INFERENCE OF LINK DELAY THROUGH MEASUREMENT REDUNDANCY IN COMMUNICATION NETWORKS

by

Ye Xia and David Tse

Memorandum No. UCB/ERL M00/57

3 November 2000

INFERENCE OF LINK DELAY THROUGH MEASUREMENT REDUNDANCY IN COMMUNICATION NETWORKS

by

Ye Xia and David Tse

Memorandum No. UCB/ERL M00/57

3 November 2000

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Inference of Link Delay through Measurement Redundancy in Communication Networks

Ye Xia* David Tse

Department of Electrical Engineering and Computer Science
University of California, Berkeley

Abstract

This paper studies the feasibility and algorithms for inferring delay at each link in a communication network based on a large number of end-to-end measurements. The restriction is that we are not allowed to measure directly on each link and can only observe the total delays on one or more network routes that include that link. It is assumed that we have considerable flexibility in choosing which route to measure. We investigate three different cases: (1) each link delay is a constant; (2) each link delay is modeled as a random variable from a family of distributions with unknown parameters; and (3) each link delay is a random variable whose distribution is completely unknown. We will answer whether such indirect measurement is possible at all, and when possible, how such measurement can be carried out.

keywords network delay measurement

*This project is funded by NSF Award #9872764.

1 Introduction

1.1 Motivation

As the internet grows, it becomes increasingly important to monitor the network performance and identify failures. These must be based on the knowledge of the complete or partial topology of the network and on the measurement of some quantities of the network. In this paper, we study the issue of network measurement, given that the network topology is completely known. The useful information one might want to know about the network includes the delay, packet loss ratio, link capacity and throughput, etc. One might be interested in the end-to-end information on some route or localized information at some routers or links. For information with randomness, it is also possible to distinguish its long-time average and instantaneous value. In building a network measurement infrastructure, there is a choice about the measurement locations, where the measurement software or hardware are placed. We can think of placing them (1) in every network router and end-hosts; (2) only in the end-hosts at the edge of the network; and (3) in selected routers and end-hosts. Here are the constraints. First, it might never be possible to have a consistent measurement infrastructure throughout the entire internet which is partitioned into a disparate array of administrative domains. Second, it is costly and practically difficult to install, maintain, and upgrade the measurement related software and hardware in a large number of routers and computers. One naturally wonders how to reduce the number of measurement locations and still be able to observe the entire network by creating redundancy in the number of routes or in the number of observations.

This paper addresses the choice of measurement locations, in particular the feasibility of measuring every link of the entire network from a subset of the network nodes. The object of measurement, which we call the link attribute, is additive in the sense that the combined attribute on multiple links is the sum of individual link attributes. The instantaneous delay, the average delay, and the amount of packet loss are all additive. Loss ratio and throughput are examples of non-additive link attributes. We will consider delay as the prototype of an additive link attribute.

We will specifically investigate a special case of choice (2) above. The measurement agents are at selected end-hosts at the edge of the network. The following is a typical question we would like to answer. Suppose we do not have a direct measurement of the average delay on a particular link, but we do have the end-to-end delay measurement on many routes that pass through that link.

Can we, then, calculate the link delay based on the end-to-end measurement?

1.2 Previous research

Our work is motivated by [1], which studies how to infer packet loss ratios on individual links in a multicast tree based on the observed loss statistics on end-to-end routes. Our work is independently done from a few other works ([3] and [5]) on the same subject of inferring link delays based on end-to-end delays. As far as we know, the deterministic analysis of section 2 has no counterpart in other works. In the study about parametric delay models in section 3, the authoritative reference on the general EM algorithm is [2]. In [8], Vardi applies the EM algorithm in a similar network setting. There, he uses a Poisson model and we use exponential and mixture of exponential models. The conclusion that Gaussian model is not identifiable is new (Theorem 3.1). In the same section, the method of moment is similar to [3], where the authors use moment method to estimate link delay variances.

The study on non-parametric delay model in section 4 has deep relation with [5]. In terms of style, we take an intuitive approach and [5] is more formal and more systematic. Both have reached the conclusion that, in order to be able to infer the link delay distribution, one needs to make an assumption that there is a non-zero probability that each link delay is zero. We arrive at this conclusion as a corollary of our deterministic study in section 2. In fact, we showed that this is almost a necessary condition for estimating link delays based on route delays. Our intuitive approach allows us to examine the applicability of our sampling technique in terms of the number of samples needed. We think our estimation technique is an intuitive variant of the estimator in [5].

Finally, [1], [3] and [5] all spend considerable effort in showing the "goodness" of their estimators. They certainly suggest one of the future directions for our work.

2 Deterministic Case

In this section, we consider link attributes as constant quantities, and hence, are subject to deterministic analysis. We will study the possibility of determining the link attribute through end-to-end measurement.

The deterministic analysis is motivated by the consideration of non-parametric probability

model of link delays, which will be discussed in detail later. We give two motivating examples here. In the first example, we would like to determine the empirical distribution of each link delay through end-to-end measurement. For each measured end-to-end delay sample, we normally need to determine the delay due to each link on that end-to-end route. In a slightly different example, suppose we have the average delay on a end-to-end route and would like to find the average delay of each link on the end-to-end route. In the following, we will state the problem more formally. First, we will introduce some graph theory terminology.

2.1 Some standard graph theory definitions

A directed graph $\mathbf{G} = [\mathbf{A}, \mathbf{U}]$ is defined to be a set $\mathbf{A}$ whose elements are called *nodes* (or *vertices*) and a set $\mathbf{U}$ whose elements $u \in \mathbf{U}$ are called *arcs* between a pair of vertices.

A *p-graph* is a directed graph with no more than p arcs between any two nodes a and b in that order. If $\mathbf{G}$ is a 1-graph, an element in $\mathbf{U}$ is represented as the ordered pair of nodes, such as (a, b).

Suppose $\mathbf{G}$ is a directed graph. A *walk* of length q is a sequence of q arcs, $P = \{u_1, ..., u_q\}$, where $u_1, ..., u_q \in \mathbf{U}$, and $u_1 = (a_0, a_1), u_2 = (a_1, a_2), ..., u_q = (a_{q-1}, a_q)$. A *path* is a walk in which no node appears more than once. A *circuit* is a closed walk whose initial and final nodes coincide, and in which no other nodes are visited more than once.

The *distance* from node a_1 to node a_2 is the length of the shortest path from a_1 to a_2.

For each $a \in \mathbf{A}$, define $d_G^+(a)$ to be the number of arcs starting from a, and $d_G^-(a)$ to be the number of arcs ending at a.

A non-directed graph, or simply a *graph*, $\mathbf{G}$, is a pair $[\mathbf{A}, \mathbf{U}]$, where $\mathbf{A}$ is the set of nodes and $\mathbf{U}$ is a set of *edges*. An edge has two endpoints, $a \in \mathbf{A}$ and $b \in \mathbf{A}$. A *chain* is a sequence of edges, $u_1, u_2, ..., u_n$, such that the terminal node of u_r is the initial node of u_{r+1}, for $r = 1, 2, ..., n-1$. A *cycle* is a chain of edges which starts and ends at the same node. A graph is *connected* if for every $a \in \mathbf{A}$, $b \in \mathbf{A}$ and $a \neq b$, there is a chain starting from node a and ending at node b. A subgraph of $\mathbf{G}$ generated by $B \subseteq A$ is a graph $\mathbf{G_B} = [B, U_B]$, where U_B is the set of edges $u \in \mathbf{U}$ such that the two endpoints of u are in B. A *connected component* of the graph $\mathbf{G}$ is a maximal connected subgraph $\mathbf{G}$ in the sense that it is not contained in any other connected subgraph of $\mathbf{G}$.

2.2 Statement of the problem

We represent a network as a directed graph $\mathbf{G} = [\mathbf{A}, \mathbf{U}]$, where $\mathbf{A}$ is the set of routers and end-hosts, and $\mathbf{U}$ is the set of communication links (directed, i.e., arcs) between elements of $\mathbf{A}$. Note that a physical link in a network is represented by two arcs going in opposite directions in a directed graph representation. The reason is that we allow the attributes, such as delays, on the two directions of the physical link to be different. Let $A_s \subseteq \mathbf{A}$ be the set of end-hosts which are designated to send packets, called *senders*, and $A_r \subseteq \mathbf{A}$ be the set of end-hosts which are designated to receive packets, called *receivers*. We call the elements in A_s *sender nodes*, the elements in A_r *receiver nodes*, and, together, they are called *measurement nodes*. Any node a with $d_G^+(a) \geq 1$ and $d_G^-(a) \geq 1$ is called a *router node*. Because a router node has at least one incoming link and at least one outgoing link, it is capable of forwarding packets from an incoming link to an outgoing link. Let us associate with each link, $u \in \mathbf{U}$, a constant x_u, which represents a fixed attribute of the link u, such as the expected delay. Assuming the attributes are additive when multiple links are involved, our objective is to study the feasibility of determining each x_u, for $u \in \mathbf{U}$, when the accumulated attributes from the senders to the receivers on all possible walks are given. An example is that we want to know whether the expected delay on each link can be recovered simply by sending packets at the senders and observing the received packets at the receivers, assuming a packet accumulates the delays as it traverses links on the walk from a sender to a receiver.

Definition 2.1 *A route from $s \in A_s$ to $r \in A_r$ is a walk on the directed graph $\mathbf{G} = [\mathbf{A}, \mathbf{U}]$ in which any circuit appears at most once. The route attribute is the accumulated attributes of the links on the route. (If a link appears on a route n times, its link attribute is accumulated n times.)*

Lemma 2.1 *Given a finite directed graph $\mathbf{G} = [\mathbf{A}, \mathbf{U}]$, A_s and A_r, the total number of routes is finite.*

Proof: Let N be the number of nodes in $\mathbf{G}$. A circuit can traverse at most N distinct nodes. The number of circuits which traverses $K \leq N$ distinct nodes is bounded by $N \times (N-1) \times ... \times (N-K+1)$ Therefore, the total number of circuits is finite. The total number of paths of length $K \leq N$ is bounded by $N \times (N-1) \times ... \times (N-K)$. So, the total number of paths is also finite. Notice each route consists of a path with a finite number of nodes. At each node, the possible number of distinct circuits is also finite. Hence, the total number of routes must be finite. ■

Although walks from a sender to a receiver in which some circuits appear more than once are not considered as routes, the definition of the route is not restrictive for the problem we are considering, because the accumulation of the link attributes around a circuit, called the circuit attribute, can be observed. The accumulated attributes on two walks which differ only in the number of times a circuit is traversed differ by a known constant.

