

Time Synchronization Algorithm for Wireless Sensor Networks Based on Frequency Offset Estimation

<https://doi.org/10.3991/ijoe.v14i06.8699>

Shaojun Yu^(✉), Li Lin, Yujian Wang, Xingyuan Chen
Kunming University, Yunnan, China
shaojunyu2786@21cn.com

Abstract—Aiming at solving the poor the classical synchronous algorithm stability in wireless sensor network and high overhead of clock phase offset and frequency offset, a synchronization algorithm (CSMS algorithm) was designed for wireless sensor networks based on frequency offset estimation. The CSMS algorithm used the low overhead phase bias and frequency offset estimation method to improve the synchronization accuracy and stability of the pair nodes. At the same time, a synchronization strategy was built based on layering and broadcast monitoring, which ensured the stability and synchronization accuracy of the algorithm, realized the synchronization with neighbor nodes and root nodes, and optimized the total synchronization cost. Among them, the CSMS algorithm was mainly divided into two stages: level discovery phase, which was used for generating a layered structure of network; synchronization phase, used to estimate clock offset and frequency offset between pairs of nodes. The experimental results showed that the CSMS algorithm can effectively balance the synchronization energy consumption, synchronization accuracy and synchronization stability. As a result, it is summed up that dynamic adjustment of the nodes clock deviation is realized, the long-term stability of synchronization is ensured, and the precision of synchronization is improved.

Keywords—wireless sensor network, timing synchronization, low-overhead

1 Introduction

Wireless sensor network (WSN) is a multi-hop and self-organizing distributed network formed by a large number of micro sensor nodes with the ability of information acquisition, processing and wireless communication. Time synchronization is the key technology of wireless sensor network. It is very important to achieve sensor data fusion, node data processing, ranging and positioning, cooperative sleep and so on. In recent years, the problem of time synchronization of WSN has been widely concerned, and a large number of research results have been obtained. According to the synchronization overhead, synchronization accuracy and synchronization stability problems, we put forward a variety of WSN time synchronization algorithms. In addition, we achieve the purpose of improving the synchronization accuracy, ensuring synchronization stability and reducing synchronization overhead from different mes-

sage transmission mechanisms, error compensation methods and topology optimization, respectively. RBS (Reference Broadcast Synchronization) is a low overhead protocol to achieve synchronization between receivers based on broadcast information, but the frequency drift of nodes is not considered, and the synchronization accuracy is low. On the basis of RBS, IRBS uses Bayesian estimation to improve the synchronization accuracy, but it cannot keep the long-term stability of synchronization and it increases the total cost of synchronization. PBS (Protocol Broadcast Synchronization) is a high precision synchronization protocol for compensating phase offset and frequency offset in the broadcast domain of a pair of nodes, but the cost of synchronization is large. DMLT combines the RBS and TPSN algorithms to realize the low overhead synchronization of the whole network, but neglects the frequency offset of the node and the synchronization precision is low. CESA proposes a synchronization algorithm based on clustering hierarchy, which combines 2 ways of unidirectional broadcast and pair wise synchronization. It uses Bayesian estimation to optimize multi hop error, which is a low overhead time synchronization protocol, but the synchronization stability is poor. TW-COSE, JSOS is a synchronization algorithm based on the clock offset and frequency offset joint correction at the transmitter side, which has high synchronization accuracy, but needs many pairs of message exchanges so that the synchronization cost is great.

Considering the three aspects of synchronization overhead, synchronization accuracy and synchronization stability, this paper proposes a time synchronization algorithm based on frequency offset estimation (CSMS algorithm). CSMS algorithm uses low overhead clock phase offset and frequency offset estimation method (SCSE), to achieve comprehensive optimal synchronization stability, synchronization accuracy and synchronization overhead. By using hierarchical, radio listening and piggybacking combined strategy; we achieve synchronization with the neighbor nodes and the root nodes and optimize the synchronization total overhead.

2 Literature review

Traditional network time synchronization schemes, such as NTP protocol and GPS system, synchronize the time of intra network nodes to the UTC time source outside the network, so as to achieve synchronization in the network and the synchronization of network nodes and UTC time outside the network. In wireless sensor networks, it is not always necessary to synchronize all nodes to UTC time. It only needs to synchronize the reference time and UTC time in the wireless sensor network and the base station nodes of other communication networks. Therefore, most of the time synchronization algorithms of wireless sensor networks pay more attention to the realization of time synchronization in the network.

