Journal of Computer Technology and Software

ISSN:2998-2383

Vol. 3, No. 9, 2024

An Enhanced Time Synchronization Algorithm for Wireless Sensor Networks: CS-TPSN Combining RBS and TPSN

Rowan Beckett
Department of Computer Science and Engineering, University of Tennessee, Knoxville, USA
rowanb092@utk.edu

Abstract: Time synchronization is critical for the efficient operation of Wireless Sensor Networks (WSNs), enabling coordinated data collection and communication among distributed nodes. This paper investigates the limitations of existing time synchronization algorithms, notably the Reference Broadcast Synchronization (RBS) and Timing-sync Protocol for Sensor Networks (TPSN). While RBS minimizes synchronization error through broadcast-based methods, it suffers from high energy consumption due to frequent data packet exchanges. TPSN enhances synchronization accuracy via two-way communication but experiences increased overhead in large-scale networks. To address these limitations, we propose the Clock Synchronization Node-TPSN (CS-TPSN) algorithm, which integrates the broadcast efficiency of RBS with the bidirectional accuracy of TPSN. By introducing synchronization nodes and reducing packet exchange frequency, CS-TPSN lowers energy consumption while preserving synchronization precision. Simulation results demonstrate that CS-TPSN reduces synchronization error, minimizes network overhead, and extends network lifespan, providing a scalable and efficient solution for multi-hop WSNs. Future work will focus on addressing mobility to further enhance the algorithm's adaptability.

Keywords: Wireless Sensor Network; Wireless Communication; Time Synchronization; RBS; TPSN.

1. Introduction

Wireless Sensor Network (WSN) is a self-organizing distributed network, and time synchronization as one of the key technologies plays an important role in WSN. The internal nodes only maintain time synchronization, to cooperate to complete the corresponding task[1,18]. With the rapid development of multi-function Sensor technology and modern Network technology. WSN had been widely used in recent years. WSN is a system based on wireless communication mode. Its characteristics are distributed, multinode, multi-hop, selforganizing system. Node sensors in wireless sensor networks are miniature embedded devices. It mainly realizes the functions of data collection, processing, sending, and information request control of the monitored object. Through multinode cooperation, a wireless sensor network is formed, and the acquired information is finally delivered to the user.

At present, the commonly used wireless sensor network time synchronization algorithm can be divided into three categories[19]: Unidirectional broadcast synchronization and bidirectional paired synchronization. And bidirectional paired synchronization can be divided into sender-to-receiver one-way synchronization and sender-to-receiver two-way synchronization.

(1)Receiver-receiver time synchroni-zation algorithm

The receiver-receiver representative algorithm is the reference broadcast syn-chronization algorithm RBS (Reference Broadcast Synchronization)[2]. This algorithm has a low synchronization error. The obvious disadvantage of this algorithm is that synchronization between nodes requires the exchange of data packets containing time information[3]. The exchange of a large number of data packets increase the complexity of algorithm and results in higher energy consumption[4,20].

(2)Sender-receiver one-way time syn-chronization algorithm

The representative algorithm of one-way time synchronization between sender and receiver is the FTSP (Flood Time Synchronization Protocol)[5] pro-posed by Miklos Maroti et al in 2004. The algorithm adopts a one-way message processing mode. To reduce the influence of communication delay on synchroniza-tion precision, the communication delay is divided into two parts. Although improve the synchronization precision, reduce the packet exchange. However, due to the flood mode, the communication distance is restricted by the energy of the central node, and the energy consumption of the network with a large range is serious.

(3)Sender-receiver bidirectional time synchronization algorithm

The representative of the sender and receiver algorithm is TPSN (Timing-sync Protocol for Sensor Network)[6], a time synchronization protocol for sensor net-works proposed by Saurabh Ganeriwal et al in 2003. The algorithm adopts the two-way communication between the central node and the subnodes to exchange time information and calculate the deviation value to achieve network time synchronization. Although it has a high clock synchronization accuracy. When the node density is large in the network, the number of node synchronization hops increases significantly, which affects the synchronization accuracy and increases the energy consumption of nodes.

