

# Joint Estimation of Clock Skew and Offset in Pairwise Broadcast Synchronization Mechanism

Xuanyu Cao<sup>1,2</sup>, Feng Yang<sup>1</sup>, Xiaoying Gan<sup>1</sup>, Jing Liu<sup>1</sup>, Liang Qian<sup>1</sup>, Xiaohua Tian<sup>1</sup> and Xinbing Wang<sup>1,3</sup>

1. Department of Electronic Engineering, Shanghai Jiao Tong University, China

2. State Key Laboratory of Integrated Services Networks, Xidian University, China

3. National Mobile Communications Research Laboratory, Southeast University, China

Email: {apogne, fengyang, ganxiaoying, jingliu\_lj, lqian, xtian, xwang8}@sjtu.edu.cn

**Abstract**—The problem of jointly estimating clock skew and offset for wireless sensor networks (WSNs) in a pairwise broadcast synchronization (PBS) protocol is considered. The random part of the delay is supposed to be an exponential random variable. We consider two estimators, i.e., joint maximum-likelihood estimator (JMLE) and generalized ML-like estimator (GMLLE) proposed by Leng and Wu [15]. For both estimators, the corresponding algorithms are explicitly derived and presented. For the GMLLE, the corresponding performance bound based on the reduced set of observations is derived and the optimal value of a user-defined parameter is identified accordingly. At last, analytical results are corroborated by numerical experiments. We observe that: (i) JMLE usually outperforms GMLLE at the cost of larger computational complexity; (ii) JMLE, while achieving the same estimation accuracy as that of the LP method presented in [23], enjoys significantly lower computational complexity than that of the latter.

**Index Terms**—Clock synchronization, Pairwise broadcast synchronization (PBS), Wireless sensor networks (WSNs).

## I. INTRODUCTION

Wireless Sensor Networks (WSNs) have a large number of critical applications such as industrial production, battlefield reconnoitring and health monitoring, etc., [1], [2] and [3]. Many of these applications require clock synchronization for transmission scheduling or data fusion. In fact, for a wireless sensor node, approximately 17% of its total energy consumption is spent on clock synchronization according to [4]. Since WSNs generally consist of many small sensors whose energy and spectrum resource are quite limited, energy efficient synchronization protocols are highly desirable. The energy consumption for synchronization is mainly composed of two parts: (1) the communication between those wireless nodes, (2) the computation needed for implementing the synchronization algorithms. Thus, to save energy, we need both accurate (to reduce necessary number of communications) and low complexity (to reduce computation related energy cost) synchronization protocols. However, there usually exists a tradeoff between these two aspects, for better accuracy often means higher computational complexity. But, as is reported by Pottie et al. [5] that the energy required to transmit 1 bit over 100 meters is equivalent to that of executing 3 million computer instructions, we usually incline to use high-accuracy synchronization mechanisms at the cost of larger

computational complexity unless the computational ability of the wireless sensor is extremely limited.

The clock synchronization in WSNs has been extensively studied in recent years and many protocols and algorithms have been proposed. Here, we just concentrate on the most relevant works, i.e., those which study the clock synchronization in a statistical signal processing point of view. Freris et al. [6] have pointed out that not all clock parameters can be determined with absolute accuracy even in a perfect situation and hence, to some extent, explain the difficulty in synchronization for WSNs. In real circumstances, the situation is even worse since the link delay between two wireless nodes usually contains a nondeterministic part [7]. Many distributions such as Gaussian, exponential, Gamma and Weibull, have been proposed to model the random part of the link delay [8]. Thanks to those models, a statistical signal processing framework of clock synchronization is formed and many previous papers have followed this line. The recent synchronization mechanisms can be roughly divided into two categories: (1) point to point synchronization protocols such as two-way message exchange mechanism and one-way message mechanism, (2) broadcast synchronization protocols such as pairwise broadcast synchronization (PBS) and reference broadcast synchronization (RBS) [9].

In a two-way message exchange mechanism, Jeske [10] derives the MLE for clock offset by assuming an exponential random delay model. Chaudhari et al. [11] derive the minimum variance unbiased estimator (MVUE) of clock offset under both symmetric and asymmetric exponential delay model. Recently, Ahmad et al. [12] study the time-varying clock offset estimation by using the factor graph theory and consider a class of delay models. Since energy is the major concern for designing synchronization protocols, people always wish to achieve long term synchronization which can make the WSNs spend less energy on repeating transmitting synchronization signals. Besides, it turns out that the rate (speed) of the clock is not exactly the same for every sensor [13], i.e., a clock skew exists. Consequently, to achieve long term synchronization, estimation of clock skew is necessary and even more important than the estimation of clock offset. For this, Noh et al. [14] study the estimation of clock skew and offset in two-way message exchange mechanism. Then Leng and Wu [15] propose three estimators for clock skew and offset by

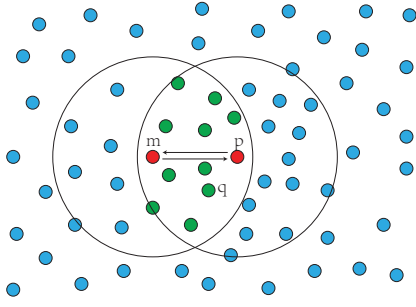


Fig. 1. PBS: When the master node  $m$  and a node  $p$  are adopting the two-way message exchange synchronization mechanism, inactive nodes located in the common transmission range of  $m$  and  $p$  (the green ones) like node  $q$  can also overhear the synchronization messages and thus estimate its clock skew and offset.

assuming a Gaussian delay model and an unknown fixed delay. Maximum likelihood estimation (MLE) is known as a good estimator in estimation theory [16] because it is asymptotically unbiased and its covariance approaches to the Cramer-Rao lower bound (CRLB) as the number of observations goes to infinity. As a result, Chaudhari et al. [17] derive the joint MLE (JMLE) of clock skew and offset under the exponential delay model. Later on, Leng and Wu [18] design a low complexity JMLE algorithm based on novel geometric analysis of the support region formed by the exponential delay model. A MLE based on reduced set of parameters is proposed and the corresponding approximate CRLB is analyzed in [19].

However, there is a big disadvantage for the two-way message exchange synchronization mechanism: the synchronization is just point-to-point and the protocol does not exploit the broadcast nature of wireless media. This drawback essentially leads to large communication overhead. Specifically, when two sensors are adopting a two-way message exchange synchronization mechanism, the sensors located in the common transmission range of these two nodes can also overhear the messages (Fig.1) and should be able to synchronize its own clock to some extent. Hence, a large number of inactive sensors can be synchronized at zero communication energy usage and this will save lots of energy for WSNs. The above is just the basic idea of PBS which is first proposed by Noh et al. [20] under a Gaussian delay model. Afterwards, Chaudhari et al. [21] apply PBS to an exponential delay model and derive the MVUE and MLE of clock offset (actually MLE is shown to coincide with MVUE in this scenario). Later on, Chaudhari et al. [22] also take the asymmetric exponential delay into consideration.

As previous works have shown, PBS and joint estimation of clock skew and offset are two important ways of saving energy for synchronization in WSNs. However, to the best of our knowledge, few works have combined these two methods under an exponential delay model. The reason may be that the analysis is challenging: PBS involves one more node in the analysis procedure compared to two-way message exchange mechanism and the introduction of clock skew usually makes the problem much more complicated for exponential delay model as previous works have shown [17]. Ahmad et al. [23] takes the first step to estimate clock skew in PBS mechanism

and convert the related estimation problem into a linear programming (LP). However, [23] has its limitation: standard LP algorithm (e.g., Newton's method) is carried out through iterations and the LP therein is related with five parameters, which results in high computational complexity. Hence, the LP method in [23] may not be suitable for practical cheap wireless sensors, whose computational ability is quite limited. In this paper, we consider joint estimation of clock skew and offset by invoking PBS. We adopt the exponential delay model which can represent the M/M/1 queue link delay better than Gaussian delay model [24]. Although the derivation for Gaussian delay model is usually much simpler than exponential delay model, the latter turns out to be more suitable for practical WSNs. The primary contributions of this paper are summarized as follows.

- We propose a low complexity JMLE algorithm. The algorithm is explicitly described and its computational complexity is shown to be  $\mathcal{O}(N^3)$  ( $N$  is the number of the rounds involved in the synchronization) in the worst case. Exploiting the special structure of the formulated optimization problem, we compress the number of parameters to two in the final search problem. So, compared with the LP based method involved with five parameters in [23], the proposed algorithm possesses significantly lower complexity. Without the need for iteration, the proposed JMLE algorithm always finds out the exact ML point, which is a widely accepted good estimator in the literature [16]. Hence, the performance of the proposed JMLE is well guaranteed.
- Given the limited computational capability of real sensor nodes, we further endeavor to find estimators with even lower complexity. A generalized ML-like estimator (GM-LLE) proposed by Leng and Wu [15] is invoked herein. Although the idea remains analogous, the algorithm and the performance bound derivation are totally different since (i) we adopt PBS here while [15] adopts a two-way message exchange mechanism, (ii) more importantly, we assume an exponential delay model rather than a Gaussian delay model used in [15] and this makes our analysis much more challenging. Note that the GMLLE here is also different from the algorithm in [19]: (i) the way to reduce the parameter set is different and the presented GMLLE depends on a user defined parameter  $k$ , (ii) the CRLB-like performance bound derived in [19] is only an approximation. In contrast, we derive the exact CRLB-like performance bound<sup>1</sup> and identify the optimal value for  $k$  accordingly. In addition, we show that the computational complexity of GMLLE is  $\mathcal{O}((N - k)^2 + I(N - k))$ , where  $I$  is the number of iterations and  $I = 3$  generally suffices to guarantee convergence. Therefore, we analytically justify that the computational complexity of GMLLE is much smaller than that of JMLE.

<sup>1</sup>We notice that the derived CRLB-like performance bound can only serve as performance benchmark for GMLLE and does not apply to JMLE since it is based on the reduced set of observations. Hence, throughout this paper, so as to avoid misunderstanding, we call this bound the *performance bound* (or *PB* for short) for GMLLE rather than CRLB.

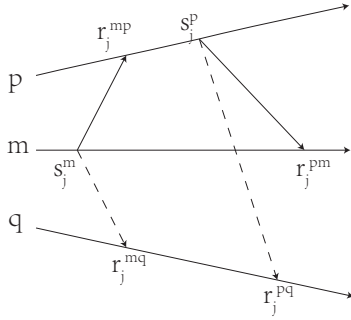


Fig. 2. PBS: Node  $m$  and node  $p$  are synchronizing by using a two-way message exchange mechanism while the inactive node  $q$  is overhearing to synchronize.

- Last but not least, numerical experiments are implemented to corroborate the theoretical analysis. The optimality of the theoretically chosen value of the user-defined parameter  $k$  for GMLLE is confirmed via simulations. We observe that JMLE usually outperforms GMLLE in terms of accuracy. However, the average CPU time for implementing JMLE algorithm is about twice that of GMLLE, which verifies that GMLLE has lower complexity than JMLE. The LP method in [23], though having the same accuracy as JMLE, consumes much more (more than 100 times of) CPU time than that of JMLE and GMLLE.

The rest of this paper is organized as follows. In Section II, we describe the system model and formulate the problem. In Section III, we derive the JMLE algorithm. In Section IV, we present the GMLLE algorithm and derive the corresponding performance bound, based on which we identify the optimal value of a user-defined parameter  $k$ . In Section V, we show the simulation results. In Section VI, we conclude this paper.

