RICE UNIVERSITY

Multiple Source Network Tomography

by

Michael G. Rabbat

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

MASTER OF SCIENCE

Approved, Thesis Committee:

_____

Robert Nowak, Chair
Associate Professor of Electrical and Computer
Engineering

_____

Mark Coates
Assistant Professor of Electrical and Computer
Engineering, McGill Univeristy

_____

Edward W. Knightly
Associate Professor of Electrical and Computer
Engineering

_____

Rudolf Riedi
Faculty Fellow in Electrical and Computer
Engineering

Houston, Texas

May, 2003

ABSTRACT

Multiple Source Network Tomography

by

Michael G. Rabbat

Assessing and predicting internal network performance is of fundamental importance in problems ranging from routing optimization to anomaly detection. The problem of estimating internal network structure and link-level performance from end-to-end measurements is called *network tomography*. This thesis investigates the general network tomography problem involving multiple sources and receivers, building on existing single source techniques. Using multiple sources potentially provides a more accurate and refined characterization of the internal network. The general network tomography problem is decomposed into a set of smaller components, each involving just two sources and two receivers. A novel measurement procedure is proposed which utilizes a packet arrival order metric to classify two-source, two-receiver topologies according to their associated model-order. Then a decision-theoretic framework is developed, enabling the joint characterization of topology and internal performance. A statistical test is designed which provides a quantification of the tradeoff between network topology complexity and network performance estimation.

Acknowledgments

Many thanks go to my advisor, Dr. Robert Nowak, for his enthusiasm, patience, and friendship which have made working on this project a pleasurable and educational experience. I would also like to thank the members of my thesis committee: Dr. Mark Coates, Dr. Edward Knightly, and Dr. Rudolf Riedi. Their comments and feedback have been very useful for refining this thesis. Special thanks go to Dr. Coates for suggesting that I investigate the multiple source network tomography problem and arrival order probing for my thesis.

I also want to express my appreciation to the members of the Digital Signal Processing group and of the Signal Processing in Network group at Rice University for the many friendships which have developed in my time here at Rice. It has been a pleasure and honor to work among so many great minds. Finally, my heartfelt gratitude goes out to my family for their never ending encouragement and support.

Contents

List of Figures

Chapter 1

Network Tomography

Developing techniques which use end-to-end performance measurements to assess network performance is an increasingly important task as the size and complexity of the Internet continue to grow. Assessing and predicting network behavior is of fundamental importance for a variety of problems such as routing optimization, replica placement, and anomaly detection. End-user applications such as streaming multimedia or video conferencing use loss and jitter to determine the appropriate service level. In order to assess the current conditions of a network, measurement-based techniques are employed.

This thesis investigates the task of using end-to-end measurements to characterize the inter-connectivity and internal performance of a network connecting multiple sources to multiple receivers. No prior knowledge of the internal network is assumed. Similar to previous techniques, the general multiple-source, multiple-receiver problem is approached by decomposing it into smaller component problems. However, the methodology adopted in this thesis differs from existing techniques in that the network inter-connectivity and internal performance are jointly characterized rather than being treated separately.

A possible approach to assessing internal network performance is to directly monitor traffic on each internal network link. However, one cannot depend on internal network elements to freely transmit vital network statistics such as traffic rates, link delays, and packet loss rates. Routers already bear the burden of managing large amounts of incoming traffic across multiple outgoing links at very high data rates. The added cost of processing and communicating performance-related statistics on demand makes monitoring the network from within an impractical approach. Additionally,

although internal monitoring is performed on select links, privacy and proprietary concerns prevent that data from being shared. Thus, the decentralized nature of the Internet makes quantitative assessment of internal network performance from within the network very difficult [8].

Developing techniques to assess internal network performance using only end-to-end measurements is an attractive and viable alternative to making internal measurements. End-to-end measurements rely only on readily available processing capabilities at the edges of the network and thus do not require special purpose support from internal network elements.

The task of inferring internal network performance using end-to-end measurements is called *network tomography*. Vardi coined the term in 1996, drawing an analogy to the medical tomography problem of imaging the internals of the human body in a non-intrusive manner [25]. Figure 1.1 captures the spirit of the network tomography problem, with a collection of hosts scattered around the perimeter of the network. The network is depicted as a cloud to represent the notion that no information is known about the current internal network state. In general, paths exist between each end-host but the inter-connectivity and performance characteristics of internal portions of the network are not readily available.

Using network tomography makes it possible to infer the network topology and to assess internal network performance. The topology is composed of links and nodes, describing the way in which packets are routed from one end-host to another and how paths from different sources to different receivers overlap. Typically, performance is described in terms of link-level packet drop rate, ECN marking rate, delay distribution, available bandwidth, or a related statistic. The problems of identifying network topology and inferring link-level performance are typically viewed as disjoint, so they are usually treated in a step-wise fashion, first determining or assuming the topology and then assessing performance. The methodology developed in this thesis differs in that a unified approach
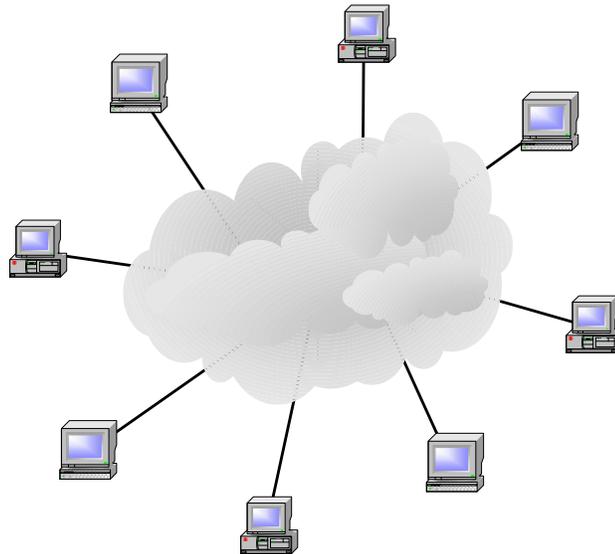
Figure 1.1 This figure shows a collection of end-hosts around the perimeter of the network cloud. A cloud is used to represent the fact that internal network characteristics such as the topology and internal performance are transparent to the end-user. Network tomography algorithms seek to characterize the internal network using end-to-end measurements only. The two tasks in network tomography are to identify the topology connecting a collection of end-hosts, and to infer link-level internal performance parameters such as packet drop rate or queueing delay distribution.

is taken such that topology and internal performance are characterized jointly.

End-to-end measurement techniques can be classified according to the manner in which measurements are acquired. Passive techniques observe already existing network traffic, and thus consume no extra bandwidth. They look for naturally occurring packet sequences which provide information about the state of the network and then correlate these across multiple observers. The main drawback of using passive techniques is the lack of measurement flexibility, since the patterns of interest may rarely occur. On the other hand, active techniques make measurements by sending packet probes between the participating hosts. These techniques center on designing probing algorithms to gauge a specific network property. Although most probing algorithms aim to be economical, active measurements still expend network resources. Thus, the tradeoff between using active and passive methods is a matter of flexibility versus resource consumption. This thesis considers active probing

techniques using multiple collaborating sources.

Using multiple sources in network tomography provides a more accurate and refined characterization of the internal network. Most work in active tomography to date has focused on the special case involving a single source transmitting probes to multiple receivers, leading to a tree-like topology. In contrast, this thesis investigates the general network tomography problem involving multiple sources and receivers. Again, making the analogy to medical tomography, by projecting through the body at more angles it is possible to resolve the image in more detail than if each source acted independently. Additionally, most existing single source active techniques in network tomography average a series of measurements to obtain link-level performance estimates. As the variance of this type of estimator is inversely proportional to the number of data points averaged, the variance of these averaging estimates can be reduced by pooling measurements made from multiple sources.

The next section presents an overview of related work in active network tomography. Then, the contributions of this thesis are formally stated.

1.1   Active Probing Methods

In single source active measurement techniques, the paths from the source to the receivers form a tree structure with the source at the root of the tree and receivers as the leaves. Active probing techniques are designed to measure a specific network property. The multicast transport mechanism was identified early on as being well suited for active probing [1, 5, 16]. Each multicast packet sent from the source is replicated whenever there is a new branch in the tree. Consequently, when a packet gets dropped or queued on a certain link, all receivers descended from that link will observe the effect of the loss or queueing behavior.

Ratnasamy and McCanne first demonstrated that these correlated drop observations could be

used to reconstruct the multicast topology and to infer the link-level loss rates [22]. Duffield et al. then rigorously established the correctness of this algorithm and developed a framework under which other metrics such as delay variance and ECN marking rate could be used in place of loss rate for inferring the topology [12, 13, 15].

Motivated by the lack of support of multicast protocols over the entire Internet infrastructure, and because the majority of Internet traffic uses the unicast transport mechanism, researchers then developed a series of measurement tools using the unicast mechanism. Many of these techniques utilize packet pair measurements to infer loss [9, 17, 19, 21] and delay [11, 14] distributions. In these measurements, packets are sent back-to-back and each packet is destined for a different receiver. Much like the correlation experienced by multicast packets, back-to-back packets are highly correlated on shared links before the paths to each receiver branch apart. Thus, the main difference between techniques using unicast and multicast probes is that the unicast measurements are made only to pairs of receivers at a time, whereas each multicast packet is transmitted to all of the receivers. Attempting to emulate this behavior, Duffield et al. proposed the use of stripes rather than packet pairs, where each packet in the stripe is sent to a different receiver [17]. However, as the length of the stripe grows, correlation is weakened throughout the stripe. Long stripes are also much more intrusive and prone to disrupt other network traffic than packet pairs are.

Researchers have also investigated the problem of identifying a single source unicast topology using special-purpose probes [3, 6, 7]. The probes in these algorithms use a collection of different sized packets sent to two receivers, noting that larger packets have a longer transmission time than smaller packets. They then employ a complexity reducing hierarchical statistical model to reconstruct the tree topology.

Whereas all previously mentioned techniques had only utilized a single source, the recent work

of Bu et al. combines measurements made independently from a collection of multicast sources [4]. Assuming the multiple source topology to be known, they establish conditions which guarantee that all links of interest are identifiable from end-to-end measurements. Furthermore, they present two algorithms for estimating loss from the combined set of measurements. However, a closer look at the problem of identifying a multiple source topology from end-to-end measurements reveals that this is no trivial task. In this thesis the multiple source tomography problem is therefore framed in more general terms. Topological and performance-related aspects are both studied without assuming any prior information.

Other related work includes the IDMaps project [18] and Global Network Positioning [20]. Although neither of these projects aims to infer internal network performance, they both utilize active end-to-end measurements to determine the distance between end-hosts and make these distances available to users. In both projects, distance is related to the latency experienced by packets travelling between two hosts which is not necessarily related to geographical distance. Such information is useful in applications such as server selection and peer-to-peer file transfer, where multiple servers or peers may all hold a desired file and the user would like to download the file from the closest provider. The scheme proposed in the IDMaps project utilizes multiple *Tracers* distributed throughout the Internet, actively calculating their distance to other hosts. Then, when an end-host needs to know its distance to another host in the Internet it queries the closest Tracer, similar to the way that one's web client queries a web server to download a web page [18].

On the other hand, in Global Network Positioning (GNP) each host is assigned a coordinate value in n-dimensional Euclidean space. A user can then calculate its distance to any other host just by acquiring the other host's coordinates [20]. Distances are calculated through a triangulation technique, from measurements made to a collection of reference hosts who already know their

coordinates.

Although both projects use end-to-end active measurements to measure the distance between hosts, these projects differ greatly from the work presented in this thesis because they only characterize internal metrics on the path level. That is, distance captures properties of the end-to-end path between hosts, but it does not relate these paths to each other in any way for different pairs of hosts. On the other hand, because network tomography algorithms infer internal connectivity in addition to characterizing internal performance, it is possible to assess the relationship between hosts from a topological perspective as well as a performance perspective. Link-level performance metrics such as delay variance and packet drop rate combine to give the end-to-end performance, therefore the information inferred by network tomography algorithms can be used to locate loss or latency within the network in addition to determining the distance between end-hosts. Thus, network tomography algorithms provide more information than algorithms which only infer the distance between hosts.

1.2 Contribution

This thesis focuses on the general network tomography problem involving multiple sources and multiple receivers. The problem is framed in general terms, assuming no prior information about the network. 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 receivers.** The single source network tomography problem can be solved by considering smaller components – from the source to pairs of receivers – one at a time. Similarly, I demonstrate that the multiple-source, multiple-receiver problem can be decomposed into two types of components: one involving a single source and two receivers, and the other involving two sources

and a single receiver. Knowledge of internal performance parameters for each of these components then suffices to reconstruct the multiple source topology (or the identifiable portion of it) and to infer performance on each link of the multiple-source, multiple-receiver topology. The simplest multiple-source, multiple-receiver problem, namely that with two sources and two receivers, comprises a pair of single-source, two-receiver components, and a pair of two-source, single-receiver components. Focus is then shifted from the more general multiple-source, multiple-receiver problem to the two-source, two-receiver problem, because this specific problem captures the challenges inherent in the more general problem.