In addition, Kyoung-Lae Noh et al, combined RBS with TPSN and proposed PBS (Pairwise Broadcast Synchronization)[7,20] clock synchronization algorithm. When the scale of WSN increases, the synchronization precision of the network will decrease with the increase of the number of synchronous hops of nodes, and meanwhile, the energy con-sumption will increase. The classical clock synchronization algorithm cannot adapt effectively to large-scale multi-hop wireless sensor networks. Therefore, this paper combines the advantages of the TPSN and RBS algorithms. TPSN has the feature of two-way data exchange, which makes clock synchronization more accurate. RBS has the characteristics of broadcast synchronization, which reduces the frequency of synchronous packet exchange and data transmission loss[21]. Time synchronization packet exchange is carried out by introducing synchronous nodes, and synchronization is practiced by broadcasting and listening for non-synchronous nodes. This not only reduces The Times of packet exchange, but also ensures synchronization accuracy. So that proposes a time synchronization algorithm, CS-TPSN (Clock Synchroni-zation Node-TPSN), which is applied to multi-hop WSN.

The organization of this paper is as follows: We first review related work in Section 2. In Section 3, we describe the design of the RBS time synchronization protocol in more detail. In Section 4, we describe the design of the TPSN time synchronization protocol in more detail. We also provide empirical data on the implementation of two wireless algorithms. In Section 5, we present a novel algorithm CS-TPSN (Clock Synchroni-zation Node-TPSN) that can reduce net-work energy consumption while ensuring synchronization accuracy. Finally, we offer our conclusions and describe our plans for future work in Section 6.

2. Related Work

In 2002, the concept of wireless sensor networks time synchronization mechanism was first proposed at the HotNets-I International Conference, and since then, it has been paid attention to by the majority of researchers. In just a few years, the single-hop time synchronization mechanism has reached maturity, among which the representative four single-hop time synchronization protocols are the RBS algorithm, TPNS algorithm, DMTS (Delay Measurement Time Synchronization Protocol) algorithm, and FTSP (Flooding Time Synchronization Protocol) algorithm[19].

In 2019, Akbar Sabriansyah Rizqika et al, proposed Reference broadcast synchronization and time division multiple access implementation WSN[8,17]. This paper addresses the main problems that exist in wireless sensor networks, specifically the issue of collisions between data and node transmissions during the data collection process. Time division multiple access(TDMA) is used to enable each node to have a different local time with the help of a node that provides synchronization markers. Each node is thus able to transmit data in accordance with the TDMA approach already implemented, while ensuring accuracy up to microseconds.

In 2019, Nadhir Boukhechem and Nadjib Badache proposed Sensor and Actuator Networks Synchronization Protocol (SANSync)[9]. This article sets the reference clock of the wireless sensors and actuators in wireless sensor networks to the same reference clock. In this way, the executive agencies can be effectively coordinated so that they can cooperate and respond quickly to the incident. This paper proposes a cluster-based SANSync, which makes use of the large transmission capacity of the actuator to improve the accuracy of time synchronization.

Transit delay affects synchronization accuracy because transmissions of messages among nodes are required for synchronization. The information transmission delay mainly consists of the following parts: sending time, media access time, transmission time, propagation time, receiving time, and receiving processing time[10]. In addition to the main delay components listed, information transmission delay also includes interrupt wait time, encoding time, decoding time, and byte alignment time. However, these delays and propagation time delays are on the order of a microsecond. The send/receive processing time, medium access time, and transmission/receive time delay are all in the order of milliseconds. Therefore, the time of microsecond transmission delays can be ignored when designing the protocol. In order to ensure the accuracy of time synchronization, it is necessary to eliminate or compensate for the millisecond time delay.

3. RBS

RBS adopts a receiver-receiver time synchronization algorithm[11]. RBS synchronizes a set of receivers with each other, rather than a traditional protocol in which the sender synchronizes with the receiver. A node periodically sends beacon messages to its neighbors using the network's physical layer broadcast. Recipients use the arrival time of the message as a reference point to compare their clocks. The message does not contain an explicit timestamp, and the exact time it was sent is not important.

3.1 Reference Broadcast Synchronization

In this section, we illustrate the basic RBS algorithm by building nodes in a single broadcast domain.

The simplest form of RBS is the broadcast of a single pulse to two receivers, allowing them to estimate their relative phase offsets. That is:

(1)A transmitter broadcast a reference packet to two receivers (i and j).

(2)Each receiver records the time that the reference was received, according to its local clock.

(3)The receivers exchange their observations. The broadcast process is shown in Figure 1.

