Dynamic Framed Slotted ALOHA Algorithm for Fast Object Identification in an RFID System

Supervisor : Professor Jae-Hyun Kim

by

Jae-Ryong Cha

School of Electrical and Computer Engineering

at the

AJOU UNIVERSITY

January, 2006

Dynamic Framed Slotted ALOHA Algorithm for Fast Object Identification in an RFID System

By

Jae-Ryong Cha

Submitted to the School of Electrical and Computer Engineering

in partial fulfillment of the requirements for the degree of

Master of Electrical Engineering

at the

AJOU UNIVERSITY

Approved by

Professor Jae-Hyun Kim, Ph.D. (Supervisor)	**Date**
Professor Sung-Keun Oh, Ph.D. (Committee)	**Date**
Professor Chae-Woo Lee, Ph.D. (Committee)	**Date**

DEDICATION

To all with whom I have gotten along and

especially to my family.

Acknowledgement

I would like to express my deepest appreciation to my supervisor, Professor Jae-Hyun Kim, for his rigorous academic advice and constant encouragement for my research. He has been willing to support my plans and ideas with proper knowledge and guidance. I would also like to thank Professor Sung-Keun Oh and Professor Chae-Woo Lee for being on my advisory committee and for his through and broad review of this thesis. My thanks are also extended to my graduate student colleagues and all WINNERs who have been not only reliable friends but also brilliant co-workers during my research, especially Seung-Min Oh, Hyun-Jin Lee, Ho-Seung Choi, Ju-A Lee, Seong-Jin Lee, Ji-Su Kim, Choong-Hee Lee, and Su-Ryun Lee. Finally, special thanks go to my parents for their support, encouragement, and endless love over the years so that I could concentrate on my study. They have done whatever they could help me throughout the course of my graduate study. Without their encouragement, support and confidence, I could have never finished my study.

Abstract

A Radio Frequency IDentification (RFID) system is a simple form of ubiquitous sensor networks that are used to identify physical objects. It permits remote, non-line-of-sight, and automatic reading. In an RFID system, when two or more tags respond to the reader's command, the collision occurs and results in interfering the reader from identifying the tags correctly. The reader must avoid this collision by using the anti-collision algorithm.

In general, there are two types of anti-collision algorithms; binary-type and ALOHA-type. This thesis will focus on the ALOHA-type anti-collision algorithm, where the Framed Slotted ALOHA (FS-ALOHA) algorithm is generally used. In the conventional FS-ALOHA algorithms, when the number of tags is more than the number of slots, the delay to identify a set of tags increases. On the other hand, in a situation that the number of tags is lower than the number of slots, the wasted slots can occur. Therefore, it needs to appropriately vary the frame size according to the number of tags.

In this thesis, we propose Dynamic Framed Slotted ALOHA (DFSA) algorithm using both Tag Estimation (TE) and Dynamic Slot Allocation (DSA) to improve the performance of the conventional FS-ALOHA algorithms. We also compare the performance of the

proposed DFSA algorithm with that of the conventional algorithms using OPNET (optimized network engineering simulator) simulation. According to the simulation, the performance of FS-ALOHA algorithms shows a rapid decrease as the number of tags increases because of the fixed frame size. So we have to use the FS-ALOHA algorithms in the specific and restricted applications. However the algorithms using both TE and DSA show the stable performance. Although the performance of the algorithms show the similar performance, the proposed DFSA algorithm is better because it enables faster tag identification and the complexity is lower so that it is easier to be implemented in an RFID system. When considering the parameters defined in ISO 18000-6 Type A, the proposed DFSA algorithm identifies approximately 777 tags per second.

Contents

List of figures

List of tables

Abbreviations

AM : Amplitude modulation

DFSA : Dynamic Framed Slotted ALOHA

DSA : Dynamic slot allocation

EPC : Electronic product code

FS-ALOHA : Framed Slotted ALOHA

FSK : Frequency shift keying

ISM : Industrial, scientific, and medical

ISO : International Organization for Standardization

LSB : Least significant bit

MSB : Most significant bit

OPNET : Optimized network engineering simulator

PSK : Phase shift keying

RF : Radio frequency

RFID : Radio frequency identification

TE : Tag estimation

UHF : Ultra high frequency

Chapter 1. Introduction

Reliable identification of multiple objects is especially challenging if many objects are present at the same time. Several technologies are available, but they all have limitations. For example, bar code is the most pervasive technology used today, but reading them requires manual, close-ranging scanning, and a line of sight between the reader device and the tag. But a Radio Frequency IDentification (RFID) system which is a simple form of ubiquitous sensor networks that are used to identify physical objects. Instead of sensing environmental conditions, an RFID system identifies the unique tags' ID or detailed information saved in them attached to objects.

A passive RFID system generally consists of a reader and many tags. A reader interrogates tags for their ID or detailed information through an RF communication link, and contains internal storage, processing power, and so on. Tags get processing power through RF communication link from the reader and use this energy to power any on-tag computations. A reader in an RFID system broadcasts a request message to the tags. Upon receiving the message, all tags send the reply back to the reader. But if there is more than one tag reply, their replies collide on the RF communication channel, and thus cannot be received by the reader. This problem is referred to as the "Tag-collision problem." An effective system must avoid this collision by using an anti-collision algorithm because the ability to identify many tags simultaneously is crucial for many applications.

In this thesis, to improve the performance of the conventional FS-ALOHA algorithms, we propose the Dynamic Framed Slotted ALOHA (DFSA) algorithm using Tag Estimation (TE) which estimates the number of tags around the reader and Dynamic Slot Allocation (DSA) which adaptively allocates the frame size according to the number of tags. The rest of this thesis is organized as follows. Chapter 2 explains what an RFID system is. In Chapter 3, after describing the general concept of the anti-collision algorithm, we introduce the anti-collision algorithms defined in UHF (ultra high frequency) band RFID Standards. In Chapter 4, we briefly introduce two algorithms proposed by Vogt. Chapter 5 numerically discusses the proposed Tag Estimation Method and Dynamic Slot Allocation Method, and then explains the basic operation of the proposed DFSA algorithm. In Chapter 6, we quantitatively compare the simulation results of the proposed DFSA algorithm with those of the conventional algorithms before concluding this thesis in Chapter 7.

