Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 9, September 2013, pg.100 – 106

SURVEY ARTICLE

A Survey of Time Synchronization Protocols for Wireless Sensor Networks

Baljinder Kaur[1], Amandeep Kaur[2]

[1]Research Scholar, M.Tech (CSE), Sri Guru Granth Sahib World University (FGS), India

[2]A.P, Dept. of Computer Sciences, Sri Guru Granth Sahib World University (FGS), India

[1] b.saini64@yahoo.com; [2] anu_virk10@yahoo.co.in

Abstract— The time synchronization is an major problem for Wireless sensor networks. There have been many research works about time synchronization in Wireless sensor network in the literature and so many methods and protocols are proposed. Many applications of sensor networks required local clocks to be synchronized for precision but different sensor network required different properties. This paper reviews the time synchronization and their need to be synchronization, and then explain the basic synchronization methods and protocols in detail.

Keywords: - wireless sensor network; time synchronization

I. INTRODUCTION

A Wireless Sensor Networks (WSN) is a set of hundreds or thousands of micro sensor nodes that have capabilities of sensing, establishing wireless communication between each other and doing computational and processing operations [1]. Each node is connected to one or several sensor. These sensor nodes are very small in size and are used for broadcast communication. Sensor networks have a wide variety of applications and systems with vastly varying requirements and characteristics. The sensor networks can be used in a vast variety of fields like military environment, disaster management, habitat monitoring, medical and health care, industrial fields, home networks, detecting chemical, biological, radiological, nuclear, and explosive material etc [1]. Structure and topology of WSN can vary from simple star network to an advanced multi-hop wireless mesh network. Power constraints, limited hardware, decreased reliability, and a typically higher density and number of failure nodes are few of the problems that have to be considered when developing protocols for use in sensor networks [2].

Figure 1 shows a typical simple wireless sensor network. As can be seen, a complete wireless sensor network usually consists of one or more base stations (or gateway), a number of sensor nodes, and the end user. Sensor nodes are used to measure physical quantities such as temperature, position, humidity, pressure etc. The output of those sensor nodes are wirelessly transmitted to the base station (or gateway) for data collection, analysis, and logging. End users may also be able to receive and manage the data from the sensor via a website from long-distance or applications in console terminal. However due to the associated cost, time and complexity involved in implementation of such networks, developers prefer to have first-hand information on feasibility and reflectivity crucial to the implementation of the system prior to the hardware implementation[2].

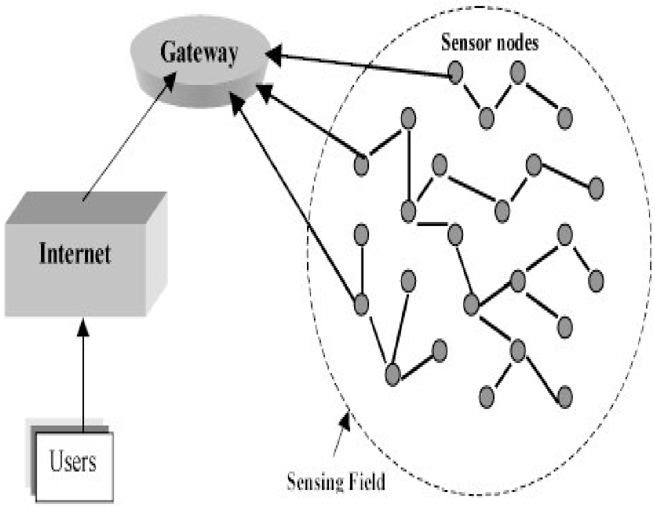

Figure 1: A Simple Wireless Sensor Networks [2]

II. TIME SYNCHRONIZATION IN WSNS

Time synchronization of the nodes has been an open research issue in wireless sensor networks. Time synchronization in distributed systems is a main component of a wireless sensor network (WSN).Which aims to provide a common time scale for local clocks of nods in the network. This common time scale is achieved by synchronizing the clock at each element to a reference time source so that the local time seen by each element of the system is approximately the same.

