CCIT Report #431 June 2003

Maximum Entropy Based One-Way Delay Estimation

Omer Gurewitz, Israel Cidon and Moshe Sidi
Electrical Engineering Department
Technion, Haifa 32000
Israel

Abstract

We present a novel approach for the estimation of one-way delays between neighboring nodes without requiring any time synchronization in the network. The approach is based on the maximum entropy principle and is taking into account the asymmetric nature of the network and links, and the fact that traffic flows are not necessarily the same in both directions. Our approach is based on conducting multiple and simple delay measurements among multiple neighboring pairs of nodes, and estimating the one-way delays by maximizing the value of an objective function (entropy based) and by exploiting the graph topology. The procedures described in the paper provide estimation for both the fixed part (i.e., propagation) and the variable part (i.e., queueing) of a link one-way delay. These procedures are easy to implement and require only minor modifications in the algorithms and message formats already used in the Internet. Numerical examples show the advantages of the proposed estimation schemes.

Keywords

Mathematical optimization, network measurements, delay estimation, one-way delay, maximum entropy.

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 the 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 IP. Delay measurements are common in such networks. For example, ad-hoc networks use delay measurements for the estimation of pair-wise distances between group members that are needed for the routing layer [1],[2]. Furthermore, continuous monitoring of the delay is essential in many applications in order to check compliance with critical delay constraints.

For many essential applications the quantity of interest is the one-way delay between a source and a destination rather than the round-trip delay. This is motivated by several factors [3]. In many networks the path from a source to a destination may be different from the path from the destination back to the source (asymmetric paths). Measuring each path independently highlights the performance difference between the two paths which may traverse radically different types of networks. Even when the two paths are symmetric, they may have different performance characteristics due to asymmetric loads or different QoS provisioning. Performance of many applications depends mostly on the performance in one direction. For example, a file transfer performance depends more on the performance from the source to 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.

The basic difficulty of measuring one-way delays is that clocks in a network are not tightly synchronized. Obviously, if clocks in the network were synchronized the task of measuring one-way delays would be simple: one end node sends a probe packet with its time stamp on it; the difference between the arriving time and the transmission time is the one-way link delay. Global Positioning Systems (GPS) provide accurate time synchronization; 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 on the Internet [4], [5], [6]. NTP is designed to synchronize clocks in a network with respect to Universal Time-Coordinated (UTC). NTP is based on symmetric link models, and even for these models it gives only a good estimate regarding the

clock offsets with respect to UTC in the network. There are other schemes which also time synchronize clocks in a network, e.g. [7], [8]. However, all these schemes are only heuristics which give an approximate solution to the synchronization problem with respect to universal clock.

Round-trip delay measurements, on the other hand, are simple to conduct and are very accurate since the same clock is used while transmitting the packet and upon its return; a common approach which is used for estimating one-way delay is to measure round-trip delays and halve them. For this approach to work well, not only the route between source and destination should be the same, but traffic loads and QoS configurations in both directions should also be the same.

In this paper we present a novel approach for the estimation of one-way delays from one-way measurements that do not require any kind of synchronization among the nodes of the network. The approach is based on the *maximum entropy principle* and is taking into account the asymmetric nature of the network, and the fact that path characteristics are not necessarily the same in both directions. The procedures described in the paper provide estimation for both the fixed part (i.e., propagation) and the variable part (i.e., queueing) of a link one-way delay. These procedures are easy to implement and require only minor modifications in the algorithms and message formats already used in the Internet.

A scheme to estimate one-way delay from cyclic-path measurements was proposed in [9]. The main assumption in [9] was that the delay of links is constant – a very limiting assumption. A major contribution of the current paper is to show how to estimate the one-way delays for general delays. Another drawback of the approach in [9] was the need to send probe packets along several hops in a cyclic manner. This required special packets along with some source-routing mechanism. In contrast, a significant contribution of this paper is to show that simple NTP-like or standard ICMP Ping packets can be used to estimate one-way delays. Finally, the estimation in [9] was based on the minimum square error principle that was computationally intensive, while the maximum entropy principle used in this paper yields itself to relatively simple computations and yet result in a better estimation accuracy for most cases checked. Related papers that use *multicast* packets for delay estimation appear in [10], [11].

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 beginning with the estimation of the fixed part of the delay and continuing with the estimation of the variable part of the delay. The high quality of the proposed estimation is assessed in Section V where numerical examples are provided.

II. The Model

The goal of this study is to design and develop a novel technique for estimating one-way delays between the nodes of the network, based on one-way measurements conducted between these nodes. We begin by introducing the network model that is used throughout this paper. We split it into two aspects: topology and delay.

A. Network Topology Model

The communication network is composed of a set of communication nodes which are connected by physical links. Naturally not all communication nodes are interested or capable in participating in the protocol. We will focus throughout this paper on an overlay network which consists of the components that do participate in the round-trip delay 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 a directed path 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 $e_{i,j}$ and the collection of links by $\mathcal{E}$. Note that each link can be composed of several physical segments. We will assume throughout the paper that all links are bidirectional, namely if $e_{i,j} \in \mathcal{E}$, then $e_{ji} \in \mathcal{E}$ (if $e_{i,j}$ exists so does e_{ji}). We will not assume though, that the two links are symmetric. On the contrary, they can be composed of different physical links and/or can have different capacities on each direction. 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 $\hat{\tau}_i$. By $\hat{\tau}_{i,j}$ we will denote the relative offset of node Λ_i clock with respect to node Λ_j clock, i.e., $\hat{\tau}_{i,j} = \hat{\tau}_i - \hat{\tau}_j$. Clearly, $\hat{\tau}_{j,i} = \hat{\tau}_j - \hat{\tau}_i = -\hat{\tau}_{i,j}$. We assume that the clock drift is negligible compared to the measurement procedure interval. Since our scheme can exploit NTP messages, jumps in clocks caused by NTP do not influence the measurements.