## II. STATEMENT OF THE PROBLEM

Fig.1 shows a WSN consisting of some sensor nodes. Node  $m$  is selected as the reference node whose clock is considered to be absolutely accurate. Node  $m$  and node  $p$  are synchronizing via a two-way message exchange mechanism. The procedure for the  $j$ -th round exchange is shown in Fig.2 and depicted as follows. Node  $m$  transmits a synchronization signal with time stamp  $s_j^m$ . Then node  $p$  receives this signal and gives it a time stamp  $r_j^{mp}$  according to its own clock. Afterwards, node  $p$  transmits a synchronization signal back to node  $m$  at time  $s_j^p$  of its own clock. Then node  $m$  receives the signal and records the time as a time stamp of  $r_j^{pm}$ . Thus the  $j$ -th round is finished. As shown in Fig.1, there may exist several silent sensor nodes located in the common transmission range of node  $m$  and  $p$ . Suppose that node  $q$  is one of such nodes. Hence, when node  $m$  and  $p$  are exchanging messages, node  $q$  also overhears them and performs some synchronization accordingly. Since node  $q$  is inactive, i.e., node  $q$  does not need to pay any energy for communication, this synchronization mechanism (PBS) turns out to be much more energy efficient than other point-to-point mechanisms. Specifically, as Fig.1 illustrates, node  $q$  will receive the message sent by  $m$  and  $p$  and give them time stamps  $r_j^{mq}$  and  $r_j^{pq}$  respectively. Note that

$q$  will also know the value of  $r_j^{mp}$  since node  $p$  is required to send this information back to node  $m$  in a two-way message exchange mechanism. In sum, node  $q$  will get the value of  $\{s_j^m, s_j^p, r_j^{mp}, r_j^{mq}, r_j^{pq}\}$ .

As for the clock model, different from [21], we assume that there exist clock skews for node  $p$  and node  $q$  in this paper. Denote the clock rate for node  $p$  and  $q$  as  $\omega_p$  and  $\omega_q$  respectively. The clock offsets certainly still exist and we denote them as  $\phi_p$  and  $\phi_q$  for node  $p$  and  $q$  respectively. Hence, if the clock time of node  $m$  is  $t$ , then the clock time of node  $p$  and  $q$  are  $\omega_p t + \phi_p$  and  $\omega_q t + \phi_q$ , respectively. Experiment results have shown that the link delay consists of two parts: deterministic delay and nondeterministic delay [7]. In this paper, we assume that the deterministic part of the delay is  $d$  for all links. We further assume that the nondeterministic part of delay is an exponential random variable with mean  $\alpha$ . We exploit symmetric delay model, i.e., the mean of the nondeterministic part of delay for all links is the same. Here, in order to capture the reality of WSNs, we assume that  $d$  and  $\alpha$  are fixed but unknown. A justification of this delay model can be found in many previous works such as [9], [15], [21] and [22].

Now, for node  $q$ , the model depicted above can be summarized by the following equations, where  $1 \leq j \leq N$  [23]:

$$r_j^{mp} = \omega_p(s_j^m + d + z_j^{mp}) + \phi_p, \quad (1)$$

$$r_j^{mq} = \omega_q(s_j^m + d + z_j^{mq}) + \phi_q, \quad (2)$$

$$r_j^{pq} = \omega_q \left( \frac{s_j^p - \phi_p}{\omega_p} + d + z_j^{pq} \right) + \phi_q. \quad (3)$$

In the above equations,  $z_j^{mp}, z_j^{mq}, z_j^{pq}$  are nondeterministic part of link delays, which are assumed to be i.i.d. exponential random variables with common mean  $\alpha$ . Compared with [21] and [22], due to the introduction of clock skew, the observations as well as parameters can not be perfectly separated and hence the MVUE is hard to find. Compared with [17], to derive JMLE, we now have two more parameters  $\omega_q$  and  $\phi_q$ , which make the dimension of the involved parameters bigger and the situation more complicated. However, if we can estimate the related parameters accurately, significant reduction of energy cost is expected as explained in Section I, which motivates us to search for good estimators in this scenario. The objective of this work is to estimate  $\omega_q, \omega_p, \phi_q, \phi_p$  accurately with low computational complexity. In the following, we will present two estimation algorithms and analyze the corresponding performance.

## III. JOINT MAXIMUM LIKELIHOOD ESTIMATOR (JMLE)

In this section, we derive the algorithm for the JMLE. Rewriting Eqn. (1), (2) and (3), we obtain:

$$\frac{r_j^{mp} - \phi_p}{\omega_p} - d - s_j^m = z_j^{mp}, \quad (4)$$

$$\frac{r_j^{mq} - \phi_q}{\omega_q} - d - s_j^m = z_j^{mq}, \quad (5)$$

$$\frac{r_j^{pq} - \phi_q}{\omega_q} - d - \frac{s_j^p - \phi_p}{\omega_p} = z_j^{pq}. \quad (6)$$

Since  $z_j^{mp}, z_j^{mq}, z_j^{pq}$  are i.i.d. exponential delay with mean  $\alpha$ , the likelihood function can be written as follows [23]:

$$L(\alpha, d, \phi_p, \phi_q, \omega_p, \omega_q) = \alpha^{-3N} \exp \left[ -\frac{1}{\alpha} \sum_{j=1}^N \left( \frac{r_j^{mp}}{\omega_p} + \frac{r_j^{mq} - \phi_q}{\omega_q} + \frac{r_j^{pq} - \phi_q}{\omega_q} - \frac{s_j^p}{\omega_p} - 3d - 2s_j^m \right) \right] \times \prod_{j=1}^N \left[ u \left( \frac{r_j^{mp} - \phi_p}{\omega_p} - d - s_j^m \right) u \left( \frac{r_j^{mq} - \phi_q}{\omega_q} - d - s_j^m \right) u \left( \frac{r_j^{pq} - \phi_q}{\omega_q} - d - \frac{s_j^p - \phi_p}{\omega_p} \right) \right]$$

Here  $u(\cdot)$  stands for the unit step function, i.e.,  $u(x) = 1$  if  $x > 0$ ,  $u(x) = 0$  if  $x < 0$ . Denote the sum inside exp as  $\xi$ , i.e.,  $\xi \triangleq \sum_{j=1}^N \left( \frac{r_j^{mp}}{\omega_p} + \frac{r_j^{mq} - \phi_q}{\omega_q} + \frac{r_j^{pq} - \phi_q}{\omega_q} - \frac{s_j^p}{\omega_p} - 3d - 2s_j^m \right)$ .

Define  $\lambda \triangleq 1/\alpha$ . When the support condition is satisfied, i.e., the quantities in the unit step function are all positive, we have  $L = \lambda^{3N} e^{-\xi\lambda}$ . Suppose parameters other than  $\lambda$  have already been given, then the  $\hat{\lambda}$  that maximizes  $L$  must satisfy  $\partial L / \partial \lambda \Big|_{\lambda=\hat{\lambda}} = 0$ . So we let  $\frac{\partial L}{\partial \lambda} = \lambda^{3N-1} e^{-\xi\lambda} (3N - \xi) = 0$ . This leads to  $\hat{\lambda} = 3N/\xi$  which is equivalent to  $\hat{\alpha} = \xi/(3N)$ . Substituting this condition back into the expression of likelihood function  $L$ , when the support condition is satisfied, we get  $L \propto \xi^{-3N}$ . Note that the sum  $\sum_{j=1}^N s_j^m$  is a constant in the expression of  $\xi$ . Therefore, in order to maximize  $L$ , the problem becomes to minimize the following expression

$$\tilde{\xi} \triangleq -3dN + \frac{1}{\omega_p} \sum_{j=1}^N (r_j^{mp} - s_j^p) + \frac{1}{\omega_q} \sum_{j=1}^N (r_j^{mq} + r_j^{pq} - 2\phi_q), \quad (7)$$

subject to the following constraints:

$$\omega_p > 0, \omega_q > 0, \quad (8)$$

$$d \leq \frac{r_j^{mp} - \phi_p}{\omega_p} - s_j^m \quad 1 \leq j \leq N, \quad (9)$$

$$d \leq \frac{r_j^{mq} - \phi_q}{\omega_q} - s_j^m \quad 1 \leq j \leq N, \quad (10)$$

$$d \leq \frac{r_j^{pq} - \phi_q}{\omega_q} - \frac{s_j^p - \phi_p}{\omega_p} \quad 1 \leq j \leq N. \quad (11)$$

The constraints (9) and (11) can be written as follows for  $1 \leq j \leq N, 1 \leq l \leq N$ :

$$r_j^{mp} - \omega_p s_j^m - \omega_p d \geq \phi_p \geq \omega_p d + s_l^p - \frac{\omega_p}{\omega_q} (r_l^{pq} - \phi_q). \quad (12)$$

Note that  $\phi_p$  only exists in constraints (9) and (11) and its range is from  $-\infty$  to  $+\infty$ . Besides, the value of  $\tilde{\xi}$  does not depend on  $\phi_p$ . Hence, (9) and (11) can be substituted by the following constraints for  $1 \leq j \leq N, 1 \leq l \leq N$ :

$$r_j^{mp} - \omega_p s_j^m - \omega_p d \geq \omega_p d + s_l^p - \frac{\omega_p}{\omega_q} (r_l^{pq} - \phi_q). \quad (13)$$

The reason of this substitution is that as long as (13) is satisfied, we can always find a  $\phi_p$  which satisfies the constraints (9) and (11) and the value of this  $\phi_p$  will not affect the value of the objective function  $\tilde{\xi}$ , which means that we can arbitrarily choose a value for  $\hat{\phi}_p$  as long as the constraint (13) is satisfied.

Now, except constraint (8), only constraints (10) and (13) exist. Rewriting (10) and (13), we get:

$$\omega_q d + \phi_q \leq r_j^{mq} - \omega_q s_j^m, \quad (14)$$

$$2\omega_q d + \phi_q \leq \frac{\omega_q}{\omega_p} (r_j^{mp} - s_l^p) + r_l^{pq} - s_j^m \omega_q, \quad (15)$$

where  $1 \leq j, l \leq N$ . Define  $f(\omega_p, \omega_q) \triangleq \min_{1 \leq j \leq N} \{r_j^{mq} - \omega_q s_j^m\}$  and  $g(\omega_p, \omega_q) \triangleq \min_{1 \leq j, l \leq N} \{(\omega_q/\omega_p)(r_j^{mp} - s_l^p) + r_l^{pq} - s_j^m \omega_q\}$ . Then (14) and (15) are equivalent to the following two constraints:

$$\omega_q d + \phi_q \leq f(\omega_p, \omega_q), \quad 2\omega_q d + \phi_q \leq g(\omega_p, \omega_q). \quad (16)$$

Note that the value of  $f$  and  $g$  does not depend on  $d$  and  $\phi_q$ . Hence, the only constraints on  $d$  and  $\phi_q$  is (16), which is a linear constraint with respect to  $d$  and  $\phi_q$ . We further notice that the objective function  $\tilde{\xi}$  in Eqn. (7) is just a linear function in terms of  $d$  and  $\phi_q$ . Hence we rewrite Eqn. (7) as follows:

$$\tilde{\xi} = -\frac{2N}{\omega_q} \left( \frac{3\omega_q}{2} d + \phi_q \right) + \frac{1}{\omega_p} \sum_{j=1}^N (r_j^{mp} - s_j^p) + \frac{1}{\omega_q} \sum_{j=1}^N (r_j^{mq} + r_j^{pq}) \quad (17)$$

$$\geq -\frac{N}{\omega_q} (f(\omega_p, \omega_q) + g(\omega_p, \omega_q)) + \frac{1}{\omega_p} \sum_{j=1}^N (r_j^{mp} - s_j^p) + \frac{1}{\omega_q} \sum_{j=1}^N (r_j^{mq} + r_j^{pq}). \quad (18)$$

The reason for (18) is that  $\frac{3\omega_q}{2} d + \phi_q = \frac{1}{2} [(2\omega_q d + \phi_q) + (\omega_q d + \phi_q)]$ . Applying (16), we immediately obtain (18) and the equality in (18) holds iff both conditions in (16) hold with equality, i.e.,

$$d = \frac{g(\omega_p, \omega_q) - f(\omega_p, \omega_q)}{\omega_q}, \quad \phi_q = 2f(\omega_p, \omega_q) - g(\omega_p, \omega_q). \quad (19)$$

