A Critical Survey and Analysis of RFID Anti-Collision Mechanisms

Lei Zhu and Tak-Shing Peter Yum, The Chinese University of Hong Kong

ABSTRACT

The anti-collision mechanism is an important part of RFID systems. Many algorithms were proposed in the literature, but most of them were designed based on the theory of Random Access (RA) systems. In this article, we review the various types of anti-collision systems and their corresponding algorithms. Different reading strategies are critically analyzed and compared by computer simulation. By comparing the different working mechanisms of RA and RFID systems, we show that a new theory is needed for the design and analysis of RFID anti-collision algorithm.

INTRODUCTION

During the past decade, the price of passive RFID tags has dropped low enough to replace the barcode tags for many applications. Retail giants including Wal-Mart in the U.S., Marks & Spenser in the UK, Metro in Germany, and Mitsukoshi in Japan, have all implemented their RFID solutions for more efficient supply chain management. In the foreseeable future, RFID tags will be cheap enough to realize item-level tagging, thus enabling a new wave of innovative applications. RFID technology, which provides efficient wireless object identification, is envisioned to bridge the physical world and the virtual world. Many large companies have set foot in this area, providing hardware and software solutions as well as contributing to a global standard. The major RFID technology providers include Philips Electronic, Texas Instruments, IBM, Intel, SAP, VeriSign, Sun Microsystems, and Alien. For details of their product lines please refer to their respective company websites.

EPCglobal has released a comprehensive set of standards that define the working mechanism for many parts of RFID systems, such as the reader, the tag, the Object Name Service (ONS), and the EPC Information Services (EPCIS). The standard on anti-collision strategies is specified in 'Class 1 Generation 2 UHF Air Interface Protocol Standard' [2].

This article offers a critical survey of anti-collision mechanisms for passive tags. Passive tags have no power source and minimum functionality. They cannot sense the media or cooperate with other tags. As they can only be polled by the reader, their replied packets will collide and be lost if multiple tags are polled [1].

Over the years, many RFID communication systems have been proposed in the literature. We summarize their common operations as follows:

- The RFID Reader broadcasts an inquiring command to initiate the communication.
- Upon hearing the Reader's command, tags in the vicinity run a function to decide whether to reply or not. The function can be deterministic (compare the prefix of its ID with the string broadcast by the reader and reply if they match) or probabilistic (generate a random number from 0 to L and reply if the number is 0).
- The reader continues to broadcast inquiring commands until it is sure that all tags are identified.

The command sets of the reader and the reply functions of the tag are usually different in different RFID systems. The reading efficiency is usually not the major concern; instead, the system robustness and system complexity are more important considerations. That is why the systems defined in standards [2, 3] are usually not the most efficient ones. Traditionally, the anti-collision systems can be divided into two main classes: Framed Aloha based systems and Tree based systems. Besides these two major classes, some new systems have been proposed recently, such as the interval based system and some hybrid versions.

The tag reading strategy of an RFID system is called *the RFID anti-collision algorithm*. Here we emphasize the difference between the "anti-collision system" and the "anti-collision 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. Most algorithms are adaptive, i.e., command parameters are changed based on the tag population. But the tag population is usually unavailable. Therefore, an RFID anti-collision algorithm usually contains two parts: *tag population estimation* and *reading strategy determination*. As shown in Fig. 1, the first part is for estimating the tag population based on the tags' replies, while the second part is for adjusting the parameters based on the estimation. In the following sections, we will review the major systems and their corresponding algorithms.

Framed Aloha-Based Anti-Collision Systems and Algorithms

Framed Aloha (FA) is a variation of slotted Aloha where a terminal (tag) is permitted to transmit once per frame. It is widely used in RFID systems to reduce collisions between tags. As examples, the anti-collision systems defined in EPC-global standards [2] and Philips Smart Label IC data sheet [3] are both Framed Aloha based. In this section, we will review three traditional types of Framed Aloha based systems and their corresponding algorithms.

Simple Framed Aloha Systems

This system [4] was designed in the last century before the emergence of the Philips and EPC-global standards. Both hardware and software are very simple, but the algorithms cannot avoid collisions effectively. The reader can broadcast only two commands: *Frame-start* and *Trigger*. The communication mechanism can be summarized as follows:

- The RFID Reader broadcasts the *Frame-start* command with an integer parameter, say 'L'. Upon hearing this command, all tags generate a random number between 0 and $L - 1$ and keep them in counters. Those that have generated '0' reply immediately.
- Subsequently, the Reader broadcasts the *Trigger* command $L - 1$ times. Upon hearing this command, all tags decrease their counter value by 1 and reply if the counter value reaches '0'.

In this system, tags do not know their transmission results as there is no feedback from the reader. They will all transmit again in the next frame. Since some tags may suffer collisions again and again (the tag starvation problem), the Reader needs to go through many frames to make sure that with high probability all tags have replied at least once. Although dynamically changing the frame size based on the collision ratio can yield some improvements, the efficiency of simple FA algorithms is still poor. In [5], the Simple Framed Aloha system is found to be efficient for estimating tag population. Identifying tags, however, is a different problem.