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 three factors: *(i) Buffering delay* which is the time needed to buffer a packet at each physical node along the path, and prepare it for transmission. *(ii) Transmission delay* which is the time needed to transmit the packet, first bit to last, by each physical node along the path comprising the link. *(iii) Propagation delay* which is the time a bit propagates along the link. Since throughout this paper all measurement packets that traverse a specific link will have the same format and size, and since the physical links comprising a link do not change, we will assume that the deterministic part of the delay on each link is constant for all packets travelling the link. Note, though that due to the asymmetric characteristic of the link, we will not assume that the constant part of the delay on the two directions of a link is 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 $x_{i,j}$ the delay on the link connecting node Λ_i and Λ_j, and by $c_{i,j}$ and $v_{i,j}$ its constant and variable parts, respectively ($x_{i,j} = c_{i,j} + v_{i,j}$). The distribution function and the density function of the two random delays will be denoted by: $F_{x_{i,j}}$, $F_{v_{i,j}}$ and $f_{x_{i,j}}$, $f_{v_{i,j}}$, respectively.

III. The Measurements

In this paper we suggest that all measurements be one-way measurements, and the message format be the one suggested by NTP which is the widely accepted standard for synchronizing clocks in the Internet [4],[6],[12].

The measurements will be carried out in the following manner: Each node is continuously sending probe packets every so often to each one of its neighbors. Time is stamped on packet k by the sender Λ_i upon transmission to node Λ_j ($T_{i,j}^{[k]}$). The receiver Λ_j stamps its local time upon receiving the packet ($R_{j,i}^{[k]}$). Each packet k will eventually have two time stamps on it: $T_{i,j}^{[k]}$ and $R_{j,i}^{[k]}$ (these time stamps are part of the standard NTP packets).

We intend to estimate one-way delays by looking at the n most recent packets. For the link $e_{i,j} \in \mathcal{E}$ connecting the two nodes Λ_i and Λ_j, let $x_{i,j}^{[k]}$ be the one-way link delay experienced by probe packet k while traveling from node Λ_i to Λ_j.

Let us also denote by $\Delta T_{i,j}^{[k]}$ 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., $\Delta T_{i,j}^{[k]} = R_{j,i}^{[k]} - T_{i,j}^{[k]}$. Note that the different times are taken according to different clocks which are not necessarily synchronized, hence the computed time $\Delta T_{i,j}^{[k]}$ is not the one-way delay but rather the sum of the one-way link delay experienced by probe packet k while traveling between node Λ_i to Λ_j and the time difference between the two clocks, i.e., $\Delta T_{i,j}^{[k]} = x_{i,j}^{[k]} - \hat{\tau}_i + \hat{\tau}_j = x_{i,j}^{[k]} - \hat{\tau}_{i,j}$. Note that $\Delta T_{i,j}^{[k]}$ can be negative.

It is interesting to note that the sum $\Delta T_{i,j}^{[k_1]} + \Delta T_{j,i}^{[k_2]}$ for arbitrary k_1 and k_2 represents the round-trip delay of a virtual packet that was 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 $\Delta T_{i,j}^{[k_1]} + \Delta T_{j,i}^{[k_2]} = x_{i,j}^{k_1} - \hat{\tau}_{i,j} + x_{j,i}^{k_2} - \hat{\tau}_{j,i} = x_{i,j}^{k_1} + x_{j,i}^{k_2}$.

IV. Delay Estimation

In section II-B we separated the link delay into two basic components, constant and variable. In this section we present the estimation procedures for each component.

A. Constant Delay Estimation

Determination of the one-way delay between nodes Λ_i and Λ_j faces the problem that the two time stamps $T_{i,j}^{[k]}$ and $R_{j,i}^{[k]}$ are based on two different local clocks which are not synchronized. Had delay measurements been taken along a cyclic path, they would have been accurate with no need for synchronization since the same clock is being used at the source and the destination (in a cyclic path the first and the last node of the path are identical).

In this section we propose a way 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 constraints the one-way link delays along the path to equal a specific value. In [9] it was proved that in an N-node connected network the maximal number of independent cyclic paths and thus independent constraints is $E-(N-1)$.

Remember that each constraint relates to the total delay along the cyclic path. In order to extract the constant delay along the $E-(N-1)$ 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 between node Λ_i and Λ_j the packet with the smallest entry $\min_k \Delta T_{i,j}^{k}$ 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., $\Delta T_{i,j}^{[\min]} = \min_k \Delta T_{i,j}^{[k]}$.

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 the 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 $\Delta T_{i,j}^{[min]}$'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 $\Delta T_{i,j}^{[min]}$ and only then applying the sum over all the minimum delays. This procedure is much more probable to yield the constant cyclic-path delay than the cyclic path measurement of [9].

We turn now for the selection of the $E-(N-1)$ independent cyclic paths. Obviously, $\frac{E}{2}$ independent cyclic paths are all the unordered pairs of adjacent nodes, i.e., the round trip on each bidirectional link. Note that even for the single hop round trip delay measurements, 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, $\Delta T_{i,j}^{[min]}+\Delta T_{j,i}^{[min]} \leq \min_{[k]}(\Delta T_{i,j}^{[k]}+\Delta T_{j,i}^{[k]})$. The remaining $\frac{E}{2}-N-1$ independent cyclic-path delays can be taken from any set of independent cyclic paths were the deterministic delay is computed as the sum of the $\Delta T_{i,j}^{[min]}$ comprising the path. An algorithm for building this set is proposed in the Appendix.

