Yik-Chung Wu, Qasim Chaudhari, and Erchin Serpedin

Clock Synchronization of Wireless Sensor Networks

Message exchange mechanisms and statistical signal processing techniques

Clock synchronization is a critical component in the operation of wireless sensor networks (WSNs), as it provides a common time frame to different nodes. It supports functions such as fusing voice and video data from different sensor nodes, time-based channel sharing, and coordinated sleep wake-up node scheduling mechanisms. Early studies on clock synchronization for WSNs mainly focused on protocol design. However, the clock synchronization problem is inherently related to parameter estimation, and, recently, studies on clock synchronization began to emerge by adopting a statistical signal processing framework. In this article, a survey on the latest advances in the field of clock synchronization of WSNs is provided by following a signal processing viewpoint. This article illustrates that many of the proposed clock synchronization protocols can be interpreted and their performance assessed using common statistical signal processing methods. It is also shown that advanced signal processing techniques enable the derivation of optimal clock synchronization algorithms under challenging scenarios.

Digital Object Identifier 10.1109/MSP.2010.938757
Date of publication: 17 December 2010

1053-5888/11/$26.00©2011IEEE

INTRODUCTION

With the help of recent technological advances in microelectromechanical systems and wireless communications, low-cost, low-power, and multifunctional wireless sensing devices have been developed. When these devices are deployed over a wide geographical region, they can collect information about the environment and efficiently collaborate to process such information, forming the so-called WSNs. WSNs are a special case of wireless ad hoc network and assume a multihop communication without a common infrastructure, where the sensors spontaneously cooperate to deliver information by forwarding packets from a source to a destination. The feasibility of WSNs keeps growing rapidly, and WSNs have been regarded as fundamental infrastructures for future ubiquitous communications due to a variety of promising potential applications: monitoring the health status of humans, animals, plants, and the environment; control and instrumentation of industrial machines and home appliances; homeland security; and detection of chemical and biological threats [1], [2].

DATA FUSION IS A BASIC OPERATION IN ALL DISTRIBUTED NETWORKS FOR PROCESSING AND INTEGRATING THE COLLECTED DATA IN A MEANINGFUL WAY.

Clock synchronization is a procedure for providing a common notion of time across a distributed system. It is crucial for WSNs in performing a number of fundamental operations:

- *Data Fusion*: Data fusion is a basic operation in all distributed networks for processing and integrating the collected data in a meaningful way. It requires some or all nodes in the network to share a common time scale.
- *Power Management*: Energy efficiency is a key designing factor for WSNs since sensors are usually left unattended without any maintenance and battery replacement service along their lifetimes. Most energy-saving operations strongly depend on time synchronization. For instance, the duty cycling (sleep and wake-up modes control) helps the nodes to save huge energy resources by spending minimal power during the sleep mode. Therefore, network-wide synchronization is essential for efficient duty cycling, and its performance is proportional to the synchronization accuracy.
- *Transmission Scheduling*: Many scheduling protocols require clock synchronization. For example, the time division multiple access scheme, one of the most popular communications schemes for distributed networks, is only applicable in a synchronized network.

Moreover, many localization, security, and tracking protocols also demand the sensor nodes to timestamp their messages and sensing events. Therefore, clock synchronization appears as one of the most important research challenges in the design of energy-efficient WSNs.

DEFINITION OF CLOCK

Every individual sensor in a network has its own clock. Ideally, the clock of a sensor node should be configured such that $C(t) = t$, where t stands for the ideal or reference time. However, because of the imperfections of the clock oscillator, a clock will drift away from the ideal time even if it is initially perfectly tuned. For example, according to the data sheet of a typical crystal-quartz oscillator commonly used in sensor networks, the frequency of a clock varies up to 40 ppm, which means clocks of different nodes can loose as much as 40 μs in a second (or 0.144 s in an hour). In general, the clock function of the ith node is modeled as

$$C_i(t) = \theta + f \cdot t, \qquad (1)$$

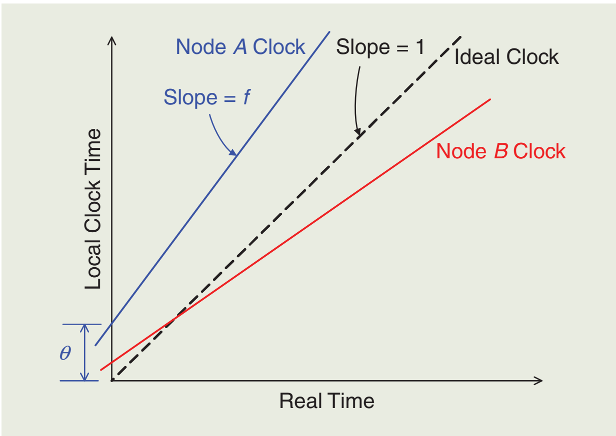
where the parameters θ and f are called clock offset (phase difference) and clock skew (frequency difference), respectively. A graphical representation of the clock model is illustrated in Figure 1.

From (1), the clock relationship between two nodes, Node A and Node B, can be represented by

$$C_B(t) = \theta^{AB} + f^{AB} \cdot C_A(t),$$

where θ^{AB} and f^{AB} stand for the relative clock offset and skew between Node A and Node B, respectively. Obviously, if two clocks are perfectly synchronized, $\theta^{AB} = 0$ and $f^{AB} = 1$. Otherwise, suppose Node A is the reference node, the task of clock synchronization is to estimate θ^{AB} and f^{AB} such that Node B can adjust its own clock or translate its timing information to the time scale of Node A when it is necessary. If there are L nodes in the network, then the global network-wide synchronization requires $C_i(t) = C_j(t)$ for all $i, j = 1, \cdots, L$, or all the relative clock offsets and skews are estimated with respect to a reference node.

In the long term, clock parameters are subject to changes due to environmental or other external effects such as temperature, atmospheric pressure, voltage changes, and hardware aging [3]. Hence, in general, the relative clock offset keeps changing with



[FIG1] Clock model of sensor nodes.

time, which means that the network has to perform periodic clock resynchronization to adjust the clock parameters.

THE CHALLENGE

Assume Node B needs to be synchronized to Node A. Node A sends its current time to Node B. If there is absolutely no delay in the message delivery, Node B can immediately know the difference between its clock and that of Node A. Unfortunately, in a real wireless network, various delays affect the message delivery, making clock synchronization much more difficult than it seems to be. In general, a series of timing message transmissions is required to estimate the relative clock skews and offsets among nodes. In some sense, clock synchronization in WSNs can be regarded as the process of removing the effects of random delays from the timing message transmissions sent across wireless channels.

The various delays present in a message delivery include the following components [4], [5]:

- *Send Time*: The time spent in building the message at the application layer, including delays introduced by the operating system when processing the send request.
- *Access Time*: The waiting time for accessing the channel after reaching the medium access control (MAC) layer. This is the most significant component and highly variable depending on the specific MAC protocol.
- *Transmission Time*: The time for transmitting a message at the physical (PHY) layer.
- *Propagation Time*: The actual time for a message to be transmitted from the sender to the receiver in a wireless channel.
- *Reception Time*: The time required for receiving a message at the PHY layer, which is the same as the transmission time.
- *Receive Time*: Time to construct and send the received message to the application layer at the receiver.

The delay components can also be categorized into two classes: deterministic (fixed portion) and stochastic (variable portion). The variable portion of delays depends on various network parameters (e.g., network status and traffic); therefore, no single delay model can be found to fit for every case. Probability density function (pdf) models that have been proposed for modeling random delays in wireless networks include Gaussian, exponential, Gamma, Weibull, and log-normal [6]–[8]. In the first half of this article, we focus on Gaussian and exponential delay models to illustrate the signal processing aspects in clock synchronization. The Gaussian model is justified if the delays are thought to be the addition of numerous independent random processes due to the central limit theorem. This is supported by [9], where the chi-square test showed that the variable portion of delays can be modeled as Gaussian distributed random variables (RVs) with 99.8% confidence. On the other hand, a single-server M/M/1 queue can fittingly represent the cumulative link delay for point-to-point hypothetical reference connections, where the random delays are independently modeled as exponential RVs [10]. The exponential delay model is also supported by experimental measurements [11], [12]. Toward the end of this article, the assumption on the distribution assumed by the network delays will be relaxed. Arbitrary network delay distributions will be assumed and the emphasis will be put toward developing clock offset estimation techniques that are robust to the distribution of network delays.

Another challenge that clock synchronization in WSNs faces is the limited and generally nonrechargeable power resources. Clock synchronization is one of the critical components contributing to energy consumption due to the highly energy consuming radio transmissions for delivering timing information. Pottie and Kaiser showed in [13] that the radio frequency energy required to transmit 1 kb more than 100 m (i.e., 3 J) is equivalent to the energy required to execute 3 million instructions. Therefore, developing efficient synchronization algorithms represents an ideal mechanism for trading computational energy for reduced communication overhead.

REMARK 1

If the time stamping occurs at the interface between the MAC- and PHY-layer, among the many sources of message delivery delay, the send, access, and receive times can be eliminated [5]. This procedure can dramatically reduce time-stamping errors at both the transmitter and receiver, and it is a strategy prescribed in most of the current standards, see e.g., IEEE 802.15.4.

REMARK 2

This article focuses on clock synchronization based on exchanging time stamps between sensor nodes. This approach is also referred to as packet coupling. This is in contrast to pulse-coupling techniques [14]–[16], which achieves synchronization by transmitting and processing PHY layer pulses directly.

FUNDAMENTAL APPROACHES TO CLOCK SYNCHRONIZATION

Clock synchronization in WSNs can be achieved by transferring a group of timing messages to the target sensors. The timing messages contain information about the time stamps measured by the transmitting sensors. There are three well-known timing message signaling approaches for clock synchronization in WSNs. These are the two-way message exchange (or sender–receiver synchronization), the one-way message dissemination, and the receiver–receiver synchronization.

TWO-WAY MESSAGE EXCHANGE

Two-way message exchange is a classical mechanism for exchanging timing information between two adjacent nodes. Examples of existing WSN clock synchronization protocols that employ this approach include timing-sync protocol for sensor networks (TPSNs) [17], tiny-sync and mini-sync [18], and lightweight time synchronization (LTS) [19]. Consider Node B as the reference node, where Node A needs to synchronize with Node B. The clock model for the two-way message exchange is depicted in Figure 2, where timing messages are assumed to be exchanged N times [17], [20]. In the kth round of message exchange, Node A sends a synchronization message to Node B at $T_{1,k}$. Node B records its time $T_{2,k}$ at the reception of that message and replies to Node A at $T_{3,k}$. The replied message contains both time stamps $T_{2,k}$ and $T_{3,k}$. Then, Node A records the reception time of Node B's reply as $T_{4,k}$. Note that $T_{1,k}$ and $T_{4,k}$ are the time stamps recorded by the clock of Node A, while $T_{2,k}$ and $T_{3,k}$ are the time stamps recorded by the clock of Node B. After N rounds of message exchanges, Node A obtains a set of time stamps $\{T_{1,k}, T_{2,k}, T_{3,k}, T_{4,k}\}_{k=1}^{N}$. The above procedure can be mathematically modeled as [21]

$$T_{2,k} = f(T_{1,k} + \tau + X_k) + \theta, \tag{2}$$
$$T_{3,k} = f(T_{4,k} - \tau - Y_k) + \theta, \tag{3}$$

where f and θ denote the relative clock skew and offset of Node A with respect to Node B, respectively, τ is the fixed delay, X_k and Y_k are the variable delays in the transmissions from Node A to Node B and from Node B to Node A, respectively.

