Adaptive Q-Algorithm for Multiple Tag Identification in EPCglobal Gen-2 RFID System

Intaek Lim, *Member, KIMICS*

***Abstract*—EPCglobal Class-1 Gen-2 protocol has been proposed for UHF-band RFID systems. In Gen-2 standard, Q-algorithm was proposed to select a frame size for the next query round 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 original Q-algorithm did not define an optimized parameter *C* for adjusting the frame size. In this paper, we propose an adaptive Q-algorithm with the different parameter C_c and C_i in accordance with the status of reply slot. Simulation results show that the proposed adaptive Q-algorithm outperforms the original Gen-2 Q-algorithm.**

***Index Terms*—RFID, Anti-collision algorithm, EPCglobal Gen-2, Q-algorithm**

I. INTRODUCTION

In RFID system, 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][2].

EPCglobal Class-1 Generation-2 and ISO/IEC 18000-6 Type C standards use the probabilistic approach [3][4]. The probabilistic algorithms are based on ALOHA-like protocol that provides slots for the tags to send their data [5]. Almost all the probabilistic algorithms use framed slot ALOHA (FSA) [6]. 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: i) tag number estimation scheme, and ii) dynamic frame size allocation scheme [7][8].

EPCglobal Class-1 Generation-2 standard proposed Q-algorithm to determine the frame size for the next query round [3]. The Gen-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 an adaptive Q-algorithm, which differentiates the parameter Q between in collision case and idle case.

This paper is organized as follows. In Section II, we briefly describe EPCglobal Class-1 Gen-2 anti-collision algorithm, and in Section III propose an adaptive Q-algorithm. Section IV shows the simulation results. Section V concludes the paper.

II. GEN-2 RFID SYSTEM

A. Gen-2 Anti-collision 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 with *Q* value. 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 shows the simplified tag state transition diagram of Generation-2 anti-collision algorithm. Upon entering an energizing RF field, a tag that is not killed enters Ready state and remains in that state until it receives a Query command. When tags receive a Query command, they draw a Q-bit number from their random number generator, load this number in the range (0, 2^Q-1) inclusive into their slot counter, and change their state into the Arbitrate state if the number is nonzero, or into the Reply state after transmitting an RN16 if the number is zero.

A tag in Arbitrate state decrements its slot counter every time it receives a QueryRep command, and it goes to the Reply state and backscatters an RN16 when its slot counter reaches zero. If it receives a QueryAdjust command, it adjusts its *Q* value according to UpDn field of QueryAdjust command, selects a new random number, and loads this number into its slot counter. A tag that reinitialized its slot counter remains in the Arbitrate state

---

Manuscript received May 11, 2010; revised May 12, 2010; accepted May 27, 2010.

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

if the slot counter is nonzero, or goes to the Reply state.

Upon entering Reply state, a tag backscatters an RN16. If the tag receives a valid acknowledgement, it transmits its <PC, EPC, CRC-16>, and enters the Acknowledged state. If the tag fails to receive an ACK command within the specified time, or receives an invalid ACK or an erroneous RN16, it returns to the Arbitrate state. After acknowledging a tag, a reader can choose to access it.

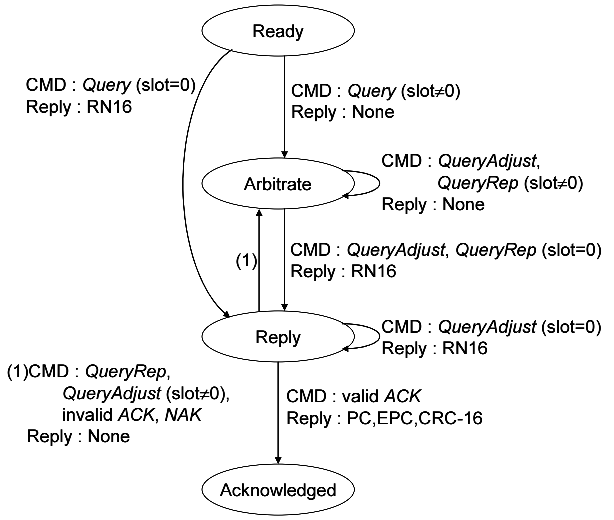

Fig. 1. Tag state diagram.

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.2 and Fig.3 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 its PC, EPC, and CRC-16.

