Sender-receiver Reference Broadcast Synchronization

Shi Kyu Bae

Department of Computer Engineering, Dong Yang University, South Korea
skbae@dyu.ac.kr

Abstract: Time Synchronization Schemes for Wireless Sensor Networks have been classified into two categories in terms of message flow; namely, Sender-receiver and Receiver-receiver. While Sender-receiver type has been used traditionally, RBS (Reference Broadcast Synchronization), the representative of Receiver-receiver approach, was developed to provide higher synchronization accuracy by removing the sender's non-deterministic delay for the critical path, which approach has been referenced by so many other WSN synchronization schemes till now. However, RBS has limitations that it consumes too much energy to exchange large amount of messages and does not support an absolute global clock efficiently over an entire network. In this paper, we propose a new synchronization scheme to provide an absolute global clock for a network, which has RBS-like approach and less message exchange so as to reduce power consumption.
ShiKyu Bae. **Sender-receiver Reference Broadcast Synchronization.** *Life Sci J* 2014;11(7):548-552]. (ISSN:1097-8135). http://www.lifesciencesite.com. 74

Keywords: Time Synchronization; Wireless Sensor Network; Delay; Power Consumption

1. Introduction

WSN(Wireless Sensor Networks), which utilizes wireless implementation of sensor nodes, can be applicable to so many areas such as environmental monitoring, military surveillance, and so on. Time Synchronization is one of the important issues in WSN, as well as other computer network area, where clock offset and clock drift(or skew) cause all nodes to be asynchrony problem with each other. Beside clock offset and skew, random transit delay should be considered because nodes participating in synchronization require transmission of messages among them. Time synchronization in WSN plays a crucial role for many WSN applications such as data fusion, assembly of distributed observations, duty cycling, transmission scheduling, localization, security, tracking etc. Most representative and successful time synchronization is NTP(Network Time Protocol)[1], which, however, is not appropriate for WSN since they were designed for other different requirements than WSN. In addition, synchronization scheme for WSN should meet following requirements; 1) limited resources and cost such as computing power and communication capability, 2) scalability for working well with any number of nodes. Energy consumption is most important among requirements.

RBS[2] was developed to provide higher synchronization accuracy by removing the sender's non-deterministic delay over a path from a sender to receiver[s], which approach has been referenced by so many other WSN synchronization schemes till now. However, RBS has some limitations that it consumes too much energy to exchange large amount of messages and cannot support an absolute global clock over an entire network efficiently.

In this paper, I propose a new synchronization scheme to provide an absolute global clock for a network with RBS-like approach. The scheme has good synchronization accuracy and uses less message exchange so as to reduce power consumption. The remainder of this paper is organized as follows. After surveying related works and analyzing delay characteristics of the existing time synchronization schemes in section 2, I propose a new time synchronization scheme in section 3. Simulations for some performance parameters are performed in the next section. This paper ends with some concluding remarks in Section 5.

2. Related Works and Delay analysis

Transit delay affects synchronization accuracy because transmissions of messages among nodes are required for synchronization.

Delay is comprised of some components; Send, Access, Transmission, Propagation, Reception, and Receive time. Access time is the waiting time for accessing the channel after reaching the MAC (Medium Access Control) layer, which is variable and most critical. Propagation time is the actual time taken to transmit a message from the sender to the receiver through the wireless channel, which is very small in WSNs.

RBS [2], which was developed in 2002, lets a sender broadcast beacon[s] for receivers' reference, and receivers except the sender participate in synchronization by exchanging their observation after recording the time that the beacon was received. RBS increases the accuracy by eliminating sender side's delay uncertainty. That is, duration (t_3-t_1) in Figure 1 is not included in clock offset measurements. And (t_4-

t_3) is ignored, if assumed that it is too small in case of WSN, where distances among nodes are too close. So, (t_6-t_4) in Figure 1 is assumed to be almost same for all receivers. Even though good synchronization accuracy, RBS cannot transmit exact global reference time efficiently (i.e. focusing on relative clock synchronization).

TPSN(Timing-sync Protocol for Sensor Networks)[3], which was developed in 2003, operates basically like NTP [1]. That is to say, NTP measures round trip delay which has all components of delay (shown as (t_6-t_1) in Figure 1), and estimates clock offset between two nodes. However, TPSN uses timestamps at MAC layer to improve delay measurement accuracy (shown as (t_6-t_3) in Figure 1), other than NTP.

