International Journal of Education and Science Research Review
E-ISSN 2348-6457
August- 2017, Volume-4, Issue-4 P-ISSN 2349-1817
www.ijesrr.org Email- editor@ijesrr.org

ANALYSIS OF TIME SYNCHRONIZATION ALGORITHM BASED ON CLUSTER FOR WIRELESS SENSOR NETWORKS

Vikas Jain
Assistant Professor
Dept. of Computer Application
C.C.S. University, Meerut

ABSTRACT

Time Synchronization algorithm based on Cluster for WSN was proposed for Wireless Sensor Network (WSN) Clustered Time Synchronization algorithm for Wireless Sensor Network. This algorithm consists of two phases: In the Cluster-Inter Synchronization phase, It adopt pair-wise packet exchange mechanism to achieve time synchronization between the Base station and cluster heads through establishing a hierarchical topology structure. In the Cluster-Intra Synchronization phase, It used reference broadcast mechanism to achieve time synchronization between the cluster heads and cluster members. The purpose of this algorithm is to set the logical clock of the cluster heads and cluster nodes with global time. The simulation result shows that this algorithm has better synchronization accuracy and lower power consumption and better synchronization precision as compared to Reference Broadcast Synchronization (RBS), Timing-Sync Protocol for Sensor Networks (TPSN) algorithms.

KEYWORDS: *Wireless Sensor Networks, Time Synchronization*

INTRODUCTION

As advances in technology have enabled the development of tiny low power devices capable of performing sensing and communication tasks, sensor networks have emerged and received the attention of many researchers. Sensor networks are a special type of ad hoc networks, where wireless devices (usually referred to as nodes in the network) get together and spontaneously form a network without the need for any infrastructure. Because of the lack of infrastructure (e.g., routers in traditional networks), nodes in an ad hoc network cooperate for communication by forwarding each other's packets for delivery from a source to its destination. This yields a multi-hop communication environment. Although they are a special type of ad hoc networks, sensor networks have their own characteristics, such as very limited energy sources, high density of node deployment and cheap & unreliable sensor nodes. With these extra limiting factors for their operation, sensor networks are designed to perform complex tasks such as emergency applications, environment monitoring, information gathering in battlefields, and many other uses, connecting the physical world to the virtual world of computers.

Clock synchronization is the process of ensuring that physically distributed processors have a common notion of time. It has a significant effect on many areas like security systems, fault diagnosis and recovery, scheduled operations, database systems, and real-world clock values.

In centralized systems, there is no need for synchronized time because there is no time ambiguity. A process gets the time by simply issuing a system call to the kernel. When another process then tries to get the time, it will get either an equal or a higher time value. Thus, there is a clear ordering of events and the times at which these events occur.

In distributed systems, there is no global clock or common memory. Each processor has its own internal clock and its own notion of time. In practice, these clocks can easily drift seconds per day, accumulating significant errors over time. Also, because different clocks tick at different rates, they may not remain always synchronized although they might be synchronized when they start. This clearly poses serious problems to applications that depend on a synchronized notion of time. In this chapter we present motivations and research directions for studying time synchronization in sensor networks and its future scope for upcoming researches.

TIME SYNCHRONIZATION PROBLEM

Computing devices are mostly equipped with a hardware oscillator assisted computer clock, which implements an approximation *C*(*t*) of real-time *t*. The angular frequency of the hardware oscillator determines the rate at which the clock runs. Even though the frequency of a clock changes over time, it can be approximated with good accuracy by an oscillator with fixed frequency. Then, for some node *i* in the network, we can approximate its local clock as:

$$Ci(t) = ai(t) + bi \qquad (1)$$

Where ***ai(t)*** is the clock *drift*, and ***bi(t)*** is the *offset* of node *i*'s clock. *Drift* denotes the rate /frequency of the clock, and *offset* is the difference in value from real time *t*. Using equation (1), we can compare the local clocks of two nodes in a network, say node 1 and node 2 as:

$$C1(t) = a12 \,.\, C2(t) + b12 \qquad (2)$$

We call a**12 the** relative drift, and b**12 the** relative offset between the clocks of node1 and node2. If two clocks are perfectly synchronized, then their relative drift is 1 -indicating the clocks have the same frequency and their relative offset is zero -meaning they have the same value at that instant. Some studies in the literature use “skew" instead of “drift", defining it as the difference between clock rates. Also, the “offset” may equivalently be mentioned as “phase offset".