Let us denote each of the chosen independent cyclic paths by $\theta_k \quad 1 \leq k \leq E-(N-1)$ 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 constant one-way link delays $\{c_{i,j} | \forall e_{i,j} \in \mathcal{E}\}$. Additional constraints are the non-negativity of the variables $\{c_{i,j} \geq 0 | \forall e_{i,j} \in \mathcal{E}\}$.

Now that we have established the constraints upon the constant delay estimation process, we turn to define an objective function to be optimized.

B. The Maximum Entropy (ME) Principle

In order to assess the quality of the one-way constant delay estimates, one has to define an objective function that allows comparison of different solutions. Note that since the constraints do not contain enough information for obtaining the real delay values, no matter how many measurements one takes, there is no objective function which can determine the delays indubitably.

In order to decide upon an objective function let us examine the following conceptual experiment. Suppose that there is a special packet which is travelling throughout 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. Assume that for the cyclic paths in Θ we can determine the probability of finding the packet along the path. Let $\alpha_k,\ 1 \leq k \leq E-(N-1)$ denote this probability for path θ_k. 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 probability of finding the packet on each link $e_{i,j} \in \mathcal{E}$. Each component $p_{i,j}$ can be assigned only positive values. Our estimation should be based on $E-(N-1)$ linear equations of the vector components $\sum_{e_{i,j}\in\theta_k} p_{i,j} = \alpha_k\ k \in \Theta$. The solution should satisfy the conditions that $p_{i,j}$ is positive and satisfies the $E-(N-1)$ cyclic path probability constraints.

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 [13],[14],[15],[16]. 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 $p_{i,j}$ follows the solution of the maximal entropy $-\sum_{e_{i,j}\in\mathcal{E}} p_{i,j}\log p_{i,j}$ under the above constraints.

We now turn to show the relation between the conceptual experiment 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 correlated with the link delay. Furthermore, the relative probabilities of finding the packet on two different links should be the ratio between their link delays. For example, if the delay along link a is twice the delay along link b it should be clear that the probability of finding the packet on link a should be twice the probability of finding the packet on link b. Therefore, since these probabilities should sum up to one, they should be equal to: $p_{i,j} = \frac{c_{i,j}}{\sum_{e_{i,j}\in\mathcal{E}} c_{i,j}}$, where $c_{i,j}$ is the constant delay along link $e_{i,j}$.

Now let us return to our original goal of determining the propagation link delays. Since we know how to extract the packets that experienced negligible queueing delay (section IV-A), we can assume that the delay is constant. It is easy to see that $\sum_{e_{i,j}\in\mathcal{E}} c_{i,j} = \sum_{e_{i,j}\in\mathcal{E}} \Delta T_{i,j}^{[min]}$. The reason is that summing $\Delta T_{i,j}^{[min]}$ over the two directions between two nodes, the clock offset is eliminated. Let $C = \sum_{e_{i,j}\in\mathcal{E}} c_{i,j}$.

The probability sums along the selected cyclic paths can be also easily determined based on the probe packets that experienced the minimum delay on each directed link (IV-A): $\alpha_k = \sum_{e_{i,j}\in\theta_k} p_{i,j} = \sum_{e_{i,j}\in\theta_k} \frac{c_{i,j}}{\sum_{e_{i,j}\in\mathcal{E}} c_{i,j}} = \sum_{e_{i,j}\in\theta_k} \frac{\Delta T_{i,j}^{[min]}}{\sum_{e_{i,j}\in\mathcal{E}} \Delta T_{i,j}^{[min]}} = \frac{1}{C}\sum_{e_{i,j}\in\theta_k} \Delta T_{i,j}^{[min]}\ k \in \Theta$.

In the constant one-way delay estimation problem we also require that the delay variable $c_{i,j}$ 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 travelling along each link. Hence the principle of Maximum Entropy (ME) can be exploited.

Consequently, the optimization problem at hand is the following:

Maximize the information theoretical entropy:

$$S = -\sum_{e_{i,j}\in\mathcal{E}} p_{i,j}\log p_{i,j} \tag{1}$$

subject to the constraints:

$$p_{i,j} \geq 0 \quad ; \quad \sum_{e_{i,j}\in\mathcal{E}} p_{i,j} = 1 \tag{2}$$

and

$$\sum_{e_{i,j}\in\theta_k} p_{i,j} = \frac{1}{C}\sum_{e_{i,j}\in\theta_k} \Delta T_{i,j}^{[min]} = \alpha_k \quad \forall\theta_k \in \Theta \tag{3}$$

To maximize (1) subject to the constraints (2) and (3) we employ the method of Lagrange multipliers. The relevant steps are briefly outlined in the following.

The Lagrangian will take the form:

$$\begin{aligned}\mathcal{L}(p_{i,j},\lambda_k) = &-\sum_{e_{i,j}\in\mathcal{E}} p_{i,j}\log p_{i,j} \\ &-\sum_{k=1}^{E-(N-1)} \lambda_k\left(\sum_{e_{i,j}\in\theta_k} p_{i,j} - \alpha_k\right) \\ &-(\lambda_0 - 1)\left(\sum_{e_{i,j}\in\mathcal{E}} p_{i,j} - 1\right)\end{aligned} \tag{4}$$

Taking the derivative,

$$\frac{\partial\mathcal{L}(p,\lambda)}{\partial p_{i,j}} = -\log p_{i,j} - \lambda_0 - \sum_{k=1}^{E-(N-1)} \lambda_k\delta(e_{i,j}\in\theta_k) = 0 \tag{5}$$

where $\delta(e_{i,j}\in\theta_k) = \begin{cases} 1 & e_{i,j}\in\theta_k \\ 0 & otherwise \end{cases}$
and the probabilities are:

$$p_{i,j} = e^{-\lambda_0-\lambda_1\cdot\delta(e_{i,j}\in\theta_1)-\quad\cdots\quad-\lambda_{E-(N-1)}\cdot\delta(e_{i,j}\in\theta_{E-(N-1)})} \tag{6}$$