Suppose the links in the directed graph $\mathbf{G}$ are labeled as $1, 2, ..., L$, where L is the total number of links. Suppose the routes are also indexed from 1 to M, where M is the number of routes. Let the vector $\mathbf{x} = (x_1, x_2, ..., x_L)^T$ in $\mathbb{R}^L$ be the link attributes, and let $\mathbf{y} = (y_1, y_2, ..., y_M)^T$ in $\mathbb{R}^M$ be the route attributes observed on each route, and let $r(i,j)$ be the number of times link j appears on route i. Let $U_i \subseteq \mathbf{U}$ be the set of links on route i. Then

$$y_i = \sum_{j \in U_i} r(i,j) x_j \tag{1}$$

Let $\mathbf{R}$ be the $M \times L$ matrix whose (i,j)-th entry is $r(i,j)$, which is called the *route matrix* for $\mathbf{G}$, we can write equation (1) in the matrix format.

$$\mathbf{Rx} = \mathbf{y} \tag{2}$$

Definition 2.2 *The directed graph* $\mathbf{G}$ *is identifiable if equation (2) has an unique solution; otherwise, we say* $\mathbf{G}$ *is unidentifiable.* $\mathbf{G}$ *is strongly unidentifiable if no component of* $\mathbf{x}$ *is uniquely determined by (2).*

Proposition 2.2 *Suppose, for any* $a_1 \in A_s \cup A_r$ *and* $a_2 \in A_s \cup A_r$, *the distance from* a_1 *to* a_2 *is at least* 2. *Then* $\mathbf{G}$ *is strongly unidentifiable.*

The condition for Proposition 2.2 says that no two measurement nodes are adjacent, i.e., are connected by a link. The focus of the deterministic analysis is to prove Proposition 2.2 and to point out its consequences. One necessary condition for solving the set of linear equations in (2) uniquely is to have $M \geq L$. We therefore assume this to be true. Essentially, we assume the number of routes is greater than the number of links. We will postpone the proof of Proposition 2.2.

Corollary 2.3 *If at least one link in* $\mathbf{G}$ *does not directly connect two measurement nodes, then* $\mathbf{G}$ *is unidentifiable.*

Proof: Remove all links which directly connect any pair of measurement nodes. Then, delete nodes which have no links attached. Links which do not connect two measurement nodes directly together with the nodes to which they are attached survive the deletion process. Hence, the resulting directed graph has at least one connected component, in which all measurement nodes are separated by a minimum distance of 2, hence, is unidentifiable. ■

Suppose each measurement node is a measuring site, where measurement software or hardware need to be placed. By the above proposition and corollary, the only way to uniquely determine the attribute of a particular link is to measure it at the two nodes directly attached to it. All nodes are required to become measurement sites in order to monitor the complete network. It is not possible to monitor the network from the edge, relying on the redundancy of routes.

We will first make the some assumptions about the directed graph $\mathbf{G}$ which do not reduce the generality of our results but simplify the exposition.

Definition 2.3 *A node or a link in* $\mathbf{G}$ *is reachable if it is on at least one route.*

Assumption 2.1 *(1) The directed graph* $\mathbf{G}$ *is connected when viewed as an undirected graph. (2) Every link is reachable.*

In the case where $\mathbf{G}$ is not connected as a graph, we can look at each connected component separately. When a link is not reachable, it certainly cannot be identified, and it won't affect the identifiability of other links. Hence, we can remove it from $\mathbf{G}$. A consequence of the assumptions is that every node is reachable. For, if a node is not reachable, none of the links connected to it is reachable. Also, every node which is not a measurement node must be a router node. Otherwise, all the links connected to it are not on any routes.

2.3 Symmetric networks

First, certain special networks are easily identifiable. We say a network is symmetric if its directed graph representation $\mathbf{G} = [\mathbf{A}, \mathbf{U}]$ has the property that all links between any two nodes $a \in \mathbf{A}$ and $b \in \mathbf{A}$ have the same link attribute. The symmetry is observed in that the link from a to b has the same attribute as the link from b to a. Under Assumption 2.1, we have

Proposition 2.4 *If a pair of nodes, a and b, are connected by at least one link from a to b and at least one link from b to a, then the attribute of these links can be uniquely determined.*

Proof: Setup any route, l_1, that passes through a link starting from a and terminating at b. Then, l_1 plus circuit starting from a, to b, and back to a can also yield a route. Call it l_2. From the difference of route attributes of l_2 and l_1, we get the attribute of the circuit above, which is twice the link attribute for the links that connect a and b. ■

Corollary 2.5 *Suppose* $\mathbf{G}$ *has the property that, for every* $a \in \mathbf{A}$ *and* $b \in \mathbf{A}$*, if there is a link from* a *to* b*, then there is a link from* b *to* a*. Then,* $\mathbf{G}$ *is identifiable.*

Proof: The attribute of every link can be uniquely determined by the previous proposition. ■

In reality, communication links in the networks are almost always full duplex. Even though the attributes of forward and backward links are usually not identical, by the proof of the proposition, the combined attributes of the forward and backward links can be uniquely determined between any two adjacent nodes. For purposes such as fault discovery, this can be adequate.

2.4 Strong unidentifiability of asymmetric networks

In this section, we will prove Proposition 2.2. Again, we assume that $\mathbf{G}$ satisfies Assumption 2.1.

Let us first get familiar with the matrix $\mathbf{R}$ in equation (2). It has the following simple properties.

- Each entry of $\mathbf{R}$ is either 0 or a positive integer.
- Each row of $\mathbf{R}$ (say i) corresponds to a route (i) and each column (say j) corresponds to a link (j). The $(i,j)^{th}$ entry of $\mathbf{R}$ is the number of times link j appears on route i.

The dimension of matrix $\mathbf{R}$ is $M \times L$, where M is equal to the number of routes, and L is the number of links. Let the rows of $\mathbf{R}$ be vectors $\mathbf{v_1}^T, \mathbf{v_2}^T, ..., \mathbf{v_M}^T$, and let the columns of $\mathbf{R}$ be vectors $\mathbf{w_1}, \mathbf{w_2}, ..., \mathbf{w_L}$. For each $i \in \{1, 2, ..., L\}$, define vectors $\mathbf{e_i} \in \mathbb{R}^L$ to be of the form $(0, ..., 0, 1, 0, ..., 0)^T$ with 1 in the i^{th} location.

Lemma 2.6 *Every column vector of* $\mathbf{R}$ *can be expressed as a linear combination of other column vectors.*

Proof: Let us fix i, for $i \in \{1, 2, ..., L\}$. The i^{th} column vector of $\mathbf{R}$ corresponds to link i in the directed graph $\mathbf{G} = [\mathbf{A}, \mathbf{U}]$. Let $a \in \mathbf{A}$ and $b \in \mathbf{A}$ be the two nodes to which link i is connected. Since all measurement nodes are separated by a distance of at least 2, at least one of these nodes,

say a, is not a measurement node. Then a must be a router node by the assumption that any link including link i must be on at least one route. Let $U_a^+ \subseteq \mathbf{U}$ be the set of outgoing links starting from node a, and $U_a^- \subseteq \mathbf{U}$ be the set of incoming links to node a. Then we have the following identity.

$$\sum_{j \in U_a^-} \mathbf{w_j} = \sum_{j \in U_a^+} \mathbf{w_j} \tag{3}$$

To show this, let us fix a k for $k \in \{1, 2, ..., M\}$. $\sum_{j \in U_a^-} \mathbf{w_j}(k)$ is the number of times the k^{th} route, numbered by the rows of matrix $\mathbf{R}$, enters node a. Similarly, $\sum_{j \in U_a^+} \mathbf{w_j}(k)$ is the number of times the k^{th} route leaves node a. Since a is a router node and is not a measurement node, any route which enters it must leave it. Hence, we get the equality of (3). Since link i is either in U_a^+ or in U_a^-, by re-arranging equation (3), we get,

$$\mathbf{w_i} = \begin{cases} \sum_{j \in U_a^+} \mathbf{w_j} - \sum_{j \in U_a^-, j \neq i} \mathbf{w_j} & \text{if } \mathrm{i} \in \mathrm{U_a^-} \\ \\ \sum_{j \in U_a^-} \mathbf{w_j} - \sum_{j \in U_a^+, j \neq i} \mathbf{w_j} & \text{if } \mathrm{i} \in \mathrm{U_a^+} \end{cases} \tag{4}$$

■

Corollary 2.7 $\mathbf{G}$ *is unidentifiable.*

Proof: By Lemma 2.6, matrix $\mathbf{R}$ does not have full rank. ■

Lemma 2.8 *For every* $i \in \{1, 2, ..., L\}$, $\mathbf{e_i} \notin span\{\mathbf{v_1}, \mathbf{v_2}, ..., \mathbf{v_M}\}$.

Proof: We need to show that there is not a non-zero row vector $\lambda = (\lambda_1, \lambda_2, ..., \lambda_M)$ such that $\mathbf{e_i} = \sum_{j=1}^{M} \lambda_j \mathbf{v_j}$. Note that $\mathbf{e_i}^T = \lambda \mathbf{R} = (\lambda \mathbf{w_1}, \lambda \mathbf{w_2}, ..., \lambda \mathbf{w_L})$. Since all components of $\mathbf{e_i}$ are 0's except that the i^{th} component is a 1, it is sufficient to show that, for any λ, $\lambda \mathbf{w_j} = 0$ for every $j \neq i$ necessarily implies $\lambda \mathbf{w_i} = 0$. By Lemma 2.6, we can write $\mathbf{w_i} = \sum_{j=1, j \neq i}^{L} \alpha_j \mathbf{w_j}$, for some $\alpha_j \in \mathbb{R}$. Then,

$$\lambda \mathbf{w_i} = \lambda \sum_{j=1, j \neq i}^{L} \alpha_j \mathbf{w_j} = \sum_{j=1, j \neq i}^{L} \alpha_j \lambda \mathbf{w_j} = 0$$

■

Finally, we are in a position to show strong unidentifiability.

