International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-1 Issue-2, June 2012

Clustered Time Synchronization Algorithm for Wireless Sensor Networks

A. P. Zurani, B. N. Mahajan

***Abstract:** A Time Synchronization algorithm based on Cluster for WSN was Proposed for WSN (Wireless Sensor Network) -(CTS) Clustered Time Synchronization algorithm for WSN.*

This algorithm consist 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:** Sensor Networks, Time Synchronization, Malicious Nodes, Battery, Clustered Synchronization, Delay, Energy, Cluster, Reference Broadcast.*

I. INTRODUCTION

In wireless sensor network large number of nodes are deployed, these nodes sense the data from the environment and communicate the same to the sink or base station. To identifying the correct event time, these nodes need to be synchronized with global clock. Therefore, time synchronization is significant feature in WSNs. Since the performance of time synchronization algorithm is greatly influenced by many factors. The energy conservation is one of the important issues in WSNs which helps to prolong the lifetime of the network.

In this paper we propose a Clustered Time Synchronization algorithm for Wireless Sensor Networks where the synchronization is performed after the formation of clusters.The root node initiates the synchronization process after formation of the cluster. In first phase all level-1 Cluster Heads (CHs) synchronized and then the level-2 cluster heads and so on. This process remains continue until all the nodes in the network has been synchronized.

Revised Manuscript Received on 30 June 2012.
* Correspondence Author
Ms A. P. Zurani*, G. H. Raisoni College of Engineering, Digdoh hills, Wanadongari, Nagpur, India.
Prof. B. N. Mahajan, Asst. Professor, G. H. Raisoni College of Engineering, Digdoh hills, Wanadongari, Nagpur, India.

The Simulation result shows that our algorithm consumes less power and provides better synchronization accuracy than Reference Broadcast Synchronization (RBS), Timing-Sync Protocol for Sensor Networks (TPSN).

The various sections of this paper are organized as follows: In Section 2 we discuss hierarchical clustering. Section 3 contains the time synchronization components. Section 4 contains the the design of the angorithm CTS. Section 5 contains implementation of CTS algorithm. Section 6 contains the simulation and analysis and finally we conclude the paper in Section 7.

II. HIERARCHICAL CLUSTERING

In wireless sensor network applications clustering is used for various purposes like data fusion, routing and optimizing energy consumption. A sensor network can be made scalable by gathering the sensor nodes into clusters. Every cluster has a cluster head (CH). In a clustered based hierarchical wireless sensor network CHs can be used to process and send the information while the sensor nodes are used to sense the data. The advantage of using the clustering for time synchronization is that CH can prolong the battery life of the individual sensors and also the network lifetime. CH can reduce the rate of energy consumption by scheduling activities in the cluster. Clustering reduce the communication overhead for synchronization. There are various protocols for forming the clusters, but in this paper we are using a top-down tree-based architectural design for creating multi-level clustering (MLC) wireless sensor networks. In this algorithm cluster-heads form a tree to reach each node in the network. As shown in Fig. 1, the root of the Multilevel Clustering tree is always the sink of a wireless sensor network.

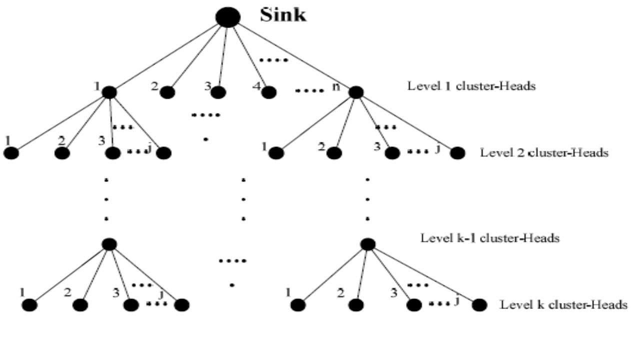

Figure 1. Multilevel clustering.

The communications within a cluster in this clustering algorithm run in rounds. Each round consists of four operational procedures, and they are the discovery, cluster head selection, cluster-head member admittance, and transmission processes.

The first three processes repeat every round and they are followed with a number of transmission operations.

First step of this algorithm is the neighbor discovery process. In the proposed multilevel clustering protocol design,
a sensor node selects a random number between 0-u to determine the number of seconds to wait before sending a HELLO broadcast message covering its maximum transmission range. This operation is to notify all neighbouring nodes regarding the existence of this specific node. During the waiting period, a sensor node listens for neighbouring nodes' discovery messages. At the end of each round, every node including the sink purges its neighbours table and begins a new neighbour discovery process. This takes care of the scenario that some nodes might have switched off due to low battery powers.

