

A Cognitive Global Clock Synchronization Method in Wireless Sensor Networks

Bilal Ahmad, Ma Shiwei, Lin Lin, Song Yang*

School of Mechatronics Engineering and Automation, Shanghai Key Laboratory of Power Station Automation Technology, Shanghai University, Shanghai

Abstract Clock synchronization in sensor networks quite recently attained great importance and researchers are concentrating their attention on it. It is utmost difficult to get clock synchronization through clock skew and phase offset. Every sensor node has its own internal clock and it is very necessary to get all the clocks of sensor nodes precisely on common time value to achieve good synchronization. In surveillance and law enforcement applications global synchronization is important as precise mapping is required for the collected sensor data. The sensor nodes form a multi-hop and ad-hoc network for such applications. The basic demand for these applications is high accuracy and reduced latency error. The main contribution of this article is: 1. Brief introduction of fault tolerant novelty of the cognitive global clock synchronization algorithm (CGCSA) and 2. In this article we have observed jointly maximum likelihood estimation (JMLE) of the clock skew and clock offset in CGCSA to check its performance in terms of synchronization accuracy and compared our scheme with receiver only synchronization (ROS) strategy.

Keywords Sensor networks, Clock Synchronization, Clock offset, Clock Skew, Algorithm

1. Introduction

Wireless sensor networks are such a scientific tool by which we can understand the elements of the real world even from far off places. Initially it was required by armed forces but later they are used by civil applications because of their diverse nature. It is such a vast and diverse field that it is impossible for a single discipline to cover the whole field. Application experts, software engineers and control engineers altogether sum this field up completely. The application of wireless sensor networks (WSNs) is very rapidly expanding to our daily life in recent years. Since the basic operation of WSNs is data fusion, time synchronization is required for basic communication among nodes in WSNs. To fulfill this need all the nodes are brought to agree on a common clock. It does not necessarily mean that all the nodes act at once since this would become a very strict form of synchronization. Any node at any time can match its relative time with another node. Energy efficiency, robustness, cost effectiveness; life time and accuracy are all prerequisites of time synchronization. We propose that cognition is also a factor which should be required for time synchronization. It is not an easy task to get maximum benefits from sensor nodes because of certain

factors like limited energy and smaller hardware structure. The timing information of sensors is mismatched because of the drifting of internal clocks of the sensors. As a result of this situation achieving of accurate information becomes difficult and thus useful transmission cannot be had. Good time synchronization is required to achieve maximum benefits.

WSNs cannot function over a long period of time because of energy deficiency and environmental factors. Moreover, during synchronization and messages exchange between two sensor nodes some quantity of energy is also consumed. As the sensor nodes remain unattended for quite some time and their batteries cannot be changed. In order to increase the life of a sensor network, it is mandatory to reduce its energy consumption or to develop fault tolerant, simple and accurate time synchronization protocols so that to make network functional in case of node failure due to less energy at the end sensor node. Overhearing is the type of synchronization which helps to decrease the dissipation of energy but at the same time it compromise on accuracy. Receiver only synchronization scheme is described in section two.

Accurate and cognitive protocols are required to solve this issue. WSNs should remain functional even in the presence of faults, only then we can get benefits from them. We call this state fault tolerance. And if the protocol removes the faults by itself and adjusts the network we call it cognition. Cognition, in a way, is self-healing process. In the present day time there does not exist any technique

* Corresponding author:

y_song@shu.edu.cn (Song Yang)

Published online at <http://journal.sapub.org/jwnc>

Copyright © 2016 Scientific & Academic Publishing. All Rights Reserved

which can overcome all types of faults. We need accurate and low complexity protocols to preserve network functionality over a longer period of time.

Since the basic purpose of the WSNs operation is data fusion, therefore to fulfill this need all the nodes are brought to agree on a common time because this is the basic requirement of the any application of WSNs. Moreover, to agree local clocks of all sensor nodes exactly on a common time is a strict form of clock synchronization which is almost impossible to achieve. Due to the environmental factors and age of the local oscillator, local clocks of the sensor nodes may slightly drift from each other that's why they are measured as parts per million. So the rate of oscillations of two sensor nodes cannot be exactly same. Similarly due to fixed and random communication delays nodes confront difference of time also. So most often the local clocks are precisely matched. In this article we presented precise clock synchronization taking in account that local frequencies of sensor nodes are slightly different from each other and value of computed offset shared between the nodes also suffers little delay.

Cognitive global clock synchronization algorithm (CGCSA) adopts centralized synchronization mechanism. Because it is easy to control centralized synchronization schemes to achieve maximum benefits but the major drawback in centralized approaches is the failure of the master node. If we have a global time synchronization agreement between the sensor nodes which can handle faults also it provides us accurate and long term clock synchronization. As self-organization is a fundamental characteristic of the WSNs so we can direct this basic characteristic of WSNs for the cognition/fault tolerance.

The rest of the paper is organized as: Some important factors directly related to the clock synchronization and the CGCSA are discussed in section 2, previous related research work is represented in section 3, section 4 describes the description of the cognitive global clock synchronization (CGCS), measurement model of the CGCS is discussed in section 5, section 6 discusses simulation results, and finally research work is concluded in the section 7.

2. Plenaries of WSNs and Clock Synchronization Related with CGCSA

WSNs fall in the category of distributed systems as many processors working together fulfill a certain application in the network communication. But WSNs are a little different from other distributed system because their communication range is comparatively shorter and their hardware structure is smaller. As WSNs are different from other distributed system with reference to their hard ware infrastructure and application, so they are faced by different problems from those of the distributed systems. In this section we will

discuss those constraints which are a challenge in designing of the protocols in WSNs. If these challenges are handled and addressed properly, we can get a great help in increasing the performance efficiency not only in the existing protocols but also in new protocol designing.

2.1. Constraints in WSNs

Batteries are the energy source for the WSNs. If the battery of a node is discharged completely then this node does not remain functional. When nodes are operational a considerable amount of energy is consumed in data processing and communication. In the applications where energy efficiency is primary goal and accuracy can be compromised to some extent, they need resource efficient and energy efficient protocols. When one or more than one nodes stop working in a network this creates a bottleneck in fetching the data. To overcome this difficulty we need to design fault tolerant/cognitive protocols.