Proof: (of Proposition 2.2) Let us view $\mathbf{R}$ as a linear transformation from $\mathbb{R}^L$ to $\mathbb{R}^M$. It is sufficient to show that, for each $i = 1, 2, ..., L$, there exists a vector $\mathbf{z_i} \in$ *kernel* $\mathbf{R}$ such that $\mathbf{z_i}^T \mathbf{e_i} \neq 0$. This is enough because if $\mathbf{z_i}^T \mathbf{e_i} \neq 0$ for a fixed $i \in \{1, 2, ..., L\}$, then the i^{th} entry of $\mathbf{z_i}$

must be non-zero. Then, if the vector $\hat{\mathbf{x}}$ is a solution to equation (2), $\hat{\mathbf{x}} + c\mathbf{z_i}$ is also a solution, for any real constant c. In particular, the i^{th} entries for all these solutions are different. That is, the delay of link i cannot be determined uniquely. Letting i vary, we can conclude the directed graph $\mathbf{G}$ is strongly unidentifiable.

Let us call the span of $\{\mathbf{v_1}, \mathbf{v_2}, ..., \mathbf{v_n}\}$ the row space of the linear transformation $\mathbf{R}$. Then, it is a well known fact that *kernel* $\mathbf{R}$ and *row space* $\mathbf{R}$ are orthogonal complement of each other. In other words, for every $\mathbf{v}$ such that $\mathbf{v} \perp$ *kernel* $\mathbf{R}$, it must be true that $\mathbf{v} \in$ *row space* $\mathbf{R}$ (See page 138 of [7].

By Lemma 2.8, $\mathbf{e_i} \notin$ *row space* $\mathbf{R}$, for every $i \in \{1, 2, ..., L\}$. Therefore, for each i, there exist a vector $\mathbf{z_i} \in$ *kernel* $\mathbf{R}$, such that $\mathbf{z_i}^T\mathbf{e_i} \neq 0$. Otherwise, $\mathbf{e_i}$ would be in *row space* $\mathbf{R}$. ■

2.5 Conclusion of deterministic analysis

The results of this section are based on deterministic analysis. In summary, the deterministic and additive attribute of a link in an asymmetric network cannot be observed unless we measure it directly from the two nodes it is attached to. We loosely say the network cannot be completely observed from a proper subset of the nodes.

Link attributes such as delay are often non-negative. In Proposition 2.2, we did not impose the constraint of non-negativity. If such a constraint is imposed, and if all link attributes are strictly positive, Proposition 2.2 is still valid, since the vector of link attributes has an open neighborhood in $\mathbb{R}^L_{\geq 0}$, where $\mathbb{R}_{\geq 0}$ is the set of non-negative real number. If some link attributes are zero, then it is possible that all link attributes can be uniquely determined. This last point makes determining a non-parametric model possible, as we will show in section 4.

3 Probabilistic Case - Parametric Models

Many link attributes, such as delay, has randomness, which, on one hand, alters the objectives we can pursue, and on the other hand, adds a different set of information that we can utilize. It is entirely possible that the network can be observed from a subset of the nodes. In this section, we will study the case where the link attributes can be specified by some parametric probability models. The objective is to recover the unknown parameters of the model through statistical inference.

A parametric probability model is one whose distribution has known functional form with some parameters. Its distribution is completely specified if the parameters are known. We will see that parametric models pose certain difficulties to our problem. First, since a parametric model is parsimoniously determined by a few parameters, we are somewhat obliged to determine all its parameters. This can be analytically difficult for additive link attributes. Second, by stipulating that the link attribute follows a parametric model, we have made a very strong assumption. Even if we can estimate the parameters correctly and efficiently, the contribution may not be great if the model is invalid when compared with the reality.

Throughout the discussion of the probabilistic models, including both the parametric and non-parametric models, we assume all link attributes are independent of each other at all time instances, and attributes of the same link at different time instances are independent.

3.1 Gaussian model

Let $\mathbf{G} = [\mathbf{A}, \mathbf{U}]$ be the directed graph satisfies assumption 2.1. Suppose there are total L links in $\mathbf{G}$ numbered as $1, 2, ..., L$, and there are total M routes, numbered as $1, 2, ..., M$. For each link i, let its link attribute X_i to be a random variable with Gaussian distribution with mean μ_i and variance σ_i^2. Since $X_1, X_2, ..., X_L$ are independent, they are multivariate Gaussian with mean vector $\mu = (\mu_1, \mu_2, ..., \mu_L)^T$ and covariance matrix $\mathbf{\Sigma} = diag(\sigma_1^2, \sigma_2^2, ..., \sigma_L^2)$. Let $\mathbf{R}$ be the route matrix associated with $\mathbf{G}$, let $\mathbf{X} = (X_1, X_2, ..., X_L)$, and let $\mathbf{Y} = (Y_1, Y_2, ..., Y_M)$ be the route attributes on route $1, 2, ..., M$. Since $\mathbf{Y} = \mathbf{RX}$, $\mathbf{Y}$ is also multivariate Gaussian with the mean vector $\mathbf{R}\mu$ and covariance matrix $\mathbf{\Sigma_Y} = \mathbf{R\Sigma R}^T$.

Denote P_θ the multivariate Gaussian distribution for the random vector $\mathbf{Y}$ with parameter $\theta \in \Theta$, where Θ is the parameter space. In this case, $\theta = (\mathbf{R}\mu,\ \mathbf{\Sigma_Y})$.

Definition 3.1 *A parametric model is identifiable if $\theta_1 \neq \theta_2$ implies $P_{\theta_1} \neq P_{\theta_2}$, for all $\theta_1,\ \theta_2 \in \Theta$. Otherwise, we say it is unidentifiable.*

Then, we have the following theorem.

Theorem 3.1 *If no measurement nodes are adjacent in the directed graph $\mathbf{G}$, the Gaussian model P_θ specified above is unidentifiable.*

Proof: Let $K = rank\ \mathbf{R}$. By Proposition 2.2, $K < L$. This implies we can select K routes so that any route vector (i.e., a row in $\mathbf{R}$) is a linear combination of the selected K routes. Therefore, without the loss of generality, we can assume $\mathbf{R}$ is a $K \times L$ route matrix with full row rank. From $\mathbf{Y} = \mathbf{RX}$, $\mathbf{Y}$ is a multivariate Gaussian with mean $\mathbf{R}\mu$ and covariance matrix $\mathbf{\Sigma_Y} = \mathbf{R\Sigma R}^T$. $\mathbf{\Sigma_Y}$ is invertible. The distribution of $\mathbf{Y}$ is completely determined by its mean vector and the covariance matrix. It depends on μ through $\mathbf{R}\mu$. Since $kernel\ \mathbf{R} \neq \{0\}$, there exist vectors $\mu^1 \neq \mu^2$ such that $\mathbf{R}\mu^1 = \mathbf{R}\mu^2$. ■

Theorem 3.1 is a rather disappointing result. Multivariate Gaussian random variables have some nice properties which make many calculations simpler. For instance, the joint distribution is completely specified by the mean and the covariance matrix. A linear transformation of the multivariate Gaussian random variables is also Gaussian. The mean and covariance of the transformed Gaussian can be obtained by linear transformations on the original mean and covariance. These nice properties works against us in our case, where the linear transformation by $\mathbf{R}$ is not injective.

It might be worth pointing out that the variances of link attributes can be measured in certain networks, as will be shown in section 3.2.2.

3.2 Exponential model

The exponential model is a commonly assumed model for the distributions of link delays. Suppose that the link attribute, X_i, of link i is characterized by an exponential distribution with parameter λ_i, denoted by $X_i \sim \exp(\lambda_i)$, for each $i \in \{1, 2, ..., L\}$, and $X_1, X_2, ..., X_L$ are independent. Since the parameters λ_i's are from the space of the positive real numbers, and there are a finite number of links in question, we will assume that all λ_i's are distinct. We will first investigate the identifiability issue of a single route.

Lemma 3.2 *The distribution of any route attribute is identifiable up to ordering.*

Proof: Without loss of generality, let us assume the route in question, denoted by r, has I links, $1, 2, ..., I$. The link attributes, $X_1, X_2, ..., X_I$, are independent exponential random variables with parameter $\lambda_1, \lambda_2, ..., \lambda_I$. Let Y_r be the route attribute, i.e., $Y_r = \sum_{i=1}^{I} X_i$. Since the moment generating function for an exponential random variable with parameter λ_i is $\frac{\lambda_i}{\lambda_i - s}$, the moment

generating function for Y_r is,

$$G(s) = \prod_{i=1}^{I} \frac{\lambda_i}{\lambda_i - s} \tag{5}$$

If the set $\lambda^1 = \{\lambda_1^1, \lambda_2^1, ..., \lambda_I^1\}$ is not the same as the set $\lambda^2 = \{\lambda_1^2, \lambda_2^2, ..., \lambda_I^2\}$, the resulting moment generating functions will be different, also. Since we can find the unique distribution function corresponding to the moment generating function of the form in equantion (5), by the definition of identifiability, the distribution of any route attribute is identifiable if we ignore the order of the I links in the route. ■

Suppose it is possible to estimate the λ_i's correctly on all routes. It is not difficult to determine the correspondence between the links and the parameters λ_i's on some directed graphs. Consider the graph of a binary tree in figure 1 with 1 sender node at the root, 8 receiver nodes at the leaves, 7 intermediate router nodes, and 15 links. There are 8 routes, each associated with one of the receivers. Number the routes $1, 2, ..., 8$ from left to the right. Suppose the estimates of $\lambda_1, \lambda_2, ..., \lambda_{15}$ based on the observations on each route are exact so that we can consider the estimates and the true parameter values are equivalent. Since the estimation based on route 1 yields $\{\lambda_1, \lambda_2, \lambda_4, \lambda_8\}$ and the estimation based on route 2 yields $\{\lambda_1, \lambda_2, \lambda_4, \lambda_9\}$, we can conclude that λ_8 must be associated with X_8 and that λ_9 must be associated with X_9. In this fashion, we can find the parameters λ_i's for all links at the bottom level. Recursively, we can move one level up in the graph and identify the parameters for X_4, X_5, X_6 and X_7. For example, consider route 1 which is associated with the parameter set $\{\lambda_1, \lambda_2, \lambda_4, \lambda_8\}$, and route 3, which is associated with the parameter set $\{\lambda_1, \lambda_2, \lambda_5, \lambda_{10}\}$. Since we have identified X_8 with λ_8 and X_{10} with λ_{10} when we consider the bottom level, we can conclude that λ_4 must be associated with X_4 and that λ_5 must be associated with X_5.

