International Journal of Technology (2016) 7: 1232-1238
ISSN 2086-9614 

# BROADCAST-BASED SKEW CORRECTION TECHNIQUE FOR WIRELESS SENSOR NETWORKS

Wasan Lasoi[1], Sataporn Pornpromlikit[1*]

[1]*Department of Electrical Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen, 40002, Thailand*

(Received: July 2016 / Revised: October 2016 / Accepted: November 2016)

## ABSTRACT

Time synchronization is a vital process in wireless sensor networks, where energy sources are highly limited. In this work, we propose a broadcast-based skew correction technique that will improve both the accuracy and the lifetime of any time synchronization protocol that only corrects time offsets among sensor nodes. Using time information transmitted periodically by the root node, each client node can compensate its software clock frequency in real time after an initial offset correction. The experimental results show that a clock skew below 0.05 us/s can be consistently achieved with appropriate compensation parameters after the correction process is stabilized.

*Keywords:* Broadcast; Offset correction; Skew correction; Time synchronization; Wireless sensor network (WSN)

## 1. INTRODUCTION

Wireless sensor networks (WSNs) have been widely used in recent years to aggregate data for remote monitoring applications. These networks consist of a large number of sensor nodes, which usually have limited energy sources and cheap unregulated crystal oscillators as local clocks. Maintaining an accurate common time frame among neighbor nodes with minimum energy consumption is nontrivial.

Essentially, the time synchronization process to achieve the highest accuracy and the longest synchronization lifetime involves two major steps: time offset correction and frequency offset (skew) correction. There are many popular time synchronization protocols in literature, such as Reference Broadcast Synchronization (RBS) (Elson et al., 2002), Timing-sync Protocol for Sensor Networks (TPSN) (Ganeriwal et al., 2003), and Flooding Time Synchronization Protocol (FTSP) (Maroti et al., 2004), which differ in how time offset and skew corrections are done network-wide.

For RBS, client nodes receive reference-broadcast messages from the root and exchange time information with one another to synchronize their clocks, assuming they receive each message at the same time. This protocol achieves high accuracy, but only works for one-hop networks. Moreover, the assumption that all client nodes receive the same message simultaneously is not valid in some applications such as those underwater (Liu et al., 2013). TPSN is practical and easy to implement for many applications, but it only corrects the time offsets among sensor nodes. Resynchronization will be required frequently in the network as client clocks drift away

*Corresponding author's email: spornpro@gmail.com, Tel. +66-84-6777545
Permalink/DOI: https://doi.org/10.14716/ijtech.v7i7.4950

from one another due to their relative skews. To solve this problem, statistical estimation techniques are applied to TPSN in (Noh et al., 2007; Chaudhari et al., 2012; Yang et al., 2012) to estimate both the time offset and the skew between a pair of nodes. These algorithms, however, have limited accuracy, especially under volatile environments where clock frequencies can vary over time.

In this work, we adopt TPSN as a protocol of choice for network-wide synchronization and introduce a real-time broadcast-based skew correction technique that has much less computational complexity and is more robust than those previously presented in the literature.

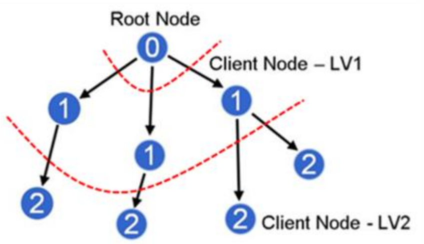



Figure 1 A TPSN hierarchical structure

## 2. PROPOSED TIME SYNCHRONIZATION PROTOCOL

Similar to TPSN, a hierarchical structure is first established in the network (Ganeriwal et al., 2003) as illustrated in Figure 1. Pair-wise synchronization is then performed in two steps from the top down throughout the network.

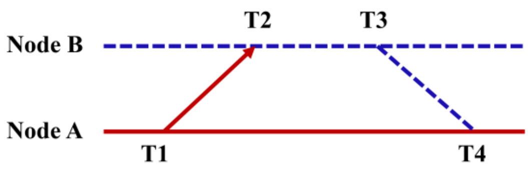



Figure 2 A two-way message exchange between a pair of nodes

### 2.1. Pair-wise Synchronization: Offset Correction

As shown in Figure 2, time information from multiple rounds of two-way message exchanges between a pair of nodes is used to estimate the clock offset. Assuming no clock skew, the propagation delay ($d$), and relative phase offset ($\theta$), can be estimated as shown in Equations 1 and 2:

$$d = \frac{(T2 - T1) + (T4 - T3)}{2} \tag{1}$$

$$\theta = \frac{(T2 - T1) - (T4 - T3)}{2} \tag{2}$$

The calculations are based on the assumption that propagation delays are symmetric for both messages. In reality, the propagation delay has some non-deterministic components that introduce errors (Elson et al., 2002).

