

Power Consumption Analysis of Prominent Time Synchronization Protocols for Wireless Sensor Networks

Shi-Kyu Bae*

Abstract—Various Time Synchronization protocols for a Wireless Sensor Network (WSN) have been developed since time synchronization is important in many time-critical WSN applications. Aside from synchronization accuracy, energy constraint should also be considered seriously for time synchronization protocols in WSNs, which typically have limited power environments. This paper performs analysis of prominent WSN time synchronization protocols in terms of power consumption and test by simulation. In the analysis and simulation tests, each protocol shows different performance in terms of power consumption. This result is helpful in choosing or developing an appropriate time synchronization protocol that meets the requirements of synchronization accuracy and power consumption (or network lifetime) for a specific WSN application.

Keywords—Wireless Sensor Networks, Time Synchronization, Energy Efficiency, Power Consumption, Performance, Analysis

1. INTRODUCTION

Extensive research in various fields for WSNs has been performed so far, and there is a Time Synchronization issue among them. Time Synchronization is not a new topic, with a traditional method like Network Time Protocol (NTP) [1] adopted for wired networks like the Internet. Time Synchronization is important for a sensor node in WSN to get information accurately from other nodes with respect to a common clock criterion. Time-critical applications expect accurate or efficient synchronization among sensor nodes in WSN. For example, data fusion, a typical WSN application, uses individual sensor readings by exchanging messages that are time-stamped by each sensor's local clock, which requires a common notion of time among the sensors [2]. There is growing interest in literature on proposals for time synchronization in WSNs, since WSN has requirements different from those of wired or other wireless networks [2]. As such, a lot of time synchronization schemes for WSN have been developed until now.

The main challenge in designing WSN is to minimize energy consumption, and the time synchronization protocols for WSN are no exception. Aside from synchronization accuracy, energy constraint should also be considered seriously for time synchronization protocols in WSNs, which typically have limited power environments. Thus, a synchronization protocol needs to be energy-efficient as well. Likewise, to prolong the lifetime of networks, the energy load should be evenly distributed among all nodes so that a certain node will not die too quickly.

Manuscript received on June 28, 2013; first revision August 19, 2013 ; accepted on October 11, 2013.

Corresponding Author : Shi-Kyu Bae (skbae@dyu.ac.kr)

* Dept. of Computer Engineering, Dongyang University, Gyoung-Sang-Buk-Do, 750-711, Korea (skbae@dyu.ac.kr)

Even though a lot of time synchronization schemes for WSN have been developed until now, there are few papers evaluating and comparing the power consumption for most of the existing time synchronization protocols for WSN. Therefore, in this paper, I analyze the prominent time synchronization protocols developed for WSN in terms of power consumption and test them through simulation.

The rest of this paper is organized as follows: Section 2 presents works related to time synchronization protocols and performance evaluation in WSN; In section 3, an energy model for WSN is described; Prominent time synchronization protocols for WSN are analyzed in Section 4; Section 5 shows the simulation test results for those protocols' performance; Finally, Section 6 presents the conclusion.

2. RELATED WORK

Many time synchronization schemes have been proposed for WSN, whose approach is classified into 3 categories in terms of message direction: *Sender-receiver*, *Receiver-receiver* [2], and *Receiver-only* [3] approach. The *Receiver-receiver* approach has a reference node that initiates synchronization; after that, only receiver nodes except the reference node participate in synchronization regardless of the reference node. Reference Broadcast Synchronization (RBS) [4] in this category lets a sender send a broadcast beacon for receivers' reference, with all the receivers except the sender exchanging messages with each other to share the timing information. RBS increases the accuracy by eliminating the sender side's delay uncertainty, even though it cannot transmit the exact global reference time. The *Sender-receiver* method indicates that one node sends a message with useful information and the other(s) receive(s) it. Timing-sync Protocol for Sensor Networks (TPSN) [5] and Flooding Time Synchronization Protocol (FTSP) [6] are classified under this synchronization category, and they can transmit a global reference clock through the network unlike RBS. NTP and TPSN basically measure round trip delay and estimate the clock offset between two nodes [5]. In the *Receiver-only* approach, which was proposed later than the two previous methods, a group of nodes is synchronized by listening to the message exchange of a pair of nodes. Pair-wise Broadcast Synchronization (PBS) [3] is the representative of this category.

