This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

Optimal probing for unicast network delay tomography

Yu Gu, Guofei Jiang, Vishal Singh and Yueping Zhang
NEC Laboratories America, Inc.
4 Independence Way, Princeton, NJ 08540, USA
Email: {yugu, gfj, vishal, yueping}@nec-labs.com

***Abstract*—Network tomography has been proposed to ascertain internal network performances from end-to-end measurements. In this work, we present *priority probing*, an optimal probing scheme for unicast network delay tomography that is proven to provide the most accurate estimation. We first demonstrate that the Fisher information matrix in unicast network delay tomography can be decomposed into an additive form where each term can be obtained numerically. This establishes the space over which we can design the optimal probing scheme. Then, we formulate the optimal probing problem into a semi-definite programming (SDP) problem. High computation complexity constrains the SDP solution to only small scale scenarios. In response, we propose a greedy algorithm that approximates the optimal solution. Evaluations through simulation demonstrate that priority probing effectively increases estimation accuracy with a fixed number of probes.**

I. Introduction

Network tomography has been proposed to ascertain the performance of internal networks from end-to-end measurements (see [1], [2] and the references there in). By sending probing packets from a source and collecting the delay and loss information of these packets from receivers at the edge of the network, network tomography is able to infer delay and loss performances of internal network links. This ability makes network tomography especially appealing to applications that concern network performances but don't have access to internal network states [3].

Existing unicast network delay tomography techniques have developed maximum likelihood estimators of the network delay distributions. These estimators converge to the true distributions with enough probing traffic. However it remains an open question that *with a fixed number of probing packets, how to obtain the most accurate estimators?* This is important for two reasons. First, given the fast transient nature of Internet traffic dynamics [4], [5], it is essential to obtain accurate estimations within a constrained time period, which implies limited probing packets. And second, it becomes more challenging when monitoring large scale networks since on one hand, low probing traffic volume is required to have minimal impact on network delay and on the other hand, enough probing traffic is required to cover the entire network in a short time period.

In this work, we propose *priority probing*: algorithms that construct an optimal probing set that is proven to offer the most accurate estimation with a fixed number of probing packets. The probing set defines the distribution of probing packets among the receivers, which in turn, determines the covariance matrix of the maximum likelihood estimators. The covariance matrix corresponds to the accuracy of the estimators. In particular, its diagonal elements are the variances of the estimators. And in this work, we propose an optimal probing set that leads to the minimum trace of the covariance matrix, i.e. the sum of variances.

The development is presented in two steps. In the first step, we prove that the Fisher information matrix for unicast network delay tomography can be decomposed into an additive form where each term can be obtained numerically. This additive form establishes the space over which we can design the optimal probing set. In the second step, we demonstrate that designing the optimal probing set is an optimal experiment design problem [6] and can be formulated in a semi-definite programming (SDP) form [7]. However, the SDP approach is constrained to small scale scenarios due to expensive computational costs associated with computing the full Fisher information matrix. In response, we propose a greedy algorithm that significantly saves both computational space and time.

The optimal probing set is determined by the network topology and the network delay distributions. While the true delay distributions may not be available, "local optimal solutions" can be obtained using previous measurement results or assuming no knowledge about the delay distributions. Simulation results demonstrate that, by utilizing the network topology information, and sometimes with previous measurement results, priority probing outperforms existing approaches without knowing the true delay distributions.

Evaluations through simulation demonstrate that priority probing is very efficient in increasing estimation accuracy and reducing probing traffic. In a network consisting of 100 end hosts, priority probing uses only about 1000 pairs of probing packets to achieve the same accuracy as existing approach using 7000. With a moderate probing rate of $20kbps$, this is equivalent to, without compromising accuracy, increasing the monitoring resolution from one measurement result every 6 minutes to one every 50 seconds.

The rest of the paper is organized as follows. Section II gives a brief review of unicast based network delay tomography and the upward-downward algorithm. Section III presents our main theorem that decomposes the Fisher information

978-1-4244-5837-0/10/$26.00 ©2010 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

matrix into additive forms. Section IV proposes the priority probing algorithms, which formulates the optimal experiment design problem and solves it using both SDP and the greedy algorithm. Section V presents evaluation results exhibiting the advantages of priority probing over existing approaches. We review related works in Section VI and conclude the paper in Section VII.

II. Background

We first introduce the terminologies used in this paper in the form of a brief review of unicast network delay tomography [8], [9] and the upward-downward algorithm [10].

A. Unicast based network delay tomography

Consider a source s and a set of n receivers $\mathcal{R} = \{1 \dots n\}$. Assume that the paths from s to $\mathcal{R}$ are stationary and form a logical tree topology $\mathcal{T} = (V, E)$. V is the set of vertices including the source s, the receivers $\mathcal{R}$ and the set of branch points. A branch point is a vertex where two paths split. E is the set of links in $\mathcal{T}$. A link $l \in E$ is a path segment between two vertices in V without passing any other vertex. Here the logical tree is an overlay network obtained from the paths between the source and the receivers and a link in the logical tree can correspond to multiple physical links. In network tomography, this logical tree topology is assumed to be known by polling routing information from routers. We also denote by L_i the set of links on the path from s to receiver i and $L_{i,j} = L_i \cup L_j$.

The measurements are performed by sending a pair of back-to-back probing packets to two distinct receivers at a time. It is expected that the two packets experience the same or similar delays on the shared links, which brings in correlations in their delay observations [9], [8]. Let (r_1^t, r_2^t) represent the receiver pair at round t and S be a sequence of T receiver pairs: $S = \{(r_1^t, r_2^t)\}$, $r_1^t \neq r_2^t \in \mathcal{R}$, $t = 1 \dots T$. Note the same pair can appear multiple times in S and be probed multiple times accordingly. For easy presentation, we assume that the probing packets are not lost and denote the delays of these two packets as an observation pair $(o_{r_1^t,t}, o_{r_2^t,t})$. Probes are sent in multiple rounds and a set of observation pairs $\{(o_{r_1^t,t}, o_{r_2^t,t})\}$ is obtained accordingly.

All delay observations are then adjusted by deducting from them the smallest delay seen on the corresponding path. It is assumed that the smallest delay represents transmission delay and propagation delay along the path and the rest of the delay represents the dynamic part, i.e. queuing delay caused by network congestion. As a consequence, unicast delay tomography monitors the dynamic delay changes in the network, which is important for applications like content streaming and online gaming. Also, it does not require clock synchronization among the source and the receivers. Clock drifting will affect the estimation accuracy, but it is usually small enough to be ignored.