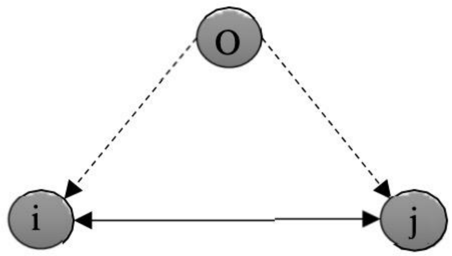

Figure 1. RBS algorithm single-hop node broadcast synchronization process

In the process of packet transmission and reception, the clock phase shift will occur. RBS corrects the clock deviation by recording the timestamp information. We assume for the moment that we already knew the clock skew and have corrected it. We can increase the precision of synchronization statistically, by sending more than one reference:

(1)A transmitter broadcast m reference packets.

(2)Each of the n receivers records the time that the reference was observed, according to its local clock.

(3)The receivers exchange their observations.

(4)Each receiver i can compute its phase offset to any other receiver j as the average of the phase offsets implied by each pulse received by both nodes i and j. That is, given n: the number of receivers. m: the number of reference broadcasts, and Tr,b: r's clock when it received broadcast b.

$$\forall i \in, j: \mathrm{Offset}[i, j] = \frac{1}{m}\sum_{k=1}^{m} T_{j,k} - T_{i,k}$$

In the process of packet transmission and reception, the clock phase shift will occur. RBS corrects the clock deviation by recording the timestamp information[12]. In the deviation calculation experiment, n nodes are given random clock offsets. M pulse transmission times are randomly selected. Each pulse is "transmitted" to each receiver by using a timestamp receiver's clock. The calculated offset matrix is shown in equation 1. Each of these O(n^2) computed offsets was then compared with the "real" offset; the maximum difference was considered to be the group dispersion.

3.2 Estimation of Clock Skew

The important characteristics of a crystal oscillator are accuracy and stability. Complex disciplines exist that can lock an oscillator's phase and frequency to an external standard. However, we can select a very simple yet effective algorithm to correct skew. Instead of averaging the phase offsets from multiple observations, we perform a least-squares linear regression. This offers a fast, closed-form method for finding the best-fit line through the phase error observations over time. The frequency and phase of the local node's clock with respect to the remote node can be recovered from the slope and intercept of the line. The linear regression model is defined as Eq.2.

$$f(x; m, c) = mx + c$$

In the linear regression model, X represents the calculated delay offset after receiving the time packet, and Y represents the predicted delay offset. m is the slope and c is the intercept. By predicting the minimum offset by the least square method, the average loss function can be minimized and the optimal solution can be obtained. The average loss function is defined as Eq.3.

$$\begin{aligned} L &= \frac{1}{N}\sum_{n=1}^{N} L_n\big(y_n, f(x_n; c, m)\big) \\ &= \frac{1}{N}\sum_{n=1}^{N}\big(y_n - f(x_n; c, m)\big)^2 \\ &= \frac{1}{N}\sum_{n=1}^{N}\big(y_n - (c + mx_n)\big)^2 \\ &= \frac{1}{N}\sum_{n=1}^{N}(y_n - c - mx_n)(y_n - c - mx_n) \\ &= \frac{1}{N}\sum_{n=1}^{N}(y_n^2 - 2y_n c - 2y_n m x_n + c^2 + 2cmx_n + m^2x_n^2) \\ &= \frac{1}{N}\textstyle\sum_{n=1}^{N}(y_n^2 - 2y_n c + 2mx_n(c - y_n) + c^2 + m^2x_n^2) \end{aligned}$$

4. TPSN

TPSN is a Sender-Receiver-based time synchronization protocol for WSN. It has two main steps to synchronizing the network as pair-wise synchronization and network wide synchronization[13,18].

4.1 Pair-wise Synchronization

The algorithm is similar to the NTP time synchronization protocol running in the traditional network [14,16]. There is a core precise time clock source node, which is responsible for the time synchronization of all nodes in the network. TPSN protocol requires that the identity ID of all nodes is unique throughout the network, and nodes can be identified. At the same time, there is a bidirectional reachable wireless link between two nodes. The protocol mainly classifies nodes according to the idea of hierarchy. Each node only needs to synchronize with the node of its upper layer to complete the synchronization of all nodes with the clock source node.

Let's have 2 nodes A and B which will have been synchronized and node A starts the synchronization. The synchronization process is shown in Figure 2.

