One-Way Delay Estimation Using Network-Wide Measurements

Omer Gurewitz, Israel Cidon, *Senior Member, IEEE*, and Moshe Sidi, *Senior Member, IEEE*

***Abstract*—We present a novel approach for the estimation of one-way delays between network nodes without any time synchronization in the network. It is based on conducting multiple and simple one-way measurements among pairs of nodes, and estimating the one-way delays by optimizing the value of a global objective function that is affected by the overall network topology and not just by individual measurements. We examine two objective functions. The first intuitive choice is the least square error (LSE). Using a novel concept of delay-induced link probabilities, we develop a second objective function that is based on the maximum-entropy (ME) principle. Extensive numerical experiments show that both functions considerably outperform the common method of halving the round-trip delays. They also show that ME outperforms the commonly used LSE.**

***Index Terms*—Delay estimation, mathematical optimization, maximum entropy, network measurements, one-way delay.**

I. Introduction

ACCURATE measurements and adequate analysis of network characteristics are essential for robust network performance and management. Such real-data analysis plays a key role in network design and in the control of its dynamic behavior. One of the most important network performance quantities is delay as it strongly influences the configuration and performance of network protocols such as routing and flow control and network services such as voice and video over the Internet Protocol (IP). Delay measurements are common in such environments and many others. Furthermore, continuous monitoring of delay is essential in many applications in order to check compliance with critical delay constraints.

In many cases, the path from a source to a destination may differ from the path from the destination back to the source. Even when the two paths are symmetric, they may have different performance characteristics due to asymmetric loads or different quality-of-service (QoS) provisioning [1], [2]. Moreover, performance of many applications depends mostly on the delays in one direction [3]. For example, streaming applications performance depends more on the characteristics of the path from the source to the destination. A typical client server transaction depends more on the quality of the path from the server to the client. Finally, for voice and video conferencing each unidirectional path is responsible for timely delivery. Consequently, the capability to measure or estimate one-way delays is very important.

The main obstacle in measuring one-way delays is that clocks in a network are not synchronized. Taking one-way measurements is quite simple. A node can send a probe packet with a time stamp on it to its neighbor. When the neighbor receives the packet, it marks its own time stamp over it. The difference between these two time stamps is a *one-way measurement*. Clearly, this one-way measurement equals the corresponding one-way delay only if the clocks of the two nodes are synchronized. Otherwise, the one-way measurement includes the corresponding one-way delay and the clock offset (that is unknown) between the nodes.

Global Positioning Systems (GPS) provide accurate time synchronization between network nodes; unfortunately, GPS are scarce in computer networks. Moreover, an embedded GPS requires continuous reception of multiple satellites which is hard to accomplish indoors or at secured data centers. Network Time Protocol (NTP) is the current standard for synchronizing clocks, with respect to Universal Time-Coordinated (UTC), in the Internet [4], [5]. NTP measures round-trip delays and uses a halving procedure to estimate the clock offsets. A recent offset synchronization method was suggested in [6] for a Pentium-based systems as an alternative to GPS synchronization. However, this method calls for a GPS level synchronized NTP server in the (delay-wise) proximity of the measurement endpoints, a requirement that is not practical many times for remote endpoints, branches, and homes and cannot be implemented in non-PC-based systems. A novel synchronization protocol based on NTP messages that provides better accuracy by optimizing a global cost function is described in [7]. However, all these clock synchronization procedures are working accurately only when the delay is symmetric. Another approach for synchronizing clocks in sensor networks based on the availability of broadcast and low propagation delay among neighboring sensors appears in [8].

Unlike one-way delay, round-trip delay measurements are simple to conduct and they are accurate since the same clock is used while transmitting the packet and upon its return; a common approach used for estimating one-way delay is to measure round-trip delays and halve them. This requires not only that the route between source and destination be the same, but that traffic loads and QoS configurations in both directions also be the same. However, as noted above, often this is not the case.

Manuscript received March 14, 2005; revised January 6, 2006. This work was supported in part by a grant from the Cisco University Research Program Fund at Community Foundation Silicon Valley. The material in this paper was presented in part at the International Symposium on Performance Eveluation of Communication Systems (SPECTS'05), Philadelphia, PA, July 2005, and at the Information Theory and Applications Workshop, University of California, San Diego, La Jolla, CA, February 2006.

The authors are with the Electrical Engineering Department, Technion–Israel Institute of Technology, Technion City, Haifa 32000, Israel (e-mail: gurewitz@tx.technion.ac.il; cidon@ee.technion.ac.il; moshe@ee.technion.ac.il).

Communicated by D. Towsley, Guest Editor.

Digital Object Identifier 10.1109/TIT.2006.874414

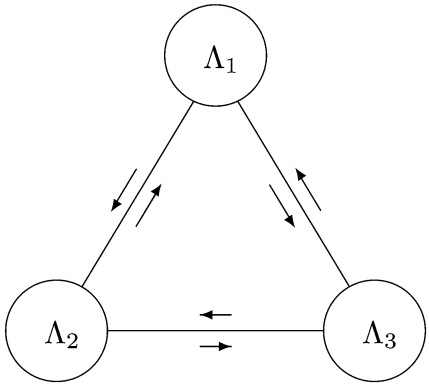

Fig. 1. Three-node network.

In this paper, we present a novel approach for the estimation of one-way delays from one-way measurements that do not require clock synchronization among the nodes of the network. The approach is based on taking one-way measurements between neighboring nodes and pose these measurements as constraints to well-defined optimization problems.

To motivate the approach and illustrate its basic ideas, let us consider the simple three-node network depicted in Fig. 1. Assume that we have the following one-way measurements (OWM) between nodes Λ_1, Λ_2, and Λ_3 (note that due to the offsets of clocks between the nodes, these measurements can have positive or negative values):

$$\begin{aligned} OWM(\Lambda_1 \rightsquigarrow \Lambda_2) &= 70 \\ OWM(\Lambda_2 \rightsquigarrow \Lambda_1) &= 30 \\ OWM(\Lambda_2 \rightsquigarrow \Lambda_3) &= 70 \\ OWM(\Lambda_3 \rightsquigarrow \Lambda_2) &= 30 \\ OWM(\Lambda_3 \rightsquigarrow \Lambda_1) &= -110 \\ OWM(\Lambda_1 \rightsquigarrow \Lambda_3) &= 210. \end{aligned}$$

Let us first concentrate on one-way measurements between node Λ_1 and node Λ_2. The sum

$$OWM(\Lambda_1 \rightsquigarrow \Lambda_2) + OWM(\Lambda_2 \rightsquigarrow \Lambda_1) = 100$$

is actually the round-trip delay between these two nodes, since it is easy to see that the offsets between the nodes are canceled in this sum. Given that the round-trip delay between the two nodes is 100, what would be the "best" estimate for the one-way delays between them? Since the meaning of "best" has not been defined yet (it will be defined later in the paper), the formal answer is unclear. However, the intuitive answer (that fits many criteria of "best") would be 50 in each direction, namely, using the halving procedure that is commonly used when no additional information is available [9], [10].

Using the same arguments for nodes Λ_1 and Λ_3 $(OWM(\Lambda_1 \rightsquigarrow \Lambda_3) + OWM(\Lambda_3 \rightsquigarrow \Lambda_1) = 100)$ would lead also to an estimate of one-way delay in each direction of 50. Similarly, for nodes Λ_2 and Λ_3 $(OWM(\Lambda_2 \rightsquigarrow \Lambda_3) + OWM(\Lambda_3 \rightsquigarrow \Lambda_2) = 100)$ would lead to an estimate of one-way delay in each direction of 50.

Based on the one-way measurements that are available, are the estimates obtained above feasible at all? Surprisingly, the answer is no. To see this, let us look at the sum

$$OWM(\Lambda_1 \rightsquigarrow \Lambda_2) + OWM(\Lambda_2 \rightsquigarrow \Lambda_3) + OWM(\Lambda_3 \rightsquigarrow \Lambda_1) = 30.$$

This sum contains no offsets and it represents the sum of the actual one-way delays from Λ_1 to Λ_2, from Λ_2 to Λ_3, and from Λ_3 to Λ_1 (clock offsets are canceled when summing the delays along a cyclic path). This implies that none of these one-way delays can be larger than 30. In particular, the one-way delays from Λ_1 to Λ_2 as well as from Λ_2 to Λ_3 and from Λ_3 to Λ_1 cannot be larger than 30 and thus cannot be 50. Furthermore, the one-way delays from Λ_2 to Λ_1, from Λ_3 to Λ_2, and from Λ_1 to Λ_3, must be at least 70 (since the round-trip delay between each pair is 100) and therefore cannot be 50 as well. We return to this example in Section VI-C.

This simple example shows that the one-way measurements impose constraints on the feasible values of the one-way delays. Our goal in this paper is to derive the "best" estimate of the one-way delays given the one-way measurements. In the sequel we show how to exploit the one-way measurements to obtain the necessary constraints on the one-way delays and derive the number of independent constraints that can be obtained. These constraints are used in the optimization problems that we explore. For these problems, we define objective functions that when optimized, provide the "best" estimate for the one-way delays.

In this paper, we investigate two different objective functions. The first objective function is very intuitive and is based on the least square error (LSE) principle. According to this principle, the solution that is sought for is the one that minimizes the square error. The second objective function is based on the *maximum entropy* (ME) principle. According to this principle, the solution that is sought for is the one that maximizes the entropy. Note that the definition of entropy requires an underlying probability space. One of the contributions of this paper is the introduction of a method to induce probabilities upon network links that are relative to the delays over these links. The objective function based on the ME principle lends itself to relatively simple computations and results in a better one-way delay estimation for most cases checked.

Both objective functions provide estimates of the fixed part (i.e., propagation) of the one-way delay. For the estimation of the variable delay one can use the same optimization methodology and common techniques that are available for the estimation of the distribution parameters. The solutions that are provided are easy to implement using standard probe packets among nodes (e.g., NTP, Internet Control Message Protocol (ICMP)). Extensive numerical experiments demonstrate that both schemes considerably outperform the traditional round-trip delay halving.

Due to the importance of measuring one-way delays, an extensive literature exists on their estimation based on other types of measurements. For example, [11] describes a related approach that uses end-to-end *multicast* packets for estimating

internal links delays. This approach requires clock synchronization at the measurement hosts, preferably via GPS. The most common approach is to halve round-trip delays to estimate one-way delays ([5], [9], [10]). The accuracy of the halving approach is highly dependent on the path symmetry in the network. However, as previously explained, the traffic on a bi-directional path is often asymmetric, [1], [2]. In [12], it was suggested to estimate internal networks delays based on end-to-end delay measurements by solving a set of linear equations, assuming that either routing is symmetric (use round-trip measurements) or clocks are synchronized (use one-way measurements). Recently, [13] combined results similar to [12] with a measurement apparatus based on attaching a centralized measurement host and the set of border routers with dedicated asynchronous transfer mode (ATM) or multiprotocol label switching (MPLS) low-delay tunnels. This yields a system that can conduct end-to-end measurements from the same host at the expense of such tunnels (maybe impossible in non service provider networks). This requirement is similar to a full synchronization at the endpoints and as a special case of the cyclic path measurement of [14]. In contrast to [13] and [14], this paper uses standard node-to-node measurement packets and does not rely on any clock synchronization at the nodes.

