

*Review*

# **An Overview of a Class of Clock Synchronization Algorithms for Wireless Sensor Networks: A Statistical Signal Processing Perspective**

**Xu Wang**<sup>1</sup>, **Daniel Jeske**<sup>2</sup> and **Erchin Serpedin**<sup>1,\*</sup>

<sup>1</sup> Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA; E-Mail: xu.wang@tamu.edu

<sup>2</sup> Department of Statistics, University of California Riverside, Riverside, CA 92521, USA; E-Mail: daniel.jeske@ucr.edu

\* Author to whom correspondence should be addressed; E-Mail: serpedin@ece.tamu.edu; Tel.: +1-979-458-2287; Fax: +1-979-862-4630.

Academic Editor: Alberto Marchetti Spaccamela

*Received: 22 May 2015 / Accepted: 31 July 2015 / Published: 6 August 2015*

---

**Abstract:** Recently, wireless sensor networks (WSNs) have drawn great interest due to their outstanding monitoring and management potential in medical, environmental and industrial applications. Most of the applications that employ WSNs demand all of the sensor nodes to run on a common time scale, a requirement that highlights the importance of clock synchronization. The clock synchronization problem in WSNs is inherently related to parameter estimation. The accuracy of clock synchronization algorithms depends essentially on the statistical properties of the parameter estimation algorithms. Recently, studies dedicated to the estimation of synchronization parameters, such as clock offset and skew, have begun to emerge in the literature. The aim of this article is to provide an overview of the state-of-the-art clock synchronization algorithms for WSNs from a statistical signal processing point of view. This article focuses on describing the key features of the class of clock synchronization algorithms that exploit the traditional two-way message (signal) exchange mechanism. Upon introducing the two-way message exchange mechanism, the main clock offset estimation algorithms for pairwise synchronization of sensor nodes are first reviewed, and their performance is compared. The class of fully-distributed clock offset estimation algorithms for network-wide synchronization is then surveyed. The paper concludes with a list of open research problems pertaining to clock synchronization of WSNs.

**Keywords:** wireless sensor networks; synchronization; clock offset; clock skew; time; estimation; algorithm; statistics

---

## 1. Introduction

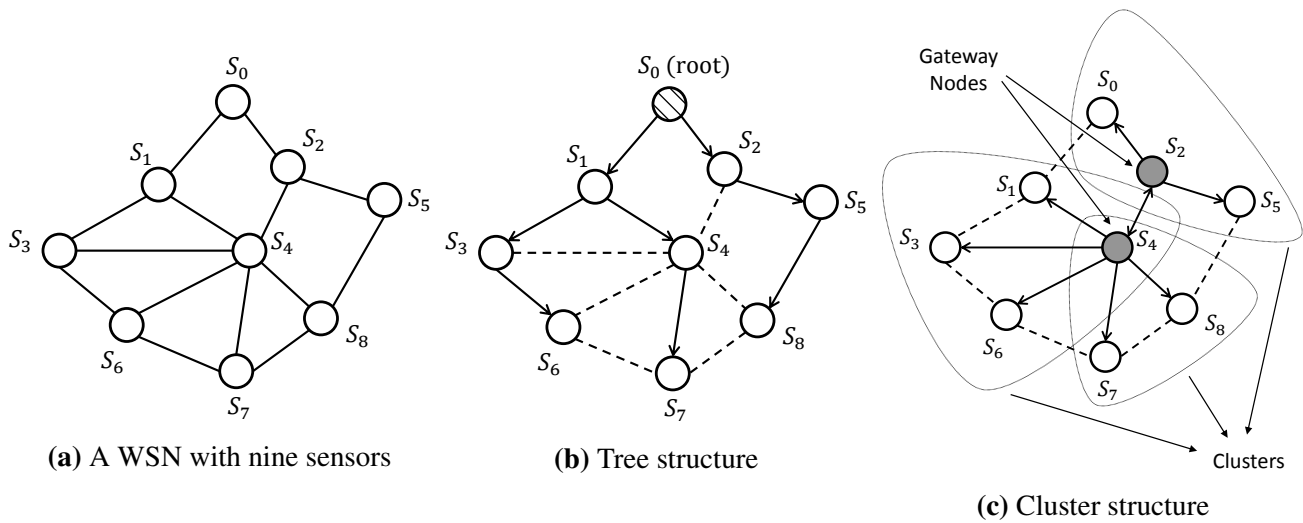
A wireless sensor network (WSN) is a group of spatially-distributed autonomous sensors, which monitor physical or environmental conditions, such as temperature, humidity, speed, pressure, *etc.*, and then transmit the recorded data to a central computing unit for analysis. WSNs originate from military-oriented applications, such as battlefield surveillance and movement monitoring. Nowadays, WSNs have witnessed a rapid growth and have been implemented in a variety of promising applications, which can be classified as follows:

- Military [1]: battlefield surveillance; forces monitoring; battle damage assessment.
- Health [2,3]: tracking and monitoring patients; telemonitoring of human physiological data.
- Environment [4,5]: air pollution monitoring; water quality monitoring; forest fire detection.
- Home [6,7]: home automation; home appliances management and monitoring.
- Industry [8]: machine health monitoring; waste monitoring; data logging.

Most of these applications require the clocks of network nodes to be synchronized, because performing a joint task requires all of the nodes to operate on a common time scale. For instance, data logging is a basic operation that collects, processes and transmits data and is used for temporal and spatial monitoring of a habitat. The advantage of data logging in WSNs over the conventional data logging lies in the property of real-time data being collected, which requires some or all nodes in the network to share a common time frame.

Each sensor in a WSN operates independently and has its own clock. Even if the clocks of sensors are initially tuned perfectly, due to the imperfections of the clock oscillator, they may drift away from the ideal time as time evolves [9]. Hence, developing efficient clock synchronization protocols is critical in WSNs. In the literature, many clock synchronization algorithms rely on the clock information from the Global Positioning System (GPS). However, GPS is not ubiquitously available and requires a high-power receiver [10]. This fact makes it impractical to implement GPS technology in the clock synchronization of WSNs, since WSNs generally consist of small sensors that have limited energy. Furthermore, sensors sometimes are positioned in an environment where the GPS signal is not available, as is the case with underwater or underground sensors for water quality monitoring. For clock synchronization applications in computer networks, the network time protocol (NTP) represents one of the oldest and standard Internet timing protocols in current use [11]. However, NTP is still not suitable for WSNs, due to the lack of energy availability and unstable working environment in WSNs. Therefore, a variety of synchronization protocols have been particularly designed for WSNs. In general, these protocols perform clock synchronization via timing message exchanges among sensors. However, due to the power requirement for each individual sensor, sensors have a limited communication range, and they are not able to communicate with every other sensor. An example of a WSN with nine sensors is shown

in Figure 1a, where each circle denotes a sensor and each edge represents the communication link (if present) between the corresponding pair of sensors.



**Figure 1.** Network topology for different synchronization approaches.

Traditional synchronization protocols for WSNs generally include two steps: first, the synchronization is conducted within a pair of neighboring nodes; second, a multi-hop structure is built, such that the synchronization can be extended into all of the nodes through a layer-by-layer synchronization. One natural approach to deal with the multi-hop structure of WSNs is to construct a tree-based network. Specifically, as illustrated in Figure 1b, a sensor is selected as the reference node, then a spanning tree rooted at this node is built, and each sensor synchronizes with its parent pairwise along the unique path from that sensor to the root. On the other hand, the cluster-based structure divides the network into several interconnected single-hop clusters, as graphically shown in Figure 1c. In each cluster, a reference node is selected, and all of the other nodes are pairwise synchronized by the reference node. The reference nodes of different clusters act as gateways to adjust the local clock of different clusters into one common time frame. In general, the core of these protocols is based on timing message exchanges between a pair of nodes, in which accurate and efficient pairwise clock synchronization algorithms play a key role. In the literature, some widely-used clock synchronization protocols, such as the timing-sync protocol for sensor networks (TPSN) [12], the tiny/mini synchronization [13], the light-weight time synchronization (LTS) [14] and the flooding time synchronization protocol (FTSP) [15], employ a two-way message exchange mechanism to adjust the clock between any two nodes.

However, the traditional synchronization protocols suffer from large overhead in building and maintaining the spanning-tree or cluster structure. In addition, some nodes act as a root (in the spanning-tree structure) or as a gateway (in the cluster structure), and the failure of such nodes may lead to the failure of a large number of nodes connecting to it. To tackle this problem, several algorithms based on a fully-distributed communication topology have been proposed. In such algorithms, there are no special nodes, such as roots and gateways, and thus, no structure needs to be built. With all of the

nodes performing exactly the same algorithm and communicating with their neighboring nodes, these protocols are robust to the topology changes and failures of nodes.

In this paper, the latest algorithms in the realm of clock synchronization for WSNs are surveyed following a statistical signal processing viewpoint. Specifically, the rest of the paper is organized as follows. Section 2 introduces the clock model and the two-way message exchange mechanism for pairwise clock synchronization. Based on the two-way message exchange mechanism, the state-of-the-art pairwise clock synchronization algorithms under Gaussian random delays are investigated in Section 3. Section 4 overviews the pairwise clock synchronization algorithms in the presence of exponential random delays. Section 5 surveys the signal processing techniques employed in WSNs when the random delays are unknown. In Section 6, synchronization algorithms that assume a fully-distributed operation are discussed. Finally, Section 7 concludes the paper and provides some open research topics.

In the literature, surveys for clock synchronization in WSNs have been presented by [9,10,16,17]. The focus of [10,16] lies in the clock synchronization protocols that are applied in WSNs. The survey paper [17] provides a technical overview of the history, recent advances and challenges in distributed clock synchronization for distributed wireless networks, while this article presents synchronization algorithms with emphasis on the mathematical and statistical techniques employed for clock synchronization parameters estimation. The work in [9] surveys the latest advances in the clock synchronization of WSNs following a signal processing viewpoint. However, only pairwise synchronization methods are investigated in [9], while our paper discusses both pairwise and fully-distributed synchronization algorithms.

## 2. System Model for Pairwise Clock Synchronization

In this section, the clock model for each sensor node is first introduced and the notions of clock offset and skew are described. Then, the classic mechanism for message exchange between two adjacent nodes, namely the two-way message exchange mechanism, is presented.

### 2.1. Clock Offset and Skew

As discussed earlier, each sensor node in a WSN works independently and has its own clock. Ideally, the clock of a sensor node is modeled as  $c(t) = t$ , where  $t$  represents the reference time [9]. However, due to the imperfections of the clock oscillator, as well as the environmental factors, such as pressure, temperature and hardware aging, the clock is subject to change as time evolves. Generally, the clock function of node  $i$  can be expressed as:

$$c_i(t) = \theta_i + f_i \cdot t \quad (1)$$

where  $\theta_i$  and  $f_i$  denote the clock offset (phase difference) and the clock skew (frequency difference) for sensor node  $i$ , respectively [9]. In this way, if the clock of Node A is selected to be the reference time, it follows that:

$$c_B(t) = \theta + f \cdot c_A(t)$$

where  $\theta$  and  $f$  stand for the clock offset and skew between Nodes A and B, respectively. A graphical interpretation of the relationship between the clocks of Nodes A and B is shown in Figure 2. It can be seen that if  $\theta = 0$  and  $f = 1$ , then Nodes A and B are perfectly synchronized. Otherwise, the synchronization process aims to estimate the clock offset  $\theta$  and skew  $f$ , such that Node B can adjust its local clock to the reference time (Node A). As time evolves, the clock offset and skew are subject to change due to various reasons, such as the changes induced by the environment and sensor hardware aging. Therefore, the clock parameters are synchronized periodically, such that they are tuned up-to-date.

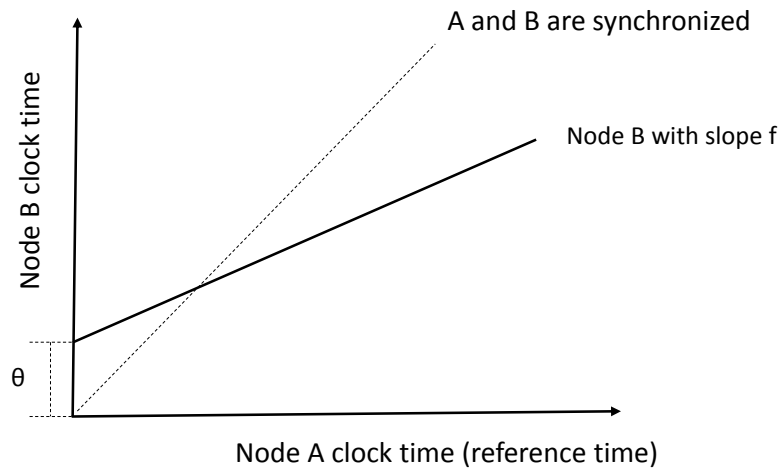


Figure 2. Clocks of sensor nodes.

