IJITKM Volume 6 • Number 2 • July-December 2013 pp. 143-150

Comparative Analysis of Time Synchronization Protocols for Wireless Sensor Network

Shikha Singh, Yogesh Chaba
Department of CSE, GJUS&T, Hisar, India

Abstract – Wireless sensor network are formed by wireless sensor nodes, having capabilities of perception, and are useful in communication. The specific features of sensor networks as limited energy, wireless channel conditions, and dynamic topology etc results in significant requirement for being synchronized. The time in which the data is measured becomes important. Thus synchronization protocols also become vital to coordinate the tasks running on different nodes of the network. Time synchronization is one of the basic middleware services of WSN. The aim of time synchronization is to keep the time of all sensor nodes in network as consistent as possible. It is the process of ensuring that physically distributed nodes in WSN have a common notion of time. A number of synchronization schemes have been designed for WSN applications to meet with these challenges such as RBS, TPSN etc. In this paper a comparative study among different time synchronization protocol RBS and TPSN is done. These protocols are studied and compared using various factors as number of packets transmitted for synchronization, number of clusters formed, etc..
Keywords – sensor networks, synchronization.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) are networks of light-weight sensors that are battery powered used majorly for monitoring purposes. These are spatially distributed autonomous sensors which cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants [20]. WSN are used to locate not only the objects whose area of location is known, but also the objects whose location is anticipated to be around a certain domain [1]. Wireless sensor networks are an increasingly attractive means to bridge the gap between the physical and virtual world. It has very broad application prospects and great potential value in many areas, such as in the military and national defense, environmental monitoring, and biomedical, remote monitoring dangerous areas, and so no [2].

A wireless sensor network consists of a number of sensor nodes with the capability of storing, processing and relaying the sensed data, and often has a base station called Sink for further computation. Wireless sensor networks are an increasingly attractive means to bridge the gap between the physical and virtual world. While WSNs are increasingly equipped to handle some of the complex functions, network processing such as data aggregation, computation and transmission activities requires these sensors to use their Time Synchronization efficiently in order to extend their effective network Throughput. Recent various areas like terrorist and animal warfare counter measures require distributed sensor networks that can be deployed easily using aircraft, and have self-organizing capabilities etc. with the advancement in the recent technology development of small, battery-powered, wireless sensor nodes is enabled. But, sensors have serious resource constraints including battery lifetime, communication bandwidth, CPU capacity and storage also, due to increase in mobility of the sensor node, may increase the complexity of sensor data because a sensor may be in a different neighborhood at any point of time [3]. Sensor nodes are prone to Time Synchronization drainage and failure, and their battery source might be irreplaceable, instead new sensors are deployed. Thus, the constant re-energizing of wireless sensor network as old sensor nodes die out and/or the uneven terrain of the region being sensed can lead to Time Synchronization imbalances or heterogeneity among the sensor nodes. WSN are useful in monitoring complex tasks. In this new class of networks there also challenges in many areas. In contrast to traditional wired networks, a WSN is highly dynamic. In WSN system, the gateway acts as the network coordinator responsible of node authentication, message buffering, and connection between the wireless networks to the wired network, where one can collect, process, analyze, and use measurement data [21].

Each node consists of processing capability, may also contain multiple types of memory (program, data and flash memories), have a RF transceiver (usually with a single omni-directional antenna), have a power source (e.g., batteries and solar cells), and accommodate various sensors [22]. For various applications in WSNs, several special requirements were proposed as follows [7]:

- Energy saving: Some synchronization schemes require energy saving to prolong network life. This requirement claims the least synchronization cost to get fixed goal.
- Stability: many thousands of sensors may have to be deployed in a large area without manual intervention for a long time. Therefore, a stable synchronization for the network seems to have great impacts.
- Scope and Availability: the geographic span of nodes that are synchronized, and completeness of coverage within that region. The scope might be as small as a pair of nodes exchanging data, or as large as the entire network.