The Philips System

The Philips System [3] introduced the 'Silence' Command[1] so that the identified tags will not contend the channel in future frames. Its anti-collision mechanism[2] is summarized as follows:

- At the beginning of a frame, the RFID Reader broadcasts the *BEGIN-ROUND (Frame-start)* 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.
- If only one tag replies, the reader can identify it and send back the *FIX-SLOT (Silence & Trigger)* command. Upon 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 contend the channel if the counters reach 0.
- If multiple tags reply or no tag replies, the reader will send back the *CLOSE-SLOT (Trigger)* command. Upon hearing this command, all the unsilenced tags will decrease their counters by 1 and contend the channel if their counters reach 0.
- After one frame ends, the Reader will begin a new frame by broadcasting the *BEGIN-ROUND* command if some tags were collided in the previous frame.

The introduction of the *Silence (FIX-SLOT)* command improves the system performance in two ways:

- **Smaller frame size**: The reader can decrease the frame size after each frame, as some tags are silenced.
- **Fewer frames**: The reader can stop reading when all tags are silenced.

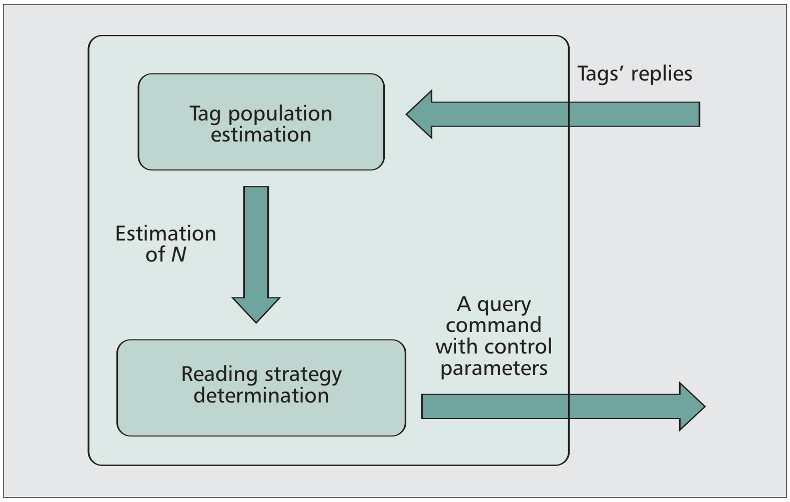

Figure 1. *The anti-collision algorithm for the RFID reader.*

Anti-Collision Algorithms for the Philips System

In the Philips system, the algorithm designer can only change the frame size L. Since the tag population is unknown, the algorithms should first estimate the tag population and then adjust L based on the estimation. Previous work focuses on *Population Estimation*. For determining the *Reading Strategy*, nearly all the previous algorithms use the classic results of Random Access systems. The classical formula [6] for expected throughput U given terminal number N and frame size L is:

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

Simple calculus shows that U is maximized at $L = N$, meaning that the *instantaneous throughput* can be optimized by setting the frame size equal to the terminal number.

Schoute [6] found that when N is large and L suitably chosen (say $L \approx N$), the number of tags attempting each slot has a Poisson distribution

[1] *This command has different names in different literature. In some papers, it is also called 'Kill' command. In the Philips System, it corresponds to the 'CLOSE-SLOT' command, which has other uses besides silencing a tag.*

[2] *The real communication is much more complex than that stated here. We omit the process not relevant to anti-collision.*