The paper is organized as follows. In Section II, we present the underlying model used throughout the paper including the network topology and the delay models. Section III describes the measurements that are needed to be conducted in the network that are used in the estimation procedures. Section IV describes the estimation procedures for estimating the propagation delay. Section V describes the estimation of the variable part of the delay. The quality of the proposed estimation is assessed in Section VI where numerical examples are provided.

II. The Model

In this section, we introduce the network model that is used throughout the paper. We split the description into two aspects: topology and delay.

A. Network Topology Model

Naturally, not all network elements are interested in or capable to participate in the one-way delay estimation procedure. We will focus throughout this paper on an overlay network which consists of the components that do participate in the one-way measurements. The participating components will be called nodes. Let $\mathcal{N}$ denote this set of nodes, $N = |\mathcal{N}|$ be the number of nodes, and Λ_i $i = 1,2,\ldots,N$ denote a specific node. We define a directed link between two nodes as the directed route between the two nodes that does not contain any other node in $\mathcal{N}$. The directed link connecting nodes Λ_i and Λ_j will be denoted by e_{ij} and the collection of links by $\mathcal{E}$. Note that since this is an overlay network, each link can be composed of several physical segments. We assume that all links are bi-directional, namely, if $e_{ij} \in \mathcal{E}$, then $e_{ji} \in \mathcal{E}$, with $E = |\mathcal{E}|$ denoting the number of links. We will not assume that the two links are symmetric so they can be composed of different physical links and/or can have different capacities. We will denote by G_i the set of nodes which are node Λ_i's neighbors in the underlying network.

Since the nodes in the network are not synchronized, let us denote the clock offsets of node Λ_i's clock with respect to a "Universal Time" by τ_i. By τ_{ij} we will denote the relative offset of node Λ_i's clock with respect to node Λ_j's clock, i.e., $\tau_{ij} = \tau_i - \tau_j$. Clearly, $\tau_{ji} = \tau_j - \tau_i = -\tau_{ij}$.

Special care should be given to network environments where clock drifts are present. The problem of frequency synchronization has been studied in the literature. Therefore, throughout this study we will assume that skew errors were removed from all measurements conducted between neighboring nodes, using any one of the techniques suggested in [15]–[18]. Note that all these techniques are based on one-way delay measurements which are compliant with our scheme. Previous delay estimation works such as [1] and [19] also made similar assumptions.

B. Delay Model

A common approach is to divide the delay into two basic components, deterministic and stochastic: The deterministic component can be further divided into two factors: i) Transmission delay—the time needed to transmit the packet by each physical node along the path. ii) Propagation delay—the time a bit propagates along the link. Since all measurement packets have the same format and size, and since the physical links comprising a link do not change, we assume that the deterministic part of the delay on each link is constant for all packets traveling the link. Note, though, that due to the asymmetric characteristics of the two directions of links, we do not assume that the constant parts of the delays along the two directions of a link are the same.

The other component comprising the link delay is the stochastic component which is usually associated with the queueing delay. This part may vary from packet to packet even when the packets have the same size and format.

Let us denote by x_{ij} the one-way delay on the link from node Λ_i to node Λ_j, and by c_{ij} and v_{ij} its constant and variable parts, respectively, $(x_{ij} = c_{ij} + v_{ij})$. The distribution function and the density function of the two random delays will be denoted by $F_{x_{ij}}$, $F_{v_{ij}}$ and $f_{x_{ij}}$, $f_{v_{ij}}$, respectively.

III. The Measurements

Extensive literature exists on how to conduct network measurements. Most employed schemes are based on an active measurement systems that measure round-trip and one-way delays over various Internet paths [1], [19]–[21]. While systems such as NTP use periodic trigger of measurement probes, all active systems quoted above schedule measurements at Poisson intervals. Some passive measurement systems can also measure packet delays, for example, the passive monitoring technique suggested in [22], [23]. Our scheme can easily use measurements results of both passive and active measurements systems. Moreover, since our scheme does not rely on clock synchronization and only requires time stamp exchange between the participating entities, the employed active or passive measurement architecture can be deployed over a non-GPS based infrastructure.

Since our scheme can use many measurement systems we only specify the basic measurement requirements. The

measurements should be one-way measurements conducted between each pair of neighbors. We will assume that each node is repeatedly sending probe packets (at Poisson or periodic times) to each one of its neighbors during the measurement time interval. Two time stamps will be extracted from each such probe packet k, the transmission time by the sender Λ_i to the receiver Λ_j which will be denoted by $T_{ij}^{[k]}$, and the receiving time by the receiver Λ_j which will be denoted by $R_{ij}^{[k]}$. Thus, each packet k sent from node Λ_i to node Λ_j contributes two time stamps: $T_{ij}^{[k]}$ and $R_{ij}^{[k]}$. Note that these time stamps are part of the standard NTP packet [5], so NTP messages can be used for our measurements.

We intend to estimate one-way delays by looking at the n most recent measurement packets. For the link $e_{ij} \in \mathcal{E}$ from node Λ_i to node Λ_j, let $x_{ij}^{[k]}$ be the one-way delay experienced by probe packet k. Let us also denote by $\Delta T_{ij}^{[k]}$ the time difference between the transmission of probe packet k by node Λ_i, according to node Λ_i clock, and the arriving time of the packet at node Λ_j according to its own clock i.e.,

$$\Delta T_{ij}^{[k]} = R_{ij}^{[k]} - T_{ij}^{[k]}.$$

Note that the two times are taken using different clocks that are not necessarily synchronized, hence, the computed time $\Delta T_{ij}^{[k]}$ is not the one-way delay but rather the sum of the one-way delay experienced by probe packet k while traveling from node Λ_i to node Λ_j and the time difference between the two clocks, i.e.,

$$\Delta T_{ij}^{[k]} = x_{ij}^{[k]} - \tau_i + \tau_j = x_{ij}^{[k]} - \tau_{ij}.$$

Note that $\Delta T_{ij}^{[k]}$ can be positive or negative.

An important observation is that the sum $\Delta T_{ij}^{[k_1]} + \Delta T_{ji}^{[k_2]}$ for arbitrary k_1 and k_2 represents the round-trip delay of a virtual packet that had it been sent as packet k_1 from Λ_i to Λ_j and returned from Λ_j to Λ_i as packet k_2. This follows from

$$\Delta T_{ij}^{[k_1]} + \Delta T_{ji}^{[k_2]} = x_{ij}^{k_1} - \tau_{ij} + x_{ji}^{k_2} - \tau_{ji} = x_{ij}^{k_1} + x_{ji}^{k_2}.$$

In other words, the clock offsets between the neighbors do not affect the latter expression. The same statement holds for any cyclic-path, i.e., if $\Lambda_{i_1} \rightsquigarrow \Lambda_{i_2} \rightsquigarrow \cdots \rightsquigarrow \Lambda_{i_l} \rightsquigarrow \Lambda_{i_1}$ is an arbitrary cyclic-path, then the sum

$$\Delta T_{i_1 i_2}^{[k_1]} + \Delta T_{i_2 i_3}^{[k_2]} + \cdots + \Delta T_{i_l i_1}^{[k_l]}$$

represents the cyclic-path delay of a virtual packet had it been sent along the path.

IV. Constant Delay Estimation

In Section II-B, we divided the one-way delay into two basic components, constant and variable. In this section we present the estimation procedures for the constant delay component.

A. From One-Way Measurements to Delay Constraints—Methodology

Estimation of the one-way delay from node Λ_i to node Λ_j poses the problem that the two time stamps $T_{ij}^{[k]}$ and $R_{ij}^{[k]}$ are based on two different local clocks which are not synchronized. In this subsection, we propose a method for estimating the one-way constant delay for each directed link, based on the sum of single hop one-way measurements along various cyclic paths. The main idea that is further elaborated below is that the sum of one-way measurements along a cyclic path eliminates the clock offsets from the measurements. This motivates us to identify as many independent cyclic paths as possible. Each such path yields a delay measurement that does not have any offset issues and it constrains the one-way delays along the path to equal a specific value. We need to construct as many independent cyclic paths as possible, i.e., as many independent constraints as possible. In order to extract the constant delay along the independent cyclic paths we have to separate each constraint into the two components comprising the delay: the constant and the variable delays.

In a network which is not permanently overloaded, one can expect that from time to time a packet transmitted over each link will experience no (or nearly no) queueing delay. Looking at the last n packets which traversed the link from node Λ_i to node Λ_j, the packet with the smallest entry $\min_k \Delta T_{ij}^k$ is the packet that experienced the smallest delay and, hence, it is the packet that experienced the smallest variable delay. Let us denote the quantities related to the packet with minimum delay with the superscript "[min]," i.e., $\Delta T_{ij}^{[\min]} = \min_k \Delta T_{ij}^{[k]}$.

Since we expect that on each link at least one packet among all will experience negligible variable delay, it is clear that the minimum value obtained by summing up the one-way measurements along a cyclic path is the constant delay along this path. This minimum value is the sum of the $\Delta T_{ij}^{[\min]}$'s along the cyclic path, i.e.,

$$c_{S,i_1} + c_{i_1,i_2} + \cdots + c_{i_m,S} = \Delta T_{S,i_1}^{[\min]} + \Delta T_{i_1,i_2}^{[\min]} + \cdots + \Delta T_{i_m,S}^{[\min]}.$$

Note that each directed link is measured separately to obtain $\Delta T_{ij}^{[\min]}$ and only then applying the sum over all the minimum delays.

Note that a different approach could be to send a series of probe packets along predefined cyclic paths and select the one that experienced the minimum cyclic-path delay, as suggested by [14]. However, there are two major drawbacks to such a scheme. First, we have to use nonstandard source routing messages, or a special algorithm in order to implement such a scheme. Second, our procedure is much more likely to yield the minimum constant cyclic-path delay or at least a tighter bound on it since

$$\Delta T_{S,i_1}^{[\min]} + \Delta T_{i_1,i_2}^{[\min]} + \cdots + \Delta T_{i_m,S}^{[\min]} \le \min_{[k]} \left(\Delta T_{S,i_1}^{[k]} + \Delta T_{i_1,i_2}^{[k]} + \cdots + \Delta T_{i_m,S}^{[k]} \right).$$

Note that the latter observation is valid even for the single-hop round-trip delay measurements, i.e., it is better to find a minimum delay in each direction separately than to look for the packet exchange that experienced the minimum round-trip delay (as performed by NTP), or formally

$$\Delta T_{ij}^{[\min]} + \Delta T_{ji}^{[\min]} \leq \min_{[k]} \left(\Delta T_{ij}^{[k]} + \Delta T_{ji}^{[k]} \right).$$

For each link $e_{ij} \in \mathcal{E}$, let $\hat{c}_{ij}$ be the estimate of c_{ij}. We formulate the estimation problem as a constrained optimization problem. The variables are $\vec{c} = \{c_{ij}\}$ (similarly, the estimates are $\hat{\vec{c}} = \{\hat{c}_{ij}\}$) and the constraints are the cyclic-path delays *computed* from the one-way measurements and the nonnegativity of the variables $\vec{c}$. To formally define these constraints, assume that we identify $\mathcal{L}$ cyclic paths for which we compute the delays. Let $a_{l,\{ij\}} = 1$ if link e_{ij} appears along the lth cyclic path and $a_{l,\{ij\}} = 0$ otherwise. Let α_l be the computation of the minimum delay obtained for the lth cyclic path. The delay constraints are given by

$$\boldsymbol{A} \cdot \vec{c} = \vec{\alpha} \tag{1}$$

where $\boldsymbol{A}$ is a $\mathcal{L}x\mathcal{E}$ matrix whose elements are $\{a_{l,\{ij\}}\}$ and $\vec{\alpha}$ is a vector whose elements are $\{\alpha_l\}$. Note that each constraint is essentially a linear equation that contains part of or all of the unknowns $\vec{c}$.

