Multicast Topology Inference from Measured End-to-End Loss

N.G. Duffield, J. Horowitz, F. Lo Presti, D. Towsley

Abstract—**The use of multicast inference on end-to-end measurement has recently been proposed as a means to infer network internal characteristics such as packet link loss rate and delay. In this paper we propose three types of algorithm that use loss measurements to infer the underlying multicast topology: (i) a grouping estimator that exploits the monotonicity of loss rates with increasing path length; (ii) a maximum likelihood estimator; and (iii) a Bayesian estimator. We establish their consistency, compare their complexity and accuracy, and analyze the modes of failure and their asymptotic probabilities.**

Keywords: Communication Networks, End-to-End Measurement, Maximum Likelihood Estimation, Multicast, Statistical Inference, Topology Discovery.

I. Introduction

A. Motivation.

In this paper we propose and evaluate a number of algorithms for the inference of logical multicast topologies from end-to-end network measurements. All are developed from recent work that shows how to infer per link loss rate from measured end-to-end loss of multicast traffic. The idea behind this approach is that performance characteristics across a number of intersecting network paths can be combined to reveal characteristics of the intersection of the paths. In this way, one can infer characteristics across a portion of the path without requiring the portion's endpoints to terminate measurements.

The use of active multicast probes to perform measurements is particularly well suited to this approach due to the inherent correlations in packet loss seen at different receivers. Consider a multicast routing tree connecting the probe source to a number of receivers. When a probe packet is dispatched down the tree from the source, a copy is sent down each descendant link from every branch point encountered. By this action, one packet at the source gives rise to a copy of the packet at each receiver. Thus a packet reaching each member of a subset of receivers encounters *identical* conditions between the source and the receivers' closest common branch point in the tree.

N.G. Duffield and F. Lo Presti are with AT&T Labs–Research, 180 Park Avenue, Florham Park, NJ 07932, USA, E-mail: {duffield, lopresti}@research.att.com. J. Horowitz is with the Dept. of Math. & Statistics, University of Massachusetts, Amherst, MA 01003, USA, E-mail: joeh@math.umass.edu. D. Towsley is with the Dept. of Computer Science, University of Massachusetts, Amherst, MA 01003, USA, E-mail: towsley@cs.umass.edu

This approach has been used to infer the per link packet loss probabilities for logical multicast trees with a known topology. The Maximum Likelihood Estimator (MLE) for the link probabilities was determined in [3] under the assumption that probe loss occurs independently across links and between probes. This estimate is somewhat robust with respect to violations of this assumption. This approach will be discussed in more detail presently.

The focus of the current paper is the extension of these methods to infer the *logical topology* when it is not known in advance. This is motivated in part by ongoing work [1] to incorporate the loss-based MLE into the National Internet Measurement Infrastructure [14]. In this case, inference is performed on end-to-end measurements arising from the exchange of multicast probes between a number of measurement hosts stationed in the Internet. The methods here can be used to infer first the logical multicast topology, and then the loss rates on the links in this topology. What we do not provide (are unable to) is an algorithm for identifying the physical topology of a network.

A more important motivation for this work is that knowledge of the multicast topology can be used by multicast applications. It has been shown in [9] that organizing a set of receivers in a bulk transfer application into a tree can substantially improve performance. Such an organization is central component of the widely used RMTP-II protocol [20]. The development of tree construction algorithms for the purpose of supporting reliable multicast has been identified to be of fundamental importance by the Reliable Multicast Transport Group of the IETF; see [7]. This motivated the work reported in [16], which was concerned with grouping multicast receivers that share the same set of network bottlenecks from the source for the purposes of congestion control. Closely related to [3], the approach of [16] is based on estimating packet loss rates for the path between the source and the common ancestor of pairs of nodes in the special case of binary trees. Since loss is a non-decreasing function of the path length, this quantity should be maximal for a sibling pair. The whole binary tree is reconstructed by iterating this procedure.

B. Contribution.

This paper describes and evaluates three methods for inference of logical multicast topology from end-to-end multicast measurements. Two of these ((i) and (ii) below) are directly based on the MLE for link loss probabilities of [3], as recounted in Section II. In more detail, the three methods are:

(i) Grouping Classifiers. We extend the grouping method of [16] to general trees, and establish its correctness. This is done in two steps. First, in Section III, we apply and extend the methods of [3] to establish a one-to-one correspondence between the the expected distribution of events measurable at the leaves, and the underlying topology and loss rates. In particular, we provide an algorithm that reconstructs arbitrary (e.g. non-binary) topologies from the corresponding distributions of leaf-measurable events. Second, in Section IV, we adapt the algorithm to work with the empirical leaf-event distributions arising from multicast end-to-end measurements. A complication arises through the fact that certain equalities that hold for the expected distributions only hold approximately for the measured distributions. We propose and evaluate three variants of the algorithm to overcome this. One is based on the above reconstruction method for general trees; the other two methods use binary grouping operations to reconstruct a binary tree, which is then manipulated to yield the inferred tree.

(ii) Maximum Likelihood Classifier. Given the measured end-to-end packet losses, the link loss estimator of [3] associates a likelihood with each possible logical multicast tree connecting the source to the receivers. The maximum likelihood classifier selects that tree for which the likelihood is maximal. This estimator is presented in Section V.

(iii) Bayesian Classifier. In this approach, the topology and link probabilities are treated as random variables with some prior distribution. In Bayesian decision theory one specifies a loss function that characterizes a penalty for misclassification, then selects the topology that minimizes the mean value of this penalty according to the posterior distribution (i.e. the conditional distribution of the parameters given the measurements). This estimator is presented in Section VI.

In all cases we establish that the classifiers are consistent, i.e., the probability of correct classification converges to 1 as the number of probes grows to infinity. We establish connections amongst the grouping-based algorithms. In particular, the general grouping-based algorithm is equivalent to the composition of the binary grouping algorithm with a pruning operation that excises links of zero loss and identifies their endpoints. The latter approach turns out to be computationally simpler.

The ML and Bayesian classifiers, embodying standard statistical methods, provide reference points for the accuracy of the grouping-based classifiers. In Section VII we use simulations to evaluate the relative accuracy of the topology classifiers, and to understand their modes of failure. We find that the accuracy of the grouping classifiers either closely matches or exceeds that of the other methods when applied to the identification of a selection of fixed unknown topologies. This finding is supported by some numerical results on the tail asymptotics of misclassification probabilities when using large numbers of probes. The simulations show the techniques can resolve topologies even when link loss probabilities are as small as about 1%, on the basis of data from a few thousand probes. This data could be gathered from a probe source of low bandwidth (a few tens of kbits per second) over a few minutes.

The ML and Bayesian classifiers are considerably more computationally complex than the grouping-based methods. This is for two reasons: (i) they exhaustively search the set of possible trees, whereas the grouping approaches progressively exclude certain topologies from consideration as groups are formed; (ii) their per-topology computational costs are greater. Since the number of possible topologies grows rapidly with the number of receivers, any decrease in per-topology cost for the ML and Bayesian classifiers would eventually be swamped by the growth in the number of possible topologies. For this reason, we expect significant decrease in complexity will only be available for classifiers that are able to search the topology space in a relatively sophisticated manner, e.g. as performed by the grouping-based algorithms. Summarizing, we conclude that binary-based grouping algorithms provide the best combination of accuracy and computational simplicity.

In Section VIII we further analyze the modes of misclassification in grouping algorithms. We distinguish the coarser notion of misgrouping, which entails failure to identify the descendant leaves of a given node. This notion is relevant, for example, in multicast congestion control, where one is interested in establishing the set of receivers that are behind each bottleneck. We obtain rates of convergence of the probability of successful grouping and classification in the regime of small link loss rates.

We conclude in Section IX; the proofs and some more detailed technical material are deferred to Section X.

C. Other Related Work.

The `mtrace` [12] measurement tool, reports the route from a multicast source to a receiver, along with other information about that path such as per-hop loss statistics. The `tracer` tool [10] uses `mtrace` to perform topology discovery. We briefly contrast some properties of those methods with those presented here. (i) Access: `mtrace` relies on routers to respond to explicit measurement queries; access to such facilities may be restricted

by service providers. The present method does not require such cooperation. (ii) Scaling: `mtrace` needs to run once per receiver in order to cover the tree, so that each router must process requests from all its descendant leaf nodes. The present method works with a single pass down the tree. On the other hand, our methods do not associate physical network addresses with nodes of the logical multicast tree. For this reason, we envisage combining `mtrace` and multicast-based estimation in measurement infrastructures, complementing infrequent `mtrace` measurements with ongoing multicast based-inference to detect topology changes.

In the broader context of network tomography we mention some recent analytic work on a different problem, namely, determination of source-destination traffic matrix from source- and destination-averaged traffic volumes; see [18], [19] for further details.

II. Loss Trees and Inference of Loss Rate

We begin by reviewing the tree and loss models used to formulate the MLE for link loss probabilities in a known topology. We identify the physical multicast tree as comprising actual network elements (the nodes) and the communication links that 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 has at least two children, except the leaf nodes (which have none) and the root (which we assume has one). We can construct the logical tree from the physical tree by the following procedure: except for the root, delete each node that has only one child, and adjust the link set accordingly by linking its parent directly to its child.

A. Tree Model.

Let $\mathcal{T} = (V, L)$ denote a logical multicast tree with nodes V and links L. We identify one node, the root 0, with the source of probes, and set of leaves $R \subset V$ with the set of receivers. We say that a link is internal if neither of its endpoints is the root or a leaf node. We will occasionally use W to denote $V \setminus (\{0,1\} \cup R)$, where 1 denotes the child node of 0, the set of nodes terminating internal links. Each node k, apart from the root, has a parent $f(k)$ such that $(f(k), k) \in L$. We will sometimes refer to $(f(k), k)$ as link k. Define recursively the compositions $f^n = f \circ f^{n-1}$ with $f^1 = f$. We say j is descended from k, and write $j \prec k$, if $k = f^n(j)$ for some positive integer n. The set of children of k, namely $\{j \in V : f(j) = k\}$ is denoted by $d(k)$. The (nearest) ancestor $a(U)$ of a subset $U \subset V$ is the $\prec$-least upper bound of all the elements of U. A collection of nodes U are said to be siblings if they have the same parent, i.e., if $f(k) = a(U)\ \forall k \in U$. A maximal sibling set comprises the entire set $d(k)$ of children of some node $k \in V$. $\mathcal{T}(k) = (V(k), L(k))$ will denote the subtree rooted at k; $R(k) = R \cap V(k)$ is the set of receivers in $\mathcal{T}(k)$.

B. Loss Model.

For each link we assume an independent Bernoulli loss model: each probe is successfully transmitted across link k with probability α_k. Thus the progress of each probe down the tree is described by an independent copy of a stochastic process $X = (X_k)_{k \in V}$ as follows. $X_0 = 1$. $X_k = 1$ if the probe reaches node $k \in V$ and 0 otherwise. If $X_k = 0$, then $X_j = 0, \forall j \in d(k)$. Otherwise, $P[X_j = 1 | X_k = 1] = \alpha_j$ and $P[X_j = 0 | X_k = 1] = 1 - \alpha_j$. We adopt the convention $\alpha_0 = 1$ and denote $\alpha = (\alpha_i)_{i \in V}$. We call the pair $(\mathcal{T}, \alpha)$ a **loss tree**. $\mathsf{P}_{\mathcal{T},\alpha}$ will denote the distribution of X on the loss tree $(\mathcal{T}, \alpha)$. In what follows we shall work exclusively with **canonical loss trees**. A loss tree is said to be in canonical form if $0 < \alpha_k < 1, \forall k \in V$ except for $k = 0$. Any tree $(\mathcal{T}, \alpha)$ not in canonical form can be reduced to a loss tree, $(\mathcal{T}', \alpha')$, in canonical form such that the distribution of $(X_k)_{k \in R}$ is the same under the corresponding probabilities $\mathsf{P}_{\mathcal{T},\alpha}$ and $\mathsf{P}_{\mathcal{T}',\alpha'}$. To achieve this, links k with $\alpha_k = 1$ are excised and their endpoints identified. If any link k has $\alpha_k = 0$, then $X_j = 0$ for all $j \prec k$, and hence no probes are received at any receiver in $R(k)$. By removal of subtrees $\mathcal{T}(k)$ rooted at such k, we obtain a tree in which all probabilities $\alpha_k > 0$. Henceforth we shall consider only canonical loss trees.

C. Inference of Loss Rates.

When a probe is sent down the tree from the root 0, we can not observe the whole process X, but only the outcome $(X_k)_{k \in R} \in \Omega = \{0,1\}^R$ that indicates whether or not the probe reached each receiver. In [3] it was shown how the link probabilities can be determined from the the distribution of outcomes when the topology is known. Set

$$\gamma(k) = \mathsf{P}_{\mathcal{T},\alpha}[\vee_{j \in R(k)} X_j = 1]. \tag{1}$$

The internal link probabilities α can be found from $\gamma = \{\gamma(k) : k \in V\}$ as follows. For $k \in V$ let $A(k)$ be the probability that the probe reaches k. Thus $A(k) = \prod_{j \succeq k} \alpha_j$, the product of the probabilities of successful transmission on each link between k and the root 0. For $U \subset V$ we write $\gamma(U) = \mathsf{P}[\vee_{k \in U} \vee_{j \in R(k)} X_j = 1]$. A short probabilistic argument shows that for any $U \subseteq d(k)$,

$$(1 - \gamma(U)/A(k)) = \prod_{j \in U} (1 - \gamma(j)/A(k)). \tag{2}$$

In particular, this holds for $U = d(k)$ in which case $\gamma(U) = \gamma(k)$. It can be shown for canonical loss trees that $A(k)$ is the unique solution of (2); see Lemma 1 in [3] or Prop 1 below. Thus given $\{\gamma(k) : k \in V\}$ one

can find $(A(k))_{k\in V}$, and hence α, by taking appropriate quotients.

Let $x = (x^{(1)}, \ldots, x^{(n)})$ with $x^{(m)} = (X_k^{(m)})_{k\in R}$ be the set of outcomes arising from the dispatch of n probes from the source. We denote the log-likelihood function of this event by

$$\mathcal{L}(\mathcal{T}, \alpha) = \log \mathsf{P}_{\mathcal{T},\alpha}[x] \tag{3}$$

Construct the empirical distributions $\widehat{\gamma}(k) = n^{-1}\sum_{m=1}^{n} \vee_{j\in R(k)} X_j^{(m)}$, i.e. the fraction of the n probes that reaches some receiver descended from k. Let $\widehat{A}$ denote the corresponding solution of (2) obtained by using $\widehat{\gamma}$ in place of γ, and $\widehat{\alpha}$ the corresponding probabilities obtained by taking quotients of the $\widehat{A}$. The following results, the proof of which can be found in [3], holds.

Theorem 1: Let $\mathcal{T}$ be a canonical loss tree.
(i) The loss model is identifiable, i.e. $P_{\mathcal{T},\alpha} = P_{\mathcal{T},\alpha'}$ implies $\alpha = \alpha'$.
(ii) with probability 1, for sufficiently large n, $\widehat{A}, \widehat{\alpha}$ are the Maximum Likelihood Estimators of A, α, i.e.,

$$\widehat{\alpha} = \arg\max_{\alpha} \mathcal{L}(\mathcal{T}, \alpha). \tag{4}$$

As a consequence of the MLE property, $\widehat{A}$ is consistent ($\widehat{A} \stackrel{n\to\infty}{\longrightarrow} A$ with probability 1), and asymptotically normal ($\sqrt{n}(\widehat{A} - A)$ converges in distribution to a multivariate Gaussian random variable as $n \to \infty$), and similarly for α; see [17].

III. Deterministic Reconstruction of Loss Trees by Grouping

The use of estimates of shared loss rates at multicast receivers has been proposed recently in order to group multicast receivers that share the same set of bottlenecks on the path from the source [16]. The approach was formulated for binary trees, with shared loss rates having the direct interpretation of the loss rate on the path from the root to the (nearest) ancestor of two receivers. Since the loss rate cannot decrease as the path is extended, the pair of receivers for which shared loss rate is greatest will be siblings; if not then one of the receivers would have a sibling and the shared loss rate on the path to their ancestor would be greater. This maximizing pair is identified as a pair of siblings and replaced by a composite node that represents their parent. Iterating this procedure should then reconstruct the binary tree.

In this section and the following section we establish theoretically the correctness of this approach, and extend it to cover general trees, i.e., those with nodes whose branching ratio may be greater than two. In this section we describe how canonical loss trees are in one-to-one correspondence with the probability distributions of the random variables $(X_k)_{k\in R}$ visible at the receivers.