In general, there are three parameters that have to be estimated: f, θ, and τ. Stacking all the time stamps from (2) and (3) in a matrix form, it follows that

$$\begin{bmatrix} T_{1,1} \\ \vdots \\ T_{1,N} \\ -T_{4,1} \\ \vdots \\ -T_{4,N} \end{bmatrix} + \tau \cdot \mathbf{1}_{2N} = \begin{bmatrix} T_{2,1} & -1 \\ \vdots & \vdots \\ T_{2,N} & -1 \\ -T_{3,1} & 1 \\ \vdots & \vdots \\ -T_{3,N} & 1 \end{bmatrix} \begin{bmatrix} 1/f \\ \theta/f \end{bmatrix} - \begin{bmatrix} X_1 \\ \vdots \\ X_N \\ Y_1 \\ \vdots \\ Y_N \end{bmatrix}, \tag{4}$$

where $\mathbf{1}_{2N}$ is the all-one column vector of dimension $2N \times 1$. Depending on whether the fixed delay τ is known or unknown, the maximum likelihood estimator (MLE) and the corresponding Cramer-Rao bound (CRB) for joint skew and offset estimation under Gaussian variable delays were derived in [21] and [22], respectively. Besides optimal MLEs, suboptimal but lower complexity algorithms were also reported in [21] and [22].

On the other hand, if there is only clock offset between the two nodes, i.e., $f = 1$, (2) and (3) can be simplified to

$$U_k = \tau + \theta + X_k\ , \tag{5}$$
$$V_k = \tau - \theta + Y_k\ , \tag{6}$$

where $U_k := T_{2,k} - T_{1,k}$ and $V_k := T_{4,k} - T_{3,k}$. Under the assumption that X_k and Y_k are independent and identically distributed zero mean Gaussian RVs, it can be shown that the MLE for θ is given by [21]

$$\hat{\theta} = \frac{1}{N}\sum_{k=1}^{N}(U_k - V_k) \tag{7}$$

and the value of τ does not affect the estimator. Interestingly, if only one round of message exchanges is performed (i.e., $N = 1$), the MLE of clock offset under the Gaussian delay model coincides with the clock offset estimator adopted in TPSN [17].

ONE-WAY MESSAGE DISSEMINATION

In the one-way message dissemination, a master node P broadcasts its timing information to many nodes, and these nodes record the arrival times of the broadcast message, as shown in Figure 3. The timing model of the kth broadcast message is the same as the first equation in the two-way message exchange, and it is given by

$$T_{2,k} = f(T_{1,k} + \tau + X_k) + \theta. \tag{8}$$

The corresponding equation when there is only clock offset is

$$T_{2,k} = T_{1,k} + \tau + \theta + X_k. \tag{9}$$

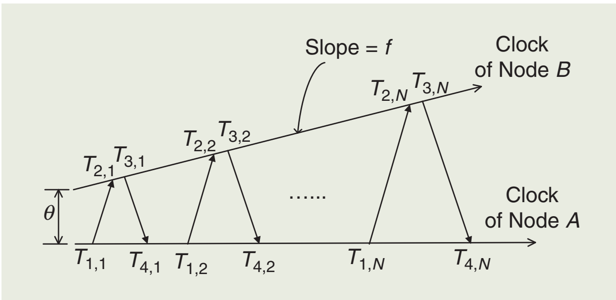
Notice that with only one-way message dissemination, the clock offset θ and the delay τ cannot be differentiated. However, assuming the fixed delay τ is negligible, and since the clock skew $f \approx 1$, (8) can be approximated by

$$T_{2,k} \approx f \cdot T_{1,k} + \theta + X_k. \tag{10}$$

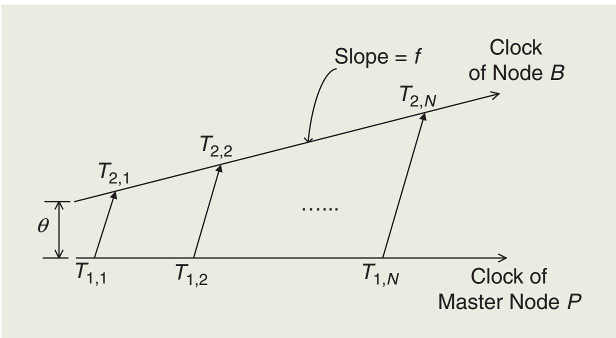
Collecting all the time-stamps and putting (10) into matrix form, the least squares (LS) estimate for f and θ can be obtained. This is the idea behind the flooding time synchronization protocol (FTSP) [5]. Reference [23] also adopts a similar approach, albeit it combines the one-way message dissemination scheme for clock skew estimation and the two-way message exchange scheme for offset estimation.

RECEIVER–RECEIVER SYNCHRONIZATION

Apart from synchronizing to the master node, two nodes that receive the same broadcast timing information can also be



[FIG2] Two-way timing message exchange between two nodes.



[FIG3] One-way message dissemination.

synchronized with each other directly. This is achieved by having them exchange their arrival time stamps with each other, as shown in Figure 4. Suppose the arrival time stamps of the common broadcast message at Node A and Node B are denoted by

$$T_{2,k}^{A} = f^{PA}(T_{1,k} + \tau^{A} + X_k^{A}) + \theta^{PA}, \tag{11}$$
$$T_{2,k}^{B} = f^{PB}(T_{1,k} + \tau^{B} + X_k^{B}) + \theta^{PB}, \tag{12}$$

where the superscripts A and B on $T_{2,k}$, τ, and X_k are used to distinguish the same quantity at two different nodes. Subtracting (12) from (11) leads to

$$T_{2,k}^{A} - T_{2,k}^{B} = f^{AB}T_{1,k} + \theta^{AB} + \underbrace{f^{PA}\tau^{A} - f^{PB}\tau^{B}}_{:=\tau'} + \underbrace{f^{PA}X_k^{A} - f^{PB}X_k^{B}}_{:=X_k'}, \tag{13}$$

where $f^{AB} := f^{PA} - f^{PB}$ and $\theta^{AB} := \theta^{PA} - \theta^{PB}$ are the relative clock skew and offset between Node A and Node B, respectively. Collecting the subtracted time stamps into a matrix form, it yields

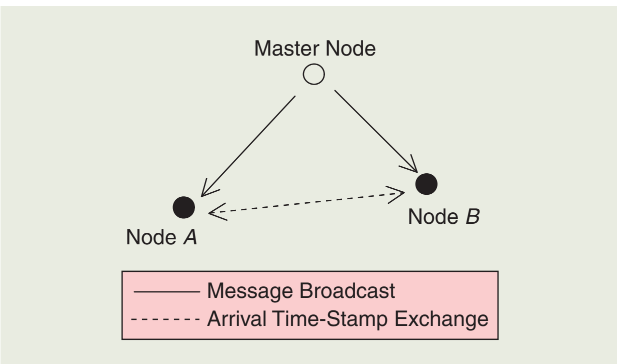
$$\begin{bmatrix} T_{2,1}^{A} - T_{2,1}^{B} \\ T_{2,2}^{A} - T_{2,2}^{B} \\ \vdots \\ T_{2,N}^{A} - T_{2,N}^{B} \end{bmatrix} - \tau' \cdot \mathbf{1}_N = \begin{bmatrix} T_{1,1} & 1 \\ T_{1,2} & 1 \\ \vdots & \vdots \\ T_{1,N} & 1 \end{bmatrix} \begin{bmatrix} f^{AB} \\ \theta^{AB} \end{bmatrix} - \begin{bmatrix} X_1' \\ X_2' \\ \vdots \\ X_N' \end{bmatrix}, \tag{14}$$

and the LS solution for f^{AB} and θ^{AB} can be developed.

As a special case, if there is no relative clock skew [i.e., $f^{AB} = 0$ in (13)] and assuming $\tau' \approx 0$, it is straightforward to show that the LS estimate of the relative clock offset $\hat{\theta}^{AB}$ is

$$\hat{\theta}^{AB} = \frac{1}{N}\sum_{k=1}^{N}\left[T_{2,k}^{A} - T_{2,k}^{B}\right], \tag{15}$$

which is equivalent to the reference broadcast synchronization (RBS) algorithm presented in [9]. Notice that, in this case, there is no need for the master node to embed its transmission timestamp $T_{1,k}$ in the broadcast message, as it does not appear in the estimator's expression.



[FIG4] Receiver-receiver synchronization.

SIGNAL PROCESSING TECHNIQUES FOR CLOCK SYNCHRONIZATION UNDER EXPONENTIAL DELAY

Previous discussions reveal that most of the existing fundamental synchronization protocols rely on standard statistical estimation techniques when the variable delay in message delivery is Gaussian distributed. It is known that when the random perturbation in observations is Gaussian distributed, the optimal parameter estimator is relatively easy to derive. Furthermore, the minimum variance unbiased estimator (MVUE), best linear unbiased estimator (BLUE), MLE, and LS estimator all coincide.

However, when the variable delay is not Gaussian, it takes more than intuition and basic signal processing techniques to derive the optimal clock synchronization algorithm. Next, we will illustrate the use of advanced signal processing techniques to derive clock parameter estimation algorithms when the delay is exponentially distributed. Later, we will further show how signal processing techniques help to tackle the more challenging situation, when the distribution of network delays is arbitrary.

MAXIMUM LIKELIHOOD ESTIMATION

The maximum likelihood method is overwhelmingly one of the most widely used approaches for parameter estimation. The particularly attractive features of the MLE are due to its asymptotic properties: it is unbiased and achieves the CRB at large enough numbers of data samples. Deriving the MLE of the clock offset in the exponential network delay model is a simple but very important first step, which was surprisingly accomplished only recently when the clock synchronization problem was addressed from a statistical signal processing viewpoint in [24] and [25].