2. NEED OF TIME SYNCHRONIZATION

Wireless sensor networks (WSN) are distributed systems consisting of a large number of sensor nodes each of which maintains a local clock. In order to facilitate data aggregation and node localization, these local clocks have to be synchronized in some way [16]. Number of areas are there which require cooperative sensing nodes involved, to agree on a common time. These situations mandate the necessity of one common notion of time in wireless sensor networks. In a wireless sensor network, sensors operate independently, so their local clocks may not be synchronized with one another. This may result in difficulties when trying to integrate and interpret information sensed at different nodes [4]. Time synchronization as an important component in WSNs is necessary for the node localization, low power listening, data fusion, TDMA, synchronized hopping system and the time synchronization error precision may changes for different applications [17].

A good protocol design should be able to scale well both in Time Synchronization heterogeneous and homogeneous settings, meet the demands of different application scenarios and guarantee reliability. A number of factors make flexible and robust time synchronization particularly important in WSNs, while here it is more difficult to achieve them than in traditional networks. It is process of ensuring that physically distributed processors have a common notion of time [5]. Like many other distributed systems, it is very difficult, to achieve an agreement on current time between different nodes in wireless sensor networks in possession of their own local clocks. Without appropriate synchronization techniques, the timestamps associated with sensing date cannot be correctly interpreted; this may, in turn, cause difficulties in certain sensor network applications [6].The broad nature of sensor network applications leads to timing requirements whose scope, lifetime, and precision differ from traditional systems [13]. Time synchronization methods for sensor networks must take into account the time and energy that they consume. In synchronization the total time consumed depends on number of messages being used for synchronization [15]. In general, the time involved in sending a message from a sender to a receiver is the result of the following four factors, all of which can vary non- deterministically [8]:

- **Send time:** The time spent at the sender to build the message.
- **Access Time:** This is the delay incurred when waiting in the network interface for access to the transmission channel.
- **Propagation time:** This is the time needed for the message to propagate from sender to receiver over the wireless medium.
- **Receive time:** This is the time needed for processing at the receiver's network interface.

Types of synchronization [11]

Different types of synchronization schemes are described based on the interaction of different senders and receivers as follow:

Sender-to-receiver synchronization: This traditional approach usually happens in three steps. Firstly the sender node periodically sends a message with its local time as a timestamp to the receiver. The receiver then synchronizes with the sender using the timestamp it receives from the sender. The message delay between the sender and receiver is calculated by measuring the total round-trip time, from the time a receiver requests a timestamp until the time it actually receives a response.

Receiver-to-receiver synchronization: In this instead of interacting with a sender, receivers exchange the time at which they received the same message and compute their offset based on the difference in reception times.

3. TIME SYNCHRONIZATION PROTOCOLS FOR WSN

Time synchronization is a major component for many applications and operating systems in distributed computing environments. In general, the uncertainty sources affecting time synchronization can be grouped into two main categories [18]:

- The clock-related uncertainty contributions which include the instability of the employed crystal oscillators (e.g. due to ageing, temperature changes, manufacturing tolerances, supply voltage fluctuations and short-term random jitter) as well as the resolution of the digital timer embedded on the node and the uncertainty affecting the control signal enabling and disabling the timer;
- The communication-related uncertainty contributions that comprise the end-to-end delay due to packet time-stamping and encoding, radio channel access, data sending and receiving, packet propagation delay etc.

There are many time synchronization protocols defined previously. However, these traditional synchronization schemes such as Network Time Protocol (NTP) are not suitable for use in WSN due to some of the specific reasons as [9]:

Energy awareness - For time synchronization protocols in traditional wired networks, using the CPU power, listening to the signaling messages from the network and supporting transmissions have a minimal impact on the operation of a communicating node. Traditional synchronization protocol such as NTP assumes CPU is always available. While in WSN CPU mostly operate in idle power mode.

Dynamic infrastructure -Traditional networks were designed on a static infrastructure. Although it is possible to use statistical information to decide which servers to use to connect, nodes are still dependent on their initial configuration. Because of the highly dynamic nature of WSN, such a static configuration is not possible. A number of different time synchronization protocols proposed for WSN have been described as follow:

3.1 Reference Broadcast Synchronization (RBS)

RBS provides synchronization for a whole network. Here the receiver nodes exchange their respective reception times of beacon and use linear regression to compute relative offsets and rate differences to each other [4]. In RBS scheme, nodes periodically send beacon messages to their neighbors using the network's physical layer broadcast. Recipients use the message's arrival time as a point of reference for comparing their clocks. The message contains no explicit timestamp, nor is it important exactly when it is sent [14]. RBS eliminates error introduced by the Send time and Access time from critical path.RBS works in simple manner. RBS consist of one broadcast beacon and two receivers. And 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, this recorded time is used by the two receivers for exchanging their timing information and further to calculate the offset. For n receivers; where n is greater than two, this may require more than one broadcast to be sent. Increasing the broadcasts will increase the precision of the synchronization. [10]

The simplest form of RBS is executed in three steps:

1. A node broadcasts a reference beacon.
2. Each node that receives the beacon, records its arrival time according to the node's local clock.
3. The nodes exchange their observations. Using this information, each node can compute it's offset to any other node.

