Efficient Dual Bias Q-Algorithm and Optimum Weights for EPC Class 1 Generation 2 Protocol

Dongwook Lee, Okkyeong Bang, Sooyeol Im, and Hyuckjae Lee
School of Electrical Engineering
Information and Communications University (ICU)
119 Mungiro, Yuseong-gu, Daejeon, 305-732, Korea
Email:{dwook, railmery, iamsyim, and hjlee}@icu.ac.kr

***Abstract*—Passive RFID tags, which have no self-battery and just backscatter the energy from a reader, share a common channel. It causes a tag-to-tag collision problem when at least two or more tags communicate to the reader simultaneously. The dynamic frame slotted Aloha (DFS-Aloha) protocol is one of the well-known anti-collision algorithms to solve this problem. To maximize the system performance and enhance the reading speed of Aloha protocols, we propose a frame-size estimator which is a modified version of Q-Algorithm. To maximize the system performance, the optimum bias values of the proposed estimator are searched by using Least Square method from mathematical model of DFS-Aloha. Simulation results show that the proposed estimation algorithm enhances the system performance. The proposed estimation algorithm takes less identification time than DFS-Aloha with the existing Q-Algorithm based on EPC Class 1 Generation 2 protocol.**

I. INTRODUCTION

A radio frequency identification (RFID) system has attracted considerable attention in supply markets all over the world. It is perceived as a good substitute for bar code system because RFID tags are contactless types, and they can store and modify a lot of data in their memories. Nevertheless, it has some open problems to be solved especially in passive RFID tag systems. The most critical problem among them may be an interference among tags because all tags share same bandwidth, and are not able to communicate with each other. When two or more tags transmit their data simultaneously, whole messages backscattered from the tags are corrupted, which is typically called a collision problem [1].

Dynamic frame slotted Aloha (DFS-Aloha) is one of the most widely used anti-collision algorithms in RFID systems [1]-[3]. In the DFS-Aloha algorithm, each tag transmits its data in a frame at a random slot to avoid collisions. Therefore, the system efficiency strongly depends on the frame sizes and the number of tags. The DFS-Aloha varies its frame size according to the current traffic. However, since the reader cannot have any information about the number of tags to read. it needs to estimate the number of tags by observing the collision pattern of the current frame and decide the next frame size which fits for the unread tags.

Most of recent low-cost RFID systems follow EPCglobal Class 1 Generation 2 (Gen 2) protocol which is approved as RFID air-interface standard for ultra high frequency (UHF) band (ISO 18000-6 Type C) [4]. In this protocol, the DFS-Aloha is adopted as anti-collision algorithm. Q-Algorithm is used as a frame-size estimator of DFS-Aloha in Gen2. It estimates not the length but the exponent of the next frame size. The estimator is simple because it just weights to the number of collided and empty slots. However, there have been no systematic ways to find the optimum weight for the estimator. Most of researches on this issue have been performed by brute-force searching using computer simulation. Unfortunately, it is hard to achieve theoretical maximum performance with the results.

In this paper, we propose an enhanced frame-size estimator based on Q-Algorithm which provides more accurate estimation. The proposed estimator is very similar with Q-Algorithm but has different weights for the number of collided and empty slots. To find the optimal weights, we formulate a mathematical model to express various observations of DFS-Aloha. Then, the averaged optimum exponents for the actual number of tags are searched under the assumption that the reader knows the exact number of tags and varies its frame size to maximize the system efficiency. Line-fitting is performed based on the data and the expected collision patterns, and then the optimum weights are finally obtained by the least square method. The simulation results show that the proposed estimator with the searched weights enhances the accuracy of estimator, compared to the existing Q-Algorithm. The proposed estimator also reduces the total number of slots for DFS-Aloha and identification time of Gen2-based protocol.

The rest of this paper is organized as follows. In Section II, Gen2 protocol, its frame-size estimator, and Q-Algorithm are briefly explained to compare with the proposed scheme. In Section III, we introduce the proposed frame-size estimator. Then, we provide how to find the optimum weights for the estimator, based on the mathematical model. The simulation environment and performance analysis are presented in Section IV. Finally, the conclusion is remarked in Section V.

