Light-Sync: a Low Overhead Synchronization Algorithm for Underwater Acoustic Networks

Davide Zennaro*, Beatrice Tomasi*, Lorenzo Vangelista*, Michele Zorzi*†
*Department of Information Engineering, University of Padova, Italy
†California Institute for Telecommunications and Information Technology, University of California at San Diego, USA
E-mails: { zennarod, tomasibe, zorzi }@dei.unipd.it, lorenzo.vangelista@unipd.it

Abstract—**Clock synchronization is a fundamental requirement in a network of sensor nodes. Underwater sensor networks are a special kind of sensor networks characterized by a strong energy constraint; on the other hand, the low communication results in a less stringent constraint on the synchronization accuracy. In this paper, we propose Light-Sync, a new synchronization protocol for underwater sensor networks in which the best synchronization accuracy is sacrificed to the benefit of a lower communication overhead and finally to a lower consumed energy. We analyze the protocol performance in terms of skew and offset estimation accuracy and compare the estimation variance to the Cramer-Rao Lower Bound (CRLB). We study the energy consumption and synchronization error through an extensive simulation study in an underwater acoustic scenario. Simulation results show that Light-Sync is a suitable choice for synchronizing underwater sensors with an extremely low power consumption.**

I. Introduction

Clock synchronization is a well known fundamental issue in wireless sensor networks (WSNs). Several protocols have been proposed so far to face the problem of reaching a network wide notion of time. Nevertheless, synchronization is still an open problem for underwater acoustic networks (UANs), mainly because they are subject to energy constraints, large propagation delays and time-varying communcations performance. For these reasons, the existing protocols for WSNs are not suitable for UANs, which thus require specifically designed solutions. However, many algorithms and protocols, e.g., time-slotted channel access schemes, require synchronization, thus spurring the interest of the community towards designing solutions tailored to the underwater acoustic scenario.

Among the synchronization protocols proposed in the literature for UANs, the most important are the studies in [1], [2], [3] and [4]. The authors of [1] propose time synchronization for high latency networks (TSHL). This protocol consists of two phases: first, each node estimates its skew with respect to the beacon node through linear regression. Second, the offset is estimated through a two-way message exchange. TSHL is shown to outperform the timing-sync protocol for sensor networks (TPSN) [5], which is a well known synchronization protocol employed in terrestrial wireless networks. The reported results clarify that TSHL has a stronger resilience to large propagation delays with respect to TPSN. Moreover, the authors of [6] propose a more energy efficient version of TSHL. In [2] the authors propose D-Sync as a cross-layer approach for clock synchronization: while an estimate of the Doppler shift is used in order to infer the propagation delay, an ordinary least square estimator is applied to provide an estimate for both clock skew and offset. However, in deriving the estimators, communication channel variability is not taken into account. MU-Sync, proposed in [3], makes use of two-way handshake message exchanges in order to estimate the clock skew and offset of a mobile node with respect to the cluster head, via linear regression. The main limiting factor of this protocol is the large number of messages exchanged between each node and the cluster head. MU-Sync is outperformed by Mobi-Sync [4], in which the estimated quantities, while still computed via linear regression, are instead iteratively updated. However, the cost of the improved synchronization accuracy is an even higher number of transmitted messages. The most energy consuming activity for an acoustic modem is signal transmission, therefore protocols which require the exchange of a large number of messages are not well suited to energy constrained scenarios such as UANs.

The authors of [7] provide a careful overview of the basics of synchronization protocols in WSNs. Specifically, they recognize three synchronization paradigms: the so called Sender-Receiver (SR), Receiver-Only (RO) and Receiver-Receiver (RR). These paradigms can be used to categorize the protocols that are based on clock parameter estimation.

In this paper, starting from the definition of SR and RO paradigms, we propose a hybrid algorithm, Light-Sync, which combines both SR and a modified version of RO in order to synchronize nodes of a UAN to a beacon buoy node. This hybrid approach provides a trade-off between synchronization accuracy and energy consumption thanks to a lower communication overhead. In particular, we compute clock skew and offset estimators with respect to the reference node and derive expressions for the Cramér-Rao Lower Bound (CRLB) for the variance of the estimation error. Numerical simulations confirm the suitability of our protocol for UANs, in terms of both energy consumption and synchronization accuracy.

II. System Model

We consider a static acoustic underwater network composed by a gateway buoy beacon node P, and N nodes anchored at the sea bed, as represented in Fig. 1. We assume that the gateway has unlimited energy resources and perfect timing information, for example by using a GPS service. In this

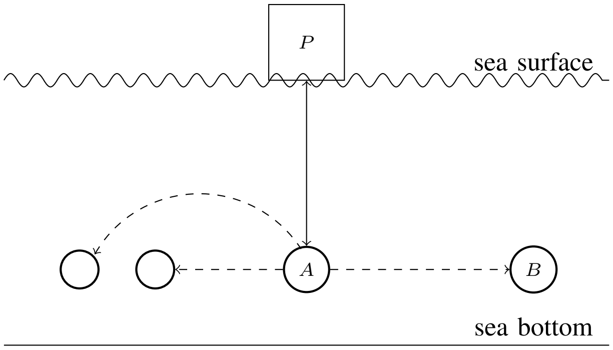

Figure 1. Underwater sensor network composed by N nodes communicating with a buoy node P on the sea surface.

regard, it provides the time reference for the sensors positioned underwater.