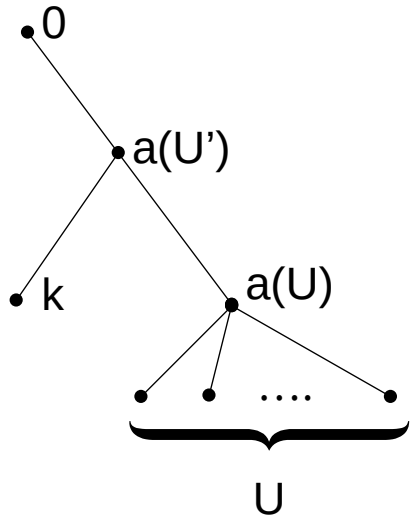


Fig. 1. $B(U') > B(U)$ where $U' = U \cup \{k\}$. Adjoining the non-sibling node k to U increases the value of B; see Prop. 1(iv).

Thus the loss tree can be recovered from the receiver probabilities. This is achieved by employing an analog of the shared loss for binary trees. This is a function $B(U)$ of the loss distribution at a set of nodes U that is minimized when U is a set of siblings, in which case $B(U) = A(a(U)))$, i.e. the complement of the shared loss rate to the nodes U. In the case of binary trees, we can identify the minimizing set U as siblings and substitute a composite node that represents their parent. Iterating this procedure should then reconstruct the tree. The definition and relevant properties of the function B are given in the following proposition.

Proposition 1: Let $\mathcal{T} = (V, L)$ be a canonical loss tree, and let $U \subset V$ with $\#U > 1$.
(i) The equation $(1 - \gamma(U)/B) = \prod_{k\in U}(1 - \gamma(k)/B)$ has a unique solution $B(U) > \gamma(U)$.
(ii) Let $B > \gamma(U)$. Then $(1 - \gamma(U)/B) > \prod_{k\in U}(1 - \gamma(k)/B)$ iff $B > B(U)$.
(iii) $B(U) = A(a(U))$ if U is a set of siblings, and hence $B(U)$ takes the same value for any sibling set with a given parent.
(iv) Let U be a set of siblings, and suppose $k \in V$ is such that $a(U \cup \{k\}) \succ a(U)$ and $a(U \cup \{k\}) \succ k$. Then $B(U \cup \{k\}) > B(U)$.

Proposition 1(iv) shows that adjoining a non-sibling non-ancestor node to a set of siblings can only increase the value of B; see Figure 1. This provides the means to reconstruct the tree $\mathcal{T}$ directly from the $\{\gamma(U) : U \subset R\}$. We call the procedure to do this the Deterministic Loss Tree Classification Algorithm (DLT), specified in Fig-

1. *Input:* The set of receivers R and associated probabilities $\{\gamma(U) : U \subset R\}$;
2. $R' := R$; $V' := R$; $L' := \emptyset$;
3. **foreach** $j \in R'$ **do** $B(j) := \gamma(j)$; **enddo**
4. **while** $|R'| > 1$ **do**
5. **select** $U = \{u_1, u_2\} \subseteq R'$ that minimizes $B(U)$;
6. **while** there exists $u \in R' \setminus U$ such that $B(U \cup \{u\}) = B(U)$ **do**
7. $U := U \cup \{u\}$;
8. **enddo**
9. $V' := V' \cup \{U\}$; $R' := (R' \setminus U) \cup \{U\}$;
10. **foreach** $j \in U$ **do**
11. $L' := L' \cup \{(U, j)\}$; $\alpha'(j) := B(j)/B(U)$;
12. **enddo**
13. **enddo**
14. $V' = V' \cup \{0\}$;
15. $L' = L' \cup \{0, U\}$;
16. $\alpha'_U = B(U)$; $\alpha'_0 = 1$;
17. *Output:* loss tree $((V', L'), \alpha')$.

Fig. 2. Deterministic Loss Tree Classification Algorithm (DLT).

1. *Input*: a loss tree $(\mathcal{T}, \alpha)$;
2. *Parameter*: a threshold $\varepsilon \geq 0$;
3. $V' := \{0\} \cup d_{\mathcal{T}}(0)$; $L' := \{(0, k) : k \in d_{\mathcal{T}}(0)\}$;
4. $U := d_{\mathcal{T}}(0)$;
5. **while** $U \neq \emptyset$ **do**
6. **select** $j \in U$;
7. $U := U \setminus \{j\} \cup d_{\mathcal{T}}(j)$;
8. **if** $((1 - \alpha_j) \leq \varepsilon) \wedge (j \neq R)$ **then**
9. $L' := (L' \cup \{(f_{\mathcal{T}'}(j), k) : k \in d_{\mathcal{T}}(j)\}) \setminus \{(f_{\mathcal{T}'}(j), j)\}$;
10. $V' := V' \setminus \{j\} \cup d_{\mathcal{T}}(j)$;
11. **else**
12. $L' := L' \cup \{(j, k) : k \in d_{\mathcal{T}}(j)\}$;
13. $V' := V' \cup d_{\mathcal{T}}(j)$;
14. **endif**;
15. **enddo**
16. *Output:* $((V', L'), \alpha')$

Fig. 3. Tree Pruning Algorithm TP(ε)

ure 2; it works as follows. At the start of each while loop from line 4, the set R' comprises those nodes available for grouping. We first find the pair $U = \{u_1, u_2\}$ that minimizes $B(U)$ (line 5), then progressively adjoin to it further elements that do not increase the value of B (lines 6 and 7). The members of the largest set obtained this way are identified as siblings; they are removed from the pool of nodes and replaced by their parent, designated by their union U (line 9). Links connecting U to its children (i.e. members) are added to the tree, and the link loss probabilities are determined by taking appropriate quotients of $B's$ (line 11). This process is repeated until all sibling sets have been identified. Finally, we adjoin the root node 0 and the link joining it to its single child (line 14).

Theorem 2: (i) DLT reconstructs any canonical loss tree $(\mathcal{T}, \alpha)$ from its receiver set R and the associated probabilities $\{\gamma(U) : U \subset R\}$.

(ii) Canonical loss trees are identifiable, i.e. $P_{\mathcal{T},\alpha} = P_{\mathcal{T}',\alpha'}$ implies that $(\mathcal{T}, \alpha) = (\mathcal{T}', \alpha')$.

Although we have not shown it here, it is possible to establish that any set R' present at line 4 of DLT has the property that $\min_{U \subset R'} B(U)$ is achieved when U is a sibling set. Consequently one could replace steps 5–8 of DLT by simply finding the maximal sibling set, i.e. select a maximal $U \subset R'$ that minimizes $B(U)$. However, this approach would have worse computational properties since it requires inspecting every subset of R'.

$B(U)$ is a root of the polynomial of degree $\#U - 1$ from Prop. 1(i). For a binary subset, $B(\{j, k\})$ is written down explicitly

$$B(\{j, k\}) = \frac{\gamma(j)\gamma(k)}{\gamma(k) + \gamma(j) - \gamma(\{j, k\})}; \qquad (5)$$

Calculation of $B(U)$ requires numerical root finding when $\#U > 5$. However, it is possible to recover $\mathcal{T}$ in a two stage procedure that requires the calculation of $B(U)$ only on binary sets U. The first stage uses the Deterministic Binary Loss Tree (DBLT) Classification Algorithm. DBLT is identical to DLT except that grouping is performed only over binary trees, thus omitting lines 6–8 in Figure 2. The second stage is to use a Tree Pruning (TP) Algorithm on the output of the DBLT. TP acts on a loss tree $((V, L), \alpha)$ by removing from L each internal link $(f(k), k)$ with loss rate $1 - \alpha_k = 0$ and identifying its endpoints $k, f(k)$. We will find it useful to specify a slightly more general version: for $\varepsilon \geq 0$, TP(ε) prunes link k when $1 - \alpha_k \leq \varepsilon$. We formally specify TP(ε) in Figure 3. In Section X we prove that composing the binary algorithm DBLT with pruning recovers the same topology as DLT for general canonical loss trees:

Theorem 3: DLT=TP(0)$\circ$DBLT for canonical loss trees.

IV. Inference of Loss Tree from Measured Leaf Probabilities

In this section, we present algorithms which adapt DLT to use the measured probabilities corresponding to the γ. Let $(X_k^{(i)})_{k \in R}^{i=1,\ldots,n}$ denote the measured outcomes arising from each of n probes. Define the processes $Y_k^{(i)}$ recur-

1. *Input*: The set of receivers R, number of probes n, receiver traces $(X_k^{(i)})_{k\in R}^{i=1,2,\ldots n}$;
2. $R' := R, V' := R;\ L' = \emptyset$;
3. **foreach** $k \in R$, **do**
4. $\quad \widehat{B}(k) := n^{-1}\sum_{i=1}^n X_k^{(i)}$;
5. $\quad$ **foreach** $i = 1,\ldots,n$, **do** $Y_k^{(i)} = X_k^{(i)}$; **enddo** ;
6. **enddo**
7. **while** $|R'| > 1$ **do**
8. $\quad$ **select** $U = \{u_1, u_2\} \subset R'$ that minimizes $\widehat{B}(U) = \frac{\sum_{i=1}^n Y_{u_1}^{(i)} \sum_{i=1}^n Y_{u_2}^{(i)}}{n\sum_{i=1}^n Y_{u_1}^{(i)} Y_{u_2}^{(i)}}$;
9. $\quad$ **foreach** $i = 1,\ldots,n$ **do** $Y_U^{(i)} = \vee_{u\in U} Y_u^{(i)}$ **enddo**
10. $\quad V := V \cup \{U\}$; $R' := (R' \setminus U) \cup \{U\}$;
11. $\quad$ **foreach** $u \in U$ **do**
12. $\quad\quad L' := L' \cup \{(U,u)\}$; $\widehat{\alpha}_u := \widehat{B}(u)/\widehat{B}(U)$;
13. $\quad$ **enddo**
14. **enddo**
15. $V' = V' \cup \{0\}$; $L' = L' \cup \{0, U\}$;
16. $\widehat{\alpha}_U = \widehat{B}(U)$; $\widehat{\alpha}_0 = 1$;
17. *Output:* loss tree $((V', L'), \widehat{\alpha})$.

Fig. 4. Binary Loss Tree Classification Algorithm (BLT).

sively by

$$Y_k^{(i)} = \vee_{j\in d(k)} Y_j^{(i)} \quad \text{with} \quad Y_k^{(i)} = X_k^{(i)},\ k \in R. \quad (6)$$

Thus $Y_k^i = 1$ iff probe i was received at some receiver descended from k; $\widehat{\gamma}(k) = n^{-1}\sum_{i=1}^n Y_k^{(i)}$ is the fraction of the probes $1,\ldots,n$ that reach some receiver descended from k. For $U \subset V$ we define $\widehat{\gamma}(U) = n^{-1}\sum_{i=1}^n \vee_{j\in U} Y_j^{(i)}$ analogously; $\widehat{\gamma}(U)$ is the fraction of probes that reach some receiver descended from some node in U. Let $\widehat{B}(U)$ be the unique solution in Prop. 1(ii) obtained by using $\widehat{\gamma}$ in place of γ. We will use the notation $(\widehat{\mathcal{T}}, \widehat{\alpha})$ to denote an inferred loss tree; sometimes we will use $\widehat{\mathcal{T}}_{\mathsf{X}}$ to distinguish the topology inferred by a particular algorithm X. P_{X}^f will denote the probability of false identification of topology $\mathcal{T}$ of the loss tree $(\mathcal{T}, \alpha)$ i.e. $P_{\mathsf{X}}^f = \mathsf{P}_{\mathcal{T},\alpha}[\widehat{\mathcal{T}}_{\mathsf{X}} \neq \mathcal{T}]$.

Theorem 4: Let $((V, L), \alpha)$ be a loss tree. Then $\lim_{n\to\infty} \widehat{B}(U) = B(U)$ for each $U \subset V$.

A. Classification of Binary Loss Trees

The adaptation of DLT is most straightforward for binary trees. By using $\widehat{B}$ in place of B in DLT and restricting the minimization of $\widehat{B}$ to binary sets we obtain the Binary Loss Tree (BLT) Classification Algorithm; we specify it formally in Figure 4. This is, essentially, the algorithm proposed in [16]. We have taken advantage of the recursive structure of the $Y_k^{(i)}$ (in line 9) in order to calculate the probabilities $\widehat{\gamma}$. Note that when BLT reconstructs an incorrect topology $\widehat{\mathcal{T}} \neq \mathcal{T}$, the definitions of quantities such as $\widehat{B}(U)$ and $Y_U^{(i)}$ extend evidently to subsets U of nodes in the incorrect topology $\mathcal{T}'$. The following theorem establishes the consistency of the estimator $\widehat{\mathcal{T}}_{\mathsf{BLT}}$; the proof appears in Section X.

Theorem 5: Let $(\mathcal{T}, \alpha)$ be a binary canonical loss tree. With probability 1, $\widehat{\mathcal{T}}_{\mathsf{BLT}} = \mathcal{T}$ for all sufficiently large n, and hence $\lim_{n\to\infty} P_{\mathsf{BLT}}^f = 0$.

B. Classification of General Loss Trees

The adaptation of DLT to the classification of general loss trees from measured leaf probabilities is somewhat more complicated than the binary case. It is shown during the proof of Theorem 5 that the $\widehat{B}(U)$ have the same relative ordering as the $B(U)$ for n sufficiently large. But for a general tree (V, L), $B(U')$ takes the same value for any subset U' of a maximal sibling set $U \subset V$. For finitely many probes, the corresponding $\{\widehat{B}(U') : U' \subset U\}$ will not in general be equal. Hence choosing to group the subset U' that minimizes $\widehat{B}(\cdot)$ will not necessarily group all the siblings in U.

In this section we present three algorithms to classify general trees. Each of these overcomes the problem described in the previous paragraph by incorporating a threshold into the grouping procedure. The set U is grouped if $\widehat{B}(U)$ is sufficiently close to being minimal. However, this can also give rise to false inclusion by effectively ignoring internal links whose loss rates do not exceed the threshold. The variety of algorithms derives from different ways to implement the threshold. We establish domains in which the algorithms correctly classify canonical loss trees. In succeeding sections we evaluate their relative efficiencies and compare their modes and frequencies of false classification.

B.1 Binary Loss Tree Pruning Classification Algorithm BLTP.

Nodes are grouped as if the tree were binary, the resulting tree is pruned with $\mathsf{TP}(\varepsilon)$ to remove all internal links with loss probabilities less than or equal to the threshold $\varepsilon > 0$. Thus for each $\varepsilon > 0$ we define $\mathsf{BLTP}(\varepsilon)$ to be the composition $\mathsf{TP}(\varepsilon)\circ\mathsf{BLT}$. A refinement $\mathsf{BLTP}'(\varepsilon)$ of $\mathsf{BLTP}(\varepsilon)$ is to recalculate the loss probabilities α' based on the measurements and the pruned topology $\mathcal{T}'$.

B.2 Binary Loss Tree Clique Classification Algorithm BLTC.

For each $\varepsilon > 0$, $\mathsf{BLTC}(\varepsilon)$ groups by forming maximal sets of nodes U in which all binary subsets U' have $\widehat{B}(U')$ sufficiently close to the true minimum over all binary sets. This amounts to replacing line 8 in Figure 4 with the following steps:

(i) select $U' = \{u', v'\}$ that minimizes $\widehat{B}(U')$;

(ii) construct the graph G of all links (u'', v'') such that $(1-\varepsilon)\widehat{B}(\{u'', v''\}) < \widehat{B}(U')$;
(iii) select U comprising the elements of the largest connected component of G that contains U'.
Note that if the grouping is done correctly, then $B(\{u', v'\})$ takes the same value for all binary subsets $\{u', v'\}$ of U. For finite but large n, the corresponding sampled $\widehat{B}(\{u', v'\})$ will differ slightly.

B.3 General Loss Tree Classification Algorithm GLT.

For each $\varepsilon > 0$, GLT(ε) is a modification of DLT that employs a threshold ε to perform the grouping based on $\widehat{B}$. Each grouping step starts by finding a binary set $\{u_1, u_2\}$ of minimal $\widehat{B}$, then adjoining further elements to it provided the resulting set U satisfies $\widehat{B}(U)(1-\varepsilon) < \widehat{B}(\{u_1, u_2\})$. The violation of this condition has the interpretation that the ancestor $a(U)$ is separated from $a(\{u_1, u_2\})$ by a link with loss rate at least ε. Thus we replace line 8 of Figure 4 by the following.