Let D_l be the queuing delay on link l. D_l is a random variable and the goal of network delay tomography is to estimate the distribution of D_l for all links. Delays are discretized into k bins $B_0 = [0, \tau)$, $B_1 = [\tau, 2\tau)$, …, $B_{k-1} = [(k-1)\tau, \infty)$ where τ is a predefined constant bin size. For a link $l \in E$, its delay D_l is then discretized into a categorical distribution random variable with parameters $\{\theta_{l,d}\}$, $0 \leq d \leq k-1$ and $\sum_d \theta_{l,d} = 1$. Link delays are assumed to be independent across links and estimating link delay distributions then turns into estimating $\overrightarrow{\theta}$, the parameters of the independent categorical delay distributions.

With $\{(o_{r_1^t,t}, o_{r_2^t,t})\}$, the set of observations collected from multiple rounds, the log likelihood function is

$$l(\overrightarrow{\theta}) = \sum_t \log\Big(\sum_{\{(l,d)\} \in \Omega(o_{r_1^t,t}, o_{r_2^t,t})} \prod_{l \in L_{r_1^t, r_2^t}} \theta_{l,d} \Big), \quad (1)$$

where $\Omega(o_{r_1^t,t}, o_{r_2^t,t})$ is all the possible combinations of link delays that satisfy the observation $(o_{r_1^t,t}, o_{r_2^t,t})$. For clear presentation, we use Ω instead of $\Omega(o_{r_1^t,t}, o_{r_2^t,t})$ in the rest of the paper.

A direct expression for $\{\theta_{l,d}\}$ that maximizes (1) is difficult to obtain. As an alternative, an Expectation Maximization (EM) algorithm is proposed that iteratively converges to the maximum likelihood estimator (MLE) of the distribution of network internal link delays [8], [9].

One of the key steps in the EM algorithm developed is to obtain $P(D_l = d | O_{r_1} = o_{r_1}, O_{r_2} = o_{r_2})$, the conditional probability of a link l having delay d given a pair of observations. This is achieved by the upward-downward algorithm.

B. Upward-Downward algorithm

The upward-downward algorithm [10] is developed to calculate belief propagation in causal trees, a special case of Bayesian networks. It efficiently calculates $P(\{O_v = o_v\})$, the probability that a subset of vertices in the tree, $\{v\}$, takes certain states $\{O_v = o_v\}$. In unicast network delay tomography, it computes $P(D_l = d | O_{r_1} = o_{r_1}, O_{r_2} = o_{r_2})$ by calculating $P(O_{r_1} = o_{r_1}, O_{r_2} = o_{r_2})$, the likelihood of observing a pair of observations (o_{r_1}, o_{r_2}), and $P(O_{v_1} = o_{v_1}, O_{v_2} = o_{v_2}, O_{r_1} = o_{r_1}, O_{r_2} = o_{r_2})$, the likelihood of observing a pair of observations (o_{r_1}, o_{r_2}) and a link l from v_1 to v_2 having delay $d = o_{v_2} - o_{v_1}$. We omit detailed steps due to space limitations and interested readers can refer to [11], [8]. These results will be used to calculate the Fisher information matrix in our later development.

The upward-downward algorithm is a polynomial time algorithm. In general, for a b-nary tree of depth q and k states per vertex, its complexity is $O(bqk^2)$. In network tomography, since for each pair of receivers, there is only one branch point with two children, the complexity is $O(qk^2)$.

III. The Fisher Information Matrix

The focus of this work is on designing S, the sequence of probing pairs used in unicast network delay tomography. Existing approaches select the probing pairs uniformly among all pairs of receivers. For example, in [8], each pair of receivers gets to be selected in a round robin fashion, and in [11], all pairs of receivers have an equal opportunity to be selected at each round.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

In this section, we will see that, each time a pair of receivers is probed, it brings new information into estimation. The amount of information brought in is determined by the structure of the two paths and the delay distributions. Generally, the more information we have, the more accurate the estimation is. Therefore, by properly selecting pairs of end hosts, we can design an S that maximizes the information.

The expected information available for estimation is naturally expressed by the Fisher information matrix [12]. One of our main results is the following theorem:

Theorem 3.1: Let $\omega(r_1, r_2)$ be the number of times (r_1, r_2) is probed according to S. The Fisher information matrix $\mathcal{I}$ for unicast network delay tomography can be calculated as

$$\mathcal{I} = \sum_{(r_1,r_2)} \omega(r_1, r_2)\mathcal{I}(r_1, r_2). \tag{2}$$

where $\mathcal{I}(r_1, r_2)$ is a function of (r_1, r_2) that can be computed numerically.

Theorem 3.1 demonstrates how the probing pair set S determines the Fisher information matrix $\mathcal{I}$ in an additive manner. This establishes the foundation over which our optimization is carried out in the next section.

The rest of the section develops a proof of Theorem 3.1, including how $\mathcal{I}(r_1, r_2)$ are obtained numerically. The development is presented in three steps. We first describe how to calculate the observed information matrix (OIM) $\mathcal{J}$, as the Fisher information matrix is the expectation of $\mathcal{J}$. Then we obtain $\mathcal{I}(r_1, r_2)$, the expected contribution to the OIM from a pair of receivers (r_1, r_2). And finally, we will see that $\mathcal{I}$ is a linear combination of $\mathcal{I}(r_1, r_2)$ with coefficients $\omega(r_1, r_2)$. At the end of the section, we also discuss the computation complexity to obtain $\mathcal{I}$.

A. The OIM

The OIM of an MLE, $\mathcal{J}$, quantifies the inverse of the covariance matrix of the estimator after a set of observations is obtained. It is defined as

$$\mathcal{J} = -\nabla\nabla^\top l(\overrightarrow{\theta}), \tag{3}$$

where $l(\overrightarrow{\theta})$ is the log likelihood function (1). For each of the matrix element in $\mathcal{J}$ we have

$$\frac{\partial^2 l(\overrightarrow{\theta})}{\partial\theta_{l',d'}\partial\theta_{l'',d''}} = \sum_t \frac{A\mathcal{L}_t - \mathcal{D}_{t,l',d'}\mathcal{D}_{t,l'',d''}}{\mathcal{L}_t^2}, \tag{4}$$

where

$$\mathcal{L}_t = \sum_{\{(l,d)\}\in\Omega} \prod_{l\in L_{r_1^t,r_2^t}} \theta_{l,d} \tag{5}$$

is the likelihood for a single pair of observations $(o_{r_1^t,t}, o_{r_2^t,t})$,

