Graduation Thesis Academic Year 2011

Congestion Avoidance in Wireless Sensor Network with Adaptive Time-wise Data Interpolation

Doan Hoai Nam

Faculty of Environment and Information Studies
Keio University

Abstract of Bachelor Thesis

Congestion avoidance in Wireless Sensor Network (WSN) is an essential research topic. It has, thus, received much attention of researchers. Congestion avoidance comprises of congestion detection and traffic control. This thesis focuses on traffic control and proposes an optimal data interpolation method, referred to as the Approximation Equation Method, and an adaptive method to adjust data reporting interval and the data queue length in accordance with permissible error criteria. When congestion is detected in a Wireless Sensor Network, usual practice is to lengthen data reporting interval. The eliminated data is usually recovered by application with an interpolation. In approximation equation's method, the optimization is done in end devices such that the transmitted data provides the best approximation within the reporting interval. With a thorough mathematical representation, the optimization can be done with a low computational power yet low electric power wireless transceiver. This method is applied to a room temperature monitoring system using one-chip ZigBee Transceiver TI CC2430. It is shown that the approximation equation's method can archive 1.7 times longer the reporting time interval than that of the simple truncation. The success of the approximation equation's method motivated the author to develop an adaptive control of reporting time interval under predefined error threshold. With the set limited error rate at 0.0125%, the adaptive time wise data interpolation method reduces the data transmission packet number by 94∼99%. However, in this case, approximation equation's method is only effective when sufficient sensor data queuing is permitted particularly where large sensor data fluctuation is observed. The adaptive control is also implemented in CC2430. It reveals that, by using only 20 data in queue, in comparison with no use of queue, the number of transmission packets can be reduced by 30%.

Keywords:
Wireless Sensor Network, Traffic congestion avoidance, Approximation equation, Data interpolation.

Doan Hoai Nam

Faculty of Environment and Information Studies
Keio University

卒業論文要旨

今日、無線センサネットワークにおいてトラフィック混雑防止は重要な要素であり、様々な研究が行われている。トラフィック混雑防止には混雑発見とトラフィック抑制の段階がある。本論文では、トラフィック抑制に焦点を当て、設定したエラー率に基づく近似方程式を用いた最適なデータ補間手法を提案する。既存手法では、無線センサネットワークの混雑を検知した場合にデータ送信頻度を長くし、間引くことにより通信量を減らす。間引かれたデータは、収集できたデータから推測、補間することとなる。本論文が提案する方法では、エンドデバイスが間引く前の全てのデータから近似式を立て、この式から導出されるデータを送信する。これにより、収集したデータから予測するする間引かれたデータの推定値と、実際の観測値とのエラーを減らすことができる。本手法は計算手法を工夫することにより、計算資源の少ないセンサデバイスにも最適化モジュールとして実装を可能とした。本論文では気温センシングシステムを ZigBee トランシーバー TI CC2430 を使ってこの手法を実装した。本論文の提案する近似方程式を用いたデータ補間を用いることにより、用いない場合と比べて、エラー率を上げずにデータ送信間隔を 1.7 倍長くすることが出来た。この結果に基づいて、近似方程式を用い、設定したエラー率によって自動でデータ送信間隔を調整するシステムを開発、実装した。この実装を用いた実験では、提案したデータ送信頻度調整手法を用いて、送信パケット数を 94~99% 抑えることができた。また、エンドデバイスが最低 20 個のデータを保持するまでは処理を行わない設定にした場合に最も効果的であり、送信パケット数を 30% 抑えることができた。

キーワード：
無線センサネットワーク, トラフィック混雑防止, 近似方程式, データ補間

慶應義塾大学環境情報学

ドアン ホアイ ナム

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Nowadays, with the ever-increasing use of environmental sensing and smart grids, in the wireless sensor networks, especially in multi-hop networks, the quantity of data volume is rapidly increasing. To reach the main goal of creating the "smart environments", in a narrow space, a large number of sensors are being used and transmission packets in the same time also increases. They have caused an increase traffic in WSN. When there is a sudden surge of simultaneous data transmission, problems arise. This type of surge causes traffic overload at a full buffer at the receiving node. This state is called traffic congestion, which is the main cause of packet drops. Subsequent retransmission of dropped packets may increase overall network delay, and cause high latency. Traffic congestion avoidance became the big problem in WSN and congestion avoidance became an essential topic for researchers. There is a variety of studies in traffic congestion detection and traffic control. Almost all of the past studies focused on reducing the traffic congestion without demanding the accuracy in the received data. Moreover, the analysis and calculation almost take place in the computer. With the more and more increase in sensor node's features, instead of sensing and forwarding data only, this thesis proposes an optimal data interpolation method, referred to as the Approximation Equation Method, and an adaptive method to adjust data reporting interval and the data queue length in accordance with permissible error criteria, which is done in a low computational power yet low electric power wireless transceiver.

1.1.1 Home Wireless Sensor Network

A wireless sensor network (WSN) is a collection of nodes organized into a cooperative network. In a WSN, we usually have a numbers of sensor nodes, which are monitoring physical or environmental conditions, such as temperature, human sensing or electricity consumption data, and one sensor sink, which is collecting data from all of sensor nodes through the network. Both sensor nodes and sensor sinks consist of processing capability (such as micro-controllers, CPUs, or DSP chips), a RF transceiver, a power source and an electronic circuit for interfacing with the sensors[1]. In multi-hop WSN, there're also routers, which are only transferring the other nodes' data to other routers or sensor sink, or both sensing data by itself and transferring the other nodes' data.

Home Wireless Sensor Network (Home WSN) is one type of Wireless Sensor Network, which is applied in a narrow area such as home, building. Most of Home WSN is a system of large number of nodes (1000s or even more nodes) in a multi-hop network.

In a basic example of home WSN system (Fig. 1.1), a large number of sensor nodes measures and sends data to sensor sink through routers in WSN. Sensor sink connects to the Internet, save data to the Data Base. Applications use data in the Data Base to do different works, such as data analysis, data calculation, data interpolation, etc...

In the practical system there're also 3 or 4-hop networks containing a large number of sensor nodes. In the narrow areas such as a building, with a multi-hop WSN system of 1000s or even more nodes, traffic throughput is always very high. Therefore, traffic congestion is always a big problem of Home Wireless Sensor Network.

1.1.2 Fixed-time Data Reporting Interval System

In Home Wireless Sensor Network, there are a lot of systems, in which data are reported by sensor node in Fixed-time interval, such as one data per minute. Especially in environmental sensing and smart grids, most of the WSN Systems is

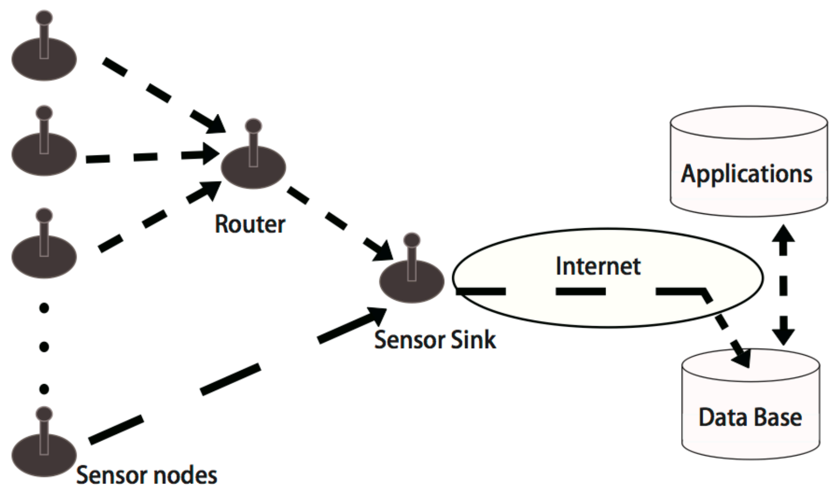


Figure 1.1: Home Wireless Sensor Network System Graph

the Fixed-time Data Reporting Interval System.

Table 1.1 shows one example of the data base of the Fixed-time Data Reporting Interval System. It is one part of the Data Base of RACOW Project that will be described in the next chapter. As shown in the table, the data of both sensors had been reported one data per minute.

Nowadays, for meeting the requirement of real-time sensing, it is required that data be reported in a short time interval. That means the number of reported data is increasing. The more data reported by sensor node, the better quality of real-time sensing we have. Large number of transmission packets in the same time has caused an increase in WSN's traffic.

1.2 Challenge and Research Goal

As described above and noted in [2], to avoid traffic congestion, sensor nodes can not report data in a short time interval, as it is required. However, small number of sensor reporting data causes the decrease in quality of data interpolation. Focusing on remedying this problem, this thesis proposes an optimal data interpolation method, referred to as approximation equation's method, and an adaptive method

Day	Time	EPCID	Data
5/11/2011	11:59:23 PM	epc:id:sgtin:0457122707.101.1	18
5/11/2011	11:59:23 PM	epc:id:sgtin:0457122707.102.3	33
5/11/2011	11:59:24 PM	epc:id:sgtin:0457122707.101.1	19
5/11/2011	11:59:24 PM	epc:id:sgtin:0457122707.102.3	20
5/11/2011	11:59:25 PM	epc:id:sgtin:0457122707.101.1	19
5/11/2011	11:59:25 PM	epc:id:sgtin:0457122707.102.3	10

Table 1.1: Sample of Fixed-time Data Reporting Interval System Data Base

to adjust data reporting interval and the data queue length in accordance with permissible error criteria. With the permissible error criteria in data interpolation, this method proposes to decrease the number of the necessary transmission packets. This way addresses the efficiency on congestion avoidance.

1.3 Structure of Thesis

The rest of this thesis is organized as follows.

Chapter 2 describes this research's related techniques, related project and the position of the research in the related project. Chapter 3 describes the related research and existing method that solves the same problem with this research. Chapter 4 presents this research's approach which uses Approximation Equation method as a solution. Chapter 5 describes the implementation of this research in ZigBee Multi-hop Sensor Network. Chapter 6 discusses results and evaluations from experiments. Finally in chapter 7 the conclusion is presented.

Chapter 2

Related Project and Techniques

This chapter presents the RACOW Project that is closely related to this research, and this research's role in the project. Next, IEEE 802.15.4 standard and ZigBee protocol will be introduced. Afterwards, the reason why ZigBee is being used in RACOW Project and this research's experimental implement will be discussed.

2.1 RACOW Project

This research is related to a RACOW Project [3]. RACOW is an abbreviation of RFID Auto-Commission Open system with WiMAX. This project using the second supplementary budget in the fiscal year 2009 of Japanese Ministry of Public Management, known as a "Promotion project of network integration control system standardization etc.", is a demonstration of reduction of environmental effect with data collection system over WiMAX.

