TIME SYNCHRONIZATION IN WIRELESS SENSOR NETWORKS: A REVIEW

Parminder Kaur 1, Abhilasha 2

1 Department of Computer Science and Engineering, GZS-PTU Campus, Bathinda, India
2 Department of Computer Science and Engineering, GZS-PTU Campus, Bathinda, India

ABSTRACT:
Time synchronization for wireless sensor networks (WSNs) has been studied in recent years as a significant research issue. Many applications based on WSNs assume local clocks at each sensor node need to be synchronized to a common time scale. It is particularly important for many of the tasks such as synchronizing event detection, data fusion, and coordinating wake and sleep cycle. Time synchronization requires accurate time stamping of events and coordination between the sensor nodes to reduce power consumption. Time synchronization protocols provide a mechanism to synchronize the local clocks of the nodes in the network to a global clock or relative to each other. During the last decade, many researchers have investigated the time synchronization in WSN, and proposed numerous time synchronization protocols. This paper reviews various existing protocols based on factors such as accuracy, efficiency, and complexity. The studied Synchronization protocols include conventional time sync protocols (RBS, TPSN, FTSP) and many other protocols aiming at providing common notion of time.

Keywords: Wireless Sensor Network, Time Synchronization, Clock

[1] INTRODUCTION

With the introduction of smart devices and technological advancement wireless sensor networks become an important part of human life. Wireless Sensor Networks are large-scale networks consists of tiny low cost sensors to process and monitor various ambient conditions such as temperature, humidity, vehicular movement, lightning condition, pressure, noise levels, the presence or absence of certain kinds of objects, mechanical stress levels on attached objects, and the current characteristics such as speed, direction[1]. Sensors can be deployed in dangerous, remote and hostile environments. Each sensor senses data from environment, readings from each of them is assembled together and forwarded to a central processing node or base station to get a single meaningful result. Time Synchronization in WSNs means to provide a common time scale for local clocks of nodes in the network. Time Synchronization is very important requirement for applications like data fusion, event detection, mobile object tracking, to name a few. For event detection time stamping of distributed events is needed to find out their correct order of appearance. Time Synchronization is also required for scheduling sleep and active modes among nodes, as sensor nodes have limited energy so for energy conservation, the power consumption of the nodes must be reduced. To reduce the consumption
the nodes must turn their transceiver on and off at appropriate time, so accurate timing is required between the nodes. Protocols that provide common time scale are clock synchronization protocols. Traditional synchronization protocols such as Network Time Protocol (NTP) [3] are not suitable for the wireless networks because of energy constraint of sensor nodes and also very difficult to implement in wireless network. In this paper various existing protocols are reviewed so that new protocols can be made on the basis of these protocols.

The rest of the paper is organized as follows. The time synchronization basics are given in next section, followed by message delay components in Section III. Section IV presents time synchronization parameters. Various existing protocols are presented in Section V followed by a table presenting pros and cons of protocols discussed in Section V. Final section concludes the paper.

[2] TIME SYNCHRONIZATION ISSUES

Time Synchronization means to provide common time scale to all the nodes in network. Some important terms:

1) Clock Offset: It is the difference of time between two clocks.
2) Clock Drift: It is the difference in frequency of the clocks at which they are ticking i.e. one clock is running faster than the other.
3) Clock Synchronization: It is the synchronization of both clock offset and clock drift. Two clocks are synchronized if they are running on the same frequency and showing similar time.

There are three reasons behind the nodes to be representing different times in their respective clocks [2].

- The nodes may have been started at different times.
- The quartz crystals at each of these nodes may be running at marginally different frequencies, causing the clock values to gradually diverge from each other.
- The frequency of the clocks can change differently over time because of aging or ambient conditions such as temperature (termed as the clock drift).

Three synchronization categories have been proposed by [2]

- event ordering, as the simplest form of synchronization
- synchronization to a reference node
- relative clocks.

In first method, the synchronization is on the order of messages or events. Here, clocks are not synchronized but just the order is maintained. In the second method there is a global timescale. All the nodes synchronize to this global clock. In third method, a node keeps information about its drift and offset in correspondence to neighbouring nodes. So nodes have an ability to synchronize to its neighbouring nodes.
Various reasons to know the time in certain applications of distributed systems are:

- The time of a day at which an event happened on a particular machine in the network.
- The relative ordering of events that occurs on different machines in the network.
- The time interval between two events that occurs on different machines in the network.

[3] MESSAGE DELAY COMPONENTS

The major opponent of precise time synchronization in wireless sensor networks is nondeterminism of time to deliver a message. Kopetz and Ochsenreiter[13] decompose the message delay into the following six components.

