This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2011 proceedings

A Tag Count Estimation Algorithm for Dynamic Framed ALOHA Based RFID MAC Protocols

Ilker Onat[1] and Ali Miri[2]

[1]School of Information Technology and Engineering, University of Ottawa, Canada
e-mail: ionat@site.uottawa.ca
[2]School of Computer Science, Ryerson University, Toronto, Canada
e-mail: samiri@scs.ryerson.ca

***Abstract*—The performance of RFID MAC algorithms is expressed in terms of the total time it takes by a reader to read all the tags in the reading range. One of the major MAC approaches for RFID systems is the framed slotted ALOHA based approach. It is proven that this class of algorithms achieve the smallest total reading time when, at each reading round, the frame length is set equal to the actual number of remaining unread tags. An important design challenge to increase the performance of this class of algorithms is to estimate the remaining tag count before each reading round. In this paper, we address this remaining tag count estimation problem for such RFID systems. The introduced algorithm is an a posteriori tag count estimation scheme from the collision statistics of the previous reading round. Our algorithm compares favorably to other a posteriori estimation algorithms. Another improvement brought by the algorithm is the reduction in reader side computational overhead of the previous schemes.**

I. Introduction

Radio frequency identification (RFID) systems provide an efficient alternative to optical barcodes for the requirements of the rapidly evolving automatic identification systems. The majority of RFID tags are passive devices powered by the reader emitted RF signals. The medium access control (MAC) schemes used in passive RFID tags are comparatively simpler than those used in active systems. Passive RFID tags cannot listen the channel for activity, hence they are dependent on the reader to initiate, manage and complete the reading process. There are two basic MAC approaches for RFID systems: *binary tree search methods* and *ALOHA-based channel access randomization* methods. In binary tree search based methods [1], simple or optimized polling procedures are used to query the tags. In general, binary tree search based algorithms are more wasteful in that they require a much bigger number of exchanges between the tags and the reader, compared to the framed slotted ALOHA based approach.

In slotted ALOHA based algorithms time is divided into slots and a station can only transmit at specific time slots. In framed slotted ALOHA [2] time slots are further grouped into frames, with N time slots per frame. In this protocol, each station can transmit only once in a given frame, on a randomly selected time slot. A collision occurs when multiple transmissions start within the same slot. In [3] authors analyzed the performance of the dynamic frame length ALOHA with capture. Various performance enhancement techniques are in use in practical ALOHA based systems. When a tag response is received, the acknowledgement from the reader can *mute* the tag to prevent the unnecessary responses, until the next activate command. A tag can be switched-off after being read, can be slowed-down with random back-off command or temporarily muted when another tag is transmitting. The same methods can be used more efficiently when slotted ALOHA is used. One of the earliest and widely used commercial examples of protocols built on framed slotted ALOHA is Philips I*Code [4]. The ALOHA based algorithms are also parts of the major RFID standards such as ISO 18000-6 [5]- [6] and ISO 14443-3 [7].

If there are more tags than the available slots, tags will be constrained and there will be excessive number of collisions requiring extra reading slots and reading rounds. When the frame size is unnecessarily larger than the tag count, many slots will be empty and minimum total reading time will not be achieved. The major performance parameter of all RFID MAC algorithms is the total time it takes by the reader to read all the tags in the reading range. This total time is expressed as the total reading slots used in all reading rounds. It is proven that framed slotted ALOHA based algorithms achieve the minimum total reading time, when, at each reading round, the frame length (reading slots count) is set equal to the actual number of remaining unread tags. Therefore for optimum performance, framed slotted ALOHA algorithms must adapt the frame size to the number of remaining tags at each reading round. Since a reader has no knowledge of actual remaining tag count before reading rounds, practical estimation algorithms must be available on the reader side.

In this paper, we address this remaining tag count estimation problem. The introduced algorithm is an a posteriori tag count estimation scheme from the collision statistics of the previous reading round. We compare the algorithm with other a posteriori estimation algorithms and demonstrate that it achieves minimum total reading time without requiring complex reader side calculations.