As far as the agents synchronization is concerned, the main building block is a clock. The clock in a sensor node is based on a crystal oscillating at a given nominal frequency f_0. The actual frequency f_k of the generic node k's oscillator is affected by environmental factors as well as factory imperfections, so that it deviates from its nominal value, that is

$$f_k = (1 + \theta_k) f_0, \tag{1}$$

with $|\theta_k| < 100$ ppm [8]. Let us define t as the absolute time reference. The counting process of node k, $n_k(t)$, counts the number of clock ticks and can be written as

$$n_k(t) = \lfloor f_k t + n_k(0) \rfloor \, , \tag{2}$$

where $n_k(0)$ is the value of the counter at time 0. By assuming that the clock resolution is infinite[1], the counting process (2) can be considered a continuous time function and the corresponding time process is reformulated as the timer

$$T_k(t) = \frac{f_k}{f_0} t + T_k(0) \ . \tag{3}$$

Recalling that node P provides the time reference to the network, i.e., $T_P(t) = t \ \forall t$ and $f_P = f_0$, we define the *clock offset* $\theta^{(kP)}_{\text{offset}}$ between nodes k and P as follows

$$\theta^{(kP)}_{\text{offset}} \triangleq T_k(0) - T_P(0) = T_k(0) \ , \tag{4}$$

and the *clock skew* $\theta^{(kP)}_{\text{skew}}$ between nodes k and P as

$$\theta^{(kP)}_{\text{skew}} \triangleq \frac{dT_k(t)}{dt} - 1 = \frac{f_k}{f_0} - 1 \ . \tag{5}$$

We can therefore write the clock state (3) as a linear function of t, with parameters $\theta^{(kP)}_{\text{skew}}$ and $\theta^{(kP)}_{\text{offset}}$

$$T_k(t) = (1 + \theta^{(kP)}_{\text{skew}}) t + \theta^{(kP)}_{\text{offset}}. \tag{6}$$

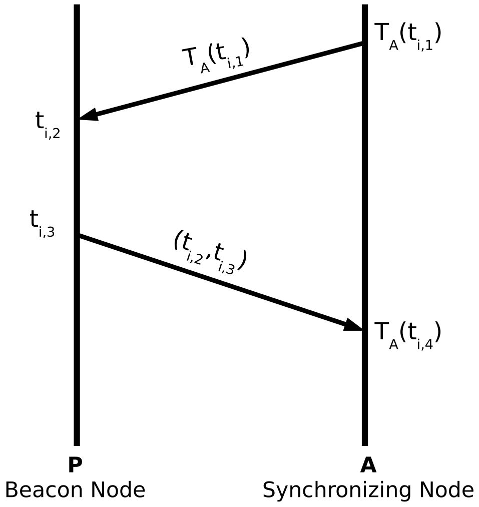

Figure 2. A scheme of the SR two-way message exchanges

Our aim is to design an information exchange protocol for any node k on the sea bed, to estimate the parameters $\theta^{(kP)}_{\text{offset}}$ and $\theta^{(kP)}_{\text{skew}}$ with a reasonable tradeoff between energy consumption and synchronization error. In the next section we describe and study the SR and RO paradigms for UAN and we introduce Light-Sync, that makes use of a customized version of both paradigms in a hybrid fashion.

III. Light-Sync

Light-Sync is a new synchronization protocol that combines two synchronization paradigms, SR and RO, in order to estimate the clock skew and offset of any node on the sea bed with respect to the reference node P with a reasonable tradeoff between energy consumption and synchronization error. As described next, SR and RO are used in two distinct phases of the proposed algorithm. In Phase 1, the SR paradigm is used to estimate clock parameters that are then exploited during Phase 2 via the use of the RO paradigm. Before introducing Light-Sync, we recall the basics of both the SR and the RO paradigm. For a more detailed description of them, we refer the reader to [7].

In the SR paradigm two nodes are actively involved in the message exchange process. We assume that node A aims at synchronizing with node P. To do this, at the generic iteration i node A starts the exchange process by sending a request message to node P, containing its transmission timestamp $T_A(t_{i,1})$. After node P has received the message at time $t_{i,2}$, it waits for a predefined amount of time ϵ, and then it replies to node A with a message carrying its receiving and transmitting timestamps, $t_{i,2}$ and $t_{i,3}$, respectively. Finally, node A records the time $T_A(t_{i,4})$ at which it receives this last message. This procedure is reported in Figure 2. The two way exchange is repeated for a sufficient number of times I such that node A is able to estimate its skew and offset with respect to node P with a certain accuracy.

[1] This assumption is valid if the step size of the clock is small compared to the time duration of the events the clock is measuring, refer to [9] for details on the practical effect of the limited clock resolution for the synchronization problem.