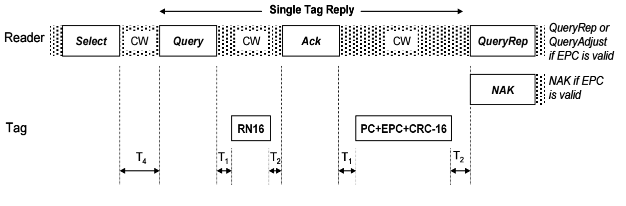

Fig. 2. In the case of single tag reply.

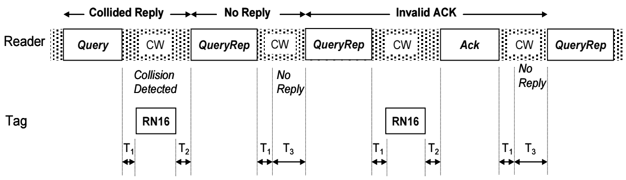

Fig. 3. In the case of collision or no reply.

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.

A. 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 $0.1<C<0.5$.

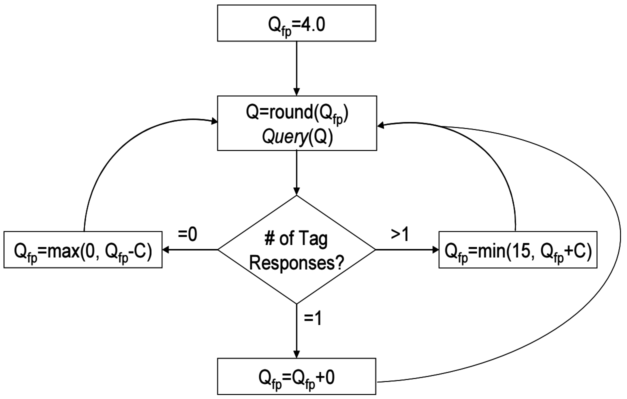

Fig. 4. Q-algorithm.

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.

III. ADAPTIVE Q-ALGORITM

In Q-algorithm, if the value of parameter *C* is relatively large, the frame size can converge to the optimal point very quickly. However, the oscillation of frame size will occur near the optimal point. On the other hand, in the case of that the value of parameter *C* is small, the convergence to the optimal frame size will be very slow, but the frame size rarely changes after converging to the optimal point. Therefore, it is expected that the value of parameter *C* will be different according to the tag's reply.

In the proposed adaptive Q-algorithm that will be described later, we adopt new parameters named C_c and C_i which mean the parameter *C* for collision and idle slot, respectively. C_c and C_i will be the important parameters that influence the performance of Gen-2 anti-collision algorithm.

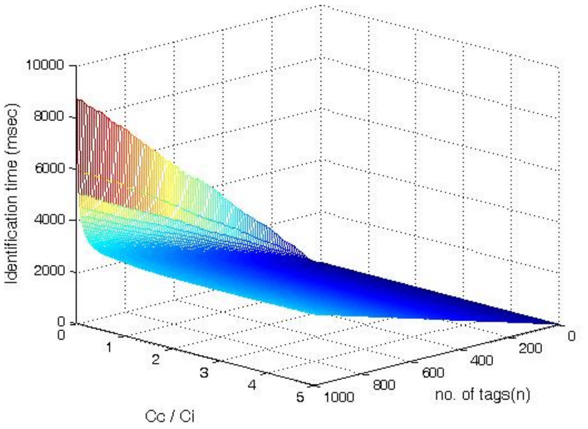

Fig. 5. Identification time vs. ratio of C_c to C_i.

Through the experiment with simulations, Fig. 5 shows the total identification time of Gen-2 anti-collision algorithm according to the ratio of C_c to C_i. The simulation parameters are listed in Table I. In the simulations, we assumed that the Query commands are transmitted as follows: 1) when there is a successful reply, the reader transmits QueryRep command; 2) when there is a collided reply or no reply, the reader sends QueryAdjust command if *Q* gets changed or QueryRep command if *Q* has no change. As shown in Fig. 3, the identification time of Gen-2 anti-collision algorithm is changed by the ration of C_c to C_i, and almost same when the ration is over 3.

TABLE 1.
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

As shown in Fig. 3, the slot duration in Gen-2 RFID system is different whether the slot has a collided reply or no reply. Let T_c and T_i denote the duration of a collided reply and the duration of no reply, respectively. T_c and T_i can be written as following:

$$T_c = T_1 + RN16 + T_2 \tag{1}$$

$$T_i = T_1 + T_3 \tag{2}$$

We assumed that in Table I TRrate is 125Kbps. With this assumption, we know that the duration of a collided reply is about 4.3 times longer than that of no reply because the values of T_c and T_i are 336μsec and 80μsec, respectively. Therefore, the identification time of Gen RFID system could be improved if we can adjust the value of parameter *C* in a way to increase the number of no reply slots while relatively decreasing the number of collided slots.