2.1.1 RACOW Project's Goal

This project aims to provide a system in which consumers can freely add and select energy and information services related to electronic equipment and devices. In this project's home sensor network, ZigBee wireless network is being used to

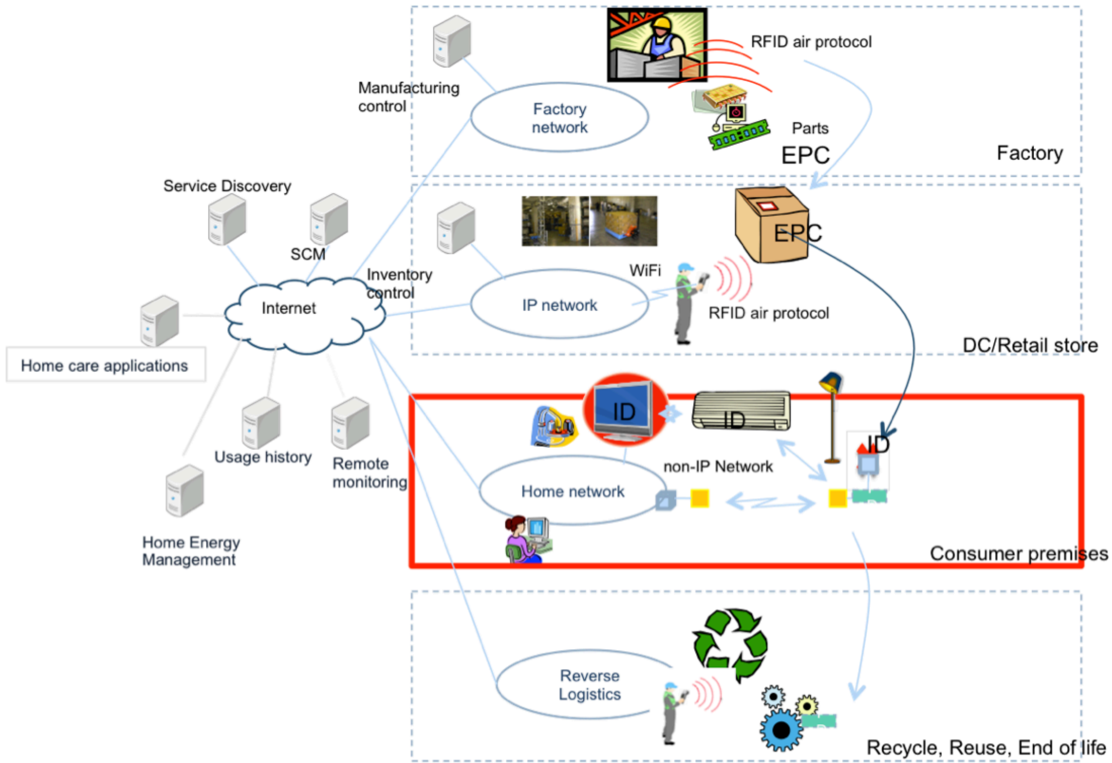


Figure 2.1: RACOW Project Construction System Graph

provide connection between domestic electronic devices, such as lamps, televisions, and refrigerators.

2.1.2 This Research's Role

In RACOW Project, with the large number of sensor nodes used and the requirement of real-time sensing, traffic controlling becomes a mandatory problem to solve. This research is proposed to support the congestion avoidance and give the optimal method for application when doing the data interpolation.

2.2 IEEE 802.15.4 and ZigBee

2.2.1 IEEE 802.15.4

IEEE 802.15.4 is a standard for specifying the physical (PHY) and media access control (MAC) layers of a personal area, low-power, wireless network [4], which was created and is maintained by IEEE [5].

2.2.2 ZigBee

ZigBee is a new standard developed by the ZigBee Alliance for personal-area networks (PANs). Consisting of more than 270 companies (including Freescale, Ember, Mitsubishi, Philips, Honeywell, and Texas Instruments), the ZigBee Alliance is a consortium that promotes the ZigBee standard for a low-rate/low-power wireless sensor and control network [6]. ZigBee is the primary protocol which builds on the 802.15.4 standard, adding a network layer capable of peer-to-peer multi-hop mesh networking, a security layer capable of handling complex security situations, and an application layer for interoperable application profile.

ZigBee specifies all the layers above MAC and PHY, including the NWK layer, APS, ZDO, and security layers. IEEE 802.15.4 does not specify anything to do with multi-hop communications, assigning address, or application level interoperability (Fig. 2.2). It's ZigBee that provides the mesh networking and multi-hop capabilities, enhances the reliability of data packet delivery, and specifies application-to-application interoperability [4].

The ZigBee Alliance developed the following application profiles:

- Smart energy
- Commercial building automation
- Home automation
- Personal, home, and hospital care
- Telecom applications
- Remote control for consumer electronics
- Industrial process monitoring and control

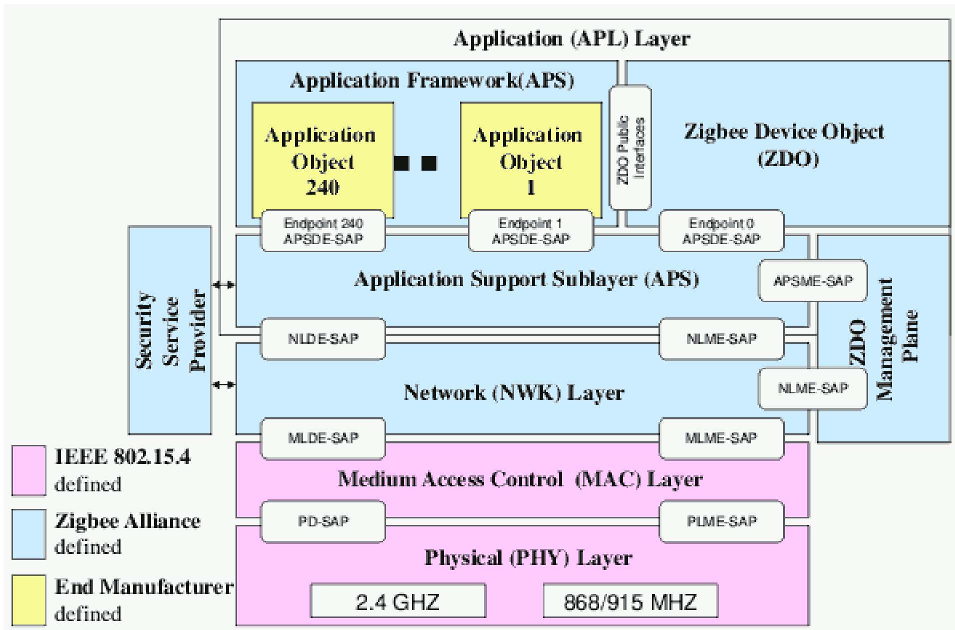


Figure 2.2: ZigBee Stack Architecture

In this research, ZigBee Wireless Network is being used as a Home WSN. As described above, ZigBee is also being used in RACOW Project because of these following properties.

- Low battery consumption

  A ZigBee End Device can be in sleep mode and still keep its association with its network. Therefore it should operate for months or even years without needing its battery replaced.

- Multi-hop network

  In tree and mesh topologies, ZigBee provides a multi-hop network; packets pass through multiple hops to reach their destination.

- Automatic routing

  ZigBee can automatically establish its network.

2.3 Data Interpolation

Interpolation is a method for estimating and constructing new data points from a discrete set of known data points. Sometimes we know the value of a function $f(x)$ at a set of points $x_1, x_2, \ldots, x_N$, but we do not have an analytic expression of $f(x)$ that let us calculate its value at an arbitrary point. The task now is to estimate $f(x)$ for an arbitrary x by drawing a smooth curve through (and perhaps beyond) the x_i. If the desired x is in between the largest and smallest of the $x_i's$, the problem is called interpolation [7].

In engineering, especially in WSN, the sensing data is simply a discrete set of data points sensed over time. Moreover, the number of data sensed by a sensor node is always much bigger than the number of data that can be reported to sink. Therefore, it is required to interpolate the intermediate value of the independent value from the data which is reported to sink.

2.4 Summary

Chapter 2 introduces the RFID Auto-Commission Open system with WiMAX (RACOW) Project, in which ZigBee Wireless Network is being used to provide a connection between electronic devices in home. IEEE 802.15.5 and ZigBee's properties and the reason why ZigBee is being chosen are also discussed. Lastly, data interpolation, the technique that is ly used in science and engineering, is presented.

Chapter 3

Related Researches and Existing Methods

The first section of this chapter shows some related researches. They are the studies about solving the problem of traffic congestion in Wireless Sensor Network. Afterward, the existing methods, which solve the same question with this research, are presented. Problems of each existing method are also addressed.

3.1 Related Research

3.1.1 Traffic Congestion Detection and Avoidance in Wireless Sensor Networks

Traffic Congestion Detection and Avoidance in Wireless Sensor Networks is described a lot in literature. Sensor network diagnosis is used to identify and be aware of the network condition that helps a lot in traffic congestion detection and avoidance. Some studies propose to monitor sensor network by collecting the information from sensor nodes. Those information is the aggregates of parameters of each residual energy of each sensor [8] or the marking data created by the proposed marking scheme identifying the packet routing part [9]. The network condition analysis done in the sink and the configuring of sensor node transmission frequency are proposed [10] [11] [12]. Some researchers propose a study aiming at improving network efficiency in congested networks, by demonstrating that choosing a suitable packet size can help to significantly reduce the packet loss rate and the number of retransmissions required. It is based on the theory that transmitting a smaller packet size with a high Bit Error Rate and a much bigger packet size with a low Bit Error Rate helps reduce transmission errors [13]. The investigators in this research only aim at configuring the suitable packet size but not concentrate on reducing errors on the receiving end when doing the data interpolation.

3.1.2 Sensor Data Interpolation

The research in [14] presents the Minimum Mean Square Error (MMSE) interpolators based upon a model of the image capture system. After being resolved by using the optimal interpolator, the observed image taken by the low resolution capturing system is proposed to provide the optimal of the desired, high resolution image. In this work, the researchers said that their experiments indicate that optimal interpolation outperforms cubic interpolation in terms of subjective image quality, without a significant increase in computational cost.

3.2 Existing Method

3.2.1 Simple Method

In solving the problem described in chapter 1.2, the Simple Method only decreases the transmission frequency, in other words, lengthen the data reporting interval. In fixed time interval, when data sensing frequency is higher than data transmission frequency, the number of sensing data is bigger than the number of data that can be transmitted. Simple Method sends only one sensing data (raw data which is sensed by sensor node) in one fixed time interval to guarantee the avoidance of traffic congestion in WSN. Therefore, in one fixed time interval, only one sensing data can be reported and saved to data base, the other sensing data is all omitted.

Figure 3.1 shows one example of using the Simple Method in Home WSN System. It is a figure of temperature data that are sensed by one sensor set in the author's laboratory room. In normal condition, data are sensed and reported once per minute by a sensor node. That means sensor sink receives have sensing data each minute. To avoid traffic congestion, the Simple Method reduces the sensing data transmission frequency. Instead of one data in one minute, it lengthen the data reporting interval, such as one data in 16 minutes (as in Fig. 3.1). When data in a specific time is required, if that data was not reported (in the other words, that data is not in the Data Base), application has to interpolate that data. In Simple Method, data in a specific time that was not reported is defined as a point in the straight-line between two data, which were reported, are in the nearest time with that specific time. In figure 3.1, data in 13:00 is defined by a point in the straight-line between data in 12:48 and 13:04. That estimated data involves the interpolation error. The more data are omitted, the more the interpolation error we have.