2. **A dichotomy of possible two source, two receiver topologies into *shared* and *non-shared* classes is established, based on the model-order complexity of their representations.** Current probing schemes based on back-to-back packet pairs are capable of measuring link-level properties on single-source, two-receiver components. In order to pool these measurements and reduce estimator variance, one must be able to identify common links from different components. Of the possible two-source, two-receiver topologies, this identification is possible only for a certain class of them where links in each component are shared. As such, this class is termed the shared class, and the class for which this identification, and thus measurement pooling, is not possible is termed the non-shared class. The model-order of the shared topology class, in terms of the number of link-level parameters to be estimated, is lower than that of non-shared topologies.

3. **A novel multiple-source probing algorithm is proposed for identifying the class of an unknown two-source, two-receiver topology.** The algorithm uses packet arrival order measurements which are easy to make and practical since they do not require any special mea-

surement infrastructure. Arrival order measurements are also robust as they are not subject to measurement noise at the receiver. Sources collaborate in probing to a pair of receivers without being synchronized. Probes are composed of packet pairs, which can also be used for performance estimates. Because a single-source algorithm would use this same type of measurement, no more bandwidth is consumed by the multiple source algorithm than would be if the sources were to probe the network individually. Thus, just by having sources collaborate it is possible to extract more information from the same type of measurements.

4. **Finally, a decision-theoretic framework is developed to enable the joint characterization of topology and internal performance.** The problem is formulated as a composite hypothesis test where the true parameters governing the link-level performance and arrival order distributions are unknown. Then, using the Generalized Likelihood Ratio Test and invoking Wilks' Theorem, a statistical hypothesis test is designed which provides a quantification of the tradeoff between network topology complexity (model-order), and network performance estimation. A threshold for the test can easily be set by specifying the probability of mistakenly deciding that a topology belongs to the non-shared class when in truth it is shared.

The remainder of this thesis is organized as follows. Chapter 2 frames the general multiple-source, multiple-receiver network tomography problem, specifically assuming no prior knowledge of the network. The chapter first reviews the decomposition of single-source topologies into single-source, two-receiver components. It then presents an extension for decomposing the multiple source, multiple receiver case into single source, two receiver components and two source, single receiver components. Chapter 3 focuses on the two-source, two-receiver problem. It establishes a dichotomy of possible two-source, two-receiver topologies, classifying each topology as either

shared or non-shared based on the model-order of the topology. Then, the multiple source probing algorithm is developed in order to distinguish between these cases, and a theoretical analysis of the algorithm is performed. Chapter 4 presents the framework for combining arrival order and back-to-back packet measurements. Results from model-based simulations illustrate the performance of the algorithm under a variety of conditions. Finally, Chapter 5 summarizes the findings of this thesis and discusses avenues for future work in the area of multiple source network tomography.

Chapter 2

The Problem With Multiple Sources

This chapter compares the single-source and multiple-source network tomography problems in terms of their decompositions in to smaller components. Investigators previously studying the single-source network tomography problem have identified a decomposition of single-source, multiple-receiver (1-by-N) network tomography problems into single-source, two-receiver (1-by-2) components. In this chapter, it is shown that the multiple-source, multiple-receiver (M-by-N) network tomography problem can be decomposed in to 1-by-2 and 2-by-1 components. The two-source, two-receiver (2-by-2) network tomography problem is then identified as the simplest multiple-source, multiple-receiver problem which also captures all of the complexity of the general M-by-N network tomography problem. Thus, solving the 2-by-2 problem is a key step towards solving the general M-by-N problem.

In the context of algorithms which use end-to-end measurements, network topology is always discussed 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. In other words, each node is either the the first common ancestor of two receivers or the first common descendent of two sources. Other approaches which require special support from internal network devices, such as `traceroute` [24], are able to identify individual routers along a single path. As a result, the topologies obtained using `traceroute` more accurately reflect the true physical topology. However, in the context of network tomography, where performance is also being inferred using end-to-end measurements, the logical topology sufficiently describes the relevant path information

for a given set of end-hosts. 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. For the purposes of solving the network tomography problem, the logical topology provides a sufficient characterization of the topology, and using end-to-end measurements remains more in line with the measurement philosophy adopted in this work.

When discovering topology using end-to-end measurements, a standard set of assumptions is made. First, it is assumed that a unique path exists between each source and each receiver at any given point in time and that these paths are stationary over reasonable periods of time. The assumption that paths are unique is justified by the fact that the most routers in the Internet forward packets based on the destination address listed in the packet header, so packets with the same destination always get forwarded along the same route. Results reported by Zhang, Paxson, and Shenker indicate that Internet routes typically remain stable for many hours [27]. Additionally, it is assumed that the paths from a source to different receivers do not rejoin after they have branched and that paths from different sources to a given receiver do not branch after they have joined. This assumption is violated when load balancing is implemented over a portion of the network being studied. In this case, techniques using end-to-end measurements group the multiple links over which the load is balanced into one virtual link. The performance estimate associated with the virtual link reflects the average performance across the load balancing links. Finally, it is assumed that network events (loss and delay) are spatially uncorrelated, that packets are temporally uncorrelated when spaced far enough apart, and that the distributions of these events are stationary over time periods lasting a few minutes. Stationarity is also addressed by Zhang et al., and they report that the stationarity of loss observed over periods of minutes often lasts for at least one hour.

The remainder of this chapter is organized as follows. First, the decomposition of the single source (1-by-N) network tomography problem into 1-by-2 components is reviewed. Examples illustrate that if the 1-by-N topology is known, then performance parameters can be inferred on links of the 1-by-N tree topology when performance parameters are known on links of the 1-by-2 components, and that if the 1-by-N topology is unknown, then it can be inferred from 1-by-2 components. It is then shown that a similar decomposition exists for the M-by-N network tomography problem. Examples again illustrate that topology identification and link-level performance labelling can be performed using the 1-by-2 and 2-by-1 components.

2.1 Decomposing the Single Source Problem

Figure 2.1 depicts the single source network tomography problem with a single source transmitting probes to multiple receivers though an uncharacterized network cloud. The two tasks in this problem are to identify the topology connecting the source and receivers and to infer performance parameters associated with each link of the topology. The topology in single source network tomography problems is tree structured, and all internal nodes of the topology are points where paths from the source branch to two or more receivers. Such nodes are termed *branching points*.

Many existing single source network tomography algorithms utilize *pairwise* measurements made on each of the 1-by-2 components to solve the 1-by-N problem [8–11, 14, 19, 21]. The phrase "pairwise measurements" is used to emphasize that measurements are made to pairs of receivers (1-by-2) at a time. When the topology is known, pairwise measurements suffice to infer performance parameters along the links of the 1-by-N tree, and when the topology is not known it can be inferred from the 1-by-2 components.

As an example of a 1-by-N network tomography problem, consider the 1-by-3 network depicted
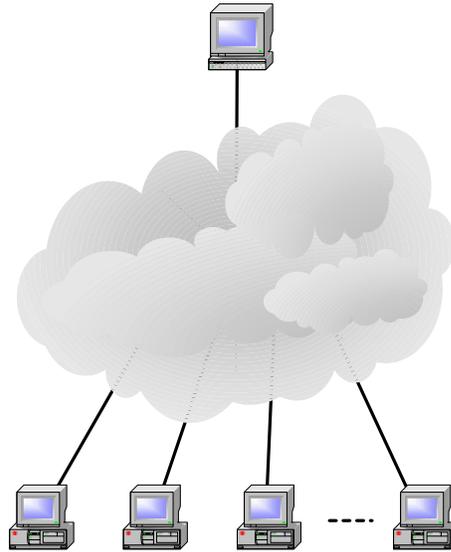
Figure 2.1   The majority of work to date in network tomography focuses on the special case of a single source transmitting probes to multiple receivers.

in Figure 2.2. This figure depicts the desired result in the network tomography problem. The figure

describes the topology in terms of a tree and performance parameters (loss in this example) are

associated with each link. Note that although no arrows are displayed on the links, every graph in

this discussion is a directed graph, with the flow of packets downward from the source to receivers.

In the Internet, paths are typically not symmetric, in that packets travelling from host A to host

B do not usually traverse the same links as packets travelling from B to A. Arrows are omitted

throughout to avoid cluttering the figures.

2.1.1   Inferring Link-Level Performance Using 1-by-2 Components

Suppose that the topology is known, but performance parameters have not been estimated. Fig-

ure 2.3 depicts the 1-by-2 components, with link-level loss rates labelled. Performance parameters

such as packet drop rate and delay variance have the property that they are non-decreasing along a

path under the assumption that network behavior is spatially independent. In other words, the loss
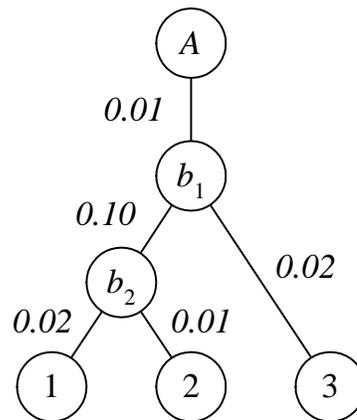
Figure 2.2 In the 1-by-N problem, the topology is a tree structure. The source, A, sits at the root of the tree and receivers 1, 2, and 3 are the leaves. Link level loss rates are shown next to each link as an example performance parameter. Although no arrows are drawn, it is always implied that the graphs depicted are directed graphs with packets fbwing downward from the source(s) to the receivers.

rate parameters depicted multiply (add logarithmically) along a path so that the end-to-end loss rate observed at receiver 1, for example, is the product of the loss rates on links (A, b_1), (b_1, b_2), and $(b_2, 1)$. By taking advantage of this non-decreasing property, the performance parameters inferred on each link of the 1-by-2 components suffice to calculate performance parameters on each link of the 1-by-3 tree. As the topology is assumed to be known, the branching point to receivers 1 and 2 is identified as b_2. The performance parameters on the links $(b_2, 1)$ and $(b_2, 2)$ of the 1-by-2 component in Figure 2.3(a) then directly correspond to those on the same links in the 1-by-3 tree. Again, by the assumption that network events are spatially independent, the loss rate on link $(b_1, 1)$ in the 1-by-2 component with receivers 1 and 3 (Figure 2.3(b)) corresponds to the product of loss rates on links (b_1, b_2) and $(b_2, 1)$ in the 1-by-3 tree. Thus by combining information from the 1-by-2 component having receivers 1 and 2 with information from the component having receivers 1 and 3, the performance on internal link (b_1, b_2) can be calculated. In this way, performance parameters on the links of the general 1-by-N tree can be calculated from those of the 1-by-2 components.

In practice, pairwise measurements are noisy, and thus the link-level parameters estimated from
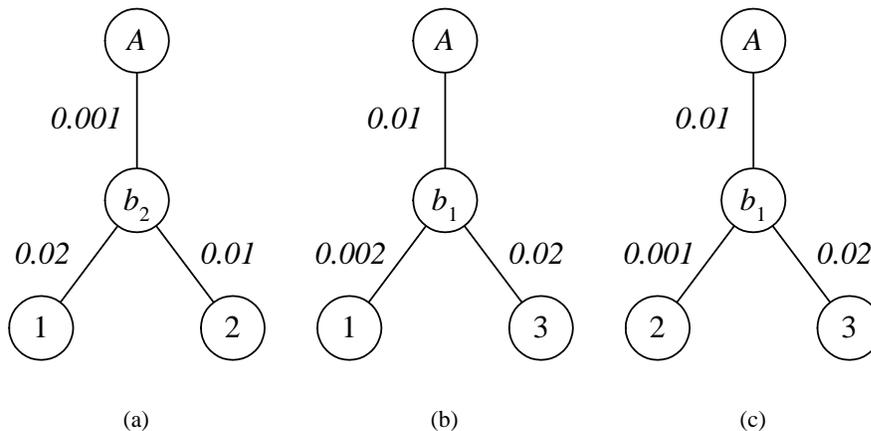
Figure 2.3   An example of how any single source tomography problem can be broken down into 1-by-2 sub-problems. In this example, having perfect estimates of the link-level parameters for the 1-by-2 sub-graphs for each pair of receivers is sufficient to reconstruct the 1-by-3 network depicted in Figure 2.2.

these measurements are not perfect. Measurements are made by sending back-to-back packet probes from the source, with one packet going to each receiver. These measurements take advantage of the fact that the back-to-back packets are highly correlated on the link from the source to the branching point, and uncorrelated on the lower legs. Thus, if a loss is observed at one receiver and not at the other, then one concludes that the loss occurred on the link after the branching point with high probability. While these pairwise estimates are noisy, it has been shown that they can still be used to achieve optimal estimates of performance on the links of the 1-by-N tree using the Expectation-Maximization (EM) algorithm [10, 14, 21]. Likewise, Bu, Duffield, Lo Presti, and Towsley have demonstrated that the EM algorithm is useful in the multiple source setting when the topology is assumed to be known [4].

2.1.2   Inferring Topology Using 1-by-2 Components

Now, suppose that nothing is known about topology or performance, so that the situation has the spirit of the network tomography problem depicted in Figure 2.1. Again, from of the non-decreasing

nature of performance parameters such as loss rate and delay variance, the 1-by-3 tree topology in Figure 2.2 can be reconstructed just from perfectly knowing the performance parameters for each 1-by-2 tree. By examining the three components in Figure 2.3 for the above example and noting that loss rates multiply along paths, one concludes that the branching point to receivers 1 and 2 must be the lowest in the tree since the loss parameters on the legs of the 1-by-2 component to these receivers are the largest. In this way, the 1-by-2 components can be used to impose a hierarchy on the branching points in a tree, thus arriving at the topology. Many single source topology identification algorithms work in exactly this fashion [6, 7, 22]. These algorithms grow a tree from the bottom up by grouping receivers (identifying branching points) based on pairwise measurements.

