The Research of Network-induced Delay Measurement Methods in Wireless Networked Control Systems based on Zigbee Network

Feiping Lu, Wen Xue and Zhaozhou Zeng
Shanghai Electric Control Institution, Shanghai, 200092
Email: lufp2000@163.com; xuewen1217@sina.cn; z.z.zeng@163.cn

Abstract. The large and random network-induced delay in ZigBee wireless network seriously reduce the performance of wireless networked control systems (WiNCS) based on ZigBee network, due to unreliable communication, limited bandwidth, as well as high bit error rate of ZigBee network. Compared with other methods, such as converting time-varying network-induced delay into a fixed value, assuming network-induced delay under a certain statistical law, real-time measurement and compensation of random network-induced delay can effectively improve the performance of WiNCS. In order to obtain the network-induced delay of WiNCS based on ZigBee network on-line, this paper deeply analyzes the composition of delay, provides three delay measurement methods, clock synchronization method, confirm mechanism method and online estimation method, and designs a WiNCS network data transmission experimental platform. Then measurement precision of this measurement methods are analyzed by experiments that based on the experimental platform. The experiments show that, these measurement methods achieve very high measurement precision by taking compensation measures.

1. Overview
At present, as a short-distance, low-complexity, low-power, low-cost wireless communication technology, ZigBee can form a large-scale wireless network at a very low price which has been increasingly used in many applications [1], such as wireless sensor networks and industrial control field. However, the large and random network-induced delay in ZigBee wireless network has brought lots of challenges in further application of wireless networked control system[2] (Wireless Networked Control System, WiNCS) based on ZigBee network, due to unreliable communication, limited bandwidth, as well as high bit error rate of ZigBee network. Network-induced delay has an important impact on WiNCS. Compared with other methods, such as converting time-varying network-induced delay into a fixed value[3], assuming network-induced delay under a certain statistical law[4], real-time measurement and compensation of random network-induced delay can effectively improve the performance of WiNCS. Therefore, the study of WiNCS based on ZigBee network’s network-induced delay is very important. The experiment based on the actual network test platform is more practical than the software simulation. Therefore, this paper designs a WiNCS network data transmission experimental platform. The experimental measurement methods of ZigBee network-induced delay are studied, and the measurement accuracy of these methods is analyzed theoretically and experimentally.

2. Network-Induced Delay Analysis
The total delay of a complete closed-loop control for WiNCS is consisted of as follow three components: firstly sensor-to-controller network-induced delay $\tau_s$, secondly controller calculation
delay $\tau$, and the last is controller-to-actuator network-induced delay $\tau_{ca}$. Among them, $\tau$ is related to the complexity of the control algorithm and the running speed of the controller. Generally $\tau$ is a fixed value. If the control algorithm is designed to be simple, $\tau$ can be ignored. But $\tau_{ns}$ and $\tau_{ca}$ both belong to network-induced delays and their values are affected by various complex factors such as network protocols and network load. According to the analysis of general network-induced delay in document [5], the basic time sequence of sending a data packet by a ZigBee network node is shown in Figure 1. The network-induced delay is composed as follows:

$$
\tau = \tau_{si} + \tau_{s1} + \tau_{h1} + \tau_f + \tau_p + \tau_{s2} + \tau_{h2} + \tau_{si}
$$

(1)

In the formula (1), $\tau_{si}$ shows the internal processing delay of the source node protocol stack, $\tau_{s1}$ shows the delay that the source node waits, $\tau_{h1}$ shows the delay that the source node hardware handles, $\tau_f$ shows the transmission delay, $\tau_p$ shows the network transmission delay, $\tau_{s2}$ shows the internal processing delay of the destination node protocol stack. $\tau_{s2}$ shows the delay that destination node waits; $\tau_{h2}$ shows the delay that handles the destination node hardware handles. $\tau_f$ can be ignored due to the very fast speed of current and radio waves, etc in the transmission medium. And related to specific hardware devices, in general, and $\tau_{h1}$ and $\tau_{h2}$ can also be ignored. $\tau_f$ is determined by the network transmission rate and the packet length. $\tau_{s1}$ and $\tau_{s2}$ depend on the network protocol, the number of packets already in the transmit buffer, the task period, the processing load of the system, and the network load. $\tau_{si}$ and $\tau_{s2}$ is related to the efficiency of the protocol stack and hardware performance.