Under the two-way message exchange mechanism and assuming symmetric exponential delays with common mean λ in the uplink and downlink, according to the signaling model depicted by (5) and (6), the likelihood function of (τ, θ, λ) can be expressed as

$$L(\tau,\theta,\lambda) = \lambda^{-2N} e^{-\frac{1}{\alpha}\sum_{k=1}^{N}(U_k + V_k - 2\tau)} \times I\left[U_{(1)} \geq \tau + \theta; V_{(1)} \geq \tau - \theta\right],$$

where $I[\cdot]$ represents the indicator function, achieving the value 1 when its argument is true and 0 otherwise, and $U_{(1)}$ and $V_{(1)}$ denote the minimum order statistics of $\{U_k\}_{k=1}^{N}$ and $\{V_k\}_{k=1}^{N}$, respectively. The likelihood function is maximized by making τ as large as possible, while having the constraints $\tau \leq U_{(1)} - \theta$ and $\tau \leq V_{(1)} + \theta$ satisfied. The support region of the constraints is shown in Figure 5, and it can be seen that the point corresponding to maximum τ is located at the intersection of the boundary lines represented by the indicator functions, which is the vertex M of the shaded triangle in Figure 5. Hence, the MLE of the vector parameter $\Theta_{\text{MLE}} = [\tau\ \theta\ \lambda]$ is given by [24]

$$\hat{\Theta}_{\text{MLE}} = \begin{bmatrix} \hat{\tau} \\ \hat{\theta} \\ \hat{\lambda} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} U_{(1)} + V_{(1)} \\ U_{(1)} - V_{(1)} \\ \overline{U} + \overline{V} - (U_{(1)} + V_{(1)}) \end{bmatrix}, \tag{16}$$

where $\overline{U}$ and $\overline{V}$ represent the sample averages of $\{U_k\}_{k=1}^{N}$ and $\{V_k\}_{k=1}^{N}$, respectively. It is found that the MLE of the clock offset

$\hat{\theta}_{ML}$ coincides with the minimum round delay estimator previously proposed by [12] through informal arguments. Interestingly, in the general case of asymmetric link delays, the MLE of clock offset $\hat{\theta}$ assumes the same expression.

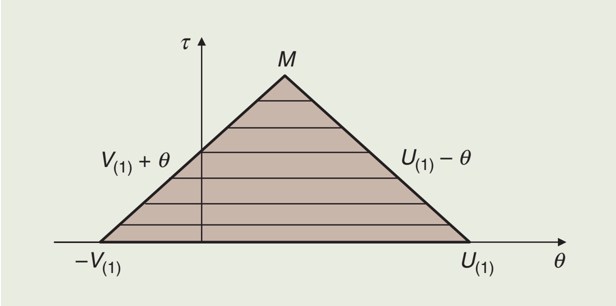
When the clock skew is also taken into account as in (2) and (3), there is no closed-form expression available for the joint estimates of clock parameters under exponential link delays. Instead, the optimization problem is solved by maximizing the likelihood function over nonlinear constraints through iterative methods and it has been addressed in [26] in the case of symmetric exponentially distributed delays. A more elegant method to find the MLE of the clock offset and skew was presented in [27], which utilizes the concept of profile likelihood whereby not only the five-dimensional optimization problem is reduced to a one-dimensional problem but also the general asymmetric delay case can be handled. Another way to get around the high-dimensional maximization of the likelihood function is to note that adding (2) and (3) eliminates the fixed delay. Then, the MLE for the clock offset and skew can be easily derived from the resultant equation [28]. On the other hand, under the RBS protocol, [29] derives the joint MLE for clock offset and skew under the exponential delay model, and the Gibbs sampler is proposed to maximize the likelihood function.

BEST LINEAR UNBIASED ESTIMATION USING ORDER STATISTICS

In many practical applications, it is difficult or impossible to find an optimal estimator due to various reasons. In such scenarios, a commonly applied methodology is to restrict the estimator to be linear in the data and find an unbiased linear estimator with minimum variance. This results in the BLUE.

It is known that when the observation noise is Gaussian distributed, BLUE provides the optimal solution by virtue of the Gauss-Markov theorem. For other distributions, including the exponential distribution, direct application of BLUE cannot guarantee any optimality. However, inspired by the results that MLE under exponentially distributed delays depends heavily on the order statistics of the observation data, [30] derived BLUE, which later turned out to possess certain optimality features.

Now assume the two-way message exchange model and the general set-up of exponential network delays for the uplink and downlink of possibly different means, denoted by α and β, respectively. Define



[FIG5] Support region for the likelihood function of clock offset estimation under exponential delays.

$$U'_k := \frac{1}{\alpha}(U_k - \tau - \theta),$$

$$V'_k := \frac{1}{\beta}(V_k - \tau + \theta)$$

as a set of independent observations on the standardized variate. Hence, their distribution will be parameter free. Furthermore, let $\{U'_k\}_{k=1}^N$ and $\{V'_k\}_{k=1}^N$ be the order statistics of $\{U'_k\}_{k=1}^N$ and $\{V'_k\}_{k=1}^N$, respectively. Using standard results for the exponential distribution [31, p. 500], the $N \times N$ symmetric positive-definite covariance matrix C for both $[U'_{(1)}\, U'_{(2)} \, \ldots \, U'_{(N)}]^T$ and $[V'_{(1)}\, V'_{(2)} \, \ldots \, V'_{(N)}]^T$ takes the common expression

$$\mathrm{C} = \begin{bmatrix} \frac{1}{N^2} & \frac{1}{N^2} & \cdots & \frac{1}{N^2} \\ \frac{1}{N^2} & \frac{1}{N^2}+\frac{1}{(N-1)^2} & \cdots & \frac{1}{N^2}+\frac{1}{(N-1)^2} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{1}{N^2} & \frac{1}{N^2}+\frac{1}{(N-1)^2} & \cdots & \sum_{k=1}^{N}\frac{1}{(N-k+1)^2} \end{bmatrix},$$

and the inverse C^{-1} could also be expressed in closed form.

Let $\Theta := [\tau\ \theta\ \alpha\ \beta]^T$ be the 4×1 vector of unknown parameters and $\mathbf{z} := [U_{(1)}U_{(2)}\ldots U_{(N)}V_{(1)}V_{(2)}\cdots V_{(N)}]^T$. Then, exploiting the following relations

$$E\left[U_{(k)}\right] = \tau + \theta + \alpha\, E\left[U'_{(k)}\right],\ E\left[V_{(k)}\right] = \tau - \theta + \beta E\left[V'_{(k)}\right],$$
$$\mathrm{var}\left[U_{(k)}\right] = \alpha^2 \mathrm{var}\left[U'_{(k)}\right],\ \mathrm{var}\left[V_{(k)}\right] = \beta^2 \mathrm{var}\left[V'_{(k)}\right],$$
$$\mathrm{cov}\left[U_{(k)}U_{(j)}\right] = \alpha^2 \mathrm{cov}\left[U'_{(k)}U'_{(j)}\right],\ \mathrm{cov}\left[V_{(k)}V_{(j)}\right] = \beta^2 \mathrm{cov}\left[V'_{(k)}V'_{(j)}\right],$$

the mean and covariance matrix of the ordered observations $\mathbf{z}$ is expressed in the equation at the bottom of the page.

$$E[\mathbf{z}] = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 & -1 & -1 & \cdots & -1 \\ \frac{1}{N} & \frac{1}{N}+\frac{1}{N-1} & \cdots & \sum_{k=1}^{N}\frac{1}{(N-k+1)} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \frac{1}{N} & \frac{1}{N}+\frac{1}{N-1} & \cdots & \sum_{k=1}^{N}\frac{1}{(N-k+1)} \end{bmatrix}^T \begin{bmatrix} \tau \\ \theta \\ \alpha \\ \beta \end{bmatrix}$$
$$:= \mathrm{Q}\Theta,$$
$$E\left[\mathbf{z}\mathbf{z}^H\right] := \mathrm{C}_z = \begin{bmatrix} \alpha^2\mathrm{C} & 0 \\ 0 & \beta^2\mathrm{C} \end{bmatrix}.$$

Consequently, the BLUE assumes the expression

$$\hat{\Theta} = (\mathbf{Q}^T\mathbf{C}_z^{-1}\mathbf{Q})^{-1}\mathbf{Q}^T\mathbf{C}_z^{-1}\mathbf{z} = \begin{bmatrix}\hat{\tau}\\ \hat{\theta}\\ \hat{\alpha}\\ \hat{\beta}\end{bmatrix} = \frac{1}{2(N-1)}\begin{bmatrix} N(U_{(1)}+V_{(1)})-(\overline{U}+\overline{V})\\ N(U_{(1)}-V_{(1)})-(\overline{U}-\overline{V})\\ 2N(\overline{U}-U_{(1)})\\ 2N(\overline{V}-V_{(1)})\end{bmatrix}. \quad (17)$$

In case of symmetric link delay (i.e., $\alpha = \beta$), it can be shown that the BLUE based on order statistics (BLUE-OS) for clock offset coincides with the $\hat{\theta}_{ML}$ in (16). Furthermore, the optimality of (17) will be revealed next in the discussion of MVUE.

MINIMUM VARIANCE UNBIASED ESTIMATION WITH RAO-BLACKWELL-LEHMANN-SCHEFFÉ THEOREM

In search of optimal estimators, mean square error (MSE) is often the chosen criterion. But from a practical viewpoint, the minimum MSE (MMSE) estimator is usually not realizable because of its dependence on the required parameter. Since the MSE is the sum of estimator variance and squared bias, and the dependence of the MMSE estimator on the unknown parameter typically arises from the bias, an alternative approach is to constrain the bias to be zero and find the estimator with minimum variance. Such an estimator is called the MVUE. Finding the MVUE necessitates the identification of the sufficient statistics and the application of the Rao-Blackwell-Lehmann-Scheffé theorem [32].

For the two-way message exchange mechanism under exponential network delays, the MVUE of clock synchronization parameters was derived in [33] by exploiting the following strategy. Considering the asymmetric case, the likelihood function for the clock offset as a function of observations $\{U_k\}_{k=1}^N$ and $\{V_k\}_{k=1}^N$ is expressed as

$$\begin{aligned} L(\tau,\theta,\alpha,\beta) = \alpha^{-N}\exp\left[-\frac{1}{\alpha}\sum_{k=1}^{N}\{U_k-\tau-\theta\}\right] \\ \cdot\beta^{-N}\exp\left[-\frac{1}{\beta}\sum_{k=1}^{N}\{V_k-\tau+\theta\}\right] \\ \cdot I\left[U_{(1)}-\tau-\theta\right]\cdots I\left[V_{(1)}-\tau+\theta\right], \quad (18)\end{aligned}$$

which can be factored as a product of the following functions

$$\begin{aligned} g_1\left(\sum_{k=1}^{N}U_{(k)},\tau,\theta,\alpha\right) &= \alpha^{-N}e^{-\frac{1}{\alpha}\sum_{k=1}^{N}\left(U_{(k)}-\tau-\theta\right)}, \\ g_2\left(\sum_{k=1}^{N}V_{(k)},\tau,\theta,\beta\right) &= \beta^{-N}e^{-\frac{1}{\beta}\sum_{k=1}^{N}\left(V_{(k)}-\tau+\theta\right)}, \\ g_3(U_{(1)},\tau,\theta) &= I\left[U_{(1)}-\tau-\theta\right], \\ g_4(V_{(1)},\tau,\theta) &= I\left[V_{(1)}-\tau+\theta\right], \\ h_1(U_k,V_k) &= 1.\end{aligned}$$

Note that $\mathbf{T} = \{\sum_{k=1}^N U_{(k)}, U_{(1)}, \sum_{k=1}^N V_{(k)}, V_{(1)}\}$ is a sufficient statistic for $\Theta = [\tau\ \theta\ \alpha\ \beta]^T$ because $h_1(U_k,V_k)$ is independent of Θ, whereas $g_1(\sum_{k=1}^N U_{(k)},\tau,\theta,\alpha)$, $g_2(\sum_{k=1}^N V_{(k)},\tau,\theta,\beta)$, $g_3(U_{(1)},\tau,\theta)$, and $g_4(V_{(1)},\tau,\theta)$ are functions depending on the data through $\mathbf{T}$.