In the other words Time synchronization algorithms provide a mechanism to synchronize the local clocks of the nodes in the network, to a global clock or relative to each other [3].Time synchronization plays a very important role because it allows the entire system to cooperate and function as a group. It is particularly important for a many of tasks such as synchronizing event detection, data fusion, and coordinating wake and sleep cycles. Due to its importance, the problem of time synchronization has been around for a long time. Many distributed applications running on WSNs need precise synchronization of the clocks among all nodes to have a consistent view of the global time. Each node runs on its own local clock and sends (receives) packets to (from) other nodes at sampling rate of its own clock. Also, to conserve the power, the nodes can enter the sleep mode after sending/receiving the packet and wakeup just before its time for another packet. If the clocks are not synchronized, the nodes can miss the packets sent by other nodes. As an alternative, the slave can have a guard time added to their wakeup time to remain active longer before going into sleep mode. This however increases the power consumption. So it’s imperative that the nodes maintain accurate time with respect to the global clock. [4]

The three basic methods of synchronization available in wireless sensor networks are:

1. Relative Ordering: In this method, the synchronization is on the order of messages or events. Here, clocks are not synchronized but just the order is maintained.
2. Relative Timing: In this method, a node keeps information about its drift and offset in correspondence to neighboring nodes. Thus, nodes have an ability to synchronize to its neighboring nodes.
3. Global Synchronization: In this method, there is a global timescale. All the network nodes to synchronize to this global clock [2].

III. SOURCES OF TIME SYNCHRONIZATION ERRORS

Currently existing time synchronization algorithms and methods work by exchanging messages. Message latency estimates are confounded by random events that lead to nondeterministic message delivery delays; this results directly to the synchronization error. We have following four components of message latency:

Send Time: The time spent at the sender to build the message. it mention a time that required to transfer the message to the network interface.

Access Time: During transmission each packet faces some delay at the medium access control (MAC) layer. Delay depends on the MAC layer and reasons for delay are wait for idle channel or wait for slot for time-division multiple access (TDMA).

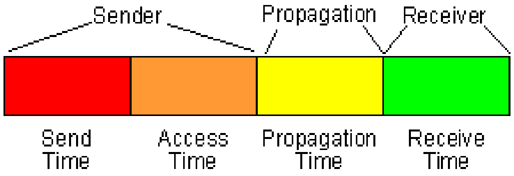

Figure 1- Breakdown of packet delay components [5]

Propagation Time:

The time required for the message to transit from sender to receivers. It include time when sender sends to receiver, time consume during transmission till reach to the receiver.

Receive Time:

The time required for the receiver's network interface to receive the message from the channel and notify the host of its arrival. It is typically the time required for the network interface to generate a message reception signal [3-7].

IV. The Need for Synchronization in WSNs

There are several reasons for addressing the synchronization problem in sensor networks. Some reasons are as following:

- Sensor nods are required to coordinate their operations to perform a particular task. Example of it is Data fusion. In which data collected at different nodes are combined into a meaningful result.
- Life time of network is depending on power. So to increase the life of network we need to used power saving schemes. For example when using power-saving modes, the nodes should sleep and wake up at coordinated times.
- To share the transmission medium we used scheduling algorithms like time-division multiple access (TDMA). It helps in the time domain to eliminate transmission collisions and conserve energy.