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

[4] TIME SYNCHRONIZATION PARAMETERS

- **Accuracy:** Accuracy of a clock is how well its time compares with global time. It is a measure of precision of synchronization. A protocol with high accuracy gives high precision. The requirement for synchronization accuracy may be on the order of a few micro seconds.
- **Energy Efficiency:** The time and energy needed to achieve synchronization. Lesser the energy required for synchronization more efficient will be the protocol.
- **Scalability:** The Scalability means that any synchronization technique must work well with any number of nodes in the network. The synchronization protocols must be sufficiently scalable with respect to network size.
- **Scope:** The synchronization process provides a global time-scale for all nodes in the network, or provides local synchronization among neighbor nodes. Because of the less scalability global synchronization is difficult to achieve in large sensor networks.
- **Delay:** Some sensor network applications such as emergency detection (e.g. gas leak detection) require the occurring event to be communicated immediately to the sink node. In this kind of applications, the network cannot tolerate any kind of delay when such an emergency situation is detected.
• **Robustness:** If a node in the network fails then it does not affect the working of other nodes in the network and the synchronization process. [2]

[5] **TIME SYNCHRONIZATION PROTOCOLS**

[5.1] **REFERENCE BROADCAST SYNCHRONIZATION PROTOCOL (ELSON 2002)**

Reference-Broadcast Synchronization [4] is a protocol in which one sensor node send reference beacon to their neighboring nodes to synchronize their local time. A reference node does not send any extra timestamp message instead, receiver nodes use arrival time of beacon as a point of reference for comparing their clocks. Each node records the time when reference was received according to its local clock and then receiver nodes exchange this recorded time to find clock offset and synchronize themselves.

[5.2] **TIMING-SYNC PROTOCOL FOR SENSOR NETWORKS (GANERIWAL 2003)**

TPSN [5] uses tree based network topology. The protocol works in two phases- In the first phase hierarchical structure is created in which each node is assigned a level. In the second phase pair-wise synchronization is performed along the edges in which each node gets synchronized by exchanging synchronization messages with a reference node one level higher in the hierarchy and opts to achieve global synchronization.

[5.3] **DELAY MEASUREMENT TIME SYNCHRONIZATION (PING 2003)**

In this Protocol [6] delay in time synchronization message transfer path is estimated. It is applicable for both single hop and multi-hop sensor networks. In DMTS a leader is selected as time master and broadcasts its time. All the receiving devices measure the time delay and set their time as received master time plus measured time transfer delay. As a result, all the devices that have received the time synchronization message can be synchronized with the leader. The time synchronization accuracy is depends mainly on how well the delay measurements are along the path.


The main goal of the LTS [7] protocol is to achieve reasonable accuracy while using modest computational resources (both in terms of memory space and CPU time). This protocol has two approaches. In the centralized approach the synchronization and periodic updates are generated from a reference node. In the decentralized approach, any node can start a synchronization round. Adjacent tree nodes exchange synchronization information with each other. LTS protocol builds a tree structure within the network and performs synchronization.

P. Kaur and Abhilasha
Tiny - Sync and Mini - Sync are two lightweight synchronization algorithms [8]. The tiny-sync algorithm acquires its name from the fact that it needs very limited resources, fewer resources than mini-sync. These algorithms are suitable for pair-wise time synchronization. In these algorithms, one node determines (tight) bounds of the relative offset and drift of its clock with respect to the one of the other node. 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.

FTSP [9] The idea of FTSP protocol is same as TPSN i.e. to achieve a network wide synchronization of the local clocks of nodes. The root node maintains the global time and periodically sends out a time sync message to all other nodes which synchronize their clocks to that of the root. FTSP uses single radio message time stamp to synchronize the multiple receivers. It also provides multi-hop synchronization.

LEETS [10] Protocol for Wireless Sensor Network has two phases: initial time synchronization and time synchronization maintaining, i.e., acquiring and tracking. In Initial time synchronization phase, the root node which is equipped with the GPS will synchronize with nodes which are switched on. In time synchronization maintaining phase each node in the network maintain node time by using a slot timer which count slots. Sensor nodes are equipped with hardware oscillators that can be used for the slot timer counting. Here to maintain the long time synchronization, tracking is needed periodically.