Pérez-Solano and Felici-Castell as stated in [1] the adaptive time window linear regression algorithm for accurate time synchronization in wireless sensor networks. Wang and others as stated in [2] discussed the linear estimation of clock frequency offset using wireless sensor networks. Xu and others as stated in [3] studied the energy-efficient time synchronization in wireless sensor networks via temperature-aware

compensation. At the same time, Kim and Lim as stated in [4] also discussed the energy-efficient time synchronization based on source clock frequency recovery in wireless sensor networks. Shi and others as stated in [5] designed the multiple two-way time message ex-change (ttme) time synchronization for bridge monitoring wireless sensor networks. Lee and Chin as stated in [6] studied the low-complexity time synchronization for energy-constrained wireless sensor networks. Liao and Barooah as stated in [7] designed an algorithm for accurate distributed time synchronization in mobile wireless sensor networks. Wang and others as stated in [8] proposed the linear estimation of clock frequency offset for time synchronization based on wireless sensor networks. Jayaprakash and Reddy as stated in [9] studied a robust blind carrier frequency offset estimation algorithm for ofdm systems. Zhang and others as stated in [10] designed a single pilot subcarrier-based sampling frequency offset estimation and compensation algorithm.

To sum up, the existing research is not accurate enough for the time node calculation of the sensor. To solve the poor the classical synchronous algorithm stability in wireless sensor network and high overhead of clock phase offset and frequency offset, a synchronization algorithm (CSMS algorithm) was designed for wireless sensor networks based on frequency offset estimation. This method that combines the CSMS algorithm can effectively balance the synchronization energy consumption, synchronization accuracy and synchronization stability.

3 An overview of time synchronization in Wireless Sensor Networks

3.1 Time synchronization message transfer delay

The uncertain factors of message transfer delay are another important factor affecting the synchronization accuracy. Through the above clock synchronization model, we know that to synchronize the node i with other node j (or reference node), the corresponding synchronization message transfer must be carried out. Therefore, it is necessary to analyze synchronization message transmission delay composition and characteristics in the process of message transfer. It also requires analyzing certain factors and uncertain factors in the process of message transfer, and then improves the efficiency of node i using these synchronous messages for calculation, analysis and adjustment of the local clock rate.

Ganeriwal and others analyzed the details of the synchronous message (Package) message transfer process in the wireless communication process, and decomposed the message transmission delay into six components from the perspective of system level. The specific composition is shown in Figure 1.

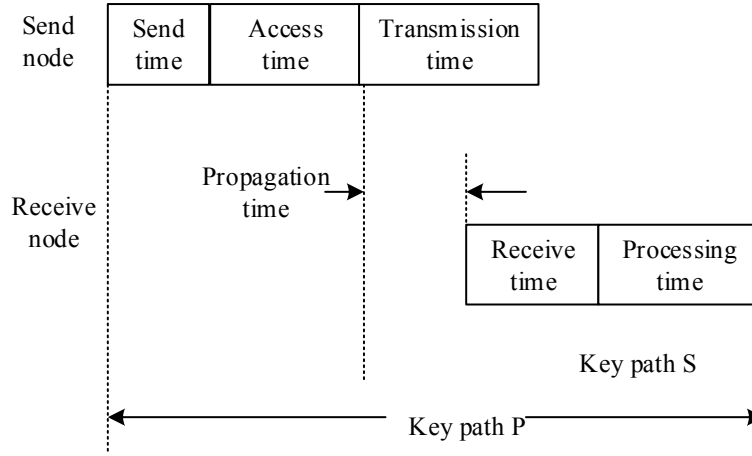


Fig. 1. Composition of time delay in message transfer

The explanation of each component of the time delay in the synchronization message passing in Figure 1 is as follows:

Send time: during this time, the send node needs to produce the corresponding time sync message, and sends to the MAC layer. The length of time depends on the MCU kernel protocol processing time, the corresponding operating system (OS) call time and interrupt processing and so on time, which are related to processing speed and load of MCU, having certain uncertainty.