Chapter 2. An RFID system

2.1. History of an RFID system

Work on RFID systems as we know them began in earnest in the 1970s. In 1972, Kriofsky and Kaplan filed a patent application for an "inductively coupled transmitter-responder arrangement." This system used separate coils for receiving power and transmitting the return signal. In 1979, Beigel filed a new application for an "identification device" that combined the two antennas; many consider his application by to be the landmark RFID application because it emphasized the potentially small size of RFID devices. In the 1970s, a group of scientists at the Lawrence Livermore Laboratory (LLL) realized that a handheld receiver stimulated by RF power could send back a coded radio signal. Such a system could be connected to a simple computer and used to control access to a secure facility. They developed this system for controlling access to sensitive materials at nuclear weapons sites. Today we would call this Livermore system an example of security through obscurity: What made the system secure was that nobody else had a radio capable of receiving the stimulating radio signal and sending back the properly coded reply. But at the time it was one of the most secure access control systems available. The scientists left LLL a few years later and created their own company to commercialize the technology. This system ultimately became one of the first building entry systems based on proximity technology and the first commercial use of RFID (Radio Frequency IDentification)[1].

2.2. Basics of an RFID system

RFID systems are a form of sensor networks that are used to identify physical objects. Instead of sensing environmental conditions such as temperature and humidity, an RFID system identifies the unique identifier (ID or EPC(electrical product code)) and other information stored in Radio Frequency (RF) tags affixed to objects. RFID readers identify tags by actively transmitting a signal to communicate with the RF tags. RFID systems fundamentally consist of two elements: the RFID tags themselves and the RFID readers[1]-[3].

Figure 1 shows the basic circuit diagram for a passive RFID system. Different from an active RFID system, tags of a passive RFID system do not have the battery for on-tag computation or data transfer. The reader generates an electromagnetic field with its antenna. A tag extracts its energy from the electromagnetic field with its antenna by charging a capacitor until the chip (IC) is able to operate. After having transferred the energy to the tags, the reader and the tag may now exchange information, e.g. reader's command, tag's data (EPC), by modulation methods like amplitude modulation (AM), phase shift keying (PSK) or frequency shift keying (FSK) of the electromagnetic field[4].

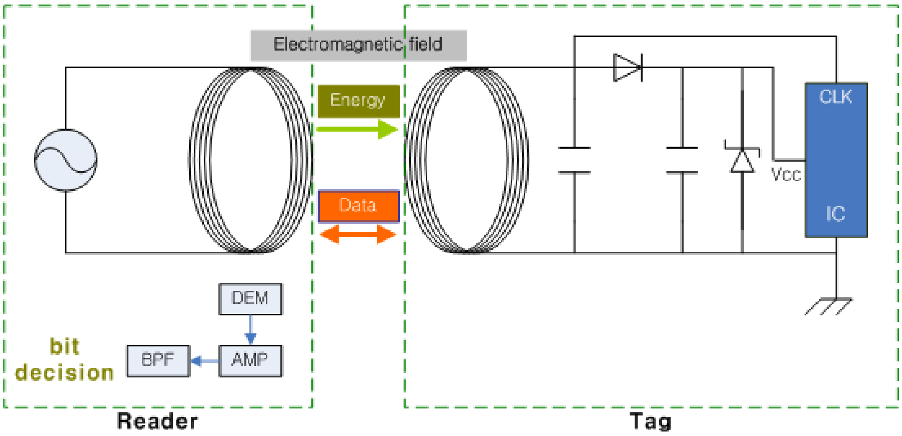

Figure 1. Basic circuit diagram for a passive RFID system

Where, DEM, AMP, and BPF mean demodulation, amplifier, and band pass filter respectively.

2.2.1. An RFID reader

An RFID reader sends a pulse of radio energy to the tag and listens for the tag's reply. The tag detects this energy and sends back a reply that contains the tag's serial number and possibly other information as well. In simple RFID systems, the reader's pulse of energy functioned as an on-off switch; in more sophisticated systems, the reader's RF signal can contain commands to the tag, instructions to read or write memory that the tag contains, and even passwords.

2.2.2. RFID tags

The tag is the basic building block of an RFID system. Each tag consists of an antenna and a small silicon chip that contains a radio receiver, a radio modulator for sending a reply back to the reader, control logic, some amount of memory, and a power system. The power system can be completely powered by the incoming RF signal, in which case the tag is known as a passive tag. Alternatively, the tag's power system can have a battery, in which case the tag is known as an active tag.

2.3. Applications of an RFID system

Radio frequency (RF) based passive identification devices are an emerging technology which influences a lot of different application domains. It lets the end user identify an item so that it can be quickly and accurately re-identified electronically, as it moves through the distribution/utilization process. RFID chips may not only be used as simple identification devices, but also more complex personal data recording devices or in sophisticated smart card applications. The implantation of small RFID chips in animals for example helps to observe their migration or disease propagation. RFID chips in clothes for example helps in large laundries to organize the individual washing procedure and to identify their owners. Access control by RFID chips in buildings instead of ordinary keys are much more flexible and thus quite common, access control at ski-lifts are much faster. RFID chips are also used to identify and to keep track of stock or production units. Recent document management systems can not only handle digital data, but are also able to identify physical archived files, which are equipped with RFID chips. The wear out of tools and machine units can be kept track with RFID chips. RFID technology may replace bar codes on food stores, record over-temperature of sensitive food, or for applications like automatic billing and refilling in mini-bars of hotel rooms. More sophisticated applications use so called smart cards with integrated encryption processor power. Applications in the domain of electronic cash are quite common. Intelligent tickets mixing access for concerts, sport events, fitness room every morning, ski-lift every afternoon may improve customer services[4].

Chapter 3. Anti-collision algorithms

3.1. Reader and tag collision problem

The use of the shared RF medium for communication with tags creates the problem of either readers or tags potentially interfering with one another's operation. Interference may be due to either readers or tags.

Reader interference occurs when two or more readers try to access the same tag at the same time. The low functionality of the tags prevents them from being able to distinguish between the communication signals from different readers. Therefore, when two or more readers may potentially communicate with the same tag, the readers must communicate at different times to ensure proper communication with the tag by each of the readers. Reader caused interference is referred to as a reader collision, and the problem of minimizing reader collisions is referred to as "Reader Collision Problem."

Tag interference occurs when two or more tags respond to the reader's ID request. That is, if the reply of tags on the reader's ID request is more than one, their replies collide due to the interference on the RF communication channel, and thus cannot be received by the reader. This is generally regarded as "Tag Collision Problem." An effective system must avoid these problems by having tags or readers respond on either different frequencies or different times. Generally, the method to resolve these problems is referred to as the "anti-collision algorithm", which is one of the core technologies in an RFID system[2],[5].

3.2. Anti-collision algorithms

The anti-collision algorithms are generally classified into two types: binary-type and ALOHA-type. Binary-type anti-collision algorithms consist of binary search algorithm, bit-by-bit binary tree algorithm, binary tree algorithm using bin slots, probabilistic binary tree algorithm[6]-[8]. Meanwhile, ALOHA-type anti-collision algorithms are composed of Framed slotted ALOHA algorithms using the fixed frame size which are used for optimizing the relatively low throughput of the ALOHA-type anti-collision algorithm and the probabilistic slotted algorithm[9],[10]. In next section, two types of algorithms mentioned above will be explained in details.