Actually, algorithms which use pairwise measurements to identify the topology are sub-optimal. This class of algorithms operates by identifying pairs of receivers which share more internal links in common before branching, and then grouping them. Each step in such a hierarchical construction is greedy. As a result, the region of the solution space over which a greedy algorithm searches is constrained after each step in the algorithm. Consequently, it is not possible to recover from a step in the wrong direction (perhaps due to noisy observations). An optimal algorithm searches over the entire solution space, unconstrained, and always finds the correct solution, however, optimal algorithms do not always exist. Despite the sub-optimality of hierarchical grouping algorithms, Duffield et al. report that grouping algorithms perform as well if not better than algorithms which directly solve for the optimal solution (ex. MLE), and the grouping algorithms are much less computationally complex than algorithms achieving the optimal solution [12]. This result is encouraging, since it suggests that a hierarchical reconstruction algorithm should also offer acceptable results for multiple source problems.

Additionally, the issue of *identifiability* arises when component measurements are used to re-

construct the network topology. The identifiability issue stems from the fact that for each type of performance metric there is an associated zero value or unobservable quantity. For instance, suppose that the loss rate on link (b_1, b_2) of the tree depicted in Figure 2.2 is zero. This would go undetected and the resulting topology would be a ternary tree, with the paths to all three receivers branching at b_1. The resulting tree in this case is termed a metric-induced topology because the topology reflects the observations at receivers and not necessarily the true logical topology [3]. Therefore, it is always possible to reconstruct the metric-induced topology by using components, and aside from the issue of identifiability the metric-induced topology is always the logical topology.

Thus, for any 1-by-N problem an acceptable solution is reachable using measurements made on the 1-by-2 component networks. Next it is shown that an analogous decomposition exists for the more general M-by-N problem.

2.2 Decomposing the Multiple Source Problem

Now the context is switched to the multiple source, multiple receiver problem considered in this thesis. Figure 2.4 depicts this setting as having multiple sources transmitting to multiple receivers, again through an uncharted network cloud. As before, identifying the topology and characterizing internal link-level performance are the two parts to this problem. Using multiple sources potentially provides a more accurate and refined characterization of topology and performance. However, determining the interactions between routes from different sources is an extremely challenging portion of this problem since measurements are not as easily correlated from two sources to a single receiver as they are from a single source to two receivers. This section presents the first contribution of this thesis by showing that the general M-by-N problem can be decomposed in to 1-by-2 and 2-by-1 components.
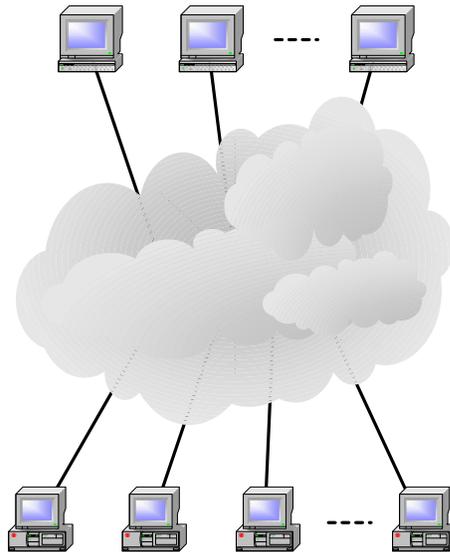
Figure 2.4   The multiple source network tomography problem consists of identifying the M-by-N topology and characterizing the link-level performance.

When multiple sources are used, another type of internal node must be defined in order to describe when the paths from two different sources to the same receiver join, namely the *joining point*. Figure 2.5 depicts a multiple source topology with joining points and branching points. In the M-by-N problem, topology identification breaks down into the tasks of determining where the paths from a given source branch to each receiver, identifying how the paths from the collection of sources join to a given receiver, and relating the branching point and joining point hierarchies.

Clearly, the idea of decomposing the 1-by-N single source network into 1-by-2 components extends to a decomposition into 1-by-2 and 2-by-1 components for the M-by-N multiple source problem. Just as each branching point is identified by at least one 1-by-2 component, each joining point is identified by at least one 2-by-1 component.

When the topology is assumed to be known, the analogy to having perfect knowledge of performance parameters on each 1-by-2 component of the 1-by-N problem is having perfect knowledge on the links of each 1-by-2 and 2-by-1 component in the M-by-N problem. With this component
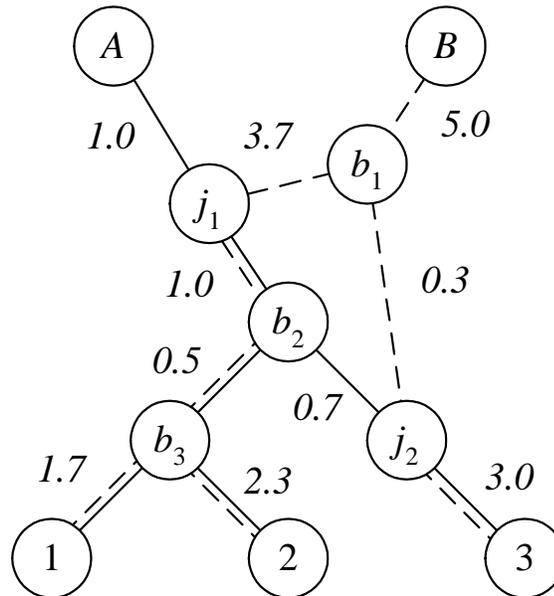
Figure 2.5   In the M-by-N problem, the topology is no longer a nicely defined tree structure. Sources A and B sit at the roots of the directed graph and receivers 1, 2, and 3 are the leaves. Internal nodes in a multiple source can be classified as either joining points or branching points. Link-level loss rates are depicted adjacent to each link.

level performance information and precise knowledge of the topology one can accurately identify

the performance parameters on the internal links of the M-by-N topology.

When the topology is not assumed to be known, grouping algorithms can be used to hierarchically arrange joining points from measurements made on the 2-by-1 components. This information, combined with the similar hierarchy of branching points and the assumption that a unique, well-behaved, path exists from each source to each receiver suffices to reconstruct the topology. The identifiability issue manifests itself in the same form as in the case of a single source, and so components still suffice to identify the metric-induced topology. Although such grouping algorithms will still be greedy and thus sub-optimal, it is expected that they will exhibit acceptable performance similar to single source grouping algorithms.

Now, it should be noted that a 2-by-2 network is the simplest multiple source, multiple receiver network, encompassing two 1-by-2 components and two 2-by-1 components. It then follows that

knowing the topology and link-level performance of every 2-by-2 component is sufficient for re-constructing an M-by-N topology, and for determining performance on the links of the M-by-N topology. For example, the 1-by-2 and 2-by-1 components derived directly from the 2-by-2 components depicted in Figure 2.6 provide enough information to accurately infer performance parameters – delay variance in this case – of the 2-by-3 network depicted in Figure 2.5.
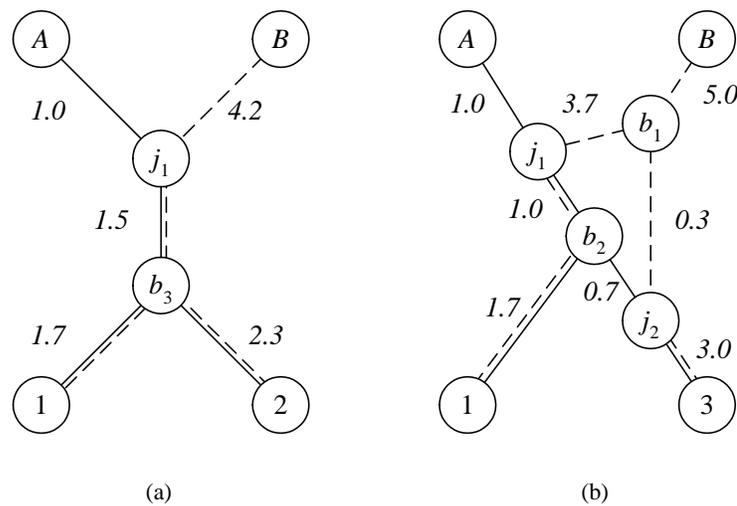
Figure 2.6   Decomposing the M-by-N into 2-by-2 components. In general, precisely knowing the topology and link-level parameters for every 2-by-2 component suffices to solve the M-by-N network tomography problem.

Thus, this thesis has identified the 2-by-2 component as the simplest multiple-source, multiple-receiver network which also captures the complexity of the general M-by-N network tomography problem. Much as solving the 1-by-2 problem results in solving the 1-by-N problem, solving the 2-by-2 problem is a large step toward solving the M-by-N problem. Recognizing its importance, for the remainder of this thesis the focus is placed on studying the 2-by-2 network tomography problem.

Chapter 3

The Two Source, Two Receiver Problem

In the last chapter the 2-by-2 problem was identified as being the fundamental multiple source, multiple receiver problem because it encapsulates two 2-by-1 components and two 1-by-2 components which. These components formed the base of the decomposition of M-by-N problems identified in this thesis. It was then concluded that solving the 2-by-2 problem would be a big step towards solving the general M-by-N network tomography problem.

This chapter examines the 2-by-2 problem further. A dichotomy of 2-by-2 network topologies is identified based on the model-order of the representation required to characterize each class. This dichotomy classifies the possible 2-by-2 network topologies as either shared or not shared. The major property of the shared topology is that it contains only one joining point and only one branching point. As a result, there are fewer links in the topology and thus fewer degrees of freedom in the model. Then a multiple source active probing algorithm is presented. This algorithm has been designed to classify a 2-by-2 network as shared or not shared. Measurements are based on packet arrival order, and are thus easy to make and not subject to measurement noise at the receivers. The algorithm is practical as it does not require any special timing infrastructure for precise synchronization between the participating hosts, and it can be implemented in either a multicast or unicast setting. In Chapter 4, the algorithm is further developed to incorporate measurements of performance.

3.1   The Shared/Non-Shared Dichotomy

Figure 3.1 below depicts some possible 2-by-2 topologies. The goal is to distinguish between these cases and to identify link-level performance parameters accordingly. In Section 2.2 it was noted that a 2-by-2 network is composed of two 1-by-2 components and two 2-by-1 components, where the latter impose a hierarchy on the joining points from the collection of sources to a given receiver. Estimating link-level performance parameters on a 2-by-1 network is an extremely challenging problem since the shared portion of the paths is below the joining point. Essentially, to make measurements analogous to those used to tease apart link-level parameters in the 1-by-2 component, one would need to send probes such they travel back-to-back on the portion of the 2-by-1 component *after* the joining point. In other words, the sources would need to send packets consistently such that they arrive at the joining point at exactly the same time. Making such precision measurements is impractical using standard packet probing techniques, and even a stretch when the end-hosts have a precision timing infrastructure is available.

| (a) | (b) | (c) | (d) |
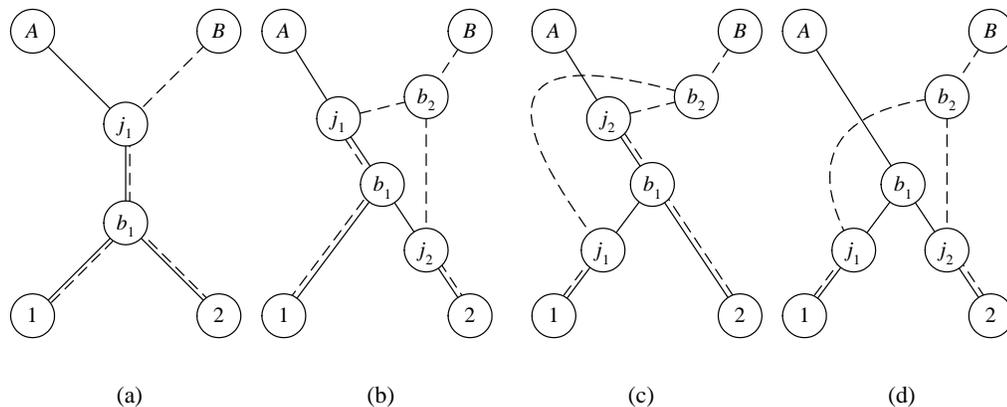
Figure 3.1   Four possible topologies for a two source, two receiver network. The shared topology in (a) has one joining point and one branching point, and fewer links over all. The non-shared topologies shown in (b), (c), and (d) each have distinct joining points for paths to each receiver. As a result 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.