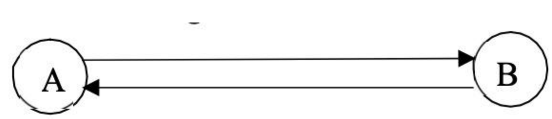

Figure 2. TPSN algorithm pair-wise synchronization process

Here are the steps to synchronize A to node B:

Node A prepares a time synchronization packet and sends it to node B to start time synchronization, and records the sending time T1.

When node B receives the time synchronization packet of node A, the receiving time is immediately recorded as T2.

After node B waits for a certain period of time, the sending time is synchronized with the reply packet to Node A and the sending time is recorded at T3.

Node A receives the time synchronization reply packet of node B, records the receiving time as T4, and calculates the time deviation through the recorded timestamp information. The specific process of data packet exchange of timestamp information is shown in Figure 3.

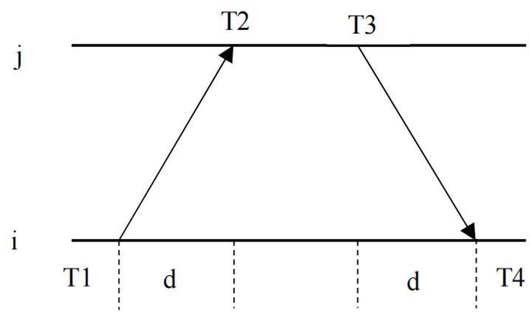

Figure 3. TPSN protocol node’s two-way information exchange process

4.2 Network-wide Synchronization

TPSN time synchronization protocol requires a clock source that provides an absolute time and is responsible for time synchronization across the network. The hierarchical network structure is adopted to divide the nodes into different levels, and the lower-level node only synchronizes the communication completion time with the corresponding upper-level node.

The execution steps of the algorithm are divided into two stages: the hierarchical discovery stage and the time synchronization stage.

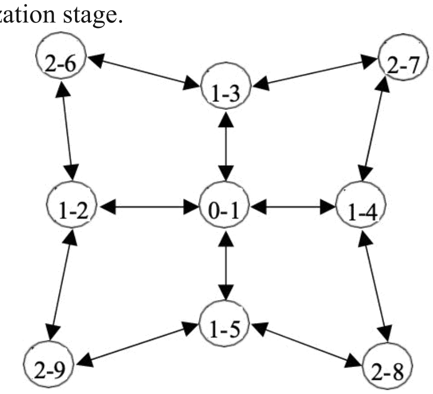

Figure 4. TPSN network topology

First, is the hierarchical discovery stage. The main work of this stage is to layer the network. TPSN builds a spanning tree where each node is given a unique ID number. Node ID numbers are recorded to maintain the parent-child node relationship in the layering process. Each node knows the hierarchy between itself and its parent node. Set the clock source node as the root node of layer 0, which is responsible for the beginning of the network hierarchy partition. The root node sends a layered packet containing hierarchical information of its own layer 0 and the node ID. When all the one-hop neighbor nodes receive the packet, the one-hop node sets its own level to 1 and its parent level to 0, and records the parent node’s ID information. Hierarchical packets are sent after the completion of the one-hop node. At this time, the node that receives the packet of layer 1 and is not layered sets its own level to 2 and its parent level to 1. Meanwhile, record the parent node ID information, and so on to complete the whole network hierarchy. The 2-hop TPSN network topology is shown in Figure 4.

Second, is the time synchronization phase. The main work of this stage is to synchronize the network node time. When the network level discovery stage is completed, the reference node sends the start packet of time synchronization, the start packet of network time synchronization, the start packet of time synchronization, and the start packet of time synchronization is sent to the whole network. After receiving the initial synchronization packet, the node of layer 1 sends the time synchronization request to the root node, which sends the synchronization reply packet after receiving it, and the node of layer 1 completes the time synchronization in turn. After the synchronization of all nodes of the first layer is completed, and the nodes of the second layer send the time synchronization request to the layer is complete, the nodes of the second layer send the time synchronization request to the nodes of the first layer, the nodes of the first layer send the synchronization reply packet after receiving it, and the nodes of the second layer complete the time synchronization accordingly, and so on to complete the time synchronization of the nodes of the whole network. The specific synchronization process is the same as described in Chapter 3.1.