ZigBee's MAC layer uses CSMA/CA technology. $\tau_{ns}$ and $\tau_{ns}$ are both random unpredictable amounts. When ZigBee network node sends data, it must first call the sending function in the application layer of the protocol stack. The data needs to pass through the APS layer, the NWK layer, the MAC layer and the PHY layer [6] to convert the data into wireless signals to send out. These transfers require the protocol stack operating system to call each layer’s processing function, which certainly requires a certain processing time. Moreover, once a newly created task triggered, it would not be executed at once without satisfying the following conditions:

a. The task being performed is completed;
b. There is no higher priority task than the current one;
c. The arrival of a time slice (the operating system must wait for a time slice before inquiring whether a new task has been generated).

This allows the processing time inside the protocol stack to have a random value $\tau_{ns}$ when sending a data packet. The protocol stack receiving data is the inverse process of sending data, and its processing time inside the protocol stack also has a certain randomness. Therefore, $\tau_{ns}$ can be decomposed into random components $\tau_{ns}$ and deterministic components $\tau_{ns}$:

$$
\tau_{ns} = \tau_{ns} + \tau_{ns}, \quad i=1,2
$$

(2)

In summary, according to formula (1), the regular pattern of delay can be analyzed. But due to its randomness, it is difficult to obtain its complete characteristics just through theoretical analysis. At the same time, considering that simulation experiments are often hard to describe various factors which
are willing to appear in practical applications, this paper will study the ZigBee network-induced delay problem from experimental measurement.

3. Network-Induced Delay Measurement Method

According to the characteristics of ZigBee network-induced delay, the following three experimental measurement methods are proposed: Clock synchronization method, confirm mechanism method, and online estimation method, and its characteristics are analyzed.

3.1 Clock Synchronization Method

The Clock synchronization method first synchronizes the clocks of all nodes in the network. The source node records the local time before sending the data packet and packages the time (also called the time stamp) in the transmitted data packet. After receiving the data packet, the receiving node immediately reads the current local time and subtracts the time stamp to obtain the network-induced delay. This method measures sensor-to-controller network-induced delay \( \tau \) of WiNCS. Due to the physical decentralization of ZigBee network nodes, the system cannot provide a unified global clock for each independent node, and each node maintains their own local clock. Since the speed and operating environment of these local clocks cannot be completely consistent, a time synchronization operation must be performed. To this end, ZigBee network nodes can use the master-slave clock synchronization method and the broadcasting clock synchronization method to achieve synchronization.

3.2 Master-slave clock synchronization method

In this method, the master node sends clock information to each slave node in the network to keep the clocks of the nodes in the network synchronized. The master-slave clock information's transmission and reception sequences are shown in Figure 2. The master node records the local time \( t_1 \) at the moment \( t_1 \) and packages it in the clock information to be transmitted. It transmits to the slave node in the network at the moment \( t_2 \), and the slave node receives the clock information at the moment \( t_3 \). Then the slave node adjusts its local time to \( t_4 = t_3 + \tau \) at the moment \( t_4 \). To eliminate the effect of the delay between the transmission and reception of clock information on the accuracy of clock synchronization, \( \tau \) should be

\[ \tau = t_4 - t_1 = t_3 + \tau_1 + \tau_2 + \tau_3 \]

In the formula (3), \( \tau_1 \) is the time difference between \( t_1 \) and \( t_2 \); \( \tau_2 \) is the time difference between \( t_3 \) and \( t_4 \). \( \tau_1 \) and \( \tau_2 \) are determined by the program code length and CPU execution speed; \( \tau_3 \) is the time difference between \( t_2 \) and \( t_3 \), that is, the network-induced delay.

\( \tau_1 \) and \( \tau_2 \) are generally small and negligible, so only the network-induced delay \( \tau_2 \) affects the accuracy of clock synchronization.