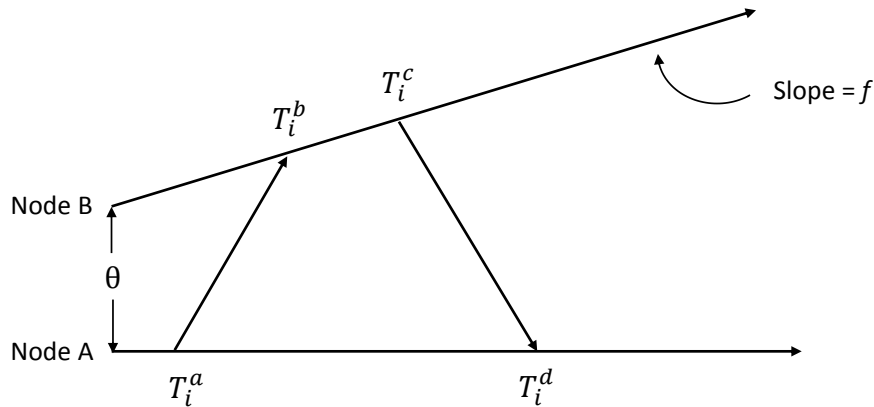
### 2.2. Two-Way Message Exchange Mechanism

The two-way message exchange is a classic mechanism for communicating the timing messages between two adjacent nodes and has been widely used in the literature [12–15]. For pairwise synchronization, one of the sensor nodes is selected as the reference node. For example, a two-way message exchange model between Node A and Node B is shown in Figure 3 in the situation when Node A is serving as the reference node. During message exchange round  $i$ , the synchronization begins at Node A, and a synchronization message containing the sending time  $T_i^a$  is sent to Node B. Next, Node B records the reception of the message at its clock time  $T_i^b$  and sends an acknowledgment message to Node A at  $T_i^c$ . The acknowledgment message contains the timestamps  $T_i^b$  and  $T_i^c$ . At last, Node A receives the acknowledgment message at  $T_i^d$ , and the message exchange round  $i$  ends. This process is repeated  $N$  times, where  $N$  stands for the required number of observations.

Based on the procedure depicted above, the clock synchronization problem can be mathematically modeled as

$$\begin{aligned} T_i^b &= f(T_i^a + d + X_i) + \theta \\ T_i^c &= f(T_i^d - d - Y_i) + \theta \end{aligned} \tag{2}$$

where  $\theta$  and  $f$  represent the clock offset and clock skew, respectively. Moreover,  $X_i$  and  $Y_i$  denote the random delays from Node A to Node B and from Node B to Node A, respectively, and  $d$  stands for the fixed portion of delays. In addition,  $T_i^a, T_i^d$  and  $T_i^b, T_i^c$  are time stamps recorded at Nodes A and B, respectively.



**Figure 3.** Two-way message exchange mechanism (with skew).

If we consider only the clock (phase) offset between two adjacent nodes, *i.e.*,  $f = 1$ , the system model in Equation (2) can be expressed as:

$$\begin{aligned} T_i^b &= T_i^a + d + X_i + \theta \\ T_i^c &= T_i^d - d - Y_i + \theta \end{aligned} \tag{3}$$

The equations in Equation (3) can be further simplified as:

$$\begin{aligned} U_i &= d + \theta + X_i \\ V_i &= d - \theta + Y_i \end{aligned} \tag{4}$$

where  $U_i$  and  $V_i$  contain the clock information and are defined as  $U_i = T_i^b - T_i^a$  and  $V_i = T_i^d - T_i^c$ , respectively.

### 3. Pairwise Clock Synchronization under Gaussian Delays

In general, probability density function (PDF) models have been employed for modeling the random portion of delays in WSNs. Examples of PDFs that have been adopted in the literature include exponential, Gaussian, Weibull and Gamma [18–21]. In this section, we use the Gaussian distribution to characterize the random delay in WSNs, *i.e.*,  $X_i$  and  $Y_i$  are Gaussian random variables with parameters  $(\mu_1, \sigma_1)$  and  $(\mu_2, \sigma_2)$ , respectively. The reasons for selecting the Gaussian distribution for modeling the random delays are as follows:

- The central limit theorem (CLT) states that the PDF of the sum of a large number of independent and identically-distributed (i.i.d.) random variables is approximately normally distributed. Therefore, the Gaussian model is appropriate if the random delays are assumed to be the summation of multiple independent random variables.
- Experimental results based on two Texas Instruments ez430-RF2500 evaluation boards were recorded in [22] to demonstrate the fitness of the Gaussian distribution in modeling the random portion of delays in WSNs.

The most representative pairwise clock synchronization algorithms in the literature focused on developing maximum likelihood estimators (MLEs) for the clock offset and skew. The MLE is one

of the most widely-used estimators in the area of parameter estimation, since it achieves asymptotically the Cramér–Rao lower bound (CRLB).

Based on the two-way message exchange model in Equation (4), where only the clock offset is considered, [18] derived the MLE for clock offset by assuming a known fixed delay  $d$  and symmetric random delays, *i.e.*,  $\mu_1 = \mu_2 = \mu$  and  $\sigma_1 = \sigma_2 = \sigma$ . In this framework, the likelihood function is expressed as:

$$p(U_i, V_i|\theta) = (2\pi\sigma^2)^{-N} e^{-\frac{1}{2\sigma^2} \left[ \sum_{i=1}^N (U_i - d - \theta - \mu)^2 + \sum_{i=1}^N (V_i - d + \theta - \mu)^2 \right]}$$

The MLE of clock offset is derived by differentiating the log-likelihood function with respect to  $\theta$  and setting the corresponding derivative to zero, and it assumes the expression:

$$\theta_{ML} = \frac{\bar{U} - \bar{V}}{2} \tag{5}$$

where  $\bar{U}$  and  $\bar{V}$  stand for the sample means of observations  $\{U_i\}_{i=1}^N$  and  $\{V_i\}_{i=1}^N$ , respectively. It can be verified that the estimator Equation (5) is unbiased and achieves the CRLB. The MLE estimator Equation (5) represents also the minimum variance unbiased estimator (MVUE).

Following the same assumptions, *i.e.*, a known fixed delay and symmetric random delays, the joint maximum likelihood estimates of the clock offset and skew were also proposed in [18] based on the two-way message exchange model in Equation (2). Specifically, the likelihood function is first formulated as:

$$p(U_i, V_i|\theta, f') = (2\pi\sigma^2)^{-N} e^{-\frac{1}{2\sigma^2} \sum_{i=1}^N \{ [f'(\theta - T_i^b) + (T_i^a + d)]^2 + [f'(\theta - T_i^c) + (T_i^d - d)]^2 \}}$$

where  $f' = 1/(1 + f)$ . Differentiating the log-likelihood function with respect to  $\theta$  leads to an equation of  $\theta$  in terms of  $f'$ . In addition, differentiating the log-likelihood function with respect to  $f'$  results in an equation of  $f'$  in terms of  $\theta$ . The joint estimates of  $\theta$  and  $f'$ , denoted as  $\hat{\theta}_{ML}$  and  $\hat{f}'_{ML}$ , can be obtained by combining those two equations, and the MLE of  $f$  is obtained as  $\hat{f}_{ML} = 1/(1 + \hat{f}'_{ML})$ , due to the invariance principle [23].

However, in some practical cases, the value of the fixed delay  $d$  is unknown and sometimes even represents a parameter that needs to be estimated, as is the case with the node localization application [24]. Therefore, efficient algorithms for joint estimation of the clock offset, clock skew and fixed delay are of great interest. Based on the two-way message exchange mechanism in Equation (2), Leng *et al.* [24] extended the result in [18] by assuming an unknown fixed delay  $d$ . First, the signaling model Equation (2) is re-formulated in matrix form as:

$$\underbrace{\begin{bmatrix} T_1^a \\ \vdots \\ T_N^a \\ -T_1^d \\ \vdots \\ -T_N^d \end{bmatrix}}_{\hat{=}\mathbf{C}_r} + d \cdot \mathbf{1} = \underbrace{\begin{bmatrix} T_1^b & -1 \\ \vdots & \vdots \\ T_N^b & -1 \\ -T_1^c & 1 \\ \vdots & \vdots \\ -T_N^c & 1 \end{bmatrix}}_{\hat{=}\mathbf{C}_h} \underbrace{\begin{bmatrix} \phi_1 \\ \phi_0 \end{bmatrix}}_{\hat{=}\Phi} - \underbrace{\begin{bmatrix} X_1 \\ \vdots \\ X_N \\ Y_1 \\ \vdots \\ Y_N \end{bmatrix}}_{\hat{=}\mathbf{Z}}$$

where  $\phi_0 \triangleq \theta/f$ ,  $\phi_1 \triangleq 1/f$  and  $\mathbf{1} = [1, 1, \dots, 1]^T$  with dimension  $2N \times 1$ .

Since  $X_i$  and  $Y_i$  are independent Gaussian random variables, *i.e.*,  $X_i \sim \mathcal{N}(0, \sigma^2)$ ,  $Y_i \sim \mathcal{N}(0, \sigma^2)$ , the log-likelihood function for  $\Phi$  and  $d$  is expressed as:

$$\ln p(\mathbf{C}_r, \mathbf{C}_h | \Phi, d) = \ln \frac{N}{2\pi\sigma^2} - \frac{\|\mathbf{C}_r + d \cdot \mathbf{1} - \mathbf{C}_h \Phi\|^2}{2\sigma^2}$$

For a fixed  $d$ , the MLE of  $\Phi$  is given by:

$$\hat{\Phi}(d) = (\mathbf{C}_h^T \mathbf{C}_h)^{-1} \mathbf{C}_h^T (\mathbf{C}_r + d\mathbf{1}) \tag{6}$$

Plugging Equation (6) into the log-likelihood function results in a compressed likelihood function with only one variable  $d$ . The MLE of  $d$  is derived by taking the derivative over the resulting log-likelihood function with respect to  $d$ , and the MLE of  $\Phi$  is obtained by plugging the corresponding estimate of  $d$  back into Equation (6). Finally, the MLEs of  $\theta$  and  $f$  are obtained from the MLE of  $\Phi$  using the invariance principle [23].

#### 4. Pairwise Clock Synchronization under Exponential Delays

In the literature, the exponential distribution is also commonly exploited to represent the random portion of delays in WSNs. The reasons behind choosing the exponential distribution include:

- For the point-to-point hypothetical reference connection (HRX) between two nodes, a single-server M/M/1queue can appropriately represent the aggregate link delay, where the random delays are modeled as independent exponential random variables [25].
- Experiment results were carried out in [26,27] to verify the appropriateness of choosing the exponential distribution to characterize the random delays in WSNs.
- Among all distributions with a fixed mean in the support  $[0, +\infty)$ , the exponential distribution achieves the maximum differential entropy, and thus, it is the least informative.

The most representative clock offset estimation methods for pairwise synchronization in the presence of exponential delays are the maximum likelihood estimator (MLE), the best linear unbiased estimator (BLUE) and the minimum variance unbiased estimator (MVUE), and they will be first reviewed in this section. The MLE for joint estimation of the clock offset and skew will be investigated later. Finally, an important statistical concept, referred to as the confidence interval for the clock offset, will be investigated.