5. CS-TPSN Improved Algorithm

How to reduce the times of packet exchange o TPSN protocol and reduce the cost of the protocol is the main direction of research on the optimization algorithms. In the TPSN algorithm, nodes communicate with each other more times. Each node needs to exchange packets with its parent node at least 4 times during a time synchronization. In order to solve the problem of too many times of packet exchanges, the synchronization mode of the synchronization node is proposed in this paper to reduce the times of packet exchange.

5.1 Optimization Design of One-hop Communication

Taking four nodes within a single hop as an example, the packet exchange process of the TPSN algorithm is shown in Figure 5.

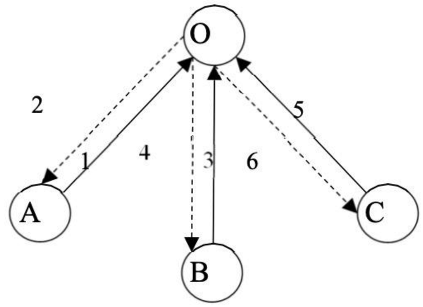

Figure 5. TPSN algorithm single hop topology structure diagram

O is the root node, while A, B, and C are all leaf nodes. Nodes synchronize time with node O in order of A, B, and C. The realization represents the time synchronization request sent by the leaf node, and the dashed line represents the feedback from the root node. It can be seen that the number of neutron nodes in the figure is 3, and the number of time-synchronized packet exchange is 6. Thus, when the number of child nodes is n, the number of packet exchanges is 2n.

According to the ideas of the RBS algorithm and TPSN algorithm, B and C nodes are all within the single-hop communication range of A and O nodes. Because the wireless channel has broadcast characteristics, the packet exchange between nodes A and O can be monitored by nodes B and C. The packet exchange topology diagram of the optimization algorithm CS-TPSN proposed is shown in Figure 6.

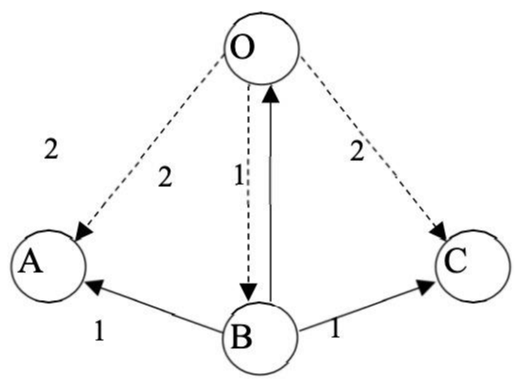

Figure 6. CS-TPSN algorithm single hop topology structure diagram

As can be seen from figure6, the number of synchronous packet exchanges at the one-hop node time is 2. Even as the number of nodes increases, a small number of packet exchanges can be guaranteed as long as they are within broadcast range. By using the broadcast listening feature, the packet switching times are reduced and the network overhead is reduced.

5.2 Optimization Design of Communication between Networks

The algorithm design is optimized through the simplification of network topology, and the design is aimed at a time synchronization algorithm. The simplified topology is shown in Figure 7.

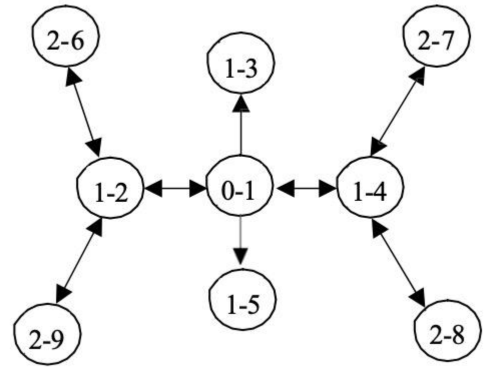

Figure 7. CS-TPSN network topology

The algorithm reduces the time of exchanging time information packets with parent and child nodes by using a broadcast monitoring function. The design of a lower-level node selects an upper-level node as the parent node, a lower-level node corresponds to a parent node, and the parent node corresponds to one or more child nodes. The parent node is called the synchronous node in the whole network. By choosing a synchronous node to communicate with the root node and other child nodes to listen, the energy consumption is reduced and the network life is prolonged.

Network topology initialization. The reference node of the network clock source initiates periodic topology updates. The data contains information such as the reference node ID, its own hierarchy, and the parent node. The node receiving the broadcast message saves the received information, sets the sending data node as the parent node, and continues to send packets containing the same information, thus completing the entire network initialization.

Calculate the two-hop node. After network initialization is completed, each node will save its own child node information. Then the node sends the child node information to the parent node. At this time, the parent node can obtain the two-hop node information by comparing it with the child node information saved by the parent node.