When sensor nodes are temporarily influenced by certain elements the communication is affected to some extent. There are some elements which completely disturb the functioning of sensor networks, these elements are temperature, humidity and chemicals. Actually these constraints affect the quality of the communication link. To solve this problem we need robust protocols.

Network density is also an important factor, effective coverage range can be short but area of coverage can be vast for which there is dense deployment of sensors and when the deployment is dense, it causes delaying factor because of multipath. Therefore effective routing protocols are required to solve this problem. Routing has two popular types (i) active routing and, (ii) proactive routing.

2.2. Coverage and Connectivity

Numbers of nodes in WSNs depend upon two factors: cost of WSNs and functionality. Coverage and connectivity are also important justifications of WSNs. Both of these come into existence because of limited communication. The deployment of sensors in different applications is of different nature. The most recent research has created the common impression that triangular form of deployment covers more geographical area than rectangular form of deployment of sensor nodes. It is because overlapping area is smaller in rectangular lattice and it needs fewer sensors. "Coverage" is such a quality of service which tells us for how long a period and in how much better way can the sensor networks be functional in the sensing field. When sensor nodes have covered any physical field then it is required to transform the sensed data into useful information by fusing that data, and to achieve it clock synchronization in sensor nodes is a necessary factor. When this data is transmitted to the sink node it is called "connectivity". To achieve useful and better coverage sensors are neither deployed much densely nor too far at long distances. The deployment of sensors can be random

as well as fixed.

2.3. Network Topology

Beside with the coverage and connectivity in the sensor network, special attention is given to the network topology also. Network topology may be star, tree, single hop or multi hop. Coverage may be sparse, dense or redundant. Connectivity may be connected or sporadic. Topology control is an extremely important issue without it nodes cannot function in any way for a longer period of time. It is quite difficult to better the performance of network without affecting its capacity and connectivity. If hop by hop short range method is applied instead of long range communication method, the possibility of the packet data loss is reduced to a considerable extent. For synchronization in mobile sensors dynamic network topology is used to make clock synchronization more robust, while in stationary sensors they are tightly coupled according to the desired application.

2.4. Clock Synchronization

Clock synchronization is an operation in which it is ensured that all physical distributed processors are functioning at a common notion of time. This function is useful for security and fault tolerance. The data of all nodes is collected in order to arrange a meaningful result. That is why clocks are synchronized at a common time. There are several types of clock synchronization; however we will discuss only those schemes which are used in or directly related with our cognitive global clock synchronization scheme.

Master-slave synchronization: In this type there is a master node which is equipped with CPU processor and all the children nodes have to set their local clocks at the local time of the master node.

Peer-peer synchronization: In this scheme every node can communication independently with any other node. In this type the possibility of the failure of master node is lessened but to control this scheme is considerably difficult. Self-organization is a basic quality of sensor networks. Through this quality it is easy to control master-slave technique and getting maximum benefits. Therefore considering these factors we have overcome the biggest danger of failure of master node resulting the communication being affected, we overcome this problem by putting cognition (inclusion of expected master node) in it.

Sender-receiver synchronization (S-R): One node sends the time stamp to another node and the other node records it sends back the acknowledgement to the first node and the message delay time is measured by calculating round trip time. Communication between master node and expected master node in CGCSA is followed by this strategy. Traditional S-R synchronization strategy adopted by [5] is shown in Figure 1.

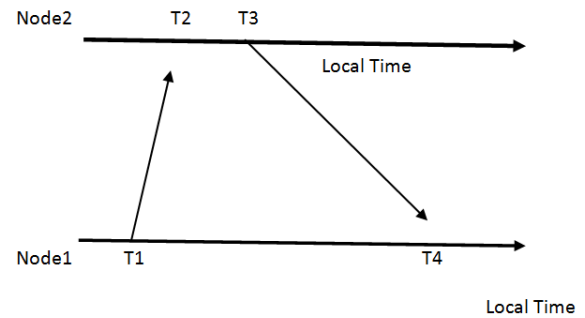


Figure 1. Sender-receiver synchronization

Receiver-receiver synchronization: In this approach when two receivers receive the same timing message, they mutually compute the difference of their reception time sent by their common sender. The major advantage of this scheme is the reduction of message delay variance. Communication between expected master node and the children nodes is followed by this strategy. The R-R synchronization scheme exploited by [4] is represented in Figure 2.

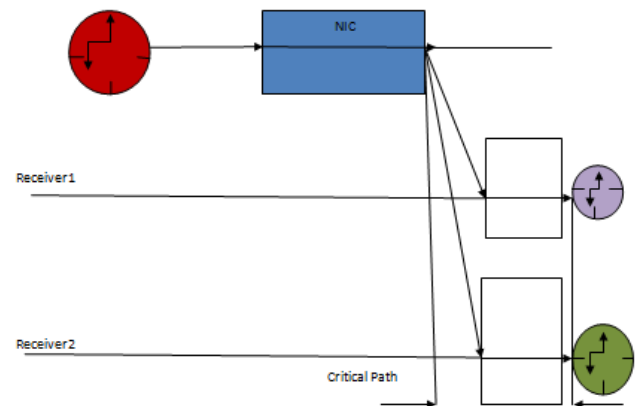


Figure 2. Traditional receiver-receiver synchronization

Receiver only synchronization: In this approach two super nodes communicate with each other for the synchronization and the third node residing nearby overhears all the messages and synchronizes itself upon reference time without communicating directly. As it can be seen from Figure 3 that node M and node N exploit pairwise synchronization and node C just overhears the message packets exchange between the nodes and synchronizes itself with the root node. Communication overhead is very low in this scheme and it saves a lot of energy but this scheme does not work efficiently for the networks dealing with large number of sensor nodes and secondly direct communication is always more accurate than the overhearing case. Another disadvantage of this scheme is that all the nodes are not active all the time. If node(s) is in idle state then it will miss the synchronization packet to overhear.