8a. **select** $U := \{u_1, u_2\} \subset S$ that minimizes $\widehat{B}(\cdot)$;
8b. **while** there exists $u \in R' \setminus U$ such that
$(1-\varepsilon)\widehat{B}(U \cup \{u\}) < \widehat{B}(u_1, u_2)\}$ **do**
8c. $U = U \cup \{u\}$;
8d. **enddo**

For clarity we have omitted the details of the dependence of $\widehat{B}$ on the $\widehat{\gamma}$; these are as described before Theorem 4.

B.4 Convergence of General Loss Tree Classifiers.

As the number of probes grows, the topology estimates furnished by BLTP(ε),BLTC(ε) and GLT(ε) converge to the true topology provided all internal link loss probabilities are greater than ε. This happens for the same reason as it does in BLT. It is not difficult to see that the deterministic versions of each algorithm, obtained by using B in place of $\widehat{B}$, reconstruct the topology. Since $\widehat{B}$ converges to B as the number of probes grows, the same is true for the classifiers using $\widehat{B}$. We collect these results without further proof:

Theorem 6: Let $(\mathcal{T}, \alpha)$ be a loss tree in which all loss probabilities $1-\alpha_k > \varepsilon'$, $k \in W$, for some $\varepsilon' > 0$. For each $\varepsilon \in (0, \varepsilon')$ and each algorithm $\mathsf{X} \in \{\mathsf{BLTP}(\varepsilon), \mathsf{BLTC}(\varepsilon), \mathsf{GLT}(\varepsilon)\}$, with probability 1, $\widehat{\mathcal{T}}_{\mathsf{X}} = \mathcal{T}$ for all sufficiently large n, and hence $\lim_{n\to\infty} P_{\mathsf{X}}^f = 0$.

Convergence to the true topology requires ε to be smaller than the internal link loss rates, which are typically not known in advance. A very small value of ε is more likely to satisfy the above condition but at the cost, as shown in Section VIII, of slower classifier convergence. A large value of ε, on the other hand, is more likely to result in systematically removing links with small loss rates. In practice, however, we believe that the choice of ε does not pose a problem. We expect, indeed, that for many applications while it is important to correctly identify links with high loss rate, it could be considered acceptable failure to detect those with small loss rates. In other words, in practice, it could be sufficient the convergence of the inferred topology to $\mathcal{T}^\varepsilon = \mathsf{TP}(\varepsilon)(\mathcal{T})$ obtained from $\mathcal{T}$ by ignoring links whose loss rates fell below some specific value ε which, in this case, would be regarded as some application-specific minimum loss rate of interest.

The results below establish the desired convergence to $\mathcal{T}^\varepsilon$ for any $\varepsilon \in (0,1)$ provided $\varepsilon \neq \alpha_k$, $k \in W$. The key observation is that since the deterministic versions of each algorithm reconstruct $\mathcal{T}^\varepsilon$, so does each algorithm, as the number of probes grows. Denote $P_X^f(\varepsilon) = \mathsf{P}_{\mathcal{T},\alpha}[\widehat{\mathcal{T}}_X \neq \mathcal{T}^\varepsilon]$. Without further proof we have:

Theorem 7: Let $(\mathcal{T}, \alpha)$ be a loss tree. For each $\varepsilon \in (0,1)$, such that $\varepsilon \neq \alpha_k$, $k \in W$, and for each algorithm $\mathsf{X} \in \{\mathsf{BLTP}(\varepsilon), \mathsf{BLTC}(\varepsilon), \mathsf{GLT}(\varepsilon)\}$, then with probability 1, $\widehat{\mathcal{T}}_{\mathsf{X}} = \mathcal{T}^\varepsilon = \mathsf{TP}(\varepsilon)(\mathcal{T})$ for all sufficiently large n, and hence $\lim_{n\to\infty} P_{\mathsf{X}}^f(\varepsilon) = 0$.

C. Effects of Model Violation

The two underlying statistical assumptions are (i) probes are independent; and (ii) conditioned on a probe having reached a given node k, the events of probe loss on distinct subtrees descended from k are independent. We now discuss the impact of violations of these assumptions.

The first observation is that the estimators remain consistent under the introduction of some temporal dependence between probes, i.e. under violation of assumption (i) above. Assuming the loss process to be ergodic, $\widehat{\gamma}$ still converges to γ almost surely, as the number of probes n grows. However, rates of convergence can be slower, and hence the variance of $\widehat{B}$ can be higher, than for the independent case. This would increase the misclassification probabilities for inference from a given number of probes n.

On the other hand, spatial dependence of loss (i.e. violations of assumption (ii) above) can lead to bias. We take spatial loss dependence to be characterized by departure from zero of an appropriate set of loss correlation coefficients. By extending an argument given for binary trees in [3, Theorem 7], it can be shown that the limit quantities $B' = \lim_{n\to\infty} \widehat{B}$ deform continuously away from the quantities B of the spatially independent case as the loss correlation coefficients move away from zero. Hence a given canonical loss tree can be recovered correctly by applying DBLT to the quantities B' provided the spatial dependence is sufficiently small, i.e., to make the B' sufficiently close to B' so that $B(U_1) > B(U_2)$ iff $B'(U_1) > B'(U_2)$ for all relevant subsets of nodes U_1 and U_2. Then by a similar argument to that of Theorem 5,

a tree with link loss rates greater than some $\varepsilon > 0$, is recovered by $\mathsf{BLTP}(\varepsilon)$ with probability 1 for a sufficiently large number n of probes, and sufficiently small spatial correlations.

We remark that the the experiments reported in Sections VII and VIII use network level simulation rather than model based simulation. Hence it is expected that the model assumptions will be violated to some extent. Nevertheless, the classifiers are found to be quite accurate.

V. Maximum-Likelihood Classifier

Let $\mathcal{T}(R)$ denote the set of logical multicast trees with receiver set R. Denote by $\widehat{\alpha}_{\mathcal{T}}$ the MLE of α in (4) for the topology $\mathcal{T}$. The *maximum-likelihood (ML) classifier* assigns the topology $\widehat{\mathcal{T}}_{\mathrm{ML}}$ that maximizes $\mathcal{L}(\mathcal{T}, \widehat{\alpha}_{\mathcal{T}})$:

$$\widehat{\mathcal{T}}_{\mathrm{ML}} = \arg\max_{\mathcal{T} \in \mathcal{T}(R)} \mathcal{L}(\mathcal{T}, \widehat{\alpha}_{\mathcal{T}}). \tag{7}$$

We prove that, if the link probabilities are bounded away from 0 and 1, the ML-classifier is *consistent* in the sense that, w.p. 1, it identifies the correct topology as the number of probes grows to infinity. For $\varepsilon > 0$, let $\mathcal{A}^{\varepsilon}_{\mathcal{T}} = \{\alpha : \varepsilon \le \alpha_k \le 1 - \varepsilon,\ k \in V \setminus \{0\}\}$.

Theorem 8: Let $\varepsilon > 0$ and let $(\mathcal{T}, \alpha)$ be a loss tree with $\alpha \in \mathcal{A}^{\varepsilon}_{\mathcal{T}}$. Then $\lim_{n\to\infty} \mathsf{P}_{\mathcal{T},\alpha}[\widehat{\mathcal{T}}_{\mathrm{ML}} \neq \mathcal{T}] = 0$.

VI. Loss-Based Bayesian Tree Classifier

Let $\mathcal{T}(R)$ denote the set of logical multicast topologies having a given receiver set R. $\mathcal{A}^0_{\mathcal{T}}$ from Section V, is the set of possible loss rates in the topology $\mathcal{T}$. A possible loss tree with topology in $\mathcal{T}(R)$ is an element of the parameter space

$$\Theta = \cup_{\tau \in \mathcal{T}(R)} (\{\tau\} \times \mathcal{A}^0_{\tau}). \tag{8}$$

Let $\pi(\tau, \alpha)$ be a prior distribution on Θ. Given receiver measurements $x = (x^{(1)}, \ldots, x^{(n)})$, the posterior distribution on Θ is

$$\pi(\tau, \alpha | x) = \pi(\tau, \alpha) f(x | \tau, \alpha) / f(x), \tag{9}$$

where $f(x|\tau,\alpha) = e^{\mathcal{L}(\tau,\alpha)}$ is the joint density of the observations and $f(x)$ their marginal density.

A *decision rule* δ provides an estimate $\delta(x) \in \Theta$ of the loss tree given receiver measurements x. The quality of a decision rule is evaluated in terms of a *loss function* $H(\theta, \theta')$, a nonnegative function on $\Theta \times \Theta$ interpreted as the loss incurred by deciding that θ' is the true parameter when, in fact, it is θ. A measure of quality of a decision rule δ is its Bayes risk $R(\delta) = E(H(\theta, \delta(x))$, where the expectation is taken with respect to the joint distribution $\pi(\tau, \alpha) f(x \mid \tau, \alpha)$. of the loss tree $\theta = (\tau, \alpha)$ and the observations x. The Bayes decision rule δ_B is the one that minimizes $R(\delta)$: it has least average loss. A standard theorem in decision theory gives δ_B in the form:

$$\delta_B(x) = \arg\min_{\theta' \in \Theta} \int_{\Theta} H(\theta, \theta') \pi(\theta | x)\, d\theta, \tag{10}$$

i.e., it is the minimizer of the *posterior risk*, which is the expected loss with respect to the posterior distribution $\pi(\theta|x)$; see Prop. 3.16 of [17] and Section 4.4, result 1 of [2].

Since our interest is in identifying the correct topology, we choose the loss function $H((\tau, \alpha), (\tau', \alpha')) = \chi[\tau \neq \tau']$ where χ is the indicator function, i.e., no loss for a correct identification of the topology, and unit loss for any misidentification. Here, the loss rates α play the role of a nuisance parameter. The Bayes classifier for the topology becomes $\widehat{\mathcal{T}}_B = \widehat{\mathcal{T}}_B(x)$, where

$$\widehat{\mathcal{T}}_B(x) = \arg\min_{\tau' \in \mathcal{T}(R)} \mathsf{P}[\tau' \neq \tau \mid x], \tag{11}$$

or, equivalently,

$$\widehat{\mathcal{T}}_B(x) = \arg\max_{\tau' \in \mathcal{T}(R)} \mathsf{P}[\tau' = \tau \mid x]. \tag{12}$$

Thus the Bayes classifier $\widehat{\mathcal{T}}_B$ yields the topology with maximum posterior probability given the data x. By definition, this classifier minimizes the misclassification probability.

A special case is the uniform prior in which all topologies in $\mathcal{T}(R)$ are taken to be equally likely, and for each topology τ, α is distributed uniformly on $\mathcal{A}^0_{\tau}$. The corresponding prior distribution, $\pi(\tau, \alpha) = \chi_{\mathcal{A}^0_{\tau}}(\alpha) / \#\mathcal{T}(R)$, is a non-informative prior, expressing "maximum ignorance" about the tree topology and link probabilities. Clearly if other prior information is available about the tree, it may be incorporated into a non-uniform prior distribution. The Bayes classifier becomes

$$\widehat{\mathcal{T}}_B(x) = \arg\max_{\tau' \in \mathcal{T}(R)} \int_{A^0_{\tau'}} f(x | \tau', \alpha)\, d\alpha. \tag{13}$$

This should be compared with the ML classifier in (7).

A. Consistency of Pseudo-Bayes Classifiers.

In practice our task is to identify the specific topology giving rise to a set of measured data. When no prior distribution is specified, the concept of the Bayes classifier, as the maximizer of the probability of correct classification, does not make sense, because "the" probability of correct classification is not defined. Nonetheless it may be convenient to construct a *pseudo-Bayes* classifier by choosing a distribution π on Θ, which plays the role of a prior, and forming the classifier in (10), which we now denote by

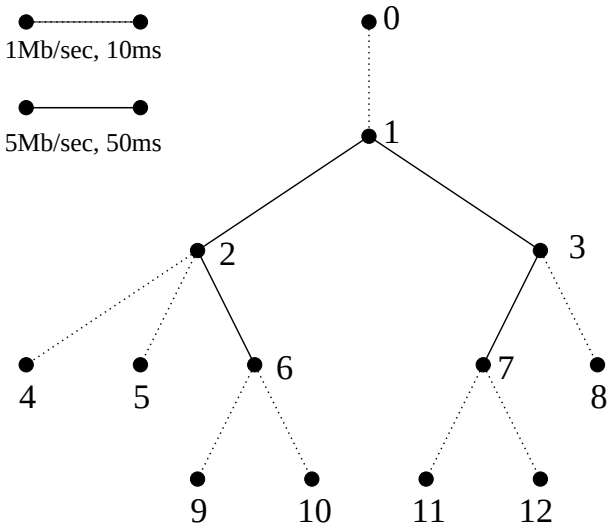


Fig. 5. Network-level simulation topology for ns. Links are of two types: *edge* links of 1Mb/s capacity and 10ms latency, and *interior* links of 5Mb/s capacity and 50ms latency.

$\widehat{\mathcal{T}}_\pi$. Classifiers constructed in this way are also consistent under a mild condition.

Theorem 9: Let π be a prior distribution on Θ, and assume that $(\mathcal{T}, \alpha)$ lies in the support of π. Then $\widehat{\mathcal{T}}_\pi$ is consistent in the frequentist sense, i.e., $\mathsf{P}_{\mathcal{T},\alpha}[\widehat{\mathcal{T}}_\pi \neq \mathcal{T}] \to 0$ as $n \to \infty$.

VII. Simulation Evaluation and Algorithm Comparison

A. Methodology.

We used two types of simulation to verify the accuracy of the classification algorithms and to compare their performance. In model-based simulation, packet loss occurs pseudorandomly in accordance with the independence assumptions of the model. This allows us to verify the prediction of the model in a controlled environment, and to rapidly investigate the performance of the classifiers in a varied set of topologies.

This approach was complemented by network-level simulations using the ns [13] program; these allow protocol-level simulation of probe traffic mixed in with background traffic of TCP and UDP sessions. Losses are due to buffer overflow, rather than being generated by a model, and hence can violate the Bernoulli assumptions underlying the analysis. This enables us to test the robustness to realistic violations of the model. For the ns simulations we used the topology shown in Figure 5. Links in the interior of the tree have higher capacity (5Mb/sec) and latency (50ms) than those at the edge (1Mb/sec and 10ms) in order to capture the heterogeneity between edges and core of a wide area network. Probes were generated from node 0 as a Poisson process with mean interarrival time 16ms. Background traffic comprised a mix of infinite FTP data sources connecting with TCP, and exponential on-off sources using UDP. The amount of background traffic was tuned in order to give link loss rates that could have significant performance impact on applications, down to as low as about 1%. One strength of our methodology is its ability to discern links with such small but potentially significant loss rates. In view of this, we will find it convenient to quote all loss rates as percentages.

B. Performance of Algorithms Based on Grouping

B.1 Dependence of Accuracy on Threshold ε.

We conducted 100 ns simulations of the three algorithms BLTP,BLTC and GLT. Link loss rates ranged from 1.8% to 10.9% on interior links; these are the links that must be resolved if the tree is to be correctly classified. In Figures 6–11 we plot the fraction of experiments in which the topology was correctly identified as a function of the number of probes, for the three algorithms, and for selected values of ε between 0.25% and 5%. Accuracy is best for intermediate ε, decreasing for larger and smaller ε. The explanation for this behavior is that smaller values of ε lead to stricter criteria for grouping nodes. With finitely many samples, for small ε, sufficiently large fluctuations of the $\widehat{B}$ cause erroneous exclusion of nodes. By increasing ε, the threshold for group formation is increased and so accuracy is initially increased. However, as ε approaches the smallest interior link loss rate, large fluctuations of the $\widehat{B}$ now cause erroneous inclusion of nodes into groups. When ε is moved much beyond the smallest interior loss rate, the probability of correct classification falls to zero. The behavior is different if we ignore failures to detect links with loss rates smaller than ε. For $\varepsilon = 5\%$ and $\varepsilon = 7\%$, in Figure 12 and 13, respectively, we plot the fraction of experiments in which the pruned topology $\mathcal{T}^\varepsilon$ was correctly identified for the three algorithms. Here the accuracy depends on the relative values of ε and the internal link loss rates. In these experiments, the actual loss rates was often very close to 5%, so that small fluctuations results in erroneous inclusions/exclusions of nodes which accounts for the significant fraction of failures for $\varepsilon = 5\%$. In Section VIII-B we shall analyze this behavior and obtain estimates for the probabilities of misclassification in the regimes described. We comment on the relative accuracy of the algorithms below.

B.2 Dependence of Accuracy on Topology.