Compute synchronous nodes. For each synchronization node, the greedy heuristic algorithm is used. First of all, the algorithm converts with independent nodes in priority. If only one node carries out data forwarding between the two-hop node and the root node, then this intermediate node is selected as the synchronous node. Then, after the independent node coverage is completed, it will judge whether it is selected as a synchronous node according to the number of two-hop nodes covered by child nodes. Priority is given to those with a large number of two-hop nodes to complete full coverage and synchronous node selection.

List of synchronization modes. After the upper node calculate the broadcast information of the synchronous node, the lower node will generate a list of synchronous modes based on this information. When the upper node sends synchronization information, the lower node receives the list and compares it with its own ID. If it is in the list, the upper node is recorded in its own synchronization node list, and a time synchronization protocol is adopted for synchronization. If it is not in the synchronization list, it will record its synchronization list as empty and synchronize with the broadcast mode.

5.3 Simulation and Modeling of Time Synchronization Algorithm

This paper is based on OPNET modeling and analysis. OPNET is a professional network environment modeling and simulation software package. It can accurately simulate and analyze all kinds of complex network performance, and network equipment manufacturers and communication operators have a wide range of applications.

CS-TPSN algorithm modeling mainly includes process hierarchy design, node model and package structure design, network structure design and so on. Process hierarchy design is the core of modeling and simulation work. The specific functions of the CS-TPSN algorithm will be realized in the process model design.

Firstly, the node model of the CS-TPSN algorithm is designed. The node consists of a transmitter, a receiver, and a processor. The node model is shown in Figure 8.

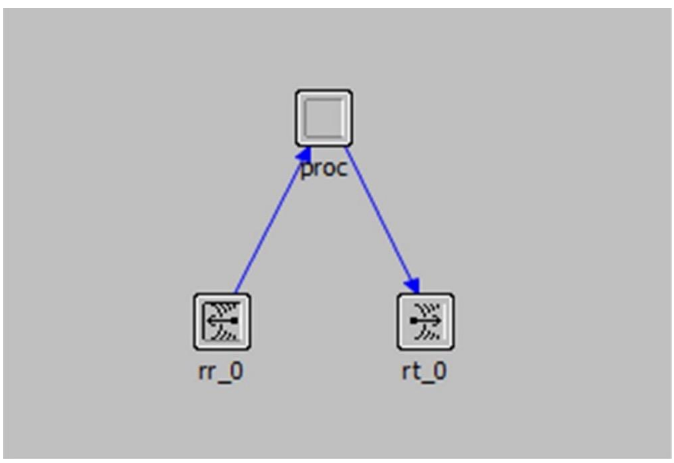

Figure 8. CS-TPSN algorithm node model

The algorithm node is divided into the Time node and the Common node. The Time node is responsible for providing the benchmark clock and is defined as the root node of layer 0. All the other nodes except the Time node are Common nodes. Both the Time node and Common node adopt the same process model, but they perform different functions after judging the node type through process function module design. The process model is shown in Figure 9.

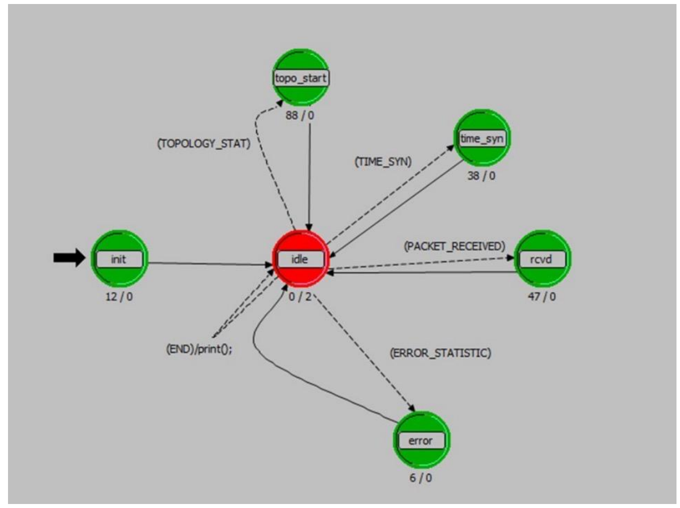

Figure 9. CS-TPSN algorithm process model

5.4 Analysis of CS-TPSN Simulation Results

We use OPNET simulation software to verify CS-TPSN algorithm performance. The performance characteristic of the CS-TPSN algorithm is described mainly from two aspects: (1) time synchronization precision; (2) the communication packet cost [15]. The time synchronization method involved in the comparative experiment includes: (1) the TPSN algorithm; (2) the RBS algorithm. And compare the other improved algorithms, showing that the result is better than it.