To understand clock synchronization properly it is

necessary to understand in detail the clock skew side by side with the clock offset. As the sensors are equipped with the very cheap clocks and if their internal clocks drift among themselves the synchronization period cannot sustain for a long period of time. All the processing devices are equipped with digital clocks and this hardware clock causes the time synchronization to hold or disturb. Each hardware clock has two components: quartz stabilizer oscillator and the counter. Counter is incremented upon rising or falling edge. Rate of the counter is defined as the derivative of the counter with respect to real time. The rate of an ideal clock would be unity at all time but a number of environmental factors like temperature, humidity, big age of crystal etc. cause the drifting of clocks among themselves. That is why clocks are measured in parts per million.

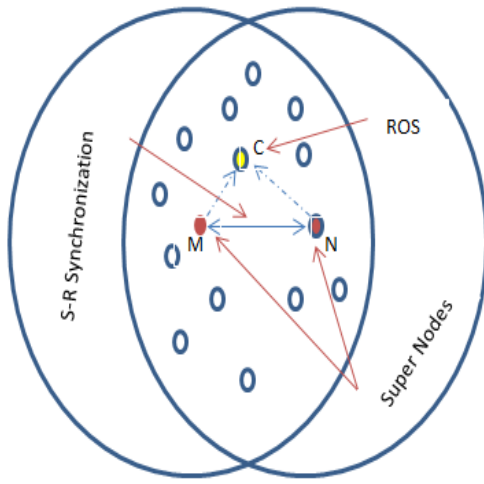


Figure 3. Receiver only synchronization

3. Literature Review

The word sensor means to perceive. Sensor is a device which converts a physical phenomenon into a signal to enable an observer to read it. Sensors are used to detect the electrical and optical signals and these act by converting the physical parameters into signals which in turn may be read electrically. Sensors are also known as a mote. A mote is always a node; however, a node is not always a mote. This is a tiny device with a small hardware and very limited memory. It runs on a minute battery. The sensor nodes gather and process the information before communicating it to the network. Sensor nodes being smaller and cheaper, and because of their various usages are rapidly making their place in the present day scientific technology like MEMS (Micro electro mechanical system). We can monitor the desired condition at any location with the help of these small sized devices. Sensors follow multi-hop communication when they are deployed at different locations.

A node's transmission range varies depending upon the protocol used in the transmission. Due to the limited resources, WSNs faces various challenges such as

self-management, power restriction, storage restrictions, security and random topology. Multiple sensor nodes combine to make a network. Some of these nodes may act as control nodes. These nodes are wirelessly connected with each other and with the base station. A node is either a sensor node or a router node in nature. Base station is a platform where all the nodes send any form of information. Sensor nodes are very densely deployed. These follow multi hop communication and function like networks. Owing to the vulnerability, their security and reliability is a big challenge. Energy consideration, scalability, data aggregation and fault tolerance are the key features of WSNs protocol designing.

A sensor network is comprised of sensor nodes, sink, sensor field and controller. Sensor nodes are the most important component of this network. They act by gathering and routing the information before transferring it to the sink. The information is transferred in very few messages leading to reduced energy consumption.

WSNs are comprised of various sensor nodes which are densely deployed, few of them are control nodes also referred as base stations. These nodes are connected with each other wirelessly and base station connects them with some other network. Sensors are deployed in ad-hoc fashion to aggregate data from observing field. Less number of sensor nodes is deployed in an ad-hoc network than a sensor network. Nodes deployment may be random or deterministic, for deterministic deployment routs are predefined while for premier case it's really very challenging. A sensor network is comprised of sensor nodes, sink, sensor field and controller. Sensor nodes are the most important component of this network. They act by gathering and routing the information before transferring it to the sink. The information is transferred in very few messages leading to the accuracy. WSNs have both advantages as well as disadvantages. Among the advantage, their implementation cost is less, the network may be deployed anywhere and they can be monitored through a global monitoring system. The disadvantages include inadequate security, difficulty in configuration and reduced speed as compared to a wired network.

To aggregate data in a sensor network, clock synchronization is an important feature. Synchronization has been studied for long time for computer systems but traditional schemes are ineffective for sensor because of their different physical properties. Because of less energy at end sensor node it's hard to synchronize nodes on common clock effectively and due to power consumption factor running the complex synchronization algorithms is difficult. In wired networks, network time protocol and global positioning system are useful methods for clock synchronization but are not suitable for wireless network because of their different characteristics. There are many factors that must be taken into account while designing a clock synchronization algorithm in sensor networks like energy efficiency, scalability, precision, robustness, life time, scope, and cost and size, thus a single protocol may

not satisfy altogether. Clock synchronization is useful for communication between nodes in the network. It is required for proximity, location and also important due to imperfections in local oscillators due to which an event is not recorded at the same time.

Wireless sensor networks have variety of applications ranging from domestic [14-16] to scientific fields [17-19], and purpose of communication is data processing [9, 10]. Looking into the needs it's required to synchronize all nodes to a global clock because the basic operation is data fusion. Sensor network applications require that sensors in the network agree on the common time. By matching the sensor location and the sensing time, the sensor network may easily predict the transmitted information. But this is not an easy task as clock skew and clock offset disturb clock synchronization a lot. Node's hardware oscillator shows clock as approximation $C_i(t)$ of real time t as

$$C_i(t) = \omega_i(t) + \phi_i \quad (1)$$

Where ω_i denotes the rate of hardware oscillator and ϕ_i denotes the time difference. Due to the imperfections and environmental changes rate of clock can be expressed as

$$1 - \rho \leq \omega \leq 1 + \rho \quad (2)$$

Sensor, often called motes are valuable tiny devices used to interact with physical world and to achieve useful information from a long distance accurately. They are often deployed in ad hoc fashion. Basic purpose is data fusion and data from each sensor is collected for a meaningful output [11-13]. Data fusion not only defines sensors location but proximity with each other also, for efficient data fusion clock synchronization is the basic requirement. Due to various limitations and bottle necks clock synchronization is hardly achieved. So far many protocols have been introduced in this field like RBS [4], TSPN [5], FTSP [6], Tiny-Sync [7], and Firefly Synchronization in Ad Hoc Networks [8]. Almost all protocols follow more or less the same timing messages exchange mechanism as followed by the pioneer protocols of the field [4] and [5], these protocols use receiver-receiver (R-R) and sender-receiver (S-R) synchronization schemes for clock synchronization respectively and are quite simple to implement.