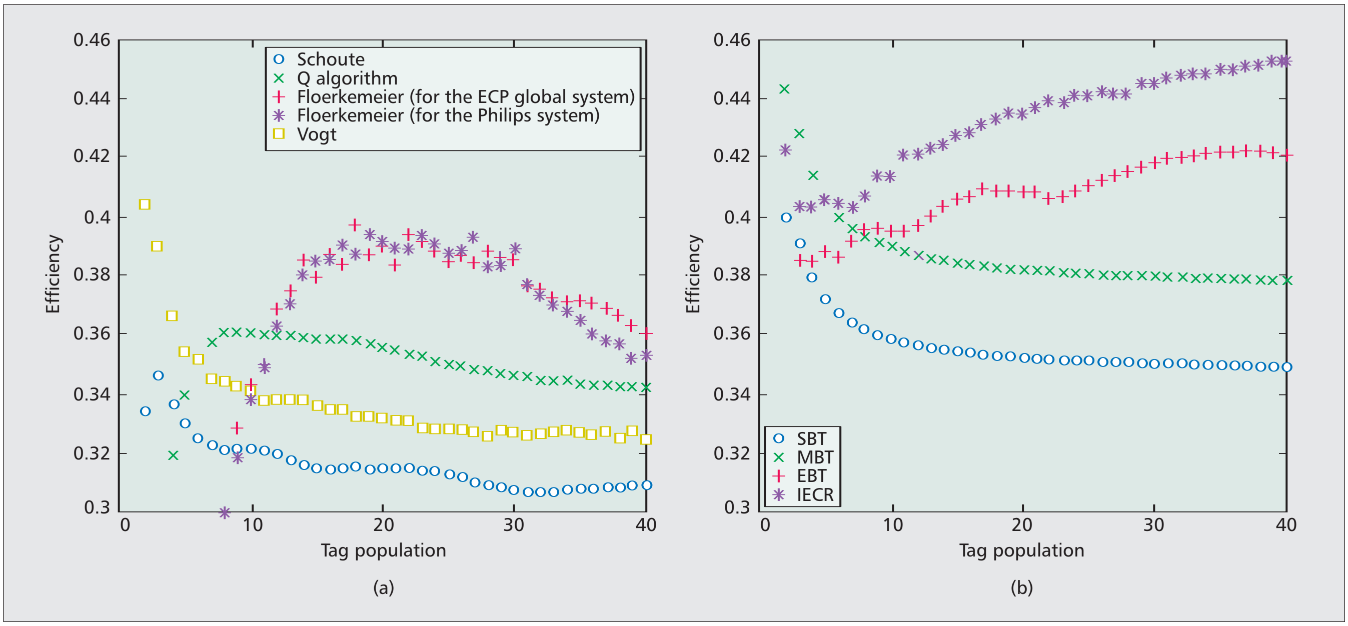

Figure 2. *Efficiency of different algorithms.*

with mean 1. So in the *Population Estimation* part, his algorithm uses $\hat{N}$ = round $(2.39s_c)$, where s_c is the number of collided slots in the last frame. Based on this, the frame size is set as $L = \hat{N} = \text{round}(2.39s_c)$. Without prior knowledge, setting L for the first frame would be a problem and it is usually set as $L = 1$.

Vogt [15] improved the *Population Estimation* strategy of Schoute's algorithm by using the statistics of empty slots s_e and singleton slots s_s in addition. $\hat{N}$ is chosen to minimize the error between the observed values of s_e, s_s, s_c and their expected values using $\hat{N}$. Similarly, $L = \hat{N}$ is used.

Floerkemeier [7] assumes that a rough estimation of the target group size is always available in the form of a distribution $\Pr\{N = n\}$. This distribution is updated by Bayesian method at the end of every frame. The *Frame Size* is chosen to maximize the expected instantaneous throughput $E[U(N, L)]$ based on the distribution. This strategy is better, as the frame size is derived from a distribution rather than a maximum-likelihood value.

Anti-collision algorithms' performance is traditionally compared by their efficiency η_N defined as the number of tags divided by the expected reading time. Figure 2 shows the simulation results of η_N for some algorithms introduced in this article.[3] In Fig. 2a, we compared some FA-based algorithms. Schoute's algorithm is the worst ($\eta_N \approx 0.31$ for large N), because it uses a very simple estimation strategy; Vogt's algorithm is about 10 percent better, because its estimation strategy uses the statistics of empty slots and singleton slots; Floerkemeier's algorithm is the best because of the use of Bayesian statistics.

The EPCglobal System

EPCglobal standards provide a more advanced RFID system. The operations are summarized as follows:

•At the beginning of a frame, the RFID Reader broadcasts the QueryAdjust (Framestart) command with an integer parameter Q. Upon hearing this command, unsilenced tags generate a random value from 0 to $2^Q - 1$. Those generating '0' reply immediately. But unlike the previous systems, the tag will only reply a short packet, namely 'RN16' (a 16-bit random number), containing its temporary ID.

•If multiple tags reply or no tag replies, the reader will send the '*QueryRep*' (Trigger) command. Upon hearing this command, unsilenced tags will decrease their counters by 1 and contend the channel when the counters reach 0.

•If only one tag replies, the reader can receive its RN16 successfully. The reader has a set of *operation* commands: reading data, writing new data, changing password, etc. The reader can select a particular tag by including its RN16 in the operation commands. After the operation, the tag will be silenced.

•The reader can use the 'QueryAdjust' command again even before the frame ends. Upon hearing this command, unsilenced tags regenerate their counter values according to the new frame size.

Compared with the Philips system, The EPCglobal system has two advanced features.

- **Slot reservation**. The tags use the temporary ID (16 bits) to reserve an operation slot for the EPC information packet (about 100 bits). Since the communication after the reservation is collision-free, the performance of anti-collision algorithms for this system is compared by expected contention time, measured by contention slot[4].
- **Frame cancellation mechanism**. The reader can initiate a new frame using the 'QueryAdjust' command whenever the current frame size is found unsuitable.

[3] *In the simulation, the initial estimation for Floerkemeier's algorithm is set as a uniform distribution from 1 to 40.*

[4] *A contention slot is defined as the time from the point that the reader broadcasts a query command to the point that tags finish replying their RN16.*

ANTI-COLLISION ALGORITHMS FOR THE EPCGLOBAL SYSTEM

In the EPCglobal system, the reader can cancel the running frame and initiate a new one at any time slot. Accordingly, the algorithm for EPCglobal systems needs to specify how to choose frame size *and* when to cancel a running frame.

