

Research Article

Research on Production Management and Optimization of Multisensor Intelligent Clothing in 5G Era

Xiang Chen ¹, YouTang Gao,¹ and Bao Tian²

¹Huang Huai University College of LEIZU Fashion Smart Manufacturing, Zhumadian City, 463000 Henan, China

²School of Intelligent Engineering, Zhengzhou University of Aeronautics, Zhengzhou Henan 450000, China

Correspondence should be addressed to Xiang Chen; chen.xiang@huanghuai.edu.cn

Received 13 September 2021; Accepted 12 November 2021; Published 6 December 2021

Academic Editor: Gengxin Sun

Copyright © 2021 Xiang Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The influence of 5G has penetrated into all aspects of people's lives. The field of garment production management is inevitably affected by 5G. The various advanced technologies it promotes can greatly promote the production and management of clothing. Learning and understanding these technologies can help you learn how to change in the changing garment factory to obtain more intelligent and efficient production methods without being excluded by age. The garment production line management system proposed in this paper is based on the garment production line, introduces Internet technology into the garment production process, and monitors all links of the garment production process through the Zigbee network. The system improves the automation degree of enterprises, greatly expands the application scope of wireless sensor networks, and improves the application level of data acquisition, monitoring, equipment maintenance, and diagnosis in China's industrial field. Wireless network node location technology is also an important supporting technology for managing wireless Zigbee networks. The visual display of a physical topology map can effectively help administrators manage and maintain wireless networks.

1. Introduction

5G is the fifth-generation wireless communication. The advantages of 5G include high speed, low delay, high throughput, and high security. It is impossible to simply understand the upgrade of traditional technology of the communication network. It should be regarded as a technological change with essential and subversive changes in various fields affecting people's production and life. Therefore, we have such a problem [1]. In the 5G era, how can the field of garment production management keep pace with the times and improve its own scientific and technological content? With the elimination of old technology, the field of garment production management is inevitably affected by 5G technology. This technology promotes production, and various advanced technologies are extremely convenient for garment factory production and garment management [2].

In the traditional garment production process, the site control is poor, and the bottleneck of the production line cannot be foreseen. In the process of production and pro-

cessing, a large number of data such as material data, labor time, and output data need to be recorded manually. In addition, when product quality problems occur in the processing process, quality tracking cannot be carried out, and the classification of responsibilities is unclear. Finally, due to the lack of on-site systematic management, it is difficult to estimate the production progress of orders and the data that have been completed in the workshop. The production cost cannot be calculated correctly [3].

The automatic garment production monitoring system is equipped with RFID reader devices in each process and RF tag cards on each garment. When clothes pass through each process, employees can use the cards to observe the current line processing process and the completion of each process in real time and estimate the overall completion time, thus realizing the overall monitoring of the production line [4]. The monitoring process can also count the ratio of each employee and the labor effect in the cycle. These play an important and decisive role in stimulating the enthusiasm of employees. Combined with Internet of Things technology

and garment production management system, it can significantly improve efficiency and reduce costs [5].

A low-cost information collection system is used to manage the distribution of clothing production lines in order to ensure traceability of products throughout the supply chain, which is very important for any product. For the garment production line, the traditional way is to rely on the team leader's personal experience or intuition without relying on data. If the management team leader or workers are not on duty, it will directly affect the production efficiency and damage the production capacity. For such problems, the system uses information collection tools to continuously use the collected data in the supply chain and analyzes the data to display it [6]. In this way, even inexperienced leaders can use the data provided for management purposes. Research shows that team change can be achieved by assignment. With the development of science and technology, various new technologies are also applied in all walks of life, and nanotechnology is also applied to the clothing industry. The characteristics of nanomaterials determine that they can play a role in promoting and innovating the clothing industry. We summarized the nanodispersion theory by searching literature and then, combined with experiments, put forward the application of nanomaterials in safe and intelligent clothing design [7].

Military protective clothing is often threatened or destroyed by chemical components, coupled with a series of shortcomings such as its own bulkiness, so its practicality has been of wide concern. Researchers made graphene electronic fabric by laminating graphene, which is light, durable, and scalable. In addition, graphene intelligent protective clothing is provided by configuring a graphene triboelectric nanogenerator [8]. Through experiments, it is found that the clothing has high sensitivity to chemical warfare agents. Nowadays, with the rapid development of digitalization, many traditional industries have begun to change to digital industries, such as the clothing industry. Everyone is competing to launch the digital customization platform for clothing, but it is still in the exploratory stage, and the satisfaction is not very high in the actual use process. In fact, it is mainly caused by the low consumer participation and poor communication of the platform. To solve this problem, we first analyze the psychological needs of customers and then optimize and transform them in combination with the construction of the sales platform and design an intelligent interactive platform for personalized clothing customization [9]. Young people are exposed to more and more diversified things, and they also have a higher personalized pursuit of clothing. In order to meet the needs, we designed an intelligent clothing matching selection system, that is, a personalized clothing recommendation solution based on a self-organizing mapping SOM neural network. The system combines the collected personalized demand information with the objective elements of customers and then establishes a database to mine personal preferences. By using this system, we can provide customers with relevant reference for clothing [10]. Experiments show that the system has good accuracy and performance and has great development potential.

2. Detailed Design and Implementation of Each Module of the System

2.1. Grid Probabilistic Positioning Algorithm. The biggest feature of the DV-Hop localization algorithm is that it does not need any auxiliary hardware equipment, but the algorithm itself is not optimized and the localization accuracy of specific network applications is not high. Therefore, the system transforms the node location problem into the problem of identifying the correct location of network nodes in each grid [11].

For $\forall k \subseteq S-A$, we set the unknown node "k" in A and set the minimum number of hops $H^k = (h_1^k, h_2^k, \dots, h_{|A|}^k)$ for each anchor node of A. The unknown node "K" can be expressed as the probability distribution in the grid by the following formula.

$$F_k = \begin{pmatrix} f_{11}^k & \cdots & f_{1n}^k \\ \vdots & \ddots & \vdots \\ f_{n1}^k & \cdots & f_{nn}^k \end{pmatrix}, \quad \forall k \subseteq S-A. \quad (1)$$

The probability of unknown nodes in each grid satisfies the following equation.

$$f_{ij}^k = \begin{cases} \text{Probability that the sensor node "k" is located at the } (i, j) \text{ position,} & (i, j) \notin A, \\ 0, & (i, j) \in A. \end{cases} \quad (2)$$

f_{ij}^k represents the probability that the sensor node "k" is located (i, j) in the grid. The calculation method of f_{ij}^k itself is expressed in equation (3). Of course, if the location (i, j) is the location of the anchor node, then the probability that the unknown node is in this grid is zero.

$$f_{ij}^k = \prod_{l=1}^{|A|} p_{ij}^{h_l^k}. \quad (3)$$

$p_{ij}^{h_l^k}$ in equation (3) represents the probability that the number of hops from the unknown node "k" to each anchor node is the number of hops represented by the vector H^k . The probability of the Poisson distribution is expressed as the following formula.

$$P_\tau = \frac{\lambda^{\tau-1} e^{-\lambda}}{(\tau-1)!}. \quad (4)$$

In formula (4), $\tau = 1, 2, \dots$ and λ are the hops and distances from the unknown node to the anchored node. Through this calculation, the probability that the unknown node "k" exists in each grid in the message topology can be obtained, in which case, only the location of the maximum probability or the estimated location of the location node "k" is determined.

2.2. Realization of RFID Data Acquisition. Users use RFID readers to collect data in the system. The identification stage includes the operation of finding the card, preventing collision, and operating the card. Authentication provides the password of the selected card through authentication, and the authentication authority of the card provides effective protection of the data in the card. You can start reading and writing the data in the card through authentication [12].

Figure 1 shows a flow of an RFID reader operation representing an implementation function of a corresponding function. Under normal circumstances, after receiving the card reading instruction, the card reader will first look for the card operation and then send a radio frequency signal to look for the response card. If a plurality of radio frequency tags exist within the identification range of the card reader, a plurality of cards respond to a card search signal sent by the card reader. At this time, the card reader adopts anticollision protocol. One of the many cards found to operate is selected. If a card is selected, we check whether we have access to the sectors that need to be read and written. The contents of the card are read and written only after passing the verification. After the final operation, the final operation card is needed. In this case, when the card cannot leave due to the test system, if the data in the card can be read again, the card reader needs to be reset.

Because the macroprocessor is single-column red, the idea of the main function design of the system is whether various tag bits polled continuously in the dead cycle are already located in it, and after initialization, they enter the dead cycle and become the state where their respective tags are polled. As you can see from the loop, if there is no identification location, the job of the system is to find the card. In the operation of the card, the buzzer prompts the user with a prompt sound of success or warning according to the operation result in Figure 2.

The system has two kinds of tag cards. One is the employee card assigned to the employee. The other is a card distributed by clothing. We call it the “packing card.” When the card reader reads different types of cards, the screen will display different contents. The acquisition terminal reads data from the card, first judges the type of the card, and makes different responses to different types of the card.