The Lagrange multiplier λ_0 is determined by substituting (6) into (2)

$$\begin{aligned}e^{-\lambda_0}\cdot\sum_{e_{i,j}\in\mathcal{E}} e^{-\lambda_1\cdot\delta(e_{i,j}\in\theta_1)-\cdots-\lambda_{E-(N-1)}\cdot\delta(e_{i,j}\in\theta_{E-(N-1)})} \\ = 1\end{aligned} \tag{7}$$

If we now define a partition function as:

$$\begin{aligned}&Z(\lambda_1,\ldots,\lambda_{E-(N-1)}) \\ &\quad\equiv\sum_{e_{i,j}\in\mathcal{E}} e^{-\lambda_1\cdot\delta(e_{i,j}\in\theta_1)-\cdots-\lambda_{E-(N-1)}\cdot\delta(e_{i,j}\in\theta_{E-(N-1)})}\end{aligned} \tag{8}$$

then (7) reduces to

$$\lambda_0 = logZ(\lambda_1,\ldots,\lambda_{E-(N-1)}) \tag{9}$$

The rest of the Lagrange multipliers $\lambda_i \quad i = 1,\ldots,(E-(N-1))$ are determined by substituting (6) and (9) in (3)

$$\frac{\partial}{\partial\lambda_k}\log Z = \alpha_k \tag{10}$$

a set of $E-(N-1)$ equations for $E-(N-1)$ unknowns. In order to solve the set of equations and determine $\lambda_i \quad i = 1,2,\ldots,E-(N-1)$, one has to use one of many iterative methods [17], [18].

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 should 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 on 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:

$$d_{i,j} = \frac{1}{Z} e^{-\lambda_1 \cdot \delta(e_{i,j} \in \theta_1) - \quad \cdots \quad -\lambda_{E-(N-1)} \cdot \delta(e_{i,j} \in \theta_{E-(N-1)})} \tag{11}$$

and therefore the one-way constant delays are given by:

$$c_{i,j} = C \cdot p_{i,j} \quad \forall e_{i,j} \in \mathcal{E} \tag{12}$$

C. Variable Delay Estimation

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

In the present work we assume that the variable delay distribution type is known and we will use the Bayesian method which utilizes the prior subjective knowledge in conjunction with the measurements (unmodified $\Delta T_{i,j}$). In addition to the estimation of the set of parameters characterizing the variable delay, $C_{i,j}$ should also be estimated. For example, if we know that the variable delay along a link has a Gama distribution, i.e., $f_{x_{i,j}}(t) = \frac{(t-C_{i,j})^{\alpha-1} e^{-\frac{(t-C_{i,j})}{\beta}}}{\beta^{\alpha}\Gamma(\alpha)}$, we can use the method of Maximum Likelihood in order to estimate the parameters α, β, $C_{i,j}$ [19]. We will omit further discussion regarding the Bayesian method since it is well covered in the literature.

V. Numerical examples

A. The Underlying Network

In order to evaluate the delay estimation attained using the suggested algorithm, we apply it on several sample networks of different sizes. The important parameters which characterize each entity in the network are randomly chosen. The propagation delay on each link is chosen for each direction separately based on normal distribution where the mean and variance are uniformly selected between 10 to 40 and between 5 to 15, respectively ($\sim U[10, 40]$ and $\sim U[5, 15]$). The queueing delay of each directed link is sampled from an Exponential distribution with mean that is randomly selected between 0.1 to 5 (uniformly). The clock offset with respect to the "Reference Time Node" is randomly chosen with a uniform distribution between -10 to 10 ($\sim U[-10, 10]$). For each link, thirty probe packets are transmitted and the estimation of both the constant delay and the variable delay is based on these packets.

B. The Results

We separate the numerical results into two parts. In the first part we examine the estimation of only the propagation link delay. In the second part we evaluate the performance of the combined suggested schemes which estimate both the propagation delay (sections IV-A and IV-B) as well as the queueing delay (section IV-C).

We start by examining the estimation of the propagation delay, based on the ME principle suggested in section IV-B. We apply our scheme on the five node network shown in Fig. 2. Since in this part we are interested in evaluating only the propagation delay estimation, we ensure that on each link at least one packet will experience no queueing delay, i.e. $\Delta T_{i,j}^{min}$ on each link is the propagation delay plus the relative clock offset between the two nodes in the two ends of the link. We run our scheme 100 times on the network.

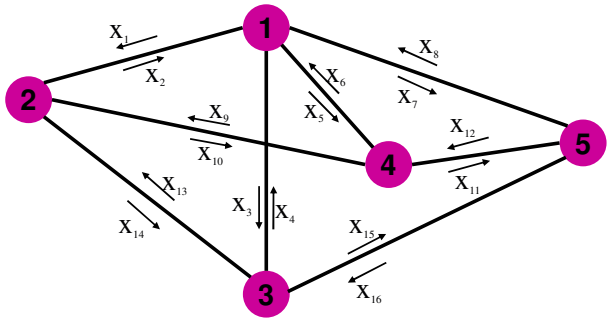

Fig. 1. 5-Node 16-link network

The propagation delay of each link is chosen from Normal distribution where the mean and variance were randomly selected once, prior to the first run.

