Optimal Framed Aloha Based Anti-Collision Algorithms for RFID Systems

Lei Zhu, *Student Member, IEEE*, and Tak-Shing Peter Yum, *Senior Member, IEEE*

***Abstract*—The anti-collision algorithm is an important part of the Radio-Frequency Identification (RFID) system. Of the various possible algorithms, the Framed Aloha based (FA) algorithms have been most widely used due to their simplicity and robustness. Previous studies have focused mainly on the tag population estimation, choosing the frame size based on the classical results of Random Access (RA) systems. We show that a new theory is needed for algorithm design for RFID systems, because RFID and RA systems are fundamentally different. The Philips RFID system is studied in this paper. We model the reading process as a Markov Chain and derive the optimal reading strategy by first-passage-time analysis. The optimal frame sizes are derived analytically and numerically.**

***Index Terms*—RFID anti-collision algorithms, framed Aloha, optimization.**

I. Introduction

In the past few years, Radio-Frequency identification (RFID) tags have found more and more everyday applications, ranging from inventory and tracking to electronic tickets and keys. The capability of tags varies widely, depending on the application. Active tags embedded with power supply and their own CPUs can process data and initiate transmission. Passive tags (including Class 1 and Class 2 tags in EPCglobal standards [2]), however, have only bare-bone functionality and no embedded power supply. Some passive tags can merely transmit a particular bit-string when probed by a reader. But they have the advantage of being relatively cheap, and for this reason passive tags are gradually replacing barcode tags in ever-widening range of applications.

In RFID systems, tags share a common communication channel. Therefore, if multiple tags transmit at the same time, their packets will collide and get lost [1]. Since passive tags cannot sense the media or cooperate with one another, the RFID reader needs to coordinate their transmissions to avoid collisions. Depending on the working principles used, previous-published RFID anti-collision algorithms can be divided into three main classes: Tree based algorithms [5][6], Framed Aloha (FA) based algorithms [8-16] and Interval based algorithms [7]. Of these three classes, only FA based algorithms are widely used in RFID communication standards [2-4], because of their simplicity and robustness.

Paper approved by R. Fantacci, the Editor for Wireless Networks and Systems of the IEEE Communications Society. Manuscript received July 14, 2009; revised November 12, 2009 and January 10, 2010.

L. Zhu is with the Department of Information Engineering, the Chinese University of Hong Kong (CUHK), Hong Kong, China (e-mail: lzhu6@ie.cuhk.edu.hk).

T.S.P. Yum is with the Hong Kong Applied Science and Technology Research Institute (ASTRI), Hong Kong, China (e-mail: peteryum@astri.org).

Digital Object Identifier 10.1109/TCOMM.2011.102910.090390

Although variations on the working mechanism of FA based RFID systems have been reported in previous studies, they all share some basic properties. In this section we use the Philips system [4] to introduce these properties. We focus only on the anti-collision part as the complete communication process is much more complicated.

1) The RFID Reader starts a frame by broadcasting the *"begin-round"* command with an integer parameter 'L'. Upon hearing this command, unsilenced tags generate a random number from 0 to $L-1$. Those generating '0' reply immediately.
2) If only one tag replies, the reader can identify it and send back the *"fix-slot"* command. On hearing this command, the replied tag will be *silenced*, i.e. will not respond to future commands, while the other tags decrease their counter values by 1 and reply if the counters reach 0.
3) If multiple tags reply or no tag replies, the reader will send back the *"close-slot"* command. On hearing this command, all the unsilenced tags will decrease their counters by 1 and reply if their counters reach 0. The collided tags will be available for reading in another frame.
4) After one frame ends, the Reader will begin a new frame by broadcasting the *"begin-round"* command, and this cycle will continue until no collision is detected.

This kind of RFID system is usually called the **Basic FA-based RFID system**, or the Philips system. It is important to differentiate between an 'RFID system' and an 'RFID algorithm'. The former is usually designed in standards or product manuals. It specifies the reader's command set and the tag's reply function. The latter is designed only for the RFID reader. It tells the reader when and how to use the commands to achieve efficiency. Besides these basic operations in the Philips system, some advanced RFID systems incorporate other commands, such as the *'split'* command in [8] (asking the collided tags to back off 1 or 2 slots to reply again) and the *'Frame-size adjust'* command in the EPCglboal system [13][14][15] (terminating the current frame and letting all the unsilenced tags regenerate their counter values according to a new L). These advanced commands can improve performance, but at the cost of greater system complexity.

As mentioned earlier, the communication time between the tags and the reader is slotted. Conventionally, the time from the point that the reader sends out a command to the point that the tags finish replying their information is called a *time slot* (*slot* for short). The time taken by the reader to identify a group of tags is conventionally called the *reading time*, and is measured in slots. In Basic FA-based RFID systems, the goal of algorithm design is to find the optimal L that can maximize the reading efficiency η defined as

$$\eta = \frac{\text{The number of tags}}{\text{The expected reading time}}.$$

Unfortunately, a function between η and L is not explicitly available. Previous studies [8-15] have therefore used the frame-local efficiency

$$U = \frac{\text{The number of tags identified in the current frame}}{\text{The current frame size}}$$

as a substitute for algorithm optimization. In the Random Access theory, a classical formula to calculate the frame-local efficiency U with terminal population N and frame size L is given in [10] as:

$$U(N, L) = \frac{N}{L}\left(1 - \frac{1}{L}\right)^{N-1}. \tag{1}$$

In (1), U can be maximized by setting the frame size equal to the terminal number, or $L = N$. On this basis, previous algorithms [8-15] either directly set the frame size L equal to the expected tag population $E[N]$ or try to find an optimal L to maximize U for a given probability distribution of N. However, these approaches are not suitable because maximizing U in every frame does not necessarily maximize overall efficiency. Overall efficiency is equal to frame-local efficiency only when all the frames are identical. Since the tag population decreases frame by frame as identified tags are silenced by the RFID reader, optimizing L based on (1) will only yield a frame-local optimal result. We will show in Section II.C that the concatenation of locally optimal results are usually significantly inferior.

In this paper, we propose a method for choosing the globally optimal frame size for the Philips system. We model the reading process as a Markov Chain and optimize the reading strategy through first-passage-time analysis. The optimal frame sizes can be obtained either analytically or via a recursive program. Simulation results demonstrate that the use of the optimal frame size delivers a better performance than previously-published algorithms. The methodology in this paper can be applied to the EPCglobal system. Some partial results are presented in [22].

In Section II, we survey these previously-published algorithms and expose an unjustified assumption used in previous attempts at reading strategy optimization. In Section III, a new model is proposed to derive the optimal reading strategy. In Section IV, we verify the optimal frame sizes by computer simulation and compare the performance of the optimal algorithm with its predecessors. In Section V, we provide an example of the application of the optimal reading strategy.

II. Survey of Previous Studies

In this section, we survey previous studies in this field and show why a new theory is needed.

In real applications, the number of tags is unknown before identification. A proper FA algorithm therefore always contains two parts: **Population Estimation** and **Frame Size Determination**. The first part is used for estimating the tag population based on tags' replies, while the second part is used for choosing the frame size using the estimation. Depending on which estimation methods are used, algorithms can be divided into the max-likelihood approach and the probability distribution approach.