II. A posteriori Remaining Tag Count Estimation Algorithms

A posteriori tag count estimation algorithms introduced in [8]–[11] make use of the collision statistics in estimating the remaining tag count for the next round. Problem is formulated as follows: assume there are n tags to be read, and let the frame length be L time slots. The probability of finding k tags on a given slot is a binomial distribution with n Bernouilli experiments and $1/L$ occupation probability.

$$B(k) = \binom{n}{k} \left(\frac{1}{L}\right)^k \left(1 - \frac{1}{L}\right)^{n-k} \tag{1}$$

Then the empty (p_e), success (p_s) and collision (p_c) probabilities for a given slot are obtained as

$$p_e = B(0) = (1 - \frac{1}{L})^n \tag{2}$$

$$p_s = B(1) = \frac{n}{L}(1 - \frac{1}{L})^{n-1} \tag{3}$$

$$p_c = 1 - p_e - p_s \tag{4}$$

since $p_e + p_s + p_c = 1$.

With n tags to be read, the probability of finding a single tag on a given slot is given by p_s. With L slots, the expected value of number of singly occupied slots is

$$E[S] = L.p_s = n(1 - \frac{1}{L})^{n-1} \tag{5}$$

Channel usage efficiency is defined as the ratio of expected value of successful slots count to the number of total slots.

$$U = \frac{E[S]}{L} = \frac{n}{L}(1 - \frac{1}{L})^{n-1} \tag{6}$$

To maximize the channel usage efficiency, the first derivative is taken with respect to L and its roots are found.

$$\frac{dU}{dL} = \frac{n(n-L)(L-1)^{n-2}}{L^{n+1}} \tag{7}$$

The maximum U is then obtained with $n = L$, which means that the number of slots should be set equal to the actual number of tags for maximum channel utilization, or throughput.

A. Chen's Method

In Chen's method [9], the probability that among L slots, there are C slots with collisions, E slots without transmission and S slots with single transmission, is modeled as a multinomial distribution with L independent trials.

$$P(E, S, C) = \frac{L!}{E!S!C!} p_e^E p_s^S p_c^C \tag{8}$$

For a read cycle with L slots, the a posteriori probability for the number of tags n, with given E, S, and C.

$$\begin{aligned} P(n|E, S, C) &= \frac{L!}{E!S!C!} \\ &\times \left[\left(1 - \frac{1}{L}\right)^n\right]^E \left[\frac{n}{L}\left(1 - \frac{1}{L}\right)^{n-1}\right]^S \\ &\times \left[1 - \left(1 - \frac{1}{L}\right)^n - \frac{n}{L}\left(1 - \frac{1}{L}\right)^{n-1}\right]^C \end{aligned} \tag{9}$$

The decision rule proposed by Chen is to set the number of tags estimate equal to n that maximizes this probability, $P(n|E, S, C)$.

However, Chen's formulation does not take into account the fact that the outcomes of the L trials are not independent. The binomial distribution is the probability distribution of the number of successes in n independent Bernoulli trials, with the same probability of success in each trial. In the framed slotted ALOHA reading process, while performing L trials on a given round, the outcome of each trial, i.e., the placement of a tag on a slot randomly, effects the subsequent trials by changing the outcome probabilities.

B. Vogt's Method

Vogt's method [8] makes use of Chebyshevs inequality. Chebyshevs inequality states that the outcome of a random experiment involving a random variable is most likely somewhere near the expected value of it. Hence Vogt's method uses the n that minimizes the distance between read results and the expected values.

$$\varepsilon_{vd}(L, E, S, C) = min_n \left| \begin{pmatrix} a_0 \\ a_1 \\ a_m \end{pmatrix} - \begin{pmatrix} E \\ S \\ C \end{pmatrix} \right| \tag{10}$$

where a_0, a_1 and a_m are expected values of the number of empty, singly occupied and collision slots. They are defined as,

$$a_0 = LB(0) = L(1 - \frac{1}{L})^n, \tag{11}$$

$$a_1 = LB(1) = n(1 - \frac{1}{L})^{n-1}, \tag{12}$$

$$a_m = L - a_0 - a_1. \tag{13}$$

Note that for any algorithm, if the initial starting frame length is brought closer to the actual tag count, the performance of the algorithm will improve. However this knowledge is not available to the reader. Therefore the reader has to start from some appropriate initial slot count.

