# Multiple Source, Multiple Destination Network Tomography

Michael Rabbat
Rice University
Houston, TX
Email: rabbat@rice.edu

Robert Nowak
Rice University
Houston, TX
Email: nowak@rice.edu

Mark Coates
McGill University
Montreal, QC
Email: coates@ece.mcgill.ca

***Abstract*—The problem of identifying topology and inferring link-level performance parameters such as packet drop rate or delay variance using only end-to-end measurements is commonly referred to as network tomography. This paper describes a collaborative framework for performing network tomography on topologies with multiple sources and multiple destinations, without assuming the topology to be known. Using multiple sources potentially provides a more accurate and refined characterization of the internal network. We present a novel multiple source active measurement procedure using a semi-randomized probing scheme and packet arrival order measurements which do not require precise synchronization between the participating hosts. A decision-theoretic framework is developed enabling the joint characterization of topology and internal performance. We design a statistical test based on the Generalized Likelihood Ratio Test and Wilks' Theorem. The test quantifies the tradeoff between network topology complexity and performance estimation, and identifies when measurements made by the two sources can be combined to achieve reduced variance performance estimates. The performance and efficacy of the algorithm are assessed through *ns-2* simulations and experiments over the Internet.**

***Method Keywords*— Statistics, Network measurements, Simulations, Experimentation with real networks/testbeds.**

## I. Network Tomography

Assessing and predicting internal network behavior is of fundamental importance in a variety of problems such as routing optimization, network management, and anomaly detection. However, acquiring direct internal measurements from all parts of the network is not practical due to the distributed nature of the Internet. Additionally, one cannot rely on internal network elements to respond with special purpose messages (i.e. ICMP timestamp exceeded) due to growing security concerns. Those who do have access to internal measurements are nearly always restricted from sharing the data for proprietary and privacy reasons.

For the purpose of network management, direct link-level measurements are critical for the low-level analysis of equipment conditions. However, traditional fault alarms are only triggered after failures occur, and passive measurements generate large amounts of data which are not easily processed online. Ciavattone et al. describe a practical system using active end-to-end measurements to augment traditional network operations measurements allowing them to proactively detect impairments and react quickly to performance degradation [1].

The problem of inferring internal network characteristics using end-to-end measurements is commonly referred to as *network tomography*, drawing an analogy to the medical tomography problem of imaging the internals of the human body in a non-intrusive manner [2, 3]. Active measurement techniques have been designed using both unicast and multicast measurements to estimate link-level performance parameters such as loss rate and delay variance [4–6], in addition to identifying topology [7–9].

This paper presents a study of the multiple source, multiple destination network tomography problem. Using multiple sources in the context of network tomography, it is possible to identify segments within a network shared by the paths connecting multiple sources and destinations. This information may be useful for identifying potential bottlenecks. Sharing statistics between sources may also be useful for optimizing the use of network resources when transferring large amounts of data. Additionally, in some cases it is possible to fuse information gleaned from multiple sources to get a more accurate and refined network characterization.

The majority of work in network tomography has revolved on active probing from a single source. Also, it is typical to focus on either (step 1) identifying the topology, or (step 2) estimating link-level performance parameters in which case it is assumed that the topology is known. This paper presents a multiple source active measurement procedure and a statistical framework enabling the joint characterization of topology and link-level performance. Jointly solving for performance parameters and topology leverages on the close coupling between link-level characteristics, routes derived from the network topology, and end-to-end measurements.

Inference and characterization of network properties using active end-to-end measurements is a challenging problem. Because the participating hosts are distributed across the network it is not practical to assume that they can be precisely synchronized. Additionally, labels which apply globally cannot be assigned to internal nodes by topology identification techniques employing end-to-end measurements. In general, internal nodes are only inferred relative to the single source from which measurements are made. Thus, the problem of identifying a multiple source topology amounts to more than just matching nodes with the same label. Finally, because active measurement techniques consume network resources it is desirable to minimize the amount of probing traffic required for accurate inference.

### A. Contributions

This paper focuses on the multiple source, multiple destination network tomography problem of characterizing the topology and performance on links connecting a collection of sources and destinations. The contributions are as follows.

1) It is shown that the general network tomography problem can be decomposed into a set of smaller components, each involving just two sources and two destinations. We can then focus on this special case and easily extend our results to more general multiple source, multiple destination networks.
2) We identify a dichotomy of possible two-source, two-destination topologies based on the model order of their representations.
3) A novel multiple-source probing algorithm is presented for determining the model order of an unknown two-source, two-destination topology.
4) A flexible decision-theoretic framework is developed enabling the joint characterization of topology and internal performance.
5) The efficacy and accuracy of the probing algorithm and statistical framework are evaluated through simulation. Additionally, as a proof-of-concept, the algorithm has been implemented and tested in experiments over the Internet and over the LAN of Rice University's ECE department.

### B. Related Work

In [10], Bu et al. describe a procedure for combining end-to-end multicast measurements made independently from multiple sources. They assume that the topology is known, and then extend techniques previously used with single-source measurements to infer link-level performance. A closer look at the problem of identifying a multiple source topology from end-to-end measurements reveals that this is no trivial task. This paper differs from previous work in that we formulate the multiple source, multiple destination network tomography problem without assuming that the topology is known ahead of time, and we develop new techniques accordingly.

Other related work includes the IDMaps project [11] and GNP [12]. Although neither of these projects directly aims to characterize the internal network, they both utilize active end-to-end measurements to determine the distance – typically measured in latency – between end-hosts. Distance maps are useful for growing overlay networks or multicast trees, for server selection, and in peer-to-peer file transfer applications. However, the distance maps inferred by these algorithms do not relate path characteristics for different pairs of hosts. On the other hand, the multiple source characterization described here contains information which can be used to localize loss or latency within the network. Additionally, by characterizing the internal network we are able to identify where paths from multiple sources to multiple destinations traverse a common (potential bottleneck) segment within the network.

The remainder of the paper is organized as follows. Section II describes interesting properties of multiple source topologies. These properties are the foundation for the novel probing algorithm aimed at characterizing the multiple source topology, described in Sections III-V. Then a statistical framework allowing the joint characterization of topology and performance is presented in Section VI. Section VII describes the characteristics of network topologies and traffic which affect the algorithm's performance. Results from simulation and experiments on real networks are presented in Sections VIII and IX, and we conclude in Section X.

## II. On the Structure of Multiple Source Topologies

Algorithms which use end-to-end measurements typically discuss network topology in terms of the *logical* topology since end-to-end measurements can only distinguish link boundaries by points where two paths either branch or join, and not by individual routers. No internal node in a logical topology has both in degree and out degree equal to one. Other approaches to topology identification which require special support from internal network devices, such as `traceroute` [13], are able to identify individual routers along a single path, and thus more accurately describe the physical topology. Thus, there is a tradeoff between using measurements requiring special internal network support which infer a more detailed description of network topology, and using end-to-end techniques which require no special internal network support but only identify the logical topology. However, in the context of network tomography, where the goal is to characterize internal performance using end-to-end measurements, the logical topology sufficiently describes connectivity between the participating hosts.