Access time: the waiting time of sending synchronous messages to the MAC layer and obtaining the access rights of the wireless channel. The time length is dependent on the current network load and the MAC protocol adopted by the network, which is also an uncertain factor.

Transmission time: time spent for decoding the synchronous message and sending in accordance with the number in turn through wireless signal after send node obtains channel utilization right. Because the synchronization message length and wireless transmission power are known, the time is certain.

Propagation time: the time taken for synchronous messages to transmit from the sending node to the receiving node in the transmission medium. The time can be determined by the transmission distance between the transmission media and the two nodes, and it is a determinable value. Table 1 is a part of the common values and features of the synchronization message transmission delay.

Table 1. Some common values and characteristics of synchronous message transmission delay

Transmission time	Common value	Characteristics
Send Time & Receive Time	0~100ms	Uncertain
Access time	10~500ms	Uncertain
Transmission Time& Reception Time	10~20ms	Certain
Propagation time	<1us	Certain

3.2 Time synchronization design elements and performance indicators

Due to the characteristics of its self-organizing network and practical application environment, WSN cannot be limited by energy except for sink nodes or Sink base station nodes. If the root node chooses the nodes inside the network selected randomly, the node will also be limited by energy. Other nodes use batteries that are not capable of power management and maintenance, such as volume, power and life. And we take into account the characteristics of the node itself, such as small volume, low computer capability and weak communication performance and so on. Therefore, in the process of time synchronization of wireless sensor networks, it is necessary to consider the factors that are often ignored in the traditional network.

Energy constraints and their importance to WSN make energy management one of the hot-spots of research on WSN. There are generally two strategies for energy efficient utilization: by balancing the traffic load between nodes, the energy consumption of each node reaches a relative equilibrium state; by reducing the energy consumption of each sensor node, the service life is prolonged. The energy consumption distribution of each module in typical sensor node is shown in Figure 2.

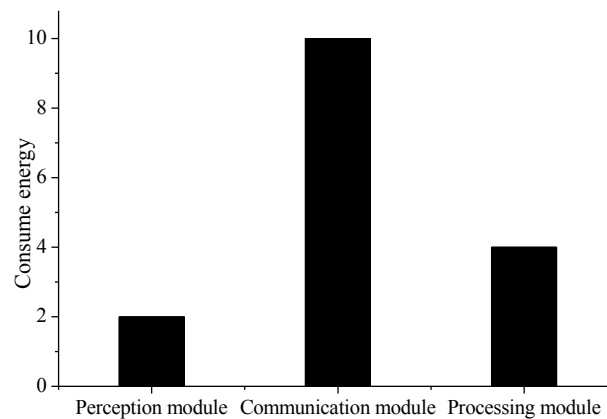


Fig. 2. Energy consumption distribution of sensor nodes

The sensor nodes mainly include the perception module, the communication module, the processing module and the power supply module, in which the power supply module provides energy for other modules. We can see from Figure 2, energy consumption in wireless sensor network is mostly used for wireless communication. On the condition that the sensor nodes energy cannot be supplemented, the number of messages mutually transferred by nodes must be effectively controlled in the synchronization process. At the same time, because the processing task of each node and the node performance problems lead to different residual energy of each node, sometimes, in order to enhance the network life cycle, it is necessary to balance the energy consumption of each sensor node in the sensor network, so as to achieve energy efficient uniform consumption. Secondly, it also shows that a relatively simple algorithm is more convenient than computer, which can increase the complexity of the computer

without bringing much energy consumption. It can also improve the synchronization accuracy, reduce the synchronization frequency and save energy. Finally, it is necessary to consider the energy consumption in the long-term synchronous operation of the whole network life cycle.

3.3 Time synchronization mechanism

According to the synchronous message transfer between nodes in WSN, we can summarize the classic algorithm as the synchronization mechanism based on sender - receiver (SR), the synchronization mechanism receiving only (RO) and the synchronization mechanism based on receiver - receiver (RR) a total of three categories. These synchronization mechanisms show different characteristics in the synchronization energy consumption, synchronization error and synchronization computational complexity as well as expansibility.