3.2.1. Binary-type anti-collision algorithms

3.2.1.1. Bit-by-bit binary tree algorithm

Bit-by-bit binary tree algorithm defined in AutoID Class 0 performs tag identification process on a bit-by-bit[11]. Each tag reply is defined by two sub-carrier frequencies, one for a binary 0, and the other for a binary 1. Because 0s and 1s are communicated as distinct tones, the reader can simultaneously receive both. After each collision-less tag-to-reader bit communication, the reader, by choosing one of the two possible binary tree branches, directs tags to either remain active, or go temporarily inactive. In particular, tags that receive a bit that matches the last bit backscattered remain active; those that do not see such a match will go temporarily inactive and wait to participate in the next identification process. The identification process continues for all bits of their ID, and results in a tag identification. Once the tag has been identified, the reader may send commands to this tag and/or put the tag to sleep (inactive state). This method is applied repeatedly for each tag in the population.

3.2.1.2. Binary tree algorithm using bin slot

Binary tree algorithm using bin slot for EPC Class 1 resolves the collision by using PingID command and bin slots which are used to receive tags' reply[12]. The reply period for PingID command consists of 8 bin slots. Each slot sequentially represents from '000' to '111' as shown in Figure 2. The procedure of the algorithm using PingID command is as follows. The reader transmits PingID command to the tags. The tags matching [VALUE] beginning at location [PTR] reply by sending 8 bits of their ID beginning with the bit at location [PTR] + [LEN], where [VALUE] is the data that the tag will attempt to match against its own ID (from the [PTR] position towards the LSB(lLeast significant bit)), [PTR] is a pointer to a location (or bit index) in their ID, and [LEN] is the length of the data being sent in the [VALUE] field. The 8-bit reply is transferred in one of eight bin slots delineated by the reader. The bin slot is selected to be equal to the value of the first 3 bits from MSB (most significant bit) of the 8-bit reply. So, the tags whose 3 bits from MSB of their ID after [VALUE] field are '000' select bin 0 and whose 3 bits from MSB of their ID after [VALUE] field is '111' select bin 7. When two or more tags select the same bin slot, the reader retransmits PingID command to the tags. If a bin slot is occupied by only one tag, the reader sends ScrollID command to the tag. The tags matching [VALUE] beginning at location [PTR] reply by sending their whole ID.

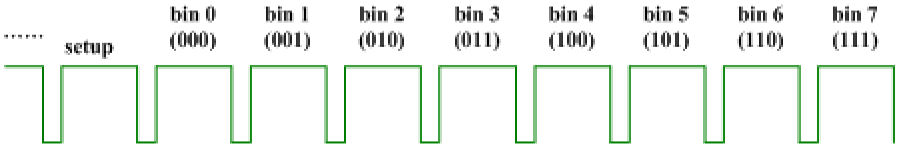

Figure 2. PingID Reply Response

3.2.1.3. Probabilistic binary tree algorithm

Probabilistic binary tree algorithm defined in ISO 18000-6 Type B uses a slot counter and a random generator of tags[13],[14]. After reader's ID request, when the collision occurs, in slot k, all tags that are not involved in the collision wait until the collision is resolved. The tags involved in the collision split randomly into two groups, by (for instance) each selecting 0 or 1. The tags in the first group, those that selected 0, retransmit in slot $k+1$ while those that selected 1 wait until all tags that selected 0 successfully transmit their ID. If slot $k+1$ is either idle or successful, the tags of the second group, those that selected 1, retransmit in slot $k+2$. If slot $k+1$ contains another collision, the same procedure is repeated.

3.2.2. ALOHA-type anti-collision algorithms

3.2.2.1. Framed Slotted ALOHA algorithm

The Framed Slotted ALOHA (FS-ALOHA) algorithm adopted in ISO 18000-6 Type A uses a mechanism that allocates tag transmissions into rounds and slots (time frame)[13]. Each slot has duration, long enough for the reader to receive a tag reply. This time frame is divided into a number of slots that can be occupied by tags and used for sending their replies. The reader determines the actual duration of a slot. After the reader has sent its request to the tags, it waits for a certain amount of time for their answers. Tags which successfully transmitted their ID are identified by a reader and go to inactive state in which they do not respond a next reader's ID request temporarily. Meanwhile, when multiple tags use the same slot, a collision occurs and data get lost. Tags which experienced the collision transfer their ID by choosing new random number in next round.

3.2.2.2. Probabilistic Slotted ALOHA algorithm

A reader implementing probabilistic Slotted ALOHA algorithm can select tags in the field before inventorying them by issuing a new command, called *Select* [15],[16]. After selecting a specific group of tags, the reader sends *Query* command including value Q which is a seed value to determine the value of the slot count. Upon receiving a *Query* command, participating tags pick a random value in the range $(0, 2^Q - 1)$, and load this value into their slot counter. Tags that pick a zero transition to the *reply* state and reply immediately. Tags that pick a nonzero value shall await another reader's command, e.g. either *QueryAdjust* or *QueryRep* command. Assuming that a single tag replies, the algorithm proceeds as follow:

(a) The tag backscatters an RN16 (16-bit random number) as it enters *reply* state,

(b) The reader acknowledges the tag with an *ACK* command containing this same RN16,

(c) The acknowledged tag transitions to the *Acknowledged* state, backscattering its PC(protocol control bits), EPC, and CRC-16.

3.3. Anti-collision algorithms in Standards

Moving into the UHF frequencies to exploit range benefits through use of unlicensed industrial, scientific, and medical (ISM) bands have led to the creation of several first-generation protocol standards. One of these traces is EPCglobal, an organization that recognized the potential of RFID early. Other standards originated with the International Organization for Standardization (ISO) as part of the ISO 18000 family, with 6 groups of documents dedicated to UHF operation.

Table 1. Major attributes of the significant UHF Standards

	Arbitration	Air interface (Rev. / For.)	EPC (bits)	Security
AutoID Class 0	Deterministic Binary Tree	Pulse Width Mod. / FSK	64, 96	24-bit
AutoID Class 1	Deterministic Slotted	Pulse Width Mod./ Pulse Interval AM	64, 96	8-bit
ISO 18000-6 Type A	Probabilistic Slotted	Pulse interval ASK / FM0	n.a.	n.a.
ISO 18000-6 Type B	Probabilistic Binary tree	Manchester-ASK / FM0	n.a.	n.a.
ISO 18000-6 Type C	Probabilistic Slotted	Pulse interval ASK / Miller, FM0	96, 256	32-bit , Access

Where, **Rev.** and **For.** mean "reverse direction" and "forward direction" respectively.

