IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 8, Issue 7, July 2019

Research on Data Sparse-Collection and Transmission System for WSN Based on NB-IoT and FPGA

Jun Jia[1], Guiling Sun[2], Yi Xu[3] and Bowen Zheng[4]

Student, College of Electronic Information and Optical Engineering, Nankai University, Tianjin, China[1,3,4]

Professor, College of Electronic Information and Optical Engineering, Nankai University, Tianjin, China[2]

Abstract: This paper designs and implements a data sparse-acquisition and transmission system for Wireless Sensor Network (WSN) based on Narrow Band –Internet of Things (NB-IoT) and Field-Programmable Gate Array (FPGA) for smart agriculture. The use of distributed wireless network nodes to collect agricultural environment information in a sparse sampling manner can overcome the disadvantages of limited bandwidth and significantly reduce the amount of data transmission, thereby helping to reduce the energy consumption of wireless network nodes and extend the use time. We use FPGA as the base station to receive the data collected by each node, and make full use of the advantages of parallel computing in FPGA for data recovery. And the method we used are matrix filling algorithm, which was named Latent Factor Model (LFM) in the recommendation system, and gradient descent algorithm. Finally, the recovered complete data will be transmitted to the cloud platform for display in the way of NB-IoT. NB-IoT is an emerging Wide Area Network (WAN) technology that has the advantages of supporting massive connections and being geographically unrestricted. The results show that the data can be perfectly recovered within the specified error range, and the algorithm runs nearly three times faster than the serial operation after adding parallel operations.

Keywords: WSN, FPGA, NB-IoT, LFM

I. INTRODUCTION

Smart agriculture applies IoT technology to traditional agriculture, using sensors and software to monitor agricultural production through mobile or computer platforms, making traditional agriculture more "smart". Relying on various sensor nodes (temperature, humidity, conductivity, carbon dioxide, light intensity, image, etc.) deployed in agricultural production sites and wireless communication networks to realize intelligent perception, intelligent early warning, intelligent decision-making, intelligent analysis, expert online guidance, etc.in agricultural production environment, and provide accurate planting, visual management, intelligent decision making for agricultural production. Wisdom agriculture solutions based on WSN technology (Zigbee) have been widely used because of low cost, simple control and convenient to network [1-3]. At the same time, there are some shortcomings. For example, the wireless network node is powered by battery, so the battery replacement must be performed periodically to ensure the operation of the node, the bandwidth of the wireless link is limited, and a large amount of data cannot be transmitted.

In this paper, the sparse sampling theory is applied to the WSN of smart agriculture [4], which overcomes the shortcomings of limited bandwidth to some extent, reduces the power consumption of sensor nodes and prolongs the use time. Using sparse sampling theory can greatly reduce the frequency of data acquisition, thus greatly improving the working time of the sensor, and using a certain data recovery algorithm can recover and reconstruct the uncollected data by the existing data, so as to meet the needs of high-density detection of agriculture environment. FPGA is a device that can realize parallel computing. It is very suitable for the accelerated design of algorithms. Using FPGA for data reconstruction can minimize the running time of the algorithm and speed up the data recovery [5-8].

NB-IoT, based on cellular network, is an emerging technology in the IoT field that supports low-power devices in the WAN's cellular data connection, it's also called Low Energy Wide Area Network (LPWAN) [9-10]. It overcomes the limitations of traditional Local Area Network (LAN) in terms of network coverage, network access, cost, etc., and is very suitable for applications in remote locations and complex environments. With the development of NB-IoT, the Agricultural Internet of Things has also begun to adopt NB-IoT technology [11-12]. Due to its advantages of low power consumption, low cost, wide coverage and large capacity, it has solved the difficulties and pain points of the current Agricultural Internet of Things in a timely manner. It provided technical support while promoting the development of the Agricultural Internet of Things [13-17]. This paper uses this technology to transmit the collected data information to the cloud platform for data monitoring and display. It can be free from geographical restrictions and does not require wired network connection. Fast data transfer can be achieved as long as the base station coverage of the telecom operators.

DOI 10.17148/IJARCCE.2019.8701

II. MATRIX FILLING ALGORITHM (LFM)

Due to known fact of the computational complexity of the Singular Value Decomposition (SVD), it is difficult to implement in practical system. Therefore, people often choose Simon Funk's improved SVD algorithm [18], named LFM. In this paper, the commonly used LFM algorithm in recommended system is applied to the WSN. Although the meaning of the hidden variable is lost (In commodity recommendation, the hidden variable refers to the user's preference for the value of the commodity attribute and the comprehensive performance of each commodity on different attributes, but the hidden variable has no practical meaning in the WSN), it is practical in practice because of the low rank of matrix.