ICMP-based techniques such as `traceroute` and those based on `traceroute` come with their own problems. Barford et al. report in their 2000 study that $13\%$ of routers in the Internet do not respond to these special purpose messages [14]. It is anticipated that this number will only increase with rising security concerns. Additionally, there is the problem of identifying routers which respond with different addresses on different interfaces. These unsolved problems are beyond the scope of this work, but they motivate the development of alternative methods for identifying topology. Also, while techniques based on `traceroute` are limited to discovering layer-3 devices, it is possible to map a layer-2 network using end-to-end measurements as demonstrated later in this work. Thus, topology identification techniques using end-to-end measurements may also be used to fill in the gaps where other procedures leave off.

### *A. Decomposing Multiple Source Networks*

This work specifically focuses on characterizing the internal network (topology and link-level performance) using end-to-end measurements made from multiple sources. Much of the previous network tomography work has utilized pair-wise measurements made from a single source. In this case, the topology takes the form of a tree. The intuition behind pair-wise measurement schemes is illustrated in the following example where one source sends packets to two destinations. All packets sent from the source traverse an initial common segment until the reach a branching point, where the paths to each destination split. Suppose two packets are sent back-to-back from the source, with one packet going to each destination. Queuing events experienced by both packets before they reach the branching point are highly correlated since the packets are traveling back-to-back through the common set of queues, or very close to each other. Queuing events occurring downstream from the branching point are uncorrelated since by that point the packets are separated. These correlated observations can be used to infer loss and delay on the shared and unshared links.

Pair-wise measurements of this sort are a common building block in many network tomography algorithms [5, 6, 15]. Ratnasamy and McCanne first introduced the idea in the multicast context for building a tree topology [7]. Duffield et al. later proved that the topology inferred using this type of measurement indeed corresponds to the maximum likelihood solution for the topology given a set of multicast measurements [9, 16]. Empirical evidence also suggests that in the unicast situation methods based on pair-wise comparisons give accurate results with high confidence. The condition of failure for these algorithms is when the weight on one link in the tree is zero. In this case the inferred topology is a *metric induced* topology [17]. Such topologies do not directly reflect the underlying physical or logical topologies, but rather they reflect the logical topology induced by the metric used to reconstruct the topology.

In the context of multiple sources, the analogous building block is the simplest multiple-source, multiple-destination topology – that composed of two sources and two destinations (a 2-by-2 component). The idea is that all internal nodes in a logical topology are either points where paths from multiple sources to a common destination join (a *joining point*), or where paths from a single source to multiple destinations branch (a *branching point*). Each 2-by-2 component – composed of the logical topology and performance parameters associated with each logical link – contains information about at least one joining point and one branching point. Having the 2-by-2 component information for every pair of sources and destinations is sufficient for reconstructing a general multiple source, multiple destination ($M$-by-$N$) network. Please see [18] for further discussion. Thus, by solving the 2-by-2 network tomography problem we have effectively solved the more general $M$-by-$N$ problem.

### *B. Shared vs. Non-Shared 2-by-2 Topologies*

Previous end-to-end measurement schemes utilizing a single source have been based on the assumption that the underlying topology takes the form of a tree [7–9, 16]. Indirectly, this basis assumes that routes between each source and destination are unique. Following this same line of reasoning, there are four possible 2-by-2 topologies as depicted in Figure 1.

One could further decompose any 2-by-2 network into two single-source, two-destination (1-by-2) components and two dual-source, single-destination (2-by-1) components. However, in order to make measurements on the 2-by-1 components analogous to the back-to-back packet pair measurements made on the 1-by-2 components it is necessary to transmit packets from each source so that

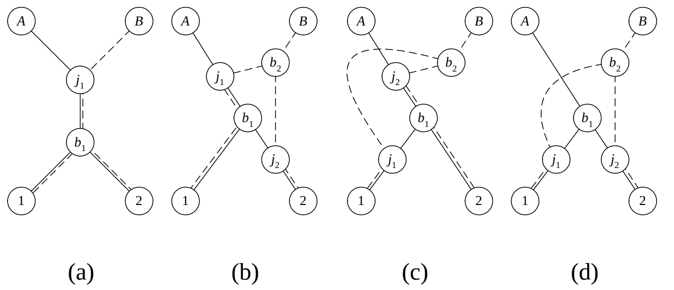


Fig. 1. Four possible topologies for a two source, two destination network. Nodes $A$ and $B$ are sources, and nodes 1 and 2 are destinations. The flow of traffic is directed downward on links, but arrowheads are omitted to avoid cluttering. The shared topology in (a) has one joining point ($j_1$) and one branching point ($b_1$), and fewer links over all. The non-shared topologies shown in (b), (c), and (d) each have distinct joining points for paths to each destination. As a result more links are required, so there are more degrees of freedom in the model. This is the basis for the dichotomy of shared and non-shared topologies.

they are correlated (back-to-back) on the common downstream link to the destination. Measurements of this sort are impractical, as one would need to have precise synchronization among hosts, knowledge of internal delays, and no competition from cross-traffic in order to transmit packets such that they arrive simultaneously at the first shared router. Rather than give up here, we ask ourselves, *"What can we infer about a 2-by-2 topology without the 2-by-1 component measurements?"*

We begin by distinguishing among the four 2-by-2 topologies based on the model order of each topology. The *shared* topology depicted in Figure 1(a) has two internal nodes and five links, and the *non-shared* topologies depicted in Figures 1(b-d) each have four internal nodes and eight links. The multiple source probing algorithm described in this paper is designed to identify the model-order of a 2-by-2 network.

Without these measurements on the 2-by-1 components, there is no clear way to further distinguish between the three non-shared topologies. However, for the purpose of network tomography where our goal is to infer link-level performance characteristics, there is not much to be gained by differentiating among the non-shared cases. Each non-shared topology has the same model order, and existing techniques for inferring performance – namely, back-to-back packet probes – do not allow us to achieve better results in the non-shared case than if each source acted independently. On the other hand, we can take advantage of topological properties in the shared case.

Link-level performance estimates are typically generated by averaging over the outcomes of multiple measurements, and it is a well-known fact that the variance of averaging estimators is inversely proportional to the number of measurements used. Using existing techniques, each source is able to characterize performance on the logical links extending from the branching point, $b_i$, to each destination. For the shared topology, these logical links are identical for both sources $A$ and $B$, and so measurements can be averaged to produce better estimates (i.e. when measurements from two sources are averaged the variance of the estimate is reduced by $1/2$).

Now, one of the drawbacks to using active measurement techniques is that network resources are consumed in the measurement process which would otherwise not be used. Keeping this in mind, next we pose the question, *"How does probing collaboratively from multiple sources affect the amount of probe traffic required?"* We find that by probing collaboratively from multiple sources it is possible to procure more information without requiring any more measurements than would be used if the sources were to transmit probes independently.

## III. Collaborative Probing from Multiple Sources

This section describes the multiple source measurement algorithm developed in this work. Measurements are developed to exploit differences between the shared and non-shared topologies. Sources transmit packets in a semi-randomized fashion, and destinations record packet arrival order. Because neither of these operations require precise time synchronization between any of the participating hosts the algorithm is easy and practical to implement.