$$\mathcal{D}_{t,l',d'} = \sum_{\{(l,d)\}\in\Omega, D_{l'}=d'} \prod_{l\in L_{r_1^t,r_2^t}, l\neq l'} \theta_{l,d}, \tag{6}$$

is $\mathcal{L}_t$'s derivative with regard to parameter $\theta_{l',d'}$, and

$$A = \begin{cases} 0, & l' = l''; \\ \sum\limits_{\substack{\{(l,d)\}\in\Omega, \\ D_{l'}=d', D_{l''}=d''}} \prod\limits_{\substack{l\in L_{r_1^t,r_2^t}, \\ l\neq l', l\neq l''}} \theta_{l,d}, & otherwise \end{cases} \tag{7}$$

is $\mathcal{L}_t$'s second order derivative with regard to parameters $\theta_{l',d'}$, $\theta_{l'',d''}$. In particular, we have the diagonal element for a particular parameter θ_{l^*,d^*}

$$\frac{\partial^2 l(\overrightarrow{\theta})}{\partial\theta_{l^*,d^*}^2} = -\sum_t \left(\frac{\mathcal{D}_{t,l^*,d^*}}{\mathcal{L}_t}\right)^2. \tag{8}$$

Equation (4) indicates that each pair of observations contributes additively to the OIM. In the following, we demonstrate how to calculate quantitatively this additive contribution from each pair of observations, and therefore, the whole OIM after a set of probing pairs.

We define

$$J_{l',d',l'',d''}(o_{r_1}, o_{r_2}) = \frac{A\mathcal{L} - \mathcal{D}_{l',d'}\mathcal{D}_{l'',d''}}{\mathcal{L}^2} \tag{9}$$

as the term in the summation in (4) and

$$\mathcal{J}(o_{r_1}, o_{r_2}) = \left(J_{l',d',l'',d''}(o_{r_1}, o_{r_2})\right) \tag{10}$$

as the corresponding matrix defining the contribution to $\mathcal{J}$ made by a pair of observations (o_{r_1}, o_{r_2}), where (l', d') defines the row index and (l'', d'') defines the column index. Note that from now on, we omit the round index t whenever there is no ambiguity.

We now demonstrate how to calculate $\mathcal{L}$, A and $\mathcal{D}_{l^*,d^*}$ numerically so that we can evaluate (9). $\mathcal{L}$ is the likelihood function for a single pair of observations (o_{r_1}, o_{r_2}). As described in Section II, we can obtain it directly using the upward-downward algorithm by assigning the two leave vertices their observed delay bins.

The key to calculating $\mathcal{D}_{l^*,d^*}$ and A is the following two observations:

$$P(D_{l^*} = d^*, o_{r_1}, o_{r_2}) = \mathcal{D}_{l^*,d^*}\theta_{l^*,d^*} \tag{11}$$

and

$$P(D_{l'} = d', D_{l''} = d'', o_{r_1}, o_{r_2}) = A\theta_{l',d'}\theta_{l'',d''} \tag{12}$$

for $l' \neq l''$.

With these observations and our introduction of the upward-downward algorithm in Section II, $\mathcal{D}_{l^*,d_l^*}$ and A can be readily obtained. Here, as described in Section II, $\mathcal{D}_{l^*,d^*}$ is obtained by evaluating the probabilities $P(O_{v_1}, O_{v_2}, O_{r_1} = o_{r_1}, O_{r_2} = o_{r_2})$ where v_1 and v_2 are determined by l^* and their states determined by d^*. And calculating A is more complicated since it requires evaluating the probabilities $P(O_{v_1'}, O_{v_2'}, O_{v_1''}, O_{v_2''}, O_{r_1} = o_{r_1}, O_{r_2} = o_{r_2})$ in the causal tree, where v_1' and v_2' are determined by l', v_1'' and v_2'' are determined by l'' and their states determined by d' and d'' respectively.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

We then have

$$\mathcal{J} = \sum_{(r_1^t, r_2^t) \in S} \mathcal{J}(o_{r_1^t}, o_{r_2^t}).$$

B. Expected impact from a pair of receivers

Now we further extend the technique and evaluate $\mathcal{I}(r_1, r_2)$, the expected contributions to the OIM from probing the pair (r_1, r_2). This quantifies the average impact on estimation accuracy by probing a pair of receivers and we will see that this is actually the building piece of the Fisher information matrix.

The derivation is straightforward as

$$\mathcal{I}(r_1, r_2) = E\mathcal{J}(o_{r_1}, o_{r_2}).$$

Let $\left(I_{l',d_l',l'',d_l''}(r_1, r_2)\right)$ be the elements in $\mathcal{I}(r_1, r_2)$. Note that $\mathcal{L} = P(o_{r_1}, o_{r_2})$, the likelihood function for a single observation (o_{r_1}, o_{r_2}), we then have

$$\begin{aligned} I_{l',d_l',l'',d_l''}(r_1, r_2) &= E J_{l',d_l',l'',d_l''}(o_{r_1}, o_{r_2}) \\ &= \sum_{o_{r_1}, o_{r_2}} A - \frac{\mathcal{D}_{l',d_l'} \mathcal{D}_{l'',d_l''}}{\mathcal{L}} \end{aligned} \quad (13)$$

C. The Fisher information matrix

Since the Fisher information matrix $\mathcal{I} = E\mathcal{J}$, $\mathcal{J} = \sum_{(r_1^t, r_2^t) \in S} \mathcal{J}(o_{r_1^t}, o_{r_2^t})$, and $\mathcal{I}(r_1^t, r_2^t) = E\mathcal{J}(o_{r_1^t}, o_{r_2^t})$, we have, given S, the Fisher information matrix

$$\mathcal{I} = \sum_{(r_1^t, r_2^t) \in S} \mathcal{I}(r_1^t, r_2^t). \quad (14)$$

Let $\omega(r_1, r_2)$ be the number of probing pair (r_1, r_2) in S, we then obtain Theorem 3.1.

D. Computation complexity

We conclude this section by a discussion of the computation complexity required to obtain the Fisher information matrix.

According to (13), calculating one element $I_{l',d_l',l'',d_l''}(r_1, r_2)$ in $\mathcal{I}(r_1, r_2)$ requires evaluating $\mathcal{L}$, $\mathcal{D}_{l,d}$, and A over all possible observations. With a fixed pair of observation, the complexity of obtaining $\mathcal{L}$ is the same as that of the Upward-Downward algorithm, $O(qk^2)$, where q is the depth of the topology tree and k is the number of bins. The computation complexity of obtaining $\mathcal{D}_{l,d}$ is $O(qk^3)$, as on average, $O(k)$ cases require to be considered when calculating $\mathcal{D}_{l,d}$. And similarly, the computation complexity of obtaining A is $O(qk^4)$. As we need to evaluate $\mathcal{L}$, $\mathcal{D}_{l,d}$, and A for all possible observation pairs, the computation cost for evaluating an element in $\mathcal{I}(r_1, r_2)$ is then $O(qk^6)$. For the subtree formed by paths to r_1 and r_2, the corresponding sub-matrix dimension is $O((qk)^2)$. As a consequence, the computation cost for evaluating $\mathcal{I}(r_1, r_2)$ is $O(q^3k^8)$.