There are uncertainties \( \tau_1 \), \( \tau_2 \), \( \tau_3 \), and \( \tau_4 \) in the ZigBee network, so it is impossible to pre-acquire \( \tau_2 \) accurately. Since \( \tau_1 \), \( \tau_2 \), and \( \tau_3 \) are difficult to estimate, \( \tau_2 \) is often replaced by \( \tau = \tau_1 + \tau_2 + \tau_3 \), that is, the slave node adjusts its local clock to \( t + \tau \) at the moment \( t_4 \). According to the above analysis, assuming that the master node clock is the ideal clock, the upper bound of \( \tau_1 + \tau_2 \) is \( \delta_1 \), the upper bound of \( \tau_1 + \tau_2 + \tau_3 \) is \( \delta_2 \), then the deviation \( e \) between the slave clock and the master clock meets the following equation after the master-slave clock is adjusted.

\[ \rho T + e \leq \delta_1 + \delta_2 + \rho T \]

Where \( \rho \) is the clock drift rate of the slave node; \( T \) is the clock information transmission period.

Broadcasting synchronize clock method

This method uses a node (reference node) that does not require clock synchronization in the network to broadcast clock information to synchronize the network nodes. The reference node records
the local time \( t \) before broadcasting the clock information and packages it into the clock information, then broadcasts it to other nodes in the network. Other nodes initialize the local clock immediately after receiving the clock information. The broadcast clock information transmission and reception timing is shown in Fig. 3. In the figure, \( t_1 \) and \( t_2 \) are respectively the time when the i-th synchronization node receives the clock information and adjusts the local time, and the meaning of \( t_1 \) and \( t_2 \) is the same as that of Fig. 2.

![Figure 2 master-slave clock synchronization timing diagram](image)

**Figure 2** master-slave clock synchronization timing diagram

![Figure 3 broadcasting clock synchronization timing diagram](image)

**Figure 3** broadcasting clock synchronization timing diagram

As with the master-slave clock synchronization method, the clock synchronization accuracy of the broadcasting clock synchronization method is also related to \( \tau \) in the equation (3). If the node that needs clock synchronization can receive the broadcast clock information at the same time, that is, \( \tau \) is equal, the clock synchronization accuracy of the method depends only on the clock precision of the node itself. If the topology is simple enough and the influence of the communication distance is neglected, it can be considered that the broadcast clock signal can reach the physical port of each node in the network at the same time, but the application layer of each node does not necessarily receive the broadcast information at the same time, which is related to \( \tau_2 \) and \( \tau_3 \). According to the above analysis, after the broadcast clock is synchronized, the clock deviation \( \varepsilon \) between the nodes to be synchronized satisfies the following formula.

\[
\rho T \leq \varepsilon \leq \delta_2 + \rho T 
\]

(5)

In the formula (5), each symbol has the same meaning as in the formula (4). Comparing equations (5) and (4), it can be seen that the upper limit of the clock bias generated by the broadcasting clock synchronization method is less than that of the master-slave clock synchronization method, which is the advantage of the broadcasting clock synchronization method. However, this method requires an additional reference node used to calibrate the clock. The above two methods often need to periodically transmit clock information due to the accumulated error caused by the differences in clocks among nodes. In order to solve the deviation among the node clocks themselves (clock drift), the drift rate between the node clock and the ideal clock can be measured offline and then compensated by the program. Later experiments will prove that clock drift has a large impact on the delay experiment results. In theory, if the clock drift is properly compensated, the two methods may not need to periodically transmit the clock information. \( \tau_1 \) and \( \tau_2 \) are generally too little to affect the clock synchronization precision, just about one time slice. In order to limit \( \tau_1 \) and \( \tau_2 \), before the clock synchronization completed, other nodes in the network are prohibited from initiating communication except for the node that sends the clock information, so \( \tau_1 \) and \( \tau_2 \) can be ignored.