As the well-known time synchronization protocols for WSN, RBS, TPSN, and FTSP have influenced many other schemes. Thus, many new derivative schemes based on these three approaches have been developed until now. There are hybrid types [7-9] that take advantage of RBS and TPSN, or enhanced versions of RBS [10] and TPSN [11]. In [7], a root node calculates the clock offset with a reference node pair-wise like TPSN and broadcasts it to all child nodes level by level at the previously constructed tree, unlike TPSN. In [8], TPSN and RBS methods were combined depending on the number of children connected to the transmitter node to reduce data transmission. Authors of [9] proposed a synchronization protocol wherein the cluster heads are synchronized with the sink pair-wise and the cluster nodes are subsequently synchronized with their respective cluster heads by broadcast. [10]'s scheme performs synchronization with ranging, i.e., selects a particular range of broadcast according to the transmission power of the sensor nodes. A TPSN-like synchronization scheme with security functionality for heterogeneous sensor networks -- utilizing powerful high-end sensors -- was proposed in [11]. Although most of these schemes in [4-11] are related to power efficiency, evaluation and

comparison with other protocols in terms of power consumption were not fully performed.

The energy model in WSN was proposed, and energy consumption at a node was analyzed [12,13]. Authors of [12] presented the energy models of core node components -- including processors, Radio Frequency (RF) modules, and sensors -- and evaluated the energy consumption of network protocols based on their node energy model. Authors of [13] developed cluster-based protocol architecture for data aggregation in WSN and analyzed the overall energy consumption for their architecture. Energy dissipation for clustering methods in WSNs was analyzed in [14], where the energy consumption for traffic due to target activities or events was considered in a sensor network detecting a target in the sensor field.

As far as we know, there are few papers evaluating and comparing the power consumption for time synchronization protocols for WSN. Therefore, our analysis of power consumption may be used for developing new protocol or enhancing current protocols in terms of power efficiency.

3. POWER CONSUMPTION ANALYSIS OF WSN

Since power consumption for communication is known to be far higher than that for computing in the case of most WSN applications, power consumption of a sensor node is dominated by communication rather than computing [12,13].

In the experiment with a sample application (i.e., Dynamic Source Routing protocol) [12], a transceiver module at each node had higher energy consumption than the processor or sensor module. At a transceiver module of each node, energy consumption in the transmit state is higher than that in the receiver state. In other words, for the same length of packet, more energy is required for sending a packet than receiving one. In the same simulation mentioned above [12], however, the transceiver module showed higher energy consumption in the receive state than in the transmit state. This is because all packets within the communication area were received by a node, which is called *Overhearing*.

Overhearing occurs when a node receives packets that are not destined to it. Due to the broadcast characteristics of wireless communication, all nodes overhear the packets transmitted from a transmitter node in reach. Thus, in most cases, the number of packets received is much greater than that of packets transmitted at each node. Overhearing, which is determined by some parameters including node density, consumes a significant amount of energy.

3.1 Power Evaluation Model

Suppose all sensor nodes are homogeneous, and there are two basic energy consumption types, i.e., *transmitting* (E_{Tx}) and *receiving* (E_{Rx}). Static WSN -- where all sensor nodes stay at a fixed position once they have been deployed -- is also assumed.

We use a simple energy dissipation model described in [13], which assumes that the radio hardware should dissipate energy to run the radio electronics and the power amplifier for the transmitter. To transmit a b -bit message to a d -distant node, a transmitter node consumes energy as shown in the following equation:

$$\begin{aligned}
 E_{Tx}(b,d) &= E_{Tx-elec}(b) + E_{Tx-amp}(b,d) \\
 &= b \cdot E_{elec} + b \cdot e_{fs} \cdot d^2, \quad d < d_0
 \end{aligned}
 \tag{1}$$

where $E_{Tx-elec}$ is the energy dissipation of the radio electronics, and E_{Tx-amp} , the energy dissipation of the power amplifier at the transmitter. E_{elec} , the electronic energy, is determined by factors such as digital coding, filtering, modulation, and spread of the signal. $e_{fs} * d^2$, the amplifier energy, is determined by the distance from the transmitter to the receiver and the bit-error rate in free space (when the distance is less than threshold distance d_0).

An energy dissipation model described in [13] also assumes that the radio hardware dissipates energy to run the radio electronics for the receiver. To receive a b -bit message from a node with distance that is less than the threshold distance (d_0), a receiver node consumes energy as follows:

$$E_{Rx}(b) = E_{Rx-elec}(b) = b * E_{elec} \tag{2}$$

where $E_{Rx-elec}$ is the energy dissipation of the radio electronics at the receiver, which is the same as $E_{Tx-elec}$ (i.e. $b * E_{elec}$), as shown in Eq.2.

Energy consumed at node i is determined by the transmission energy by itself, with the receiving energy caused by packets transmitted from neighbor nodes within transmission range R . Therefore, energy consumption at each node can be expressed as:

$$E_i = E_{Tx}(b,R) + \sum_{j=0, j \neq i}^n E_{Rx}(b) * I(j) \tag{3}$$

where n is the number of nodes existing within the communication range and $I(j)$ is an indicator function with 1 or 0 whether node j participates in transmitting or not.