Sensor nodes have limited resources and are prone to faults [1]. Sensor hardware and communication should be fault tolerant so that network remains functional [21]. The life time of the sensor network is increased by activation of sensors on the basis of residual energy content in [2]. A distributed algorithm is presented in [7] to overcome the faults. In [26] Li et al solved joint estimation problem by maximizing profile likelihood function in terms of clock skew. As sensors are prone to faults so fault tolerant clock synchronization protocols in sensor networks are required to be fault tolerant, so that if some node goes faulty it may be discarded or replaced by some other node.

4. The CGCSA Description

Our proposed scheme works in two phases i.e. Synchronization and cognition.

4.1. Synchronization and Cognition

Since sensor nodes are deployed in far flung areas therefore fault tolerant and accurate transmission is needed. Considering the needs we have designed CGCS protocol so that we can produce a simple, accurate and efficient time synchronization protocol. In our design the node with high energy level will be selected as the master node and if any child node fails to function our algorithm will discard it, as shown in Figure 4. And if master node or expected master node fails to function, this matter will be dealt with logically and intelligently. In other words, if the expected master node fails the master node will check its table and selects a new expected master node, and incase the master node fails, the expected master node will start functioning as the master node and will select its expected master node. The master node will form its table with the help of information provided to it by acknowledgement (ACK) messages from all the nodes.

Nodes with different residual energy may present in the sensor network. Nodes with low energy cannot contest for the master node in CGCSA. Communication between master and expected master node ensures the energy state of main controlling node(s). The likelihood is a better tool for checking the performance of fault tolerance protocols. The master node and the expected master node exchange a periodic message which helps to keep the alternative paths alive in order to maintain the network reliability in case of any faults. Excessive amount of energy and bandwidth is consumed during the master node's connection with the rest of the nodes. Therefore, like the unicast protocol, in our protocol the master node will only keep itself linked with the expected master node.

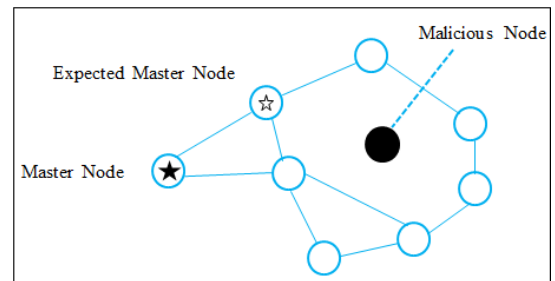


Figure 4. Synchronization and cognition

It can be seen from Figure 5 that the node which receives timing information from master node compares the clock value for adjustment, computes the offset and sends it to next hop. This mechanism reduces message overhead and in turn propagation time is also reduced. It is not essential for any individual receiver to send back the ACK to its sender except to the master node. Meanwhile master node and expected master node communicate with each other to

check their effectiveness. If node “N” is unable to receive message from master node then node “N” appears as master node and broadcasts the request message to the next hop to change and update their receiver ID. On the other hand if master node “M” is unable to receive ACK message from “N” continuously three times then it means that node “N” is down.

So the parent node checks its table and sends a request message to “C” to communicate with it as a secondary

expected master node. Each node sends an ACK message to the parent node, due to centralized system round trip time is minimized. ACK message includes ID information and energy level of the sensor nodes. Only parent node maintains the table which contains the ID information and clock time of all the nodes. CGCSA adopts S-R synchronization and R-R synchronization schemes so achieves benefits of both strategies at the same time i.e. accuracy and energy efficiency at some extent.