The synchronization mechanism based on receiver - receiver (RR): the basic idea of RR synchronization mechanism is all nodes to be synchronized receive the synchronous message sent by a "third party" node and record the moment of receiving the synchronization message. Then, the time information is transmitted between nodes to be synchronized, and at last, the mean value of time deviation between each node and other nodes is calculated. We can choose one node as the reference node, and other nodes to be synchronized (except for the nodes broadcasting information) and this node complete the synchronization operation. The most classical algorithm is the RBS algorithm proposed by J.Elson and others. And the schematic diagram of the RBS synchronization algorithm is shown in Figure 3.

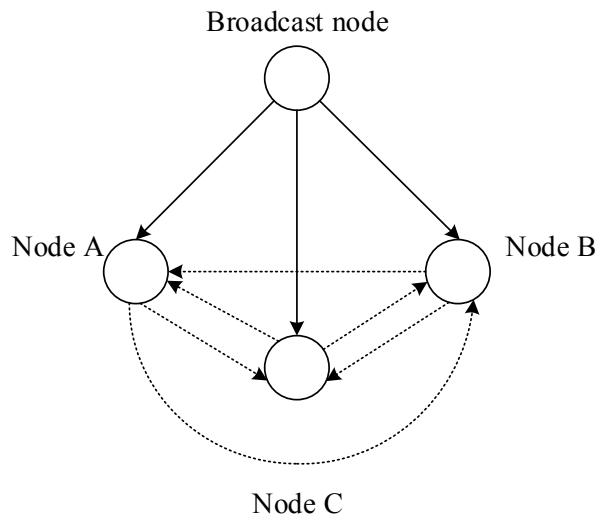


Fig. 3. RBS synchronization algorithm principle

Sender - receiver synchronization mechanism SR: the basic idea of SR synchronization mechanism is that, the sending nodes send the packet information containing time information to the receiving node. Then, the receiving node records the moment of receiving the synchronous message and processes it again, including packet unpacking and then sending the reply into synchronous message. After the sending node receives the reply packet, according to the time information obtained, the local clock is adjusted and the time synchronization process is completed. The classical TPSN algorithm using SR mechanism is used to illustrate the principle of SR synchronization mechanism. The TPSN algorithm adopts hierarchical network structure. Sink nodes are generally used to provide standard time for the whole network. The lower nodes communicate with the upper nodes for two-way message exchange and correct the local clock offset. Finally, the time synchronization between the whole network node and the Sink node is completed.

4 Time synchronization algorithm design for wireless sensor networks based on frequency offset estimation

4.1 Node clock model

Hardware clock model: the hardware clock model of sensor node i can measure time through the frequency of crystal oscillator, and its local clock at physical time t can be expressed as (1).

$$C_i(t) = \frac{1}{f_0} \int_{t_0}^t f_i dt + C_i(t_0) \quad (1)$$

In the above formula, f_i suggests the standard frequency of a nodal oscillator, f_0 indicates the actual working frequency of the nodal oscillator and $C_i(t_0)$ refers to the local time of node at the moment of t_0 . Because the clock frequency is affected by internal and external factors and it is a small and slow process, it can be assumed that the frequency of the clock is unchanged in a relatively short time. Then, the above formula can be expressed as:

$$C_i(t) = \alpha_i(t - t_0) + \beta_i \quad (2)$$

In (2), α_i suggests the relative frequency and β_i refers to the clock readings at the initial time. It needs to be explained that the frequency of the oscillator is changing, but generally there is a certain limit. In consequence, α_i in (2) indicates the relative frequency. The difference between the node clock readings relative to the standard time is defined as the clock frequency offset (abbreviated as frequency offset). In addition, due to the limitation of manufacturing process, the actual working frequency f_0 is different from the standard frequency, and the crystal frequency of each sensor node is also different. The initial value of the clock (or the initial value between two

nodes) is different, which is defined as clock phase offset, abbreviated as offset. Clock drift is one of the main factors that affect synchronization accuracy. Another factor is the message transfer delay needing to exchange messages synchronously when it is required to achieve synchronization between nodes.