The card operation trigger event is when the reader reads the legitimate tags belonging to the system, that is, the above two tags. In order to complete a friendly interaction with users, prompt information will be displayed on the LCD screen when operating the card. At the same time, if external instructions are needed in operation, keyboard operation will be involved. There is no mobile banking on the monitor. The word touch code of the word to be displayed is stored in an array in advance. It is just read directly when necessary. Keyboard operations are performed through external interruptions.

In order to count the accounting of employees, employees should first use their own employee cards before using the packaging cards. Therefore, when the card reader identifies the card, it must first determine whether the employee card is used in the current system. If there is an employee card recorded on the system, it will show the code

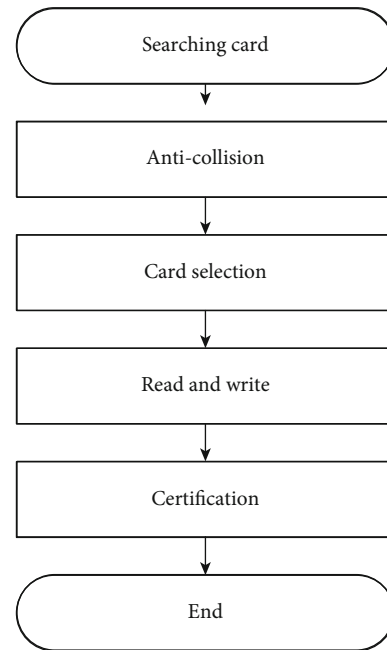


FIGURE 1: RFID reader operation flow.

of the current operator. Otherwise, the monitor will swipe the employee card first and then finish reading the card. If the unregistered card is detected, the LCD panel will also pay attention to the corresponding operation. Normally, if the confirmation card is an employee card, the current operator is prompted and the current employee number stored in the local cache is updated in Figure 3.

In order to improve CPU utilization, the macroprocessor uses the interrupt mechanism, because the macroprocessor operates in single-column red. Interruptions used during system implementation will interrupt timer interruptions and serial data reception. Timer interruption is mainly used for timing display. For example, the information of the current user will be automatically displayed after a certain period of time after swiping the card. Reception of serial data is interrupted. This is mainly used for receiving serial data. After receiving the serial data, the Serial Port Receive Interrupt Service Handling Subroutine is executed if the location of the tag is detected.

The crosstalk interrupt handler is responsible for receiving serial data. Crosstalk is received per byte, and the register flag bit RI is set to 1. The received data is set on the register SBUF. Therefore, our job is to set the received data into an arrangement. The process flow is shown in Figure 4.

If the register RI is set to 1, it indicates that new data has arrived at the serial port, where the data is received in byte form, and it is first determined whether the received byte is a frame header, and if it is the auxiliary flag bit Frame Start, it is determined whether the current byte is a data frame header or a frame trailer. Because the header and trailer use the same recognition, for Frame Start recognition, if 0 is the header, 1 represents the trailer. The flag position is set to 1 after the initial reception of the frame header, and if 0xC0 is encountered again after the

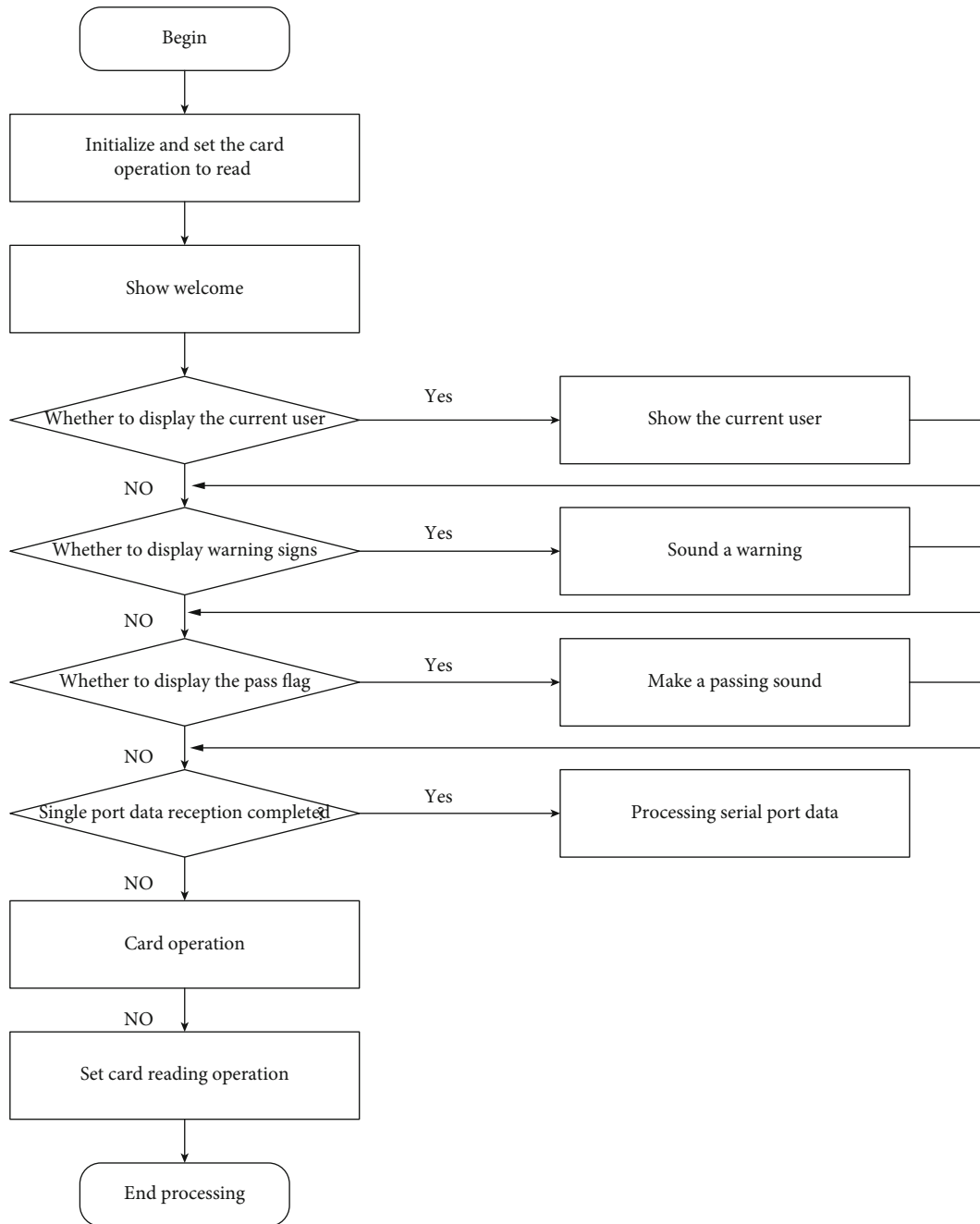


FIGURE 2: Main function execution flow chart.

reception of data, the current byte can be determined to be the last of the frame.

2.3. Design and Implementation of Reliable Communication in Zigbee. Zigbee is a wireless network protocol with low speed and short distance transmission. The bottom layer is the media access layer and physical layer which adopt the IEEE 802.15.4 standard specification. The main features are low speed, low power consumption, low cost, support for a large number of network nodes, support for a variety of network topologies, low complexity, fast, reliable, and safe. Zigbee is a new wireless communication technology, which is

suitable for a series of electronic components and devices with short transmission range and low data transmission rate. It is a wireless network technology with short distance, low cost, and low power consumption, and it is also a technology suitable for star network topology. The Zigbee network transport software module sends data to the trusted module via crosstalk. The coordinator module is physically connected with the ARM9 gateway, and the main logic is implemented on the Linux platform, but the coordinator itself is only responsible for a data transceiver. There is not much processing logic [13]. Then this time, only the program of the Zigbee terminal is introduced.