In FTSP(Flooding Time Synchronization Protocol)[4] developed in 2004, the authors proposed to use broadcast, not unicast, which was used in TPSN, and uses timestamps at MAC layer for the similar reason as TPSN.

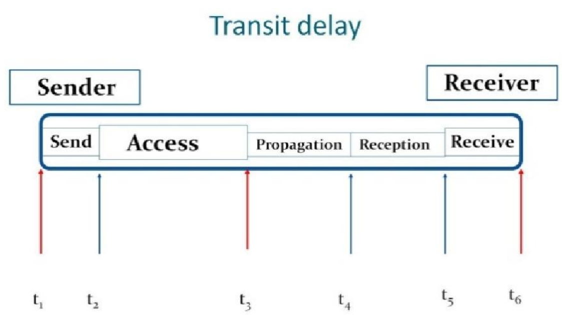

Figure 1 Transit delay components

These Time Synchronization schemes have been classified into two categories in terms of message flow; *SR(Sender-receiver)* and *RR(Receiver-receiver)* [5]. SR type indicates that one node send a message while the other receive it, which includes TPSN and FTSP. In RR approach, receivers mainly participate in synchronization, rather than the sender, which is the case of RBS. In wireless networks synchronization, using broadcast (i.e. RBS and FTSP) is more advantageous than one using unicast like TPSN.

Meanwhile, synchronization schemes have two different approaches according to different goals to synchronize; *Absolute* and *Relative* synchronization, which are similarly classified into *internal* synchronization versus *external* synchronization in [5]. Absolute (or external) synchronization is referenced to global reference time. In Relative (or internal) synchronization, a global reference time base is not available or not necessary. So the protocol attempts and focuses on minimizing clock offsets among nodes. In the light of absolute or relative clock synchronization, SR type schemes synchronize with either absolute or relative clock, while receiver-receiver type schemes focus on only relative clock.

My motivation for this paper is as follows; Accuracy of synchronization is determined by not only algorithm but also communication layer implemented. TPSN and FTSP have good accuracy of synchronization by being implemented at MAC layer. That is, as dominant access delay is eliminated transit delay is determined by remaining delay components. However, it means having some disadvantages like lack of flexibility. In contrast, RBS depends on synchronization points at receivers, not delay between a sender and receivers, and can be implemented at upper layer, which is more flexible than TPSN and FTP. Sender-receiver method is vulnerable to variance in message delay between the sender and the receiver. However, receiver-receiver method cannot propagate the global reference time to other nodes directly, even though it has advantage of the reduction of the message-delay variance.

So, I propose a new synchronization scheme, which is called SRBS(Sender-receiver Reference Broadcast Synchronization), to provide an absolute global clock for a network with RBS-like approach. The scheme has good synchronization accuracy and uses less message exchange so as to reduce power consumption.

3. Proposed Scheme

3.1. Limitaion of RBS

Although RBS can provide higher accuracy by eliminating sender side's delay uncertainty, it has a limitation of being unable to synchronize with a sender node, which cannot be synchronized to global reference clock eventually. Beside, all receivers cannot synchronize each other because some nodes are getting out of the broadcast reach. Take a look at the example in Figure 2.

1) *Inefficient synchronization with a global clock*: If a sender broadcast a beacon, each receiver will record the received time of the beacon from the sender in its local clock, and exchange time information with other receivers. In Figure 1, for example, Rx_1 and Rx_4 can get time offset each other after exchanging the time information, except Sender, which means two nodes are synchronized with relative clock, not global clock. In order to synchronize with Sender node, another beacon broadcast[s] at other nodes and the similar message exchanges are required.

2) *Not all receivers are synchronized*: If a sender broadcast a beacon, each receiver tries to exchange time information with other receivers after recording the received time of the beacon

from the sender in its local clock. In the example case of Figure 1, one of receivers (Rx_1) succeeds to exchange the time information with Rx_4, but cannot reach to Rx_2 and RX_3. In order to synchronize all nodes, more rounds by multiple beacon broadcasts at other nodes would be required.