The total energy consumption of a network for WSN applications or protocols will be the sum of all nodes' energy consumption in the network as shown in Fig. 1.

$$E_{network} = \sum_{i=0}^N E_i \tag{4}$$

where N is the number of nodes composing the network.

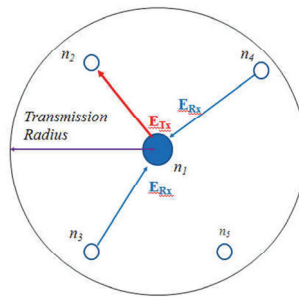


Fig. 1. Power consumption at a node

When sending a packet, a node not only consumes power as much as the transmitting power but also makes other neighbor nodes consume power as much as the receiving power. In other words, due to the broadcast nature of wireless communication, all the neighbor nodes surrounding a transmitter node overhear the transmitted packets and consume the receiving energy.

When a b -bit message is sent by the transmitter node, energy consumed at all nodes within a transmission range (R) in general WSNs is expressed as:

$$\begin{aligned}
 E_{area}(b,R) &= E_{Tx}(b,R) + (n(t)-1) E_{Rx}(b) \\
 &= b * E_{elec} + b * e_{fs} * R^2 + (n(t)-1) * b * E_{elec} \\
 &= n(t) * b * E_{elec} + b * e_{fs} * R^2
 \end{aligned}
 \tag{5}$$

where E_{area} is the total energy consumed caused by the transmitter node at the transmitter itself and all neighbor nodes within a transmission range, and $n(t)$, the number of nodes within a transmission range including the transmitter that is centered (see Fig. 2). $n(t)$ is the function of time depending on the node density (number of nodes per area) and mobility. Since static WSNs without mobility are assumed, $n(t)$ depends only on node density, becoming constant. Subsequently, E_{area} can be rewritten as follows from Eq. 5:

$$E_{area}(b,R) = n * b * E_{elec} + b * e_{fs} * R^2
 \tag{6}$$

where n is the number of nodes including the transmitter node within a transmission range, depending on the node density deployed in a sensor field in static WSNs.

Thus, E_{area} can be represented with node density $\rho = \frac{n}{\pi R^2}$ as follows:

$$\begin{aligned}
 E_{area}(b,R) &= n * b * E_{elec} + b * e_{fs} * R^2 \\
 &= \rho * \pi * R^2 * b * E_{elec} + b * e_{fs} * R^2 \\
 &= (\rho * \pi * E_{elec} + e_{fs}) * b * R^2
 \end{aligned}
 \tag{7}$$

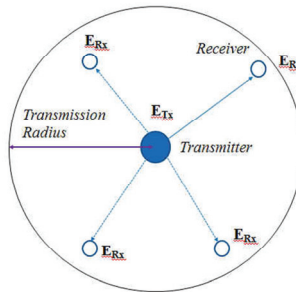


Fig. 2. Power consumption within a communication range by a transmitter

where ρ is the average node density calculated by the number of nodes per area for the network area.

Eq. 7 shows that E_{area} is proportional to the size of messages and R^2 if node density ρ is constant, which means a static (i.e., not mobile), uniformly deployed WSN. If transmission range R is determined and fixed at the transmitter in a uniformly deployed static WSN (i.e., constant node density ρ), E_{area} is only proportional to the size of messages. Therefore, the total energy consumption of a network for WSN applications can be evaluated in terms of the total size of the transmission messages generated in the entire network.

$$\begin{aligned}
 E_{network}(b) &= \sum_{k=0}^M E_{area}(b) \\
 &= (\rho * \pi * E_{elec} + e_{fs}) * R^2 * \sum_{k=0}^M b
 \end{aligned}
 \tag{8}$$

where M represents the total number of messages occurring at all nodes in the network.

4. PROMINENT WSN TIME SYNCHRONIZATION PROTOCOLS

We regard RBS, TPSN, and FTSP as the prominent time synchronization protocols for WSN for the reason mentioned in section 2. In this section, I analyze the number of transmission messages of each scheme, because the total energy consumption of a network for each WSN application can be evaluated in terms of the total size of the transmission messages generated in the entire network in Eq. 8 in the previous section.

4.1 RBS

In RBS, the reference beacon is assumed to arrive at all the receiver nodes at almost the same instant, because the propagation time is negligible in the network where the range is relatively small. RBS was originally designed to work for single-hop network but later extended to a multi-hop network scheme using gateway nodes. All receiver nodes in RBS exchange messages with each other upon receiving a beacon message from a transmitter node. Thus, the maximum number of messages in RBS per round is:

$$M_{\text{RBS}} = 1 + (n-1)(n-2) \quad (9)$$

where n is the number of nodes existing within the communication range. In other words, RBS requires $O(n^2)$ message exchanges for a single-hop network of n nodes.