To improve the identification time, we also adopt the probabilistic approaches while selecting the value of parameter *C*. Let P_c and P_i denote the probabilities that a slot is collision and idle, respectively. It is assumed that a frame consists of *N* slots and there are *n* tags in the reader's identification range. Also, we assume that the tag selects one of *N* slots with the equal probability because it generates a uniformly distributed random number within the range from 1 to frame size. For a given slot, the number of tags selected that is a binomial distribution with *n* Bernoulli experiments and 1/*N* occupied probability. Therefore, the probability P_c and P_i can de defined as follows.

$$P_c = \lim_{n\to\infty}\left[1-\left(1-\frac{1}{N}\right)^n\left(1+\frac{1}{N}\right)^n\right] = \left(1-\frac{2}{e}\right) \quad (3)$$

$$P_i = \lim_{n\to\infty}\left(1-\frac{1}{N}\right)^n = \frac{1}{e} \quad (4)$$

As described above, the probability that a slot has no reply is higher than the probability that a slot has a collided reply. Also, the duration of a collided reply is longer than the duration of no reply. Therefore, the value of parameter C must be applied differently according to the slot status. In this paper, we propose an adaptive Q-algorithm, which uses the parameter C_c and C_i in the case of collided reply and no reply, respectively.

The basic design principles for the proposed adaptive Q-algorithm are as follows. As mentioned above, the duration of collided slot is longer than the duration of idle slot. Therefore, when the slot has a collided reply, it is necessary for the frame size to be increased as soon as possible. Also, when the slot has no reply, the frame size should be decreased slowly because the probability that a slot is idle is greater than the probability that a slot is collision. In the adaptive Q-algorithm, we always let C_c to be larger than C_i. Fig. 6 shows the operation of adaptive Q-algorithm. The basic operation of adaptive Q-algorithm is very similar to the original one. The only difference is the differentiation of parameter C between in collision cases and idle cases. We can see that it is easy for the reader to obtain T_c and T_i. And we can use P_c and P_i as constant values. Therefore, the proposed adaptive Q-algorithm does not impose any change on tags while a little bit overhead at the reader.

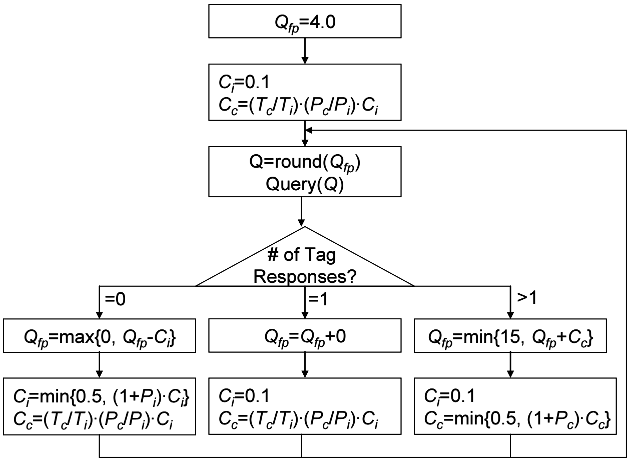

Fig. 6. Operation of adaptive Q-algorithm.

IV. SIMULATION RESULTS

In this section, we evaluate the performance of proposed adaptive Q-algorithm through the computer simulations. The system parameters for simulation are same as Table I in Section III. Also it is assumed that the transmission scenarios of Query command are same as Section III. We compare the adaptive Q-algorithm with the original Gen-2 Q-algorithm with the parameter C=0.3. All the results of the simulation are averaged after iterating it 100 times.

We evaluate the total identification time and identification speed, and show these in Fig. 7 and Fig. 8, respectively. The total identification time means the time consumed until all the tags within the reader's identification range are recognized by a reader. The identification speed means the ratio of the total number of identified tags over one second. As shown in the figures, the adaptive Q-algorithm outperforms the Gen-2 Q-algorithm by around 4.5%.