Denote the following expressions:

$$\tilde{f}(\omega_p, \omega_q) = \frac{f(\omega_p, \omega_q)}{\omega_q} = \min_{1 \leq j \leq N} \left\{ \frac{r_j^{mq}}{\omega_q} - s_j^m \right\}, \quad (20)$$

$$\tilde{g}(\omega_p, \omega_q) = \frac{g(\omega_p, \omega_q)}{\omega_q} = \min_{1 \leq j, l \leq N} \left\{ \frac{r_j^{mp} - s_l^p}{\omega_p} + \frac{r_l^{pq}}{\omega_q} - s_j^m \right\} = \min_{1 \leq j \leq N} \left\{ \frac{r_j^{mp}}{\omega_p} - s_j^m \right\} + \min_{1 \leq l \leq N} \left\{ -\frac{s_l^p}{\omega_p} + \frac{r_l^{pq}}{\omega_q} \right\}. \quad (21)$$

Suppose the optimal value for  $d$  and  $\phi_q$  have already been chosen as in Eqn. (19). Then the inequality in (18) becomes equality and we can rewrite (18) as follows

$$\tilde{\xi} = -N(\tilde{f}(\omega_p, \omega_q) + \tilde{g}(\omega_p, \omega_q)) + \frac{1}{\omega_p} \sum_{j=1}^N (r_j^{mp} - s_j^p) + \frac{1}{\omega_q} \sum_{j=1}^N (r_j^{mq} + r_j^{pq}). \quad (22)$$

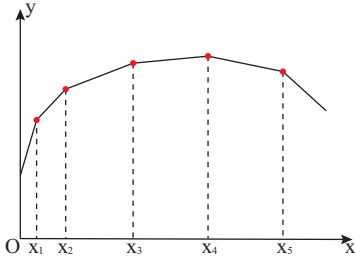


Fig. 3. An illustration for the curve of general expression  $y = \min_{1 \leq j \leq N} \{a_j + b_j x\}$ . The curve is divided into several line segments by some turning points  $x_i$ . The skews of the line segments should decrease as  $x$  increases.

So far the estimation problem has been reduced to the problem of minimizing Eqn. (22). Note that this is a LP problem with respect to two positive parameters  $\frac{1}{\omega_p}$  and  $\frac{1}{\omega_q}$ . Hence, the optimal point of  $(\frac{1}{\omega_p}, \frac{1}{\omega_q})$  should lie in the set of the turning points identified by the minimization operator in the expression of  $\tilde{f}(\omega_p, \omega_q)$  and  $\tilde{g}(\omega_p, \omega_q)$ .

All the minimization operators in  $\tilde{f}$  and  $\tilde{g}$  take the form of  $\min_{1 \leq j \leq N} \{a_j + b_j x\}$ . The general curve of this function is depicted in Fig.3. The curve of  $y = \min_{1 \leq j \leq N} \{a_j + b_j x\}$  is divided into several line segments by some turning points. The skews of the line segments should decrease as  $x$  increases. The first line segment should be the one with the minimum  $a_j$ . Denote  $j_1 = \arg \min_j \{a_j\}$  and  $y = a_{j_1} + b_{j_1} x$  should be the first line segment. Denote  $j_2$  the index for the second line segment. Since the skew must decrease, we have  $j_2 = \arg_j \{ \text{the solution for } a_j + b_j x = a_{j_1} + b_{j_1} x \text{ is minimum} | b_{j_1} > b_j \}$ . This procedure continues until the index  $j_l = \arg \min_j \{b_j\}$  is selected. Then all the turning points are founded. This process is summarized as Algorithm 1.

---

**Algorithm 1** JMLE: Finding the turning points for  $\min_{1 \leq j \leq N} \{a_j + b_j x\}$

---

Arrange  $b_j$  in decreasing order  $\{b_{j_l}\}$  such that  $b_{j_l} \geq b_{j_{l+1}}$ .  
 Find  $i_1 = \arg \min_j \{a_j\}$  and  $l_1$  such that  $j_{l_1} = i_1$ ,  $m = 1$ .  
**while**  $l_m \neq N$  **do**  
   Find  $l_{m+1} = \arg \min_l \left\{ \frac{a_{j_l} - a_{j_{l+1}}}{b_{j_l} - b_{j_{l+1}}} \mid l > l_m \right\}$  and turning  
   point  $x_m = \frac{a_{j_{l_{m+1}}} - a_{j_{l_m}}}{b_{j_{l_m}} - b_{j_{l_{m+1}}}}$ .  
    $m = m + 1$ .  
**end while**  
 Output the turning points  $\{x_i\}_{1 \leq i \leq m-1}$ .

---

After applying Algorithm 1 to the three minimization operators in the expressions of  $\tilde{f}$  and  $\tilde{g}$ , we get three sets of turning points for  $1/\omega_p$ ,  $1/\omega_q$  and  $\frac{1/\omega_q}{1/\omega_p}$  respectively. Denote the three sets as  $\{1/\omega_p = a_j\}_{1 \leq j \leq m_1}$ ,  $\{1/\omega_q = b_j\}_{1 \leq j \leq m_2}$  and  $\left\{ \frac{1/\omega_q}{1/\omega_p} = c_j \right\}_{1 \leq j \leq m_3}$ . Each turning point of the three sets is actually one line in the two dimensional plane of  $(1/\omega_p, 1/\omega_q)$ . These lines together tessellate the plane  $(1/\omega_p, 1/\omega_q)$  into several small regions. The optimal point of  $(1/\omega_p, 1/\omega_q)$  must lie in the intersections of these lines since  $\tilde{\xi}$  is a linear function (in terms of  $1/\omega_p$  and  $1/\omega_q$ ) with fixed coefficients inside each

of these small regions. Therefore, we just do a search among those intersections and find the one with the minimum  $\tilde{\xi}$ . This will give the JMLE  $(\hat{\omega}_p, \hat{\omega}_q)$ . Then, invoking Eqn. (12) and (19), we can give the JMLE  $(\hat{\phi}_p, \hat{\phi}_q)$ . The detailed process is summarized in Algorithm 2.

---

**Algorithm 2** JMLE: Finding the estimates  $(\hat{\omega}_p, \hat{\omega}_q, \hat{\phi}_p, \hat{\phi}_q)$

---

- 1: Exploit Algorithm 1 to get three sets of turning points:  $\{1/\omega_p = a_j\}_{1 \leq j \leq m_1}$ ,  $\{1/\omega_q = b_j\}_{1 \leq j \leq m_2}$  and  $\left\{ \frac{1/\omega_q}{1/\omega_p} = c_j \right\}_{1 \leq j \leq m_3}$ .
  - 2: Identify three sets of intersection points  $(1/\omega_p, 1/\omega_q)$ :  $\{(a_j, b_l)\}_{1 \leq j \leq m_1, 1 \leq l \leq m_2}$ ,  $\{(a_j, a_j c_l)\}_{1 \leq j \leq m_1, 1 \leq l \leq m_3}$  and  $\left\{ \left( \frac{b_j}{c_l}, b_j \right) \right\}_{1 \leq j \leq m_2, 1 \leq l \leq m_3}$ .
  - 3: Exploit the above intersections and Eqn. (22) to compute  $\tilde{\xi}$ . Find the one which gives the minimum  $\tilde{\xi}$ . Accordingly, output the corresponding  $(\hat{\omega}_p, \hat{\omega}_q)$ .
  - 4: Exploit Eqn. (12) and (19) to give the estimates  $(\hat{\phi}_p, \hat{\phi}_q)$ . Note that when using (12),  $\hat{\phi}_p$  can be chosen arbitrarily as long as (12) is satisfied.
- 

Note that the number of turning points is  $\mathcal{O}(N)$ , i.e., the number of lines in the  $(1/\omega_p, 1/\omega_q)$  is  $\mathcal{O}(N)$ . Hence, the number of intersection points needed to be checked during the search for JMLE is  $\mathcal{O}(N^2)$ . In addition, the number of instructions needed to calculate  $\tilde{\xi}$  is  $\mathcal{O}(N)$  for each intersection point. Therefore, the computational complexity for the proposed JMLE algorithm is  $\mathcal{O}(N^3)$  in the worst case. One thing we should mention is that this is the worst case complexity. In Section V, simulation results show that the experimental complexity can be significantly lower than this upper bound and almost scales linearly with  $N$ .

*Remark 1:* The proposed JMLE possesses both low complexity and high accuracy. The low complexity results from the fact that the ultimate optimization problem (22) is a LP problem only related to two parameters. Besides, since we give the exact JMLE based on all the raw observations, the proposed JMLE algorithm shows good accuracy.

#### IV. GENERALIZED ML-LIKE ESTIMATOR (GMLLE)

Due to the quite limited computational capability of the sensor nodes in WSN, we endeavor to find estimators with even lower computational complexity than the proposed JMLE algorithm in the previous section. Specifically, we invoke the generalized ML-like estimator (GMLLE) proposed in [15]. Although the method of reducing the number of parameters is similar, the overall algorithm and performance analysis here are very different since (i) we adopt PBS here while [15] adopts a two-way message exchange mechanism; (ii) more importantly, we assume an exponential delay model rather than a Gaussian delay model used in [15]. Both differences have made our work much more challenging.

##### A. Algorithm

The basic idea behind GMLLE in [15] is that by subtraction of the observation equations, the fixed part of delay and the

clock offset can be removed and thus the number of unknown parameters is reduced. The idea still works for exponential delay model in a PBS protocol herein. Specifically, subtracting the  $j$ -th and  $(j+k)$ -th equations in (4), (5) and (6), we get:

$$\frac{r_{j+k}^{mp} - r_j^{mp}}{\omega_p} - s_{j+k}^m + s_j^m = z_{j+k}^{mp} - z_j^{mp}, \quad (23)$$

$$\frac{r_{j+k}^{mq} - r_j^{mq}}{\omega_q} - s_{j+k}^m + s_j^m = z_{j+k}^{mq} - z_j^{mq}, \quad (24)$$

$$\frac{r_{j+k}^{pq} - r_j^{pq}}{\omega_q} - \frac{s_{j+k}^p - s_j^p}{\omega_p} = z_{j+k}^{pq} - z_j^{pq}, \quad (25)$$

where  $1 \leq j \leq N-k$ . Here  $\lceil N/2 \rceil \leq k \leq N-1$  is a user-defined parameter. One thing we should mention is that the GMLLE is different from the algorithm in [19]. The algorithm in [19] add together the two equations involved in the two-way message exchange mechanism in every round. This may not work well in the PBS mechanism since by doing so for Eqn. (4), (5) and (6), we still have to deal with four parameters  $\omega_p, \omega_q, \phi_p, \phi_q$  and only the fixed delay  $d$  disappears. In contrast, by invoking a user defined parameter  $k$ , GMLLE can eliminate three parameters  $\omega_p, \omega_q$  and  $d$ .