On the other hand, the joint pdf of $U_{(1)}, U_{(2)}, \cdots, U_{(N)}$ is given by

$$p(U_{(1)},U_{(2)},\ldots,U_{(N)}) = N!\alpha^{-N}e^{-\frac{1}{\alpha}\sum_{k=1}^{N}\{U_{(k)}-\tau-\theta\}}\cdot\prod_{k=1}^{N}I\left[U_{(k)}-\tau-\theta\right], \quad (19)$$

whereas the pdf of the minimum order statistic $U_{(1)}$ is also exponential with a mean α/N. With the transformation $z_k = (N-k+1)(U_{(k)}-U_{(k-1)})$, $k = 1,\ldots,N$, and $U_{(0)} := \tau+\theta$, (19) can be equivalently expressed as [30]

$$p(z_1,z_2,\cdots,z_N) = \alpha^{-N}e^{-\frac{1}{\alpha}\sum_{k=1}^{N}z_k}\cdot\prod_{k=1}^{N}I[z_k],$$

i.e., z_k are independent exponential RVs with the same mean α. Also, since each $z_k \sim \exp(\alpha)$, each z_k assumes a Gamma distribution $z_k \sim \Gamma(1,\alpha)$ too. Using the relationship $\sum_{k=1}^N (U_{(k)}-U_{(1)}) = \sum_{k=2}^N z_k$, and the fact that each of $z_2, z_3, \cdots, z_N$ is independent of z_1 [and hence of $U_{(1)}$, since $z_1 = N(U_{(1)}-\tau-\theta)$], $\sum_{k=1}^N (U_{(k)}-U_{(1)}) \sim \Gamma(N-1,\alpha)$ and is independent of $U_{(1)}$.

Through a similar reasoning, it can be inferred that $\sum_{k=1}^N (V_{(k)}-V_{(1)}) \sim \Gamma(N-1,\beta)$ and is independent of $V_{(1)}$. Therefore, the one-to-one function $\mathbf{T}' = \{\sum_{k=1}^N (U_{(k)}-U_{(1)}), U_{(1)}, \sum_{k=1}^N (V_{(k)}-V_{(1)}), V_{(1)}\}$ of $\mathbf{T}$ is also sufficient for estimating Θ because the sufficient statistics are unique within one-to-one transformations [32]. Consequently, $\mathbf{T}'$ consists of four independent RVs that in terms of the three-parameter Gamma distribution assume the distributions

$$\begin{aligned} r &= \sum_{k=1}^{N}(U_{(k)}-U_{(1)}) \sim \Gamma(N-1,\alpha,0), \\ s &= \sum_{k=1}^{N}(V_{(k)}-V_{(1)}) \sim \Gamma(N-1,\beta,0), \\ & U_{(1)} \sim \Gamma(1,\alpha/N,\tau+\theta),\ V_{(1)} \sim \Gamma(1,\beta/N,\tau-\theta),\end{aligned}$$

respectively.

Finally, it is straightforward to prove that $\mathbf{T}'$, or equivalently $\mathbf{T}$, is complete and minimal [30]. Therefore, what remains is to find an unbiased estimator for Θ as a function of $\mathbf{T}$, which is also the MVUE according to Rao-Blackwell-Lehmann-Scheffé theorem. Since the BLUE-OS in (17) is such an unbiased estimator of Θ as a function of $\mathbf{T}$ and for this reason, it is the MVUE too.

MLE VERSUS MVUE IN MSE

From our previous discussion, it follows that, for asymmetric exponential delays in the uplink and downlink with different means, the MVUE is given by

$$\hat{\theta}_{\text{MVUE}} = \frac{N(U_{(1)} - V_{(1)}) - (\bar{U} - \bar{V})}{2(N-1)}, \tag{20}$$

while the MLE is $\hat{\theta}_{\text{MLE}} = (U_{(1)} - V_{(1)})/2$. One may wonder which estimator is better in terms of MSE? To answer this question, note that the MVUE is not necessarily the best estimator. It is only the best among unbiased estimators. If a biased estimator is devised having reduced variance relative to MVUE at the price of an insignificant increase in its squared bias, then the biased estimator might outperform the MVUE in the MSE sense.

TWO-WAY MESSAGE EXCHANGE IS A CLASSICAL MECHANISM FOR EXCHANGING TIMING INFORMATION BETWEEN TWO ADJACENT NODES.

For the considered modeling setup, the MSEs of the MVUE and MLE can be expressed in closed form, respectively, as

$$\text{MSE}(\hat{\theta}_{\text{MVUE}}) = \frac{1}{4N(N-1)}(\alpha^2 + \beta^2),$$
$$\text{MSE}(\hat{\theta}_{\text{MLE}}) = \frac{1}{2N^2}(\alpha^2 + \beta^2 - \alpha\beta).$$

The MLE performs better than MVUE in the MSE sense when $\text{MSE}(\hat{\theta}_{\text{MVUE}}) > \text{MSE}(\hat{\theta}_{\text{MLE}})$, or equivalently

$$\frac{N}{2} - 1 < \frac{\alpha\beta}{(\alpha - \beta)^2}.$$

From the above equation, it can be seen that the MLE is better than the MVUE when the means of the uplink and downlink delays are very close to each other. Otherwise, the MVUE is better. This observation is illustrated in Figure 6, in which $N = 15$, $\alpha = 2$, and β is varied across the interval $[\alpha - 2, \alpha + 2]$.

REMARK 3

Notice that most of the techniques in this section were proposed in the context of the two-way message exchange mechanism. Since after performing a mild approximation, the system of equations for the one-way message dissemination becomes linear [see (10)], the analysis of the one-way message dissemination framework under the exponential delay model generates similar results to those corresponding to the two-way message exchange mechanism. For receiver–receiver synchronization, only [29] derives the joint MLE for clock offset and skew. The derivation of estimators assuming other optimization criteria (e.g., MVUE and BLUE) in the exponential delay environment is an interesting research topic for future studies.

SIGNAL PROCESSING TECHNIQUES FOR DELAYS WITH ARBITRARY DISTRIBUTION

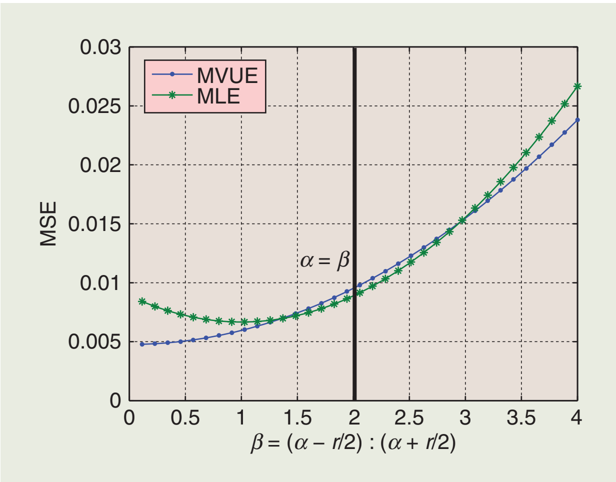
In synchronizing the clocks in a WSN, it might happen that the underlying pdf of the network delay model is not known in advance, and, hence, the performance of estimators specially designed for a particular distribution can vary a lot. Therefore, there is a need for developing statistical signal processing estimation techniques that are robust to the unknown network delay distributions or can adapt to different delay distributions. In this section, we consider three such statistical signal processing techniques.

LINEAR PROGRAMMING ESTIMATION

A linear programming (LP) problem is defined as the problem of maximizing or minimizing a linear function subject to linear equality or inequality constraints. For the one-way message dissemination scheme, note from (10) that if the link delays are coming from a nonnegative distribution, estimation of clock skew and offset can be cast as a linear program

$$\begin{aligned} &\text{minimize} \quad \sum_{k=1}^{N} (T_{2,k} - T_{1,k} f - \theta), \\ &\text{subject to} \quad \theta \le T_{2,k} - T_{1,k} f \qquad \forall \quad k = 1, 2, \ldots, N. \end{aligned}$$

The above linear program can be solved through many different techniques such as the simplex algorithm, ellipsoid method, or interior point methods. The solution to this linear program gives the ML estimate if the transmission delays are exponentially



[FIG6] The MSE of the MLE and MVUE under different α and β.

distributed. Even if the delay distribution is not exponential, it is quite logical to use a linear program to estimate the clock parameters, and, hence, this approach was elegantly put forward in [11]. LP was also employed to solve the clock synchronization problem in the context of wireless ad hoc networks based on the one-way message dissemination mechanism in [34].

THE MAXIMUM LIKELIHOOD METHOD IS OVERWHELMINGLY ONE OF THE MOST WIDELY USED APPROACHES FOR PARAMETER ESTIMATION.

On the other hand, clock synchronization under the two-way message exchange mechanism can also be cast into the LP problem [35]. From (2) and (3), we can write

$$\frac{1}{f}T_{2,k} - \frac{\theta}{f} - (\tau + X_k) = T_{1,k},$$
$$\frac{1}{f}T_{3,k} - \frac{\theta}{f} + (\tau + Y_k) = T_{4,k}.$$

Assuming Node B replies Node A immediately (see Figure 2) after receiving the timing message (i.e., $T_{2,k} = T_{3,k}$), and the delays τ, X_k, and Y_k are nonnegative, the above two equations can be represented in terms of the constraints: $T_{1,k} \le f T_{2,k} - \theta' \le T_{4,k}$, where $f' := 1/f$ and $\theta' := \theta/f$. It is proposed in [35] that the upper limit of the optimal set-valued estimate of f' is given by the following linear program

$$\max_{f',\theta'} \quad [f' \theta'] \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$
$$\text{subject to} \quad T_{1,k} \le f' T_{2,k} - \theta' \le T_{4,k} \quad \forall \quad k = 1, 2, \ldots, N, \tag{21}$$

and a lower limit estimate of f' is given by a similar linear program, but with maximization replaced by minimization. Suppose $(\overline{f'}, \hat{\theta}'_1)$ is the solution for (21) and $(\underline{f'}, \hat{\theta}'_2)$ is the solution for the same linear program but with minimization, [35] proves that $[\underline{f'}, \overline{f'}] \times [\underline{\theta'}, \overline{\theta'}]$ is a consistent set-valued estimate that minimizes the product $(\overline{f'} - \underline{f'})(\overline{\theta'} - \underline{\theta'})$ where $\overline{\theta'} := \max(\hat{\theta}'_1, \hat{\theta}'_2)$ and $\underline{\theta'} := \min(\hat{\theta}'_1, \hat{\theta}'_2)$.