A. The Max-likelihood Approach

Schoute [10] noticed that when N is large and L suitably chosen (say $L \approx N$), the number of tags contending each slot has a Poisson distribution with mean 1. So in the *Population Estimation* part, his algorithm uses $\hat{N} = \text{round}(2.39 s_c)$,[1] where s_c is the number of collided slots in the last frame. On this basis, in the *Frame Size Determination* part, the frame size is set as $L = \hat{N}$. This choice is obviously based on (1). It tries to maximize the instantaneous throughput by setting the frame size equal to the expected terminal number.

Vogt [11] improved the *Population Estimation* strategy of Schoute's algorithm by also using the statistics of empty slots s_e and singleton slots s_s. Tag population is estimated to be the value $\hat{N}$ that minimizes the error between the observed values of s_e, s_s, s_c and their expected values using $\hat{N}$. In the *Frame Size Determination* part, it also uses $L = \hat{N}$.

Kodialam [16] proposed a new *Population Estimation* strategy based on the Central Limit Theorem. That is when the number of contending tags is large enough, the number of collision slots and empty slots in a frame should obey the Normal distribution. This method makes it possible to obtain the estimation accuracy as well as the max-likelihood tag population. But after deriving $\hat{N}$, it also sets $L = \hat{N}$.

Another example is the Q algorithm in EPCglobal standards [2]. The reader maintains a floating-point variable Q_{fp}. It decreases a typical value C when no tag replies, increases C when multiple tags reply, and stays unchanged when only 1 tag replies.[2] The tag population is estimated as $\text{round}(2^{Q_{fp}})$ while the frame size is set to 2^Q, where $Q = \text{round}(Q_{fp})$. In [13][14], the efficiency of the Q algorithm was obtained with different choices of C and Q_{fp} and a number of suggestion were made for improving the estimation strategy.

In summary, algorithms of this type compute the maximum-likelihood tag population $\hat{N}$ based on the reading results and set $L = \hat{N}$ as the frame size. They may adopt different approaches in the *Population Estimation* part, but follow the same strategy in the *Frame Size Determination* part.

B. Probability Distribution Approach

Floerkemeier [12][15] assumes that a rough estimation of the target group size is always available in the form of a distribution $\Pr\{N = n\}$. As a new *Population Estimation* strategy, it updates the population distribution by Bayesian method at the end of every frame. Based on this distribution, the *Frame Size* is chosen as

$$L^* = \arg\max_{L \in \Upsilon} E[U] = \arg\max_{L \in \Upsilon} \sum_{n=0}^{N_{max}} U(N = n, L) \Pr(N = n), \tag{2}$$

where Υ is the set of possible frame sizes while N_{max} is a practical limit of the tag population.

[1]For the first frame where s_c is not available, $\hat{N}$ is obtained from the initial estimation of the tag population.

[2]In EPCglobe standards, it is recommended that $0.2 \leq C \leq 0.5$ and the initial $Q_{fp} = 4$

This approach can track the value of N more accurately. Since a random variable N is completely specified by its distribution, and Bayesian method ensures no information loss in estimation, the *Population Estimation* part of Floerkemeier's algorithm is flawless, but the use of (1) in the *Reading Strategy Determination* part is still unwise. We explain why in the following section.

C. The Need for a New Theory

Previous studies have focused on the *Population Estimation* and have suggested a number of ways of obtaining a more accurate estimation. For the *Frame Size Determination* part, these studies all assumed that optimal performance can be achieved by maximizing the frame-local efficiency U. As we mentioned before, (1) is obtained from the theory of RA system. Since a terminal in RA systems would still attempt the channel after a successful transmission, the 'contending group' can be assumed unchange during a long enough period. The long-term efficiency of a RA system is therefore equal to the expected frame-local efficiency U calculated by (1). However, in RFID systems, identified tags are silenced by the reader, so that the tag population decreases during the reading process. When the frames are not identical, a concatenation of locally optimal solutions is not globally optimal. Suppose, for example, the target group size is distributed as

$$\Pr\{N=n\} = \begin{cases} 0.99 & , \quad n=0 \\ 0.01 & , \quad n=10 \end{cases}$$

From (2), the suitable frame size should be $L = 10$, as it can maximize the throughput of the current frame. However, since this group is very likely empty, it is better to use $L = 1$ to check whether it contains tags or not, even though the throughput of this checking frame is 0.

III. Reading Strategy Optimization

We now present our optimal strategy for *Frame Size Determination*. In this paper, we derive the optimal strategy only for the Basic FA based RFID system. The application of this theory to the EPCglobal system is in [22].

A. The Optimal Strategy

To choose a suitable frame size L, the reader needs the information of the target group size. As discussed in Section II, this information can be fully described by a probability distribution. In applications, a rough distribution is often available as the reader has information of its previous readings. In the worst case where N is completely unknown, a uniform distribution on $[0, N_{max}]$ can be assumed as we cannot favor any value over the others. The value of this N_{max} is determined by the nature of the application concerned. For example, for a typical supermarket shopping cart, N_{max} can be safely set to 1000.

During the reading process, let $Bel(N)$ denote the *belief* of N, or the conditional distribution of N based on all available information [17]. To simplify the notation, let $v_n = Bel(N = n)$ and $\mathbf{v} = (v_0, v_1, \ldots, v_{max})$. Obviously the accuracy of the belief affects the reading efficiency. Let $T(n \mid \mathbf{v})$ denote the average reading time, measured by slots, for these n tags when the initial belief is $\mathbf{v}$. The expected reading time for a group with population distribution $\mathbf{v}$ is then

$$\mathcal{T}(\mathbf{v}) = \sum_{n=0}^{N_{max}} v_n T(n \mid \mathbf{v}).$$

Our goal is to find the optimal frame size L^* that can maximize the overall efficiency for any given distribution $\mathbf{v}$, or

$$L^* = \arg\max_{L\in\Upsilon} \eta = \arg\max_{L\in\Upsilon} \left\{ \frac{E[N]}{\mathcal{T}(\mathbf{v})} \right\}. \tag{3}$$

Since $E[N] = \sum_{n=1}^{N_{max}} n v_n$ independent of the frame size L, (3) becomes

$$L^* = \arg\min_{L\in\Upsilon} \{\mathcal{T}(\mathbf{v})\}. \tag{4}$$

It should be noted that (4) is different from (2), as it is designed to maximize the overall efficiency instead of the frame-local efficiency. Thus L^* is the global optimal frame size. To find it, however, requires the function of $\mathcal{T}(\mathbf{v})$. We now show how to derive this function from the reading mechanism of basic FA based RFID systems.

In an intelligent system, the optimal decision only depends on the current information, or the *belief* of all the relevant variables [17]. When applied to RFID systems, the optimal frame size only depends on $Bel(N)$. We let $V_j = Bel(N_j) = (v_0, v_1, \ldots, v_{max})$ denote the state of the reading process at the end of frame j, where N_j is the unresolved tag population at the end of frame j. Since identified tags are silenced by the reader, we always have $N_j \geq N_{j+1}$. Further, let

$$\mathscr{V} = \left\{ (v_0, v_1, \ldots, v_{max}) \,\middle|\, v_i \geq 0, \ \sum_{i=0}^{N_{max}} v_i = 1 \right\}$$

denote the set of all possible states. For a group with the initial estimation $\Pr(N)$, let $V_0 = \Pr(N)$ be the initial state and $V_T = (1, 0, 0, \ldots, 0)$ be the terminating state.

Theorem 1: Following a distribution-based anti-collision algorithm, the reading process $V_0V_1V_2\ldots V_T$ is a Markov Chain.