Table 1 compares the major attributes of the significant UHF Standards[17]. The first-generation EPC had two protocols, Class 0 and Class 1, and the same reader could not read both unless it was a multi-protocol reader. ISO also approved two UHF air-interface protocols, 18000-6 Type A and 18000-6 Type B, as international standards. So there have been four UHF standards. Meanwhile, in 2004, EPCglobal has submitted the Gen 2 protocol to ISO for approval. As of January 2005, EPCglobal Class 1 Generation 2 air interface protocol was adopted to Standards, ISO 18000-6 Type C in UHF band, which is expected that it will be one UHF protocol globally in a passive RFID system.

Chapter 4. Related works

This thesis focuses on the performance evaluation of ALOHA-based anti-collision algorithms. For the objective performance evaluation of the proposed algorithms, we would like to consider many other algorithms. But actually there are few papers with regard to ALOHA-based algorithms. We found a good reference paper written by Vogt in 2002. So, in this section, we explain the two algorithms proposed by Vogt briefly. The first proposed by Vogt is the algorithm using the minimum bound of collided slots. The basic concept is that a collided slot includes more than at least two tags. So the number of estimated tags can be calculated by just multiplying the number of collided slots by two[18].

$$Vogt_{\min} = \text{the number of collided slots} \times 2. \qquad (1)$$

The second algorithm employs *Chebyshev's inequality*. That is, it uses the distance between the slot status (successful, collided, and idle information) and the expected value vector to determine the number of tags for which the distance becomes minimal. Accordingly, after a round, the reader estimates the number of tags by determining the value n when the distance of between read results (c_0: number of idle slots, c_1: number of successful slots, and c_k: number of collided slots) and expected value vector A ($a_0^{N,n}$, $a_1^{N,n}$, and $a_{\geq 2}^{N,n}$) is minimum as shown in equation (2).

$$Vogt_{\text{est}} = \min(n)\left[\begin{pmatrix} a_0^{N,n} \\ a_1^{N,n} \\ a_{\geq 2}^{N,n} \end{pmatrix} - \begin{pmatrix} c_0 \\ c_1 \\ c_k \end{pmatrix}\right]. \quad (2)$$

Where N and n mean the frame size and the number of tags respectively. For more details about the algorithms, refer to [18].

In the next section, we find the problems of the conventional FS-ALOHA algorithms using the fixed frame size before explaining the proposed algorithm which uses both how to estimate the number of tags around the reader and how to adaptively allocate the frame size according to the number of estimated tags.

Chapter 5. Proposed DFSA algorithm

5.1. Proposed DFSA algorithm

In FS-ALOHA algorithm, when the number of tags is more than the number of slots, the delay to identify a set of tags increases. On the other hand, in a situation that the number of tags is lower than the number of slots, the wasted slots can occur.

During the tag identification process, a reader does not know how many tags there are. So a reader needs to estimate the number of tags around the reader. This method is regarded to as Tag Estimation (TE)[9]. Vogt proposed two TEs. But, because he did not consider the inactive state in which tags do not temporarily respond reader's next request in his simulation results, there is the limitation to apply for those results to evaluate the performance of an RFID system[10].

A reader also needs to allocate the optimal frame size to enhance the throughput of a system. This kind of method is called as Dynamic Slot Allocation (DSA). DSA is introduced in [13], however there are no detailed methods how to dynamically allocate the frame size. Vogt also proposed DSA, but the result is just by the simulation.

In this thesis, to resolve these problems, we propose a DFSA algorithm using TE which estimates the number of tags around a reader and DSA which allocates the optimal frame size adaptively according to the number of estimated tags. In next section, we explain TE and DSA in details.

5.1.1. Tag Estimation (TE)

In this section, we use the ratio of the number of collided slots to the frame size for estimating the number of tags.

Given L slots in a frame and n tags, the probability that r out of n tags transfers their ID in a slot is given by

$$P(X = r) = \binom{n}{r}\left(\frac{1}{L}\right)^{r}\left(1-\frac{1}{L}\right)^{n-r}. \tag{3}$$

The number r of tags in a particular slot is called the occupancy number of the slots[19]. The expected value of the number of slots with occupancy number r is given by

$$E[X = r] = L\binom{n}{r}\left(\frac{1}{L}\right)^{r}\left(1-\frac{1}{L}\right)^{n-r}. \tag{4}$$

To estimate the number n of tags, we define the collision ratio (C_{ratio}), which means the ratio of the number of slots with collision to the frame size, is given by

$$C_{ratio} = 1 - \left(1 - \frac{1}{L}\right)^n \left(1 + \frac{n}{L-1}\right). \qquad (5)$$

After a round, we know the frame size and the collision ratio. Based on this information, we estimate the number of tags using equation (5).

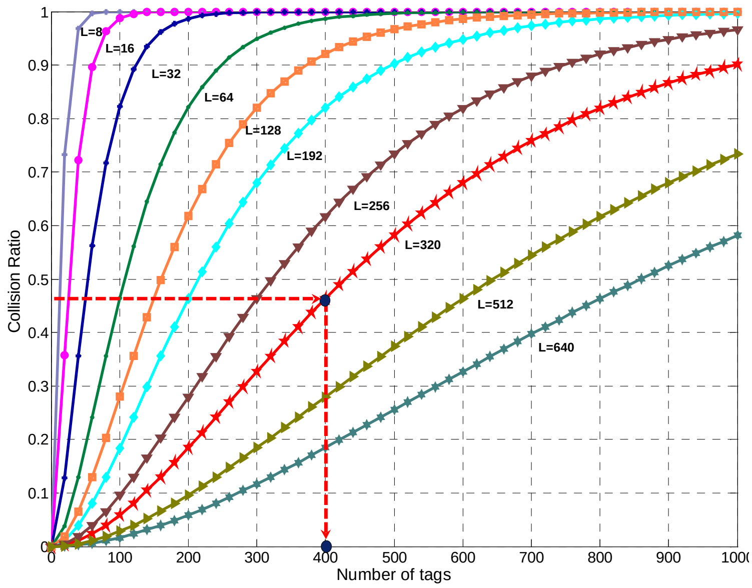

Figure 3. Collision ratio vs. number of tags

Figure 3 shows the collision ratio for the number of tags. Let n_{est} be the number of tags estimated by equation (5). In Figure 3, if the frame size is 320 and the collision ratio is 0.46323 measured by the reader, the number n_{est} of estimated tags is 400.

5.1.2. Dynamic Slot Allocation (DSA)

In this section, we explain two DSAs to determine the optimal frame size. The first method (DSA I) uses the transmission delay of a tag and the second method (DSA II) uses the throughput of a system.