We performed 1000 model-based simulations using randomly generated 24-node trees with given maximum branching ratios 2 and 4. Link loss rates were chosen at random in the interval $[1\%, 10\%]$. Figure 14 shows

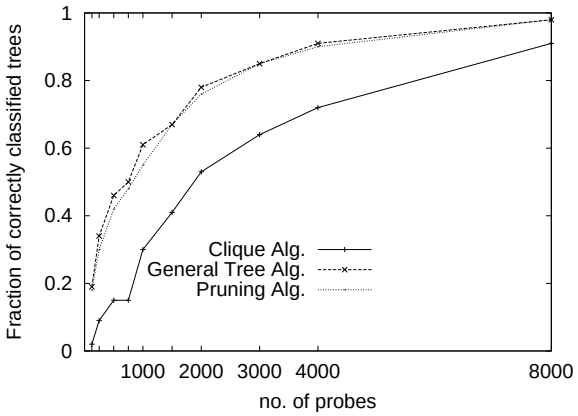


Fig. 6. $\epsilon = 0.25\%$.

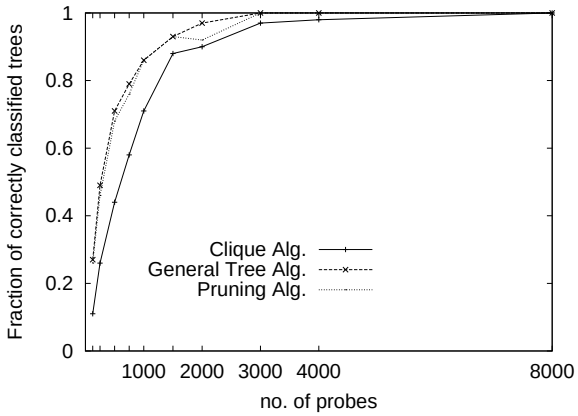


Fig. 7. $\epsilon = 0.5\%$.

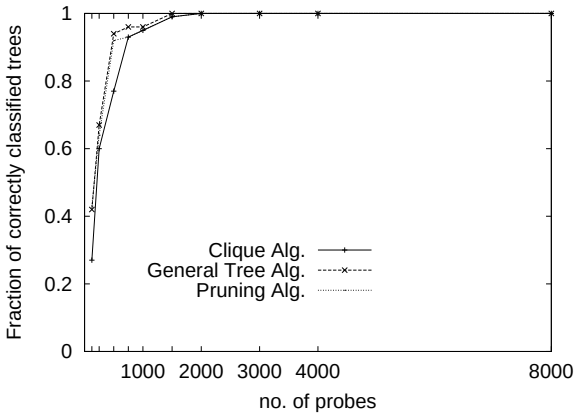


Fig. 8. $\epsilon = 1.0\%$.

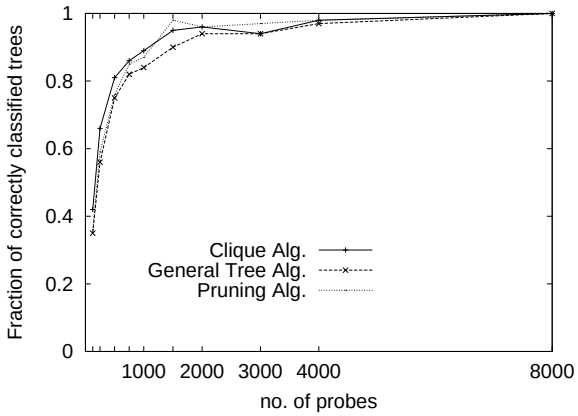


Fig. 9. $\epsilon = 2.0\%$.

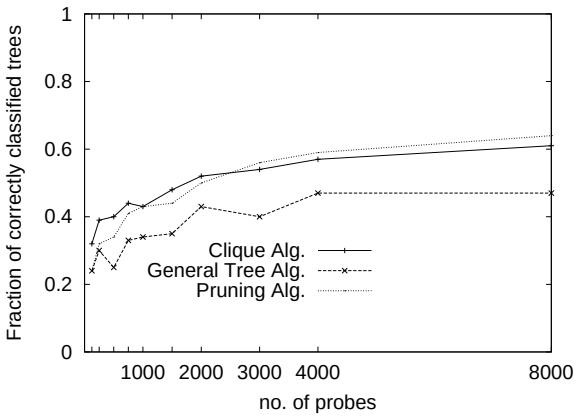


Fig. 10. $\epsilon = 3.0\%$.

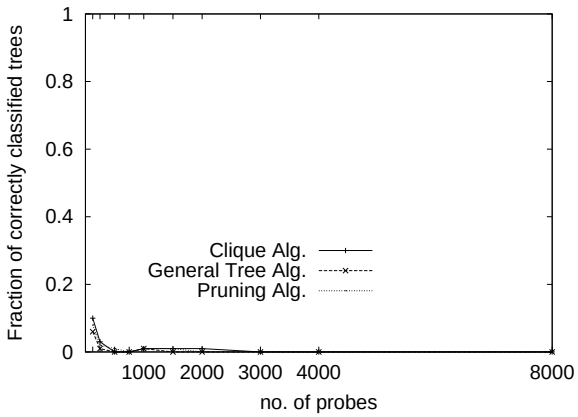


Fig. 11. $\epsilon = 5.0\%$.

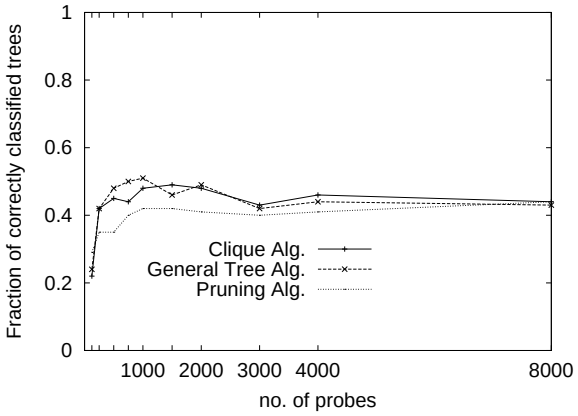


Fig. 12. $\epsilon = 5\%$.

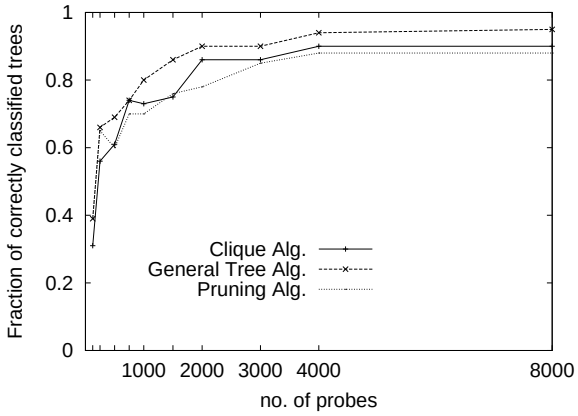


Fig. 13. $\epsilon = 7\%$.

the probability of successful classification for BLTP(ε), BLTC(ε) and GLT(ε) for $\varepsilon = 0.25\%$. In both cases this grows to 1, but convergence is slower for trees with higher branching ratios. We believe this behavior occurs due to the larger number of comparisons of values of $\widehat{B}$ that are made for trees with higher branching ratio, each such comparison affording an opportunity for misclassification.

B.3 Comparison of Grouping Algorithm Accuracy.

In all experiments reported so far, with one exception, the accuracies of BLTP and GLT were similar, and at least as good as that of BLTC. The similar behavior of BLTP and GLT is explained by observing that the two algorithms group nodes in a similar manner. In BLTP, a link is pruned from the reconstructed binary tree if its inferred loss rate is smaller than ε. In GLT, a node is added to a group if the estimated common loss of the augmented group is within ε of the estimated common loss of the original group. The operation of BLTC is somewhat different, checking all possible pairs amongst candidate nodes for grouping. Incorrect ordering in any test can result in false exclusion from a sibling set. We observe also that the performance gap between BLTC and the other algorithms is sensitive to the value of ε and to the branching ratio. The exceptional case in which BLTC performs better than the other algorithms is in the inference of binary trees: here BLTC performs slightly better because of the stricter grouping condition is employs, making it less likely to group more than two nodes.

B.4 Computational Complexity of Grouping Algorithms.

Of the two best performing grouping algorithms, namely BLTP and GLT, we observe that BLTP has smaller computational complexity for several reasons. First, $\widehat{B}$ is given explicitly for binary groups, whereas generally it requires numerical root finding. Second, although the algorithms have to calculate $\widehat{B}$ for up to $O(\#R^3)$ groups, in typical cases GLT requires additional calculations due to the larger sibling groups considered. Thirdly, observe that each increase of the size of sets considered in GLT is functionally equivalent to one pruning phase in BLTP. Thus in GLT, the threshold ε is applied throughout the algorithm; in BLTP it is applied only at the end. We expect this to facilitate adaptive selection of ε in BLTP. Comparing now with BLTC, we observe that this algorithm requires, in addition to the calculation of shared losses, the computation of a maximal connected subgraph, an operation that does not scale well for large numbers of nodes. For these reasons we adopt BLTP as our reference grouping algorithm since it is the simplest and has close to the best accuracy. In the next section, we compare its performance with that of the ML and Bayesian classifiers.

C. *Comparison of* BLTP *with the ML and Bayesian Classifiers*

C.1 Complexity.

In this section we compare our reference grouping algorithm, BLTP, with the ML and Bayesian classifiers. Here we consider the simplest implementation of these classifiers whereby we proceed by exhaustive search of the set $\mathcal{T}(R)$ of all possible topologies during evaluation of the maxima (7) and (13). By contrast, all the grouping algorithms proceed by eliminating subsets of $\mathcal{T}(R)$ from consideration; once a set of nodes is grouped, then only topologies which have those nodes as siblings are considered.

The Bayesian classifier further requires numerical integration for each candidate topology. In order to reduce its complexity we took the prior for the link rates to be uniform on the discrete set $\{1\%, \ldots, 10\%\}$, with all topolo-

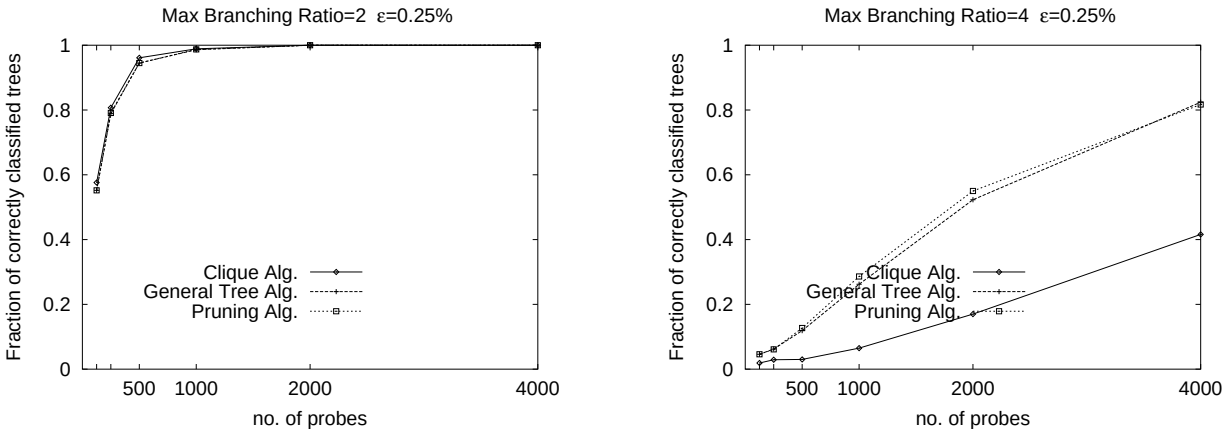


Fig. 14. DEPENDENCE OF ACCURACY ON BRANCHING RATIO: convergence is faster for binary trees (left); GLT and BLTP outperform BLTC for non-binary trees (right).

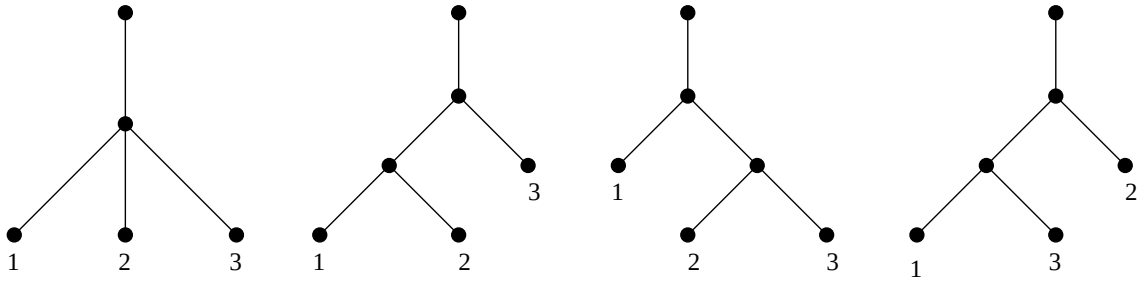


Fig. 15. ML AND BAYESIAN CLASSIFIER: The four possible topologies with three receivers.

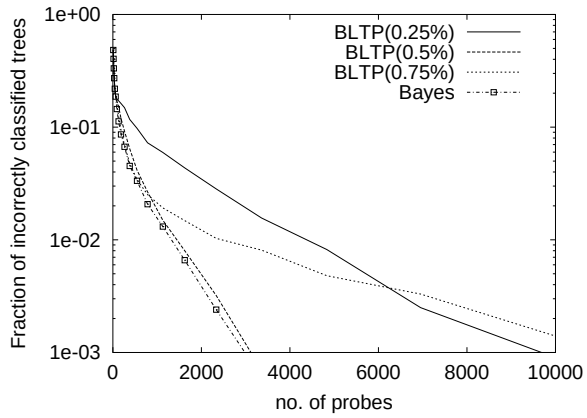
gies equally likely; we also precomputed the joint distributions $f(x|\tau,\alpha)$. Due to these computational costs, we were able to compare BLTP with the ML classifier for only up to five receivers, and restricted the Bayesian classifier to the smallest non-trivial case, that of three receivers. The four possible three-receiver trees are shown in Figure 15. In this case, the execution time of the Bayesian classifier was one order of magnitude longer than that of the ML classifier, and about two orders of magnitude longer than that of BLTP.

C.2 Relative Accuracy.

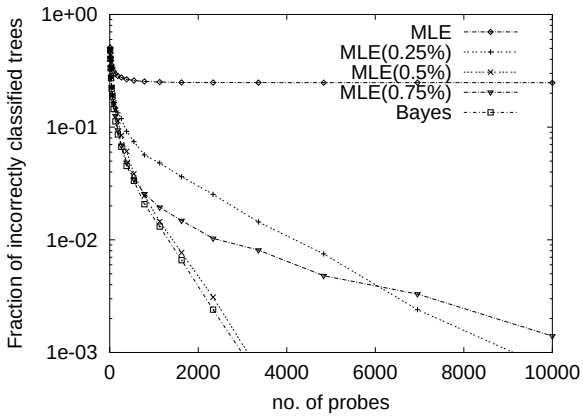
We conducted 10,000 simulations with the loss tree (τ,α) selected randomly according to the uniform prior. As remarked in Section VI, the Bayesian Classifier is, by definition, optimal in this setting. This is seen to be the case in Figure 16, where we plot the fraction of experiments in which the topology was incorrectly identified as function of the number of probes, for the different classifiers (for clarity we plot separately the curves for the ML and BLTP(ε) classifiers). Accuracy of BLTP greatly varies with ε: it gets close to optimal for the intermediate value of $\varepsilon = 0.5\%$, but rapidly decreases otherwise as ε approaches either 0 or the smallest internal link loss rate. It is interesting to observe that the ML classifier fails 25% of the time. This occurs when τ is the non-binary tree at the left in Figure 15. The reason is that the likelihood function is invariant under the insertion of links with zero loss. Statistical fluctuations present with finitely many probes lead to tree with highest likelihood to be a binary tree obtained by insertion of links with near-zero loss. This behavior does not contradict the consistency property of the ML classifier in Theorem 8; if links with loss less than some $\varepsilon > 0$ are excluded from consideration, then for sufficiently large number of probes, the spurious insertion of links will not occur.

The effect of these insertions can be suppressed by pruning after ML classification. Setting $\mathrm{ML}(\varepsilon) = \mathsf{TP}(\varepsilon) \circ \mathrm{ML}$ we find the accuracy almost identical with that of BLTP(ε); this is plotted in Figure 16(b). A more detailed inspection of the experiments shows that BLTP selects the maximum likelihood topology most of the time.