In order to evaluate our results we compare them with two other schemes. In the first scheme, denoted by "H" for halving, 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. $c_{i,j} = \frac{\Delta T_{i,j}^{min} + \Delta T_{j,i}^{min}}{2}$. Note that the Halving scheme is based on the same concept as NTP of halving the minimum round trip delay attained on a link. 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. On the other hand when measuring delay on a link it is sufficient to require that the two nodes in both sides of the link will have their clock synchronized only with respect to each other hence we suggest an NTP-like scheme between the two adjacent nodes. The second compared scheme denoted by "MS", is the one suggested in [9]. This scheme is based on minimizing a mean square-like cost function. As previously explained this scheme suggests only a technique for estimating the propagation delays. The technique is based on cyclic path delay measurements assuming there are no queueing delays. When adding queueing delays to the model it is much more accurate to locate the packets that suffer no queueing delays on a per link basis and construct the cyclic paths later as explained in section IV-A, rather than look for the packets that experienced the minimum delay along cyclic paths.

Figure 2 shows the results of the 100 runs on a variety of selected links. The y axis on each graph presents the fraction of runs where the propagation delay offset between the estimated value and the real propagation delay is not greater than the value described by the x axis.

Figure 2 demonstrates significant improvement in terms of the delay estimation of the "ME" scheme over the other two schemes. For example in link 1 the estimated link propagation delay never exceeds 5.4 time units in 100 runs where for the other two schemes the maximum delay error is 9.2 and 11.0 for the MS and Halving respectively. Taking into account the scalability and implementation problems of the MS technique [9] makes the suggested scheme very attractive even when estimating only the propagation delay.

The Maximum Entropy scheme can be extended to handle a delay which is not constant. For example assuming that the minimum delay attained on each link ($\Delta T_{i,j}^{[min]}$) is varied over time and the distribution type is known, we want to estimate some of the relevant parameters. In such cases the Maximum Entropy can be used by computing each link propagation delay over time and then use common parameter estimation techniques [20].

We ran the same simulation as before, 100 times over the 5-node 16-link network where the delay on each link has Normal distribution, and we estimated the distribution based on the Mean and Variance. We compared the results with the halving technique and a modification of the MS technique suggested in [9] (originally [9] deals only with constant delays).

Figure 3 shows that the estimation obtained by the ME is much better than the other two schemes. We added to each graph the I-divergence distance [21] which measures the difference between estimated and real distribution. It is interesting to note the results of link $e_{5,6}$ which was forced to be symmetric, i.e. to have the

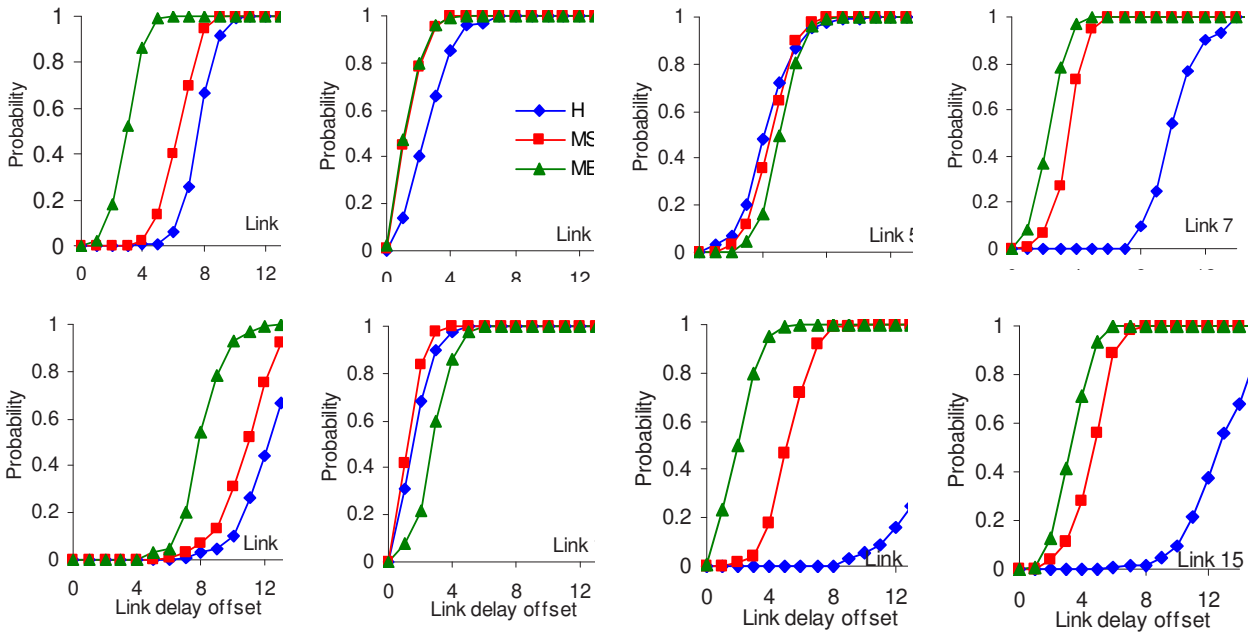

Fig. 2. The fraction of runs that the offset between the estimated link delay and the real link delay is not greater than t, for selected links in the 5-node 16-links network