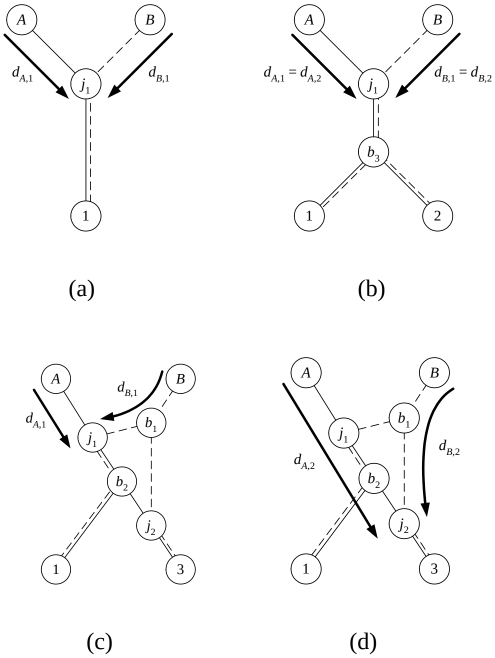
Initially, to facilitate in explaining the algorithm we make the following idealistic assumptions:

1) there is no cross-traffic in the network so that there is no variability in delay along any link,
2) sources are precisely synchronized,
3) routes between end-hosts are unique, and
4) packets do not get reordered within the network.

These assumptions are relaxed/justified in Section IV.

### A. Packet Arrival Order

Consider the "Y" shaped topology which describes the routes from sources $A$ and $B$ to destination 1 as shown in Figure 2(a). Under the assumptions listed above, if $A$ and $B$ both transmit a packet to 1 at some time $t_0$, then the packets arrive at 1 in the same order they arrive at joining point $j_1$. More precisely, let $d_{A,1}$ denote the delay incurred by packets traveling from $A$ to $j_1$, and let $d_{B,1}$ denote the delay incurred by packets traveling from $B$ to $j_1$. The order in which packets arrive at destination 1 indicates the sign of the quantity $\delta_1 = d_{B,1} - d_{A,1}$. That is, if the packet from $A$ arrived at the destination first then $\delta_1 > 0$, and if the packet from $B$ arrived first then $\delta_1 < 0$. Thus, this notion of packet arrival order is directly related to the difference in delays incurred from the sources to the joining point.



(a) (b)

(c) (d)

Fig. 2. (a) Packet arrival order at the destination is the same as the order in which they arrive at the joining point, $j_1$. This order is determined by the delays incurred by packets traveling from the sources to the joining point. (b-d) Shared and non-shared topologies are depicted with delays to each joining point labeled. The joining point in the shared topology is common for paths to both destinations. Joining points to each destination are unique in the non-shared topology. The collaborative multiple source probing algorithm hinges on this idea to identify whether a topology is shared or not.

The shared topology is unique in that there is only one joining point. That is, the joining point in the shared topology is shared by paths going to both destination 1 and destination 2. Thus, when packets are transmitted by the sources they pass through this common joining point regardless of which destination the packets are destined for. On the other hand, in any of the non-shared topologies, there are two joining points; one joining point for packets going to destination 1 and the other for packets going to destination 2. Figures 2(b-d) display delays from the sources to each joining point for shared and non-shared topologies. The collaborative multiple source probing algorithm exploits this feature using packet arrival order measurements to distinguish between shared and non-shared topologies.

### B. Multiple Source Probes

The basic multiple source probe is as follows. At time $t_0$, source $A$ sends a pair of packets spaced apart by $\Delta$ seconds with the first packet headed for destination 1 and the second packet going to destination 2. The space between packets is chosen to be sufficiently large so that the inter-packet spacing is not affected by differences in bandwidths on upper and lower links of the topology. Specifically, $\Delta > packetsize/b_{min}$, where $b_{min}$ is the minimum bandwidth of all links in the 2-by-2 network. This criterion ensures that the packets will traverse the network independently.

Source $B$ sends packets in a similar configuration, but with a random offset introduced between the transmit times of corresponding packets. That is, if source $A$ transmits packets at times $t_0$ and $t_0 + \Delta$, then source $B$ transmits packets at times $t_0 + u$ and $t_0 + \Delta + u$ to destinations 1 and 2 respectively, where $u$ is a random variable distributed uniformly over the interval $[-D, +D]$ and $D$ is much larger than $\Delta$. These four packets constitute a single probe. By sending repeated measurements and varying the offset $u$ over a range of values, the difference in delays to each joining point is indirectly measured. An example of such a series of probes is depicted in Figure 3.

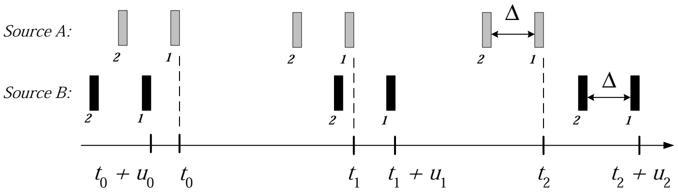


Fig. 3. This figure depicts a series of probes. The inter-packet spacing, $\Delta$, is chosen to be large enough so that queuing events affecting the two packets are independent. The offset variables $u_i$ are independent random draws, uniformly distributed over the interval $[-D, +D]$, and $D$ is much larger than $\Delta$ in practice.

Let $\alpha_1 = \pm 1$ denote the packet arrival order at destination 1, with $\alpha_1 = +1$ indicating that the packet from source $A$ arrived before the packet from source $B$ and $\alpha_1 = -1$ indicating that the packet from source $B$ arrived first. Similarly, let $\alpha_2$ denote packet arrival order at destination 2. The arrival order at destination one can be written as

$$\begin{aligned} \alpha_1 &= \text{sign}((t_0 + u + d_{B,1}) - (t_0 + d_{A,1}), \quad (1) \\ &= \text{sign}(\delta_1 + u), \quad (2) \end{aligned}$$

with a similar expression for $\alpha_2$, the arrival order at destination 2. Define the arrival order statistic to be

$$z = 1\{\alpha_1 \neq \alpha_2\}, \quad (3)$$

where $1\{\cdot\}$ is the indicator function. Thus, $z$ takes value 1 only when the arrival order at each destination is different. In terms of modeling, $z$ is a Bernoulli random variable with a parameter $\rho$ which quantifies the probability of observing different arrival orders at each destination.