Proof: Let $V_j = (v_0, v_1, \ldots, v_{max}) \in \mathscr{V}$ be the current state. For a distribution-based algorithm, the next frame size l should be fixed given V_j. Let $V_{j+1} = (u_0, u_1, \ldots, u_{max})$ be the belief of tag population at the end of frame $j+1$. Obviously V_{j+1} depends on the reading results of frame $j+1$ as well as the previous beliefs.

In frame $j+1$, let random variable S_0, S_1, S_c denote the number of empty slots, singleton slots and collided slots. Since $S_0 + S_1 + S_c = l$ and $S_0, S_1, S_C \geq 0$, there are at most $\binom{l+2}{2} = \frac{1}{2}(l+1)(l+2)$ different outcomes in frame $j+1$. Thus for a given frame size, there are at most $\frac{1}{2}(l+1)(l+2)$ different choices of V_{j+1} that satisfy $\Pr(V_{j+1} \mid V_j V_{j-1} \ldots V_0) > 0$.

In every frame, tags randomly choose their transmission delay. Using the urn problem terminology, the probability that s_1 urns (slots) contain only 1 ball (tag), s_c urns contain more than 1 balls and the others are empty can be obtained in [18] as:

$$\Pr\{S_c = s_c, S_1 = s_1 \mid N_j = n, L = l\} = \tag{5}$$
$$\binom{l}{s_0, s_1, s_c} \frac{n!}{(n-s_1)!\,l^n} \sum_{\substack{m_1, m_2, \ldots, m_{sc} \geq 2, \\ m_1+m_2+\cdots+m_{sc}=n-s_1}} \binom{n-s_1}{m_1, m_2, \ldots, m_{sc}},$$

where $m_1, m_2, \ldots, m_{sc}$ denote the number of tags in each of the s_c collided slots. Further, we can substitute the belief of N_j to obtain

$$\Pr\{S_c = s_c, S_1 = s_1 \mid L = l\} = \sum_{n=0}^{N_{max}} v_n \Pr\{S_c = s_c, S_1 = s_1 \mid N_j = n, L = l\}. \quad (6)$$

At the end of frame $j+1$, we can obtain the values of s_0, s_1 and s_c. By Bayes formula, the posterior distribution of N_j can be updated as.:

$$\begin{aligned} v'_i &= \Pr\{N_j = i \mid S_c = s_c, S_1 = s_1, L = l\} \\ &= \frac{\Pr\{S_c = s_c, S_1 = s_1 \mid N_j = i, L = l\}\, v_i}{\Pr\{S_c = s_c, S_1 = s_1 \mid L = l\}} \end{aligned} \quad (7)$$

As tags in the singleton slots are successfully identified and silenced, we have $N_{j+1} = N_j - s_1$ with distribution given as

$$\begin{aligned} u_i &= \Pr\{N_{j+1} = i \mid S_c = s_c, S_1 = s_1, L = l\} \\ &= \Pr\{N_j = i + s_1 \mid S_c = s_c, S_1 = s_1, L = l\} \\ &= v'_{i+s_1}, \qquad i = 0, 1, 2, \ldots \end{aligned} \quad (8)$$

Note the transition probability from state $V_j = (v_0, v_1, \ldots, v_{max})$ to $V_{j+1} = (u_0, u_1, \ldots, u_{max}) = \Pr\{N_{j+1} \mid S_c = s_c, S_1 = s_1, L = l\}$ is just $\Pr\{S_c = s_c, S_1 = s_1 \mid L = l\}$. As specified in (5) and (6), this transition probability does not depend on states $V_{j-1}V_{j-2}\ldots V_0$. The states $V_0V_1V_2\ldots V_T$ then form a Markov Chain. □

For a given state V_j, let $\mathcal{S}(V_j, l) \subset \mathscr{V}$ denote the set of possible V_{j+1}, or

$$\mathcal{S}(V_j, l) = \Big\{V_{j+1} \in \mathscr{V} : \Pr(V_{j+1} \mid V_j, L = l) > 0\Big\}.$$

As shown in Theorem 1, $|\mathcal{S}(V_j, l)| \le \frac{1}{2}(l+1)(l+2)$. Let $\mathcal{T}_o(V_j)$ denote the expected first passage time from V_j to V_T using the *optimal reading strategy*, or

$$\mathcal{T}_o(V_j) = \min_{L \in \Upsilon} \mathcal{T}(V_j).$$

According to the Markov Chain theory [19], we have

$$\begin{aligned} &\mathcal{T}_o(V_j) \\ &= \text{The reading time for the next frame} + \text{The expected remaining reading time after the next frame} \\ &= \text{The optimal frame size for the next frame} \\ &\quad + \sum_{V_{j+1} \in \mathscr{V}} \Pr(\text{the process moves to } V_{j+1}) \\ &\quad * \big(\text{The expected first passage time from } V_{j+1} \text{ to } V_T\big) \\ &= L^* + \sum_{V_{j+1} \in \mathcal{S}(V_j, L^*)} \Pr(V_{j+1} \mid V_j, L^*)\mathcal{T}_o(V_{j+1}) \\ &= \min_l \Bigg\{ l + \sum_{V_{j+1} \in \mathcal{S}(V_j, l)} \Pr(V_{j+1} \mid V_j, l)\mathcal{T}_o(V_{j+1}) \Bigg\} \end{aligned} \quad (9)$$

In (9), the state $V_{j+1} \in \mathcal{S}(V_j, l)$ and the transition probability $\Pr(V_{j+1} \mid V_j, l)$ are given by (8) and (6).

This formulation is often called the Adaptive Markov Decision Process [20]. We now show how to solve (9) analytically and numerically.

B. The Analytical Solution of L^*

In this section, we show how to solve (9) by some examples. Since (9) is a recursive function, we begin from small N_{max} cases.

Case 1: $N_{max} = 2$

In this case, the tag population can only be 0, 1 or 2. Given $V_j = \mathbf{v} = (v_0, v_1, v_2)$, the probability of different outcomes of frame $j+1$ can be obtained from (6) as:

$$\begin{aligned} &\Pr\{S_c = 0, S_1 = 0 | L = l\} = v_0; \\ &\Pr\{S_c = 0, S_1 = 1 | L = l\} = v_1; \\ &\Pr\{S_c = 0, S_1 = 2 | L = l\} = \tfrac{l-1}{l} v_2; \\ &\Pr\{S_c = 1, S_1 = 0 | L = l\} = \tfrac{1}{l} v_2. \end{aligned}$$

From (7) and (8), we get the belief of N_{j+1} as

$$\begin{aligned} &\Pr\{N_{j+1} = 0 | S_c = 0, S_1 = s_1, L = l\} = 1; \\ &\Pr\{N_{j+1} = 2 | S_c = 1, S_1 = 0,\ L = l\} = 1. \end{aligned}$$

Thus V_{j+1} has only two possible choices independent with the value of l as

$$\mathcal{S}(V_j, l) = \Big\{(1,0,0),\, (0,0,1)\Big\}$$

and the transition probability is

$$\begin{aligned} \Pr\Big((1,0,0)\,\Big|\,(v_0, v_1, v_2),\, l\Big) &= \frac{l-1}{l} v_2 + v_1 + v_0 \\ \Pr\Big((0,0,1)\,\Big|\,(v_0, v_1, v_2),\, l\Big) &= \frac{1}{l} v_2 \end{aligned}$$