In practice we want to classify a fixed but unknown topology. In this context the uniform prior specifies a pseudo-Bayesian classifier, as in Section VI. Note that this classifier is not necessarily optimal for a fixed topology. We conducted a number of experiments of 10,000

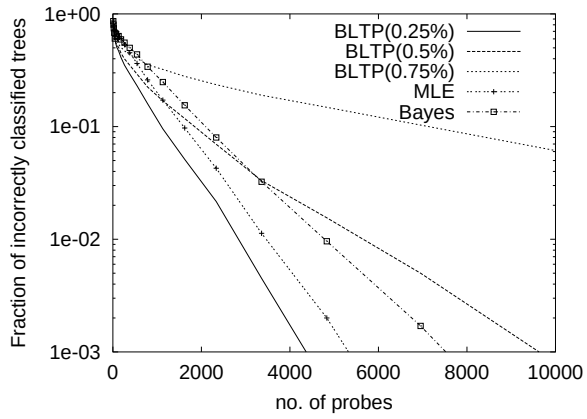


(a) random topology



(b) random topology

Fig. 16. MISCLASSIFICATION IN ML, BAYESIAN AND BLT CLASSIFIER: (τ, α) RANDOMLY DRAWN ACCORDING TO THE PRIOR DISTRIBUTION. (a) Bayes and BLTP(ε) classifier. (b) Bayes and ML classifiers.



fixed topology

	Expt.	Approx.
BLTP(0.25%)	0.00158	0.00130
BLTP(0.5%)	0.00048	0.00058
BLTP(0.75%)	0.00014	0.00014
ML	0.00105	0.00079

Fig. 17. MISCLASSIFICATION IN ML, BAYESIAN AND BLT CLASSIFIER: FIXED (τ, α). LEFT: fraction of misclassified topologies. RIGHT: Comparison of experimental and approximated tail slopes.

simulations of the three algorithms with fixed loss trees. The relative accuracy of the algorithms was found to vary with both topology and link loss rates. However, in all example we found a value of ε for which BLTP(ε) accuracy either closely approached or exceeded that of the ML and Bayesian classifiers. As an example, in Figure 17 we plot the results for the first binary tree topology in Figure 15 with all loss rates equal to 10% but that of the sole internal link, which has loss rate 1%. In this example, the ML classifier is more accurate than the pseudo-Bayesian classifier. BLTP(ε) accuracy improves as ε is decreased, and eventually, for $\varepsilon = 0.25\%$, it exceeds that of the pseudo-Bayesian and ML classifier.

These experimental results are supported by approximations to the tail slopes of the log misclassification probabilities, as detailed in Section VIII. For the same example, we display in Figure 17 (right), the estimated experimental and numerical approximated tail slopes of the ML and BLTP classifiers. For a given classifier these agree within about 25%. Finally, not reported in the Figure, we also verified that the ML(ε) classifiers provide the same accuracy as BLTP(ε).

D. Summary.

Whereas the Bayesian classifier is optimal in the context of a random topology with known prior distribution, similar accuracy can be achieved using BLTP(ε) or ML(ε) with an appropriately chosen threshold ε. In fixed topologies, the corresponding pseudo-Bayes classifier is not necessarily optimal. In the fixed topologies for which we were able to make comparisons, better accuracy could be obtained using BLTP(ε) or ML(ε) with an appropriate threshold ε. The accuracy of BLTP(ε) and ML(ε) are similar: most of the time BLTP selects the ML topology with maximum likelihood.

BLTP has the lowest complexity, primarily because

each grouping operation excludes subsets of candidate topologies from further consideration. By contrast, the ML and Bayesian classifiers used exhaustive searches through the space of possible topologies. Since the number of possible topologies grows rapidly with the number of receivers, these methods have high complexity. A more sophisticated search strategy could reduce complexity for these classifiers, but we expect this to be effective only if the number of topologies to be searched is reduced (e.g. in the manner of BLTP). With larger numbers of receivers, any fixed reduction in the per-topology computational complexity would eventually be swamped due to the growth in the number of possible topologies.

VIII. Misgrouping and Misclassification

In this section, we analyze more closely the modes of failure of BLTP, and estimate convergence rates of the probability of correct classification. Since this classifier proceeds by recursively grouping receivers, we can analyze topology misclassification by looking at how sets of receivers can be misgrouped in the estimated topology $\widehat{\mathcal{T}}$. We formalize the notion of correct receiver grouping as follows. $R_{\mathcal{T}}$ will denote the set of receivers in the logical multicast topology $\mathcal{T}$.

Definition 1: Let $(\mathcal{T}, \alpha)$ be a loss tree with $\mathcal{T} = (V, L)$, and let $(\widehat{\mathcal{T}}, \widehat{\alpha})$ be an inferred loss tree with $\widehat{\mathcal{T}} = (\widehat{V}, \widehat{L})$. The receivers $R_{\mathcal{T}}(i)$ descended from a node $i \in W$ are said to be correctly grouped in $\widehat{\mathcal{T}}$ if there exists a node $\widehat{\imath} \in \widehat{V}$ such that $R_{\mathcal{T}}(i) = R_{\widehat{\mathcal{T}}}(\widehat{\imath})$. In this case we shall say also that node i is correctly classified in $\widehat{\mathcal{T}}$.

Observe that we allow the trees rooted at i and $\widehat{\imath}$ to be different in the above definition; we only require the two sets of receivers to be equal.

Correct receiver grouping and correct topology classification are related: in the case of binary trees, the topology is correctly classified if and only if every node $k \in W$ is correctly classified. This allows us to study topology misclassification by looking at receiver misgrouping. To this end, we need to first introduce a more general form of the function $B(\cdot)$ to take into account expressions which may arise as result of classification errors. Observe that in (6) for $k \in V$ we defined $Y_k^{(i)}$ as $Y_k^{(i)} = \vee_{j \in d(k)} Y_j^{(i)} = \vee_{j \in R_{\mathcal{T}}(k)} Y_j^{(i)}$. In line 9 of BLT we have for the newly formed node U, $Y_U^{(i)} = \vee_{u \in U} Y_u^{(i)} = \vee_{j \in S} Y_j^{(i)}$, for some subset S of $R_{\mathcal{T}}$. By construction S is the set of receivers of the subtree of $\widehat{\mathcal{T}}$ rooted in U (which has been obtained by recursively grouping the nodes in S). It is clear that $S = R_{\mathcal{T}}(k)$ for some node $k \in V$ if the subtree has been correctly reconstructed, but, upon an error, can be otherwise a generic subset of $R_{\mathcal{T}}$. Therefore, in BLT we need to consider the following more general expression

$$\begin{aligned} \widehat{B}(S_1, S_2) &:= \frac{\sum_{i=1}^n \vee_{j \in S_1} Y_j^{(i)} \sum_{i=1}^n \vee_{j \in S_2} Y_j^{(i)}}{n \sum_{i=1}^n (\vee_{j \in S_1} Y_j^{(i)}) \cdot (\vee_{j \in S_2} Y_j^{(i)})} \\ &= \frac{\widehat{\gamma}(S_1)\widehat{\gamma}(S_2)}{\widehat{\gamma}(S_1) + \widehat{\gamma}(S_2) - \widehat{\gamma}(S_1 \cup S_2)} \end{aligned} \quad (14)$$

where S_1 and S_2 are two non empty disjoint subsets of $R_{\mathcal{T}}$. Analogous to Theorem 4, $\lim_{n \to \infty} \widehat{B}(S_1, S_2) = B(S_1, S_2)$, where

$$\begin{aligned} B(S_1, S_2) &:= \frac{\mathsf{P}[\vee_{j \in S_1} X_j = 1]\mathsf{P}[\vee_{j \in S_2} X_j = 1]}{\mathsf{P}[\vee_{j \in S_1} X_j \cdot \vee_{j \in S_2} X_j = 1]} \\ &= \frac{\gamma(S_1)\gamma(S_2)}{\gamma(S_1) + \gamma(S_2) - \gamma(S_1 \cup S_2)}. \end{aligned} \quad (15)$$

(15) can be regarded as a generalization of (5) where we consider a pair of disjoint sets of receivers instead of pair of nodes.

A. Misgrouping and Misclassification in BLT

We start by studying misgrouping in binary trees under BLT. Consider the event G_i that BLT correctly groups nodes in $R_{\mathcal{T}}(i)$ for some $i \in W$. This happens if grouping operations do not pair any nodes formed by recursive grouping $R_{\mathcal{T}}(i)$, with any nodes formed similarly from the complement $R_{\mathcal{T}} \setminus R_{\mathcal{T}}(i)$, until no candidate pairs in $R_{\mathcal{T}}(i)$ remain to be grouped.

Lemma 1: A sufficient condition for correct grouping of i is that

$$\widehat{D}(S_1, S_2, S_3) := \widehat{B}(S_1, S_3) - \widehat{B}(S_1, S_2) > 0 \quad (16)$$

for all $(S_1, S_2, S_3) \in \mathcal{S}(i) = \{(S_1, S_2, S_3) : S_1, S_2 \subset R_{\mathcal{T}}(i), S_3 \subseteq R_{\mathcal{T}} \setminus R_{\mathcal{T}}(i), S_k \neq \emptyset, k = 1, 2, 3, S_k \neq S_\ell, k \neq \ell\}$.

Therefore $G_i \supseteq Q_i = \cap_{(S_1, S_2, S_3) \in \mathcal{S}(i)} Q(S_1, S_2, S_3)$ where $Q(S_1, S_2, S_3)$ denotes the event that (16) holds. This provides the following upper bound for probability of misgrouping i, denoted by

$$P_i^f := \mathsf{P}[G_i^c] \leq \sum_{(S_1, S_2, S_3) \in \mathcal{S}(i)} \mathsf{P}[Q^c(S_1, S_2, S_3)] \quad (17)$$

A.1 Estimation of Misclassification Probabilities.

We now consider the asymptotic behavior of P_i^f, first for large n, then for small loss probabilities $\overline{\alpha} = 1 - \alpha$. Let $s(k) := \sum_{l \succeq k} \overline{\alpha}_l$, $k \in V$, and set $D(\cdot) = \mathsf{E}[\widehat{D}(\cdot)]$.

Theorem 10: Let $(\mathcal{T}, \alpha)$ be a canonical loss tree. For each $i \in W$, $\sqrt{n} \cdot (\widehat{D}(S_1, S_2, S_3) - D(S_1, S_2, S_3))$, $(S_1, S_2, S_3) \in \mathcal{S}(i)$, converges in distribution, as the number of probes $n \to \infty$, to a Gaussian random variable with mean 0 and variance $\sigma^2(S_1, S_2, S_3)$, with

$D(S_1,S_2,S_3) = B(S_1,S_3) - B(S_1,S_2)$. Moreover, as $\|\overline{\alpha}\| = \max_{k\in V}\overline{\alpha}_k \to 0$, then:
(i) $D(S_1,S_2,S_3) = s(a(S_1\cup S_2)) - s(a(S_1\cup S_3)) + O(\|\overline{\alpha}\|^2)$;
(ii) $\sigma^2(S_1,S_2,S_3) = s(a(S_1\cup S_2)) - s(a(S_1\cup S_3)) + O(\|\overline{\alpha}\|^2)$;
(iii)

$$\min_{(S_1,S_2,S_3)\in\mathcal{S}(i)} \frac{D^2(S_1,S_2,S_3)}{\sigma^2(S_1,S_2,S_3)} = \overline{\alpha}_i + O(\|\overline{\alpha}\|^2), \quad (18)$$

where, for small enough $\|\overline{\alpha}\|$, the minimum is attained for S_1,S_2,S_3 such that $a(S_1\cup S_2) = i$ and $a(S_1\cup S_3) = f(i)$.

Theorem 10 suggests we approximate $\mathsf{P}[Q^c(S_1,S_2,S_3)]$ by $\Psi\left(-\sqrt{n}\cdot\frac{D(S_1,S_2,S_3)}{\sigma(S_1,S_2,S_3)}\right)$, where Ψ is the cdf of the standard normal distribution. Thus for large n and small $\|\overline{\alpha}\|$, Theorem 10 and (17) together suggest that we approximate the misgrouping probability

$$P_i^f \approx e^{-\overline{\alpha}_i\frac{n}{2}} \quad (19)$$

Here we have used the fact that the P_i^f should be dominated by the summand with the smallest (negative) exponent according to (18). Thus, asymptotically for many probes, the probability of correctly identifying a group of receivers descended from node i is determined by the loss rate of link i alone, and is larger for lossier links. Moreover, the stated relations between the minimizing (S_1,S_2,S_3) in (iii) say that the likely mode of failure is to mistakenly group a child of i with the sibling of i.

In binary trees, the topology is correctly classified when all groups are correctly formed. Hence $P^f_{\mathsf{BLT}} \le \sum_{i\in W} P_i^f \approx \max_{i\in W} P_i^f$, and we expect $\log P^f_{\mathsf{BLT}}$ to be an asymptotically linear with function of n with negative slope $\overline{\alpha}^f/2$, where

$$\overline{\alpha}^f = \min_{i\in W}\overline{\alpha}_i. \quad (20)$$

Thus, in the regime considered, the most likely way to misclassify a tree is by incorrectly grouping siblings whose parent node j terminates the least lossy internal link, mistakenly grouping the sibling of j with one of its children.

We remark that the preceding argument can be formalized using Large Deviation theory [5]. However, calculation of the decay rate appears computationally infeasible, although one can recover the leading exponent $\overline{\alpha}^f/2$ in the small $\|\overline{\alpha}\|$ regime.

A.2 Experimental Evaluation.

Although we have derived the slope $\overline{\alpha}^f$ through a series of approximations, we find that it describes experimental misclassification and misgrouping reasonably well. We performed 10,000 experiments with an eight-leaf perfectly balanced binary tree. On each experiment, the loss rates are a random permutation of the elements of the set $\{0.5\%, 1\%, \ldots, 7\%, 7.5\%\}$. In this way, the smallest loss rate is fixed to 0.5%. In Figure 18 we plot the proportion of links, that had loss rates greater than or equal to a given threshold ϕ, and were misclassified. As the number of probes increases, misclassification is due exclusively to misgrouping of low loss rate links: in this set of experiments, no link with loss rate higher than 2% was misclassified once the number of probes exceeded 700.

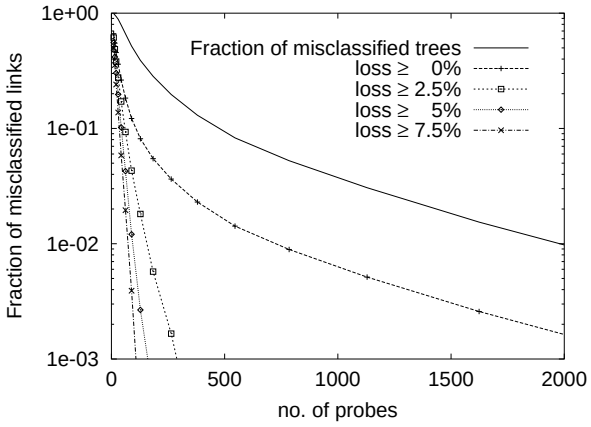
According to (19), the different curves should be asymptotically linear with negative slope approximately $\phi/2$ (then adjusted by a factor $\log_{10} e$ since the logarithms are to base 10). On the table in Figure 18(right) we display the estimated experimental and approximated slopes. Agreement is good for $\phi = 2.5\%$ and 5%. We believe the greater error for $\phi = 7.5\%$ may be due to the departure from the leading order linear approximations of (18) for larger values of $\overline{\alpha}_k$; also relatively few points are available for estimation from the experimental curves. In the figure, we also plot the log fraction of times BLT correctly identify the topology; as expected, this curve exhibits the same asymptotic linear slope of the fraction of misgrouped links, i.e., the one for $\phi = 0\%$.

B. *Misgrouping and Misclassification in* BLTP(ε)

We turn our attention to the errors in classifying general trees by the reference algorithm BLTP(ε). In the following, without loss of generality, we will study the errors in the classification of the pruned tree $(\mathcal{T}^\varepsilon,\alpha^\varepsilon) = \mathsf{TP}(\varepsilon)(\mathcal{T},\alpha)$, with $\mathcal{T}^\varepsilon = (V^\varepsilon, L^\varepsilon)$, under the assumption that $\varepsilon \neq \alpha_k$, $k\in W$. This will include, as a special case, when ε is smaller than the internal link loss rates of the underlying tree, i.e., when $\mathcal{T}^\varepsilon = \mathcal{T}$, the analysis of the misclassification of $\mathcal{T}$. $W^\varepsilon = V^\varepsilon \setminus (\{0,1\}\cup R_{\mathcal{T}^\varepsilon})$ will denote the set of nodes in $\mathcal{T}^\varepsilon$ terminating internal links.