5.4.1 Time Synchronization Error Analysis

The algorithm is synchronized once every 5min, after complete network topology synchronization, and the average time error after each time synchronization is calculated.

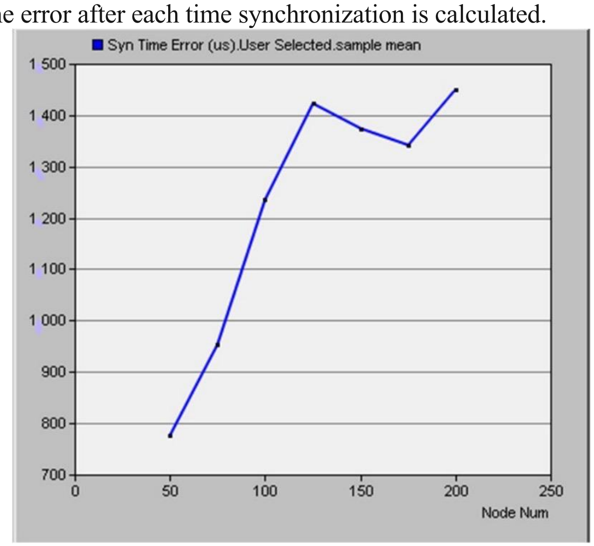

Figure 10. Multi-hop time synchronization error of CS-TPSN algorithm

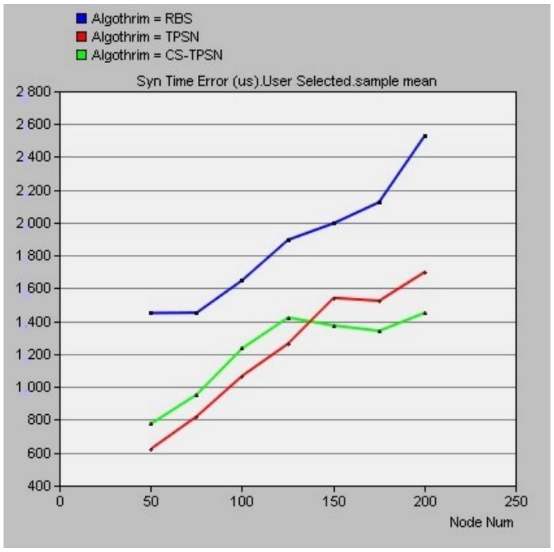

Figure 11. An error simulation analysis of CS-TPSN algorithm and classical time synchronization algorithm

The simulation is set at 50 to 200 nodes, with 25 nodes as an interval. The synchronization error of the algorithm is tested separately. After the synchronization of the whole network node is completed, the error between the time of the reference node and the time of the reference node is averaged as the error calculation. The result is shown in Figure 10.

The error of the CS-TPSN algorithm, traditional RBS algorithm, and TPSN algorithm with the number of nodes increasing is shown in Figure 11.

6. Conclusion

This paper mainly analyzes the characteristics of the RBS and TPSN algorithms, describes their respective communication process and packet exchange mode in detail, add points out the problems of the two algorithms. Combining the advantages of the two algorithms, the improved algorithm CS- TPSN is proposed by changing the topology. The simulation results show that the improved algorithm can reduce the synchronization error, reduce the network overhead and extend the network life. However, in the algorithm design, the mobile node is not fully considered, which is also the direction of future research and improvement.

References

[1] Y.W. Du, N. Xu, H.L Li, W. Li, K. Feng. A hierarchical time synchronization algorithm for WSN[J], Procedia Computer Science, (2018), p. 131.

[2] Jeremy E, Lewis G, Deborah E. Fine-grained network time synchronization using reference broadcasts[J], ACM SIGOPS Operating Systems Review, (2002), p. 36(SI).

[3] Trong-Hop D, Myungsik Y. Continuous Reference Broadcast Synchronization with Packet Loss Tolerance[J], Wireless Personal Communications, Vol. 86 (2016) No. 4.

[4] Y.W. Du, N. Xu, H.L. Li, et al. A hierarchical time synchronization algorithm for WSN[J], Procedia Computer Science, (2018), p. 131.