Substituting them into (9), we have

$$\begin{aligned} \mathcal{T}_o(\mathbf{v}) &= \min_l \Bigg\{ l + \frac{1}{l} v_2\, \mathcal{T}_o\Big((0,0,1)\Big) \\ &\qquad + \Big(\frac{l-1}{l} v_2 + v_1 + v_0\Big)\mathcal{T}_o\Big((1,0,0)\Big) \Bigg\} \\ &= \min_l \Big\{ l + \frac{4v_2}{l} \Big\}, \end{aligned} \quad (10)$$

where $\mathcal{T}_o\Big((1,0,0)\Big) = 0$ as $(1,0,0)$ is the terminating state while $\mathcal{T}_o\Big((0,0,1)\Big) = 4$ is the expected contention time for a group with exactly 2 tags, which can be obtained in *'Case 3'*. Solving (10) yields:

$$\begin{aligned} L^* &= \begin{cases} 1 & , \quad v_2 < \frac{1}{2} \\ 2 & , \quad v_2 \ge \frac{1}{2} \end{cases} \\ \mathcal{T}_o(\mathbf{v}) &= \begin{cases} 1 + 4v_2 & , \quad v_2 < \frac{1}{2} \\ 2 + 2v_2 & , \quad v_2 \ge \frac{1}{2} \end{cases} \end{aligned}$$

To compare, we derived the frame size and average contention time of Floerkemeier's algorithm from (3) as

$$\begin{aligned} L &= \begin{cases} 1 & , \quad v_1 > v_2 \\ 2 & , \quad v_1 \le v_2 \end{cases} \\ \mathcal{T}_f(\mathbf{v}) &= \begin{cases} 1 + 4v_2 & , \quad v_1 > v_2 \\ 2 + 2v_2 & , \quad v_1 \le v_2 \end{cases} \end{aligned}$$

Therefore, when the distribution $\mathbf{v}$ satisfies $v_1 < v_2 < 0.5$, we have $\mathcal{T}_f(\mathbf{v}) > \mathcal{T}_o(\mathbf{v})$, or Floerkemeier's strategy is not optimal.

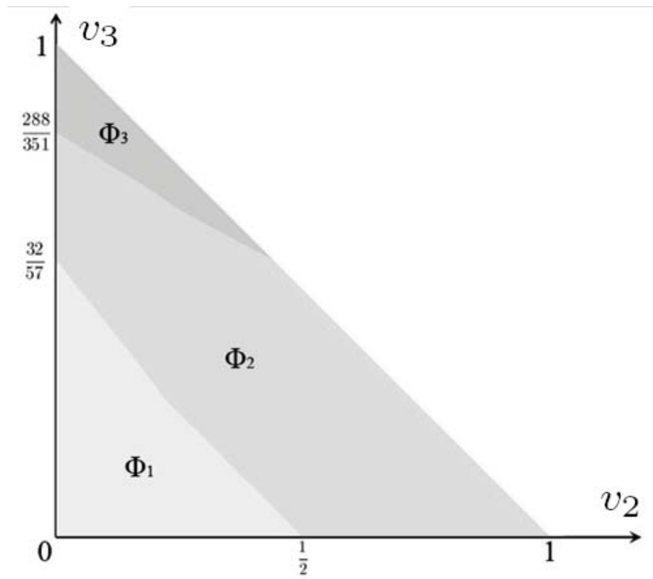

Fig. 1. The solution set for $N_{max} = 3$ case.

Case 2: $N_{max} = 3$

Next, we move on to $N_{max} = 3$ and derive the recursive function as

$$\mathcal{T}_o(\mathbf{v}) = \min_l \left\{ l + \frac{12v_3(l-1)}{l^2} + (\frac{v_2}{l} + \frac{v_3}{l^2})\mathcal{T}_o(\mathbf{u}) \right\}, \quad (11)$$

where $\mathbf{u}$ is the distribution of N_{j+1} on condition that $S_c = 1$ and $S_1 = 0$ in frame $j+1$, or

$$\mathbf{u} = (u_0, u_1, u_2, u_3) = \left(0, 0, \frac{lv_2}{v_3 + lv_2}, \frac{v_3}{v_3 + lv_2}\right).$$

Solving (11), the optimal reading strategy can be similarly obtained as:

$$L^* = \begin{cases} 1 & , \ \mathbf{v} \in \Phi_1 \\ 2 & , \ \mathbf{v} \in \Phi_2 \\ 3 & , \ \mathbf{v} \in \Phi_3 \end{cases}$$

$$\mathcal{T}_o(\mathbf{v}) = \begin{cases} 1 + (v_2 + v_3) f\left(\frac{v_3}{v_2+v_3}\right) & , \ \mathbf{v} \in \Phi_1 \\ 2 + 3v_3 + \frac{1}{4}(2v_2 + v_3) f\left(\frac{v_3}{2v_2+v_3}\right) & , \ \mathbf{v} \in \Phi_2 \\ 3 + \frac{8}{3}v_3 + \frac{1}{9}(3v_2 + v_3) f\left(\frac{v_3}{3v_2+v_3}\right) & , \ \mathbf{v} \in \Phi_3 \end{cases}$$

where $f(x)$ is a recursive function as

$$f(x) = \begin{cases} \frac{8}{3}x + 4 & , \ 0 < x \leq \frac{9}{16} \\ 3 + \frac{8}{3}x + \frac{1}{9}(3 - 2x) f\left(\frac{x}{3-2x}\right) & , \ \frac{9}{16} < x < 1 \end{cases}$$

while Φ_1, Φ_2 and Φ_3 are distribution regions specified in Fig. 1.

For $N_{max} > 3$ cases, the optimal strategies can be obtained similarly, though the computation becomes more complex.

Case 3: Tag Population N known

We show this as a special case, because the result of this case will be used in other tag population unknown cases. As an example, to solve (10), we used $\mathcal{T}_o\Big((0,0,1)\Big) = 4$, which is a result of N known case.

Let $\mathcal{F}(n)$ denote the average reading time using the optimal strategy for the tag population known case, or $\mathcal{F}(n) = \mathcal{T}_o(\mathbf{v})$ when $v_n = 1$. After one frame of reading, the population decreases to $n' = n - s_1$, where s_1 is the number of singleton slots. So the remaining reading time is $\mathcal{F}(n - s_1)$. Substituting $\mathcal{F}(n)$ and $\mathcal{F}(n - s_1)$ into (9), we have

$$\mathcal{F}(n) = \min_l \left\{ l + \sum_{s_1=0}^{l} \sum_{s_c=0}^{l-s_1} \Pr\{S_c = s_c, S_1 = s_1 \mid l\} * \mathcal{F}(n - s_1) \right\}.$$

The explicit form of the system equation after rearranging is

$$\mathcal{F}(n) = \min_l \left\{ \frac{l + \sum_{s_1=1}^{l} \sum_{s_c=0}^{l-s_1} \Pr\{S_c = s_c, S_1 = s_1 \mid l\} \mathcal{F}(n - s_1)}{1 - \sum_{s_c=1}^{l} \Pr\{S_c = s_c, S_1 = 0 \mid l\}} \right\}. \quad (12)$$

Solving (12), the optimal frame size is obtained as $L^* = n$ while the average reading time is

$$\mathcal{F}(n) = \frac{n + \sum_{s_1=1}^{n} \sum_{s_c=0}^{n-s_1} \Pr\{S_c = s_c, S_1 = s_1 \mid L = n\} \mathcal{F}(n - s_1)}{1 - \sum_{s_c=1}^{n} \Pr\{S_c = s_c, S_1 = 0 \mid L = n\}}. \quad (13)$$

The algorithm efficiency $\eta = \frac{n}{\mathcal{F}(n)}$ is shown in Fig. 2. We can see that η is always above the efficiency upper bound of Random Access systems [10]. This is also a proof that formula (1) is not suitable for RFID systems.