Notice that if we only calculate the diagonal elements, $I_{l,d,l,d}(r_1, r_2)$, it only requires to obtain $\mathcal{L}$ and $\mathcal{D}_{l,d}$, as $A = 0$ for diagonal elements. In this case, the computation cost of obtaining the diagonal elements is $O(q^2k^6)$, much more efficient than that of obtaining the whole matrix.

IV. Priority probing

Theorem 3.1 demonstrates how the probing sequence S determines the Fisher information matrix $\mathcal{I}$. The inverse of the Fisher information, $\mathcal{I}^{-1}$, is the covariance matrix of the MLE, which determines the estimation accuracy. As a consequence, by properly design S, we can obtain a covariance matrix $\mathcal{I}^{-1}$ that leads to the most accurate estimators. This is an optimal experiment design problem [6] and we call the resulting probing schemes *priority probing*.

There are several optimal criteria for optimal experiment design problems, including E-Optimality, which minimizes the maximum eigenvalue of $\mathcal{I}^{-1}$; A-Optimality, which minimizes the trace of $\mathcal{I}^{-1}$; and D-Optimality, which minimizes the determinant of $\mathcal{I}^{-1}$. All of them can be properly formulated into semi-definite programming (SDP) problems [7]. Here, we focus on A-Optimality since the trace of $\mathcal{I}^{-1}$ has a direct explanation as the sum of estimation variances.

A. SDP approach

Let T be the predefined size of S, A-Optimality problem is then to determine $\omega(r_1, r_2)$, the number of probing pair (r_1, r_2) in S, that minimizes the trace of $\mathcal{I}^{-1}$. This can be formalized in a convex optimization form:

$$\begin{array}{ll} \min & \sum t_i \\ \text{s.t.} & t_i \geq e_i' \mathcal{I}^{-1} e_i \quad i = 1, \ldots, k|E| \\ & \mathcal{I} = \sum_{(r_1,r_2)} \omega(r_1, r_2) \mathcal{I}(r_1, r_2) \\ & \sum_{(r_1,r_2)} \omega(r_1, r_2) = T \\ & \omega(r_1, r_2) \geq 0 \end{array} \quad (15)$$

where e_i is the i^{th} unit-vector of $\mathbb{R}^{k|E|}$ and t_i corresponds to the i^{th} diagonal element in $\mathcal{I}^{-1}$. Using Schur complement [7], we can rewrite the first set of inequality constraints in a semi-definite form

$$\begin{pmatrix} \mathcal{I} & e_i \\ e_i' & t_i \end{pmatrix} \succeq 0.$$

The optimization problem (15) can then be solved using SDP.

While the SDP approach provides the optimal probing set, we have seen in the previous section that the computation cost for evaluating $\mathcal{I}(r_1, r_2)$ is $O(q^3k^8)$. It grows fast as the topology tree becomes higher and, more seriously, as the number of bins increases. In response, we develop a greedy approach that requires much less computation cost and whose performances are close to the optimal as presented in the simulation results.

B. A greedy approach

We elect to approximate the diagonal of the covariance matrix by considering only the diagonal elements in $\mathcal{I}$ and assuming all other elements are 0. This is the best bet without calculating the off-diagonal elements, which have an expensive computation cost of $O(q^3k^8)$. In Section V, we will see that this approach provides a close approximation to the optimal solution and the experimental results demonstrate that besides faster and easier computation, this approximation also provides more accurate estimation than existing approaches.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

We define $\mathcal{I}(t)$ as the Fisher information matrix of the MLE after t rounds of measurements. And let $\Lambda(t)$ be the set of diagonal elements of $\mathcal{I}(t)$ and $\lambda_{l,d}(t) \in \Lambda(t)$ be the diagonal elements. As we consider only the diagonal elements of $\mathcal{I}(t)$, the sum of estimation variances is then

$$S(t) = \sum_{l,d} |\lambda_{l,d}^{-1}(t)|. \tag{16}$$

The greedy algorithm targets at minimizing $S(t)$ at each step t. We first obtain, for each pair of receivers (r_1, r_2), $\{I_{l,d,l,d}(r_1, r_2)\}$, the set of expected contributions to the diagonal elements $\{\lambda_{l,d}(t)\}$ of the Fisher information matrix.

The algorithm maintains the state of $\Lambda(t)$. Initially, all elements of $\Lambda(0)$ are set to 0. At round t, we evaluate for each pair (r_1, r_2) the expected value of $\Lambda(t)$ after round t

$$\lambda_{l,d}(r_1, r_2) = \lambda_{l,d}(t-1) + I_{l,d,l,d}(r_1, r_2) \tag{17}$$

and obtain the following metric

$$S(r_1, r_2) = \sum_{l,l_d} |(\lambda_{l,d}(r_1, r_2))^{-1}|. \tag{18}$$

After the evaluation is performed over all pairs, the pair (r_1^*, r_2^*) that leads to the minimum $S(r_1, r_2)$ is selected for round t. Then $\Lambda(t)$ is updated as

$$\lambda_{l,d}(t) = \lambda_{l,d}(t-1) + I_{l,d,l,d}(r_1^*, r_2^*), \tag{19}$$

And the pair for round $t+1$ is selected similarly.

Also in the first few rounds, since there are 0's in $\Lambda(t)$, $S(r_1, r_2)$ cannot be evaluated and those pairs that remove the most number of 0's are selected. After there is no 0 in $\Lambda(t)$, the above procedure is repeated until $t = T$. Figure 1 demonstrates the greedy algorithm in pseudo code.

The computation complexity of generating a sequence of probing pairs using the greedy approach is determined by two parts: first, to obtain the diagonal elements $I_{l,d,l,d}(r_1, r_2)$ from all pairs of receivers, and second, to generate the probing sequence. From the previous section, the total cost of obtaining $I_{l,d,l,d}(r_1, r_2)$ for a single pair (r_1, r_2) is $O(q^2k^6)$ and therefore the computation cost for the first part is $O(n^2q^2k^6)$. Once $I_{l,d,l,d}(r_1, r_2)$ is obtained, the complexity of generating the probing sequence is $O(Tn^3k)$, evaluating all pairs of receivers at each round. In this work, we have implemented the greedy algorithm in Java and during our experiments, 7000 probing pairs of a network containing 100 receivers with 5 delay bins can be generated within a few minutes using a commodity Intel(R) Xeon(R) $1.86GHz$ processor and our code is far from optimized.