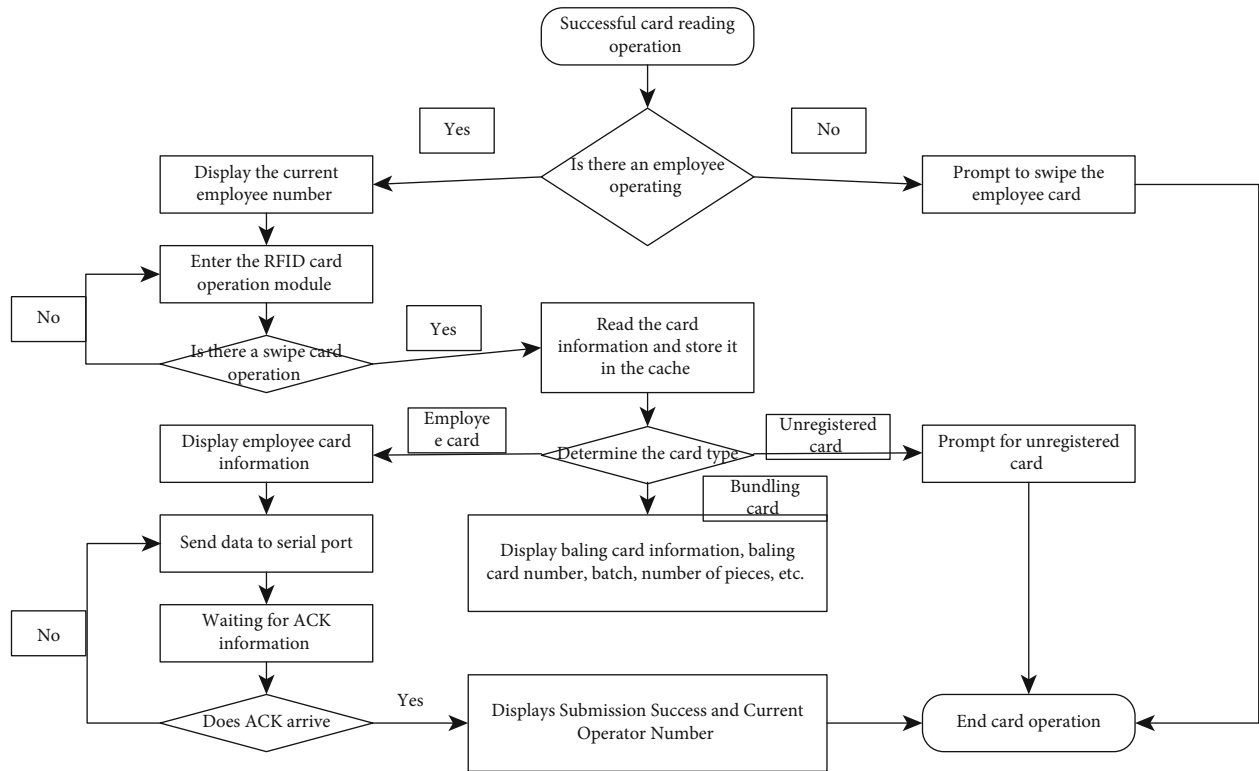


FIGURE 3: Main function execution flow chart.

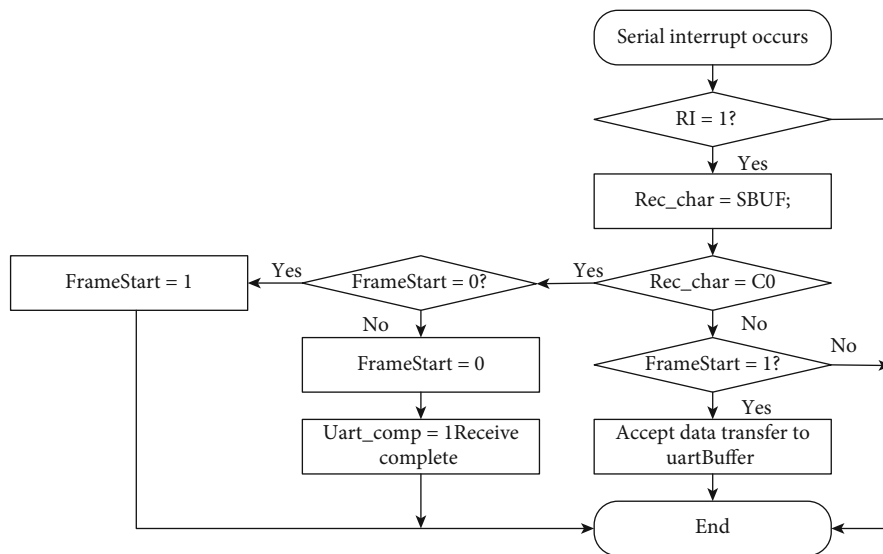


FIGURE 4: Flow chart of serial interrupt service program processing.

The whole Zigbee programming process is based on the model of event processing, which promotes the execution of the whole program through events. Figure 5 is a state switching diagram of the state machine. The coordinator node selects whether to receive data packets according to the report sequence number. The gateway node does not receive error messages. After the sequence number is reset, the state machine is in state 9 (trans_state_t_9), where tasks performed by the system send heartbeat packet messages to

the gateway every 10 seconds. When the RFID terminal transmits a message to the Zigbee terminal, the Zigbee terminal determines the correctness of the message, modifies its format, generates a newspaper, inserts the newspaper into a waiting transmission matrix, and generates an event j (trans_state_convert_arc_t_j). When the event exists, the modified report is sent to the gateway together with the sequence number of the current state machine, and a timer is started to set the receiving waiting time of the ACK. If

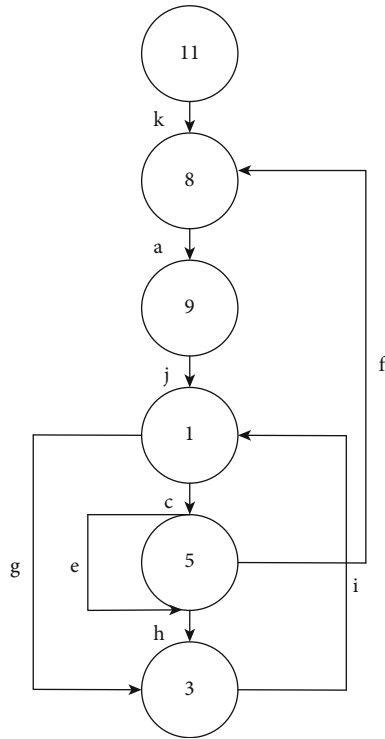


FIGURE 5: Implementation flow chart of Zigbee terminal based on state machine.

the Zigbee terminal does not receive an ACK within the specified time, we believe that an error occurred in the process of sending the newspaper. This report was not sent and needs to be sent again. After the newspaper is sent, according to whether the confirmation letter is received within the specified time, the system enters state 3 (*trans_state_t_3*) and state 5 (*trans_state_t_5*), respectively, indicating that the system enters the retransmission stage in state 5. If the retransmission times exceed the threshold set by the system, the event of resetting the serial number will occur, and the system enters state 8. In this case, the network failure or the serial number may be unsatisfactory in Tables 1 and 2.

According to the state machine, a normal state switching sequence, a state switching sequence when the transmission timeout is less than a threshold value, and a state sequence when the system fails or the sequence number is wrong are provided.

- (1) *Transmission of messages under normal circumstances.* The usual sequence message transmission is $k \rightarrow a \rightarrow b \rightarrow j \rightarrow g \rightarrow i \rightarrow g \rightarrow i \rightarrow \dots$; that is, the program states are 11, 8, 9, 1, and 3, and it switches cyclically between state 1 and state 3. States 11, 8, and 9 are the state machine startup and initialization processes.
- (2) *Temporary congestion on the network causes some messages to be lost.* In this case, some packets will be delayed or lost in the system network, and in this case, some packets may be transmitted repeatedly. At this point, the reported transport sequence is $k \rightarrow a$

$\rightarrow b \rightarrow j \rightarrow c \rightarrow e \rightarrow e \rightarrow \dots \rightarrow e \rightarrow h \rightarrow i \rightarrow c \rightarrow g \rightarrow i$
 \dots

- (3) *Network failure or message disorder.* In this case, the sequence of states corresponding to the above-mentioned switching graph is $k \rightarrow a \rightarrow b \rightarrow j \rightarrow c \rightarrow e \rightarrow e \rightarrow \dots \rightarrow e \rightarrow f \rightarrow a \rightarrow b$.

2.4. Implementation of Multiprotocol Conversion Gateway Module. The ARM9 gateway is the function of protocol conversion of the whole system. It is a service process that transforms the original data of the wireless sensor network collected by the Zigbee coordinator into TCP/IP data and transmits it to the background server. The overall workflow of the gateway is shown in Figure 6.

The three major functions of the gateway program are to obtain serial data, the boundary of the original data, and the repackaging of the data. Each function is shown in Table 3.

2.4.1. Obtain Serial Port Data. The gateway acquires raw data frames of undefined boundaries transmitted by the coordinator via crosstalk and stores them in the raw data queue, where each complete data frame is initiated by a delimiter 0xFE and the delimiter is a boundary that can be used to set the data frames. It is the thread *uart_rcv_loop* that gets the thread to receive the serial message. The thread execution flow chart is shown in Figure 7.

The data read from the serial is the original byte stream. The program checks the length of the data received from the queue. If the data is complete, the data is saved to the queue. After the data is lost, the current operation is recorded in the error log. Each queue has a length. After reaching a certain length, the queue data is written as a buffer memory.

2.4.2. Data Delimitation and Encapsulation. The data boundary separates mixed data frames and discards incomplete data frames. The thread that completes this operation is *package_analysis*. The thread task extracts packets from the original data *g_rawdata_queue*, determines each packet, and reencapsulates each processed data frame into a packet in a new format, thus inserting it into the queue *g_package_queue*, waiting for the network thread to be sent for processing. The process diagram is shown in Figure 8.

The thread polls the original packet queue *g_rawdata_queue*, and if the queue is not empty, the data is fetched at the head of the queue. In the extracted queue array, the read state is set to be ready to receive the frame header, and the frame header identification 0xfe is searched at this time; after finding the frame header, the data reading state is set to prepare for receiving the frame length, and then, the frame length byte is read. The specific processing flow chart is shown in Figure 9.