B. Constraints Characteristics

An interesting and important question is how many independent cyclic paths exist that yield independent constraints (equations). If the number of independent constraints $(\mathcal{L})$ were the same as the number of unknown one-way delays (E), then the one-way delays could have been computed exactly from the set of equations $\boldsymbol{A} \cdot \vec{c} = \vec{\alpha}$. However, in Theorem 1 we show that, in an N-node connected network, the maximal number of independent delay constraints (equations) that can be obtained from the computations of the cyclic-path delays (based on the one-way measurements) is smaller than the number of links by $(N-1)$.

Theorem 1: The maximal number of independent delay constraints (equations) obtained from the cyclic-path delay computations in an N-node connected network is $E - (N-1)$.

Theorem 1 is based on the result known as *Euler's formula* for graphs [24]. According to *Euler's formula* for graphs: if $\mathcal{G}$ is a connected undirected graph, then *no. of independent circuits* = *no. of edges* − *no. of vertices* $+1$. This result was extended for the number of independent circuits in strongly connected *directed* graphs [25]. The dimension of the cycle basis or the maximal number of linearly independent circuits is known in graph theory as the *cyclomatic number*.

Theorem 1 implies directly that in an N-node *connected* network $(\mathcal{N}, \mathcal{E})$, by using a correct choice of $(N-1)$ links and $E-(N-1)$ cyclic paths whose delays are computed from the one-way measurements, we can represent the one-way delays of all the E links.

Next, we construct the cycle basis, i.e., select the $E-(N-1)$ independent cyclic paths (see the algorithm for the undirected in graph [26]). First, we choose a spanning tree T. Using each link $e_{ij} \notin T$ we construct a cycle by connecting its two end nodes Λ_i and Λ_j via the unique path in T. This cycle completion is possible since all links $e_{ij} \in T$ are bi-directional. Note that this algorithm allows us to construct a first set of only $E-2(N-1)$ cycles. In addition, we take a second set of $N-1$ round trips along each edge of the spanning tree to conclude with a total of $E-(N-1)$ independent cyclic paths. The set is independent since any cycle of the first set contains a link which is exclusive to that cycle. Each cycle from the second set does not contain any of the first set of exclusive links, and hence is independent from the first set. In addition, all cycles of the second set are mutually exclusive (each cycle comprises two exclusive links) hence they are also independent.

Let us denote each of the chosen independent cyclic paths by θ_k $1 \leq k \leq E-(N-1)$ and the set of all the paths by $\Theta = \{\theta_k | 1 \leq k \leq E-(N-1)\}$. The variables to be determined in the $E-(N-1)$ independent constraints are the E one-way delays $\{c_{ij} | \forall\, e_{ij} \in \mathcal{E}\}$. An additional constraint is the nonnegativity of the variables $\{c_{ij} \geq 0 | \forall\, e_{ij} \in \mathcal{E}\}$.

Let set Ω define all the values of $\vec{c}$ with $c_{ij} > 0$ that comply with the constraints in (1). Clearly, this set is convex. Note that if any further information is available about the c_{ij}, it can be incorporated as additional constraints in the definition of Ω.

Now that we have established the constraints upon the constant delay estimation process and the feasible region upon these constraints, we have to determine which solution out of all feasible ones that comply with the constraints should be picked as the one-way constant delays.

C. The Optimization Problem

Since the one-way measurements do not contain enough information for obtaining the real delay values, no matter how many measurements are taken, there is no scheme that can uniquely determine the delays. Our approach is to pose the problem as a constrained optimization problem. The constraints were defined in Section IV-A. In order to complete the optimization problem setup we have to specify the objective function. The objective function will also help in assessing the quality of the one-way constant delay estimates.

In the sequel, we investigate two objective functions. The first which is very intuitive is based on the LSE principle. The second is based on the ME principle.

1) Least Square Error (LSE): Let us re-examine the domain Ω which defines all the values of $\vec{c}$ with $c_{ij} > 0$ that comply with the equality constraints (see (1)). Obviously, any point in the established domain can be the "true" constant one-way delays; hence, it seems self-evident to pick the constant one-way delay $\vec{\tilde{c}}$ which lies in Ω and yields the LSE, or to solve the following:

$$\min \left\{ \int_{\Omega} |\vec{c} - \vec{\tilde{c}}|^2 d\vec{c} \right\} \tag{2}$$

under the constraints

$$\Omega = \{\vec{c} \,|c_{ij} > 0; \boldsymbol{A} \cdot \vec{c} = \vec{\alpha}\}.$$

Note that $\vec{c} - \vec{\tilde{c}}$ consist of all the links in $\mathcal{E}$. The next theorem states that instead of minimizing the function $\sum_{\text{all links}} (c_{ij} - \hat{c}_{ij})^2$ over all links, $\forall\, e_{ij} \in \mathcal{E}$, we can

minimize a similar function only upon the specifically chosen $(N-1)$ links of an N-node connected network. To state the theorem we denote by $w_1, w_2, \ldots, w_{N-1}$ the variables that correspond to the one-way delays of the $N-1$ chosen links. Similarly, let $\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{N-1}$ be their respective estimates.

Theorem 2: The target function

$$\sum_{\forall\, e_{ij}\in\mathcal{E}} (c_{ij}-\hat{c}_{ij})^2$$

can be presented as a function of the $(N-1)$ one-way delays of the chosen links as follows:

$$\sum_{k=1}^{N-1}\sum_{l=1}^{N-1} D_{kl}\cdot(w_k-\hat{w}_k)(w_l-\hat{w}_l) \tag{3}$$

where D_{kl} are constants.

Proof: According to Theorem 1, in the N-node connected network $(\mathcal{N},\mathcal{E})$, there are E links (their delays are our variables), and only $E-(N-1)$ independent cyclic paths (which are the set of independent equations). Clearly, each one-way constant delay c_{ij} can be presented as a linear combination of the chosen $(N-1)$ independent one-way constant delays

$$c_{ij}=\gamma_{ij}+\sum_{k=1}^{N-1} a_{ij}^{(k)} w_k$$

where $a_{ij}^{(k)}$ are integers and γ_{ij} are constants. We take our estimates $\hat{c}_{ij}$ to have the same form, i.e.,

$$\hat{c}_{ij}=\gamma_{ij}+\sum_{k=1}^{N-1} a_{ij}^{(k)} \hat{w}_k.$$

Now let us concentrate on one element in the sum of the target function, namely, let us develop $(c_{ij}-\hat{c}_{ij})^2$

$$\begin{aligned}(c_{ij}-\hat{c}_{ij})^2 &= \left(\gamma_{ij}+\sum_{k=1}^{N-1} a_{ij}^{(k)} w_k-\gamma_{ij}-\sum_{k=1}^{N-1} a_{ij}^{(k)}\hat{w}_k\right)^2\\ &= \left(\sum_{k=1}^{N-1} a_{ij}^{(k)}\left(w_k-\hat{w}_k\right)\right)^2\\ &= \sum_{k=1}^{N-1}\sum_{l=1}^{N-1} a_{ij}^{(k)} a_{ij}^{(l)}(w_k-\hat{w}_k)(w_l-\hat{w}_l).\end{aligned}$$

Using the last result and summing over all links yields (3). From this derivation it is also clear that $D_{kl}=D_{lk}$ for all lk.

An example of Theorem 2 is the target function of the fully connected network for which $D_{k,k}=2(N-1)\;\forall\, k$ and $D_{kl}=2\;\forall\, k\neq l$.

In order to complete the analysis, we still have to find

$$\min\left\{\int_\Omega |\vec{c}-\hat{\vec{c}}|^2 d\vec{c}\right\}$$

which according to Theorem 2 is of the form

$$\min\left\{\int_\Omega \sum_{k=1}^{N-1}\sum_{l=1}^{N-1} D_{kl}\cdot(w_k-\hat{w}_k)(w_l-\hat{w}_l)d\vec{w}\right\}.$$

Let us partially differentiate the above with respect to each variable $\hat{w}_p$, equate it to zero, and use the fact that $D_{kl}=D_{lk}$ to obtain

$$\begin{aligned}&\int_\Omega\left(\sum_{l=1}^{N-1} D_{p,l}\cdot(w_l-\hat{w}_l)\right)d\vec{w}\\ &\quad=\sum_{l=1}^{N-1} D_{p,l}\left(\int_\Omega (w_l-\hat{w}_l)d\vec{w}\right)\\ &\quad=0.\end{aligned}$$

A possible solution is

$$\int_\Omega (w_l-\hat{w}_l)d\vec{w}=0,\qquad 1\le l\le N-1$$

or

$$\hat{w}_l=\int_\Omega \frac{1}{\int_\Omega d\vec{w}} w_l d\vec{w},\qquad 1\le l\le N-1. \tag{4}$$

Note that this solution is unique due to the convexity of Ω. It is interesting to see that the best estimate $\hat{w}_l$ is some kind of averaging of w_l over Ω which further supports our intuition that the LSE is a good choice for the objective function.

2) Maximum Entropy (ME): In this subsection we suggest a different objective function for estimating the one-way constant delay that is based on the ME principle. To that end, we first need to develop the underlying probabilistic foundation for proper use of the entropy notion. We achieve this by using a novel concept of delay-induced link probabilities as described in the sequel.

D. Probabilistic Interpretation

Let us examine the following conceptual experiment: Suppose that there is a special packet which is hopping throughout the network. This packet does not follow a predefined route but each time it finishes traversing a link it randomly picks a network-wide link with equal probability out of all the network directional links and transmit the packet on this link. Note that the experiment is conceptual, hence, when the packet arrives to a node over one of its incoming links it does not necessarily leave on one of the outgoing links as described in Fig. 2(a). Also note that since we randomly pick each link with equal probability among all links in the network, in the long run, the packet will traverse each link the same number of times. However, the time spent by the packet each time it travels over a link depends on its delay. Our goal is to estimate the probabilities of finding the packet on each directional link.

Formally, we are trying to estimate the components of a vector $\vec{p}$ whose components are the probabilities of finding the packet on each link $e_{ij}\in\mathcal{E}$. Each component p_{ij} can be assigned only positive values $(p_{ij}\ge 0)$. The sum of all probabilities should equal one $(\sum_{e_{ij}\in\mathcal{E}} p_{ij}=1)$. If we have no additional knowledge regarding the probabilities (e.g., no link delay information is known), the most reasonable probability assignment would be $p_{ij}=\frac{1}{|E|}\;\forall\, e_{ij}\in\mathcal{E}$, i.e., the probability of finding the packet on each link is the same.