Suppose there is a sparse matrix $A_{U\times I}$. Only a small number of values are known in the matrix, which in this paper represents the data collected by the sensor, where the number of rows U represents the number of sensors, and the number of columns I represents the total number of data collected by each sensor at different times. The unknown value, which we need to predict and fill, means that the sensor was not collecting data, so the matrix can be regarded as a sparse sampling of the information. Since there is a full rank decomposition for any matrix, the matrix $A_{U\times I}$ can be expressed as the product of two matrices $P_{U\times K}$ and $Q_{K\times I}$:

$$A_{U\times I} = P_{U\times K} \times Q_{K\times I}\#(1)$$

Where $K = \mathrm{Rank}(A)$ represents the rank of the matrix. $P_{U\times K}$ and $Q_{K\times I}$ can be trained using the values known in $A_{U\times I}$, such that the product of P and Q is best fitted to the known data in $A_{U\times I}$, and the unknown data can be obtained by multiplying by a row of P and a column of Q:

$$a'_{ui} = p_u^T q_i\#(2)$$

Assuming that the known data in the matrix $A_{U\times I}$ is a_{ui}, the error between the true value and the predicted value is:

$$e_{ui} = a_{ui} - a'_{ui}\#(3)$$

And then, the total sum of squares for error (SSE) can be calculated using known data:

$$SSE = \sum_{u,i} e_{ui}^2 = \sum_{u,i}\left(a_{ui} - \sum_{k=1}^{K} p_{uk} q_{ki}\right)^2 \#(4)$$

In order to prevent over-fitting of the training, a regularization penalty is added to the objective function SSE to penalize all variables of P and Q:

$$SSE = \frac{1}{2}\sum_{u,i} e_{ui}^2 + \frac{1}{2}\lambda\sum_{u}|p_u|^2 + \frac{1}{2}\lambda\sum_{i}|q_i|^2\#(5)$$

In the (5), $\frac{1}{2}$ is to make the guidance more convenient, as long as the SSE is minimized by training, then P and Q can best fit A. In this paper, the gradient descent method is used to train and fit in FPGA. The gradient of SSE in the variable p_{uk} (the value of the u-th row and the k-th column of the matrix P) can be obtained by using the derivative chain rule:

$$\frac{\partial}{\partial p_{uk}} SSE = \frac{\partial}{\partial p_{uk}}\left(\frac{1}{2}e_{ui}^2 + \frac{1}{2}\lambda|p_u|^2 + \frac{1}{2}\lambda\sum_{i}|q_i|^2\right) = e_{ui}\frac{\partial}{\partial p_{uk}}e_{ui} + \lambda p_{uk}\#(6)$$

Due to:

$$e_{ui} = a_{ui} - a'_{ui} = a_{ui} - p_u^T q_i = a_{ui} - \sum_{k=1}^{K} p_{uk} q_{ki}\#(7)$$

Thus:

$$\frac{\partial}{\partial p_{uk}} e_{ui} = \frac{\partial}{\partial p_{uk}}\left(a_{ui} - \sum_{k=1}^{K} p_{uk} q_{ki}\right)\#(8)$$

The only item related to the derivation is $p_{uk} q_{ki}$ in (8), so the result of the derivation is:

$$\frac{\partial}{\partial p_{uk}} e_{ui} = -q_{ki}\#(9)$$

Therefore:

$$\frac{\partial}{\partial p_{uk}} SSE = e_{ui}\frac{\partial}{\partial p_{uk}} e_{ui} + \lambda p_u = -e_{ui} q_{ki} + \lambda p_{uk} \#(10)$$

Now the gradient of the objective function at p_{uk} is obtained. According to the gradient descent method, change to the direction of the negative gradient of p_{uk}. Let the updated step size (learning rate) be η, then the update of p_{uk} is:

$$p_{uk} \coloneqq p_{uk} - \eta(-e_{ui} q_{ki} + \lambda p_{uk}) \coloneqq (1-\lambda\eta)p_{uk} + \eta e_{ui} q_{ki} \#(11)$$

Similarly, the updated version of q_{ki} can be obtained:

$$q_{ki} \coloneqq q_{ki} - \eta(-e_{ui} p_{uk} + \lambda q_{ki}) \coloneqq (1-\lambda\eta)q_{ki} + \eta e_{ui} p_{uk} \#(12)$$

There are two methods for iterative updating of matrices P and Q, namely batch gradient descent and stochastic gradient descent. The updated value of this iteration only can be used in the next iteration for batch gradient. However, the value used by the current sample in this iteration may be the value of the previous sample update for stochastic gradient. Randomness can bring many benefits, such as helping to avoid local optimal solutions, so this paper uses random gradient descent method to update.

III. SYSTEM DESIGN

A. Overall architecture

The system consists of data acquisition, data recovery and data transmission and display. The overall structure of the system is shown in Fig.1.

