CHAPTER 1

NETWORK TOMOGRAPHY: A REVIEW AND RECENT DEVELOPMENTS

Earl Lawrence[a], George Michailidis[b], Vijayan N. Nair[b] and Bowei Xi[c]

[a] *Statistical Sciences Group*
Los Alamos National Laboratory
Los Alamos, NM 87545
earl@lanl.gov

[b] *Department of Statistics*
The University of Michigan
Ann Arbor, MI 48109-1107
{*gmichail,vnn*}*@umich.edu*

[c] *Department of Statistics*
Purdue University
West Lafayette, IN 47907
xbw@stat.purdue.edu

The modeling and analysis of computer communications networks give rise to a variety of interesting statistical problems. This paper focuses on network tomography, a term used to characterize two classes of large-scale inverse problems. The first deals with passive tomography where aggregate data are collected at the individual router/node level and the goal is to recover path-level information. The main problem of interest here is the estimation of the origin-destination traffic matrix. The second, referred to as active tomography, deals with reconstructing link-level information from end-to-end path-level measurements obtained by actively probing the network. The primary application in this case is estimation of quality-of-service parameters such as loss rates and delay distributions. The paper provides a review of the statistical issues and developments in network tomography with an emphasis on active tomography. An application to Internet telephony is used to illustrate the results.

Key Words: E-M algorithm, Inverse problems, Missing data, Probing experiments.

1. Introduction

There has been a great deal of interest recently, in both the engineering and research communities, on the modeling and analysis of communications and computer networks. This paper provides a review of network tomography and describes some interesting statistical issues, challenges, and recent developments. The term network tomography, introduced by Vardi (1996), has been used in the literature to characterize two broad classes of inverse problems. The first is passive tomography where aggregate data are collected at the router level. The goal is to disaggregate these to obtain finer-level information. The most common application, which was the original problem studied in Vardi (1996), is estimation of the origin-destination traffic matrix of a network. The second is active tomography where the network is actively "probed" by sending packets from a source to several receiver nodes, all located on the periphery of the network. Here one can collect only end-to-end path-level information, and the goal is to use this to recover individual link-level information. We will provide a brief review of both of these areas but focus more on the latter, as this has been the subject of our own research. There are also several other interesting statistical problems, especially related to network data obtained from a single network link, that arise in the study of communications and computer networks. These will not be be discussed here. See, however, the collection of papers in Adler et al. (1998), Park and Willinger (2000) and references therein).

The work in network tomography has been stimulated by the demand for sophisticated techniques and tools for monitoring network utilization and performance by network engineers and internet service providers (ISP). This need has increased further in recent years due to the complexity of new services (such as video-conferencing, Internet telephony, and on-line games) that require high-level quality-of-service (QoS) guarantees. The tools and techniques are also important for network management tasks such as fault and congestion detection, ensuring service-level-agreement compliance, and dynamic replica management of Web services, just to name a few (Coates et al. (2002a)).

There are two categories of methods in network tomography: (i) node-oriented methods that collect packet and network flow information passively through monitoring agents located at local network devices such as routers, switches, and hosts; and (ii) path-oriented methods that collect information about connectivity and latency in a network by actively sending probe packets through the network from nodes located on its periphery. The first category of tools are geared towards network operators who use the information for capacity planning and management decisions. Their main shortcoming is that they require access to all the network

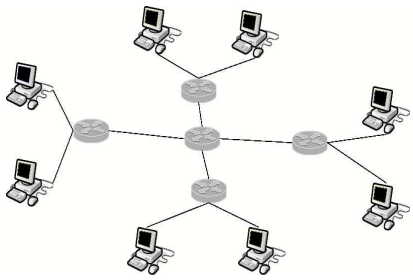

Fig. 1. Layout of a small network showing routers/nodes and links

elements (routers/switches) in order to deploy monitoring agents to collect the information. Furthermore, the amount of data generated can be substantial. The second category of tools collect data on network performance measures that are indirectly related to the parameters of interest and does not require cooperation from the internal nodes of the network. For both types, however, the collected data have to be appropriately processed (through the solution of different types of statistical inverse problems) to obtain the information of interest. (Castro et al., 2004).

We provide here a brief background on network traffic flow so that readers can follow the discussion. A more detailed and accessible introduction can be found in Marchette (2001). Throughout, we represent a network by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ the set of links. Figure 1 is an example of a small network with computers/workstations connected by routers and links. When a file is transferred from one location in the network to another (or one node to another), the file's content is first broken into pieces, called packets. Information about origin-destination, reassembly instructions (such a sequence numbers), and error-correcting features are also added to the packet. The origin-destination information is used by the network elements (routers and switches) to deliver the packets to the intended recipient. One can think of the routers (internal nodes in Figure 1) as the intersections in a road network. Packets are queued at routers, awaiting their transmission to the next router according to some protocol (first-in-first-out is common, but there are others). Physically, a queue consists of a block of computer memory that temporarily stores the packets. If the queue (memory) is full when a packet arrives, it is discarded and, depending on the transmission protocol, the sender may or may not be alerted. Otherwise, it waits until it reaches the front of the queue and is forwarded to the next router on the way to its destination. This queuing mechanism is responsible for observed packet losses and, to a large extent, for packet delays.

2. Passive Tomography

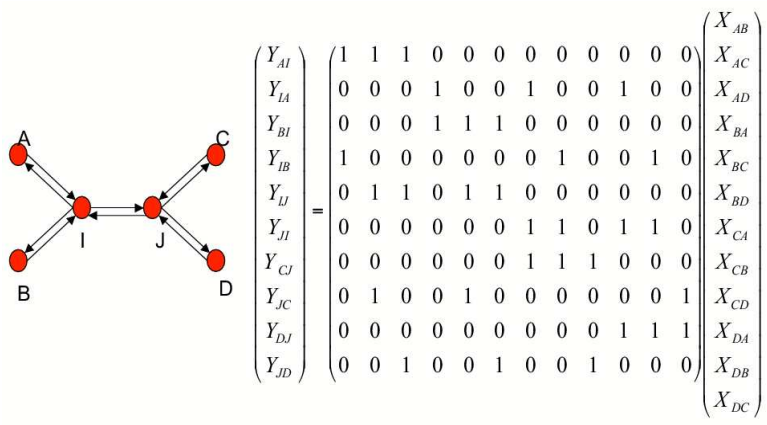

Fig. 2. A small network and its associated origin-destination traffic matrix.