The next step is the cluster-head selections. After the neighbour discovery process, a cluster-head can select cluster heads at the next level. For example, at the Level-0, the sink picks I number of nodes from its neighbour table as the next level cluster-heads and broadcasts the message as notification.
The transmission power level of sending such a message depends on the distance estimates of selected nodes recorded
in the sink's neighbour table.

The last step of the multi-level clustering algorithm is transmission process. In a transmission cycle, each member node transmits its sensed data to its cluster-head in its allocated TDMA slot. Slot time is the time allocated to a member node by its cluster-head.

In this way the proposed design creates a highly interconnected tree graph in network, and the sensed data can
get to the sink from far away.

III. TIME SYNCHRONIZATION

Time synchronization means that bringing all the sensor nodes to a common notion of time. Synchronization in WSNs is also necessary to know the order of events that has sensed by the nodes and it is also important for data fusion.

A. Clock Synchronization:

It is the synchronization of both clock drift and clock offset. The two clocks are synchronized if they are running on the same frequency and showing similar time. Following are some terms that we are going to use in our algorithm.

1) Clock Drift: It is the difference in frequency of the clocks at which they are ticking.
2) Clock Offset: It is the difference of time between two clocks.
3) Accuracy: Accuracy of a clock is how well its time compares with global time.
4) Efficiency: The time and energy needed to achieve synchronization.

B. Packet Delay Model

Non-determinism is the major opponents of precise time synchronization in wireless sensor networks. J. Elson decomposes the packet delay into the following six components.

1) Send Time: The time required by the sender to construct the packet and deliver to the MAC layer. It is nondeterministic due to processor load.
2) Access Time: The waiting time for packets to get access to the wireless channel. It is most affecting part in any synchronization method and is also non-deterministic.
3) Transmission Time: It is the time that the sender takes to transmit the packet bit by bit at the physical layer. It depends on the length of the packet and transmission baud rate.
4) Propagation Time: The time taken by the packet to travel from the sender to receiver, on a wireless link. It is deterministic and dependents on the distance between sender and receiver.
5) Reception Time: The delay it takes for the receiver to receive the packet. It depends on the packet length and the transmission baud rate.
6) Receive Time: It is the time taken by the receiver to process the incoming packet.

IV. DESIGN OF THE ALGORITHM CTS

As shown in figure 2, the network was comprised of base station, cluster head nodes and many non-cluster – head members:

1) Network composed by n sensor nodes is uniformly distributed in a random deployment region and has formed topology structure using cluster structure routing protocol.
2) Cluster head node can directly communicate with the base station, or through the cluster head node forward indirectly related to the base station.
3) Cluster head node has the information of cluster members and can directly communicate with them.
4) Cluster head can change the transmitting power and established the first inter-cluster communication radius Ri, cluster communication within a radius of Rc, Ri> Rc.

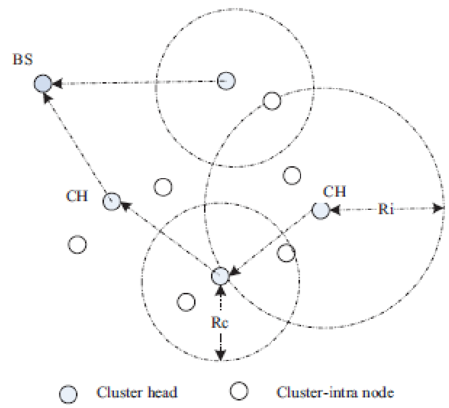

Figure 2. the network model of CTS algorithm

In the cluster-inter synchronization phase, through the construction of the base station with the cluster head node-level topology, we adopted two-way message exchange to synchronize the first cluster node with the base station. Only cluster head nodes participate this cluster-inter synchronization phase and pair-wise packet exchange mechanism is used between base station and cluster head.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

First of all, the base station starts this phase through a special radio packet, the neighbor cluster head nodes of the base station receive the packet and then response a message which contains the local time of the cluster head. The base station sends a ACK after listening this response message and appoint this node to be level one. Neighbor head nodes receive the confirmation message and set their own level to be one, then calculate the difference between cluster head node and base station using the information in the packet, finally adjust the clock to synchronization with the base station.

Every cluster head can synchronize with its previous level node, and finally every cluster head node can be synchronized with base station.