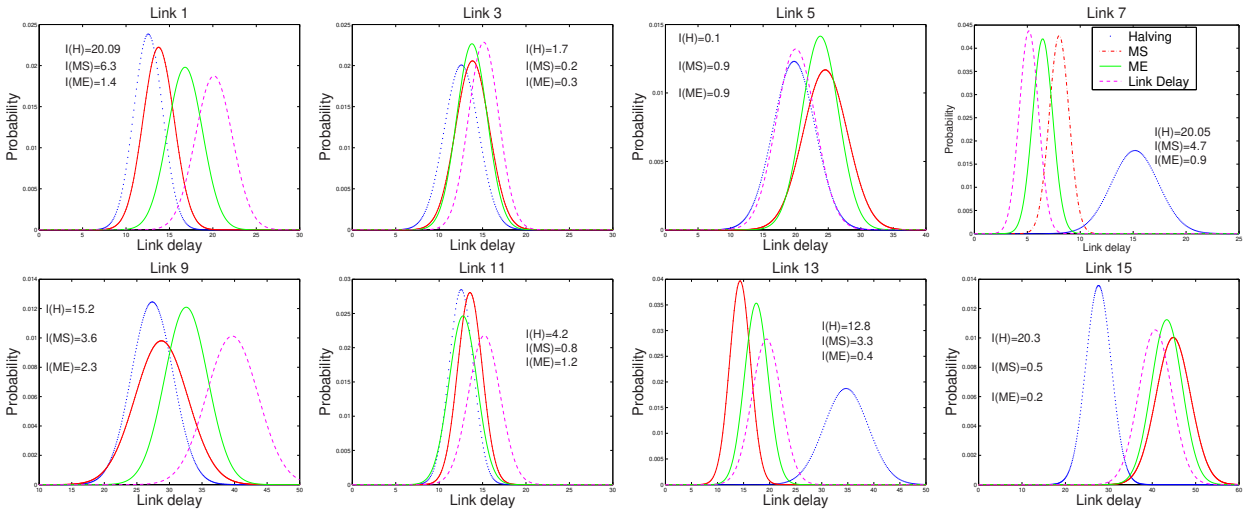

Fig. 3. Estimation of the Normal distribution propagation delay in the 5-node 16-link network.

same Mean and Variance as link $e_{6,5}$. The halving scheme is naturally the best for symmetric links however the difference from the ME is not very big (on link $e_{5,6}$ the I-divergence distance is 0.1 and 0.9 from the link delay to the estimation based on halving and ME respectively). On the other hand on links that are not symmetric the improvement of ME over the other two schemes is very significant (on link $e_{1,2}$ the I-divergence distance is 20.9 and 1.4 from the link delay halving and ME respectively).

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

In order to evaluate the total delay estimation attained using the suggested algorithm, we applied it on two networks of different sizes. The important parameters which characterize each entity in the network were randomly chosen. 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 mean that was randomly selected between 0.1 to 5 (uniformly). The clock offset with respect to the "Reference Time Node" is randomly chosen with a uniform distribution between -10 to 10

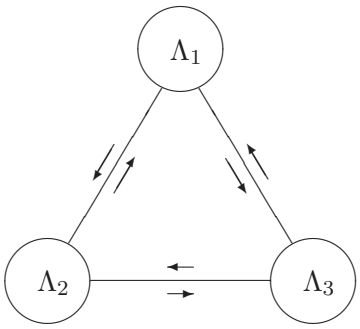

Fig. 4. 3-node fully connected network.

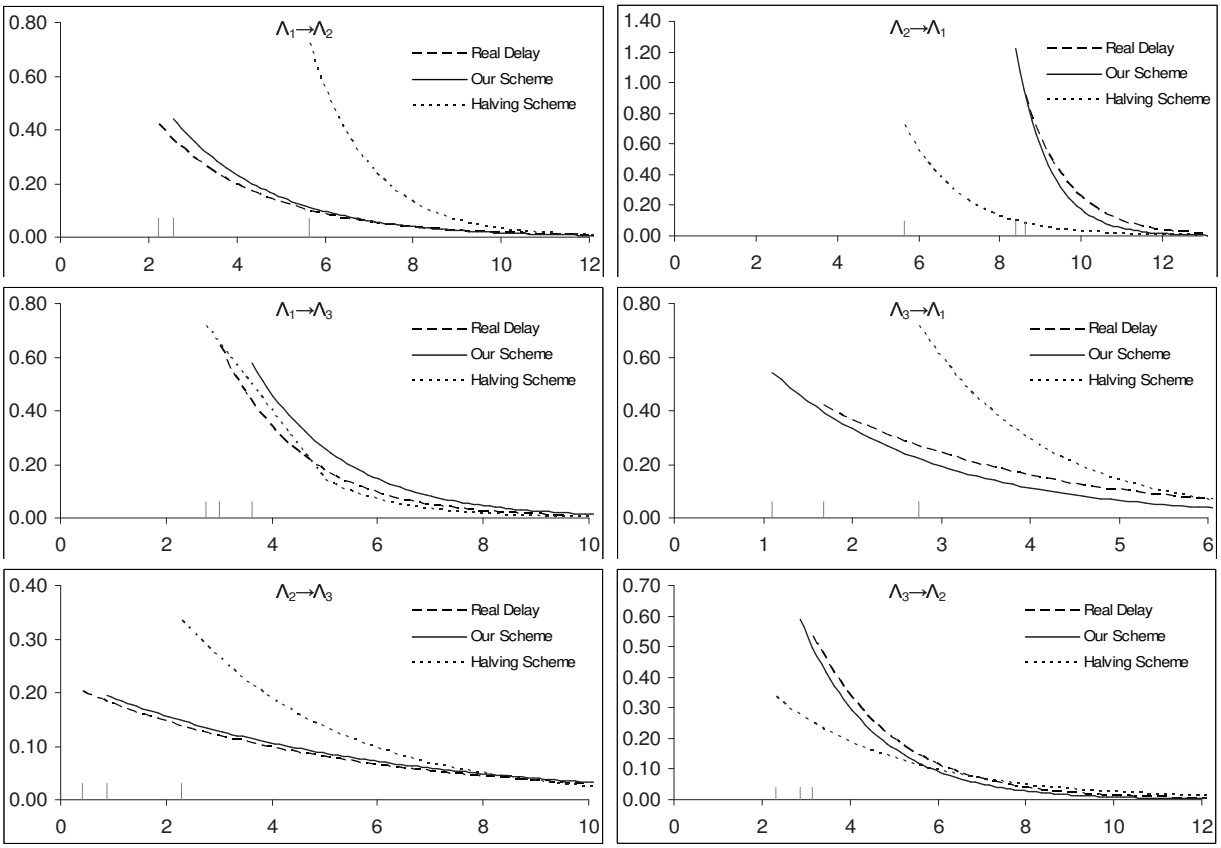

Fig. 5. Probability density function

($\sim U[-10, 10]$). For each link, thirty probe packets are transmitted and the estimation of both the constant delay and the variable delay is based on these packets. The results are based on thirty packets exchange on each link.