The goal in traffic (or origin-destination) matrix estimation is to obtain information about the distributions of traffic flowing from $\mathcal{V}_i$ to $\mathcal{V}_j$ for all pairs of nodes i and j in the network. Of course, one would also have to study various sources of variation such as time-of-day, day-of-the-week, and other effects to characterize the traffic pattern. This information is used by network engineers for capacity planning and network management operations.

In this application, monitoring agents are placed at the individual nodes, and data on total number of packet counts traversing the node are collected. The packets do contain origin-destination (OD) information, but due to volume of the data, it is impractical to access individual packets to collect this information. So only total packet count data are available and are obtained using the Simple Network Management Protocol (SNMP).

Let $|\mathcal{V}|$ and D denote the number of nodes and OD pairs in the network respectively. Let us restrict attention to a fixed time period where the traffic patterns are fairly homogeneous. Let Y_t be a $|\mathcal{V}|$ column vector containing the number of packets traversing all the nodes in period t for $t = 1, ..., T$ during the time of study, with $\bar{Y}$ being the average number of packets in the entire period. Finally, let R be a $|\mathcal{V}| \times D$ routing matrix (corresponding to the permissible routes through the network), and let X be a D column vector that represents the unknown OD flows traversing the network. The routing matrix can be deterministic (entires are 0-1) or random (entries are probabilities); the latter refers to the case with multiple

paths in a network due to load balancing considerations. The routing matrix actually changes over time, usually on the order of a few hours (Paxson (1997)), and is typically estimated by computing shortest paths using the Interior Gate Protocol link weights that indicate congestion levels, together with known information about the network topology. An example of the traffic matrix problem is shown in Figure 2.

We can then write

$$\bar{Y} = RX.$$

The statistical inverse problem is to reconstruct the distribution of X from the aggregate level data $\bar{Y}$. In general, $D >> |\mathcal{V}|$, usually $D = \mathcal{O}(|\mathcal{V}|^2)$, so this is a highly ill-posed inverse problem and cannot be solved without additional assumptions or regularization. We provide below a review of several approaches in the literature for addressing this. See also Papagiannaki et al. (2004) for a discussion on why direct data collection and estimation of X is intractable using today's monitoring technologies.

Vardi (1996) modeled the traffic flows as Poisson, i.e., the X_j's are independent Poisson random variables with means λ_j's. The Poisson assumption provides additional estimating equations because the variance is equal to the mean, so the higher order information can be used for estimation. Vardi (1996) studied maximum likelihood estimation using the EM algorithm. However, as shown there, the EM algorithm may not converge to the MLE. More importantly, the algorithm becomes computationally intractable for large networks. Vardi (1996) studied several heuristic methods as alternatives, among which the following method-of-moments estimation was the most promising. Recall that $\bar{Y}_j,\ j = 1, ...|\mathcal{V}|$ provide $|\mathcal{V}|$ estimating equations. In addition, the sample variances and covariances $S_{ij} = \sum_{t=1}^{T}[Y_{it} - \bar{Y}_i][Y_{jt} - \bar{Y}_j]/(T-1)$ provide another $\frac{|\mathcal{V}|\times(|\mathcal{V}|+1)}{2}$ equations. Note than $E(S_{ij})$ reduces to the expected value of the variance of the counts in the shared links, and so it is again a linear function of the λ_j's. Hence, letting $\mathbf{S}$ denote a vector of the elements of the variance-covariance matrix and letting $\Lambda = (\lambda_1, ...\lambda_{|E|})^T$, where $|E|$ is the number of edges in the network, we have the linear model

$$\begin{bmatrix} E(\bar{Y}) \\ E(\mathbf{S}) \end{bmatrix} = \begin{bmatrix} R \\ B \end{bmatrix} \Lambda$$

for a suitable matrix B. Now, the data on the left-hand side are approximately normal when T is large, so we can use the resulting large-sample normal theory and weighted-least squares to estimate Λ and develop other inferential procedures. We note that Vardi proved identifiability for all practical networks. A Bayesian approach under the same framework was considered in Tebaldi and West (1998).

The goal was slightly different, dealing with prediction of the actual OD traffic counts instead of the distribution of the counts. The Poisson assumption implies that the variance of X is proportional to the mean. Cao et al. (2000) relaxed this assumption by considering a general model of the form $Var(X) \propto E(X)^{\alpha}$ and developed estimation methods.

This first generation of models do not work well in estimating the distribution of X in high speed networks as they are very sensitive to the assumptions (Poisson, normal with a specific mean-variance relationship), which did not quite hold for real network traffic data (Medina et al. (2002)). This has led to a new generation of models that employ extra information or other assumptions. The two most prominent approaches are the *tomogravity* model (Zhang et al. (2003a and 2003b)) and the method of routing changes (Soule et al. (2004)). The tomogravity model is based on the premise that the OD flow $X(i,j)$ between nodes i and j is proportional to the total amount of traffic departing node j, $X(\cdot,j) = \sum_{i \in V} X(i,j)$, and the total amount of traffic entering node i, $X(i,\cdot)$; i.e. $X(i,j) \propto X(i,\cdot) \times X(\cdot,j)$ (Zhang et al. (2003a)). This model assumes complete independence between the sources and destinations, which tends to be violated in backbone networks due to the so-called hot-potato routing policies adopted by their operators (operators offloading peering traffic at the nearest exit point). A modification of the simple tomogravity model capturing such issues was also proposed in Zhang et al. (2003a). Another modification that embedded the tomogravity model in a regularization framework was proposed in Zhang et al. (2003b), where the problem was formulated as

$$\min ||Y - RX||_2^2 + \lambda^2 K(X|X'), \text{ subject to } X > 0,$$

where X' is the solution to the traffic estimation problem under the generalized gravity model, $K(X|X')$ the Kullback-Leibler divergence measure and $||\cdot||$ denotes the L_2 norm. In contrast, the route change method (Soule et al (2004)) attempts to overcome the under-constrained nature of the problem by manipulating the link weights and thereby inducing additional routing matrices R.