### 3.3 Confirmation mechanism method

The principle of Confirmation mechanism method: The source node’s application layer first records the local time \( t \) before sending the data packet. After receiving the confirmation information of the destination node, the local time \( t_1 \) is recorded again, so \( t_1 - t_0 \) is the network-induced delay. This method can measure the controller-to-actuator network-induced delay \( \tau_{ca} \) of WiNCS.
It is true that the use of the method is related to the confirmation mechanism of the ZigBee protocol. In the ZigBee protocol, for non-broadcast information, there are two types of Confirmation information: end-to-end confirmation (APS ACK) and single hop confirmation (MAC ACK). The MAC ACK is the default. As long as there is data transmission between the two nodes, there will be a MAC ACK, and the MAC ACK is completed in hardware. The APS ACK is an confirmation message sent from the destination node to the source node at the APS layer, and it is possible to select whether or not an APS ACK is required in the transmission function. Therefore, the ZigBee network's confirmation mechanism can be divided into three types: application layer confirmation method, MAC layer confirmation method and APS layer confirmation method.

1) Application layer confirmation method

The source node application layer records the local time $t_1$ before sending the data frame. After receiving the data frame, the destination node application layer feeds the data back to the source node. The source node application layer records the local time $t_2$ immediately after receiving the feedback data. The delay is one-half of the difference between $t_2$ and $t_1$. The application layer confirmation method is very accurate and is not only suitable for ZigBee networks, but also for other networks including CAN networks.

2) MAC layer confirmation method

For ZigBee network, if a node receives a data frame whose destination address is its own address, the hardware will automatically generate a MAC ACK acknowledgement frame, and the MAC ACK acknowledgement frame is sent to the acknowledged frame immediately, without using the CSMA/CA mechanism to contend for the channel. Therefore, the MAC ACK takes less time, and the delay measurement result is directly replaced with a smaller error.

3) APS layer confirmation method

In the case of multi-hop routing, APS layer acknowledgment method must be used to know when the packet arrives at the destination node. When the data is transmitted by APS layer confirmation method, it needs to be set in the transmit function. After receiving the data frame, the destination node will send an ACK acknowledgement frame to the destination node at the APS layer. The APS layer ACK takes a long time, and the error is big when the delay measurement result is replaced by $t_2 - t_1$. The method of correcting the error by the experimental method is given in section 3.2 of this paper.

3.4 Online estimation method

Clock synchronization method requires clock synchronization information to be transmitted in the network and a time stamp is attached to the packet. The confirmation mechanism method requires the destination node to return confirmation information. These two methods increase the network burden and require other nodes to cooperate. The online estimation method only uses the internal resource calculation of the node can measure the sensor-to-controller network-induced delay $\tau_w$ of the WiNCS without occupying network resources and the participation of other nodes in the network. Combined with the data transmission sequence diagram shown in Figure 4, the detailed implementation process of the entire online delay estimation method is as follows:

![Figure 4 Online estimation method data transmission timing diagram](image-url)
The source node periodically sends a packet to the destination node in the period of h. Suppose the destination node receives the kth data packet is $t_k$ and the time when the k+1th data packet is received is $t_{k+1}$. Assuming that the network-induced delay of the kth packet is $\tau_k$, then the network-induced delay of the k+1th packet $\tau_{k+1}$ can be expressed by

$$\tau_{k+1} = \tau_k + (t_{k+1} - t_k) - h$$  \hspace{1cm} (6)

The network-induced delay of the kth data frame is not predictable. Suppose $\tau_k = 0$, the result calculated according to equation (6) may be non-negative or negative. Assuming that the result is non-negative, the network delay of the k+1th data frame can be directly expressed by equation (6), and the network-induced delay of the k+1th data frame is greater than or equal to the kth data frame. Assuming that the result is a negative value, the network-induced delay of the k+1th data frame is 0, and the delay of the k+1th data frame is less than the delay of the kth data frame. According to this recursive algorithm, as long as the shortest network delay occurs during the actual transmission, the network-induced delay calculated later is non-negative. If the shortest network-induced delay is zero, the network-induced delay calculated after the shortest delay occurs is the actual delay. However, in network transmission, the shortest delay cannot be zero, so the final result needs to be added with the shortest delay to be the real delay.

