Performance Evaluation of Q-Algorithm with Tag Number Estimation Scheme

Intaek Lim, *Member, KIMICS*

***Abstract*—EPCglobal Class-1 Gen-2 standard proposed Q-algorithm to select a frame size for the next query round. Q-algorithm calculates the frame size without estimating the number of tags. Therefore, the Q-algorithm has advantage that the reader's algorithm is simpler than other algorithms. However, it is impossible to allocate the optimized frame size. Also, the conventional Q-algorithm does not define an optimized parameter value *C* for adjusting the frame size. In this paper, we propose a modified Q-algorithm with the tag number estimation scheme, and evaluate the performance with computer simulations.**

***Index Terms*—RFID, Tag number estimation, Anti-collision algorithm, Q-algorithm**

I. INTRODUCTION

An RFID system consists of radio frequency tags attached to objects that need to be identified and one or more electromagnetic readers. When there is more than one tag in the identification range of a reader, all or some tags may send their response back to the reader at the same time. If two or more tags answer, their messages will collide on the RF channel and cannot be correctly received by the reader. This may lead to mutual interference, which is referred to as a collision. A technical scheme that handles multiple-access without any interference is called as an anti-collision algorithm [1].

EPCglobal Class-1 Generation-2 and ISO/IEC 18000-6 Type C standards use the probabilistic approach [2][3]. The probabilistic algorithms are based on ALOHA-like protocol that provides slots for the tags to send their data. Almost all the probabilistic algorithms use framed slot ALOHA (FSA) [4]. A lot of researches have been performed to enhance the performance of FSA algorithm. Among those algorithms, DFSA (Dynamic Framed Slot SLOHA) dynamically allocates the frame length based on the number of tags within the reader's identification range. There are two main research areas in DFSA algorithm [5]: i) tag number estimation scheme, and ii) dynamic frame size allocation scheme.

EPCglobal Class-1 Generation-2 standard proposed Q-algorithm to determine the frame size for the next query round [2]. Q-algorithm calculates the frame size without conducting a tag number estimation. Therefore, it wastes less computational cost and is simpler than other DFSA algorithms. But the constant parameter *C* value, which is used for calculating the next frame size, is not optimized. Also, it adjusts the frame size of next query round only with the slot status. It may lead to allocate a non-optimal frame size. Therefore, in this paper, we propose a modified Q-algorithm with the tag number estimation scheme.

This paper os organized as follows. In Section II, we describe EPCglobal Class-1 Generation-2 anti-collision algorithm, and Section III describes Q-algorithm. Section IV presents the tag number estimation scheme and the modified Q-algorithm. Section V shows the simulation results. Section VI concludes the paper.

II. GEN-2 ALGORITHM

EPCglobal Inc. proposed Class-1 Generation-2 RFID protocol operating at the 860MHz–960MHz UHF band. EPCglobal Class-1 Generation-2 uses a frame-based slot ALOHA as an anti-collision algorithm for identifying tags within the reader's identification range.

The reader begins a query round by transmitting a Query command. After issuing a Query command to initiate a query round, the reader transmits one or more QueryRep commands to detect each slot during a query round. If there is only one tag reply for a Query command, it is a successful query round. However, if there is no tags reply or multiple tags reply, we consider it as a failure.

Fig.1 and Fig.2 illustrate the timing diagrams for the case of single tag reply and collision or no tags reply, respectively. As shown in the figures, if only one tag transmits its RN16 for a Query command, the reader successfully receives without collisions, and then transmits an ACK command. If the tag receives the ACK command with a correct RN16, it backscatters it PC, EPC, and CRC-16.

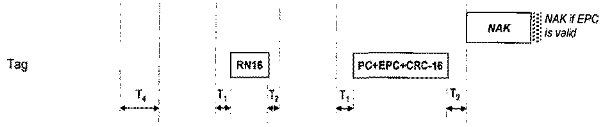


Fig. 1. Single tag reply.

Manuscript received January 7, 2010; revised January 12, 2010; accepted February 18, 2010.

Intaek Lim is with the Department of Embedded IT, Pusan University of Foreign Studies, Busan, 608-738, Korea (Email: itlim@pufs.ac.)

