An Improved Time Synchronization Algorithm for Wireless Sensor Networks

Ji-hong Sun
Luohe Vocational College of Food, Luohe, China
jjuf5931656@126.com

Huanzheng Shao
Luohe Vocational College of Food, Luohe, China
nbmyt5577@126.com

Abstract—Wireless sensor network (WSN) has been paid more and more attention to by the international academic and industrial fields. What's more, it has become a hot research topic with great attention and expectations. There are a lot of technical problems involved in wireless sensor networks, among which, time synchronization technology is one of the supporting technologies in wireless sensor networks. And it is rather essential for the application of wireless sensor network. Two classical time synchronization algorithms are introduced and their advantages and disadvantages are compared. Based on the existing algorithms, an improved time synchronization algorithm is put forward. The improved algorithm was simulated by NS2. The results showed that the proposed algorithm had better time synchronization accuracy and lower network cost than RBS and TPSN algorithm. In summary, the improved time synchronization algorithm has a rather good performance.

Keywords—Wireless sensor, time synchronization algorithm, TPSN algorithm

1   Introduction

Wireless sensor network (WSN) is a distributed system, and the problem of time synchronization is an integral part that cannot be ignored of all distributed systems [1]. In wireless sensor networks, since that there is the clock skew between different nodes, it is necessary to use time synchronization techniques to provide consistent time standards for different nodes. In wireless sensor networks, each node has its own local clock, and Internet users often need to know the specific time of an event, so when it is necessary to provide the time of events, all the clocks of the nodes have to be consistent. Wireless sensor networks need to make a real-time information collection for the monitoring of the target, and most of sensors can only monitor a feature of the objects. Users cannot fully grasp other information of the target, such as direction, location, movement speed and so on. It it is expected to obtain more comprehensive information, it is necessary to make a collaborative processing or data integration of

the data collected by the other sensor nodes. Data fusion and collaborative processing often require that the data collected by different nodes is uniform in time, so time synchronization is essential [2-3].

In this paper, we introduced two classical time synchronization algorithms and compared their advantages and disadvantages. The improved algorithm was simulated by NS2. The results showed that the proposed algorithm had better time synchronization accuracy and lower network cost than RBS and TPSN algorithm.

2   State of the art

In August 2002, J. Elson and K.Romer first presented a wireless sensor network at the Hot Nets-I International Conference. J.Elson and K.Romer first described the importance of time synchronization to wireless sensor networks, and proposed RBS (Reference-Broadcast Synchronization) algorithm. The algorithm is intended to provide a synchronization algorithm based on the receiver-receiver mechanism. In 2011, Luca Schenato et al. proposed the ATS (Average Time Sync) algorithm, which used two cascaded consistency algorithms to adjust the compensation parameters. The average node virtual clock has a consistent settling time. In 2012, Michael Kevin Maggs proposed CCS (Consensus Clock Synchronization) algorithm. The algorithm uses the averaging algorithm to compensate for the offset of the node time. In 2013, Zhao Dengchang et al. proposed TSMA (Time Synchronization using Max and Average Consensus) algorithm. Based on the ATS algorithm and the CCS algorithm, the algorithm uses the maximum coincidence algorithm to achieve the compensation of the drift, and obtains the faster drift rate. At the same time, TSMA algorithm uses MAC layer timestamp to get higher synchronization accuracy. In 2015, through the study of RBS time synchronization algorithm (RBS) for wireless sensor networks, such as time synchronization problem of multi-hop network, ZhouYa et al. proposed RBS ring algorithm based on ring network topology (References Broadcast Bing Synchronization, RBRS). On the basis of RBS, the algorithm adopts the method of broadcast packet and least squares linear regression to realize the whole network time synchronization. The algorithm is compared with the existing RBS optimization algorithm from the aspects of synchronization error and overhead. It is found that the algorithm has the advantages of small accumulation of errors and a significant reduction in synchronization overhead compared with the initial algorithm. In addition, it can realize the time synchronization of the whole network. In 2016, Yang Zhiqiu et al. proposed a low-power time synchronization algorithm and fault-tolerant time synchronization algorithm based on the traditional time synchronization algorithm. By setting up the time synchronization test platform, the low power time synchronization algorithm is validated from the aspects of algorithm precision, correctness and robustness. It is shown that the low-power time synchronization algorithm of cluster structure is more stable than traditional algorithm.