A third generation of traffic matrix estimation methods incorporated temporal considerations, namely data are obtained over $t = 1, ..., T$ periods and the goal is to estimate the temporal evolution of the OD flows. Cao et al. (2000) developed two approaches for estimating the parameters of X as they evolved over time. The first was based on a a moving window with a locally time-homogeneous approach within each window while the second used a more formal temporal model. Soule et al. (2005) proposed the following state-space model:

$$Y_t = RX_t + u_t, \quad t = 1, \cdots, T \tag{2.1}$$

$$X_t = CX_{t-1} + e_t, \quad t = 1, \cdots, T, \tag{2.2}$$

where the first equation is the traditional traffic matrix model, with u_t representing measurement error, and the second equation posits both a temporal model for the underlying traffic state as well as spatial dependence between OD flows through the non-diagonal elements of the C matrix, and e_t captures the traffic system noise process. Liang et al. (2005) proposed a modification of the above model, and among other things, allowed for occasionally direct measurements for selected OD flows that aid in calibrating the parameters of the model.

The traffic matrix estimation problem has proved useful to network operators, especially for capacity planning purposes. In fact, many of the proposed techniques originated from telecommunication research groups such as AT&T and Sprint Labs and have been applied to networks involving up to 1000 nodes. The ill-posed nature of the problem requires the imposition of various modeling assumptions that in many cases have had a negative impact on the accuracy of the results (Medina et al. (2002), Roughan (2005)). The focus over the years has shifted to understanding both the temporal and spatial variability of OD flows, as attested to by the latest models. Some of the research challenges involve the quest for models that can capture more accurately the characteristics of the OD flows in today's high speed networks, fast and scalable estimation techniques and the simulation of realistic traffic matrices (Roughan, 2005).

3. Active Tomography

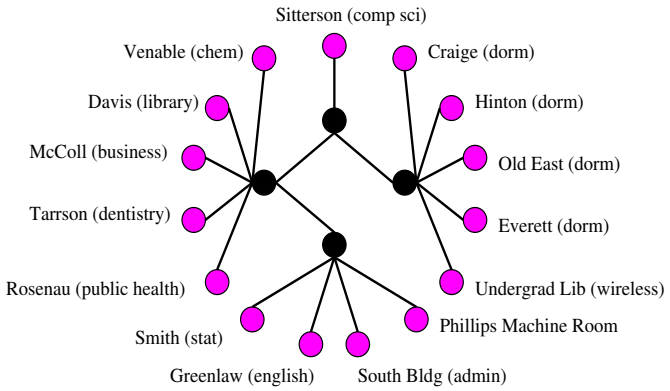

Fig. 3. The physical topology of the UNC campus network.

A second class of problems deals with estimation of the quality-of-service parameters of the network such as delays and loss rates at the individual link/router

level. This information is used to characterize and monitor the performance of the network over time, detect congestions or other anomalies in the network, and ensure compliance with service-level agreements. The difficulty and the challenge arise from the fact that many service providers do not own the entire network and hence do not have access to the internal nodes. Active tomography provides an interesting and convenient alternative by "probing" the network from nodes located on the periphery and using this to recover information about the internal links.

3.1. *Background and Probing Experiments*

To describe the details, consider the network shown in Figure 3. It depicts part of the campus network at the University of North Carolina. We will come back to a real application dealing with this network, but in this section we will use it to describe the active tomography problem. We can study the performance of the internal links of this network by sending "probe" packets from a source (in this case Sitterson) to all the other nodes on the periphery of the network (receiver nodes). Special equipment placed on the source and receiver nodes is used to send the packets and collect end-to-end information on losses and delays. The packets can be sent from the source node to the receiver(s) using a unicast or multicast transmission protocol. In a unicast protocol, the packets are sent to one receiver at a time; however, such schemes cannot estimate all the internal link-level parameters. In multicast protocols, packets are sent simultaneously to any specified set of receivers. The higher-order information in multicast schemes (performance of losses and delays in shared links) allows one to reconstruct internal link-level information. Some networks have turned off multicast transmission due to security reasons. In such situations, back-to-back unicast schemes, where packets are sent within nanoseconds of each other to two or more receivers, have been proposed in order to mimic multicast transmissions (Tsang et al., (2003)).

The logical topology for the probing experiment associated with this UNC campus network is shown on the left panel of Figure 4. This corresponds to a tree topology with source node 0 (Sitterson) at the top and receiver nodes 4, 5, 6, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, and 18 as the leaves. Note that we can observe only end-to-end measurements (0-4, 0-10, 0-15, etc.) on losses and delays, and we have to use this information to reconstruct all the internal link-level information (0-1, 1-2, 2-9, etc.). This is the inverse problem. In practice, the number of nodes can vary from about a dozen for small area networks (such as a campus network) to several hundred in wide area networks. However, the investigators can reduce the size of the network by collapsing the links (combining links and nodes) if they

are interested in just a coarse investigation of the network. A detailed study will require looking at all the nodes.

The traditional approach to probing experiments has been based on full multicast transmission where the probes are sent to all the receivers in the network (or back-to-back unicasts intended to mimic the multicast scheme). The difficultly is that this scheme is quite inflexible. One rarely wants to send the same number of probes to all the receiver nodes. Rather, we want to be able to investigate different regions of the network with different intensities and even possibly at different times. In Xi, Michailidis, and Nair (2005) and Lawrence, Michailidis, and Nair (2005a), we have proposed the use of a flexible class of probing experiments (referred to as flexicast experiments) for active tomography. This consists of $\mathcal{C} = \{C_h, N_h\}$, a collection of sub-experiments C_h with probe size N_h that lead to identifiability of all the link-level parameters. The individual sub-experiment C_h covers only part of the network and by itself cannot estimate all the parameters in the subnetwork. However, by judiciously designing the subexperiments, we can estimate all the parameters in the entire network of interest. This class of experiments is particularly useful in network monitoring where we want to study different subregions of a network depending on where anomalies, such as congestion, occur.