The Q algorithm is used in EPCglobal standards [2]. As shown in Fig. 3, the RFID 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.[5] The tag population is estimated as $2^{Q_{fp}}$. Based on this estimation, the reading strategy is designed as:

• Set the frame size to 2^Q, where Q = round(Q_{fp}).

• Cancel the frame when the estimation prefers another frame size, i.e., the value of round(Q_{fp}) changes. In [8], the efficiency of the Q algorithm was obtained with different choices of C and Q_{fp} and some methods to improve the estimation strategy were proposed. The performance of the Q algorithm is also shown in Fig. 2a. In the simulation, $Q_{fp} = 4$ and $C = 0.4$ are set as recommended in the standards. The Q algorithm is about 8 percent better than Vogt's algorithm. This improvement is obtained by the introduction of a frame-cancellation mechanism.

In [9], Floerkemeier proposed an improved strategy. Similar to that in [7], a rough estimation of N in the form of a distribution is assumed known. In the tag *Population Estimation* part, the Bayesian estimation method is used *after each slot* for updating the distribution of N. Based on this updated distribution, the reading strategy is designed as:

- Set the frame size to L^* which maximizes the expected instantaneous throughput.
- Cancel the frame when the updated distribution calls for a new L^*.

Floerkemeier's algorithm performs much better than the Q algorithm when the initial estimation is accurate. However, the trade-off is complexity, as in each slot it needs to do Bayesian estimation approximately N times and solve a nonlinear equation to obtain L^*.

TREE-BASED ANTI-COLLISION SYSTEMS AND ALGORITHMS

Tree-based algorithms are those whose reading process can be modeled by a tree. The RFID Reader can split a collided group and ask a subgroup to reply. With the initial group being the root, the inquired groups and their split descendants form a tree. There are two types of Tree-based algorithms studied in the literature.

THE BINARY TREE SYSTEM

Binary Tree systems use a splitting probability of 0.5 to resolve collided groups. The operations are:

• Before starting, all tags set their counter values to 1.

• The RFID Reader can broadcast the *Trigger* command. Upon hearing this command, tags with counter value 0 will be silenced while all others decrease the counter values by 1 and reply when their counters reach 0.

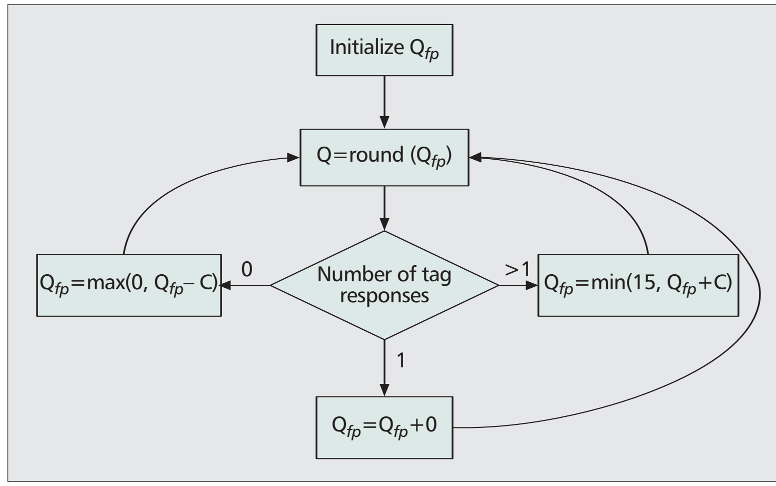

Figure 3. *Mechanism of the Q-algorithm.*

• The RFID Reader can broadcast the *Split* command. Upon hearing this command, tags with counter value 0 will add a random bit to the counter and reply if the bit is 0 while the others will increase the counters by 1.

This system is very simple, requiring only two commands. It is, however, less robust. To see this, Consider the case that the channel is unreliable and some replies from tags may not reach the reader. In FA-based systems, these tags will retransmit in the future frames. But in Binary Tree systems, replies not reaching the reader would mean there are no tags with counter value 0. The reader then broadcasts the *Trigger* command and lets all tags decrease their counter values. This causes ill-read tags to be ignored by the reader.

ANTI-COLLISION ALGORITHMS FOR THE BINARY TREE SYSTEM

The Simple Binary Tree (SBT) algorithm is our first example. It uses the Split command for the collided groups and the Trigger command for the others.