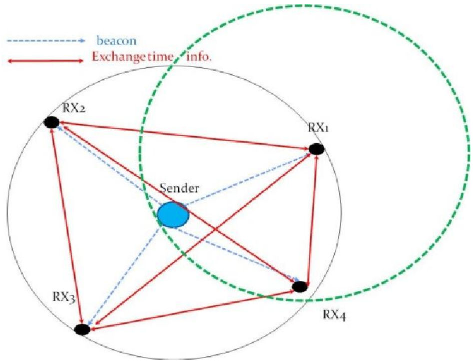

Figure 2 An example operation of RBS

3.2. A Proposed Scheme

I propose a new time synchronization scheme to provide global reference clock synchronization with higher accuracy by eliminating sender side's delay uncertainty like RBS.

In this scheme, there are three types of role for each node; *broadcaster*, *reference* and *normal* (See Figure 3). *Broadcaster* is the same role as beacon sender in RBS. In addition, broadcaster delivers the reference time to several receivers by broadcast within transmission range at the same time. *Reference*, which is selected among the one of the receivers of beacon by broadcaster, provides a reference time to the other nodes. *Normal* node will be synchronized by broadcaster with the time which is supplied from reference.

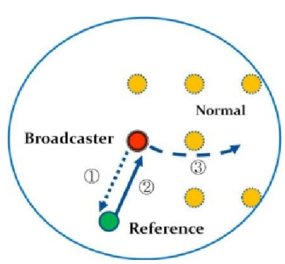

Figure 3 Roles of each node

3.2.1 Root-hop Synchronization

Let's call a cluster which includes the root node a '*root-hop*', and assume that the root node has been synchronized with an external global clock server in advance. In root-hop synchronization, root node plays a role of both reference and broadcaster. The basic concept of the root-hop synchronization is as follows;

1) A Root node (node 'R' in the Figure 4) as a broadcaster broadcasts a '*Attention*' packet to trigger the *Attention* for the receivers and requests to *Assist* node, which is one of the receiver nodes (node 'A' in the Figure 4). Selecting the assist node will be explained later.

2) Assist node broadcasts a beacon (called *ReadyTime*) with more transmission power to be able to reach all the previous receivers including the root node. The broadcast range is determined by the sum of the previous broadcast range's radius plus the distance between the root node and the assist node. That is to say, the broadcast range from the node 'A' should be extended to 'blue circle' as shown in Figure 4. For smaller transmission power at node 'A', node 'A' had better be as close as possible from the root node so that the distance between them is smaller. If the root node already knows the position of other nodes, it can select the closest node as an assist node. The radius of transmission range from node 'A' will double the one from node 'R' maximally if node 'A' is located at the border of red circle.

3) Each receiver (including the root node) records the S*tartTime* in its local time. The receiver nodes that have received *ReadyTime* packet without *Attention* packet (the region outside of red circle in Figure 4) ignore the packet.

4) The root node (as a both *reference and broadcaster simultaneously*) broadcasts a '*Start*' packet with recorded *StartTime*, which is receive time of the *ReadyTime* in the root node's local clock (i.e. reference clock).

5) All receivers set their local clock with respect to clock offset by using the received '*StartTime*' from the root node, which is synchronized to the reference clock. It means that all receivers within the same hop are synchronized to the reference clock.

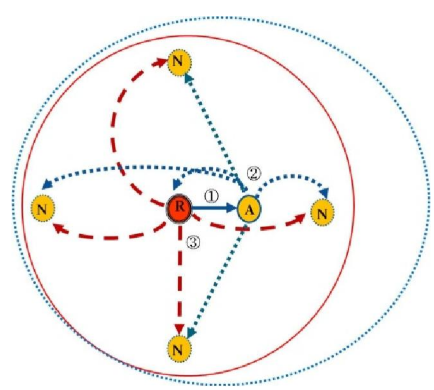

Figure 4 Operation of the proposed scheme at 'Root hop'

3.2.2 Multi-hop Synchronization

In root-hop synchronization, root node plays a role of both reference and broadcaster. For multi-hop network, one of the nodes synchronized to the reference becomes a new reference or broadcaster. Special nodes called 'Leader' are determined by a hierarchical structure, and play roles as either reference or broadcaster.

In second hop which is one-hop distant from the root-hop, the root node and the first 'Leader' node play roles as reference and broadcaster, respectively, as shown in Figure 5. The Leader node (shown as 'L_1' in the figure 5) synchronizes all receivers except already synchronized nodes (where are located in red circle in the figure) within its broadcast area together with reference (the root node shown 'R' in the figure here). This procedure is similar to the one of the root-hop synchronization, and rather simpler.