We have developed necessary and sufficient conditions under which the flexicast experiments lead to identifiability (estimability) of all the link-level parameters. We first need the notion of a k-cast scheme and a splitting node. In a k-cast scheme, a probe packet is sent to a specified set of k receiver nodes. It is uniquely specified by the receiver nodes. For example, $\langle 15, 16, 17, 18\rangle$ and $\langle 4, 5, 10, 11\rangle$ are two four-cast schemes for the network in the left panel of Figure 4. A splitting node, as the name suggests, is an internal node at which a k-scheme splits. For example, the four-cast scheme $\langle 15, 16, 17, 18\rangle$ splits at node 9 while $\langle 4, 510, 11\rangle$ splits at nodes 1,2, and 3.

Proposition: Let $\mathcal{C}$ be a collection of probing experiments $\{C_h, N_h\}$ and $\mathcal{T}$ be a general tree network topology. Then, all the internal link loss rates are identifiable if and only if (a) every internal node is a splitting node for some $C_h \in \mathcal{C}$ and (b) all receiver nodes are covered by $\mathcal{C}$. The same conditions are also necessary and sufficient for estimating link delay distributions provided they are discrete.

Proofs can be found in Xi et al. (2005) and Lawrence et al. (2005a). Additional conditions for identifiability of continuous delay distributions are given in Lawrence (2005).

To get some insight into the Proposition, consider the logical topology (tree) on the left panel of Figure 4. Suppose that $\mathcal{C}$ consists of the following three

subexperiments: $C_1 = \langle 4,5,6,7,8\rangle$, $C_2 = \langle 10,11,12,13,14,15\rangle$, and $C_3 = \langle 15,16,17,18\rangle$. All the receiver nodes are covered, and 2, 3, and 9 are splitting nodes for C_1, C_2 and C_3 respectively. However, the internal node 1 is not a splitting node, so this experiment will not be able to recover all the link-level parameters. A modified experiment with C_1 as before and $C_2' = \langle 10,15\rangle, C_3' = \langle 11,12,13,14,15\rangle$, and $C_4' = \langle 16,17,18\rangle$ will allow for estimation of all the parameters. Of course, there are many other ways of modifying the original experiment to get identifiability.

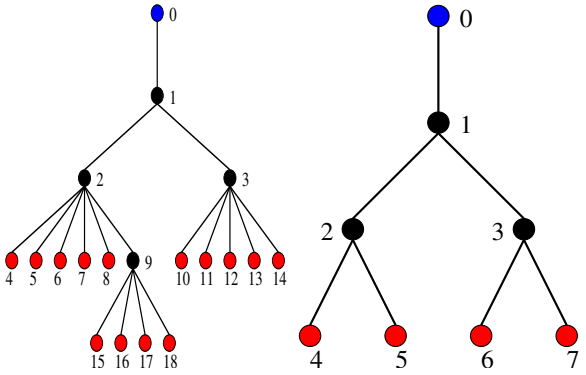

Fig. 4. The logical topology for the UNC study (left); A 3-layer symmetric binary tree (right).

This raises the question of whether there are "optimal" approaches to constructing the flexicast experiments. This is a difficult question in general, as there are many ways to define optimality. From the point of view of statistical efficiency, a multicast scheme that sends probes to all the receivers in the network is the most optimal as it provides the highest order of dependence among the shared links and hence is most informative. However, as we have already noted, it is not very flexible. Moreover, it generates a lot of probing traffic. Among the flexicast experiments, a collection of bicast (or two-cast) and unicast subexperiments has the least data complexity since the highest dimension is that of a bicast scheme which is multinomial with dimension four. For such a collection, one can find *minimal* experiments (smallest collections) that lead to identifiability of all the internal link parameters as follows: (a) For each internal node s, use exactly *one* bicast pair b whose splitting node is s; (b) Choose these bicast pairs to maximize the number of receiver nodes that are covered; and (c) Choose unicast schemes to cover the remaining receiver nodes $r \in \mathcal{R}$ that are not covered by the bicast pairs.

To understand the details, consider the three-layer symmetric binary tree in the

right panel of Figure 4. The full multicast experiment sends packets to all the seven receiver nodes and hence is a seven-dimensional multinomial experiment with 2^7 outcomes. A bicast experiment can be based on all possible pairs (21 pairs). However, a minimal experiment that can estimate all the internal links requires only three bicast pairs, for example, $C_1 = \langle 4, 5\rangle$, $C_2 = \langle 6, 7\rangle$ and $C_3 = \langle 5, 6\rangle$. This is not unique as we can replace C_3 by $C_3' = \langle 4, 7\rangle$ or several others.

The active tomography problem has been studied by several authors in the literature. The problem for loss rates was formulated in Caceres et al. (1999), where the multicast transmission scheme was also introduced and an algorithm that computes estimators that are asymptotically equivalent to the MLES was derived. The problem for delays was discussed in Lo Presti et al. (2002), where a heuristic algorithm was proposed for calculating a nonparametric estimate of the link delay distributions. Liang et al. (2003) developed a pseudo-likelihood approach for the delay problem by considering all possible pairwise probes from the full-multicast experiment. Shih et al. (2003) presented an estimator for the back-to-back probing scheme that models link delay using a point mass at zero and a finite mixture of Gaussian distributions.

3.2. *Inference for Loss Rates*

Inference for loss rates has been studied in the literature under the following stochastic framework. Let $Z_r(m) = 1$ if the m-th probe packet sent from the source node reached receiver node $r \in \mathcal{R}$, the set of all receiver nodes, and 0 otherwise. Define hypothetical random variables as follows: $X_i(m) = 1$ if packet m traverses successfully link $i \in \mathcal{E}$, the set of all links, and 0 otherwise. The collected data are analyzed under the following model (Caceres et al., (1999), Lo Presti et al. (2002), Castro et al. (2003), and others). For the loss rate problem, let $\alpha_i(m) = P(X_i(m) = 1)$, i.e., the probability that the probe packet traverses successfully the link between nodes $f(i)$ and i, and reaches node $i \in \mathcal{T} - \{0\}$. It is assumed that the $X_i(m)$'s are independent across i and m. Further, $\alpha_i(m) \equiv \alpha_i$ for all probes m (temporal homogeneity). Then, $P(Z_r(m) = 1) = \Pi_{s\in\mathcal{P}(0,r)}\alpha_s$. Further, $P(X_j(m) = 1,\ \forall\ j \in \mathcal{D}(i)) = \Pi_{s\in\mathcal{P}(0,i)}\alpha_s \times \Pi_{j\in\mathcal{D}(i)}\alpha_j$.