The original packet queue *g_rawdata_queue* in the thread extracts data from the column header if the column is not empty. The data extracted at a time may include a plurality of raw packets each starting with 0xff so that the data extracted from the queue is treated as a byte stream. In the extracted queue arrangement, the read state is first set to be ready to receive the frame header, and the frame header

TABLE 1: Definition and description of state in system.

Status	Abbreviation	Meaning
trans_state_t_1	1	The data transmission request is received
trans_state_t_3	3	ACK arrival
trans_state_t_5	5	The retransmitted data frame is in a timeout state
trans_state_t_8	8	The state in which the reset sequence number frame request is sent after receiving other states
trans_state_t_9	9	Reset the status of sequence frames after successful transmission
trans_state_t_11	11	Restart

TABLE 2: Definition and description of events in the system.

Events	Abbreviation	Meaning
trans_state_convert_arc_t_c	c	The message was sent out of time for the first time
trans_state_convert_arc_t_g	g	The message sent receives ACK within a given time
trans_state_convert_arc_t_i	i	After this message is successfully sent, the next message will be sent
trans_state_convert_arc_t_e	e	The message is sent out of time for the second time or more, but it is less than the maximum number of transmissions
trans_state_convert_arc_t_f	f	The timeout times of message sending are greater than the maximum transmission times; request to send a reset sequence number frame
trans_state_convert_arc_t_h	h	Receive ACK within a given time when the message is sent again after timeout
trans_state_convert_arc_t_a	a	Sending a reset sequence frame to the coordinator
trans_state_convert_arc_t_j	j	Start sending messages
trans_state_convert_arc_t_k	K	State machine initialization, requiring a reset frame to be sent

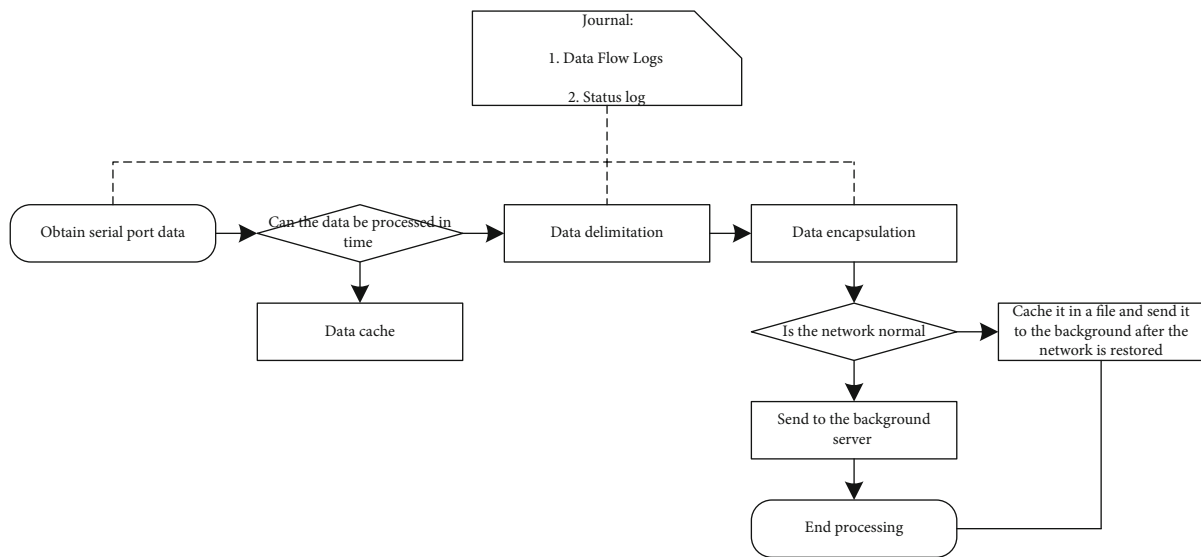


FIGURE 6: Business flow chart of gateway program.

identification 0xfe is sought at this time. After finding the frame header, the data reading state is set, the frame length is prepared to be received, and then, the frame length byte

is read out. The frame content is read from the frame length byte. Finally, format conversion is carried out, and the conversion process is the process of data reencapsulation. The

TABLE 3: Function description of each thread in gateway programming.

Thread function name	Functional description
uart_rcv_loop	Processing data received from the coordinator
package_analyse	Delimitation and reencapsulation of original data packets
network_loop	Packet sending
pingtest_loop	Test the connection between the gateway and the server

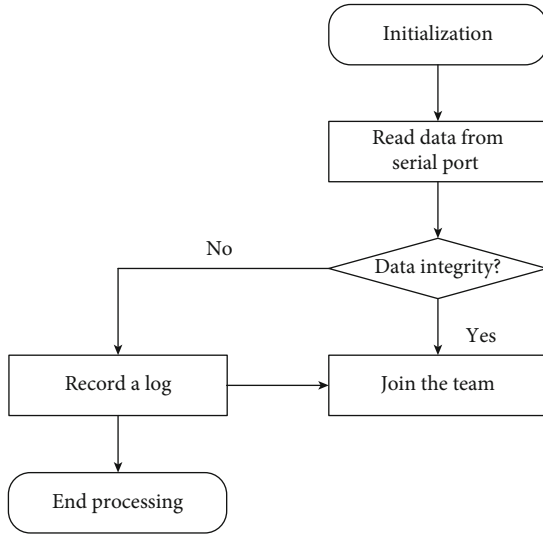


FIGURE 7: Flow chart of obtaining serial data thread execution.

processed new packet is inserted into the packet (g_package_queue). If the original queue is empty, the thread will go to sleep at intervals. The specific processing flow chart is shown in Figure 9.

2.4.3. Packet Sending. Packet sending sends the packet data to the background service program through the TCP/IP protocol. The thread of this process is the network_loop. A schematic diagram of the processing of this thread is shown in Figure 10 which is a flow chart of the design of a packet sending thread.

The network sending thread first creates a socket connection and sends a connection request to the server. If the request is successful, it reads the encapsulated packet queue and sends the data to the background server via the network. If this process fails, the socket is closed and the above process is reexecuted. When an error exception occurs during an operation, it is recorded in the error exception log.

2.4.4. Log Submodule. In order to improve the detection possibility of the system, the gateway module uses the log detection unit and now maintains the data flow log and the status log. By observing the data flow log, you can know the action details of the gateway. The status log is used to confirm whether the gateway is faulty in Figure 11.

2.5. File Cache Submodule. The gateway module uses two cache files in the storage of sequence data and the packaged

data storage. In the gateway module, the upper limit of the queue size is set to 1024, and when the data volume exceeds 1024, it will be cached by external files. The logic of file caching is as follows:

- (a) After processing the data in the queue (assuming there are N columns left blank at that time), you need to confirm the cache file and save N data at the end of the queue
- (b) When new data enters the queue, the cache file is checked. If the cache file is not empty, the new data is saved directly to the cache file. If the cache file is empty, make sure the queue is full. If it is not filled, the new data is queued and filled, and the new data will be temporarily saved in the cache file

3. Design and Analysis of Probability Location Algorithm Based on DV-Hop Grid

This section will describe in detail the combined DV-Hop localization process and grid probabilistic localization algorithm execution process and, according to the specific application scenarios in this paper, improve the algorithm execution of the average hop distance and hop between nodes in the calculation method. Finally, the different factors that affect the bit accuracy of the method are analyzed.

3.1. Calculation of Average Jump Distance. In this part, the method of calculating the average hop distance between nodes by using the uniformity of nodes in the system is improved. In the current hypothesis, the number of nodes in a uniform network is the density of nodes (per square meter), the communication radius between nodes is $R(m)$, and the total number of nodes in a circle has a radius R . The N nodes are arranged in the form of Figure 12, in which adjacent nodes of different layers represent equilateral triangles.

In the figure, there are about 19 nodes in a circle with radius R as the center, and the nodes are divided into two layers. The first layer (from inside to outside) contains 6 nodes, the second layer has 18 nodes, and generally, there are $6m$ nodes in the m layer. The estimated total number of nodes n is suitable for the following equation.

$$n \leq \frac{m(6 + 6m)}{2}. \quad (5)$$

Equation (5) can transform the resulting equation (6) and solve the equation (6) to obtain equation (7). The value

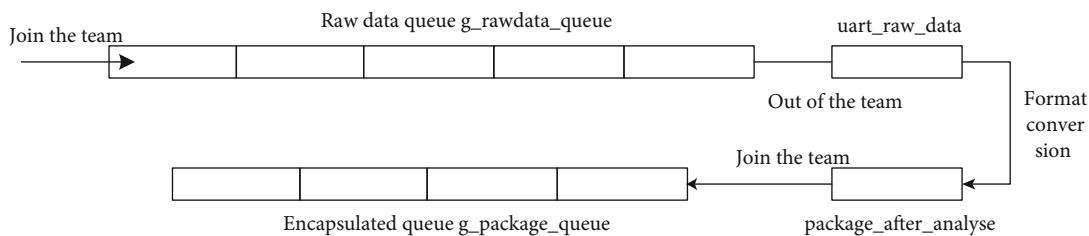


FIGURE 8: Schematic diagram of delimiting and encapsulating processing threads.