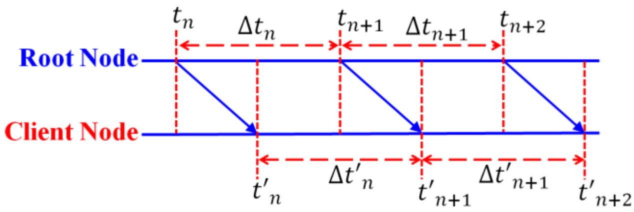



Figure 3 Broadcast-based skew correction

## 2.2. Pair-wise Synchronization: Broadcast-based Skew Corretion

After the initial offset correction, where the time offset is minimized, the root starts broadcasting its reference messages periodically as illustrated in Figure 3. After each received message, the client node can calculates and corrects its skew and offset (due to the skew). If the broadcast period is fixed at $\tau = \Delta t_n = \Delta t_{n+1} = \Delta t_{n+2} = \ldots$, the clock skew ($D_{n+1}$) of the client node at time $t_{n+1}$ can be estimated as shown in Equation 3:

$$D_{n+1} = \frac{\Delta t_n^{'} - \Delta t_n}{\Delta t_n} = \frac{\left(t_{n+1}^{'} - t_n^{'}\right)}{\tau} - 1 \qquad (3)$$

The extra offset ($\theta_{n+1}$) at $t_{n+1}$ due to the skew during one broadcast period is calculated as shown in Equation 4:

$$\theta_{n+1} = \tau D_{n+1} \qquad (4)$$

To correct the skew, the client node periodically compensates its own time (by a number of ticks) according to the following equation where $\beta$ = compensation factor, which is introduced to improve the steady-state error and the stability of this feedback-like mechanism as shown in Equation 5:

$$t_{n+1}^{''} = t_{n+1}^{'} - \beta \tau D_{n+1} \qquad (5)$$

## 3. RESULTS AND DISCUSSION

As a proof of concept, a star-network prototype has been implemented, employing CC2530 System-on-Chip (SoC) transceivers from Texas Instruments and SimpliciTI communication protocol, which stacks on top of the IEEE 802.15.4 MAC layer. The SimpliciTI protocol is chosen because of its open source code and the fact that time stamps can be inserted directly in the medium access control (MAC) layer avoiding some non-deterministic delay components introduced in the application layer.

As shown in Figure 4, the measured clock skew is slowly reduced and approaches a very small value as it gets to the steady state, where the client node can switched off its transceiver and sleep to save its energy source. Without any further correction, the client time would drift away from the root time due to the uncorrected skew.

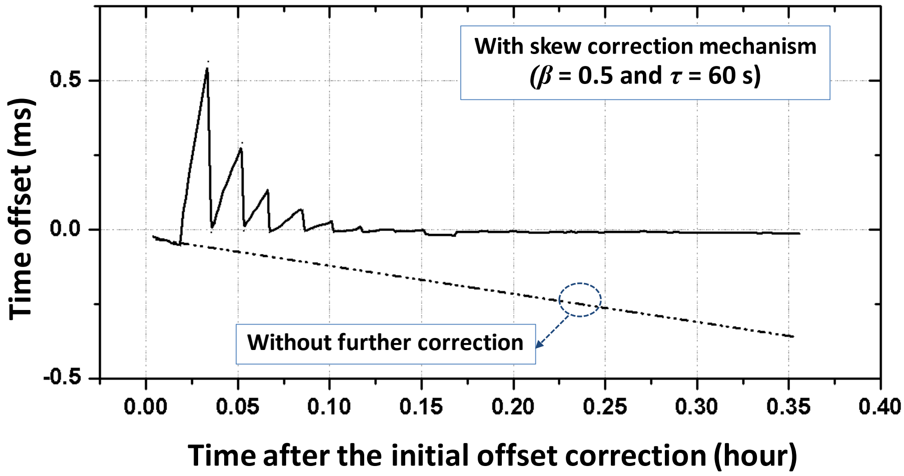
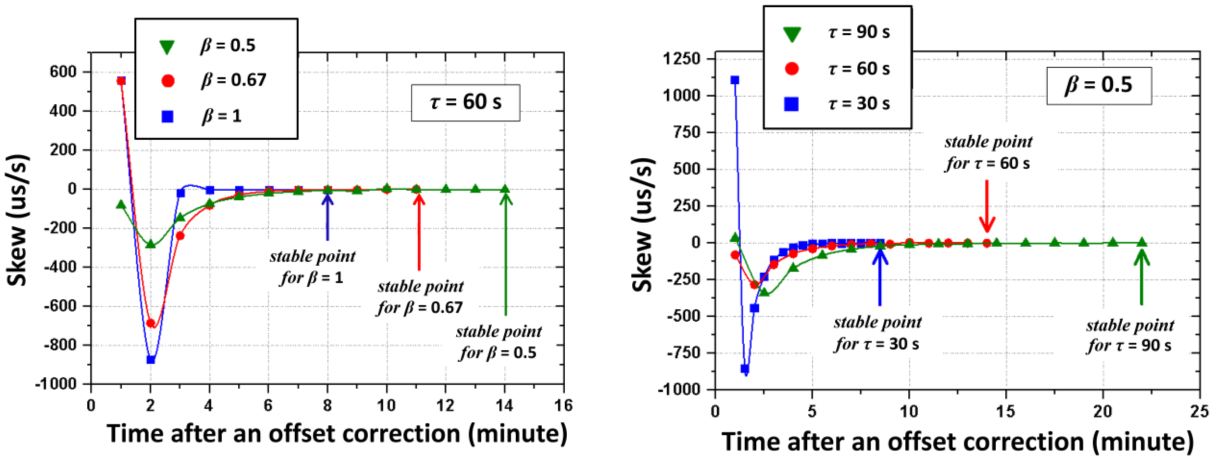