Require: $\{I_{l,d,l,d}(r_1, r_2)\}$ for all pairs (r_1, r_2)
  $\Lambda(t) = 0$
  $t = 1$
  while $t \leq T$ **do**
    if there is 0 in $\Lambda(t)$ **then**
      $Pairs(t) = (r_1, r_2)$ that removes maximum number of 0 in $\Lambda(t)$
    else
      $MinimumS = A\_LARGE\_NUMBER$
      for all $(r_1,\ r_2)$ **do**
        for all $l,\ l_d$ **do**
          $\lambda_{l,d}(r_1, r_2) = \lambda_{l,d}(t-1) + I_{l,d,l,d}(r_1, r_2)$
        end for
        $S(r_1, r_2) = \sum_{l,l_d} |(\lambda_{l,d}(r_1, r_2))^{-1}|$
        if $S(r_1, r_2) < MinimumS$ **then**
          $MinimumS = S(r_1, r_2)$
          $Pairs(t) = (r_1, r_2)$
        end if
      end for
    end if
    for all $l,\ l_d$ **do**
      $\lambda_{l,d}(t) = \lambda_{l,d}(t-1) + I_{l,d,l,d}(Pairs(t))$
    end for
    $t = t + 1$
  end while
  return $Pairs$

Fig. 1. Pseudocode for the greedy algorithm

C. Missing information

The optimal probing sequence is determined by both the network topology and the network delay distribution. Existing approaches, i.e. the uniform probing schemes, utilize neither. Priority probing utilizes both information. However, before any measurement is performed, the delay distribution information is unknown. This is referred to as local optimality in the literature [13], [14].

One of the common practices in this situation is to utilize only the topology information and assume no knowledge about the delay distributions. We will see in the next section that most of the time this generates better estimations than the uniform approaches. Also, after one set of probing is performed, estimations of the delay distribution can be obtained. These estimations can then be used as input to the priority probing algorithm and generate a second probing set. In one of our experiments presented in Section V, we will see one example of this approach. There are other approaches such as sensitivity analysis of the probing set regarding the delay distribution and we leave them as future works.

V. Simulation results

In this section, we present evaluation results for priority probing. We prefer the method of simulation as it enables us to record the true network performances and provides a controlled environment for an objective comparison between priority probing and existing approaches. Evaluation in real network environment is currently under study and will be presented in our future works together with other practical considerations on applying network delay tomography in real networks.

The experimental results demonstrate that the probing sequences generated by priority probing effectively improve the accuracy obtained by network tomography. Starting from a simple small topology to a large topology containing 310

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

nodes and 483 links, we demonstrate that as the tomography tree becomes larger and more complex, the benefit of the priority probing scheme becomes more significant. In our largest experiment, which consists of 100 end hosts, priority probing uses only about 1000 probing pairs to achieve the same accuracy as existing approach using 7000 pairs, which increases the monitoring resolution from one estimation result every 6 minutes to one every 50 seconds.

A. Simulation setup

For each experiment, we specify the topology and the receivers in ns-2. The tree topology from the source to the receivers is calculated using ns-2's built-in routing algorithm. This tree topology is used as an input to our priority probing algorithm. Our priority probing algorithm also takes as input the performances of links in the tree topology if available. In the following experiments, unless specified, the priority probing algorithm assumes a uniform distribution over the parameter space for all link parameters. The generated probing pairs are then used as input to the simulation. As comparison, we also generate uniform probing sequences where all pairs of receivers are probed in a round robin fashion. Each simulation is therefore carried twice. In the first run, the priority probing sequence is used and in the second run, the uniform probing sequence is used.

The background traffic is a mixture of TCP sessions and Pareto on-off UDP sources. The TCP sessions are generated by the PackMime-HTTP [15] web traffic generator. The average number of connections per second is set to 100. Both the burst time and the idle time of the Pareto traffic are set to $2s$ with shape parameter set to 1.5. When the state is on, the Pareto traffic generator sends UDP packets with a constant rate of $3Mbps$ and the UDP packet size is set to 600 bytes.

This background traffic is applied to individual logical links in the tomography tree during the experiments. This is because 1) we are targeting at an objective comparison between priority probing and uniform probing and therefore follow the independent link behavior assumption made by the network tomography algorithm; and 2) a logical link in the network tomography tree potentially corresponds to a number of physical links in the underlying network and the aggregation effects from traffic on these physical links can present as independent behaviors among the logical links.

In each experiment, 7000 probing pairs are generated from each probing scheme. 7000 is selected because we observe that at in all cases, the decreasing of the estimation errors has become small before 7000 rounds of probes are performed. Every $50ms$, a pair of receivers are probed. The size of the probing packet is 64 bytes. This results in a probing data rate of $20.48kbps$ from the source. The simulation finishes after all 7000 rounds of probing are done, about 350 seconds.

For all the delays collected, the minimum delay ever observed on the corresponding path is deducted. Then, they are discretized into 5 bins with $10ms$ for each bin. If a probing packet is lost, it is assigned a delay in the largest bin. Delay distributions of the links in the tree topology are then obtained using the EM algorithm developed in [8], [9].

In order to obtain the ground truth, we record the queuing delays of all the links in the tree topology every $10ms$ by sampling the queue lengths. When the simulation is finished, these queuing delays are also discretized into the 5 bins and the resulting empirical probability distributions are used as $\{\theta_{l,d}\}$, the true link delay parameters.

We measure the accuracy of the estimation using averaged sum of absolute errors over all link parameters, a metric that is consistent with the goal of priority probing.[1] In order to demonstrate the estimation accuracy as the number of probing pairs increases, we define

$$E(t) = \frac{\sum_{l,d} |\hat{\theta}^t_{l,d} - \theta_{l,d}|}{L \times k} \tag{20}$$

as the averaged sum of errors after t rounds of probings, where $\hat{\theta}^t_{l,d}$ is the estimation of $\theta_{l,d}$ after t round. $E(t)$ ranges from 0 to $2/k = 0.4$, with the number of bins $k = 5$. The smaller $E(t)$ is, the more accurate the estimations are, with $E(t) = 0$ when the estimation is exactly the ground truth. We didn't take relative errors since some $\theta_{l,d} = 0$.