In GMLLE, we constrain  $k$  to be larger than  $N/2$  in order to avoid the statistical dependence of the R.H.S. of Eqn. (23), (24) and (25). In fact, the random variables in the R.H.S. of (23), (24) and (25) are i.i.d. Laplace variables with zero mean. Specifically, the distribution is  $p(x) = \frac{1}{2\alpha} \exp\left(-\frac{|x|}{\alpha}\right)$ . Consequently, the likelihood function is as follows:

$$\begin{aligned} L(\omega_p, \omega_q, \alpha) &= \left(\frac{1}{2\alpha}\right)^{3(N-k)} \\ &\times \exp \left[ -\frac{1}{\alpha} \sum_{j=1}^{N-k} \left( \left| \frac{r_{j+k}^{mp} - r_j^{mp}}{\omega_p} - s_{j+k}^m + s_j^m \right| \right. \right. \\ &+ \left. \left| \frac{r_{j+k}^{mq} - r_j^{mq}}{\omega_q} - s_{j+k}^m + s_j^m \right| \right. \\ &+ \left. \left. \left| \frac{r_{j+k}^{pq} - r_j^{pq}}{\omega_q} - \frac{s_{j+k}^p - s_j^p}{\omega_p} \right| \right) \right]. \quad (26) \end{aligned}$$

Now we apply MLE to the above likelihood function and the situation becomes much simpler than the strict JMLE discussed in Section III. To maximize  $L(\omega_p, \omega_q, \alpha)$ , we only need to minimize the following expression:

$$\begin{aligned} \xi \triangleq &\sum_{j=1}^{N-k} \left( \left| \frac{r_{j+k}^{mp} - r_j^{mp}}{\omega_p} - s_{j+k}^m + s_j^m \right| \right. \\ &+ \left| \frac{r_{j+k}^{mq} - r_j^{mq}}{\omega_q} - s_{j+k}^m + s_j^m \right| \\ &+ \left. \left| \frac{r_{j+k}^{pq} - r_j^{pq}}{\omega_q} - \frac{s_{j+k}^p - s_j^p}{\omega_p} \right| \right). \quad (27) \end{aligned}$$

Again we are confronting with a LP problem. Instead of conducting an exhaustive search, we now use a low complexity but still accurate iteration method to minimize  $\xi$  as in [19] and [29]. Suppose we have already got an estimate  $\hat{\omega}_q$ , then

to minimize  $\xi$  in terms of  $\omega_p$  is equivalent to minimizing the follows expression (the middle absolute value term in Eqn. (27) is omitted since it is independent of  $\omega_p$ ):

$$\begin{aligned} \xi_1 \left( \frac{1}{\omega_p} \right) &= \sum_{j=1}^{N-k} \left( \left| r_{j+k}^{mp} - r_j^{mp} \right| \left| \frac{1}{\omega_p} - \frac{s_{j+k}^m - s_j^m}{r_{j+k}^{mp} - r_j^{mp}} \right| \right. \\ &+ \left. \left| s_{j+k}^p - s_j^p \right| \left| \frac{1}{\omega_p} - \frac{r_{j+k}^{pq} - r_j^{pq}}{s_{j+k}^p - s_j^p} \frac{1}{\hat{\omega}_q} \right| \right). \quad (28) \end{aligned}$$

The minimization of  $\xi_1$  is a weighted median value problem [29]. For this, we should rearrange the sequence  $\left\{ \frac{s_{j+k}^m - s_j^m}{r_{j+k}^{mp} - r_j^{mp}} \right\}_{1 \leq j \leq N-k} \cup \left\{ \frac{r_{j+k}^{pq} - r_j^{pq}}{s_{j+k}^p - s_j^p} \frac{1}{\hat{\omega}_q} \right\}_{1 \leq j \leq N-k}$  in increasing order. Note that regardless of the value of  $\hat{\omega}_q$ , we can rearrange  $\left\{ \frac{s_{j+k}^m - s_j^m}{r_{j+k}^{mp} - r_j^{mp}} \right\}_{1 \leq j \leq N-k}$  and  $\left\{ \frac{r_{j+k}^{pq} - r_j^{pq}}{s_{j+k}^p - s_j^p} \frac{1}{\hat{\omega}_q} \right\}_{1 \leq j \leq N-k}$  in increasing order separately in advance. So, when a new  $\hat{\omega}_q$  is generated, we only need to merge two increasing sequences into one long combined sequence still in increasing order. This can reduce the complexity significantly. After merging the two sequences in this way, we rewrite  $\xi_1$  into the following form

$$\xi_1 = \sum_{j=1}^{2(N-k)} a_j \left| \frac{1}{\omega_p} - b_j \right|. \quad (29)$$

where  $\{b_j\}_{1 \leq j \leq 2(N-k)}$  is in increasing order. Then we find the minimum  $M$  such that  $\sum_{j=1}^M a_j \geq \frac{1}{2} \sum_{j=1}^{2(N-k)} a_j$ . Thus optimal  $\frac{1}{\omega_p}$ , i.e., the weighted median value, is  $b_M$ .

After updating the estimate  $\hat{\omega}_p$ , we again update  $\hat{\omega}_q$  in a similar way and keep iterating until convergence. Numerical experiments show that three rounds of iteration are usually enough for convergence (one round means updates for both  $\hat{\omega}_p$  and  $\hat{\omega}_q$  once). As for the initial estimate, we choose  $\hat{\omega}_p = 1, \hat{\omega}_q = 1$  because the clock skew is generally very small in real WSNs [13]. The reason that we do not exploit a LS initial estimate as [19] is that the pseudo-inverse of the matrix involved in the LS estimation may incur big computation overhead. Actually, the complexity of computing the pseudo-inverse turns out to be even larger than that of the iterations in GMLLE. It is meaningless to spend a majority of computational capability on just finding a coarse initial point.

The iterations of GMLLE raise the convergence issue, which is analyzed as follows. Note that each iteration finds the corresponding minimum points for either  $\omega_p$  or  $\omega_q$ , i.e., the value of the target function  $\xi$  always decreases after each iteration. Specifically, suppose that the estimates at the  $k$ -th iteration are  $\omega_p^{(k)}$  and  $\omega_q^{(k)}$ . We have:

$$\begin{aligned} \xi \left( \frac{1}{\omega_p^{(k)}}, \frac{1}{\omega_q^{(k)}} \right) &\geq \xi \left( \frac{1}{\omega_p^{(k+1)}}, \frac{1}{\omega_q^{(k)}} \right) \\ &\geq \xi \left( \frac{1}{\omega_p^{(k+1)}}, \frac{1}{\omega_q^{(k+1)}} \right). \end{aligned}$$

Furthermore, the objective function  $\xi$  in Eqn. (27) is convex with respect to  $\frac{1}{\omega_p}$  and  $\frac{1}{\omega_q}$ . Therefore, the proposed GMLLE will finally converge to the global optimum point.

With the estimates  $\hat{\omega}_p$  and  $\hat{\omega}_q$  available, we can now easily give good estimates for the rest of the parameters by exploiting

the results of [21]. Plugging the estimates  $\hat{\omega}_p$  and  $\hat{\omega}_q$  back into Eqn. (4), (5) and (6), we obtain:

$$U_j \triangleq \frac{r_j^{mp}}{\hat{\omega}_p} - s_j^m = d + \frac{\phi_p}{\hat{\omega}_p} + z_j^{mp}, \quad (30)$$

$$V_j \triangleq \frac{r_j^{mq}}{\hat{\omega}_q} - s_j^m = d + \frac{\phi_q}{\hat{\omega}_q} + z_j^{mq}, \quad (31)$$

$$W_j \triangleq \frac{r_j^{pq}}{\hat{\omega}_q} - \frac{s_j^p}{\hat{\omega}_p} = d + \left( \frac{1}{\hat{\omega}_q} - \frac{1}{\hat{\omega}_p} \right) \phi_p + z_j^{pq}. \quad (32)$$

Viewing  $\phi_p/\omega_p$  and  $\phi_q/\omega_q$  as new clock offsets, we get a pure clock offset estimation problem. Thanks to [21], the MLE and MVUE are the same in this case and can be written as follows:

$$\hat{\phi}_p = \hat{\omega}_p(V_{(1)} - W_{(1)}), \quad (33)$$

$$\hat{\phi}_q = \hat{\omega}_q(2V_{(1)} - U_{(1)} - W_{(1)}), \quad (34)$$

$$\hat{d} = U_{(1)} + W_{(1)} - V_{(1)}, \quad (35)$$

$$\hat{\alpha} = \frac{1}{3N} \sum_{j=1}^N (U_j + V_j + W_j) - \frac{1}{3}U_{(1)} - \frac{1}{3}V_{(1)} + \frac{2}{3}W_{(1)}. \quad (36)$$

where  $U_{(1)} \triangleq \min_{1 \leq j \leq N} \{U_j\}$ ,  $V_{(1)}$  and  $W_{(1)}$  are similarly defined.

The computational complexity is analyzed as follows. The ordering of the sequence  $\left\{ \frac{s_{j+k}^m - s_j^m}{r_{j+k}^{mp} - r_j^{mp}} \right\}$ ,  $\left\{ \frac{r_{j+k}^{pq} - r_j^{pq}}{s_{j+k}^p - s_j^p} \right\}$ ,  $\left\{ \frac{s_{j+k}^m - s_j^m}{r_{j+k}^{mq} - r_j^{mq}} \right\}$  and  $\left\{ \frac{s_{j+k}^p - s_j^p}{r_{j+k}^{pq} - r_j^{pq}} \right\}$  can be accomplished with a complexity of  $\mathcal{O}((N-k)^2)$  while the iteration part will incur  $\mathcal{O}(I(N-k))$  complexity, where  $I$  is the iteration rounds. Thus the total complexity is at most  $\mathcal{O}((N-k)^2 + I(N-k))$  and we choose  $I = 3$  in practice.

*Remark 2:* The low complexity of the GMLLE algorithm is justified by the simulation results in Section V. Note that if we use some advanced sorting algorithms (e.g., quick sort) to do the sequence ordering in GMLLE, the complexity can be even lower. We do not consider this here.

## B. Performance Bound

The performance bound for GMLLE in a two-way message exchange protocol under the assumption of Gaussian delay model is derived as CRLB based on reduced set of observations in Eqn. (23)-(25). This bound is critical in evaluating how much performance loss is incurred by the manipulation of the observations. Compared to [15], since we are studying a PBS protocol under the assumption of exponential delay here, we need to derive the performance bound for GMLLE again. This is not a trivial job since the conditions are quite different and much more complex. Note that the derivation of the performance bound also differs from that of [19]: (i) the algorithm and synchronization mechanism are different and our GMLLE algorithm involves a user defined parameter  $k$ ; (ii) the CRLB-like performance bound derived in [19] is just an approximation while we derive the exact one for GMLLE in this section. Note that the derived performance bound can only serve as performance benchmark for GMLLE and does not apply to JMLE for it is based on the reduced set of observations.

Denote  $y_{j,k}^{mp} \triangleq r_{j+k}^{mp} - r_j^{mp}$ ,  $y_{j,k}^{mq} \triangleq r_{j+k}^{mq} - r_j^{mq}$  and  $y_{j,k}^{pq} \triangleq r_{j+k}^{pq} - r_j^{pq}$ , where  $1 \leq j \leq N-k$ . Denote  $\mathcal{L}(a, b)$  the Laplace distribution:  $p(x) = \frac{1}{2b} \exp\left(-\frac{|x-a|}{b}\right)$ . Then, from Eqn. (23)-(25), we have  $y_{j,k}^{mp} \sim \mathcal{L}(\omega_p(s_{j+k}^m - s_j^m), \alpha\omega_p)$ ,  $y_{j,k}^{mq} \sim \mathcal{L}(\omega_q(s_{j+k}^m - s_j^m), \alpha\omega_q)$  and  $y_{j,k}^{pq} \sim \mathcal{L}\left(\frac{\omega_q}{\omega_p}(s_{j+k}^p - s_j^p), \alpha\omega_q\right)$ . Then the joint probability density function (PDF) of  $y_{j,k}^{mp}$ ,  $y_{j,k}^{mq}$  and  $y_{j,k}^{pq}$ , where  $1 \leq j \leq N-k$ , is as follows:

$$f(y^{mp}, y^{mq}, y^{pq}) = \left(\frac{1}{2\alpha}\right)^{3(N-k)} \left(\frac{1}{\omega_p}\right)^{N-k} \left(\frac{1}{\omega_q}\right)^{2(N-k)} \\ \times \exp \left[ -\frac{1}{\alpha} \sum_{j=1}^{N-k} \left( \left| \frac{y_{j,k}^{mp}}{\omega_p} - s_{j+k}^m + s_j^m \right| \right. \right. \\ \left. \left. + \left| \frac{y_{j,k}^{mq}}{\omega_q} - s_{j+k}^m + s_j^m \right| + \left| \frac{y_{j,k}^{pq}}{\omega_q} - \frac{s_{j+k}^p - s_j^p}{\omega_p} \right| \right) \right]. \quad (37)$$

*Remark 3:* Note that we regard  $y^{mp}$ ,  $y^{mq}$ ,  $y^{pq}$  as the only random outputs of the system, i.e., the observations. And  $s^p$ ,  $s^m$  are just the previously known system inputs and are not observations. Strictly speaking, Eqn. (26) in Subsection IV-A is just an approximation of the likelihood function when  $\omega_p$  and  $\omega_q$  are near to 1 while Eqn. (37) is the standard likelihood function. However, maximizing (37) is too complicated since it cannot be reduced to a LP problem. So we use the approximate likelihood function (26) when designing algorithm in Subsection A. Now, to derive the performance bound, we will turn to the standard likelihood function in (37).