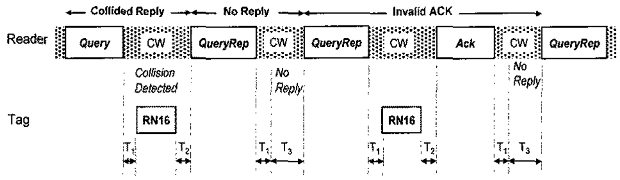


Fig. 2. No tags reply or multiple tags reply.

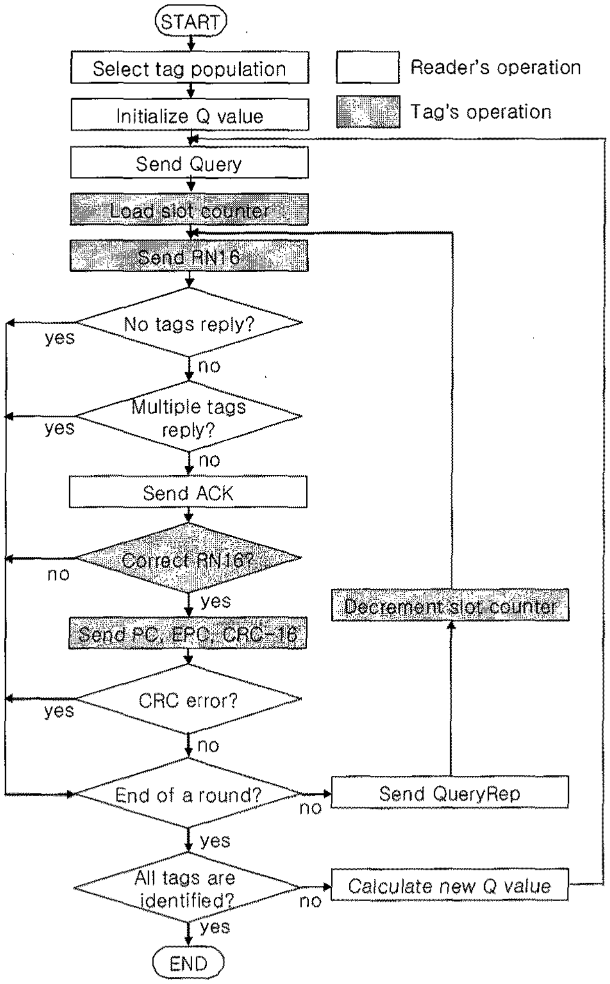


Fig. 3. Flowchart for Gen-2 anti-collision algorithm.

Fig.3 shows the operation for anti-collision algorithm that is proposed by EPCglobal Class-1 Generation-2. At first, the reader selects a tag population for a query process by transmitting a Select command. And it transmits a Query command to decide which tags participate in the query round. Query command contains a slot-count parameter Q, and the initial value of Q is 4.

Upon receiving a Query command, participating tags pick a random value in the range (0, 2^Q-1), inclusive, and load this value into their slot counter. Tags that pick a zero backscatter an RN16, which is a 16-bit random number. If the reader receivers an RN16 without collision, it acknowledges the tag with an ACK command containing this same RN16. If the tag receives the ACK command with a correct RN16, it transmits its PC, EPC, and CRC-16. The reader checks CRC errors with the received CRC-16 value. If no CRC errors are found, the reader assumes that this tag is successfully identified. After that, it reads from or writes to the identified tag.

In the EPCglobal Class-1 Generation-2 algorithm, the number of slots in the frame of query round is 2^Q slots. If a query round is not expired, the reader continues an identification process for the next slot by transmitting a QueryRep command. On the other hand, if unidentified tags still remain though a query round is terminated, the reader issues a new Query command to initiate another query round. The new Query command also contains a slot-count parameter Q, which is calculated through Q-algorithm described later.

III. Q-ALGORITHM

EPCglobal Class-1 Generation-2 proposed Q-algorithm to determine the number of slots in the next query round. Q-algorithm basically calculates the slot-count parameter Q based on the slot status that tags are responded. The slot status is classified into three categories: success, collision, and empty slot.