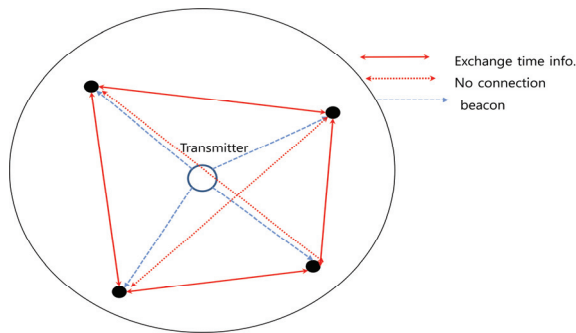


Fig. 3. Operation of RBS

4.2 TPSN

TPSN is broken up into two phases: *level discovery* and *synchronization*. The *level discovery phase* creates the hierarchical topology of the network wherein each node is assigned a level (i.e., group of links with the same depth from the root node). Only one node resides on level zero, the root node. In the *synchronization phase*, all level nodes synchronize with the lower-level nodes with two-way communications. This synchronizes all nodes with the root node shown in Fig. 4.

Suppose N is the number of nodes of an entire network. Since the hierarchical tree of TPSN is constructed by simple flooding-based method, transmissions as many as N times are required.

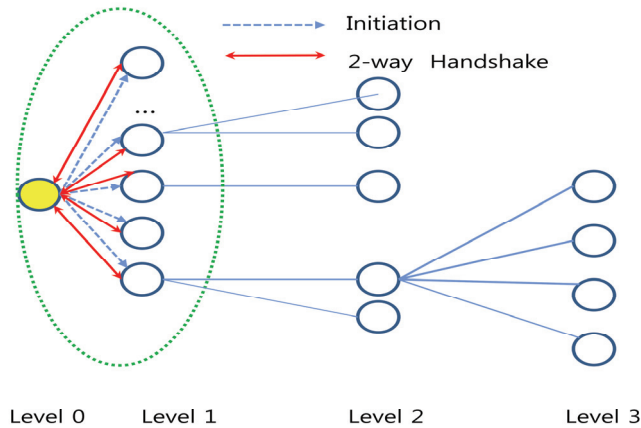


Fig. 4. Operation of TPSN

If h is the number of parent (or non-leaf) nodes at the tree, h ranges from 1 to $(N-1)$. One initiation message per parent node and one round-trip message exchange per child (or link) are required in the synchronization phase of TPSN as shown in Fig. 4. Thus, the total number of message transmissions per round (M_{TPSN}) can be calculated as follows:

$$M_{\text{TPSN}} = h + 2(N-1) = 2N + h - 2 \tag{10}$$

4.3 FTSP

FTSP operates node by node within a communication range from a center node. Since all nodes broadcast a message at least once, unlike simple flooding [6], the number of messages per round is:

$$M_{\text{FTSP}} = N \tag{11}$$

In FTSP, the order of synchronization operations is determined by the position of nodes (or distance from the previous node) as shown in Fig. 5 and is different from TPSN, which is determined by the previously constructed tree.

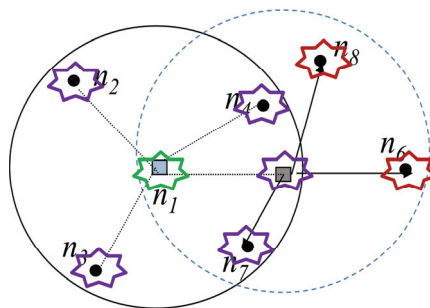


Fig. 5. Operation of FTSP

5. SIMULATION TEST

The power dissipation of RBS, TPSN, and FTSP was tested through simulation with Network Simulator NS2- 2.33 [15].

5.1 Simulation environment

Basically, simulation has been performed at the topology (with 3 rows by c variable columns) as shown in Fig. 6.

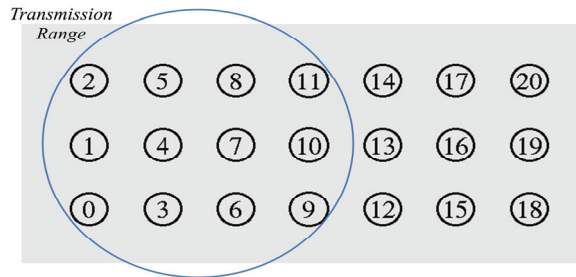


Fig. 6. Basic simulation topology

The distance among adjacent nodes in x or y coordinate is a uniform 100m, and the communication region diameter is 550m. Node density (number of nodes per area) and network size will be changed by narrowing the gap among neighbor nodes or adding nodes based on this basic topology in other simulation scenarios. The parameters for energy model at a node in simulation are as follows: *Transmission Power Consumption* (0.660W), *Receive Power Consumption* (0.395W), and *Idle Power Consumption* (0.035W).