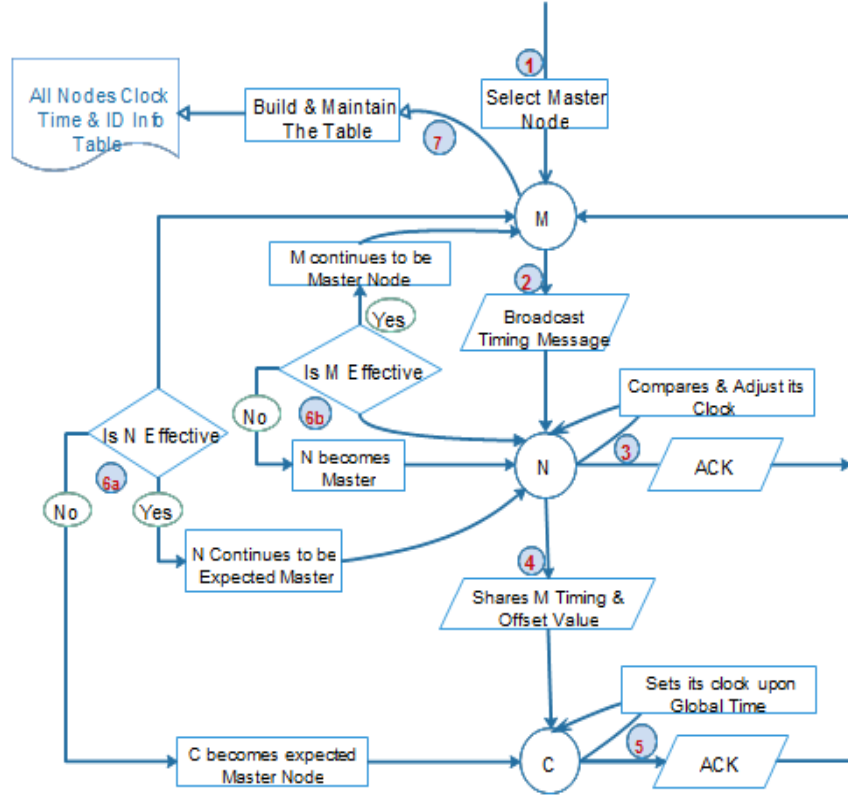


Figure 5. Block diagram of the algorithm

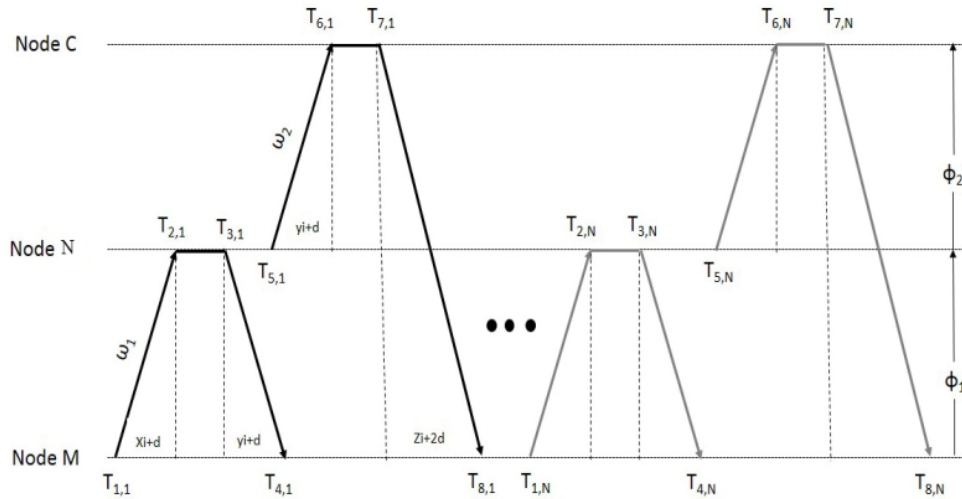


Figure 6. Clock synchronization model of the CGCSA

5. Clock Model for the CGCS

The CGCS algorithm proposed is a new approach in clock synchronization domain to deal with faults in sensor networks and to achieve accurate clock synchronization for meaningful data fusion. The algorithm not only synchronizes the sensor nodes on mutually common time but also checks for the faulty nodes and takes decisions likewise.

Due to environmental factors these tiny sensor nodes cannot survive malicious attacks in hostile environments [20, 22]. Implementation of cognitive algorithm is essential to discard malicious nodes and if master node gets faulty another node must take its place immediately so that the network remain synchronized. Taking an account on all bottlenecks like limited energy, smaller bandwidth and presence of faulty nodes, our aim turned into a novel clock synchronization algorithm which not only checks for the faults but also encounters with those faults very well. Beside of sender to receiver (S-R) synchronization and receiver to receiver (R-R) synchronization there exists another approach which is called as receiver only synchronization (ROS) [25]. In ROS a node overhears synchronization messages of two super nodes without communicating by itself, due to reduced transmissions this scheme saves energy.

Message exchange mechanism for the CGCSA is illustrated in Figure 6.

CGCSA assumes S-R and R-R approaches that exploit the timing exchange mechanism, as depicted in Fig. 3. Send time and receive time estimate round trip time (RTT) and RTT is exploited to synchronize the clocks in the network.

Figure 6 presents a number of N message exchanges between the nodes. The timestamps can be expressed as

$$T_{2,i} = \omega_1(T_{1,i} + x_i + d) + \phi_1 \quad (3)$$

$$T_{3,i} = \omega_1(T_{4,i} - x_i - d) + \phi_1 \quad (4)$$

$$T_{6,i} = \omega_2(T_{5,i} + y_i + d) + \phi_2 \quad (5)$$

$$T_{7,i} = \omega_2(T_{8,i} - z_i - 2d) + \phi_1 + \phi_2 \quad (6)$$

Where $T_{1,i}, \dots, T_{8,i}$ are the time stamps between the three nodes, ϕ_1 is the clock offset between node M and node N while ϕ_2 is the clock offset between node N and

the node C. Similarly ω_1 is the clock skews present between node M and node N and ω_2 is the clock skew existing between node N and node C. d is the fixed propagation delay and x_i, y_i, z_i are the random delays. Latency and jitter affect the performance of clock synchronization to a great extent. Fixed delays can be compensated easily but the latency changes with the passage of time. Random delays which can cause end to end latency are divided into four categories: send time, access time, propagation time and receive time.

Equations (3), (4), (5) and (6) can be written in the form of random delays, like

$$x_i = \frac{T_{2,i} - \phi_1}{\omega_1} - T_{1,i} - d \quad (7)$$

$$x_i = -\frac{T_{3,i} - \phi_1}{\omega_1} + T_{4,i} - d \quad (8)$$

$$y_i = \frac{T_{6,i} - \phi_2}{\omega_2} - T_{5,i} - d \quad (9)$$

$$z_i = -\frac{T_{7,i} - 2\phi_1}{\omega_2} + T_{8,i} - 2d \quad (10)$$

When a sensor node sends a message to another node it confronts mainly these types of delays: (a) send time, (b) access time, (c) propagation time, and (d) receive time. Sensor nodes synchronize with one another by sharing their present state. Clocks can agree on mutually common time through global positioning system (GPS) but this is an expensive solution. If GPS is used then there is no use of WSNs being cheap. Moreover, if sensors are deployed inside a building then clocks cannot be synchronized on GPS because of nonexistence of line of sight. The protocol introduced by us not only achieves synchronization but also deals with faults. Clock synchronization becomes a difficult task because of less energy at the end sensor node; as the life of a node depends on its battery life. Sensors are disposable when their energy ends they cease to function. So fault tolerant clock synchronization protocols are mandatory to achieve accurate data fusion for long time.

The random delays from equations (7), (8), (9) and (10) are subject to exponential distribution. The joint maximum likelihood function for the above equations can be written as

$$L(d, \omega_1, \omega_2, \phi_1, \phi_2) = \lambda^{-4N} \exp \left[-\sum_{i=1}^N [2x_i + y_i + z_i] \right] = \lambda^{-4N} \exp \left[-\sum_{i=1}^N \left[\left(\frac{T_{2,i} - \phi_1}{\omega_1} - T_{1,i} - d \right) + \left(-\frac{T_{3,i} - \phi_1}{\omega_1} + T_{4,i} - d \right) + \left(\frac{T_{6,i} - \phi_2}{\omega_2} - T_{5,i} - d \right) + \left(-\frac{T_{7,i} - 2\phi_1}{\omega_2} + T_{8,i} - 2d \right) \right] \right] \quad (11)$$

Taking derivative of the equation (11) w.r.t $\omega_1, \omega_2, \phi_1, \phi_2$ and for getting the value of each individual parameter all other parameters are fixed and result is set equal to the 0. After derivation we get

$$\hat{\phi}_1 = \frac{\sum_{i=1}^N [\omega'_1 (T_{2,i} + T_{3,i}) - (T_{1,i} + T_{4,i})]}{2N\hat{\omega}'_1} \quad (12)$$

$$\hat{\phi}_2 = \frac{\sum_{i=1}^N [(T_{5,i} + T_{8,i} - 2d) \times (T_{6,i}^2 + T_{7,i}^2) - (T_{6,i} + T_{7,i}) \times Q]}{\sum_{i=1}^N [(T_{6,i} + T_{7,i}) \times (T_{5,i} + T_{8,i} - d)] - 2NQ} \quad (13)$$