Differently from SR, in the RO paradigm at least three nodes are involved, i.e., a third node indicated as node B. Specifically, node B is supposed to be in the coverage range of both nodes P and A, so that it receives and decodes messages transmitted by both nodes A and P. Node B exploits these messages in order to estimate its clock parameters and therefore synchronize with respect to the reference P [10]. However, in this work we relax the coverage assumption and address the problem of clock synchronization when node B is in the coverage range of A but not in that of the reference P. The motivation for this approach is that we assume that node A is the closest in distance to the buoy node or it uses more transmission power so that it is responsible for communicating with the beacon node P. Given that underwater acoustic communications are subjected to channel quality fluctuations, the other nodes can fall outside the coverage range of the buoy node P for some time, while remaining in the transmission range of node A. Under this assumption and in the worst case scenario, while node A is synchronizing to node P, the generic node B on the sea bed receives the messages sent by node A but not those sent by node P, and needs to exploits just these messages in order to estimate its clock deviation parameters with respect to the buoy node P. Figure 3 describes the message exchange in this extended scenario.

We infer the skew and offset estimator for any node B with respect to the reference beacon node P under the following assumptions: nodes are tethered to the sea bed and propagation delays are statistically distributed according to a Gaussian random variable with an unknown variance and a known average which depends on the node distance and the sound velocity.

A. Algorithm Description

In this section we describe Light-Sync in detail. It consists of two phases: in Phase 1 the goal is to let a node A on the sea bed to synchronize to the beacon node P through estimation of its clock offset and skew with respect to the beacon buoy node P using the SR paradigm. In Phase 2 results of the previous phase are used to let all the remaining nodes on the sea bed synchronize to the reference node. We introduce the estimator formulas for both clock skew and offset of a node's clock with respect to the reference P and introduce the CRLB for evaluating the performance of such estimators.

1) Phase 1*:* Node A computes the estimates of its own clock skew and offset with respect to the reference node P via a two way handshake procedure repeated for I times. We assume that node A waits for Δ seconds between two successive transmissions.

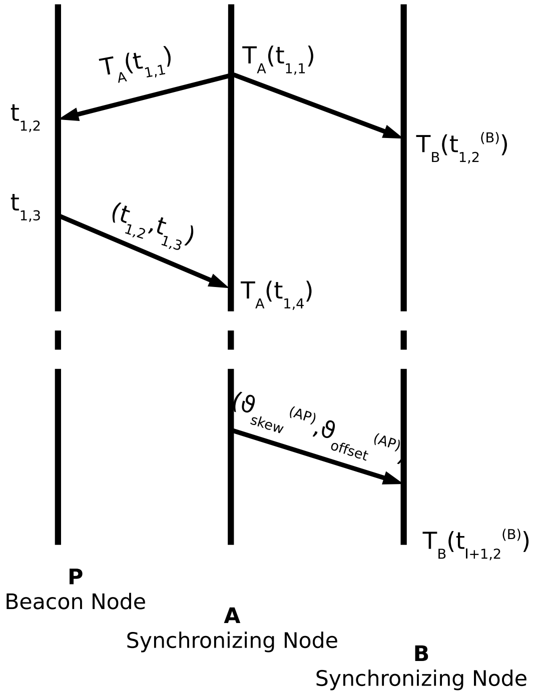

Figure 3. A scheme of the modified SR and RO two-way message exchanges: node B receives the messages sent by node A only.

Regarding the scenario described in Section II and depicted in Figure 2, we can write the equations (see (7) below) which relate the clock state of node A to the time reference of node P during iteration $i \in [1, \dots, I]$. The impairments in the message exchange come from the random delays encountered in the procedure. Such delays can be divided into two categories, depending on their nature [11]. Let $d_{i,AP}$ be the deterministic part of the message transmission delay between nodes A and P at iteration i. We assume $d_{i,AP}$ to be known, since it can be precomputed in the deployment phase of the network or estimated via an extra message exchange between node P and node A, possibly exploiting piggybacking. On the other hand, the random delays in uplink and downlink are modelled as $i.i.d.$ zero mean Gaussian random variables $X_{i,AP}$ and $Y_{i,AP}$, respectively, with common variance equal to σ_{AP}^2. The overall delays in uplink and downlink can then be written as $d_{i,AP} + X_{i,AP}$ and $d_{i,AP} + Y_{i,AP}$, respectively.

Given the previous discussion, the following relation between absolute times holds [12]

$$t_{i,2} = t_{i,1} + d_{i,AP} + X_{i,AP} \ , \tag{7}$$

and we can write the i-th timestamp transmitted by node A as

$$T_A(t_{i,1}) = (1 + \theta_{\text{skew}}^{(AP)})(t_{i,2} - d_{i,AP} - X_{i,AP}) + \theta_{\text{offset}}^{(AP)} \ , \tag{8}$$

where we used (6). Analogously, by considering that

$$t_{i,4} = t_{i,3} + d_{i,PA} + Y_{i,AP} \tag{9}$$

and using (6) again, we can write the i-th timestamp received by node A as

$$T_A(t_{i,4}) = (1 + \theta_{\text{skew}}^{(AP)})(t_{i,3} + d_{i,AP} + Y_{i,AP}) + \theta_{\text{offset}}^{(AP)} \ . \tag{10}$$