Figure 4 shows an example of reading three tags. Initially, all tags have counter values set to 1 and reply upon receiving the *Trigger Command*. This state corresponds to the root node ε of the tree. In Fig. 4, three tags are shown collided in ε. Based on this collided reply, the reader broadcasts a *Split Command*. As shown, two tags choose to add 0 to their counter and reply immediately to form the left-hand child of ε, while one tag chooses to add 1 and stay dormant to form the right-hand child of ε. When a collision is detected from the replies of the left-hand group, SBT uses the *Split Command* again. This continues until no collision is detected. In this example, the tree has seven nodes and so it takes the reader seven time slots to identify the three tags. Mathematical analysis shows that SBT is only efficient for small groups of tags. As shown in Fig. 2b, its efficiency drops quickly to 0.347 for large tag population. The Tree-based algorithms usually perform better than Aloha

[5] *In EPCglobe standards, it is recommended that 0:2 · C · 0:5 and the initial Qfp = 4.*

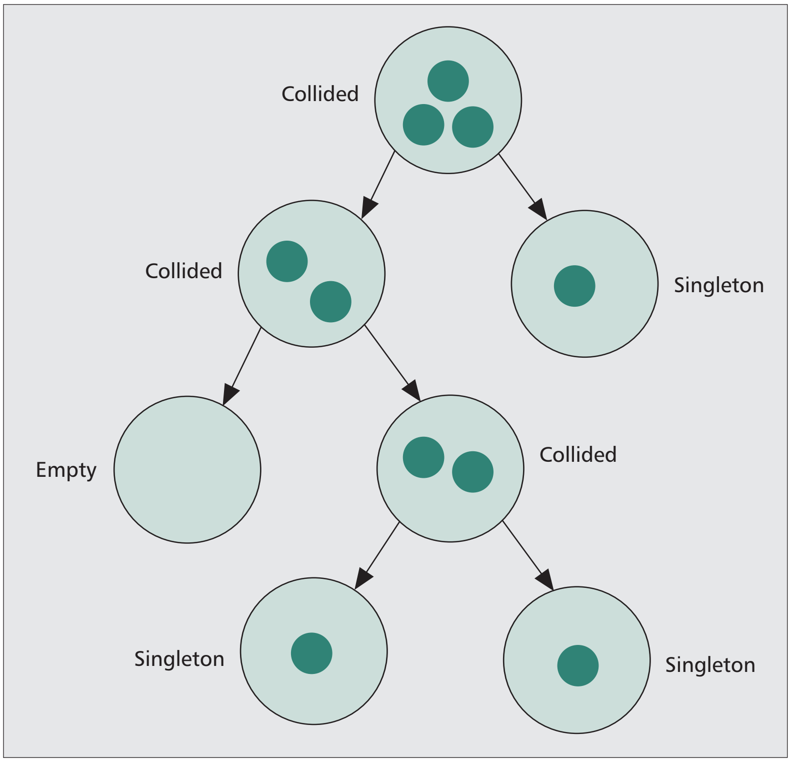

Figure 4. *Reading process of Simple Binary Tree algorithm.*

based ones, because splitting collided groups repetitively is more efficient than putting them together for the next frame.

The Modified Binary Tree (MBT) algorithm is our second example. It uses the *Split Command* for collided groups and also other groups that are known to contain multiple tags. As shown in Fig. 4, since Group '0' is collided and Group '00' is empty, Group '01' is certain to contain multiple tags. So MBT uses a *Split* directly instead of a *Trigger* followed by a *Split* to save one time slot. Figure 2b shows that the efficiency of MBT approaches 0.376 for a large population, about nine percent better than SBT.

THE QUERY TREE SYSTEM

In Query Tree Systems, all tags are assigned a unique ID. The ID can be the EPC code or a random binary string generated at the initializing stage. In each slot, the RFID Reader broadcasts a binary string. Tags compare the prefix of its ID with the string received and reply if they match.

The Query Tree system only needs the *Trigger Command* to resolve collisions. If the *group* of tags with prefix $'b_1b_2...b'$ (b_i ε{0, 1}) is collided and the reader broadcasts the string $'b_1b_2...b_x0'$ in the next slot, about half of the tags in the previous group will reply. This is just like splitting the group with probability $p = 0.5$. If the reader broadcasts $'b_1b_2...b_x00'$ instead, the previous group will be split with $p = 0.25$. This ability to vary p is very useful for handling a large tag population.

On the other hand, a Query Tree system has the following problems that limit its application in RFID standards.

- **System Fragility**: Similar to the case in section III-A, if some replies do not reach the reader, it may assume there are no tags with the prefix and those tags involved will not be identified in the future.
- **ID collision**: It is possible that two tags generate the same ID and become indistinguishable.

ANTI-COLLISION ALGORITHMS FOR THE QUERY TREE SYSTEM

Similar to the Aloha-based algorithms, tag population estimation is also important for fast reading. In [10], Chiang proposed an estimation method that resembles Schoute's method [6]. The reader begins with a certain length of prefix, say l, and sequentially broadcasts all possible combinations of prefixes with length l, i.e., from '000···00', '000···01' to '111···11'. If the collision ratio is larger than a threshold, the reader will increase l and trigger all the possible combinations again. If the collision ratio is low, every collided slot will be further split using the MBT strategy.