Reference Broadcast Synchronization (RBS) is a time synchronization protocol in which nodes make use of the broadcast possibility of the network for sending reference beacons to their neighbors. The major fact is that these beacons do not contain a timestamp. Receivers use the arrival time of these beacons for comparing their clocks [9].

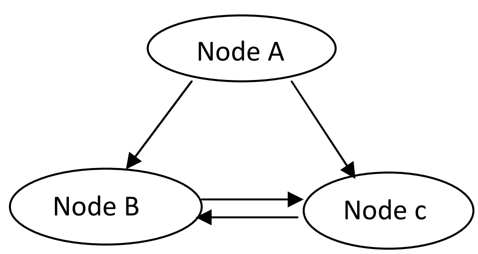


Figure 1: RBS Message Exchange

A beacon can be sent by any node in the network, thus no special nodes are needed. This protocol helps in removing two of the main error sources of clock synchronization, which are uncertainties at Send Time and Access Time. Furthermore, the difference between propagation times is negligible compared to the uncertainty at Receive Time, which becomes the only error source. This protocol works as shown in the figure below. Here Node A transmit beacon to nodes B and C. Later B and C nodes interchange the time of receiving this beacon according to their local clock. And in this way they synchronize.

3.2 Timing-sync Protocol for Sensor Network

In TPSN protocol firstly hierarchical structure is established over the network and then a pair wise synchronization is performed along the edge of this structure to establish a global timescale throughout the network [19]. The disadvantage is that a message acknowledgement pair is used for a pair of nodes prohibiting the use of message broadcasts resulting in higher benefits [13].

TPSN is a Sender-Receiver based time synchronization protocol for WSN. It basically involves two phases, the level discovery phase and the synchronization phase as follow. They work in the following way:

Level discovery phase:

It creates the hierarchical topology of the network in which each node is assigned a level. Only one node resides on level zero, the root node. Once the root node is determined, it will initiate the level discovery. The root, level zero, node will send out the level discovery packet to its neighboring nodes. The level discovery packet consists of the identity and level of the sending node. The neighbors of the root node will then assign themselves as level one. They will in turn send out the level discovery packet to their neighboring nodes. This process will continue until all nodes have received the level discovery packet and are assign a level [10].

Synchronization phase:

All i level nodes will synchronize with i-1 level nodes. This will synchronize all nodes with the root node TPSN is easy to implement and an effective time synchronization protocol but it has some weaknesses [10].

In the following fig.2 two way message passing is shown. This messaging can synchronize a pair of nodes by following this method. The times T1, T2, T3, and T4 are all measured times. Node A will send the synchronization pulse packet at time T1 to Node B. This packet will contain Node A's level and the time T1 when it was sent. Node B will receive the packet at time T2. Time T3 is when Node B sends the acknowledgment packet to Node A.

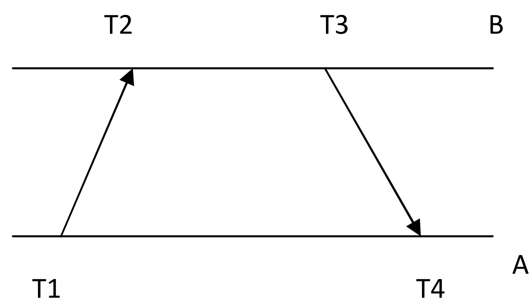


Figure 2: Two way message passing

That packet will contain the level number of Node B as well as times T1, T2, and T3. By knowing the drift, Node A can correct its clock and successfully synchronize to Node B. TPSN is easy to implement and an effective time synchronization protocol but it has some weaknesses [12].