The above reasoning leads to the following lemma.

Lemma 3.3 *For an exponential model in which all parameters λ_i's are distinct, it is possible to construct an identifiable directed graph in which the measurement nodes are not necessarily adjacent.*

This is in contrast with both the deterministic case and the Gaussian model.

In order to estimate the parameters of link attributes based on the observations of the route attribute, it is natural to consider a maximum-likelihood estimator. However, maximizing the likelihood of the sum of independent exponentials is difficult. Consider the following example.

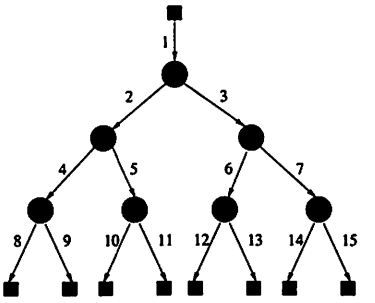

Figure 1: Binary Tree

Let the random variable Y be the sum of three independent exponential random variables with parameters λ_1, λ_2 and λ_3. Suppose we make N independent observations of Y and let $Y^{(n)}$ denote the n^{th} observation. Let the random vector $\mathbf{Y} = (Y^{(1)}, Y^{(2)}, ..., Y^{(N)})$. Let $P_{\mathbf{Y}}(y^{(1)}, y^{(2)}, ..., y^{(N)} \mid \lambda)$ denote the density of $\mathbf{Y}$, where $\lambda = \{\lambda_1, \lambda_2, \lambda_3\}$. $P_{\mathbf{Y}}$ is also known as the likelihood function. After some algebra, we get,

$$\begin{aligned} P_{\mathbf{Y}}(y^{(1)}, y^{(2)}, ..., y^{(N)} \mid \lambda) &= \prod_{n=1}^{N} \{ \frac{\lambda_1\lambda_2\lambda_3}{(\lambda_2 - \lambda_1)(\lambda_2 - \lambda_1)} e^{-\lambda_1 y^{(n)}} \\ &+ \frac{\lambda_1\lambda_2\lambda_3}{(\lambda_1 - \lambda_2)(\lambda_3 - \lambda_2)} e^{-\lambda_2 y^{(n)}} + \frac{\lambda_1\lambda_2\lambda_3}{(\lambda_1 - \lambda_3)(\lambda_2 - \lambda_3)} e^{-\lambda_3 y^{(n)}} \} \end{aligned}$$

We want to obtain the parameters λ that maximize the likelihood function $P_{\mathbf{Y}}$ or the *log* likelihood function. However, we know no easy way to directly apply the traditional optimization techniques by taking the derivatives with respect to the parameters. The difficulty is due to the sum in the expression of the likelihood function.

There are two solutions to our dilemma. The first one is an iterative technique, called the EM algorithm, which is widely used in the maximum likelihood based parameter estimation with incomplete data. The second approach is to deviate from the strict maximum likelihood based estimation and use the method of moments. We will discussion each of these two techniques in the following.

3.2.1 EM algorithm

Let us consider a single route with I links, numbered as $1, 2, ..., I$. Let the link delay to be $X_i \sim exp(\lambda_i)$ for link i, $i = 1, 2, ..., I$, and the X_i's are independent. We assume the parameters λ_i's are different. Let the route delay to be Y. Then, $Y = \sum_{i=1}^{I} X_i$. Suppose N independent samples of Y

are observed, denoted by $\{Y^{(1)}, Y^{(2)}, ..., Y^{(N)}\}$. At each time n, the delay of link i is denoted $X_i^{(n)}$. Let $\lambda = (\lambda_1, \lambda_2, ..., \lambda_I)$, $\mathbf{Y} = (Y^{(1)}, Y^{(2)}, ..., Y^{(N)})$ and $\mathbf{X} = (X_i^{(n)}, i = 1, 2, ..., I, n = 1, 2, ..., N)$.

The EM algorithm is an iterative algorithm for finding the parameters λ that maximize the log likelihood, denoted $l_\lambda(\mathbf{Y}) = \log P_\mathbf{Y}(\mathbf{Y} \mid \lambda)$ [2]. Notice that the random variables $\mathbf{Y}$ are observed. The random variables $\mathbf{X}$ are not observed, and are called hidden random variables. In our case, if $\mathbf{X}$ were observed, then $\mathbf{Y}$ would be completely determined. At each step of the iteration, we have a current estimate of the parameters, denoted by $\lambda^{(t)}$, where t stands for the t^{th} iteration. At the t^{th} iteration, let the conditional density of the hidden variables $\mathbf{X}$ conditional on the observed random variables $\mathbf{Y}$ be $P_{\mathbf{X}|\mathbf{Y}}(\mathbf{X} \mid \mathbf{Y}; \lambda^{(t)})$.

The E-step of the EM algorithm is to write the *complete log likelihood* (of all random variables) with a generic parameters λ, under the assumption that the hidden variables are also observed. Then the conditional expectation of the complete log likelihood is evaluated with respect to the conditional density $P_{\mathbf{X}|\mathbf{Y}}(\mathbf{X} \mid \mathbf{Y}; \lambda^{(t)})$. The M-step of the algorithm is to find a new set of parameters, denoted by $\lambda^{(t+1)}$, that maximize the expected complete log likelihood computed in the E-step.

We now apply the EM algorithm to our case. The complete likelihood is denoted by $L_\lambda^c(\mathbf{X}) = P_\mathbf{X}(\mathbf{X} \mid \lambda)$. $L_\lambda^c(\mathbf{X})$ is,

$$L_\lambda^c(\mathbf{X}) = \prod_{n=1}^{N} \prod_{i=1}^{I} \lambda_i e^{-\lambda_i X_i^{(n)}} = \prod_{n=1}^{N} (\prod_{i=1}^{I} \lambda_i) e^{-\sum_{i=1}^{I} \lambda_i X_i^{(n)}}$$

Then the complete log likelihood, denoted by $l_\lambda^c(\mathbf{X})$, is,

$$l_\lambda^c(\mathbf{X}) = \sum_{n=1}^{N} (\sum_{i=1}^{I} log\lambda_i - \sum_{i=1}^{I} \lambda_i X_i^{(n)}) \tag{6}$$

Suppose in the t^{th} iteration step, the estimate for the parameters is $\lambda^{(t)}$. Then, the E-step of the EM algorithm is to take the expected value of the complete log likelihood over the conditional density $P_{\mathbf{X}|\mathbf{Y}}(\mathbf{X} \mid \mathbf{Y}; \lambda^{(t)})$. In our case,

$$\mathbb{E}[l_\lambda^c(\mathbf{X}) \mid \mathbf{Y}, \lambda^{(t)}] = \sum_{n=1}^{N} (\sum_{i=1}^{I} log\lambda_i - \sum_{i=1}^{I} \lambda_i \mathbb{E}[X_i^{(n)} \mid \mathbf{Y}, \lambda^{(t)}]) \tag{7}$$

The M-step is to choose $\lambda^{(t+1)}$ to maximize the expected log likelihood $\mathbb{E}[l_\lambda^c(\mathbf{X}) \mid \mathbf{Y}, \lambda^{(t)}]$. That is

$$\lambda^{(t+1)} = \text{argmax}_{\lambda \in \Theta} \mathbb{E}[l_\lambda^c(\mathbf{X}) \mid \mathbf{Y}, \lambda^{(t)}] \tag{8}$$

where the parameter space Θ is, $\Theta = \{(\lambda_1, \lambda_2, ..., \lambda_I) \ : \ \lambda_i > 0, i = 1, 2, ..., I\}$. The M-step in our case is fairly simple. By taking derivative with respect to the λ_i's, the optimization yields,

$$\lambda_i^{(t+1)} = \frac{N}{\sum_{n=1}^{N} \mathbb{E}[X_i^{(n)} \mid \mathbf{Y}, \lambda^{(t)}]} \qquad \text{for i = 1, 2, ..., I} \tag{9}$$

Hence, the key for applying the EM algorithm is to compute $\mathbb{E}[X_i^{(n)} \mid \mathbf{Y}, \lambda^{(t)}]$, for $i = 1, 2, ..., I$ at each iteration step. By independence, $\mathbb{E}[X_i^{(n)} \mid \mathbf{Y}, \lambda^{(t)}] = \mathbb{E}[X_i^{(n)} \mid Y^{(n)}, \lambda^{(t)}]$, where $Y^{(n)} = \sum_{i=1}^{I} X_i^{(n)}$. We will next compute this value. For convenience, we drop the time index n and iteration index t. Let $P_Y(\cdot \mid \lambda)$ denote the density for the random variable Y. It can be shown that

$$P_Y(y \mid \lambda) = \sum_{i=1}^{I} \Big(\prod_{j=1, j\neq i}^{I} \frac{\lambda_j}{\lambda_j - \lambda_i} \Big) \lambda_i e^{-\lambda_i y} \tag{10}$$

Then, for any $\mathbf{x} = (x_1, x_2, ..., x_I)$ such that $x_i \geq 0$ for all i, and $\sum_{i=1}^{I} x_i = \text{y}$,

$$P_{\mathbf{X}|Y}(\mathbf{x} \mid y; \lambda) = \frac{\prod_{i=1}^{I} \lambda_i e^{-\lambda_i x_i}}{P_Y(y \mid \lambda)} \tag{11}$$

Because $y = \sum_{i=1}^{I} x_i$, there are only $I - 1$ independent x_i's in equation (11). In principle, we can find the expression for the conditional density $P_{X_i|Y}(x_i \mid y; \lambda)$ by integrating the right-hand side of equation (11) with respect to the other $I - 2$ variables. But, we can do this more simply by noting that $Y = X_i + W_i$, where $W_i = \sum_{k=1, k\neq i}^{I} X_k$ has the density $P_{W_i}(w_i \mid \lambda)$. By the general expression in equation (10) for the density of the sum of independent exponentials,

$$P_{W_i}(w_i \mid \lambda) = \sum_{k=1, k\neq i}^{I} \Big(\prod_{j=1, j\neq k, j\neq i}^{I} \frac{\lambda_j}{\lambda_j - \lambda_k} \Big) \lambda_k e^{-\lambda_k w_i}$$

Therefore,

$$P_{X_i|Y}(x_i \mid y; \lambda) = \frac{\lambda_i e^{-\lambda_i x_i} P_{W_i}(y - x_i \mid \lambda)}{P_Y(y \mid \lambda)} \qquad \text{for } 0 \leq x_i \leq y \tag{12}$$

The conditional expectation of X_i can then be computed. We get the closed-form expression.

$$\mathbb{E}[X_i \mid Y, \lambda] = \frac{\sum_{k=1, k\neq i}^{I} \big(\prod_{j=1, j\neq k}^{I} \frac{\lambda_j}{\lambda_j - \lambda_k} \big) \big[\frac{\lambda_k e^{-\lambda_k Y}}{\lambda_i - \lambda_k} - \frac{\lambda_k e^{-\lambda_i Y}}{\lambda_i - \lambda_k} - \lambda_k Y e^{-\lambda_i Y} \big]}{P_Y(Y \mid \lambda)} \tag{13}$$

Substituting the result from equation (13) into equation (9) with the correct time and iteration indices, we then have an iterative procedure to compute the parameters λ.