V. Requirements of Synchronization Schemes for WSNs

We have many requirements in wireless sensor networks so its result that we must satisfy these when designs an algorithm for time synchronization. It has fewer chances that each algorithm satisfies every requirement. These requirements for any synchronization technique:

- Accuracy: Accuracy or Precision of synchronization technique is depends on the particular application. The results of accuracy maybe vary in microseconds.
- Robustness: If any node in the network is break down or go out of then it does not affect the working of other nods in the network and synchronization scheme.
- Scalability: In some applications, tens of thousands of sensors might be deployed. Any synchronization technique must work well with any number of nodes in the network. According to time any numbers of nods can be increased or decreased.
- Energy efficiency: We know that Network nodes have limited energy resources. All network protocols including synchronization ones should consider this limitation.
- Cost: Due to advanced technologies, network nodes are becoming so small and inexpensive. So its results that synchronization algorithm should not have too much cost and too much large in the size.
- Scope: Time synchronization algorithm may provide a common time or global time to all nodes in the network. So it's too costly in large network and too provided global synchronization.
- Delay: Many applications, like detecting gas leak, require an immediate response. For those kinds of applications, the total required time to synchronize the network must be as low as possible [3-4].

VI. Synchronization Methods for WSNs

There are many time synchronization protocols in wireless sensor networks. These protocols are not too much differing from each other. Today we have many protocols but basic idea behind them is always there, but improving in these protocols is doing there. Three basic protocols are Reference Broadcast Synchronization (RBS), Timing-sync Protocol for Sensor Networks (TPSN), and Flooding Time Synchronization Protocol

(FTSP). These three protocols are the major timing protocols currently in use for wireless networks. Some main protocols are as following:

Traditional Time Synchronization (TTS):

Network time protocol (NTP) is the traditional synchronization method which worked only in wired network. It's also known as Two-way message exchange in which messages are exchanged between a pair of nods, In order to obtain a definitive relation between the two clocks with a single message exchange, two basic assumptions need to be made.

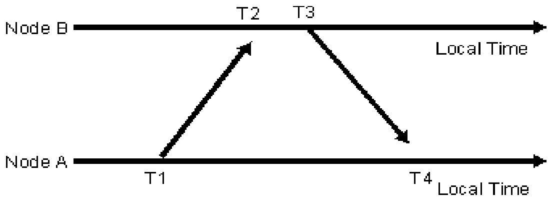

Figure 2. Two - way message exchange between a pair of nodes

1. The offset between the clocks is constant in the small time period during the message exchange.
2. The propagation delay is the same in both directions [3].

Reference Broadcast Synchronization (RBS):

Elson, Girod and Estrin proposed Reference Broadcast Synchronization (RBS) for sensor network and it is a receiver to receiver synchronization. A third party will broadcast a beacon to all the receivers. The receivers will compare their clocks to one another to calculate their relative phase offsets and transmit the recorded times to each other. The time of reference is based on when the nodes receive the beacon. Since the time synchronization protocol is receiver to receiver synchronization, the sender can be removed from the critical path. The simplest form of RBS is one broadcast beacon and two receivers. The timing packet will be broadcasted to the two receivers. The receivers will record when the packet was received according to their local clocks. Then, the two receivers will exchange their timing information and be able to calculate the offset. This is enough information to retain a local timescale. RBS differs from the traditional sender to receiver synchronization by using receiver to receiver synchronization. The reference beacon is broadcasted across all nodes. Once it is received, the receivers note their local time and then exchange timing information with their neighboring nodes. The nodes will then be able to calculate their offset.[3,9]

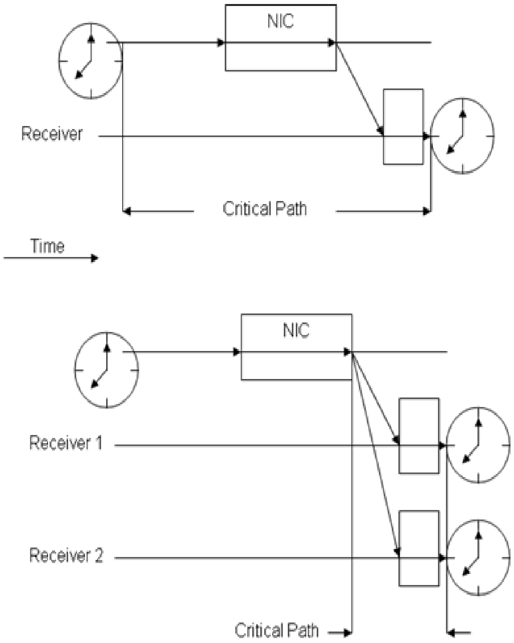

Figure 3 - Comparison of a traditional synchronization system with RBS[3]