5.2 Relationship between the number of transmission messages and power consumption

First, I measured the power consumption of a network when several messages were transmitted by a specific node to show the relationship between the number of transmission messages and power consumption as in Fig. 7. The total power consumption at the network seems to be directly proportional to the number of transmission messages.

Fig. 8 shows that power consumed at each node in the test is different due to transmission or receiving according to its position. In this case, node #7 is a transmit node, with the other nodes surrounding it. Nodes with node #0 to node #23 (except node #7) are located within reach from node #7, so they consumed power due to overhearing. Note, however, that nodes #24, #25, and #26 are out of reach from node #7, so the network hardly consumed power besides idle state energy.

5.3 Number of transmission messages after the synchronization operation for each scheme

Even though RBS was designed for the one-hop network other than TPSN and FTSP, I have measured the performance of RBS separately at each cluster to compare with TPSN and FTSP

in the same environment. Fig. 9 shows the number of transmission messages measured with respect to variable node density for each scheme. In Fig. 9, the basic topology with 18 nodes mentioned above is represented as $\rho_0(\#18)$. $2 * \rho_0(\#36)$ and $4 * \rho_0(\#72)$ mean double $\rho_0(\#18)$ and quadruple $\rho_0(\#18)$ in node density, respectively.

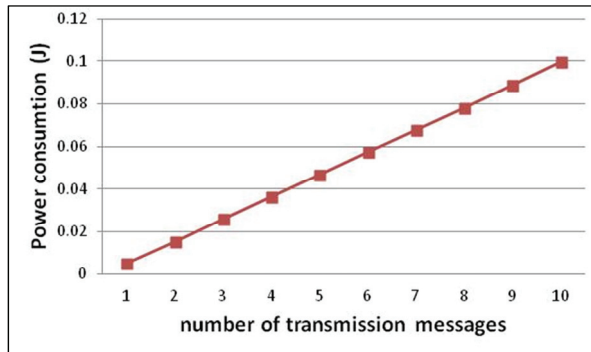


Fig. 7. Power consumption with respect to the number of transmission messages at a node

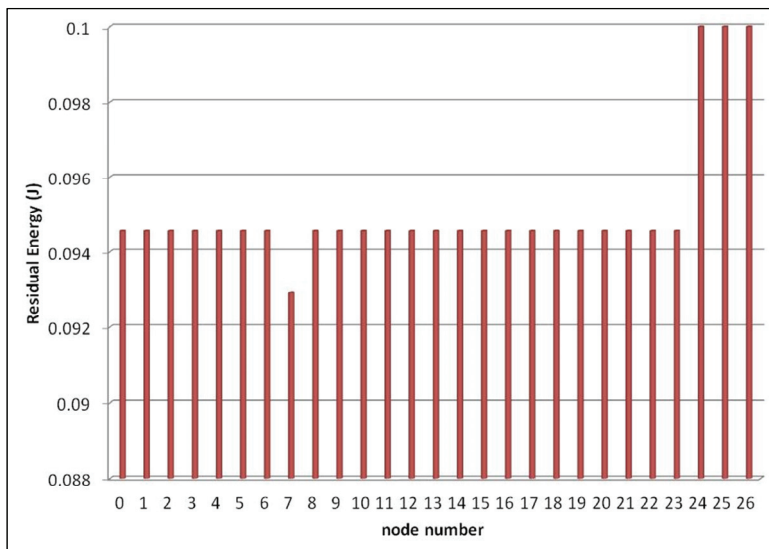


Fig. 8. Residual energy remaining at each node after the transmission of 10 messages at node #7

The number of messages in FTSP is directly proportional to the node density (number of nodes per area). The number of messages in TPSN is more than twice as many as in FTSP. “TPSN+” marked in Figs. 9~11 means the total number of messages (or power consumption) for constructing a tree and one-round synchronization in TPSN, because tree construction should be done at the first phase or whenever the topology changes. TPSN’s number of messages is not exactly proportional to the number of nodes as the number of nodes increases. Such is considered attributable to the fact that different trees were made for different topologies and retransmissions due to the collision occurring among neighbor nodes. Meanwhile, since the number of messages in RBS is proportional to the square of the number of nodes, it is biggest

when comparing TPSN and FTSP.