[5] Miklós Maróti, Branislav Kusy, Gyula Simon, Ákos Lédeczi. The flooding time synchronization protocol[P], Embedded networked sensor systems, (2004).

[6] Saurabh Ganeriwal, Ram Kumar, Mani B. Srivastava. Timing-sync protocol for sensor networks[P], Embedded networked sensor systems, (2003).

[7] Noh K L, Serpedin E, Qaraqe K. A New Approach for Time Synchronization in Wireless Sensor Networks: Pairwise Broadcast Synchronization[J], IEEE Transactions on Wireless Communications, Vol. 7 (2008) No.9, p. 3318-3322.

[8] Akbar Sabriansyah Rizqika, Ichsan Mochammad Hannats Hanafi, Darmawan Aulia Arif. Reference broadcast synchronization and time division multiple access implementation on WSN[J], TELKOMNIKA (Telecommunication Computing Electronics and Control), Vol. 17 (2019) No. 1.

[9] Nadhir Boukhechem, Nadjib Badache. SANSync: An Accurate Time Synchronization Protocol for Wireless Sensor and Actuator Networks[J], Wireless Personal Communications, Vol. 105 (2019) No. 3.

[10] Shi Kyu Bae. Sender-receiver Reference Broadcast Synchronization[J], Life Science Journal, Vol. 11 (2014) No. 7.

[11] Ki-Hyeon KIM, Won-Kee HONG, Hie-Cheol KIM. Low-Cost Time Synchronization Protocol for Wireless Sensor Network[J], IEICE Transactions on Communications, Vol. B (2009) No. 4.

[12] W. Ping, S.W. Xu. Research of Dynamic Clustering Time Synchronization Algorithm in Wireless Sensor Networks[A]. Intelligent Information Technology Application Association. Future Communication, Computing, Control and Management (ICF4C 2011 LNEE 141) [C], Intelligent Information Technology Application Association, Vol. 7 (2011).

[13] Niranjan Panigrahi, Pabitra Mohan Khilar. An evolutionary based topological optimization strategy for consensus-based clock synchronization protocols in wireless sensor network[J], Swarm and Evolutionary Computation, Vol. 22 (2015).

[14] Ranjan, Rajeev, Varma, Shirshu. Collision-Free Time Synchronization for Multi-Hop Wireless Sensor Networks[J], Journal of Computational Intelligence and Electronic Systems, Vol. 1 (2012) No. 2.

[15] J.Q. Xu et al. Design & Simulation of WSN Equal-cluster-based Multi-hop Routing Algorithm. Journal of System Simulation, Vol. 23 (2011) No. 05, p. 992-997.

[16] G.S.S. Chalapathi, Vinay Chamola, S. Gurunarayanan. A testbed validated simple time synchronization protocol for clustered wireless sensor networks for IoT[J], Journal of Intelligent & Fuzzy Systems, Vol. 36 (2019) No. 5.

[17] X.H. Wei, K.X. Cheng. Time Synchronization of Distributed Systems based on IEEE1588[J], World Scientific Research Journal, Vol. 7 (2021) No. 3.

[18] L.ZH. Cui, J. Cao, Z.L. An, et al. Research on time synchronization of linear pulse-coupled oscillators model with delay in nearest neighbor wireless multi-hop networks[J], International Journal of Distributed Sensor Networks, Vol.17 (2021) No. 7.

[19] F.L. Shan, J.G. Yu, Z.F. Wang, et al. Research and Design of Wireless Network Time Synchronization Module Based on IEEE1588 Protocol[A]. Asia Pacific Institute of Science and Engineering. Proceedings of 4th International Conference on Data Mining, Communications and Information Technology (DMCIT 2020) [C]. Asia Pacific Institute of Science and Engineering: Chengdu Shylock Education Consulting Co., LTD, (2020).

[20] X.B. Sun, et al. Design and development of a wireless sensor network time synchronization system for photovoltaic module monitoring. International Journal of Distributed Sensor Networks. Vol. 16 (2020) No. 5.

[21] Dur-e-Shawar Agha, Fozia Hanif Khan, Rehan Shams et al. A Secure Crypto Base Authentication and Communication Suite in Wireless Body Area Network (WBAN) for IoT Applications[J]. Wireless Personal Communications: An International Journal, Vol. 103 (2018) No. 6.