Now suppose that we do know the probability of finding the packet along one of the links comprising several predefined

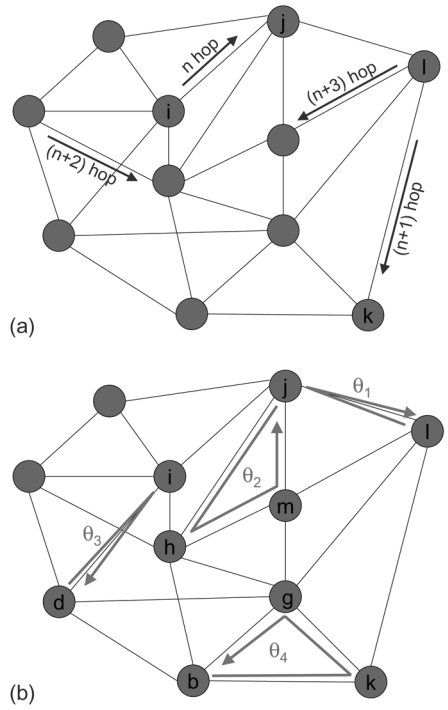

Fig. 2. The conceptual experiment. (a) A packet is traveling the network. This packet does not follow a predefined route. Instead, each time it finishes traversing a link we randomly pick a link with equal probability out of all the network directional links and transmit the packet on this link. (b) We know the probability of finding the packet along one of the links comprising a predefined path for several cyclic paths. $P_{\theta_1} = p_{lj} + p_{jl} = \beta_1$; $P_{\theta_2} = p_{jh} + p_{hm} + p_{mj} = \beta_2$; $P_{\theta_3} = p_{di} + p_{id} = \beta_3$; $P_{\theta_4} = p_{bk} + p_{kg} + p_{gb} = \beta_4$.

cyclic paths. For example, assume that in Fig. 2(b) the probability of finding the packet along one of the links comprising the path denoted by θ_1 is β_1 and the probability of finding the packet along one of the links comprising paths $\theta_2, \theta_3, \theta_4$ is β_2, β_3 and β_4, respectively, i.e., $\text{Prob}\{$ packet is on one of the links comprising $\theta_1\} = p_{lj}+p_{jl} = \beta_1; \text{Prob}\{$ packet is on one of the links comprising $\theta_2\} = p_{jh} + p_{hm} + p_{mj} = \beta_2; \text{Prob}\{$ packet is on one of the links comprising $\theta_3\} = p_{di}+p_{id} = \beta_3; \text{Prob}\{$ packet is on one of the links comprising $\theta_4\} = p_{bk} + p_{kg} + p_{gb} = \beta_4$. What should be the probabilities p_{ij} of finding the packet on each directional link e_{ij} given the additional knowledge?

The modified probability assignment should satisfy the given data

$$p_{ij} \geq 0, \qquad \sum_{e_{ij}\in\mathcal{E}} p_{ij} = 1, \qquad \sum_{e_{ij}\in R_k} p_{ij} = \beta_k$$

where R_k denotes a cyclic path in the set of cyclic paths where we know the probability of finding the packet along each one of them. Re-examining the conceptual experiment described above, it seems that the ME principle should be a good choice for estimating the probability of finding the virtual packet along each link. Evidently, *entropy* is the most natural function to measure the lack of knowledge about a certain system which makes it the most suitable function to find the desired probabilities in the proposed conceptual experiment [27]–[30]. Quoting E. T. Jaynes: "Information theory provides a constructive criterion for setting up probability distribution on the basis of partial knowledge, and leads to a type of statistical inference which is called maximum-entropy. It is the least biased estimate possible on the given information." Consequently, the determination of the probabilities p_{ij} follows the solution of the maximal entropy $-\sum_{e_{ij}\in\mathcal{E}} p_{ij}\log p_{ij}$ under the above constraints.

Formally, we have to find the probability assignment $p_{ij}\ \forall\ e_{ij} \in \mathcal{E}$ which satisfies the conditions that p_{ij} is positive and satisfies the cyclic path probability constraints

$$p_{ij} \geq 0, \qquad \sum_{e_{ij}\in\mathcal{E}} p_{ij} = 1, \qquad \sum_{e_{ij}\in R_k} p_{ij} = \beta_k$$

and maximizes the information theory entropy

$$S_I = -\sum_{e_{ij}\in\mathcal{E}} p_{ij}\log p_{ij}. \tag{5}$$

E. Reduction From Conceptual Experiment to One-Way Delay Estimation

We now explain the relation between the conceptual experiment suggested in Section IV-D and the problem of estimation of the constant one-way delays. To that end, let us examine the probability of finding the packet on each link (in the experiment). Clearly, if the only delay experienced by the packet is the constant delay, the probability of finding the packet along a specific link is proportional to the link delay. For example, assume that the delay along link a is twice the delay along link b. In the long run, the number of times the packet will traverse both links is the same; this means that the packet will spend twice as much time on link a than on link b. Consequently, the probability of finding the packet on link a is twice the probability of finding the packet on link b. Applying the same for all links and that all probabilities sum up to one, we get

$$p_{ij} = \frac{c_{ij}}{\sum_{e_{ij}\in\mathcal{E}} c_{ij}}$$

where c_{ij} is the constant delay along link e_{ij}.

We return to our original goal of determining the constant delays. It is easy to see that

$$\sum_{e_{ij}\in\mathcal{E}} c_{ij} = \sum_{e_{ij}\in\mathcal{E}} \Delta T_{ij}^{[\min]}.$$

The reason is that when summing $\Delta T_{ij}^{[\min]} + \Delta T_{ji}^{[\min]}$ over the two directions of any two neighboring nodes i and j, the clock offset between the nodes is added once and subtracted once. Let $C = \sum_{e_{ij}\in\mathcal{E}} c_{ij}$.

The probability sums along the selected cyclic paths can be also easily determined based on the *same* probe packets that experienced the minimum delay over each directed link (Section IV-A)

$$\beta_k = \sum_{e_{ij}\in\theta_k} p_{ij} = \sum_{e_{ij}\in\theta_k} \frac{c_{ij}}{\sum_{e_{ij}\in\mathcal{E}} c_{ij}}$$

$$= \sum_{e_{ij}\in\theta_k} \frac{\Delta T_{ij}^{[\min]}}{\sum_{e_{ij}\in\mathcal{E}} \Delta T_{ij}^{[\min]}} = \frac{1}{C} \sum_{e_{ij}\in\theta_k} \Delta T_{ij}^{[\min]}, \qquad k \in \Theta.$$

In the constant one-way delay estimation problem we also require that the delay variable c_{ij} takes only positive values and complies with the cyclic path-delay measurements. As can be seen, the problem of estimating the one-way constant delays is the same as estimating the probability distribution of finding the packet in the suggested experiment traveling along each link. Hence the principle of ME can be exploited.

The reduction from conceptual experiment to one-way delay estimation can be summarized as

$$c_{ij} \geq 0 \Leftrightarrow p_{ij} \geq 0$$
$$p_{ij} = \frac{c_{ij}}{\sum_{e_{ij}\in\mathcal{E}} c_{ij}} \Leftrightarrow \sum_{e_{ij}\in\mathcal{E}} p_{ij} = 1$$
$$\sum_{e_{ij}\in\theta_k} c_{ij} = \alpha_k \Leftrightarrow \sum_{e_{ij}\in R_k} p_{ij} = \beta_k = \frac{\alpha_k}{\sum_{e_{ij}\in\mathcal{E}} c_{ij}}.$$

Consequently, the optimization problem at hand is as follows.

Maximize the information-theoretical entropy

$$S = -\sum_{e_{ij}\in\mathcal{E}} p_{ij} \log p_{ij} \tag{6}$$

subject to the constraints

$$p_{ij} \geq 0, \qquad \sum_{e_{ij}\in\mathcal{E}} p_{ij} = 1 \tag{7}$$

and

$$\sum_{e_{ij}\in\theta_k} p_{ij} = \frac{1}{C} \sum_{e_{ij}\in\theta_k} \Delta T_{ij}^{[\min]} = \frac{1}{C}\alpha_k = \beta_k, \qquad \forall\, \theta_k \in \Theta. \tag{8}$$

To maximize (6) subject to the constraints (7) and (8) we employ the method of Lagrange multipliers. The relevant steps are briefly outlined in the following.

The Lagrangian will take the form

$$\mathcal{L}(p_{ij}, \lambda_k) = -\sum_{e_{ij}\in\mathcal{E}} p_{ij} \log p_{ij} - \sum_{k=1}^{E-(N-1)} \lambda_k \left(\sum_{e_{ij}\in\theta_k} p_{ij} - \alpha_k \right) - (\lambda_0 - 1) \left(\sum_{e_{ij}\in\mathcal{E}} p_{ij} - 1 \right). \tag{9}$$

Taking the derivative

$$\frac{\partial \mathcal{L}(p,\lambda)}{\partial p_{ij}} = -\log p_{ij} - \lambda_0 - \sum_{k=1}^{E-(N-1)} \lambda_k \delta(e_{ij} \in \theta_k) = 0 \tag{10}$$

where

$$\delta(e_{ij} \in \theta_k) = \begin{cases} 1, & e_{ij} \in \theta_k \\ 0, & \text{otherwise} \end{cases}$$

and the probabilities are

$$p_{ij} = e^{-\lambda_0 - \lambda_1 \cdot \delta(e_{ij}\in\theta_1) - \cdots - \lambda_{E-(N-1)} \cdot \delta(e_{ij}\in\theta_{E-(N-1)})}. \tag{11}$$

The Lagrange multiplier λ_0 is determined by substituting (11) into (7)

$$e^{-\lambda_0} \cdot \sum_{e_{ij}\in\mathcal{E}} e^{-\lambda_1 \cdot \delta(e_{ij}\in\theta_1) - \cdots - \lambda_{E-(N-1)} \cdot \delta(e_{ij}\in\theta_{E-(N-1)})} = 1. \tag{12}$$

If we now define a partition function as

$$Z(\lambda_1, \ldots, \lambda_{E-(N-1)} \equiv \sum_{e_{ij}\in\mathcal{E}} e^{-\lambda_1 \cdot \delta(e_{ij}\in\theta_1) - \cdots - \lambda_{E-(N-1)} \cdot \delta(e_{ij}\in\theta_{E-(N-1)})} \tag{13}$$

then (12) reduces to

$$\lambda_0 = logZ(\lambda_1, \ldots, \lambda_{E-(N-1)}). \tag{14}$$

The rest of the Lagrange multipliers λ_i $i = 1, \ldots, (E - (N-1))$ are determined by substituting (11) and (14) in (8) producing

$$\frac{\partial}{\partial \lambda_k} \log Z = \alpha_k; \tag{15}$$

a set of $E-(N-1)$ equations for $E-(N-1)$ unknowns. In order to solve the set of equations and determine λ_i $i = 1, 2, \ldots, E-(N-1)$, one can use one of many iterative methods [31], [32].