Fig.4 shows an algorithm that the reader might use for setting the slot-count parameter Q in a query round. In the figure, Q_{fp} is a floating-point representation of Q. As shown in the figure, the reader updates Q_{fp} in accordance with the slot status at every slot. When a collision occurs, it adds the constant C value to the previous Q_{fp}, because it means the frame length is smaller than the number of tags. If the slot is empty, which means that there are no tag responses in the slot, the reader subtracts the constant C value from the previous Q_{fp}, because the frame length is larger than the ideal one. When a new query round begins, the reader rounds Q_{fp} to an integer value Q in the Query command. Typical values for the parameter C are $0.1<C<0.5$.

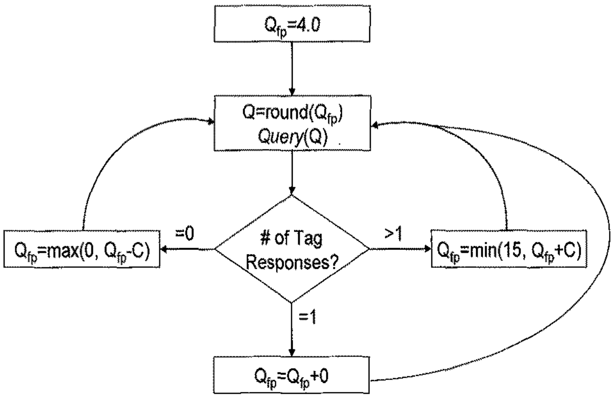


Fig. 4. Q-algorithm.

EPCglobal Class-1 Generation-2 standard suggests that the reader typically uses small values of C when Q is large and large values of C when Q is small. However, the performance of Gen-2 anti-collision algorithm, which uses the frame-based slot ALOHA, is dependent on the number of tags in the reader's identification range and frame length. Therefore, the reader must choose a constant parameter C value according to the number of tags in a query round.

In Q-algorithm of Gen-2 anti-collision algorithm, the slot-count parameter for the next query round is incremented or decremented by the constant parameter C value according to the slot status. Therefore, it is anticipated that the performance of Gen-2 anti-collision algorithm will mainly depend on the constant parameter C value and the number of tags within the identification range of reader.

Fig.5 shows the total identification time of Gen-2 algorithm according to the constant parameter C value and number of tags. The total identification time means the number of all consumed slots for identifying all tags. As depicted in the figure, the identification times of Gen-2 algorithm are various with the constant parameter C and number of tags. If C is 0.5, the reader can identify faster than other cases when the number of tags increase. In the case of constant parameter C value with 0.2, 0.3, and 0.4, the identification time increases sharply when the number of tags is 200, 300, and 350, respectively. Moreover, when the number of tags is over 400, the identification time is similar to each other with the C value between 0.2 and 0.4.

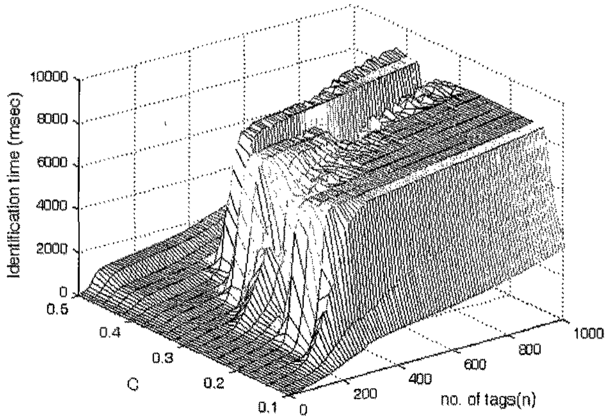


Fig. 5. Identification time according to C value.

Table 1 shows the optimal C values, which are obtained by simulations, to maintain the optimal identification time with regard to the number of tags. For maintaining the optimal performance of Gen-2 anti-collision algorithm, it is necessary to choose the optimized constant parameter C values as shown in Table 1.

TABLE 1.
OPTIMAL C VALUES VS. NUMBER OF TAGS.