We remark that, at this point, the observations available at node A for iteration i are $T_A(t_{i,1})$, $t_{i,2}$, $t_{i,3}$ and $T_A(t_{i,4})$. Assuming that $t_{i,3} = t_{i,2} + \epsilon$, with ϵ a design parameter, and using equations (8) and (10), the observations can be recombined as

$$\begin{aligned} z_i \triangleq & \frac{1}{2}\left[(T_A(t_{i,1}) - t_{i,2}) + (T_A(t_{i,4}) - t_{i,3})\right] = \\ = & \theta_{\text{skew}}^{(AP)}\left[t_{i,2} + \frac{\epsilon}{2} + \frac{Y_{i,AP} - X_{i,AP}}{2}\right] + \\ & + \theta_{\text{offset}}^{(AP)} + \frac{Y_{i,AP} - X_{i,AP}}{2} \ , \end{aligned} \tag{11}$$

where we have introduced a new variable z_i. Since it is generally true that

$$t_{i,2} \gg \epsilon + Y_{i,AP} - X_{i,AP}\theta_{\text{skew}}^{(AP)} \ , \tag{12}$$

we can write z_i as follows

$$z_i \simeq \theta_{\text{skew}}^{(AP)} t_{i,2} + \theta_{\text{offset}}^{(AP)} + w_i \ , \tag{13}$$

where the random variable

$$w_i \triangleq \frac{Y_{i,AP} - X_{i,AP}}{2} \sim \mathcal{N}\left(0, \sigma_w^2 \triangleq \frac{\sigma_{AP}^2}{2}\right) \tag{14}$$

summarizes the Gaussian random delays. By stacking the observations z_i, $i = 1, \ldots, I$, in vector form, we eventually have the following linear system in matrix form

$$\boldsymbol{z} = \boldsymbol{H}_{1,I}\boldsymbol{\theta}^{(AP)} + \boldsymbol{w} \tag{15}$$

where

$$\begin{aligned} \boldsymbol{z} & \triangleq [z_1 z_2 \cdots z_I]^T \\ \boldsymbol{\theta}^{(AP)} & \triangleq [\theta_{\text{skew}}^{(AP)}\theta_{\text{offset}}^{(AP)}]^T \\ \boldsymbol{w} & \triangleq [w_1 w_2 \cdots w_I]^T \end{aligned} \tag{16}$$

and

$$\boldsymbol{H}_{1,I} \triangleq \begin{bmatrix} t_{1,2} & 1 \\ t_{2,2} & 1 \\ \vdots & \vdots \\ t_{I,2} & 1 \end{bmatrix} \tag{17}$$

collects node P's timestamps. Since the observation model is linear, the best linear unbiased estimator (BLUE) can be computed as [13]

$$[\hat{\theta}_{\text{skew}}^{(AP)}\hat{\theta}_{\text{offset}}^{(AP)}]^T = (\boldsymbol{H}_{1,I}^T\boldsymbol{H}_{1,I})^{-1}\boldsymbol{H}_{1,I}^T\boldsymbol{z} \ , \tag{18}$$

while the CRLB for the variance of the estimation error turns out to be [13]

$$\boldsymbol{C}_{AP} = \sigma_w^2(\boldsymbol{H}_{1,I}^T\boldsymbol{H}_{1,I})^{-1}. \tag{19}$$

Since the noise affecting the observations $\boldsymbol{z}$ is Gaussian, the estimator (18) is efficient, i.e., the variance of the estimation error reaches the CRLB [13]. Phase 1 ends with node A transmitting an extra broadcast message carrying its clock skew and offset estimates (18).

2) Phase 2: We now address the problem of synchronizing a node B to node A by assuming that node B is in the coverage area of node A but not in the coverage area of node P: node B hears timestamp messages from A toward P but not vice-versa as shown in Figure 3. Node B timestamps the local time instants $T_B(t_{i,2}^{(B)})$ at which it receives the messages from node A. As we have seen, after Phase 1 node A has computed an estimate of its offset and skew with respect to the reference node. After receiving the last message from node A, node B is able to compute the estimates $\hat{t}_{i,1}$ of $t_{i,1}$ as

$$\hat{t}_{i,1} \triangleq \frac{T_A(t_{i,1}) - \hat{\theta}_{\text{offset}}^{(AP)}}{1 + \hat{\theta}_{\text{skew}}^{(AP)}} \ . \tag{20}$$

This time estimate is useful to node B for getting an estimate of its clock parameters. In fact, similarly to Phase 1, given that the deterministic portion of the delay $d_{i,AB}$ is known and the random portion of the downlink delay $X_{i,AB}$ is a zero mean Gaussian random variable with variance σ_{AB}^2, the following relations hold

$$\begin{aligned} t_{i,2}^{(B)} = & t_{i,1} + d_{i,AB} + X_{i,AB} \\ T_B(t_{i,2}^{(B)}) = & (1 + \theta_{\text{skew}}^{(BP)})t_{i,2}^{(B)} + \theta_{\text{offset}}^{(BP)} = \\ = & (1 + \theta_{\text{skew}}^{(BP)})(t_{i,1} + d_{i,AB} + X_{i,AB}) + \theta_{\text{offset}}^{(BP)}. \end{aligned} \tag{21}$$

