Analysis of EMU-Sync: A Time Synchronization Protocol for Underwater Mobile Networks

Chairat Phongphanphanee[1], Pollawat Vonlopvisut[2], Nitthita Chridchoo[3] and Lunchakorn Wuttisittikulkij[2]

[1]Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Phayathai Rd., Pathumwan, Bangkok 10330, Thailand
[2]Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University, Phayathai Rd., Pathumwan, Bangkok 10330, Thailand
[3]Department of Telecommunication Engineering, Faculty of Engineering, Nakhon Pathom Rajabhat University, 85 Malaiman Rd., Muang, Nakhon Pathom, 73000, Thailand
E-mail: chai.p@chula.ac.th, pollawat.von@gmail.com, nitthita@yahoo.com, wlunchak@chula.ac.th

Abstract

Although there are numerous studies in time synchronization for underwater mobile networks, none of these provide the analysis on the effect of node mobility on its performance. In this paper we propose an analysis model with a cluster node and a neighboring node moving in one-dimension. We compare the skew estimation error of Enhanced Mobile Underwater Synchronization (EMU-Sync) protocol and Mobile Underwater Synchronization (MU-Sync) protocol. The analysis shows that MU-Sync provides better accuracy when the cluster node is static or the relative velocity approaches zero. But EMU-Sync can outperform MU-Sync when the cluster node is moving and the relative velocity of the neighboring node is not zero.

Keywords: Underwater network, time synchronization, sensor node, propagation delay

1. Introduction

Time synchronization for terrestrial wireless sensor networks have been well developed, but these known techniques cannot be directly applied to underwater environment for the following reasons [1][2]. Underwater wireless sensor networks rely on acoustic communications, rather than radio or optical communications due to their severely high attenuation though water. In addition, the underwater acoustic channel has unique characteristics, including low available bandwidth, long propagation delay, node mobility and high error probability.

Several underwater time synchronization protocols have been proposed in literature such as Time Synchronization for High Latency (TSHL) [1], MU-Sync [2], EMU-Sync [3], Doppler Synchronization (D-Sync) [4] and Doppler Assisted Synchronization (DA-Sync) [5]. TSHL was designed primarily for high latency networks. TSHL estimates skew and offset by using linear regression. However, TSHL assumes that all sensor nodes are in static state thus not applicable for mobile networks. In contrast, MU-Sync, a cluster-based protocol, was designed for mobile networks. It estimates skew and offset by using two steps of linear regressions. The first linear regression is aimed to resolve the response time, whereas the second linear regression is used to estimate the final values of skew and offset. Although MU-Sync is more accurate than THSL in mobile context, MU-Sync encounters some loss of accuracy due to propagation delay estimation error in some mobility scenario involving the movement of neighboring and cluster nodes. EMU-Sync was proposed as an enhanced version of MU-Sync by exploiting the time stamp data of messages sent between clusters and neighboring nodes in both directions instead of one direction in MU-Sync. The simulation results showed that EMU-sync offers better performance than MU-Sync. However, there was no in depth analysis to explain the reason of performance gain in EMU-Sync protocol. Therefore, in this paper, we aim to provide an analysis of the EMU-Sync performance in comparison to MU-Sync. We present some analysis of the effect of long and dynamic propagation delay spread on the estimation of skew and how to reduce errors.

It is worth noting that there are other interesting protocols in literature, such as D-Sync and DA-Sync. These protocols are known to offer good error performance, but they require some physical layer enhancement in nodes to acquire additional information, i.e. Doppler shift, to help calculate the propagation delay more accurately. This process clearly requires high complexity and might be difficult to achieve.

The rest of this paper is organized as follows. In Section II, we analyze error of EMU-Sync and MU-Sync. Next, the results are given in Section III. Finally, we give our conclusion in Section IV.