Figure 4 Measured time offset of a client node after the initial offset correction, with and without skew correction mechanism

The values of $\tau$ and $\beta$ are varied to study their relevance to the accuracy and reliability of the proposed technique. From Figure 5, the skew correction process will reach its steady state faster with a larger $\beta$ and a smaller $\tau$. However, a smaller $\beta$ will, on average, lead to a better stability and skew at the steady state, as shown in Figure 6. With $\beta$ = 0.5, the measured skew is consistently below 0.05 us/s after the skew correction phase is stabilized.




(a) (b)

Figure 5 Measured skew of a client node vs. time during the skew correction process: (a) with different values of $\beta$; and (b) with different values of $\tau$

The broadcast period, $\tau$, should be long enough to reduce the effect of non-deterministic variations in propagation delays. From the experiments, a larger value of $\tau$ will result in a better corrected skew at the steady state, but its effect diminishes for the value of $\tau$ beyond 30 seconds, and especially for a compensation ($\beta$ < 1) as shown in Figure 6.

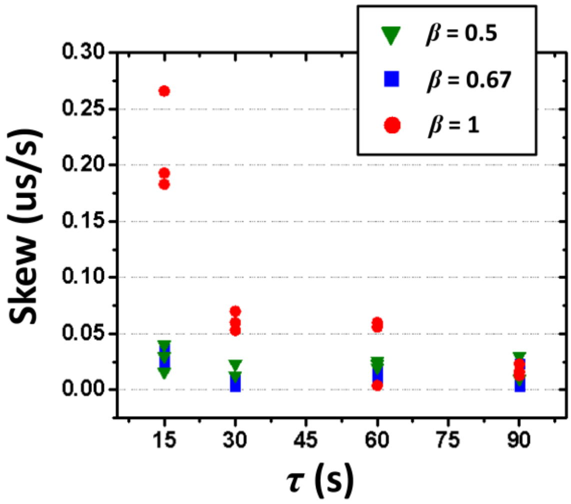



Figure 6 Measured skew of a client node after the skew correction with different values of $\tau$ and $\beta$. The experiment was repeated three times for each combination of $\tau$ and $\beta$