In [11], Popovski proposed the Estimation Binary Tree (EBT) algorithm. This algorithm estimates the tag population based on the identified-tag ratio. Specifically, during the reading process, the reader estimates the initial tag population as the number of identified tags divided by their proportion of the overall tag population, or n_s/p_s. As an example, let the first identified tag have ID prefix '0010'. Then we have $n_s = 1$ and $p_s = 3/32$. Therefore, the estimated population is 32/3. After the estimation, the length of next prefix is set around $\log_2 \hat{N}_\varepsilon$. This choice corresponds to setting $L = \hat{N}$ in a FA-based system. As shown in Fig. 2b, the efficiency of EBT is higher for a large tag population and approaches 0.44, about 17 percent better than MBT. This increasing efficiency results from the increasing accuracy of population estimation.

A COMPARISON OF A FA-BASED SYSTEM AND A TREE-BASED SYSTEM

We compare the FA-based system and Tree-based system in three aspects.

•The FA-based system is simpler. During the contention period, commands are broadcast to all the tags, so tags do not need to check the "destination ID"; they only need to perform a counter decrement operation. On the other hand, tags in the Tree-based system have to do a string comparison in every time slot.

•The FA-based system is more robust. When the channel is not perfect, replies from tags may fail to reach the reader. When this occurs, the tags will not be silenced and will still contend the channel in the next frame. For the pure Tree-based system, if a tag's reply does not reach the RFID reader, the reader may assume the triggered group is empty. That group will not have a chance to reply again. To fix this problem, a "complete checking" is needed at the end of the reading process. A second problem with Tree-based systems is "ID collision." Since a tags' ID has finite length, it is possible that two tags may choose the same ID and become indistinguishable. Therefore, the RFID reader needs a special mechanism to resolve them. The common method is to ask them to regenerate ID after the other tags are identified.

• The FA-based system is less efficient. As shown in Fig. 2, the efficiency of the Q algorithm is even lower than that of the Simple Binary Tree algorithm. In FA based systems, the collided tags are merged into one group and split again in the next frame. This practice is not as efficient as the Tree-based systems where tags in collided slots are split repetitively.

Other Types of Anti-Collision Systems and Algorithms

Recently, some novel RFID anti-collision systems have been proposed in the literature. We introduce two of them as follows.

The Interval-Based System and Algorithm

The Interval-based system [11] proposed by Popovski in 2004 has a different working principle. Before reading, every tag generates a random binary string as its ID. In each slot, the RFID Reader broadcasts a special Trigger command containing two binary strings as the upper bound and lower bound of an interval. Upon hearing this command, tags check whether their IDs fall inside this interval or not and reply if they do.

Interval-based algorithms eliminate the "probability choice limitation" of Tree-based and Aloha-based algorithms. By varying the size of the interval, the reader can choose any fractions of the whole group. In other words, the reader can ask tags to reply with a probability that better suits a particular tag-population estimate. On the other hand, this system has the same system fragility and ID collision problems of the Query Tree system. Tags also need to perform string comparison twice within one slot. This prolongs the time slot and increases the hardware and software complexity.

Popovski also proposed the Interval Estimation Conflict Resolution (IECR) algorithm [11]. It uses the identified-tag ratio based method to estimate population (similar to EBT) and chooses the reply interval based on the classical result in Random Access systems. The efficiency of IECR depends on the initial knowledge of the target group size. Figure 2b shows that the efficiency of IECR (tag population unknown case) is much higher than Tree-based and Aloha-based algorithms. But the trade-off is system complexity.

Hybrid Systems and Algorithms

There are many ways to form hybrid systems based on the previous systems. Hybrid systems usually perform better at the expense of higher hardware and software complexity. Some, however, are not realistic. The *Framed Aloha system with Tree-based splitting* [12] is an example for which Aloha and Tree-based operations are used together. The reader can assign a frame for tags as in Framed Aloha based systems and split a collided group as in Tree-based systems. The Framed-Slotted Aloha with Tag Estimation and Binary Splitting algorithm [12] is designed for this system. It begins with FA-based reading using a strategy similar to [6]. When the collision ratio of a certain frame is smaller than a threshold, the reader invokes Tree-based splitting to resolve collided slots.

Reading Strategy Optimization

As illustrated earlier, most anti-collision algorithms are designed based on the classic result of Random Access systems. In this section, we show that formula 1 is not suitable for optimizing the anti-collision algorithms by comparing the RFID systems and RA systems. Furthermore, a new optimization model is proposed.

Comparison of RFID and RA Systems

The traditional RA systems include packet satellite systems and the early Ethernet systems. Terminals in these systems share a common channel and follow the framed aloha mechanism to avoid collisions. Under saturated condition, terminals always have packets to send. Comparing this to RFID systems, we identify two main differences

The traditional RA systems include packet satellite systems and the early Ethernet systems. Terminals in these systems share a common channel and follow the framed aloha mechanism to avoid collisions. Under saturated condition, terminals always have packets to send.

The Contention Group — In RA systems, the terminals continue to contend the channel in the following frames after a successful transmission, or the number of terminals N does not change during the contention process. Therefore, the optimal frame size L^* and expected instantaneous throughput for every frame is the same according to Eq. 1. Thus the steady-state throughput of the system is equal to the instantaneous throughput.