4. EVALUATION OF PROTOCOLS BY SIMULATION WORK AND RESULTS

Time synchronization can be achieved by number of protocols. But these protocols when compared over different parameters varied. This paper presents the simulation of RBS and TPSN protocols for time synchronization of WSN. For hardware implementation TinyOS is being used for time synchronization, but the work presented here was done using MATLAB7.5.0 simulator. The lifetime of a WSN during time synchronization can be defined as the time elapsed until the first node dies, the last node dies, or a fraction of nodes die. To maximize network lifetime, a trade-off is considered between total Time Synchronization energy consumption and Time Synchronization balancing energy among sensors. Here the scenario also works for cluster-based time synchronization for WSN, there is one sensor called as CH which acts as router. All non-CH nodes transmit their data to their CH, which routes it to the remote PN. However, since CHs consume more Time Synchronization energy in aggregating and routing data. We present an optimal allocation of states to sensors which maximizes the efficiency of sensors' Time Synchronization capability. A stationary sensor network is assumed to be deployed to observe events. In this work various parameters are being set already as E-initial, as initial energy while Time Synchronization of nodes. E-Transmission, being Time Synchronization energy consumed during transmission. Others are being estimated to obtain a comparison of working of these protocols as E-remaining by E-initial - E-transmission, T-life =

Network Throughput of node. Remaining Network Throughput of node = (E-remaining/E-initial) * T-Life, Other involved measures are T-s =sending time-stamp of last data packet, T-R = receiving time stamp of last data packet for time synchronization. In TPSN, each node would send residual Time Synchronization along with the sending time stamp T-S and the remaining lifetime of battery to the base station. When the base station receives the packet, it calculates T-R - T-S (the difference between receiving timestamp and current time stamp) And If difference > = remaining lifetime of node the node will become=cluster head else if remaining lifetime = max among all nodes of the cluster choose the node as cluster head, in this way it works.

When a WSN grows large, the synchronization error may increase. In a scalable approach when a WSN is extended, the computing time used by the new nodes should automatically be synchronized to the time of the existing nodes in the network. When a WSN grows large, the synchronization error may increase. This is especially a problem in TPSN, where static master nodes are used. RBS appears most scalable among these protocols. Since in RBS, the new nodes do not need to perform any special actions. They can just behave like the existing nodes that are already in the network. In TPSN, the send time error is minimized by using time-stamping packets .Here the timestamp is added to the packet when it is ready to be transmitted. The receive time in TPSN is also minimized by time-stamping received messages. RBS eliminates both send time and access time. The beacon message that is initially sent does not contain a timestamp. When the master node fails, the whole network can no longer be synchronized. This problem can occur in TPSN. RBS is not based on a hierarchy, and is therefore probably the most robust protocol among all. The following result graphs were obtained as results of these protocols simulation for synchronizations. These shows the number of dead and alive nodes while synchronization rounds which were fixed to be maximum, r = 5000 with a wireless sensor network scenario of number of nodes (n) being 100. In Figure 3 the case of RBS, clearly shows the number of dead nodes starts increasing around 1500 rounds whereas after 2500 round it becomes constant to around 100. Including Figure. 4 it is also evident that in case of TPSN these number of dead nodes starts to pull up after 1000 rounds only. It simply represents as the synchronization rounds starts increasing the number of dead nodes in TPSN starts increasing more randomly then in RBS as the message exchange in TPSN are more resulting in more energy loss as compared to RBS. And in TPSN it becomes constant at to 100 at around 1500 round. So number of dead nodes also reaches constancy in TPSN earlier than RBS.