III. Proposed Algorithm

By checking the results of a reading round, we already know that there are at least $2C$ colliding tags that are waiting to be read over the next rounds. In order to approximate the actual number of unread tags we use this number as the lower bound. For a given reading round, let C_n represent the expected number of collision slots where n tags are involved. Then,

- The expected value of collision slots count, $E(C) = (C_2 + C_3 + ... + C_n)$
- The expected value of number of remaining tags after a reading round, $E(N_{remaining}) = (2C_2 + 3C_3 + ... + nC_n)$
- The expected value of the lower bound for remaining tags count is $2E(C) = (2C_2 + 2C_3 + ... + 2C_n)$
- The expected value of the unknown parameter in our experiment is the value of $E(N_{remaining}) - 2E(C) = (C_3 + 2C_4 + 3C_5 + ... + (n-2)C_n)$

Let's define this unknown parameter as A. We know that, the expected value of $C_n = L.B(n)$. Then the expected value of A is equal to

$$E(A) = (C_3 + 2C_4 + 3C_5 + ... + (n-2)C_n) = \sum_{i=1}^{n-2} L.i.\frac{n!}{(n-i-2)!(i+2)!}\left(\frac{1}{L}\right)^{i+2}\left(1-\frac{1}{L}\right)^{n-2-i} \quad (14)$$

For the optimum estimation case, i.e., when the number of tags is equal to number of slots ($n = L$), we plot the expected value of A in Figure-1. We observe that, $E(A)$ increases almost linearly with increasing L. In the same figure, we also plot the expected number of successful slots $E(S)$ and we observe a linear dependency between $E(S)$ and $E(A)$.

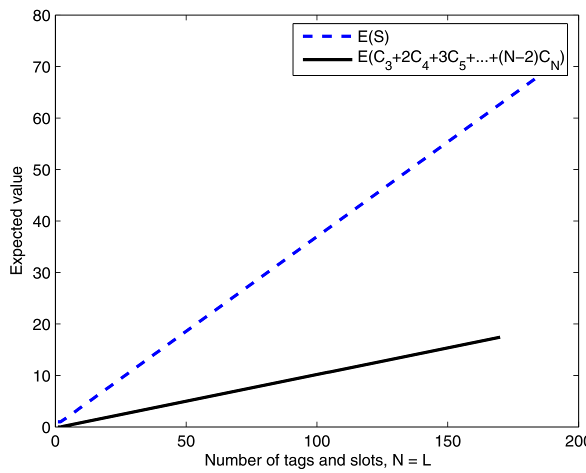

Fig. 1: Expected values of S and A

Therefore we propose to approximate $E(A)$ as a linear function of $E(S)$ using

$$E(A) \cong \frac{1}{3.5}E(S) \quad (15)$$

where the constant $1/3.5$ obtained numerically. When we set the remaining tag count to $N_{actual} = 2C + S/3.5$, we obtain very close performance results as the optimum oracle estimation, which sets the number of slots as equal to the actual number of remaining tags, as can be seen in Figure-2 and Figure-3. Since we now know this linear dependence on S, using trial an error, we balance the number of rounds and the number of slots required. We set the frame length L, as the remaining tag count estimate $N_{estimate} = 2C + S$, where S represents the successful slots count and C represents the collision slots count of the current round[1]. Our estimation algorithm results in lower estimation error, lower number of total slots and lower number of rounds when compared to Vogt's and Chen's methods without requiring complex minimization or maximization routines on the reader side.

The simulations are performed according to the Algorithm-1.

[1]Here we should emphasize that, if our tag count formulation was to be expressed as the current round's tag count estimation, it would be $N_{estimate} = 2C + 2S$. For the next round, since S tags are successful, the remaining tag count estimate will be $2C + S$.