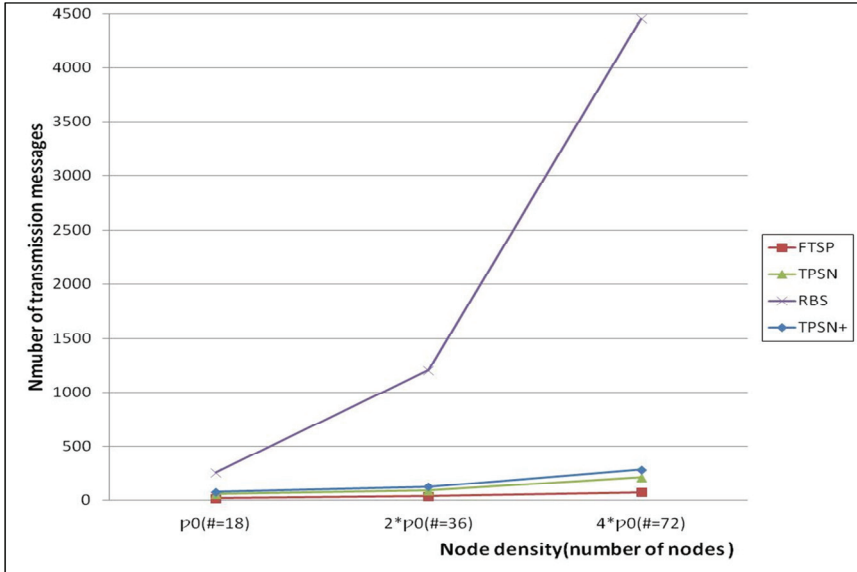


Fig. 9. Comparison of the number of transmission messages with respect to variable node density for each scheme

Fig. 10 shows the number of transmission messages measured for each scheme with respect to the variable network size with increasing node numbers and maintaining the same node density. In Fig. 10, the basic topology with 18 nodes mentioned above is represented as $\rho_0(\#18)$, and $\rho_0(\#36)$ and $\rho_0(\#72)$ indicate double $\rho_0(\#18)$ and quadruple $\rho_0(\#18)$ in network size, respectively.

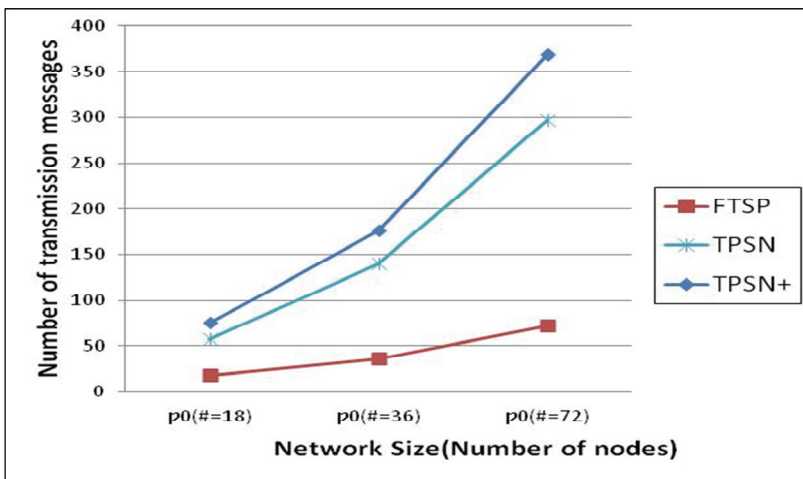


Fig. 10. Comparison of the number of transmission messages with respect to the variable network size for each scheme

5.4 Power consumption after the synchronization operation for each scheme

Fig. 11 and Fig. 12 present the result of power consumption for the compared schemes.

Power consumption is shown to be a function of node density and the amount of transmitted data in Eq. 8. As the transmitted data size grows with increasing node density (i.e., more nodes), power consumption shows a non-linear relationship with node density as in Fig. 11. Power consumption is proportional to the number of messages if node density is fixed. As a result, Fig. 11 shows a form similar to Fig. 9, and Fig. 12, a form similar to Fig. 10.

Therefore, I can conclude that power consumption can be evaluated in terms of the number of transmission messages (actually, the size of transmitted messages).

Fig. 13 was redrawn from Fig. 11 and Fig. 12 to show the difference in power consumption for TPSN and FTSP with respect to node density (changing the network area with the number of nodes kept constant).