Advantages of RBS

- It eliminates the uncertainty of the sender by removing the sender from the critical path. By removing the sender, the only uncertainty is the propagation and receives time.
- The propagation time is negligible in networks where the range is relatively small.
- By removing the sender and propagation uncertainty the only room for error is the receiver uncertainty. Figure 2 illustrates this concept[3-10]

Disadvantage

- The disadvantage to this approach is the additional time synchronization messages needed to exchange among the nodes
- RBS also is limited by the transmission range[11]

Time Sync Protocol for Sensor Networks (TPSN):

TPSN is a Time synchronization protocol that can be applied to spanning tree based network. The protocol constitutes of two levels; level discovery phase where each node determines its node level in Wireless sensor network. In the second level called synchronization level, the synchronization phase begins at root node and propagates to other levels. Here, all nodes in 'I' level gets synchronized to i-1 level nodes[3].The level discovery phase is run on network deployment. First, the root node should be assigned. If one node was equipped with a GPS receiver, then that could be the root node and all nodes on the network would be synced to the world time. If not, then any node can be the root node and other nodes can periodically take over the functionality of the root node to share the responsibility [12].

The synchronization phase is two-way communications between two nodes. As mentioned before this is a sender to receiver communication. Similar to the level discovery phase, the synchronization phase begins at the root node and propagates through the network [10].

Advantages of TPSN

- It minimizes the delay during transmission due to minimize in delay factor components like send time, access time, propagation time, and receive time.
- There is no issue of transmission range because it is a multi-hop protocol.
- It is claimed that TPSN has a 2 to 1 better precision than RBS and that the sender to receiver synchronization is superior to the receiver to receiver synchronization.

Flooded Time synchronization Protocol (FTSP):

The FTSP was designed at Vandebuilt University and implemented using Berkeley Mica2 motes [3]**.** FTSP is sender to receiver synchronization. This protocol is similar to TPSN, but it improves on the disadvantages to TPSN. It is similar in the fact that it has a structure with a root node and that all nodes are synchronized to the root.

In the FTSP, the nodes can form a mesh network where each node sends out time sync message to every other node, or they can form a star network where only one node at a time acts as a master and all the other nodes act as slaves. The master node periodically sends out a time sync message. All the slaves in the network receive this message and synchronize their local clocks to the master clock. If the slaves' clock is off only by a constant, ideally one packet would be necessary to get synchronized with the global clock. But, various issue like temperature, aging produce a drift in the slave's clock that needs to be periodically synchronized to the master clock.[1]

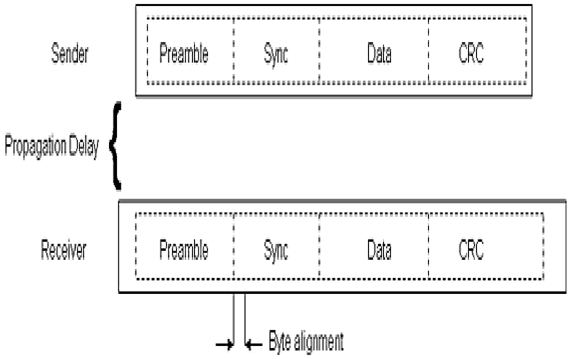

Figure 4 - Data packets transmitted with FTSP

Advantages of FTSP

- FTSP is robust in flooding of synchronization messages and node failure.
- It's also provides the ability for dynamic topology changes.

- It eliminates maximum delay components but not only the propagation time error.
- It utilizes the multiple time stampings and linear regression to estimate clock drift and offset.