3.2.2 Method of moments

The the following heuristic based on the method of moments can be an alternative for estimating the parameters in the exponential model. Again, consider the example shown in figure 1. From $Y_1 = X_1 + X_2 + X_4 + X_8$ and $Y_2 = X_1 + X_2 + X_4 + X_9$, the covariance of Y_1 and Y_2 is

$$\mathrm{Cov}(Y_1, Y_2) = \mathrm{Var}(X_1 + X_2 + X_4)$$

From

$$\mathrm{Var}(Y_1) = \mathrm{Var}(X_1 + X_2 + X_4) + \mathrm{Var}(X_8)$$

we get

$$\mathrm{Var}(X_8) = \mathrm{Var}(Y_1) - \mathrm{Cov}(Y_1, Y_2)$$

Both quantities on the right hand side above can be estimated based on measurement samples collected for the routes. Since the variance of an exponential random variable with parameter λ_i is $1/{\lambda_i}^2$, the parameter λ_8 can be estimated. In a similar fashion, the parameters λ_9 to λ_{15} can all be estimated from the variances for bottom level links. We can then move up one level from the bottom and estimate the parameters for link 4, 5, 6 and 7. For example, taking $Y_1 = X_1 + X_2 + X_4 + X_8$ and $Y_3 = X_1 + X_2 + X_5 + X_{10}$, we get

$$\mathrm{Var}(X_4) = \mathrm{Var}(Y_1) - \mathrm{Cov}(Y_1, Y_3) - \mathrm{Var}(X_8)$$

By now, we have estimates for all quantities on the right hand side. It is obvious this algorithm can be continue upward until all link parameters are determined.

We stress that this method requires multicast on the binary tree for collecting the samples for Y_i's. In other words, each time a packet is sent at the root node, 2^{n-1} samples are collected at the bottom level, one for each route, where n is the depth of the tree.

In the following, we will discuss some issues related to this heuristic approach.

1. The estimator based on the method of moments may not be asymptotically efficient, while a maximum likelihood estimator usually is. For an exponential random variable, the sample mean estimator and the maximum likelihood estimator are the same. We might lose efficiency when the sample variance is used in estimating the parameter. Nevertheless, we expect the estimator proposed here to be reasonably efficient, particularly when the depths of the binary tree is small. The major advantage of this estimation scheme is its computational simplicity.

2. In the Gaussian model, the variances can be estimated using the method of moments. However, since the means and and variances for the Gaussian model are unrelated, our method gives no information about the means. Our deterministic analysis shows that applying the method of moments only to the first moments will not yield unique estimates of the model parameters. That is why, in the exponential model, we need to rely on the second moments.

3. The method here can be used also for the one-parameter Gamma model, in which each link attribute is considered to be a Gamma random variable with known order n_i and unknown parameter λ_i. The density is of the form $\frac{\lambda_i e^{-\lambda_i x}(\lambda_i x)^{n_i-1}}{(n_i-1)!}$, where $x \geq 0$, and the variance is $\frac{n_i}{\lambda_i^2}$.

The method of moments is in fact a family of estimation methods that rely on the moments of the random variables. Here is another one of these methods. Again, consider the example shown in figure 1. From $Y_1 = X_1 + X_2 + X_4 + X_8$ and $Y_2 = X_1 + X_2 + X_4 + X_9$, we get $Y_1 - Y_2 = X_8 - X_9$. Notice that $Y_1 - Y_2$ can be observed. Since the mean of an exponential random variable with parameter λ_i is $1/\lambda_i$ and the variance is $1/{\lambda_i}^2$, we can write

$$\mathbb{E}[Y_1 - Y_2] = 1/\lambda_8 - 1/\lambda_9$$

$$\mathrm{Var}(Y_1 - Y_2) = 1/{\lambda_8}^2 + 1/{\lambda_9}^2$$

Given a random variable Z, let us define the following notations. Let $\hat{m}(Z)$ be the sample mean and $\hat{\sigma^2}(Z)$ be the sample variance. Then, assuming certainty equivalence, i.e., the estimate of the mean and the variance are in fact the mean and the variance of a random variable, it is reasonable to set up the following equations, and solve for λ_8 and λ_9.

$$\hat{m}(Y_1 - Y_2) = 1/\lambda_8 - 1/\lambda_9 \tag{14}$$

$$\hat{\sigma^2}(Y_1 - Y_2) = 1/{\lambda_8}^2 + 1/{\lambda_9}^2 \tag{15}$$

The positive solutions to the above equations are unique.

$$1/\lambda_8 = \frac{\hat{m}(Y_1 - Y_2) + \sqrt{2\hat{\sigma^2}(Y_1 - Y_2) - (\hat{m}(Y_1 - Y_2))^2}}{2} \tag{16}$$

$$1/\lambda_9 = \frac{-\hat{m}(Y_1 - Y_2) + \sqrt{2\hat{\sigma^2}(Y_1 - Y_2) - (\hat{m}(Y_1 - Y_2))^2}}{2} \tag{17}$$

In the similar fashion, all parameters associated with the links at the bottom level can be solved. Then, we can move one level up in the tree and try to solve the parameters associated with links

in that level. For example, $Y_1 - Y_3 = X_4 + X_8 - X_5 - X_{10}$. Since the parameters for X_8 and X_{10} have already been estimated, we consider them known constants. By independence of the X_i's and by certainty equivalence, we set up the following equations and want to solve for λ_4 and λ_5.

$$\hat{m}(Y_1 - Y_3) = 1/\lambda_4 + 1/\lambda_8 - 1/\lambda_5 - 1/\lambda_{10} \quad (18)$$

$$\hat{\sigma^2}(Y_1 - Y_3) = 1/{\lambda_4}^2 + 1/{\lambda_8}^2 + 1/{\lambda_5}^2 + 1/{\lambda_{10}}^2 \quad (19)$$

As a final note on the exponential model, if each link delay is a sum of J independent exponentials with a constant J parameters, the techniques used for the simple exponential model can be applied.

3.3 Mixture of exponential model

Some may argue that the exponential model is too simple to model link delays. In this section, we present a different model, the mixture of exponentials model. This model is motivated by the observation that the link delay distribution may have different decaying tail at different time, possibly due to the difference in the traffic load. If we consider each decaying tail as a mode, the link delay switches among these modes. It has been shown that power decaying tail can be approximated quite nicely with this model [6].

More specifically, let us model the delay at link i, X_i, as a mixture of J exponentials, each with a parameter $\lambda_{i,j}$, for $j = 1, 2, ..., J$, where J is a known constant. Let $\pi_{i,j}$ be the probability that X_i takes one of these J exponentials, with $\sum_{j=1}^{J} \pi_{i,j} = 1$. Let $\lambda_i = (\lambda_{i,j}, \lambda_{i,2}, ..., \lambda_{i,J})$. We can write the density for X_i.

$$P_{X_i}(x_i \mid \lambda_i) = \sum_{j=1}^{J} \pi_{i,j} \lambda_{i,j} e^{-\lambda_{i,j} x_i} \quad (20)$$

Again assume $Y = \sum_{i=1}^{I} X_i$, where Y is the route delay. We would like to estimate the parameters λ and π based on the observations of Y. We formulate a maximum likelihood estimator and apply the EM algorithm, because the estimation problem can be considered in a setting with unobserved hidden variables. First, the link delays, X_i's, are unobserved. Second, we can regard the particular mode a link delay chooses as a hidden variable. Let us define a vector-valued random variable Q_i associated with each X_i taking values in the set $\{\mathbf{e_1}, \mathbf{e_2}, ..., \mathbf{e_J}\}$, where $\mathbf{e_j} \in \mathbb{R}^J$ is a vector $(0, ..., 0, 1, 0, ..., 0)^T$, with the 1 in the j^{th} position. Denote $\bar{\mathbf{X}} = (X_i^{(n)})$ and $\bar{\mathbf{Q}} = (Q_i^{(n)})$, where $i = 1, 2, ..., I$ and $n = 1, 2, ..., N$. The variables in bold with a bar on the top are two-dimensional

variables. Then, the complete likelihood is,

$$L^c_{\lambda,\pi}(\bar{\mathbf{X}}, \bar{\mathbf{Q}}) = \prod_{n=1}^{N}\prod_{i=1}^{I}\prod_{j=1}^{J}(\pi_{i,j}\lambda_{i,j}e^{-\lambda_{i,j}X_i^{(n)}})^{Q_{i,j}^{(n)}} \tag{21}$$