Number of tags (n)	C
$n \leq 107$	0.1
$108 \leq n \leq 126$	0.2
$127 \leq n \leq 183$	0.3
$184 \leq n \leq 321$	0.4
$322 \leq n \leq \infty$	0.1

IV. TAG NUMBER ESTIMATION

The reader estimates the number of tags by combining the information obtained after a query round with the result of probabilistic analysis. It is assumed that a frame consists of N slots and there are n tags in the reader's identification range. Let N_e and N_s be the number of empty slots and successful slots observed after the read cycle, respectively. If we assume that the number of empty slots and successful slots observed after the read cycle is equal to their expected values, respectively [6], then N_e and N_s can be given as follows.

$$N_e = N\left(1-\frac{1}{N}\right)^n \tag{1}$$

$$N_s = n\left(1-\frac{1}{N}\right)^{n-1} \tag{2}$$

By solving Eq.(1) and (2) for the number of tags n, we obtain

$$n=(N-1)\frac{N_s}{N_e} \tag{3}$$

The number of tags that will be involved in the next query round is equal to the number of colliding tags n_c. Therefore, the number of colliding tags can be estimated by subtracting N_s from Eq.(3) as follows.

$$n_c=(N-N_e-1)\frac{N_s}{N_e} \tag{4}$$

Because the reader can obtain the number of successful slots and empty slots after the read cycle, it also can estimate the number of unidentified tags from Eq.(4). However, as shown in Eq.(4), if there are neither successful slots nor empty slots, we cannot use the above equation. Therefore, the proposed scheme supplements the result of probabilistic analysis for the exact estimation.

If we let α be the average number of tags in each collision slot, α can be given by

$$\alpha = \frac{\sum_{r=2}^{n} rN\binom{n}{r}\left(\frac{1}{N}\right)^{r}\left(1-\frac{1}{N}\right)^{n-r}}{N - N\left(1-\frac{1}{N}\right)^{n} - n\left(1-\frac{1}{N}\right)^{n-1}} \qquad (5)$$

Fig.6 shows the average number of tags in a collision slot when the frame size is equal to the number of tags (N=n) and when the frame size is optimal value obtained from the analysis results of FSA algorithm (N=optimal) [7]. In FSA algorithm, the optimal performance for the system efficiency and identification delay can be obtained when the frame size is equal to the number of tags. Therefore, as shown in the figure, the average number of tags in each collision slot is 2.4. If we let N_c be the number of collision slots, the number of unread tags are give by

$$n_c = 2.4N_c \qquad (6)$$

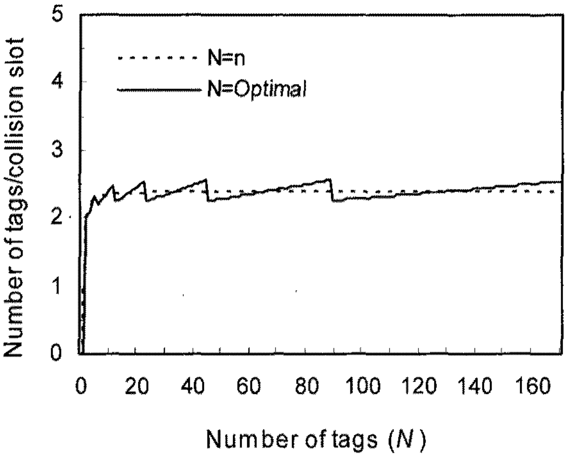


Fig. 6. Number of collision tags per collision slot.

By combining the observed result in Eq.(4) with the probabilistic result in Eq.(6), we can estimate the number of unread tags as following.

$$n_c = \begin{cases} (N - N_e - 1)\dfrac{N_s}{N_e} & \text{, if } N_e \neq N_s \neq 0 \\ 2.4N_c & \text{, otherwise} \end{cases} \qquad (7)$$

Fig.7 compares the estimation error for the proposed estimation scheme when the frame size is 128. When the number of tags is large, the estimation error increases. But when number of tags is below 310, there is almost no estimation error.

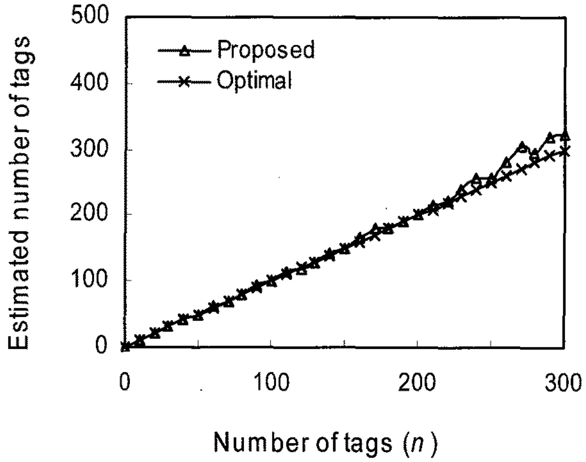


Fig. 7. Comparison for the estimated number of tags.