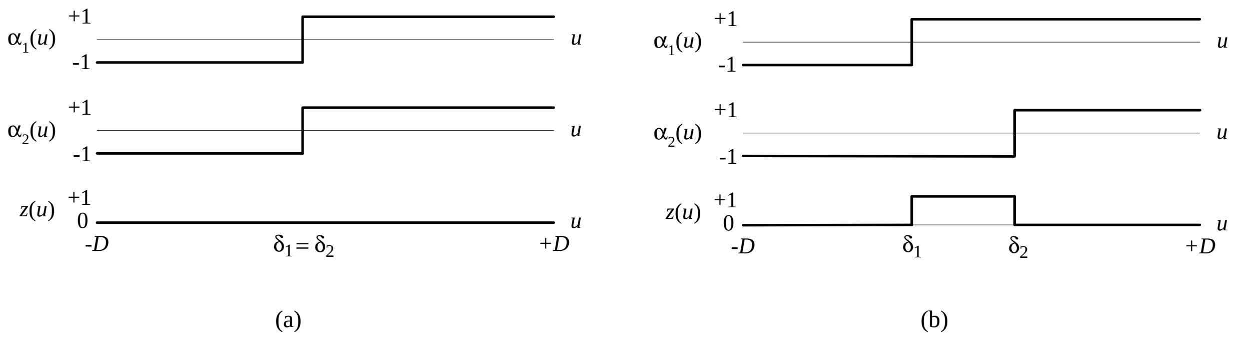


Fig. 4. Displaying $\alpha_1$, $\alpha_2$, and $z$ as functions of the random offset $u$ for both (a) shared and (b) unshared topologies. The arrival order statistic, $z$, is modeled as a Bernoulli random variable with parameter $\rho$, and an estimate of this parameter is used to determine whether a topology is shared or not.

Observe from Fig. 2(b) that for the shared topology, $\delta_1 = \delta_2$. Disregarding the effects of cross-traffic, it is always true that $\alpha_1 = \alpha_2$, and thus $z = 0$ for a shared topology. The random offset, $u$, determines the sign of each $\alpha_i$, but the outcomes at each destination are identical so $u$ has no effect on the value of $z$.

On the other hand, for any non-shared topology, it is unlikely that the delay differences to each destination, $\delta_1$ and $\delta_2$, are the same. Then, for a certain range of offset values, the packet arrival order will be different at the two destinations. Thus, the random offset, $u$, acts as a mechanism for exploring the behavior of an unknown 2-by-2 network. Figure 4 depicts the relationship between the $\alpha_i$, $z$, and $u$ for both shared and non-shared topologies. This illustration makes clear the point that for non-shared topologies arrival orders will be different at each destination for a certain range of $u$.

In practice, probes are sent at a frequency $1/T$, with the offset taking different values $u^{(1)}, u^{(2)}, u^{(3)}, \ldots, u^{(n)}$ for each probe. The probing period $T$ is chosen to be large enough that the experiences of each probe are statistically independent. In our experiments we set $T$ to twice the maximum round-trip time (RTT) for any source-destination pair. Assuming the probes have independent experiences and that the queuing distribution is stationary for the duration of an experiment ($\sim 2-5$min.), the $z^{(i)}$ are independent and identically distributed Bernoulli random variables.

The probing procedure samples the function $z(u)$ in a random fashion. Then, keeping track of $z^{(1)}, z^{(2)}, \ldots, z^{(n)}$, one can calculate $\hat{\rho} = \frac{1}{n}\sum_i z^{(i)}$, an estimate of the probability of observing different arrival orders at each destination. For shared topologies, the estimate $\hat{\rho} = 0$, and for non-shared topologies then $\hat{\rho} > 0$. Section VI describes a more precise formulation of this decision process in terms of a statistical hypothesis test.

## IV. Relaxing Assumptions

### A. Cross-Traffic Distorts Packet Spacing

In reality, cross-traffic in the network induces a random queuing delay on each link. This is accounted for by modeling the delays from sources to joining points as a random process, $d_{S,R}(t)$. Queuing due to cross-traffic has the effect of distorting the spacing, $\Delta$, between packets in each probe. Consequently, the probability of observing different arrival orders at each destination is no longer zero for the shared topology, but should be some small value due to queuing delay. For non-shared topologies, queuing "blurs the edges" of the region between $\delta_1$ and $\delta_2$, where different arrival order observations occur.

To gauge whether the mechanism inducing different arrival order events is just cross-traffic (shared) or a combination of cross-traffic and topological characteristics (non-shared), a modified probing measurement is developed which measures the percentage of different arrival order events due to cross-traffic alone. Similar to the probes described in Section III-B, each source sends two packets spaced by time $\Delta$, with the same timing and random offset as before, only that all four packets in the probe are transmitted to a single destination. For a single-destination probe of this form, a different arrival order event occurs when the arrival order of the first packets sent from each source is different from the arrival order of the second packets. Such an event may occur if the spacing $\Delta$ is distorted by queuing.

Let $\rho_1$ denote the probability that a different arrival order event occurs when all packets are sent to destination 1. Define the following packet arrival order observations for these measurements.

$$\begin{aligned} \alpha_1' &= \text{sign}(d_{B,1}(t_0+u) - d_{A,1}(t_0) + u) \\ \alpha_1'' &= \text{sign}(d_{B,1}(t_0+\Delta+u) - d_{A,1}(t_0+\Delta) + u \end{aligned}$$

The arrival order statistic $z_1 = 1\{\alpha_1' \neq \alpha_1''\}$, is a Bernoulli random variable with parameter $\rho_1$ describing the probability that cross-traffic will affect the arrival order for packets going to destination 1. For a set of $n$ measurements, we calculate $\hat{\rho}_1 = \frac{1}{n}\sum_i z_1^{(i)}$. This estimate reflects the percentage of different arrival order events due to queuing on the links leading from each source to joining point $j_1$, for the paths leading to destination 1.

A similar experiment is performed, but with all four packets going to destination 2. These single-destination experiments mimic conditions under the shared topology, with both sets of packets going through a single joining point. An estimate, $\hat{\rho}_2$, is obtained that reflects the amount of queuing on the links leading from each source to joining point $j_2$, for the paths leading to destination 2.

Now, when the topology is shared, there is only one joining point, so every different arrival order event is due to queuing. Because the joining point is the same for the paths to either destination, $\rho_1 = \rho_2$. Additionally, $\rho_1 = \rho_2 = \rho$ when the topology is shared since the only mechanism causing different arrival orders in the shared case is queuing along the paths to the one joining point.

When the topology is not shared, the joining points are different to each destination as are the paths from each source to the joining points. In this case, queuing behavior may be different for links to each joining point so it is not necessarily true that $\rho_1 = \rho_2$. For non-shared topologies, $\rho$ should be larger than $\rho_1$ and $\rho_2$ since it is also expected that different arrival order events will be observed in the experiment involving both destinations due to the different mean delays from sources to each joining point.

Thus, intuitively, when $\hat{\rho}_1$, $\hat{\rho}_2$, and $\hat{\rho}$ are very similar we declare that the topology is shared. When $\hat{\rho}$ is significantly larger than the other two estimates we conclude that the topology is not shared. A formal decision procedure is developed in Section VI.

### B. *Dealing With Coarse Source Synchronization*

A major advantage to using packet arrival order measurements is that no precision timing infrastructure is required to make the measurements. The destinations only need to record the order in which packets arrive. It is not practical to assume that a precision timing infrastructure will be in place between the sources, either. It is practical, however, to assume that the sources can achieve a coarse awareness of each others relative time though a handshaking mechanism. To this extent, we expect that the sources will be able to synchronize to within 5-10 milliseconds of each other at the beginning of an experiment.