Using an approach similar to [14], node B subtracts from $T_B(t_{i,2}^{(B)})$ the known value of the deterministic delay $d_{i,AB}$ and the estimate (20), i.e.,

$$y_i^{(B)} \triangleq T_B(t_{i,2}^{(B)}) - d_{i,AB} - \hat{t}_{i,1} \ . \tag{22}$$

By combining (21) and (22) the observations can be written in vector form as

$$\boldsymbol{y}^{(B)} = \hat{\boldsymbol{H}}_{1,I}\boldsymbol{\theta}^{(BP)} + \boldsymbol{v}^{(B)} \ , \tag{23}$$

where

$$\boldsymbol{y}^{(B)} \triangleq [y_1^{(B)}y_2^{(B)} \cdots y_I^{(B)}]^T \tag{24}$$

$$\boldsymbol{\theta}^{(BP)} \triangleq [\theta_{\text{skew}}^{(BP)}\theta_{\text{offset}}^{(BP)}]^T \tag{25}$$

$$\boldsymbol{v}^{(B)} \triangleq [v_1^{(B)}v_2^{(B)} \cdots v_I^{(B)}]^T \tag{26}$$

$$\hat{\boldsymbol{H}}_{1,I} \triangleq \begin{bmatrix} \hat{t}_{1,1} & 1 \\ \hat{t}_{2,1} & 1 \\ \vdots & \vdots \\ \hat{t}_{I,1} & 1 \end{bmatrix} \tag{27}$$

with $v_i^{(B)}$ being the sum of two terms, i.e.,

$$v_i^{(B)} \triangleq v_{i,1}^{(B)} + v_{i,2}^{(B)} \ , \tag{28}$$

in which the impairments coming from the random delays and the error in estimating $t_{i,1}$ have been separated. More specifically, under the assumption that $d_{i,AB}\theta_{\text{skew}}^{(BP)} \ll X_{i,AB}(1 + \theta_{\text{skew}}^{(BP)})$, which is verified in the considered underwater case, we can write

$$
\begin{aligned}
v_{i,1}^{(B)} \triangleq& (1+\theta_{\text{skew}}^{(BP)})X_{i,AB} + d_{AB}\theta_{\text{skew}}^{(BP)} \\
\simeq& (1+\theta_{\text{skew}}^{(BP)})X_{i,AB} \ , \qquad (29)
\end{aligned}
$$

so that $v_{i,1}^{(B)} \sim \mathcal{N}\left(0, \sigma^2_{v_{i,1}^{(B)}}\right)$, with

$$\sigma^2_{v_{i,1}^{(B)}} = (1+\theta_{\text{skew}}^{(BP)})^2 \sigma^2_{BP} \ . \qquad (30)$$

On the other hand, the error in the estimation process (20) is taken into account in the second noise term, i.e.,

$$v_{i,2}^{(B)} \triangleq (1+\theta_{\text{skew}}^{(BP)})(t_{i,1} - \hat{t}_{i,1}) \ . \qquad (31)$$

In order to describe the statistics of $v_{i,2}^{(B)}$, we need to recall that the estimator (18) is the minimum variance estimator, therefore we can write the estimation errors as

$$
\begin{aligned}
e_{i,s} &\triangleq \hat{\theta}_{\text{skew}}^{(AP)} - \theta_{\text{skew}}^{(AP)} && (32)\\
&\sim \mathcal{N}\left(0, C_{AP}(1,1)\right) && (33)\\
e_{i,o} &\triangleq \hat{\theta}_{\text{offset}}^{(AP)} - \theta_{\text{offset}}^{(AP)} && (34)\\
&\sim \mathcal{N}\left(0, C_{AP}(2,2)\right) \ . && (35)
\end{aligned}
$$

Moreover, the error in estimating $t_{i,1}$ can be expressed as

$$t_{i,1} - \hat{t}_{i,1} = \frac{1}{1 + e_{i,s} + \theta_{\text{skew}}^{(AP)}} \left[e_{i,s} t_{i,1} + e_{i,o}\right] \ ,$$

and its variance $\sigma^2_{t,i}$ is given by

$$
\begin{aligned}
\sigma^2_{t,i} \triangleq& \mathbb{E}\left[\left(t_{i,1} - \hat{t}_{i,1}\right)^2\right] \\
=& t_{i,1}^2 \mathbb{E}\left[\frac{e_{i,s}^2}{\left(1 + e_{i,s} + \theta_{\text{skew}}^{(AP)}\right)^2}\right] + \\
&+ C_{AP}(2,2) \cdot \mathbb{E}\left[\frac{1}{\left(1 + e_{i,s} + \theta_{\text{skew}}^{(AP)}\right)^2}\right] \ ,
\end{aligned}
$$

so that we can write

$$v_{i,2}^{(B)} \sim \mathcal{N}\left(0, \sigma^2_{v_{i,2}^{(B)}} \triangleq \sigma^2_{t,i}\left(1+\theta_{\text{skew}}^{(BP)}\right)^2\right) . \qquad (36)$$

As a result, since $v_{i,1}^{(B)}$ and $v_{i,2}^{(B)}$ are independent, the overall noise term is Gaussian distributed, i.e.,

$$v_i^{(B)} \sim \mathcal{N}\left(0, \sigma^2_{v_i^{(B)}} = \sigma^2_{v_{i,1}^{(B)}} + \sigma^2_{v_{i,2}^{(B)}}\right) . \qquad (37)$$