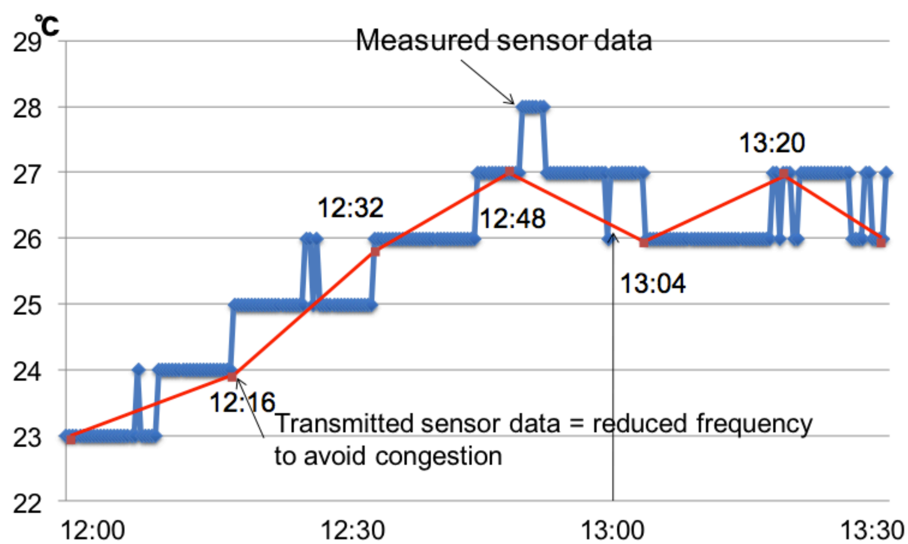


Figure 3.1: Example of the Simple Method in Temperature Sensing System

3.2.2 Store and Forward Method

Based on the fact that the header of Over-the-Air Data frame is quite large (in ZigBee, the OTA Data Frame is up to 127 bytes and the header is up to 47 bytes) (Fig. 3.2), as described in [15], instead of sending data immediately, storing some data in a sensor node and forwarding all at once helps decrease transmission packet number that supports traffic avoidance (Fig. 3.3). However, this method has its own problem. By sending all the sensing data, this method provides the best accuracy when doing the data interpolation. Nevertheless, with the fixed OTA data payload length, the sensor node can store only the extremely limited data number that does not help a lot in traffic avoidance.

3.3 Summary

There has been a variety of research on how to recognize and prevent congestion in wireless sensor networks. There is also an existing method for traffic congestion avoidance by reducing transmission packet number that is done by a sensor node. However, the requirement of data interpolation accuracy and the automatically configured transmission packet number have not been discussed.

This research focuses on remedying the existing method's problems. For avoiding traffic congestion in Wireless Sensor Network, an Adaptive Time-wise Data Interpolation is presented. They are an optimal data interpolation method, referred to as approximation equation 's method, and an adaptive method to adjust data reporting interval and the data queue length in accordance with permissible error criteria.

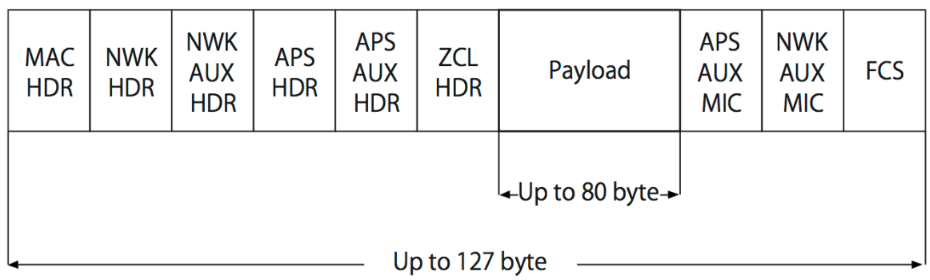


Figure 3.2: ZigBee Over-the-Air Data Frame

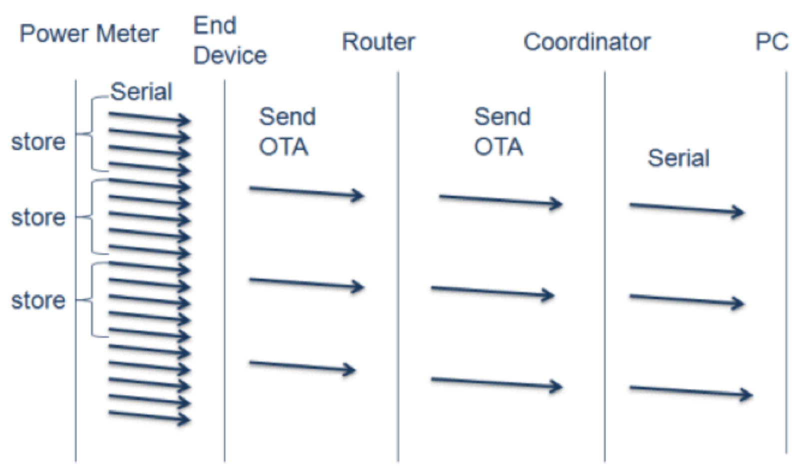


Figure 3.3: Store and Forward Data Sending Diagram

Chapter 4

Approximation Equation Method

This chapter presents Approximation Equation Method - an optimal data interpolation method for sensor nodes to calculate and report sensing data in WSN. In the first section, this method's main goal will be described. Afterwards, in the second section, this method's theory will be presented. In this section, the way to calculate the approximation equation and the problem of the approximation degree will be introduced. Lastly, in the third section, the evaluation on the efficiency of approximation equation degree will be shown.

4.1 Approximation Equation Method's Goal

In a high traffic throughout WSN, to avoid the traffic congestion each sensor node has to lengthen the time reporting interval. Simple Method (as described in section 3.2.1) shows good results only where small sensor data fluctuation is observed. Where large sensor data fluctuation is observed, this method makes high error rate when doing the data interpolation. However, shortening the time reporting interval will increase the data packet transmission, which causes the traffic congestion's capability.

In this research, an optimal data interpolation method for sensor nodes is proposed. This method doesn't increase the sensor node's transmission but optimizes the reporting data. By using my method, sensor nodes do not send the raw sensing data as in Simple Method but send the data providing better result when doing data interpolation in application. Figure 4.1 shows the imagination of using Approximation Equation Method in the temperature sensing system. Instead of real sensing data, the sensor node sends optimal data. When application does the data interpolation, such as interpolating the data at 13:00, the proposal method will have less interpolation error than the Simple Method.

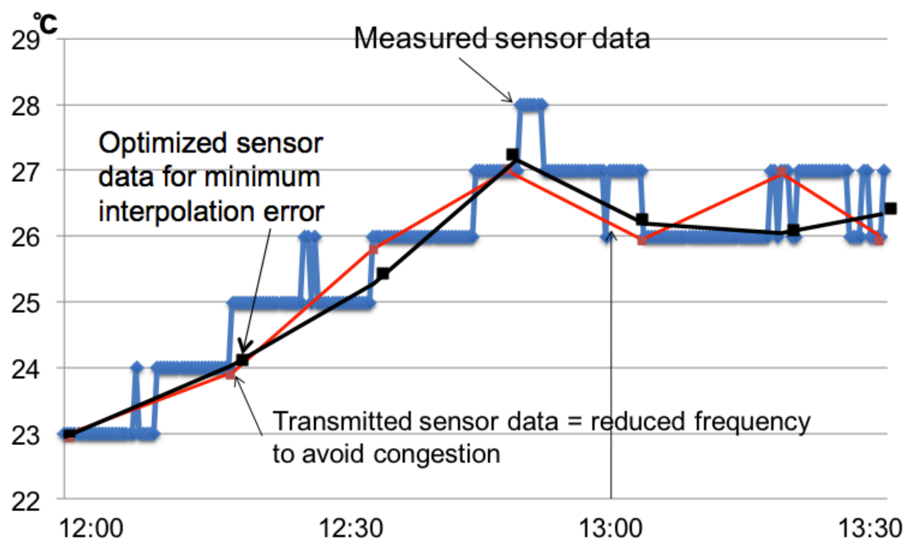


Figure 4.1: Imagination of the Approximation Equation Method

In the following, the way of calculating the Approximation Equation Method and setting the approximation equation to provide both the desirable result in data interpolation and the good performance in a sensor node, which has very lack of memory and processing capability, are presented.

4.2 Approximation Equation Method's Theory

In a sensor node, although the data sensing frequency is higher than the data transmission frequency, by using this method sensor node doesn't omit any sensing data. The sensor node uses all sensing data to calculate an approximation equation. After that it reports one data calculated by the approximation equation to the sensor sink.

Let's imagine that a sensor node has $k+1$ sensing data (x_i, y_i) $i = 0 \ldots k$ and needs to calculate the approximation equation of degree n (Formula 4.1) of that $k+1$ sensing data

$$y = \sum_{i=0}^{n} a_i x^i = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0 \tag{4.1}$$

To have the least error when doing the data interpolation, the error distribution of real sensing data and the data that are calculated using the approximation equation (formula 4.2) must be at minimum.

$$I = \sum_{i=0}^{k} (y_i - \sum_{j=0}^{n} a_j x_i^j)^2 \tag{4.2}$$

$$I = \sum_{i=0}^{k} (y_i - \sum_{j=0}^{n} a_j i^j)^2 \tag{4.3}$$

Because $k+1$ data are continuous, we can simply assign x_i as i. Formula 4.2 becomes formula 4.3.

Formula 4.3 has $n+1$ variables: $a_n, a_{n-1}, \ldots, a_0$. Sensor node has to calculate $a_n, a_{n-1}, \ldots, a_0$ to find the approximation equation of these k data. In formula 4.3, I is at the minimum when all the differential of I by $a_i (i = 0 \ldots n)$ are 0.

However, in this method, to guarantee that application's data interpolation can work well, the first data of $k+1$ data y_0 must be the data that was sent to sensor sink right before these $k+1$ data are being sensed. In the other words, y_0 must be the latest data saved in the database. Therefore, based on the (x_0, y_0), x_0 = 0, we have an equation of one variable related to other n variables. Now, we only have to calculate n variables, for example, $a_{n-1}, \ldots, a_0$. We take the differential of I by $a_i, i = 0 \ldots n-1$ and have n equations:

$$\frac{\delta I}{\delta a_i} = 0, i = 0 \ldots n-1 \tag{4.4}$$

After calculating the approximation equation of these k data, sensor node can calculate all the new data $\hat{y_i}$ of y_i.

Each equation with different degree requires a different quantity of information for the application to know about the equation and do the interpolation.

In the simplest case, the linear equation, sensor node only needs to send one data in one time, because the linear equation can be calculated based on the 2 points (2 data). In this case, sensor node sends the new data $\hat{y_k}$ of y_k. Now in the database we have two data y_0 and $\hat{y_k}$. Application can calculate the equation using the $\hat{y_k}$ conjunction with the y_0. In higher degree equation, sensor node needs to send more than one data, such as square equation requires two data, cubic equation requires three data.

By identifying the equation, application can interpolate all the data which are not reported.

4.3 Efficiency of Approximation Equation Degree

Higher degree of the approximation equation provides more variables. So it provides the better accuracy in data interpolation. However, higher degree will require more computational complexity burden to sensor node. In sensor node, which has lack of memory, overflow is possible.

This section shows the evaluation on the efficiency of approximation equation degree in data interpolation. The external temperature data of Z building of Keio University Shonan Fujisawa Campus in one week (05 $\sim$ 12/May/2011) were being used to do this simulation. The sensing data is one data per minute but the sensor node is set that only send one data per fixed number minutes.

Error distribution average is calculated as below.

$$e\bar{r}r = \frac{I}{k} = \frac{\sum_{i=0}^{k}(\hat{y_i} - y_i)}{k} \tag{4.5}$$

y_i : the real sensing data.

$\hat{y_i}$: the data after doing the data interpolation. The way to do the data interpolation will be described after.

As described in section 4.2, each equation with different degree requires a different quantity of sensing data. Therefore, higher degree equation has to calculate with larger number of data. For example, with the set data reporting interval is 1 data per 20 minutes, linear equation calculates with 21 data each time but square equation calculates with 41 data each time.

When doing the data interpolation, with the linear equation, it uses 2 data and the data between that 2 data is calculated as a point in the linear line between that

2 points. With the square equation, it used 3 data. The data, which is not received, is calculated as a point in the square line of the square equation calculated before.