Some comments about these assumptions are in order. The temporal homogeneity assumption is not critical as the time frame for the probing experiment is in the order of minutes, but the effect of spatial dependence merits further study as examples using the network simulator tool and real data sets indicate.

Recall that the flexicast experiment $\mathcal{C}$ is made up of a collection of independent subexperiments C_h. Each subexperiment is a k-cast experiment (for some k), so it can be viewed as a k-dimensional multinomial experiment. More specif-

ically, each outcome is of the form $\{Z_{r_1}, ..., Z_{r_k}\}$ where $Z_{r_j} = 1$ or 0 depending on whether the probe reached receiver node r_j or not. Let $N_{(r_1,...,r_k)}$ denote the number of outcomes corresponding to this event, and let $\gamma_{(r_1,...,r_k)}$ be the probability of this event. Then the log-likelihood for the experiment C_h is proportional to $\gamma_{(r_1,...,r_k)} \log(N_{(r_1,...,r_k)}$ and that for $\mathcal{C}$ is the sum of the log-likelihoods for the individual experiments. However, the $\gamma_{(r_1,...,r_k)}$'s are complicated functions of the αs, the link-level loss rates, so one has to use numerical methods to obtain the MLEs.

This belongs to the class of missing data problems, so the EM-algorithm is a natural approach to computing the MLEs (Coates and Nowak, 2000; Coates et al., 2002; Castro el al., 2004; Xi, 2003). Xi et al. (2005) developed the structure of the EM-algorithm explicitly for flexicast experiments described above. In our experience, the EM algorithm works well when coupled with a collection of bicast and unicast experiments for small to moderate networks. For large networks, however, they are computationally not practical.

A class of fast estimation methods based on least-squares has been developed in Michailidis, Nair, and Xi (2005). This is done by transforming the loss-estimation problem to a linear inverse problem as follows. Consider the the 3-layer symmetric binary tree in the right panel of Figure 4, and suppose we use a three-cast experiment to the receivers $\langle 4, 5, 6\rangle$. There are eight possible outcomes $(1, 1, 1), (1, 1, 0), (1, 0, 0), ...(0, 0, 0)$; denote the corresponding number of the outcomes by $N_{(1,1,1)}, N_{(1,1,0)}$ and so on. We can ignore the last one as there are only seven linearly independent observations. Consider the one-to-one transformation of these seven events to the following: $(1, 1, 1), (1, 1, +), (1, +, +), ...$ where a '+' indicates either a '1' or a '0'. The new outcomes are obtained by replacing all the '0's with '+'s. Let $M_{(i,j,k)}$ denote the number of these outcomes. Now, if N_h denotes the total number of probes for the subexperiment h, we can write $E(M_{i,j,k})$ as N_h times the product of appropriate link-level α's. For instance, $E(M_{(1,1,+)}) = N_h\alpha_1\alpha_2\alpha_4\alpha_5$. Similarly, $E(M_{(1,+,1)}) = N_h\alpha_1\alpha_2\alpha_3\alpha_4\alpha_6$. This expression where the expectations are products of the appropriate link-level loss rates holds in general for any $k-$cast experiment. This suggests fitting a log-linear model to the estimated probabilities $Y^h_{(i_1,...i_k)} = M^h_{(i_1,...i_k)}/N_h$. In other words, if Y^h denotes the vector of estimated probabilities for subexperiment h, then

$$Y^h = R^h\beta^h + \epsilon^h,$$

where $\beta_j = \log(\alpha_j)$, R^h is a matrix of ones and zeros that depend on the logical topology of the subexperiment and ϵ^h is a vector of errors. The expected value of the errors tends to zero as the probe size $N_h \to \infty$. Also, the errors are correlated

in general, although the variance-covariance structure can be obtained easily due to its block-diagonal form.

Now, by stacking up the vectors of Y^hs for all the subexperiments, we get a linear system of equations that can be used to estimate all the internal link-level parameters α_js. The ordinary least-squares algorithm provides a non-iterative and very fast estimation scheme. Since the ϵ^hs are correlated, a more efficient estimation scheme is based on iteratively-reweighted LS. These and other schemes and their properties are studied in Michailidis et al. (2005). It was found that the IRWLS estimators are very close to the MLEs even in reasonably small samples. One can also compute the asymptotic variance-covariance matrix of the estimators based on the LS schemes easily, leading to explicit construction of standard errors and hypothesis tests. This is another advantage of these LS schemes over the MLEs obtained using the EM algorithm.

3.3. *Inference for Delay Distributions*

Let X_k denote the (unobservable) delay on link k, and let the cumulative delay accumulated from the root node to the receiver node r be $Y_r = \sum_{k \in \mathcal{P}_{0,r}} X_k$, where $\mathcal{P}_{0,r}$ denotes the path from node 0 to node r. The observed data are end-to-end delays consisting of Y_r for all the receiver nodes. A common approach for inference that can accommodate the heavy-tailed nature of Internet measurements is based on discretizing the continuous delays using a common bin size q. Let $X_k \in \{0, q, 2q, \ldots, bq\}$ be the discretized delay accumulated on link k where bq is the maximum delay. Let $\alpha_k(i) = \mathrm{P}\{X_k = iq\}$. Our objective is to estimate the delay distributions or the $\alpha_k(i)$'s for $k \in \mathcal{E}$ and i in $\{0, 1, \ldots, b\}$ using the end-to-end data Y_rs.