We use YALMIP [16] to model the optimization problem and SDPA [17] to solve the SDP problem. The simulations are performed using the network simulator ns-2 [18].

B. Simulation results

We now present some of the simulation results we obtained. All the following experiments have been duplicated using multiple random seeds and similar results are observed.

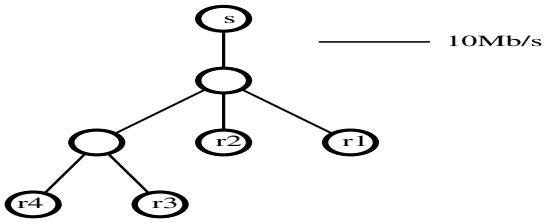

Fig. 2. A simple topology illustrating priority probing

1) An illustrating example: We first demonstrate the sequences generated by both the SDP approach (the optimal solution) and the greedy algorithm using a simple topology as in Figure 2. There are four receivers, two of them have a distance of two hops from the source and the other two have a distance of three hops. Intuitively, if we consider only the topology and ignore the actual link delay distribution, we would expect more probings on receivers further away from the source as more links required to be evaluated on the paths to these receivers.

And this is exactly what the SDP approach and the greedy algorithm generate with an assumption of uniform distribution

[1] Strictly speaking, benefits from second order statistics should be validated using empirical deviations. Here we take typical simulation results instead because the simulation usually takes quite long time.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

pair	SDP	greedy	uniform
(r_1, r_2)	3198	1317	1170
(r_1, r_3)	107	1243	1170
(r_1, r_4)	107	596	1170
(r_2, r_3)	107	595	1170
(r_2, r_4)	107	1243	1170
(r_3, r_4)	3374	2026	1170

TABLE I
DISTRIBUTIONS OF PROBING PAIRS

over the link parameters. Table I demonstrates the distribution of the 7020 probing pairs from the SDP approach and the greedy algorithm. Notice these receiver pairs form three types of topologies: (r_3, r_4) forms a topology where there are two links in the upstream branch; (r_1, r_3), (r_1, r_4), (r_2, r_3) and (r_2, r_4) all have one link in one downstream branch and two links in the other; and (r_1, r_2) leads to a regular three link binary tree. Accordingly, (r_3, r_4) gets the most probings. The same number of probings sent to (r_1, r_3), (r_1, r_4), (r_2, r_3) and (r_2, r_4) from the SDP approach reflect their equivalence in topology. The difference among (r_1, r_3), (r_1, r_4), (r_2, r_3) and (r_2, r_4) from the greedy algorithm is an artifact of the order in which these pairs are evaluated: (r_1, r_3) is always evaluated first, so it gets the most probings. This suppresses the number for (r_1, r_4) and (r_2, r_3), which then increases the number for (r_2, r_4). However, due to their topological equivalence, the distribution given by the greedy algorithm is equivalent to each of them getting the same probing.

We set all the link bandwidth to $10Mbps$ with a queue size of 200 packets. A simple drop-tail queue management is used. Assuming the packet size is 600 bytes, the queuing delay then ranges from $0ms$ to $96ms$. Background traffic is applied to all the links.

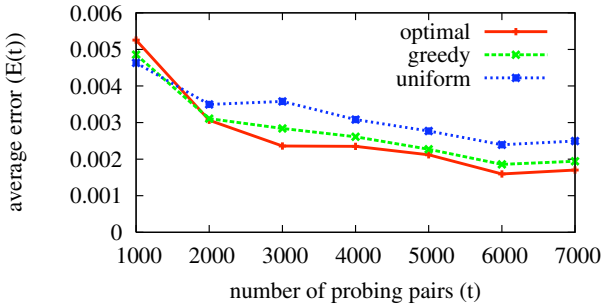

Fig. 3. optimal vs. greedy vs. uniform in the simple topology

Figure 3 presents the result of this experiment using the SDP approach, the greedy algorithm and the uniform approach. We see that in this small topology, the priority probing has exhibited its advantage over the uniform scheme and the greedy algorithm generates estimation results that are very close to the SDP approach. We shall see that as the topology gets larger and more complex, the advantage of using priority probing get more obvious.

2) A three level tree: Now consider a synthetic tree topology demonstrated in Figure 4. Here, each branch point has three children and the height of the tree is set to three. There are 27 leaves and each leave node corresponds to a receiver. For this case, it is already too expensive for us to compute the whole $\mathcal{I}(r_1, r_2)$ matrices and from now on we only present results from the greedy algorithm.

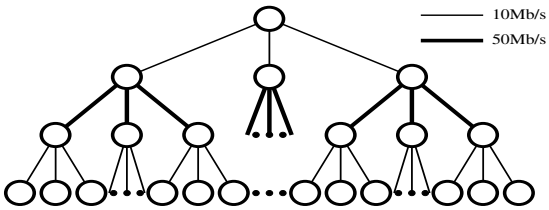

Fig. 4. A three level tree

The first level and third level links have a capacity of $10Mbps$ and the second level links have a capacity of $50Mbps$, reflecting that in real Internet slow access networks are connected by over-provisioned high bandwidth core networks. All links are set with a propagation delay of $5ms$. Since network tomography handles only dynamic part of the delay, this does not affect its accuracy. Background traffic is applied to all the top level links and half of the lower level links randomly selected.

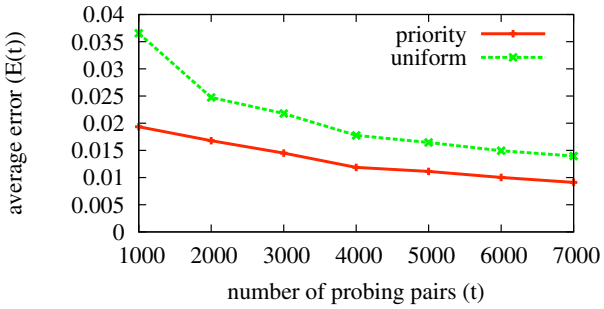

Fig. 5. priority vs. uniform in the three level tree topology experiment

Figure 5 presents the result of this experiment. We see that with only about 3000 pairs of probing, the priority probing scheme has achieved an accuracy reached by the uniform scheme with 7000 pairs. When both schemes use 7000 probes, the priority probing scheme is almost 50% more accurate than the uniform scheme.