2. Error Analysis

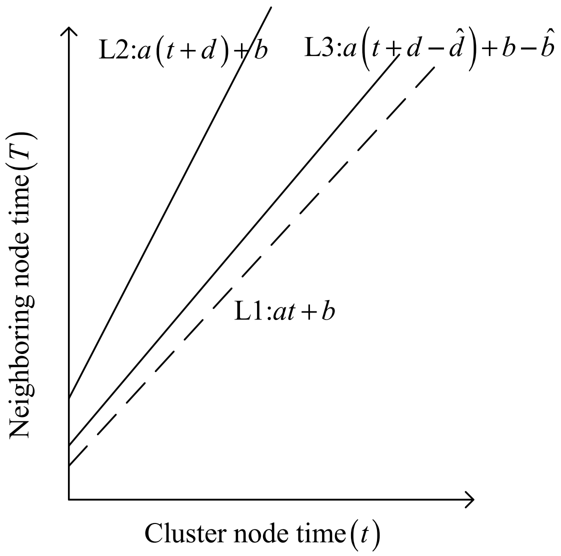

Fig.1 Impact of skew, offset and propagation delay

In general, time synchronization uses two parameters namely skew and offset as equation (1).

$$T = at + b \tag{1}$$

where a, b are skew and offset respectively and t, T are cluster node time and neighboring node time respectively. d is the propagation delay. Basically, skew and offset can be found by using linear regression. In Fig.1, L1 governs the relation between reference and local time. L2 is time synchronization protocol but it ignores long and dynamic propagation delay in UWSNs. Skew and offset can be determined for time synchronization protocols for UWSNs in L3. Because of unique characteristics and node mobility in UWSNs, it is impossible to calculate real propagation delay. Therefore, most of the existing time synchronization protocols for UWSNs use half of the round trip time to calculate propagation delay.

MU-Sync and EMU-Sync were designed to operate in two phases, namely the skew and offset acquisition phase and the synchronization phase. In the first phase, the protocol estimates the clock skew and offset by performing linear regression twice over a set of n reference beacons. The first regression gives $\hat{a}$, an estimated skew, that allows the cluster head to compute the propagation delay that each reference packet encounters and the adjusted timing of the neighboring node time stamp.

For MU-Sync, propagation delay uses half of the round trip time as in equation (2).

$$\hat{d}_{1,i} = \frac{t_{4,i} - t_{1,i} + \frac{\left(T_{2,i} - T_{3,i}\right)}{\hat{a}}}{2} \tag{2}$$

where $t_{1,i}$, $t_{4,i}$ are local time stamps at the cluster node and $T_{2,i}$, $T_{3,i}$ are the time stamps at the neighboring node.

Estimation of the final skew $\hat{\hat{a}}$, can be determined by equation (2).

$$\hat{\hat{a}}_{MU} = \frac{n\sum t_{1,i}\hat{T}_{2,i} - \sum t_{1,i}\sum \hat{T}_{2,i}}{n\sum t_{1,i}^2 - \left(\sum t_{1,i}\right)^2} \tag{3}$$

$$\hat{T}_{2,i} = T_{2,i} - \hat{d}_{1,i} \tag{4}$$

where $\hat{T}_{2,i}$ is the adjusted time stamp.

By substitute (2) and (4) into (3) can be transformed to equation (5) we obtain the ratio of skew estimated error, ξ_{MU}, as:

$$\xi_{MU} = \frac{\hat{\hat{a}}_{\mathrm{MU}} - a}{a} \tag{5}$$

$$\xi_{MU} = \frac{n\sum t_{1,i}\left(d_{1,i} - \frac{\hat{d}_{1,i}}{a}\right) - \sum t_{1,i}\sum\left(d_{1,i} - \frac{\hat{d}_{1,i}}{a}\right)}{n\sum t_{1,i}^2 - \left(\sum t_{1,i}\right)^2} \tag{6}$$

For EMU-Sync, the cluster node additionally performs the linear regression on $T_{3,i}$ and $t_{4,i}$ data :

$$\hat{a}_N = \frac{(n-1)\sum T_{3,i}\hat{t}_{4,i} - \sum T_{3,i}\sum \hat{t}_{4,i}}{(n-1)\sum T_{3,i}^2 - \left(\sum T_{3,i}\right)^2} \tag{7}$$

where $\hat{t}_{4,i} = t_{4,i} - \hat{d}_{2,i}$ and

$$\hat{d}_{2,i} = \frac{T_{2,i+1} - T_{3,i} + \frac{\left(t_{4,i} - t_{1,i+1}\right)}{\hat{a}_N}}{2} \tag{8}$$

where $\hat{a}_N$ is the estimated skew obtained from the first linear regression from the neighboring node perspective i.e., $t_{4,i}$ over $T_{3,i}$.