Let $(\widehat{\mathcal{T}},\widehat{\alpha})$ denote the tree produced by BLT, the final estimate $\widehat{\mathcal{T}}^\varepsilon$ is obtained from $\widehat{\mathcal{T}}$ by pruning links whose loss rate is smaller or equal than ε, i.e., $(\widehat{\mathcal{T}}^\varepsilon,\widehat{\alpha}^\varepsilon) = \mathsf{TP}(\varepsilon)(\widehat{\mathcal{T}},\widehat{\alpha})$. In contrast to the binary case, incorrect grouping by BLT may be sufficient but not necessary for misclassification. For BLTP(ε), incorrect classification occurs if any of the following hold:
(i) at least one node in $\mathcal{T}^\varepsilon$ is misclassified in $\widehat{\mathcal{T}}^\varepsilon$;
(ii) TP(ε) prunes links from $\widehat{\mathcal{T}}$ that are present in $\mathcal{T}^\varepsilon$; or
(iii) TP(ε) fails to prune links from $\widehat{\mathcal{T}}$ that are not present in $\mathcal{T}^\varepsilon$.
Observe that (i) implies that a node i such that $\overline{\alpha}_i \le \varepsilon$ can be misclassified and still $\widehat{\mathcal{T}}^\varepsilon = \mathcal{T}^\varepsilon$ provided the all the resulting erroneous links are pruned.

We have approximated the probability of errors of type (i) in our analysis of BLT. Errors of type (ii) are excluded



ϕ	Expt.	Approx.
0%	0.0005	0.0011
2.5%	0.0051	0.0054
5.0%	0.0097	0.0109
7.5%	0.0248	0.0163

Fig. 18. MISCLASSIFICATION AND MISGROUPING IN BLT. LEFT: fraction of links misclassified with loss $\geq \phi$, for $\phi = 0\%, 2.5\%, 5.0\%.7.5\%$. RIGHT: Comparison of experimental and approximated tail slopes.

if for all $i \in W^\varepsilon$:

$$\widehat{D}(S_1,S_2,S_3,\varepsilon) := \widehat{B}(S_1 \cup S_2, S_3)(1-\varepsilon) - \widehat{B}(S_1,S_2) > 0 \quad (21)$$

for all $(S_1,S_2,S_3) \in \mathcal{S}(i)$, since this condition implies that all estimated loss rates of links in the actual tree are greater than ε. Errors of type (iii) are excluded if $\widehat{B}(S_1,S_3) - \widehat{B}(S_1,S_2) > 0$ and $\widehat{B}(S_1 \cup S_2, S_3)(1-\varepsilon) - \widehat{B}(S_1,S_2) \leq 0$, or if $\widehat{B}(S_1,S_3) - \widehat{B}(S_1,S_2) \leq 0$ and $\widehat{B}(S_1 \cup S_3, S_2)(1-\varepsilon) - \widehat{B}(S_1,S_3) \geq 0$ for all $(S_1,S_2,S_3) \in \mathcal{S}(\varepsilon)$ where $\mathcal{S}(\varepsilon) = \{(S_1,S_2,S_3) : S_j \subset R;\ S_j \neq \emptyset;\ S_j \cap S_k = \emptyset, j \neq k;\ (S_1 \cup S_2 \cup S_3) \cap R_{\mathcal{T}}(i) = \emptyset \vee (S_1 \cup S_2 \cup S_3) \subseteq R_{\mathcal{T}}(i) \vee \exists j \in \{1,2,3\}\, R_{\mathcal{T}}(i) \subseteq S_j, i \in W^\varepsilon\}$. The latter conditions ensure that all the links in the binary tree produced by BLT, which are either results of node misgrouping or corresponding to fictitious links due to binary reconstruction, have estimated loss rate less than or equal to ε, and are hence pruned. Summarizing, let $Q(S_1,S_2,S_3,\varepsilon)$ be the event that (21) holds for a given (S_1,S_2,S_3), and $G(\varepsilon)$ the event that the topology is correctly classified. Then $G(\varepsilon) \supseteq \cap_{k \in V^\varepsilon \setminus R_{\mathcal{T}}}(Q_k \cap Q_k^{(ii)}(\varepsilon)) \cap Q^{(iii)}(\varepsilon)$ where $Q_k^{(ii)}(\varepsilon) = \cap_{(S_1,S_2,S_3)\in\mathcal{S}(k)} Q(S_1,S_2,S_3,\varepsilon)$ and $Q^{(iii)}(\varepsilon) = \cap_{(S_1,S_2,S_3)\in\mathcal{S}(\varepsilon)}(Q(S_1,S_2,S_3) \cap Q(S_1,S_2,S_3,\varepsilon)^c) \cup (Q(S_1,S_2,S_3)^c \cap Q(S_1,S_3,S_2,\varepsilon)^c)$. Consequently, we can write a union bound for the probability of misclassification:

$$P^f_{\mathsf{BLTP}(\varepsilon)} := P[G(\varepsilon)^c] \quad (22)$$
$$\leq \sum_{k \in W^\varepsilon} \left(\mathsf{P}[Q_k^c] + \mathsf{P}[Q_k^{(ii)}(\varepsilon)^c] \right) + \mathsf{P}[Q^{(iii)}(\varepsilon)^c]$$

and each term in (22) can in turn be bounded above by a sum similar to the RHS of (17). For the last term, in particular, observe that

$$Q^{(iii)}(\varepsilon)^c = \cup_{(S_1,S_2,S_3)\in\mathcal{S}(\varepsilon)}(Q(S_1,S_2,S_3)^c \cup \quad (23)$$
$$Q(S_1,S_2,S_3,\varepsilon)) \cap (Q(S_1,S_2,S_3) \cap Q(S_1,S_3,S_2,\varepsilon))$$
$$\subseteq \cup_{(S_1,S_2,S_3)\in\mathcal{S}(\varepsilon)}(Q(S_1,S_2,S_3,\varepsilon) \cup Q(S_1,S_3,S_2,\varepsilon))$$
$$= \cup_{(S_1,S_2,S_3)\in\mathcal{S}(\varepsilon)} Q(S_1,S_2,S_3,\varepsilon),$$

so that $\mathsf{P}[Q^{(iii)}(\varepsilon)^c] \leq \sum_{(S_1,S_2,S_3)\in\mathcal{S}(\varepsilon)} Q(S_1,S_2,S_3,\varepsilon)$.

B.1 Misclassification Probabilities and Experiment Duration.

We examine the asymptotics of the misclassification probability $P^f_{\mathsf{BLTP}(\varepsilon)}$ for large n and small $\|\overline{\alpha}\|$, by the same means as in Section VIII-A. This amounts to finding the mean $D(S_1,S_2,S_3,\varepsilon)$ and asymptotic variance $\sigma^2(S_1,S_2,S_3,\varepsilon)$ of the distribution of $\widehat{D}(S_1,S_2,S_3,\varepsilon)$, then finding the dominant exponent D^2/σ^2 over the various (S_1,S_2,S_3). Let $\overline{\alpha}^f(\varepsilon) = \min_{i\in W^\varepsilon} \overline{\alpha}_i$ denote the smallest internal link loss rate of $\mathcal{T}^\varepsilon$ larger than ε and $\overline{\alpha}^p(\varepsilon) = \max_{i\in W\setminus W^\varepsilon} \overline{\alpha}_i$ the largest internal link loss rate of $\mathcal{T}$ smaller than ε or $\overline{\alpha}^p(\varepsilon) = 0$ if no such loss rate exists (which occurs when ε is smaller than all internal links loss rate). The proof of the following result is similar to that of Theorem 10 and is omitted.

Theorem 11: Let $(\mathcal{T},\alpha)$ be a canonical loss tree. For each $0 \leq \varepsilon < 1$, $(S_1,S_2,S_3) \in \cup_{i\in W^\varepsilon}\mathcal{S}(i) \cup \mathcal{S}(\varepsilon)$, $\sqrt{n}\cdot(\widehat{D}(S_1,S_2,S_3,\varepsilon) - D(S_1,S_2,S_3,\varepsilon))$ converges in distribution, as the number of probes $n \to \infty$, to a Gaussian random variable with mean 0 and variance $\sigma^2(S_1,S_2,S_3,\varepsilon)$. Furthermore, as $\|\overline{\alpha}\| = \max_{k\in V} \overline{\alpha}_k \to 0$ and $\varepsilon/\|\overline{\alpha}\| \to c \in (0,\infty)$,

(i) $D(S_1,S_2,S_3,\varepsilon) = s(a(S_1 \cup S_2)) - s(a(S_1 \cup S_3)) - \varepsilon + O(\|\overline{\alpha}\|^2)$;

(ii) $\sigma^2(S_1,S_2,S_3,\varepsilon) = |s(a(S_1 \cup S_2)) - s(a(S_1 \cup S_3))| + O(\|\overline{\alpha}\|^2)$;

(iii) If $(S_1, S_2, S_3) \in \mathcal{S}(i)$, $i \in W^\varepsilon$,

$$\min_{(S_1,S_2,S_3)\in\mathcal{S}(i)} \frac{D^2(S_1,S_2,S_3,\varepsilon)}{\sigma^2(S_1,S_2,S_3,\varepsilon)} = \frac{(\overline{\alpha}_i - \varepsilon)^2}{\overline{\alpha}_i} + O(\|\overline{\alpha}\|^2) \tag{24}$$

and

$$\min_{i\in W^\varepsilon} \min_{(S_1,S_2,S_3)\in\mathcal{S}(i)} \frac{D^2(S_1,S_2,S_3,\varepsilon)}{\sigma^2(S_1,S_2,S_3,\varepsilon)} = \frac{(\overline{\alpha}^f(\varepsilon) - \varepsilon)^2}{\overline{\alpha}^f(\varepsilon)} + O(\|\overline{\alpha}\|^2). \tag{25}$$

If $(S_1, S_2, S_3) \in \mathcal{S}(\varepsilon)$,

$$\min_{(S_1,S_2,S_3)\in\mathcal{S}(\varepsilon)} \frac{D^2(S_1,S_2,S_3,\varepsilon)}{\sigma^2(S_1,S_2,S_3,\varepsilon)} = \begin{cases} O(\varepsilon^2/\|\overline{\alpha}\|^2) & \text{if } \overline{\alpha}^p(\varepsilon) = 0 \\ \frac{(\varepsilon-\overline{\alpha}^p(\varepsilon))^2}{\overline{\alpha}^p(\varepsilon)} + O(\|\overline{\alpha}\|^2) & \text{if } \overline{\alpha}^p(\varepsilon) > 0 \end{cases} \tag{26}$$

In (27) above, for clarity we distinguish the expressions for $\overline{\alpha}^p(\varepsilon) = 0$ and $\overline{\alpha}^p(\varepsilon) > 0$. Observe that the result for $\overline{\alpha}^p(\varepsilon) = 0$ in (27) can be actually obtained by taking the limit of the expression for $\overline{\alpha}^p(\varepsilon) > 0$, which is of the form $\Big((\varepsilon - \overline{\alpha}^p(\varepsilon))^2 + O(\|\overline{\alpha}\|^3)\Big) \,/\, \Big(\overline{\alpha}^p(\varepsilon) + O(\|\overline{\alpha}\|^2)\Big)$.

Using the same reasoning as was used in Section VIII-A, we expect that the logarithms of the probabilities of errors of type (i), (ii) and (iii) to be asymptotically linear in the number of probes n, with slopes that behave respectively as

$$c^{(i)} = \overline{\alpha}^f(\varepsilon)/2, \quad c^{(ii)} = \frac{(\overline{\alpha}^f(\varepsilon) - \varepsilon)^2}{2\overline{\alpha}^f(\varepsilon)}, \tag{27}$$

$$c^{(iii)} = \begin{cases} O(\varepsilon^2/\|\overline{\alpha}\|^2) & \text{if } \overline{\alpha}^p(\varepsilon) = 0 \\ \frac{(\varepsilon-\overline{\alpha}^p(\varepsilon))^2}{2\overline{\alpha}^p(\varepsilon)} & \text{if } \overline{\alpha}^p(\varepsilon) > 0 \end{cases}$$

The dominant mode of misclassification is that with the lowest slope in (27), which then dominates the sum in (22) for large n. Hence we approximate the misclassification probability to leading exponential order by

$$P^f_{\mathsf{BLTP}(\varepsilon)} \approx e^{-(n/2)\min\{c^{(i)},c^{(ii)},c^{(iii)}\}}. \tag{28}$$

Since $c^{(i)} \geq c^{(ii)}$, type (ii) errors always dominate type (i). Between type (ii) and (iii), the prevailing type of errors depends on the relative magnitude of $\overline{\alpha}^f(\varepsilon)$, $\overline{\alpha}^p(\varepsilon)$ and ε, which satisfy $\overline{\alpha}^p(\varepsilon) < \varepsilon < \overline{\alpha}^f(\varepsilon)$. Type (ii) becomes prevalent as $\varepsilon \to \overline{\alpha}^f(\varepsilon)$ since then $c^{(ii)} \to 0$; similarly, type (iii) dominates as $\varepsilon \to \overline{\alpha}^p(\varepsilon)$. Thus, ε should be chosen large enough to avoid the type (iii) errors, but small enough so that the probability of type (ii) does not become large. Unfortunately, this is not possible unless information on the actual link loss rates is available. We believe, nevertheless, that this does not represent a problem in practice. Indeed, as the analysis above indicates,

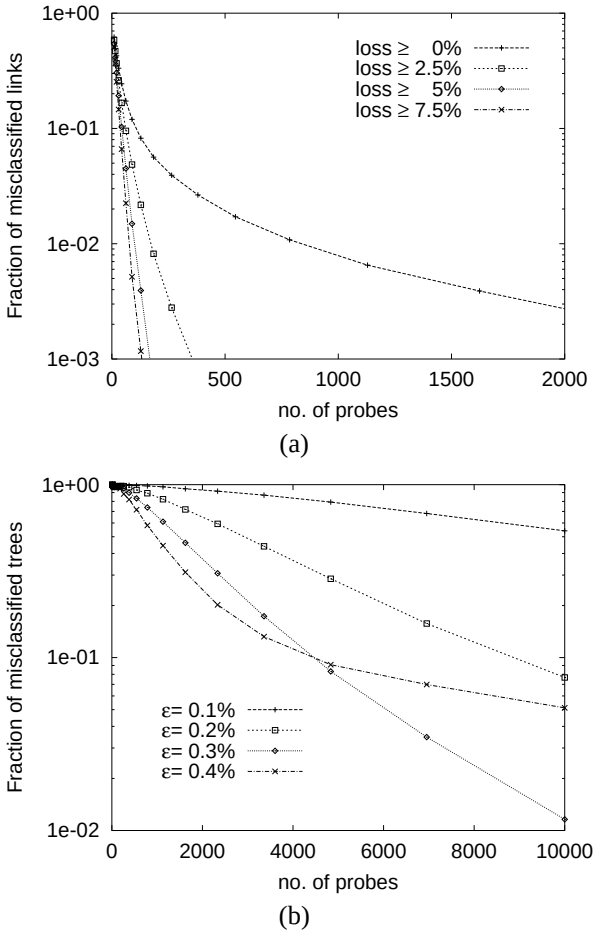


Fig. 19. MISCLASSIFICATION AND MISGROUPING IN $\mathsf{BLTP}(\varepsilon)$: (a) fraction of misclassified links with loss $\geq \phi$, for $\phi = 0\%, 2.5\%, 5.0\%.7.5\%$; (b) fraction of misclassified trees for $\varepsilon = 0.1\%, 0.2\%, 0.3\%, 0.4\%$.

for enough large n, the most likely way $\mathsf{BLTP}(\varepsilon)$ misclassify a tree is by either pruning the link which the least loss rate higher than ε (a type (ii) error) or by not pruning that with the the largest loss rate smaller than ε (a type (iii) error); either way, the resulting inferred tree would differ from the actual by the at most one link, approximatively, that with the loss rate closest to ε.