The next Leader node synchronizes all receivers except already synchronized nodes within its broadcast area together with the previous leader node as reference at the same way. This step above will be repeated to the end of hop throughout a network.

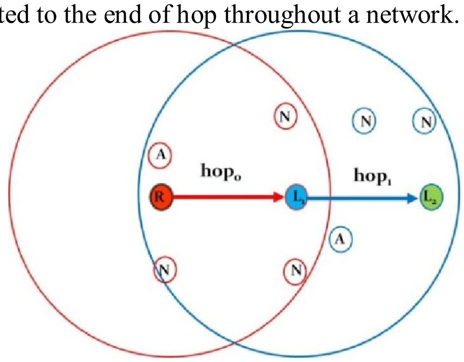

Figure 5 Multi-hop operation of the proposed scheme

4. Simulation Test

The proposed synchronization scheme was tested by simulation with Network Simulator, NS2-2.34 [8]. The simulation has been performed with 18 nodes of 3-hop hierarchy in the topology shown in (Figure 6). Distance among adjacent nodes in x or y coordinate is 100m uniformly, and communication region diameter is 550m. 802.11 MAC and AODV routing protocols are used in this test.

4.1 Delay Measurement

At first, transit delay values of neighbor nodes within broadcast range, which are centered at node 4 as a root node, were measured for representative time synchronization schemes, such as RBS, TPSN, and FTSP. In condition of no clock offset and no clock drift among nodes (i.e perfect synchronization), delay was measured for three schemes which are implemented in application layer to compare synchronization error due to delay.

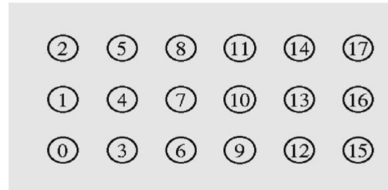

Figure 6 Test topology

As shown in Table 1, delay values in TPSN from root node to each node are larger than those in FTSP. The reason is that TPSN calculates forward delay from round trip delay and need more waiting delay for contending or escaping collision for with other nodes. In RBS or SRBS, delay between the root node and other nodes is meaningless. So, at each node, average value of delays between itself node and other receiver node except the root node (node #4) are measured. (-) sign means that the node runs ahead of other receiver nodes in average when calculating clock offset. RBS or SRBS has the order of micro second, compared to FTPS in a mili-second order and TPSN in a second order.

Synchronization errors for RBS and TPSN were reported to be in about 10us to 30us range [2][3]. And the precision of 1.5μs in the single hop was said to be shown by authors of FTSP [4]. It means that RBS-like approach implemented in application layer can get the equivalent accuracy to TPSN and FTSP, which should be implemented in MAC layer and hardware dependent.

Table 1. Delay measured for each scheme

node	FTSP_A(sec)	TPSN_A(sec)	RBS / SRBS(sec)
0	0.001344471	1.095030965	0.00000008
1	0.001344333	0.082974991	-0.00000130
2	0.001344471	1.074582124	0.00000008
3	0.001344333	1.072306015	-0.00000130
4	0	0	x
5	0.001344333	0.097617201	-0.00000130
6	0.001344471	0.095665465	0.00000008
7	0.001344333	1.081822037	-0.00000130
8	0.001344471	0.107335439	0.00000008
9	0.001344745	0.145587979	0.00000282
10	0.001344667	0.099347809	0.00000204
mean	0.001344463	0.495227003	-3.46945E-19

4.2 Synchronization Result

To test synchronization effect, I set the root node to have initial time offset of 0 usec and other nodes to have initial time offset value range from -5 to

5 usec from random uniform distribution, which represents asynchronous situation.

Figure 7 shows the result of synchronization of proposed scheme. Test topology shown in Figure 6 was also used. And root node is node #1 and node #0 is selected as an assist node. Node #4 and node # 10 are the leader nodes for each hop.

In other nodes than the root node (denoted node #1 in the Figure 7), local times were corrected by referencing to the root node, which means every node in the network was synchronized successfully. Maximum synchronization error in a 3-hop topology showed less than 0.412 usec. RMSE(Root Mean Square Error) of synchronization errors and RMSE of synchronization error per hop after synchronization are 0.261 usec and 0.087 usec, respectively.