Software clock model: sensor nodes use their hardware clock readings to build corresponding software clock models. The purpose of building software clock is to convert the readings of local clock's hardware clock, so as to achieve synchronization with physical time. From (2), it can be obtained that:

$$t = \frac{1}{\alpha_i} C_i(t) + (t_0 - \frac{\beta_i}{\alpha_i}) \quad (3)$$

In order to synchronize the clock and physical time of nodes, we only need to convert the software clock of sensor nodes into corresponding logical time, and get the clock frequency correction factor and the clock phase offset correction coefficient from (3). In the same way, in order to synchronize the sensor node i with another sensor node j , the two nodes clock relationship can be expressed as:

$$C_j(t) = \alpha_{ij} C_i(t) + \beta_{ij} \quad (4)$$

In the above formula, α_{ij} and β_{ij} indicate the relative clock frequency offset and phase offset between two nodes, respectively. Then, the clock frequency correction coefficient and the clock phase offset correction coefficient can be obtained by (4), and the node i synchronizes with the node j through the correction coefficient.

In general, in wireless sensor network, (4) is used to complete the node synchronization (if another node is a Sink node, the node can be considered the frequency offset of 1 and phase offset of 0), and two different synchronization ways can be obtained: clock frequency offset compensation and clock phase deviation compensation. For node i , if we can get the deviation from the node clock phase, at this time, node i and node j will achieve synchronization, which is called phase offset compensation. The synchronization between the nodes achieved in this way is simple, but the frequency deviation between the two nodes is not considered. As shown in Figure 4, the local time offset between node i and node j is $D_{t_1}^{i \rightarrow j}$ at the moment of t_1 ; and at the moment of t_2 , the local time bias between node i and node j is changed to $D_{t_2}^{i \rightarrow j}$ and $D_{t_2}^{i \rightarrow j} = D_{t_1}^{i \rightarrow j} + RD_{t_1 \rightarrow t_2}^{i \rightarrow j}$, in which $RD_{t_1 \rightarrow t_2}^{i \rightarrow j}$ is caused by the difference of clock crystal frequency in the time period of $t_1 \sim t_2$.

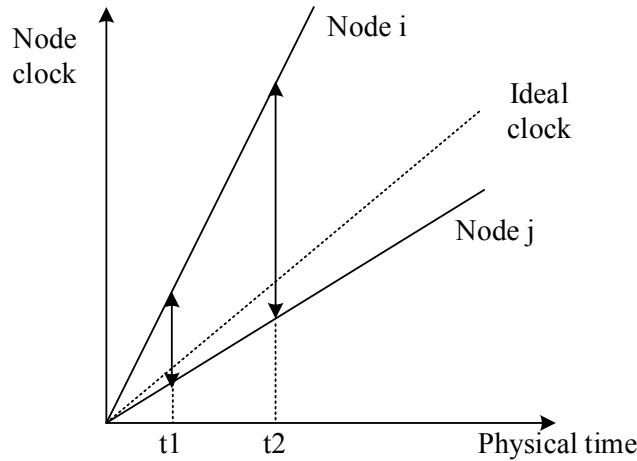


Fig. 4. The influence of clock frequency offset

Therefore, in the case of uncompensated clock frequency offset α_{ij} , the deviation between clock $C_i(t)$ and clock $C_j(t)$ will increase with time, it will quickly exceed the synchronization accuracy of application requirements, and increase the number of synchronization operations again. In addition, because of the impact of clock drift in the node, the two node clocks after calibration phase offset (through clock frequency offset and phase offset compensation synchronous operation), after a period of time, will appear asynchrony phenomenon again. And as time goes by, the phase deviation between two clocks will become increasingly large. Therefore, time synchronization algorithm needs to periodically exchange messages and update time information, so as to maintain a certain accuracy synchronization. Compared with only the compensation clock phase offset β_{ij} and the both compensation of clock phase offset β_{ij} and frequency offset α_{ij} , the synchronization algorithm between node E and node J can improve the synchronization accuracy while reducing the number of synchronization cycles. Therefore, the time for sensor node clock requiring time synchronization is not necessarily accurate or precise. And only when the node realizes the synchronization with physical time can the above three time indexes be met. For most of the wireless sensor network applications, it does not require sensor nodes to maintain synchronization with the physical time. Therefore, the time synchronization studied in this paper is a time synchronization algorithm maintaining certain accuracy and precision.