BOOTSTRAP BIAS CORRECTION

Bootstrap is an approach for statistical inference based on building a sampling distribution for a statistic by resampling from the data at hand (see, e.g., [36] and [37]). For small sample sizes, such a method is usually superior to large sample techniques. On the downside, its computational complexity is considerably greater than the standard techniques described earlier in this article. We now discuss the application of bootstrap bias correction in the context of clock offset estimation.

Bootstrap bias correction typically reduces the bias of an estimator at the expense of increased variance but with an overall effect of reduced MSE. As explained in [38], suppose that an unknown probability distribution F assumes the data $\mathbf{x} = (x_1, x_2, \cdots, x_N)$ by random sampling. We want to estimate a real-valued parameter θ. For now, we will assume the estimator to be any statistic $\hat{\theta} = s(\mathbf{x})$. The bias of $\hat{\theta} = s(\mathbf{x})$ is defined to be the difference between the expectation of $\hat{\theta}$ and the value of the parameter θ, $B(\hat{\theta}) = E_F[s(\mathbf{x})] - \theta$. In practice, we may not know the distribution F or the true value of θ, so $B(\hat{\theta})$ cannot be computed. However, we can approximate the bias with the bootstrap estimate, which is defined as

$$\hat{B}(\hat{\theta}) = E_{\hat{F}}[s(\mathbf{x})] - \hat{\theta}, \tag{22}$$

where $\hat{F}$ is the empirical distribution constructed from $\mathbf{x}$. To compute $E_{\hat{F}}[s(\mathbf{x})]$, we generate independent bootstrap samples $\mathbf{x}^{*1}, \mathbf{x}^{*2}, \ldots, \mathbf{x}^{*M}$ from $\hat{F}$, evaluate the bootstrap replications $\hat{\theta}^*(m) = s(\mathbf{x}^{*m})$, and approximate the bootstrap expectation $E_{\hat{F}}[s(\mathbf{x}^*)]$ by the average

$$E_{\hat{F}}[s(\mathbf{x})] = \frac{1}{M}\sum_{m=1}^{M} s(\mathbf{x}^{*m}).$$

Therefore, the bias-corrected estimator is

$$\hat{\theta}_{BC} = \hat{\theta} - \hat{B}(\hat{\theta}) = 2\hat{\theta} - \frac{1}{M}\sum_{m=1}^{M} s(\mathbf{x}^{*m}).$$

In the context of clock synchronization, two sensor nodes exchange timing packets to obtain the data sets $\{U_k\}_{k=1}^N$ and $\{V_k\}_{k=1}^N$, as defined in (5) and (6), and suppose the estimator under consideration is $\hat{\theta} = s(\{U_k\}_{k=1}^N, \{V_k\}_{k=1}^N)$. For the nonparametric bootstrap method, the empirical distributions of $\{U_k\}_{k=1}^N$ and $\{V_k\}_{k=1}^N$, denoted by $\hat{F}$ and $\hat{G}$, are constructed. From $\hat{F}$ and $\hat{G}$, the samples $\{U_1^*, U_2^*, \cdots, U_N^*\}$ and $\{V_1^*, V_2^*, \cdots, V_N^*\}$, called the bootstrap resamples, are redrawn. Then, the distribution of $\hat{\theta}$ is approximated by the empirical distribution of $\hat{\theta}^* = s(\{U_k^*\}_{k=1}^N, \{V_k^*\}_{k=1}^N)$ derived from the bootstrap resamples.

In case when some partial information about the true distributions of $\{U_k\}_{k=1}^N$ and $\{V_k\}_{k=1}^N$, denoted by F and G, is available, the parametric bootstrap technique can be applied. For example, if F and G are known to obey a particular distribution but with unknown mean μ, we should draw resamples from that distribution with mean $\hat{\mu}$, where $\hat{\mu}$ is estimated from the samples $\{U_k\}_{k=1}^N$ and $\{V_k\}_{k=1}^N$.

It must be emphasized that not all bootstrap bias-corrected estimators have to be evaluated via resampling methods. It is interesting to observe that [39] derives a closed-form expression for the bootstrap bias corrected estimator for clock offset in the two-way message exchange scenario. From (5) and (6), the marginal distributions of U_k and V_k are defined as $F(u) := F(u - \theta - \tau)$ and $G(v) := G(v + \theta - \tau)$, respectively, and it is assumed that $F(u)$ and $G(v)$ are nonnegative such that $U_k \ge \theta + \tau$ and $V_k \ge -\theta + \tau$ hold. Moreover, their joint distribution is $H(u, v) = F(u)G(v)$ due to the independence of the transmission delays in both directions. The nonparametric estimator of $H(u,v)$ is $\hat{H}(u,v)=\hat{F}(u)\hat{G}(v)$, where $\hat{F}(u)$ and $\hat{G}(v)$ are the empirical probability distributions based on the observations $\{U_k\}_{k=1}^{N}$ and $\{V_k\}_{k=1}^{N}$, respectively.

Assume that the bias of the MLE $\hat{\theta}_{ML}=(U_{(1)}-V_{(1)})/2$ when applied to unknown distributions F and G is of interest

$$B(\hat{\theta}_{\mathrm{ML}})=\frac{1}{2}E_H(U_{(1)}-V_{(1)})-\theta$$
$$=\frac{1}{2}\left(\int_0^\infty[1-F(u)]^N du-\int_0^\infty[1-G(v)]^N dv\right)-\theta.$$

The bootstrap estimate of this bias is

$$\hat{B}(\hat{\theta}_{\mathrm{ML}})=\frac{1}{2}\left(\int_0^\infty\left[1-\hat{F}(u)\right]^N du-\int_0^\infty\left[1-\hat{G}(v)\right]^N dv\right)-\hat{\theta}_{\mathrm{ML}}.$$

Now defining $U_{(0)}=V_{(0)}=0$ and $U_{(N+1)}=V_{(N+1)}=\infty$, we can write

$$1-\hat{F}(u)=\sum_{k=1}^{N+1}\frac{N-k+1}{N}I\left[U_{(k-1)}\le u\le U_{(k)}\right],$$
$$1-\hat{G}(v)=\sum_{k=1}^{N+1}\frac{N-k+1}{N}I\left[V_{(k-1)}\le v\le V_{(k)}\right].$$

From the above three equations, it can be shown that

$$\hat{B}(\hat{\theta}_{\mathrm{ML}})=\frac{1}{2}\sum_{k=1}^{N}\left\{\left(\frac{N-k+1}{N}\right)^N-\left(\frac{N-k}{N}\right)^N\right\}\times(U_{(k)}-V_{(k)})-\hat{\theta}_{\mathrm{ML}}.$$

Finally, a bias-corrected estimator can be expressed as

$$\hat{\theta}_{BC}=\hat{\theta}_{\mathrm{ML}}-\hat{B}(\hat{\theta}_{\mathrm{ML}})$$
$$=U_{(1)}-V_{(1)}-\frac{1}{2}\sum_{k=1}^{N}\left\{\left(\frac{N-k+1}{N}\right)^N-\left(\frac{N-k}{N}\right)^N\right\}(U_{(k)}-V_{(k)}).$$

In general, the MSE performance of bootstrap-bias corrected estimate is better than the exponential MLE $\hat{\theta}_{\mathrm{ML}}$ when they are applied to nonexponential delays [39]. Figure 7 compares the performance of the MLE of clock offset derived under exponential delay and its corresponding bootstrap bias-corrected estimator when applied to Gamma distributed delays with two degrees of freedom and the means for uplink and downlink are one and five, respectively. It can be seen that the bootstrap estimator performs consistently better than the original MLE. Notice that for clock offset estimation only, the LP approach will generate an estimator identical to the MLE, so its performance is not presented here.

EXTENSION TO NETWORK-WIDE SYNCHRONIZATION CAN BE DIRECTLY ACHIEVED BY BUILDING A HIERARCHICAL STRUCTURE (SPANNING TREE) AND PAIRWISE SYNCHRONIZATION IS PERFORMED BETWEEN ADJACENT LEVELS.

COMPOSITE PARTICLE FILTERING

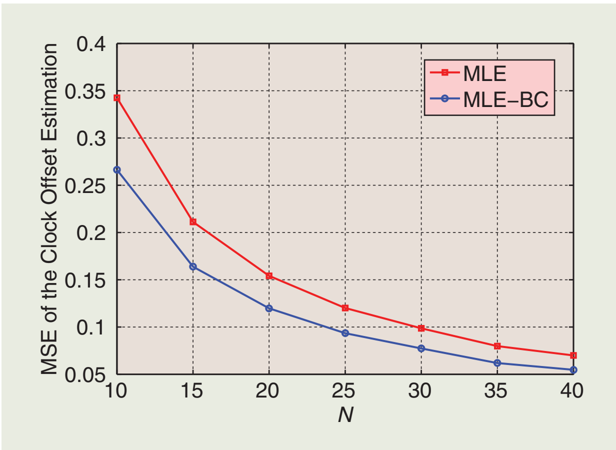
We now turn our attention toward a robust approach, which works extremely well in networks where the delay distributions are non-Gaussian and nonexponential, and even time-varying. The idea is to model the clock estimation problem in state-space form and make use of the optimal Bayesian framework for state estimation. First, notice that, from (5) and (6), we can write

$$\underbrace{\begin{bmatrix}U_k\\V_k\end{bmatrix}}_{:=\mathrm{y}_k}=\underbrace{\begin{bmatrix}1&1\\1&-1\end{bmatrix}}_{:=\mathrm{B}}\underbrace{\begin{bmatrix}\tau\\\theta\end{bmatrix}}_{:=\mathrm{x}_k}+\underbrace{\begin{bmatrix}X_k\\Y_k\end{bmatrix}}_{:=\mathrm{n}_k},\qquad(23)$$

where n_k can assume any distribution. Since x_k is fixed or varying only very slowly, the unknown state can be modeled as obeying a Gauss-Markov dynamic model of the form

$$\mathrm{x}_k=\mathrm{x}_{k-1}+\mathrm{v}_{k-1}\,,\qquad(24)$$

where the additive process noise component v_{k-1} can be modeled as Gaussian with zero mean and covariance matrix $E[\mathrm{v}_{k-1}\mathrm{v}_{k-1}^T]=\mathrm{Q}_{k-1}$. Now (23) and (24) form the state-space model. The objective is to derive the MMSE estimator of the unknown state x_k, which is the conditional mean state estimator $\hat{\mathrm{x}}_k=E\{\mathrm{x}_k|\mathrm{y}_{1:k}\}$, where $\mathrm{y}_{1:k}=[\mathrm{y}_1\mathrm{y}_2\cdots\mathrm{y}_k]^T$ denotes the set of observed samples up to time k [41].



[FIG7] MSE performance of MLE derived for exponential delay and its bootstrap bias-corrected version applied to Gamma delay distribution.