The foregoing arguments allow us to also estimate the number of probes N required for inference with misclassification probability δ in a tree with minimum link loss rate $\overline{\alpha}^f$. This is done by inverting the approximation (28) to obtain that N is approximately

$$\begin{cases} -\frac{2\overline{\alpha}^f(\varepsilon)\log\delta}{(\varepsilon-\overline{\alpha}^f(\varepsilon))^2} & \text{if } \overline{\alpha}^p(\varepsilon) = 0 \\ -2\max\left\{\frac{\overline{\alpha}^f(\varepsilon)}{(\varepsilon-\overline{\alpha}^f(\varepsilon))^2}, \frac{\overline{\alpha}^p(\varepsilon)}{(\varepsilon-\overline{\alpha}^p(\varepsilon))^2}\right\}\log\delta & \text{if } \overline{\alpha}^p(\varepsilon) > 0 \end{cases} \tag{29}$$

Note that for BLT, or when $\varepsilon \ll \overline{\alpha}^f$, this reduces to the simple form $N \approx -2\log(\delta)/\overline{\alpha}^f$.

We conclude by observing that in the above analysis, we have implicitly assumed that $W^\varepsilon \neq \emptyset$. Nevertheless, for large enough ε, $W^\varepsilon = \emptyset$ which corresponds to the case when $\mathcal{T}^\varepsilon$ is a degenerate tree where all leaf nodes are siblings. In this case, it is clear that misclassification occurs only because of type (iii) errors. The misclassification analysis for this special case can then be obtained by taking into account type (iii) errors alone.

B.2 Experimental Evaluation.

We performed 10,000 experiments in a 21 node tree with mixed branching ratio 2 and 3. On each experiment, the loss rates are a random permutation of the elements of the set $\{0.5\%, 1\%, \ldots, 9.5\%, 10\%\}$, thus having the same smallest link loss as in the experiments for BLT. In Figure 19 we plot the fraction of links, that had loss rates greater than or equal to a given threshold ϕ, and were misclassified. These appear very similar to those for BLT in Figure 18. In Figure 19(b) we also plot the fraction of misclassified trees using $\mathsf{BLTP}(\varepsilon)$ for different values of ε, all smaller than the smallest loss rate of 0.5%. With this choice, $\overline{\alpha}^p(\varepsilon) = 0$ and $\overline{\alpha}^f(\varepsilon) = 0.5\%$. As expected, accuracy is best for intermediate ε. The difference in shape between the last and the first three curves indicates the change between the two different regimes of misclassification. For ε smaller than 0.4%, misclassification is dominated by erroneous exclusion of nodes from a group, while for $\varepsilon = 0.4\%$, misclassification is mostly determined by erroneous pruning of the link with the smallest loss rate (which is 0.5%) because of statistical fluctuation of its inferred loss rate below ε. In the latter case, we can use (27) to compute the tail slope obtaining 4.3×10^{-4}, in good agreement with the estimated experimental slope which is 4.1×10^{-4}.

B.3 Asymptotic Misclassification Rates for the ML-Classifier

We sketch how the theory of large deviations [5] can be used to bound the asymptotic probability of misclassification by the ML estimator. The expressions obtained here were used to determine the ML tail slopes in the table in Figure 17. First, observe that $\mathsf{P}_{\mathcal{T},\alpha}(\widehat{\mathcal{T}}_{\mathrm{ML}} \neq \mathcal{T}) = \sum_{\tau \neq \mathcal{T}} \mathsf{P}_{\mathcal{T},\alpha}(\widehat{\mathcal{T}}_{\mathrm{ML}} = \tau)$. For $\tau \neq \mathcal{T}$, each term in this sum can be bounded above by $\mathsf{P}_{\mathcal{T},\alpha}\left(\cup_{\alpha' \in \mathcal{A}_\tau} \{n^{-1} \sum_{i=1}^n g(X^{(i)}; \tau, \alpha') > 0\}\right)$, where $g(x; \tau, \alpha') = \log(p(x; \tau, \alpha')/p(x; \mathcal{T}, \alpha))$ and $p(x; \mathcal{T}, \alpha)$ the probability of the outcome $x \in \Omega = \{0,1\}^R$ under the loss tree $(\mathcal{T}, \alpha)$. Let $\mu_n = n^{-1} \sum_{i=1}^n \delta_{X^{(i)}}$ denote the empirical distribution of the first n quantities $X^{(i)}$ (here δ_x is the unit mass at x), and for each τ and $\alpha' \in \mathcal{A}_\tau$ let $\Gamma_{\tau,\alpha'} = \{\nu \in M_1(\Omega) : \sum_{x \in \Omega} g(x; \tau, \alpha')\nu(x) \geq 0\}$ (here $M_1(\Omega)$ is the set of probability measures on Ω) and set $\Gamma_\tau = \cup_{\alpha' \in \mathcal{A}_\tau} \Gamma_{\tau,\alpha'}$. Since the $g(X^{(i)}, \tau, \alpha')$ are IID random variables, we can use Sanov's Theorem [5] to conclude that

$$\begin{aligned} \limsup_{n\to\infty} \frac{1}{n} \log \mathsf{P}_{\mathcal{T},\alpha}(\widehat{\mathcal{T}}_{\mathrm{ML}} = \tau) \\ \leq \limsup_{n\to\infty} \frac{1}{n} \log \mathsf{P}_{\mathcal{T},\alpha}(\mu_n \in \Gamma_\tau) \\ \leq - \inf_{\nu \in \Gamma_\tau} K(\nu \mid f(\cdot; \mathcal{T}, \alpha)). \end{aligned} \tag{30}$$

Here, for $\nu, \eta \in M_1(\Omega)$, $K(\nu \mid \eta) = \sum_{x \in \Omega} \nu(x) \log(\nu(x)/\eta(x))$ is the Kullback-Leibler "distance", or entropy of ν relative to η. By further minimizing the right-hand term of (30) over all $\tau \neq \mathcal{T}$, we obtain an asymptotic upper bound for the decay rate of the misclassification probability as n increases. For each τ, the minimization can be carried out using the Kuhn-Tucker theorem; we use the form given in [15].

We mention that a lower bound of the following form can be found:

$$\begin{aligned} \liminf_{n\to\infty} \frac{1}{n} \log \mathsf{P}_{\mathcal{T},\alpha}(\widehat{\mathcal{T}}_{\mathrm{ML}} \neq \mathcal{T}) \quad \geq \\ - \inf\{K(\tau', \alpha' | \mathcal{T}, \alpha) : \tau' \neq \mathcal{T}, \alpha' \in \mathcal{A}_{\tau'}\} \end{aligned} \tag{31}$$

IX. Summary and Conclusions

In this paper we have proposed and established the consistency of a number of algorithms for inferring logical multicast topology from end-to-end multicast loss measurements. The algorithms fall in two broad classes: the grouping algorithms (BLTP, BLTC and GLT), and the global algorithms (ML and Bayesian).

The computational cost of the grouping approaches is considerably less for two reasons: (i) they work by progressively excluding subsets of candidate topologies from consideration while the global algorithms inspect all topologies; and (ii) their cost per inspection of each potential sibling set is lower. Of the grouping algorithms, the BLTP approach of treating the tree as binary then pruning low loss links is simplest to implement and execute.

Of the algorithms presented, only the Bayesian is able to identify links with arbitrarily small loss rates. All the other classifiers require a parameter $\varepsilon > 0$ that acts as a threshold: a link with loss rate below this value will be ignored and its endpoints identified. The threshold is required in order that sibling groups not be separated due to random fluctuations of the inferred loss rates. However, we do not believe that the necessity of a threshold presents an obstacle to their use in practice, since it is the identification of high loss links that is more important for performance diagnostics. In practice we expect ε to be chosen according to an application-specific notion of a minimum relevant loss rate.

By construction, the Bayesian classifier has the greatest accuracy in the context of classification of topologies

drawn according to a known random distribution. However, the performance gap narrows when classifying a fixed unknown topology, and in fact the Bayesian classifier has slightly worse performance than the others in this context. We conclude that BLTP offers the best performance, having the lowest computational cost for near optimal performance.

This selection of BLTP(ε) motivates analyzing its error modes, and their probabilities. Although the analysis is quite complex, a simple picture emerges in the regime of small loss rates $\overline{\alpha}_k$ and many probes n, and errors are most likely to occur when grouping the children of the node j that terminates the link of lowest loss rate.

The leading exponents for the misclassification that were calculated in Section VIII can be used to derive rough estimates of the number of probes required in practice. Consider the problem of classifying a general topology whose smallest link loss rate in 1%. According to (29), the number of probes required for a misclassification probability of 1% (using $\varepsilon = 0.5\%$) is about 4000. (In a binary topology using BLT the number required drops to about 1000). Using small (40 byte) probes at low rate of a few tens of kbits per sec, measurements involving this many probes could be completed within only a few minutes.

We note that the grouping methods extend to a wider class of estimators by replacing the shared loss estimate with any function on the nodes (i) that increases on moving away from the root; and (ii) whose value at a node can be consistently estimated from measurements at receivers descended from that node. Examples of such quantities include the mean and variance of the cumulative delay from the root to a given node; see [6] and [11].

Finally, a challenging problem is to take the resulting logical multicast trees and mapping the constituent nodes onto physical routers within real networks. This remains beyond our capability at this time.

X. Proofs of the Theorems

The proof of Proposition 1 depends in the following Lemma.

Lemma 2: Let $g_i > 0$ for $i = 1, 2, \ldots, m$; let g be such that $\min_i g_i < g < \sum_i g_i$; and set $f(b) := (1 - g/b) - \prod_i (1 - g_i/b)$. Then the equation $f(b) = 0$ has a unique solution $b^* > g$. Furthermore, given $b > g$ then $f(b) > 0$ if and only if $b > b^*$.

Proof of Lemma 2: Set $c_i = g_i/g < 1$ so that $\sum_i c_i > 1$. Let $h_1(x) = (1 - x)$, $h_2(x) = \prod_i (1 - c_i x)$ and $h = h_1 - h_2$, so that $f(b) = h(g/b)$. We look for zeroes of h. For $x \in [0, 1]$ $h_1''(x) = 0$, $h_2''(x) = h_2(x) \left\{ \left(\sum_i q_i(x)\right)^2 - \sum_i q_i(x)^2 \right\} > 0$ where $q_i(x) = c_i/(1 - c_i x) > 0$. Hence h is strictly concave on $[0, 1]$. Now $h(0) = 0$, $h(1) < 0$ and $h'(0) = -1 + \sum_i c_i > 0$. So since h is concave and continuous on $[0, 1]$ there must be exactly one solution x^* to $h(x) = 0$ for $x \in (0, 1)$ and hence one solution b^* to $f(b) = 0$ for $b > g$. Furthermore, given $x \in (0, 1)$, $h(x) > 0$ iff $x < x^*$ and hence given $b > g$, $f(b) > 0$ iff $b > b^*$. ∎

Proof of Proposition 1: Clearly $\min_{k \in U} \gamma(k) < \gamma(U) < \sum_{k \in U} \gamma(U)$ in a canonical loss tree and hence (i) and (ii) follow from Lemma 2. (iii) is then a restatement of (2), established during the proof of Prop. 1 in [3].

(iv) Write $U' = U \cup \{k\}$. We refer to Figure 1, where we show the logical multicast subtree spanned by k, U and their descendents, together with $a(U), a(U')$ and the root 0. From (i), $B(U')$ is the solution of the equation

$$\left(1 - \frac{\gamma(U')}{B(U')}\right) = \left(1 - \frac{\gamma(k)}{B(U')}\right) \prod_{j \in U} \left(1 - \frac{\gamma(j)}{B(U')}\right). \tag{32}$$

and $B(U)$ is the solution of

$$(1 - \gamma(U)/B(U)) = \prod_{j \in U} (1 - \gamma(j)/B(U)) \tag{33}$$

Now suppose that $B(U) \geq B(U')$. We shall show that this leads to a contradiction. Since then $B(U) \geq B(U') > \gamma(U')$, we can apply (i) and (ii) to (32) to obtain

$$\begin{aligned} 1 - \frac{\gamma(U')}{B(U)} &\geq \left(1 - \frac{\gamma(k)}{B(U)}\right) \prod_{j \in U} \left(1 - \frac{\gamma(j)}{B(U)}\right) \\ &= \left(1 - \frac{\gamma(k)}{B(U)}\right) \left(1 - \frac{\gamma(U)}{B(U)}\right), \end{aligned} \tag{34}$$

with the right-hand equality obtained by substitution of (33). Applying (2) at the node $a(U')$ we have

$$1 - \frac{\gamma(U')}{A(a(U'))} = \left(1 - \frac{\gamma(k)}{A(a(U'))}\right) \left(1 - \frac{\gamma(U)}{A(a(U'))}\right). \tag{35}$$

Since the assumption $B(U) \geq B(U')$ implies that $B(U) > \gamma(U')$, then comparing (34) with (35) and using (ii) again we find $A(a(U')) \leq B(U) = A(a(U))$. This is a contradiction since $a(U') \succ a(U)$ and T canonical implies $A(a(U')) > A(a(U))$. ∎

While proving that DLT reconstructs the tree correctly, we find it useful to identify a subset S of V as a **stratum** if $\{R(k) : k \in S\}$ is a partition of R. If DLT works correctly, then before each execution of the while loop at line 4 of Figure 2, the set R' is a stratum and the set (V', L') of nodes and links is consistent with the actual tree (V, L) in the sense that it decomposes over subtrees rooted at the stratum R', i.e., $V' = \cup_{k \in R'} V(k)$ and $L' = \cup_{k \in R'} L(k)$. This is because any correct iteration of the loop that groups the children of node k has the effect

of joining subtrees rooted at nodes in $d(k)$, while modifying the partition $\{R(k) : k \in R'\}$ by replacing elements $\{R(j) : j \in d(k)\}$ by $R(k)$. The proof of Theorem 2 depends on the following Lemma that collects some properties of strata.

Lemma 3: If S is a stratum in a logical multicast tree (V, L) then

(i) If $k \in S$ then no ancestor or descendant of k lies in S.

(ii) Exactly one of the following alternatives applies to each non-root node k in V: (a) $k \in S$; (b) k has an ancestor in S; (c) k has at least two descendants in S.

Proof of Lemma 3: (i) If $j, k, \in S$ and $j \prec k$ then $R(j) \subset R(k)$, contradicting the partition property. (ii) If $k \notin S$, then there exists $j \in S$ obeying one of the alternatives $j \succ k$ or $j \prec k$, for otherwise $R(j)$ does not overlap with any element of the partition $\{R(j) : j \in S\}$. By (i), the alternatives are exclusive. There exists $j \in S$ with $j \succ k$, it is unique, by (i). If not, there exists $j \in S$ with $j \prec k$. In this case k cannot be a leaf node and hence $R(j) \subsetneq R(k)$ since k has branching ratio at least 2. Hence there must be at least one more node $j' \in S$ with $j \prec k$, since otherwise the partition $\{R(j) : j \in S\}$ would not cover R. ■

Proof of Theorem 2: (i) Suppose that DLT yields an incorrect tree, and consider the first execution of the loop during which (V', L') becomes inconsistent. Inconsistency could occur for the following reasons only:

1. *If the minimizing pair $\{u_1, u_2\}$ are not siblings.* Then there exists $t \prec a(u_1, u_2)$ that is the parent of either u_1 or u_2; say $t \succ u_1$. Since $u_1 \in R'$, by Lemma 3(i) no ancestor of u_1 – including t – can be in R'. Hence by Lemma 3(ii), there must be at least one node u_2' in addition to u_1 with the property that $u_2' \prec t$ and $u_2' \in R'$. Since the loss tree is canonical, $B(u_1, u_2') \leq A(t) < A(a(u_1, u_2)) = B(u_1, u_2)$, contradicting the minimality of $B(u_1, u_2)$. Hence the minimizing pair are siblings.
2. *If not all sibling nodes of u_1, u_2 are members of R'.* Let there be a sibling s of u_1 that is not in R'. Since $u_1 \in R'$, then by Lemma 3(i) no ancestor of u_1 – and hence no ancestor of its sibling s – can lie in R'. Since s itself is not in R', by Lemma 3(ii), there exist $s_1, s_2 \in R'$ with ancestor s. Since the loss tree is canonical, $B(s_1, s_2) \leq A(s) < A(a(u_1, u_2)) = B(u_1, u_2)$, contradicting the minimality of $B(u_1, u_2)$. Hence all siblings of u_1, u_2 are members of R'.
3. *If not all sibling nodes of u_1, u_2 are included in U of steps 5–7.* This would violate Prop 1(iii).
4. *If a node that is not a sibling of u_1, u_2 is included in U.* This would violate Prop 1(iv).