Tiny - Sync and Mini – Sync protocol:

Tiny - Sync and Mini - Sync are two lightweight synchronization algorithms proposed by Sichitiu and Veerarittiphan These algorithms use a method similar to the conventional two - way messaging scheme, but obtain a relation between clocks in a rather different way. In this clock can be approximated by an oscillator with a fixed frequency. . Both Tiny-Sync and Mini-Sync use multiple round-trip measurements and a line-fitting technique to obtain the offset and rate difference of the two nodes.

Traditional two-way messaging is used to collect data points, which are used to apply tight bounds on relative drift and relative offset between two nodes. To create data point using two nodes; 1 and 2, node 1 sends a probe message with timestamp to. Node 2 timestamps the received message with tb and sends back an acknowledgment to node 1, immediately or after sometime, which timestamps this acknowledgment with tr[3]. Fig. 5 shows the previous described sequence.

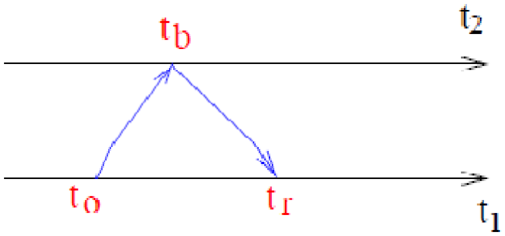

Figure 5. Probe message exchange [12]

Lightweight Tree -Based Synchronization: Some wireless sensor networks need a precision in order of micro seconds, while others need it in order of milliseconds. The algorithms proposed in deal with this fact by assuming that the precision to be achieved is given, and hence, the complexity and computational time depend on this given value.

First algorithm consists of two phases; a spanning tree is constructed in phase one. Then a sender to receiver synchronization is done along the n-1 edges of the spanning tree in phase two. In this centralized algorithm, all nodes synchronize to a central one, called sink or root node. Also, it sets the total time to synchronize the whole network

In the second algorithm, nodes decide when they need to be synchronized based on their distance from the reference node, clock drift, and the required precision. So, any node in the network can initiate the synchronization phase with a reference point, and all nodes along the path to that reference point must be synchronized first. As some nodes do not need to be synchronized all the times, this algorithm removes the overhead of unnecessary synchronization by allowing nodes to make decisions. Aggregation of synchronization requests is used to make this algorithm more efficient. Before the node sends synchronization requests, it checks with its neighbors to see if any request is pending, if any, it will add its request to the pending one, so the path will be synchronized only once. In suggested algorithm divides the whole network into sub trees. In each sub tree, children synchronize to their parent or root node. Then, all parents synchronize with each other [3-14].

Post - facto Synchronization:

Post-facto synchronization was a work by Elson and Estrin, which has led afterward to their RBS scheme. They proposed it to eliminate the week points of traditional synchronization schemes, for NTP. In these local clocks of the sensor nodes should normally run unsynchronized in their own pace and should synchronize only when necessary.). Post-facto synchronization can also be termed as reactive synchronization, while the traditional schemes are proactive, requiring the clocks of sensor nodes to be synchronized before an event of interest occurs [15].

Time-Diffusion-Synchronization Protocol: The time – diffusion protocol (TDP) is a synchronization protocol that maintains an equilibrium time throughout the network, allowing only a small deviation from the equilibrium. The deviation tolerance can be adjusted based on the specific sensor network application. TDP has a two type of periods in long term. These are active and inactive periods. At every d seconds during the active period, some nodes are elected as master nodes that broadcast timing information to their neighbors at every μ seconds. Nodes receiving timing information from the master nodes self - determine to become diffused leader nodes that further broadcast the timing information to their neighbors [3].