5.1.2.1. Dynamic Slot Allocation I (DSA I)

To obtain the optimal frame size, we consider delay (D), which is the time taken by the tags to transfer their ID successfully.

$$D = \text{number of retransmission} \times \text{frame size}. \quad (6)$$

Because the value of the frame size is known, we need to find the number of retransmission to calculate the delay. The probability p that one tag transmits at the particular slot in a frame is $1/L$. Then the probability that one tag successfully transmits its ID during a slot is given by

$$P_{succ} = p \times (1-p)^{n-1}. \quad (7)$$

And the probability that one tag successfully transmits its ID in a frame (L) is given by

$$P_{succ,L} = p \times (1-p)^{n-1} \times L = (1-p)^{n-1}. \quad (8)$$

Let $P_{succ}(k)$ be the probability that one tag transmits its ID successfully in k^{th} frame. Then $P_{succ}(k)$ is

$$P_{succ}(k) = P_{succ,L}(1 - P_{succ,L})^{k-1}. \tag{9}$$

Using the mean of geometric distribution, the average number of retransmissions for one tag is

$$\begin{aligned} E[X = k] &= \sum_{k=1}^{\infty} kP_{succ}(k) \\ &= \frac{1}{\left(1 - \frac{1}{L}\right)^{n-1}}. \end{aligned} \tag{10}$$

Therefore, we get D from equation (6) and (10).

$$D = \frac{L}{\left(1 - \frac{1}{L}\right)^{n-1}}. \tag{11}$$

It now remains to drive the optimal frame size ($L_{optimal}$). By calculating L when D is minimum, we obtain the optimal frame size.

$$\frac{d}{dn} D = \frac{d}{dn} \frac{L}{\left(1 - \frac{1}{L}\right)^{n-1}} = 0. \tag{12}$$

Then we get

$$L_{optimal} = n. \tag{13}$$

5.1.2.2. Dynamic Slot Allocation II (DSA II)

In this section, we define p same as in the section 5.1.2.1 to drive the optimal frame size. Then the probability that no tag transmits its ID during a slot is

$$P_{idle} = (1-p)^n. \tag{14}$$

The probability that one tag transmits successfully its ID during a slot is given by

$$P_{succ} = np(1-p)^{n-1}. \tag{15}$$

Then, the probability that there is a collision in a slot is

$$\begin{aligned} P_{coll} &= 1 - P_{idle} - P_{succ} \\ &= 1-(1-p)^n - np(1-p)^{n-1}. \end{aligned} \tag{16}$$

We now define throughput S as follows.

$$S = \frac{P_{succ}}{P_{succ} + P_{coll} + P_{idle}} = P_{succ}. \tag{17}$$

The maximum throughput happens when

$$\frac{dP_{succ}}{dp} = n(1-p)^{n-1} - n(n-1)p(1-p)^{n-2} = 0. \tag{18}$$

From equation (18) we get

$$p = \frac{1}{n}. \tag{19}$$

Accordingly, the maximum throughput occurs when $L = n$. We get the optimal frame size($L_{optimal}$) as follows.

$$L_{optimal} = n. \tag{20}$$

Figure 4 depicts the throughput of a system for the frame size. From Figure 4, we can get the optimal frame size by determining the same value with the number of estimated tags.

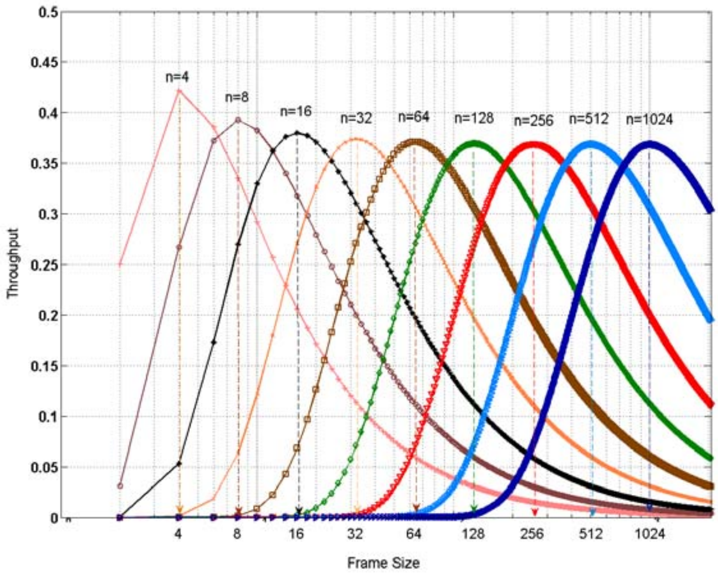

Figure 4. Throughput vs. frame size

From equation (13) and (20), we found that the optimal frame size is the same considering the delay or throughput in a system.

5.2. Basic operation of DFSA algorithm

In this section, we explain the basic operation of the proposed DFSA algorithm. After initialization, a reader sends initial ID request command including initial frame size. By using the replies from the tags, a reader counts the number of collided slots. After a round, a reader estimates the number of tags using equation (5) and the number of collided slots. A reader begins the next round with the new frame size. Table 2 shows briefly pseudo code of DFSA algorithm.

Table 2. Pseudo code of the proposed DFSA algorithm

• Initialization

```
slot_count =1;
coll_count =0;
frame_size = init_frame_size=16;
```

• Sending of initial ID request command

```
send_init_ID_req(init_frame_size)
```

• Calculation of the number of collided slots

```
while(slot_count<=frame_size)
  {
        If (collision occurred)
              coll_count = coll_count+1;

        slot_count = slot_count+1;
        send_ID_req (slot_count);
  }
```

• Estimation of the number of tags

```
estimated_num_tag = tag_est(coll_count);
frame_size = estimated_num_tag;
```

Where,

slot_count : variable indicating tags' ID reply order; if tags' internal slot count is equal to slot_count, send tag's ID, if not, wait.

frame_size : variable(seed value) for tags to select their random number in the range of (1, frame_size).

coll_count : variable for calculating how many slots occur in a frame.

estimated_num_tag : variable for saving the number of estimated tags.

tag_est () : function for estimating the number of tags using the number of collided slots after a round.

send_init_ID_req () : function for sending initial ID request.

send_ID_req () : function for sending ID request after initial ID request.

Chapter 6. Performance evaluation

6.1. Simulation environments

6.1.1. Frame structure for identification process

Figure 5 shows the frame structure for calculating the tag identification time which is also based on ISO 18000-6 Type A. The algorithm proceeds by reader-driven method. We assume that each tag has a 64-bit serial number[16] and errors in wireless communication channel do not occur so that there is no retransmission caused by the errors. It is also assumed that the frame size of next round is equal to the number of tags estimated in the previous round because the results of DSA I and DSA II are same.