Note that in wire-line networks, where the physical routes between neighboring nodes do not change frequently, the deterministic part of the delay does not change often and the algorithm can be run only sporadically. Therefore, in such a case, the convergence rate is of secondary importance. Also note that in both wire-line and wireless networks if delays are not drastically changed over short times, subsequent runs of the algorithms converge much faster and the next run starts using the delay values of the previous run.

After solving the set of Lagrange multipliers, the probabilities are

$$d_{ij} = \frac{1}{Z} e^{-\lambda_1 \cdot \delta(e_{ij}\in\theta_1) - \cdots - \lambda_{E-(N-1)} \cdot \delta(e_{ij}\in\theta_{E-(N-1)})} \tag{16}$$

and therefore the one-way constant delays are given by

$$c_{ij} = C \cdot p_{ij} \quad \forall\, e_{ij} \in \mathcal{E} \tag{17}$$

V. Variable Delay Estimation

The estimation of the one-way variable delay on link e_{ij} $\forall\, e_{ij} \in \mathcal{E}$ should be based on measurements taken by the probe packets exchanged between the neighboring nodes Λ_i and Λ_j. Each measurement $\Delta T_{ij}^{[k]}$ is made up of the delay experienced by probe packet k and the clock offset between the two nodes, $\Delta T_{ij}^{[k]} = x_{ij}^{[k]} - \tau_{ij}$. Recall that the one-way delay can be separated into constant and variable delays, i.e., $\Delta T_{ij}^{[k]} = c_{ij}^{[k]} + v_{ij}^{[k]} - \tau_{ij}$. Since we assume that the clock offset is constant (clock drifts are removed), we can unite the two constant parts (clock offset and constant delay) into one that will be denoted by C_{ij} where $C_{ij} = c_{ij}^{[k]} - \tau_{ij}$. The random variable, which represents the variable one-way delay, has hence the distribution of the set of measurements shifted by the constant C_{ij}, and its probability density function is

$f_{v_{ij}}(t) = f_{\Delta T_{ij}}(t - C_{ij})$, where we have denoted by $f_{\Delta T_{ij}}$ the probability density function of the measurements.

Several schemes can be used for estimating which distribution best describes the filtered measurement [33], [34]; other works relate to the problem of modeling variable delays [19]. In this work, for completeness of the model in the numerical results section, we assume that the variable delay distribution type is a Gamma distribution as suggested by [1], [35], i.e.,

$$f_{x_{ij}}(t) = \frac{(t - C_{ij})^{\alpha-1} e^{-\frac{(t-C_{ij})}{\beta}}}{\beta^{\alpha}\Gamma(\alpha)}$$

and we use the maximum-likelihood method in order to estimate the parameters α, β, C_{ij} [36] from our measurements ΔT_{ij}.

VI. Numerical Results

In order to illustrate the quality of the delay estimation attained using the suggested algorithm, we apply it to several sample networks with different size topologies and a variety of link delays characteristics.

We divide the validation section into three different parts. We start by presenting the network topology setup. In the second part, we present the results of estimating the propagation link delay. In the third part, we evaluate the performance of the combined estimation of both the propagation delay (Section IV-C1 and C2) as well as the queueing delay (Section V).

A. Simulation Models

In order to evaluate the one-way delay estimation schemes presented in Sections IV and V, we first need to construct the network topology setup based on the existing extensive literature of modeling network topologies.

As explained in Section II-A, we focused throughout the paper on an overlay network that consists of the components that participate in the round-trip delay measurements. The network topology we choose to implement is based on the random graph model of [37]. Other parameters besides connectivity such as the delay parameters, clock offsets, etc., were chosen based on the literature.

We model all delays as described by a shifted exponential distribution, which is a special case of the gamma distribution. The suggested schemes were evaluated over a wide variety of parameters (shift as well as exponential parameters), for each run we will specify the relevant parameters.

As can be seen in the analytic part of the paper, our one-way delay estimations are totally unaffected by clocks' offset values. However, for completeness we randomly chose the offsets to vary with a uniform distribution between -10 and 10 ($\sim U[-10, 10]$) which is based on [38].

B. Propagation Link Delay

We return to the three-node example presented in Section I. The measured delays were: $\Delta_{1,2} = 70$, $\Delta_{2,1} = 30$, $\Delta_{2,3} = 70$, $\Delta_{3,2} = 30$, and $\Delta_{3,1} = -110$, $\Delta_{1,3} = 210$. Recall that based only on single-link round-trip delays, and on halving the measured round-trip delays on each link, results in nonfeasible delay estimates. On the other hand, the estimated delays according to both the LSE and the ME are: $\hat{c}_{1,2} = \hat{c}_{2,3} = \hat{c}_{3,1} = 10$ and $\hat{c}_{2,1} = \hat{c}_{3,2} = \hat{c}_{1,3} = 90$, which is also intuitive since it does not favor any link over the other links. Note, however, that this is not the only feasible solution. According to the given measurements $\hat{c}_{1,2} = 15$, $\hat{c}_{2,3} = 5$, $\hat{c}_{3,1} = 10$ and $\hat{c}_{2,1} = 85$, $\hat{c}_{3,2} = 95$, $\hat{c}_{1,3} = 90$ is also a feasible solution.

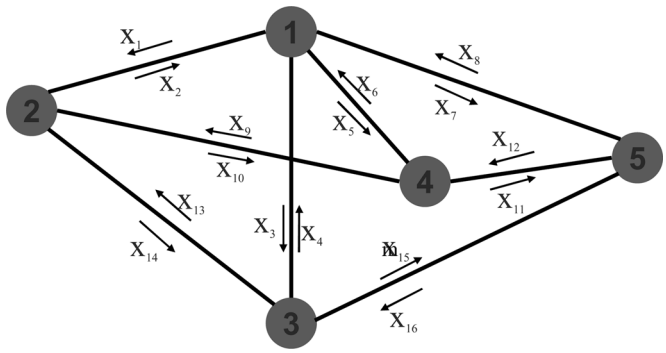

Fig. 3. Five-node 16-link network.

Next we examine the estimation of the propagation delay, based on the LSE and ME principles suggested in sections Section IV-C1 and C2, respectively. We apply our scheme to the five-node network shown in Fig. 3. Since, in this part, we are interested in evaluating only the propagation delay estimation, we assume that at least one packet experiences no queueing delay on each link, i.e., $\Delta T_{ij}^{\min}$ on each link is the propagation delay plus the relative clock offset between the two nodes in the two ends of the link. The propagation delay on each link is chosen for each direction separately based on a normal distribution where the mean and variance are uniformly selected between 10 to 40 and between 5 and 15, respectively ($\sim U[10, 40]$ and $\sim U[5, 15]$). We run our scheme 100 times on the network where the mean and variance were randomly selected once, prior to the first run.

In order to evaluate our results, we compare them with a reference scheme. According to this scheme, denoted by "H" for halving, we take into account only round-trip delays hence the propagation delay of each link is computed simply by halving the minimum round-trip delay attained on the link by one or two different packets, i.e.,

$$c_{ij} = \frac{\Delta T_{ij}^{\min} + \Delta T_{ji}^{\min}}{2}.$$

The halving scheme is based on NTP [5] and on [9], [10] that halves the minimum round-trip delay attained on a link. Note that a scheme that synchronizes the clocks in the network using NTP and then measures the delays will be less accurate than the halving scheme since NTP is an hierarchical scheme meant for synchronizing clocks with respect to a specific clock, the reference time node, hence, the farther the nodes are from the reference node the less accurate their clocks are with respect to it and therefore with respect to each other.

Fig. 4 shows the results of the 100 runs over a variety of selected links. The y axis in each graph presents the fraction of runs where the propagation delay difference between the estimated value and the real propagation delay is not greater than the value described in the x axis.

Fig. 4 demonstrates significant improvement in terms of the delay estimation of both the "LSE" and "ME" schemes over

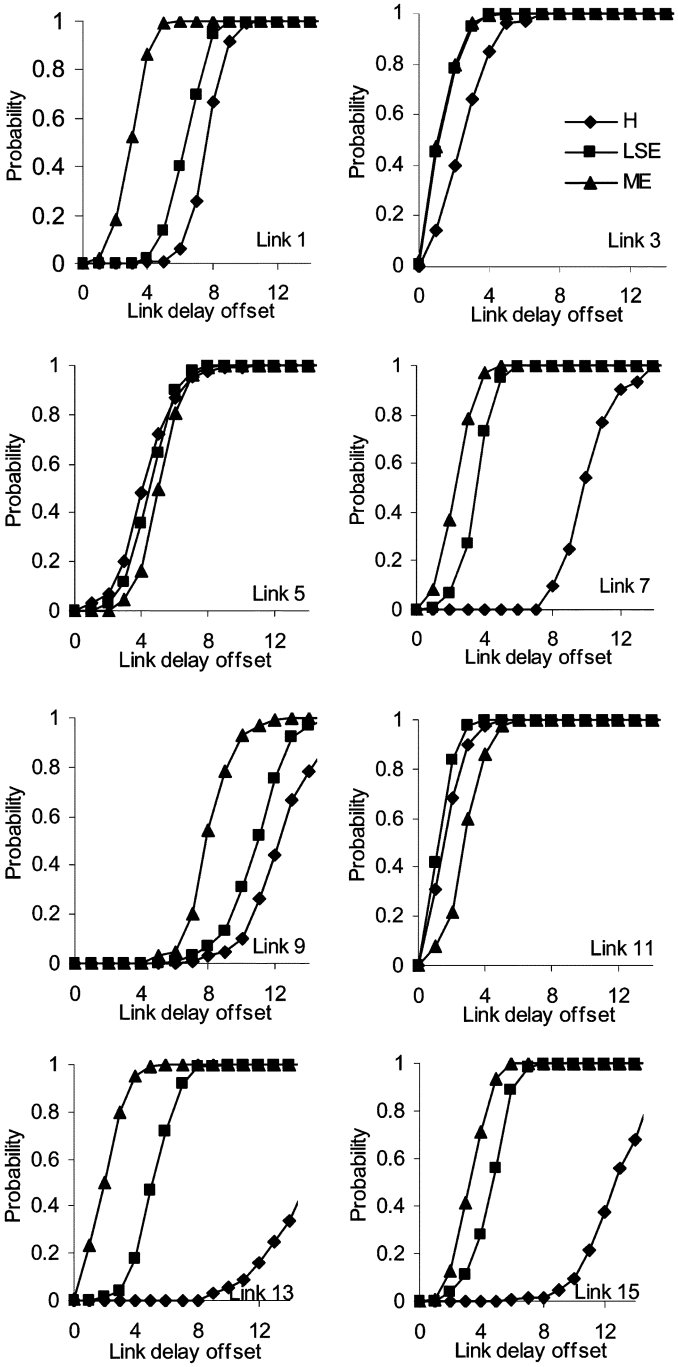

Fig. 4. The fraction of runs that the difference between the estimated link delay and the real link delay is not greater than t, for selected links in the fivr-node 16-links network.