##### 4.1. Clock Offset Estimation under Exponential Delays

###### 4.1.1. Maximum Likelihood Estimator

In the two-way message exchange mechanism Equation (4) where only clock offset exists, we assume that  $X_i$  and  $Y_i$  are exponentially-distributed random variables with means  $\lambda_1$  and  $\lambda_2$ , respectively. Assuming a known fixed delay, Abdel-Ghaffar [25] reported that the MLE of the clock offset does not exist when the exponential random delays are assumed to be known and symmetric, *i.e.*,  $\lambda_1 = \lambda_2 = \lambda$ ,



since the likelihood function does not admit a unique maximum. However, Jeske [28] derived a closed-form solution for the MLE of clock offset by assuming an unknown fixed delay and an unknown mean for the exponential delay. In this way, a joint estimation of  $\theta$ ,  $d$  and  $\lambda$  can ensure a unique maximum of the likelihood function. Specifically, assuming symmetric exponential delays with a common mean  $\lambda$ , the likelihood function for  $[d, \theta, \lambda]^T$  is expressed as:

$$p(U_i, V_i|d, \theta, \lambda) = \lambda^{-2N} e^{-\left(\sum_{i=1}^N U_i + \sum_{i=1}^N V_i - 2Nd\right)/\lambda} \times I[U_{(1)} \geq d + \theta, V_{(1)} \geq d - \theta] \tag{7}$$

where  $I[\cdot]$  denotes the indicator function. Let the first order statistics  $U_{(1)}$  and  $V_{(1)}$  denote the minimum values of observations  $\{U_i\}_{i=1}^N$  and  $\{V_i\}_{i=1}^N$ , respectively. It can be shown that for fixed values of  $d$  and  $\theta$ , the optimal value of  $\lambda$  that maximizes the likelihood function is given by  $\lambda = (\bar{U} + \bar{V})/2 - d$ . Therefore, the likelihood function Equation (7) can be simplified to:

$$p(U_i, V_i|d, \theta) = \left(\frac{\bar{U} + \bar{V}}{2} - d\right)^{-2N} e^{-2N} \times I[U_{(1)} \geq d + \theta, V_{(1)} \geq d - \theta]$$

It can be observed that the indicator function poses a high impact on the value of the likelihood function, and moreover, the indicator function depends on  $U_{(1)}$  and  $V_{(1)}$ . Hence, in order to maximize the likelihood function, different assumptions on  $U_{(1)}$  and  $V_{(1)}$  are discussed in [28], e.g.,  $U_{(1)} > V_{(1)} > 0$ , and the region where  $p(d, \theta|U_i, V_i) > 0$  is illustrated separately for each assumption. The value of  $\theta$  that maximizes the likelihood function is found graphically from the corresponding support region. It is shown that all of the different assumptions on  $U_{(1)}$  and  $V_{(1)}$  lead to the same MLEs for the vector  $[d, \theta, \lambda]^T$ :

$$\begin{bmatrix} \hat{d}_{ML} \\ \hat{\theta}_{ML} \\ \hat{\lambda}_{ML} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} U_{(1)} + V_{(1)} \\ U_{(1)} - V_{(1)} \\ \bar{U} + \bar{V} - U_{(1)} - V_{(1)} \end{bmatrix} \tag{8}$$

Our primary interest is the MLE of the clock offset  $\hat{\theta}_{ML} = (U_{(1)} - V_{(1)})/2$ , but in the meantime, the MLEs for  $d$  and  $\lambda$  are also derived. It can be verified that  $\hat{\theta}_{ML}$  is unbiased and has a variance  $\lambda^2/(2N)$ . Interestingly, if  $d$  is unknown and the variable portions of delays are not symmetric, i.e.,  $\lambda_1 \neq \lambda_2$ , the MLE of  $\theta$  also admits the form  $\hat{\theta}_{ML} = (U_{(1)} - V_{(1)})/2$ . However, in this case, the MLE is biased.

Different from the graphical analysis used in [28], [29] analytically derived the MLE of clock offset under exponential delays using convex optimization tools. In particular, the system model in Equation (4) is re-written as:

$$U_i = \xi + X_i, \quad V_i = \psi + Y_i \tag{9}$$

where  $\xi \triangleq d + \theta$  and  $\psi \triangleq d - \theta$ . The goal is to determine the MLEs of  $\xi$  and  $\psi$ , i.e.,  $\hat{\xi}_{ML}$  and  $\hat{\psi}_{ML}$ , such that the MLE of  $\theta$  can be obtained from:

$$\hat{\theta}_{ML} = (\hat{\xi}_{ML} - \hat{\psi}_{ML}) / 2 \tag{10}$$

based on the invariance principle [23]. Towards this end, the density functions of  $U_i$  and  $V_i$  are expressed as:

$$p(U_i|\xi) = \lambda_1^N \exp\left(-\lambda_1 \sum_{i=1}^N (U_i - \xi)\right) I(U_{(1)} - \xi) \tag{11}$$

$$p(V_i|\psi) = \lambda_2^N \exp\left(-\lambda_2 \sum_{i=1}^N (V_i - \psi)\right) I(V_{(1)} - \psi) \tag{12}$$

The resulting MLE of  $\xi$  can be formulated as a convex optimization problem:

$$\begin{aligned} \hat{\xi}_{ML} &= \arg \max_{\xi} \exp(N\lambda_1\xi) \\ \text{s.t. } &\xi \leq U_{(1)} \end{aligned}$$

and it follows that  $\hat{\xi}_{ML} = U_{(1)}$ . The analysis of  $\psi$  is analogous to the case of  $\xi$ , and it yields that  $\hat{\psi}_{ML} = V_{(1)}$ . Thus, the MLE of  $\theta$  follows from Equation (10) and admits the form:

$$\hat{\theta}_{ML} = \frac{U_{(1)} - V_{(1)}}{2}$$

which coincides with the result in Equation (8). The advantage of this analytical approach is that it provides a more general derivation of MLEs, which can be applied to obtain the MLEs for the delays under other types of distributions, which include Gaussian and log-normal as special cases [29].

#### 4.1.2. Best Linear Unbiased Estimator

In the statistical signal processing field, the minimum mean square error (MSE) is commonly selected as the criterion to measure the performance of estimators. The MSE of an estimator  $\hat{\theta}$  is defined as:

$$\text{MSE}(\hat{\theta}) = \text{var}(\hat{\theta}) + [\text{bias}(\hat{\theta})]^2 \tag{13}$$

where  $\text{var}(\hat{\theta})$  stands for the variance of  $\hat{\theta}$  and  $\text{bias}(\hat{\theta}) \triangleq \text{E}(\hat{\theta}) - \theta$ .

In general, the MVUE is referred to as the ‘‘optimal’’ estimator, since it minimizes the variance in the class of unbiased estimators. However, it frequently occurs that the MVUE is difficult or impossible to be derived or might not even exist [23]. For such scenarios, a suboptimal estimator is required. A common approach is to constrain the estimator to be linear in terms of the observations and to find the linear estimator that is unbiased and achieves the minimum variance. Such an estimator is referred to as the BLUE. The BLUE of the clock offset under asymmetric exponential delays was first derived by Jeske and Sampath in [30] using the order statistics of the observations  $[U_{(1)}, U_{(2)}, \dots, U_{(N)}, V_{(1)}, V_{(2)}, \dots, V_{(N)}]^T$  where  $U_{(1)} \leq U_{(2)} \leq \dots \leq U_{(N)}$  and  $V_{(1)} \leq V_{(2)} \leq \dots \leq V_{(N)}$ . Specifically, the BLUE is formulated as a linear combination of the order statistics:

$$T = \sum_{i=1}^N a_i U_{(i)} + \sum_{i=1}^N b_i V_{(i)} \tag{14}$$

The variance of estimator  $T$  and the unbiasedness condition are further expressed in terms of  $a_i$  and  $b_i$ . Using Lagrange multipliers, it can be shown that the minimum of the variance subject to the

unbiasedness condition is achieved when  $b_i = -a_i$  with  $a_1 = 1/2 + 1/(2N)$  and  $a_i = -1/[2N(N - 1)]$  for  $2 \leq i \leq N$ . Thus, the BLUE of the clock offset admits the form:

$$\hat{\theta}_{\text{BLUE-A}} = \frac{N(U_{(1)} - V_{(1)}) - (\bar{U} - \bar{V})}{2(N - 1)} \tag{15}$$

On the other hand, the BLUE of the clock offset under symmetric exponential delays can also be derived following the aforementioned steps, and it is expressed as:

$$\hat{\theta}_{\text{BLUE-S}} = \frac{U_{(1)} - V_{(1)}}{2}$$

Using statistical signal processing techniques, [31] later re-derived the joint BLUEs for the fixed delay, the clock offset and the mean of exponential delays under symmetric and asymmetric exponential delays, as shown in Equations (16) and (17), respectively.

$$\begin{bmatrix} \hat{d}_{\text{BLUE-S}} \\ \hat{\theta}_{\text{BLUE-S}} \\ \hat{\lambda}_{\text{BLUE-S}} \end{bmatrix} = \frac{1}{2(N - 1)} \begin{bmatrix} N(U_{(1)} + V_{(1)}) - (\bar{U} + \bar{V}) \\ (N - 1)(U_{(1)} - V_{(1)}) \\ N(\bar{U} + \bar{V}) - N(U_{(1)} + V_{(1)}) \end{bmatrix} \tag{16}$$

$$\begin{bmatrix} \hat{d}_{\text{BLUE-A}} \\ \hat{\theta}_{\text{BLUE-A}} \\ \hat{\lambda}_{1\text{BLUE-A}} \\ \hat{\lambda}_{2\text{BLUE-A}} \end{bmatrix} = \frac{1}{2(N - 1)} \begin{bmatrix} N(U_{(1)} + V_{(1)}) - (\bar{U} + \bar{V}) \\ N(U_{(1)} - V_{(1)}) - (\bar{U} - \bar{V}) \\ 2N(\bar{U} - U_{(1)}) \\ 2N(\bar{V} - V_{(1)}) \end{bmatrix} \tag{17}$$

These results were further used as a prerequisite to find the MVUE of the clock offset, a result that will be presented in the next sub-section.

### 4.1.3. Minimum Variance Unbiased Estimator

As discussed earlier, the MVUE is an unbiased estimator that exhibits lower variance than any other unbiased estimator. Finding the MVUE usually requires usage of the concept of sufficient statistics, the Neyman–Fisher factorization theorem and the Rao–Blackwell–Lehmann–Scheffe theorem [23] (p. 101).

Let  $\Phi$  denote the parameters to be estimated and  $\mathbf{z}$  represent the set of the observations; then, the Neyman–Fisher factorization theorem states that if the likelihood function  $p(\mathbf{z}|\Phi)$  can be factorized as:

$$p(\mathbf{z}|\Phi) = g(T(\mathbf{z}), \Phi)h(\mathbf{z}),$$

where  $g$  is a function depending on  $\mathbf{z}$  only through  $T(\mathbf{z})$  and  $h$  is a function depending on  $\mathbf{z}$  only, then  $T(\mathbf{z})$  is a sufficient statistic for  $\Phi$ . In a two-way message exchange mechanism under symmetric exponential delays,  $\Phi = [d, \theta, \lambda]^T$  and  $\mathbf{z} = [U_1, U_2, \dots, U_N, V_1, V_2, \dots, V_N]^T$ . The MVUE of  $\Phi$  was derived in [31] by implementing the following procedure. Express first the likelihood function in terms of the unit step function  $u[\cdot]$  as follows:

$$p(\mathbf{z}|\Phi) = \lambda^{-2N} \exp \left[ -\frac{1}{\lambda} \sum_{i=1}^N (U_i + V_i - 2d) \right] \cdot u[U_{(1)} - d - \theta] \cdot u[V_{(1)} - d + \theta] \tag{18}$$

and factorize Equation (18) as:

$$p(\mathbf{z}|\Phi) = g_1\left(\sum_{i=1}^N (U_i + V_i), d, \lambda\right) \cdot g_2(U_{(1)}, d, \theta) \cdot g_3(V_{(1)}, d, \theta) \cdot h(\mathbf{z})$$

where:

$$g_1\left(\sum_{i=1}^N (U_i + V_i), d, \lambda\right) = \lambda^{-2N} \exp\left[-\frac{1}{\lambda} \sum_{i=1}^N (U_i + V_i - 2d)\right]$$

$$g_2(U_{(1)}, d, \theta) = u[U_{(1)} - d - \theta], \quad g_3(V_{(1)}, d, \theta) = u[V_{(1)} - d + \theta], \quad h(\mathbf{z}) = 1$$

In the above equations,  $h(\mathbf{z})$  is independent of the unknown parameters  $\Phi$  and  $g_1, g_2, g_3$  are functions depending on the data only through  $\mathbf{T} = \{\sum_{i=1}^n (U_i + V_i), U_{(1)}, V_{(1)}\}$ . Using the Neyman–Fisher factorization theorem, it turns out that  $\mathbf{T}$  is a sufficient statistic. Moreover, the Rao–Blackwell–Lehmann–Scheffe theorem (p. 109 in [23]) claims that a sufficient statistic  $\mathbf{T}$  is complete if there is only one function  $c(\cdot)$  of  $\mathbf{T}$  that is unbiased, and this function leads to the MVUE, *i.e.*,  $\Phi_{MVU} = c(\mathbf{T})$ . Therefore, the remaining task is to prove that  $\mathbf{T}$  is complete or, equivalently, that only one function of  $\mathbf{T}$  is unbiased, and to find that function.