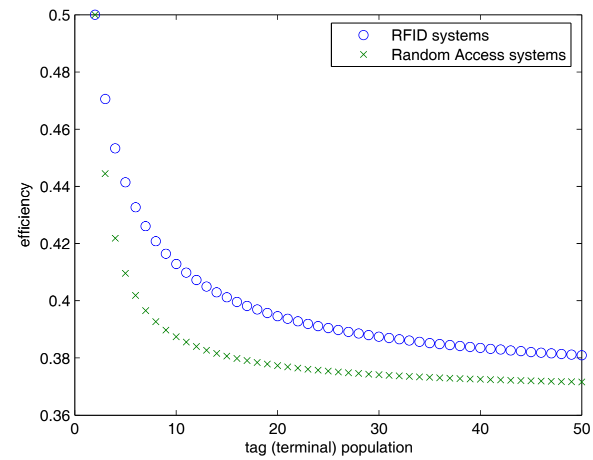

Fig. 2. The efficiency bound for Basic FA based systems and Random Access systems when N is known.

C. The Numerical Solution of L^*

In [20][21], a general method is proposed to solve the optimal strategy in an Adaptive Markov Decision Process by running an Iterative Program. Following the method in [21], we show an outline of the solution as follows:

For a group of tags with initial estimation V_0, the average reading time and optimal frame size can be derived from (9) as

$$\mathcal{T}_o(V_0) = \min_l \left\{ l + \sum_{V_1 \in \mathcal{S}(V_0,\, l)} \Pr(V_1 \,|\, V_0,\, l)\mathcal{T}_o(V_1) \right\},$$
$$L^* = \arg\min_l \left\{ l + \sum_{V_1 \in \mathcal{S}(V_0,\, l)} \Pr(V_1 \,|\, V_0,\, l)\mathcal{T}_o(V_1) \right\}.$$

Thus given the values of $\{\mathcal{T}_o(V_1)\}$, where $V_1 \in \mathcal{S}(V_0, l)$, both $\mathcal{T}_o(V_0)$ and L^* can be solved numerically. If V_1 is the terminating state, we can simply use $\mathcal{T}_o(V_1) = 0$. Otherwise, $\mathcal{T}_o(V_1)$ should be similarly computed from the values of $\{\mathcal{T}_o(V_2)\}$ where $V_2 \in \mathcal{S}(V_1, l)$. The process continues until the terminating state V_T is reached. This program is guaranteed to converge, because in a transient Markov Chain,

$$\lim_{j\to\infty} \Pr(V_j = V_T) = 1.$$

The computation complexity is proportional to the number of states. For a particular state V_j, the number of states it will visit after one frame is $|\mathcal{S}(V_j, l)| \leq \frac{1}{2}(l+1)(l+2) \sim N^2$. In other words, the number of states increases with $\mathcal{O}(N^{2\alpha})$, where α is the number of frames used to identify all the tags. As derived in [10], the number of frames $\alpha \sim \mathcal{O}(\log N)$. In sum, therefore, the computation complexity is $\mathcal{O}(N^{\log N})$. This approach is prohibitive when N is large, so we go on to introduce a more efficient and accurate approximation method for the solution based on theorem 2. The proof is in Appendix I.

Theorem 2: For a group of tags with population distribution $\Pr(N) = \mathbf{v} = (v_0, v_1, v_2, \ldots, v_{max})$, variance $\mathrm{Var}(N) \to 0$ and expectation $E[N] \to k$, its average reading time satisfies

$$\mathcal{T}_o(\mathbf{v}) = \sum_{j=0}^{N_{max}} v_j\, \mathcal{G}_{k-j}(j), \tag{14}$$

where $\mathcal{G}_m(n)$ is defined in (15) and $L = m + n$, $X = \min(L - S_1, \frac{n-S_1}{2})$.

Based on Theorem 2, we can stop the iteration and calculate the approximate value of $\mathcal{T}_o(\mathbf{v})$ by (14) as soon as $\mathrm{Var}(N) < \delta$, where δ is a small enough value. After each frame, the feedbacks from the tags help the reader to update the estimation and hence decrease the estimation variance. According to [16], if the first frame size is appropriately chosen, the reader can estimate the tag population within two or three frames with an accuracy greater than 99.95%. In other words, $\mathrm{Var}(N)$ will drop to 0.6 for a group of about 100 tags within a few of frames. Thus the computation complexity is reduced to $\mathcal{O}(N^\beta)$, where $\beta = \mathcal{O}(1)$. The error bound of this approximation is depicted in Fig. 3 for different variance bound. As it shows, the error bound is nearly a constant as tag population increases. For $E[N] = 20$ and $\mathrm{Var}(N) = 0.6$, the error bound is around 0.25, only 0.5% error compared with $\mathcal{F}(20)$. The derivation of this error bound is in Appendix II.

A computer program based on the theorem discussed above is shown in Fig. 4. The input is a vector $\mathbf{v}$ representing the distribution of N. The program first checks its variance. If it is smaller than a threshold δ, the average reading time is calculated from (14); if not, it is solved numerically. Our experiment shows that the stopping condition will be fulfilled after several frames.

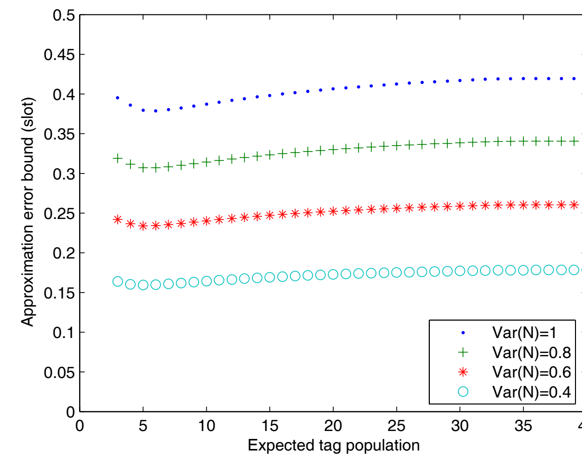

Fig. 3. The approximation error bound for different variance.

```
Function [T_v, L_v]=OptimalAloha(v)                 // realize(9)

if (var(N) < δ)                            // the stopping condition
    T_v = T(v);                                      // using (14)
    L_v = E[N];
    return;
end if

l =round(E[N]); T_v = ∞;
for L_t = l − Δl : l + Δl                       // numerically find L*
    T_t =0;
    for s_1 = 0 : L_t
        for s_e = 1 : L_t − s_1
            Calculate Pr{s_e, s_1|L_t} from (6);
            Calculate u from (7) and (8);
            [T_u, L_u]=OptimalAloha(u);
            T_t = T_t + Pr{s_e, s_1|L_t}T_u;
        end for
    end for
    if(T_v > T_t)
        T_v = T_t; L_v = L_t;
    end if
end for
```

Fig. 4. The iterative program to compute the optimal frame size.

IV. Performance Comparison

To compare the performance of various algorithms in different cases, we need a class of population distribution whereby the mean and variance of N can be varied. In our computer simulation, we choose

$$Pr\{N = n\} = \begin{cases} \frac{1}{Z}\left(|n-\beta| + \beta\right)^\alpha & , \quad 0 \leq n \leq 2\beta, \\ 0 & , \quad \text{others} \end{cases} \tag{16}$$

where Z is the normalization constant, α and β are parameters. It can be shown that $E[N] = \beta$ independent of α while the coefficient of variation c_v can be changed from 0 to the maximum value of 1 by setting different values of α. The uniform distribution is a special case of (16) when $\alpha = 0$. Note that this distribution is chosen for convenience. The Poisson distribution and Binomial distribution are not favored because once the mean is given, the variance is fixed.