While the possibility of completely solving the 2-by-2 problem currently looks grim, it is possible to get a step closer to the solution by considering the 2-by-2 problem in terms of the number of links, and thus link-level parameters, in each topology, rather than trying to precisely determine the true topology. From this perspective, the problem is similar to a model-order selection problem, where the goal is to appropriately characterize the underlying mechanism without using more of a description than is necessary. With existing measurement techniques, accurate estimates of parameters for links of the the 1-by-2 components are obtainable. These 1-by-2 components partially compose a 2-by-2 network. If each source was to approach the network tomography problem independently, then these 1-by-2 components would be the end result. However, for the particular 2-by-2 topology depicted in Figure 3.1(a) there is a common branching point for the paths from either source to the receivers. As a result, the downstream legs of each 1-by-2 component correspond to exactly the same logical links in the 2-by-2 network. This special 2-by-2 case is termed the *shared* topology in this thesis, since the lower links of each 1-by-2 component are shared. Consequently, sources can easily reduce the variance of the link parameter estimates on these shared links by averaging their individual estimates. That is, rather than treating the problem as two independent components, better estimates are obtained by combining estimates in the shared case. In any of the non-shared topologies (Figures 3.1(b-d)), there are no logical links in the individual 1-by-2 components which perfectly overlap. Consequently, there is no easy way to exploit this type of measurement made from each source in the 2-by-2 problem without also having parameter estimates for each 2-by-1 component.

A consequence of the common branching point in the shared topology is that the number of parameters to be estimated is reduced. Consider only the two 1-by-2 components for the 2-by-2 problem since current single source techniques allow for estimating these. In general there are six

links (three in each component), but when the topology is shared, the parameters corresponding to the lower links in each component are restricted to be the same, so that the number of parameters is reduced to four. Hence, the problem of distinguishing between shared and non-shared topologies has a model-order selection flavor to it. The higher order, non-shared model will always appear better since it over-fits to the data, regardless of the true underlying topology. However, the shared model is unable to fit to the data well when the true topology is not shared because of the restrictions imposed on the parameters.

3.2 Collaborative Probing From Multiple Sources

This section describes a multiple source probing algorithm developed as part of this thesis. The sources send packets in a semi-randomized fashion, and then measurements are made by recording packet arrival order at the receivers. These measurements are easy to make and require no special timing infrastructure. The intuition behind the algorithm is as follows. Assuming that packets do not get reordered, the arrival order of packets sent simultaneously from each source to a given receiver is identical to the order in which those packets arrive at the joining point of the paths from each source to that receiver. Thus, arrival order is completely determined by events occurring on the links leading up to the joining points. In the shared topology there is one unique joining point for the paths to both receivers, and in the non-shared topologies the joining points to each receiver are distinct. The design of the probing scheme presented here is based on distinguishing between these two cases based on the number of joining points.

The assumption has explicitly been made that packets do not get reordered within the network, but unfortunately this is not necessarily the case. Bellardo and Savage have recently performed a study on packet reordering in IP networks [2]. They conclude that the probability of two packets

being reordered as they travel through the network is highly correlated to the amount of time between when the packets are transmitted, with the probability of packets being reordered decreasing dramatically as the space between send times increases. Packets travelling 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 ordering will be preserved. In the algorithm described in this section, packets will occasionally arrive at a joining point very close to each other. The reordering potentially induced by these measurements is treated as noise and incorporated with the randomness due to queueing with cross-traffic.

3.2.1 Probe Structure

The basic multiple source probe is as follows. Each source sends one packet to receiver 1 and one packet to receiver 2. The transmission times of packets from each source are spaced apart by Δ seconds. The parameter Δ 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,

$$\Delta > \frac{\text{packet size}}{b_{\min}}, \tag{3.1}$$

where $b_{\min}$ is the lowest bandwidth of all links in the 2-by-2 network. This criterion ensures that the packets will traverse the network independently, and by making this spacing greater than 200 microseconds it is additionally ensured that these packets will not be reordered before they branch to the receivers.

Now, a random offset is introduced between the transmit times of corresponding packets at each source. Let t_0 denote the time at which the first packet is sent from source A. Then the first

packet is sent from source B at time $t_0 + u$ where u is a random variable distributed uniformly over the interval $[-D, D]$. In practice, D is much larger than Δ. By varying the offset u over a range of values, the difference in delays to each joining point is indirectly measured. These four packets constitute a single probe. Many probes are sent to obtain a series of packet arrival order measurements, as depicted in Figure 3.2.
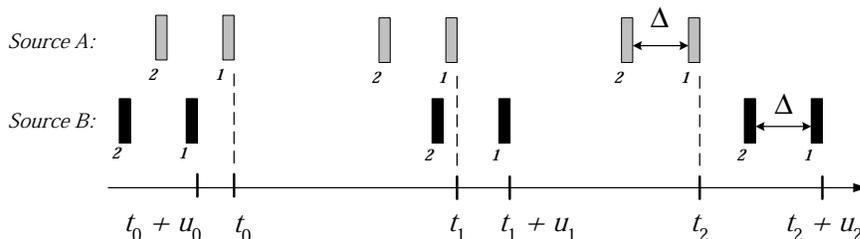
Figure 3.2   This figure depicts a series of probes. The inter-packet spacing, Δ, is chosen to be large enough so that queueing events affecting the two packets are independent. The offset variables u_i are independent random draws, uniformly distributed over the interval $[-D, D]$, and D is much larger than Δ in practice.

The receivers record the order in which the packets from each source arrive. The arrival order at each receiver is determined at the joining point to that receiver, assuming no reordering. Then, for the shared topology it is expected that the arrival order will always be the same at both receivers since the packets going to receiver 1 and to receiver 2 both traverse the same links before they reach the joining point. When the topology is not shared, there are two joining points, and so the paths from the sources to each of those joining points are different. As a result, for a certain range of offsets u, the arrival order will be different at each receiver. Thus, when the arrival observed at the receivers is the same for a series of probes, one concludes that the topology is shared. When different arrival orders are observed at each receiver then one concludes that the topology is not shared.

Now, because of cross-traffic, packets may be queued. Queueing essentially distorts the spacing, Δ, which each source inserts between packet transmission times. Consequently, it is possible that

some different arrival order measurements may occur even in the shared case. Thus, a baseline is needed for differentiating between cross-traffic and topological structure as the cause of different arrival order events. By keeping the same probe structure as defined above, but transmitting all four packets being transmitted to the same receiver rather than to both receivers, an estimate of the percentage of different arrival order events induced by cross-traffic to that receiver is obtained. This measurement process can be repeated to the other receiver, and then three estimates of different arrival order rates are obtained: one for probes with packets sent to both receivers, and two for probes with packets sent to a single receiver. In the shared case, because of the uniqueness of the joining point, all three of these rates should be identical. On the other hand, in any non-shared case, the paths to each receiver join at different points, so there is no reason for these three rates to be the same. In particular, if the topology is not shared, then the probability of observing different arrival order events when packets in the probes are sent to both receivers should be larger than either of the rates for single receiver probes because of the distinctness joining points. This process is elaborated on below.

3.2.2   Theoretical Analysis

The explanation above is rephrased, now, in more theoretical terms. First, a notation is introduced to describe delays along the logical links from each source to the joining point for paths to a particular receiver. Let $d_{S,R}(t)$ denote the delay process along the logical link from source S to joining point to receiver R. Figure 3.3 depicts shared and unshared topologies with labels assigned to the logical links leading up to each joining point.

Begin by considering a simplified scenario where there is no cross traffic so that delays on each link are constant terms (propagation delay). Again, it is assumed that packets are not reordered so
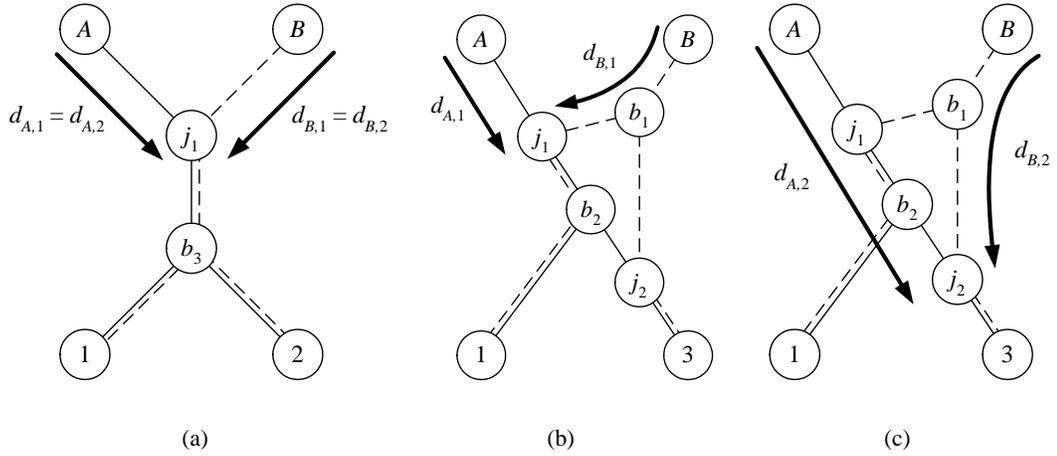
Figure 3.3   This figure depicts shared and unshared topologies with delays to each joining point labelled. The joining point in the shared topology is common for paths to both receivers. Joining points to each receiver are unique in the non-shared topology. The probing algorithm hinges on this idea to identify whether a topology is shared or not.

that the arrival order of packets destined for a given receiver is the same order in which the packets arrive at the joining point of their paths to that receiver. As before, source A sends a packet to receiver 1 at time t_0 and source B sends a packet to receiver 1 at time $t_0 + u$. The arrival order at receiver 1 can then be defined as

$$\alpha_1 \quad = \quad \text{sign}\left((t_0 + u + d_{B,1}) - (t_0 + d_{A,1})\right), \tag{3.2}$$

$$= \quad \text{sign}(d_{B,1} - d_{A,1} + u), \tag{3.3}$$

where $\alpha_1 = +1$ if packets from source A arrive first and $\alpha_1 = -1$ if packets from B arrive first. A similar expression can also be written for α_2, the arrival order of packets destined for receiver 2. Define the arrival order statistic to be

$$z \quad = \quad 1(\alpha_1 \neq \alpha_2), \tag{3.4}$$

where $1(\cdot)$ is the indicator function. Thus, z takes value 1 only when the arrival order at each receiver is different.

Observe that for the shared topology, $d_{A,1} = d_{A,2}$ and $d_{B,1} = d_{B,2}$. Thus, momentarily disregarding the effects of cross-traffic, it should always be true that $\alpha_1 = \alpha_2$ for a shared topology. Thus, in the shared case $z = 0$ always. In this case the random offset, u, determines the sign of each α_i, but the outcomes at each receiver are still identical so u essentially has no effect.

On the other hand, for any non-shared topology it is not likely that the delay differences to each receiver, $d_{B,1} - d_{A,1}$ and $d_{B,2} - d_{A,2}$, are the same. Then, for a certain range of offset values, the packet arrival order will be different at the two receivers. For example, set $d_{A,1} = 1$, $d_{A,2} = 4$, $d_{B,1} = 3$, and $d_{B,2} = 9$, as shown in Figure 3.4. Then there are three possible outcomes. When $u > -2$, packets from source A arrive first at both receivers. When $u < -5$ packets from source B arrive first at both receivers. When $-5 < u < -2$ then the packet from source B arrives first at receiver 1 and the packet from source A arrives first at receiver 2.

(a)   (b)
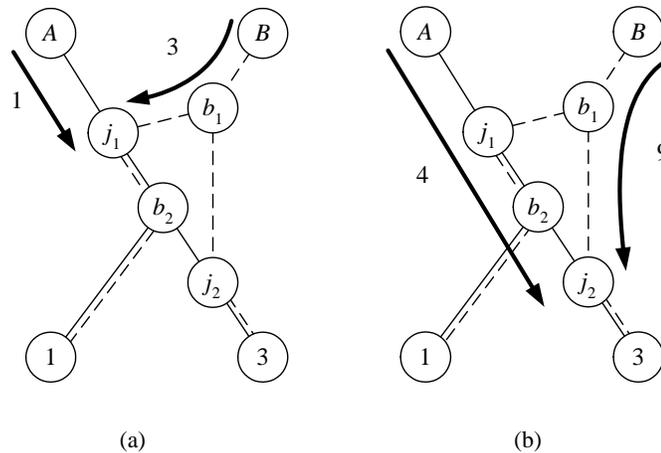
Figure 3.4   Non-shared topology example. In this example, the arrival order at each receiver will be different when the random offset, u, takes values between -2 and -5, because of the difference in delays to each joining point.

Thus, the random offset, u, acts as a mechanism for exploring the behavior of an unknown 2-by-2 network. Figure 3.5 below depicts the relationship between the α_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 receiver for a certain range of u. Essentially z is a Bernoulli random variable with parameter ρ. From this perspective, ρ is the probability that different arrival orders are observed at the receivers. In practice, many probes are sent, with the offset taking different values $u^{(1)}, u^{(2)}, \ldots, u^{(n)}$ for each probe. In essence, this procedure samples the function $z(u)$ in a random fashion. Then, keeping track of $z^{(1)}, z^{(2)}, \ldots, z^{(n)}$, one can calculate

$$\hat{\rho} = \frac{1}{n} \sum_i z^{(i)}, \tag{3.5}$$

an estimate of the probability of observing different arrival orders at each receiver. When this estimated probability is zero the algorithm declares that the 2-by-2 topology is shared, otherwise the algorithm declares that it is not shared.

Now, in reality there is cross-traffic in the network which induces a random queueing delay on each link. Thus the delays from sources to joining points are really delay processes, $d_{S,R}(t)$. In this case, the arrival order function at each receiver depends on the delay processes in addition to the offset, u. Consequently, the probability of observing different arrival orders at each receiver is no longer zero for the shared topology, but should be some small value due to queueing delay. For non-shared topologies, this probability reflects the different delay values and queuing behavior on links to the separate joining points.