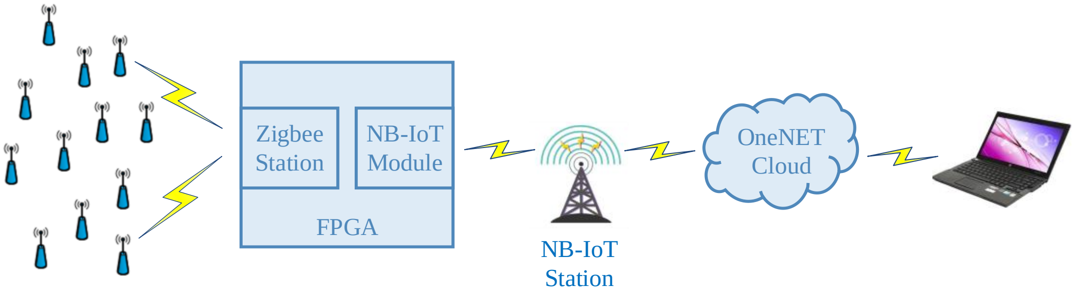

Fig.1 The overall structure of the system

The data acquisition part collects the farmland environment information used the wireless sensor node based on Zigbee network, and send the data to the Zigbee base station in real time. After the data is received, the LFM algorithm is executed inside the FPGA to recover and reconstruct the missing data. The reconstructed data will be sent to the cloud platform through the NB-IoT module and the telecom carrier's base station. At this time, the terminal (computer) can view the obtained data by logging in to the cloud platform, thereby realizing the monitoring of the farmland environment information.

B. Sparse sampling

Assume that there is a complete data set matrix $R_{U\times I}$. Define a sparse sampling matrix $E_{U\times I}$ of the same size as $R_{U\times I}$. There are only 0 and 1 elements in the matrix $E_{U\times I}$, and the number of 1 is much smaller than the number of 0, and the distribution of 1 has no regularity. The sparse sampling of the data set can be completed by the point multiplication operation of the data set matrix and the sparse sampling matrix, thereby obtaining a sparse sample $A_{U\times I}$ of the data set:

$$A_{U\times I} = R_{U\times I} \bullet E_{U\times I} \#(13)$$

In this paper, the sampling of data is done by collecting data through the sensor, and a sparse matrix $E_{U\times I}$ is generated in advance with a certain sampling rate.

$$\mathbf{E}_{U\times I} = \begin{bmatrix} \mathbf{0} & \mathbf{1} & \mathbf{1} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{1} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{1} & \mathbf{0} & \mathbf{1} & \cdots & \mathbf{0} \end{bmatrix} \#(14)$$

U represents the total number of Zigbee sensor nodes connected to the base station. To distinguish each sensor node, each sensor is numbered separately, and the u-th row represents the sensor of number u. I represents the number of times each node needs to perform data collection in performing an LFM algorithm, and the i-th column represents the i-th sensor data acquisition. Of course, the data acquisition or not depends on the value of the sparse matrix actually, such as, for the matrix in (14), $e_{13} = 1$, which represents the data acquisition of the No.1 sensor during the third data acquisition, then the value of a_{13} in the sample matrix is the collected data; $e_{21} = 0$, which means that the NO.2 sensor does not collect data during the first data acquisition, then the value of a_{21} in the sample matrix is recorded as 0. Sparse sampling of the data is done in this way.

C. Data Recovery

The LFM algorithm is implemented in the FPGA, which has abundant hardware logic resource and can realize parallel operation. And the known acquisition data is used to train and fit the matrix $P_{U\times K}$ and $Q_{K\times I}$.

Let $k = \min(U, I)$, initialize $P_{U\times K}$ and $Q_{K\times I}$ to a random matrix between 0 and 1, and store it in the ROM of the FPGA for use. The sparse samples $A_{U\times I}$ are scanned in order from left to right and top to bottom. When the scanned element a_{ui} is non-zero, it is calculated in the FPGA:

$$e_{ui} = a_{ui} - a_{ui}^{'} = a_{ui} - \sum_{k=1}^{K} p_{uk} q_{ki} \#(15)$$

Due to the parallelism of the FPGA, the above operation only needs K+1 clock cycles to complete. The first clock performs K independent multiplications in parallel: $p_{u1}q_{1i}$, $p_{u2}q_{2i}$, $\cdots p_{uk}q_{ki}$. The second to K-th clocks perform K-1 summation operation, and the (K+1)-th clock performs a subtraction operation, and finally obtains the value of e_{ui}. The operation flow is as shown in Fig.2.