Let $\vec{\alpha}_k = [\alpha_k(0), \alpha_k(1), \ldots \alpha_k(b)]'$ and let $\vec{\alpha} = [\vec{\alpha}_0', \vec{\alpha}_1', \ldots, \vec{\alpha}_{|E|}']'$. The observed end-to-end measurements consist of the number of times each possible outcome $\vec{y}$ was observed from the set of outcomes $\mathcal{Y}^h$ for a given scheme h. Let $N_{\vec{y}}^h$ denote these counts. These are distributed as multinomial random variables with corresponding path-level probability $\gamma_c(\vec{y}; \vec{\alpha})$. So the log-likelihood is given by

$$l(\vec{\alpha}; \mathbf{Y}) = \sum_{c \in \mathcal{C}} \sum_{\vec{y} \in \mathcal{Y}^c} N_{\vec{y}}^c \log[\gamma_c(\vec{y}; \vec{\alpha})].$$

In principle, one can maximize it numerically to get the MLEs. As in the loss case, this can be viewed as an instance of a missing data problem, and the EM algorithm provides a convenient approach for computing the MLEs. This has been studied in the literature (Lawrence, Michailidis, and Nair (2003) for full multicast experiments and Lawrence et al. (2005a) for flexicast experiments). However, the

computations in the E-step are quite involved, so it can work only with very small networks for full multicast experiments. The use of the EM algorithm is more manageable when coupled with the flexicast experiments, but even then it is practical only in moderate-sized networks. There have been heuristic estimation methods that have been proposed in the literature for the full multicast situation. The first, by LoPresti et al. (2002), tries to mimic the clever algorithm for the loss case in Caceres et al. (1999) and relies on solving higher-order polynomials. However, this algorithm does not use all the data and can be very inefficient (Lawrence et al. (2005a)). Liang and Yu (2003) proposed a pseudo-likelihood method where one considers only data from all pairs of probes and ignores the third and higher order information. The all-pairs-bicasts by Liang and Yu (2003) is similar in spirit to a flexicast experiment with all pairs of bicasts, although they will all be independent in the flexicast set up. Also, as we showed earlier, one can use many fewer independent bicasts than all possible pairs to estimate the link delay parameters. Even then, the complexity of the EM algorithm grows exponentially with the number of layers in the tree.

To handle larger networks, Lawrence et al. (2005a) developed a *grafting* method which fits the EM to subtrees and uses a heuristic method based on a fixed point algorithm to combine the results across the subtrees. This is a very fast algorithm, and extensive numerical work has shown that its small sample performance is favorable compared to the estimator in Lo Presti et al. (1999) and the pseudo-likelihood estimator in Liang and Yu (2003). Further, the efficiency loss is relatively small compared to the full MLE.

Lawrence et al. (2005a) also studied inference for continuous delay distributions and developed moment-based estimation methods for the means and variances of the delay distribution assuming a mixture model for the individual link-delay distributions of the form

$$F_j(x) = p_j \delta_{\{0\}} + (1 - p_j) G_j(x).$$

Here $\delta_{\{0\}}$ denotes point mass at 0 (i.e., no delay with probability p_j) and the continuous part has a mean-variance relationship of the form $V_j = \phi \mu_j^{\theta}$ where μ_j and V_j are, respectively the mean and variance of $G_j(\cdot)$.

4. An Application

We illustrate the results using real data collected from the campus network at the University of North Carolina, Chapel Hill. The loss rates in this network were negligible, so we will focus on delays. We have collected extensive amounts of data but report here only selected results. See also Xi et al. (2005) for an application of the results to network monitoring.

Voice over IP (VoIP) or Internet telephony is a technology that turns analog voice signals into digital packets and then uses the Internet to transmit them to the intended receivers. The main difference with classical telephony is that the call does not use a dedicated connection with reserved bandwidth, but instead packets carrying the voice data are multiplexed in the network with other traffic. The quality of service (QoS) requirements in terms of packet losses and delays for this application are significantly more stringent than other non-real time applications, such as e-mail. Hence, assessing network links to ensure that they are capable of supporting VoIP telephony is an important part of the technology. The University of North Carolina (UNC) is currently in the planning phase of deploying VoIP telephony. As part of this effort, monitoring equipment and software capable of placing such phone calls were installed throughout the campus network. Specifically, the software allowed the emulation of VoIP calls between the monitoring devices. It can then synchronize their clocks and obtain very accurate packet loss and delay measurements along the network paths.

Fifteen monitoring devices had been deployed in a variety of buildings and on a range of different capacity links through the UNC network. The locations included dorms, libraries, and various academic buildings. The links included large capacity gigabit links, smaller 100 megabit links, and one wireless link. Monitoring VoIP transmissions between these buildings allowed one to examine traffic influenced by the physical conditions of the link and the demands of various groups of users. Figure 3 gives the logical connectivity of the UNC network. Each of the nodes on the circle have a basic machine that can place a VoIP phone call to any of the other endpoints. The three nodes in the middle are part of the core (main routers) of the network. One of these internal nodes, the upper router linked to Sitterson Hall, also connects to the gateway that exchanges traffic with the rest of the Internet.

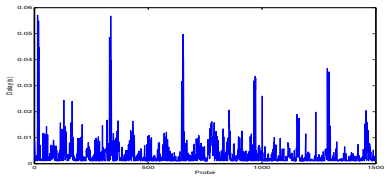

Fig. 5. Traffic trace of packet delays generated by a single phone call across the UNC network.

The data were collected using a tool designed by Avaya Labs for testing a network's readiness for VoIP. There are two parts to this tool. First, there are the monitoring devices that are computers deployed throughout the network with the

capability of exchanging VoIP-style traffic. These devices run an operating system that allows them to accurately measure the time at which packets are sent and received. The machines collect these time stamps and report them back to the second part of the system: the collection software. This software remotely controls the devices and determines all the features of each call, such as source-destination devices, start time, duration, and protocol which includes the inter-packet time intervals. The software collects the time stamps when the calls are finished and processes them. The processing consists of adjusting the time stamps to account for the difference among the machines' clocks, and then calculating the one-way end-to-end delays.