The optimal state estimation consists of two main steps: time update and measurement update. Suppose at time $k-1$, the posterior distribution $p(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1})$ is known, the time update step resumes to obtain the predictive distribution

CENTRALIZED SIGNAL PROCESSING TECHNIQUES CAN ONLY HELP SOLVING THE PROBLEM OF NODE-TO-NODE SYNCHRONIZATION AND POSSIBLE AD HOC EXTENSIONS TO NETWORK-WIDE SYNCHRONIZATION.

$$p(\mathrm{x}_k|\mathrm{y}_{1:k-1}) = \int p(\mathrm{x}_k|\mathrm{x}_{k-1})p(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1})d\mathrm{x}_{k-1}, \quad (25)$$

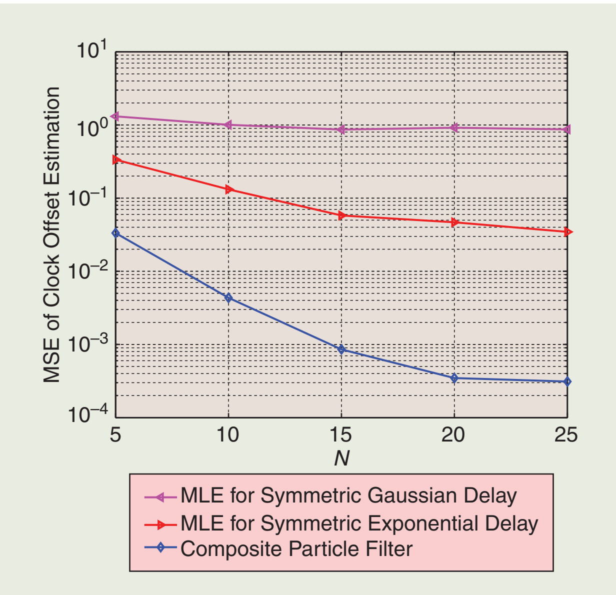
where the transition density $p(\mathrm{x}_k|\mathrm{x}_{k-1})$ is related to the state transition equation (24). On the other hand, in the measurement update step, based on the new observation y_k and the predictive distribution $p(\mathrm{x}_k|\mathrm{y}_{1:k-1})$, the marginal posterior distribution of state at time k is obtained as

$$p(\mathrm{x}_k|\mathrm{y}_{1:k}) = C_k p(\mathrm{x}_k|\mathrm{y}_{1:k-1})p(\mathrm{y}_k|\mathrm{x}_k)\ , \quad (26)$$

where

$$C_k = \left(\int p(\mathrm{x}_k|\mathrm{y}_{1:k-1})p(\mathrm{y}_k|\mathrm{x}_k)d\mathrm{x}_k\right)^{-1} \quad (27)$$

is a normalization constant and $p(\mathrm{y}_k|\mathrm{x}_k)$ is the likelihood of the observation y_k given x_k and is related to the observation equation (23). In general, when the model is linear with Gaussian noise and the prior knowledge about the initial state x_0 is Gaussian, the Kalman filter provides the mean and covariance update sequentially and is the optimal Bayesian solution. If the noise is not Gaussian, there may not be closed-form expression to (25) and (26), and particle filtering [40], in which a set of particles with weights are used to approximate the shape of the distribution, becomes an attractive alternative to the closed-form solution.



[FIG8] MSE performance of MLEs derived for Gaussian and exponential delays, and the composite particle filter, applied to Gamma delay distribution.

In the composite particle filtering, the predictive and posterior distributions are modeled by Gaussian mixture models [40], and each component is updated using the Kalman filter or particle filter. This solution presents a smaller computational complexity than a pure particle filter solution, as the procedure called resampling is avoided [40]. More specifically, let the posterior distribution at time $k-1$ assume the following decomposition

$$p(\mathrm{x}_{k-1}|\mathrm{y}_{1:k-1}) \approx \sum_{g=1}^{G} w_{(k-1)g}\mathcal{N}(\mathrm{x}_{k-1};\boldsymbol{\mu}_{(k-1)g}, \mathrm{P}_{(k-1)g}), \quad (28)$$

where G is the number of mixing components, $w_{(k-1)g}$ is the mixing weight, and $\mathcal{N}(\mathrm{x};\boldsymbol{\mu},\mathrm{P})$ denotes the Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance P. Plugging (28) into the time update step (25), the predictive distribution takes the form

$$p(\mathrm{x}_k|\mathrm{y}_{1:k-1}) = \sum_{g=1}^{G} w_{(k-1)g}\int p(\mathrm{x}_k|\mathrm{x}_{k-1})\mathcal{N}(\mathrm{x}_{k-1};\boldsymbol{\mu}_{(k-1)g}, \mathrm{P}_{(k-1)g})d\mathrm{x}_{k-1}. \quad (29)$$

Since the state transition (24) is linear and the noise v_{k-1} is Gaussian distributed, the predictive distribution can be obtained by a bank of G Kalman filters and the result is [40]

$$p(\mathrm{x}_k|\mathrm{y}_{1:k-1}) \approx \sum_{g=1}^{G} \bar{w}_{kg}\mathcal{N}(\mathrm{x}_k;\bar{\boldsymbol{\mu}}_{kg}, \bar{\mathrm{P}}_{kg}), \quad (30)$$

where $\bar{w}_{kg} = w_{(k-1)g}$, $\bar{\boldsymbol{\mu}}_{kg} = \boldsymbol{\mu}_{(k-1)g}$ and $\bar{\mathrm{P}}_{kg} = \mathrm{P}_{(k-1)g} + \mathrm{Q}_{k-1}$.

After obtaining the predictive distribution and plugging it into the measurement update equation (26), it follows that

$$p(\mathrm{x}_k|\mathrm{y}_{1:k}) = C_k\sum_{g=1}^{G} \bar{w}_{kg}p(\mathrm{y}_k|\mathrm{x}_k)\mathcal{N}(\mathrm{x}_k;\bar{\boldsymbol{\mu}}_{kg}, \bar{\mathrm{P}}_{kg}). \quad (31)$$

Sine $p(\mathrm{x}_k|\mathrm{y}_{1:k})$ is eventually approximated by a mixture of Gaussian pdfs, each term on the right-hand side of (31) given by $p(\mathrm{y}_k|\mathrm{x}_k)\mathcal{N}(\mathrm{x}_k;\bar{\boldsymbol{\mu}}_{kg}, \bar{\mathrm{P}}_{kg})$ is approximated with a Gaussian with mean and covariance calculated using samples obtained from importance sampling. Suppose the J samples generated from each $p(\mathrm{y}_k|\mathrm{x}_k)\mathcal{N}(\mathrm{x}_k;\bar{\boldsymbol{\mu}}_{kg}, \bar{\mathrm{P}}_{kg})$ are $\mathrm{x}_{kg}^{(j)}$ with the corresponding weighting factors $\gamma_{kg}^{(j)}(j = 1, \ldots, J)$. Then, the updated mean and covariance of each of the G components are given by

$$\boldsymbol{\mu}_{kg} = \sum_{j=1}^{J}\gamma_{kg}^{(j)}\mathrm{x}_{kg}^{(j)} \Big/ \sum_{j=1}^{J}\gamma_{kg}^{(j)}\ , \quad (32)$$

$$\mathrm{P}_{kg} = \sum_{j=1}^{J}\gamma_{kg}^{(j)}(\mathrm{x}_{kg}^{(j)} - \boldsymbol{\mu}_{kg})(\mathrm{x}_{kg}^{(j)} - \boldsymbol{\mu}_{kg})^T \Big/ \sum_{j=1}^{J}\gamma_{kg}^{(j)}. \quad (33)$$

Then, the posterior distribution is given by

$$p(\mathrm{x}_k|\mathrm{y}_{1:k}) \approx \sum_{g=1}^{G} w_{kg}\mathcal{N}(\mathrm{x}_k; \boldsymbol{\mu}_{kg}, \mathrm{P}_{kg}) \;, \tag{34}$$

where $w_{kg} = \tilde{w}_{kg} / \sum_{g=1}^{G} \tilde{w}_{kg}$ and $\tilde{w}_{kg} = \bar{w}_{kg} \sum_{j=1}^{J} \gamma_{kg}^{(j)} / (\sum_{g=1}^{G} \sum_{j=1}^{J} \gamma_{kg}^{(j)})$.

Finally, the conditional mean state estimate and the corresponding error covariance are calculated as follows:

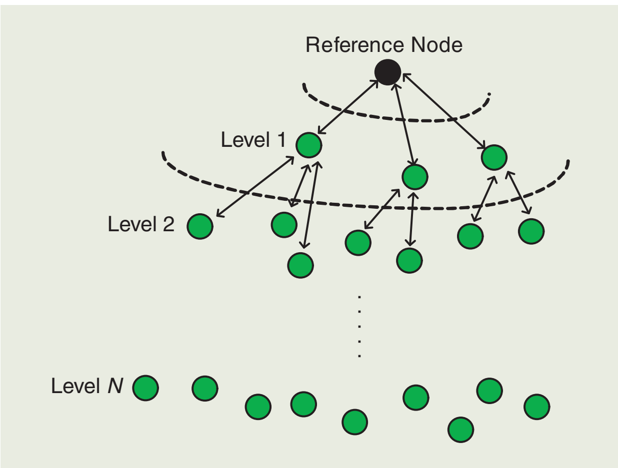
$$\hat{\mathrm{x}}_k = \sum_{g=1}^{G} w_{kg}\boldsymbol{\mu}_{kg}, \qquad \hat{\mathrm{P}}_k = \sum_{g=1}^{G} w_{kg}(\mathrm{P}_{kg} + (\hat{\mathrm{x}}_k - \boldsymbol{\mu}_{kg})(\hat{\mathrm{x}}_k - \boldsymbol{\mu}_{kg})^T].$$

It should be noted that the composite particle filtering approach allows tracking of time-varying clock offset, which represents a more realistic model than a constant phase.

Figure 8 compares the performance of the MLEs of clock offset derived under symmetric Gaussian and exponential delays and the composite particle filter. The message delivery delay is Gamma distributed with two degrees of freedom, and the means for uplink and downlink are two and one, respectively. For the composite particle filter, $\mathrm{Q} = 10^{-4}\mathrm{I}$, the number of particles and Gaussian mixture model components are 100 and three, respectively. It can be seen that the composite particle filter performs much better than the MLEs derived under the symmetric Gaussian or exponential delay assumption. This comes at the expense of increased computational complexity and knowledge of the observation noise.

FROM PAIRWISE SYNCHRONIZATION TO NETWORK-WIDE SYNCHRONIZATION

All of the techniques discussed so far focus on synchronization between a pair of neighboring nodes. Extension to network-wide synchronization can be directly achieved by building a hierarchical structure (spanning tree) and pairwise synchronization is performed between adjacent levels. Representative clock synchronization protocols that employ such approach are TPSN [17], LTS [19], and FTSP [5].



[FIG9] Level-by-level clock synchronization.