In the Cluster-intra phase, it adopted reference broadcast mechanism to finish the time synchronization between cluster heads and cluster members. Reference broadcast is not send by cluster head but one of the cluster members which elected by cluster head. Then this special node sends the reference broadcast using communication radius Ri, and cluster members within this radius can be synchronized with cluster head. After that, cluster head use communication radius Ri to send packet which contains the serial number and cluster head local time, the relevant nodes can synchronize themselves with cluster head by comparing their time. At last, it used linear least square to achieve a good precision.

V. IMPLEMENTATION OF ALGORITHM CTS

A. Cluster-inter time synchronization phase

The mission of this phase is to construct a multi-levels topology between base station and cluster head nodes , and complete the synchronization between base station and cluster head nodes.

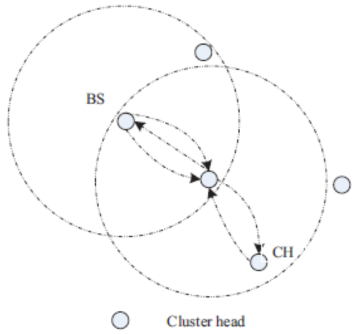

Figure 3. the algorithm principle in cluster-inter phase

As shown in figure 3, Cluster-inter time synchronization phase mainly includes the following steps:

1) The base station, as the root node of the network, starts this phase by broadcasting a packet Sync_start, which includes the level of the root node 0.

2) Neighbor cluster head node of base station (such as cluster head A) received Synn_start packet and immediately set its level to be 1.After waiting a random time, it broadcast packet Sync_req, which contains the node level and send time T1.

3) After receiving packet Sync_req , Root node broadcasts reply packet Sync_ack a short random time later, which includes T2, time of receiving packet Sync_req and T3, time of broadcasting packet Sync_ack.

4) Neighbor cluster nodes of root node(such as cluster A) records this time T4 of receiving Sync_ack. Through the following analysis, the neighbor cluster head nodes can be synchronized with the root node.

5) Neighbor cluster nodes of A(such as cluster head C) listen this packet Sync_req ,and immediately change its own level, and broadcast packet Sync_req in random time, which includes node level and send time T1.

6) Other cluster head nodes in accordance with this process continue, and at last each cluster node in the network will complete the time synchronization process.

7) As long as the cluster head node level has been set, it will no longer listening time synchronization packet.

The cluster head node can calculate the time difference after getting T1, T2, T3 and T4 and synchronize with the upper. As shown in figure 4, node S is the first i-level node, node R is the i-1 level nodes; T1 and T4 are said to be the time measurement of the local node in different time and T2 and T3 belong to node R; d is the propagation delay and δ means the time deviation of the two nodes.

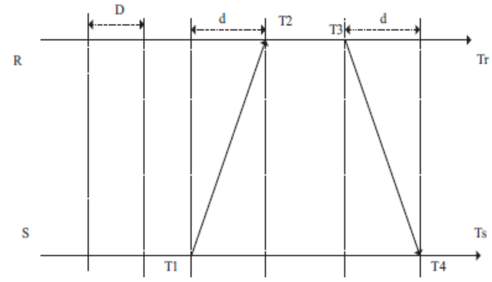

Figure 4. message exchange of node pair

Time synchronization request packet is send by node S at T1 to node R, which include the level of S and T1.This packet is received by R at T2 ,so T2= (T1+ d + δ), and then node R broadcasts reply packet to S at T3 ,which includes the level of R and T1, T2 and T3. Node S can be responded at T4, so

T4= (T3+ d + δ), so we can conclude:

$$\delta = [(t_2 - t_1) + (t_3 - t_4)] / 2$$

$$d = [(t_2 - t_1) + (t_3 - t_4)] / 2$$

After time deviation is caculated by node S , then we can synchronize node S with R.

The pseudocode of this cluster-inter synchronization phase as followed:

```
Base station :
        broadcast (Sync_start, level=0)
        if  receive ( Sync_req) then
        send ( Sync_ack , T1, T2, T3)
Neighbour cluster nodes :
          receive ( Sync_start , level)
          if ( level = null) then
        {
            level++;
            wait a short random time ;
            send ( Sync_req, level, T1) ;
            receive( Sync_ack);
            {
              record ( T1, T2, T3, T4);
              d = ( (T2 -T1) - (T4 -T3) ) / 2;
              calculate (d,  δ )
              Sync( d,  δ )
             }

         }
           else drop packet;
```


Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

B. Cluster-intra time synchronization phase

In the Cluster-intra phase, it adopts reference broadcast mechanism to achieve the time synchronization between cluster heads and cluster members. First, the cluster head node selects a cluster member act as a reference broadcast sender, and assigns a serial number to this node which uses the same transmission power with cluster head node an the communication distance is Ri. As long as the cluster head node receive s this reference broadcast packet, which maintains serial number assigned by the cluster head, it uses the transmission power which match along with communication radius Ri to broadcast a packet including the serial number and the reference-broadcast-receiving time. The other cluster members which received this serial number and receiving time can synchronize with cluster head node through comparing the serial number and the time difference when receiving the synchronization packet sending by the cluster head. The serial number matches along with that round receiving time and at the same time marks the round time. to some extent a reflection of the accuracy of Synchronization.

Selecting the reference broadcast sender is the point of achieving the cluster-intra synchronization. First of all , the worst-case scenario is considered, as showed in figure 5.

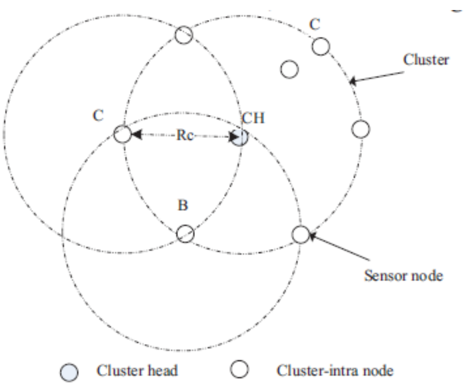

Figure 5. worst case scenario in vertical phase

Clustered Time Synchronization Algorithm

- The root node (RN) initiates the synchronization process by multicasting the Syn_start packet to level-1 CHs in the proposed algorithm.
- The Syn_start packet contains packet sending timestamp t_1.
- After receiving Syn_start packet by all the level-1 CH,the CH with CH ID = 1 responds back by sending the Syn_ack packet which contain t_1, t_2 and t_3 where t_1 is RN send time, t_2 is the CH packet receive time and t_3 is CH packet send time.
- Now RN calculate the propagation delay (d) and multicast Syn_pkt that contains (d, t) the delay and global time. When the level-1 CHs receive Syn_pkt with d and t then each CH will compute its δ (drift) and set the local clock according to the global time.
- When all the level-1 CHs synchronized then these CHs becomes RN for level-2 CHs and the same process will be repeated.
- To synchronize the sensor nodes their CH becomes RN and the same process implemented.

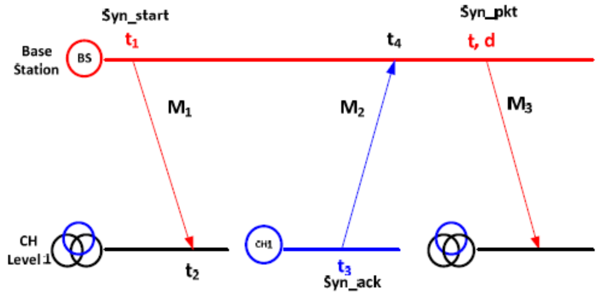

Figure 6. Message Delay Estimation.

The purpose of this algorithm is to set the logical clock of the CHs and cluster nodes with global time. The delay (d) and offset δ can be defined as

$$d = [(t_2 - t_1) + (t_3 - t_4)]/2$$
$$\delta = t + d - LocalTime$$

Where t is global time, d is the delay which will be constant for single hop communication. δ is the time deviation of the two nodes (i.e offset) and LocalTime is CHs/nodes local time.

VI. SIMULATION AND ANALYSIS

This section compares the performance of proposed algorithm with RBS and TPSN synchronization protocols.. The performance evaluation includes two parts: Energy consumption and synchronization delay. The sensors are simulated to deploy over a square sized area of l00m x l00m with adjustable communication range and fixed sensing range.

Simulation is performed using ns-2 , a discrete event network simulator. We have compared the performance of Clustered Time Synchronization (CTS) with RBS and TPSN synchronization protocols

Figure 7. illustrates the performance comparison of Clustered Time Synchronization (CTS) with RBS and TPSN in terms of energy consumption. As shown in Fig. 4, energy consumption of CTS is less than RBS and TPSN protocols in all cases thus it is energy-efficient.