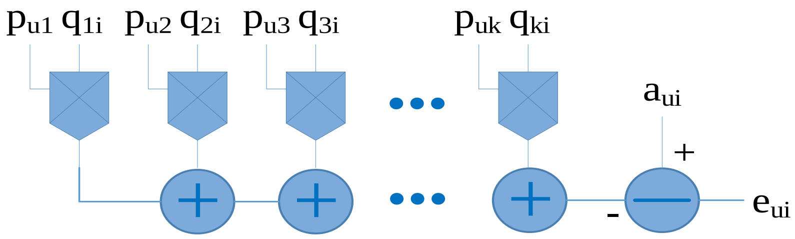

Fig.2 The operation flow of e_{ui}

Updates to p_{uk} and q_{ki} can also be done simultaneously, and only three clock cycles are required for each p_{uk} or q_{ki} update. The first clock calculates 1-λη and $e_{ui}\, q_{ki}$ (The value of λη can be predetermined and stored in the ROM of the FPGA). The second clock calculates the remaining multiplication, and the last clock calculates the addition, as shown in Fig.3. Therefore, it takes only 3K clock cycles for each round of p_{uk} and q_{ki} (k=1, 2 ⋯ K), and the total operation time is 4K+1 clock cycles after adding the operation time of e_{ui}.

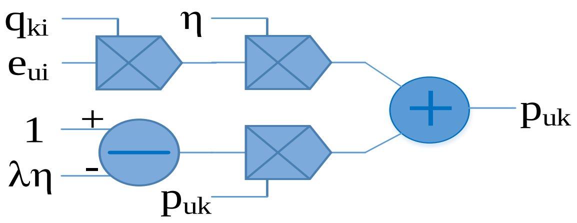

Fig.3 The update operation flow of p_{uk}

In the sparse sample $A_{U\times I}$ obtained by the sampling rate f, the non-zero value elements have $f \cdot UI$, and the zero-value elements have $(1 - f) \cdot UI$. Therefore, for a complete scan of the matrix $A_{U\times I}$, there are $(1 - f) \cdot UI$ clock cycles scanned to zero value, and $f \cdot UI$ clock cycles are scanned to non-zero value. The update of the matrices P and Q requires $f \cdot UI \cdot (4K + 1)$ clock cycles, so the total clock cycle for a complete iteration is $T_1 = [1 + (1 + 4K)f] \cdot UI$. If the number of iterations is set to N times, restoring the complete data set $R_{U\times I}$ requires a total of $N \cdot [1 + (1 + 4K)f] \cdot UI$ clock cycles. The accuracy of predictions for unknown data can be determined by mean absolute error (MAE) and root mean square error (RMSE). Because they can be used to measure the deviation between the predicted value and the real value. The accuracy of data recovery is adjusted by setting the thresholds of MAE and RMSE. The smaller the MAE and RMSE, the higher the accuracy of the recovered data and the more iterations are required.

$$\mathrm{MAE}(A,A') = \frac{1}{n}\sum_{u\in U,i\in I}|a_{ui} - a'_{ui}| = \frac{1}{n}\sum_{u\in U,i\in I}\left|a_{ui} - \sum_{k=1}^{K} p_{uk}\,q_{ki}\right| \#(16)$$

$$\mathrm{RMSE}(A,A') = \left[\frac{1}{n}\sum_{u\in U,i\in I}(a_{ui} - a'_{ui})^2\right]^{\frac{1}{2}} = \left[\frac{1}{n}\sum_{u\in U,i\in I}\left(a_{ui} - \sum_{k=1}^{K} p_{uk}\,q_{ki}\right)^2\right]^{\frac{1}{2}} \#(17)$$

Where a_{ui} represents a non-zero value in the sparse sample $A_{U\times I}$, that is, the collected real data, and n represents the number of non-zero values.

D. Data transmission

By setting the threshold of the error, the true error after the iteration is smaller than the threshold. At this time, the latest matrix P and Q are multiplied to obtain the complete data set $R_{U\times I}$, wherein the sparse sample $A_{U\times I}$ is used. The true value in $A_{U\times I}$ replaces the value of the same position in $R_{U\times I}$, and the other values are predicted values.

The complete data obtained is sent to the platform side to realize the data display function, thereby completing the design of the entire system. Taking into account the geographical environment, using NB-IoT to achieve this function can overcome the shortcomings of no wired network in a remote environment, and the technology has wide coverage, low cost, low power consumption, support for massive connections, and can realize a large number of nodes in. This paper uses China Mobile's NB-IoT module M5310 to connect with the cloud platform. China Mobile's base station covers a wide range, so it can meet the access needs of most regions.

The access protocol adopts the Internet of Things application layer protocol, named lightweight Machine to Machine (LWM2M) and defined by open Mobile Alliance (OMA). The networking mode is NB-IoT, and the platform side adopts the Internet of Things open cloud platform OneNET created by China Mobile. The structure of the LWM2M protocol stack is shown in Fig.4.

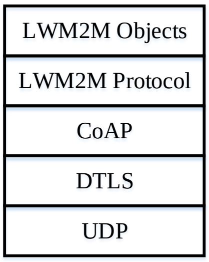

Fig.4 The structure of the LWM2M protocol stack

Each object corresponds to a specific functional entity of the client, for example: LWM2M Server Object, whose object ID is 2; LWM2M Access Control Object, whose object ID is 3. It also defines the objects of some commonly used sensors, such as: Illuminance Sensor, whose object ID is 3301; Conductivity, whose object ID is 3327. Each object can have multiple resources, and each resource represents an object attribute or function. For example, some resource descriptions of the GPS Location sensor with object ID 3336 are shown in Table 1.