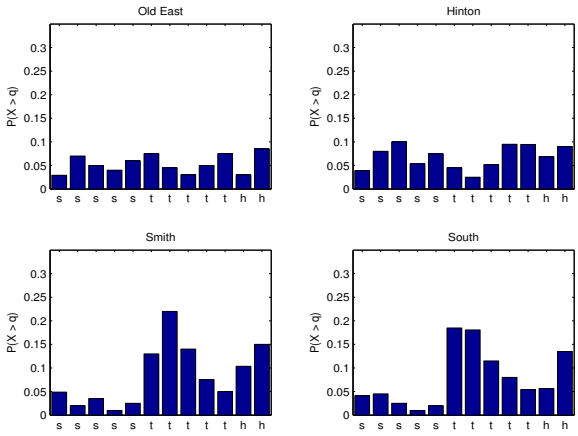

Fig. 6. Probability of large delay throughout 2/26/2005 (s), 3/1/2005 (t), and 3/3/2005 (h) at two dorms, Old East and Hinton and two university buildings, South and Smith. s, t and h, denote Saturday, Tuesday and Thursday, respectively. On the y-axis, the probability of a delay larger than 1 ms is depicted.

Figure 5 shows the delays for the packets of one phone call (flow) between two devices. The data contain information about the entire path between a pair of end-points, which spans several links. For example, a phone call placed between the dorm and the library follows a path that goes through three main routers. Many of the features found in other types of network data can be seen here: heavy-tailed marginal distributions and significant autocorrelation between consecutive observations. We will see that, by using the techniques developed in this paper, we are able to reconstruct link-level information about delays from the end-to-end path-level data.

For the data collection, Sitterson served as the root, and we used seven bicast

pairs to cover the 14 receiver nodes:

$$\mathcal{C} = \{\langle 4,5\rangle, \langle 6,7\rangle, \langle 8,10\rangle, \langle 11,12\rangle, \langle 13,14\rangle, \langle 15,16\rangle, \langle 17,18\rangle\}.$$

The network allowed only unicast transmission protocol, so back-to-back probing was used to simulate multicast transmissions. The span of time between the two packets comprising the back-to-back probe was on the order of a few nanoseconds while the time between successive probes was one tenth of a second. Prior experimentation using the call synthesis tool and this probing method leads us to believe that the correlation between the two packets on the shared links is close to one. Most of the probing sessions resulted in 200 packets to each pair (one session ran considerably longer and produced more probes due to operator error).

In this paper, we consider data collected on 2/26/2005, 3/1/2005, and 3/3/2005 (corresponding to a Saturday, Tuesday, and Thursday respectively) during the Winter semester at UNC. In addition to methodology confirmation, this data will allow us to contrast the weekend/weekday behavior of the network. The data collection is somewhat irregular, but there are five collections throughout the day on Saturday and Tuesday, and a morning and noon collection on Thursday. Analysis was conducted using the discrete delay MLE approach. The unit size was chosen as approximately 1 ms. This results in most of the mass occurring in the 'zero' bin, which gives us a useful statistic to track over time.

Figure 6 gives some selected results from this analysis. Each bar represents the probability of a delay of one unit or larger in each of four locations for each time period. Some interesting results can be noted. The university buildings South and Smith both show very few delays due to limited traffic on the Saturday collections; there are very small probabilities of delays, half of a percent or less, throughout the day. The weekdays show a typically diurnal pattern with a peak midday that tapers off. In many respects, the dorms, Old East and Hinton, show opposite patterns. The activity on the weekend is not much less than during the week. During the week, the traffic, particularly at Hinton the large freshman dorm, actually dips during the day when the students are busy with classroom activities and rises at night.

Figure 7 shows partial results from a more detailed analysis of the data. Here the unit size is about half that of the previous analysis. The plots give the first five bins on the Hinton and South links on Saturday and Tuesday. We see that the whole distribution is relatively stable on the dorm link despite the weekend/weekday difference. In the office building, we see the mass shift outward from Saturday to Tuesday, but most of the mass still falls within the first five bins indicating fairly good link performance. Additional analyses of the UNC network data can be found in Lawrence et al. (2005a, 2005b)).

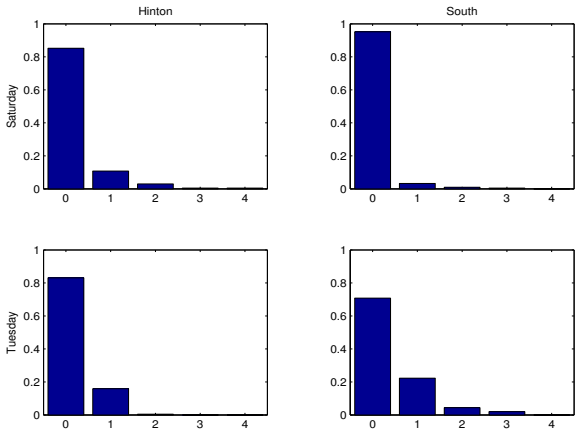

Fig. 7. Detailed distributions with unit sizes of .5 ms at Hinton and South on Saturday, 2/26, and Tuesday, 3/1, during the mid-afternoon.

5. Concluding Remarks

This paper has provided an overview of network tomography and some of the interesting statistical issues that arise from the inverse problems. The discussion so far assumed that the routing matrix (logical topology) is known or can be determined easily. This is sometimes possible as the traceroute tool, based on the Internet control message protocol (ICMP), reports all the network devices (routers and hosts) along a node. Unfortunately, many routers have disabled the protocol and do not respond to the traceroute requests. As a result, there has been a lot of interest in developing tools for topology identification. Various statistical techniques such as clustering, maximum likelihood, and Bayesian methods have been used, based on measurements from active tomography experiments (see Coates et al. (2002b), Castro et al. (2004), Rabbat et al. (2002), Shih et al. (2004) and references therein). However, the ill-posed nature of the problem and its exponential complexity make topology identification very challenging.

The tools and techniques that have been developed thus far have proved useful for characterizing network performance, detecting anomalies such as congestion, and capacity planning. Nevertheless, some of the underlying assumptions, especially those dealing with the spatio-temporal behavior, are somewhat restrictive and merit further study. Another on-going challenge (largely unaddressed in the literature) is the lack of distributed algorithms. All the proposed methods require the existence of a central data repository, which limits the applicability of tomography techniques to on-line network monitoring. Moreover, fast algorithms are

critical for implementing the techniques in real time. Therefore, useful methodology needs to strike a balance between computational complexity and statistical efficiency.