According to the above description, assuming that the shortest network delay is $\tau_1$, the recursive algorithm in the destination node can be performed as follows:

a. Record the time $t_1$ when the first packet arrives, and make the network-induced delay of the first packet $\tau_1$;

b. The network-induced delay of the subsequent data packet is calculated according to equation (6). If the result is bigger than or equal to $\tau_1$, the network delay is taken as the calculation result of equation (6). If the result is less than $\tau_1$, the network delay is taken as $\tau_1$.

According to the above estimation method, as long as the data is transmitted with the minimum network-induced delay, the estimation result of the subsequent data is the actual delay.

4. Experimental analysis of network-induced delay measurement method

This section uses the measurement method proposed in Section 2 to test the network-induced delay on the ZigBee-based WiNCS data transmission characteristic test platform and analyzes the accuracy of the measurement method.

4.1 ZigBee-based WiNCS data transmission characteristic test platform

The test platform is shown in Figure 5, consisting of a computer, a probe node and four ZigBee network nodes. Node 1 (configured as a coordinator that emulates controllers' function) and three other nodes (configured as routers or terminal devices that emulate sensor and actuator's functions) form a star network. The computer can communicate with node 1 through the serial port to issue experimental commands and receive experimental data. The computer can also cooperate with the Packet Sniffer software and the probe node to monitor the data transmission in the ZigBee network without causing interference to the ZigBee network [7]. The ZigBee network node uses the CC2530 chip as the core with integrated processor module and wireless communication module function and uses TI's IEEE 802.15.4 and ZigBee alliance standard communication protocol stack Z-Stack.
4.2 Network-induced delay measurement experiment

The structure of the network-induced delay measurement experimental is shown in Figure 5. Node 1 and Node 2 perform point-to-point communication. Because the application layer confirmation method measures the point-to-point network-induced delay very accurately, this section will first use this method to measure the delays of packets with different valid data lengths. Then, based on the results obtained by the method, the measurement errors of the MAC layer confirmation method and the APS layer confirmation method, and the accuracy of the clock synchronization method and the online estimation method are analyzed.

1) Confirmation mechanism experiment

In this experimental environment, the maximum data length that the application layer can send to the MAC layer is 85 bytes. The effective data length of the application layer is 5, 25, 45, 65, and 85 bytes respectively. The network delay results measured by the application layer confirmation method, the MAC layer confirmation method and the APS confirmation method are shown in Table 1.

| Effective data (byte) | 5   | 25  | 45  | 65  | 85  |
|-----------------------|-----|-----|-----|-----|-----|
| application layer     | 9.34| 10.28| 11.21| 12.18| 13.12|
| MAC                   | 9.43| 10.26| 11.1 | 11.94| 12.96|
| APS                   | 17.75| 18.64| 19.51| 20.38| 21.49|

From the experimental results of the application layer confirmation method, the transmission delay occupies a small portion of the ZigBee network-induced delay (the maximum packet length that the ZigBee physical layer can transmit is 127 bytes and the delay of the 127-byte wireless data transmitted over the network is only 4.064 ms). The network-induced delay is linear with the effective data length. For every 20 bytes of valid data of the application layer, the network-induced delay is increased by about 0.95 ms, however, the data transmission delay of 20 bytes is only 0.64 ms. It can be seen that the increasing data length will result in the delay increasing significantly.

Table 2: Measurement error of APS confirmation method under different effective data lengths

| Effective data (byte) | 5   | 25  | 45  | 65  | 85  |
|-----------------------|-----|-----|-----|-----|-----|
| Delay error (ms)      | 8.4 | 8.36| 8.51| 8.2 | 8.37|

The result of MAC layer confirmation method and application layer confirmation method is similar, not exceeding 0.2 ms. In general, it can be used without compensation. Based on the application layer confirmation method, the network-induced delay measurement error measured by APS layer confirmation method is shown in Table 2. The error is large, and the error average is 8.33 ms. Therefore, if the network-induced delay is measured by APS layer confirmation method (usually used...
in the case of multi-hop transmission), the measured network-induced delay must be subtracted by 8.33 ms for each additional route to obtain a higher measurement error.