Table I Partial Resource of GPS Sensor

Resource ID	Description	Data Type	Permission
5514	Latitude	String	Read
5515	Longitude	String	Read

The LWM2M protocol defines some logical operations, such as Read, Write, Execute, Create, and Delete. These logical operations enable data interaction between the cloud platform and the terminal device.

The transport layer protocol of LWM2M is Constrained Application Protocol (CoAP), and the lower layer can be UDP. Datagram Transport Layer Security (DTLS) is used to ensure security between the client and the server.

To access the device, we need to create a product on the cloud platform and enter the International Mobile Equipment Identity (IMEI) number of the module and the International Mobile Subscriber Identification (IMSI) number of the IoT card. The module side needs to call the AT command of the module to log in to the cloud according to the process and report the data. After the device is successfully accessed, the data can be sent to the cloud platform side and displayed in the terminal. The access process for M5310 is shown in Fig.5. The AT command flow for docking the OneNET platform is shown in Fig.6.

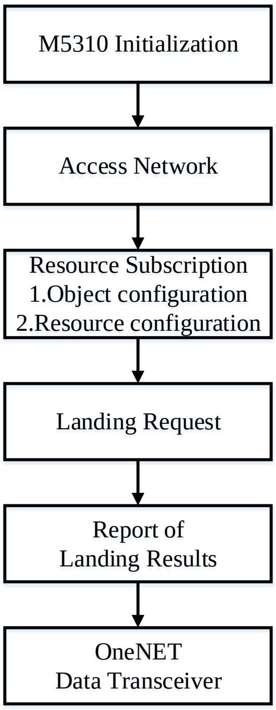

Fig.5 The access process for M5310

```
(1)        Inspection after power-on
AT
AT+CSQ
AT+CEREG?
AT+CGATT?
(2)        Device creation, Resource subscription and Login
AT+MIPLCONFIG=72,1003……501,1,1
AT+MIPLADDOBJ=0,3303,0
AT+MIPLNOTIFY=0,3303,0,5700,4,”20”,1
AT+MIPLOPEN=0,30
(3)        Data transceiver
AT+MIPLNOTIFY=0,3303,0,5700,4,”21.05”,1
(4)        Device logout
AT+MIPLCLOSE=0
AT+MIPLDELOBJ=0,3303,0
AT+MIPLDEL=0
```

Fig.6 The AT command flow for docking the OneNET platform

IV. RESULT ANALYSIS

The data collected in this paper is temperature information. The size of the data set is U=20，I=60, that is, there are 20 sensor nodes, and each node collects 60 times for data recovery. Set the sampling rate f=0.2, and give the random sampling matrix $E_{20\times60}$, then the proportion of element 1 in the matrix is 20%, as shown in Fig.7.

$$\mathbf{E}_{20\times60}=\begin{bmatrix}\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\cdots&\mathbf{0}\\\mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\cdots&\mathbf{0}\\\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}&\cdots&\mathbf{0}\\\mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\cdots&\mathbf{1}\\\vdots&\vdots&\vdots&\vdots&\vdots&\ddots&\vdots\\\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{1}&\mathbf{0}&\cdots&\mathbf{0}\end{bmatrix}$$

Fig.7 The random sampling matrix

DOI 10.17148/IJARCCE.2019.8701

If data collection is set every one minute, it takes one hour to acquire 60 times. Therefore, each time data recovery is performed, the temperature data within one hour is restored. The sparse sample data $A_{20\times 60}$ collected a certain time is shown in Fig.8. It can be seen that the data set is sparse, only a small part of data are the collected real data, and others are zero.

$$\mathbf{A}_{20\times 60}=\begin{bmatrix} \mathbf{0} & \mathbf{19.30} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{19.47} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{19.40} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{22.98} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{19.59} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{19.46} & \mathbf{19.17} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}$$

Fig.8 The sparse sample data

Let η = 0.001， λ = 0.02, after 500 iterations, MAE = 0.0682 and RMSE = 0.0877, the recovered complete data set $R_{20\times 60}$ is shown in Fig.9. The value in parentheses is the original data.