The effects of queueing delay can be described in the following fashion in relation to Figure 3.5. When the two sources' sending times are offset such that packets arrive at the joining point(s)

(a)

(b)
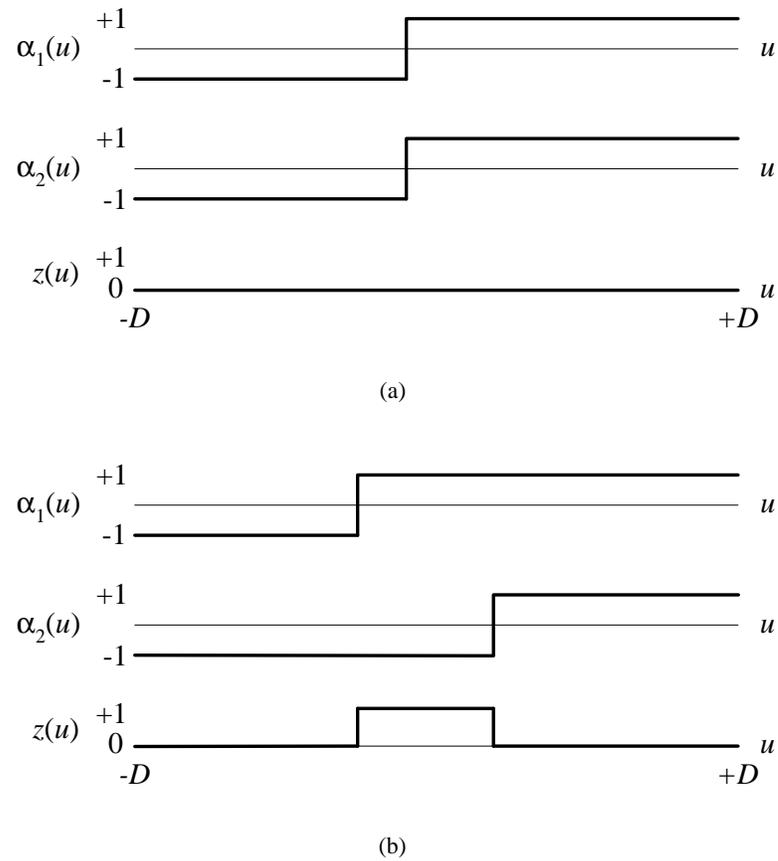
Figure 3.5   Displaying α_1, α_2, and z as functions of the random offset u for both shared and unshared topologies. The arrival order statistic, z, is used to determine whether a topology is shared or not.

at roughly the same time then it is possible for queueing delay to bump packets around so that

the arrival orders are different at each receiver, even for the shared topology. On the other hand,

when packets are already offset such that they reach the joining point at very distinct times then

queueing delay will not affect the arrival order outcomes. In other words, queueing delay only

affects measurements for the limited range of offsets such that the packets arrive at the joining point

close to each other.

3.2.3 Single Receiver Probing to Gauge Cross-Traffic Effects

To gauge whether the mechanism inducing different arrival order events is just cross-traffic (shared) or a combination of cross-traffic and the topology (non-shared), a modified probing scheme is developed. Similar to the probes described above, each source sends two packets spaced by time Δ with the same timing and offset as before, only that all four packets are destined for one receiver. Then, for a single receiver probe, 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.

Let ρ_1 denote the probability that a different arrival order event occurs when all packets are sent to receiver 1. Define the following packet arrival order observations for these measurements.

$$\alpha_1' = \operatorname{sign}(d_{B,1}(t_0 + u) - d_{A,1}(t_0) + u) \tag{3.6}$$

$$\alpha_1'' = \operatorname{sign}(d_{B,1}(t_0 + \Delta + u) - d_{A,1}(t_0 + \Delta) + u). \tag{3.7}$$

Likewise, the arrival order statistic is

$$z_1 = 1(\alpha_1' \neq \alpha_1''), \tag{3.8}$$

and the estimate of the probability of the arrival orders being different is

$$\hat{\rho}_1 = \frac{1}{n} \sum_i z_1^{(i)}, \tag{3.9}$$

using n probes. Note that this probability estimate reflects the amount of queueing on the links leading from each source to the joining point to receiver 1. A similar experiment can be performed, but with all packets going to receiver 2, and then an estimate $\hat{\rho}_2$ is obtained, that reflects the probability

of observing different arrival orders at receiver 2 due to queueing on the upstream links. To avoid confusion, let $\hat{\rho}_{12}$ denote the estimated probability for the original measurements described, where the first packet from each source goes to receiver 1 and the second packet goes to receiver 2.

Now, when the topology is shared, there is only one joining point so every different arrival order event is due to queueing delay. Because the joining point is the same for the paths to receiver 1 as it is for the paths to receiver 2, $\rho_1 = \rho_2$. Similarly, $\rho_1 = \rho_2 = \rho_{12}$ when the topology is shared since the only mechanism causing different arrival orders in the shared case is queueing along the paths to the one joining point.

When the topology is not shared, the joining points are different to each receiver as are the paths from each source to the joining points. In this case, queueing behavior may be different for links going to each joining point so it is not necessarily true that $\rho_1 = \rho_2$. For non-shared topologies, ρ_{12} should be larger than ρ_1 and ρ_2 since it is also expected that different arrival order events will be observed in the experiment involving both receivers due to the different mean delays from sources to each joining point.

A formal decision procedure is developed in Chapter 4. The intuition is that when $\hat{\rho}_1$, $\hat{\rho}_2$, and $\hat{\rho}_{12}$ are very similar it is concluded that the topology is shared. When $\hat{\rho}_{12}$ is significantly larger than the other two estimates it is concluded that the topology is not shared.

3.2.4 Performance and Multicast versus Unicast

The performance of this test is related to the ability to resolve the difference between the influence of queueing and influence of topological differences in the estimate $\hat{\rho}$. Further examination of Figure 3.5 reveals the relationship between the percentage of different arrival order events observed and random offset range, $[-D, D]$. The boundaries of the range of offsets which result in different

arrival order events are determined by the difference in mean delays to each joining point. As the difference in delays from each source to each joining point increases, the region becomes wider. Thus, if the joining points are more distinct in the sense that delay differences to each joining point are very different, then it is easier to identify the non-shared topology. As D increases, the ratio of the width of the range over which different arrival order events are observed to the width of the sampling range ($2D$) decreases. Consequently, to observe different arrival order events more probes must be sent. Thus, it would be nice to avoid having very large D in order to keep the number of probes required for an accurate estimate at a manageable level.

On the other hand, in order to determine whether there is one or more joining points, measurements must be made over the entire range where different arrival order events occur to determine its "width". Thus making D too small has the effect of focusing in on a specific range of offsets, and this must be avoided. In practice, a reasonable value is found by setting D equal to the maximum measured round trip time from any source to any receiver. This value for D ensures that the range of offsets resulting in different arrival order events is covered completely, and generally gives accurate results with a reasonable number of probes (1000 or less).

Additionally, it should be noted that in the description of the probing scheme above no mention was ever made as to whether the probes are composed of unicast or multicast packets. This is because the design of this algorithm is such that it works regardless of the transport mechanism. The advantages to using multicast measurements are also evident when more than two receivers are being considered. In this case, because each multicast packet goes to every receiver, one set of measurements suffices to determine whether the topology is shared or not for each pair of receivers.

3.2.5   Source Synchronization

A major advantage of arrival order estimates is that no special timing infrastructure is required to make precision measurements. The receivers only need to record the order in which packets arrive. Thus far it has been said that source A transmits packets at times t_0 and $t_0 + \Delta$, and that source B transmits packets at $t_0 + u$ and $t_0 + \Delta + u$. Using t_0 in both sources transmit times suggests that they are precisely synchronized. However, it will be shown that precise synchronization is not necessary and that only a coarse level of synchronization is required.

CPU clocks tend to be fairly reliable these days, so the time difference between source clocks can be characterized as a constant offset plus a difference in rate. Thus, letting $\tau_A(t)$ and $\tau_B(t)$ denote each source's perception of time, set

$$\tau_B(t) = \beta \tau_A(t) + \kappa. \tag{3.10}$$

Without loss of generality, let $\tau_A(t) = t$.

First consider the case where $\beta = 1$, so that the clock rates at each machine are identical and the only discrepancy in time is a constant offset. Again, source A sends packets at times t_0 and $t_0 + \Delta$. Now, the times when source B sends packets become $t_0 + \kappa + u$ and $t_0 + \Delta + \kappa + u$. The result of rewriting (3.2), the expressions for arrival order, is

$$\alpha_1 = \text{sign}\,(d_{B,1} - d_{A,1} + (u + \kappa))\,, \tag{3.11}$$

$$\alpha_2 = \text{sign}\,(d_{B,2} - d_{A,1} + (u + \kappa))\,. \tag{3.12}$$

The constant offset κ shifts the mean of the effective random offset so that now $u' = u + \kappa$ is

uniformly distributed over the interval $[-D+\kappa, D+\kappa]$. As long as this range of offsets still contains the interval over which different arrival order events occurs then the algorithm is unaffected. In practice, it is possible using a handshaking protocol to get the sources synchronized to within a few milliseconds. This is a slight shift so the constant offset is not a concern.

Next, the situation when there is a constant offset and a rate difference is analyzed. Suppose that probes are sent at some frequency $1/T$. Then the expression for the kth arrival order at receiver r is

$$\alpha_r(k) \quad = \quad \text{sign}\left((\tau_B(kT) + u + d_{B,r}) - (\tau_A(kT) + d_{A,r})\right), \tag{3.13}$$

$$= \quad \text{sign}\left(d_{B,r} - d_{A,r} + (u + \kappa + kT(\beta - 1))\right). \tag{3.14}$$

Now the distribution of the offset has a time-varying mean. It has already been emphasized that the main region of interest is the region of offsets where different arrival orders occur. In the shared case this region is a very narrow point around the offset where the arrival order flips from being -1 to $+1$, or vice versa. The time-varying mean will have the effect of distorting the different arrival order region and making it look wider, since the point is effectively shifting across offsets over the course of the experiment. In this case, the reference measurements made to a single receiver will also exhibit the distorted different arrival order region, thus the effect of the time-varying mean will resemble that of queueing due to cross-traffic.

It has already been determined that queueing effects are only visible near the cross-over points, when the offsets are such that packets from both receivers arrive at the joining points near each other. Away from these points queueing delay has no effect on the packet arrival order, so if the original offset interval magnitude D is sufficiently large so that the edge of this interval never gets near a cross-over point then the rate difference in the sources' clocks will not corrupt the measurements.

In practice, it has been found that by setting D using the maximum round-trip time as described above, typical rate differences do not significantly degrade the procedure.

3.3 Summary

To recap, this chapter focused on studying the 2-by-2 problem, since it captures all of the complexity of the more general M-by-N problem. While it is not clear how to make measurements which will distinguish between all of the possible 2-by-2 topologies, there is a clear dichotomy into shared and non-shared classes of topologies. This dichotomy is relevant for network tomography and is based on the number of degrees of freedom in the model for each topology. Then a novel multiple source probing scheme was presented which can be used to identify whether the underlying topology in a 2-by-2 problem is shared or not. The algorithm is based on measurements of packet arrival order which are easy to make, and benefits from not requiring precise time synchronization between any of the participating hosts.

To summarize the algorithm, a 2-by-2 network is considered, where the underlying topology is unknown. The sources send many probes while varying u. Packet arrival order is recorded at the receivers, and an estimate for the probability of the arrival orders being different at each receiver is obtained. When this estimate is very small the topology is shared, otherwise it is not shared. The next chapter demonstrates that back-to-back probes can be incorporated into the algorithm for performance assessment, and presents a decision-theoretic framework for incorporating measurements of performance into the multiple source probing algorithm.

Chapter 4

Combining Measurements of Performance and Topology

In Chapter 2 the general multiple source network tomography problem was examined. It was concluded that studying the 2-by-2 problem, the simplest multiple source, multiple receiver problem, would offer an insight to the general problem. That provided the motivation to focus on the 2-by-2 network in Chapter 3, where it was recognized that the shared topology is distinct from other 2-by-2 topologies in the model-order sense. In light of this dichotomy, a multiple source probing algorithm was developed based on the packet arrival order metric.

This chapter demonstrates how performance measurements such as packet drop rate or delay variance can be incorporated in to the probing scheme in both the multicast and unicast settings. Then the problem of distinguishing between shared and non-shared networks is formulated using statistical decision theory. This chapter describes how to set up likelihood functions for loss and packet arrival order, and how to appropriately set a threshold based by choosing the probability that the algorithm declares a 2-by-2 network to be not shared when in fact the true topology is shared. Experimental results demonstrate the robustness and efficacy of this approach.