However, in modern RFID systems, such as the Philips system and the EPCglobal system, identified tags are silenced by the reader, leading to tag population decrease during the reading process. When the frames are not identical, a concatenation of locally optimal solutions is not globally optimal. To see this, suppose 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 Floerkemeier's algorithm, 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.

The Advanced Features of RFID Systems — The classic theory for RA systems is designed for simple framed Aloha systems. When new commands (such as a cancellation of a frame and splitting a group of tags) are introduced to RFID systems, Eq. 1 is no longer sufficient to characterize the instantaneous throughput. As an example, suppose the target group contains exactly three tags. From Eq. 1, the suitable frame size should be $L = 3$ and the efficiency is $U = 0.44$. However in the EPCglobal system, if we choose $L = 2$ and follow the strategy shown in Fig. 5, the efficiency can achieve 0.6.

Further studies are needed for designing anti-collision algorithms for noisy channels under software/hardware constraints, with the understanding that the algorithm robustness is a more important requirement than the reading efficiency.

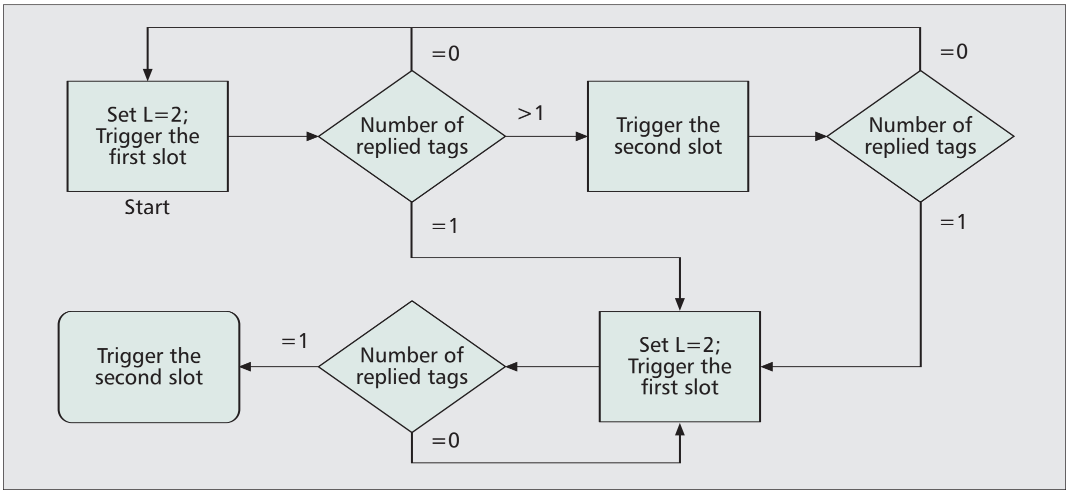

Figure 5. *The optimal strategy to read three tags.*

A New Approach for Anti-Collision Algorithm Optimization

To optimize the anti-collision algorithm, we need to specify three things:

- The optimization objective
- The system variables
- The utility function (a functional relationship between the variables and the optimization objective)

In the following, we discuss them one by one.

The Optimization Objective — The conventional criterion for performance comparison is reading efficiency η_N. This index is simple and intuitive but not suitable for performance optimization, because η_N depends on the distribution of N. As an example, Floerkemeier's algorithms [7, 9] need the distribution of N as an input and their efficiency η_N is higher when the variance of N is smaller.

To compare algorithms, we have to specify the application we have in mind. For a particular application, we assume the tag population statistics are given or roughly known.[6] Let $d_n = \Pr\{N = n\}$ and $\mathbf{d} = (d_0, d_1, \cdots)$. Further let $T(n|\mathbf{d})$ denote the average reading time of n tags, given the tag population distribution is $\mathbf{d}$, and let $\mathbf{T} = (T(0|\mathbf{d}), T(1|\mathbf{d}), \cdots)$. Then, the expected reading time $\mathcal{T}(\mathbf{d})$ for distribution $\mathbf{d}$ is dTT. The optimization objective can now be specified as to minimize $\mathcal{T}(\mathbf{d})$ for a given distribution $\mathbf{d}$. This approach covers the tag population known case as a special case by letting $\mathbf{d} = (0, 0, \cdots, 0, 1, 0, \cdots)$ and $\mathcal{T}(\mathbf{d}) = T(N|\mathbf{d})$.

The System Variables — To model an anti-collision system, we need to specify variables representing the commands and the system parameters. For example, the only parameter we can change in the Philips system is the frame size. So an integer variable L is enough. But in the EPCglobal system where the running frame can be canceled, we need in addition a Boolean variable for its indication. For other systems, we can similarly define variables for each command.

The Utility Function — If an explicit function in the form $\mathcal{T}(\mathbf{d}) = f(\mathbf{v}, \mathbf{d})$, where $\mathbf{v}$ is the set of variables, is given, then $\mathcal{T}(\mathbf{d})$ can be minimized using classical optimization methods. Unfortunately, this explicit function does not exist for most RFID systems. Take the Philips system as an example. The expected remaining reading time depends not only on the current frame size but also on sizes of the subsequent frames. To get around this problem, previous research chose the instantaneous throughput (which only depends on the current frame size) $U(\mathbf{d}) = f(L, \mathbf{d})$ obtained from Eq. 1 as the objective function. But as we have mentioned, maximizing $U(\mathbf{d})$ frame by frame does not lead to minimizing $\mathcal{T}(\mathbf{d})$.