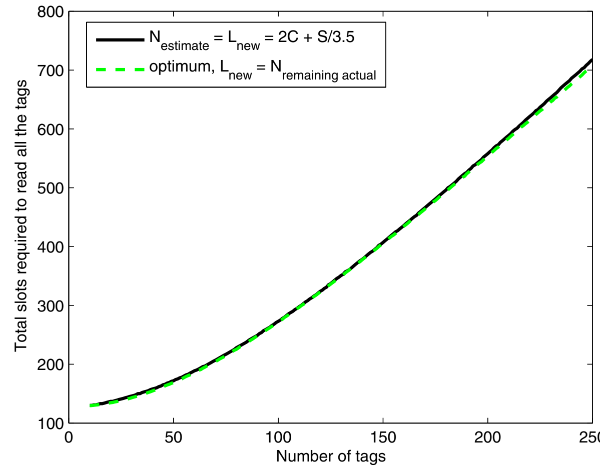

Fig. 2: Total slots required, $L_{new} = 2C + S/3.5$ vs optimum

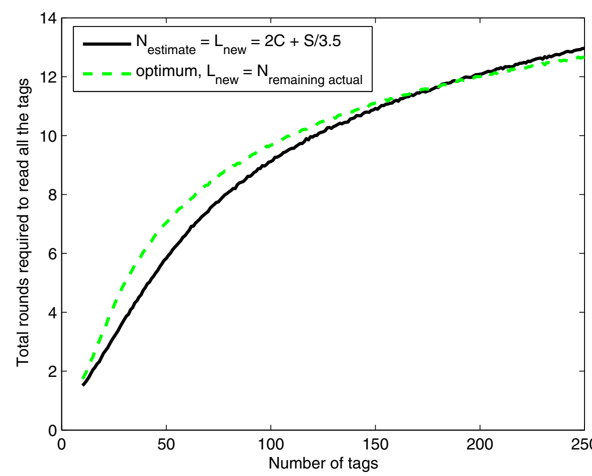

Fig. 3: Total rounds required, $L_{new} = 2C + S/3.5$ vs optimum

Algorithm 1 Remaining Tag Count Estimate

Initial frame size, L_{init};
Current frame size, $L_{current} = L_{init}$;
$totalRoundsCount = 0$;
$totalSlotsCount = 0$;
while (there are unread tags) **do**
  $totalRoundsCount++$;
  $totalSlotsCount = totalSlotsCount + L_{current}$;
  Initiate read cycle with the current frame size, $L_{current}$;
  //Every remaining tag randomly selects a slot
  Count empty (E), successful (S) and collision (C) slots
  // Remaining tag count estimate is then
  $N_{estimate} = L_{new} = 2C + S$;
  $L_{current} = L_{new}$;
  // repeat the cycle while there are unread tags
end while

IV. Simulation Results

We compare the following remaining tag count estimation algorithms where $N_{estimate}$ represents the remaining tag count estimate :

- Optimum, $N_{estimate} = N_{actual}$
- Lower bound, $N_{estimate} = 2C$
- Vogt's algorithm
- Chen's algorithm
- Our algorithm, $N_{estimate} = 2C + S$

A. Estimation Error

In this section we analyze the estimation error metric and the related tradeoffs. The normalized total number of errors in reading all the tags for a given number of actual tags is the sum of single read cycle errors. In certain cases, this error function can be used to judge the quality of the tag count estimator. Error in a single read cycle is defined as

$$error = \frac{|n - \hat{n}|}{n} \tag{16}$$

where $\hat{n}$ is the tag count estimation. The cumulative average estimation error for different number of actual slot counts is plotted in Figure-4.

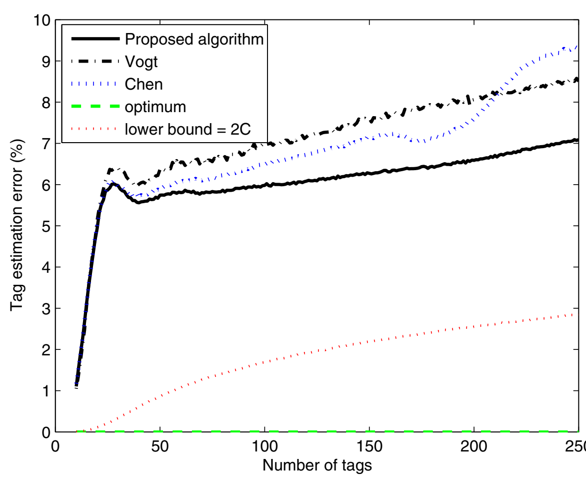

Fig. 4: Tag estimation error

When the results related to estimation error metric are analyzed, we can see that a smaller average estimation error does not always result in smaller reading rounds count. For example, the *lower bound* algorithm gives a small cumulative average estimation error, however it has a high total number of required rounds. The downside of large number of rounds even if the total reading time is small, is the signalling and processing overhead which is difficult to predict and model in simulations. In practice, this overhead can be high and algorithms that have small total slots count while keeping the total number rounds under control are preferable.