In summary, the research on the time synchronization of wireless sensor networks is mainly focused on two aspects: (1) To minimize the dependence of the synchronization algorithm on the time server and channel quality, thereby shortening the "criti-

cal path" that may cause synchronization errors. (2) To study energy efficient synchronization algorithm from the perspective of energy consumption. Various algorithms are designed for a particular environment. Although a number of good time synchronization algorithms are proposed, they are designed to improve the performance and ignore some other performance. There is a gap with the practical application. It needs to further improve the original algorithm or design a new algorithm, so that the network has better scalability and stronger network topology to adapt. In this paper, two classical time synchronization algorithms are introduced and their advantages and disadvantages are compared. Based on the existing algorithms, an improved time synchronization algorithm is put forward. At last, the improved algorithm was simulated by NS2.

3 Typical time synchronization algorithm

3.1 RBS time synchronization algorithm

RBS (Reference broadcast synchronization) algorithm is a classical algorithm of time synchronization. It is not complicated in the realization method, and the requirements on the storage space is not harsh, so it can meet most of the precision requirements on time synchronization and the application requirements are not strict.

The RBS synchronization algorithm requires that the message be exchanged many times. Whether between the sending node and the receiving node or between the receiving node and the receiving node, it needs to the repeated message exchange, so as to achieve time synchronization [4]. However, the disadvantages of this method are obvious, and the repeated broadcast messages and messages exchange between different nodes will affect the synchronization accuracy and increase the network overhead. RBS protocol principle diagram is as follows. The sending node first of all broadcasts a reference packet, and there are two nodes that can receive the group data within the radio range. Each node records the time for receiving this packet, and of course, this time is to record according to the node's local clock. Then, two nodes exchange their time recorded, and the time difference is the difference between the receiving time. One of the nodes, according to the time difference, modifies its local clock, and in this way, the two receiving nodes get time synchronization. Figure 1 is the schematic diagram of the RBS algorithm.
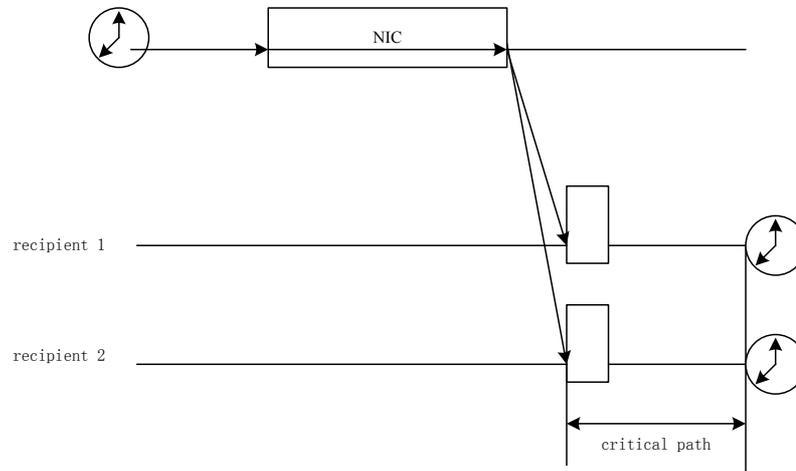
Fig. 1. Schematic diagram of RBS algorithm

3.2 TPSN time synchronization algorithm

TPSN (Timing-sync Protocol for Sensor Networks) synchronization algorithm is based on pair-wise mechanism, which can provide a wide range of wireless sensor network time synchronization.