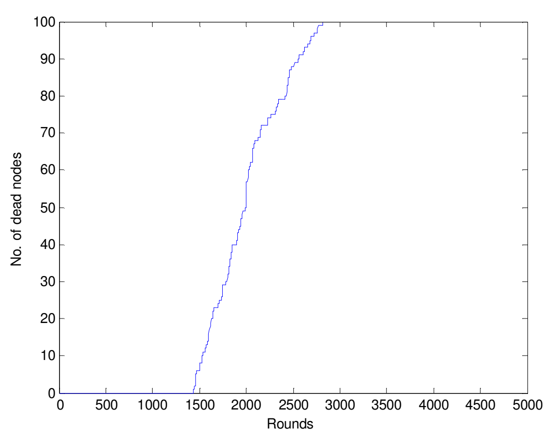


Figure 3: Dead nodes in RBS

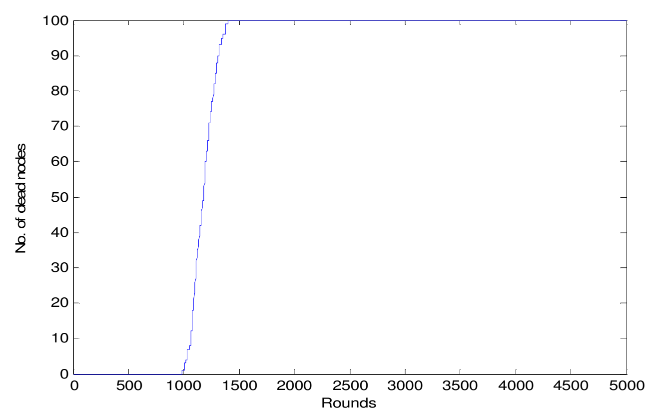
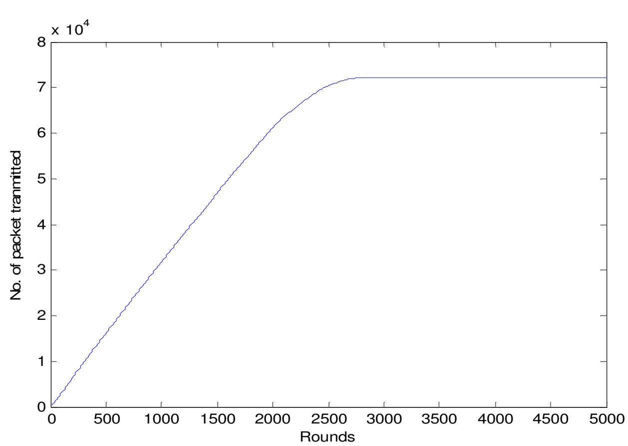


Figure 4: Dead nodes in TPSN



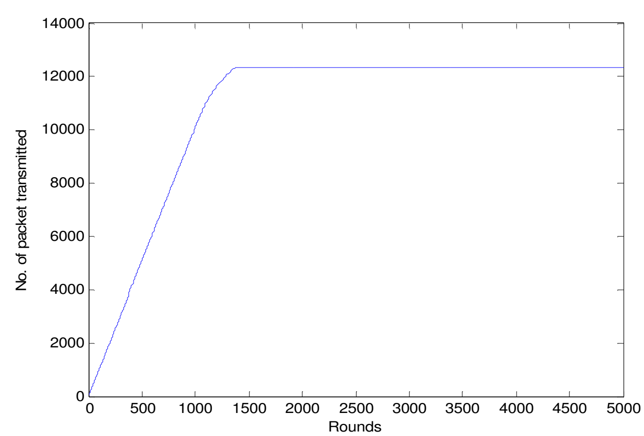


Figure 5: Packet transmissions in RBS

Figure 6: Packet transmissions in TPSN

Here Figure 5 shows the packets transmitted for synchronization in the case of RBS at around synchronization round 500,reached around 1 x 10^4 while Figure 6 verifies the case for TPSN where at around number of synchronization round being 500, the packets transmitted for synchronization reached around 2000,while reaching 2500 the number of packet transmitted reaches 12000.