II. GEN2 PROTOCOL AND Q-ALGORITHM

Gen2 is a global UHF air-interface protocol standard, where the DFS-Aloha is implemented as shown in Fig. 1 and 2. The reader transmits the information about frame with 22-bit Query. It notifies the beginning of a frame and the exponent of the frame size to the tags. At every beginning of the slot, the reader transmits 4-bit QueryRep to the tags. Then, the tags generate random number ranging from 0 to frame size-1 and

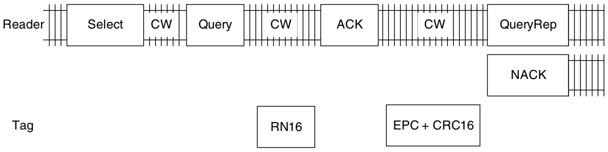

Fig. 1. Example of Gen 2 protocol for single tag reply.

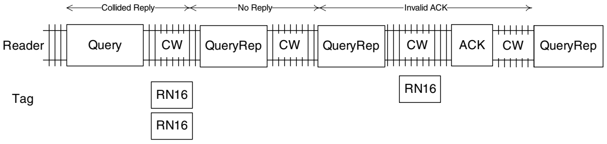

Fig. 2. Example of Gen 2 protocol.

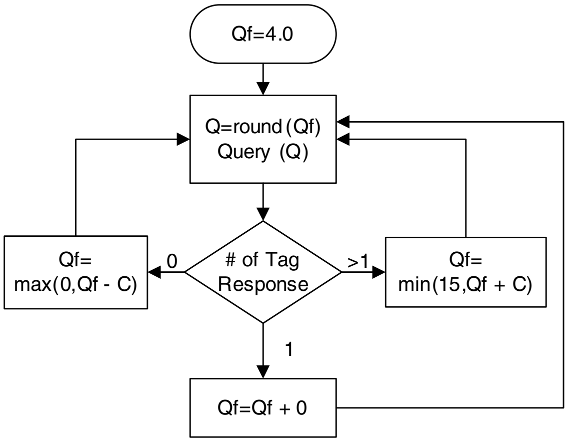

Fig. 3. Q-Algorithm in Gen 2 protocol.

count the number of QueryRep. If the counted number matches to the generated random number, the tag responds to the query of the reader. To reduce the slot time, the tag transmits 16-bit temporary ID, RN16, during its slot time. Thereby, it can save the time for collided and empty slots. If a tag successfully transmits its RN16 without error or collision, the reader sends ACK to the tag to receive desired data including 96 or 256-bit Electronic Product Code (EPC) and 16-bit CRC.

A frame-size estimator is the most key part of DFS-Aloha because the reader is not able to know how many tags exist in its region. If the estimated frame size is too large with respect to the actual number of tags, the number of empty slots will be increased. On the other hand, if it is too small, most slots in the frame will be failed due to collision. Consequently, the imperfect frame-size estimation degrades the performance of DFS-Aloha. The Q-Algorithm introduced in EPCglobal Class 1 Gen 2 protocol is very simple estimator for DFS-Aloha unlike other estimators in [5]-[8] that requires complicate computations or large memory. The flow chart of Q-Algorithm is illustrated in Fig. 3.

Let 2^{Q_i} denote the ith frame length, L_i, where Q_i represents the ith frame size, i.e.,

$$L_i = 2^{Q_i}. \tag{1}$$

For Q-Algorithm, the collided or empty slots can be regarded as lack or excess of slots in the frame. Specifically, when collision occurs in the slot of the ith frame, the Q-Algorithm increases the exponent of the ith frame length, Q_i. On the other hands, it decreases Q_i when the slot is empty. By doing so, if the ith frame is finished, the exponent of the $(i+1)$th frame length, Q_{i+1} is determined by

$$Q_{i+1} = \text{round}\left(Q_i + c\left(N_c - N_e\right)\right), \tag{2}$$

where N_c and N_e are the respective number of collided and empty slots in the ith frame, round(x) represents the integer nearest x, and the parameter c means a bias term which plays an important role in the frame-size estimation.