The time difference between source clocks can be characterized as a constant offset and a difference in rate. Letting $\tau_A(t)$ and $\tau_B(t)$ denote each source's perception of time, set $\tau_B(t) = \beta\tau_A(t) + \kappa$. Without loss of generality, let $\tau_A(t) = t$. Suppose that the probes are sent at some frequency $1/T$, so that source $A$ ideally sends the first packet in each probe at times $t_0, t_0 + T, t_1 + T, \ldots$. Note that $T$ can be set as large as desired, and typically we choose $T = 2D$. Rewriting (2), we find that the expression for the $k^{th}$ arrival order at destination $r$ is

$$\alpha_r(k) = \text{sign}(d_{B,r} - d_{A,r} + (u + \kappa + k\beta T)).$$

Thus, we can think of discrepancies in relative source clocks purely in terms of their effect on the distribution of random offset variable $u$. The constant offset, $\kappa$, acts as an initial offset so that on the first probe ($k = 0$), $u$ is drawn uniformly from $[-D + \kappa, D + \kappa]$. Then the rate difference, $\beta$, shifts the uniform random offset interval by $T\beta$ at each probe transmission. As long as $\delta_1, \delta_2 \in [-D + \kappa + k\beta T, D + \kappa + k\beta T]$ for every $k$ then the probability of observing a different arrival order event on any individual trial is the same. Thus, by choosing the parameter $D$ sufficiently large we see that synchronization discrepancies in the form of a constant offset and rate difference in the source clocks have no effect on our collaborative multiple source measurement procedure.

On the other hand, the size of $D$ is inversely proportional to performance. Intuitively, the feature we are taking advantage of in relation to Figure 4 is the area between $\delta_1$ and $\delta_2$ where $z(u) = 1$. The values of $\delta_1$ and $\delta_2$ are determined by the network topology (transmission delays) and by current network conditions (mean delay), thus $\delta_1$ and $\delta_2$ are essentially fixed for the duration of an experiment. Intuitively, the larger we make $D$, the smaller $|\delta_1 - \delta_2|/2D$ becomes. The number of probes required to achieve a given level of estimator accuracy is inversely proportional to $2D$. Therefore, there is a tradeoff between making $D$ large enough to account for the lack of precise synchronization, and minimizing $D$ to reduce the amount of network resources consumed. Observe that

$$|\delta_i| < \max(d_{A,i}, d_{B,i}) < \max(\text{RTT}_{A,i}, \text{RTT}_{B,i}). \quad (4)$$

That is, the delay difference to a given destination is bounded from above by the maximum RTT from a source to a receiver. In the experiments and simulations reported, we choose $D$ to be

$$D = \max_{S\in\{A,B\},\, R\in\{1,2\}} (\text{RTT}_{S,R}), \quad (5)$$

the maximum round-trip time from a source to destination, and find that we are able to make accurate inferences with a reasonable number of probes.

### C. The Effects of Packet Reordering and Multiple Paths

The assumption that paths between a source and destination are unique is motivated by the fact that the majority of routers in the Internet use routing tables to determine next hops based off of destination addresses. Zhang, Paxson, and Shenker report that Internet routes typically remain stable for many hours [19], suggesting that routing tables are not apt to change rapidly. While it is possible that they may change at any time, by restricting our measurement periods to be roughly five minutes in duration it is less likely that routes will change during an experiment.

Additionally, the assumption was made that packets do not get reordered within the network. In a recent study on packet reordering in IP networks, Bellardo and Savage conclude that the probability of two packets being reordered as they travel through the network is highly correlated with the space in time between the packets as they traverse through the network [20]. The probability of reordering decreases dramatically as the space between packets increases. Specifically, packets traveling more than 200 microseconds apart experience reordering with probability less than 0.01. Thus, unless two packets arrive at a joining point within 200 microseconds of each other, it is safe to assume that their ordering will be preserved. In the collaborative multiple source probing algorithm described above, packets will occasionally arrive at a joining point very close to each other. These probes are the very same ones which are susceptible to the effects of queuing. Accordingly, we group the effects of reordering with the random delay due to queuing, and together these effects are treated as noise.

It is possible that load balancing may be employed on the network of interest. In this case incoming traffic is transmitted over two or more paths in parallel in order to reduce the load on any part of the network. This situation violates our assumption that paths between the source and destination are unique, and may be a cause of packet reordering. Because our algorithm uses end-to-end measurements, the load-balanced links carrying probe traffic will appear as a single virtual link, and the inferred performance characteristic for the virtual link will reflect the average behavior across all links in the load balancing system.

## V. Incorporating Performance Measurements

This section briefly describes how a slight modification to the probe structure described in Section III allows us to make dual-destination measurements (Sec. III-B), single-destination measurements (Sec. IV-A), and measurements of performance all using the same probing structure. Then we can combine these measurements to jointly characterize topology and link-level performance. This is achieved by modifying the original probes (Figure 3) so that each packet in the probe goes to *both* destinations. If multicast packets are being used then no modification needs to be made since each packet effectively is transmitted to all destinations in the multicast group. When unicast packets are being used the modification is made by replacing each single packet with a back-to-back packet pair. Many single-source active probing techniques have been developed using back-to-back probes or stripes of many back-to-back packets to infer link-level performance parameters such as loss rate and delay variance [4–6, 15, 21]. The resulting probe structure is depicted in Figure 5.

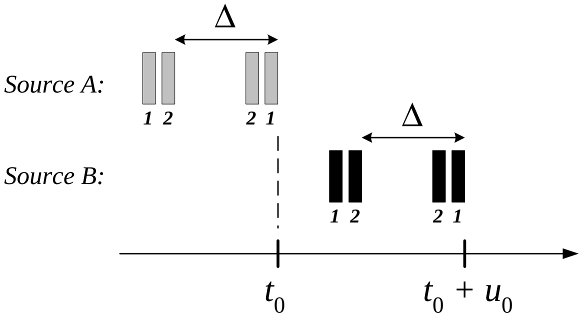


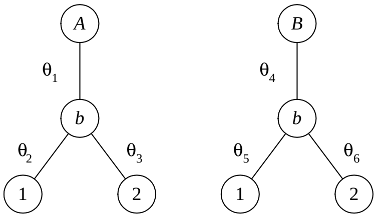
Fig. 5. Modifi ed multiple source probes. Each rectangle represents a packet and the numbers beneath each rectangle indicate the packet's destination. In the unicast setting we replace each packet with a back-to-back packet pair in order to acquire measurements which can also be used to estimate link-level performance. In each back-to-back packet pair, one packet goes to destination 1 and the other to destination 2. Back-to-back packets are used to measure link-level performance because their experiences are highly correlated on parts of their paths before the branching point.

With this type of probe structure, one can look at the arrival order of the first pair of packets at destination 1 and the second pair of packets at destination 2 to get dual destination measurements. Alternatively, by comparing the arrival orders of both pairs of packets arriving at destination 1 (or all arriving at destination 2) one gets a single destination measurement. Finally, the outcomes of packets within a back-to-back probe can be used to estimate internal performance parameters. Thus, by having the sources collaborate and by adding structure to the back-to-back probes, the resulting set of measurements can be used to infer more information than if the two sources had independently employed an active measurement scheme using back-to-back probes.