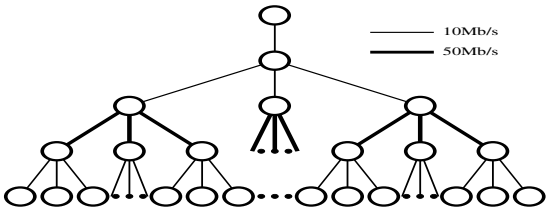

Fig. 6. A four level tree

3) A topology with one more link: Now we look at a topology with one more link than that in the previous experiment, as in Figure 6. In this case, everything is the same as in the

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2010 proceedings
This paper was presented as part of the main Technical Program at IEEE INFOCOM 2010.

previous experiment except a new link is added as the top level. The same type of background traffic is applied to the top link and, in order to diversify the situation, the background traffic in the middle link of the second level is removed.

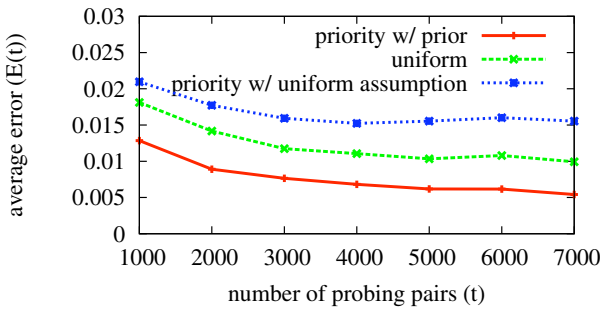

Fig. 7. benefit of delay information in the four level tree topology experiment

First, we still generate 7000 probing pairs using priority probing scheme with uniform link parameter distribution assumption. As the topology changes, the probing sequence generated by priority probing is different from that generated in the previous experiment.

Figure 7 presents the result of this experiment. We see that with the uniform distribution assumption of the link delays, the accuracy achieved by priority probing scheme is not as good as that by the uniform scheme. Situations like this do not happen often in our evaluations. In this case, some initial measure of the link delay distributions can help.

As described in Section IV-C, we first generate 1000 pairs of probing using the priority probing scheme with uniform parameter distribution assumption. Then the link parameters are estimated using the network tomography algorithm with the 1000 pairs of observations. After that, the priority probing scheme takes these delay estimations as inputs and generates another 7000 probing pairs. This time, the accuracy achieved by the priority probing scheme becomes higher than that by the uniform scheme, as can be seen from Figure 7.

It would be interesting to see what happens when delay estimation and probing set design perform iteratively, i.e. design the rest of the probing pattern whenever there is updated delay estimation results available. Besides consideration on the computation cost, several interesting problems remain open with this regard including the convergence property of this process. We retain this as a future work.

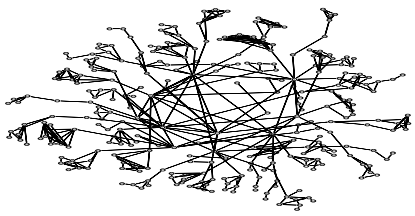

Fig. 8. Topology generated by gt-itm

4) A larger case: In the last example, we look at a larger case. We use GT-ITM [19] to generate a topology of 310 nodes. The topology consists of a core network of 10 nodes. Each node in the core network connects to 5 stub networks with each stub network consisting of an average of 6 nodes. The 10 nodes in the core network are connected as a random graph with the probability of a duplex link appearing between two nodes set to 0.6 and the nodes in the stub networks are connected with the probability set to 0.4. This reflects the high connectedness of the Internet core network [20]. 483 duplex links are generated. Links between nodes in the core network have a capacity of $50Mbps$ and others have a capacity of $10Mbps$. Random propagation delays between $1ms$ an $61ms$ are applied to the links.

We randomly pick 100 receivers from the 300 nodes in the stub networks. A node in the core network is selected as the source. The routing from the source to the receivers forms a logical tree consisting of 124 logical links with some receivers located on the branch points. The number of logical links from the receivers to the source ranges from 1 to 5 with an average of 3.09. Background traffic is applied to half of the logical links randomly selected.

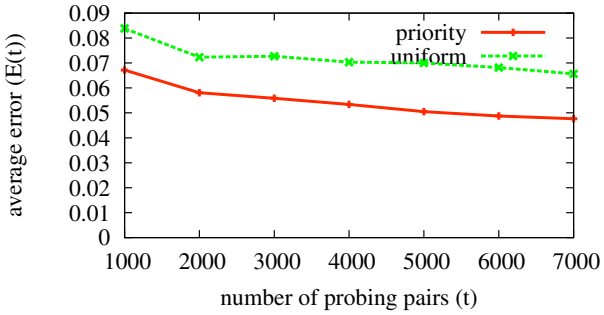

Fig. 9. Smaller estimation error from priority probing in a large network

Figure 9 presents the result of this experiment. In this experiment, priority probing uses only about 1000 probing pairs to achieve the same accuracy as existing approach using 7000 pairs. With the fixed probing rate of around $20kbps$, this is equivalent to maintain the same estimation accuracy while increase the monitoring resolution from 6 minutes per measurement to 50 seconds per measurement. Compared with previous results, the advantage of the priority probing scheme becomes more significant. And as the network size increases, this advantage is expected to become more significant.

VI. Related works

There have been extensive research efforts on network tomography. The Multicast Inference of Network Characteristics (MINC) project [1] provides techniques that apply multicast to estimate network losses [21], delay [22], delay variances [23] and the topology [24]. However, the application of multicast based network tomography is constrained by limited support over the Internet and the different delay and losses experienced by multicast traffic and unicast traffic [8], [25]. In response, unicast network tomography techniques were developed. With unicast, [11], [25] estimate network losses and [8], [9] estimate

network delay. [26] proposes a pseudo-maximum-likelihood algorithm and one of its applications is to obtain delay estimation from multicast measurements. A brief review of the network tomography literature is provided in [2].

Recently, Arya et al. [27], [28] extend network tomography techniques to infer temporal loss and delay properties in the tomography tree. Lawrence et al. [29] demonstrated using 'flexicast' to infer network performances and related optimal schemes have been discussed in [29], [14]. Chen et al. [30] discuss identifiability of link performance distribution in network tomography and propose mixture models for network delay tomography. In [31], Nguyen and Thiran proposed a Boolean algebra based algorithm to infer congested links. Also Fragouli et al. discuss the benefit of having multiple sources and destinations in [32]. Our work determines optimal probing pairs for unicast network delay tomography under the setting of Upward-Downward algorithm with a single source.

VII. Conclusion and future work

In this work, we propose *priority probing* to answer the question: with a fixed number of probing packets, how to obtain the most accurate estimator? We prove that the Fisher information matrix can be decomposed into an additive form where information accumulates as each probing is performed. Then, we formulate the optimal probing set design problem into a SDP problem. High computation complexity constrains the SDP solution to only small scale scenarios. In response, we propose a greedy algorithm that approximates the optimal solution. Evaluation results demonstrate that priority probing is very promising in increasing the estimation accuracy with a limited probing/time budget.