As for Phase 1, we can compute the BLUE estimator for $\boldsymbol{\theta}^{(BP)}$ in (23), which is given by [13]

$$[\hat{\theta}_{\text{skew}}^{(BP)} \hat{\theta}_{\text{offset}}^{(BP)}]^T = (\hat{\boldsymbol{H}}_{1,I}^T \hat{\boldsymbol{H}}_{1,I})^{-1} \hat{\boldsymbol{H}}_{1,I}^T \boldsymbol{y}^{(B)} \ , \qquad (38)$$

and the CRLB for the estimation error as [13]

$$\boldsymbol{C}_{AB} = \sigma^2_{v_i^{(B)}} (\hat{\boldsymbol{H}}_{1,I}^T \hat{\boldsymbol{H}}_{1,I})^{-1} . \qquad (39)$$

We recall that matrix $\hat{\boldsymbol{H}}_{1,I}$, defined in (27) and used by the estimator (38), consists of the time estimates $\hat{t}_{i,1}$, in (20), obtained from $\hat{\theta}_{\text{skew}}^{(AP)}$ and $\hat{\theta}_{\text{offset}}^{(AP)}$, which are noisy estimates of the clock skew and offset between nodes A and P, as defined in (18).

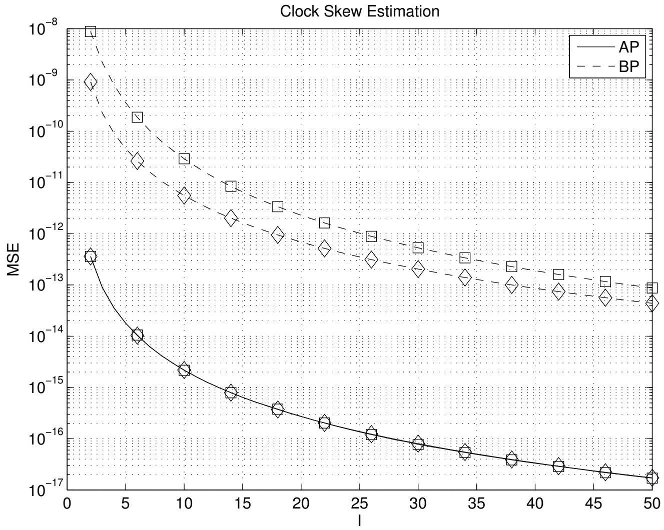

Figure 4. Variance of the estimation error vs I in estimating $\theta_{\text{skew}}^{(AP)}$ with $\Delta = 100$ s. Square markers: estimation variance - Diamond markers: CRLB.

IV. Performance Analysis

In this section, we perform a statistical analysis of the proposed estimators (18) and (38) for Light-Sync and provide simulation results.

The underwater sensor nodes are deployed randomly in a circular area of radius $R = 5000$ m. The distance between the buoy and node A is set to 100 m. The nominal oscillation frequency is $f_0 = 32768$ Hz while the standard deviations for the skew and the offset are set to 50 ppm and 10^{-2}, respectively. Thanks to data gathered in real experiments, we are able to determine the deterministic portions of the delays d as well as the deviation parameters σ_{AP} and σ_{AB}. Finally, we set $\epsilon = 10^{-2}$.

A. Estimation performance

Since the estimators proposed in Section III are unbiased, we study the performance of the synchronization protocol in terms of mean squared error (MSE) in clock parameter estimation

$$\text{MSE}\,(\theta) \triangleq \mathbb{E}\left[\left(\theta - \hat{\theta}\right)^2\right] \ ,$$

and compare it with the corresponding CRLB by letting the number of iterations I vary. We notice that the estimation accuracy is independent of the number of nodes N in the network, since Phase 1 is itself independent of this parameter and Phase 2 involves each sensor node separately, without any information exchange between them.

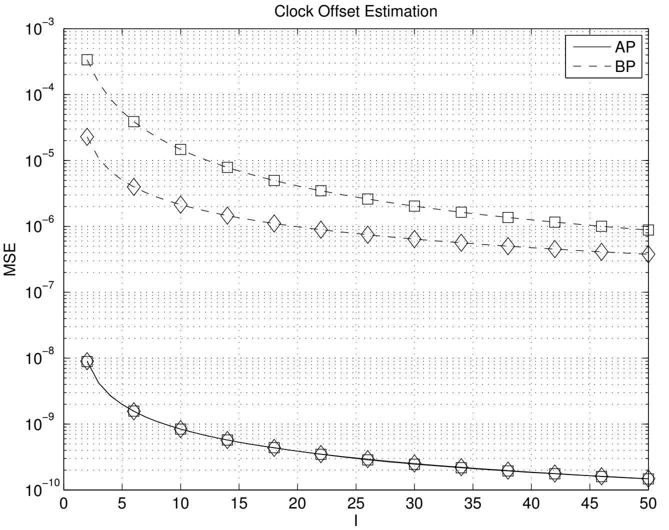

Figure 5. Variance of the estimation error vs I in estimating $\theta^{(AP)}_{\text{offset}}$ with $\Delta = 100$ s. Square markers: estimation variance - Diamond markers: CRLB.