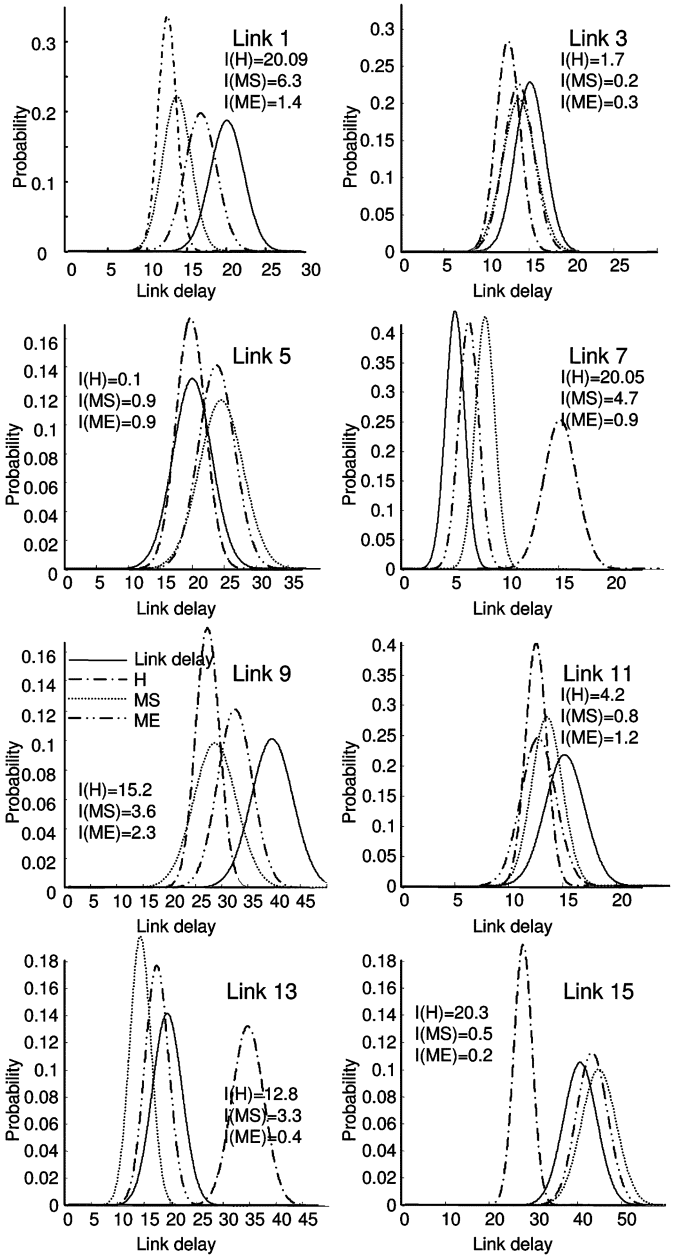

Fig. 5. Estimation of the normal distribution propagation delay in the five-node 16-link network. H—halving, MS—least square error, ME—maximum entropy.

the halving scheme. For example, in link 7, the estimated link propagation delay never exceed 5.1 and 3.7 time units in 100 runs for the LSE and ME, respectively, whereas for the halving scheme the maximum delay error is 13.8. Symmetric or nearly symmetric delay mean and variance such as in links 5,6 and 11,12 make the halving scheme the natural choice for estimating the one-way link delay. It is important to note that the delay estimation on such links by LSE and ME does not fall much behind the halving scheme. On the other hand, on asymmetric links, the LSE and ME schemes significantly outperforms the halving scheme.

Our schemes can be extended to handle nonconstant delays. For example, when the minimum delay attained on each link $\left(\Delta T_{ij}^{[\min]}\right)$ behaves according to a known distribution type, we may want to estimate some of the relevant parameters. In such cases, both the Maximum Entropy as well as the Least Square Error can be used by computing each link propagation delay over time and use common parameter estimation techniques [39].

We ran the same simulation as before, 100 times over the fivr-node 16-link network where the delay at each link has normal distribution. We estimated the distribution based on the mean and variance. We compared the results of the LSE and ME with the halving technique.

Fig. 5 shows that the estimates obtained by the LSE and ME are much better than those obtained by the halving scheme. We

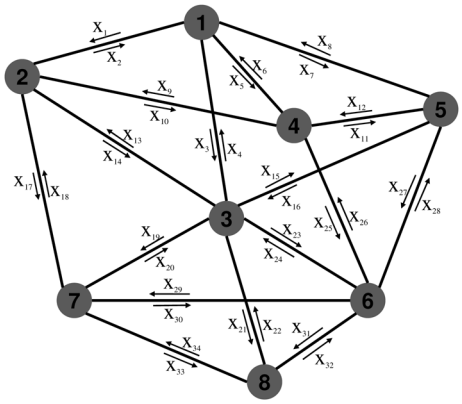

Fig. 6. Eight-node 17 bidirectional link network.

added to each graph in this figure the I-divergence distance [33], which measures the difference between the estimated and true distribution. It is interesting to note the results of link 5, which was forced to be symmetric, i.e., to have the same mean and variance as link 6. The halving scheme is naturally the best for symmetric links. However, the difference between the two schemes is not very big. For link 5, the I-divergence distance is 0.1 and 0.9 from the link delay to the estimation based on halving and both LSE and ME, respectively. On the other hand, in links that are not symmetric, the improvement of LSE and ME over the halving scheme is very significant. For instance, for link 1 the I-divergence distance is $20.9, 6.3$, and 1.4 from the link delay halving, LSE and ME, respectively. Again, ME outperforms the LSE scheme.

As previously explained, the halving scheme performs well on symmetric links. Next we explore the correlation between the number of asymmetric links in the network and the performance of the suggested schemes. We ran the three schemes on the eight-node 34–link network depicted in Fig. 6. We varied the number of asymmetric links, starting with no asymmetric link (all 17 bidirectional links symmetric), continuing with one asymmetric link and 16 symmetric links and consequently continuing by changing each time two more symmetric links to asymmetric links until all 17 bidirectional links were asymmetric.

Based on [1], [35], we modeled the total delay by a shifted Erlang distribution (which is a special case of the gamma distribution in the case that one of the parameters takes only integer values). The shift parameter (propagation delay) of each link is chosen based on uniform distribution ($\sim U[0, 10]$). The Erlang parameters α and θ were randomly selected between 1 to 5 and between 0.1 to 1, respectively. On each link, eight probe packets are used as suggested by NTP and ΔT_{ij} were measured based on these packets. Each network setup was ran 20 times where each time the symmetric links and all parameters were randomly selected. Note that on the symmetric links, only the propagation delay was selected to be the same and the queueing delays were randomly selected, separately for each link. Also note that on the asymmetric links, the propagation delay was chosen such that there was at least two time units difference between the two link directions (this insures that asymmetric links are indeed asymmetric).

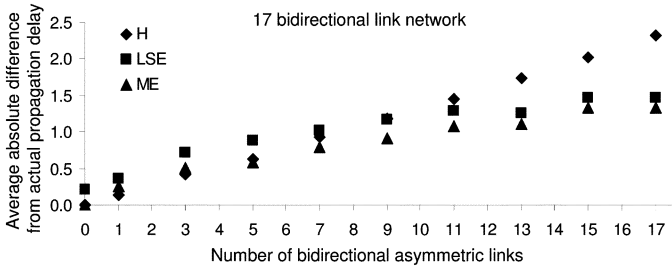

Fig. 7. No queueing delay.

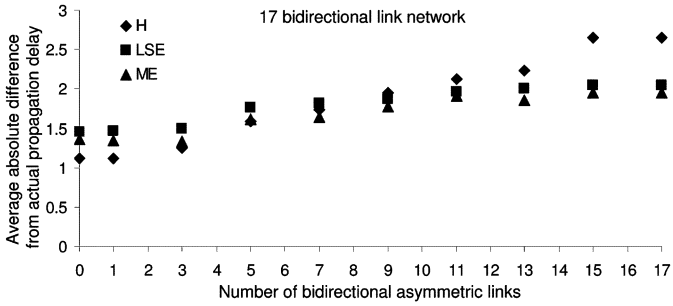

Fig. 8. With queueing delay.

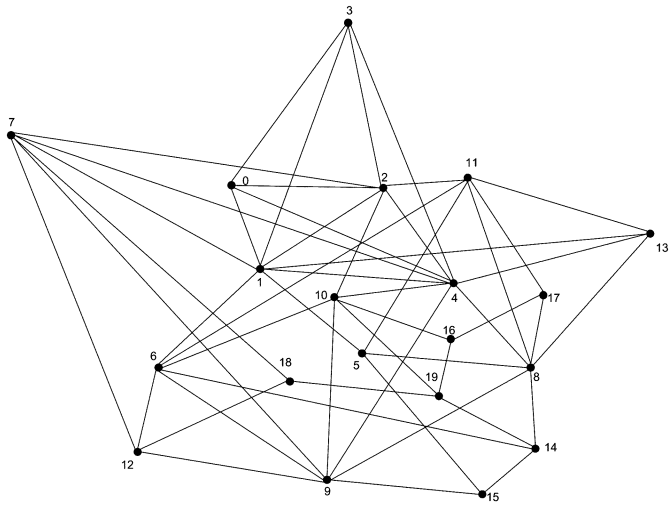

Fig. 9. 20-node 102-link network.

In order to better understand the effect of symmetry we ran the setup without queueing delay (Fig. 7) and with queueing delay (Fig. 8). Note that without queueing delay, the halving scheme performs perfectly for the symmetric links. From the respective figures we observe that as long as the number of asymmetric links is small, the halving scheme is indeed better. Yet, when the number of asymmetric links increase (beyond three (five) without (with) queueing delay), the estimation error of the propagation delay is smaller with the ME scheme compared to the halving scheme. We also observe that the ME scheme outperforms the LSE scheme for any number of asymmetric links and the LSE scheme is better than the halving scheme beyond nine asymmetric links.

To complete this part of the numerical results we run the ME over a larger network described in Fig. 9 and compare the results with the halving scheme.

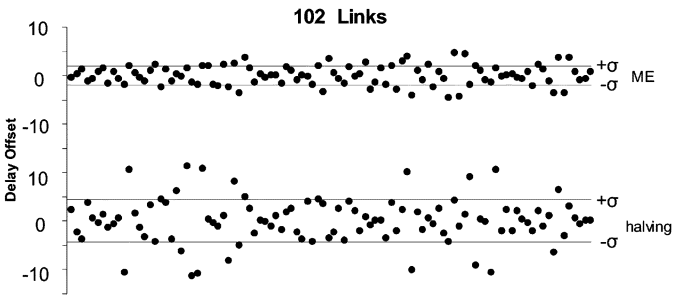

Fig. 10. The difference between the measured minimum delay and the estimated minimum delay. The standard deviation σ equals 1.9 and 4.4 for the ME and halving schemes, respectively.

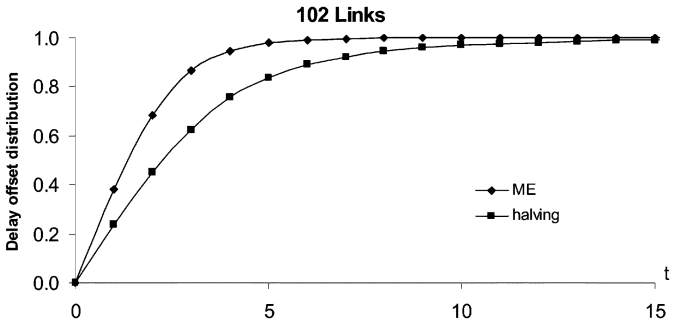

Fig. 11. The fraction of links with estimated propagation delay difference with respect to the minimum delay obtained on their respective links is not greater than t.