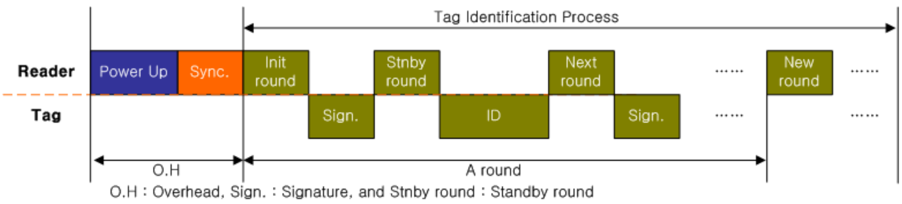

Figure 5. Frame structure for identification process

Table 3 represents the parameters used for calculating the tag identification time. Forward link data rate is 33 kbps and in the case of backward, data rate is 160 kbps so that *O.H* is 480 us , *Init round* is 180 us , *Standby round* is 180 us , *Next round* is 180 us , *New round* is

180 *us* , *Sign.* is 25 *us* , and *ID* is 400 *us* in Figure 5[16].

Table 3. Parameters for tag identification time

O.H	*Init round*	*Standby round*	*Next round*	*New round*	*Sign.*	*ID*
480us	180us	180us	180us	180us	25us	400us

6.1.2. Network model for simulation

Figure 6 shows the network model for the simulation. We use OPNET simulator to evaluate the performance of the proposed algorithm. From the simulation, we obtain the performance metrics such as the number of rounds or the number of total slots until a tag is identified when the number of tags increases from 100 to 1000 by 100.

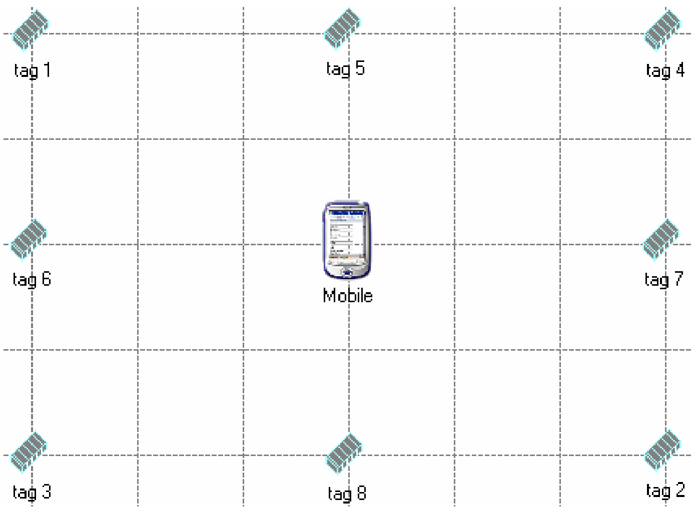

Figure 6. Network model for simulation

6.2. Simulation results

6.2.1. Number of slots per round

Figure 7 and Figure 8 show the number of slots per round when the number of tags is 100 and 500 respectively. Vogt-min, which uses the only collision information to estimate the number of tags, determine the smaller number of tags than others at the initial rounds so that they have the larger round size. But Vogt-est and DFSA use four information; the number of collided, idle, and successful slots including the frame size. Because they may estimate more number of tags at the initial time, more idle slots can occur so that the number of unused slots is increased.

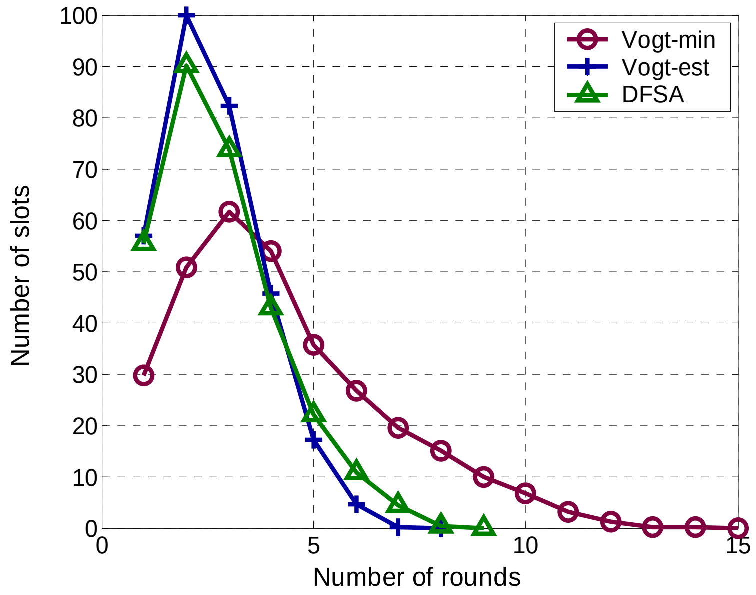

Figure 7. Number of slots per round when 100 tags

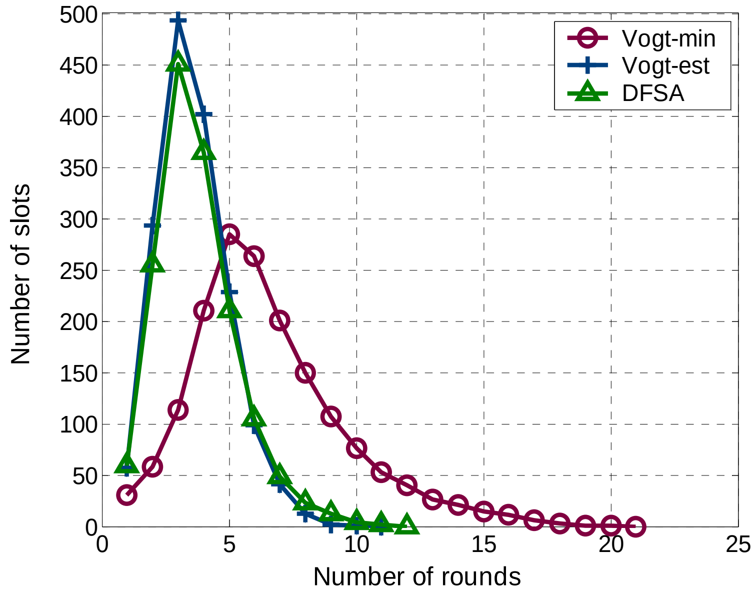

Figure 8. Number of slots per round when 500 tags

6.2.2. Number of slots per tag vs. number of tags

6.2.2.1. Conventional FS-ALOHA algorithms

Figure 9 depicts the number of slots per tag as the number of tags increases. Slot-128 and Slot-256 mean FS-ALOHA algorithms using the fixed frame size with 128 and 256 respectively. Vogt-min represents the algorithm estimating the number of tags proposed by Vogt. In FS-ALOHA algorithms, the number of slots per tag increases sharply after the specific number of tags. But Vogt-min shows the stable performance regardless of the number of tags because it adaptively varies the frame size for the number of tags.