Acknowledgments: The research was supported in part by NSF grants DMS-0204247, CCR-0325571, and DMS-0505535. The authors would like to thank the following for their help on the collection, modeling, and analysis of data from the UNC network: Jim Landwehr, Lorraine Denby and Jean Meloche from the Avaya Labs, Yinghan Yang, Jim Gogan and his team from the Information Technology Division at UNC, Don Smith from CS Department at UNC, and Steve Marron from the Statistics Department at UNC.

References

1. Adler, R.J., Feldman, R. and Taqqu, M. (1998), *A practical guide to heavy tails: statistical techniques and applications*, Birkhauser, Boston
2. Cao, J., Davis, D., Vander Wiel, S. and Yu, B. (2000), Time varying network tomography: router link data, *Journal of the American Statistical Association*, 95, 1063-1072.
3. Castro, R., Coates, M., Liang, G., Nowak, R.D. and Yu, B. (2004), Network tomography: recent developments, *Statistical Science*, 19, 499-517.
4. Castro, R., Coates, M.J., and Nowak, R.D (2004), Likelihood based hierarchical clustering, *IEEE Transactions on Signal Processing*, 52, 2308-2321.
5. Coates, M.J., Hero, A.O., Nowak, R.D and Yu, B. (2002), Internet tomography, *Signal Processing Magazine*, 19, 47-65.
6. Coates, M.J., Castro, R., Gadhiok, M., King, R., Tsang Y., and Nowak, R. (2002), Maximum likelihood network topology identification from edge-based unicast measurements, *ACM Sigmetrics Proceedings*, Marina Del Rey, CA.
7. Duffield, N.G., Horowitz, J., Lo Presti, F., and Towsley, D. (2002), Multicast topology inference from measured end-to-end loss, *IEEE Transactions in Information Theory*, 48, 26-45.
8. Lawrence E. (2005), *Flexicast Network Delay Tomography*, Unpublished doctoral dissertation, Department of Statistics, The University of Michigan.
9. Lawrence, E., Michailidis, G. and Nair, V.N. (2003), Maximum likelihood estimation of internal network link delay distributions using multicast measurements, *Proceedings of the Conference on Information Systems and Sciences*, Johns Hopkins University, March 12-14, 2003.
10. Lawrence, E., Michailidis, G. and Nair, V.N. (2005a), Flexicast delay tomography (submitted).
11. Lawrence, E., Michailidis, G. and Nair, V.N. (2005b), Local area network analysis using end-to-end delay tomography, *Proceedings Large Scale Network Inference Workshop,* Banff, Canada and also to appear in *Performance Evaluation Review*
12. Liang, G. and Yu, B. (2003), Maximum pseudo-likelihood estimation in network tomography, *IEEE Transactions on Signal Processing*, 51, 2043-2053.
13. Liang, G., Taft, N. and Yu, B. (2005), A fast lightweight approach to origin destination IP traffic estimation using partial measurements, (submitted).

14. Lo Presti, F., Duffield, N.G., Horowitz, J. and Towsley, D. (2002), Multicast based inference of network internal delay distributions, *ACM/IEEE Transactions on Networking*, 10, 761-775.
15. Marchette, D.J. (2001), *Computer intrusion detection and network monitoring: a statistical viewpoint*, Springer, New York.
16. Michailidis, G., Nair V.N. and Xi, B. (2005), Fast least squares algorithms for estimating and monitoring network link losses based on active probing schemes, (preprint).
17. Papagiannaki, K., Taft, N. and Lakhina, A. (2004), A distributed approach to measure traffic matrices, *ACM Internet Measurement Conference Proceedings*, Taormina, Italy.
18. Park, K. and Willinger W. (2000), *Self-similar network traffic and performance evaluation*, Wiley Interscience, New York.
19. Paxson, V. (1997), End-to-end routing behavior in the Internet, *IEEE/ACM Transactions on Networking*, 5, 601-615.
20. Rabbat, M., Nowak, R., and Coates, M.J. (2004), Multiple source, multiple destination network tomography, Proceedings IEEE Infocom, Hong Kong.
21. Roughan, M. (2005), First order characterization of Internet traffic matrices, Proceedings ISI, Sydney, Australia.
22. Shih, M.F. and Hero, A.O. (2003), Unicast based inference of network delay distributions with finite mixture models, *IEEE Transactions on Signal Processing*, 51, 2219-2228.
23. Shih, M.F., and Hero, A.O. (2004), Network Topology Discovery Using Finite Mixture Models, Proceedings of IEEE Int. Conf. on Acoust. Speech and Sig. Proc., Montreal.
24. Soule, A., Nucci, A., Leonardi, E. Cruz, R. and Taft, N. (2004), How to identify and estimate the largest traffic matrix elements in a dynamic environment, *ACM Sigmetrics Proceedings*, New York, NY.
25. Soule, A., Lakhina, A., Taft, N., Papagiannaki, K., Salamatian, K., Nucci, A., Crovella, M. and Diot, C. (2005), Traffic matrices; balancing measurements, inference and modeling, *ACM Sigmetrics Proceedings*, Banff, Canada.
26. Tebaldi, C. and West, M. (1998), Bayesian inference of network traffic using link count data (with discussion), *Journal of the American Statistical Association*, 93, 557-576.
27. Tsang, Y., Coates, M. and Nowak, R.D. (2003), Network delay tomography, *IEEE Transactions on Signal Processing*, 51, 2125-2135.
28. Vardi, Y. (1996), Estimating source-destination traffic intensities from link data, *Journal of the American Statistical Association*, 91, 365-377.
29. Xi, B. (2004), *Estimating internal link loss rates using active network tomography*, Unpublished doctoral dissertation, Department of Statistics, The University of Michigan.
30. Xi, B., Michailidis, G. and Nair, V.N. (2005), Estimating network loss rates using active tomography, (submitted).
31. Zhang, Y., Roughan, M., Duffield, N. and Greenberg, A. (2003), Fast accurate computation of large scale IP traffic matrices from link loads, *ACM Sigmetrics Proceedings*, San Diego, CA.
32. Zhang, Y., Roughan, M., Lund, C. and Donoho, D. (2003) An information theoretic approach to traffic matrix estimation, *ACM SIGCOMM Proceedings*, Karlsruhe, Germany.