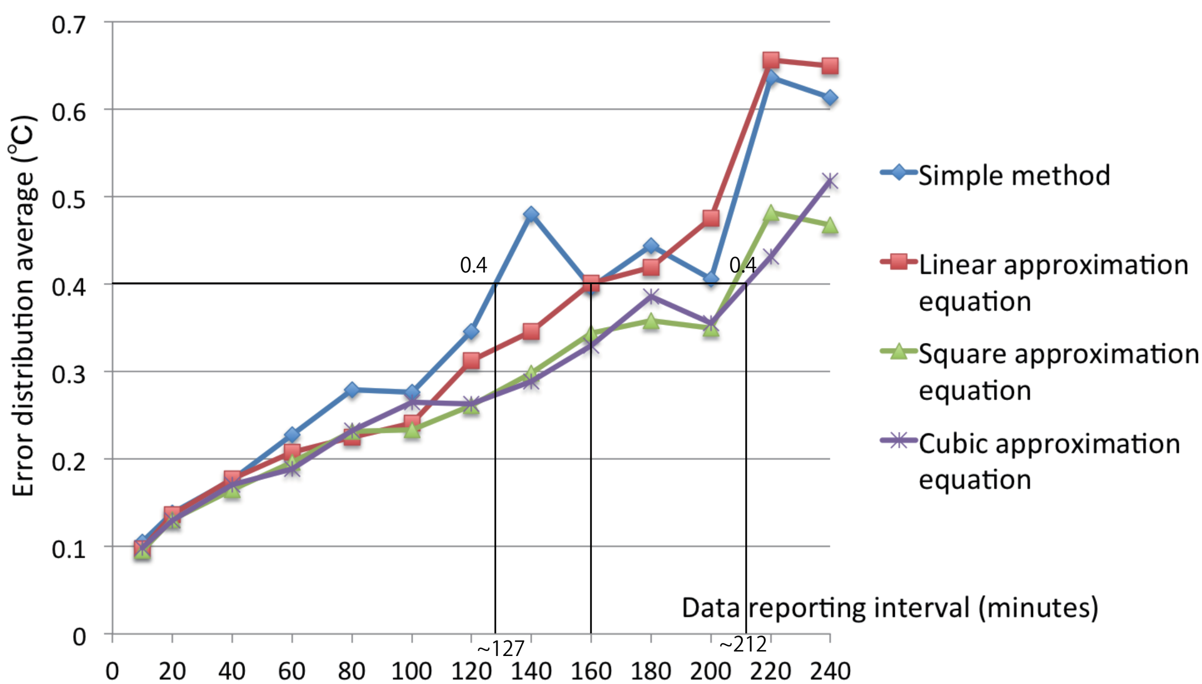


Figure 4.2: Evaluation on the Efficiency of Approximation Equation Degree

In this simulation, with the same error distribution rate at 0.4, the cubic approximation equation method provided 1.7 times longer the reporting time interval than that of the Simple Method. Even the linear approximation equation method provided 1.25 times longer the reporting time interval than that of the Simple method. (Fig. 4.2)

4.4 Summary

In this chapter, the proposal approximation equation method's goal and theory are presented. Lastly, the evaluation on the efficiency of approximation equation degree is discussed. This evaluation is applied to a room temperature monitoring system. It is shown that the approximation equation's method can archive 1.7 times longer the reporting time interval than that of the simple truncation. Even the linear approximation equation method can archive 1.25 times longer the reporting time interval than that of the simple truncation. The success of the approximation equation's method motivated the author to develop an adaptive control of reporting time interval under predefined error threshold, which will be described in the next chapter.

Chapter 5

Implementation

Traffic in a Wireless Sensor Network can further be mitigated if sensors adaptively changes its reporting time interval while the accuracy in the receiving application is preserved. This observation leads the author to develop the an adaptive control of reporting time interval under predefined error threshold with linear interpolation in sensor node. This chapter presents an implementation in Texas Instrument CC2430. The first section describes the design of this method in the experiment. The next section shows the structure and the detail of each parts of the system.

5.1 Implementation Platform

As mentioned in the chapter 2, ZigBee is used as the home wireless sensor network environment in the implementation experiment. Texas Instrument's CC2430s are used as the data transceivers. The CC2430 is the first single-chip, IEEE 802.15.4 compliant and ZigBee SoC (System on Chip) RF transceiver with integrated micro-controller. It provides a highly integrated, flexible low-cost solution for applications. Figure 5.1 shows the diagram of the CC2430. Table 5.1 shows the common features of the CC2430. [16]

CC2430EM (CC2430 Evaluation Module), a small plug-in module for Smart RF04EB, is used as a reference design for antenna and RF layout. Smart RF04EB is a motherboard for Evaluation Module. It provides a RS232 interface for com-

RF/Layout	2.4 GHz IEEE 802.15.4 compliant RF transceiver
Microcontroller	8051Microcontroller core
	32/64/128 KB flash memory
	8 KB RAM
Peripherals	2 USARTs, support for serial protocols
Development tools	ZigBee protocol stack

Table 5.1: The CC2430's Features

munication with sensors for receiving sensing data (in sensor node) or with PC for sending data (in sensor sink) (Fig. 5.4). Z-stack, a free-download protocol stack for ZigBee provided by Texas Instrument, is used to implement the CC2430 chip. IAR Embedded Workbench is used as the IDE.

5.2 Implementation Design

As the result of the evaluation on the efficiency of approximation equation degree that was mentioned in the previous chapter, linear equation is chosen to take the implementation. Let's take a deep look at how the linear equation is calculate in the sensor node.

As describe before, in linear equation's case, we have two variables: a_1, a_0. Because the equation has to contain the point $(0, y_0)$ so there is only one variable need to calculated. That leads the thinking that instead of calculating the variable a_1 or a_0, we calculate directly the new data $\hat{y_k}$ of y_k.

Formula of the linear equation containing $(0, y_0)$ and $(k, \hat{y_k})$ is in formula 5.1.

$$y = \frac{\hat{y_k} - y_0}{k} x + y_0 \tag{5.1}$$

We can calculate the error distribution I as in 4.3 and the differential of I by $\hat{y_k}$.

Let the differential of I by $\hat{y_k}$ is 0 so we can have the equation of $\hat{y_k}$.

To prevent overflow in a micro-controller, the formula is simplified. We have the equation of $\hat{y_k}$ as in formula 5.2.

$$\hat{y_k} = \frac{6\sum_{i=0}^{k}(i * y_i) - y_0 (k+1)(k-1)}{(k+1)(2k+1)} \tag{5.2}$$

$\hat{y_k}$ can be calculated using the real sensing data y_i (with $i = 0 \ldots k$).

5.3 System Implementation

This system is divided into two parts:

- Approximation equation calculation and reporting data module in sensor node
- Reporting interval controlling and data receiving module in sensor sink

The details of each part are described as followings.

5.3.1 Approximation Equation Calculation and Reporting Data Module in Sensor node

Main function of this module is calculating data using the Approximation Equation Method, comparing with set limitted error rate, and sending data to sensor sink. Another function of this module is changing the set error rate based on the data receiving from sensor sink. Afterward, each function will be described.

Changing error rate function

In sensor node, there is a variable named $errorSet$. It is the limited error distribution average (EDA). In the other words, it is the highest acceptable error distribution average, which is provided by the sensor sink. The way to calculate the error distribution average is described in the equation 4.5. Sensor node calculates and sends the optimal data so that when application interpolates the data, which isn't received, using the optimal data the error distribution average is not higher than the $errorSet$.

When the sensor sink sends the new value to sensor node, this function receives that data and change the value of the variable $errorSet$.

Data calculating and reporting function

This function's work is calculating an optimal data based on the sensing data and the limited error distribution average $errorSet$ (as described before). Sensor node senses data, calculates, and reports data so that the data reporting interval can be long as much as possible, but still satisfies the limited error distribution average $errorSet$.

First of all, when receiving the first data, sensor node sends that data to sensor sink and saves that value to $y0$ (will be used before when calculate the adaptive data using the Linear Approximation Equation Method).

In sensor node, there is a queue using for storing data. The reason why the queue is being used and the efficiency of the number of data in the queue will be described in the next chapter. From the second data, when receiving new data, sensor node checks the number of data in the queue. After it matches the set number $queuenumber$, sensor node calculates the optimal data, which can be sent to sensor sink if the EDA matches the limited value, based on the Linear Approximation Equation Method. After that, it calculates the error distribution average and compare with the limited value. If it is higher than the limited value, sensor node will calculate and find the previous latest optimal data, when the EDA still smaller than the limited value, sends it to sensor sink and saves that value to $y0$. All the data in the order before that optimal data will be removed from the queue. If it is not, sensor node saves the sensing data in the queue and waits for the new data. (Fig. 5.2)

There are 4 important variables in this module.

• $y0$

This variable defines the previous data that were sent to sensor sink before. By using that variable, sensor node knows the value of the last data sent to sensor sink, which is necessary when calculating the optimal data using the Linear Approximation Equation Method and the error distribution average.

• $queuenumber$

When the number of data in the queue matches this number, sensor node starts calculating the optimal data and the error distribution average. The merit of this variable will be described before in the Evaluation chapter.

• xy

It's a variable used in calculation the approximation equation. In formula 5.2, $\sum_{i=0}^{k}(i * y_i)$ is calculated by this variable. xy is calculated using the data in the queue and the order of those data. Each time the data in queue is changing, xy is being recalculated. To prevent overflow, xy is used as uint32 type.

• $seqnum$

This variable is used to numbering the packet sent to sensor sink. By using this variable, application can define that there is packet loss in WSN or not. If the receiving packets in sink are continuous, there is no packet loss. If it is not, there is packet loss. This thing is very important in WSN and in this method. If there is no packet loss, the data interpolation can satisfy the limited error distribution average value. However, if there is packet loss, doing data interpolation in that time period can not provide the EDA under limited.

In this module, there are 2 important functions doing 2 main works: calculating the optimal data and calculating the error distribution average.

• Calculating the optimal data

The optimal data $\hat{y_n}$ is calculated using this function. When receiving a new sensing data, sensor node checks the number of data stored in the queue. When the number of data in queue reaches the $queuenumber$, this function is called to calculate tne $\hat{y_n}$. As shown in formula 5.2, this function uses the xy, $y0$ variables to calculate the the $\hat{y_n}$. Formula 5.2 has the division, so there is a possible that $\hat{y_n}$ is not an integer. However the sending data is an integer value, so the $\hat{y_n}$ must be converted into the integer value. When calculating using formula 5.2, the numerator is multiplied by 10 to know the value of the first number of the fractional part. If that value is smaller than 5, $\hat{y_n}$ is the decimal part value. If it is not, $\hat{y_n}$ is the decimal part value plus 1.

• Calculating the error distribution average (EDA)

As described before, the error distribution average is calculated as the same as in equation 4.5. Because the Linear Approximation Equation Method is being used, the $\hat{y_i}$ is calculate based on the last data sent to sensor sink $y0$ and the new optimal data $\hat{y_n}$. $\hat{y_n}$ is calculated as below.

$\hat{y_i} = (\hat{y_n} - y0) * i/number$

The packet sent from sensor node to sensor sink contains: $seqnum$ and the calculated optimal data.

Preparing for the evaluation in the next chapter, the method without using the approximation equation is also implemented. In this module, there is not the "calculating the optimal data" function. It simply sends the pervious raw sensing data if the EDA is higher than the limited EDA (Fig.5.3).

5.3.2 Reporting interval controlling and data receiving module in sensor sink

In sensor sink, there are two modules that do the data receiving and reporting interval controlling works.

- Data receiving module

  This module's function is receiving data, which is sent Over-the-Air from sensor nodes, and sending those data to computer via RS232C port.

- Reporting interval controlling module

  This module's function is sending commands from sensor sink to sensor node. The command is received from RS232C port, which is connected to a computer. In this experiment, the commands are the limited error distribution average value and number of data in the queue in sensor node. When sensor nodes receive those values, it changes its $errorSet$ and $queuenumber$.