Where

$$Q \equiv \sum_{i=1}^N [(T_{1,i}T_{2,i} + T_{3,i}T_{4,i}) + (T_{2,i} - T_{3,i})d]$$

$$\hat{\omega}_1 = \frac{\sum_{i=1}^N (T_{2,i} + T_{3,i}) - 2N\hat{\phi}_1}{\sum_{i=1}^N (T_{3,i} + T_{4,i})} \quad (14)$$

$$\hat{\omega}_2 = \frac{\sum_{i=1}^N (T_{6,i} + T_{7,i}) - 2N\hat{\phi}_2}{\sum_{i=1}^N (T_{5,i} + T_{8,i} - d)} \quad (15)$$

MSE is a better performance index of an estimator. We purpose estimator which outperform ML estimators in terms of MSE. Numerical simulations are carried out for the clock model. The results are represented and discussed briefly in the next section.

6. Simulations and Discussions

In this section we discuss the numerical simulations which are carried out through MATLAB depending upon the measurement model to justify the performance of our proposed clock synchronization scheme. There exist three famous types of clock synchronization strategies i.e. S-R synchronization followed by TPSN [5], R-R synchronization adopted by RBS [4] and receiver only synchronization (ROS) presented in [25]. As CGCSA achieves synchronization as the hybrid of S-R and R_R synchronization so its performance is compared with the third type which is ROS scheme.

The parameters set up for the simulations are represented in Table 1.

Figure 7 and Figure 8 illustrate the mean square error (MSE) of the estimated clock offset ϕ_1 and ϕ_2 versus the number of rounds of messages exchange. Curves in both plots for the values (1, 1.10, and 1.80) for ω_1 and ω_2 clearly show that the MSE of the estimated clock offset

show decreasing pattern and convergence as long as the number of rounds of messages exchange increase and it establishes the effectiveness of proposed estimator. It can also be seen in Figure 7 and Figure 8 that MSE of the clock offset is directly proportional to the clock skew values i.e. lesser is the clock skew value, lower will be the MSE of the estimated clock offset and vice versa. MSE of the clock offset is lowest without clock skew error.

Table 1. Synchronization parameters

Parameter	Symbol	Values
Clock offset	ϕ_1 and ϕ_2	1.5-2
Clock skew	ω_1 and ω_2	1-1.80
Fixed delay	d	0.1-0.8
Number of observations	N	30-35

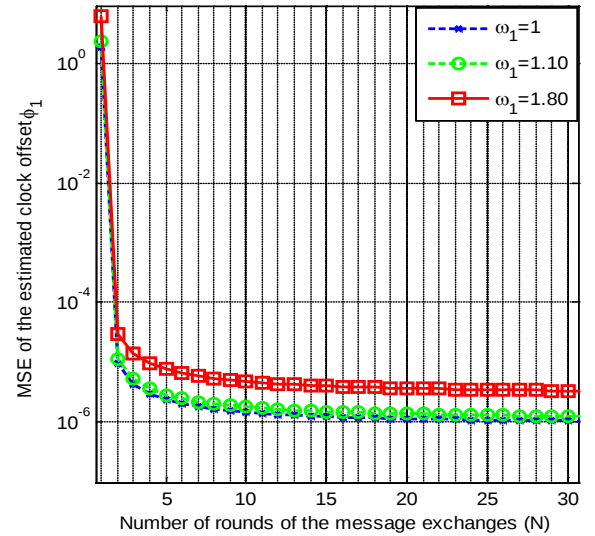


Figure 7. MSE of the estimated clock offset between node M and node N versus number of rounds of message exchanges

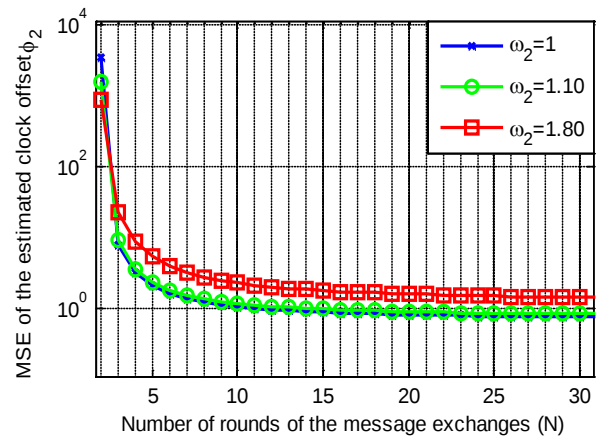


Figure 8. MSE of the estimated clock offset between node N and node C versus number of rounds of message exchanges

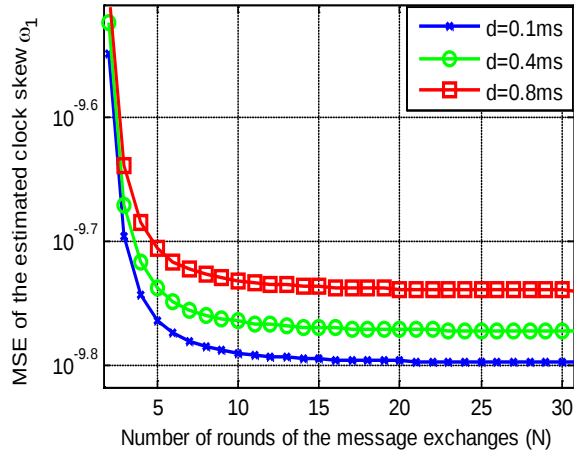


Figure 9. MSE of the estimated clock skew ω_1 versus number of rounds of message exchanges