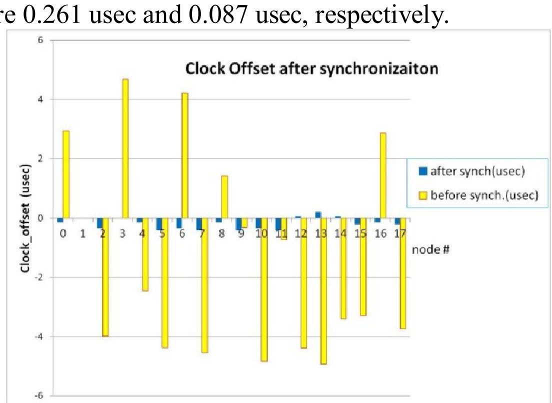

Figure 7 Synchronization result of the proposed scheme

As the number of message exchange is a barometer for energy consumption, reducing the message exchange is important factor for designing synchronization protocols in power efficient WSNs. SRBS needs 3 broadcast messages per hop. So, 9 messages are required for 3-hop network in this test, compared to 18 for FTSP, 57 for TPSN, and 50 for RBS, respectively. Here, RBS was measured only for one-hop because of being designed so originally.

As a result, power consumption for SRBS during one synchronization round was 0.0936(J), compared to 0.0762(J) for FTSP, 1.1777(J) for TPSN, and 0.5996(J) for RBS (even in case of single hop), respectively (See Figure 8). For this test, the parameters for energy model at a node in simulation were 0.660W for Transmission, 0.395W for Receive, and 0.035W for Idle, and 20bytes per message was transmitted. Considering that FTSP requires at least 8 synchronization rounds for acquiring expectation value of clock offset, it is shown that SRBS outperforms TPSN, FTSP, and original RBS during the same time interval in terms of power consumption.

7/1/2013

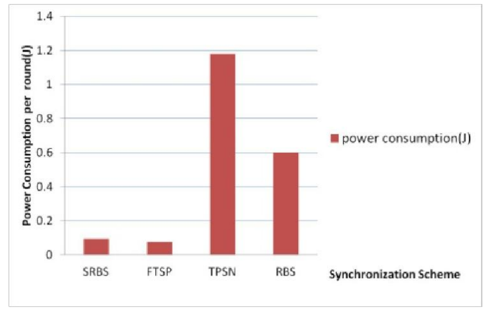

Figure 8 Comparison of power consumption per round for each scheme

Conclusion

Time synchronization plays a crucial role for many WSN applications. Traditional time synchronization schemes do not meet the requirements of sensor networks; energy efficiency, limited resources, dynamics, and scalability etc.

RBS, which has been being referenced by so many other schemes [6],[7], was developed to provide higher synchronization accuracy by removing the sender's non-deterministic delay for the critical path. However, RBS has limitations that it consumes too much energy to exchange large amount of messages and does not support an absolute global clock efficiently over an entire network. In this paper, I propose a new synchronization scheme called SRBS, which bases upon RBS approach, to provide an absolute global clock with good accuracy for a network, and use less message exchange in order to reduce power consumption.

References

1. David ML. Computer Network Time Synchronization: the Network Time Protocol on Earth and in Space. CRC Press 2011.
2. Jeremy E, Lewis G, Deborah E. Fine-grained network time synchronization using reference broadcasts. ACM OSDI 2002:1-17.
3. Saurabh G, Ram K, Mani BS. Timing-Synch Protocol for Sensor Networks. ACM Sensys 2003.
4. Miklós M, Branislav K, Gyula S. Ákos L. The Flooding Time Synchronization Protocol. ACM SenSys'04 2004.
5. Bharath S, Ugo B, Ajay DK. Clock Synchronization for Wireless Sensor Networks: A Survey. Ad Hoc Networks 2005.
6. Xin Q, Xingfa S, Guojun D, Jianhui Z, Changping L. Clapping and Broadcasting Synchronization in Wireless Sensor Network. Sixth International Conference on Mobile Ad-hoc and Sensor Networks 2010:140-145.
7. ÁLVARO M, ROBERTO C, JOSÉ LSR, VICTORIA'N C, A'NGEL A, MOHAMMAD SO. Synchronization of multihop wireless sensor networks at the application layer: IEEE Wireless Communications 2011: 82-88.
8. The Network Simulator ns-2: Documentation, http://www.isi.edu/nsnam/ns/ns-documentation.html, November, 2011.