4.2 Estimation model

There are differences in the internal crystal oscillators for different nodes, so there is a difference in the frequency of the clock. As time goes on, the clock deviation between nodes will increase continuously. Therefore, the estimation and correction of the node clock frequency deviation can not only improve the synchronization accura-

cy, but also maintain the long-term stability of the synchronization. The traditional algorithm cannot accurately reflect the actual error. Although the proposed clock deviation model is close to the reality, the calculation is complex and the synchronization cost is great. In many practical applications, the requirement of synchronization accuracy is not strict, but the energy consumption is limited strictly. As a result, it is very important to simplify the clock parameter estimation model under the premise of ensuring the effective synchronization accuracy. In view of the above problems, a low overhead clock offset and frequency offset estimation method based on the sender - receiver is proposed.

Figure 5 is a timing information exchange model between two points when there is a clock phase bias and a frequency offset. The time stamps $T_1^{(A)}$, $T_4^{(A)}$ and $T_5^{(A)}$ of the information exchange are based on the local clock of the node A, and $T_2^{(B)}$, $T_3^{(B)}$ and $T_6^{(B)}$ are based on the local clock of the node B. At the moment of $T_1^{(A)}$, the node A synchronously packets to the node B broadcast time, which contains the topology level, identification number and time-stamp $T_1^{(A)}$ of the node A. The node B receives the synchronization packet at the moment of $T_2^{(B)}$, and after waiting for a period of time, sends a synchronous request message packet to node f}. It contains the topology level, recognition number and time stamps $T_1^{(A)}$, $T_2^{(B)}$ and $T_3^{(B)}$ of node B. The node A always receives the synchronization request message packet at the moment of $T_4^{(A)}$ and after waiting for a period of time, and then sends a synchronous acknowledgement packet to node B. The group contains the topology level, recognition number and time stamps of $T_1^{(A)}$, $T_2^{(B)}$, $T_3^{(B)}$, $T_4^{(A)}$ and $T_5^{(A)}$ of node f}. At last, the node B receives the acknowledgement packet at the moment of $T_6^{(B)}$.

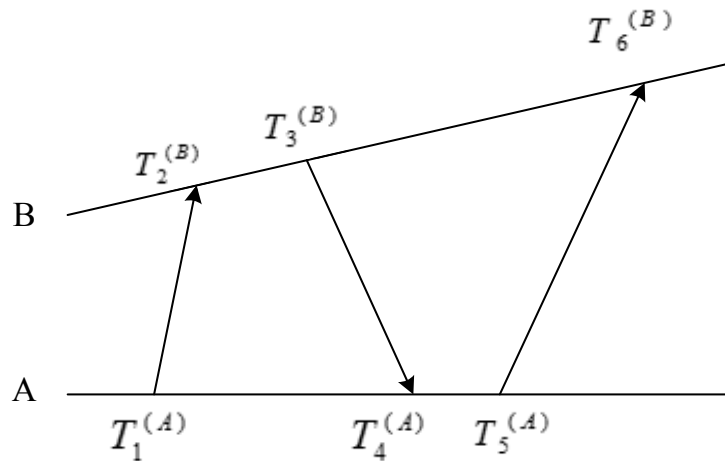


Fig. 5. Two nodes timing information exchange model

4.3 Level discovery stage and synchronization stage

The discovery is to create a hierarchical structure in the network and the stage includes the following steps. First of all, select a root node, and assign the root node zeroth; secondly, divide the root node broadcast level discovery packet (LDP). LDP contains the node ID and level of the group; then, each node that receives LDP defines the level to the next layer of received packet level, and sends a new level discovery packet containing its own node. Once the node defines the layers, it ignores the later level discovery packet, so as to avoid network flooding. The last step is repeated, until each node in the network successfully defines the level. After the above steps, the network topology, as shown in Figure 6, can be built.