There are several related problems still remain open. First, if we assign two bins to each link indicating the Bernoulli distribution of packet losses, the setting becomes a network *loss* tomography problem. It remains to see how priority probing performs in that aspect and with variants as in [25]. Second, it remains to see how to adapt the algorithm in various related settings such as variable sized bins [8] and multi-source multi-receiver environments [32], [33].

References

[1] MINC: Multicast-based Inference of Network-internal Characteristics, "http://gaia.cs.umass.edu/minc/."
[2] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, "Network tomography: recent developments," *Statistical Science*, vol. 19, pp. 499–517, 2004.
[3] Y. Gu, L. Breslau, N. Duffield, and S. Sen, "Gre encapsulated multicast probing: A scalable technique for measuring one-way loss," *IEEE INFOCOM 2008*, pp. 1651–1659, April 2008.
[4] H. Pucha, Y. Zhang, Z. M. Mao, and Y. C. Hu, "Understanding network delay changes caused by routing events," *SIGMETRICS Perform. Eval. Rev.*, vol. 35, no. 1, pp. 73–84, 2007.
[5] B. Zhang, T. S. E. Ng, A. Nandi, R. Riedi, P. Druschel, and G. Wang, "Measurement based analysis, modeling, and synthesis of the internet delay space," in *IMC '06: Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*. New York, NY, USA: ACM, 2006, pp. 85–98.
[6] F. Pukelsheim, *Optimal Design of Experiments*. Society for Industrial and Applied Mathematic; illustrated edition edition (March 24, 2006).
[7] H. Wolkowicz, R. Saigal, and L. Vandenberghe, Eds., *Handbook of semidefinte programming Theory, algorithms, and applications*. Kluwer Academic Publishers.
[8] N. G. Duffield, J. Horowitz, F. L. Presti, and D. F. Towsley, "Network delay tomography from end-to-end unicast measurements," in *IWDC'01*. London, UK: Springer-Verlag, 2001, pp. 576–595.
[9] M. J. Coates and R. D. Nowak, "Network tomography for internal delay estimation," in *In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2001.
[10] O. Ronen, J. Rohlicek, and M. Ostendorf, "Parameter estimation of dependence tree models using the em algorithm," *Signal Processing Letters, IEEE*, vol. 2, no. 8, pp. 157–159, Aug 1995.
[11] M. Coates and R. Nowak, "Network loss inference using unicast end-to-end measurement," in *Proc. ITC Conf. IP Traffic, Modeling and Management*, 2000, pp. 28–1.
[12] M. J. Schervish, *Theory of Statistics*. Springer, 1995.
[13] H. Chernoff, "Locally optimal designs for estimating parameters," *The Annals of Mathematical Statistics*, vol. 24, no. 4, pp. 586–602, December 1953.
[14] B. Xi, G. Michailidis, and V. N. Nair, "Estimating network loss rates using active tomography," *Journal of the American Statistical Association*, vol. 101, pp. 1430–1448, December 2006.
[15] J. Cao, W. Cleveland, Y. Gao, K. Jeffay, F. Smith, and M. Weigle, "Stochastic models for generating synthetic http source traffic," *IEEE INFOCOM 2004*, vol. 3, pp. 1546–1557 vol.3, March 2004.
[16] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.
[17] K. Fujisawa, M. Kojima, K. Nakata, and M. Yamashita, "The sdpa family and how to choose," http://sdpa.indsys.chuo-u.ac.jp/sdpa/software.html.
[18] The Network Simulator ns-2, "http://www.isi.edu/nsnam/ns/."
[19] GT-ITM: Georgia Tech Internetwork Topology Models, "http://www.cc.gatech.edu/projects/gtitm/."
[20] S. Tauro, C. Palmer, G. Siganos, and M. Faloutsos, "A simple conceptual model for the internet topology," *Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE*, vol. 3, pp. 1667–1671 vol.3, 2001.
[21] R. Caceres, N. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal loss characteristics," *Information Theory, IEEE Transactions on*, vol. 45, no. 7, pp. 2462–2480, Nov. 1999.
[22] F. L. Presti, N. G. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal delay distributions," *IEEE/ACM Trans. Netw.*, vol. 10, no. 6, pp. 761–775, 2002.
[23] N. Duffield and F. Lo Presti, "Multicast inference of packet delay variance at interior network links," *IEEE INFOCOM 2000*, vol. 3, pp. 1351–1360 vol.3, Mar 2000.
[24] N. Duffield, J. Horowitz, F. Lo Presti, and D. Towsley, "Multicast topology inference from measured end-to-end loss," *Information Theory, IEEE Transactions on*, vol. 48, no. 1, pp. 26–45, Jan 2002.
[25] N. Duffield, F. L. Presti, V. Paxson, and D. Towsley, "Network loss tomography using striped unicast probes," *IEEE/ACM Transaction of Networking*, vol. 14, no. 4, pp. 697–710, 2006.
[26] G. Liang and B. Yu, "Pseudo likelihood estimation in network tomography," *IEEE INFOCOM 2003*, vol. 3, pp. 2101–2111 vol.3, March-3 April 2003.
[27] V. Arya, N. G. Duffield, and D. Veitch, "Multicast inference of temporal loss characteristics," *Perform. Eval.*, vol. 64, no. 9-12, pp. 1169–1180, 2007.
[28] V. Arya, N. Duffield, and D. Veitch, "Temporal delay tomography," *IEEE INFOCOM 2008*, pp. 276–280, April 2008.
[29] E. Lawrence, G. Michailidis, and V. N. Nair, "Network delay tomography using flexicast experiments," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, pp. 785–813(29), November 2006.
[30] A. Chen, J. Cao, and T. Bu, "Network tomography: Identifiability and fourier domain estimation," *IEEE INFOCOM 2007*, pp. 1875–1883, May 2007.
[31] H. X. Nguyen and P. Thiran, "The boolean solution to the congested ip link location problem: Theory and practice," in *IEEE INFOCOM 2007*, May 2007, pp. 2117–2125.
[32] C. Fragouli, A. Markopoulou, R. Srinivasan, and S. Diggavi, "Network Monitoring: it depends on your points of view," in *Proceedings ITA Workshop'07*, Jan. 2007.
[33] M. Rabbat, R. Nowak, and M. Coates, "Multiple source, multiple destination network tomography," in *Proc. of IEEE Infocom*, 2004.