Generally speaking, CRLB does not exist for exponential distribution since the support region is related to the unknown parameters and the regularity condition for CRLB is not satisfied. However, when they are manipulated according to the algorithm of GMLLE, the observations are changed to Laplace random variables, which are suitable for deriving CRLB. We first perform some parameter transformations:  $\theta_1 \triangleq 1/\omega_p$ ,  $\theta_2 \triangleq 1/\omega_q$ ,  $\lambda \triangleq 1/\alpha$ . Then the joint PDF becomes as follows:

$$f(y^{mp}, y^{mq}, y^{pq}) = \left(\frac{\lambda}{2}\right)^{3(N-k)} \theta_1^{N-k} \theta_2^{2(N-k)} \\ \times \exp \left[ -\lambda \sum_{j=1}^{N-k} \left( \left| y_{j,k}^{mp} \theta_1 - s_{j+k}^m + s_j^m \right| \right. \right. \\ \left. \left. + \left| y_{j,k}^{mq} \theta_2 - s_{j+k}^m + s_j^m \right| + \left| y_{j,k}^{pq} \theta_2 - (s_{j+k}^p - s_j^p) \theta_1 \right| \right) \right]. \quad (38)$$

Now we derive the CRLB with respect to  $(\theta_1, \theta_2, \lambda)$  and this CRLB based on the reduced set of observations is the desired performance bound for GMLLE. The derivation of CRLB is just the derivation of the Fisher information matrix (FIM). This is accomplished by Lemma 1.

*Lemma 1:* The FIM based on the reduced set of observations in Eqn. (38) is as follows, with respect to  $(\theta_1, \theta_2, \lambda)$ .

$$\mathbf{FIM} = \begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{12} & F_{22} & F_{23} \\ F_{13} & F_{23} & F_{33} \end{pmatrix}$$

where

$$\begin{aligned}
F_{11} &= \frac{3(N-k)}{\lambda^2}, \quad F_{12} = \frac{N-k}{\lambda\theta_1}, \quad F_{13} = \frac{2(N-k)}{\theta_2}, \\
F_{22} &= \frac{N-k}{\theta_1^2} + \frac{\lambda^2}{\theta_1^2} \sum_{j=1}^{N-k} (s_{j+k}^m - s_j^m)^2 \\
&\quad + \lambda^2 \sum_{j=1}^{N-k} (s_{j+k}^p - s_j^p)^2, \\
F_{23} &= -\frac{\theta_1\lambda^2}{\theta_2} \sum_{j=1}^{N-k} (s_{j+k}^p - s_j^p)^2, \\
F_{33} &= -\frac{(N-k)^2}{\theta_2^2} + \frac{2(N-k)}{\theta_2^2} + \frac{\lambda^2}{\theta_2^2} \sum_{j=1}^{N-k} (s_{j+k}^m - s_j^m)^2 \\
&\quad + \frac{\lambda^2\theta_1^2}{\theta_2^2} \sum_{j=1}^{N-k} (s_{j+k}^p - s_j^p)^2.
\end{aligned}$$

*Proof:* The derivation is somewhat complicated and is available in Appendix A. ■

Thanks to Lemma 1, the performance bound for  $(\lambda, \theta_1, \theta_2)$  is just the diagonal elements of the inverse matrix of the FIM, i.e., for any unbiased estimator  $(\hat{\lambda}, \hat{\theta}_1, \hat{\theta}_2)$ :

$$\text{Var}(\hat{\lambda}) \geq [\mathbf{FIM}]_{11}^{-1}, \quad (39)$$

$$\text{Var}(\hat{\theta}_1) \geq [\mathbf{FIM}]_{22}^{-1}, \quad (40)$$

$$\text{Var}(\hat{\theta}_2) \geq [\mathbf{FIM}]_{33}^{-1}. \quad (41)$$

Closed form of the performance bound can be got directly from (39)-(41). However the expression is too complicated and hence not given here. According to the parameter transformation rule of CRLB [16], the performance bound for the original parameter vector  $(\alpha, \omega_p, \omega_q)$  is as follows:

$$\text{Var}(\hat{\alpha}) \geq \alpha^4 [\mathbf{FIM}]_{11}^{-1}, \quad (42)$$

$$\text{Var}(\hat{\omega}_p) \geq \omega_p^4 [\mathbf{FIM}]_{22}^{-1}, \quad (43)$$

$$\text{Var}(\hat{\omega}_q) \geq \omega_q^4 [\mathbf{FIM}]_{33}^{-1}. \quad (44)$$

### C. Identifying Optimal User-defined Parameter $k$

Obviously, the performance of the GMLLE depends on the choice of a user-defined parameter  $k$ . In this subsection, we endeavor to select the optimal  $k$  which can give us the best estimation performance, i.e., the minimum mean square error (MMSE). In this subsection, we make efforts to find such an optimal  $k$  analytically. In Section V, computer simulations will give us the same result which corroborates our theoretical analysis.

When the observations are reduced to Eqn. (23)-(25), we have derived the corresponding performance bound in Subsection IV-B. Note that the GMLLE is indeed a MLE based on the reduced observations (it is not true MLE since the true MLE should be based on the original observations, i.e., GMLLE is only suboptimal). From classical estimation theory, we know that MLE is asymptotically unbiased and approaches to the CRLB when the number of observations is large enough [16]. Consequently, the performance of GMLLE can be well

described and predicted by the CRLB-like performance bound derived in Subsection IV-B, i.e., smaller performance bound means better performance. Based on the above analysis, the problem turns out to be finding the  $k$  which can minimize the performance bound. In the following, when referring to performance bound, we mean the performance bound for  $\theta_2$  since: (i) the estimation of the clock skew of the inactive node is the most important one when requiring long term synchronization; (ii) the accuracy of the estimation of the clock offset  $\phi_q$  is related with the accuracy of the estimation of skew through Eqn. (34).

Without loss of generality, we assume that  $s_{j+1}^m - s_j^m = H > 0$ ,  $s_{j+1}^p - s_j^p = G > 0$ , since the interval time of transmission can be chosen in advance, where  $G$  and  $H$  are two constants. The rationality of such an assumption is well argued in [15]. Under this assumption, the optimal value of  $k$  can be selected by the following theorem.

**Theorem 1:** The optimal choice of the user-defined parameter  $k$  is the integer nearest to  $(2/3)N$ .

*Proof:* Under the previously mentioned assumption, the FIM can be further reduced to the form of  $[\mathbf{FIM}] = (N-k) \cdot [\mathbf{A}]$  (the detailed expression of  $\mathbf{A}$  is complex and omitted here). Since<sup>2</sup>  $\text{PB}(\theta_2) = [\mathbf{FIM}]_{33}^{-1} = \frac{1}{N-k} [\mathbf{A}]_{33}^{-1}$ , we need to compute  $[\mathbf{A}]_{33}^{-1}$  which can be written as follows:

$$[\mathbf{A}]_{33}^{-1} = \frac{\det \begin{pmatrix} A_{11} & A_{12} \\ A_{12} & A_{22} \end{pmatrix}}{\det([\mathbf{A}])}.$$

After some algebraic calculations, the performance bound for  $\theta_2$  can be expressed as:

$$\begin{aligned}
\frac{1}{\text{PB}(\theta_2)} &= (N-k) \times \\
&\left( \frac{-4 - 4\lambda^2 H^2 k^2 - 8k^2 G^2 \lambda^2 \theta_1^2 - 3\theta_1^4 k^4 \lambda^4 G^4}{2\theta_2^2 + 3\lambda^2 \theta_2^2 H^2 k^2 + 3k^2 G^2 \lambda^2 \theta_1^2 \theta_2^2} \right. \\
&\quad \left. - \frac{N-k}{\theta_2^2} + \frac{2}{\theta_2^2} + \frac{\lambda H^2 k^2}{\theta_2^2} + \frac{\lambda^2 \theta_1^2 k^2 G^2}{\theta_2^2} \right).
\end{aligned}$$

Not surprisingly, the value of  $\text{PB}(\theta_2)$  depends on the unknown parameters and hence the optimal  $k$  cannot be determined strictly when we do not know the value of those parameters. However, for practical concern, we can invoke some approximations to make the problem solvable. First, the values of  $\theta_1$  and  $\theta_2$  are generally very close to 1 (note that they are just the reciprocal of the clock skews) and thus can be approximated as 1 here. Second, the number of observations  $N$  is usually much larger than  $H, G$  and  $\lambda$ . Note that  $N/2 \leq k < N$ , we only need to consider those terms with the highest sum degree of  $N$  and  $k$  and simply omit the other terms. Third, we regard the variable  $k$  as a real number rather than an integer to make the optimization feasible analytically. Thanks to the above three approximation assumptions, we can write the derivative of the quantity  $1/\text{PB}(\theta_2)$  with respect to

<sup>2</sup>By PB, we mean performance bound.



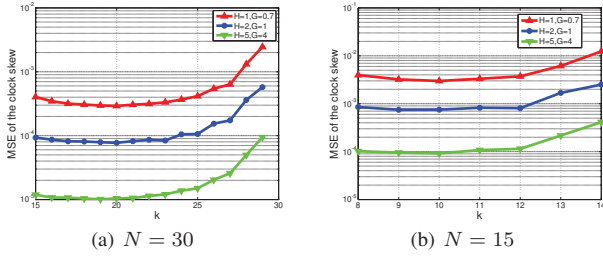


Fig. 4. GMLLE: MSE of the clock skew estimate  $\hat{\omega}_q$  with different  $G, H$  and  $k$ . The number of observations are fixed to be 30 (left) or 15 (right) respectively. For all values of  $G, H$ , the optimal value of  $k$  is approximately 20 (left) or 10 (right) as predicted by Theorem 1.

$k$  as:

$$\begin{aligned} & \frac{d}{dk} \left( \frac{1}{\text{PB}(\theta_2)} \right) \\ & \approx k^5 (3k - 2N) (9\lambda^5 H^6 + 9\lambda^6 H^4 G^2 + 9\lambda^6 H^2 G^4 + \\ & \quad 18\lambda^5 H^4 G^2 + 9\lambda^5 H^2 G^4) / (2 + 3\lambda^2 H^2 k^2 + 3k^2 G^2 \lambda^2)^2. \end{aligned}$$

Note that the factor which can decide the sign of the derivative is just  $(3k - 2N)$ . Thus, for all the values in the range of  $N/2 \leq k < N$ , there is only one point that can make the above derivative be zero and this is just the optimal point we want. Letting the derivative be zero, we always have  $k = (2/3)N$  regardless of the specific values of  $N, G, H$  and  $\lambda$ . Thus, we finish the proof. ■

In Theorem 1, the choice coincides with that of [15] although the delay model, synchronization mechanism and derivation are totally different. In Section V, simulations support the theoretical analysis in this subsection and further indicate that even when  $N$  is not that large (e.g.,  $N = 15$ ), the optimality for the chosen  $k$  still holds. This validates the robustness of Theorem 1.