There is also a "receiving" application in the computer that connecting to the sensor sink. This application does two works:

- Receiving data from sensor sink

  The data is received via the RS232 port. The data is saved to Data Base.

- Sending command to sensor node

  The commands are inputted by user. This module converts and sends commands to RS232 port. Through the RS232 port, they are sent to sensor sink. The commands are for configuring a limited EDA and the number of data saving in queue in each sensor node. Before sending command to sensor node, sensor node network address must be inputted. Sensor sink received this value when sensor node joined the network at the first time. Sensor node sends its IEEE address together with its network address. Using those value, sensor node can identify which node has which network address.

  This application was written in C Sharp. (Fig. 5.5)

Table 5.2 shows the implement environment of the “receiving” application.

Laptop PC	Lenovo Thinkpad T510
OS	Window XP Professional SP2 (32bit)
Program language	C Sharp

Table 5.2: Environment of the “receiving” application

5.4 Summary

In this chapter, I have discussed in detailed the implementation of an adaptive method to adjust data reporting interval and the data queue length in accordance with permissible error criteria using the Linear Approximation Equation Method in TI CC2430 chip. The calculation function is simplified to implement in a low computational power yet low electric power wireless transceiver. The application that receiving data and sending commands in sensor sink has been presented. In the next chapter, the system's evaluation will be discussed.

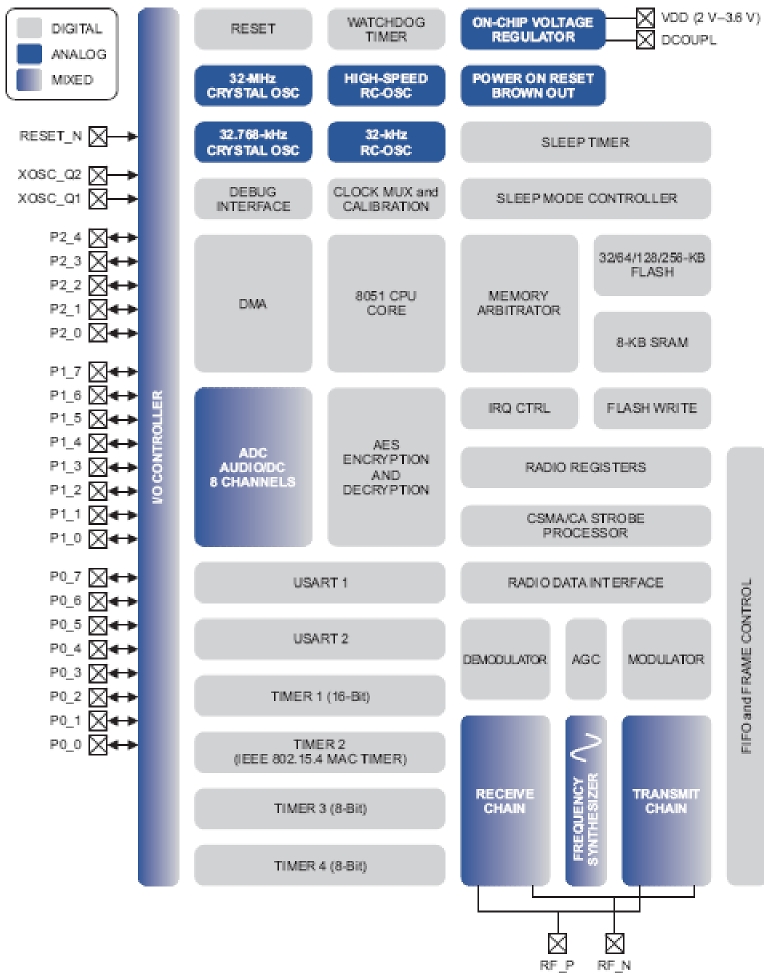


Figure 5.1: CC2430's Diagram

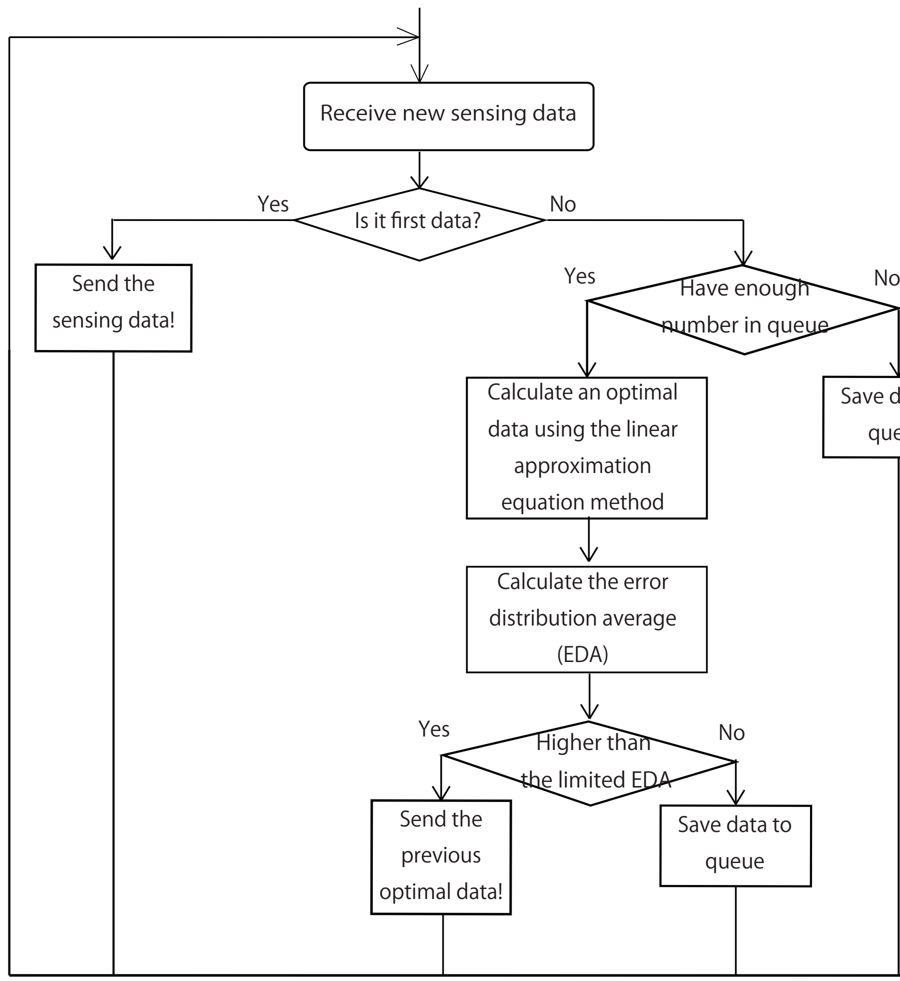


Figure 5.2: Flow chart of sensing data solving module in sensor node using the Linear Approximation Equation Method

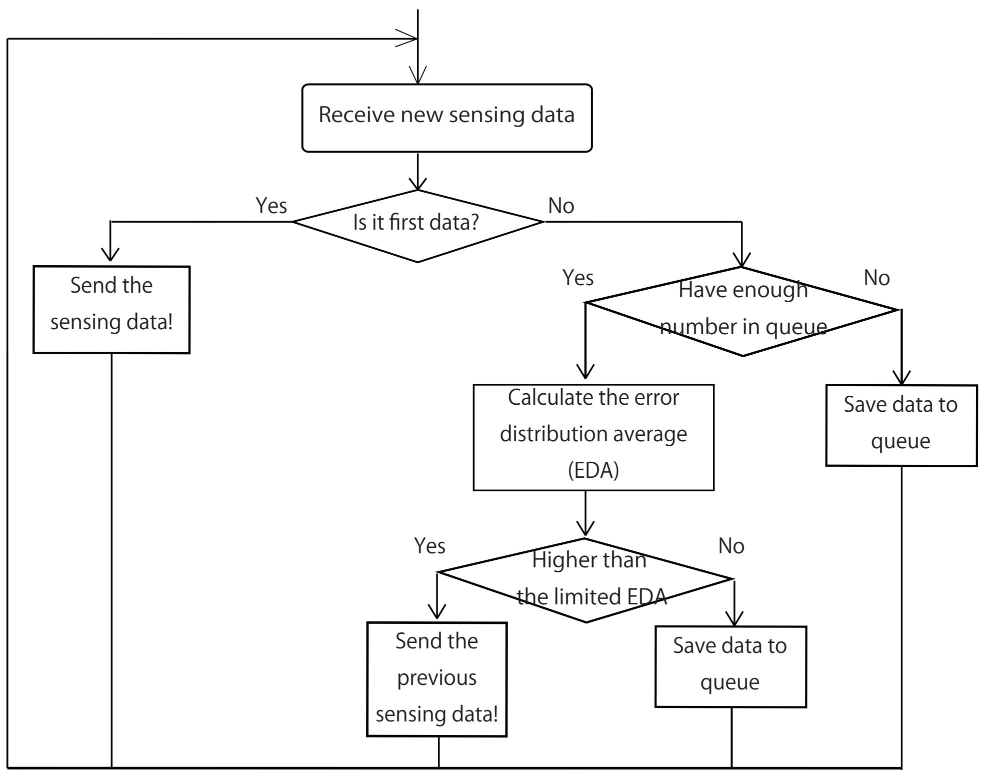


Figure 5.3: Flow chart of sensing data solving module in sensor node without using the approximation equation

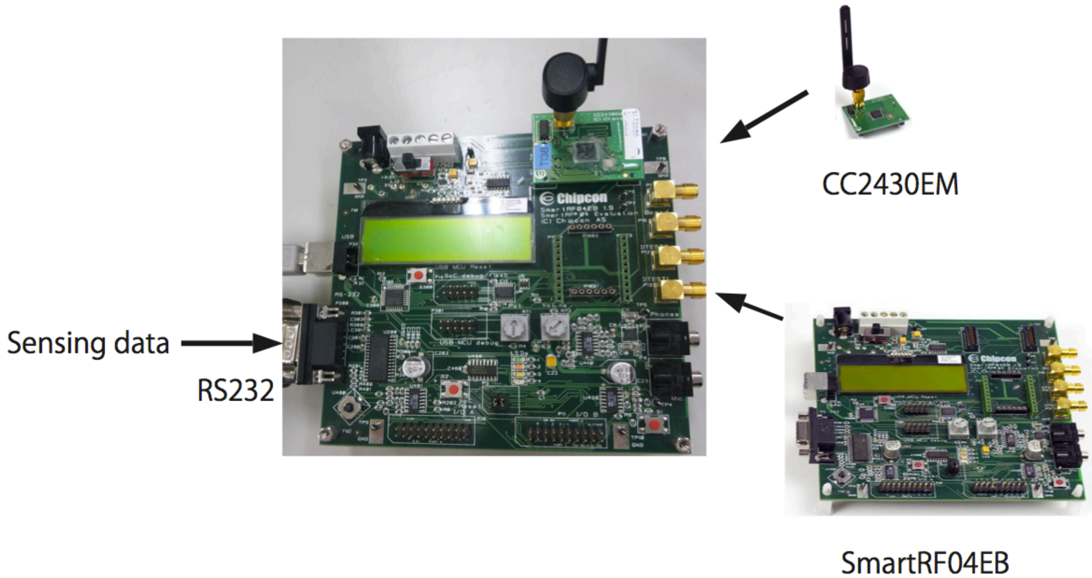


Figure 5.4: CC2430 Kit

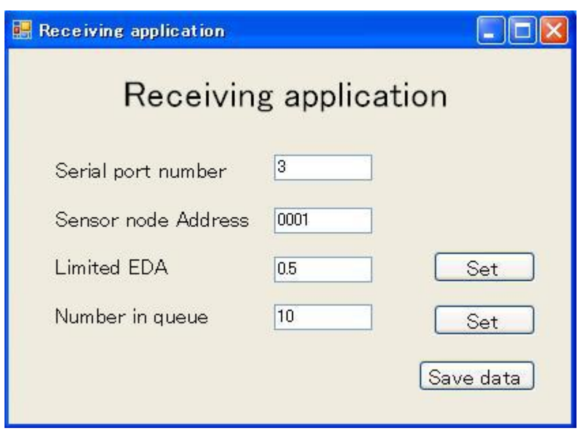


Figure 5.5: Receiving Application GUI

Chapter 6

Evaluation

This chapter demonstrates the result of evaluations on the Linear Approximation Equation Method and the efficiency of the length of the data queue in a sensor node.