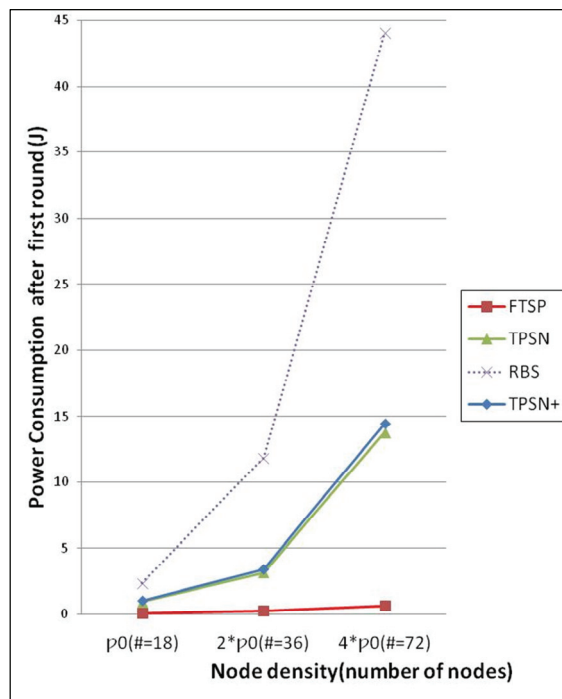


Fig. 11. Comparison of power consumption for each scheme with variable node density

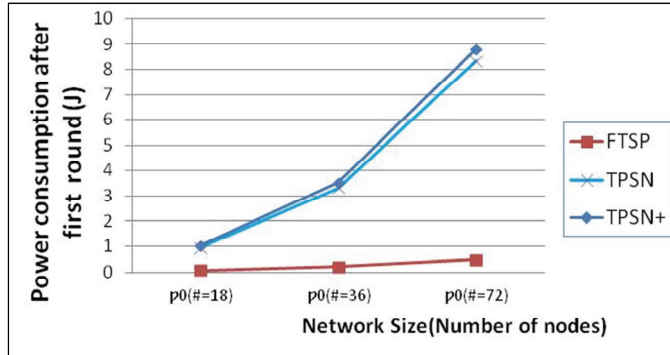


Fig. 12. Comparison of power consumption with respect to the variable network size for each scheme

In TPSN, power dissipated at low density in the 36-node scenario is shown to be slightly greater than at high density. The number of parent nodes in the tree is considered to increase as the size of the network expands (i.e., low density), so that the total number of messages increases. Note, however, that power dissipation at high density in the 72-nodes scenario is much greater than at low density. This is because collision due to too many adjacent nodes makes more retransmissions, so more power is dissipated. By the way, FTSP consumed more energy at high node density overall than at low node density because of overhearing. Retransmission due to collision does not occur in the case of FTSP, since FTSP uses not unicast but broadcast.

Therefore, FTSP has better performance in terms of power consumption than TPSN, especially in a dense network.

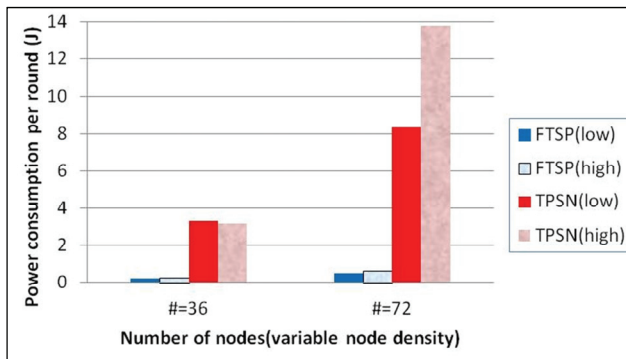


Fig. 13. Comparison of power consumption with respect to node density for FTSP and TPSN

In general, network lifetime is defined as the time interval from the start of operation until the death of the first alive node in network. In this paper, however, I define the network lifetime as the number of synchronization rounds from the start of operation before the first death of at least one node caused by the depletion of residual energy in the entire network because synchronization is performed round by round, not continuously in time.

Fig. 14 shows the lifetimes (number of rounds) tested in a condition with initial energy (1 Joule) and round interval (1 second) for FTSP and TPSN. FTSP has longer lifetime than TPSN.

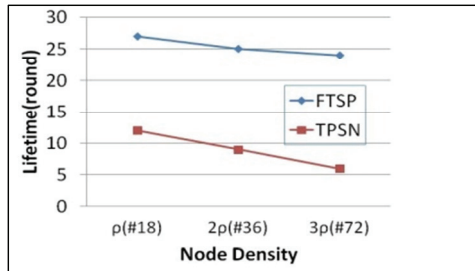


Fig. 14. Comparison of the Lifetimes of FTSP and TPSN

FTSP consumed less energy than TPSN overall. Moreover, FTSP enabled the power consumption of each node to be distributed evenly (see Fig. 15). In contrast, in TPSN, power at each node was consumed unequally, with a certain node (one of the parent nodes in the TPSN tree) consuming the most power than any other node; thus making the network lifetime shorter as shown in Fig. 15.