Fig. 8 shows the operation of Q-algorithm with the tag number estimation scheme. At the initial query round, the operation of modified Q-algorithm is same as the original Q-algorithm with a random C values. The reader collects the number of collision slots, idle slots, and successful slots at every round. After a query round, the reader estimates the number unidentified tags by using Eq.(7), and starts the next query round with the new C value in Table 1 based on the estimated number of tags.

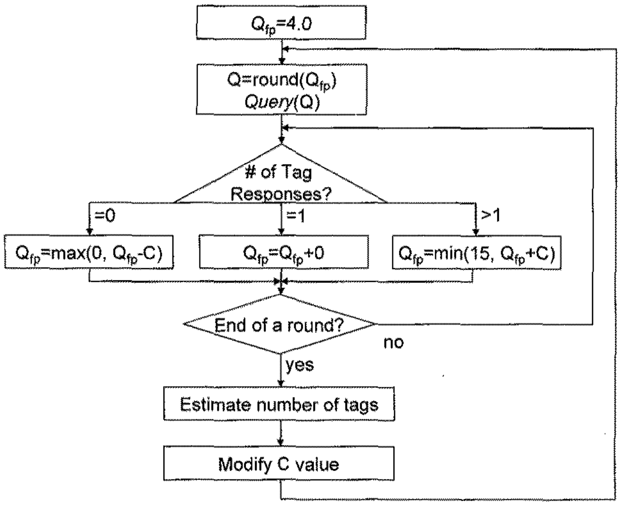


Fig. 8. Modified Q-algorithm.

V. SIMULATION RESULTS

In this paper, computer simulations are performed to evaluate the performance of modified Q-algorithm with the tag number estimation scheme. The system parameters for simulation are shown in Table 2, which are based on the EPCglobal specification.

TABLE 2.
SIMULATION PARAMETERS.

Parameter	Value
T_{ari}	12.5μs
0_{length}	12.5μs
1_{length}	18.75μs
RT_{cal}	31.25μs
TR_{cal}	64μs
LF	125KHz
T_{pri}	8μs
RT_{rate}	64Kbps
TR_{rate}	125Kbps
T_1	80μs
T_2	80μs
T_3	0μs
T_4	62.5μs
R=>T preamble	120.25μs
R=>T frame sync	56.25μs
T=>R preamble	48μs

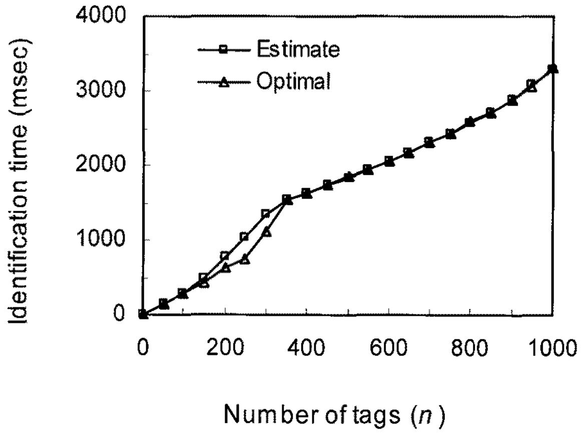


Fig. 9. Comparison between estimation and optimal.

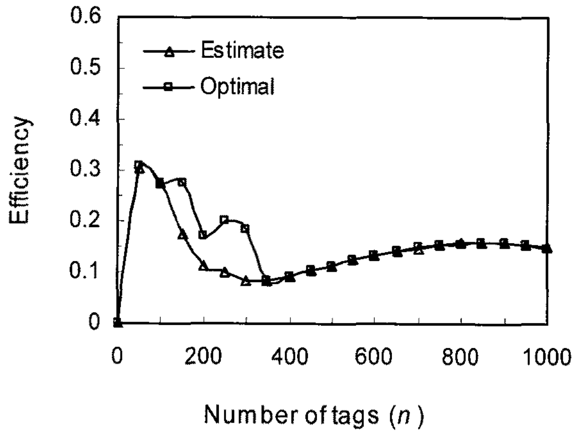


Fig. 10. Comparison of efficiency.