## V. NUMERICAL EXPERIMENTS AND DISCUSSIONS

In this section, MATLAB simulations are implemented to verify the effects of the proposed JMLE and GMLLE. All the reported data points are obtained through an average of the results of ten thousand simulation runs. According to Theorem 1, the optimal value of the user-defined parameter  $k$  should be  $(2/3)N$ . In order to justify this property in the simulation, we set the involved parameters as follows:  $\omega_p = 1.005, \omega_q = 0.995, \phi_p = -4, \phi_q = 5, d = 3, \alpha = 1, N = 30$ , which are all practical values frequently used in the literature. Fig.4(a) shows the MSE of the estimates of the two clock rates  $\omega_p, \omega_q$  with different  $G, H$  and  $k$  under the fixed number of observations, i.e., 30. In this case, the optimal value of  $k$  should be 20 according to the theoretical result in Theorem 1. Fig.4(a) clearly corroborates this conclusion. Though sometimes the strictly optimal value is 19, the difference between the MSE when  $k$  equals to 19 and 20 is very slight. In practice, we tend to use large  $k$  if the MSE does not increase much, since the computation overhead decreases with  $k$  increasing (the complexity of GMLLE is an increasing function of  $(N - k)$ ). Fig.4(a) also tells us that generally speaking, the larger the

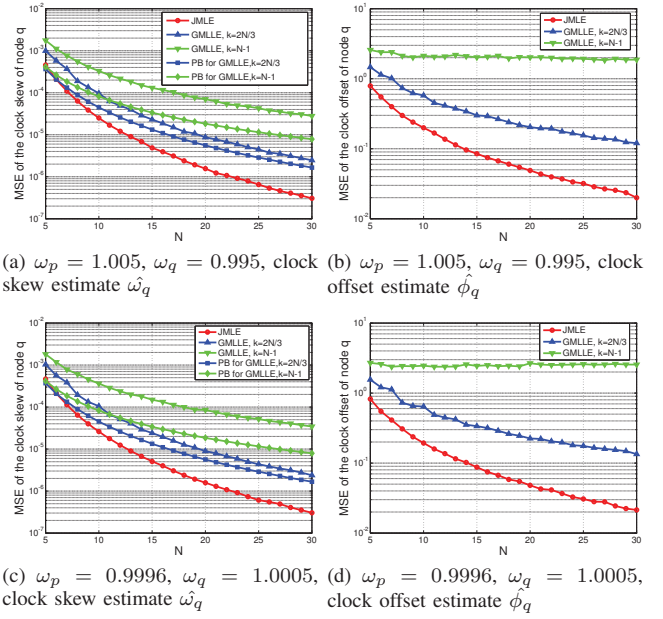


Fig. 5. Comparison of the JMLE and GMLLE: (i) JMLE, (ii) GMLLE with  $k = (2/3)N$ , (iii) GMLLE with  $k = N - 1$ . The red lines are for (i). The blue lines are for (ii). The green lines are for (iii). For clock skew estimation by GMLLE, the corresponding performance bound (PB) is also plotted. The figure shows that JMLE outperforms GMLLE and an optimal value of  $k$  (i.e.,  $k = (2/3)N$ ) is important for accuracy.

values of  $G$  and  $H$ , the better the performance of GMLLE. This is reasonable since GMLLE depends on the gap between observations and if  $G$  and  $H$  become larger, this gap is also amplified which leads to more accurate estimates. This property also holds for JMLE. Generally speaking, as  $G$  and  $H$  become larger or  $\alpha$  becomes smaller (note that  $\alpha$  is the mean of the random delay, which acts like the noise), the performance of both JMLE and GMLLE will become better. So, in practice, we need to ensure that  $(G + H)/\alpha$  is above a certain level, e.g., 20. Since the derivation of the optimal value of  $k$  in Subsection IV-C depends on the assumption that  $N$  is large, one may be curious about the situation when  $N$  is not very large. For this, we set  $N = 15$  and do the same simulation. The results are reported in Fig.4(b). We observe that the optimality of the analytically selected  $k$  still holds. The above property also holds for clock offset estimation since the accuracy of the offset estimation depends heavily on that of the skew estimation. The simulation figures for offset estimation are omitted due to the space limitation.

Now we endeavor to compare the performance of JMLE and GMLLE. We fix  $G = H = 10$ . We test two sets of clock rates  $\omega_p = 1.005, \omega_q = 0.995$  and  $\omega_p = 0.9996, \omega_q = 1.0005$ . In the latter case, the clock skews are much smaller, which enables us to see if the values of clock skews influence the estimation performance. Note that the values of these parameters are reasonable since the order of the quantity  $|\omega - 1|$  is roughly  $10^{-3}$  times those of  $d, \alpha$  and the clock offset  $\phi$  in real world WSNs [13]. In Fig.5(a) and Fig.5(b), for  $\omega_p = 1.005, \omega_q = 0.995$ , we compare three estimators: (i) JMLE, (ii) GMLLE with  $k = (2/3)N$ , (iii) GMLLE with

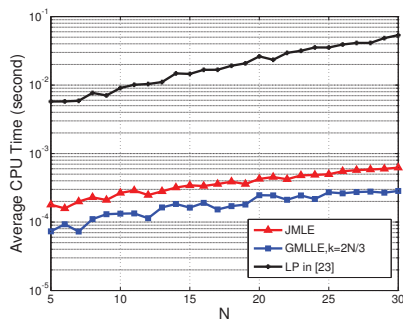


Fig. 6. A comparison of the average CPU time for running JMLE, GMLLE and the LP based algorithm proposed in [23]. We observe that (1) the running time of JMLE is about twice as much as that of GMLLE (note that the y-axis is in logarithmic scale), (2) the running time of the LP based algorithm in [23] is much larger than that of JMLE and GMLLE.

$k = N - 1$ . The computational complexity decreases from (i) to (iii) with the performance degrading, which can be seen from Fig.5(a) and Fig.5(b) obviously. The performance bound for the clock skew estimation in GMLLE is also depicted in Fig.5(a). When  $k = (2/3)N$ , or more generally speaking, when  $k$  is around the optimal user-defined value, the performance bound is very near to the MSE in GMLLE. This justifies that our performance bound based analysis of choosing optimal  $k$  in Section IV-C is reasonable. One thing we should mention is that, for estimation of clock offsets, GMLLE with  $k = N - 1$  actually fails to provide good estimate as shown in Fig.5(b). The MSE does not decrease as the number of observations increases. For this concern, if we want to use GMLLE, we'd better choose  $k = (2/3)N$  to get good estimates of the offsets. This essentially explains the necessity of selecting the optimal  $k$ . In Fig.5(c) and Fig.5(d), we conduct the same simulation when  $\omega_p = 0.9996, \omega_q = 1.0005$ . It is shown that MSE for both skew and offset estimates remain roughly the same, which means that the performance of JMLE and GMLLE are independent of the value of clock skews.

Next, we test the CPU time for running the JMLE, GMLLE and the LP method in [23] so as to compare their computational complexity. The average CPU time is reported in Fig.6. First, we observe that the CPU time of JMLE is roughly twice as much as that of GMLLE, which indicates that GMLLE algorithm has a lower complexity. In previous sections, the worst case complexity of JMLE and GMLLE has been analyzed theoretically. However, as shown in Fig.6, the experimental complexity of both algorithms are much smaller than the analytical upper bounds. In the simulation, the CPU time of both JMLE and GMLLE grow roughly linearly with  $N$  (though it is not quite obvious from Fig.6 since the y-axis is in logarithmic scale). Note that, when choosing estimators based on the tradeoff between complexity and accuracy, unless the computational ability of the wireless sensor is extremely limited, we tend to apply computationally complicated algorithm to achieve better accuracy since the energy cost of computation is much smaller than that of communications in WSNs generally [5]. Hence, JMLE, though a little complicated, is still a practical choice. Second, it can be seen from Fig.6 that the CPU time of the LP method in

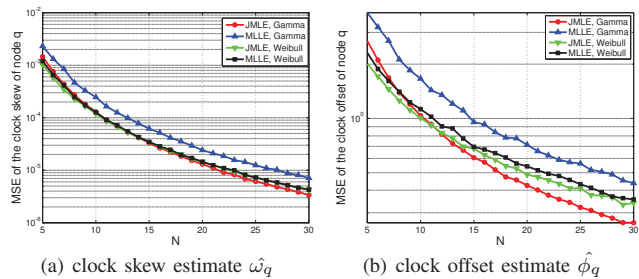


Fig. 7. (Robustness) The random delay is either Gamma distribution or Weibull distribution: (i) JMLE, (ii) GMLLE with  $k = (2/3)N$ . The red lines and green lines represent JMLE with a Gamma(2,1) delay and a Weibull(2,2) delay, respectively. The blue lines and black lines represent GMLLE with a Gamma(2,1) delay and a Weibull(2,2) delay, respectively.

[23] is much larger than (about 100 times of) that of either JMLE or GMLLE. In the simulation of the LP method of [23], we invoke the MATLAB function *linprog* and set the maximum number of iterations to be  $\lfloor \frac{N}{3} \rfloor$  since the minimum number of iterations necessary for obtaining the same accuracy as that of JMLE is about  $\lfloor \frac{N}{3} \rfloor$  in our numerical setting, i.e., any further reduction of the maximum number of iterations will make the accuracy of the LP method degrade significantly. We observe that JMLE, while achieving the same accuracy as LP method in [23], enjoys much lower complexity. Hence, the proposed algorithms are much more practical for real world large scale WSNs which consist of many cheap sensors with low computational capability.

Finally, we examine the robustness of JMLE and GMLLE when the delay model is mismatched because the random part of the delay may not be an exponential random variable exactly in real environment. For this, we apply JMLE and GMLLE to Gamma distributed random delay and Weibull distributed random delay, respectively. For Gamma delay, the shape parameter is 2 and the scale parameter is 1. For Weibull distribution, both shape parameter and scale parameter are 2. For GMLLE, we choose the optimal  $k$  as  $(2/3)N$ . The simulation results for skew estimation and offset estimation are shown in Fig.7(a) and Fig.7(b), respectively. JMLE still outperforms GMLLE in a Gamma(2,1) delay while the performance gap between JMLE and GMLLE in a Weibull(2,2) delay is small. For both estimators, the MSE, although larger than that of exact exponential delay model, still decreases quickly as  $N$  becomes large. Therefore, the robustness of the two estimators is verified.

## VI. CONCLUSION AND FUTURE WORK

In this paper, we study joint estimation of clock skew and offset in a PBS mechanism. Specifically, we investigate two estimators: JMLE and GMLLE. For both estimators, corresponding algorithms are derived and presented explicitly. For GMLLE, we derive its performance bound based on the reduced observations and identify the optimal user-defined parameter in both theoretical and experimental approach. Simulations are carried out to corroborate our algorithms and analysis. This paper combines PBS and joint estimation of skew and offset, which are two effective ways to save energy

in clock synchronization in WSNs. The algorithms presented will provide fundamental schemes to designing better WSNs [26], [27].

There are several interesting directions for future work. First, the assumption that the fixed delay  $d$  is the same for all links is imprecise for almost all real WSNs. Actually, many localization algorithms have utilized the difference in  $d$  to determine the position of sensors. Hence, taking the difference in  $d$  into account is a very practical concern. Second, in real WSNs, the clock parameters may not be a constant. Rather, the clock skew and offset may vary slowly with the time. Some pioneering works, e.g., [12] and [25], have already taken the time varying offset into consideration. But there still lack works on time varying clock skews in PBS mechanism. If one can solve the synchronization problem for time varying clock parameters in PBS mechanism, then significant energy reduction is expected since we do not need to resynchronize the clocks that frequently. Finally, further work on synchronizing the clocks over the entire network in a statistical signal processing perspective is expected [28].

#### ACKNOWLEDGMENT

This paper is supported by National Fundamental Research Grant (No. 2011CB302701); NSF China (No. 61271219,61202373); Shanghai Basic Research Key Project (No. 11JC1405100); China Ministry of Education Fok Y-ting Tung Fund (No. 122002); China Ministry of Education New Century Excellent Talent (No. NCET-10-0580); China Postdoctoral Science Foundation Grant 2011M500774 and 2012T50417, STCSM Grant 12JC1405200 and SEU SKL project #2012D13.