## VI. Decision-Theoretic Framework

This section describes a statistical framework and hypothesis test for deciding whether or not the topology of a 2-by-2 network is shared. The framework is flexible, taking as inputs either arrival order measurements, delay variance measurements, loss measurements, or any combination thereof. When multiple sets of measurements are used (e.g. arrival order and loss), the test jointly solves for the topology characterization and performance estimates. Due to constraints on the length of this paper, we limit our discussion to the case where arrival order measurements and loss measurements are both used. For a complete outline of the framework please see [18].

Suppose the sources send $N$ probes. Each destination keeps track of packet arrival order and loss. Let $\boldsymbol{z}$ denote the set of arrival order measurements and let $\boldsymbol{y}$ denote the set of loss measurements for an experiment. Denote by $\theta_1, \ldots, \theta_6$ the link-level loss rates, corresponding to links as depicted in the two 1-by-2 networks in Figure 6.



(a) (b)

Fig. 6. Two 1-by-2 components which comprise the 2-by-2 problem. We would like to estimate the link-level performance parameters, $\theta_1, \ldots, \theta_6$, averaging the estimates from each source when the topology is shared.

Let $H_S$ denote the hypothesis that the 2-by-2 topology is shared, and let $H_N$ denote the hypothesis that the topology is not shared. Let $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_6)$ denote the general six-dimensional vector of loss rates, and let $\boldsymbol{\rho} = (\rho, \rho_1, \rho_2)$ denote the three dimensional vector of different arrival order probabilities.

Under each hypothesis the joint likelihood function is written as $p(\boldsymbol{y}, \boldsymbol{z}|H_i, \boldsymbol{\theta}, \boldsymbol{\rho})$. A decision is made by choosing the hypothesis which maximizes the likelihood given the observations. We factor the likelihood function into

$$p(\boldsymbol{y}, \boldsymbol{z}|H_i, \boldsymbol{\theta}, \boldsymbol{\rho}) \quad = \quad p(\boldsymbol{y}|H_i, \boldsymbol{\theta})\; p(\boldsymbol{z}|H_i, \boldsymbol{\rho}), \qquad (6)$$

implying that the loss measurements and arrival order measurement are statistically independent. Independence follows from the assumption that the inter packet-pair spacing, $\Delta$, is large enough that queuing effects experiences by the first and second back-to-back packet probes sent from each source are independent, as described in Section III-B.

Now, the true parameters $\boldsymbol{\rho}$, $\boldsymbol{\theta}$ are unknown variables. We take the generalized likelihood ratio test (GLRT) approach to solving this composite hypothesis problem. In the GLRT, the unknown distribution parameters $\boldsymbol{\theta}$ and $\boldsymbol{\rho}$ are replaced with their maximum likelihood estimates under each model. Under $H_N$, we have $\boldsymbol{\theta} \in [0,1]^6$ and $\boldsymbol{\rho} \in [0,1]^3$. On the other hand, under $H_S$, the model order of the network is reduced. Consequently, the parameter space is restricted so that $\theta_2 = \theta_5$, and $\theta_3 = \theta_6$. Thus, under $H_S$, we have $\boldsymbol{\theta} \in [0,1]^4$ and $\boldsymbol{\rho} \in [0,1]^1$. The GLRT can be written as

$$\Lambda(\boldsymbol{y}, \boldsymbol{z}) = \frac{\max\limits_{\boldsymbol{\theta} \in [0,1]^6, \boldsymbol{\rho} \in [0,1]^3} p(\boldsymbol{y}|H_N, \boldsymbol{\theta}) p(\boldsymbol{z}|H_N, \boldsymbol{\rho})}{\max\limits_{\boldsymbol{\theta} \in [0,1]^4, \boldsymbol{\rho} \in [0,1]^1} p(\boldsymbol{y}|H_S, \boldsymbol{\theta}) p(\boldsymbol{z}|H_S, \boldsymbol{\rho})}. \qquad (7)$$

Then a decision is made according to

$$\Lambda(\boldsymbol{y}, \boldsymbol{z}) \underset{H_S}{\overset{H_N}{\gtrless}} \eta, \qquad (8)$$

for some threshold $\eta$. When the likelihood ratio is greater than the threshold, the test declares that the topology is not shared. Otherwise the test declares it is shared.

In general, setting a threshold for the GLRT is a difficult task when no uniformly most powerful test exists and when *a priori* probabilities are not available for each hypothesis. However, for the composite hypothesis test as formed above, a threshold can be set using Wilks' Theorem for the asymptotic behavior of the log likelihood ratio statistic [22]. Let $\lambda(\boldsymbol{y}, \boldsymbol{z}) = 2 \log \Lambda(\boldsymbol{y}, \boldsymbol{z})$. Then under mild assumptions about the regularity of the likelihood functions $p(\boldsymbol{y}|H_i, \boldsymbol{\theta})$ and $p(\boldsymbol{z}|H_i, \boldsymbol{\rho})$ – which are satisfied in our case – Wilks' Theorem states that under the shared (null or restricted) hypothesis, $\lambda(\boldsymbol{y}, \boldsymbol{z}) \xrightarrow{d} \chi^2_\nu$, where $\nu$ is the difference in the number of degrees of freedom under each hypothesis. In other words, using loss and arrival order measurements, $\lambda(\boldsymbol{y}, \boldsymbol{z})$ converges in distribution to a chi-squared random variable with four degrees of freedom under $H_S$. By knowing the distribution of the log likelihood ratio statistic under the shared hypothesis it is possible to determine a threshold, $\eta$, by setting the probability of mistakenly declaring that a topology is not shared when it is really shared (Type I error). For example, to have a Type I error rate of $2\%$ set $\eta = 0.429$.

## VII. Characteristics of Internet Topologies and Traffic Affecting Performance

As described above, Wilks' Theorem tells us how to set a threshold based on choosing the Type I error rate. In general, it is difficult to precisely quantify the error rate when the true topology is not shared (Type II error) because it depends on the magnitude of diversity between the delay differences on links, and these are parameters we do not know. In this section we offer an intuitive explanation of the characteristics of Internet topologies and traffic which will affect performance in the non-shared scenario. In the next section we further evaluate the performance through simulation.

We begin by relating the problem when the true topology is shared to the classic signal-in-noise detection problem. We would like to decide whether or not the topology is not-shared given a set of noisy measurements. The signal is the “bump” region of offsets between $\delta_1$ and $\delta_2$ where different arrival order events are observed. Noise takes the form of queuing due to cross-traffic which can both cause different arrival order events and same arrival order events where they would otherwise not occur. In such a problem, the error rate is usually parameterized by a signal-to-noise ratio, with performance improving as this ratio increases.

Signal power is related to the width of the bump squared, $(E|\delta_1 - \delta_2|)^2$. Each $\delta_i$ is a difference in delays along two paths to the same receiver. The larger the bump, the stronger the signal, the easier it is to detect. If there is a large variation in mean path delay – for instance, due to the geographic locations of different hosts – then it is very likely that the region between the $\delta_i$ will be wide. Noise power, on the other hand, can be written as $var(\delta_1 - \delta_2)$. This quantity describes the variance due to cross-traffic. The more bursty the background traffic, the stronger the noise. Thus the Type II error rate of our algorithm depends on mean path delay which is related to the topology, and the traffic property, delay variance.