6.1 Sample Data Preparation

In section 4.3, temperature sensing data was used in the simulation experiment. At that experiment, the method using the Linear Approximation Equation Method shows the better result than the simple method in data interpolation. However, to show the efficiency of using linear approximation equation, not only temperature sensing data but various type of data has to be used to evaluate. This section shows how various type of sample data is prepared to do the evaluation experiment.

All the periodic function $f(x)$ that is integrable on $[-\pi, \pi]$ can be described using the Fourier series as in formula 6.1. [17]

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty}(a_n \cos nt + b_n \sin nt) \tag{6.1}$$

Simplifying the function by setting a_n and b_n with the specified value. Data are created using a function $f(x)$ as below.

$$f(x) = a_0 + a_0 * \sin \pi x \frac{n}{L} \tag{6.2}$$

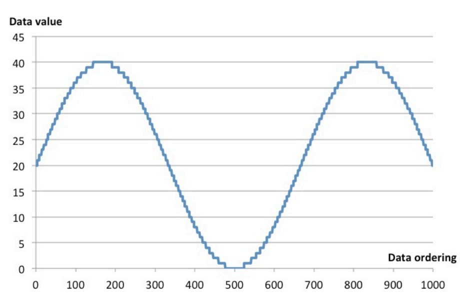
L: number of data. In this time, L is 1000.

n: number of half-periods of $\sin$ function in L data.

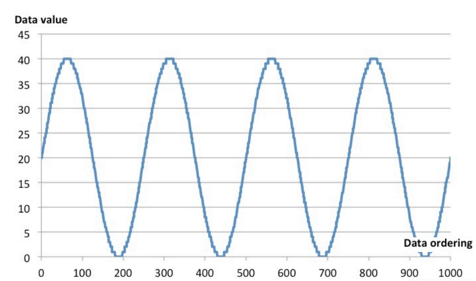
$a(0)$ is used to set the value of data. In this experiment, $a(0)$ is set to 20. So the data is in [0 . . . 40]. The data range is 40.

Using equation 6.2, 15 types of datasets are created with $n = 1 \ldots 15$. It's the number of waves consist in the Fourier series. Each set contains 1000 data.

The sample data preparation program was written in Ruby Programming (around 70 lines). The data is set up so that all data are positive and integer. The sample data are created and saved to text file.



(a) 3 half-periods



(b) 8 half-periods

Figure 6.1: Sample Data

Figures 6.1 show the sample data created with 3 and 8 half-periods.

6.2 Experimental Setup

This experiment uses two CC2430EMs, two Smart RF 04EB Boards and two laptop PCs. Each CC2430EM is set on one EB Board. Each board has a RS232 port used to connect to a laptop PC.

One node works as a sensor node (End-Device). A PC has a "sending" application (written by C Sharp), which reads data from file (created before) and sends data to node via RS232 port. End-Device node receives data and sends to Coordinator via Over-the-Air.

One node works as a sensor sink (Coordinator). This node receives data from End-Device node and sends data to a PC via RS232 port. In this PC, a “receiving” application (as described in the previous chapter) receives this data and save to Data Base. (Fig. 6.2)

The limited EDA and the number of data in the queue in each sensor node are sent to each sensor node at the beginning (Fig. 6.3). Those value are inputted in the application. Sensor sink sends those value to correspondence sensor node with the inputted IEEE Address. (Fig.5.5)

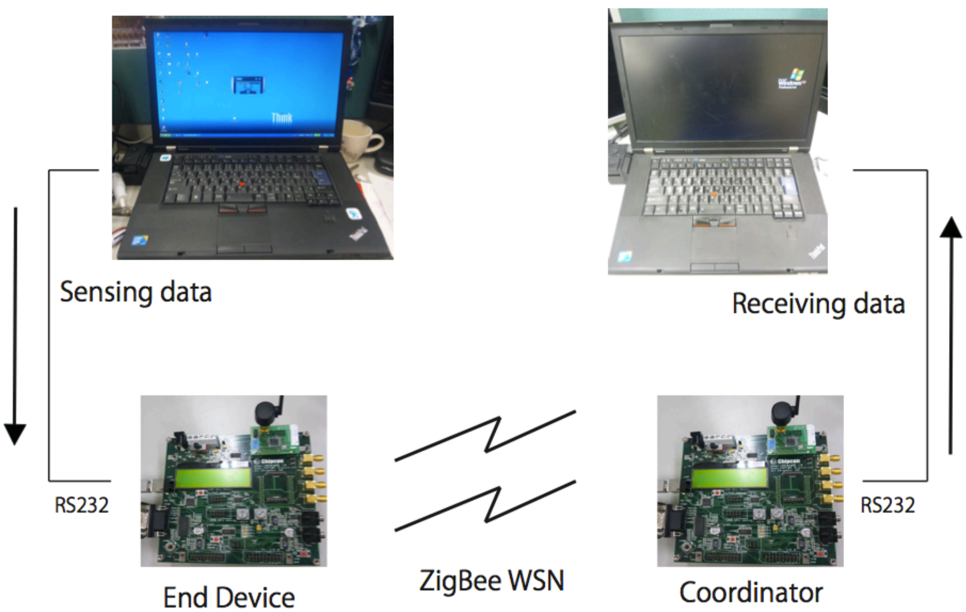


Figure 6.2: Experiment Diagram

6.3 Evaluation Method

In this experiment, both the Linear Approximation Equation Method (Fig. 5.2) and the method without using the approximation equation (Fig.5.3) are implemented to End-Device nodes. End-Device nodes receive the same data (the sam-

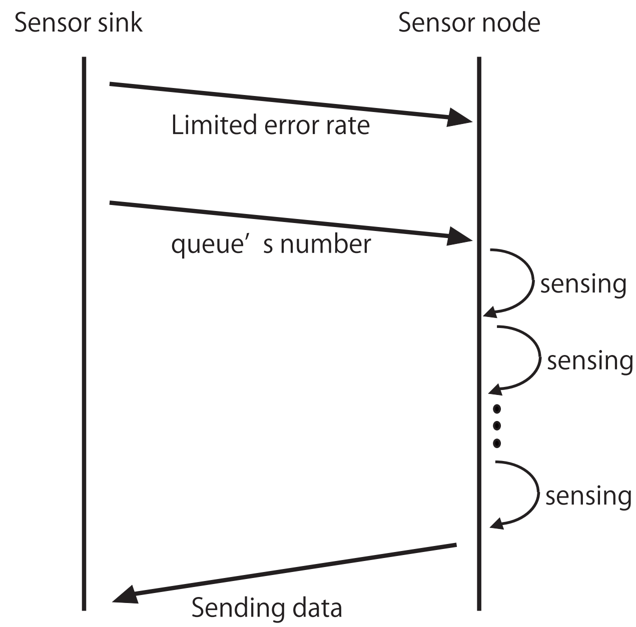


Figure 6.3: System graph

ple sensing data created before), do the calculating (which is different in each method) and send optimal data so that the EDA of data interpolation using the reported data isn’t higher than the limited EDA value.

By comparing the number of transmission packets at the same time period of the two method when using the same data, two evaluations will be taken.

- Evaluation on the efficiency of the linear approximation equation by comparing with the method doesn’t use the approximation equation.
- Evaluation on the efficiency of the number of data in queue in sensor node.

6.4 Result

The first experiment was done when the number of data in queue was set to 1. It means sensor node does not use queue to store data. It calculates immediately after receiving new sensing data. The limited EDA is set to 0.0125%. Because the data range is 40, the limited EDA value set in sensor node is $0.0125 * 40 = 0.5$.

With using the Adaptive Time-wise Data Interpolation Method, instead of sending 1000 data, with the set of little data error rate (0.0125%), the transmission packet number is reduced to $7 \sim 53$ packets. (Fig. 6.4)

In this case, the method without using the approximation equation showed a better result than the Linear Approximation Equation Method did. Using the same set of 1000 data, the number of transmission packets using the method without using the approximation equation was fairly smaller than the number of transmission packet using the linear approximation equation. The difference of the two values becomes largest when the number of half-periods in set of 1000 data is 8. And these two values are almost the same when the number of half-periods in set of 1000 data is 14 and 15 (Fig. 6.5). The Linear Approximation Equation Method did not show a good result in this case.

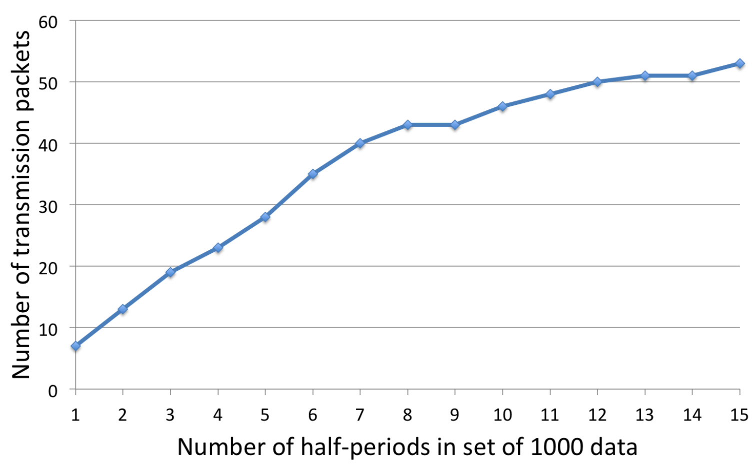


Figure 6.4: The adaptive time-wise Data Interpolation with EDA = 0.0125%, number of data in queue = 1

The result of the first experiment can be predicted as followings.