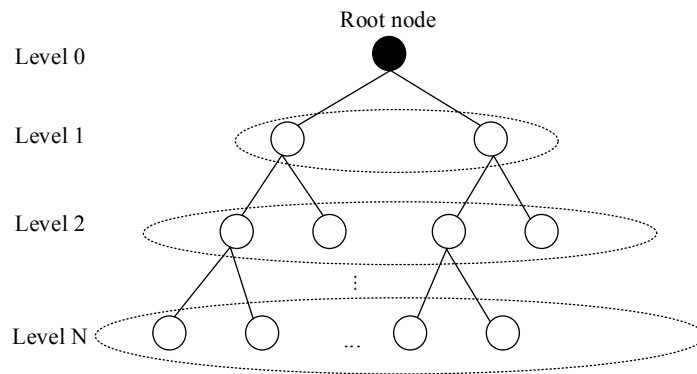


Fig. 6. Hierarchical network topology

5 Experiment and analysis

5.1 Experimental structure analysis

In order to evaluate the performance of CSMS algorithm, this paper performs a synchronous simulation based on the algorithm, and compares two classical synchronization algorithms, TPSN and TW-COSE. TPSN does not consider the frequency offset factor of the node. TW-COSE takes into account the frequency offset, and achieves synchronization by multiple pairs of message exchange and phase offset estimation. The algorithm complexity is relatively high.

The experimental topology of linear and layered networks, as shown in figure 7 and figure 8, is used in the experiment. In the experiment, node 1 is a synchronous initiating node, and its time is considered as a standard time. The total running time of nodes (standard time) is 2700s and all nodes are started at the same time in the initial time. Thereafter, every 90s, node 1 launches a new round of synchronization and the total synchronization is 30 rounds. Starting from node 1, we launch the next round synchronization until the next round begins and this process is regarded as a synchronization cycle. The frequency drift of the node is generated randomly in the range of -

20~20us/s. In each round of synchronization, the message delay varies randomly between 0.01~0.4s. When the TW-COSE algorithm is simulated, the number of exchange times of two-way messages in the first synchronization of the paired nodes is set to 4. The performance of CSMS, TPSN and TW-COSE in synchronization accuracy, synchronization stability and synchronization overhead are compared and analyzed in the experiment.

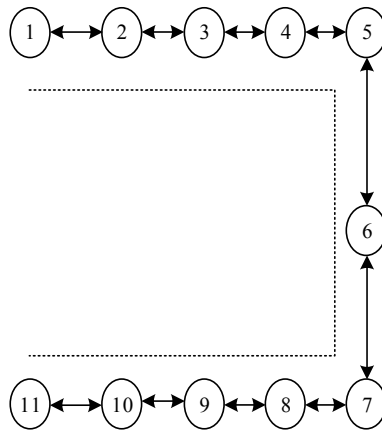


Fig. 7. Experimental simulation linear topology

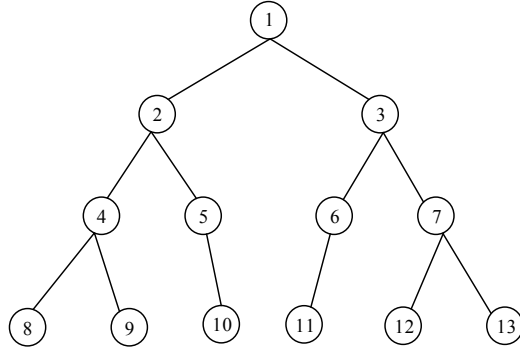


Fig. 8. Experimental simulation hierarchical network topology

5.2 Synchronization overhead analysis

With synchronous message number as a measure index of synchronization overhead, figure 9 is a comparison of the synchronization overhead of three algorithms when the pair nodes are synchronized. From Figure 9, it is known that the synchronization overhead of CSMS algorithm, in terms of pairs of nodes synchronization, is between the classical algorithm TPSN and the complex frequency offset estimation algorithm TW-COSE.

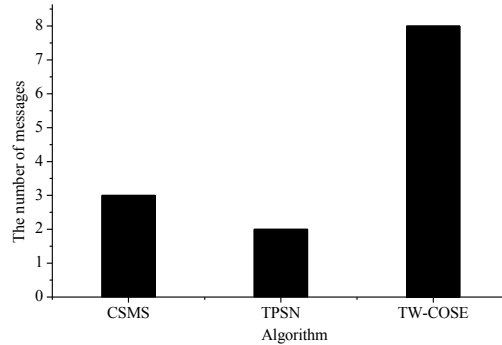


Fig. 9. A comparison diagram of the number of synchronous messages required for the completion of synchronization of pairs of nodes

Table 2 compares the synchronization overhead of the three algorithms when the N nodes are synchronized. From table 2, it is known that when N nodes are synchronized, the synchronization overhead of the CSMS algorithm is similar to that of the TPSN algorithm, but is far lower than that of the TW-COSE algorithm. Therefore, the CSMS algorithm can effectively reduce the synchronization cost under the premise of achieving high synchronization accuracy.