The authors in [31] employed a one-to-one transformation of  $\mathbf{T}$ , denoted as  $\mathbf{T}' = \{\sum_{i=1}^N (U_i + V_i - U_{(1)} - V_{(1)}), U_{(1)}, V_{(1)}\}$ . Then, it was shown that  $\mathbf{T}'$  is complete by assuming that there are two functions of  $\mathbf{T}'$  leading to unbiasedness, *i.e.*,  $E(c(\mathbf{T}')) = E(h(\mathbf{T}')) = \Phi$ , and then, proving that, actually,  $c(\mathbf{T}') = h(\mathbf{T}')$ . It turns out that  $\mathbf{T}$  is also complete, since the sufficient statistics are unique within one-to-one transformations [23]. To this end, what remains to prove is to find an unbiased estimator for  $\Phi$  as a function of  $\mathbf{T}$ , which represents the MVUE according to the Rao–Blackwell–Lehmann–Scheffe theorem. It seems difficult to find three unbiased functions of  $\mathbf{T}$  for each of  $d, \theta$  and  $\lambda$  just by inspection. However, it can be observed that the BLUE  $\Phi_{BLUE-S}$  in Equation (16) is a function of  $\mathbf{T}$  and is also unbiased. Therefore, it is concluded that the BLUE is also the MVUE:

$$\hat{\Phi}_{MVUE-S} = \begin{bmatrix} \hat{d}_{MVUE-S} \\ \hat{\theta}_{MVUE-S} \\ \hat{\lambda}_{MVUE-S} \end{bmatrix} = \frac{1}{2(N-1)} \begin{bmatrix} N(U_{(1)} + V_{(1)}) - (\bar{U} + \bar{V}) \\ (N-1)(U_{(1)} - V_{(1)}) \\ N(\bar{U} + \bar{V}) - N(U_{(1)} + V_{(1)}) \end{bmatrix} \tag{19}$$

Following an analogous approach, it was reported in [31] that the MVUE of the clock offset is also equivalent to the BLUE in the presence of asymmetric exponential delays:

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

#### 4.1.4. Comparison of Estimators

A comparison of MLE, BLUE and MVUE under both symmetric and asymmetric exponential delays is shown in Table 1. The criteria adopted in Table 1 include the estimator formula in terms of observations, bias, variance and MSE. It is observed that the BLUE and MVUE coincide in the presence of both symmetric and asymmetric distributions. Furthermore, all three estimators admit the

same expression when the random delays are symmetrically exponential distributed. Under asymmetric exponential delays, albeit unbiased, the MLE can still outperform the MVUE when:

$$\frac{\lambda_1^2 + \lambda_2^2 - \lambda_1 \lambda_2}{2N^2} < \frac{\lambda_1^2 + \lambda_2^2}{4N(N - 1)}$$

or equivalently:

$$N < \frac{2\lambda_1 \lambda_2}{(\lambda_1 - \lambda_2)^2} + 2 \tag{20}$$

It can be seen that the MLE achieves a better performance when the means of the exponential delays are close to each other. In this way, the right-hand side of Equation (20) resumes being a large number, and the inequality Equation (20) generally holds with a reasonable selection of  $N$ .

**Table 1.** Comparison of estimators.

Clock Offset	Symmetric Delays				Asymmetric Delays			
	Formula	Bias	Variance	MSE	Formula	Bias	Variance	MSE
MLE [28,29]	$\frac{U_{(1)} - V_{(1)}}{2}$	0	$\frac{\lambda^2}{2N^2}$	$\frac{\lambda^2}{2N^2}$	$\frac{U_{(1)} - V_{(1)}}{2}$	$\frac{\lambda_1 - \lambda_2}{2N}$	$\frac{\lambda_1^2 + \lambda_2^2}{4N^2}$	$\frac{\lambda_1^2 + \lambda_2^2 - \lambda_1 \lambda_2}{2N^2}$
BLUE [30,31]	$\frac{U_{(1)} - V_{(1)}}{2}$	0	$\frac{\lambda^2}{2N^2}$	$\frac{\lambda^2}{2N^2}$	$\frac{N(U_{(1)} - V_{(1)}) - (\bar{U} - \bar{V})}{2(N-1)}$	0	$\frac{\lambda_1^2 + \lambda_2^2}{4N(N-1)}$	$\frac{\lambda_1^2 + \lambda_2^2}{4N(N-1)}$
MVUE [31]	$\frac{U_{(1)} - V_{(1)}}{2}$	0	$\frac{\lambda^2}{2N^2}$	$\frac{\lambda^2}{2N^2}$	$\frac{N(U_{(1)} - V_{(1)}) - (\bar{U} - \bar{V})}{2(N-1)}$	0	$\frac{\lambda_1^2 + \lambda_2^2}{4N(N-1)}$	$\frac{\lambda_1^2 + \lambda_2^2}{4N(N-1)}$

#### 4.2. Joint Estimation of Clock Offset and Skew under Exponential Delays

In general, the clock synchronization in WSNs involves two steps. The first step assumes synchronizing the nodes by adjusting the clock phase offset (close offset), while the second step resumes correcting the clock frequency offset (clock skew). The clock skew correction plays a critical role in clock synchronization, especially for long-term synchronization, since it reduces the number of exchanged messages and, thus, brings down the energy consumption. Therefore, we discuss the joint estimation of clock offset and skew in this section.

In the literature, based on a two-way message exchange mechanism, the studies for joint estimation of clock offset and skew mainly concentrated on the development of MLEs, and there is no closed-form solution for the joint estimation of clock offset and skew under exponential delays. Roughly speaking, the algorithms for developing MLEs can be categorized into two types: in the first category, a suboptimal model is built by removing some nuisance parameters, and the corresponding MLEs are derived, while in the second category, the joint estimation is conducted directly.

4.2.1. Removing Nuisance Parameters

Based on the two-way message exchange mechanism in Equation (2) and assuming symmetric exponential delays, the likelihood function based on observations  $\{U_i\}_{i=1}^N$  and  $\{V_i\}_{i=1}^N$  is given by:

$$p(\mathbf{T}_i|d, \theta, f, \lambda) = \lambda^{-2N} \exp \left[ -\frac{1}{\lambda} \left( \sum_{i=1}^N \frac{T_i^b - T_i^c}{f} - \sum_{i=1}^N (T_i^a - T_i^d) - 2Nd \right) \right] \times \prod_{i=1}^N I \left[ \frac{T_i^b - \theta}{f} - T_i^a - d \geq 0; T_i^d - \frac{T_i^c - \theta}{f} - d \geq 0 \right] \tag{21}$$

It is observed that the likelihood function Equation (21) assumes a complicated expression, and the MLEs might be difficult to derive. According to this observation, an algorithm for joint estimation of clock offset and skew, referred to as the ML-like estimator (MLL), was proposed in [18] by using the first and last observations of timing message exchanges. Specifically, from the first equation in Equation (2), subtracting  $T_1^b$  from  $T_N^b$  yields:

$$T_N^b - T_1^b = f \cdot (T_N^a - T_1^a + X_N - X_1) \tag{22}$$

Similarly, from the second equation in Equation (2), subtracting  $T_1^c$  from  $T_N^c$  leads to:

$$T_N^c - T_1^c = f \cdot (T_N^d - T_1^d + Y_1 - Y_N) \tag{23}$$

Such an approach helps to remove the explicit dependency on the nuisance parameters  $\lambda$ ,  $d$  and  $\theta$ . In addition, an ML-like estimate  $\hat{f}_{\text{MLL}}$  of the clock skew is derived using Equation (22) and (23). Finding an estimate of the clock offset is achieved by rewriting Equation (2) as:

$$U'_i = d' + \theta + X'_i \\ V'_i = d' - \theta + Y'_i$$

where  $U'_i = T_i^b - T_i^a \hat{f}_{\text{MLL}}$ ,  $V'_i = T_i^d \hat{f}_{\text{MLL}} - T_i^c$ ,  $X'_i = X_i \hat{f}_{\text{MLL}}$ ,  $Y'_i = Y_i \hat{f}_{\text{MLL}}$ , and  $d' = d \hat{f}_{\text{MLL}}$ . It can be seen that the above model shares the same form as the no-skew model in Equation (4). Therefore, an ML-like estimate of the clock offset is obtained  $\hat{\theta}_{\text{MLL}} = (U'_{(1)} - V'_{(1)})/2$ . The above algorithm for joint estimation of clock offset and skew is computationally simple, but it exhibits a suboptimal performance compared to the maximum likelihood estimator, since it exploits a reduced set of statistics. That is the reason why the above estimator is referred to as the ML-like estimator.

Another approach to remove the nuisance parameter was reported in [32] by adding two equations in Equation (2). It follows that:

$$T_i^b + T_i^c = f(T_i^a + T_i^d) + 2\theta + f(X_i - Y_i) \tag{24}$$

where the nuisance parameter  $d$  is removed. Dividing the above expression by  $f$  and defining  $\theta_1 = 1/f$  and  $\theta_2 = \theta/f$  leads to a new model represented in matrix form as follows:

$$\begin{bmatrix} T_1^a + T_1^d \\ \vdots \\ T_N^a + T_N^d \end{bmatrix} = \begin{bmatrix} T_1^b + T_1^c & -2 \\ \vdots & \vdots \\ T_N^b + T_N^c & -2 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} + \begin{bmatrix} Y_1 - X_1 \\ \vdots \\ Y_N - X_N \end{bmatrix} \tag{25}$$

Assuming symmetric exponential delays, it is easy to verify that  $Z_i \triangleq X_i - Y_i$  follows a Laplace distribution with location parameter zero and scale parameter  $1/\lambda$ . Thus, the log-likelihood function for model Equation (25) is expressed as:

$$\ln p(\mathbf{T}_i|\theta_1, \theta_2) = N \ln \frac{\lambda}{2} - \lambda \sum_{i=1}^N |T_i^a + T_i^d - \theta_1(T_i^b + T_i^c) + 2\theta_2| \tag{26}$$

Therefore, the MLEs of  $\theta_1$  and  $\theta_2$  can be found by maximizing Equation (26), and the corresponding estimators for  $\theta$  and  $f$  are obtained using the relationships  $\theta_1 = 1/f$  and  $\theta_2 = \theta/f$ .

#### 4.2.2. Direct Joint Estimation of Clock Offset and Skew

Assuming symmetric exponential delays with a known mean  $\lambda$ , [33] addressed the problem of MLEs for clock offset and skew by optimizing over nonlinear constraints iteratively. In terms of the estimates  $[d, \theta, f]^T$ , four different cases were considered: (1)  $d$  and  $\theta$  known,  $f$  unknown; (2)  $\theta$  known,  $d$  and  $f$  unknown; (3)  $d$  known,  $\theta$  and  $f$  unknown; (4)  $d, \theta, f$  unknown. The ultimate goal is to jointly estimate  $[d, \theta, f]^T$  in the fourth case. However, similar to the derivation of the MLE in [28] when no skew is considered, the MLEs are also determined by exploiting the boundary conditions of the support regions. For Cases 1 and 2, the support regions are 2D, and the discussion of these two cases gives an insight to further analyze Cases 3 and 4, where the support regions admit a 3D visualization.

Li and Jeske [34] proposed a computationally-simpler algorithm to find the MLEs of clock offset and skew. Asymmetric exponential delays with unknown means were assumed in [34], such that the estimation algorithm can be applied to a more general framework. In this case, the likelihood function resumes as:

$$p(\mathbf{T}_i|d, \theta, f, \lambda_1, \lambda_2) = \frac{\lambda_1^N \lambda_2^N}{f^{2N}} \prod_{i=1}^N \exp \left[ -\frac{\lambda_1}{f}(T_i^b - fT_i^a - \theta - fd) - \frac{\lambda_2}{f}(fT_i^d - T_i^c + \theta - fd) \right] \tag{27}$$

$$\times I[\min(T_i^b - fT_i^a - \theta - fd) \geq 0; \min(fT_i^d - T_i^c + \theta - fd) \geq 0]$$

Re-parameterizing in terms of  $\theta_1 = \theta + fd$  and  $\theta_2 = -\theta + fd$  leads to:

$$p(\mathbf{T}_i|f, \theta_1, \theta_2, \lambda_1, \lambda_2) = \frac{\lambda_1^N \lambda_2^N}{f^{2N}} \prod_{i=1}^N \exp \left[ -\frac{\lambda_1}{f}(T_i^b - fT_i^a - \theta_1) - \frac{\lambda_2}{f}(fT_i^d - T_i^c - \theta_2) \right] \tag{28}$$

$$\times I[\min(T_i^b - fT_i^a - \theta_1) \geq 0; \min(fT_i^d - T_i^c - \theta_2) \geq 0]$$

To find the MLEs for likelihood function Equation (27), or equivalently Equation (28), the concept of profile likelihood function is employed in [34], such that the MLEs can be alternatively found by maximizing the profile likelihoods. In particular, fixing  $f, \theta_1$  and  $\theta_2$ , the conditional MLEs of  $\lambda_1$  and  $\lambda_2$  are shown to be:

$$\hat{\lambda}_1(f, \theta_1, \theta_2) = \frac{fN}{\sum_{i=1}^n (T_i^b - fT_i^a - \theta_1)}, \quad \hat{\lambda}_2(f, \theta_1, \theta_2) = \frac{fN}{\sum_{i=1}^n (T_i^d - fT_i^c - \theta_2)}$$

Plugging the above equations into Equation (28) yields the profile likelihood as a function of  $f, \theta_1$  and  $\theta_2$ , which is denoted as  $L_p(f, \theta_1, \theta_2)$ . Following the same procedure with respect to  $L_p(f, \theta_1, \theta_2)$ , the conditional MLEs for  $\theta_1$  and  $\theta_2$  are expressed as:

$$\hat{\theta}_1(f) = \min(T_i^b - fT_i^a), \quad \hat{\theta}_2(f) = \min(fT_i^d - T_i^c)$$

To this end, the original five-dimensional optimization problem Equation (27) is simplified to the maximization of a one-dimensional function  $L_p(f, \min(T_i^b - fT_i^a), \min(fT_i^d - T_i^c))$ , whose solution represents the MLE for clock skew, and it is denoted as  $\hat{f}_{JML}$ . Additionally, the MLE for clock offset is obtained via:

$$\hat{\theta}_{JML} = \frac{\min(T_i^b - \hat{f}_{JML}T_i^a) - \min(\hat{f}_{JML}T_i^d - T_i^c)}{2}$$

since the corresponding estimates of  $\theta_1$  and  $\theta_2$  are  $\min(T_i^b - \hat{f}_{JML}T_i^a)$  and  $\min(\hat{f}_{JML}T_i^d - T_i^c)$ , respectively.