We start with the simple example of a three node network depicted in Figure 4. 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 obtained by any packet 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 thirty 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.

Figure 5 shows the total delay density function of the six one-way links obtained by the two schemes,

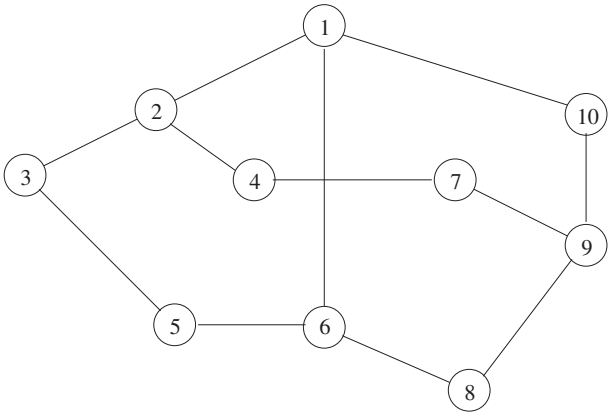

Fig. 6. 10-node 24-link connected network.

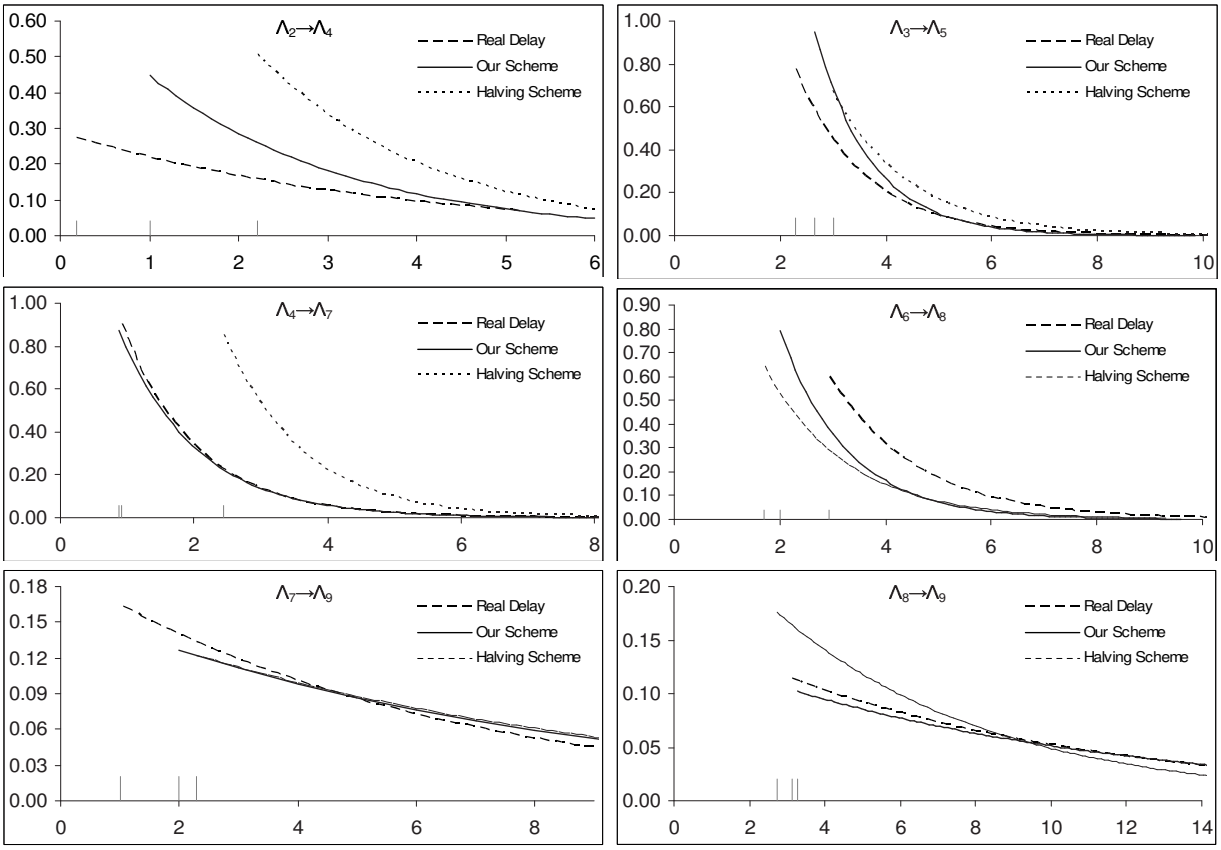

Fig. 7. Probability density function

the suggested scheme and the halving scheme, compared with 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. Figure 5 clearly depicts the 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 Figure 6. Figure 7 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.

VI. 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 approach for estimating the constant one-way delay is based on one-way measurements and exploiting the maximum entropy principle. The variable one-way delay estimation is based on focusing on the link between neighboring nodes.

The suggested schemes are easy to implement and can be incorporated in current Internet standards (e.g., using routine NTP messages). Numerical results show that our approach works well and substantially outperforms other known schemes, such as NTP based ones. Good delay estimation can be achieved even on a small group of nodes. This makes our approach attractive and applicable since there is no need to flood the network with measurement messages.