(ii) Since (i) allows the reconstruction of the loss tree from the outcome distribution, distinct loss trees can not give rise to the same outcome distributions, and hence the canonical loss tree is identifiable. ■

Proof of Theorem 3: Consider a maximal set $U = \{u_1, u_2, \ldots, u_m\}$ of siblings that is formed by execution of the while loop at line 6 in DLT; see Figure 2. We assume the non-trivial case that $m > 2$ and assume initially that U is unique. By Prop. 1(iii), $B(\cdot)$ is minimal within $R^{(0)} := R'$ on any pair of nodes from $S^{(0)} := U$. The action of DBLT can be described iteratively over $\ell = \{0, 1, \ldots, m\}$ as follows. After selecting $U^{(\ell)} = \{u_1^{(\ell)}, u_2^{(\ell)}\}$ in line 5, all pairs in $S^{(\ell+1)} = (S^{(\ell)} \setminus U^{(\ell)}) \cup \{U^{(\ell)}\}$ minimize $B(\cdot)$ over all pairs in $R^{(\ell+1)} = (R^{(\ell)} \setminus U^{(\ell)}) \cup \{U^{(\ell)}\}$ with the same minimum $B(U)$. This is because $(1 - \gamma(U^{(\ell)})/B(U)) = \prod_{u \in \mathcal{U}(\ell)}(1 - \gamma(u)/B(U))$ where $\mathcal{U}(\ell)$ denotes the members of U that are descended from $U^{(\ell)}$ in the binary tree built by DBLT. Hence $(1 - \gamma(U^{(\ell)})/B(U)) = (1 - \gamma(u_1^{(\ell)})/B(U))(1 - \gamma(u_2^{(\ell)})/B(U))$ and so $B(U^{(\ell)}) = B(U)$ by Prop. 1(i).

Thus for each step in DLT that groups the nodes in U, there are $m - 1$ steps of DBLT that successively group the same set of nodes. Since $B(U^{(\ell)}) = B(U)$ for all ℓ, each node j added in DBLT has $\alpha_j = 1$, apart from the last one. Therefore, TP(0) acts to excise all links between the binary nodes $U^{(0)}, \ldots, U^{(m)} - 1$. Thus DLT $=$ TP(0) $\circ$ DBLT. If U in not unique, the same arguments apply, except now there can be alternation of grouping operations acting on different maximal sibling sets. ■

Proof of Theorem 4: Since each $\widehat{\gamma}(U)$ is the mean of n independent random variables then by the Strong Law of Large Numbers, $\widehat{\gamma}(U)$ converges to $\mathsf{E}[\widehat{\gamma}(U)] = \gamma(U)$ almost surely as $n \to \infty$. In Theorem 1 of [3] it is shown that $B(U)$ is a continuous function of $\{\gamma(a(U)), \{\gamma(k) : k \in U\}\}$, from which the result follows. ■

Proof of Theorem 5: Let U denote a generic binary subset of R' that minimizes $B(\cdot)$ when DBLT is applied to $(\mathcal{T}, \alpha)$. Assume initially that the minimizing U is unique. Since the loss tree is canonical, $B(U) < B(U')$ for any other candidate binary set U'; by the convergence property of Theorem 4, $\widehat{B}(U) < \widehat{B}(U')$ for all n sufficiently large. Hence the nodes in U are grouped correctly.

But it may happen, by coincidence, that the minimizing U is not unique. Then there are pairs $U^{(1)}, \ldots, U^{(m)}$ that minimize B. Since the tree is canonical, then after each $U^{(\ell)} \in R'$ has been grouped, the remaining pairs are still minimizers of $B(\cdot)$ amongst all pairs of the reduced set $(R' \setminus U^{(\ell)}) \cup \{U^{(\ell)}\}$ in line 10 of Figure 4. Hence DBLT picks these pairs successively for grouping until all pairs have been picked.

With BLT, the $\widehat{B}(U^{(\ell)})$ are no longer equal. But for sufficiently large n they will still all be less than $\widehat{B}(U')$ for any other candidate pair U', by Theorem 4. Thus BLT will successively group the pairs $U^{(1)}, \ldots, U^{(m)}$ in some

random order that depends on the relative magnitude of the $\widehat{B}(U^{(\ell)})$. But the order is not important, since the end result is just to have the pairs formed as DBLT would have. ∎

Proof of Theorem 8: It suffices to show that $\lim_{n\to\infty} \mathsf{P}_{\mathcal{T},\alpha}(\widehat{\mathcal{T}}_{\rm ML} = \mathcal{T}') = 0$ for each $\mathcal{T}' \neq \mathcal{T}$, Let $p(x;\mathcal{T},\alpha)$ denote the probability of the outcome $x \in \{0,1\}^R$ under the loss tree $(\mathcal{T},\alpha)$. Under our assumptions, if $\mathcal{T}' \neq \mathcal{T}$, the Kullback-Leibler information

$$I\left((\mathcal{T},\alpha),(\mathcal{T}',\alpha')\right) := \mathsf{E}_{\mathcal{T},\alpha}(\log(p(X;\mathcal{T},\alpha)/p(X;\mathcal{T}',\alpha'))) \tag{36}$$

is a continuous function of $\alpha' \in \mathcal{A}^{\varepsilon}_{\mathcal{T}'}$, and is strictly positive because of identifiability. Thus there is a number $\delta > 0$ such that $I\left((\mathcal{T},\alpha),(\mathcal{T}',\alpha')\right) \geq \delta$ for all $\alpha' \in \mathcal{A}^{\varepsilon}_{\mathcal{T}'}$. Now

$$\mathsf{P}_{\mathcal{T},\alpha}(\widehat{\mathcal{T}}_{\rm ML} = \mathcal{T}') \leq \tag{37}$$
$$\mathsf{P}_{\mathcal{T},\alpha}\left(\cup_{\alpha'\in\mathcal{A}^{\varepsilon}_{\mathcal{T}'}}\left\{\frac{1}{n}\sum_1^n \log\frac{p(X^{(i)};\mathcal{T}',\alpha')}{p(X^{(i)};\mathcal{T},\alpha)} \geq 0\right\}\right).$$

Since $\alpha' \in \mathcal{A}^{\varepsilon}_{\mathcal{T}'}$, the density $p(x;\mathcal{T}',\alpha')$ is bounded away from zero, hence the conditions of Jennrich's [8] uniform strong law of large numbers are satisfied. Thus, $\mathsf{P}_{\mathcal{T},\alpha}$-almost surely,

$$\frac{1}{n}\sum_1^n \log\frac{p(X^{(i)};\mathcal{T}',\alpha')}{p(X^{(i)};\mathcal{T},\alpha)} \longrightarrow -I\left((\mathcal{T},\alpha),(\mathcal{T}',\alpha')\right) \leq -\delta \tag{38}$$

uniformly in $\alpha' \in \mathcal{A}^{\varepsilon}_{\mathcal{T}'}$, whence the RHS of (37) converges to zero as $n \to \infty$. ∎

Proof of Theorem 9: Recall from the proof of Theorem 8 that the Kullback-Leibler information $I(\theta,\theta')$ is a continuous function of θ', and, because of identifiability, has a unique minimum, namely 0, at $\theta' = \theta$. Given any neighborhood U of $\theta \in \Theta$, it follows that, for sufficiently small $\varepsilon > 0$, the set $C_\varepsilon = \{\theta' : I(\theta,\theta') < \varepsilon\}$ is contained in U. Using Theorem 7.80 of Schervish [17], we have, for $n \to \infty$,

$$\pi(U|x) \longrightarrow 1,\ \mathsf{P}_\theta - a.s. \tag{39}$$

Consider the pseudo-Bayes classifier $\widehat{\mathcal{T}}_\pi$, which now takes the form

$$\widehat{\mathcal{T}}_\pi(x) = \arg\max_{\tau\in\mathcal{T}(R)} \pi(\tau \times \mathcal{A}^0_\tau|x). \tag{40}$$

From (39) we obtain that, $\mathsf{P}_{\mathcal{T},\alpha}$ almost surely, $\pi(\{\mathcal{T}\} \times \mathcal{A}^0_{\mathcal{T}}|x) \longrightarrow 1$ and $\pi(\{\tau\} \times A_\tau|x) \longrightarrow 0$ for $\tau \neq \mathcal{T}$, whence $\widehat{\mathcal{T}}_\pi(x) = \mathcal{T}$ for sufficiently large n, $\mathsf{P}_{\mathcal{T},\alpha}$ almost surely. ∎

Proof of Lemma 1: Assume that a number of groupings have been formed, after which k_1, k_2 are candidate nodes descended from i, while k_3 is some other candidate node not descended from i. Since the grouping thus far is correct, k_3 cannot be i or an ancestor of i, and hence $R(k_3) = S_3 \subset R_{\mathcal{T}} \setminus R_{\mathcal{T}}(i)$. Let $S_j = R_{\mathcal{T}}(k_j)$, $j = 1,2$. All the S_j are disjoint. By arguments similar to those used in the proof of Theorem 2, $B(\{k_1,k_3\}) = B(S_1,S_3) > B(S_1,S_2) = B(\{k_1,k_2\})$. Thus correct grouping of k_1, k_2 by BLT is guaranteed if (16) holds for all $(S_1,S_2,S_3) \in \mathcal{S}(i)$. ∎

Proof of Theorem 10: Since for each $S \subseteq R$, $\widehat{\gamma}(S)$ is the mean of i.i.d. random variables $\widehat{Y}^{(i)}_S$, the variables $\sqrt{n}\cdot(\widehat{\gamma}-\gamma)$, $\widehat{\gamma} = \{\widehat{\gamma}(S)\}_{S\subseteq R}$, converge to a multivariate Gaussian random variable as $n \to \infty$. Since $\widehat{D}$ is a differentiable function $\mathcal{D}$ of $\widehat{\gamma}$, the Delta method insures that the stated convergence holds.

To prove (i) observe that since $a(S_1), a(S_3) \prec a(S_1 \cup S_3)$ then $B(S_1,S_3) = A(a(S_1 \cup S_3))$. Since S_1 and S_2 may not satisfy $a(S_1), a(S_2) \prec a(S_1 \cup S_2)$–this may occur whenever there was a grouping error in any of the steps that lead to the construction of node S_1 and/or node S_2–we need to explicitly write the expression for $B(S_1,S_2)$,

$$\begin{aligned} B(S_1,S_2) &= \frac{\mathsf{P}[\vee_{j\in S_1} X_j = 1]\mathsf{P}[\vee_{j\in S_2} X_j = 1]}{\mathsf{P}[\vee_{j\in S_1} X_j \cdot \vee_{j\in S_2} X_j = 1]} \\ &= \mathsf{P}[X_{a(S_1\cup S_2)} = 1] \\ &\times \frac{\mathsf{P}[\vee_{j\in S_1} X_j = 1 | X_{a(S_1\cup S_2)} = 1]}{\mathsf{P}[\vee_{j\in S_1} X_j = 1 | X_{a(S_1\cup S_2)} = 1, \vee_{j\in S_2} X_j = 1]} \\ &= A(a(S_1 \cup S_2))\psi_{(S_1,S_2)} \end{aligned} \tag{41}$$

where $\psi_{(S_1,S_2)} := \frac{\mathsf{P}[\vee_{j\in S_1} X_j = 1 | X_{a(S_1\cup S_2)}=1]}{\mathsf{P}[\vee_{j\in S_1} X_j = 1 | X_{a(S_1\cup S_2)}=1,\ \vee_{j\in S_2} X_j = 1]}$. Observe from Proposition 1(iv) that $\psi_{(S_1,S_2)} \leq 1$. Intuitively, the smaller $\psi_{(S_1,S_2)}$, the greater the error committed so far in classifying the subtree rooted at i. (i) then follows as for $\|\overline{\alpha}\| \to 0$ it is easy to verify that $A(k) = 1 - s(k) + O(\|\overline{\alpha}\|^2)$ and $\chi_{(S_1,S_2)} = 1 - O(\|\overline{\alpha}\|^2)$. To prove (ii), a standard application of the Delta method shows that the collection of $\sqrt{n}(\widehat{B}(S_1,S_2) - B(S_1,S_2))$ converge as $n \to \infty$ to a multivariate Gaussian random variable with mean zero and covariance matrix

$$\nu_{(S_1,S_2),(S_3,S_4)} = \sum_{S,S'\subseteq R} \frac{\partial B(S_1,S_2)}{\partial\gamma(S)}\, C_{S,S'}\, \frac{\partial B(S_3,S_4)}{\partial\gamma(S')}. \tag{42}$$

where $C_{S,S'} = \mathsf{Cov}[\widehat{Y}^{(i)}_S, \widehat{Y}^{(i)}_{S'}]$. Now, following the same lines of Theorem 5 in [3], we have that $C_{S,S'} = s(a(S \cup S')) + O(\|\overline{\alpha}\|^2)$, and $\frac{\partial B(S_1,S_2)}{\partial\gamma(S)} = \delta_{(S_1\cup S_2),S} + O(\|\overline{\alpha}\|^2)$ by direct differentiation. Therefore, we have $\nu_{(S_1,S_2),(S_3,S_4)} = C_{(S_1\cup S_2),(S_3\cup S_4)} + O(\|\overline{\alpha}\|^2)$. Hence,

$$\begin{aligned} \sigma^2(S_1,S_2,S_3) &= \nu_{(S_1,S_3),(S_1,S_3)} + \nu_{(S_1,S_2),(S_1,S_2)} \\ &\quad -2\nu_{(S_1,S_2),(S_1,S_3)} + O(\|\overline{\alpha}\|^2) \end{aligned} \tag{43}$$

$$= \quad s(a(S_1 \cup S_2)) - s(a(S_1 \cup S_3)) + O(\|\overline{\alpha}\|^2)$$

Finally, (iii) follows as $s(a(S_1 \cup S_2)) - s(a(S_1 \cup S_3))$ is minimized when $a(S_1 \cup S_2) = i$ and $a(S_1 \cup S_3) = f(i)$. ■

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, 38(5), 152–159, May 2000.

[2] J. Berger, *Statistical Decision Theory and Bayesian Analysis*, 2nd ed., Springer, 1985.

[3] R. Caceres, N.G. Duffield, J.Horowitz and D. Towsley, "Multicast-Based Inference of Network Internal Loss Characteristics", IEEE Trans. on Information Theory, 45: 2462–2480, 1999.

[4] R. Caceres, N.G. Duffield, J.Horowitz F. Lo Presti and D. Towsley, "Statistical Inference of Multicast Network Topology", in Proc. 1999 IEEE Conf. on Decision and Control, Phoenix, AZ, Dec. 1999.

[5] A. Dembo and O. Zeitouni, *Large Deviation Techniques and Applications.* Jones and Bartlett, Boston-London, 1993.

[6] N.G. Duffield and F. Lo Presti, "Multicast Inference of Packet Delay Variance at Interior Network Links", in Proc. IEEE Infocom 2000, Tel Aviv, March 2000.

[7] M. Handley, S. Floyd, B. Whetten, R. Kermode, L. Vicisano, M. Luby. "The Reliable Multicast Design Space for Bulk Data Transfer," RFC 2887, Internet Engineering Task Force, Aug. 2000.

[8] R. Jennrich, "Asymptotic properties of nonlinear least-squares estimators", *Ann. Math. Stat.* 40:633–643, 1969.

[9] B.N. Levine, David Lavo , and J.J. Garcia-Luna-Aceves, "The Case for Concurrent Reliable Multicasting Using Shared Ack Trees," Proc. ACM Multimedia Boston, MA, November 18–22, 1996.

[10] B.N. Levine, S. Paul, J.J. Garcia-Luna-Aceves, "Organizing multicast receivers deterministically according to packet-loss correlation," Proc. ACM Multimedia 98, Bristol, UK, September 1998.

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

[13] `ns` – Network Simulator. See http://www-mash.cs.berkeley.edu/ns/ns.html

[14] 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.

[15] M.J.D. Powell, "Gradient conditions and Lagrange multipliers in nonlinear programming". Lecture 9 in L.C.W. Dixon, E. Spedicato, G.P. Szegö (eds.), "Nonlinear optimization: theory and algorithms", Birkhäuser, 1980, p. 210

[16] S. Ratnasamy & S. McCanne, "Inference of Multicast Routing Tree Topologies and Bottleneck Bandwidths using End-to-end Measurements", in Proc. IEEE Infocom'99, New York, March 1999

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

[18] R.J. Vanderbei and J. Iannone, "An EM approach to OD matrix estimation," Technical Report, Princeton University, 1994

[19] Y. Vardi, "Network Tomography: estimating source-destination traffic intensities from link data," J. Am. Statist. Assoc., 91: 365–377, 1996.

[20] B. Whetten, G. Taskale. "Reliable Multicast Transport Protocol II," IEEE Networks, 14(1), 37–47, Jan./Feb. 2000.