Placing a distribution on these properties is not a simple task, as they can vary greatly depending the scale of the network considered, geographic location of hosts, time of day, and so on. Our intuition tells us, however, that in general path lengths will vary greatly for reasons of geography, and that in most places the network infrastructure is over-provisioned, so that queuing delay is relatively low. In a study of round-trip delays, Acharya and Saltz report that there is large temporal and spatial variation in RTTs, but that jitter in RTT observations is small [23]. This result seems to favor a strong signal-to-noise ratio. It is also possible that a situation could occur where the network is relatively homogeneous, with transmission delays more or less the same between every source-destination pair. If this were the case and if cross-traffic were extremely bursty then performance would be degraded. However, it should be noted that the signal-to-noise ratio can always be improved by taking more measurements.

## VIII. Simulation Results

Next, we evaluate our multiple source algorithm using the *ns-2* simulator [24]. Both loss and arrival order measurements were used in the simulation. Packet delays and losses are due to congestion as probes compete with cross-traffic. Following [10], infinite TCP flows produce the majority of the background traffic, as TCP is the dominant transport protocol on the Internet. A few exponential on-off flows are also included, with the over all mix of background traffic such that link-level loss rates vary between $0.01\%$ and $2\%$. Probes in the simulation are composed of multicast packets.

The simulated topology is depicted in Figure 7. Note the bi-directional flow of probe traffic on one link. The 2-by-2 networks for destination pairs $(1, 2)$ and $(3, 4)$ are shared, and those for all other pairs of destinations are non-shared. The simulation was repeated 500 times, with different random seeds. Each trial consists of 1000 probes transmitted over 200 simulated seconds. All settings were chosen to reflect a realistic scenario.

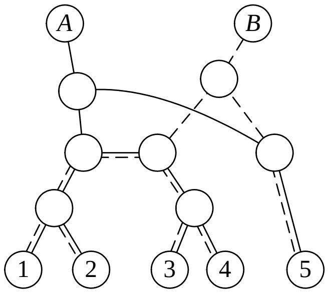


Fig. 7. Simulated topology. Solid lines indicate the paths taken by probes from source $A$ and dashed lines indicated the paths taken by probes from source $B$. The 2-by-2 networks for destination pairs $(1, 2)$ and $(3, 4)$ are shared, and those for all other pairs of destinations are non-shared.

Figure 8 depicts a histogram of the values taken by the joint log likelihood ratio, $\lambda(\boldsymbol{y}, \boldsymbol{z})$, using 1000 loss and arrival order measurements when the true topology was shared. According to Wilks’ Theorem, the values taken by this function should asymptotically be distributed according to a chi-squared random variable with four degrees of freedom. The chi-squared distribution is shown as a solid line for reference. The histogram conforms fairly well to the distribution, so we are reassured that Wilks’ asymptotic result indeed holds when at least 1000 probes are used.

Next we assess the performance of our algorithm. Figure 9 shows a plot of the Type I error rate versus one minus the Type II error rate. This type of plot is sometimes referred to receiver-operator characteristics, or ROC curves. Note that the origin is in the upper left-hand corner of the figure, and that the indices on each axis range between $0$

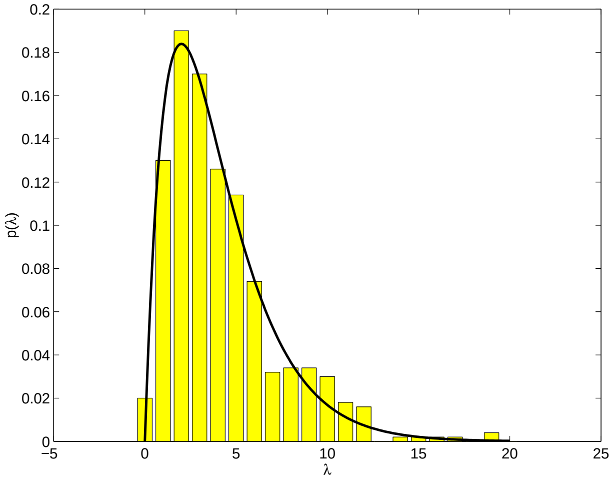


Fig. 8. Histogram of joint log likelihood ratio values where the true topology was shared. According to Wilks' Theorem the joint log likelihood ratio should have a chi-squared distribution with four degrees of freedom. The solid line corresponds to this reference distribution.

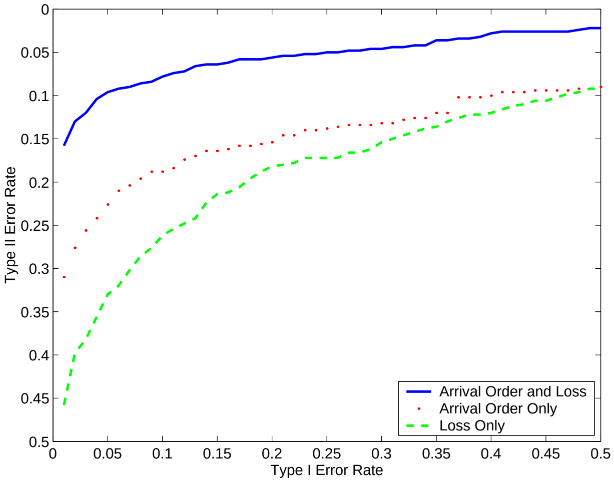


Fig. 9. Type II versus Type I error rates for detectors using both loss and arrival order measurements, only loss measurements, and only arrival order measurements. We choose a value for the Type I error in order to set the threshold, and the resulting Type II error is depicted along the $y$-axis. The joint detector (using arrival order and loss measurements) exhibits the best performance.

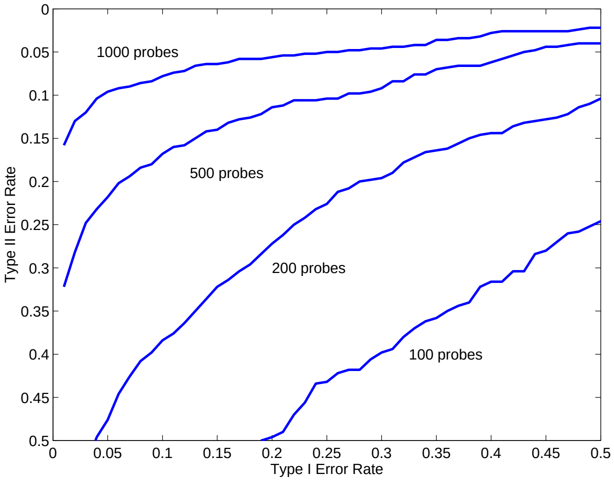


Fig. 10. This figure depicts ROC curves for the joint arrival order, loss detector. Different curves show the performance achieved as the number of probes is increased from 100 to 1000. The performance quickly increases as the number of probes goes up, especially for low Type I error rates, where we would prefer to operate.

and $0.5$. Three curves are shown for the cases when only arrival order measurements are used, only loss measurements are used, and both arrival order and loss measurements are used. In order to set a threshold for the statistical test, we choose the Type I error rate (along the $x$-axis). The resulting Type II error rate is depicted along the $y$-axis. Ideally, we would like these curves to go through the top left corner, where there is no error of either type. Note that the detector using combined arrival order and loss measurements outperforms both the loss only and arrival order only detectors.