$$\mathbf{R}_{20\times 60}=\begin{bmatrix} \mathbf{19.98} & \mathbf{19.41(19.30)} & \mathbf{19.12} & \mathbf{20.24} & \mathbf{19.28} & \cdots & \mathbf{18.71} \\ \mathbf{20.47} & \mathbf{19.80} & \mathbf{19.38(19.47)} & \mathbf{21.70} & \mathbf{19.66} & \cdots & \mathbf{18.53} \\ \mathbf{20.34} & \mathbf{19.25} & \mathbf{19.75} & \mathbf{21.78} & \mathbf{19.47(19.40)} & \cdots & \mathbf{18.95} \\ \mathbf{20.87} & \mathbf{19.61} & \mathbf{22.90(22.98)} & \mathbf{21.10} & \mathbf{19.99} & \cdots & \mathbf{19.72(19.59)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{19.12} & \mathbf{19.35(19.46)} & \mathbf{19.19(19.17)} & \mathbf{19.18} & \mathbf{18.50} & \cdots & \mathbf{18.42} \end{bmatrix}$$

Fig.9 Recovery data set obtained by performing LFM 500 times

When η = 0.001, λ = 0.02, Fig.10 shows the number of iterations required for different MAE thresholds. Fig.11 shows the number of iterations required for different RMSE thresholds. It can be seen that the smaller the MAE or RMSE, the more iterations are needed. That is, the more iterations are needed for better fit, and the more accurate the results will be.

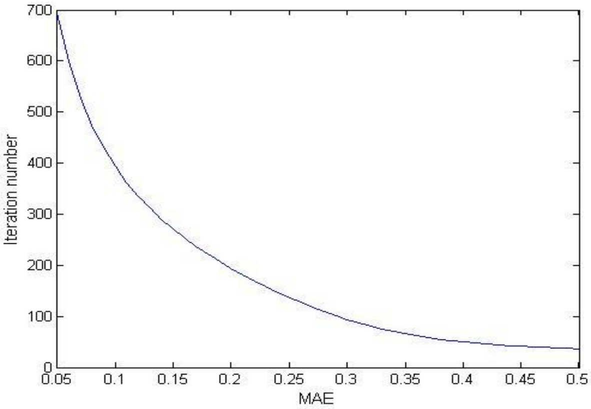

Fig.10 The relationship between the number of iterations and MAE

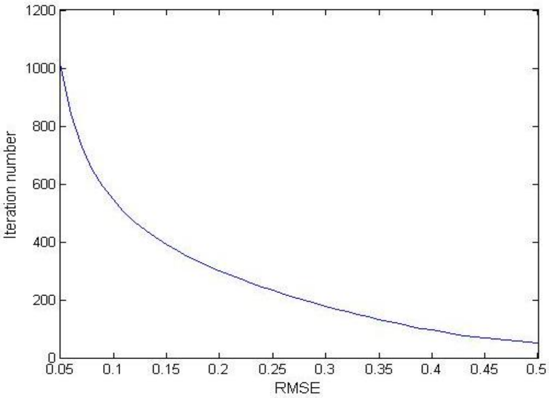

Fig.11 The relationship between the number of iterations and RMSE

DOI 10.17148/IJARCCE.2019.8701

When $\eta = 0.001$, $\lambda = 0.02$, and the number of iterations is 1000, Fig.12 gives the MAE values corresponding to different sampling rates f, and Fig.13 shows the RMSE values corresponding to different sampling rates f. It can be seen that as the sampling rate increases, the MAE and RMSE tend to increase as a whole, because the number of actual data collected increases, resulting in a weaker fit to all data when the iterations are the same number of times, resulting in an increase in error. At the same time, when the sampling rate f is less than 0.4, the curve rises faster, while when it is greater than 0.4, the curve fluctuation is more obvious.

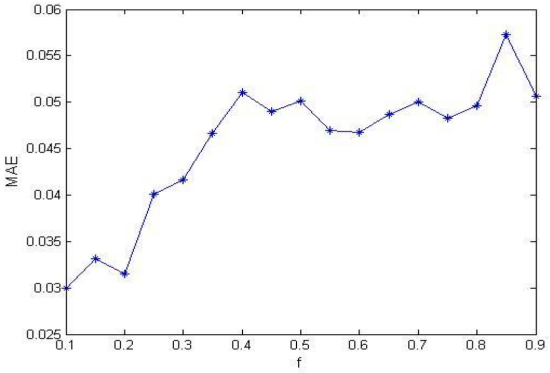

Fig.12 Relationship between MAE and sampling rate f

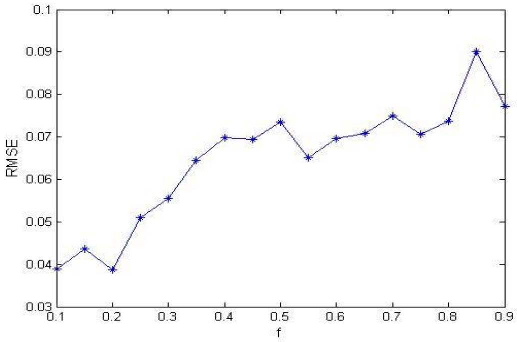

Fig.13 Relationship between RMSE and sampling rate f

In practical applications, the error values (MAE and RMSE), the number of iterations, and the sampling rate can be comprehensively considered according to the requirements for accuracy, operation speed, and power consumption. Since the temperature accuracy requirements in the farmland environment are not very high, and the wireless sensor nodes are sensitive to power consumption, a higher MAE or RMSE value can be selected to reduce the number of iterations, and a lower sampling rate is selected to reduce power consumption. This will help speed up data recovery. In this paper, when the sampling rate is greater than 0.4, the error does not improve significantly, but it will lead to a sharp increase in power consumption (If power consumption is recorded as W when sampling rate f equals 1, the actual power consumption can be regarded as fW). When the sampling rate is less than 0.4, and the smaller it is, it only needs to be iterated less times to get smaller error. Considering this, the sampling rate is set to 0.2 in this paper.