### 4.3. Confidence Interval for Clock Offset

From a statistical point of view, a confidence interval (CI) is used to describe the amount of uncertainty associated with a sample estimate of a parameter. More specifically, if a CI is constructed among a series of experiments, the proportion of such an interval that contains the true value of the parameter is given by the confidence level. Mathematically, suppose  $X$  to be a random sample from a probability distribution with parameter  $\theta$ , a CI is an interval with endpoints  $\{s(X), b(X)\}$ , such that the following property holds:

$$\Pr(s(X) < \theta < b(X)) = \gamma \tag{29}$$

where  $\gamma$  stands for the confidence level.

In terms of the confidence interval for clock offset under exponential delays, Li *et al.* [35] derived satisfactory CIs using an approximate pivotal quantity (APQ). By assuming asymmetric exponential delays, the authors in [35] derived the relative likelihood function for  $\theta$ , say  $\Lambda(\theta)$ , based on the likelihood function from model Equation (4). It was then shown that  $\Lambda(\theta)$  is an asymptotic pivot in the sense that its limiting distribution does not depend on any unknown parameter. Thus, the asymptotic pivot property of  $\Lambda(\theta)$ , as well as its easily invertible form lead to a way to obtain an asymptotic  $100(1 - \alpha)\%$  CI for  $\theta$  as follows:

$$\begin{aligned} s(U_i, V_i) &= \max \left\{ \frac{U_{(1)} - V_{(1)}}{2} - \frac{\bar{U} - U_{(1)}}{2} \left( \frac{1}{\alpha^{\frac{1}{N}}} - 1 \right), -V_{(1)} \right\} \\ b(U_i, V_i) &= \min \left\{ \frac{U_{(1)} - V_{(1)}}{2} + \frac{\bar{V} - V_{(1)}}{2} \left( \frac{1}{\alpha^{\frac{1}{N}}} - 1 \right), -U_{(1)} \right\} \end{aligned} \tag{30}$$

A generalized pivotal quantity (GPQ) was also employed in [35] to propose a generalized confidence interval for  $\theta$ . Even though generalized confidence intervals generally do not have frequentist coverage probabilities that are exactly equal to the nominal value, the proposed GPQ actually leads to frequentist coverage probabilities that are very close to the nominal value. Furthermore, APQ and GPQ CIs under exponential delays were also extended in [35] to any distribution within the one-parameter scale family.

Further discussions about CIs of clock offset were carried out in [36,37]. Specifically, under exponential network delays, [36] explored sequential intervals with a preset width and proved that these CIs provide a satisfactory solution to minimize the number of samples. In this way, sequential CIs can be used to obtain the smallest sample size required to achieve a specified precision for the clock offset estimation. On the other hand, by considering the correlation between the uplink and downlink random



delays, [37] extended the CI procedure to bivariate models where the random delays are formulated as Freund and Marshall–Olkin bivariate exponential delays. In addition, the MLEs for clock offset under Freund and Marshall–Olkin models were also derived.

### 5. Pairwise Clock Synchronization under Unknown Random Delays

Although the assumptions of Gaussian and exponential distributions for the random delay component of WSNs are plausible, it might happen that the underlying PDF of the network random delays is unknown in advance. In fact, it is well known that the distribution of the network random delays is difficult to characterize succinctly [30]. Therefore, the performance of estimators particularly designed for a certain distribution may degenerate greatly under another type of distribution, and there is a need for developing efficient estimation methods that are robust to the unknown random delays of WSNs. Based on the classic two-way message exchange mechanism, we describe several robust estimation methods in this section, namely the bootstrap bias correction [30], the composite particle filtering [38] and the least squares estimator [39].

#### 5.1. Bootstrap Bias Correction

Bootstrap bias correction is a statistical approach that aims to reduce the bias of an estimator at the expense of increased variance, but with a total effect of reduced MSE [40]. For example, the MLE of clock offset derived under exponential delays is given by  $\hat{\theta} = (U_{(1)} - V_{(1)})/2$ . However, the random delays may not follow an exponential distribution exactly, and the MLE derived under the exponential assumption may perform poorly under other distributions. For such scenarios, the bootstrap bias correction approach can be applied to reduce the bias and eventually the MSE of the estimator.

Suppose the observations  $\mathbf{x}$  follow an unknown distribution  $F$  and the estimator is a statistic  $\hat{\theta} = s(\mathbf{x})$ ; the bias of  $\hat{\theta}$  is defined as  $B(\hat{\theta}) = E_F[s(\mathbf{x})] - \theta$ . The bootstrap estimate of the bias can be expressed as:

$$\hat{B}(\hat{\theta}) = E_{\hat{F}}[s(\mathbf{x})] - \hat{\theta} \tag{31}$$

and the bias corrected estimator admits the form:

$$\hat{\theta}_{BC} = \hat{\theta} - \hat{B}(\hat{\theta}) \tag{32}$$

where  $\hat{F}$  denotes the empirical distribution of observations.

In the context of clock offset estimation under exponential delays, the observations between two nodes are denoted as  $\{U_i\}_{i=1}^N$  and  $\{V_i\}_{i=1}^N$ , and the MLE is given by  $\hat{\theta} = (U_{(1)} - V_{(1)})/2$ . The bias of  $\hat{\theta}$  resumes as:

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

where  $F(u)$  and  $G(v)$  stand for the unknown distributions of observations  $\{U_i\}_{i=1}^N$  and  $\{V_i\}_{i=1}^N$ , respectively. The bootstrap estimate of  $B(\hat{\theta})$  can be further expressed as:

$$\hat{B}(\hat{\theta}) = E_{\hat{F}} \left( \frac{U_{(1)} - V_{(1)}}{2} \right) - \hat{\theta} = \frac{1}{2} \left( \int_0^\infty [1 - \hat{F}(u)]^N du - \int_0^\infty [1 - \hat{G}(v)]^N dv \right) - \hat{\theta} \tag{33}$$

Defining  $U_{(0)} = V_{(0)} = 0$  and  $U_{(N+1)} = V_{(N+1)} = \infty$  leads to the empirical distribution [30]:

$$\begin{aligned}
 1 - \hat{F}(u) &= \sum_{i=1}^{N+1} \frac{N-i+1}{N} I[U_{(i-1)} \leq u < U_{(i)}] \\
 1 - \hat{G}(u) &= \sum_{i=1}^{N+1} \frac{N-i+1}{N} I[V_{(i-1)} \leq v < V_{(i)}]
 \end{aligned}
 \tag{34}$$

Plugging Equation (34) back into Equation (33) yields  $\hat{B}(\hat{\theta})$  and eventually the bias-corrected estimator of  $\hat{\theta}$ :

$$\hat{\theta}_{BC} = U_{(1)} - V_{(1)} - \frac{1}{2} \sum_{i=1}^N \left[ \left( \frac{N-i+1}{N} \right)^N - \left( \frac{N-i}{N} \right)^N \right] [U_{(i)} - V_{(i)}]
 \tag{35}$$

Following the same procedure described above, the bias-corrected estimator of the BLUE under asymmetric exponential delays, *i.e.*, Equation (15), was also derived in [30]. Experimental results in [30] showed that under common distribution assumptions other than exponential, *e.g.*, Gamma, Weibull and log-normal, the bias-corrected estimator of BLUE has the smallest MSE, while  $\hat{\theta}_{BC}$  also outperforms the MLE and the BLUE derived for exponential delays.

The work in [41] investigated the BLUE and its corresponding bias-corrected estimator under a Pareto distribution: a heavy-tailed distribution that was adopted to model network delays for recent applications [42–44]. The authors in [41] examined the effectiveness of bootstrap bias correction of different estimators under varying assumptions for network delays. Some interesting examples were reported where bootstrap bias correction fails in the sense that the MSE increases or even the absolute bias increases. For instance, a somewhat surprising result is that the bootstrap bias correction of the exponential BLUE leads to a large absolute bias under Pareto network delays. Among these estimators and their corresponding bootstrap bias corrections, the bias-corrected estimator in Equation (35) was considered to be a significantly more robust estimator in [41].

Moreover, another bias correction method, referred to as the Jackknife bias-correction, was reported in [45]. The Jackknife bias-corrected estimator of the exponential MLE,  $\hat{\theta} = (U_{(1)} - V_{(1)})/2$ , was proposed and compared to the corresponding bootstrap bias-corrected estimator Equation (35). Simulation results in [45] illustrated that the Jackknife bias correction does a better job at reducing bias, but is generally eroded by a variance increase, which yields a larger MSE.

### 5.2. Composite Particle Filtering

The idea of composite particle filtering is to model the estimation problem in the form of a state-space representation and to exploit an optimal Bayesian approach for state estimation [9]. In terms of clock offset estimation, the system model in Equation (4) is rewritten in compact form matrix representation:

$$\underbrace{\begin{bmatrix} U_i \\ V_i \end{bmatrix}}_{\triangleq \mathbf{y}_i} = \underbrace{\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}}_{\triangleq \mathbf{B}} \underbrace{\begin{bmatrix} d \\ \theta \end{bmatrix}}_{\triangleq \mathbf{x}_i} + \underbrace{\begin{bmatrix} X_i \\ Y_i \end{bmatrix}}_{\triangleq \boldsymbol{\omega}_i}
 \tag{36}$$

where  $\omega_i$  can assume any distribution. Since the value of clock offset is time varying, it is assumed to obey a Gauss-Markov dynamic model:

$$\mathbf{x}_i = \mathbf{x}_{i-1} + \boldsymbol{\nu}_{i-1} \tag{37}$$

where the noise term  $\boldsymbol{\nu}_{i-1}$  is modeled as a Gaussian random variable with zero mean and covariance matrix  $\mathbf{Q}_{k-1} = E[\boldsymbol{\nu}_{i-1}\boldsymbol{\nu}_{i-1}^T]$ . The aim is to obtain the minimum mean square error (MSEE) estimator of state  $\mathbf{x}_i$ , which is expressed as the conditional mean state estimator:

$$\hat{\mathbf{x}}_i = E[\mathbf{x}_i | \mathbf{y}_{1:i}] \tag{38}$$

where  $\mathbf{y}_{1:i} = [y_1, \dots, y_i]^T$  stands for the observations up to time  $i$ .

The optimal state estimator consists of two steps. In the first step, the posterior distribution  $p(\mathbf{x}_{i-1} | \mathbf{y}_{1:i-1})$  is updated and supposed known, then the predictive distribution is updated as follows:

$$p(\mathbf{x}_i | \mathbf{y}_{1:i-1}) = \int p(\mathbf{x}_i | \mathbf{x}_{i-1}) p(\mathbf{x}_{i-1} | \mathbf{y}_{1:i-1}) d\mathbf{x}_{i-1} \tag{39}$$

where the transition PDF is obtained from Equation (37). In the second step, the posterior distribution at time  $i$  is evaluated:

$$p(\mathbf{x}_i | \mathbf{y}_{1:i}) = C_i \cdot p(\mathbf{x}_i | \mathbf{y}_{1:i-1}) p(y_i | \mathbf{x}_i) \tag{40}$$

where  $p(y_i | \mathbf{x}_i)$  represents the likelihood function obtained from Equation (36), and:

$$C_i = \left( \int p(\mathbf{x}_i | \mathbf{y}_{1:i}) p(y_i | \mathbf{x}_i) d\mathbf{x}_i \right)^{-1}$$

is a normalization constant.