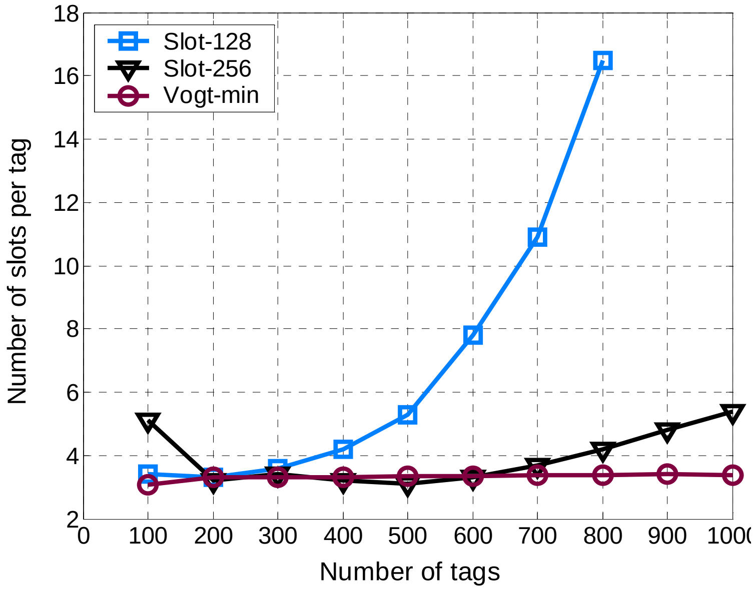

Figure 9. Number of slots per tag vs. number of tags

6.2.2.2. Proposed DFSA algorithm

Figure 10 shows the number of slots per tag versus the number of tags. Distinct from FS-ALOHA algorithms, the algorithms using Tag Estimation (TE) show the saturated range of the number of slots per tag. When applying the parameters defined in ISO 18000-6 Type A to those algorithms, the range of the number of slots per tag is approximately from 3 to 3.4.

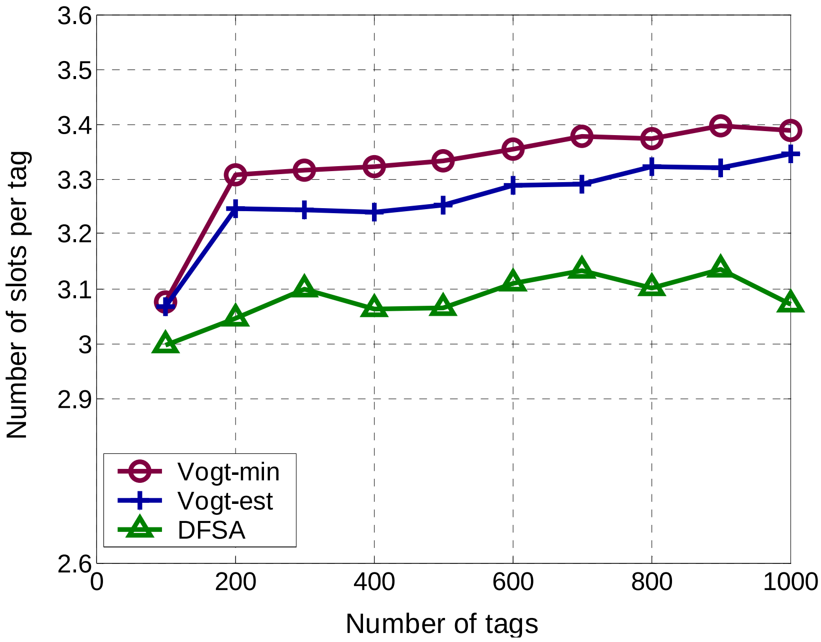

Figure 10. Number of slots per tag vs. number of tags

6.2.3. Identification time vs. number of tags

6.2.3.1. Conventional FS-ALOHA algorithms

In this section, we compare the conventional algorithms in the viewpoint of the identification time for the number of tags as shown in Figure 11.

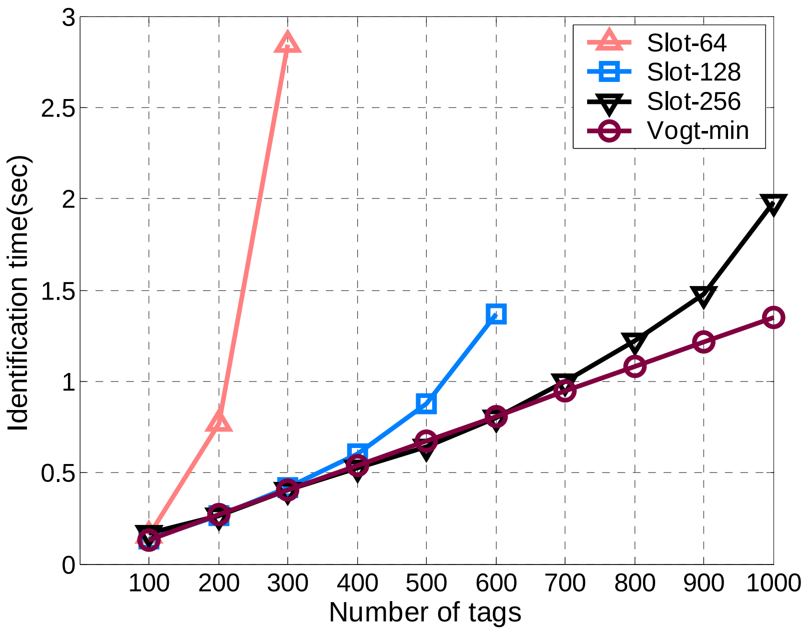

Figure 11. Identification time vs. number of tags

As mentioned before, we use the same parameters and frame structure defined in ISO 18000-6 Type A which is represented in Figure 5. The performance of FS-ALOHA algorithms shows a rapid decrease as the number of tags increases because FS-ALOHA algorithms employ the fixed frame size. So we have to use FS-

ALOHA algorithms in the specific applications; they must be applied within the allowable range of the number of tags to prevent the RFID system from being unstable. Because the performance of FS-ALOHA algorithms varies according to the number of tags, it is difficult to calculate the number of tags. But if we consider when the number of tags is 300 in which it is more likely that there are appropriate applications, the number of identified tags per second is 106 for Slot-64, 720 for Slot-128, 737 for Slot-256, and 745 for Vogt-min respectively.