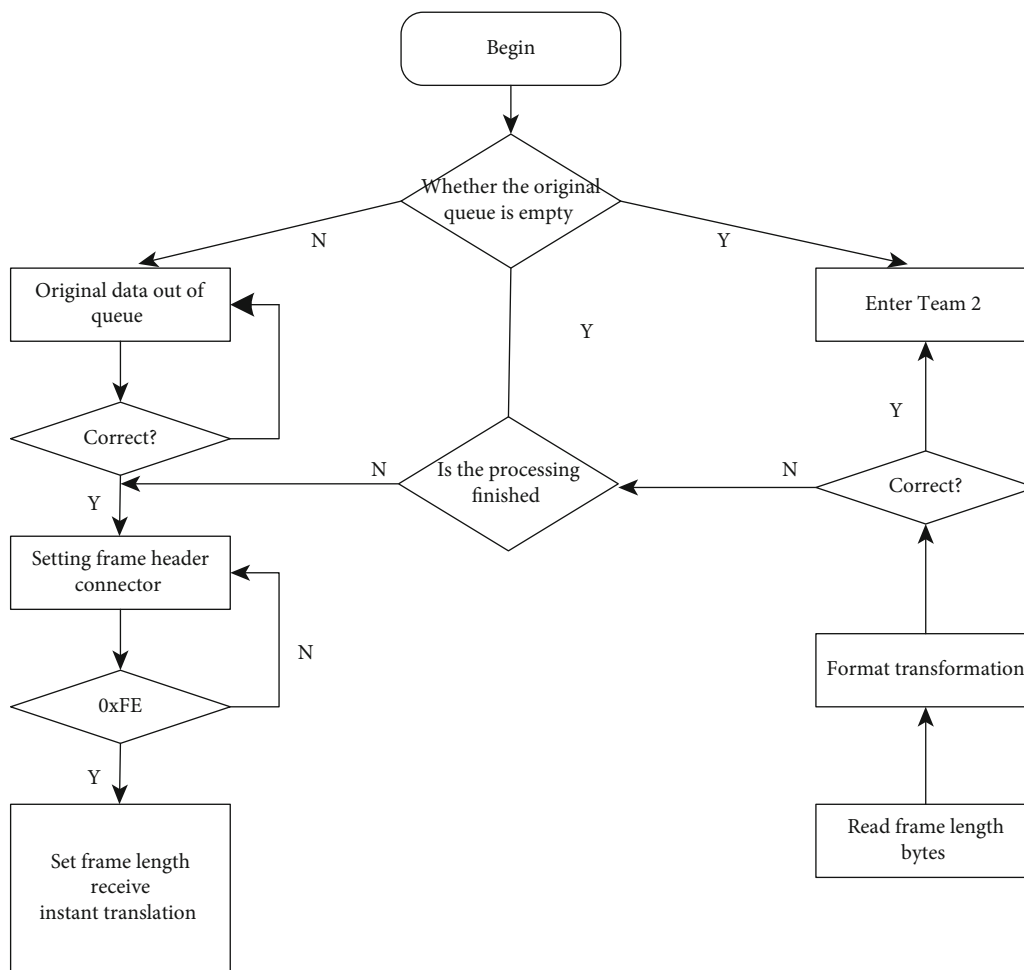


FIGURE 9: Flow chart of frame delimitation processing.

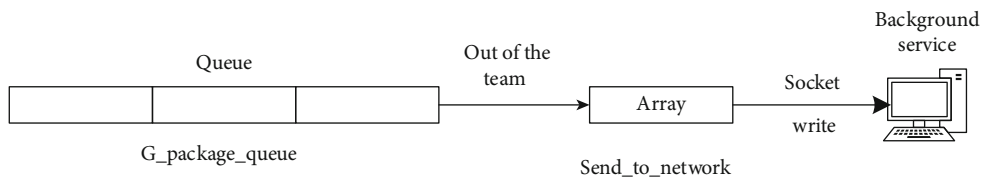


FIGURE 10: Schematic diagram of thread processing of network sending data packets.

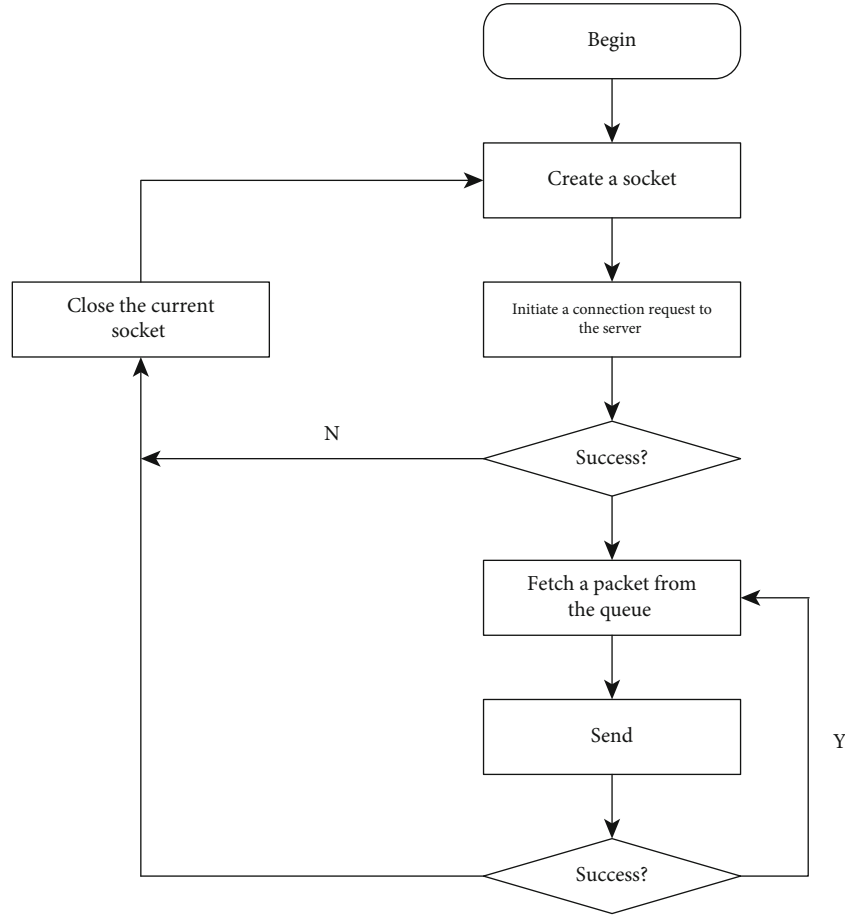


FIGURE 11: Network send thread design flow chart.

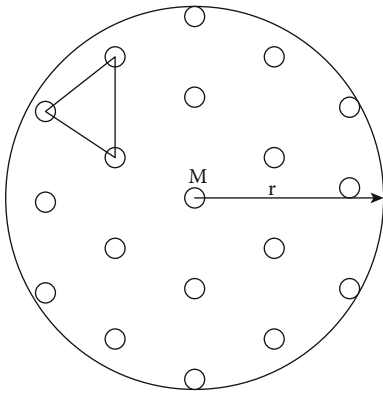


FIGURE 12: Node distribution diagram of single hop network.

of m is greater than 0, and the final value of m is shown in equation (7).

$$m^2 + m - \frac{n}{3} \geq 0, \quad (6)$$

$$m \geq \frac{-1 + \sqrt{1 + 4n/3}}{2} \text{ or } m \leq \frac{-1 - \sqrt{1 + 4n/3}}{2}, \quad (7)$$

$$m \geq \frac{-1 + \sqrt{1 + 4n/3}}{2}. \quad (8)$$

Equation (10) can be obtained by appropriately scaling equation (8). Then, formula (9) is brought in, and the result of taking the minimum value of M to M is shown in formula (11).

$$n = \rho \pi r^2, \quad (9)$$

$$m \geq \sqrt{\frac{n}{3}} - \frac{1}{2}, \quad (10)$$

$$m \approx r\sqrt{\rho} - \frac{1}{2}. \quad (11)$$

From formula (11), it can be calculated that the circular area of n nodes is arranged in about m layers, so the distance between nodes is $H = R/M$. After expansion, it is shown in the following formula.

$$h = \frac{r}{m} \approx \frac{r}{r\sqrt{\rho} - 1/2} = \frac{2}{\sqrt{\rho} - 1/r}. \quad (12)$$

Because the value of h is greater than zero, we take $h = r$ when $\sqrt{\rho} - 1/r \leq 0$. Considering comprehensively, the

average jump distance h between nodes can be obtained as shown in the following equation.

$$h = \begin{cases} \frac{2}{\sqrt{\rho} - 1/r}, & \sqrt{\rho} - \frac{1}{r} > 0, \\ r, & \sqrt{\rho} - \frac{1}{r} \leq 0. \end{cases} \quad (13)$$

Equation (13) is the average hop distance between nodes calculated in the ideal case of uniform arrangement of nodes.

3.2. Algorithm Design Flow. The process of mesh location determination algorithm based on the DV-Hop is similar to the DV-Hop location determination algorithm. The algorithm firstly calculates the minimum hops from each node in the network to the anchor node by broadcasting packets to the anchor node. In the location determination algorithm, each location node of course estimates that the barrier constructed by all nodes is the same [14]. The finishing execution flow of the algorithm is shown in Figure 13.