2) The experiment of clock synchronization method

Because the broadcasting clock synchronization method has less error than the master-slave clock synchronization method in the ZigBee network, this section selects the broadcasting clock synchronization method to measure the network-induced delay and sets the node 3 as the reference node. The effective data length of the application layer is 40 bytes. Without the clock drift compensation, the measured results of the network-induced delay are shown in Figure 6.

![Figure 6 Clock method for measuring network-induced delay without clock compensation](image)

Figure 6 Clock method for measuring network-induced delay without clock compensation

It can be seen from FIG. 6 that since the local clocks of the transmitting and receiving nodes are not completely consistent, the network-induced delay measurement result increases linearly, and thus the local clock of the receiving node is faster than the transmitting node. According to the average slope of the delay rise in FIG. 6, the clock drift rate of the receiving node relative to the transmitting node can be calculated as $1.7 \times 10^{-5}$, and the measured result of the delay after the clock drift of the receiving node is compensated by software is as shown in FIG. 7.

![Figure 7 Clock synchronization method for measuring network-induced delay with clock compensation](image)

Figure 7 Clock synchronization method for measuring network-induced delay with clock compensation

It can be seen from Fig. 7 that the result of clock drift compensation tends to be stable. The average of network-induced delays is 11.02 ms and the error is less than 0.2%. Therefore, it can be seen that Clock synchronization method has very high precision with multiple transmission method and clock compensation. The clock drift rate measured in this experiment is on the order of 10-5, which also demonstrates the conclusion of the literature [8] on the order of magnitude of the clock drift rate.

3) The experiment of Online estimation method

The online estimation method needs to know the minimum delay of packet transmission of a certain length, which can be measured by the aforementioned confirmation mechanism method. The online estimation recursive (6) has a transmission period $h$, but the clocks of the source node and the destination node may not be completely consistent. The cumulative error of the node clock will lead to a result similar to that shown in Fig. 6. For this reason, the online estimation method is also required Clock drift compensation. The effective data length of the application layer is 40 bytes. The minimum network-induced delay is 10.3 ms measured by the confirmation mechanism method. In the case of clock drift compensation, the average network-induced delay is 11.12 ms and the error is about 1%. It can be seen that the online estimation method also has higher precision with clock drift compensation.
5. Conclusions
In this paper, in order to measure the ZigBee-based WiNCS network-induced delay in real time, the composition of the ZigBee network delay is analyzed, and the general rule of network delay is obtained. Because it is difficult to obtain the complete characteristics of network-induced delay from the theoretical analysis, three experimental measurement methods for ZigBee network-induced delay are proposed and introduced in detail. In order to analyze and compare the measurement accuracy of various measurement methods, a ZigBee-based WiNCS data transmission test platform was constructed. Experimental analysis shows that with some compensation measures, various measurement methods can achieve high measurement accuracy.

6. References
[1] CHEN Jing-yun,ZHOU Xiang-ping. Zigbee Wireless Communication Technology in Industrial Controls[J]. Radio Engineering, 2006, 36(6): 61-64
[2] Nicholas J. Ploplys, Paul A. Kawka, Andrew G. Alleyne. Closed-loop control over wireless networks. IEEE Control Systems Magazine, 2004, 24(1): 58-71
[3] [21] Kyung Chang Lee, Suk Lee. Performance evaluation of switched ethernet for networked control systems [A]. Proceedings of the 28th Annual Conference of the IEEE Industrial Electronics Society [C], Spain. 2002: 3170-3175
[4] Ge Yuan, Chen Qigong, Jiang Ming, Liu Zhen’an. Stability analysis of networked control systems based on hidden Markov models. Chinese Journal of Scientific Instrument, 2008, 29(2): 273-278
[5] Zhang W, Branicky M S, Phillip S M. Stability of networked control systems. IEEE Control Systems Magazine. 2001, 21(1): 84-99
[6] ZigBee Alliance. Http://www.ZigBee.org
[7] Ti. DN034-YAGI 2.4 GHz PCB Antenna [EB/OL]
[8] Rodrigue, L., Guimaraes, M., Rufino, J.. Fault-tolerant clock synchronization in CAN. In Proceedings of the 19th IEEE Real-Time Systems Symposium, 1998: 1594-1607