The final estimated skew can be obtain from:

$$\hat{\hat{a}}_{EMU} = \frac{\left(\hat{\hat{a}}_{MU} + \frac{1}{\hat{\hat{a}}_N}\right)}{2} \tag{8}$$

Thus we obtain the ratio of skew estimation error as:

$$\xi_{EMU} = \frac{\left(\xi_{MU} + \frac{1}{\hat{\hat{a}}_N a} - 1\right)}{2} \tag{9}$$

Next we model a one dimensional movement of two sensor nodes as depicted in Fig.2, where v_c, v_n are velocity of cluster and neighboring node respectively. We also neglect jitter and assume that the response times, $T_{3,i} - T_{2,i}$ and $t_{1,i+1} - t_{4,i}$ are zero.

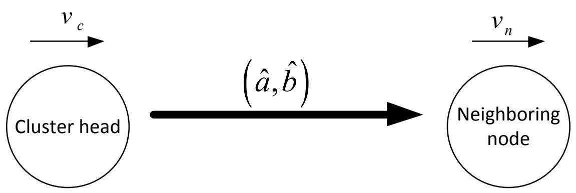

Fig. 2 Movement of cluster and neighboring node

Therefore, ξ_{MU} from equation (6) can be transformed to equation (10) which is the velocity form:

$$\xi_{MU} = \frac{2av_r\left(v_s+v_c\right)-v_r\left(v_s+v_c\right)-v_r\left(v_s-v_r-v_c\right)-v_r^2}{2a\left(v_s+v_c\right)\left(v_s-v_r-v_c\right)} \quad (10)$$

Where $v_r = v_n - v_c$ is the relative velocity between the neighboring node and the cluster node and v_s is speed of sound, approximately 1500 m/s. We can also calculate the ratio of skew estimated error for EMU-Sync as in equation (11)

$$\xi_{EMU} = \frac{a^2v_r\left(v_s+v_c\right)-v_r\left(v_s-v_r-v_c\right)}{2a^2\left(v_s+v_c\right)\left(v_s-v_r-v_c\right)} \quad (11)$$

3. Results

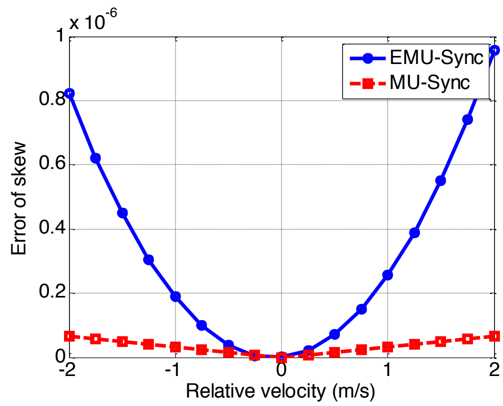

Fig.3 The effect of relative velocity in the skew estimation when $v_c = 0$

Fig.3 shows the error of skew estimation when the cluster node is stationery. Under this condition MU-Sync performs better than EMU-Sync. The error of EMU-Sync is higher when sensor nodes move faster toward each other or apart from each other whereas the error of MU-Sync remains almost constant. The increase in the skew error is around 0.75 ppm. Fig.4 shows the error of skew when the cluster node moves at 2 m/s. The skew estimation error of MU-Sync increase linearly with the relative velocity , while the error of EMU-Sync stays below 1 ppm.

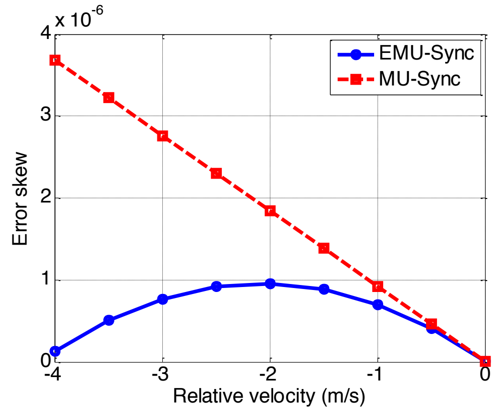

Fig.4 The effect of relative velocity in the skew estimation when $v_c = 2$ m/s

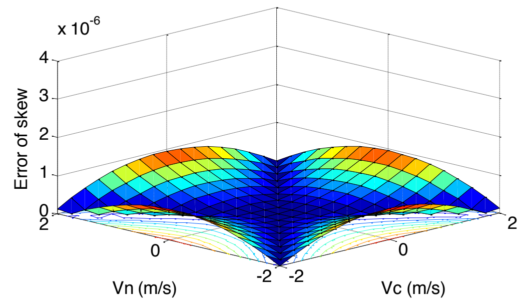

Fig.5 The effect of relative velocity and cluster node velocity in the skew estimation of EMU-Sync