The calculation formula of parameter input is shown in the following equation.

$$\lambda = \frac{\text{Distance from grid point to anchor node}}{\text{Corrective factor}}. \quad (14)$$

λ in the above formula is the distance between grid point and anchor node in the unit of transmission hops, and the correction factor is calculated by equation (13). $p_{ij}^{h_i^k}$ is found, and $p_{ij}^{h_i^k}$ represents the probability that the unknown node k is located at the location (i, j) and reaches the anchor node through the h_i^k hop.

Formula (14) is the correction coefficient calculated theoretically, and the secondary factor needs to be further adjusted for the specific actual environment. In the next section, when the simulation experiment is carried out in the last paper, the small amplitude correction coefficient is given to obtain the best positioning effect of the system, and the ideal correction coefficient in the simulation environment is obtained under different network scales. Even in a real environment, this parameter must be adjusted to achieve the best positioning effect based on the specific network size and layout environment.

The experiment shows that the transmission hops of an unknown node and anchor node are integers, and the transmission hops have almost 0.5 error. In addition, considering that there is a certain error in obtaining the transmission hop count from the unknown node to the anchor node, the local average hop count is taken in this implementation, that is, the following equation.

$$S_i = \frac{\sum_{j \in n(i)} h_j + h_i}{|n(i)| + 1} - 0.5. \quad (15)$$

$n(i)$ sends an adjacent node having a hop count of 1 from a surrounding node of the unknown node I to the node

I itself. h_i and h_j are the hops of node i , node j , and anchor node, respectively. The formula calculates a local average for transmitting hops from the unknown node I to the anchor node. The hop count from the adjacent node to the anchor node is the reference information for calculating the hop count from the adjacent node to the anchor node. This calculation of hop count from unknown optimized node to anchor node is one of the improved parts based on the DV-Hop positioning algorithm. When the network scale becomes larger, the local average hop count is due to the significant differences between communication models of each part of the network, so it can further reflect the hop count from the location node to the anchor node. At this time, the hop count from the nodes around the node to the anchor node has certain reference significance for the hop count from the node itself to the anchor node.

3.3. Simulation Experiment and Result Analysis. According to the algorithm description provided in the previous section, the influence of grid size and the number of error nodes from two latitudes of the total network ratio on the algorithm error are observed. In this paper, we use formula (12) to calculate the error.

$$\text{error} = \sqrt{(x_{\text{dest}} - x_i)^2 + (y_{\text{dest}} - y_i)^2}, \quad (16)$$

where $(x_{\text{dest}}, y_{\text{dest}})$ is the location coordinates estimated by nodes located at positions (x_i, y_i) .

Experiments show that the positioning accuracy of the DV-Hop-based probabilistic mesh location determination algorithm is directly related to the number of anchor nodes and the size of the network itself. Figure 14 is a diagram showing the change of positioning accuracy related to the number of anchor nodes when the network sizes are 30, 54, and 96, respectively.

It can be seen from the figure that with the increase of the number of anchoring nodes, the positioning accuracy index increases. When there are 20 anchored nodes, the accuracy of 96 nodes is only about 50%. This is because, as the network size increases, the proportion of anchored nodes decreases, affecting the correct speed.

In the experiment, the position configuration of anchor nodes is important, which directly affects the positioning accuracy. Figure 15 shows that when the anchor point is 5 and the total number of nodes is 20, the correct number of position nodes changes with the position of anchor nodes. In the figure, 50 tests indicate that the position of each locked node is randomly selected. It can be seen that the location of anchored nodes has a significant impact on the positioning accuracy of nodes.

From Figure 15, if the number of anchored nodes in the network significantly affects the error of the node positioning algorithm, and the proportion of the anchoring score increases to 30%, the error of the improved probabilistic grid positioning algorithm is as follows: it can be seen that it is about 7% of the communication distance. The location error of the DV-Hop algorithm and unimproved probabilistic grid algorithm is higher than this value.

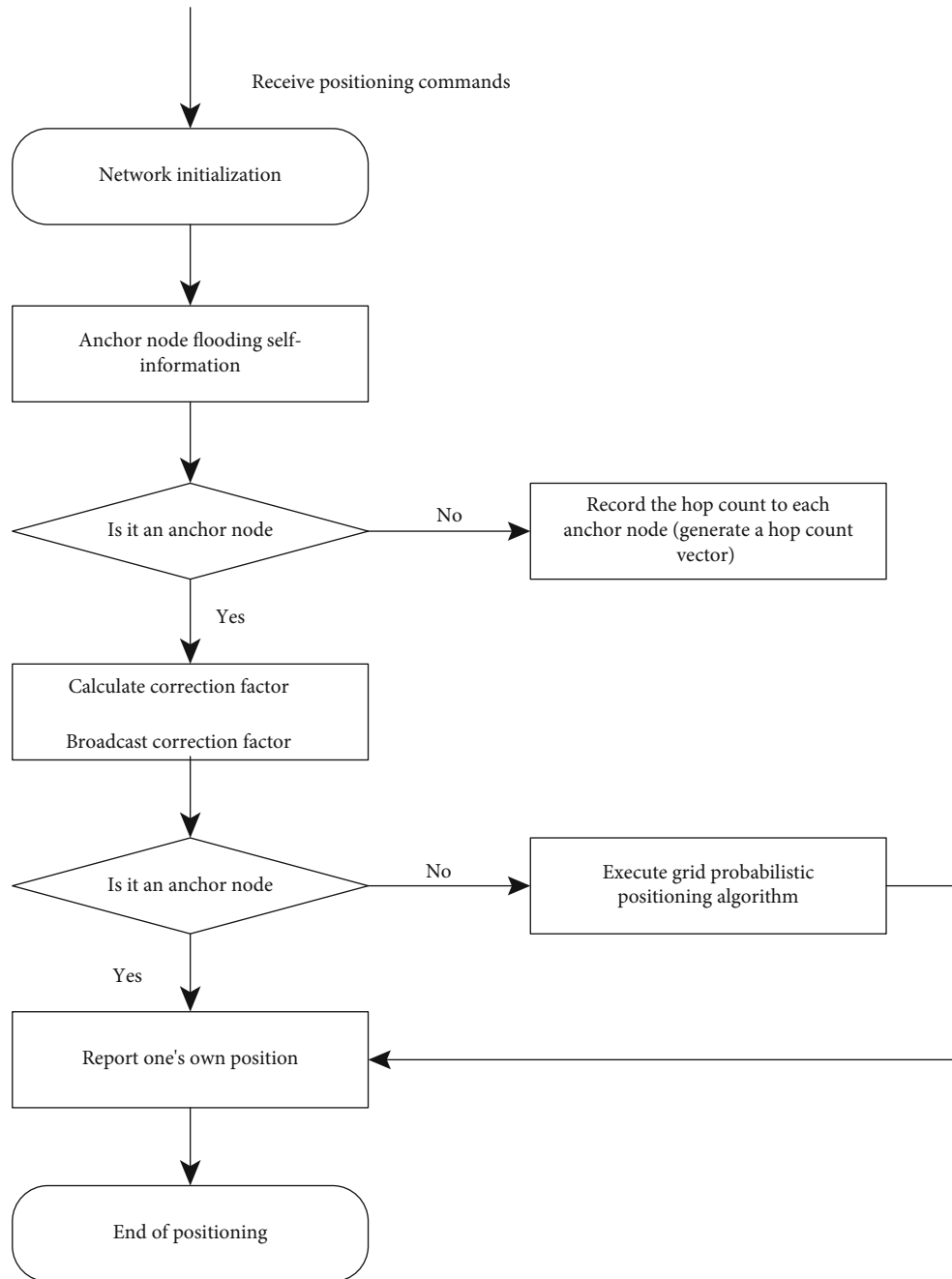


FIGURE 13: Flow chart of grid algorithm execution.

The selection of the correction coefficient has a significant influence on the positioning results. This paper introduces the change of the modification factor of the network scale through simulation experiment. According to the trend of the simulation curve, the correction coefficient can be dynamically adjusted according to the scale of the network in the practical application. If the ratio of anchor nodes in the network is assumed to be 10%, the curve of the optimal correction coefficient is shown here in Figure 16 according to the scale of the network expansion.

When the network size is more than 160 square meters, the configuration interval between nodes is 2 meters, and the change of correction coefficient begins to be stable.

4. Performance Test

4.1. Zigbee Network Performance Test. Figure 16 shows the operational topology of a single production line during system execution. The red node in the figure is the coordinator node, the blue node is the root node, and the yellow node is the terminal node. The whole network represents the tree structure.