Generally, the reader uses small value of c when Q is large but large value of c when Q is small. In Gen 2, c is a floating-point number ranging from 0.1 to 0.5. The lower constraint of c, i.e., $c = 0.1$, allows the estimator to escape the situation that the weighting never impacts on the frame size because of its small value. For example, the frame size may not change well if the frame size is 4 and c is 0.1 even though all slots in the frame are collided or empty. In case of big frame size, however, the constraint is not necessary. Rather it restricts the performance of the estimator. Until now, although there have been several researches on the optimal value of c for the various frame sizes, most of them have been found not by systematic way but by exhaustive search with computer simulations. Thus, the estimation approach of the existing estimators may be not efficient enough.

III. DBQ-ALGORITHM

In this section, we propose an enhanced approach that assigns different weights to the collided and empty slots, which is referred to as a Dual-bias Q-Algorithm (DBQ-Algorithm). By assigning different weights to collided and empty slots, the estimator fully exploits the information about the current traffic. In this regard, the $(i+1)$th frame size can be determined by DBQ-Algorithm, which is given as

$$Q_{i+1} = \text{round}\left(Q_i + k_1 N_c - k_2 N_e\right), \tag{3}$$

where k_1 and k_2 are weights for collided and empty slots and they can be changed, relying upon the given Q_i.

A similar algorithm introduced in [10] is based on the assumption that the reader knows the number of tags. However, it is not feasible because tags cannot communicate with each other. Moreover, the derivation of parameters requires very long frame size. The scheme also does not specify the optimum k_1 and k_2 for each frame size. These reasons motivate us to investigate more systematic ways to determine optimal solution for DBQ-Algorithm, based on the mathematical model which will be introduced in the following subsection.

A. Mathematical Model

Now, we introduce the mathematical model to determine optimal solution for DBQ-Algorithm. For the m identification operations under observation, we define Q_{ij}, N_{ci}, and N_{ei}

TABLE I
DETERMINED FRAME SIZES FOR PERFECT ESTIMATOR.

The number of tags	Optimum frame length (size, Q)
1-5	4 (2)
6-11	8 (3)
12-22	16 (4)
23-44	32 (5)
45-88	64 (6)
89-180	128 (7)
181-355	256 (8)
356-700	512 (9)
701-1420	1024 (10)

where $1 \leq j \leq m$, as the frame size or equivalently the exponent of frame length, the number of collided slots, and that of empty slots for the ith frame at the jth observation. A size vector of the ith frames, $\mathbf{q}_i$, can then be written as

$$\mathbf{q}_i = \left[\begin{array}{cccc} Q_{i1} & Q_{i2} & \cdots & Q_{im} \end{array} \right]^T, \tag{4}$$

and two vectors ($\mathbf{n_c}$ and $\mathbf{n_e}$), whose elements denote the corresponding number of collided and empty slots, are also given by

$$\mathbf{n}_c = \left[\begin{array}{cccc} N_{c1} & N_{c2} & \cdots & N_{cm} \end{array} \right]^T,$$
$$\mathbf{n}_e = \left[\begin{array}{cccc} N_{e1} & N_{e2} & \cdots & N_{em} \end{array} \right]^T. \tag{5}$$

Using (5), a collision pattern matrix $\mathbf{H}_{m\times 2}$ can be expressed as

$$\mathbf{H} = \left[\begin{array}{cc} \mathbf{n_c} & \mathbf{n_e} \end{array} \right]. \tag{6}$$

Finally, the updated size vector of m frames can be approximated using (3) as

$$\mathbf{q}_{i+1} \approx \mathbf{q}_i + \mathbf{H} \cdot \left(\begin{array}{c} k_1 \\ -k_2 \end{array} \right). \tag{7}$$

B. Parameter Searching

To find the optimum weights for DBQ-Algorithm, we first need the desired estimation results obtained from the perfect estimator. Here, the perfect estimator means that the reader knows the exact number of tags, and it has capability of deciding the next optimum frame size to maximize reading speed. The estimated length and exponent (or size) of frame for the perfect estimator are averaged by the Monte Carlo simulation. In the simulation, we assume that there are no tags that attend or leave while the reader performs inventory operation. In addition, the identified tags can not participate in the remaining inventory procedures so that unnecessary competition between identified and unidentified tags never happen. Thus, the number of tags to read becomes decreasing.