Figure 9 illustrates this approach. One (or more) node with accurate time is elected as the reference node, and a spanning tree is built with the reference node as the root. Clock synchronization is then carried out through the spanning tree from the root to leaves, one level at a time. Since the synchronization error accumulates along the tree, the accuracy of each pairwise synchronization is important. However, statistical signal processing techniques can help to design optimal clock synchronization algorithms and to mitigate error accumulation. Furthermore, spatial averaging in each layer is proposed in [16] to improve synchronization performance in large-scale networks. On the other hand, by exploiting MSE analysis and performance bounds (such as CRB), the synchronization accuracy at any node in the network can be predicted and used to determine how many data samples are necessary to achieve a certain synchronization accuracy. This might be an important feature for applications that require tight synchronization accuracy as is the case with localization and tracking of targets.

Instead of building a tree structure and constraining each node to communicate with one parent only, [42] proposes to model the whole network as a directed graph with $n+1$ nodes, where each edge represents a pair of nodes that can communicate with each other. The main idea of the global synchronization algorithm in [42] is based on the observation that for each closed loop $\mathcal{L}$ in the network, the relative clock offset must sum to zero

$$\sum_{i,j \in \mathcal{L}} \theta^{ij} = 0. \tag{35}$$

With these constraints, the relative offsets θ^{ij} can be translated into absolute nodal offset ξ_j with respect to a reference node using

$$\boldsymbol{\theta} = \mathrm{A}^T \boldsymbol{\xi}, \tag{36}$$

where $\boldsymbol{\theta}$ is a vector containing all the relative offsets θ^{ij}, A stands for the incidence matrix containing the connection information between different nodes, and $\boldsymbol{\xi} = [\xi_0, \xi_1, \ldots, \xi_n]^T$. Therefore, if an estimate of relative offset θ^{ij} for each pair of directly connected nodes i and j is obtained using pairwise synchronization, the absolute nodal offset vector can be determined in the LS sense. This technique is known as LS spatial smoothing. A distributed algorithm based on coordinate descent has also been developed in [42] such that only local connectivity information is required, and it takes the expression

$$\xi_i = \frac{1}{D_i} \sum_{j \leftrightarrow i} (\xi_j + \hat{\theta}^{ji}) \;, \tag{37}$$

where D_i represents the number of direct neighbors of Node i, and $\sum_{j \leftrightarrow i}$ stands for the summation with respect to all the nodes that can directly communicate with Node i. The above iterative solution admits a very simple interpretation: each node computes its update by averaging all its neighbors' absolute nodal offset estimates and relative offset estimates. Error variance and convergence analysis of this approach are reported in [43]. Incidentally, this distributed spatial smoothing algorithm coincides in spirit with the diffusion algorithms proposed in [44]. Recently, a distributed synchronization technique based on coupled phase-locked loops (PLLs) [15] also assumes the interpretation of spatial averaging. Furthermore, in the terminology of PLLs, the clock synchronization algorithms proposed herein could be interpreted as the optimal way of deriving the time error detector.

In addition to building a tree and spatial smoothing, more recently, a new approach called pairwise broadcast synchronization (PBS) was introduced in [45]. This approach allows a sensor to synchronize itself by overhearing timing messages from a neighboring two-way message exchange without sending out any packet itself. In a one-hop sensor network where every node is a neighbor of each other, a single PBS message exchange between two nodes would facilitate all nodes to synchronize, thus significantly reducing the communication overhead for achieving clock synchronization. Further extensions of PBS to multihop scenarios were also discussed in [46] and [47].

ANOTHER APPROACH TO ACHIEVE GLOBAL CLOCK SYNCHRONIZATION IS TO ESTIMATE THE CLOCK SYNCHRONIZATION PARAMETERS OF THE WHOLE NETWORK DIRECTLY FROM THE TIME STAMPS AND AT THE SAME TIME.

On the other hand, in RBS [9], the concept of gateway node is used to extend adjacent nodes synchronization to synchronization between two nodes that cannot directly communicate with each other. This idea is illustrated in Figure 10. Nodes P_1 and P_2 send out synchronization beacons, and they create two overlapping neighborhoods, where Node B lies in the overlapping area. Since Node A and Node B lie within the same neighborhood, their clock relationship (i.e., clock offset and skew) can be established from the Node P_1's reference broadcast. Similarly, the clock relationship between Node B and Node C can be established from the Node P_2's reference broadcast. Therefore, the clock relationship between Node A and Node C can be computed with Node B acting as a gateway.

More specifically, assuming only clock offset is estimated and there is only one round of synchronization beacons from P_1 and P_2, the clock offset difference between Node A and Node C is given by

$$\hat{\theta}^{CA} = T_2^{P_2\to C} - T_2^{P_2\to B} + T_2^{P_1\to B} - T_2^{P_1\to A}, \qquad (38)$$

where $T_2^{P_2\to C}$ is the arrival time of P_2's beacon recorded at Node C, and the other notations are defined similarly. In case the message delivery delay is Gaussian distributed and assuming the propagation delays are negligible, it follows quickly that (38) is an unbiased estimator for $\theta^C - \theta^A$ [25]. With more than two gateway nodes, this concept can be further extended to multihop scenarios. That is, an unbiased estimate of the pairwise offset between two nodes $\theta^{ij} := \theta^i - \theta^j$ can be obtained from any appropriate path between the two nodes, which in general is composed of the nodes $i, P_1, i_1, P_2, i_2, \ldots, i_{k-1}, P_k, j$. Therefore, the corresponding unbiased estimator can be expressed as

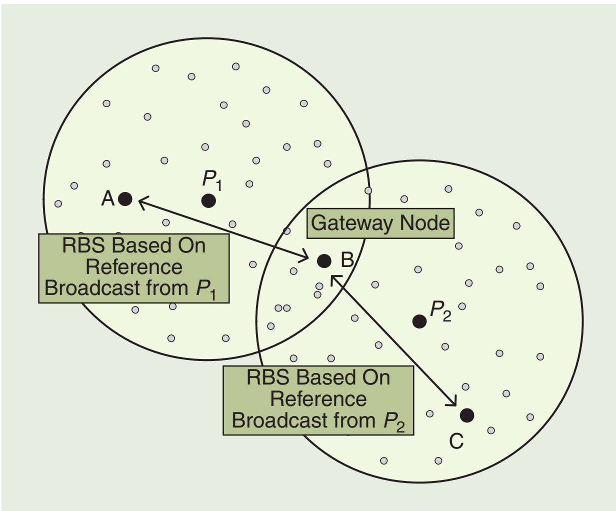
$$\hat{\theta}^{ij} = T_2^{P_1\to i} - T_2^{P_1\to i_1} + T_2^{P_2\to i_1} - T_2^{P_2\to i_2} - \cdots + T_2^{P_k\to i_{k-1}} - T_2^{P_k\to j}. \qquad (39)$$

Improved accuracy can be achieved by taking into account properly weighted combinations of alternating paths between the two nodes. Building on such a concept, it is argued in [25] that finding the MVUE of $\theta^i - \theta^j$ is related to the problem of determining the effective resistance between two nodes of a resistive network. A desirable feature of such an approach is that it produces globally consistent estimates in the sense that for any triple (n, p, q), the MVUEs of $\theta^n - \theta^p$, $\theta^p - \theta^q$, and $\theta^q - \theta^n$ sum up to zero.

Another approach to achieve global clock synchronization is to estimate the clock synchronization parameters of the whole network directly from the time stamps and at the same time. This was discussed in the context of the one-way timing message dissemination scheme in [25]. Assuming that only clock offsets need to be estimated, we generalize (10) for the transmission from a master Node P_k to a Node i

$$T_2^{P_k\to i} = T_1^{P_k} + \theta^i + X^{P_k\to i}, \qquad (40)$$

where $T_1^{P_k}$ is the transmission time at master Node P_k, and $X^{P_k\to i}$ denotes the variable portion of delays from master Node P_k to Node i. Treating $T_1^{P_k}$ as unknown and assuming $X^{P_k\to i}$ as zero mean independent Gaussian RVs with variances V_{ki}, the joint pdf of $T_2^{P_k\to i}$ takes the expression



[FIG10] **Extension of RBS to network-wide synchronization.**

$$L(T_2^{P_k\to i}|T_1^{P_k},\theta^i)=\prod_{k,i}\frac{1}{\sqrt{2\pi V_{ki}}}\exp\left[-\frac{1}{2V_{ki}}\left(T_2^{P_k\to i}-T_1^{P_k}-\theta^i\right)^2\right].$$

Differentiating the likelihood function with respect to each $T_1^{P_k}$ and θ^i, we can obtain two sets of coupled equations. Finally, a two-step iterative process can be employed to find the solution to the system of equations

$$T_1^{P_k}=\frac{\sum_i\left(T_2^{P_k\to i}-\theta^i\right)/V_{ki}}{\sum_i 1/V_{ki}}\qquad\text{for each } k,$$

$$\theta^i=\frac{\sum_k\left(T_2^{P_k\to i}-T_1^{P_k}\right)/V_{ki}}{\sum_k 1/V_{ki}}\qquad\text{for each } i.$$

It is claimed in [25] that the iterative process converges to a solution. However, the major drawback of this approach is that the variances of the transmission delays V_{ki} need to be known for each (k,i) pair, which is extremely difficult to know in advance in a practical setting.

DISTRIBUTED SIGNAL PROCESSING TECHNIQUES WILL REVEAL THE OPTIMAL WAY OF INFORMATION PASSING, THUS SAVING UNNECESSARY COMMUNICATION OVERHEAD.

CONCLUDING REMARKS AND OPEN PROBLEMS

The fundamental role of signal processing techniques was demonstrated in the context of clock synchronization in WSNs. This article explains many existing intuitive clock synchronization protocols and gives some directions to the necessary ingredients for devising an optimal estimator operating under an unconventional environment. However, centralized signal processing techniques can only help solving the problem of node-to-node synchronization and possible ad hoc extensions to network-wide synchronization. The next important step is the application of decentralized signal processing techniques (e.g., distributed estimation and detection) to the clock synchronization problem, and this naturally results in desirable distributed clock synchronization algorithms. Furthermore, distributed signal processing techniques will reveal the optimal way of information passing, thus saving unnecessary communication overhead.

On the other hand, the techniques presented in this article assume the transmissions are line of sight, and the reference node is perfectly accurate. Unfortunately, these two assumptions may not be valid in practice. The effects and mitigation of nonline-of-sight transmissions and imperfect anchors, which have been researched in localization applications, are largely unattended currently in clock synchronization. One possible solution is the use of robust estimation schemes such as the M-estimator [48]. In fact, the clock synchronization and the localization problem are two closely related areas. The propagation delay τ in this article is actually related to the distance between transmitter and receiver. Based on this observation, the joint synchronization and localization under line-of-sight transmission has been recently studied in [49], and it is found that the two problems are coupled and it is beneficial to jointly solve the two problems in the same time. Furthermore, the problem of devising joint synchronization and localization algorithms assuming nonline-of-sight transmission is also currently under investigation. With these promising results, it is believed that the signal processing techniques exploited in localization applications could be further applied in the context of clock synchronization and vice versa.