Here, we make the notation more compact by letting the discrete variables $\bar{\mathbf{Q}}$ vary. Note that $Q_{i,j}^{(n)}$ stands for the j^{th} entry of the vector $Q_i^{(n)}$, which is either 1 or 0. For each index i, only one of the J terms indexed by j survives. The complete log likelihood is,

$$l^c_{\lambda,\pi}(\bar{\mathbf{X}}, \bar{\mathbf{Q}}) = \log L^c_{\lambda,\pi}(\bar{\mathbf{X}}, \bar{\mathbf{Q}}) = \sum_{n=1}^{N}\sum_{i=1}^{I}\sum_{j=1}^{J} Q_{i,j}^{(n)}(\log \pi_{i,j} + \log \lambda_{i,j} - \lambda_{i,j}X_i^{(n)}) \tag{22}$$

To simplified the notation, let $\langle\cdot\rangle$ represent the conditional expectation operator $\mathbb{E}[\ \cdot\ |\ \mathbf{Y}, \lambda^{(t)}, \pi^{(t)}]$. The expected value of the complete log likelihood is,

$$\langle l^c_{\lambda,\pi}(\bar{\mathbf{X}}, \bar{\mathbf{Q}})\rangle = \sum_{n=1}^{N}\sum_{i=1}^{I}\sum_{j=1}^{J}(\langle Q_{i,j}^{(n)}\rangle \log \pi_{i,j} + \langle Q_{i,j}^{(n)}\rangle \log \lambda_{i,j} - \lambda_{i,j}\langle Q_{i,j}^{(n)}X_i^{(n)}\rangle) \tag{23}$$

In the M-step of the EM algorithm, we would like to find $\lambda^{(t+1)}$ and $\pi^{(t+1)}$ so that $\langle l^c_{\lambda,\pi}(\bar{\mathbf{X}}, \bar{\mathbf{Q}})\rangle$ is maximized. This is a constraint optimization problem, subject to the constraint that $\sum_{j=1}^{J}\pi_{i,j} = 1$ for $i = 1, 2, ..., I$. It is easy to show that,

$$\pi_{i,j}^{(t+1)} = \sum_{n=1}^{N}\langle Q_{i,j}^{(n)}\rangle / N \tag{24}$$

$$\lambda_{i,j}^{(t+1)} = \sum_{n=1}^{N}\langle Q_{i,j}^{(n)}\rangle / \sum_{n=1}^{N}\langle X_i^{(n)}Q_{i,j}^{(n)}\rangle \quad \text{for } i = 1, 2, ..., I \text{ and } j = 1, 2, ..., J \tag{25}$$

Therefore, the key is to compute $\langle Q_{i,j}^{(n)}\rangle$ and $\langle X_i^{(n)}Q_{i,j}^{(n)}\rangle$. Since samples at different time instances are assumed to be independent, the conditioning in the above expressions of conditional expectations is on $Y^{(n)}$ alone. In order to compute the above expectations, we need to evaluate the posterior probabilities conditional on the observation of $Y^{(n)}$. We will drop the time index n and iteration index t in the following analysis.

Let the random vectors $\mathbf{X} = (X_i)$ and $\mathbf{Q} = (Q_i)$, for $i = 1, 2, ..., I$, and $\mathbf{x}$ and $\mathbf{q}$ be the corresponding non-random versions of them. One strategy is to start with the joint conditional density, $P_{\mathbf{X},\mathbf{Q}|Y}(\mathbf{x}, \mathbf{q}\ |\ y; \lambda, \pi)$. More explicitly, for any $\mathbf{x}$ such that $x_i \geq 0$ for all i, and $\sum_{i=1}^{I} x_i = y$, we need to compute

$$P_{\mathbf{X},\mathbf{Q}|Y}(\mathbf{x}, Q_{1,j_1} = 1, Q_{2,j_1} = 1, ..., Q_{I,j_I} = 1\ |\ y; \lambda, \pi) = \frac{\prod_{i=1}^{I}\pi_{i,j_i}\lambda_{i,j_i}e^{-\lambda_{i,j_i}x_i}}{P_Y(y\ |\lambda, \pi)} \tag{26}$$

for all I-tuples $(j_1, j_2, ..., j_I) \in \{1, 2, ..., J\}^I$, where

$$P_Y(y \mid \lambda, \pi) = \sum_{j_1, j_2, ..., j_I \in \{1,2,...,J\}^I} [\sum_{i=1}^{I} \lambda_{i,j_i} e^{-\lambda_{i,j_i} y} \prod_{k=1, k \neq i}^{I} \frac{\lambda_{k,j_k}}{\lambda_{k,j_k} - \lambda_{i,j_i}}] \prod_{i=1}^{I} \pi_{i,j_i} \tag{27}$$

We can then compute $P_{X_i|Y}$ and $P_{X_i,Q_i|Y}$ by marginalizing the joint conditional density. The procedure developed here is a general one for mixture models, including the mixture of exponentials and the mixture of exponentials and Gaussians. However, the computation complexity is exponential in the number of links, I, due to the discrete nature of the random variables, Q_i's. For example, if we choose $k = 2$ and $L = 10$, then we need to compute equation (26) 2^{10} times for each observation of Y, and the outer sum in equation (27) has also 2^{10} terms.

A simpler approach is to first compute the conditional density $P_{\mathbf{X}|Y}(\mathbf{x} \mid y; \lambda, \pi)$.

$$P_{\mathbf{X}|Y}(\mathbf{x} \mid y; \lambda, \pi) = \frac{\prod_{i=1}^{I} (\sum_{j=1}^{J} \pi_{i,j} \lambda_{i,j} e^{-\lambda_{i,j} x_i})}{P_Y(y \mid \lambda, \pi)} \tag{28}$$

where

$$P_Y(y \mid \lambda, \pi) = \int_A \prod_{i=1}^{I} (\sum_{j=1}^{J} \pi_{i,j} \lambda_{i,j} e^{-\lambda_{i,j} x_i}) dx_1 dx_2 ... dx_{I-1} \tag{29}$$

where the set

$$A = \{(x_1, x_2, ..., x_{I-1}) : x_i \geq 0, i = 1, 2, ..., I-1, \text{and} \sum_{i=1}^{I-1} x_i \leq y\}$$

Then, the conditional density $P_{X_i,Q_i \mid Y}$ can be computed.

$$P_{X_i,Q_i|Y}(x_i, Q_{i,j} = 1 \mid y; \lambda, \pi) = P_{Q_i|X_i,Y}(Q_{i,j} = 1 \mid x_i, y; \lambda, \pi) P_{X_i|Y}(x_i \mid y; \lambda, \pi) \tag{30}$$

Given X_i, Q_i is independent of Y. Hence,

$$P_{Q_i|X_i,Y}(Q_{i,j} = 1 \mid x_i, y; \lambda, \pi) = P_{Q_i|X_i}(Q_{i,j} = 1 \mid x_i; \lambda, \pi) \tag{31}$$

$$= \frac{\pi_{i,j} \lambda_{i,j} e^{-\lambda_{i,j} x_i}}{\sum_{r=1}^{J} \pi_{i,r} \lambda_{i,r} e^{-\lambda_{i,r} x_i}} \tag{32}$$

The transform method can further help us to reduce the computation complexity, because of the simple form of the moment generating function for the exponential distribution. In particular, we need not calculate the integration in equation (29). The moment generating function of Y can be expressed as $G_Y(s) = \prod_{i=1}^{I} G_{X_i}(s)$, where G_{X_i}'s are the moment generating functions for the X_i's. Because $G_{X_i}(s) = g_i(s) / \prod_{j=1}^{J} (\lambda_{i,j} - s)$, where $g_i(s)$ is a polynomial in s whose degree is less than J, we get,

$$G_Y(s) = \frac{\prod_{i=1}^{I} g_i(s)}{\prod_{i=1}^{I} \prod_{j=1}^{J} (\lambda_{i,j} - s)} \tag{33}$$

We can perform partial fraction expansion on $G_Y(s)$ and get,

$$G_Y(s) = \sum_{i=1}^{I} \sum_{j=1}^{J} \frac{\alpha_{i,j}\lambda_{i,j}}{\lambda_{i,j} - s} \tag{34}$$

for some real numbers $\alpha_{i,j}$. Hence,

$$P_Y(y \mid \lambda, \pi) = \sum_{i=1}^{I} \sum_{j=1}^{J} \alpha_{i,j} e^{-\lambda_{i,j} y} \tag{35}$$

Let us write $Y = X_i + W_i$, where $W_i = \sum_{k=1, k\neq i}^{I} X_k$. Then, by the general expression shown in equation (34),

$$P_{W_i}(w_i \mid \lambda, \pi) = \sum_{k=1, k\neq i}^{I} \sum_{j=1}^{J} \beta_{k,j} e^{-\lambda_{k,j} w_i} \tag{36}$$

for some real numbers $\beta_{k,j}$. Then,

$$P_{X_i|Y}(x_i \mid y; \lambda, \pi) = \frac{(\sum_{j=1}^{J} \pi_{i,j}\lambda_{i,j} e^{-\lambda_{i,j} x_i}) P_{W_i}(y - x_i \mid \lambda, \pi)}{P_Y(y \mid \lambda, \pi)} \quad \text{for } 0 \leq x_i \leq y \tag{37}$$

Given the conditional density $P_{X_i,Q_i|Y}$, in principle, $\langle X_i Q_{i,j} \rangle$ can be computed. We next show how to compute $\langle Q_{i,j} \rangle$. To do this, we need to compute $P_{Q|Y}(Q_{i,j} = 1 \mid y; \lambda, \pi)$. In the following, notations are simplified when there is no confusion.

$$\begin{aligned}
P(Q_{i,j} = 1 \mid y) &= \int_{x_i \leq y} P(Q_{i,j} = 1, x_i \mid y) dx_i \\
&= \int_{x_i \leq y} P(Q_{i,j} = 1 \mid x_i, y) P(x_i \mid y) dx_i \\
&= \int_{x_i \leq y} P(Q_{i,j} = 1 \mid x_i) P(x_i \mid y) dx_i \\
&= \int_{x_i \leq y} \frac{P(Q_{i,j} = 1) P(x_i \mid Q_{i,j} = 1)}{P(x_i)} P(x_i \mid y) dx_i
\end{aligned}$$

4 Probablistic Case: Non-parametric Models

4.1 Introduction

When assuming a parametric model, one needs to address the issue of relevance of the model, i.e., how close the model resembles reality. Currently, there is no generally agreed model for link delays. Moreover, as we have seen, it can be difficult to infer the model parameters. These points

motivate the study on the identification of non-parametric models through sampling. The objective is to find a way to identify the distribution of a particular link attribute based on the end-to-end measurement, and to ask how well we can do this. To obtain a non-parametric distribution by sampling is generally a slow process. However, if we can postulate a relevant parametric model based on the non-parametric model obtained through sampling, the effort is well justified. If we limit our goals to estimate only partial information such as the mean and variance of the link attribute, the non-parametric approach becomes even more viable.