TPSN algorithm is an algorithm for global time synchronization, which will bring a lot of energy consumptions after the completion of each global time synchronization. In addition, if the root node fails, the algorithm will select a new root node and run the TPSN algorithm, then the system burden increases.

TPSN algorithm is the first time synchronization algorithm for wireless sensor networks relevant to the system. It uses hierarchical network topology [5], and first of all establishes the level of all the nodes in the network. Then, between each node and the upper node, we use the message exchange way to get time synchronization, and eventually realize time synchronization in the whole network. The algorithm is mainly divided into two steps: the first step is layering, marking a level for each node, the node in n level needs to communicate with its superior, that is, n-1 level; the second step is time synchronization. All the tree nodes are synchronized, node in n level and node in n-1 level communicate and complete time synchronization, and finally all nodes and the root node are synchronized, to complete synchronization of the whole network. Figure 2 is a schematic diagram of TPSN layering.
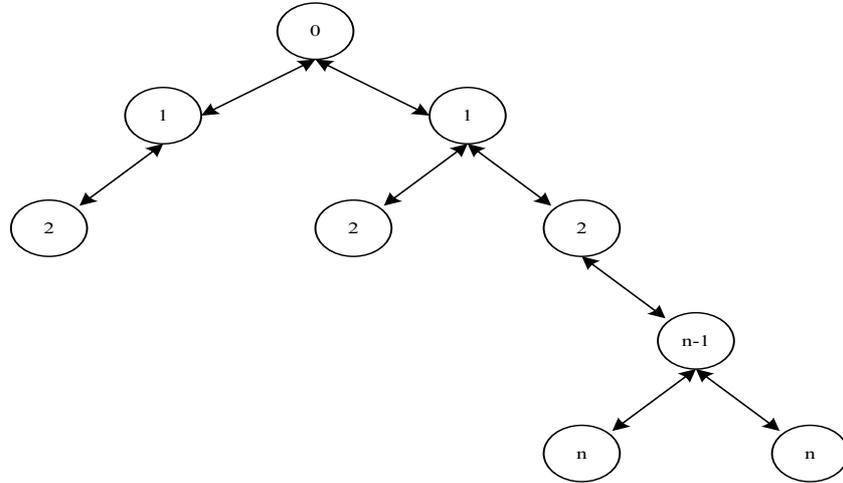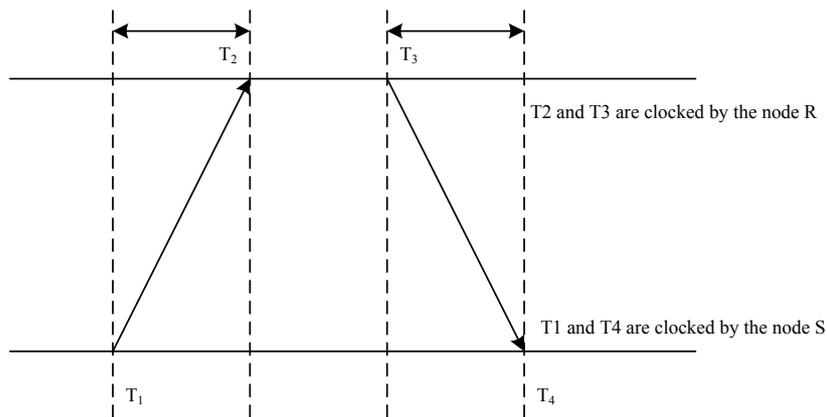
Fig. 2. Schematic diagram of TPSN layering