TDP [11] provides network-wide time synchronization that maintains global time synchronization within an adjustable bound. It allows the sensor network to reach an equilibrium time and maintain the equilibrium time throughout the network within a certain tolerance. TDP achieves global synchronization by multi-hop flooding: the base station initiates the protocol by sending a special timing message to the entire network. Some of the nodes, upon receiving the message, become masters by using a leader election procedure. The master nodes start the time diffusion procedure involving electing diffused leaders (similar to the master election algorithm), multi-hop flooding and iterative weighted averaging of timing from different master nodes.
[5.8] TIME SYNCHRONIZATION PROTOCOL BASED ON SPANNING TREE (LIMING 2008)

This STP Protocol [12] is based on spanning tree topology. In this protocol first a spanning tree of all the nodes in a network is created which is then divided into multiple sub-trees. Each sub-tree is a set of nodes having a father node and several child nodes. Before the whole network is synchronized to root node of tree, the sub-tree synchronization is performed. In each sub-tree synchronization the child nodes synchronizes with father node by exchanging messages and then father nodes synchronizes with root node and network-wide synchronization is achieved.

Table 1. Comparison of protocols based on parameters used in synchronization

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Sync Precision</th>
<th>Energy Efficiency</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBS</td>
<td>29.1μs</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>TPSN</td>
<td>16.9 μs</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>DMTS</td>
<td>32 μs</td>
<td>Very High</td>
<td>Low</td>
</tr>
<tr>
<td>LTS</td>
<td>Unknown</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>TINY-SYNC</td>
<td>945 μs</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>FTSP</td>
<td>1.48 μs</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>LEETS</td>
<td>30 μs</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>TDP</td>
<td>100 μs</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>STP</td>
<td>Unknown</td>
<td>Low</td>
<td>Low</td>
</tr>
</tbody>
</table>

Table 2. Pros and cons of synchronization protocol

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBS (Elson 2002)</td>
<td>It eliminates random delays on the sender side.</td>
<td>1. Additional message exchange is necessary to communicate the local time-stamps between the nodes. 2. Low transmission Range</td>
</tr>
<tr>
<td>TPSN (Ganeriwal 2003)</td>
<td>1. It eliminates the access time, byte alignment time and propagation time by making use of the implicit acknowledgments to transmit information back to the sender. 2. It has two times better precision than RBS.</td>
<td>1. It does not estimate the clock drift of nodes which limits its accuracy. 2. It does not handle dynamic topology changes 3. Use of two-way communication prohibits the use of message broadcasting which results in higher communication load.</td>
</tr>
<tr>
<td>DMTS (Ping 2003)</td>
<td>1. Low computational complexity and high energy efficiency. 2. Reduces the number of message exchanges.</td>
<td>The protocol can be applied only to low resolution and low frequency external clocks</td>
</tr>
<tr>
<td>LTS (Greunen 2003)</td>
<td>Robust and works well in the presence of dynamic links and fading.</td>
<td>The accuracy of synchronization decreases linearly in the depth of the synchronization tree</td>
</tr>
</tbody>
</table>
**Table:**

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Provisions</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tini-Sync – Mini-Sync</td>
<td>1. The protocols provide a tight, deterministic synchronization scheme with low storage and computational complexity. 2. Suitable for sensor networks that are highly constrained in bandwidth and computational power. 3. The protocols are tolerant of message losses.</td>
<td>1. No scalability and robustness. 2. High Convergence time 3. The sensor network is logically organized as a hierarchy, making it inapplicable to mobile sensor networks.</td>
</tr>
<tr>
<td>FTSP (Maroti 2004)</td>
<td>1. It is robust in flooding of synchronization messages and node failure. 2. It provides the ability for dynamic topology changes. 3. Eliminates maximum delay components.</td>
<td>1. It does not eliminates propagation delay component 2. Not scalable.</td>
</tr>
<tr>
<td>TDP (Su 2005)</td>
<td>1. Tolerance of message losses and more mobility 2. It provides synchronization even without external servers.</td>
<td>It leads to high complexity and its convergence time is also very high.</td>
</tr>
<tr>
<td>LiMing (2008)</td>
<td>Less Synchronization error than RBS.</td>
<td>Synchronization error increases as number nodes increases so not scalable.</td>
</tr>
</tbody>
</table>

[6] **CONCLUSION**

Wireless Sensor nodes have limited battery power, so most of the research work revolves around energy conservation of the nodes. However in designing sensor networks clock synchronization among nodes is also very crucial challenge. This paper presented a survey and analysis of existing clock synchronization protocols for wireless sensor networks, based on factors including precision, efficiency, complexity and also gives the pros and cons of protocols. The review paper will be a helpful reference for designers to compare and contrast their results with the protocols that are widely in use.
REFERENCES