In [13, 14], we have derived the *implicit* functions $\mathcal{T}(\mathbf{d})$ for the Philips and EPCglobal systems. The optimal reading strategies were obtained by solving those functions. It is proved that in the optimal case the efficiency of the FA-based system can achieve 0.4, 30 percent better than Q algorithm.

Conclusion

In this article, we have reviewed the major types of RFID anti-collision systems. Different systems were critically analyzed on the basis of simplicity, robustness, and efficiency. The performance of different algorithms was compared by computer simulation. The salient differences between the RFID anti-collision mechanism and Random Access mechanism were identified and a new model for anti-collision algorithm optimization is proposed. Further studies are needed for designing anti-collision algorithms for noisy channels under software/hardware constraints, with the understanding that the algorithm robustness is a more important requirement than the reading efficiency.

[6] *When the tag population is completely unknown,* N *is uniformly distributed between 0 and* N_{max}, *as we cannot favor any value over the others.*

References

[1] EPCglobal, EPCglobal Class 1 Generation 2 UHF Air Interface Protocol Standard Version 1.0.9, 2005.

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

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

[4] B. Zhen, M. Kobayashi, and M. Shimizu, "Framed Aloha

for Multiple RFID Objects Identification," *IEICE Trans. Commun.*, vol. E88-B, no. 3, Mar. 2005.
[5] M. Kodialam and T. Nandagopal, "Fast and Reliable Estimation Schemes in RFID Systems," *MobiCom'06*, Los Angeles, USA, Sept. 2006.
[6] F. C. Schoute, "Dynamic Frame Length ALOHA," *IEEE Trans. Commun.*, COM-31, no. 4, Apr. 1983, pp. 565–68.
[7] C. Floerkemeier, "Transmission Control Scheme for RFID Object Identification," *Proc. Pervasive Wireless Networking Wksp. at IEE PERCOM 2006*, Pisa, Italy, 2006.
[8] M. Buettner and D. Wetherall, "An Empirical Study of UHF RFID Performance," *MobiCom*, San Francisco, California, USA, 2008.
[9] C. Floerkemeier, "Bayesian Transmission Strategy for Framed ALOHA Based RFID Protocols," *IEEE Int'l. Conf. RFID, Gaylord Texan Resort*, Grapevine, TX, USA, Mar. 2007.
[10] K. W. Chiang, C. Q. Hua, and T. S. P. Yum, "Prefix-Randomized Query-Tree Protocol for RFID System," *IEEE Int'l. Conf. Commun.*, Istanbul, Turkey, June, 2006.
[11] P. Popovski, F. H. P. Fitzek, and R. Prasad, "Batch Conflict Resolution Algorithm with Progressively Accurate Multiplicity Estimation," *ACM DIALM-POMC*, Oct. 2004.
[12] J. Park, M. Y. Chung, and T. J. Lee, "Identification of RFID Tags in Framed-Slotted ALOHA with Tag Estimation and Binary Splitting," *Commun. and Electronics*, 2006.
[13] L. Zhu and P. T. S. Yum, "The Optimization of Framed Aloha based RFID Algorithm," *12th ACM-IEEE Int'l. Conf. Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Tenerife, Canary islands, Spain, Oct. 2009.
[14] L. Zhu and P. T. S. Yum, "Design and Analysis of Framed Aloha based RFID Anti-collision Algorithms," *IEEE Global Commun. Conf.*, Hawaii, USA, Dec. 2009.
[15] H. Vogt, "Efficient Object Identification with Passive RFID Tags," *1st Int'l. Conf., PERVASIVE 2002*, vol. 2414 of Lecture Notes in Computer Science (LNCS), Zurich, Switzerland, Aug. 2002, Springer-Verlag, pp. 98–113.

Biographies

TAK-SHING PETER YUM [S'76, A'78, SM'86] (tsyum@ie.cuhk.edu.hk) was born in Shanghai. He received primary and secondary school education in Hong Kong. He went to Columbia University and was awarded B.S., M.S., M.Ph., and Ph.D. degrees in 1974, 1975, 1977 and 1978 respectively. He joined Bell Telephone Laboratories in April 1978 working on switching and signaling systems for 2.5 years. Then, he taught at National Chiao Tung University, Taiwan for 2 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 and Technology Research Institute of Hong Kong (www.astri.org). He has published widely in Internet research with contributions to routing, buffer management, deadlock handling, message resequencing and multiaccess 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.

LEI ZHU [S'09] (lzhu6@ie.cuhk.edu.hk) received the B.S. degree in Electronic Information Science and Technology from University of Science and Technology of China in 2006. He is a Ph.D. candidate in 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.