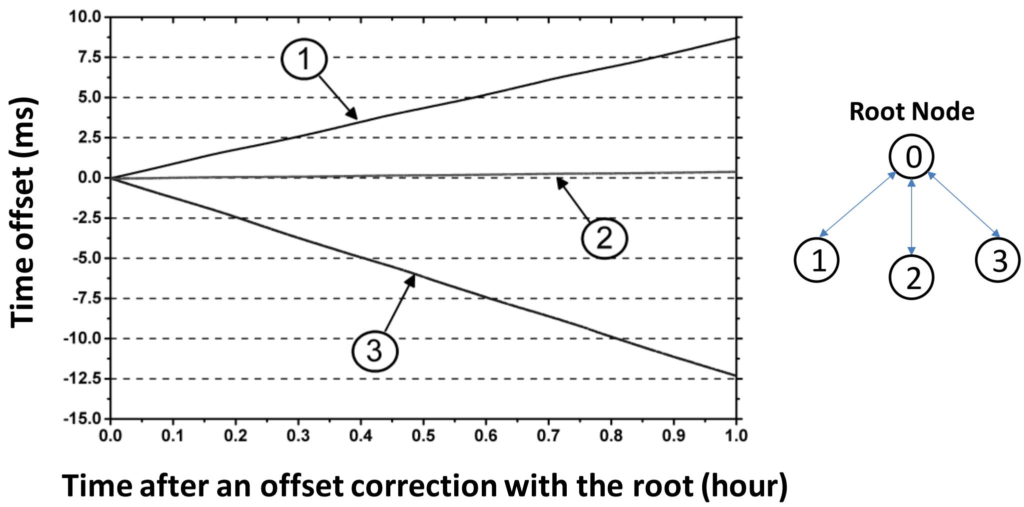
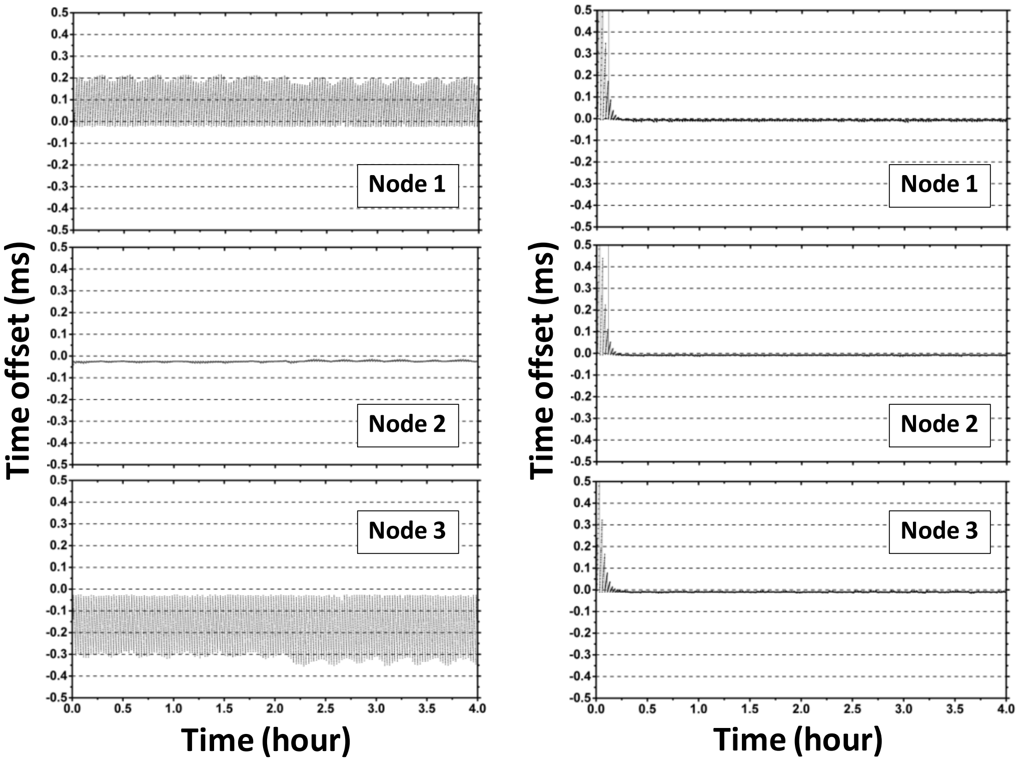