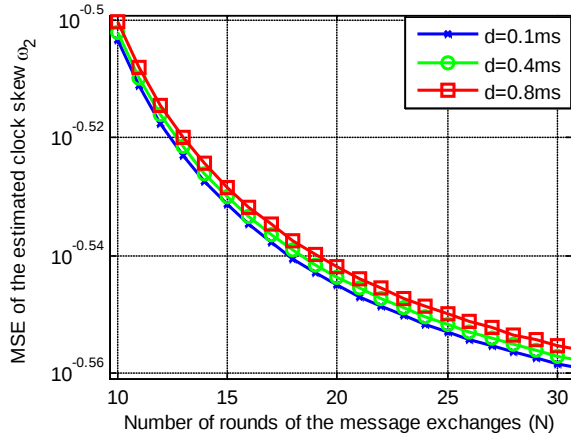


Figure 10. MSE of the estimated clock skew ω_2 versus number of rounds of message exchanges

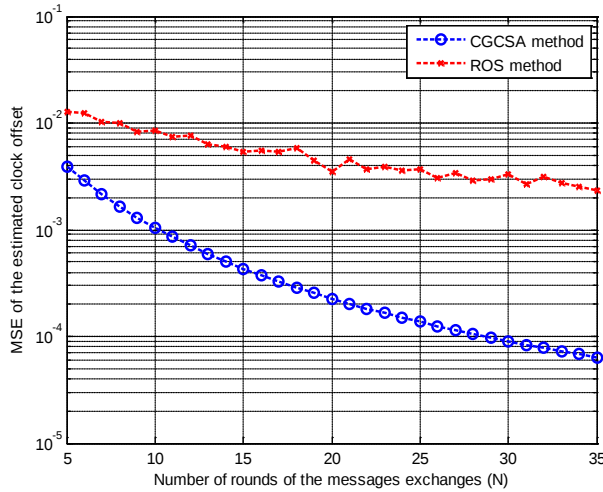


Figure 11. Comparison of the MSEs of the clock offset of proposed scheme with ROS scheme

The MSE of the estimated clock skew ω_1 and ω_2 versus the number of rounds of messages exchange is

shown in Figure 9 and Figure 10 respectively. The fixed propagation delay d is set to 0.1ms, 0.4ms and 0.8ms. We can see from both the figures that MSE of the clock skew converges with the passage of timing rounds. Distance between the nodes is directly concerned with the fixed propagation delay. Higher is the fixed propagation delay, longer is the distance and higher is the MSE of the estimated clock skew.

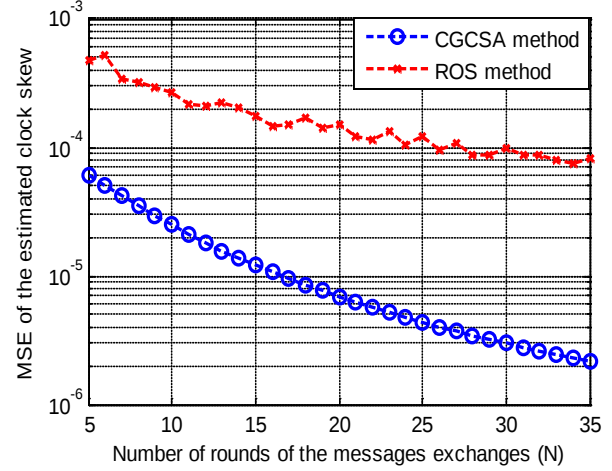


Figure 12. Comparison of the MSEs of the clock skews of proposed scheme with ROS scheme

Performance of our proposed scheme is compared with receiver only synchronization (ROS) and results are presented in Figure 11 and Figure 12 in terms of clock offset and clock skew estimation for the equal set of values. The basic reason of comparing with ROS scheme was that in ROS strategy 3rd node residing nearby overhears without communication itself, while in CGCSA the 3rd node actively takes part in communication by receiving and sending the messages. Although it increases message overhead as compared to the overhearing and consumes more energy than ROS but active participation of all nodes increases the accuracy. Simulations results show the better performance of CGCSA. From simulation results we can conclude that:

Convergence of lower bounds equal to zero reflects three aspects: (a) asymptotic efficiency, (b) lesser number of transmissions required for synchronization via CGCS and (c) accuracy.

No doubt ROS scheme is more energy efficient due to the overhearing of the third node but it compromises accuracy but our scenario is accurate because all nodes actively take part in synchronization.

This paper presented a new clock synchronization scheme in WSNs whose main idea is to share estimated value as a reference for the network wide synchronization to achieve global agreement, this reduces burden on the master node and saves on energy of the master node by not forwarding the requests from master node. Its novel fault tolerant characteristics make it robust to dynamic network topologies. From the results and the novelty of the CGCSA

we have found some characteristics which are summarized in Table 2.

Table 2. Basic characteristics of CGCSA

Deployment	Manual
Mobility	Immobile
Energy	Battery
Modality	Radio
Infrastructure	Ad-hoc
Topology	Single hop, multi hop, graph
Quality of service	Real time
Fault tolerance	Novelty of cognition
Scalability	Achievable through clustering
Accuracy	High
Application	Flexible for any wherever accuracy is desired e.g. for target tracking and law enforcement applications

Our algorithm not only detects faults in message exchanges but also replaces the faulty nodes and makes logical decisions as shown in Fig. 2. The algorithm rely on bilateral synchronization between master node and expected master node possibly with a tree topology and it follows unilateral synchronization between master node and the child node(s) based on a reference clock value shared the expected master node.

Lemma 1: Let N_{CGCS} be the required timing messages in CGCS then $N_{CGCS} = 3N + L(L-1)/2$, where L stands for the number of overall sensors in the network and N stands for the number of messages exchange between the nodes.

Proof: Since every node in the network is connected to the master node and there are $L-1$ branches in hierarchical tree. $2N$ Timing messages are required for synchronization between the master node and the expected master node, moreover expected master node shares time readings upon message received from master node to the child node to compensate relative clock offset between them. Thus $N_{CGCS} = 3N + L(L-1)/2$.

7. Conclusions

In this article useful and important feature cognition has been introduced and we study joint estimation of clock skew and offset in RTS mechanism. We investigated JMLE estimator. This paper combines the performance of RTS and a brief introduction of our fault tolerant novel algorithm. The proposed scheme is fully distributed, includes skew compensation even for the higher values of the clock skew like 1.80, computationally simple, accurate and easy to implement. It is obvious from the simulations that it reduces end to end latency. We have compared our algorithm with ROS scheme and found it more accurate. Message

complexity is low in CGCSA and as a result accuracy is high. Clustering phenomenon is required for its implementation in larger networks. Same algorithm can be used for inter clustering synchronization and intra clustering synchronization, it will reduce energy consumption.