Fig. 3. Schematic diagram of TPSN algorithm

The synchronization principle of TPSN algorithm is shown in Figure 3. The S node in the graph is the n level node, R node level is n-1, T_1 and T_4 are the time measured for the local clock in different stages, T_2 and T_3 are the time measured for the node R of the local clock in different stages, Δ refers to the time deviation for the node R and the node S, and d indicates the time delay of messages in the communication process. Because the signal transmission speed is very fast, we assume that the time for the sending and receiving communication is the same. At the moment of T_1, S node sends a synchronization request to node R; packet information includes S node level and its packet transmission time T_1, and node R receives the synchronization

request packet at the moment of T_2. There are $T_2 = (T_1 + d + \Delta)$ and R node sending a response packet in the time of T_3, and the packet contains the level of node R. In addition, it also contains the previous time, T_1, T_2, and T_3. The time for S to receive a response message is T_4, then there is $T_4 = (T_3 + d - \Delta)$, and we can know that:

$$\Delta = \frac{(T_2 - T_1) - (T_4 - T_3)}{2} \tag{1}$$

$$d = \frac{(T_2 - T_1) + (T_4 - T_3)}{2} \tag{2}$$

Finally, the node S modifies the local clock by calculating the time offset between it and the node R, to achieve time synchronization with the node R.

4   Improved time synchronization algorithm

4.1   Algorithm description

In wireless sensor networks, the idea of clustering is widely used in data fusion, routing and network cost reduction. Clustering idea has its own cluster head election strategy [6] so that the cluster head manages the network, to make the topology easy to be managed, and the use of data fusion compresses the amount of data transmission, and saves the energy costs. Based on the classic wireless sensor network time synchronization algorithm, we summarize the synchronization error and energy consumption factors, and design an improved time synchronization algorithm, that is, clustering-based low energy consumption time synchronization (CLECTS). The algorithm requirements that wireless sensor network first of all generates the cluster topology, and then makes the base station and cluster to achieve time synchronization, and then realizes the cluster node time synchronization, so as to achieve the ultimate goal of the whole network synchronization.

The election of cluster head uses LEACH protocol, LEACH protocol divides the whole wireless sensor network into several clusters, and the cluster head uses cycle random mode [7]. In this way, the load of wireless sensor network is relatively average to all the sensor nodes, and then, it reduces the energy consumption of the network, and improves the network lifetime. The most important thing in the process is to choose a cluster head. LEACH requires that all sensor nodes randomly choose a number between 0-1, and once the value of a chosen node is less than a fixed threshold, then the node is selected as the cluster head node [8].

The following formula is used to calculate the predefined threshold $T(n)$:

$$T(n) = \begin{cases} \dfrac{p}{1 - p^* \left(r \; mod \; \dfrac{1}{p} \right)} & n \in G \\ 0 & \text{other} \end{cases} \tag{3}$$

The algorithm consists of three steps, namely cluster head selection, cluster membership adding and information transmission. The first two steps are continuously cycling, followed by a series of information transmission processes. The first step of the algorithm is to select the cluster heads and the selected cluster heads can select the cluster head nodes of the next cluster. For example, in the zeroth layer, the root node selects the first I node in its next layer node group as the cluster head node of the next layer of cluster and continues to broadcast the message. The energy consumed by the sending message is determined by the distance between the root node and the node for its communication.

The stage after the cluster head selection is the cluster member node adding stage. When the cluster head node is selected, all the cluster heads broadcast the message to all other nodes with the maximum power. The node selects a signal with the strongest signal and replies to a message, which means that the cluster head is added to the cluster, that is to say, the cluster node closet to the node.

The last stage of multi layer cluster network is the process of information transmission. In the process of signal transmission for a round, each member node sends sensor data to its cluster head node.

In this way, a highly correlated tree diagram is formed in the network, and the data of the sensor node can be transmitted from the far distance to the root node.