During operation, the system tests the performance of the local Zigbee network of each line, and the Zigbee network is divided into a network establishment phase and a data transmission phase. It takes about 90% of the time in the process of network construction. Here, in the

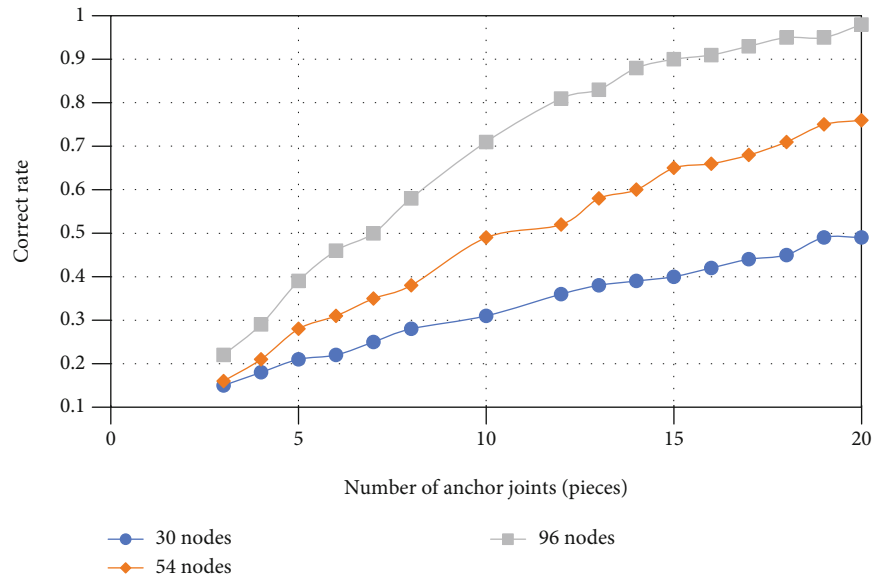


FIGURE 14: The change of the correct rate of positioning nodes with the number of anchor nodes.

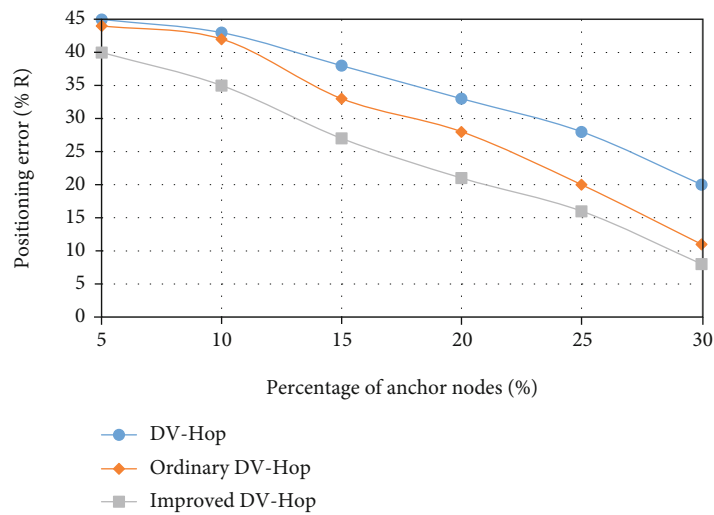


FIGURE 15: Influence of the proportion number of anchor nodes on positioning error.

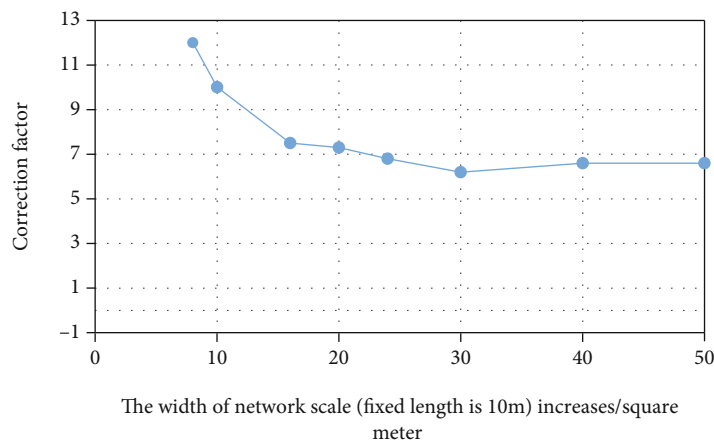


FIGURE 16: The curve of correction factor with the deployment scale of this network.

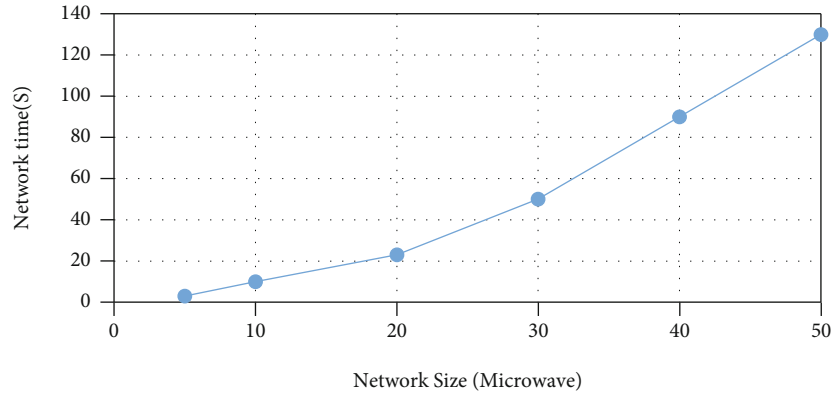


FIGURE 17: Relationship between the number of network nodes and the time required for networking.

TABLE 4: Gateway throughput test results.

Send rate (pieces/s)	Gateway output rate (pieces/s)
0.5	0.57
1	1.12
2	2.3
20	20.8
50	58
200	243

takes more time to maintain the terminal nodes in the network.

4.2. *Gateway Performance Test.* The throughput of the gateway is tested. It is the ability to transfer data without using cache files, that is, without losing data packets. The test results of this paper are shown in Table 4.

During the test, the input speed limit is 200 seconds. It can be seen from the table that the output speed increases linearly with the input speed, so the processing capacity of the gateway is in the crosstalk limit speed (38400 bps is used here), and the performance of the unsaturated gateway fully undertakes the transmission task of the Zigbee network. In addition, because the smaller the timer interval in the Zigbee protocol stack, the greater the error, and the greater the error for the two items at the back of the table.

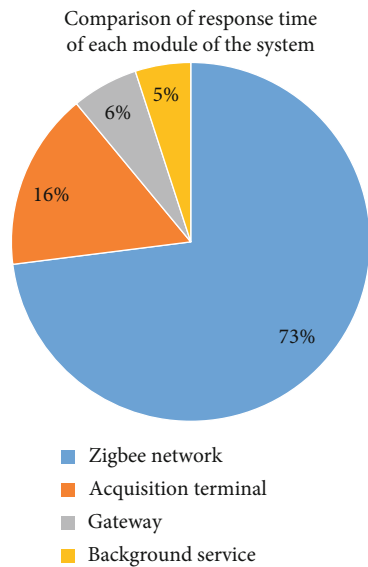


FIGURE 18: Response time ratio of each module of the system.

construction of the network, the time consumption of networks with different sizes is tested. The test results are shown in Figure 17.

As shown in the figure, with the increase of the network scale, the time required for network construction is getting longer and longer. If the network size is less than 80 nodes, the Zigbee network can be built within 10 seconds. If the network size is too large, the association space that coordination nodes must maintain will become larger, because it

4.3. *Overall Response Time of Platform.* The data is generated by the staff using the card from the collection terminal and then inserted into the database through the collection terminal, Zigbee network, gateway, and background server program. The whole process takes about 1.6 s. The system has made 28 measurements and obtained the average response time of each module. For example, the response time of each module is shown in Figure 18.

The Zigbee network module accounts for about 73% of the time. The coordinator needs to confirm the data sent by the terminal. This time, it is a two-way street. On the other hand, the processing power of the processors used by Zigbee coordinators and terminals is limited. When a terminal transmits a large amount of data, caching processing is needed to legally check the data. The background business processing module is equivalent to the time ratio of the gateway module. These processing tasks are similar, but the processing power of the background service is better than that of the gateway module.

4.4. *Platform Performance Optimization Test*

4.4.1. *Data Acquisition Concurrent Stress Test.* The employee’s production data will eventually be inserted into the database through the wireless network. The concurrency of network transmission is not considered here. We simulated the production reports delivered to us during the

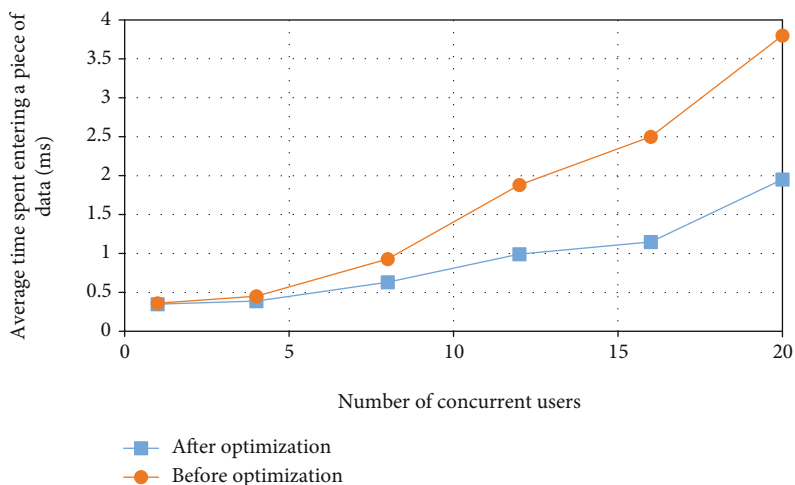


FIGURE 19: Data acquisition concurrent stress testing.