Table 2. The number of synchronization messages required by N nodes to complete synchronization

Algorithms	M
CSMS	2N+1
TPSN	2N
TW-COSE	16N

6 Conclusions

In this paper, based on the sender - receiver synchronization method, a low overhead clock phase bias and frequency offset estimation method (SCSE) is proposed. At the same time, it is extended to the entire wireless sensor network. The research result is as follows:

First, a CSMS algorithm is proposed to synchronize with neighbor nodes and root nodes.

Second, by using broadcast monitoring, the synchronization overhead of the whole network is further optimized. The effectiveness of the algorithm is verified by simulation experiments. The experimental results show that, compared with TPSN and TW-COSE synchronization algorithm, CSMS algorithm can ensure long-term synchronization stability and effectively improve the synchronization accuracy.

Finally, the CSMS algorithm balances synchronization overhead, synchronization accuracy and synchronization stability and achieves comprehensive optimal synchronization effect.

7 References

- [1] Pérez-Solano, J. J., & Felici-Castell, S. Adaptive time window linear regression algorithm for accurate time synchronization in wireless sensor networks. *Ad Hoc Networks*, 2015, vol. 24(6), pp. 92-108. <https://doi.org/10.1016/j.adhoc.2014.08.002>
- [2] Wang, H., Zeng, H., & Wang, P. Linear estimation of clock frequency offset for time synchronization based on overhearing in wireless sensor networks. *IEEE Communications Letters*, 2016, vol. 20(2), pp. 288-291. <https://doi.org/10.1109/LCOMM.2015.2510645>
- [3] Xu, M., Xu, W., Han, T., & Lin, Z. Energy-efficient time synchronization in wireless sensor networks via temperature-aware compensation. *Acm Transactions on Sensor Networks*, 2016, vol. 12(2), pp. 12. <https://doi.org/10.1145/2876508>
- [4] Kim, K. S., Lee, S., & Lim, E. G. On energy-efficient time synchronization based on source clock frequency recovery in wireless sensor networks. *Computer Science*, 2015, vol. 16(3), pp. 481-486.
- [5] Shi, F., Tuo, X., Yang, S. X., Li, H., & Shi, R. Multiple two-way time message exchange (ttme) time synchronization for bridge monitoring wireless sensor networks. *Sensors*, 2017, vol. 17(5), pp. 1027. <https://doi.org/10.3390/s17051027>
- [6] Lee, Y. R., & Chin, W. L. Low-complexity time synchronization for energy-constrained wireless sensor networks: dual-clock delayed-message approach. *Peer-to-Peer Networking and Applications*, 2016, vol. 10(4), pp. 1-10.
- [7] Liao, C., & Barooah, P. An algorithm for accurate distributed time synchronization in mobile wireless sensor networks from noisy difference measurements. *Asian Journal of Control*, 2016, vol. 18(6), pp. 1983-1994. <https://doi.org/10.1002/asjc.1348>
- [8] Wang, H., Zeng, H., & Wang, P. Linear estimation of clock frequency offset for time synchronization based on overhearing in wireless sensor networks. *IEEE Communications Letters*, 2016, vol. 20(2), pp. 288-291. <https://doi.org/10.1109/LCOMM.2015.2510645>
- [9] Jayaprakash, A., & Reddy, G. R. Robust blind carrier frequency offset estimation algorithm for ofdm systems. *Wireless Personal Communications*, 2017, vol. 94(3), pp. 1-15. <https://doi.org/10.1007/s11277-016-3650-9>
- [10] Zhang, Z., Zhang, Q., Li, Y., Song, Y., Zhang, J., & Chen, J. A single pilot subcarrier-based sampling frequency offset estimation and compensation algorithm for optical imdd ofdm systems. *IEEE Photonics Journal*, 2016, vol. 8(5), pp. 1-9. <https://doi.org/10.1109/JPHOT.2016.2607700>

8 Authors

Shaojun Yu, Li Lin, Yujian Wang, and Xingyuan Chen are with the School of Information and Technology, Kunming University, Yunnan, China.

Article submitted 13 February 2018. Resubmitted 15 March 2018. Final acceptance 23 April 2018. Final version published as submitted by the authors.