At this point, we specifically consider the variable part of the link delay as the subject of study. We do not consider the fixed propagation delay on a link, since there are various ways to determine it. For instance, it can be computed if the locations of the routers are known. In addition, because the propagation delays are the same in both directions of transmission, by the result for symmetric networks in section 2.3, it can be determined from end-to-end measurement. In practice, people use the *ping* program to measure it. In the following, we will assume the propagation delays are zero. The term, delay, is reserved to the variable delay.

Strictly speaking, our result from the deterministic analysis has ruled out the possibility to determine a genuine non-parametric model by sampling the end-to-end route attributes. For, this will necessarily involves measuring a set of route attributes at each sampling time instances and determining the contributions from each of the links on the routes. Hence, we do need to add to the non-parametric model a mild assumption which can be easily validated.

Assumption 4.1 *There exists an $\kappa > 0$, such that the probability that the delay of any link is zero is greater than κ.*

The assumption is based on our observation that the end-to-end delay on a route is typically very small (in the range of a few milliseconds). The small delay indicates the absence of queueing delay, which can easily be up to hundreds of milliseconds. We think that the queues at the routers along a route are empty most of the time. The small variable delay is possibly due to the variability of the processing delay by the router's processors. The assumption made in 4.1 idealizes this situation by assuming the variable processing delay is zero. Besides the usual independent assumptions among the link delays, we make no more assumptions. The delay model of each link appears to be non-parametric with a mild assumption.

For each link i, let the probability that the link delay is zero be η_i, where $\kappa < \eta_i \leq 1$. These

parameters need not to be estimated separately, even though they can be. They affect the efficiency of the estimation procedure and the estimation error when the estimation is not exact. The uniform lower bound of the $\{\eta_i\}$'s, κ, in assumption 4.1 can be considered close to zero. It is only used for showing a convergence result in a lemma later.

To determine a link delay distribution, we would like to obtain independent link delay samples from the end-to-end delay samples. According to the deterministic analysis, this can not be done in general if we do not know any probabilistic structure of the delays. In our case, assumption 4.1 provides extra information on the probabilistic structure that enables us to do so. The condition under which this can be done is contained in the following lemma, which is a direct consequence of the deterministic result.

Lemma 4.1 *A link delay sample for link i can be determined from the route delay samples collected at the same sampling instance if and only if there exists at least one route containing link i on which all links except i have zero delay.*

Assumption 4.1 makes the condition stated in the lemma possible. In fact, it guarantees the existence of the condition in the asymptotic regimes of large number of routes and/or large number of time samples. The procedure we will propose for estimating the link delay distribution appears to be naive at the first glance. However, the above lemma shows that it is in fact the best we can do. There is a serious inherent inefficiency in estimating the distributions of the link delays by end-to-end measurement because only selected samples will be accepted as valid samples. The acceptance rate of samples fundamentally limits the practicality of the sampling procedure. To increase the overall sample population, we need to explore the statistical redundancy in both the spatial and temporal domains. In other words, we want to take advantages of the large number of routes and of the large number of independent time samples. We will see that the nature of the spatial and the temporal redundancy are the same.

4.2 Sampling theory for link delays

4.2.1 Route redundancy

This subsection shows why it is possible to obtain one valid link delay sample. Suppose we would like to estimate the distribution of the delay X_l of link l. Let us assume there exist M routes on

which link l is the only common link between any of the two routes. Let us fix a time instance. If M is large enough, we would expect that almost certainly, on at least one of the M routes, all but link l have zero delay. More formally, let the number of links on the i^{th} route be $I_i + 1$. Denote the link delays, except the common link l, on the i^{th} route $X_{i,j}$, for $j = 1, 2, ..., I_i$, and let the route delay of route i be Y_i. Hence, $Y_i = \sum_{j=1}^{I_i} X_{i,j} + X_l$, for $i = 1, 2, ..., M$. Define a new random variable $W_M = \min_{i \in \{1,2,...,M\}} Y_i$. Then we would expect that, $W_M \xrightarrow{a.s.} X_l$, as $M \longrightarrow \infty$. Assumption 4.1 is a sufficient condition for this to be true.

Lemma 4.2 *Under assumption 4.1,* $W_M \xrightarrow{a.s.} X_l$, *as* $M \longrightarrow \infty$.

Proof: Let $V_M = \min_{i \in \{1,2,...,M\}} Y_i - X_l$. It suffices to show $V_M \xrightarrow{a.s.} 0$ as $M \longrightarrow \infty$. Fix an $\epsilon > 0$. Since the Y_i's are non-negative random variables, it is sufficient to show $P(V_M > \epsilon \text{ infinitely often }) = 0$. By the Borel-Cantelli lemma([4]), this is true if $\sum_{M=1}^{\infty} P(V_M > \epsilon) < \infty$. For fixed M, let $\alpha_i = P(Y_i > \epsilon)$, for $i = 1, 2, ..., M$. By assumption 4.1, there exists a $0 \leq \delta < 1$, such that $\alpha_i = P(Y_i - X_l > \epsilon) < \delta$, for $i = 1, 2, ..., M$. Indeed, we can take $\delta = 1 - \kappa^{I_i}$. Now,

$$P(V_M > \epsilon) = P(Y_1 - X_l > \epsilon,\ Y_2 - X_l > \epsilon, ..., Y_M - X_l > \epsilon) = \prod_{i=1}^{M} \alpha_i \leq \delta^M$$

Hence, $\sum_{M=1}^{\infty} P(V_M > \epsilon) = \sum_{M=1}^{\infty} \delta^M < \infty$. ■

Lemma 4.2 says that if there are enough redundant routes sharing a single common link l, then we can take the minimum of all route delays as the link delay of link l. The convergence result can be easily generalized to cases in which link l is the only common link to *all* routes, but some proper subsets of the routes also share other links.

We can define some notion of the convergence rate, and show that convergence rate is exponential in the number of routes, M. However, the value of M for which the convergence is considered "reasonable" fast crucially depends on other parameters. Moreover, the number of routes that can be established in practice is limited by, first, the difficulty and cost in managing a large number of routes, and second, by the fanout of the routers. We will consider what temporal redundancy can offer.

4.2.2 Temporal redundancy

We, first, reiterate the assumption that delay samples at two different sampling instances are independent. Let us revisit the scenario discussed in lemma 4.2, where M routes share a single

common link l. Suppose N independent set of samples of the route delays are observed at N different time instances. Define $Y_i^{(n)} = \sum_{j=1}^{I_i} X_{i,j}^{(n)} + X_l^{(n)}$, for $i = 1, 2, ..., M$, and for $n = 1, 2, ..., N$. Define the random variable $W_{M,N} = \min\{Y_i^{(n)} : i \in \{1, 2, ..., M\}, n \in \{1, 2, ..., N\}\}$, and tentatively assume X_l is a constant. Then, similar to the case of lemma 4.2, we can show

$$\lim_{MN \longrightarrow \infty} W_{M,N} = X_l \qquad a.s. \tag{38}$$

We emphasize that the equality in equation (38) is asymptotic in the product of M and N. In other words, the effects on the convergence rate from the number of routes and from the number of time samples are the same. In practice, we can either take a large number of routes, or a large number of time samples, depending on the situations.

4.2.3 Practicality of the sampling theory

In the following, we will look at a simple example to demonstrate the feasibility and constraints of the sampling technique alluded by the previous discussion.

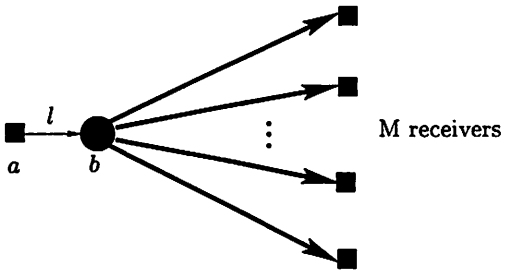

Figure 2: Singlely-overlapped Routes

Figure 2 shows a special case of the situation discussed in the previous two subsections. We want to measure the delay of link l between the sender node a and router node b. In the figure, there are M routes from a via b, where the M routes branch out. The thinner directed line from a to b, stands for a link (link l in this case) as usual. Each thicker directed line represents a path which may traverse multiple nodes. Only the ending nodes for those paths are shown. Suppose each route contains $I + 1$ links. For every link, let the probability that the link delay is zero be η, where $0 < \eta \leq 1$. Suppose we set up a multicast connection so that when a packet is sent at node a, each of the receivers of the M routes receives a copy of the packet. We then have M route delays at each sampling instance. Let total N packets be sent from node a at N different sampling times.

Then, a total of MN route delay samples can be collected. If X_l is a constant, then (38) follows. We would like to know the value of MN with which we can obtain a valid link delay sample. The probability that the minimum of the MN end-to-end delays is equal to X_l is the probability that at least one of the MN path delays from node b to the receivers is zero. Denote this probability as $1 - h(\eta, I, MN)$, where $h(\eta, I, MN) = (1 - \eta^I)^{MN}$. We hope $h(\eta, I, MN)$ approaches 0 for reasonable values of MN. Our first impression is $h(\eta, I, MN)$ tends to 0 exponentially fast in MN. However, a second look tells that $h(\eta, I, MN)$ depends very crucially on the value of η^I. Because η^I is exponential in I, it can be extremely small. For example, for $\eta = 0.1$ and $I = 10$, $\eta^I = 10^{-10}$. For a fixed probability $0 < \alpha < 1$, let us solve MN for which,

$$h(\eta, I, MN) = (1 - \eta^I)^{MN} = \alpha \tag{39}$$

Then,

$$MN = \frac{\log \alpha}{\log(1 - \eta^I)} \tag{40}$$

For small η^I,

$$MN \approx \frac{-\log \alpha}{\eta^I} \tag{41}$$

Notice that MN is proportional to $1/\eta^I$. The approximation of (41) is good for η^I upto 0.2. When $\eta^I = 10^{-10}$, by (41), $MN = 4.6 \times 10^{10}$, 9.2×10^{10}, 13.8×10^{10}, and 18.4×10^{10}, for $\alpha = 10^{-2}$,10^{-4}, 10^{-6}, and 10^{-8}, respectively. These values for MN are apparently quite large. Another important observation is, once the value MN is large enough (in the range of $1/\eta^I$), the effect of exponential convergence kicks in. We can achieve an extremely low value of $h(\eta, I, MN)$ with moderate increase of MN.