In the composite particle filtering algorithm adopted in [38], the posterior and predictive distributions are recursively modeled by finite Gaussian mixtures, and the components of the Gaussian mixtures are updated using the Kalman filter in conjunction with particle filters. Computer simulation results illustrate that the performance of the composite particle filter algorithm is superior in the presence of delays with unknown distribution (such as Gamma, Weibull, log-normal or arbitrary mixtures) relative to the performance exhibited by the MLEs derived specifically under the assumption of Gaussian or exponential delays.

### 5.3. Least Squares Estimators

The least squares approach in [39] provides a general framework for joint estimation of clock offset and skew, irrespective of the network delay distribution type. The model adopted by the least squares approach is obtained by adding two equations in Equation (2) together and reduces to:

$$\frac{T_i^b + T_i^c}{2} = \theta + f \frac{T_i^a + T_i^d}{2} + f \frac{X_i - Y_i}{2} \tag{41}$$

Thus, we can regard Equation (41) as a linear regression model in terms of  $\{(T_i^b + T_i^c)/2, (T_i^a + T_i^d)/2\}_{i=1}^N$ , where  $\theta$  represents the intercept,  $f$  stands for the slope and  $f(X_i - Y_i)/2$  denotes

the error term. Based on the model Equation (41),  $\theta$  and  $f$  can be naively estimated via a standard least squares approach as follows:

$$\hat{f}_{\text{LS}} = \frac{\sum_{i=1}^N (T_i^b + T_i^c)(T_i^a + T_i^d - \bar{T}^a - \bar{T}^d)}{\sum_{i=1}^N (T_i^a + T_i^d - \bar{T}^a - \bar{T}^d)^2} \quad (42)$$

and:

$$\hat{\theta}_{\text{LS}} = (\bar{T}^b + \bar{T}^c)/2 - \hat{f}_{\text{LS}}(\bar{T}^a + \bar{T}^d)/2 \quad (43)$$

where  $\bar{T}^j$  for  $j = a, b, c, d$  represent sample means.

However, in general, neither the mean nor the covariance of the error term  $f(X_i - Y_i)/2$  is zero, which may result in poor estimation performance. Therefore, several alternative least squares estimators were proposed in [39] to further improve the estimation accuracy of clock offset and skew. An example of an enhanced least squares estimator is referred to as the constrained least squares (CLS) estimator, and it is found by minimizing the sum of squared residuals subject to a set of constraints on  $\theta$  and  $f$  formed by considering the time stamp model in Equation (2):

$$T_i^b \geq \theta + fT_i^a, \quad T_i^c \leq \theta + fT_i^d \quad (44)$$

The disadvantage of the CLS lies in the challenging computations to solve the minimization problem. Therefore, two additional least squares estimators, referred to as the feasibility checked least squares (FLS) estimator and the Paxson-based estimator, respectively, were proposed in [39] to reduce the computational complexity. Experimental results showed that the FLS estimator and the Paxson-based estimator perform comparably and sometimes better than the CLS estimator, while exhibiting the advantage of much less computational complexity.

## 6. Fully-Distributed Clock Synchronization Algorithms

The centralized signal processing techniques can only help address the problem of pairwise synchronization. As discussed in Section 1, the network-wide synchronization requires a specified network structure to extend the pairwise synchronization approach to all of the sensor nodes. This increases the overhead of the communication and makes the network vulnerable to node failures. More recent works on clock synchronization mainly concentrated on applying decentralized signal processing methods (e.g., distributed estimation techniques) to develop clock synchronization algorithms in a fully-distributed manner.

### 6.1. System Model

In a fully-distributed clock synchronization algorithm, the synchronization is not performed pairwise, but simultaneously among all of the sensor nodes. Therefore, instead of estimating the clock offset difference and the relative clock skew between two nodes, we aim to find the clock offset and skew for each individual sensor node. A two-way message exchange between nodes  $i$  and  $j$  in the  $n$ -th round

of their message exchange is modeled in Figure 4. Following the same message exchange procedure described in Section 2.2, we can write:

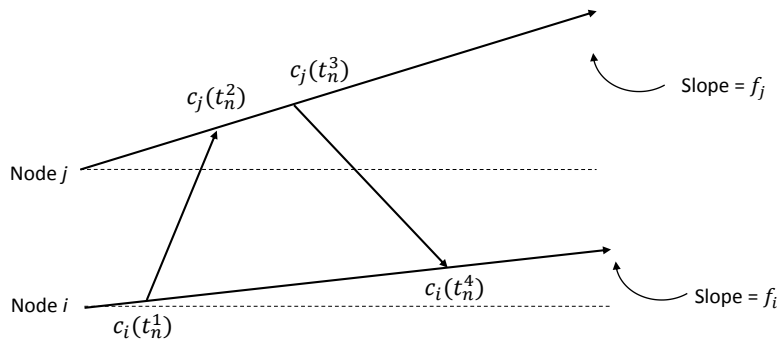
$$\frac{1}{f_j}[c_j(t_n^2) - \theta_j] = \frac{1}{f_i}[c_i(t_n^1) - \theta_i] + d_{ij} + X_{ij} \tag{45}$$

$$\frac{1}{f_j}[c_j(t_n^3) - \theta_j] = \frac{1}{f_i}[c_i(t_n^4) - \theta_i] - d_{ji} - X_{ji} \tag{46}$$

where  $d_{ij}$  and  $d_{ji}$  denote the fixed delays from node  $i$  to node  $j$  and from node  $j$  to node  $i$ , respectively, and  $X_{ij}, X_{ji}$  stand for the random delays. If only the effect of clock offset is considered, *i.e.*,  $f_i = f_j = 1$ , and the network topology is assumed stationary, *i.e.*,  $d_{ij} = d_{ji} = d$ , the system model in Equations (45) and (46) reduces to the more simplified model:

$$c_j(t_n^2) - \theta_j = c_i(t_n^1) - \theta_i + d + X_{ij} \tag{47}$$

$$c_j(t_n^3) - \theta_j = c_i(t_n^4) - \theta_i - d - X_{ji} \tag{48}$$



**Figure 4.** Two-way message exchange between nodes  $i$  and  $j$ .

### 6.2. Fully-Distributed Clock Synchronization Algorithms under Gaussian Delays

The Gaussian assumption for the random delay component of WSNs is justifiable due to the CLT and many experimental results (see, e.g., [22]). The state-of-the-art of fully-distributed clock synchronization algorithms under Gaussian random delays are discussed in the following sub-section.

#### 6.2.1. Belief-Based Synchronization Algorithms

Existing fully-distributed algorithms in [22,46,47] are based on the belief propagation method in the Bayesian framework. Under the assumption of Gaussian random delays, [46] proposed a belief-based algorithm to estimate the clock offset in a distributed manner. Specifically, the system model is modified by adding Equation (47)–(48), and it leads to:

$$T_{\{i,j\},n} \triangleq c_j(t_n^2) + c_j(t_n^3) - c_i(t_n^1) - c_i(t_n^4) = 2\theta_j - 2\theta_i + Z_n \tag{49}$$

where  $Z_n \triangleq X_{ij} - X_{ji}$ . Stacking all of the observations in a vector as  $\mathbf{T}_{i,j} = [T_{\{i,j\},1}, \dots, T_{\{i,j\},N}]$  leads to the likelihood function denoted as  $p(\mathbf{T}_{i,j}|\theta_i, \theta_j)$ . In the Bayesian scenario, the clock offset of

each sensor node, denoted as  $\theta_i$ , is regarded as a random variable with prior distribution  $p(\theta_i)$ . The estimate is obtained by maximizing the posterior distribution of the clock offset.

The posterior distribution of  $\theta_i$  for a network with only two nodes  $i$  and  $j$  is expressed as:

$$g_i(\theta_i) = \int p(\theta_i, \theta_j | \mathbf{T}_{i,j}) d\theta_j \propto \int p(\mathbf{T}_{i,j} | \theta_i, \theta_j) p(\theta_i) p(\theta_j) d\theta_j$$

Extending the above idea to a WSN with  $M$  sensor nodes, the posterior distribution of clock offset  $\theta_i$  is expressed as:

$$g_i(\theta_i) = \int_{\theta_1, \dots, \theta_{i-1}, \theta_{i+1}, \dots, \theta_M} \dots \int p(\theta_1, \dots, \theta_M | \{\mathbf{T}_{ij}\}_{i=1, \dots, M, j \in \beta_i}) d\theta_1 \dots d\theta_{i-1} d\theta_{i+1} d\theta_M \quad (50)$$

where  $\beta_i$  stands for the indices of neighboring sensors of node  $i$ . It can be seen that the joint distribution depends on interactions among all of the variables, and the integration is almost impossible to calculate. Therefore, belief propagation (BP) is employed to compute the posterior distribution without fully integrating Equation (50).

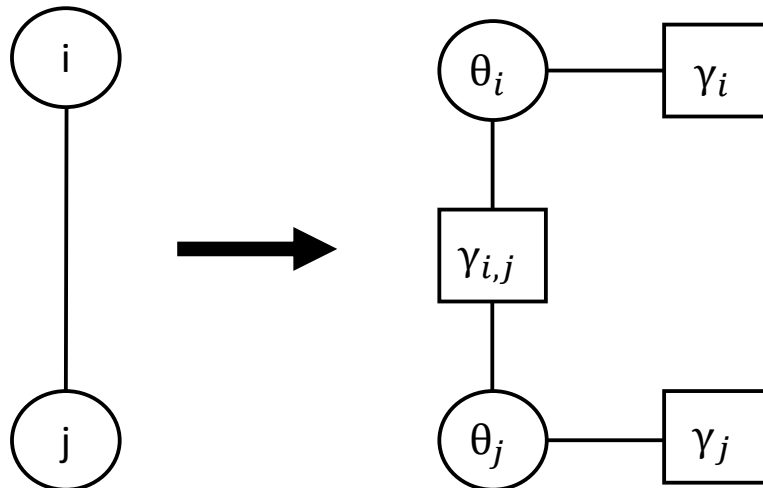
BP takes advantage of the graphical model structure. Factor graphs represent one of the most widely used graphical models and have been adopted in [46]. An example of a factor graph for two nodes  $i$  and  $j$  is depicted in Figure 5. The two sensor nodes  $i$  and  $j$  are represented in terms of the variable nodes (circles), and they are connected to a square factor node  $\gamma_{ij}$ , which stands for the likelihood function  $p(\mathbf{T}_{ij} | \theta_i, \theta_j)$ . On the other hand, nodes  $i$  and  $j$  are also connected to factor nodes  $\gamma_i$  and  $\gamma_j$ , which represent the prior distributions of  $\theta_i$  and  $\theta_j$ , respectively. During each iteration, *i.e.*, for the iteration  $l$ , the BP algorithm in [46] is carried out via the following three steps:

- (1) Each variable node  $\theta_i$  transmits its current belief  $b_i^{(l)}(\theta_i)$  to all of its neighboring factor nodes  $\{\gamma_{ij}, j \in \beta_i\}$ .
- (2) Acting like an intermediate node, the message from a factor node  $\gamma_{ij}$  to a variable node  $\theta_i$  is calculated based on the belief received from  $\theta_j$ :

$$m_{\gamma_{i,j} \rightarrow \theta_i}^{(l)}(\theta_i) = \int p(\mathbf{T}_{i,j} | \theta_i, \theta_j) b_j^{(l)}(\theta_j) d\theta_j \quad (51)$$

- (3) After variable node  $\theta_i$  receives all of the messages from its neighboring factor nodes, *i.e.*,  $\{m_{\gamma_{i,j} \rightarrow \theta_i}^{(l)}(\theta_i)\}_{j \in \beta_i}$ , it updates its belief  $b_i^{(l+1)}(\theta_i)$  as follows:

$$b_i^{(l+1)}(\theta_i) = p(\theta_i) \prod_{j \in \beta_i} m_{\gamma_{i,j} \rightarrow \theta_i}^{(l)}(\theta_i) \quad (52)$$



**Figure 5.** The factor graph of nodes  $i$  and  $j$ .

If no prior information is given about  $p(\theta_i)$ , we set  $p(\theta_i)$  to a Gaussian PDF with zero mean and very large variance. Furthermore, the belief of each node is initially set as its prior distribution, *i.e.*,  $b_i^{(0)}(\theta_i) = p(\theta_i)$ . Beliefs and messages are then iteratively updated at variable nodes and factor nodes, respectively. After convergence, the belief at each node reaches the posterior distribution exactly when the network topology is loop free and approximately when the network topology presents loops [48]. Thus, the maximum *a posteriori* probability (MAP) estimate is obtained by maximizing the converged belief at each node. Since each node runs the same algorithm only with its neighbors, the estimates are achieved in a fully-distributed manner.