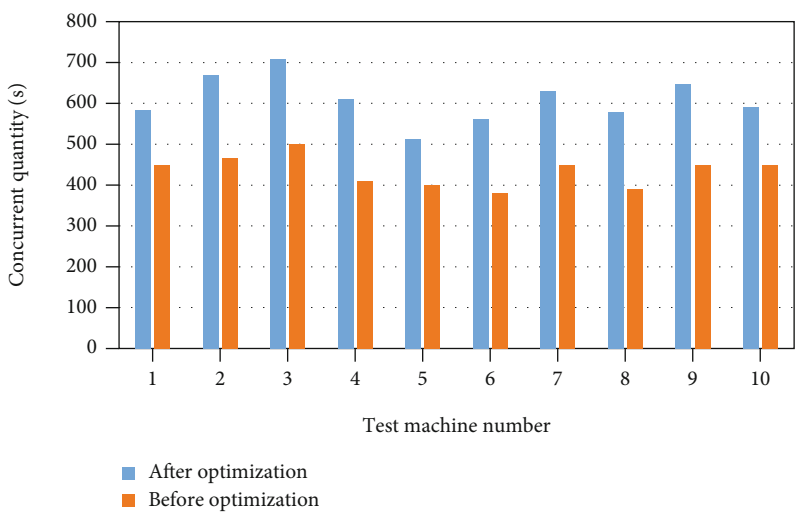


FIGURE 20: Concurrent stress test results of system client.

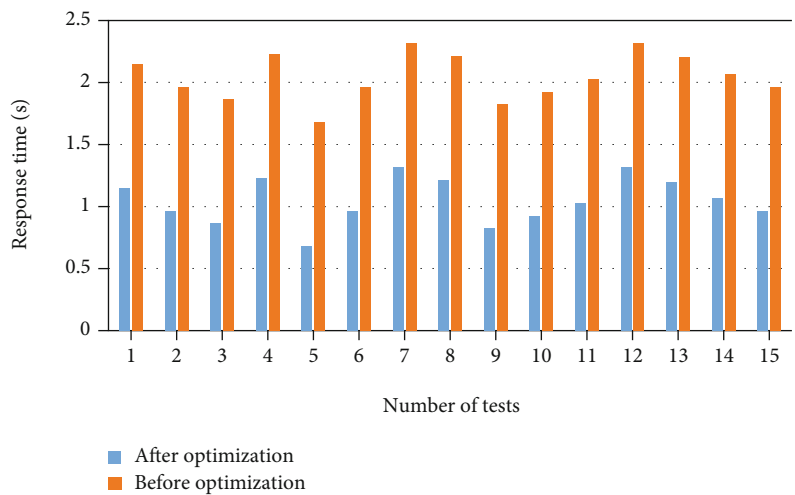


FIGURE 21: System page response time.

production process, increased the number of users linked to the database, and tested the average insertion time of data with different connection numbers. The following Figure 19 shows the results of data acquisition and simultaneous stress testing. When the number of simultaneous connections is 20, the average time for inserting one data after system optimization is 1.95 ms, which can meet the system requirements.

4.4.2. Concurrent Stress Test of System Client. In order to test the simultaneous user volume of the system, it is obviously impossible to install the system on hundreds of computers. In order to solve the above problems, first set the username and password of the database as regular passwords. We use the test database and can freely change the test account information. First, we modify the database with the program, so that the username and password of the test database are the same. At the same time, the program is changed, so that the system login program accepts the parameters of the command line, generates the simulated username and password for batch input, and executes the batch file to simulate the system registration and use of multiple users. In 10 computers, the simulated login and simultaneous stress test were carried out on the system client. Figure 20 shows the resulting test results. The test results show that the optimized system will be kept in the hands of more than 600 users at the same time, which can fully meet the scale requirements of client software installation and be used in large garment factories.

4.4.3. Performance Test of System Page Response Time. The paging response time of the system refers to the time from the time when the user sends out the paging request to the time when the corresponding data is obtained. The system page is the medium for the interaction between the system and the user. The paging response time of the system is the most intuitive evaluation standard for the user. The response time of the production status page was tested, and the test results shown in Figure 21 are obtained:

For the optimized system, the average response time of the production status page is 1.04 s. At the same time, the pages of different systems are tested in the same way, and the response time of each page is maintained at about 1 s. As for the paging response time of the system, there is an unwritten standard in the industry, with the principle of 2/5/10 seconds. The system with paging response time within 2 seconds is considered to bring excellent experience to users. It can bring a good user experience in 5 seconds and bad user experience in 10 seconds. The average user response time of this system is maintained at about 1 second, which can bring users a good experience.

5. Conclusion

This paper is aimed at the background of increasingly obvious advantages such as technological innovation and product manufacturing upgrade. The garment production line management system proposed in this paper is based on the garment production line, introduces Internet technology

into the garment production process, and monitors every link of the garment production process through the Zigbee network. The system improves the automation degree of enterprises, greatly expands the application scope of wireless sensor networks, and promotes the application level of data acquisition, monitoring, equipment maintenance, and diagnosis in China's industrial fields. Through the Internet of Things technology, the clothing production process can be improved, production efficiency can be improved, and industrial upgrading can be realized.

Data Availability

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Conflicts of Interest

The authors declared that they have no conflicts of interest regarding this work.

Acknowledgments

This work was supported in part by the Major Entrusted Project of the Zhumadian Industrial Innovation and Development Research Institute "Clothing Network Customization and Intelligent Manufacturing Training Platform Development and Application Promotion" Project No.: Huanghuai College [2020] No. 188-5.

References

- [1] J. Vestin, A. Kassler, S. Laki, and G. Pongracz, "Toward in-network event detection and filtering for publish/subscribe communication using programmable data planes," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 415–428, 2021.
- [2] Z. Guan, Y. Xu, H. Jiang, and G. Jiang, "International competitiveness of Chinese textile and clothing industry - a diamond model approach," *Journal of Chinese Economic and Foreign Trade Studies*, vol. 12, no. 1, pp. 2–19, 2019.
- [3] X. Guan, X. Guo, L. Zhang, H. Zhao, and S. Xie, "Research on the development direction of marketing models of foreign trade enterprises in the new era," *E3S Web of Conferences*, vol. 251, article 01016, 2021.
- [4] C. Gu, R. Zhou, L. Hu, and G. Gao, "A method of garment factory workers' performance monitoring using control chart based on RFID system," *The International Journal of Advanced Manufacturing Technology*, vol. 107, no. 3-4, pp. 1049–1059, 2020.
- [5] X. Tang, "Research on smart logistics model based on Internet of Things technology," *IEEE Access*, vol. 8, pp. 151150–151159, 2020.
- [6] O. Ayadi, N. Cheikhrouhou, and F. Masmoudi, "A decision support system assessing the trust level in supply chains based on information sharing dimensions," *Computers & Industrial Engineering*, vol. 66, no. 2, pp. 242–257, 2013.
- [7] Y. Xin, D. Zhang, and G. Qiu, "Application of nanomaterials in safety intelligent clothing design," *Integrated Ferroelectrics*, vol. 216, no. 1, pp. 262–275, 2021.

- [8] Y. Jin, D. Ka, S. Jang et al., "Fabrication of graphene based durable intelligent personal protective clothing for conventional and non-conventional chemical threats," *Nanomaterials*, vol. 11, no. 4, p. 940, 2021.
- [9] Y. Wang and Z. Liu, "Personalized custom clothing for intelligent interaction design," in *Emerging Trends in Intelligent and Interactive Systems and Applications: Proceedings of the 5th International Conference on Intelligent, Interactive Systems and Applications (IISA2020)*, pp. 698–709, Shanghai, China, 2021.
- [10] H. Liu and H. Ma, "Design of intelligent clothing selection system based on neural network," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pp. 1789–1792, Chengdu, China, 2019.
- [11] X. Guo, Z. Chen, X. Hu, and X. Li, "Multi-source localization using time of arrival self-clustering method in wireless sensor networks," *IEEE Access*, vol. 7, pp. 82110–82121, 2019.
- [12] Z. Xiao, W. Li, Y. Wu et al., "Preliminary study on regional technology architecture and planning of ubiquitous power internet of things part one overall architecture," *Procedia Computer Science*, vol. 175, pp. 752–757, 2020.
- [13] W. Hamdy, N. Mostafa, and H. Layaway, "An intelligent warehouse management system using the internet of things," *International Journal of Engineering & Technology Sciences*, vol. 32, no. 1, pp. 59–65, 2020.
- [14] N. Liu, J. S. Pan, J. Wang, and T. T. Nguyen, "An adaptation multi-group quasi-affine transformation evolutionary algorithm for global optimization and its application in node localization in wireless sensor networks," *Sensors*, vol. 19, no. 19, p. 4112, 2019.