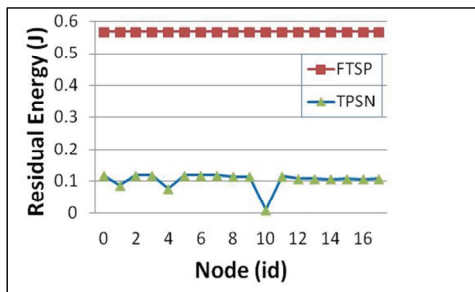


Fig. 15. Distribution of Residual Energy over the network for FTSP and TPSN

5. CONCLUSION

Many time synchronization protocols for WSN have been developed for various WSN applications because time synchronization is essential. Aside from synchronization accuracy, energy constraint should be considered seriously for time synchronization protocols in WSNs, which typically have limited power environments. In this paper, an analysis of prominent WSN time synchronization protocols in terms of power consumption has been performed, with testing by simulation. In the analysis and simulation tests, FTSP showed better performance in power consumption and network lifetime than TPSN and RBS, unless I consider other factors such as synchronization accuracy, implementation environment, reliability, and resynchronization frequency (or count of synchronization rounds). This result is helpful in choosing or developing an appropriate time synchronization protocol that meets the requirements of synchronization accuracy and power consumption (or network lifetime) for a specific WSN application.

REFERENCES

- [1] D. L. Mills, *Computer Network Time Synchronization: The Network Time Protocol on Earth and in Space*, 2nd ed. Boca Raton, FL: CRC Press, 2011.
- [2] B. Sundararaman, U. Buy, and A. Kshemkalyani, "Clock synchronization for wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 3, no. 3, pp. 281-323, 2005.
- [3] K. L. Noh, E. Serpedin, and K. Qaraqe, "A new approach for time synchronization in wireless sensor networks: pairwise broadcast synchronization," *IEEE Transactions on Wireless Communications*, vol. 7, no. 9, pp. 3318-3322, 2008.
- [4] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation*, Boston, MA, 2002.
- [5] S. Ganeriwala, R. Kumar, and M. Srivastava, "Timing-synch protocol for sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, Los Angeles, CA, 2003, pp. 138-149.
- [6] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, Baltimore, MA, 2004, pp. 39-49.
- [7] S. Rahamatkar and A. Agarwal, "An approach toward lightweight, reference-based, tree-structured time synchronization in WSN," in *Advances in Computer Science and Information Technology (Communications in Computer and Information Science Vol. 131)*, N. Meghanathan, B. Kaushik, and D. Nagamalai, Eds. Heidelberg, Germany: Springer Berlin, 2011, pp. 189-198.
- [8] R. Akl and Y. Saravanos, "Hybrid energy-aware synchronization algorithm in wireless sensor networks," in *Proceedings of the 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Athens, Greece, 2007.
- [9] S. Chauhan and L. K. Awasthi, "Adaptive time synchronization for homogeneous WSNs," *International Journal of Radio Frequency Identification and Wireless Sensor Networks*, vol. 1, no. 1, pp. 34-43, 2011.
- [10] S. Jain and Y. Sharma, "Optimal performance reference broadcast synchronization (OPRBS) for time synchronization in wireless sensor networks," in *Proceedings of the International Conference on Computer, Communication and Electrical Technology*, Tamil Nadu, India, 2011, pp. 171-175.
- [11] X. Du, M. Guizani, Y. Xiao, and H. H. Chen, "Secure and efficient time synchronization in heterogeneous sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 4, pp. 2387-2394, 2008.
- [12] H. Y. Zhou, D. Y. Luo, Y. Gao, and D. C. Zuo, "Modeling of node energy consumption for wireless sensor networks," *Wireless Sensor Networks*, vol. 3, no. 1, pp. 18-23, 2011.
- [13] W. B. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660-670, 2002.
- [14] M. Ilyas and I. Mahgoub, *Sensor Network Protocols*. Boca Raton, FL: CRC/Taylor & Francis, 2006, pp. 8.25-8.26.
- [15] *The Network Simulator ns-2: Documentation* (2011) [Online]. Available: <http://www.isi.edu/nsnam/ns-documentation.html>

**Shi-Kyu Bae**

He received his BS in Electronics Engineering from Kyungpook National University and worked for Samsung Electronics Co. He received his MS and PhD degrees in Computer Engineering from Kyungpook National University in Korea. He has been a professor at Dongyang University in Korea. He was a visiting professor at Iowa State Univ. in the USA. His research interests include Computer Networks, Multimedia Communications, and Wireless Sensor Networks.