We believe $MN = 10^6$ is a rough upper bound for a reasonable measurement set up. With this value of MN, we show the upper bounds for $I + 1$ for different values of η in Table 1, based on the approximation in (41). Recall that $I + 1$ is the number of links per route in the current example; and η is the probability that the link delay is zero.

In today's internet, a typical route across North America traverses less than 20 links. Many of these 20 or so links are within the local area network of the sender or the receiver, and hardly have any delay. It is likely that one is interested in the delay statistics of only a few of these 20 links. The parameters, η, are probably different for each link, and we now index them by the links. Now $\prod_{i=1}^{I} \eta_i$, which is the probability that the delay of the path consisting of links 1, 2, ..., and I is

η	$I+1$, $\alpha = 10^{-6}$	$I+1$, $\alpha = 10^{-18}$
0.1	6	5
0.3	10	9
0.6	23	21
0.8	51	46
0.9	107	97

Table 1: Upper bounds on the number of links per route

zero, takes the role of η^I. From the approximation of in (41) with $\prod_{i=1}^{I} \eta_i$ replacing η^I, we get,

$$\prod_{i=1}^{I} \eta_i \approx -\frac{\log \alpha}{MN} \tag{42}$$

For $MN = 10^6$, $\prod_{i=1}^{I} \eta_i$ is 1.38×10^{-5} and 4.14×10^{-5} for $\alpha = 10^{-6}$ and $\alpha = 10^{-18}$, respectively. Our measurement of the Internet delay shows that the actual value for this product is significantly greater than 0.1 even for very long routes. Hence, the sampling scheme alluded here is practical. In fact, it works for even small values of MN. For instance, suppose we choose $MN = 100$. Then, $\prod_{i=1}^{I} \eta_i$ is 1.38×10^{-1} and 4.14×10^{-1} for $\alpha = 10^{-6}$ and $\alpha = 10^{-18}$, respectively. Hence, as long as routes are non-congested half of the times, the sampling scheme works.

4.3 Rejection-based sampling scheme

In this subsection, we introduce our rejection-based sampling scheme, whose salient feature is that only those true link delay samples are accepted and the rest are rejected.

4.3.1 Illustrative example - two-level trees

Figure 3a shows a simple network of a two-level tree with two routes. We wish to collect samples for the delay of link l_1. Let route $(1,1)$ be from node a_1 to c_1. The index indicates route $(1,1)$ contains link u_1 and link l_1. Let route $(1,2)$ be from node a_1 to c_2. We will collect N set of route delay samples on route $(1,1)$ and route $(1,2)$ via a multicast connection from a_1 to c_1 and c_2. At each sampling time instance, n, the delay of route $(1,1)$ is $Y_{1,1}^{(n)} = X_1^{(n)} + Z_1^{(n)}$, and the delay of route $(1,2)$ is $Y_{1,2}^{(n)} = X_1^{(n)} + Z_2^{(n)}$, where X_1 is the link delay for link u_1, and Z_i is the link delay for link l_i for $i = 1, 2$. Among the N set of time samples, let us restrict ourselves to the set S of

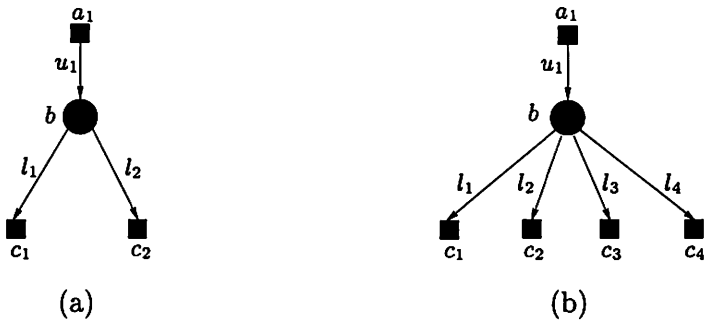

Figure 3: Two-level Tree

time instances for which $Y_{1,2}^{(n)} = 0$, i.e., $S = \{n \in \{1, 2, ..., N\} : Y_{1,2}^{(n)} = 0\}$. Then, it must be true that $Y_{1,1}^{(n)} = Z_1^{(n)}$ for $n \in S$. The rejection-based sampling is very simple. At each time instance n, if $Y_{1,2}^{(n)} > 0$, then reject the sample; if $Y_{1,2}^{(n)} = 0$, then accept the sample $Y_{1,1}^{(n)}$ as a sample for the delay of link l_1. Due to the symmetry of the network, we can also extract the delay samples for link l_2 from the same sample population $\{Y_{1,1}^{(n)}, Y_{1,2}^{(n)}\}_{n=1}^{N}$. Because the link delays are independent to each other, the sample distribution for link l_1 approaches the distribution of Z_1 as $N \longrightarrow \infty$. Given the distributions for Z_1 and $Y_{1,1}$, the distribution for X_1 can be computed. Figure 3b illustrates a setup in which route redundancy is further exploited at the lower level.

4.3.2 Sampling on general network

In the examples shown in figure 3, the link we are measuring terminates at a receiver node. In more general situations, the link to be measured can be deep inside the network, not connected to any measurement nodes. We now discuss the sampling technique that deals with general situations such as the example shown in figure 4. There, the thinner line from node c to node d, denoted by l, stands for a link as usual. We want to sample the delay of link l. The thicker lines are paths whose intermediate nodes are not drawn. Let route 1 be (a, d, e), let route 2 (b, c, d, e), and let route 3 be (b, c, f). Let Y_i be the delay for route i, for $i = 1, 2, 3$.

In all previous examples, the network can be covered by a single multicast tree. A sampling instance, n, is the time when a packet is sent from the root of the multicast tree. In the current example, we do not have a single multicast tree. Let us consider a notion of sampling period, a short time interval on which we collect one set of route delays. Suppose link delays change slow enough so that they stay constant during each sampling period. The sampling technique can be as

follows. At each sampling instance n, get the measurement for the route delays $Y_i^{(n)}$ for $i = 1, 2, 3$, and collect N set of such measurements. Let $S = \{n \in \{1, 2, ..., N\} : Y_1^{(n)} = 0, Y_3^{(n)} = 0\}$. S is the set of sampling instances when the delays on route 1 and route 3 are both zero. At those sampling instances, the delay samples for link l, denoted by X_l, is the delay on route 2. That is,

$$X_l^{(n)} = Y_2^{(n)} \qquad \text{for } n \in S \tag{43}$$

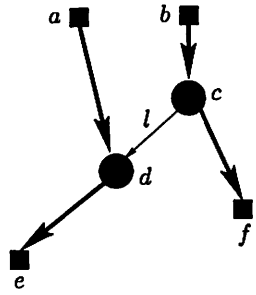

Figure 4: Multi-level Tree - multiple senders

The difficulty of algorithm lies in that link delays may not be constant during each sampling period. It is quite likely that the link delay samples are sensitive to the precise sampling times. In that case, for the above algorithm to work, the packet sent from node a and the packet sent from node b should be arranged to arrive at node d simultaneously. Due to the apparent difficulty, we propose a different strategy that does not directly collect link delay samples but collects path delay samples. If we know the delay distribution of two paths which differ only in one link, we can compute the delay distribution for that link. For the example in figure 4, let W_1 and W_2 be the delays for path (d, e) and (c, e), respectively. Since $W_2 = W_1 + X_l$, the probability mass functions for these random delays are related by $P_{W_2} = P_{W_1} \otimes P_{X_l}$, where $\otimes$ stands for convolution. Here, we assume that delays are discretized. If P_{W_2} and P_{W_1} are known, P_{X_l} can be computed by the standard procedure of deconvolution.

The method for collecting path delay samples is an obvious extension to the case of the two-level trees shown in figure 3. Simply interpret all directed lines in figure 3 as paths, and replace link delays with path delays in the corresponding sampling algorithm.

5 Conclusion

This paper address the issue of determining or estimating link delays based on the observation of end-to-end route delays. We have looked at three situations: the link delay is a constant (deterministic case), it is random and has a distribution from a model family (parametric model), or it is random with unknown distribution (non-parametric model). The deterministic analysis also serves as one of the building blocks for the non-parametric case.

The purpose of this paper is more on developing a broad understanding and families of techniques for solving the link delay inference problem than on evaluating a single technique for a specific situation. We certainly expect that some topics of the paper can be pursued further. For example, the applicability and performance of the EM algorithm to the problem of network delay inference need to be evaluated with realistic data. However, we hope that our study can provide insights to more specific inference problems.

References

[1] R. Cáceres, N.G. Duffield, J. Horowitz, and D. Towsley. Multicast-Based Inference of Network-Internal Loss Characteristics. *IEEE Transactions in Information Theory*, 45:2462–2480, 1999.

[2] A.P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of Royal Statistical Society*, 39:1–38, 1977.

[3] N.G. Duffield and F. Lo Presti. Multicast Inference of Packet Delay Variance at Interior Network Links. In *Proc. INFOCOM 2000*, Tel Aviv, Israel, March 2000.

[4] Richard Durrett. *Probability - Theory and Examples.* Duxbury Press, 2nd edition, 1996.

[5] F. Lo Presti, N.G. Duffield, J. Horowitz, and D. Towsley. Multicast-Based Inference of Network-Internal Delay Distributions. *Preprint*, 1999.

[6] David Starobinski and Moshe Sidi. Modeling and Analysis of Power-Tail Distributions via Classical Teletraffic Methods. *To appear in Queueing Systems.*

[7] Gilbert Strang. *Linear Algebra and Its Applications.* Saunders College Publishing, 3rd edition, 1998.

[8] Y. Vardi. Network Tomography: Estimating Source-Destination Traffic Intensities From Link Data. *Journal of the American Statistical Association*, 91:365–377, March 1996.