The sparse sample $A_{U\times I}$ obtained by the sampling rate f, adding a parallel operation to the FPGA for a complete iteration requires a total clock period of $T_1 = [1 + (1 + 4K)f] \cdot UI$. If a complete iteration is performed in a full serial mode, it takes 2K clock cycles to calculate e_{ui} and takes 5K clock cycles to calculate a round of p_{uk} or q_{ki} (k=1, 2 ⋯ K), and the final complete clock cycle is $T_2 = (1 + 12Kf) \cdot UI$. So the time ratio:

$$\frac{T_2}{T_1} = \frac{1 + 12Kf}{1 + (1 + 4K)f} \#(18)$$

When the sampling rate $f = 0.2$, the curve of $\frac{T_2}{T_1}$ with the rank K of the matrix is shown in Fig.14. It can be seen that when K is large enough, T_2 is 3 times of T_1 in the same clock cycle, that is, the larger the sparse sample matrix is, the more the speed can be increased by nearly 3 times after adding parallel operation, which greatly reduces the running time of the algorithm. In this paper, $K = 20$, the operation speed is increased by 2.849 times.

DOI 10.17148/IJARCCE.2019.8701

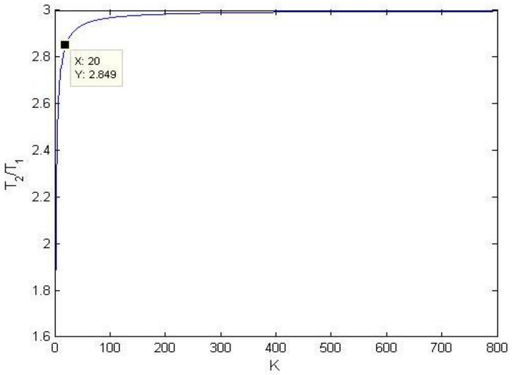

Fig.14 The curve of $\frac{T_2}{T_1}$ with the rank K of the matrix

V. SUMMARY

This paper implements the design of data sparse acquisition and transmission system for WSN based on NB-IoT and FPGA. The system realizes the reception of Zigbee node data, the recovery of sparse data, and the function of transmitting the recovery data with FPGA as the core control end. Adopting the emerging WAN technology NB-IoT to realize long-distance data transmission and display on China Mobile's cloud platform.

The system introduces the LFM algorithm, based on recommendation system, and the gradient descent method as the core algorithm to realize the recovery of the temperature data. Only a part of the data is acquired at a lower sampling rate to obtain all the data sets, which greatly reduces the power consumption to a certain extent and meets the demand for high-density data in the farmland environment. Using hardware FPGA to implement LFM algorithm has higher running speed than software implementation, because FPGA has the advantage of parallel computing.

The system achieves the collection and recovery of temperature data for the purpose of reducing power consumption and increasing the speed of operation. It is also applicable to other types of data, and can be applied not only to smart agriculture but also to other Internet of Things, and has a broad application prospects. In practice, if the scale of the FPGA is sufficient, multiple e_{ui} simultaneous calculations and multiple sets of p_{uk} and q_{ki} can be simultaneously updated, which will further increase the speed of the operation. In practice, a trade-off between FPGA scale and speed can be realized as needed.

ACKNOWLEDGMENT

This work is in part supported by the National Nature Science Foundation of China under Grant No.61771262 and the National High-tech Research and Development Program of China under Grant No.2017ZXHLNC00100 and No.18ZXRHNC00140.