The reason is clear that due to clustering the sensor nodes within the cluster have not to transmit for long distances and message exchange is also very less as compare to the RBS and TPSN that save a significant amount of energy. Another major factor that reduces the energy consumption as compare to RBS and TPSN is the reduction in message exchange. Our proposed scheme reduces the number of message exchange as compare to RBS and TPSN. In WSNs most of the energy is consumed for transmitting and receiving of messages, therefore reduction in message exchange also reduce the energy consumption.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

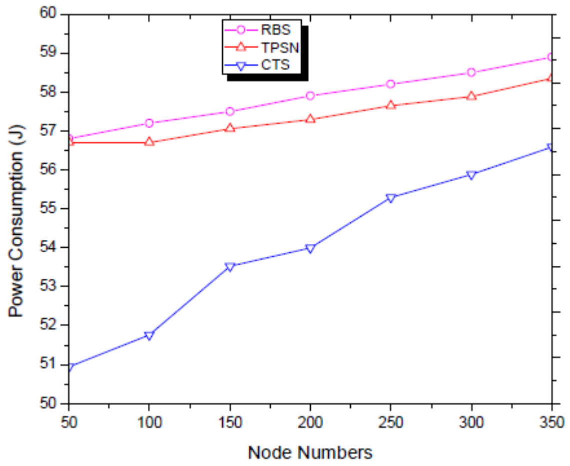

Figure 7. Power Consumption.

Figure 7. illustrates the performance of our algorithm comparing to RBS and TPSN protocols in terms of synchronization delay. As it is clear from Fig. 5 that proposed algorithm can quickly process the completion of network synchronization when the size of the network is not very large.

The above simulation results shows that clustered time synchronization is able to save the energy and provides faster synchronization speed.

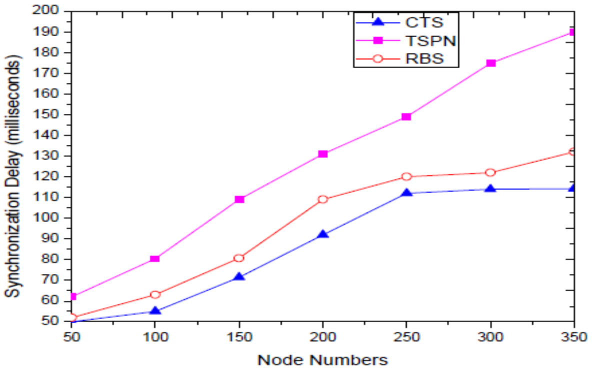

Figure 8. Synchronization Delay.

VII. CONCLUSION

A Time Synchronization algorithm based on Cluster for Wireless Sensor Networks was proposed and after the cluster formed we begin to synchronize the entire nodes of the entire network. In the Cluster-inter synchronization phase, it adopted pair-wise packet exchange mechanism t achieve the time synchronization between the Base Station and cluster heads through establishing a hierarchical topology structure. In the Cluster-intra phase, it adopted reference broadcast mechanism to finish the time synchronization between cluster heads and cluster members.The simulation results showed that the CTS algorithm, compared to RBS and TPSN algorithm, had faster convergence speed, lower power consumption and better synchronization precision.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci. Wireless Sensor Networks: A Survey. Computer Networks, Vol. 38, No. 4, pp. 393-422, March 2002.

[2] T. Haenselmann, "Sensornetworks," Wireless Sensor Network textbook, 2006.

[3] F. Sivrikaya and B. Yener, "Time Synchronization in Sensor Networks: A Survey," Network, IEEE Vol. 18, No. 4, pp. 45–50, 2004.

[4] D. Mills, "Network Time Protocol (Version 3) Specification, Implementation and Analysis," Technical Report, University of Delaware, 1992.

[5] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," Proceedings of Fifth Symposium on Operating Systems Design and Implementation (OSDI 2002), Vol. 36, pp. 147–163, 2002.

[6] K. Romer, "Time synchronization in ad hoc networks," Proceedings of ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'01), pp. 173–182, 2001.

[7] S. Ganeriwal, R. Kumar, and M. Srivastava, "Timing-Sync protocol for sensor networks," Proceedings of First International Conference on Embedded Networked Sensor Systems, Los Angeles, California, November 2003.

[8] S. Seareesavetrat, C. Pomavalai, R. Varakulsiripunth, "A light-weight fault-tolerant time synchronization for wireless sensor networks," *8th International Conference on ITS Telecommunications, 2008. ITST 2008,* pp. 182-186, 2008.

[9] Mingxia Xu, Minjian Zhao, Shiju Li, "Lightweight and energy efficient time synchronization for sensor network," *International Conference on Wireless Communications, Networking and Mobile Computing, 2005.* Vol.2, pp. 947- 950, 2005.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)