4.1   Topology and End-to-end Performance Measurements Are Related

Traditionally, researchers in network tomography treat the problems of topology identification and link-level performance analysis separately. The problem is commonly formulated in terms of the linear model

$$y = A\theta + \epsilon, \tag{4.1}$$

where $\boldsymbol{y}$ is the vector of end-to-end measurements, $\boldsymbol{A}$ is the *routing matrix*, $\boldsymbol{\theta}$ is a vector of link-level properties such as loss, delay, or marking probability when analyzing ECN, and $\boldsymbol{\epsilon}$ is the noise vector capturing the randomness of the problem [8]. Binary entries in the routing matrix describe how packets are routed from source to receiver. When the topology is not known the focus of the problem is on determining values of the matrix $\boldsymbol{A}$. When estimating the link-level parameters, $\boldsymbol{\theta}$, it is assumed that the routing matrix $\boldsymbol{A}$ is known, resulting in the formation of an ill-posed inverse problem.

The link-level performance metrics of interest – loss rate, mean delay, and delay variance – each have the property of being non-decreasing along a path. That is, if θ_l denotes the log success rate (success rate = 1 - loss rate) on a link l, then for two consecutive links in a path, l_1 and l_2, it is always true that $\theta_{l_1} \leq \theta_{l_1} + \theta_{l_2}$, with equality only when the success rate on link l_2 is 1 (no drops). This is exactly how measurements to pairs of receivers are used to reconstruct the topology, and a topology is said to be identifiable when the loss rate on every link is strictly positive.

Similarly, knowing the topology constrains how the end-to-end observations relate to link-level parameters. By knowing that the path from A to 1 is composed of some set of links $\mathcal{P}(A, 1)$, the link-level estimates are constrained so that

$$\hat{y}_{A,1} = \sum_{l \in \mathcal{P}(A,1)} \hat{\theta}_l. \tag{4.2}$$

Thus, the estimated performance parameters must add up to match the end-to-end observations.

Hence, recognizing the interactions between topology and end-to-end measurements, it makes sense to simultaneously optimize over both of these criteria rather than treating the problem in a two-step fashion. This is one consideration taken into account when formulating the likelihood

ratio test.

4.2   Multiple Source Performance Measurements

Recall the breakdown of the 2-by-2 topologies into shared and non-shared classes as described in Section 2.2. This distinction was based on the number of degrees of freedom in each model. The non-shared models have six degrees of freedom, found by considering the 2-by-2 network as two 1-by-2 components. The shared model, on the other hand, has four degrees of freedom since the link-level parameters on the lower links of each 1-by-2 component are constrained to be the same.

In many single source network tomography algorithms repeated measurements are made to a given pair of receivers, and then estimates of the internal parameters are obtained by averaging and conditioning on the observations made at each receiver. It is a well known fact that for a collection of i.i.d. random variables, the variance of the sample mean, is inversely proportional to the number of samples being averaged over. Thus, if the sources know that the 2-by-2 topology is shared for a given pair of receivers, then they can pool their measurements together to obtain lower variance estimates on the shared links.

Ignoring arrival order measurements for the time being, suppose each source independently made the usual pairwise measurements as described at the end of Section 2.1.1. One could consider comparing the link level estimates made from each source to determine whether routes are shared or not for each pair of receivers. If the downstream parameter estimates are nearly the same for two 1-by-2 components, one could conclude that the topology is probably shared. Otherwise, one would declare that the topology is probably not shared. While this scheme is legitimate when the parameters are distinct on each link, this condition is not guaranteed in real networks. In fact, when estimating loss rates in the current over-provisioned state of the Internet, one expects very low loss

rate estimates on most links. Thus, such an approach is not robust to situations where the link level behavior is similar even though in truth the topology is not shared.

Multiple source probes, as discussed in the previous section, are a more robust measurement of topology since the packet arrival order depends directly on a topological feature – the number of joining points in the 2-by-2 topology. However, packet arrival order does not directly tell us anything about network internal performance. The next section describes how performance measurements can be incorporated into the multiple source probing framework described in the previous chapter.

4.3   Probing For Performance and Topology from Multiple Sources

Although the arrival order measurements made in the multiple source probing scheme only directly help to determine whether or not the 2-by-2 topology is shared, it is not difficult to incorporate performance measurements in to the picture. Then one can simultaneously solve for maximum likelihood estimates of the performance parameters and make a decision as to whether or not the topology is shared.

Multicast. In a multicast setting, each packet sent by a source will travel down the multicast tree to every receiver in the multicast group. For probes sent by one source, the portion of each receiver's observation due to events on shared links is perfectly correlated. If a packet is dropped on a link in the tree before the paths to the two receivers branch apart then this loss will be observed at both receivers.

As a result, in the multicast setting one can consider every packet sent in our multiple source probing algorithm to also be a probe in a performance parameter inference algorithm. We already ensure that the packets sent from each receiver are independent by choosing the inter-packet spacing Δ according to (3.1). Thus, when using multicast packets, the multiple source probing algorithm

does not need to be modified to be used for both estimating link-level performance and for distinguishing between shared and unshared topologies.

Unicast. The major difference between the unicast and multicast methodologies is that unicast packets only ever have one receiver. As a result, pair-wise measurements in the unicast setting take the form of back-to-back packet probes sent from a source, with each packet destined for a different receiver. Then back-to-back packets are highly correlated as they travel on the common links before the branching point. The correlation of back-to-back packets is not perfect, as in the multicast setting, but still adequate.

The multiple source probing structure can be modified to accommodate performance measurements by replacing each packet in the original probe structure (Figure 3.2) with a packet pair. The new probing structure is depicted in Figure 4.1. By using this structure, back-to-back packet probe measurements are available for estimating link level performance parameters, and the arrival order estimates are available for classifying the topology as shared or not shared. Additionally, the packets in each probe destined for receiver 1 are exactly those needed for the single receiver measurements used to calculate $\hat{\rho}_1$ as described in Section 3.2.3. Similarly, an estimate of $\hat{\rho}_2$ is obtained by only considering the packets sent to receiver 2. Thus, it is possible to extract all of the measurements of interest using this single type of probe.

4.4 A Decision-Theoretic Framework For Combining Measurements

This section presents a framework for combining each of these measurements to obtain the proper parameter estimates, according to our estimate of the underlying topology. Again, the 2-by-2 problem is considered, assuming no prior information about the topology or link-level performance.

Suppose the sources send N probes. Each receiver keeps track of packet arrival order and of
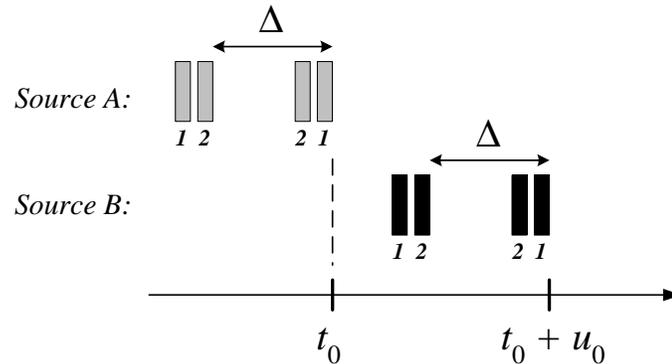
Figure 4.1   Modifi ed multiple source probes. Each rectangle represents a packet and the numbers beneath each rectangle indicate the packet's destination. In the unicast setting we replace each packet with a back-to-back packet pair in order to acquire measurements which can also be used to estimate link-level performance. In each back-to-back packet pair, one packet goes to receiver 1 and the other to receiver 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.

an end-to-end performance measurement – either loss, mean delay, delay variance, or possibly all

three. Let y denote the combined set of performance statistics from both receivers, and let z denote

the combined set of arrival order statistics for an experiment. Denote by $\theta_1, \ldots, \theta_6$ the link-level

performance parameters corresponding to links as depicted in the two 1-by-2 networks in Figure 4.2.
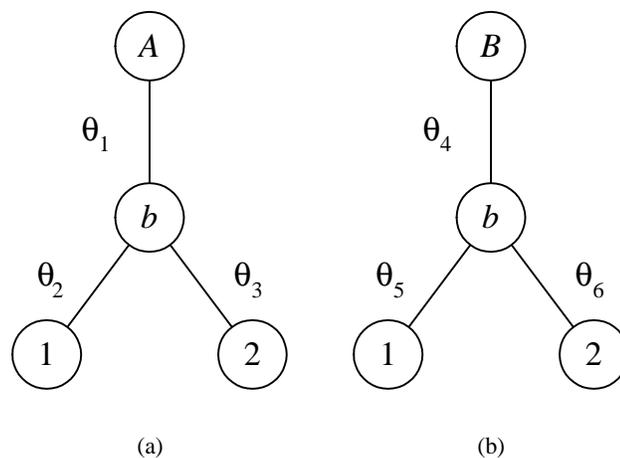
Figure 4.2   Two 1-by-2 components which comprise the 2-by-2 problem. The goal is to estimate the link-level performance parameters, $\theta_1, \ldots, \theta_6$, averaging the estimates from each source when the topology is shared.

Let H_S denote the hypothesis that the 2-by-2 topology is shared, and let H_N denote the hy-

pothesis that the topology is not shared. Let $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_6)$ denote the general six-dimensional vector of performance parameters and let $\boldsymbol{\rho} = (\rho_1, \rho_2, \rho_{12})$ denote the three dimensional vector of different arrival order probabilities. Recall that ρ_1 corresponds to the probability of a different arrival order observation when all packets are sent to receiver 1, ρ_2 corresponds to the probability of a different arrival order observation when all packets are sent to receiver 2, and that ρ_{12} corresponds to the probability of a different arrival order event when the first packet from each source is sent to receiver 1 and the second packet from each source is sent to receiver 2.

Under each hypothesis the joint likelihood function is written as $p(\boldsymbol{y}, \boldsymbol{z} | H_i, \boldsymbol{\theta}, \boldsymbol{\rho})$. Then the hypothesis is chosen which maximizes the likelihood of the observations. We can factorize the likelihood function of the observations into

$$p(\boldsymbol{y}, \boldsymbol{z} | H_i, \boldsymbol{\theta}, \boldsymbol{\rho}) = p(\boldsymbol{y} | H_i, \boldsymbol{\theta}) p(\boldsymbol{z} | H_i, \boldsymbol{\rho}), \tag{4.3}$$

implying that the performance measurements and events arrival order are statistically independent. Independence of the performance measurements follows from the assumption that the inter packet-pair spacing, Δ, is sufficiently large so that the first and second back-to-back packet probes sent from each source are independent, as reflected in (3.1). Recall that the performance measurements are made using intra-pair comparisons. Arrival order measurements, on the other hand, are made using inter-pair comparisons. As the pairs are spaced so that they are independent, the two types of measurements are also statistically independent.

Now, the true link level parameters are unknown variables. One common way of solving such composite hypothesis problems is using the generalized likelihood ratio test (GLRT). In the GLRT, the unknown distribution parameters $\boldsymbol{\theta}$ and $\boldsymbol{\rho}$ are replaced with their maximum likelihood esti-

mates under each model. Let $\boldsymbol{\Theta}_N$ denote the six-dimensional space of unrestricted performance parameters. For example, if the θ_i represent link-level loss rates, then $\boldsymbol{\Theta}_N = [0, 1]^6$, and if the θ_i represent delay variances, then $\boldsymbol{\Theta}_N = \mathbb{R}^6$. Let $\boldsymbol{\Theta}_S$ denote the space of performance parameters with the restrictions $\theta_2 = \theta_5$ and $\theta_3 = \theta_6$. Similarly, the space of unrestricted different arrival order probabilities is $[0, 1]^3$, and the space of different arrival order probabilities with the restriction that $\rho_1 = \rho_2 = \rho_{12}$ is $[0, 1]^1$, since probabilities must take values between zero and one. Then the GLRT can be written as

$$\Lambda(\boldsymbol{y}) = \frac{\max\limits_{\boldsymbol{\theta} \in \boldsymbol{\Theta}_N, \boldsymbol{\rho} \in [0,1]^3} p(\boldsymbol{y}|H_N, \boldsymbol{\theta})p(\boldsymbol{z}|H_N, \boldsymbol{\rho})}{\max\limits_{\boldsymbol{\theta} \in \boldsymbol{\Theta}_S, \boldsymbol{\rho} \in [0,1]^1} p(\boldsymbol{y}|H_S, \boldsymbol{\theta})p(\boldsymbol{z}|H_S, \boldsymbol{\rho})}. \tag{4.4}$$

Then a correct decision is made according to

$$\Lambda(\boldsymbol{y}, \boldsymbol{z}) \underset{H_S}{\overset{H_N}{\gtrless}} \eta, \tag{4.5}$$

for some threshold η. When the likelihood ratio is greater than the threshold, the test proclaims that the topology is not shared, otherwise the test proclaims it is shared.

In general, it is very difficult to set a threshold for the GLRT 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 [26]. Let $\lambda(\boldsymbol{y}, \boldsymbol{z}) = 2 \log \Lambda(\boldsymbol{y}, \boldsymbol{z})$. Then under some very mild assumptions about the regularity of the likelihood functions $p(\boldsymbol{y}|H_i, \boldsymbol{\theta})$ and $p(\boldsymbol{z}|H_i, \boldsymbol{\rho})$, Wilks' Theorem states that under the null hypothesis (the shared hypothesis in this case), $\lambda(\boldsymbol{y})$, converges in distribution to a chi-squared random variable with number of degrees of

freedom equal to the difference of the number of degrees of freedom under each hypothesis. In other words, if the number of degrees of freedom under hypothesis H_N is $\nu_N = |\Theta_N| + 3$ and the number of degrees of freedom under hypothesis H_S is $\nu_S = |\mathcal{R}_S| + 1$, then

$$\lambda(\boldsymbol{y}, \boldsymbol{z}) \xrightarrow{d} \chi^2_\nu, \tag{4.6}$$

with $\nu = \nu_N - \nu_S$. By knowing the distribution of the log likelihood ratio statistic under the shared hypothesis it is possible to set a threshold by choosing a probability of false alarm, P_F, the probability that the topology is declared to be not shared when it is shared in truth.