4.2 Algorithm energy model

The energy consumption model of transmitting and receiving 1 bit is given when the distance is d. In the range of the received signal and noise ratio, the transmission of L bit with the distance of d, the energy E_{Tx} consumed by the message sending and the energy E_{Rx} consumed by the receiving message are:

$$E_{Tx}(l,d) = \begin{cases} l^* E_{elec} + l^* \varepsilon_{fs}^{\ *} d^2 & d \geq d_0 \\ l^* E_{elec} + l^* \varepsilon_{mp}^{\ *} d^4 & d \geq d_0 \end{cases} \tag{4}$$

$$E_{Rx} = l^* \, E_{elec} \quad （5）$$

By using the algorithm proposed in this paper, we consider such a two-layer network: the total number of nodes is N, the number of cluster heads is k, the number of the first cluster heads is k_1, and the total number of membership nodes is $N - k$.

The base station is located at level 0, and the energy consumption of the base station is not taken into account in the whole network. In order to calculate the energy

consumption of the nodes in the two-layer network, we adopt the top-down method, and set the energy consumed of the first layer:

$$E_{Total,CL1} = 3k_1(1E_{elec}) + 1E_{elec} + 1^*\varepsilon_{mp}^*d_{toBS}^4$$
$$+ 2(k - k_1)\left[1E_{elec} + 1^*\varepsilon_{fs}^*d_{toCH}^2\right] \tag{5}$$

The energy consumption of the second layer:

$$E_{Total,CL2} = 3(k - k_1)1E_{elec} + k_1\left[1E_{elec} + 1^*\varepsilon_{fs}^*d_{toCH}^2\right]$$
$$+ 2(N - k)\left[1E_{elec} + 1^*\varepsilon_{fs}^*d_{toCH}^2\right] \tag{6}$$

The total energy consumption for $N - k$ sensor nodes sending and receiving 1 bit message:

$$E_{Total,NON-CH} = 2(N - k)1E_{elec} + (k - k_1)\left[1E_{elec} + 1^*\varepsilon_{fs}^*d_{toCH}^2\right] \tag{7}$$

The total energy consumed for the two-layer network to make time synchronization is:

$$E_{TOTAL} = E_{Total,CL1} + E_{Total,CL2} + E_{Total,NON-CH} \tag{8}$$

$$E_{TOTAL} = (4N + 1)1^*E_{elec} + (2N - k)1^*\varepsilon_{fs}^*d_{toCH}^2 + 1^*\varepsilon_{mp}^*d_{toBS}^4 \tag{9}$$

Through the above analysis, we can know that the wireless sensor network cluster-based low energy consumption time synchronization algorithm CLECTS described in this paper can effectively reduce the energy consumption, which is mainly achieved by reducing the transmission of data packets. According to the algorithm description, it can be seen that in the process of synchronization between the cluster head and the base station, only 3 data need to be transmitted at the beginning, and then the data exchange in the synchronization process is completed by continuing to transmit 1 data. In the process of synchronization of the members of a cluster, it only needs to transmit 3 packets, so the whole synchronization process only needs to transmit 6 packets. Some traditional synchronization algorithms, such as TPSN and RBS, we assume a network composed of n sensor nodes, then in order to achieve time synchronization, it is necessary to transmit $2^*(n - 1)$ packets.