In this case, after calculating the EDA, if the value is higher than the limited one, the sensor node sends the previous optimal data. When the new data comes, sensor node has 2 data and calculates the EDA using those 2 data and the previous optimal data. Because the previous optimal data of the method without using the

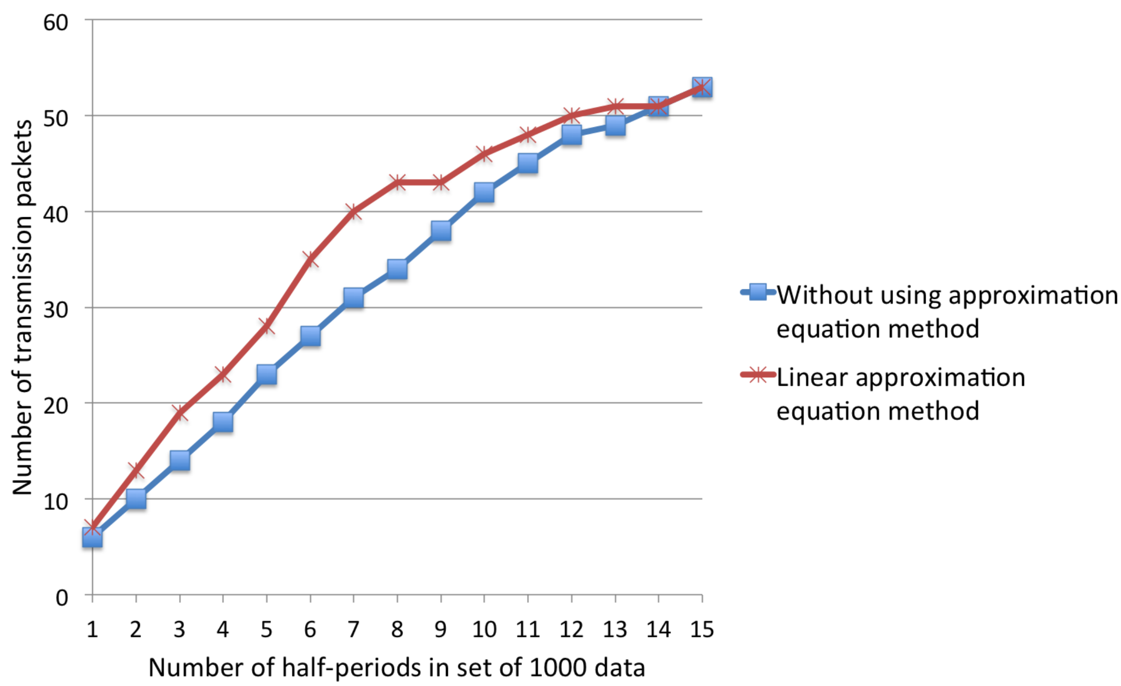


Figure 6.5: Compare the two methods with EDA = 0.0125%, number of data in queue = 1

approximation equation is the real sensing data, which is more concerned with the other 2 data than the Linear Approximation Equation Method's optimal data. Therefore, the possibility of the EDA of those 3 data using linear approximation method is fairly higher than the method without using the approximation equation. This could make the Linear Approximation Equation Method send data continuously that increases the number of transmission packet a lot. To prevent this problem from occurring, the number of data in queue should be set bigger than 1.

In **the second experiment**, the number of data in queue in a sensor node was set to 15. The limited EDA was kept as 0.0125%. the Linear Approximation Equation Method showed a better result than the method without using the approximation equation when the number of half-periods in set of 1000 data is bigger than 9. When that value was 8 or 9, the results of those two methods were fairly the same. (Fig 6.6)

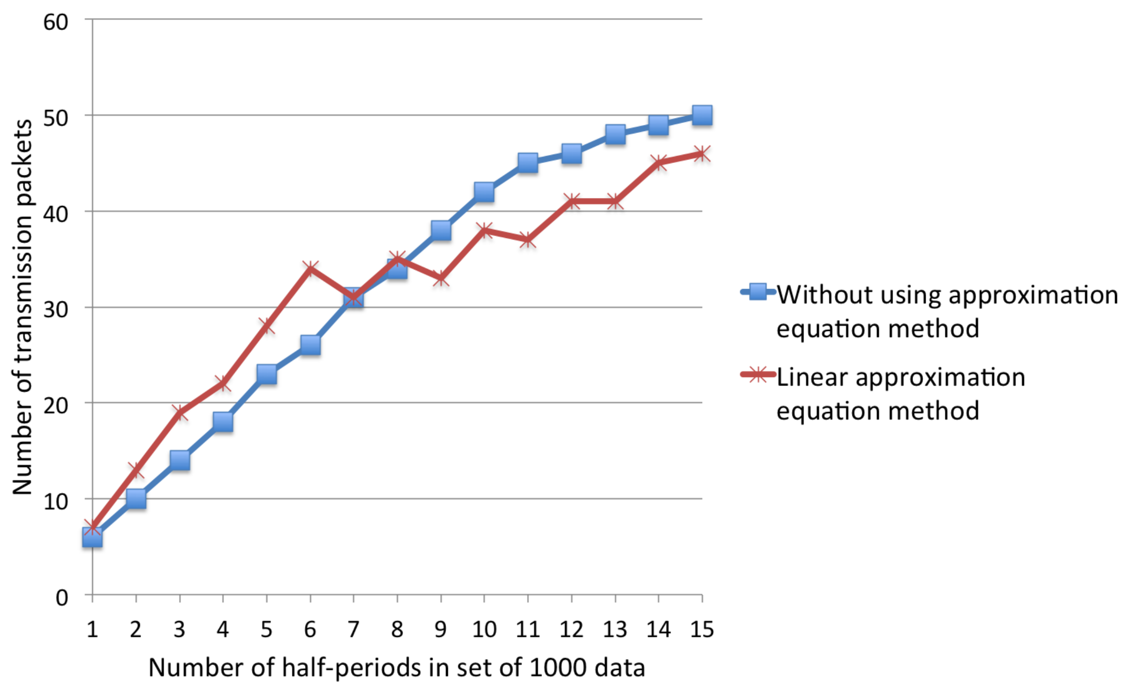


Figure 6.6: Compare the two methods with EDA = 0.0125%, number of data in queue = 15

The number of data in queue is increased to 20 and the limited EDA was kept as 0.0125% in **the third experiment**. In this experiment, the Linear Approximation Equation Method showed an even better result than the previous two experiment even though the result of the method without using the approximation equation did not change. (Fig. 6.7) From the case that the number of half-periods in set of 1000 data is 7, the Linear Approximation Equation Method showed a better result than the the method without using the approximation equation did.

It can be assumed from the results of the three experiments that the efficiency of the number of data in queue.

The fourth experiment evaluates the efficiency of the number of data in queue in both the Linear Approximation Equation Method and the method with-

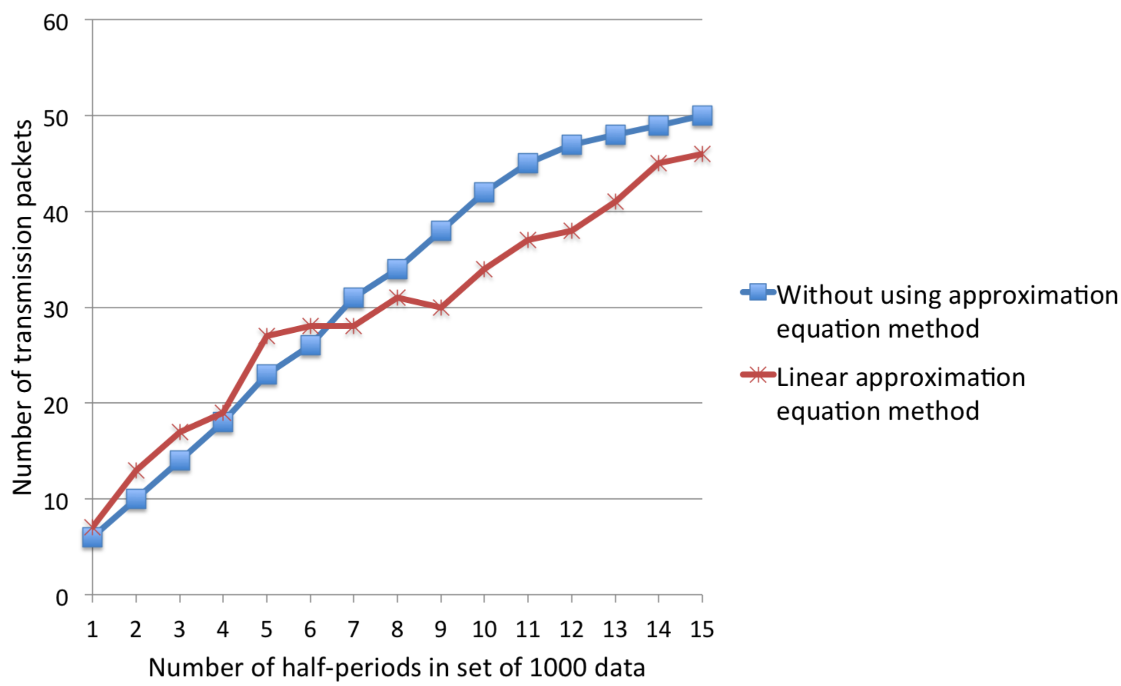


Figure 6.7: Compare the two methods with EDA = 0.0125%, number of data in queue = 20

out using the approximation equation. The limited EDA in this experiment was kept as 0.0125%. The number of data in queue was set to 1, 5, 10, 15, 20.

In this experiment, the results of the method without using the approximation equation did not change a lot when the number of data in queue was changing (Fig. 6.8). Therefore, in the method without using the approximation equation, there is no need to use queue in the sensor node. It will also save the memory and won't take any delay on sending data, the data can be sent immediately.

On the other hand, when using the Linear Approximation Equation Method, the more number of data stored in queue, the better result we had (Fig. 6.9). In this experiment, with the number of data in queue set to 20, the number of transmission packets was the least of all cases (number of half-periods in set of 1000 data in from 1 to 15). On the contrary, the number of data in queue set to

1 required the most number of transmission packets in all cases. Figure 6.9 also shows that when the number of data in queue does not show efficiency on the data with number of half-periods in set of 1000 data is 1 or 2. In that case, the number of transmission packets are the same when the number of data in queue is set to the different number. Let's take a deep look at the case of data that with them the Linear Approximation Equation Method shows the better result than the method without using the approximation equation in figure 6.7. At those cases, number of transmission packets when the data in queue is set to 20 was much smaller than the data in queue was set to 5 or 10. In the best result case (number of half-periods in set of 1000 data is 9), by setting the number of data in queue from 5 to 20, the number of transmission packet reduced from 43 to 30. It means the number of transmission packet reduced by about 30%.

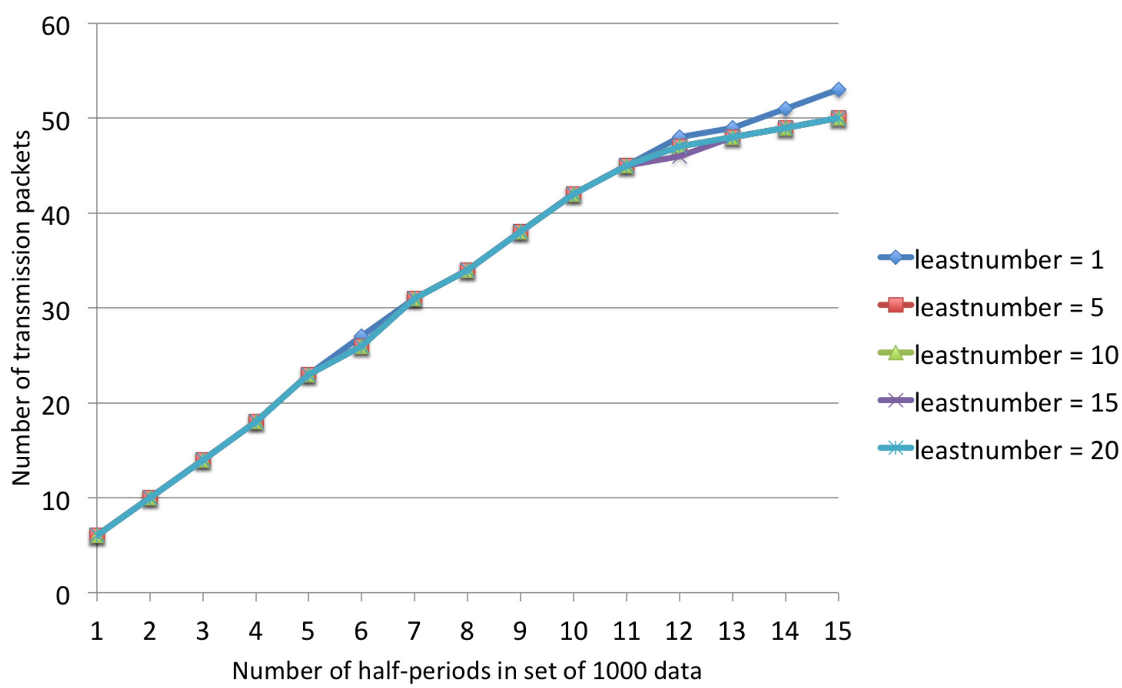


Figure 6.8: The efficiency on the number of data in queue in the method without using the approximation equation, EDA = 0.0125%