#### APPENDIX A

##### PROOF OF LEMMA 1: DERIVATION OF THE FISHER INFORMATION MATRIX

*Proof:* Taking logarithm of Eqn. (38) yields the following:

$$\begin{aligned} & \ln f(y^{mp}, y^{mq}, y^{pq}) \\ &= 3(N-k) \ln \left( \frac{\lambda}{2} \right) + (N-k) \ln \theta_1 + 2(N-k) \ln \theta_2 \\ & \quad - \lambda \left( \sum_{j=1}^{N-k} \left| \theta_1 y_{j,k}^{mp} - s_{j+k}^m + s_j^m \right| \right. \\ & \quad + \sum_{j=1}^{N-k} \left| \theta_2 y_{j,k}^{mq} - s_{j+k}^m + s_j^m \right| \\ & \quad \left. + \sum_{j=1}^{N-k} \left| \theta_2 y_{j,k}^{pq} - \theta_1 (s_{j+k}^p - s_j^p) \right| \right). \end{aligned}$$

Denote  $\xi$  the quantity inside the parentheses. Taking derivative with respect to  $\lambda, \theta_1, \theta_2$  respectively, we obtain:

$$\begin{aligned} \frac{\partial \ln f}{\partial \lambda} &= \frac{3(N-k)}{\lambda} - \xi, \\ \frac{\partial \ln f}{\partial \theta_1} &= \frac{N-k}{\theta_1} - \lambda \left( \sum_{j=1}^{N-k} y_{j,k}^{mp} \operatorname{sgn} \left( \theta_1 y_{j,k}^{mp} - s_{j+k}^m + s_j^m \right) \right. \\ & \quad \left. + \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right) \operatorname{sgn} \left( \theta_1 \left( s_{j+k}^p - s_j^p \right) - \theta_2 y_{j,k}^{pq} \right) \right), \\ \frac{\partial \ln f}{\partial \theta_2} &= \frac{2(N-k)}{\theta_2} - \lambda \left( \sum_{j=1}^{N-k} y_{j,k}^{mq} \operatorname{sgn} \left( \theta_2 y_{j,k}^{mq} - s_{j+k}^m \right. \right. \\ & \quad \left. \left. + s_j^m \right) + \sum_{j=1}^{N-k} y_{j,k}^{pq} \operatorname{sgn} \left( \theta_2 y_{j,k}^{pq} - \theta_1 \left( s_{j+k}^p - s_j^p \right) \right) \right), \end{aligned}$$

where  $\operatorname{sgn}(\cdot)$  denotes the sign function, i.e.,  $\operatorname{sgn}(x) = 1$  if  $x > 0$ ,  $\operatorname{sgn}(x) = -1$  if  $x < 0$ . Recall the definition of  $y^{mp}, y^{mq}, y^{pq}$ , we can rewrite the above equations as:

$$\begin{aligned} \frac{\partial \ln f}{\partial \theta_1} &= \frac{N-k}{\theta_1} - \lambda \left( \sum_{j=1}^{N-k} y_{j,k}^{mp} \operatorname{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right. \\ & \quad \left. + \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right) \operatorname{sgn} \left( z_j^{pq} - z_{j+k}^{pq} \right) \right), \\ \frac{\partial \ln f}{\partial \theta_2} &= \frac{2(N-k)}{\theta_2} - \lambda \left( \sum_{j=1}^{N-k} y_{j,k}^{mq} \operatorname{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right. \\ & \quad \left. + \sum_{j=1}^{N-k} y_{j,k}^{pq} \operatorname{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right), \\ \xi &= \sum_{j=1}^{N-k} \left| z_{j+k}^{mp} - z_j^{mp} \right| + \sum_{j=1}^{N-k} \left| z_{j+k}^{mq} - z_j^{mq} \right| \\ & \quad + \sum_{j=1}^{N-k} \left| z_{j+k}^{pq} - z_j^{pq} \right|. \end{aligned}$$

Before we proceed to derive FIM, we first summarize some useful equations, which will be used frequently in the later derivation, in Proposition 1. The proof of these equations can be accomplished in a step-by-step approach and is omitted here because of the limited space.

**Proposition 1:** Some frequently used formulas are listed as

follows.

$$\begin{aligned}
\mathbb{E} \left[ |z_{j+k}^{mp} - z_j^{mp}| \right] &= \mathbb{E} \left[ |z_{j+k}^{mq} - z_j^{mq}| \right] = \mathbb{E} \left[ |z_{j+k}^{pq} - z_j^{pq}| \right] = \frac{1}{\lambda}; \\
\mathbb{E} \left[ \left( z_{j+k}^{mp} - z_j^{mp} \right)^2 \right] &= \mathbb{E} \left[ \left( z_{j+k}^{mq} - z_j^{mq} \right)^2 \right] \\
&= \mathbb{E} \left[ \left( z_{j+k}^{pq} - z_j^{pq} \right)^2 \right] = \frac{2}{\lambda^2}; \\
\mathbb{E}(\xi) &= \frac{3(N-k)}{\lambda}, \mathbb{E}(\xi^2) = \frac{3(N-k)(N-k+1)}{\lambda^2}; \\
\mathbb{E} \left[ y_{j,k}^{mp} \operatorname{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right] &= \frac{\omega_p}{\lambda}; \\
\mathbb{E} \left[ y_{j,k}^{mq} \operatorname{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right] &= \mathbb{E} \left[ y_{j,k}^{pq} \operatorname{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right] \\
&= \frac{\omega_q}{\lambda}; \\
\mathbb{E} \left[ y_{j,k}^{mp} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right] &= \frac{2\omega_p}{\lambda^2}; \\
\mathbb{E} \left[ y_{j,k}^{mq} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right] &= \mathbb{E} \left[ y_{j,k}^{pq} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right] = \frac{2\omega_q}{\lambda^2}; \\
\mathbb{E} \left[ \left( y_{j,k}^{mp} \right)^2 \right] &= \omega_p^2 \left( \left( s_{j+k}^m - s_j^m \right)^2 + \frac{2}{\lambda^2} \right); \\
\mathbb{E} \left[ \left( y_{j,k}^{mq} \right)^2 \right] &= \omega_q^2 \left( \left( s_{j+k}^m - s_j^m \right)^2 + \frac{2}{\lambda^2} \right); \\
\mathbb{E} \left[ \left( y_{j,k}^{pq} \right)^2 \right] &= \frac{\omega_q^2}{\omega_p^2} \left( s_{j+k}^p - s_j^p \right)^2 + \omega_q^2 \cdot \frac{2}{\lambda^2}.
\end{aligned}$$

In what follows, we proceed to derive the elements of FIM one by one.

$$\begin{aligned}
F_{11} &= \mathbb{E} \left[ \left( \frac{\partial \ln f}{\partial \lambda} \right)^2 \right] \\
&= \frac{9(N-k)^2}{\lambda^2} - \frac{6(N-k)}{\lambda} \mathbb{E}(\xi) + \mathbb{E}(\xi^2) \\
&= \frac{3(N-k)}{\lambda^2}. \\
F_{12} &= \mathbb{E} \left( \frac{\partial \ln f}{\partial \lambda} \frac{\partial \ln f}{\partial \theta_1} \right) \\
&= \frac{6(N-k)^2}{\lambda \theta_1} - \frac{N-k}{\theta_1} \mathbb{E}(\xi) \\
&\quad - 3(N-k) \sum_{j=1}^{N-k} \mathbb{E} \left( y_{j,k}^{mp} \operatorname{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right) \\
&\quad + \lambda \mathbb{E} \left[ \xi \left( \sum_{j=1}^{N-k} y_{j,k}^{mp} \operatorname{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right. \right. \\
&\quad \left. \left. + \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right) \operatorname{sgn} \left( z_j^{pq} - z_{j+k}^{pq} \right) \right) \right] \\
&= \frac{6(N-k)^2}{\lambda \theta_1} - 3(N-k)^2 \frac{\omega_p}{\lambda} \\
&\quad + \frac{\omega_p}{\lambda} (N-k)(3N-3k+1) \\
&= \frac{\omega_p(N-k)}{\lambda}.
\end{aligned}$$

$$\begin{aligned}
F_{13} &= \mathbb{E} \left( \frac{\partial \ln f}{\partial \lambda} \frac{\partial \ln f}{\partial \theta_2} \right) \\
&= \frac{6(N-k)^2}{\lambda \theta_2} - \frac{2(N-k)}{\theta_2} \mathbb{E}(\xi) \\
&\quad - 3(N-k) \left( \sum_{j=1}^{N-k} \mathbb{E} \left( y_{j,k}^{mq} \operatorname{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right) \right. \\
&\quad \left. + \sum_{j=1}^{N-k} \mathbb{E} \left( y_{j,k}^{pq} \operatorname{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right) \right) \\
&\quad + \lambda \mathbb{E} \left[ \xi \left( \sum_{j=1}^{N-k} y_{j,k}^{mq} \operatorname{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right. \right. \\
&\quad \left. \left. + \sum_{j=1}^{N-k} y_{j,k}^{pq} \operatorname{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right) \right] \\
&= \frac{6(N-k)^2}{\lambda \theta_2} - \frac{2(N-k)}{\theta_2} \cdot \frac{3(N-k)}{\lambda} \\
&\quad - 6(N-k)^2 \cdot \frac{\omega_q}{\lambda} + \frac{\omega_q}{\lambda} \cdot (N-k)(6N-6k+2) \\
&= \frac{2\omega_q(N-k)}{\lambda}.
\end{aligned}$$

$$\begin{aligned}
F_{22} &= \mathbb{E} \left[ \left( \frac{\partial \ln f}{\partial \theta_1} \right)^2 \right] \\
&= \left( \frac{N-k}{\theta_1} \right)^2 + \lambda^2 \mathbb{E} \left[ \left( \sum_{j=1}^{N-k} y_{j,k}^{mp} \operatorname{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right. \right. \\
&\quad \left. \left. + \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right) \operatorname{sgn} \left( z_j^{pq} - z_{j+k}^{pq} \right) \right)^2 \right] \\
&\quad - \frac{2\lambda(N-k)}{\theta_1} \mathbb{E} \left( \sum_{j=1}^{N-k} y_{j,k}^{mp} \operatorname{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right. \\
&\quad \left. + \sum_{j=1}^{N-k} \operatorname{sgn} \left( z_j^{pq} - z_{j+k}^{pq} \right) \right) \\
&= \left( \frac{N-k}{\theta_1} \right)^2 + \lambda^2 \left( \sum_{j=1}^{N-k} \omega_p^2 \left( \left( s_{j+k}^m - s_j^m \right)^2 + \frac{2}{\lambda^2} \right) \right. \\
&\quad \left. + \left( \frac{\omega_p}{\lambda} \right)^2 (N-k)(N-k-1) \right. \\
&\quad \left. + \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right)^2 \right) - \frac{2\lambda(N-k)}{\theta_1} \cdot (N-k) \cdot \frac{\omega_p}{\lambda} \\
&= \frac{N-k}{\theta_1^2} + \frac{\lambda^2}{\theta_1^2} \sum_{j=1}^{N-k} \left( s_{j+k}^m - s_j^m \right)^2 \\
&\quad + \lambda^2 \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right)^2.
\end{aligned}$$