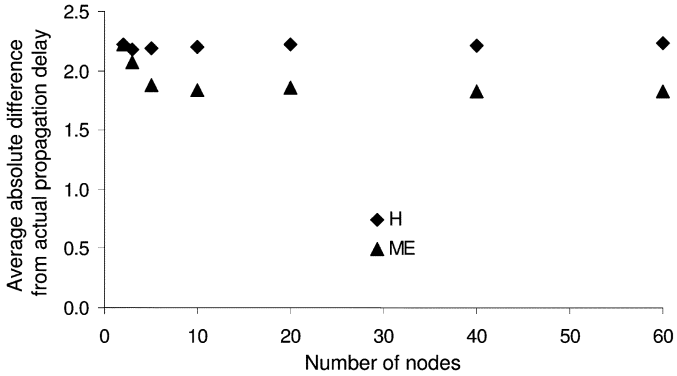

Fig. 12. Average absolute difference from actual propagation delay.

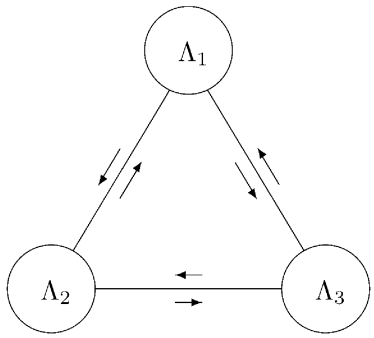

Fig. 13. Three-node fully connected network.

In Fig. 10, we demonstrate the difference between the measured minimum delay and the estimated minimum delay. In the graph, the x-axis denotes the link ID and the y-axis is the difference between the minimum delay experienced on the link minus the estimated minimum delay. Fig. 10 clearly depicts that the ME scheme that bases its estimation on more than one path traversing each link is much less dispersed than the halving scheme. In order to emphasize the difference, we add to each graph the standard deviation-σ. Note that the region between the $+\sigma$ and the $-\sigma$ in the halving scheme is much wider than the one in the ME scheme.

Next, we operated the ME and halving schemes over the 20–nodes 102–link network described in Fig. 9 100 times, where each time we chose parameters randomly. Based on the results, we draw the distribution of the difference between the estimated propagation delay and the minimum delay measured on the link. In Fig. 11, the y-axis presents the fraction of links that their estimated propagation delay difference with respect to the minimum delay obtained on the link is not greater than the time depicted by the x-axis value. Fig. 11 clearly demonstrates the significant improvement of the use of the ME over the halving scheme. For example, 70% of the links have their estimated delay less than two time units from the propagation delay when using the ME compared to only 45% of the links in the halving scheme.

Finally, we run the halving and the ME for a series of topologies with different number of nodes (2, 3, 5, 10, 20, 40, and 60 nodes) and depict the average absolute difference between the estimated and the actual propagation delays in Fig. 12. It is clear from this figure that the halving scheme is not influenced by the size of the network. ME yields the same quality estimates as halving (two-node two-link network). However, its quality improves as the size of the network increases. Obviously, the marginal gain becomes smaller and smaller; we do not expect it to become zero even when the network is very big (since the number of constraints (cycles) always fall behind the number of links by $N-1$).

C. One-Way Link Delay

The third part of our numerical results is dedicated to evaluating the performance of the combined schemes that estimate the propagation delay as well as the queueing delay.

In order to evaluate the quality of total delay estimation attained using the suggested algorithm, we applied it to two networks of different sizes.

As before, the delays are assumed to be distributed according to a shifted exponential distribution. The propagation delay on each link was chosen for each direction separately based on uniform distribution ($\sim U[0,10]$). The queueing delay of each directed link was sampled from an exponential distribution with a mean that was randomly selected between 0.1 and 5 (uniformly). The clock offset with respect to the "reference time node" is randomly chosen from a uniform distribution between -10 and 10 ($\sim U[-10,10]$) based on [38]. For each link, 30 probe packets are transmitted and the estimation of both the constant delay and the variable delay is based on these packets.

We start with the simple example of a three-node network depicted in Fig. 13. In order to evaluate our scheme, we compare it to a scheme which estimates the propagation delay on each directional link by halving the minimum round-trip delay

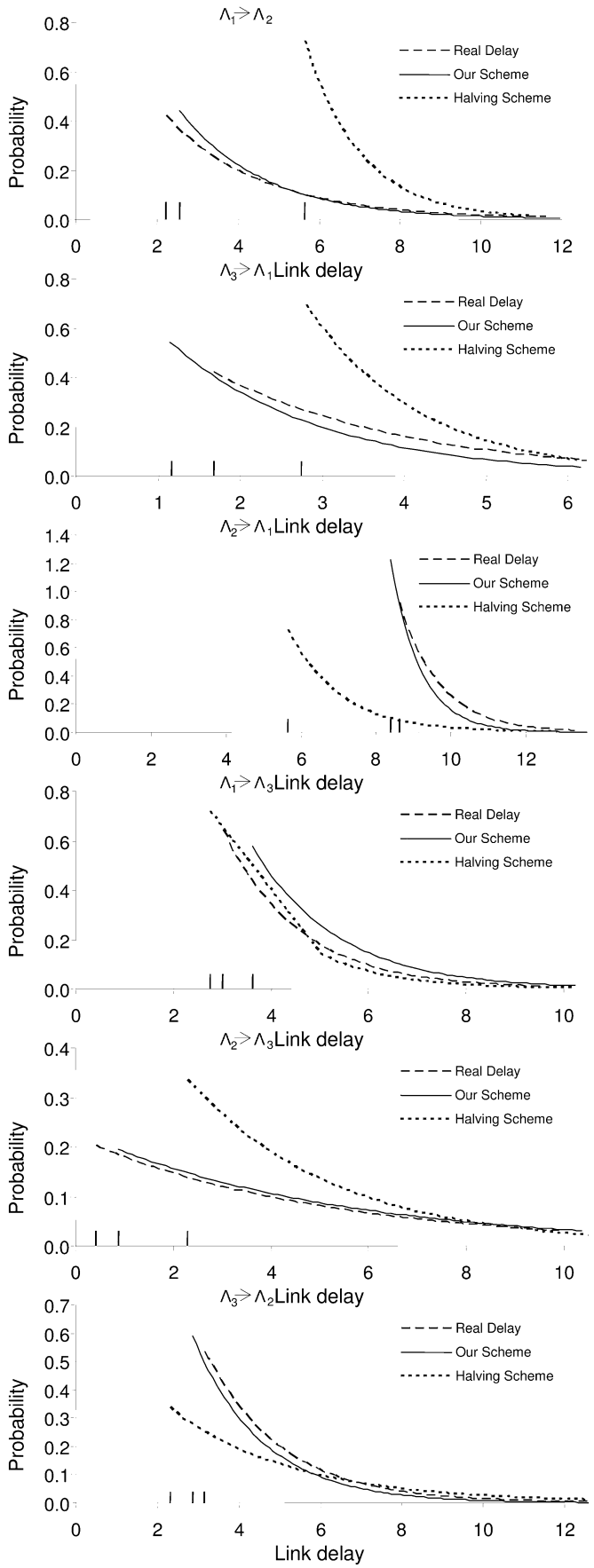

Fig. 14. Probability density function.

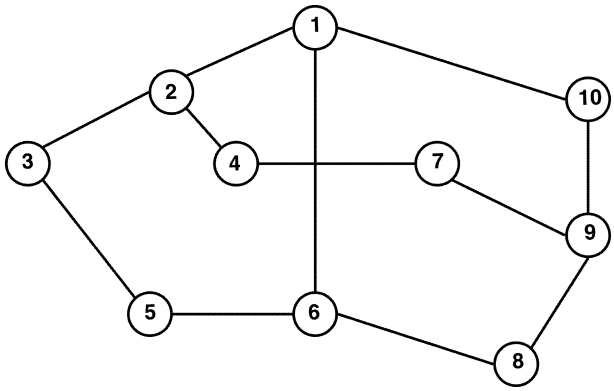

Fig. 15. 10-node 24-link connected network.

obtained by any packet exchange on the bidirectional link (the same as halving in the first part). The variable delay according to this scheme is obtained by measuring the average round-trip delay experienced by the 30 packets, and halving the obtained average. Note that by relying on round-trip delay measurements, we eliminate the clock offset from the measurements. We denote this scheme as halving. The propagation delay according to our scheme was estimated only according to the maximum entropy scheme.

Fig. 14 shows the total delay density function of the six one-way links obtained by our scheme and the halving scheme, compared to the real one-way link delay density function. Note that the point in which each graph starts (the minimum x-axis value obtained by the graph, which is marked) is the constant delay part. Fig. 14 clearly depicts a significant improvement of the total delay estimation using the suggested scheme over the common halving technique, even in the simple case of a three node network.

As a second example, consider the ten-node network with $E = 24$ directional links depicted in Fig. 15. Fig. 16 presents the total delay (constant + variable) density function of six selected links.

As with the previous example, it can be seen clearly that the estimation resulting from our suggested scheme is much better both for estimating the constant delay and for estimating the variable delay.

VII. Implementation Example

Besides the encouraging results obtained by our scheme, which are demonstrated in Section VI, another advantage of our algorithm is the ability to implement it on existing networks without modifications. Unlike previous schemes that require mechanisms that are not commonly supported in current IP networks such as multicast [11] or source routing [14], the suggested scheme can be implemented as is in IP networks. In this section, we describe an implementation example of the suggested scheme in a Cisco router based network.