We first set $E[N] = 10$ and change c_v from 0.3 to 1. Fig. 5 shows the choices of the first frame size by Schoute's,

$$G_m(n) = \begin{cases} \dfrac{L + \sum_{S_1=1}^{L} \sum_{S_c=0}^{X} \Pr\{S_c, S_1|L\} * \mathcal{G}_m(n-S_1) + \sum_{k=0}^{m+1} \sum_{S_1=0}^{n+m} \Pr\{S_c = \frac{n-S_1+k}{2}, S_1|L\} * \mathcal{G}_k(n-S_1)}{1 - \sum_{s_c=1}^{X} \Pr\{S_c = s_c, S_1 = 0 \mid L\}}, & m < 0 \\ \dfrac{L + \sum_{S_1=1}^{L} \sum_{S_c=0}^{L-S_1} \Pr\{S_c, S_1|L\} * \mathcal{G}_m(n-S_1)}{1 - \sum_{S_c=1}^{L} \Pr\{S_c, S_1 = 0|L\}}, & m \geq 0 \end{cases} \quad (15)$$

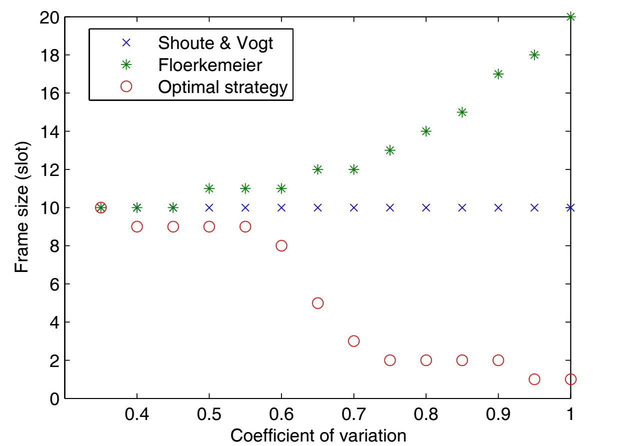

Fig. 5. The choices of the first frame size by different algorithms when $E[N] = 10$.

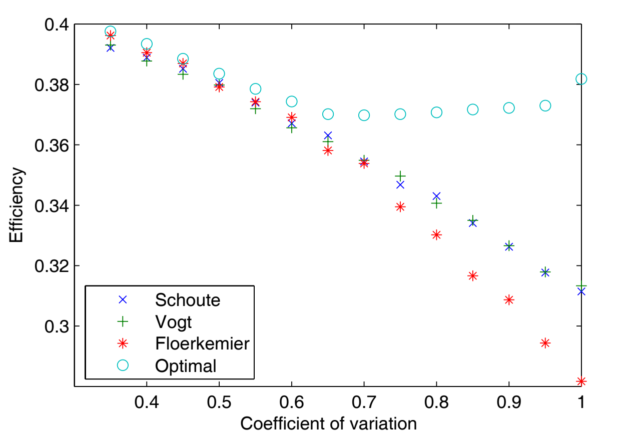

Fig. 6. Efficiency of different algorithms when $E[N] = 10$.

Vogt's, Floerkemeier's and the optimal algorithms.[3] Schoute's and Vogt's algorithms always set the first frame size $L = E[N] = 10$ regardless of the value of c_v; Floerkemeier's algorithm sets $L = 10$ when $c_v < 0.5$ and increases L with c_v; the optimal algorithm also sets $L = 10$ for small c_v but decreases L with c_v. Fig. 6 shows the efficiency ($\eta = \frac{E[N]}{E[T]}$) of the four algorithms from computer simulation (10^4 samples for each point) as a function of c_v. We observe:

[3]We do not simulate Kodialam's algorithm [16], because that algorithm is designed only for estimating the tag population instead of identifying tags.

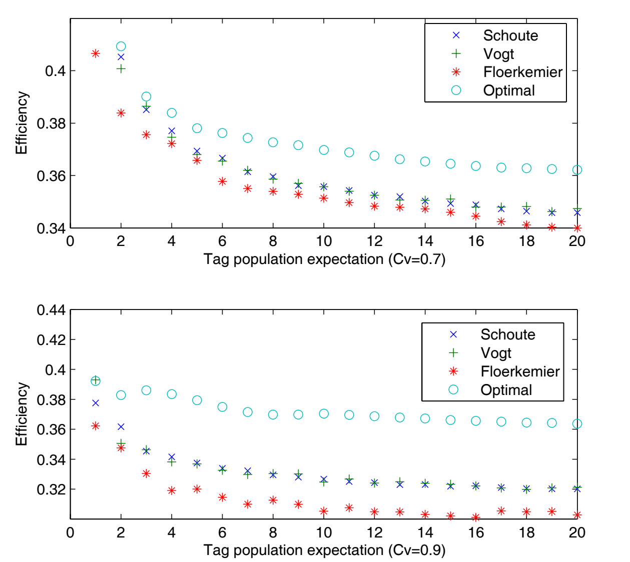

Fig. 7. Efficiency of different algorithms when $c_v = 0.7$ and $c_v = 0.9$.

- The performance of Schoute's and Vogt's algorithms are barely distinguishable;
- Floerkemeier's algorithm is marginally better than Schoute's when $c_v \leq 0.6$ but poorer when $c_v > 0.6$; and
- The optimal algorithm performs best, and its performance is quite insensitive to c_v.

Next, we let $E[N]$ vary from 1 to 20 while keeping c_v equal to 0.7 and 0.9 respectively. Fig. 7 shows the performance of the four algorithms. Again, the optimal algorithm is performs markedly more efficiently than all the others.

V. An Application

The direct use of the optimal algorithm in real application is not recommended, as it requires a lot of computation and storage in the reader. However, the optimal frame sizes are very useful for algorithm design.

As shown in Section IV, previous algorithms [10-12] are quite efficient when the tag population variance is small. Since the variance decreases after each frame of reading, a simple and efficient approach is to use the optimal frame sizes for the first several frames and revert to the previous strategies thereafter.

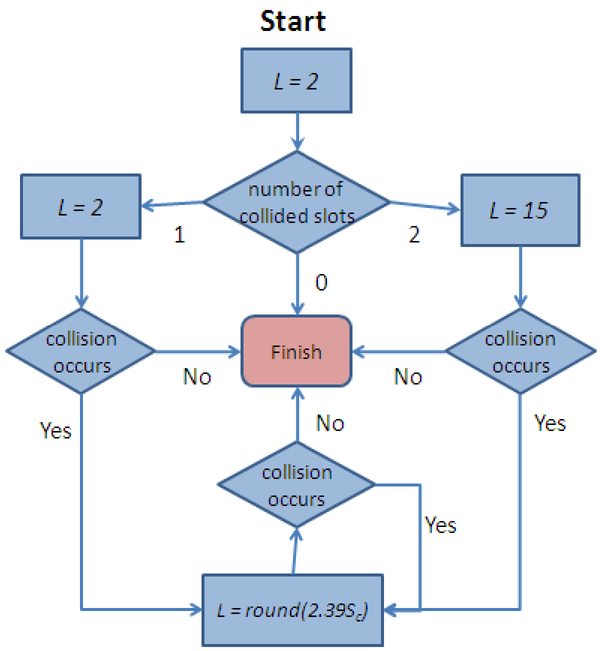

Fig. 8. Improved Schoute's algorithm for a distribution with $E[N] = 10$ and $c_v = 0.8$.