Figure 7 A star-network experiment setup with three client nodes

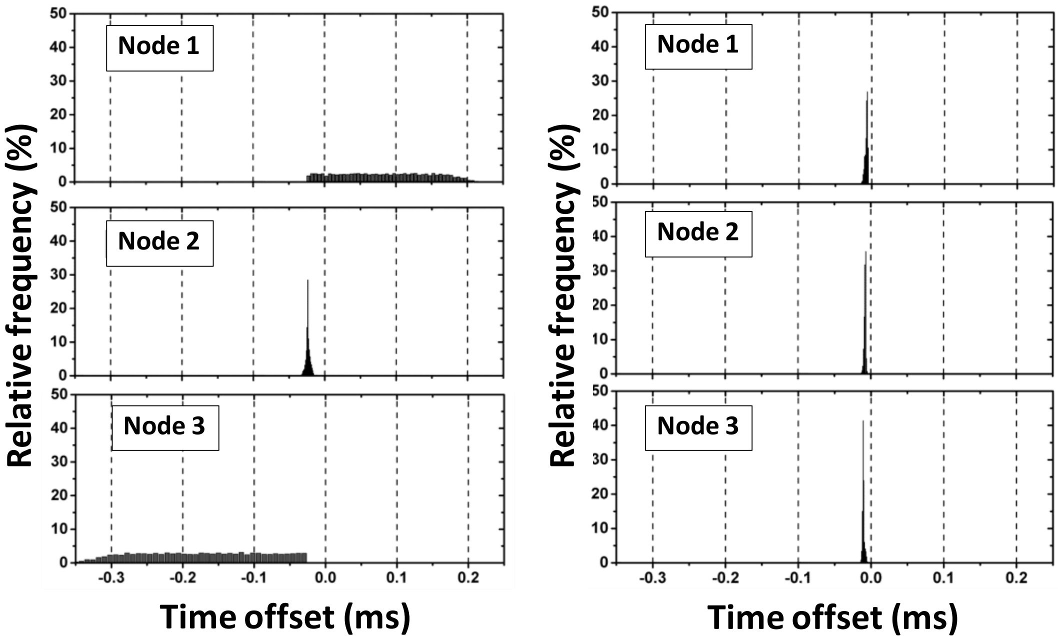
Based on the study, further experiments have been done on a star network with one root and three client nodes, as illustrated in Figure 7. All three nodes have different clock skews relative to the root node. Notice that Node 2 already has a relatively small skew.

By setting $\beta$ = 0.5 and $\tau$ = 90 seconds, the proposed time synchronization mechanism achieves consistent accuracy across the network, as shown in Figure 8b, despite a large variation in oscillator characteristics among nodes. The respective histograms of the measured results (the last-three-hour portion) are also presented in Figure 9b to illustrate the effectiveness of the proposed technique. Without the skew correction (Figure 8a), each node will have to correct its offset with the root frequently with much larger energy consumption and limited accuracy, as shown by the widely-spread histograms in Figure 9a.




(a) (b)

Figure 8 Measured time offsets of the three client nodes: (a) with an offset correction with the root node every 90 seconds (no skew correction); and (b) with the proposed skew correction mechanism ($\beta = 0.5$ and $\tau = 90$ s)




(a) (b)

Figure 9 Histograms of the measured time offsets: (a) with an offset correction with the root node every 90 seconds (no skew correction); and (b) with the proposed skew correction mechanism ($\beta = 0.5$ and $\tau = 90$ s)

## 4. CONCLUSION

A TPSN-based protocol with a broadcast-based skew correction mechanism is proposed in this work. During the skew correction phase, the root sends out periodic reference messages with time information that the client can used to compensate its clock skew in real time. The experimental results show that a skew below 0.05 us/s can be consistently achieved after the correction process is stabilized with appropriate compensation parameters.

## 5. ACKNOWLEDGEMENT

This work is financially supported by the young research development project of Khon Kaen University.

## 6. REFERENCES

Chaudhari, Q.M., 2012. A Simple and Robust Clock Synchronization Scheme. *IEEE Transactions on Communications*, Volume 60(2), pp. 328–332

Elson, J., Girod, L., Estrin, D., 2002. Fine-grained Network Time Synchronization using Reference Broadcasts. *In*: Proceedings of the 5th Symposium on Operating Systems Design and Implementation, Volume 36(SI), pp. 147–163

Ganeriwal, S., Kumar, R., Srivastava, M.B., 2003. Timing-sync Protocol for Sensor Networks. *In*: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems 2003, pp. 138–149

Liu, J., Zhou, Z., Peng, Z., Cui, J-H., Zuba, M., Fiondella, L., 2013. Mobi-Sync: Efficient Time Synchronization for Mobile Underwater Sensor Networks. *IEEE Transactions on Parallel and Distributed Systems*, Volume 24(2), pp. 406–416

Maroti, M., Kusy, B., Simon, G., Ledeczi, A., 2004. The Flooding Time Synchronization Protocol. *In*: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems 2004, pp. 39–49

Noh, K-L., Chaudhari, Q.M., Serpedin, E., Suter, B.W., 2007. Novel Clock Phase Offset and Skew Estimation using Two-way Timing Message Exchanges for Wireless Sensor Networks. *IEEE Transactions on Communications*, Volume 55(4), pp. 766–777

Yang, Z., Cai, L., Liu, Y., Pan, J., 2012. Environment-aware Clock Skew Estimation and Synchronization for Wireless Sensor Networks. *In*: Proceedings of the IEEE INFOCOM 2012, pp. 1017–1025