As our future work we are focusing on energy consumption during synchronization periods and designing of the CGCSA with re-election the Ex-master node as a Master node. There are various interesting open research directions for future work. First the assumption of fixed d is same for all links is imprecise for the real sensor network. Second estimation of communication link delays will be quite interesting.

ACKNOWLEDGMENTS

This work was financially supported by Shanghai Science and Technology Foundation (13510500400).

REFERENCES

- [1] J. Lyer, H. Yu, H. Kim, E. J. Kim, K. Hawan, P. S. Mah, Assuming k-coverage in the presence of mobility and wear-out faults in wireless sensor networks, *IJSNet.*, vol. 5, no. 1, pp. 58-65, 2009.
- [2] S. He, J. Chen, D. K. Yau, H. Shao, Y. Sun, Energy efficient capture of stochastic events under periodic network coverage and coordinated sleep, *IEEE Trans. Parall. Distrib. Syst.*, vol. 23, no. 6, pp. 1090-1102, 2012.
- [3] C. Caione, D. Brunelli, L. Benini. Distributed compressive sampling for lifetime optimization in dense wireless sensor networks, *IEEE Trans. Ind. Inform.*, vol. 8, no. 1, pp. 30-40, 2012.
- [4] J. Elson, L. Girod, and D. Estrin, Fine-grained network time synchronization using reference broadcasts, *ACM SIGOPS Operating Systems Review*, vol. 36, no. SI, pp. 147-163, 2002.
- [5] S. Ganeriwal, R. Kumar, M. B. sirivastava, Timing-sync protocol for sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, ACM, pp. 138-149, 2003.
- [6] M. Maróti, B. Kusy, G. Simon, A. Ledeczi, The flooding time synchronization protocol, In *Proceedings of the 2nd international conference on embedded networked sensor systems*, ACM, pp. 39-49, 2004.
- [7] S. Yoon, C. Veerarittiphan and M. L. Sichitiu, Tiny-sync: Tight time synchronization for wireless sensor networks, *ACM Transactions on Sensor Networks*, vol. 3, no. 2, pp. 1-34, 2007.
- [8] A. Tyrrell, G. Auer, C. Bettstetter, Fireflies as role models for synchronization in ad hoc networks. In *proceedings of the 1st international conference on Bio inspired models of network, information and computing systems*, ACM, pp. 1-7, 2006.
- [9] J. Hill, M. Horton, R. Kling, The platforms enabling wireless

- sensor networks, *Communications of the ACM*, vol. 47, no. 6, pp. 41-46, 2004.
- [10] Autonomous sensing and communication in a cubic millimeter, <http://robotics.eecs.berkeley.edu/pister>, Smart Dust, 2008.
- [11] A. Dobra, M. Garofalakis, J. Gehrke, R. Rastogi, Processing complex aggregate queries over data streams, In proceedings of the ACM SIGMOD international conference on Management of data, pp. 61-72, 2002.
- [12] S. Madden, R. Szewczyk, M. J. Franklin, and D. Culler, Supporting aggregate queries over ad-hoc wireless sensor networks *Mobile Computing Systems and Applications*, In proceedings Fourth IEEE Workshop on IEEE, pp. 1-9, 2002.
- [13] A. Woo, S. Madden, R. Govindan, Networking support for query processing in sensor networks. *Communications of the ACM*, vol. 47, no. 6, pp. 47-52, 2004.
- [14] R. P. Council. Maryland Department of Transportation, Managing the Transportation System, 1983.
- [15] P. Johnson, D. Andrews, Remote continuous physiological monitoring in the home. *Journal of telemedicine and telecare*, vol. 2 no. 2, pp.107-113, 1996.
- [16] Q. Li, M. D. Rosa, and D. Rus, Distributed algorithms for guiding navigation across a sensor network. In proceedings of the 9th annual international conference on Mobile computing and networking, ACM, pp. 313-325, 2003.
- [17] K. Sohraby, D. Minoli, T. Znati, *Wireless sensor networks: technology, protocols, and applications*, John Wiley & Sons, 2007.
- [18] V. Ashok, S. Sutagundar, S. Manvi, Context aware multisensor image fusion for military sensor networks using multi-agent system, vol. 2, no. 1, pp. 147-167, 2011.
- [19] K. Shanmugavalli, and K. Fathima, *Wireless Sensor Network Based Surveillance System to Track Enemy Intrusion at Borders*, vol. 3, no. 1, pp. 138-144, 2014.
- [20] K. Sun, et al., *Secure and Resilient Time Synchronization in Wireless Sensor Networks*. Springer, pp. 347-367, 2007.
- [21] C. Zhang, *Stabilizing clock synchronization for wireless sensor networks*, University of Iowa, pp. 335-349, 2007.
- [22] J. H. Hoepman, A. Larsson, E. M. Schiller, and P. Tsigas, *Secure and self-stabilizing clock synchronization in sensor networks. Stabilization Safety and Security of Distributed Systems* Springer, pp. 340-356, 2007.
- [23] A. L. Villas, G. L. Daniel, G. Maia, W.P. Richard, J. Ueyama, A.F.L. Antonio, An energy efficient joint localization and synchronization solution for wireless sensor networks using unmanned aerial vehicle, *Wireless Networks*, pp. 1-14, 2014.
- [24] J. Lee, J. Kim, E. Serpedin, Clock offset estimation in wireless sensor networks using bootstrap bias correction, *Wireless Algorithms Systems and Applications*, Springer, pp. 322-329, 2008.
- [25] K-L Noh, E. Serpedin, and K. Qaraqe, A new approach for time synchronization in wireless sensor networks pairwise broadcast synchronization, *IEEE Transactions On wireless Communication*, vol. 7, no. 9, pp. 3318-3322.
- [26] J. Li and D. R. Jeske, Maximum likelihood estimation of clock offset and skew under exponential delays, *Appl. Stoch. Models Bus. Ind.*, 25, pp. 445-459, 2009.