SYNCHRONIZATION DIFFICULTIES

One of the major difficulties involved in synchronizing components of a distributed network is the time it takes for a synchronization signal to reach the outlying components. Suppose, for example, we have a network of three nodes all in line with each other (Figure-1).

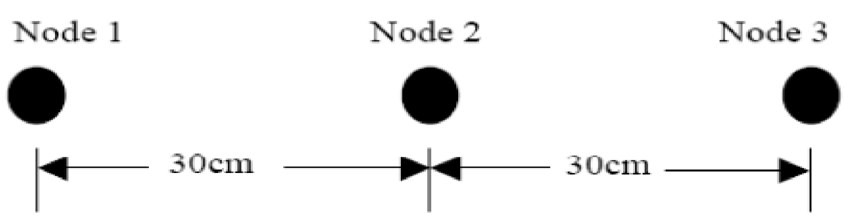

Figure 1: Three node, distributed network.

Next suppose that the node 1 is defined as the master clock. When synchronization is required node 1 will reset its clock and send a signal to nodes 2 and 3 to do the same. If the signal is travelling at the speed of light (approximately 3x108ms-1) then by the time the signal has reached node 3, the clock in node 2 is at 1ns and the master clock is at 2ns. This is a problem if the required accuracy is better than 2ns across the whole network.

SYNCHRONIZATION METRICS

In this section we present a broad set of requirements for the synchronization problem. These requirements can also be regarded as the metrics for evaluating synchronization schemes on sensor networks. However, there are tradeoffs between the requirements of an efficient synchronization solution, thus a single scheme may not satisfy them altogether.

CONCLUSION

The synchronization problem on a network with n devices corresponds to the problem of equalizing the computer clocks of different devices. The synchronization can be either *global*, trying to equalize $Ci(t)$ for all $i = 1 to\ n$, or it can be *local*, trying to equalize $Ci(t)$ for some set of nodes (mostly the ones that are spatially close). Equalizing just the instantaneous values (correcting the offsets) of clocks is not enough for synchronization since the clocks will drift away afterwards. Therefore a synchronization scheme should either equalize the clock rates/frequencies as well as offsets, or it should repeatedly correct the offsets to keep the clocks synchronized over a time period.

REFERENCES

1. D. L. Mills, "Modeling and Analysis of Computer Network Clocks", Technical Report, 92-5-2, Electrical Engineering Department, University of Delaware, 2007.
2. Erchin Serpedin, "Adaptive multi-hop timings synchro-nization for wireless sensor networks", Int. Symp. Signal Processing and Its Applications, (IEEE-ISSPA) pp. 1-6, 2014.
3. F. Perget, R. Plana, "Cross- layered Synchronization Protocol for Wireless Sensor Networks", The Ninth IEEE International Conferenceon Networks (ICN) pp. 167-172, 2010.
4. Akyildiz, "Time-Diffusion Synchronization Protocols for Sensor Networks", IEEE/ACM Transactions on Networking, pp 384-397, 2015.
5. J. Van Greunen & J. Rabaey, "Lightweight Time Synchronization for Sensor Networks", Proc. 2nd ACM Int. Workshop on Wireless Sensor Networks and Applications, San Diego, California, pp. 11–19, 2016.
6. Qianping Wang, Yan Zhao, "Time Synchronization algorithm based on Cluster for WSN", The 2nd IEEE International Conference (ICIME), pp 126-130, 2016.
7. S. Ganeriwal R Kumar, M Srivastava "Timing-Sync Protocol for Sensor Networks", Proc. First ACM Int. Conf. on Embedded Networked Sensor Systems, Los Angeles, California, pp. 138-149, Nov. 2017.
8. Sandipan Basu, "An Efficient Approach of Election Algorithm in Distributed Systems" Indian Journal of Computer Science and Engineering (IJCSE), vol. 2(1), pp. 16., 2016.
9. Y. Li, A. Syed, J. Wills, W. Ye, "Underwater sensor networking: research challenges and potential applications", Proceedings of the Technical Report ISI-TR-05-603, USC/ Information Sciences Institute, 2015.