5 Simulation experiments and results

5.1 Simulation platform

The simulation of NS2 (Network Simulator version 2) mainly contains the following two aspects; first, the level about programming on the OTcl, which is mainly that NS2 makes use of the existing network module to simulate, no need to modify the NS2, and it can complete the simulation as long as writing the OTcl script; second, the programming level based on the overall framework. NS2 itself, at the time, has no module for the network simulation, needing to expand NS2, add the necessary module needed for the network simulation, then recompile NS2, and finally write the OTcl script. NS2 network simulation flow chart is shown in Figure 4.
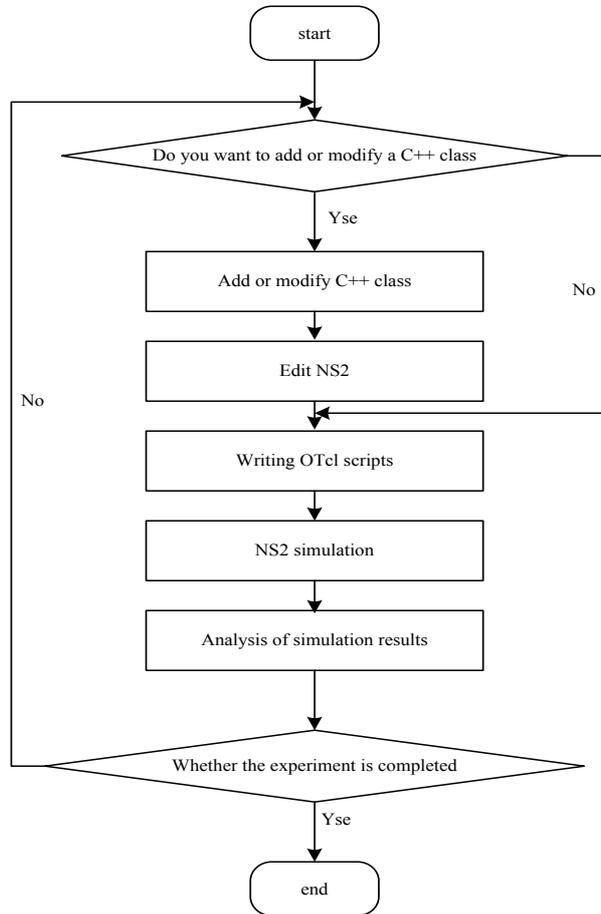
Fig. 4. NS2 simulation flow chart

Specifically divide into the following steps:

(1) According to the needs of the users, the NS2 is extended to modify or add new modules to the NS2 module;

(2) For the writing of script, make use of OTcl language. Scripting is mainly used to configure the user settings network environment structure and communication link attributes;

(3) To establish agency of Agent protocol, which are used to determine the communication model and to bind protocol for the equipment;

(4) Trace can record the event on a certain type in the simulation process and store in the trace file. NS2 takes trace file to record and save the whole process of simulation, and analyzes the trace file after the end of the simulation, which is a key step in NS2 simulation [9];

(5) To determine the total time of simulation, and then also involve the writing of some preparation information so that we have completed the preparation of the OTcl script;

(6) To write the completed OTcl script using the ns command to implement and get the trace file;

(7) To analyze the obtained trace file, get the data we need and analyze it.

5.2　　Simulation parameters

The performance of the improved algorithm is compared with the RSB and TPSN algorithms. The performance of the algorithm is mainly reflected in two aspects: energy consumption and synchronization accuracy [10]. In the simulation experiment, first of all, node is deployed in the area of 100m*100m, the transmission distance is 30m in the simulation, and the RxThresh value for wireless transmission range of 30m calculated by the threshold tool is 2.13643e-07. The simulation scene is arranged as follows:

In the scope of the 100m*100m network simulation, randomly deploy 100 static nodes, the node uses omni-directional antenna, the antenna transmission range is 30m, and the wireless transmission is two-way error-free transmission. Among them: the initial energy of the node is 1 J; the energy loss of the transmitting and receiving data is 50nJ/bit; the energy consumption of the data fusion is 5nJ/bit; the size of the data packet is 500bit, and the setting of the property is implemented in the OTcl script.

It can be very clearly displayed in Table 1:

Table 1.　Simulation parameters

Parameters	Parameters value
Total number of nodes	100
Area for the network	100m*100m
Base station location	[50,175]
Length of data packets	500bits
Initial energy of nodes	1J
E_{elec}	50 nJ(bit)$^{-1}$
ε_{fs}	10pJ/bit/m^2
E_{mp}	0.0013pJ/bit/m^4

5.3　　Simulation result

In order to make the simulation results more accurate, it is necessary to carry out several simulations to simulate different scenarios for 40 times, and the final results are statistically analyzed and the mean value is used as the final result. Finally, the number of nodes is increased, and several experiments are carried out and recorded.