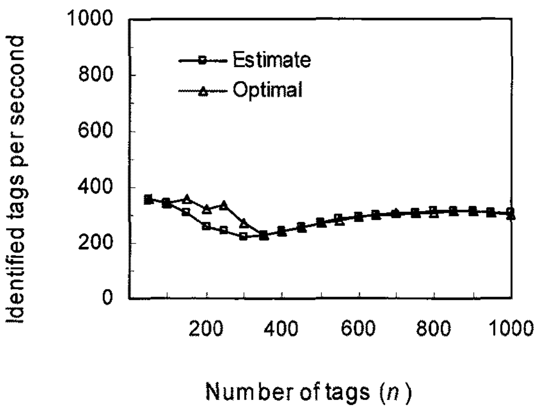


Fig. 11. Comparison of identification speed.

Fig.9 compares the optimal identification time of Q-algorithm with the modified Q-algorithm according to the number of tags.. Fig.10 shows the comparison of system efficiency. The system efficiency is defined as the number of identified tags per a slot. Fig.11 depicts the identification speed. The identification speed means the number of identified tags per a second. The optimal identification time that is plotted in the figures is obtained with the optimal constant parameter C values shown in Table 1. On the other hand, the line estimate is obtained from the modified Q-algorithm with the tag number estimation scheme. As shown in the figures, if we apply the optimal C values after estimating the number of tags, the modified Q-algorithm can maintain the optimal performances.

VI. CONCLUSIONS

Q-algorithm proposed by EPCglobal calculates the frame size without estimating the number of tags. Therefore, it wastes less computational cost and is simple. But it has a drawback that it cannot consider the number of unidentified tags within the reader's identification range in allocating the frame size. Also, the constant parameter C value, which is used for calculating the next frame size, is not optimized. This paper proposed a modified Q-algorithm, which allocates the frame size with the optimized C value after estimating the number of tags.

REFERENCES

[1] M. A. Bonucelli, F. Lonetti, and F. Martelli, "Instant Collision Resolution for Tag Identification in RFID networks," *Ad Hoc Networks*, vol.5, pp.1120-1232, 2007.

[2] EPCglobal, "EPC Rdio-Frequency Idetity Protocols Class-1 Generation-2 UHF RFID Protocols for Communication at 860MHz-960MHz, Ver.1.2.0," *EPCglobal Inc.*, Oct. 2008.

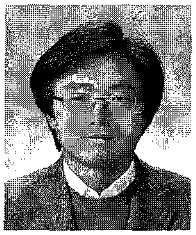
[3] ISO/IEC, "Information Technology – Radio Frequency Identification for Item Management – Part 6: Parameters for Air Interface Communication at 860-960 MHz, 19000-6," *ISO/IEC*, 2006.

[4] W. Chen, and G. Lin, "An Efficient Anti-Collision Method for tag Identification in a RFID System," *IEICE Trans. Commun.*, vol.E89-B, no.12, pp.3386-3392, Dec. 2006.

[5] H. Vogt, "Efficient Object Identification with Passive RFID Tags," First International Conf. on Pervasive Computing, *LNCS*, vol.2414, pp.99-113, Springer-Verlag, 2002.

[6] I. Lim, "A Scheme for Estimating Number of Tags in FSA-based RFID Systems," *International Journal of MICS*, vol.7, no.2, pp.164-169, June 2009.

[7] I. Lim, and J. Choi, "Dynamic FSA Algorithm for Fast Tag Identification in RFID Systems," *Journal of KIMICS*, v.10, no.5, pp.806-811, May 2006.



Intaek Lim received the B.S. degree in computer science from University of Ulsan, Ulsan, Korea, in 1984, and the M.S. degree in computer science and statistics from Seoul National University, Seoul, Korea, in 1986. He received the Ph. D. degree in computer engineering from University of Ulsan, Ulsan, Korea, in 1998. From 1986 to 1993, he was a Senior Researcher at Samsung Electronics Co., Ltd. In 1998, he joined the faculty at Pusan University of Foreign Studies, Busan, Korea, where he is currently a Professor in the Department of Embedded IT. From July 2006 to June 2007, He was a visiting scholar at Cleveland State University in USA. His research interests include the MAC protocol design, ad-hoc network, RFID, and mobile computing.