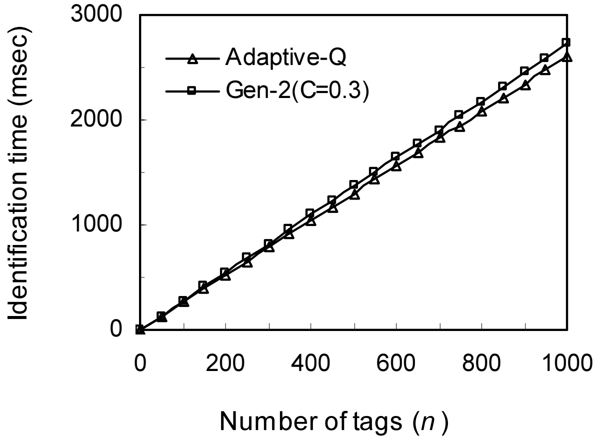

Fig. 7. Comparison of total identification time.

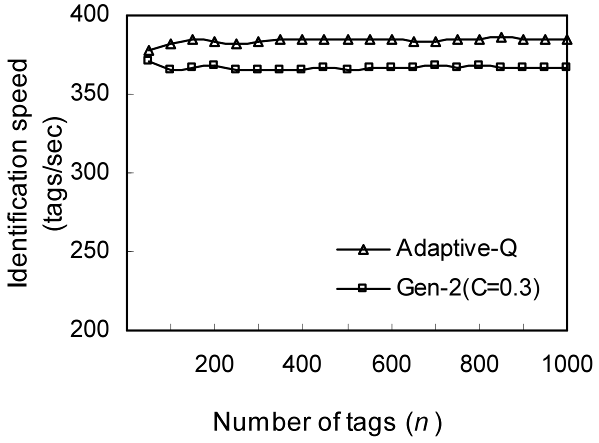

Fig. 8. Comparison of identification speed.

In addition, Fig. 9 shows the collision ratio according to the number of tags. The collision ratio is defined as the ration of the number of collided slots over the total number of slots consumed for identifying all tags. In the adaptive Q-algorithm, we considered the idle probability of slot as well as the collision probability when calculating the value of parameter C_c and C_i. Therefore, as shown in Fig. 9, the collision ratio of adaptive Q-algorithm is 65% less than the Gen-2. In general, because the duration of collided slot is longer than no reply slot, the fewer number of collisions is, the faster the reader will identify all tags. Therefore, as shown in Fig. 7, the total identification time of adaptive Q-algorithm is shorter than the Gen-2 because the collision ratio of adaptive Q-algorithm is less than the Gen-2.

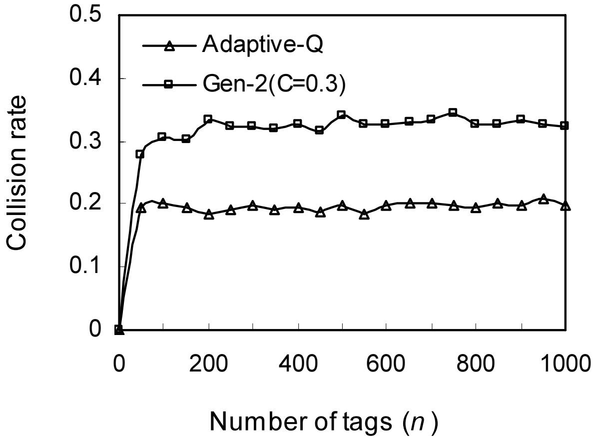

Fig. 9. Comparison of collision ratio.

V. CONCLUSIONS

This paper proposed an adaptive Q-algorithm, which enhances the performance of EPCglobal Gen-2 RFID system. We introduced new parameters C_c and C_i when calculating the slot count parameter Q. The adaptive Q-algorithm updates the parameter Q faster in the case of collided reply than in the case of no reply. The simulation results demonstrated that the adaptive Q-algorithm shows high performance compared with the original Gen-2 Q-algorithm. Especially, the number of collided slots is 65% less than the original Gen-2 Q-algorithm.

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] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, Carl Hanser Verlag GmbH & Co., 2002.

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

[4] 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.

[5] 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.

[6] B. Zhen, M. Kobayashi, and M. Shimizu, "Framed ALOHA for Multiple RFID Objects Identification," *IEICE Trans. Commun.*, vol.E88-B, no.3, pp.991-999, Mar. 2005.

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

[8] C. Wang, M. Daneshmand, and K. Sohraby, "Optimization of Tag Reading Performance in Generation-2 RFID Protocol," *Computer Commun.*, vol.32, Issue 11, pp.1346-1352, July 2009.

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.