6.2.3.2. Proposed DFSA algorithm

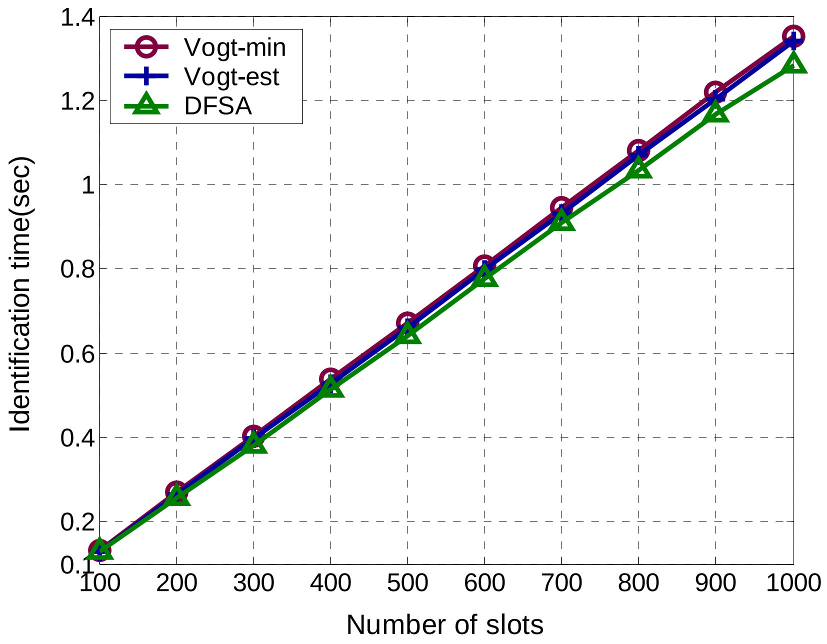

Figure 12. Identification time vs. number of tags

We now evaluate the performance of the algorithms estimating the number of tags around the reader. In the case of FS-ALOHA algorithms, the performance varies according to the number of tags. However the algorithms using TE show the stable performance. Although the performance of the algorithms show the similar performance, the proposed DFSA algorithm is better because it enables faster tag identification and the complexity is lower so that it is easier to be implemented in the system. When considering the parameters defined in ISO 18000-6 Type A, the proposed DFSA algorithm which shows the best performance identifies approximately 777 tags per second.

Chapter 7. Conclusion

In this thesis we proposed the DFSA (Dynamic Framed Slotted ALOHA) algorithm using Dynamic Slot Allocation (DSA) and Tag Estimation (TE) to improve the performance of conventional ALOHA-type anti-collision algorithms. We also compared the performance of the proposed DFSA algorithm with that of the conventional ALOHA-type algorithms using OPNET simulation. To more quantitatively evaluate the performance of each algorithm, we applied the same parameters and frame structure defined in ISO 18000-6 Type A to all the algorithms.

According to the simulation, the performance of FS-ALOHA algorithms shows a rapid decrease as the number of tags increases because FS-ALOHA algorithms employ the fixed frame size. So we have to use FS-ALOHA algorithms in the specific and restricted applications. On the other hand, the algorithms using TE and DSA show the stable performance. Although the performance of the algorithms shows the similar performance, the proposed DFSA algorithm is better because it enables faster tag identification and the complexity is lower so that it is easier to be implemented in the RFID system. When considering the parameters defined in ISO 18000-6

Type A, the proposed DFSA algorithm, which shows the best performance, can identify approximately 777 tags per second on an average. In conclusion, it will be thought over that although FS-ALOHA algorithms have to be applied within the allowable range of the number of tags to prevent the RFID system from being unstable, they have merits in the viewpoint of the simplicity of the implementation in an RFID system. However, considering not only stable but also more flexible RFID systems regardless of the number of tags, it is expected that the proposed DFSA algorithm using TE and DSA be used in an RFID system where the ability to simultaneously identify many tags is crucial for many applications.

Reference

[1] S. Garfinkel and H. Holtzman, *Understanding RFID technology*, pp. 15-36, Jun. 2005.

[2] S. Sarma, J. Waldrop, and D. Engels, "Colorwave : An anti-collision algorithm for the reader collision problem," in *Proc. ICC '03*, vol. 2, May 2003, pp. 1206-1210.

[3] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, "Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems," in *Proc. IEEE SPC*, Mar. 2003.

[4] M. Jaoccurt, A. Ehrsam, and U. Gehrig, "Contactless Identification Device With Anti-collision Algorithm," in *Proc. IEEE CSCC'99*, Athens, Jul. 1999, pp. 269-273.

[5] S. Sarma, D. Brock, and D. Engels, "Radio Frequency Identification and Electronic Product Code," in *Proc. IEEE MICRO,* 2001.

[6] J. R. Cha and J. H. Kim, "Fast Anti-collision Algorithm in Ubiquitous ID System," KICS, vol. 29, 8A, pp. 942-949, Aug. 2004.

[7] H. S. Choi, J. R. Cha, and J. H. Kim, "Fast Wireless Anti-collision Algorithm in Ubiquitous ID System," in *Proc. IEEE VTC 2004*, L.A., USA, Sep. 2004.

[8] H. S. Choi, J. R. Cha, and J. H. Kim, "Improved Bit-by-bit Binary Tree Algorithm in Ubiquitous ID System," in *Proc. IEEE PCM 2004*, Tokyo, Japan, Nov. 29-Dec. 3, 2004, pp. 696-703.

[9] J. R. Cha and J. H. Kim, "ALOHA-based Anti-collision Algorithm Estimating the Number of Tags in an RFID System," KICS, vol. 30, 9A, pp. 814-821, Sep. 2005.

[10] J. R. Cha and J. H. Kim, "Novel Anti-collision Algorithms for Fast Object Identification in an RFID System," in *Proc. ICPADS2005*, Fukuoka, Japan, Jul. 2005, pp. 63-67.

[11] Auto-ID Center, *Draft Protocol Specification for a Class 0 Radio Frequency Identification tag.*, Feb. 2003.

[12] Auto-ID Center, *Draft Protocol Specification for a Class 1 Radio Frequency Identification tag.*, Feb. 2003.

[13] ISO/IEC 18000-6:2003(E), *Part 6: Parameters for air interface communications at 860-960 MHz*, Nov. 2003.

[14] J. L. Massey, "Collision Resolution Algorithms and Random-Access Communications," UCLA, LA, Tech. Rep. UCLAENG -8016, Apr. 1980.

[15] EPC™, *Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz – 960 MHz Version 1.0.9.*, Dec. 2004.

[16] ISO/IEC 18000-6:2005(E), *Part 6C : Parameters for air interface communications at 860 MHz to 960 MHz*, Nov. 2003.

[17] R. Glidden et al. , "Design of Ultra-Low-Cost UHF RFID Tags for Supply Chain Applications," *IEEE Commun. Mag.*, Aug. 2004, pp. 140-151.

[18] H. Vogt, "Efficient Object Identification with Passive RFID Tags," in *Proc. ICPC2002,* Zürich, 2002, pp. 98-113.

[19] N. L. Johnson and S. Kotz., *Urn Models and Their Applications.,Wiley*, 1977.