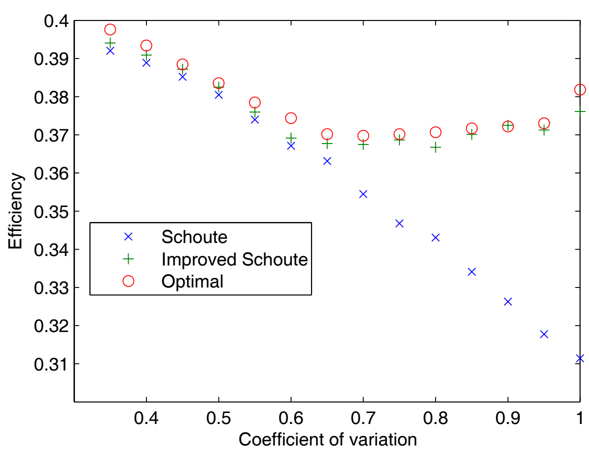

Fig. 9. Efficiency of different algorithms when $E[N] = 10$.

As an example, we improved Schoute's algorithm by using $L = L^*$ in the first two frames. Thereafter, we set $L = \text{round}(2.39s_c)$ as in Schoute's algorithm. Specifically, for the distribution in (16) with $E[N] = 10$ and $c_v = 0.8$, the reading strategy of the Improved Schoute's algorithm is shown in Fig. 8. It stipulates the use of a small frame size ($L = 2$) for the first frame to check if N is "relatively" large or "relatively" small. If collisions occur in both slots, N is estimated as a large value and subsequently a large frame size ($L^* = 15$) will minimize collision. If a collision occurs in only one slot, N should be relatively small and $L^* = 2$ is optimal. Fig. 9 shows that the improved algorithm can track the performance of the optimal algorithm quite closely.

In practice, the reader does not need to store L^*. For inventory applications where many groups of tags are read sequentially, it is more efficient to dynamically update the estimation of N and the choice of L. As an example, after 100 groups are identified, the reader may update the distribution of N using Bayes formula and recalculate L^* for the next 100 groups.

VI. Conclusion

In this paper, we have proposed a new theory for the FA based RFID anti-collision systems. Based on this theory, we have analytically derived the optimal frame size and the average reading time. Simulation results have confirmed that this approach results in a significantly better performance than previously-published algorithms have been able to achieved. The methodology developed in this paper can be extended to other RFID systems, including the Tree based systems, Interval based systems and advanced FA based systems. An example is in [22], where the optimal frame size is derived with the inclusion of the 'Frame-size Reset' command. During the course of our research, we were delighted to discover that the Adaptive Markov Decision Process theory could be used to solve our problem.

Appendix I

Lemma 1: *Let* $\mathbf{v}$ *and* $\mathbf{u}$ *denote two distributions and* $\mathbf{v} \neq \mathbf{u}$. *We have*

$$\mathcal{T}_o(\mathbf{v}) = \sum_{n=0}^{N_{max}} v_n T_o(n \,|\, \mathbf{v}) \leq \sum_{n=0}^{N_{max}} v_n T_o(n \,|\, \mathbf{u}),$$

where $T_o(n \,|\, \mathbf{v})$ *denote the average reading time of* n *tags following the optimal strategy when the initial estimation is* $\mathbf{v}$.

We do not have the space here to provide a rigorous proof, but this lemma can be easily justified by intuition. For a group with distribution $\mathbf{v}$, the average reading time based on the correct distribution $\mathbf{v}$ is always less than that based on the wrong distribution $\mathbf{u}$.

Lemma 2: *For a group of tags with population distribution* $\Pr(N) = \mathbf{v} = (v_0, v_1, v_2, \ldots, v_{max})$, *variance* $Var(N) \to 0$ *and expectation* $E[N] \to k$, *the optimal frame size* $L^* = k$.

Proof: Let $\mathbf{u} = (u_0, u_1, u_2, \ldots, u_{max})$ denote another distribution with $u_k = 1$. As derived in Section III.B Case 3, the optimal frame size for $\mathbf{u}$ is $L^* = k$ and the average reading time is $\mathcal{F}(k)$. From Lemma 1, we have

$$\begin{aligned} \mathcal{T}_o(\mathbf{v}) &\leq \sum_n v_n T_o(n \,|\, \mathbf{u}) \\ &= v_k T_o(k \,|\, \mathbf{u}) + \sum_{n \neq k} v_n T_o(n \,|\, \mathbf{u}) \\ &= v_k \mathcal{F}(k) + (1 - v_k) Y, \end{aligned} \tag{17}$$

where Y is averaged reading time on condition that $N \neq k$, or

$$Y = \frac{\sum_{n \neq k} v_n T_o(n \,|\, \mathbf{u})}{\sum_{n \neq k} v_n} < \infty.$$

On the other hand, if the first frame size is chosen as $L = m \neq k$, we let $T'(n \,|\, L = m)$ denote the average reading time of n tags when the first frame size is m and the following frame sizes are optimal. Obviously $\mathcal{F}(k) < T'(k \,|\, L = m)$ because for k tags to read, the optimal frame size $L^* = k$. Based on these, we have the average reading time using $L = m$ as

$$t = \sum_n v_n T'(n \mid L = m) > v_k T'(k \mid L = m) = v_k \mathcal{F}(k) + v_k \Delta, \quad (18)$$

where $\Delta = T'(k \mid L = m) - \mathcal{F}(k) > 0$. Since Δ and Y are both constant and $v_k \to 1$ when $\mathrm{Var}(N) \to 0$, we obtain $v_k \Delta > (1 - v_k)Y$. Combining with (17) and (18), we have $\mathcal{T}_o(\mathbf{v}) < t$. Thus we claim that the optimal frame size $L^* = k$. □

Proof of Theorem 2:

As defined in Section III, $\mathcal{G}_m(n)$ is the average reading time of n tags with the estimated group size distribution $\mathbf{v} = (v_0, v_1, v_2, \ldots, v_{max})$ satisfying 1) $v_{m+n} \to 1$ and 2) $\mathrm{Var}(N) \to 0$. In other words, $\mathcal{G}_m(n)$ is the average reading time of n tags with the wrong information that there are $n+m$ tags. By Lemma 2, the optimal frame size for $\mathbf{v}$ is $m+n$. We now derive the formulas of $\mathcal{G}_m(n)$ for different values of m.

Case 1: $m \geq 0$, or the estimated group size is larger than the real one. After s_1 tags are identified in the current frame, the real group size becomes $n - s_1$ while the estimated number of remaining tags is $n + m - s_1$. Based on the replies, the reader will not detect the estimation mistake until all the tags are successfully identified. Similar to (12), we have the system function as

$$\mathcal{G}_m(n) = L + \sum_{s_1=0}^{L} \sum_{s_c=0}^{L-s_1} \Pr\{S_c = s_c, S_1 = s_1 \mid L\} * \mathcal{G}_m(n - s_1),$$

where $L = m + n$. Then we can obtain explicit form as (15) by rearranging the terms.