Figure 5 shows the synchronization accuracy of the three algorithms. The graph shows that the synchronization accuracy of the algorithm is better than RBS and TPSN algorithm.
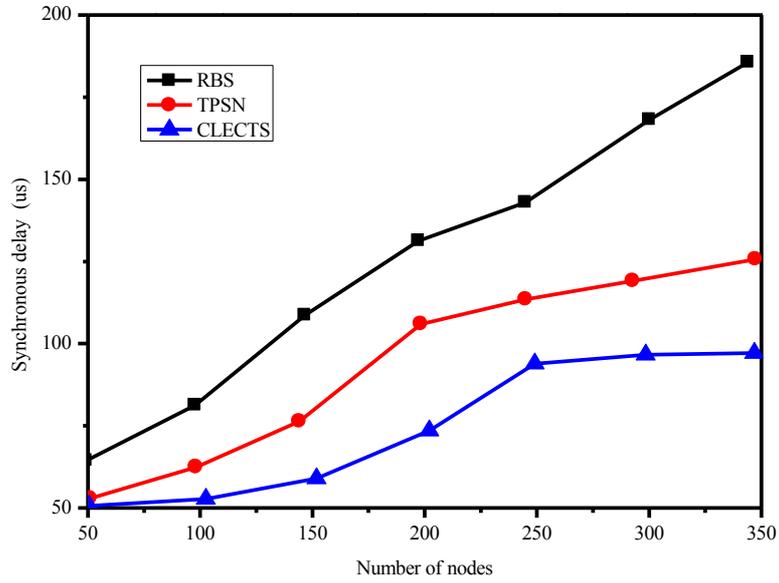
Fig. 5. Synchronous delay

Figure 6 shows the relationship of the message transmission amount with the increase of the number of nodes. The horizontal coordinate refers to the number of nodes, and the vertical coordinate indicates the energy consumption of message transmission.
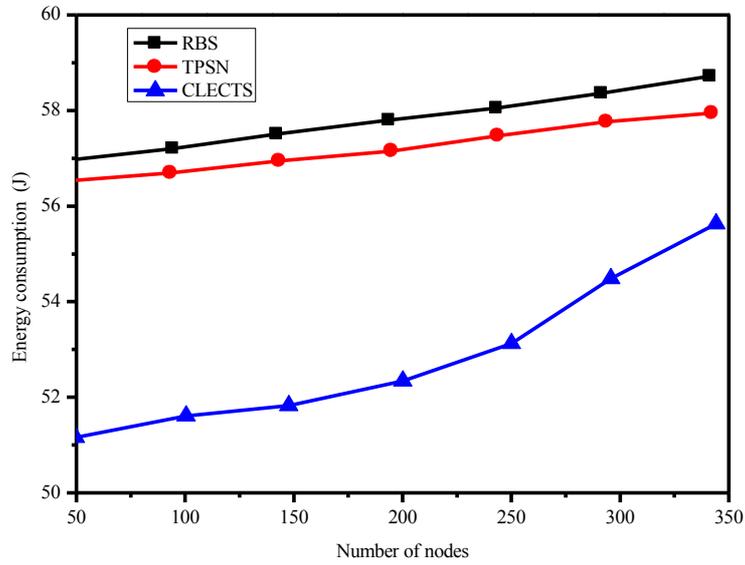

Fig. 6. Energy consumption

Through the analysis of simulation results of the above image, it can be obtained that the improved time synchronization algorithm is better than RBS and TPSN algorithm in synchronous time, and with the increase of the number of nodes in the network, the difference in the network overhead shows an increasing trend, which is an obvious advantage. This is because the consumption transmission distance of network node cluster in the improved synchronization algorithm is short. In addition, in the synchronization process, it reduces the number of message exchanges, and reduces the network synchronization overhead. As can be seen in the figure, compared to RBS and TPSN algorithm, the more the number of nodes, the more obvious the advantages of this algorithm. In addition, the improved algorithm has low complexity, and the calculation is not complicated, so the requirement of CPU in the network is relatively low, suitable for being applied in wireless sensor networks.