Table I summarizes the optimum estimated frame length (size) for given the number of tags when the perfect estimator is used. Fig.4 illustrates the averaged estimated exponents ($\hat{Q}_2$) in terms of the number of tags when the current exponents of frames (Q_1) are 4, 6, 8, and 10. In addition, the averaged estimated length of frames are provided in Fig.5. For example, from Fig. 4 and 5, if the current frame length is $L_1 = 1024$ ($Q_1 = 10$) and there exist 500 tags in the frame, 256 ($Q_1 = 8$) is the best for the next frame size, considering the number of unread tags.

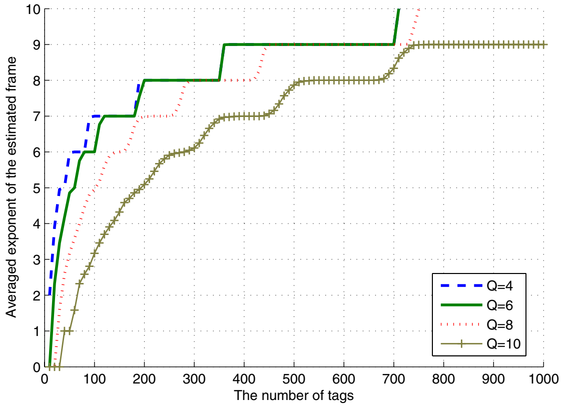

Fig. 4. Averaged exponents of the estimated frame (perfect knowledge).

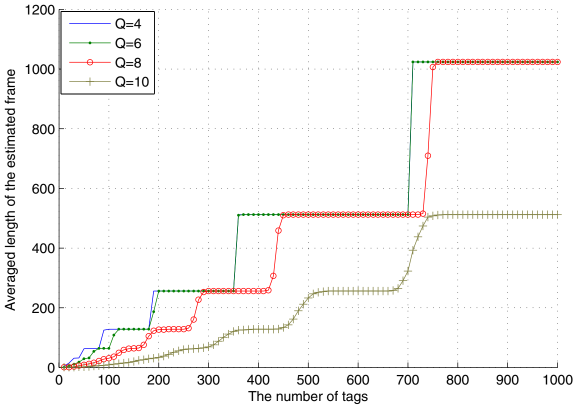

Fig. 5. Averaged lengths of the estimated frame (perfect knowledge).

After finding the reference data for the given frame size and the number of tags, we also find the expected number of collided slots $\bar{n}_c$ and empty slots $\bar{n}_e$. For the given frame size L $(= 2^{Q_1})$ and the number of tags N, the expected number of empty and collided slots are given by [3]

$$\bar{n}_e = L\left(1 - \frac{1}{L}\right)^N,$$
$$\bar{n}_c = L - L\left(1 - \frac{1}{L}\right)^N - N\left(1 - \frac{1}{L}\right)^{N-1}. \tag{8}$$

From (8), the averaged collision pattern matrix $\mathbf{H}$ can be rewritten as

$$\mathbf{H} = \left[\begin{array}{cc} \bar{\mathbf{n}}_\mathbf{c} & \bar{\mathbf{n}}_\mathbf{e} \end{array} \right]. \tag{9}$$

The DBQ-Algorithm has to work like the perfect estimator if its collision pattern is same as (8). Hence, based on the data searched for the perfect estimator, line fitting is performed to find the optimum k_1 and k_2 for given Q.

The maximum limitation is needed for DBQ-Algorithm because the reader cannot correctly decide the number of tags

TABLE II
BIASES FOR DBQ-ALGORITHM AND Q-ALGORITHM

Q (L)	k_1	k_2	c
1 (2)	0.4000	0.4000	0.5
2 (4)	0.4000	0.4000	0.5
3 (8)	0.3000	0.4000	0.4
4 (16)	0.1617	0.2627	0.3
5 (32)	0.0798	0.1392	0.3
6 (64)	0.0416	0.0704	0.3
7 (128)	0.0206	0.0378	0.3
8 (256)	0.0099	0.0200	0.2
9 (512)	0.0075	0.0111	0.2
10 (1024)	0.0061	0.0061	0.1

if all of the slots in the frame are collided. Thus, the data are selected whose expected number of empty slots in (8) are less than 0.1 ($\bar{n_e} \geq 0.1$) so that the estimator work correctly even when only one slot is empty.