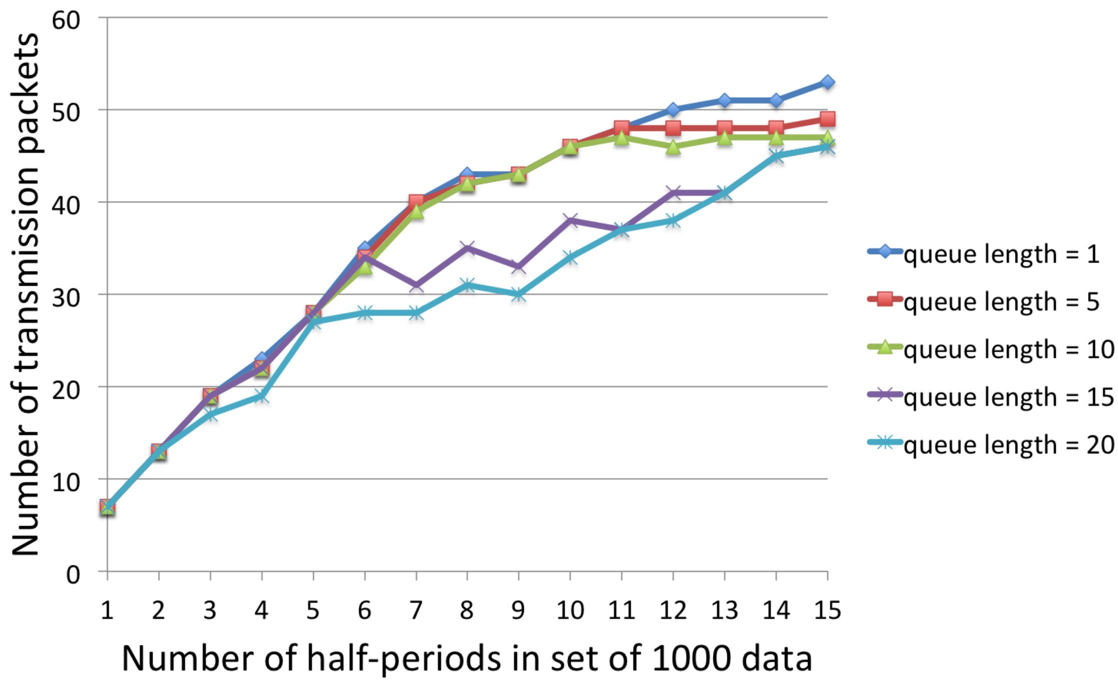


Figure 6.9: The efficiency on the number of data in queue in the Linear Approximation Equation Method, EDA = 0.0125%

6.5 Summary

In this chapter, the evaluation on the efficiency of the linear approximation method by comparing with the method without using the approximation equation and the evaluation on the efficiency of the number of data in queue in sensor node have been demonstrated. The results of those experiments show the conclusion that will be addressed.

Firstly, with using the Adaptive Time-wise Data Interpolation, in this experiment, with the set limited error rate at 0.0125%, instead of sending all the 1000 packets, the transmission packet number is only $7 \sim 53$ packets (the numbers are different with the different data set). It means that the data transmission packets number is reduced by 94∼99%.

Secondly, when the number of queue was set to 1, it means the queue isn't being used, the method without using the approximation equation showed the better result than the Linear Approximation Equation Method. Therefore, in the WSN environment that required real time data, the data must be sent immediately after sensing, or in the sensor node that can not provide a good memory enough for storing and calculating data, the method without using the approximation equation should be used for data reporting in sensor node and data interpolation to avoid the traffic congestion.

Thirdly, with the use of queue in a sensor node to store data before calculation, the Linear Approximation Equation Method showed the better result in traffic avoidance than the method without using the approximation equation. Where small sensor data fluctuation is observed, the method without using the approximation equation showed the better result. However, in practical system, almost the sensing data are changing a lot and is not stable. Therefore, the linear approximation equation with the set of data in queue should be applied for providing the better in traffic congestion avoidance. The more data storing in queue, the better result we have. Based on the requirement of the practical WSN, that number should be adequately set. In my experiment, by keeping the error distribution average at 0.0125% and setting the number of data in queue to 20 instead of 1, the number of transmission packets reduces about 30%. The queue in sensor node does not provide any efficiency for the method without using the approximation equation, so there is not any need of using queue in the method without using the approximation equation.

Chapter 7

Conclusion

This thesis proposes an optimal data interpolation method, referred to as the Approximation Equation Method, and an adaptive method to adjust data reporting interval and the data queue length in accordance with permissible error criteria.

In Wireless Sensor Network, it is usual practice to truncate sensor data reporting to avoid traffic congestion. This thesis reveals such practice results in data interpolation error in receiving applications. The Approximation Equation Method optimally adjusts the reporting sensor data such that the interpolation in receiving applications provide the minimum error over the truncated samples duration. The method is implemented in Texas Instruments CC2430 ZigBee Sensor Network which continuously monitors external temperature of Z building of Keio University Shonan Fujisawa Campus. A thorough mathematical derivation enables the method can be implemented in low computational power MCUs. It is shown that the Approximation Equation Method can achieve 1.7 times longer data reporting interval than that of simple data truncation method.

Traffic in a Wireless Sensor Network can further be mitigated if sensors adaptively changes its reporting time interval while the accuracy in the receiving application is preserved. This observation leads the author to develop the an adaptive control of reporting time interval under predefined error threshold with linear interpolation in sensor node. The method is again implemented to CC2430 and evaluated its performance against emulated periodical sensor data.Using this method, with the set limited error rate at 0.0125%, the data transmission packets number is reduced by 94$\sim$99%. However, it is observed that the interpolation in sensor node does not always results in less traffic. For a better result, instead of calculating immediately after having new sensing data, a queue set in sensor node to store data is effective. The penalty of this method is that the data is not calculated immediately. It entails delay between the time of having sensing data and sending data to sensor sink in sensor node. The evaluation reveals that the number of transmission packet can be reduced by 30% with the proposed queued interpolation method in comparison with the experiment, in which sensor node do not use queue to store data.

Acknowledgement

It is a pleasure to thank the many people who made this thesis possible.

First and foremost, I would like to express my deepest gratitude to my advisor Associate Professor Jin Mitsugi for the continuous support of my study and research, for his patience, his enthusiasm, his inspiration, and his great efforts to explain things clearly and simply to me. Without his help, I would not even think of writing this thesis.

I would like acknowledge and extend my sincere gratitude to Professor Hideyuki Tokuda, Professor Jun Murai, Associate Professor Hiroyuki Kusumoto, Professor Osamu Nakamura, Associate Professor Kazunori Takashio, Associate Professor Rodney D. Van Meter III, Associate Professor Keisuke Uehara, Lecturer Jin Nakazawa for the valuable and technical comments on my thesis.

I would like to thank Hanoi University of Science and Technology and HEDSPI Project, for giving me a chance to study in Keio University.

I would like to thank Professor Hagino and JICE for the metal support thorough all the time I am studying in Keio University.

I would like to show my gratitude to Assistant Professor Hada, Associate Professor Inaba, Mr. Nakane, Mr. Shigeya, Mr. Matsumoto, Mr. Shiraishi, who have given me numerous supports from the beginning when I have just joined the Auto-ID Labs Japan 2 years ago. They shared a lot of their experience with me and aided me in times of need.

I would like to thank Mr. Yonemura, who has helped me a lot with my research and my Japanese.

I would like to thank Mr. Yoshida, Mr. Thanh, Ms. Meryl for their assistance with my thesis writing.

Much thank for all my labmates in Auto-ID Labs Japan, Ms. Shiori, Mr. Emura, Mr. Yamaguchi, Mr. Sugimoto, Ms. Kim, Mr. Yamada, Mr. Hiroishi, Ms. Chisato, Mr. Miyazaki, Mr. Nojima, Mr. Sato, Mr. Yokoishi, Mr. Igarashi, Mr. Miki, Ms. Mayu, Mr. Osono, Mr. Uchiyama, for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in my two year in lab.

Thank all my friends from Vietnamese student group in Keio for their sharing and aiding me.

Lastly, and most importantly, I wish to thank my family. I cannot express my gratitude for all the things you have given me. Thank you!

Bibliography

[1] John A. Stankovic. Wireless sensor networks. *Computer*, 41:92–95, 2008.

[2] ドアン・ホアイ・ナム、米村茂、三次仁、中村修、村井純. データ補間を考慮したセンサデータ送信間隔の最適化. **電子情報通信学会ソサイエティ大会**, 講演論文集 2(B-19-12):430, 2011.

[3] RACOW(RFID Auto-Commissioning Open system with WiMAX). `http://www.racow.net/27bjup`. [Online; accessed 16-Jan-2012].

[4] Drew Gislason. *ZigBee Wireless Networking*, pages 274–277. Newnes, 2008.

[5] IEEE: Advancing Technology for Humanity. `http://www.ieee.org/index.html`. [Online; accessed 16-Jan-2012].

[6] Ata Elahi with Adam Gschwender. *ZigBee wireless sensor and control network*, page 31. Upper Saddle River, N.J. : Prentice Hall, 2010.

[7] Saul A. Teukolsky William T. Vetterling William H. Press, Brian P. Flannery. *Numerical Recipes in C - The Art of Scientific Computing*, page 85. Cambridge University Press, 1988.

[8] D. Estrin J. Zhao, R. Govindan. Residual energy scan for monitoring sensor netoworks. *Proc. IEEE WCNC*, 1:356–362, 2002.

[9] Mo Li Yunhao Liu, Kebin Liu. Passive diagnosis for wireless sensor networks. *IEEE/ACM Transactions on networking*, 18(4):1132–1144, August 2010.

[10] I. Akyildiz Y. Sankarasubramaniam, O. Akan. Event-to-sink reliable transport in wireless sensor networks. *Proc. of the* 4^{th} *ACM Symposium on Mobile Ad Hoc Networking & Computing*, (MobiHoc 2003):177–188, 2003.

[11] L. Krishnamurthy C. Y. Wan, A. T. Campbell. Psfq: A reliable transport protocol for wireless sensor networks. *Proc. of First ACM International Workshop on Wireless Sensor Networks and Applications*, (WSNA 2002):49–58, 2002.

[12] Andrew T. Campbell Cheh-Yin Wan, Shane B. Eisenman. Coda: Congestion detection and avoidance in sensor networks. *Sensys 03*, pages 266–278, 2003.

[13] Jiankun Hu Yaakob N., Khalil I. Performance analysis of optimal packet size for congestion control in wireless sensor networks. 9^{th} *IEEE International Symposium on Network Computing and Applications*, pages 210–213, 2010.

[14] Monson H. Hayes III Jeffery R. Price. Sensor optimal image interpolation. *Signals*, Systems and Computers:1262–1266, 1999.

[15] Doan Hoai Nam. Store and forward power consumption monitoring system using ZigBee. `http://web.sfc.wide.ad.jp/~perry/term.pdf`, 2011. [Online; accessed 16-Jan-2012].

[16] Texas Instruments. *CC2430 Data Sheet (rev. 2.1)*, 2010.

[17] 大石進一. **フーリエ解析**, pages 24–25. 岩波書店, 2010.