6   Conclusion

Based on the comparison of the existing time synchronization algorithms, we proposed an improved time synchronization algorithm for wireless sensor nodes in the network, and for node clustering in the network and the selection of cluster head, we used popular LEACH protocol, which can reduce the energy consumption of cluster head node. The network synchronization is divided into cluster head synchronization and cluster inside synchronization two processes, finally achieving the time synchronization of the whole network, and through the analysis, deriving the advantages of the algorithm, compared with RBS and TPSN algorithm, in saving network overhead advantages. In addition, the method of time marking in MAC layer is adopted, which eliminates the uncertain delay in the process of message passing and improves the precision of network time synchronization. In this algorithm, the amount of message exchange is less in the process of synchronization, which is the main reason to save the energy consumption and improve the synchronization accuracy. The NS2 platform is used to simulate the algorithm, and the results show that the algorithm has better time synchronization accuracy and lower network overhead compared with RBS and TPSN algorithm. In practical applications, the security of wireless sensor networks must be considered. The researches on the security of the algorithm are still little, needed to be further strengthened to increase the stability of the system.

7   References

[1] Benzaid, C., Saiah, A., & Badache, N. (2014). Secure pairwise broadcast time synchronization in wireless sensor networks. Euromicro International Conference on Ieee, parallel, Distributed and Network-Based Processing (pp.1-6). IEEE.

[2] Dutta, R., Gupta, S., & Das, M. K. (2014). Low-energy adaptive unequal clustering protocol using fuzzy c-means in wireless sensor networks. Wireless Personal Communications, 79(2), 1187-1209. https://doi.org/10.1007/s11277-014-1924-7

[3] Hunnekens, B., Dino, A. D., Nathan, V. D. W., Van Dijk, N., & Nijmeijer, H. (2015). Extremum-seeking control for the adaptive design of variable gain controllers. IEEE Transac-

tions on Control Systems Technology,23(3), 1041-1051. https://doi.org/10.1109/TCST.2014.2360913

[4] Jothi, S., & Chandrasekaran, M. (2014). Fuzzy based optimal clustering protocol for maximizing lifetime in wsn. Research Journal of Applied Sciences Engineering & Technology, 8(6), 714-725. https://doi.org/10.19026/rjaset.8.1027

[5] Kalaiselvi, T., & Murugan, T. S. (2016). An improved load balanced connection aware clustering hierarchy protocol for military application in mobile ad hoc network. Research Journal of Applied Sciences Engineering & Technology, 12(6), 658-667. https://doi.org/10.19026/rjaset.12.2714

[6] Kumari, S., Das, A. K., Wazid, M., Li, X., Wu, F., & Choo, K. R., et al. (2016). On the design of a secure user authentication and key agreement scheme for wireless sensor networks. Wireless Personal Communications, 1-18.

[7] Mostefaoui, A., Boukerche, A., Merzoug, M. A., & Melkemi, M. (2015). A scalable approach for serial data fusion in wireless sensor networks. Computer Networks, 79, 103-119. https://doi.org/10.1016/j.comnet.2014.12.008

[8] Ranjan, R., Wang, M., Perera, C., Jayaraman, P. P., Zhang, M., & Strazdins, P., et al. (2015). City data fusion: sensor data fusion in the internet of things. International Journal of Distributed Systems & Technologies, 7(1), 15-36.

[9] Rathi, & Viswanathan. (2014). Two phase clustering method for leach protocol for effective cluster head selection. Journal of Computer Science, 10(3), 366-375. https://doi.org/10.3844/jcssp.2014.366.375

8 Authors

Ji-hong Sun is Associate Professor at the Department of Computer Science, Luohe Vocational College of Food, Luohe, China (jjuf5931656@126.com).

Huanzheng Shao is Associate Professor at the Department of Computer Science, Luohe Vocational College of Food, Luohe, China (nbmyt5577@126.com).