REFERENCES

[1]. Dan L I U, Xin C, Chongwei H, et al. Intelligent agriculture greenhouse environment monitoring system based on IOT technology[C]//2015 International Conference on Intelligent Transportation, Big Data and Smart City. IEEE, 2015: 487-490.
[2]. Sahitya G, Balaji N, Naidu C D, et al. Designing a wireless sensor network for precision agriculture using zigbee[C]//2017 IEEE 7th International Advance Computing Conference (IACC). IEEE, 2017: 287-291.
[3]. Chikankar P B, Mehetre D, Das S. An automatic irrigation system using ZigBee in wireless sensor network[C]//2015 International Conference on Pervasive Computing (ICPC). IEEE, 2015: 1-5.
[4]. Qin Z, Fan J, Liu Y, et al. Sparse representation for wireless communications: A compressive sensing approach[J]. IEEE Signal Processing Magazine, 2018, 35(3): 40-58.
[5]. Huang G, Wang L. An FPGA-based architecture for high-speed compressed signal reconstruction[J]. ACM Transactions on Embedded Computing Systems (TECS), 2017, 16(3): 89.
[6]. Ahmedsaid A, Amira A, Bouridane A. Improved SVD systolic array and implementation on FPGA[C]//Proceedings. 2003 IEEE International Conference on Field-Programmable Technology (FPT)(IEEE Cat. No. 03EX798). IEEE, 2003: 35-42.
[7]. Ledesma-Carrillo L M, Cabal-Yepez E, Romero-Troncoso R J, et al. Reconfigurable FPGA-Based unit for Singular Value Decomposition of large mxn matrices[C]//2011 International Conference on Reconfigurable Computing and FPGAs. IEEE, 2011: 345-350.
[8]. Ibrahim A, Valle M, Noli L, et al. FPGA implementation of fixed point CORDIC-SVD for E-skin systems[C]//2015 11th Conference on Ph. D. Research in Microelectronics and Electronics (PRIME). IEEE, 2015: 318-321.
[9]. Sowmya P, Ujwala M. An Iot Based Reconfigurable Smart Sensor Interface For Industrial Monitoring System[J]. International Journal of Research, 2018, 5(01): 1630-1635.
[10]. Li Y, Cheng X, Cao Y, et al. Smart choice for the smart grid: Narrowband Internet of Things (NB-IoT)[J]. IEEE Internet of Things Journal, 2018, 5(3): 1505-1515.
[11]. TongKe F. Smart agriculture based on cloud computing and IOT[J]. Journal of Convergence Information Technology, 2013, 8(2).

[12]. Gondchawar N, Kawitkar R S. IoT based smart agriculture[J]. International Journal of advanced research in Computer and Communication Engineering, 2016, 5(6): 2278-1021.
[13]. Gubbi J, Buyya R, Marusic S, et al. Internet of Things (IoT): A vision, architectural elements, and future directions[J]. Future generation computer systems, 2013, 29(7): 1645-1660.
[14]. Ray P P. Internet of things for smart agriculture: Technologies, practices and future direction[J]. Journal of Ambient Intelligence and Smart Environments, 2017, 9(4): 395-420.
[15]. Dlodlo N, Kalezhi J. The internet of things in agriculture for sustainable rural development[C]//2015 international conference on emerging trends in networks and computer communications (ETNCC). IEEE, 2015: 13-18.
[16]. Kapoor A, Bhat S I, Shidnal S, et al. Implementation of IoT (Internet of Things) and Image processing in smart agriculture[C]//2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS). IEEE, 2016: 21-26.
[17]. Jayaraman P, Yavari A, Georgakopoulos D, et al. Internet of things platform for smart farming: Experiences and lessons learnt[J]. Sensors, 2016, 16(11): 1884.
[18]. Kurucz M, Benczúr AA, Csalogány K Method for large scale SVD with missing value[C]/Proceedings of KDD cup & workshop.2007,12: 31-38

BIOGRAPHIES

Jun Jia was born in Hebei province, China, in 1994. He received B.S. degree in Microelectronics Science and Engineering from Nankai University, Tianjin, China, in 2017. He is pursuing his M.S. degree at Nankai University. His current research is digital signal processing based on FPGA and smart agriculture based on NB-IoT and LoRa. Mr. Jia's awards and honors include National Inspirational Scholarship, Comprehensive First Class Scholarship and Triple-A student in Nankai University.

Guiling Sun received Ph.D from Nankai University in 2004 and has become the professor of Nankai University. She is also the vice director of the Tianjin Institute of Communications, the vice director of EDA Technology Research Society in North China and the leader of Electronic technology experiment and practice teaching team of Tianjin city. She has published more than 90 papers and 2 research monographs in a number of areas about the Wireless Sensor Networks, Internet of things, Compressed Sensing, information detection and intelligent control systems and signal processing.

Yi Xu was born in Hunan Province, China, in 1991. She received the B.S. degree in 2014 and M.S. degree in 2016 both in electrical engineering from Nankai University, Tianjin, China. She is currently pursuing the Ph.D. degree with the College of Electronic Information and Optical Engineering, Nankai University. Her research interests include signal processing and Wireless Sensor Networks.

Bowen Zheng was born in Hebei Province, China, in 1993. He received the B.S. degree in 2016 and M.S. degree in 2018 both in Electronic Information Science and Technology from Nankai University, Tianjin, China. He is currently pursuing the Ph.D. degree in the College of Electronic Information and Optical Engineering, Nankai University. His research interests are compressed sensing and machine learning.

DOI 10.17148/IJARCCE.2019.8701