The implementation example is based on the Cisco Service Assurance Agents (SAA), which provide a variety of service monitoring information including time-delay reporting [40]. SAA is part of every Cisco device equipped with IOS version 12.0 and above. Each participating SAA will monitor and

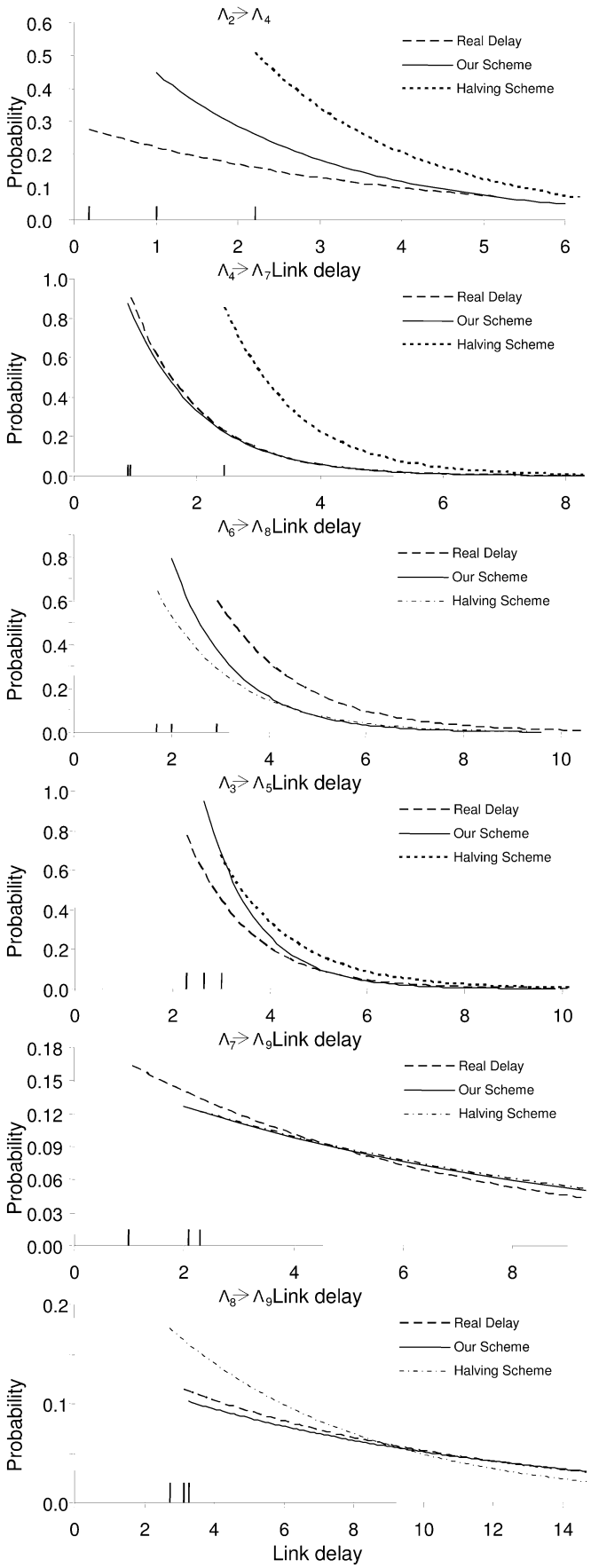

Fig. 16. Probability density function.

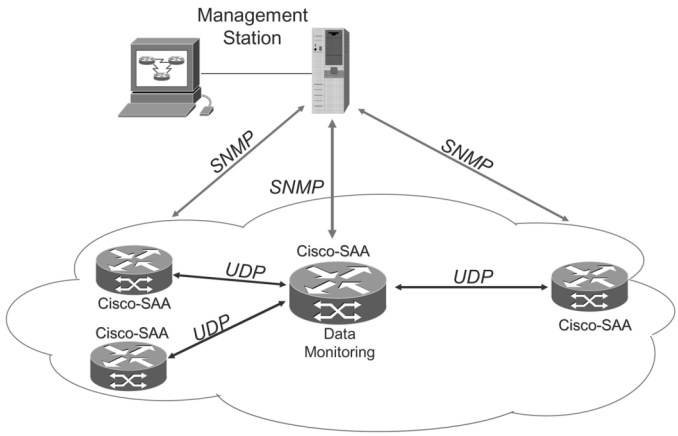

Fig. 17. Implementation example: Each SAA monitors the minimum one-way delay to each of its neighboring routers. A management station collects the data via SNMP and computes the delays.

filter the minimum one-way delay to each of its neighboring routers. Note that each router knows its neighbors and therefore can conduct the measurements only with neighboring routers, hence no source routing is needed. The data gathered by all the SAAs is collected by a management station using Simple Network Management Protocol (SNMP) (Fig. 17). The management station computes the cyclic paths and calculates the constraints from the collected data, as explained in Section III and estimates the delays on each link according to the schemes suggested in Sections IV and V.

VIII. Discussion

This study focuses on the essential problem of achieving accurate one-way link delay estimates in an unsynchronized network. We introduce a novel approach for estimating the constant and the variable parts of the one-way delays. The new approach for estimating the constant one-way delay is based on one-way single-hop measurements based on standard ICMP of NTP probes and exploiting the global objective function optimization principle. The variable one-way delay estimation is based on measuring and analyzing the link between neighboring nodes. We proposed two objective functions.The natural choice of LSE and the better ME scheme make a major improvement over the standard halving.

The suggested schemes are easy to implement and can be incorporated in current Internet standards (e.g., using NTP or ICMP probe messages). Numerical results show that our approach works well and substantially outperforms other known schemes. Good delay estimation can be achieved even in a small network, for example, an overlay network that consists only of a small fraction of nodes.

Acknowledgment

The authors would like to thank Prof. Hanoch Levy for helpful discussions.

References

[1] V. Paxson, "End-to-end internet packet dynamics," *IEEE/ACM Trans. Netw.*, vol. 7, no. 3, pp. 277–292, Jun. 1999.

[2] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and C. Diot, "Packet-level traffic measurements from the sprint IP backbone," *IEEE Network*, vol. 17, no. 6, pp. 6–16, 2003.

[3] G. Almes, S. Kalidindi, and M. Zekauskas, "A One-Way Delay Metric for IPPM," report, RFC 2679, 1999.

[4] D. L. Mills, "Improved algorithms for synchronizing computer network clocks," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, pp. 245–254, Jun. 1995.

[5] ——, "Network Time Protocol (Version 3) Specification, Implementation and Analysis," Univ. Delaware, Newark, Network Working Group Report RFC-1305, 1992.

[6] D. Veitch, S. Babu, and A. Pasztor, "Robust synchronization of software clocks across the internet," in *Proc. Internet Measurement Conf.*, Taormina, Italy, Oct. 2004, pp. 219–232.

[7] O. Gurewitz, I. Cidon, and M. Sidi, "Network time synchronization using clock offset optimization," in *Proc. Int. Conf. Network Protocols (ICNP 2003)*, Atlanta, GA, Nov. 2003, pp. 212–221.

[8] J. Elson, R. Karp, C. Papadimitriou, and S. Shenker, "Global synchronization in sensornets," in *Proc. 6th Latin American Symp. Theoretical Informatics (LATIN'04)*, Buenos Aires, Argentina, Apr. 2004, pp. 609–624.

[9] Measuring Delay, Jitter, and Packet Loss With Cisco IOS SAA and RTTMON. [Online]. Available: http://www.cisco.com/warp/public/126/saa.html

[10] E. A. Zeitoun, C. N. Chuah, S. Bhattacharrya, and C. Diot, "An as-level study of internet path delay characteristics," in *Proc. IEEE GLOBECOM 2004*, vol. 3, Dallas, TX, Nov. 2004, pp. 1480–1484.

[11] F. LoPresti, 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, Dec. 2002.

[12] Y. Shavitt, X. Sun, A. Wool, and B. Yener, "Computing the unmeasured: An algebraic approach to internet mapping," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 1, pp. 67–78, Jan. 2004.

[13] H. Burch, "Measuring an IP network in situ," Ph.D. dissertation, Carnegie Mellon Univ., School of Computer Science, Pittsburgh, PA, 2005.

[14] O. Gurewitz and M. Sidi, "Estimating one-way delays from cyclic-path delay measurements," in *Proc. IEEE INFOCOM 2001*, Anchorage, AK, Apr. 2001, pp. 1038–1044.

[15] V. Paxson, "On calibrating measurements of packet transit times," in *Proc. ACM SIGMETRICS 1998*, Madison, WI, Jun. 1998, pp. 11–21.

[16] S. Moon, P. Skelley, and D. Towsley, "Estimation and removal of clock skew from network delay measurements," in *Proc. IEEE INFOCOM 1999*, New York, NY, Mar. 1999.

[17] L. Zhang, Z. Liu, and C. H. Xia, "Clock synchronization algorithms for network measurements," in *Proc. IEEE INFOCOM 2002*, New York, NY, Jun. 2002, pp. 160–169.

[18] A. Pasztor and D. Veitch, "PC based precision timing without GPS," in *Proc. ACM SIGMETRICS*, vol. 30, Los Angeles, CA, Jun. 2002, pp. 1–10.

[19] M. Garetto and D. Towsley, "Modeling, simulation and measurements of queuing delay under long-tail internet traffic," in *ACM Sigmetrics 2003, Proc. Int. Conf, Measurement and Modeling of Computer Systems*, San Diego, CA, Jun. 2003, pp. 47–57.

[20] S. Kalidindi and M. J. Zekauskas, "Surveyor: An infrastructure for internet performance measurements," in *Proc. 9th Internet Soc. Conf. (INET)*, San Jose, CA, 1999.

[21] V. E. Paxson, "Measurements and Analysis of End-to-End Internet Dynamics," Ph.D. dissertation, Lawrence Berkeley Nat. Lab., Univ. California, Berkeley, 1997.

[22] C. Fraleigh, C. Diot, B. Lyles, S. Moon, P. Owezarski, D. Papagiannaki, and F. Tobagi, "Design and deployment of a passive monitoring infrastructure," in *Proc. In Passive and Active Measurement Workshop*, Amsterdam, The Netherlands, Apr. 2001.

[23] D. Papagiannaki, S. Moon, C. Fraleigh, P. Thiran, F. Tobagi, and C. Diot, "Analysis of measured single-hop delay from an operational backbone network," in *Proc. IEEE INFOCOM*, vol. 2, New York, NY, Jun. 2002, pp. 535–544.

[24] F. Harary, *Graph Theory*. Reading, MA: Addison Wesley, 1994.

[25] T. J. McCabe, "A complexity measure," *IEEE Trans. Softw. Eng.*, vol. SE-2, pp. 308–320, 1976.

[26] P. C. Kainen, "On robust cycle bases," in *Graph Theory, Combinatorics, Algorithms, and Applications*, Y. Alavi, D. M. Jones, D. R. Lick, and J. Liu, Eds. Amsterdam, The Netherlands: Elsevier, Jul. 2000, Electronic Notes in Discrete Mathematics 11, pp. 439–437.

[27] C. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 398–403, 1948.

[28] E. T. Jaynes, "Information theory and statistical mechanics i," *Phys. Rev.*, vol. 106, pp. 620–630, 1957.

[29] ——, "On the rationale of maximum-entropy methods," *Proc IEEE*, vol. 70, no. 9, pp. 939–952, Sep. 1982.

[30] ——, *Probability Theory: The Logic of Science*. St. Louis, MO: Washington Univ. Press, 1996.

[31] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*. Philadelphia, PA: SIAM, 1995.

[32] S. Boyd and L. Vandenberghe. (2002) Convex Optimization. [Online]. Available: www.stanford.edu/class/ee364 and www.ee.ucla.edu/ee236b

[33] D. Kazakos and P. Papantoni-Kazakos, *Detection and Estimation*. Rockville, MD: Computer Sci., 1990.

[34] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley-Interscience, 2000.

[35] A. Mukherjee, "On the dynamics and significance of low frequency components of internet load," *Internetworking: Research and Experience*, vol. 5, no. 4, pp. 163–205, 1994.

[36] N. L. Johnson and S. Kotz, *Continuous Univariate Distributions-1*. New York: Wiley, 1970.

[37] E. W. Zegura, K. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proc. IEEE INFOCOM 1996*, San Francisco, CA, Mar. 1996, pp. 594–602.

[38] N. Minar. (1999) A Survey of the NTP network. [Online]. Available: http://www.media.mit.edu/nelson/research/ntp-survey99/

[39] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.

[40] *Cisco IOS Configuration Fundamentals Configuration Guide*, 2003.