Appendix

I.

In this section we suggest an algorithm which computes $\frac{E}{2} - N + 1$ independent cyclic paths that are not single link round trips.

The algorithm suggested is iterative. It starts with a set of two connected nodes and in each iteration it adds another node to the set, a node which is connected to at least one node which is already in the previous iteration set. The number of cyclic paths which should be extracted for each additional node equals the number of bidirectional links which connect the node to the set minus one. For example if we start with a two node connected network and we add another node which is connected to both nodes we should extract $2 - 1 = 1$ cyclic path (in addition to the two round trip paths on both links which connect the node to the set). Let us denote by $\Lambda_i^{[k]}$ the node added to the set in the k-th iteration, by $E_i^{[k]}$ the number of bidirectional links connecting it to the set obtained in the $k-1$-th iteration, i.e. the number of neighbors in the set $[\Lambda_i^{[0]}, \Lambda_i^{[0]}, \Lambda_i^{[1]}, \ldots, \Lambda_i^{[k-1]}]$ and by $G_i^{[k]}$ the set of node $\Lambda_i^{[k]}$s neighbors which are already in the set prior to the k-th iteration. The cyclic paths we add in each iteration are extracted as follows: Pick one node from $G_i^{[k]}$ (a neighbor of node $\Lambda_i^{[k]}$ which is in the set). Denote this node by $P_i^{[k]}$ ($P_i^{[k]}$ can be selected arbitrarily). Extract the $E_i^{[k]} - 1$ cyclic paths that start from node $\Lambda_i^{[k]}$ traverse to node $P_i^{[k]}$ continue to one of the nodes in $G_i^{[k]}$ excluding $P_i^{[k]}$ ($G_i^{[k]} \setminus P_i^{[k]}$) via any non cyclic path which does not pass through any other node in $G_i^{[k]}$ (a minimum hop distance within the set can be such a path), and return to node $\Lambda_i^{[k]}$. Since node $\Lambda_i^{[k]}$ has $2E_i^{[k]}$ (E bidirectional) which connects it to the set, and the number of paths that can be extracted in the manner that was described above is $|G_i^{[k]}| - 1$ where $|G_i^{[k]}|$ denotes the number of nodes in group $|G_i^{[k]}|$, and since $E_i^{[k]} = |G_i^{[k]}|$ we have that $|G_i^{[k]}| - 1 = E_i^{[k]} - 1$ which is exactly the number of cyclic paths we need.

References

[1] P. Sharma, "Scalable session messages in SRM using self-configuration", Technical Report, January 1998, http://netweb.usc.edu/puneetsh/papers/ssm.ps

[2] S. Floyd, V. Jacobson, C.G. Liu, S. McCanne and L. Zhang, "A reliable multicast framework for light-weight sessions and application level framing", IEEE/ACM Transactions on Networking, Vol. 5, pp. 784-803, December 1997.

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

[4] D.L. Mills, "Internet time synchronization: the Network Time Protocol", IEEE Trans. Communications Vol. 39, pp. 1482-1493, October 1991.

[5] D.L. Mills, "Improved algorithms for synchronizing computer network clocks", IEEE/ACM Trans. Networking Vol. 3, pp. 245-254, June 1995.

[6] D.L. Mills, "Network Time Protocol (Version 3) specification, implementation and analysis", Network Working Group Report RFC-1305, University of Delaware, 1992.

[7] M. Tsuru, T. Takine and Y. Oie, "Estimation of clock offset from one-way delay measurement on asymmetric path", Symposium on Applications and the Internet (SAINT) Workshops Narar City, Nara, Japan, January 2002.

[8] O. Gurewitz, I. Cidon and M. Sidi, "Network Time Synchronization Using Clock Offset Optimization", ICNP, 2003.

[9] O. Gurewitz and M. Sidi, "Estimating One-way Delays from Cyclic-Path Delay Measurements", pp. 1038-1044, Infocom 2001,

[10] F. LoPresti, N.G. Duffield, J. Horowitz and D. Towsley, "Multicast -Based Inference of Network-Internal Delay Distributions", Tech. Rep. 99-55 UMass CMPSCI, 1999.

[11] N. Duffield and F. Lo Presti, "Multicast Inference of Packet Delay Variance at Interior Network Links", Infocom 2000.
[12] D.L. Mills, "Simple Network Time Protocol (SNTP) Version 4 for IPv4,IPv6 and OSI", Network Working Group Report RFC-2030, University of Delaware, 1996.
[13] C. Shannon, "A Mathematical Theory of Communication", Bell System Technical Journal, Vol. 27, pp. 398-403, 1948.
[14] E.T. Jaynes, "Information theory and statistical mechanics I", Physical Review, Vol. 106, pp 620-630, 1957.
[15] E.T. Jaynes, "On the rationale of maximum-entropy methods", Proc IEEE, Vol. 70, pp. 939-952, 1982.
[16] E.T. Jaynes, "Probability Theory: The Logic of Science", Preprint: Washington University, 1996 http://bayes.wustl.edu/etj/prob.html
[17] C.T. Kelley, "Iterative Methods for Linear and Nonlinear Equations", SIAM, Philadelphia, PA, 1995.
[18] S. Boyd, L. Vandenberghe, "Convex Optimization", 2002, www.stanford.edu/class/ee364 and www.ee.ucla.edu/ee236b
[19] N.L. Johnson and S. Kotz, "Continuous univariate distributions-1", John Wiley and Sons, New York, 1970.
[20] A. Papoulis, "Probability, random variables and stochastic processes", 3rd edition, McGraw-Hill, New York 1991
[21] D. Kazakos, P. Papantoni-Kazakos, "Detection and estimation", Computer Science Press, 1990.