1) Estimators optimality: Figs. 4 and 5 show the MSE and the corresponding CRLB for the estimation of skew and offset, respectively, when varying the number of iterations I. It can be observed that the proposed estimators for $\theta^{(AP)}_{\text{skew}}$ and $\theta^{(AP)}_{\text{offset}}$ are efficient since the MSE reaches the CRLB. On the other hand, the MSE in estimating $\theta^{(BP)}_{\text{skew}}$ and $\theta^{(BP)}_{\text{offset}}$ is slightly greater than that obtained for AP. Moreover, these estimators are not efficient, since the MSE does not reach the CRLB. This can be explained by noting that the estimator in equation (38) uses the noisy estimates in equation (18) as noticed in Section III-A2.

2) Beacon interval: Figs. 6 and 7 show the MSE in estimating $\theta^{(BP)}_{\text{skew}}$ and $\theta^{(BP)}_{\text{skew}}$, respectively, when using different values of the pre-defined time Δ between two successive message transmissions by node A in Phase 1. It can be observed how prolonging Δ increases the skew estimation accuracy. In fact, intuitively speaking, the longer the interval between two noisy points on a line, the more accurate the estimate of the slope. Regarding clock offset estimation, the MSE in estimating the clock offset decreases as Δ increases, but the CRLB does not depend strongly on such a parameter, showing that increasing Δ has a positive effect also on estimating the clock offset.

B. Synchronization Error

The fundamental benefit of correctly estimating clock skews and offsets is that a correction to the clock timers (6) can be applied, i.e., the corrected version of the timer $T_k(t)$ can be written as

$$\hat{T}_k(t) = \frac{1}{1+\hat{\theta}^{(kP)}_{\text{skew}}} \left(T_k(t) - \hat{\theta}^{(kP)}_{\text{offset}}\right). \tag{40}$$

Fig. 8 shows the differences between the corrected timers (40) and the absolute reference time. It can be observed that the clock errors decrease as the clock parameter estimation is improved.

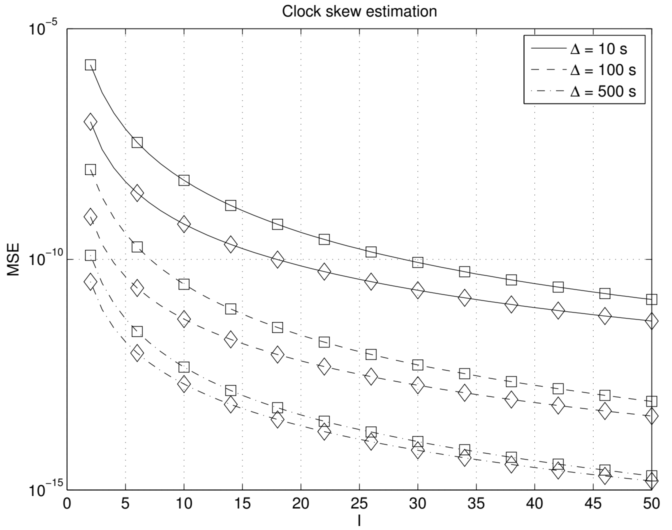

Figure 6. Variance of the estimation error for $\hat{\theta}^{(BP)}_{\text{skew}}$ varying Δ. Square markers: estimation variance - Diamond markers: CRLB.

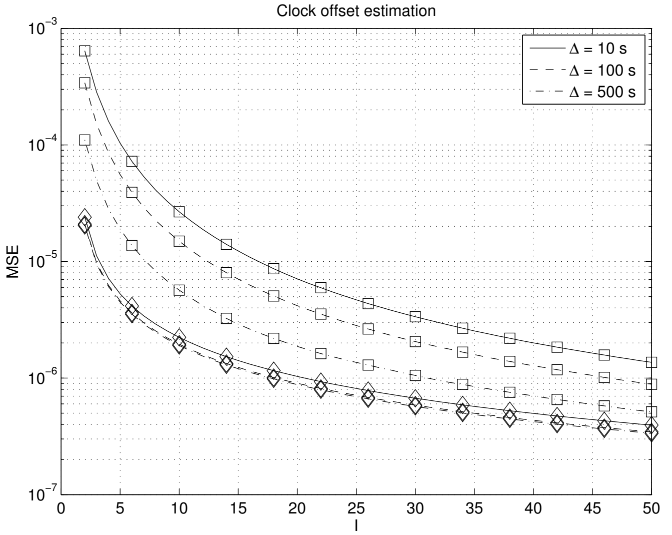

Figure 7. Variance of the estimation error for $\hat{\theta}^{(BP)}_{\text{offset}}$ varying Δ. Square markers: estimation variance - Diamond markers: CRLB.

C. Communication overhead

We now study the communication protocol overhead in terms of number of transmitted messages as a function of the number of iterations, I. During Phase 1 of Light-Sync, $2I+1$ messages are exchanged, I of them actually transmitted by the buoy (that is ideally provided with a long lasting energy resource), whereas no message is transmitted by node A during Phase 2. Therefore the total communication overhead of Light-Sync consists of $I+1$ messages transmitted by the sensor nodes on the sea bed. Note that this number is independent of the number of nodes N.