4.5   Simulation Evaluation

This section presents the results of model based simulations used to test the performance of the algorithm described above. In the experiments, back-to-back packet probes are used to infer loss rate. Results show that the log likelihood ratio statistic reaches the asymptotic chi-squared distribution after only using a few hundred probes. Furthermore, the multiple source probing algorithm is an extremely robust detector capable of identifying non-shared topologies over 90% of the time while only misclassifying a shared topology as non-shared in 1 out of 10 cases.

In our simulations, each logical link is characterized by a constant propagation delay plus an $M/M/1/K$ queue. While this model does not capture the intricacies of real Internet traffic, it is a useful tool for assessing the performance of our algorithm under a variety of conditions. The $M/M/1/K$ queueing distribution is characterized by the queue's length and the ratio of the intensity of incoming traffic to the service rate of the queue [23]. By varying these parameters it is possible to emulate both low utilization and high utilization scenarios. The situation is also ex-

amined where back-to-back packets do not remain highly correlated so that the loss estimates are not accurate. For each simulation scenario ten thousand trials were performed, varying link-level characteristics throughout. The number of probes used varied between 100 and 1000. All packets have the same size in these simulations.

4.5.1 Justifying the Asymptotic Result

Figures 4.3 and 4.4 depict histograms of the joint log likelihood function (using arrival order and loss measurements), when the true topology is shared. The data in these histograms comes from a series of simulations where the link-level characteristics are heterogeneous, varying from data point to data point, and thus reflecting a wide range of general scenarios. The propagation delay on each link varies between 10 and 150 milliseconds, queue lengths vary between 10 and 20 packets, and the intensity ratio varies between 0.8 and 0.95 which in turn, makes the loss rate on each link vary between 0.5% and 4%.

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 in each figure for reference. When only 100 probes are used, as depicted in Figure 4.3, the asymptotic behavior apparently is not completely exhibited, but the major trends are there. After 1000 probes (Figure 4.4), however, the histogram takes the shape of the chi-square distribution. Supposing that one limits the bandwidth to be used by this algorithm to 1kbps, then for 48 byte probes, 1000 probes can be transmitted in roughly 5 minutes.
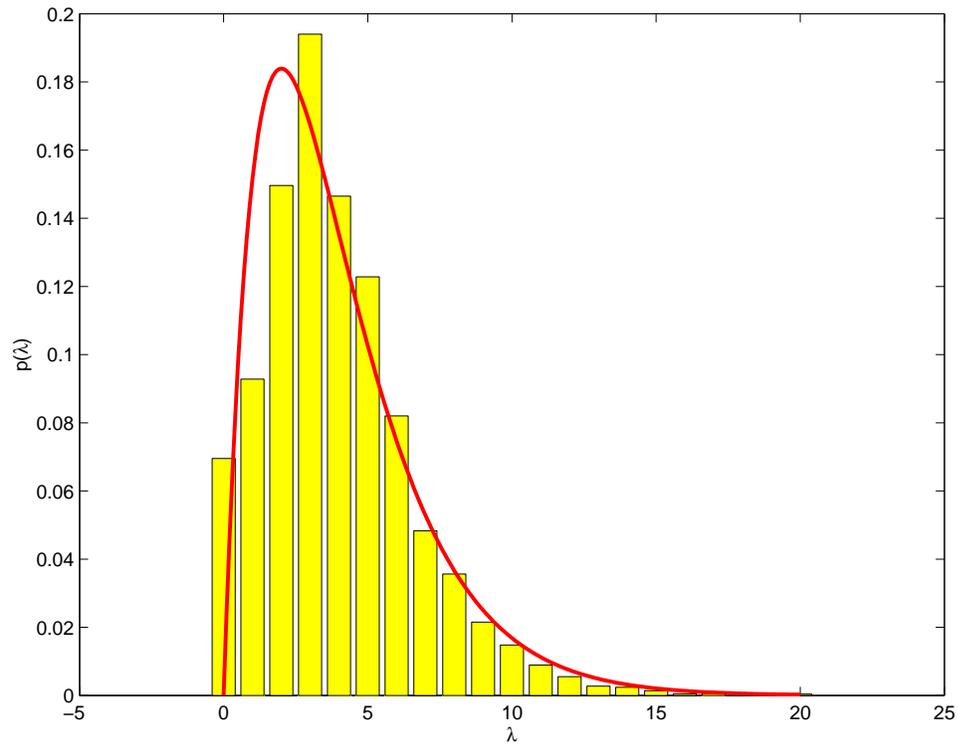
Figure 4.3 Histograms of the joint log-likelihood ratio using 100 probes. According to Wilks' Theorem, this statistic should be distributed according to a chi-squared distribution with four degrees of freedom. The chi-squared distribution is plotted on top of the histogram. After only using 100 probes the histogram fits the chi-squared distribution fairly well.

4.5.2 Algorithm Performance

Next, the performance of our GLRT-based algorithm is assessed by examining the receiver-operator characteristic (ROC) curve in a variety of conditions. In each ROC curve the x-axis is the probability of false alarm (P_F), or in other words, the probability of declaring that a topology belongs to the non-shared class when in truth it is shared. The y-axis in each figure corresponds to the probability of detection (P_D), the probability that a topology is correctly identified as not shared. As both axes are probabilities, they can take values between zero and one. The false alarm probability or probability that the algorithm incorrectly declares that a network is not shared when in truth it is shared, is a parameter to be set by the user. Then, using this parameter and the result
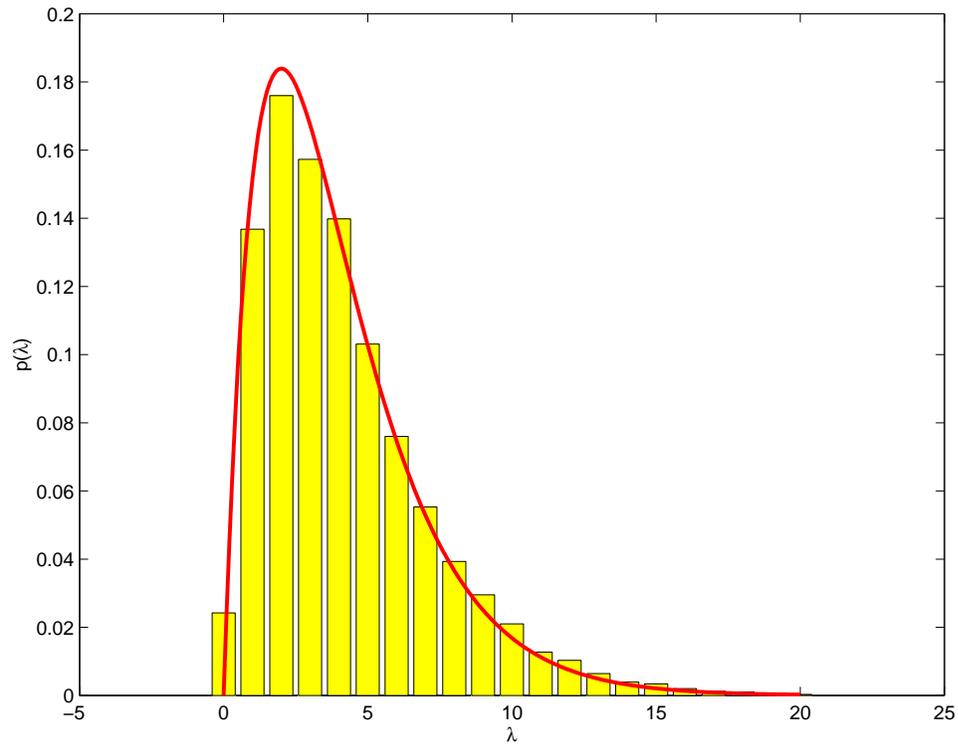
Figure 4.4   Histogram of the joint log-likelihood ratio when 1000 probes are used, and a chi-squared distribution with four degrees of freedom plotted on top. After using 1000 probes, the histogram fits the distribution perfectly. Thus, the asymptotic results given by Wilks' theorem can by applied to set a threshold for the statistical test.

from Wilks' Theorem, the threshold is determined. The performance of the statistical test is then reflected by the probability of correctly identifying a topology as not shared. Thus, the ideal ROC curve would have $P_D = 1$ for every value of P_F in the interval $[0, 1]$.

Figure 4.5 shows the ROC curve for the same simulation data used to generate the histograms above. Note the values along each axis. Only values of P_F ranging from 0 to 0.5 are displayed as one typically wishes to operate at low P_F. In this range, the algorithm performs very well, so only values of P_D from 0.5 to 1 are displayed. Again, this data set consists of ten thousand trials, where the link-level properties vary on each trial. In this case 1000 probes were used. ROC curves are plotted for detectors using only arrival order measurements, using only loss measurements, and using both sets of measurements. Because the variation in loss rates is not very large, the detector using loss rates

does not do very well alone. Essentially, this detector compares the loss rates estimated by each source on the downstream legs, declaring the topology to be shared when estimates match between sources. Using arrival order measurements only, the detector performs marginally better than using loss only. However, the joint detector offers the best performance across the board, achieving a P_D of approximately 0.9 when P_F is set as low as 0.5. Thus, by combining loss and arrival order measurements it is possible to do much better than when just using arrival order or loss estimates separately.
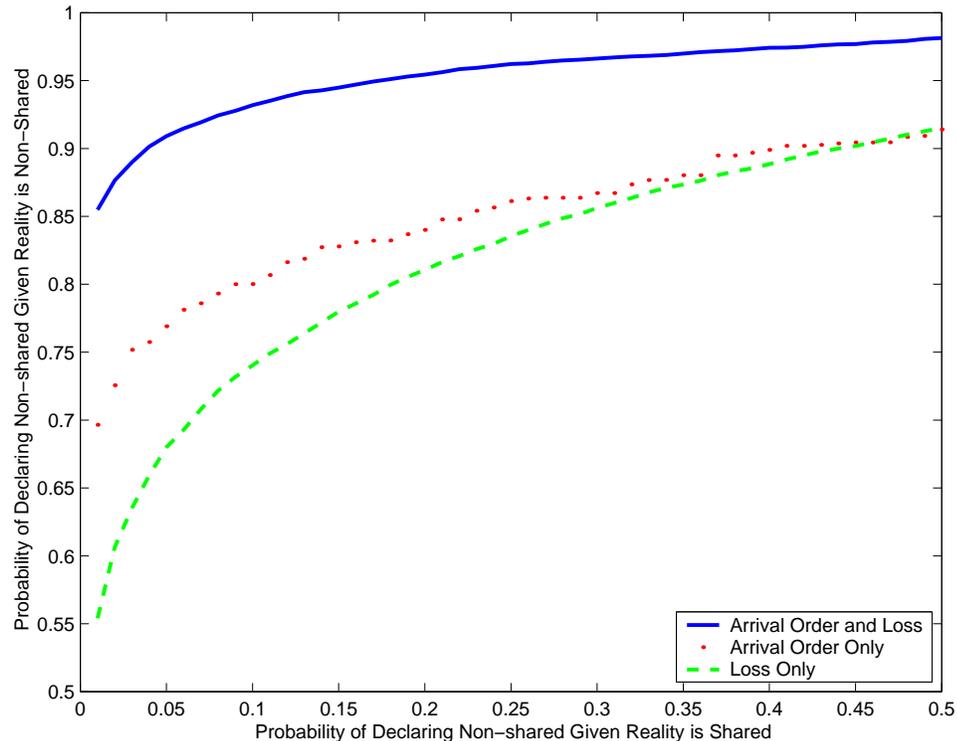
Figure 4.5   ROC Curves in heterogeneous network conditions. The probability of mistakenly declaring a shared topology to be non-shared, shown on the x-axis, is a parameter set by the user in order to determine a threshold value for the test. The higher the curve, then, the better the performance of the detector in terms of being able to distinguish between shared and non-shared topologies. The joint detector, using arrival order and loss measurements, exhibits the best performance.