VII. OTHER EXITING TIME SYNCHRONIZATION PROTOCOL

In wireless sensor networks, the main problem for time synchronization protocols is the send time, access time, propagation time, and the receive time. We eliminates these by using RBS, TPSN, and FTSP time synchronization protocols and it gives better results in form of accuracy and effectiveness. But today to gain better results we have many new time synchronization protocols. These time synchronization protocols used basic idea from RBS, TPSN, and FTSP. So these are some other popular algorithms Rate adaptive time synchronization (RATS) by Ganeriwal et al, Asynchronous diffusion protocol proposed by Li and Rus, Reachback fi refl y algorithm (RFA), Adaptive - rate synchronization protocol (ARSP) etc.

VIII. CONCLUSION AND FUTURE WORK

This paper presents a survey and analysis the synchronization problem and common challenges for synchronization, discussed the need for synchronization and requirements of synchronization methods and existing time synchronization protocols for wireless sensor networks. The above detailed analysis of time synchronization methodology will guide and help us to introduce new or enhancement of time synchronization protocol According to our survey all the time synchronization protocols for wireless sensor networks have been designed to satisfy different requirements and parameters. All protocols are based on basic idea of RBS, TPSN, and FTSP. We found that RBS and TPSN perform very well in terms of accuracy.

We presented the various algorithms on time synchronization Protocols. So in future we will introduce any new algorithm or method in time synchronization based on RBS, TPSN, and FTSP. It will be possible that we enhanced any existing algorithm which gives us much better results from previous results.

REFERENCES

[1] M Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The Flooding Time Synchronization Protocol," in Proceedings of the 2nd International Conference on Embedded Network Sensor Systems (SenSys ’04), pp. 39-49.

[2] Dennis Cox, Emil Jovanov, Aleksandar Milenkovic, “Time Synchronization for ZigBee Networks”

[3] Fikret Sivrikaya and Bulent Yener, Time Synchronization in Sensor Networks: A Survey, IEEE Network, 18(4): Pages : 45-50, August 2004

[4] Matthias Keller, Jan Beutel, Andreas Meier, Roman Lim, and Lothar Thiele. Learning from sensor network data. In SenSys ’09: Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, pages 383–384, New York, NY, USA, 2009. ACM.

[5] Timing Synchronization in Sensor Networks, http://www.ee.ucla.edu/~saurabh/time_synchronization/

[6] Characterizing Time Synchronization, http://www.circlemud.org/~jelson/writings/timesync/node2.html

[7] S. Ping, “Delay Measurement Time Synchronization For Wireless Sensor Networks,” Technical Report, Intel Research Berkeley Lab, 2003

[8] J. Elson, L. Girod, and D. Estrin, “Fine-Grained Time Synchronization using Reference Broadcasts," Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI 2002), Boston, MA, December 2002.

[9] Hyojung Lee, Wonpil Yu, Youngmi Kwon, “Efficien RBS in Sensor Networks,” in IEEE ITNG, 2006.

[10] S. Ganeriwal, R. Kumar, M. Srivastava. ”Timing Sync Protocol for Sensor Networks,” ACM SenSys ’03, 2003.

[11] Feng Wang; Jiangchuan Liu, “RBS: A Reliable Broadcast services for large – Scale Low Duty- Cycled Wireless Sensor Networks,” ICC '08. IEEE International Conference on Communications

[12] S. Ganeriwal, R. Kumar, and M. Srivastava, “Timing Sync Protocol for Sensor Networks," ACM SenSys, Los Angeles, November 2003.

[13] M.L. Sichitiu and C. Veerarittiphan, “Simple, Accurate Time Synchronization for Wireless Sensor Networks, "Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE Volume 2, 20-20 March 2003 pps.1266 - 1273 vol.2.

[14] Mingxia Xu, Minjian Zhao, Shiju Li Lightweight and Energy Efficient Time Synchronization for Sensor Network, IEEE International Conference on Wireless Communications, Networking and Mobile Computing, Volume 2: Pages 947-950., Sept 2005

[15] J. Elson , Time Synchronization in Sensor Networks , PhD Thesis, University of California Los Angeles, 2003