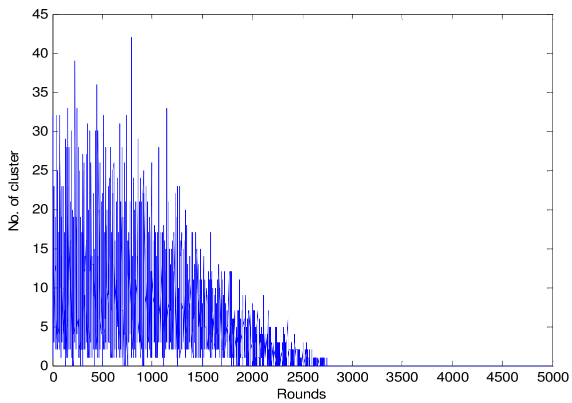


Figure 7: Cluster formation in RBS

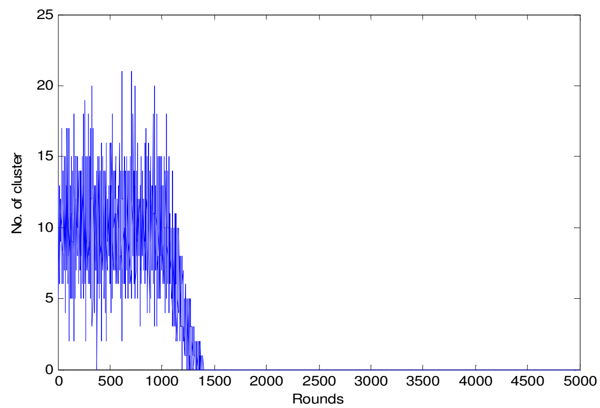


Figure 8: Cluster formation in TPSN

Here clustering is also used while synchronization among the nodes. Figure 7 Shows the case of RBS where the clusters formed are much more. As less amount of energy is being consumed in RBS for synchronization so the total remaining energy is much more. Here the clusters formed are more when compared to number of rounds for synchronization.

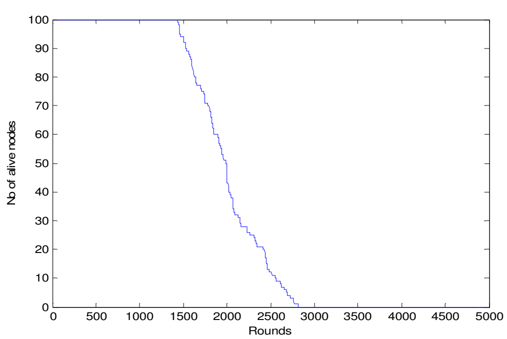


Figure 9: Alive nodes in RBS

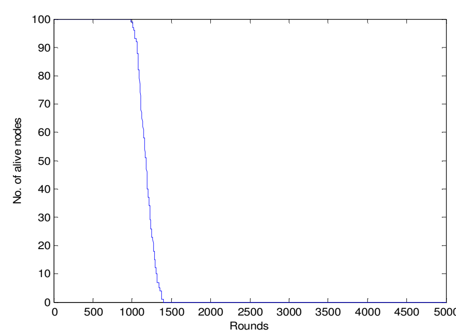


Figure 10: Alive nodes in TPSN

Another area of comparison is the number of alive nodes between the two protocols. Figure 9 is used to determine the number of alive nodes representation in case of RBS. Here around 1500 being the synchronization rounds, the number of alive nodes starts decreasing from 100. And till rounds are around 2800 the number of alive nodes decreases to 0. While Figure 10 shows number of alive nodes in the case of TPSN, where it starts decreasing from a value 100 when the synchronization round are 1000 to a value 0 when the synchronization round are about 1400. These results gives a complete scenario of the working of both the protocols involved in the time synchronization process for wireless sensor network. On the basis of the above comparison among both protocols following table can be concluded as below.

TABLE 1. STUDY OF DIFFERENT PROTOCOLS AFTER RESULTS

Protocol	No of alive nodes	No of dead nodes	Packet transmission	Rate of cluster formation while synchronization

RBS	HIGH	LESS	HIGH	HIGH
TPSN	LESS	HIGH	LESS	LESS

5. CONCLUSION

Wireless sensor networks have tremendous advantages for monitoring object movement and environmental properties but require some degree of synchronization to achieve the improved results. Thus time synchronization acts as important issue .Since WSN are different from traditional wired networks, so the existing time synchronization protocol such as network time protocol are not appropriate for WSN. In this paper RBS and TPSN time synchronization protocols for WSN are described which are compared over factor as clusters formed, dead and alive nodes in synchronization, packet transmitted etc. These clearly shows that RBS serves as better protocol then TPSN since it requires less number of synchronization messages involved. Also overall energy and resources involved in RBS is less than TPSN. Since the number of packets required for synchronization are also less in RBS, so it serves better. None of this protocol is much better or much worse, but RBS serves little better over TPSN. More work can be done in future, considering the various attacks on these protocols. More work could be done by comparing these to other synchronization protocols.