Next, it is verified that the rate at which shared topologies are incorrectly declared to be non-shared (P_F) matches the parameter set by the user. Figure 4.6 shows a plot of the estimated false

alarm probability versus the desired false alarm probability for the detectors using loss measurements only, using arrival order measurements only, and using both loss and arrival order measurements. In all three cases the plot is approximately a line with slope 1, confirming that the false alarm probability exhibited does in fact match what was expected.
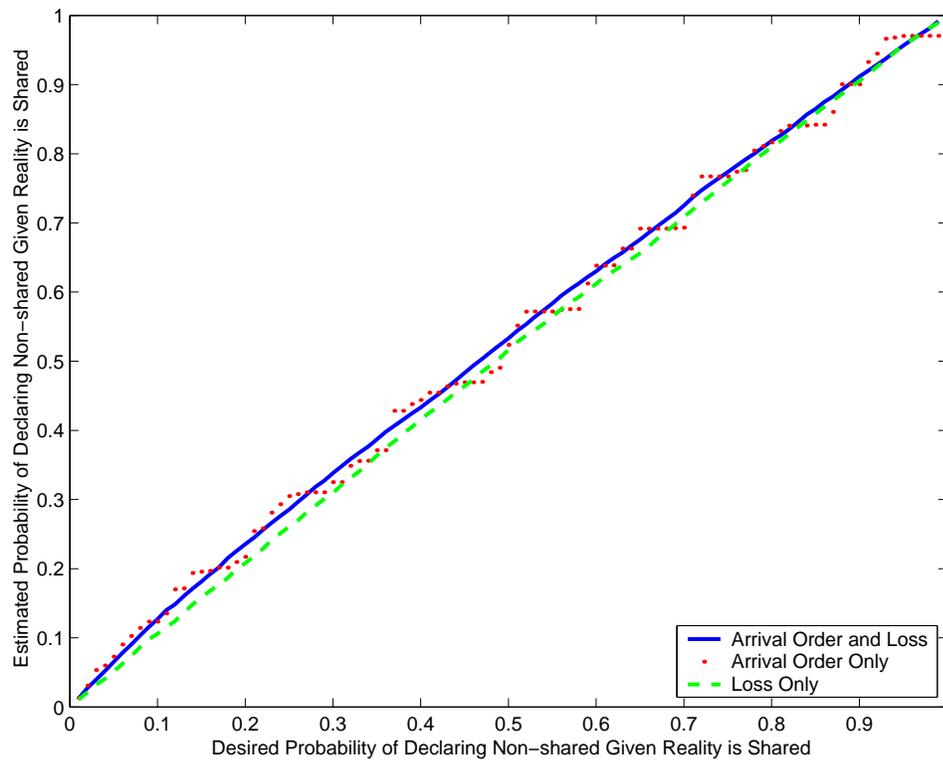
Figure 4.6   Plotting the estimated false alarm probability against the desired false alarm probability. All three curves fit a straight line with slope 1, which verifies that the false alarm probability exhibited by the detector will indeed match that set by the user when determining a value for the threshold.

Figure 4.7 depicts the ROC curves for a simulation in which packets within a pair do not exhibit correlated losses. Loss estimates in this case are useless, degrading the performance of the joint detector even more so than when the utilization is low. In these situations, however, the robust arrival order detector still exhibits the same level of accuracy.

Finally, in Figure 4.8 the performance of the joint detector is examined in homogeneous network conditions, as the number of probes used increases from 100 to 1000. Clearly, the performance
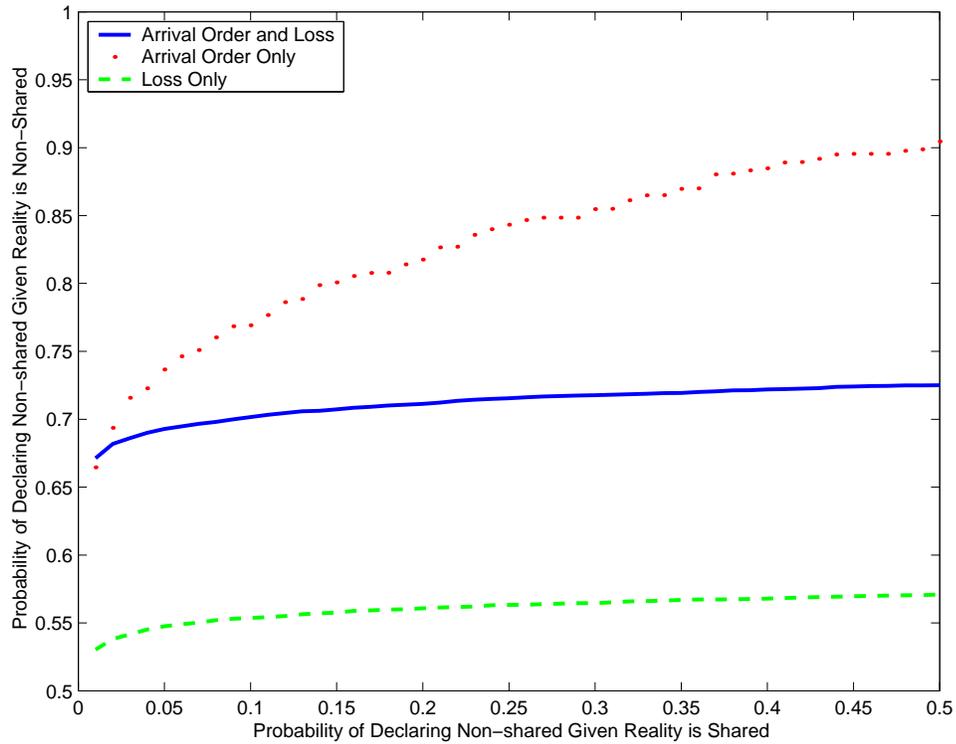
Figure 4.7 ROC curves when back-to-back packet probe measurements are uncorrelated. In this case, the loss estimates are meaningless, and so the performance of the test using only loss measurements is very poor. Consequently, the performance of the detector using both loss and arrival order measurements is degraded, however the detector using arrival order only still exhibits an acceptable level of performance.

of the detector improves when more probes are used. The increase in performance is especially pronounced for values of P_F around 0.5 where one would typically like to operate. Thus, it is concluded that a desirable performance rate can be achieved using a reasonable number (500-1000) of probes.
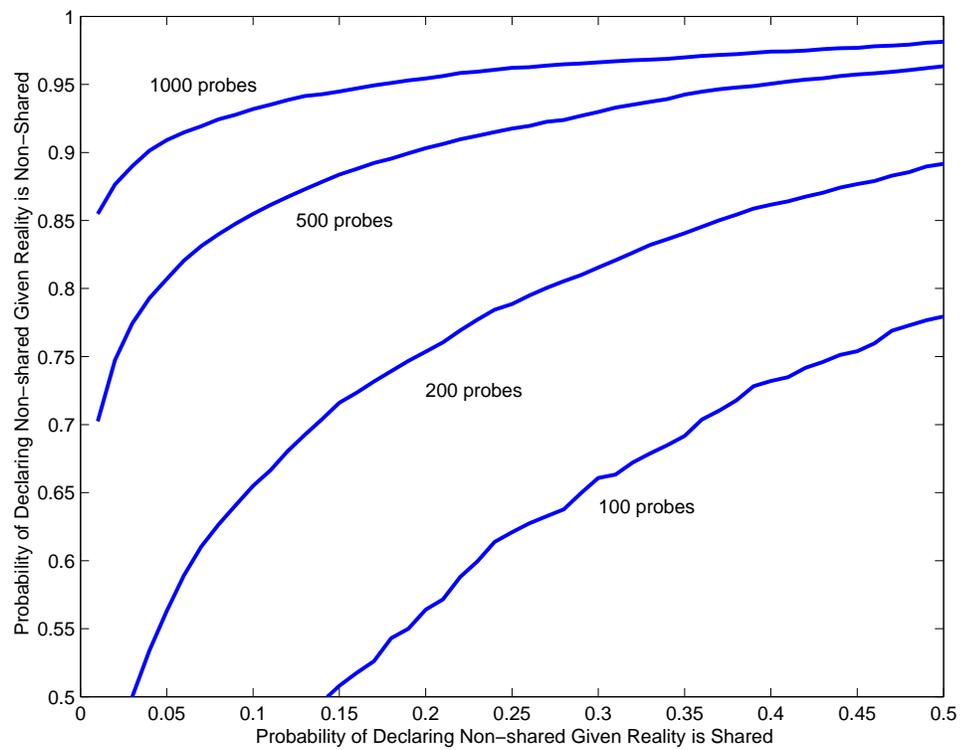
Figure 4.8 ROC curves plotted, varying the number of probes used. The detector performance, especially for very low values of P_F, increases very quickly as the number of probes used increases.

Chapter 5

Conclusion and Discussion

By making coordinated measurements from multiple sources, it is possible to obtain a more accurate and refined characterization of the network topology and performance. This goal of this thesis was to explore ways in which this could be achieved. The first step was to decompose the general multiple-source, multiple-receiver network tomography problem in to smaller two-source, two-receiver components. While considering the general problem in terms of 2-by-2 components may not scale very well to very large networks, it is an important initial step towards solving the general large-scale problem.

Focusing then on the special 2-by-2 network tomography problem, a dichotomy of the possible 2-by-2 topologies was identified. The distinguishing property of the shared topology is that estimates generated by sources from the independent measurements can be averaged when links are shared, to obtain lower variance estimates. Because some links are shared, the overall number of links in the topology is also reduced in the shared case. This leads to thinking of the problem in terms of model-order complexity. To determine whether the topology of an unknown 2-by-2 network is shared or not shared a multiple source probing algorithm was designed using packet arrival order measurements. By utilizing a randomized probing scheme the algorithm does not necessitate a specialized timing infrastructure, making it practical to implement. It was then shown how back-to-back packet measurements used to assess internal network performance could be integrated into the probing scheme. Consequently, no more probes are used than would be if the sources were to independently implement existing single-source network tomography algorithms. However, by cooperating in the probing procedure, the sources are able to extract more information about the

network. Finally, a decision-theoretic framework was developed to combine performance measurements and arrival order measurements into one test, rather than using the typical two-step approach of first characterizing the topology and then inferring internal performance. Parameters of the algorithm are easily set. The approach developed in this thesis benefits because of jointly treating topology and performance measurements, as these two network characteristics each impose constraints on the other. Results from model-based simulations indicate the potential of this algorithm.

Current ongoing work involves testing the multiple source probing procedure in a network simulator in order to better understand the effects of cross-traffic on the algorithm. Additionally, an implementation is being assembled for tests on the Internet to further assess performance.

Additionally, the theoretical contributions of this thesis open the door to further study of the general network tomography problem. It remains for a probing procedure to be developed which can make performance measurements from two sources to a single receiver in order to obtain the 2-by-1 components of a more general network tomography problem. Within the decomposition framework established in this thesis, this is a major task. Completely solving the 2-by-2 network tomography problem is a large step towards understanding the general network tomography problem so that more scalable algorithms can be designed.

Network tomography algorithms have the potential to monitor topological and performance related trends over time. In addition to tracking the performance of the Internet, such tools provide a mechanism for identifying phenomena related to correlations between these characteristics. The goal of network tomography, then, is to design tools for deployment throughout the Internet in a large-scale network measurement infrastructure.

References

1. A. Adams, T. Bu, R. Cáceres, N. Duffield, T. Friedman, J. Horowitz, F. Lo Presti, S. Moon, V. Paxson, and D. Towsley. The use of end-to-end multicast measurements for characterizing internal network behavior. *IEEE Communications Magazine*, May 2000.

2. J. Bellardo and S. Savage. Measuring packet reordering. In *Proceedings of the ACM Sigcomm Internet Measurement Workshop*, Marseille, France, November 2002.

3. A. Bestavros, J. Byers, and K. Harfoush. Inference and labeling of metric-induced network topologies. Technical Report BUCS-TR-2001-010, Computer Science Department, Boston University, Boston, MA, May 2001.

4. 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.

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

6. R. Castro, M. Coates, M. Gadhiok, R. King, R. Nowak, E. Rombokas, and Y. Tsang. Maximum likelihood network topology identification from edge-based unicast measurements. Technical Report TR-0107, ECE Department, Rice Univeristy, Houston, TX, November 2001.

7. 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.

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

9. M. Coates and R. Nowak. Network inference from passive unicast measurements. Technical Report TR-0002, ECE Department, Rice University, Houston, TX, January 2000.

10. M. Coates and R. Nowak. Unicast network tomography using em algorithms. Technical Report TR-0004, ECE Department, Rice University, Houston, TX, September 2000.

11. M. J. Coates and R. D. Nowak. Sequential monte carlo inference of internal delays in nonstationary communication networks. *IEEE Transactions on Signal Processing*, 50(2):366–376, February 2002.

12. N. 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.

13. N. 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.

14. N. Duffield, J. Horowitz, F. Lo Presti, and D. Towsley. Network delay tomography from end-to-end unicast measurements. In *Proceedings of the 2001 International Workshop on Digital Communications*, Taormina, Italy, September 2001.

15. N. G. Duffield, J. Horowitz, and F. Lo Presti. Adaptive multicast topology inference. In *Proceedings of IEEE Infocom*, Anchorage, Alaska, April 2001.

16. 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.

17. 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.

18. 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.

19. K. Harfoush, A. Bestavros, and J. Byers. Robust identification of shared loss using end-to-end unicast probes. In *Proceedings of IEEE ICNP*, Osaka, Japan, October 2000.

20. T. E. Ng and H. Zhang. Predicting internet network distance with coordinates-based approaches. In *Proceedings of IEEE Infocom*, New York, NY, June 2002.

21. R. D. Nowak and M. J. Coates. Unicast network tomography using the EM algorithm. Submitted to IEEE Transactions on Information Theory, November 2001.

22. 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.

23. S. Ross. *Introduction to Probability Models*. Academic Press, 2002.

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

25. Y. Vardi. Network tomography: Estimating source-destination traffic intensities from link data. *Journal of the American Statistical Association*, 91(433):365–377, March 1996.

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

27. Y. Zhang, V. Paxson, and S. Shenker. The stationarity of internet path properties: Routing, loss, and throughput. Technical report, ACIRI, May 2000.