Each of the curves depicted in Figure 9 show results for when 1000 probes are used. Next, we analyze the performance by varying the number of measurements fed in to the algorithm. Figure 10 depicts the ROC curve for the joint detector, varying the number of probes used by the algorithm. As expected, the Type II error rate decreases quickly as the number of probes increases. When 1000 probes are used, it is possible to achieve a Type II error rate as low as $10\%$ with the Type I error rate at $5\%$. Thus, it is possible to achieve desirable performance using a moderate number of probes.

## IX. Internet Experiments

As a proof-of-concept, we have implemented the multiple source probing algorithm using UDP probes and tested it in two diverse settings. In both experiments only arrival order measurements were used. The first setting consists of a collection of hosts scattered around the Internet. The two sources were located in Montreal, Quebec, and Houston, Texas. Destinations were situated in Portugal, Illinois, Wisconsin, and Michigan, and both Berkeley and San Diego, California. This configuration offered examples of both shared and non-shared 2-by-2 topologies. Results were verified against topologies obtained using `traceroute`. In this experiment we were able to successfully characterize each 2-by-2 network using 1000 probes per destination pair.

The second set of experiments were performed using 18 hosts on an operational LAN at Rice University. Results were validated with assistance from the network administrators. It should be noted that the topology connecting these hosts is mainly composed of layer-2 devices, with only a single layer-3 router spanning the LANs in different buildings. Using only end-to-end measurements, the algorithm was able to correctly determine shared/non-shared topology characteristics for each pair of destinations. We believe that the positive results of these two experiments indicate the strength and versatility of the multiple source probing algorithm described here.

## X. Conclusion and Discussion

Multiple source topologies can be decomposed in to 2-by-2 networks, thus by solving the 2-by-2 problem we have essentially solved the $M$-by-$N$ problem. The possible 2-by-2 networks can further be broken down into shared and non-shared classes based on their model order (number of links and nodes). There are two main reasons we are interested in this dichotomy. If the topology is shared then measurements can be combined from both sources to achieve reduced variance estimates of link-level parameters on the downstream links. Additionally, when the topology is shared then we have more information about topology (namely some information about the placement of joining points) than we would have if each source had actively probed without collaborating.

Packet arrival order is determined at the first shared queue. This was the basis of our multiple source probing algorithm. Main highlights of the algorithm include the fact that precise synchronization is not required, either multicast or unicast packets can be used, and no more packets are required than would have been used if the sources probed without collaborating even though we know more at the end of the day. Because the algorithm is founded on a principle directly related to topology, namely that the arrival order of packets is determined at the joining point – the algorithm is robust to cross-traffic and can operate effectively under a variety of conditions.

## References

[1] L. Ciavattone, A. Morton, and G. Ramachandran, "Standardized active measurements on a tier 1 ip backbone," *IEEE Comm. Mag.*, June 2003.

[2] Y. Vardi, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 365–377, March 1996.

[3] M. Coates, A. Hero, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Processing Magazine*, May 2002.

[4] R. Cáceres, N. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal loss characteristics," *IEEE Transactions on Information Theory*, vol. 45, pp. 2462–2480, November 1999.

[5] M. Coates and R. Nowak, "Network loss inference using unicast end-to-end measurements.," in *ITC Seminar on IP Traffic, Measurement, and Modeling*, Monterey, CA, Sep. 2000.

[6] A. Bestavros, K. Harfoush, and J. Byers, "Robust identification of shared loss using end-to-end unicast probes," in *Proc. IEEE Conf. Network Protocols*, Osaka, Japan, Nov. 2000, *Errata* available as Boston University CS Tech. Report 2001-001.

[7] S. Ratnasamy and S. McCanne, "Inference of multicast routing trees and bottleneck bandwidths usin end-to-end measurements," in *Proceedings of IEEE INFOCOM 1999*, New York, NY, March 1999.

[8] M. Coates, R. Castro, and R. Nowak, "Maximum likelihood network topology identification from edge-based unicast measurements," in *Proceedings of ACM Sigmetrics*, Marina Del Rey, CA, June 2002.

[9] N.G. Duffield, J. Horowitz, F. Lo Presti, and D. Towsley, "Multicast topology inference from measured end-to-end measurements," in *ITC Seminar on IP Traffic, Measurement, and Modeling*, Monterey, CA, September 2000.

[10] T. Bu, N. Duffield, F. Lo Presti, and D. Towsley, "Network tomography on general topologies," in *Proceedings of ACM Sigmetrics*, Marina Del Rey, CA, June 2002.

[11] P. Francis, S. Jamin, C. Jin, Y Jin, D. Raz, Y. Shavitt, and L. Zhang, "Idmaps: A global internet host distance estimation service," *IEEE/ACM Transactions on Networking*, October 2002.

[12] T.S.E. Ng and H. Zhang, "Predicting internet network distance with coordinates-based approaches," in *Proceedings of IEEE Infocom*, New York, NY, June 2002.

[13] *traceroute – a tool for printing the route packets take to a network host. http://ee.lbl.gov/traceroute.tar.Z.*

[14] P. Barford, A. Bestavros, J. Byers, and M. Crovella, "On the marginal utility of network topology measurements," in *Proc. of IMW*, San Francisco, CA, Nov. 2000.

[15] N.G. Duffield, F. Lo Presti, V. Paxson, and D. Towsley, "Inferring link loss using striped unicast probes," in *Proceedings of IEEE Infocom*, Anchorage, Alaska, April 2001.

[16] N.G. Duffield, J. Horowitz, F. Lo Presti, and D. Towsley, "Multicast topology inference from measured end-to-end loss," To appear in IEEE Transaction in Information Theory.

[17] A. Bestavros, J. Byers, and K. Harfoush, "Inference and labeling of metric-induced network topologies," Tech. Rep. BUCS-TR-2001-010, Computer Science Department, Boston University, Boston, MA, May 2001.

[18] M. Rabbat, "Multiple source network tomography," M.S. thesis, Rice University, Houston, TX, May 2003.

[19] Y. Zhang, V. Paxson, and S. Shenker, "The stationarity of internet path properties: Routing, loss, and throughput," Tech. Rep., ACIRI, May 2000.

[20] J. Bellardo and S. Savage, "Measuring packet reordering," in *Proceedings of the ACM Sigcomm Internet Measurement Workshop*, Marseille, France, November 2002.

[21] N.G. Duffield and F. Lo Presti, "Multicast inference of packet delay variance at interior network links," in *Proceedings of IEEE Infocom 2000*, Tel Aviv, Israel, March 2000.

[22] S.S. Wilks, "The large-sample distribution of the likelihood ratio for testing composite hypotheses," *Annals of Mathematical Statistics*, March 1938.

[23] A. Acharya and J. Saltz, "A study of internet round-trip delay," Tech. Rep. CS-TR-3736, U. Maryland, Jan. 1996.

[24] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heideman, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in network simulation," *IEEE Computer*, May 2000.