6. REFERENCES

1. Shivangi Raman, Amar Prakash, Kishore Babu Pulla, Prateek Srivastava, Ashish Srivastava, Shveta Singh, “Wireless sensor networks: A Survey of Intrusions and their Explored Remedies”, Proc of International Journal of Engineering Science and Technology, India, Vol. 2, pp. 962-969 (2010)
2. Xuhui Chen, Peiqiang Yu, ”Research on hierarchical mobile wireless sensor network architecture with mobile sensor nodes”, Proc of International Conference on Biomedical Engineering and Informatics, China, Vol. 7, pp.2863-2867(2010)
3. Mittal. R, Bhatia, M.P.S.,” Wireless sensor networks for monitoring the environmental activities”, Proc of Second International Workshop on Education Technology and Computer Science, China, Vol 2, pp.186-191(2010)
4. Surendra Rahamatkar, Ajay Agarwal, Narendra Kumar,” Analysis and Comparative Study of Clock Synchronization Schemes in Wireless Sensor Networks”, Proc of International Journal on Computer Science and Engineering, India, Vol 2, pp. 536-541(2010)
5. Jeremy Elson, and Kay Romer, “Wireless Sensor Networks: A New Regime for Time Synchronization”, Proc. of First workshop on hot topics I networks, New Jersey, pp. 393–422(2002)
6. Jing Feng, Richard W. Pazzi, Azzedine Boukerce,”The Impact of Discrete Clock on Time Synchronization in Wireless Sensor Networks”, Proc of IEEE Conference on Local Computer Networks, Canada, pp. 1054-1060(2009)
7. Gowrishankar.s, T.G.Basavaraju, Manjaiah D.H, and Subir Kumar Sarkar, “Issues in wireless sensor networks”, Proc. of the world Congress on Engineering, U.K, Vol. I, pp.176-187 (2008)
8. Palchaudhuri S., Saha A.K., and Johnsin, D.B., “Adaptive Clock Synchronization in Sensor Networks”, Proc. of IEEE on Information Processing in Sensor Networks, Houston, TX, USA, pp. 340-348(2004)
9. Bertold Van Voorst, “Time Synchronization in wireless sensor networks”, Proc. of conference on IT, Netherlands(2006)
10. Prakash Ranganathan, Kendall Nygard, “Time synchronization in wireless sensor networks: A Survey”, Proc of International Journal of UbiComp (IJU), USA, Vol.1, No.2 (2010)
11. F.Sivrikaya and B. Yener, “Time Synchronization in Sensor Networks: A Survey”, Proc. of IEEE on Network, USA, Vol.18, pp.45-50(2004)
12. Kulakli, A.B., Erciyes, K., “Time synchronization algorithms based on Timing-sync Protocol in Wireless Sensor Networks”, Proc. of IEEE on Computer and Information Sciences, Izmir, pp.1-5(2009)
13. Woo Hyuk Jang, and Youngmi Kwon, “Compensated Time Synchronization in WSN with Multiple Reference Nodes”, Proc. of IEEE on Networked Computing and Advanced Information Management, Daejeon, Vol.1, pp.229-234(2005)
14. Hyojung Lee, “Efficient RBS in Sensor Networks”, Proc. of IEEE on Information Technology: New Generations, Daejeon, pp.279-284(2006)

15. Miklos maruti, Barnislav Kusy, Gyula simon, Akos ledeczi, "The flooding time synchronization protocol", Proc. of the 2nd international conference on Embedded networked sensor systems, USA , pp.39-49(2004)
16. Yin Xianglan, "ASTS: An Agile Secure Time Synchronization Protocol for Wireless Sensor Networks", Proc. of IEEE on Wireless Communications, Networking and Mobile Computing , pp.2808-2811(2007)
17. Fuqiang Wang, "The Design and Realization of High-precision Time Synchronization Protocol in Wireless Sensor Networks based FTSP ", Proc. of 8th world congress on Intelligent Control and Automation , China, pp.6983-6987(2010)
18. D. Macii, D. Petri, "An Adaptive-Rate Time Synchronization Protocol for Wireless Sensor Networks", Proc. of IEEE on Instrument and Measurement Technology, Italy, pp.1-6(2007)
19. Xiaolu liu, Shumin Zhou, "Evaluation of several time synchronization protocols in WSN", Proc. of IEEE on Information Science and Management Engineering, China, Vol.1, pp.488-491(2010)
20. http://en.wikipedia.org/wiki/Wireless_sensor_network
21. http://zone.ni.com/devzone
22. http://computer.howstuffworks.com/mote.htm