However, only the problem of clock offset estimation is investigated in [46]. Moreover, the algorithm is performed in a synchronous way, *i.e.*, every node updates its belief only after receiving messages from all of its neighboring factor nodes. Due to the random packet dropouts, the synchronous algorithm may converge slowly. To account for these aspects, Du *et al.* [47] generalized the belief-based algorithm to jointly estimate the clock offset and skew under Gaussian random delays using both synchronous and asynchronous algorithms. In the asynchronous algorithm, some nodes are allowed to update their beliefs more frequently as long as they receive messages from their neighbors within a preset time period.

Following the two-way message exchange model in Equations (45) and (46), the authors in [47] eliminated the fixed delay  $d$  by adding these two equations together. It turns out that:

$$\mathbf{A}_{j,i}\beta_j + \mathbf{A}_{i,j}\beta_i = z_{j,i} \tag{53}$$

where  $\mathbf{A}_{j,i}$  and  $\mathbf{A}_{i,j}$  are  $N$ -by-two matrices with the  $n$ -th row representing  $[c_j(t_n^2) + c_j(t_n^3), -2]$  and  $-[c_i(t_n^1) + c_i(t_n^4), -2]$ , respectively,  $\beta_j \triangleq [1/f_j, \theta_j/f_j]^T$ ,  $\beta_i \triangleq [1/f_i, \theta_i/f_i]^T$  and  $z_{j,i} = [X_{j,1} - Y_{j,1}, \dots, X_{j,N} - Y_{j,N}]^T$ . Thus, the likelihood function can be obtained from Equation (53), and it is denoted by  $p(\mathbf{A}_{i,j}, \mathbf{A}_{j,i}|\beta_i, \beta_j)$ .

The message passing procedure adopted in [47] is slightly different from the one in [46]. In Step 1, instead of broadcasting the belief to all of its neighboring factor nodes, the message transmitted from variable node  $\theta_i$  to factor node  $\gamma_{ij}$  is evaluated via:

$$m_{\theta_i \rightarrow \gamma_{i,j}}^{(l)}(\beta_i) = \prod_{k \in \beta_i \setminus j} m_{\gamma_{i,k} \rightarrow \theta_i}^{(l-1)}(\beta_i) \tag{54}$$

which is simply the product of the incoming messages on the other links. Thus, in Step 2, the message from a factor node  $\gamma_{ij}$  to a variable node  $\theta_i$  is updated via:

$$m_{\gamma_{i,j} \rightarrow \theta_i}^{(l)}(\beta_i) = \int p(\mathbf{A}_{i,j}, \mathbf{A}_{j,i} | \beta_i, \beta_j) m_{\theta_j \rightarrow \gamma_{i,j}}^{(l)}(\beta_j) d\beta_j \quad (55)$$

and the belief is further updated as:

$$b_i^{(l+1)}(\beta_i) = \prod_{j \in \beta_i} m_{\gamma_{i,j} \rightarrow \theta_i}^{(l)}(\beta_i) \quad (56)$$

It is illustrated analytically in [47] that the asynchronous algorithm converges regardless of the network topology, and the MSE of the estimation parameters achieves the centralized CRLB asymptotically. Simulation results further show that the asynchronous algorithm converges faster than the synchronous one.

In parallel to the work in [47], Etzlinger *et al.* [22] proposed a BP algorithm and a mean field (MF) algorithm and adopted a Bayesian model with Gaussian random delays. The MAP estimates of clock offset and skew are derived using some simplifications of the measurement likelihoods and prior distributions. The message update rule in the BP algorithm is analogous to the one in [47] (see, e.g., Equations (54) and (55)). As concerns the implementation of the MF algorithm, each variable node  $\theta_i$  broadcasts its current belief  $b_i^{(l)}(\theta_i)$  to all of its neighboring factor nodes, and the same step as in [46] is followed. However, in the second step, the message transmitted from a variable node  $\theta_i$  to a factor node  $\gamma_{i,j}$  is alternatively evaluated as:

$$m_{\gamma_{i,j} \rightarrow \theta_i}^{(l)}(\theta_i) = \exp \left( \int \log(p(T_{i,j} | \theta_i, \theta_j)) b_j^{(l)}(\theta_j) d\theta_j \right) \quad (57)$$

Numerical results corroborate the attractive performances exhibited by both BP and MF algorithms.

### 6.2.2. Consensus-Based Synchronization Algorithms

The fully-distributed algorithms reported in [49–51] rely on the average consensus principle. However, the message delays are not considered in these algorithms, and their performance degenerates significantly when message delays exist [46]. Other consensus-based distributed synchronization algorithms, such as [52,53], typically suffer from slow convergence speed and require a high number of data packages [22]. To overcome this issue, Berger *et al.* [54] recently presented a fully-distributed low-complexity consensus-based synchronization algorithm for embedded WSNs. Another fully-distributed clock synchronization method based on a cascade of two consensus algorithms was proposed by Schenato *et al.* [55], which exhibits the advantage of being asynchronous and being robust to packet losses, node failures and communication topology changes. The details of these consensus-based algorithms are omitted herein, since they are not based on the statistical framework described in Section 6.1. Xiong *et al.* [56] generalized the consensus-based algorithm to estimate the clock offset by taking into account both fixed delays, as well as random Gaussian delays. In each iteration of the generalized consensus-based algorithm, each node processes and decodes the timestamps from its neighbors and then updates its local clock time using a weighted average of the time differences with its neighbor nodes. Such a strategy naturally leads to the introduction of a special network topology referred



to as the time delay balanced network in which average timing consensus is achieved in the presence of Gaussian random delays.

### 6.3. Fully-Distributed Clock Synchronization Algorithms under Exponential Delays

Compared to the fully-distributed clock synchronization algorithms that assume Gaussian network delays, fewer works have been proposed to account for exponential network delays in a fully-distributed manner. Considering only the estimation of clock offset, Zennaro *et al.* [57] pioneered a fully-distributed algorithm under exponential delays based on a factor-graph representation of the network and a max-product message update algorithm. From the two-way message exchange model in Equations (47) and (48), since the MLE for the clock offset difference  $\theta_j - \theta_i$  is given by  $S_{ij} = [U_{ij,(1)} - V_{ij,(1)}] / 2$ , where  $U_{ij,(1)}$  and  $V_{ij,(1)}$  denote the first order statistics of  $U_{ij,n} \triangleq c_j(t_n^2) - c_i(t_n^1)$  and  $V_{ij,n} \triangleq c_i(t_n^4) - c_j(t_n^3)$ , respectively, the modified system model is expressed as:

$$S_{ij} = \theta_j - \theta_i + Z_{ij} \tag{58}$$

where  $Z_{ij} \triangleq (X_{ij,(1)} - Y_{ij,(1)}) / 2$  is a Laplace random variable, since  $X_{ij}$  and  $Y_{ij}$  are both assumed to be exponentially distributed. Thus, the likelihood function  $p(S_{ij}|\theta_i, \theta_j)$  is formulated based on the model in Equation (58). The factor graph representation of the network is identical to the one employed in [46]. In terms of the message update rule, the message transmitted from variable node  $\theta_i$  to factor node  $\gamma_{ij}$  is given by:

$$m_{\theta_i \rightarrow \gamma_{i,j}}^{(l)}(\theta_i) = p(\theta_i) \prod_{k \in \beta_i \setminus j} m_{\gamma_{i,k} \rightarrow \theta_i}^{(l-1)}(\theta_i) \tag{59}$$

Moreover, the message from a factor node  $\gamma_{i,j}$  to a variable node  $\theta_i$  is updated by the “max” operator:

$$m_{\gamma_{i,j} \rightarrow \theta_i}(\theta_i) = \max_{\theta_j} [m_{\theta_j \rightarrow \gamma_{i,j}}(\theta_j) p(S_{ij}|\theta_i, \theta_j)] \tag{60}$$

and the belief update is expressed as:

$$b_i^{(l)} = p_i(\theta_i) \prod_{j \in \beta_i} m_{\gamma_{i,j} \rightarrow \theta_i}(\theta_i) \tag{61}$$

Under exponential random delays, [58] proposed a fully-distributed synchronization algorithm for joint estimation of clock offset and skew. The joint maximum likelihood estimation of the clock offset and skew is first formulated as a linear programming (LP) problem and solved in a centralized way. With the help of auxiliary replica variables, the original LP problem is re-formulated as an equivalent optimization problem with the structure more amenable for decomposition, which is further decoupled into a series of subtasks. By applying the classic alternating direction method of multipliers (ADMM), these subtasks are solved in a distributed way while still guaranteeing the convergence to the centralized solution.

## 7. Conclusions and Open Problems

Wireless sensor networks can be applied to a variety of applications, and most of these applications require synchronization among the sensor nodes. Based on the two-way message exchange system

model described in Section 2.2, this paper surveyed the most representative pairwise and fully-distributed clock synchronization algorithms from a statistical signal processing viewpoint. The interested reader is referred to [59–63] for references based on other system models. Specifically, similar to Equation (2), [59] adopted a two-way message exchange mechanism where no random delay is considered and the fixed portion of delays is different for the uplink and downlink (*i.e.*, replace  $d$  in Equation (2) by  $d_{AB}$  and  $d_{BA}$ ). The main result is that while estimating the clock skew between two nodes is possible, it is impossible to determine the clock offset for pairwise synchronization, unless the delays in two-way message communication are symmetric, *i.e.*,  $d_{AB} = d_{BA}$ . The same result was later extended in [60] for the network case. Recently, timestamp-free synchronization algorithms were reported in [61,62] to avoid the overhead brought by timestamp exchanges. The approach employed in [61] is based on conveying implicit timing information in the physical layer via the timing responses from the receiver to the transmitter. The work in [62] avoids timestamp exchanges using the round-trip-time data, which can be measured by a time-to-digital converter in the master node. The clock parameters, such as the offset and skew, as well as the unknown range between the master and slave nodes were experimentally estimated and verified with an accuracy in the order of a nano-second. Without considering the random delays, a fully-distributed network-wide synchronization algorithm was presented in [63] by assuming the constraint that the relative clock offsets in any network loop sum to zero. Recently, Jeske [64] developed a statistical process control (SPC) algorithm that can be used in the context of the two-way message exchange mechanism to detect translations in the delay distribution.

In reflecting on the research problems which have been addressed, we have observed several open research topics that need further investigation. On the one hand, the algorithms presented in this article all assume a line-of-sight transmission. The fixed portion of delays  $d$  herein is actually related to the propagation delay between the transmitter and the receiver nodes. It may occur that some obstacles exist in the signal path, and they might reflect or simply absorb the signal. In this case, the received signal strength and the propagation delay may be significantly affected, and thus, the mathematical system model proposed in this article might not be plausible. To our best knowledge, the clock synchronization problem under non-line-of-sight transmissions is still open. In addition, even though many references that exploit the statistical system model from Section 2.2 have been reported in the literature, more convincing experimental results based on large-scale WSN testbeds are necessary for a more accurate clock synchronization system model. Other possible future research topics lie in the investigation of the convergence rate and computational complexity of the proposed distributed synchronization algorithms. Due to the limited energy of sensors in WSNs, designing new distributed synchronization schemes with reduced energy consumption, low implementation complexity and a fast convergence rate represents another fruitful research direction.

## Acknowledgments

This work was supported by NSF Award CCF-1318338.