$$\begin{aligned}
F_{23} &= \mathbb{E} \left( \frac{\partial \ln f}{\partial \theta_1} \frac{\partial \ln f}{\partial \theta_2} \right) \\
&= \frac{2(N-k)^2}{\theta_1 \theta_2} - \frac{\lambda(N-k)}{\theta_1} \mathbb{E} \left( \left( \sum_{j=1}^{N-k} y_{j,k}^{mq} \times \right. \right. \\
&\quad \left. \left. \text{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right) + \sum_{j=1}^{N-k} y_{j,k}^{pq} \text{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right) \\
&\quad - \frac{2(N-k)\lambda}{\theta_2} \mathbb{E} \left( \sum_{j=1}^{N-k} y_{j,k}^{mp} \text{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right. \\
&\quad \left. + \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right) \text{sgn} \left( z_j^{pq} - z_{j+k}^{pq} \right) \right) \\
&\quad + \lambda^2 \mathbb{E} \left[ \left( \sum_{j=1}^{N-k} y_{j,k}^{mp} \text{sgn} \left( z_{j+k}^{mp} - z_j^{mp} \right) \right. \right. \\
&\quad \left. \left. + \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right) \text{sgn} \left( z_j^{pq} - z_{j+k}^{pq} \right) \right) \right. \\
&\quad \left. \times \left( \sum_{j=1}^{N-k} y_{j,k}^{mq} \text{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right. \right. \\
&\quad \left. \left. + \sum_{j=1}^{N-k} y_{j,k}^{pq} \text{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right) \right] \\
&= \frac{2(N-k)^2}{\theta_1 \theta_2} - \frac{2\lambda(N-k)}{\theta_1} \cdot \frac{\omega_q}{\lambda} \cdot (N-k) \\
&\quad - \frac{2(N-k)\lambda}{\theta_2} \cdot (N-k) \cdot \frac{\omega_p}{\lambda} + \frac{2}{\theta_1 \theta_2} \cdot (N-k)^2 \\
&\quad - \frac{\theta_1 \lambda^2}{\theta_2} \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right)^2 \\
&= -\frac{\theta_1 \lambda^2}{\theta_2} \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right)^2.
\end{aligned}$$

$$\begin{aligned}
F_{33} &= \mathbb{E} \left( \frac{\partial \ln f}{\partial \theta_2} \right)^2 \\
&= \frac{4(N-k)^2}{\theta_2^2} + \lambda^2 \mathbb{E} \left[ \left( \sum_{j=1}^{N-k} y_{j,k}^{mq} \text{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right. \right. \\
&\quad \left. \left. + \sum_{j=1}^{N-k} y_{j,k}^{pq} \text{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right)^2 \right] \\
&\quad - \frac{4(N-k)\lambda}{\theta_2} \mathbb{E} \left( \sum_{j=1}^{N-k} y_{j,k}^{mq} \text{sgn} \left( z_{j+k}^{mq} - z_j^{mq} \right) \right. \\
&\quad \left. + \sum_{j=1}^{N-k} y_{j,k}^{pq} \text{sgn} \left( z_{j+k}^{pq} - z_j^{pq} \right) \right) \\
&= \frac{4(N-k)^2}{\theta_2^2} + \frac{\lambda^2}{\theta_2^2} \sum_{j=1}^{N-k} \left( s_{j+k}^m - s_j^m \right)^2 + 2\omega_q^2(N-k) \\
&\quad + 2\omega_q^2(N-k)(N-k-1) + \frac{\lambda^2 \theta_1^2}{\theta_2^2} \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right)^2 \\
&\quad + \frac{2(N-k)}{\theta_2^2} + (N-k)^2 \omega_q^2 - \frac{8(N-k)^2}{\theta_2^2} \\
&= -\frac{(N-k)^2}{\theta_2^2} + \frac{2(N-k)}{\theta_2^2} + \frac{\lambda^2}{\theta_2^2} \sum_{j=1}^{N-k} \left( s_{j+k}^m - s_j^m \right)^2 \\
&\quad + \frac{\lambda^2 \theta_1^2}{\theta_2^2} \sum_{j=1}^{N-k} \left( s_{j+k}^p - s_j^p \right)^2.
\end{aligned}$$

Thereby, we finish the proof.  $\blacksquare$

## REFERENCES

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102-114, Aug. 2002.
- [2] K. Romer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Commun.*, vol. 11, no. 6, pp. 54-61, Dec. 2004.
- [3] N. Bulusu and S. Jha, *Wireless Sensor Networks: A Systems Perspective*. Norwood, MA: Artech House, 2005.
- [4] J. V. Greunen and J. Rabaey, "Lightweight time synchronization for sensor networks," in *Proceedings of the SIGMOD International Conference on Data Management*, 2002, pp. 61-72. ACM, 2002.
- [5] G. Pottie and W. Kaiser, "Wireless integrated network sensors," *Commun. ACM*, vol. 43, no. 5, pp. 51-58, May 2000.
- [6] N. M. Freris, S. R. Graham, and P. R. Kumar, "Fundamental limits on synchronizing clocks over networks," *IEEE Trans. Auto. Control*, vol. 56, no. 6, pp. 1352-1364, June 2011.
- [7] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The flooding time synchronization protocol," in *Proc. 2nd International Conf. Embedded Networked Sensor Systems*, 2004.
- [8] C. Bovy, H. Mertodimedjo, G. Hooghiemstra, U. Uijterwaal, and P. Miegheem, "Analysis of end-to-end delay measurements in internet," in *Proc. Passive Active Meas. Workshop*, Fort Collins, CO, pp. 26-33, Mar. 2002.
- [9] 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.
- [10] D. Jeske, "On the maximum likelihood estimation of clock offset," *IEEE Trans. Commun.*, vol. 53, no. 1, pp. 53-54, Jan. 2005.
- [11] Q. Chaudhari, E. Serpedin, and K. Qaraqe, "On minimum variance unbiased estimation of clock offset in a two-way message exchange mechanism," *IEEE Trans. Inform. Theory*, vol. 56, no. 6, pp. 2893-2904, June 2010.
- [12] A. Ahmad, D. Zennaro, E. Serpedin, and L. Vangelista, "A factor graph approach to clock offset estimation in wireless sensor networks," *IEEE Trans. Inf. Theory*, vol. 58, no. 7, pp. 4244-4260, July. 2012.

- [13] F. Cristian, "Probabilistic clock synchronization," *Distrib. Comput.*, no. 3, pp. 146-158, 1989.
- [14] 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.
- [15] 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.
- [16] S. M. Kay, *Fundamentals of Statistical Signal Processing, Vol. I, Estimation Theory*, Englewood Cliffs, NJ: Prentice Hall, 1993.
- [17] 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.
- [18] M. Leng and Y.-C. Wu, "Low-complexity maximum-likelihood estimator for clock synchronization of wireless sensor nodes under exponential delays," *IEEE Trans. Signal Processing*, vol. 59, no. 10, pp. 4860-4870, October 2011.
- [19] M. Leng and Y.-C. Wu, "On joint synchronization of clock offset and skew for Wireless Sensor Networks under exponential delay," Proceedings of IEEE International Symposium on Circuit and Systems (ISCAS), 2010, pp. 461-464.
- [20] 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.
- [21] Q. M. Chaudhari, E. Serpedin, and J. S. Kim, "Energy efficient estimation of clock offset for inactive nodes in wireless sensor networks," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 582-596, Jan. 2010.
- [22] Q. M. Chaudhari, E. Serpedin, and K. Qaraqe, "Some improved and generalized estimation schemes for clock synchronization of listening nodes in wireless sensor networks," *IEEE Trans. Commun.*, vol. 58, no. 1, pp. 63-67, January 2010.
- [23] A. Ahmad, A. Noor and E. Serpedin, "Joint clock offset and skew estimation for inactive nodes in wireless sensor networks," Conference on Information Sciences and Systems (CISS), Maryland, USA, Mar. 2011, pp. 1-6.
- [24] 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.
- [25] H. Kim, X. Ma and B. R. Hamilton, "Tracking Low-Precision Clocks With Time-Varying Drifts Using Kalman Filtering," *IEEE/ACM Trans. Netw.*, vol. 20, no. 1, pp. 257-270, Jan. 2012.
- [26] Y. Liu, Y. Zhao, L. Chen, J. Pei, J. Han, "Mining frequent trajectory patterns for activity monitoring using radio frequency tag arrays," *IEEE Trans. Parallel and Distributed Systems*, vol. 23, no. 11, pp. 2138-2149, 2012.
- [27] Z. Li, Y. Liu, M. Li, J. Wang, Z. Cao, "Exploiting ubiquitous data collection for mobile users in wireless sensor networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 24, no. 2, pp. 312-326, 2013.
- [28] M. Leng and Y.-C. Wu, "Distributed Clock Synchronization for Wireless Sensor Networks Using Belief Propagation," *IEEE Trans. Signal Processing*, vol. 59, no. 11, pp. 5404-5414, 2011.
- [29] C. Gurwitz, "Weighted median algorithms for  $L_1$  approximation," *BIT Numerical Mathematics*, 301-310. June 1990.



**Xuanyu Cao** is currently pursuing his B.E. degree in electronic engineering in Shanghai Jiao Tong University, China. His research interests are in the area of statistical signal processing as well as asymptotic analysis in wireless networks.



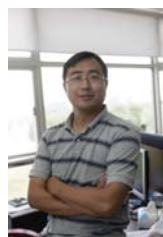
**Feng Yang** received the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2007. Since 2008, he has worked on Shanghai Jiao Tong University. His research interests include multimedia communication and OFDM system design.



**Xiaoying Gan** received the Ph.D. degree in Department of Electronic Engineering from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2006. From 2009 to 2010, she was a visiting researcher with the California Institute for Telecommunications and Information Technology, University of California, San Diego. She is currently with the Institute of Wireless Communication Technology, Department of Electronic Engineering, SJTU, where she is an Associate Professor. Her current research interests concern cognitive network, dynamic spectrum allocation and cellular system optimization.



**Jing Liu** received her Ph.D. degree in Communications and Information Systems from Xidian University, Xian, China, in 2005. Since July 2005, she joined the Department of Electronic Engineering, Shanghai Jiao Tong University, China, where she is currently an Associate Professor. From 2012 to 2013, she worked as a visiting scholar at University of Waterloo, ON, Canada. Her research interests focus on wireless body area network, wireless sensor network and cognitive access networks.



**Liang Qian** joined the Shanghai Jiaotong University (SJTU) in 1989, where he is involved in teaching and research in signal and image processing and wireless communications systems. He received his Ph.D. in Communications and Information Processing from SJTU in 2005, and had been a visiting scholar in Institute of information processing, University of Karlsruhe, in 2003. Dr. Qian's research interests include digital signal processing for wireless cellular system, satellite signal processing for navigation system, emergency wireless access for public security, as well as applications, performance and implementation of above algorithms. Among his current research projects, He is working on system-level integration and advanced algorithms for next generation wireless communications (LTE-A), and performance analysis of adaptive signal and image processing algorithms.



**Xiaohua Tian** received his B.E. and M.E. degrees in communication engineering from Northwestern Polytechnical University, Xi'an, China, in 2003 and 2006, respectively. He received the Ph.D. degree in the Department of Electrical and Computer Engineering (ECE), Illinois Institute of Technology (IIT), Chicago, in Dec. 2010. He is currently a postdoc in Department of Electronic Engineering of Shanghai Jiaotong University, China. He won the Highest Standards of Academic Achievement 2011 of IIT, Fieldhouse Research Fellowship 2009 of IIT, which

is awarded to only one student over IIT each year, and the IEEE EFSOI 2010 Student Travel Grant Award. His research interests include application-oriented networking, Internet of Things and wireless networks. He serves as the guest editor of International Journal of Sensor Networks, publicity co-chair of WASA 2012. He also serves as the Technical Program Committee member for Communications QoS, Reliability, and Modeling Symposium (CQRM) of GLOBECOM 2011, Wireless Networking of GLOBECOM 2013 and WASA 2011.



**Xinbing Wang** received the B.S. degree (with hons.) from the Department of Automation, Shanghai Jiaotong University, Shanghai, China, in 1998, and the M.S. degree from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 2001. He received the Ph.D. degree, major in the Department of electrical and Computer Engineering, minor in the Department of Mathematics, North Carolina State University, Raleigh, in 2006. Currently, he is a faculty member in the Department of Electronic Engineering, Shanghai Jiaotong

University, Shanghai, China. His research interests include scaling law of wireless networks and cognitive radio. Dr. Wang has been an associate editor for IEEE/ACM Transactions on Networking, IEEE Transactions on Mobile Computing, and the member of the Technical Program Committees of several conferences including ACM MobiCom 2012, ACM MobiHoc 2012, IEEE INFOCOM 2009-2014.