Case 2: $m < 0$, or the estimated group size is smaller than the real one. If the number of the collided slots $s_c \leq \frac{n+m}{2}$, the reader will not discover the mistake and update the estimated group size as $n+m-s_1$. But the reader will know something is wrong if the number of collided slots $s_c > \frac{n+m}{2}$, because $n+m$ tags cannot produce so many collisions. Since $\mathrm{Var}(n) \to 0$, the estimated group size should be updated as $N = 2s_c$. Similarly, we have the system function as

$$\begin{aligned}\mathcal{G}_m(n) \quad &= \quad L + \sum_{S_1=0}^{L} \sum_{S_c=0}^{X} \Pr\{S_c, S_1|L\} * \mathcal{G}_m(n - S_1) \\ &\quad + \sum_{k=0}^{m+1} \sum_{S_1=0}^{n+m} \Pr\{S_c = \frac{n - S_1 + k}{2}, S_1|L\} * \mathcal{G}_k(n - S_1),\end{aligned}$$

where $L = m + n$ and $X = \min(L - S_1, \frac{n-S_1}{2})$. (15) can be obtained by reshuffling the terms.

Appendix II

Consider two groups, Group A and Group B with tag population N_A and N_B respectively. Suppose Group A satisfies: $E[N_A] = k$ and $\mathrm{Var}(N_A) = \delta > 0$; Group B satisfies: $N_B = k$. Let $\mathbf{v}$ and $\mathbf{u}$ denote the distribution of N_A and N_B respectively. We can prove

$$\mathcal{T}_o(\mathbf{u}) \leq \mathcal{T}_o(\mathbf{v}) \leq \sum_{n=0}^{N_{max}} v_n T_o(n \mid \mathbf{u}). \quad (19)$$

The first inequality holds because when the expectation is same, the group with known population needs less time. The second inequality holds as a result of Lemma 1. By definition, $\mathcal{T}_o(\mathbf{u}) = \mathcal{F}(k)$; $T_o(n \mid \mathbf{u}) = \mathcal{G}_{k-n}(n)$. Thus (19) becomes

$$\mathcal{F}(k) \leq \mathcal{T}_o(\mathbf{v}) \leq \sum_{n=0}^{N_{max}} v_n \mathcal{G}_{k-n}(n). \quad (20)$$

According to (14), we use $\widehat{\mathcal{T}_o}(v) = \sum_{n=0}^{N_{max}} v_n \mathcal{G}_{k-n}(n)$ as an approximation of $\mathcal{T}_o(\mathbf{v})$. Then the error bound of this estimation can be derived from (20) as

$$\left|\widehat{\mathcal{T}_o}(v) - \mathcal{T}_o(v)\right| \leq \left|\mathcal{F}(k) - \sum_{n=0}^{N_{max}} v_n \mathcal{G}_{k-n}(n)\right|. \quad (21)$$

References

[1] K. Finkenzeller, *RFID Handbook*, 2nd edition. John Wiley & Sons, 2003.

[2] EPCglobal, EPCglobal Class 1 Generation 2 UHF Air Interface Protocol Standard Version 1.0.9, 2005. [Online]. Available: http://www.epcglobalinc.org/

[3] International Organization for Standardization, Information technology - RFID for item management - Part 6: "Parameters for air interface communications at 860 MHz to 960 MHz," 2004.

[4] Philips Semiconductors, I-CODE Smart Label RFID Tags. [Online]. Available: http://www.semiconductors.philips.com/acrobat_download/other/identification/SL092030.pdf

[5] J. Myung, W. Lee, J. Srivastatva, and T. K. Shih, "Tag-splitting: adaptive collision arbitration protocols for RFID tag identification," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 6, June 2007.

[6] K. W. Chiang, C. Q. Hua, and T. S. P. Yum, "Prefix-randomized query-tree protocol for RFID systems," in *Proc. IEEE ICC*, 2006.

[7] P. Popovski, F. H. P. Fitzek, and R. Prasad, "Batch conflict resolution algorithm with progressively accurate multiplicity estimation," in *Proc. ACM DIALM-POMC*, Oct. 2004.

[8] J. Park, M. Y. Chung, and T. J. Lee, "Identification of RFID tags in framed-slotted ALOHA with tag estimation and binary splitting," *Commun. Electron.*, 2006

[9] J. Mosely and P. A. Humblet, "A class of efficient contention resolution algorithms for multiple access channels," *IEEE Trans. Commun.*, vol. 33, no. 2, Feb. 1985.

[10] F. C. Schoute, "Dynamic frame length ALOHA," *IEEE Trans. Commun.*, vol. 31, no. 4, pp. 565–568, Apr. 1983.

[11] H. Vogt, "Efficient object identification with passive RFID rags," in *Proc. 1st International Conf., PERVASIVE, Volume 2414 of Lecture Notes in Computer Science (LNCS)*, pp. 98–113, Aug. 2002.

[12] C. Floerkemeier, "Transmission control scheme for RFID object identification," in *Proc. Pervasive Wireless Networking Workshop (IEE PERCOM)*, 2006.

[13] B. Zhen, M. Kobayashi, and M. Shimizu, "Framed ALOHA for multiple RFID objects identification," *IEICE Trans. Commun.*, vol. E88-B, no. 3, Mar. 2005.

[14] M. Buettner and D. Wetherall, "An empirical study of UHF RFID performance," in *Proc. MobiCom*, 2008.

[15] C. Floerkemeier, "Bayesian transmission strategy for framed ALOHA based RFID protocols," in *Proc. IEEE International Conf. RFID*, Mar. 2007.

[16] M. Kodialam and T. Nandagopal, "Fast and reliable estimation schemes in RFID systems," in *Proc. MobiCom*, Sep. 2006.

[17] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, 1st edition, Morgan Kaufmann Publishers.

[18] W. Feller, *An Introduction to Probability Theory and Its Applications*, 2nd edition, John Wiley & Sons.

[19] J. Keilson, *Markov Chain Models - Rarity and Exponentiality*. Springer-Verlag, 1979.

[20] D. J. White, *Markov Decision Processes*. John Wiley & Sons, 1993.

[21] J. J. Martin, *Bayesian Decision Problems and Markov Chains*. London: Wiley, 1967.

[22] L. Zhu and T. S. P. Yum, "Design and analysis of framed ALOHA based RFID anti-collision algorithms," in *Proc. IEEE Global Commun. Conf.*, Dec. 2009.

Lei Zhu (S'09) received the B.S. degree in electronic information science and technology from the University of Science and Technology of China in 2006. He is a Ph.D. candidate at the Chinese University of Hong Kong. His general research interests include protocol and algorithm design, RFID systems and random access systems, and performance evaluation for computer networks, wireless mobile, and sensor networks.

Tak-Shing Peter Yum (S'76-A'78-SM'86) was born in Shanghai. He received primary and secondary school education in Hong Kong. He went to Columbia University and was awarded BS, MS, MPh, and PhD degrees in 1974, 1975, 1977, and 1978 respectively. He joined Bell Telephone Laboratories in April 1978, working on switching and signaling systems for two and one half years. Then, he taught at National Chiao Tung University, Taiwan, for two years before joining The Chinese University of Hong Kong in 1982. He was appointed department chairman two times and was elected dean of engineering for two terms. He is currently senior advisor of Applied Science at the Technology Research Institute of Hong Kong (www.astri.org).

Professor Yum has published widely in Internet research, with contributions to routing, buffer management, deadlock handling, message resequencing, and multi-access protocols. He then branched out to work on cellular network, lightwave networks, video distribution networks, and 3G networks. His recently research is in the areas of RFID, sensor networks, and wireless positioning technologies.