After selecting data, least square (LS) method in [9] is performed. The LS approach attempts to minimize the square of the distance from the data of perfect estimator for each frame-size. If we fix the current frame-size as some specific $\dot{Q}_1$ for m inventory rounds observations, we can easily determine the best k_1 and k_2 for the specific frame size so that the searched weights provide minimum squared-error between the estimation result by the perfect estimator and that by the DBQ-Algorithm. In this case, the $\mathbf{q}_1$ can be expressed as a vector with the same elements ($\dot{Q}_1$) because we fix L_1 and Q_1 to find the best biases for each frame size. By multiplying the pseudo-inverse of $\mathbf{H}$, we get

$$\begin{pmatrix} k_1 \\ -k_2 \end{pmatrix} = \left(\mathbf{H}^T \mathbf{H} \right)^{-1} \mathbf{H}^T \left(\mathbf{q}_2 - \mathbf{q}_1 \right). \tag{10}$$

The searched biases are listed in Table II. The biases are different for the frame size at the estimation time. In addition, they decrease as the frame size increases. To evaluate the performance compared to the existing Q-Algorithm, c values are also listed in the table. They are obtained by the computer search. Although c is less than 0.1 for larger than 32 frame sizes, the small weights do not make problem in the inventory round. In fact, the weights less than lower constraint of Q-Algorithm do not impact on the system performance unless the frame size is too small.

IV. NUMERICAL RESULTS

In this section, we consider a system equipped with a reader and multiple tags. The channel between the reader and the tags is considered to be ideal. It means that the error due to propagation delay, path loss, and noise is ignored. Thus, all signals from the tags are received with equal power to the reader. Here, we also assume that there are no tags that participate or go out during the inventory procedure. Identified tags are inactivated and do not attend to the next frames. The tested number of tags is increased from 10 to 500. The frame size is chosen among powers of 2 to reduce feedback information from a reader.

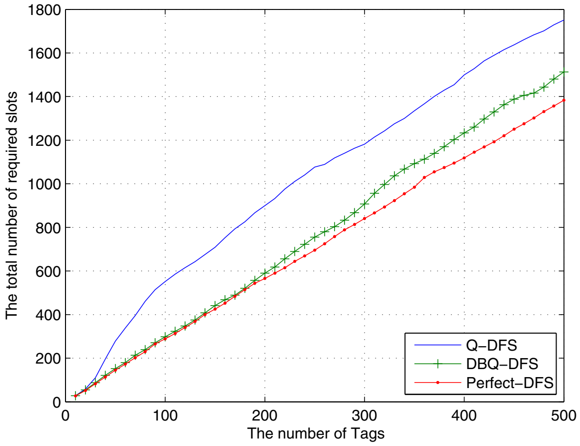

Fig. 6. The number of total required slots.

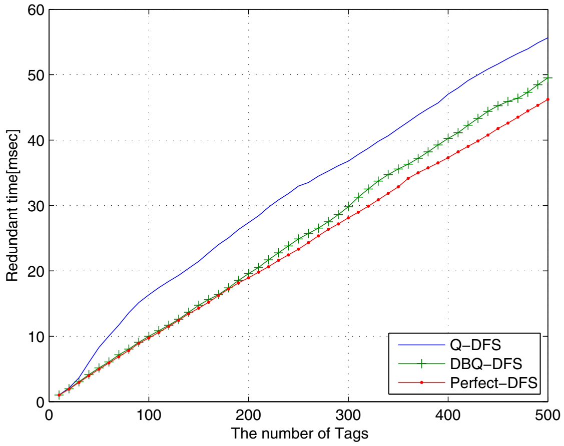

Fig. 7. Redundant time for DFS-Aloha.

Time of DFS-Aloha is measured based on Gen2 specification. Lengths of commands, temporary ID (RN16), and Q-Algorithm follow Gen2 in Fig. 1 and 2. The weights of Q-Algorithm are set to be the values listed in Table II. Both data rates of up and downlink are set to be 80 Kbps. Hence, the tag spends 0.2 ms (i.e., 16-bit time) for one slot time. Note that in the simulations, the initial frame sizes of all are fixed as 16.

Fig. 6 presents the averaged total slots to read all tags in DFS-Aloha with Q and DBQ-Algorithm. The number of slots used for DFS-Aloha is linearly increased as the number of tags increases. The proposed DBQ-Algorithm requires less slots to identify all tags. Moreover, it approaches to the performance of the perfect estimator. When the number of tags is 500, DBQ-Algorithm saves about 200 slots compared to Q-Algorithm. That is, DBQ-Algorithm exhibits performance improvement of about 80% for the given weights in Table II, compared to the existing Q-Algorithm.It shows that the proposed DBQ-Algorithm enhances the accuracy of frame-size estimator without any complicate computations.