As a matter of comparison, for I iterations TSHL requires $I+N$ messages transmitted by the beacon node (that must be in the coverage range of all nodes, which is not needed in

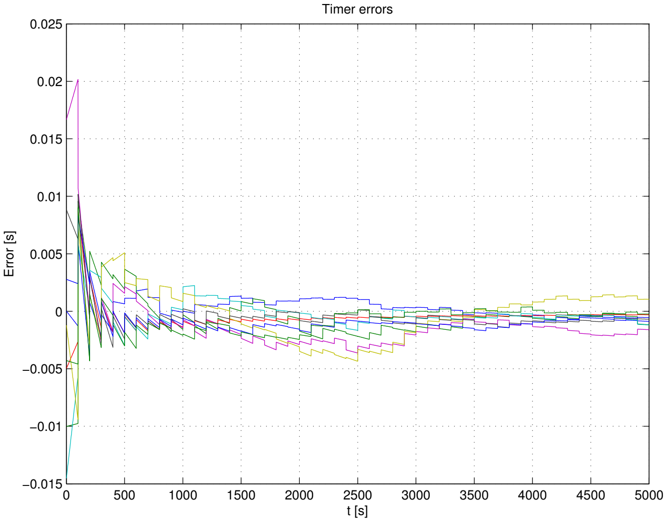

Figure 8. Timer errors for the $N = 10$ nodes in the network.

Light-Sync) plus 1 message transmitted by each node of the network, which means a number of messages which grows linearly with the number of nodes N, thus leading to a higher and less scalable communication overhead with respect to Light-Sync.

V. Conclusions

In this work we introduced Light-Sync, a new synchronization algorithm for UANs which allows sensor nodes deployed on the sea bed to synchronize to a beacon node fluctuating on the sea surface, representing the time reference for the network. Light-Sync provides a trade-off between synchronization accuracy and power consumption, therefore it is extremely well suited for power constrained and low bitrate UANs. Estimator expressions were derived for clock deviation parameters using classical estimation tools. Finally, the performance of Light-Sync was analysed via an extensive simulation study in terms of MSE and synchronization error, showing that Light-Sync can be successfully employed in UANs.

Acknowledgment

This work has been supported in part by the European Commission under the Seventh Framework Programme (Grant Agreement 258359 – CLAM) and by the US National Science Foundation under Grant no. CNS-1035828.

References

[1] A. Syed and J. Heidemann, "Time synchronization for high latency acoustic networks," in *Proc. of the IEEE Infocom*, Barcelona (Spain), 2006.

[2] F. Lu, D. Mirza, and C. Schurgers, "D-Sync: Doppler-Based Time Synchronization for Mobile Underwater Sensor Networks," in *Proc. of ACM WUWNet*, Woods Hole, MA, Sep. 2010.

[3] N. Chirdchoo, W. S. Soh, and K. C. Chua, "MU-Sync: A Time Synchronization Protocol for Undewater Mobile Networks," in *Proc. of ACM WUWNet*, San Francisco, CA, Sep. 2008.

[4] J. Liu, Z. Zhou, Z. Peng, and J. H. Cui, "Mobi-Sync: Efficient Time Synchronization for Mobile Underwater Sensor Networks," in *Proc. of ACM WUWNet*, Berkeley, CA, Nov. 2009.

[5] R. Kumar, S. Ganeriwal, and M. B. Srivastava, "Timing-sync Protocol for Sensor Networks," in *ACM Conf. Embedded Networked Sensor Systems*, 2003.

[6] Y. Wang, B. Huang, F. Wang, and F. Zhang, "Reduce time synchronization cost for high latency and resource-constrained sensor networks," in *Proc. of the Seventh International Conference on Networking*, Cancun (Mexico), 2008.

[7] I. K. Rhee, J. Lee, J. Kim, E. Serpedin, and Y. C. Wu, "Clock Synchronization in Wireless Sensor Networks: An Overview," *Sensors*, vol. 9, no. 1, pp. 56–85, 2009.

[8] B. Sadler, "Fundamentals of energy-constrained sensor network systems," *IEEE Aerospace and Electronic Systems Magazine*, vol. 20, no. 8, pp. 17 –35, Aug. 2005.

[9] B. M. Sadler, "Local and broadcast clock synchronization in a sensor node," *IEEE Trans. Signal Processing*, vol. 13, no. 1, pp. 9–12, 2006.

[10] K. Noh, E. Serpedin, and K. 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, 2008.

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

[12] H. Abdel-Ghaffar, "Analysis of synchronization algorithms with time-out control over networks with exponentially symmetric delays," *IEEE Transactions on Communications,*, vol. 50, no. 10, pp. 1652 – 1661, Oct. 2002.

[13] S. M. Kay, *Fundamentals of statistical signal processing. Estimation theory*. Prentice-Hall, 1993.

[14] M. Maroti, B. Kusy, G. Simon, and A. Lédeczi, "The flooding time synchronization protocol," in *ACM Conf. On Embedded Networked Sensor Systems*, 2004, pp. 39–49.