## Author Contributions

Xu Wang contributed to draft writing. Daniel Jeske and Erchin Serpedin proposed the framework of the paper and helped to improve the manuscript. All authors read and approved the manuscript.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Durisic, M.P.; Tafa, Z.; Dimic, G.; Milutinovic, V. A survey of military applications of wireless sensor networks. In Proceedings of the IEEE 2012 Mediterranean Conference on Embedded Computing (MECO), Bar, Montenegro, 19–21 June 2012; pp. 196–199.
2. Alemdar, H.; Ersoy, C. Wireless sensor networks for healthcare: A survey. *Comput. Netw.* **2010**, *54*, 2688–2710.
3. Mainwaring, A.; Culler, D.; Polastre, J.; Szewczyk, R.; Anderson, J. Wireless sensor networks for habitat monitoring. In Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, Atlanta, GA, USA, 28 September 2002; pp. 88–97.
4. Corke, P.; Wark, T.; Jurdak, R.; Hu, W.; Valencia, P.; Moore, D. Environmental wireless sensor networks. *IEEE Proc.* **2010**, *98*, 1903–1917.
5. Ye, D.; Gong, D.; Wang, W. Application of wireless sensor networks in environmental monitoring. In Proceedings of the IEEE 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS), Shenzhen, China, 19–20 December 2009; Volume 1, pp. 205–208.
6. Herring, C.; Kaplan, S. Component-based software systems for smart environments. *IEEE Pers. Commun.* **2000**, *7*, 60–61.
7. Gokbayrak, A.B.; Divarci, S.; Urhan, O. Wireless sensor network gateway design for home automation applications. In Proceedings of the IEEE 22nd Signal Processing and Communications Applications Conference (SIU), Trabzon, Turkey, 23–25 April 2014; pp. 1770–1773.
8. Flammini, A.; Ferrari, P.; Marioli, D.; Sisinni, E.; Taroni, A. Wired and wireless sensor networks for industrial applications. *Microelectron. J.* **2009**, *40*, 1322–1336.
9. Wu, Y.C.; Chaudhari, Q.; Serpedin, E. Clock synchronization of wireless sensor networks. *IEEE Signal Process. Mag.* **2011**, *28*, 124–138.
10. Rhee, I.K.; Lee, J.; Kim, J.; Serpedin, E.; Wu, Y.C. Clock synchronization in wireless sensor networks: An overview. *Sensors* **2009**, *9*, 56–85.
11. Mills, D.L. Internet time synchronization: The network time protocol. *IEEE Trans. Commun.* **1991**, *39*, 1482–1493.
12. Ganeriwal, S.; Kumar, R.; Srivastava, M.B. Timing-sync protocol for sensor networks. In Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, Los Angeles, CA, USA, 5–7 November 2003; pp. 138–149.
13. Sichitiu, M.L.; Veerarittiphan, C. Simple, accurate time synchronization for wireless sensor networks. In Proceedings of the IEEE Wireless Communications and Networking (WCNC), New Orleans, LA, USA, 16–20 March 2003; Volume 2, pp. 1266–1273.

14. Van Greunen, J.; Rabaey, J. Lightweight time synchronization for sensor networks. In Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, San Diego, CA, USA, 19 September 2003; pp. 11–19.
15. Maróti, M.; Kusy, B.; Simon, G.; Lédeczi, Á. The flooding time synchronization protocol. In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, MD, USA, 3–5 November 2004; pp. 39–49.
16. Youn, S. A comparison of clock synchronization in wireless sensor networks. *Int. J. Distrib. Sens. Netw.* **2013**, *2013*.
17. Simeone, O.; Spagnolini, U.; Bar-Ness, Y.; Strogatz, S.H. Distributed synchronization in wireless networks. *IEEE Signal Process. Mag.* **2008**, *25*, 81–97.
18. Noh, K.L.; Chaudhari, Q.M.; Serpedin, E.; Suter, B.W. Novel clock phase offset and skew estimation using two-way timing message exchanges for wireless sensor networks. *IEEE Trans. Commun.* **2007**, *55*, 766–777.
19. Ahmad, A.; Noor, A.; Serpedin, E.; Nounou, H.; Nounou, M. On clock offset estimation in wireless sensor networks with Weibull distributed network delays. In Proceedings of the IEEE 20th International Conference on Pattern Recognition (ICPR), Istanbul, Turkey, 23–26 August 2010, 2010; pp. 2322–2325.
20. Papoulis, A. *Random Variable and Stochastic Processes*; McGraw-Hill: New York, NY, USA, 1991.
21. Bovy, C.; Mertodimedjo, H.; Hooghiemstra, G.; Uijterwaal, H.; Van Mieghem, P. Analysis of end-to-end delay measurements in Internet. In Proceedings of the ACM Conference on Passive and Active Measurements (PAM), Fort Collins, CO, USA, 25–26 March 2002.
22. Etzlinger, B.; Wymeersch, H.; Springer, A. Cooperative synchronization in wireless networks. *IEEE Trans. Signal Process.* **2013**, *62*, 2837–2849.
23. Kay, S. *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*; Prentice Hall: Englewood Cliffs, NJ, USA, 1993.
24. Leng, M.; Wu, Y.C. On clock synchronization algorithms for wireless sensor networks under unknown delay. *IEEE Trans. Veh. Technol.* **2010**, *59*, 182–190.
25. Abdel-Ghaffar, H.S. Analysis of synchronization algorithms with time-out control over networks with exponentially symmetric delays. *IEEE Trans. Commun.* **2002**, *50*, 1652–1661.
26. Moon, S.B.; Skelly, P.; Towsley, D. Estimation and removal of clock skew from network delay measurements. In Proceedings of the IEEE 18th Annual Joint Conference of the IEEE Computer and Communications Societies, New York, NY, USA, 21–25 March 1999; Volume 1, pp. 227–234.
27. Paxson, V. On calibrating measurements of packet transit times. *ACM SIGMETRICS Perform. Eval. Rev.* **1998**, *26*, 11–21.
28. Jeske, D.R. On maximum-likelihood estimation of clock offset. *IEEE Trans. Commun.* **2005**, *53*, 53–54.
29. Ahmad, A.; Zennaro, D.; Serpedin, E.; Vangelista, L. A factor graph approach to clock offset estimation in wireless sensor networks. *IEEE Trans. Inf. Theory* **2012**, *58*, 4244–4260.
30. Jeske, D.R.; Sampath, A. Estimation of clock offset using bootstrap bias-correction techniques. *Technometrics* **2003**, *45*, 256–261.

31. Chaudhari, Q.M.; Serpedin, E.; Qaraqe, K. On minimum variance unbiased estimation of clock offset in a two-way message exchange mechanism. *IEEE Trans. Inf. Theory* **2010**, *56*, 2893–2904.
32. Leng, M.; Wu, Y.C. On joint synchronization of clock offset and skew for wireless sensor networks under exponential delay. In Proceedings of the IEEE International Symposium on Circuits and Systems, Paris, France, 30 May–2 June 2010; pp. 461–464.
33. Chaudhari, Q.M.; Serpedin, E.; Qaraqe, K. On maximum likelihood estimation of clock offset and skew in networks with exponential delays. *IEEE Trans. Signal Process.* **2008**, *56*, 1685–1697.
34. Li, J.; Jeske, D.R. Maximum likelihood estimators of clock offset and skew under exponential delays. *Appl. Stoch. Model. Bus. Ind.* **2009**, *25*, 445–459.
35. Li, J.; Jeske, D.R.; Pettyjohn, J. Approximate and generalized pivotal quantities for deriving confidence intervals for the offset between two clocks. *Stat. Methodol.* **2009**, *6*, 97–107.
36. Li, J.; Jeske, D.R. Sequential Fixed Width Confidence Intervals for the Offset Between Two Network Clocks. *Seq. Anal.* **2009**, *28*, 475–487.
37. Pettyjohn, J.; Jeske, D.R.; Li, J. Estimation and Confidence Intervals for Clock Offset in Networks with Bivariate Exponential Delays. *Commun. Stat.-Theory Methods* **2013**, *42*, 1024–1041.
38. Kim, J.S.; Lee, J.; Serpedin, E.; Qaraqe, K. A robust approach for clock offset estimation in wireless sensor networks. *EURASIP J. Adv. Signal Process.* **2010**, *2010*, 19.
39. Pettyjohn, J.S.; Jeske, D.R.; Li, J. Least squares-based estimation of relative clock offset and frequency in sensor networks with high latency. *IEEE Trans. Commun.* **2010**, *58*, 3613–3620.
40. Efron, B.; Tibshirani, R.J. *An Introduction to the Bootstrap*; CRC Press: Boca Raton, FL, USA, 1994.
41. Jeske, D.R.; Chakravartty, A. Effectiveness of bootstrap bias correction in the context of clock offset estimators. *Technometrics* **2006**, *48*, 530–538.
42. Fujimoto, K.; Ata, S.; Murata, M. Playout control for streaming applications by statistical delay analysis. In Proceedings of the IEEE International Conference on Communications (ICC 2001), Helsinki, Finland, 11–14 June 2001; Volume 8, pp. 2337–2342.
43. Loguinov, D.; Radha, H. Effects of channel delays on underflow events of compressed video over the Internet. In Proceedings of the IEEE 2002 International Conference on Image Processing, New York, NY, USA, 22–25 September 2002; Volume 2, pp. II–205–II–208.
44. Ding, L.; Goubran, R.A. Speech quality prediction in VoIP using the extended E-model. In Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM'03), Francisco, CA, USA, 1–5 December 2003; Volume 7, pp. 3974–3978.
45. Jeske, D.R. Jackknife Bias Correction of a Clock Offset Estimator. In *Advances in Mathematical and Statistical Modeling*; Springer: New York, NY, USA, 2008; pp. 245–254.
46. Leng, M.; Wu, Y.C. Distributed clock synchronization for wireless sensor networks using belief propagation. *IEEE Trans. Signal Process.* **2011**, *59*, 5404–5414.
47. Du, J.; Wu, Y.C. Distributed clock skew and offset estimation in wireless sensor networks: Asynchronous algorithm and convergence analysis. *IEEE Trans. Wirel. Commun.* **2013**, *12*, 5908–5917.
48. Yedidia, J.S.; Freeman, W.T.; Weiss, Y. Understanding belief propagation and its generalizations. *Explor. Artif. Intell. New Millenn.* **2003**, *8*, 236–239.

49. Li, Q.; Rus, D. Global clock synchronization in sensor networks. *IEEE Trans. Comput.* **2006**, *55*, 214–226.
50. Giridhar, A.; Kumar, P. Distributed clock synchronization over wireless networks: Algorithms and analysis. In Proceedings of the 2006 45th IEEE Conference on Decision and Control, San Diego, CA, USA, 13–15 December 2006; pp. 4915–4920.
51. Schenato, L.; Gamba, G. A distributed consensus protocol for clock synchronization in wireless sensor network. In Proceedings of the 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, 12–14 December 2007; pp. 2289–2294.
52. Zennaro, D.; Dall’Anese, E.; Erseghe, T.; Vangelista, L. Fast clock synchronization in wireless sensor networks via ADMM-based consensus. In Proceedings of the IEEE 2011 International Symposium on Modeling and Optimization in Mobile, *Ad Hoc* and Wireless Networks (WiOpt), Princeton, NJ, USA, 9–13 May 2011; pp. 148–153.
53. Maggs, M.K.; O’Keefe, S.G.; Thiel, D.V. Consensus clock synchronization for wireless sensor networks. *IEEE Sens. J.* **2012**, *12*, 2269–2277.
54. Berger, A.; Pichler, M.; Klinglmayr, J.; Potsch, A.; Springer, A. Low-Complex Synchronization Algorithms for Embedded Wireless Sensor Networks. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 1032–1042.
55. Schenato, L.; Fiorentin, F. Average TimeSynch: A consensus-based protocol for clock synchronization in wireless sensor networks. *Automatica* **2011**, *47*, 1878–1886.
56. Gang, X.; Shalinee, K. Analysis of distributed consensus time synchronization with Gaussian delay over wireless sensor networks. *EURASIP J. Wirel. Commun. Netw.* **2009**, *2009*, doi:10.1155/2009/528161.
57. Zennaro, D.; Ahmad, A.; Vangelista, L.; Serpedin, E.; Nounou, H.; Nounou, M. Network-wide clock synchronization via message passing with exponentially distributed link delays. *IEEE Trans. Commun.* **2013**, *61*, 2012–2024.
58. Luo, B.; Cheng, L.; Wu, Y.C. Fully-distributed joint clock synchronization and ranging in wireless sensor networks under exponential delays. In Proceedings of the 2014 IEEE International Conference on Communication Systems (ICCS), Macau, China, 19–21 November 2014; pp. 152–156.
59. Graham, S.; Kumar, P. Time in general-purpose control systems: The control time protocol and an experimental evaluation. In Proceedings of the 43rd IEEE Conference on Decision and Control (2004 CDC), Paadise Island, Bahamas, 14–17 December 2004; Volume 4, pp. 4004–4009.
60. Freris, N.M.; Graham, S.R.; Kumar, P. Fundamental limits on synchronizing clocks over networks. *IEEE Trans. Autom. Control* **2011**, *56*, 1352–1364.
61. Brown, D.R.; Klein, A.G. Timestamp-free network synchronization with random pairwise message exchanges. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 5754–5758.
62. Dwivedi, S.; de Angelis, A.; Zachariah, D.; Handel, P. Joint Ranging and Clock Parameter Estimation by Wireless Round Trip Time Measurements. *IEEE Trans. Sel. Areas Commun.* **2015**, *99*, doi:10.1109/JSAC.2015.2430521.

63. Solis, R.; Borkar, V.S.; Kumar, P. A new distributed time synchronization protocol for multihop wireless networks. In Proceedings of the 2006 45th IEEE Conference on Decision and Control, San Diego, CA, USA, 13–15 December 2006; pp. 2734–2739.
64. Jeske, D.R. CUSUM algorithm for detecting translations in gamma distributions. *Qual. Reliab. Eng. Int.* **2015**, *31*, doi:10.1002/qre.1827.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).