ACKNOWLEDGMENT

This work was supported by Qtel and QNRF-NPRP.

AUTHORS

Yik-Chung Wu (ycwu@eee.hku.hk) obtained his Ph.D. degree in electrical engineering from Texas A&M University in 2005. He is currently an assistant professor at the University of Hong Kong as well as an associate editor for *IEEE Communications Letters*. His research interests include general areas of signal processing and communication systems.

Qasim Chaudhari (qasim.mahmood@gmail.com) obtained his Ph.D. degree in electrical engineering from Texas A&M University, College Station, in 2008. He is currently an assistant professor at Iqra University, Islamabad, Pakistan. Before entering academia, he worked with the SoC Tools Group of Communications Enabling Technologies, Islamabad, and later with the HSDPA performance test team of Qualcomm Inc., San Diego. His research interests include digital communications, estimation and detection theory and channel estimation and synchronization in WSNs.

Erchin Serpedin (serpedin@ece.tamu.edu) is currently a professor in the Department of Electrical and Computer Engineering at Texas A&M University, which he joined after receiving his Ph.D. degree in electrical engineering from the University of Virginia in 1999. He was an associate editor for several journals including *IEEE Transactions on Information Theory*, *IEEE Transactions on Wireless Communications*, *IEEE Transactions on Signal Processing*, *IEEE Transactions on Communications*, *IEEE Signal Processing Letters*, *Signal Processing* (Elsevier), and *IEEE Communications Letters*. His research interests include the areas of statistical signal processing and wireless communications.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Comput. Networks*, vol. 38, no. 4, pp. 393–422, Mar. 2002.

[2] N. Bulusu and S. Jha, *Wireless Sensor Networks: A Systems Perspective*. Norwood, MA: Artech House, 2005.

[3] J. R. Vig, "Introduction to quatz frequency standards," Army Research Laboratory Electronics and Power Sources Directorate Tech. Rep., SLCET-TR-92-1, Oct. 1992.

[4] H. Kopetz and W. Ochsenreiter, "Clock synchronization in distributed real-time systems," *IEEE Trans. Comput.*, vol. 36, no. 8, pp. 933–939, Aug. 1987.

[5] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The flooding time synchronization protocol," in *Proc. 2nd Int. Conf. Embedded Networked Sensor Systems*, ACM Press, Nov. 2004, pp. 39–49.

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

[7] A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering*. 2nd ed. Reading, MA: Addison-Wesley, 1993.

[8] C. Bovy, H. Mertodimedjo, G. Hooghiemstra, H. Uijterwaal, and P. Mieghem, "Analysis of end-to-end delay measurements in Internet," in *Proc. Passive and Active Measurements Workshop*, Fort Collins, CO, Mar. 2002, pp. 26–33.

[9] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proc. 5th Symp. Operating System Design and Implementation*, Dec. 2002, pp. 147–163.

[10] H. S. Abdel-Ghaffar, "Analysis of synchronization algorithm with time-out control over networks with exponentially symmetric delays," *IEEE Trans. Commun.*, vol. 50, pp. 1652–1661, Oct. 2002.

[11] S. Moon, P. Skelley, and D. Towsley, "Estimation and removal of clock skew from network delay measurements," in *Proc. IEEE INFOCOM*, New York, Mar. 1999, pp. 227–234.

[12] V. Paxson, "On calibrating measurements of packet transit times," in *Proc. 7th ACM Sigmetrics Conf.*, June 1998, pp. 11–21.

[13] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Commun. ACM*, vol. 43, no. 5, pp. 51–58, May 2000.

[14] Y.-W. Hong and A. Scaglione, "A scalable synchronization protocol for large scale sensor networks and its applications," *IEEE JSAC*, vol. 23, no. 5, pp. 1085–1099, May 2005.

[15] O. Simeone, U. Spagnolini, Y. Bar-Ness, and S. H. Strogatz, "Distributed synchronization in wireless networks," *IEEE Signal Processing Mag.*, vol. 25, no. 5, pp. 81–97, Sept. 2008.

[16] A. Hu and S. D. Servetto. (2006). A scalable protocol for cooperative time synchronization using spatial averaging [Online]. Available: http://arxiv.org/PS$_c$ache/cs/pdf/0611/0611003v1.pdf

[17] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *Proc. SenSys 03*, Los Angeles, CA, Nov. 2003, pp. 138–149.

[18] M. L. Sichitiu and C. Veerarittiphan, "Simple, accurate time synchronization for wireless sensor networks," in *Proc. IEEE WCNC*, New Orleans, LA, Mar. 2003, pp. 1266–1273.

[19] J. Van Greunen and J. Rabaey, "Lightweight time synchronization for sensor networks," in *Proc. 2nd ACM Int. Conf. Wireless Sensor Networks and Applications (WSNA)*, San Diego, CA, 2003, pp. 11–19.

[20] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, "Clock synchronization for wireless sensor networks: a survey," *Ad-Hoc Networks*, vol. 3, no. 3, pp. 281–323, Mar. 2005.

[21] K. Noh, Q. Chaudhari, E. Serpedin, and B. Suter, "Novel clock phase offset and skew estimation two-way timing message exchanges for wireless sensor networks," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 766–777, Apr. 2007.

[22] M. Leng and Y.-C. Wu, "On clock synchronization algorithms for wireless sensor networks under unknown delay," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 182–190, Jan. 2010.

[23] P. Huang, M. Desai, X. Qiu, and B. Krishnamachari, "On the multihop performance of synchronization mechanisms in high propagation delay networks," *IEEE Trans. Comput.*, vol. 58, no. 5, pp. 577–590, May 2009.

[24] D. Jeske, "On the maximum likelihood estimation of clock offset," *IEEE Trans. Commun.*, vol. 53, no. 1, pp. 53–54, Jan. 2005.

[25] R. Karp, J. Elson, C. Papadimitriou, and S. Shenker, "Global synchronization in sensornets," in *Proc. 6th Latin American Symp. Theoretical Informatics*, Apr. 2004, pp. 609–624.

[26] Q. Chaudhari, E. Serpedin, and K. Qaraqe, "On maximum likelihood estimation of clock offset and skew in networks with exponential delays," *IEEE Trans. Signal Processing*, vol. 56, no. 4, pp. 1685–1697, Apr. 2008.

[27] J. Li and D. Jeske, "Maximum likelihood estimators of clock offset and skew under exponential delays," *Appl. Stochastic Models Bus. Ind.*, vol. 25, no. 4, pp. 445–459, Apr. 2009.

[28] M. Leng and Y.-C. Wu, "On joint synchronization of clock offset and skew for wireless sensor networks under exponential delay," in *Proc. IEEE ISCAS*, Paris, France, May 2010, pp. 461–464.

[29] I. Sari, E. Serpedin, K.-L. Noh, Q. Chaudhari, and B. Suter, "On the joint synchronization of clock offset and skew in RBS-protocol," *IEEE Trans. Commun.*, vol. 56, no. 5, pp. 700–703, May 2008.

[30] Q. Chaudhari, E. Serpedin, and K. Qaraqe, "On minimum variance unbiased estimation of clock offset in distributed networks," *IEEE Trans. Inform. Theory*, vol. 56, no. 6, pp. 2893–2904, June 2010.

[31] N. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions*, 2nd ed. New York: Wiley, vol. 1, 1994.

[32] S. M. Kay, *Fundamentals of Statistical Signal Processing, Vol. I. Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[33] Q. Chaudhari, E. Serpedin, and Y.-C. Wu, "Improved estimation of clock offset in sensor networks," in *Proc. IEEE Int. Conf. Communications (ICC)*, Dresden, Germany, June 2009, pp. 1–4.

[34] B. Scheuermann, W. Kiess, M. Roos, M. Mauve, and F. Jarre, "On the time synchronization of distributed log files in networks with local broadcast media," *IEEE/ACM Trans. Networking*, vol. 17, no. 2, pp. 431–444, Apr. 2009.

[35] M. Lemmon, J. Ganguly, and L. Xia, "Model-based clock synchronization in networks with drifting clocks," in *Proc. Pacific Rim Int. Symp. Dependable Computing*, Los Angeles, CA, pp. 177–184, Dec. 2000.

[36] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*. London: U.K.: Chapman & Hall, 1993.

[37] A. M. Zoubir and D. R. Iskander, *Bootstrap Techniques for Signal Processing*. Cambridge: U.K.: Cambridge Univ. Press, 2004.

[38] J. Lee, J. Kim, and E. Serpedin, "Clock offset estimation in wireless sensor networks using bootstrap bias correction," in *Proc. Int. Conf. Wireless Algorithms, Systems, and Applications*, Dallas, TX, Oct. 2008, pp. 322–329.

[39] D. Jeske and A. Sampath, "Estimation of clock offset using bootstrap bias-correction techniques," *Technometrics*, vol. 45, no. 3, pp. 256–261, Aug. 2003.

[40] J. H. Kotecha and P. M. Djurić, "Gaussian sum particle filtering," *IEEE Trans. Signal Processing*, vol. 51, no. 10, pp. 2602–2612, Oct. 2003.

[41] B. D. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.

[42] R. Solis, V. Borkar, and P. R. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," in *Proc. 45th IEEE Conf. Decision and Control*, San Diego, CA, Dec. 2006, pp. 2734–2739.

[43] A. Giridhar and P. Kumar, "Distributed clock synchronization over wireless networks: algorithms and analysis," in *Proc. 45th IEEE Conf. Decision and Control*, San Diego, CA, Dec. 2006, pp. 4915–4920.

[44] Q. Li and D. Rus, "Global clock synchronization in sensor networks," *IEEE Trans. Comput.*, vol. 55, no. 2, pp. 214–226, Feb. 2006.

[45] K.-L. Noh, E. Serpedin, and K. A. Qaraqe, "A new approach for time synchronization in wireless sensor networks: Pairwise broadcast synchronization," *IEEE Trans. Wireless Commun.*, vol. 7, no. 9, pp. 3318–3322, Sept. 2008.

[46] K.-L. Noh, Y.-C. Wu, K. Qaraqe, and B. Suter, "Extension of pairwise broadcasting clock synchronization for multi-cluster sensor networks," *EURASIP J. Adv. Signal Process.*, (Special Issue on Distributed Signal Processing Techniques for Wireless Sensor Networks), vol. 2008, pp. 1–10, 2008.

[47] K.-Y. Cheng, K.-S. Lui, Y.-C. Wu, and V. Tam, "A distributed multihop time synchronization protocol for wireless sensor networks using pairwise broadcast synchronization," *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 1764–1772, Apr. 2009.

[48] S. H. Zhao and S. C. Chan, "A novel algorithm for mobile station location estimation with none line of sight error using robust least M-estimation," in *Proc. IEEE ISCAS*, 2008, Seattle, WA, May 2008, pp. 1176–1179.

[49] J. Zheng and Y.-C. Wu, "Joint time synchronization and localization of an unknown node in wireless sensor networks," *IEEE Trans. Signal Processing*, vol. 58, no. 3, pp. 1309–1320, Mar. 2010.