In Fig.5 we can observe the error of skew of MU-Sync and EMU-Sync respectively with possible movement when each sensor node can move within the range [-2 to 2] m/s. In EMU-Sync when the velocity reaches the maximum of 2 m/s then error of skew approaches to zero because in equation (6) we take into account of skew error for both cluster and neighboring node

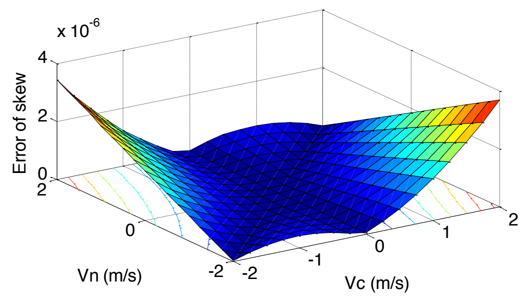

Fig.6 The effect of relative velocity and cluster node in the skew estimation of MU-Sync

Fig.6 shows that MU-Sync yields good estimation when $v_c = 0$ and when the relative velocity approaches zero, $v_n - v_c \to 0$

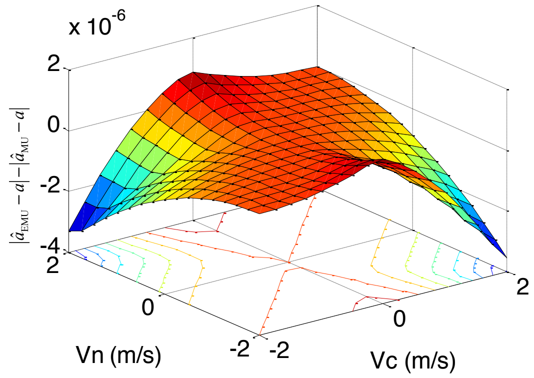

Fig.7 Error differentiation of EMU-Sync and MU-Sync

The differentiation of the skew error in EMU-Sync and MU-Sync is shown in Fig.7. The results are computed from $|\hat{a}_{\mathrm{EMU}} - a| - |\hat{a}_{\mathrm{MU}} - a|$ and we can see that that the skew errors of EMU-Sync are lower than MU-Sync when both nodes move with different velocity.

4. Conclusions

In this paper, we analyze the error of EMU-Sync and MU-Sync, which are the time synchronization protocols for UWSNs. From the results, it can be observed that the errors of MU-Sync are less when the cluster node is static or the relative velocity approaches zero. EMU-Sync can outperform MU-Sync when the cluster node moves and the velocity of the neighboring node is different from the cluster node.

References

[1] A. A. Syed and J. Heidemann, Time Synchronization for High Latency Acoustic Networks, in Proc. INFOCOM 2006, April 2006, pp. 112.

[2] N. Chirdchoo, W.-S. Soh, and K. C. Chua. Mu-sync: A time synchronization protocol for underwater mobile networks. In WuWNet08, September,15 2008.

[3] P.Vonlopvisut, R.Annur, L.Wuttisittikkulkij and N.Chirdcoo. Enhancement of MU-Sync: A Time Synchronization Protocol for Underwater Mobile Networks. In ITC-CSCC 2014, July 2014, pp 671-674

[4] F. Lu, D. Mirza, and C. Schurgers. D-sync: Doppler-based time synchronization for mobile underwater sensor networks. WUWNet10, 2010.

[5] J. Liu, Z. Wang, M. Zuba, Z. Peng, J.H. Cui and S. Zhou. DA-sync: A Doppler Assisted Time Synchronization Scheme for Mobile Underwater Sensor Network. In IEEE Transaction on Mobile Computing, 2014.