Fig. 7 shows the redundant time of Gen2 protocol. It means the time to communicate between reader and tags except time for transmitting the real data (i.e. 96-bit or 256-bit EPC). Because the time for transmitting EPC is fixed for the given number of tags, the redundant time shows the information overhead of DFS-Aloha. Hence, the result does not depend on the length of EPC. As shown in Fig. 7, DBQ-Algorithm reduces the redundant time of DFS-Aloha with the same pattern as Fig 6.

V. CONCLUSION

Q-Algorithm introduced in Gen2 protocol is very simple frame-size estimator. It does not need complicate computations and large memory capacity. In Q-Algorithm, weight c plays an important role for its identification speed. However, there have been no systematic approaches to find its optimum values. As a result, Q-algorithm shows irregular performance.

In this paper, we have proposed DBQ-Algorithm that enhances the accuracy of frame-size estimator in DFS-Aloha. The estimator estimates the exponent of next frame size by assigning different weights for the number of collided and empty slots for the current frame. It is simple estimator but its performance is influenced by its weight like Q-Algorithm.

To find the best weights for DBQ-Algorithm, the samples of perfect estimator are searched. And, the expected collision patterns for the given samples are driven by the statistic equations. Then, the weights are found by least square method. By comparing Q-Algorithm with DBQ-Algorithm based on Gen2 protocol, we show that the proposed estimator can enhance the identification time of RFID systems with minimum modification of the existing estimator.

If desired estimation data of perfect estimator are given, the desired data will always give us the best weights for the given frame sizes. Thus, we believe that the algorithm we proposed is implemented not only Gen2 protocol whose frame sizes are restricted as powers of 2 but also any other DFS-Aloha schemes that continuous frame size is available as its frame size.

ACKNOWLEDGMENT

This work is supported by the ”Development of Sensor tag and Sensor node technology for RFID/USN” project of ETRI through IT Leading R&D Support Programs of MIC (Ministry of Information and Communication), Korea.

REFERENCES

[1] K. Finkenzeller and R. Waddington, *RFID Handbook: Radio-Frequency Identification Fundamentals and Applications*, John Wiley & Sons, January 2000.

[2] H. Vogt, “Multiple object identification with passive RFID tags,” *2002 IEEE International Conference on Systems, Man and Cybernetics*, Oct. 2002.

[3] F. C. Schoute, “Dynamic frame length ALOHA,” *IEEE Transactions on Communications*, COM-31(4):565-568, April 1983.

[4] EPCglobal, EPC. radio-frequency identity protocols class-1 generation-2 UHF rfid protocol for communications at 860MHz-960MHz version 1.0.9., http:\\www.epcglobalinc.org\standards_technology.

[5] J. Park, M. Y. Chung, and T. Lee, “Identification of RFID tags in framed-slotted Aloha with robust estimation and binary selection,” *IEEE Commun. Lett.*, vol. 11, no. 5, pp 452-454, May 2007.

[6] S. R. Lee, S. D. Joo, and C.W. Lee, “An Enhanced Dynamic Framed Slotted ALOHA Algorithm for RFID Tag Identification,” *Proceedings of IEEEMobiQuitous05,* 2005.

[7] C. Floerkemeier, “Bayesian transmission strategy for framed ALOHA based RFID protocols ,” *2007 IEEE International Conference on RFID*, March 2007.

[8] C. Floerkemeier and M. Wille, “Comparison of transmission schemes for framed ALOHA based RFID protocols,” *Proc. of Inter. SAINT2006 IPv6 Workshops*, November 2005.

[9] Steven M. Kay,*Fundamentals of Statistical Singal Processing Volume 1 Estimtion Theory*, Prentice Hall, 1993.

[10] Donghwan Lee, Kyungkyu Kim, Wonjun Lee, “ Q+-Algorithm : An Enhanced RFID Tag Collision Arbitration Algorithm,” *Ubiquitous Intelligence and Computing: Lecture Notes in Computer Science*, vol.4611/2007, pp.23-32, Aug. 2007.