B. Total reading time and total reading rounds

The total reading time is defined as the total number of slots multiplied with the slot duration. Initial slot count doesn't change the relative performance of the algorithms. In our simulations we selected the initial slot count, L_{init} as 128. Optimum and lower bound algorithms have the minimum total slots required. However they have very high number of required rounds, close to three times the number required by other algorithms. Figure-5 shows the performance of our algorithm in terms of the total time slots required. Another important metric measuring the performance of tag count estimation algorithms is the total rounds required to read all the tags. Figure-6 compares the performance of our scheme with the previous algorithms. Again, the number of total rounds metric is important because each reading round carries a handshaking and control messaging overhead, whose severity depends on the underlying communications system design. In any case, small number of rounds is an important design goal for estimation algorithms.

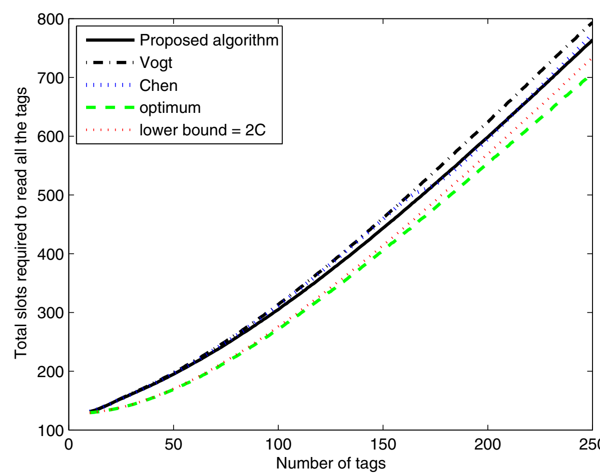

Fig. 5: Total slots required

Our algorithm achieves better results for every comparison parameter. It has a smaller tag estimation error. It finishes in a smaller number of total slots and a smaller number of total rounds compared to the other algorithms. The benefits of our algorithm are more pronounced when the computational overhead is considered. Both [8] and [9] have comparatively higher computation time on the reader side. In order to come up with an estimate, Chen's algorithm finds the tag count that maximizes a nonlinear function. This requires the evaluation of that function which includes four factorials, and six exponentiations, for every tag count candidate in the entire scope of the candidates. Likewise, Vogt's algorithm finds the tag count that minimizes a nonlinear function by evaluating that function for the entire scope. Vogt's function contains six exponentiation, while our algorithm requires a single multiplication and addition for an estimation.

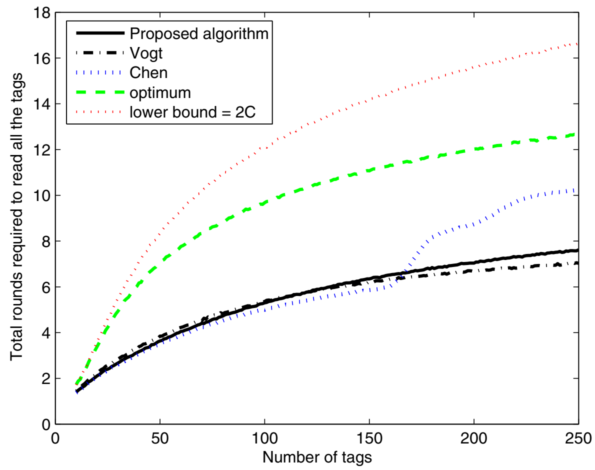

Fig. 6: Total rounds required

V. Conclusions

In this work, we introduced a new tag number estimation scheme. Our algorithm performs better than the existing a posteriori tag count estimation algorithms. It also eliminates complex reader side minimization and maximization calculations performed between reading rounds in the previous algorithms. Current major RFID standard parts that deal with the collision avoidance problem such as ISO 18000-6 and ISO 14443-3 does not specify frame length estimation method for framed slotted ALOHA based algorithms. Our algorithm can be efficiently used in practice.

References

[1] D. Hush and C. Wood, "Analysis of tree algorithms for RFID arbitration," in *the Proceedings of the IEEE International Symposium on Information Theory*, p. 107, 1998.

[2] F. Schoute, "Dynamic frame length ALOHA," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 565–568, 1983.

[3] J. Wieselthier, A. Ephremides, and L. Michaels, "An exact analysis and performance evaluation of framed ALOHA with capture," *IEEE Transactions on Communications*, vol. 37, no. 2, pp. 125–137, 1989.

[4] Philips Semiconductors, *Philips I*Code1 System Design, AN00025*. available at www.nxp.com/acrobat_download2/other/identification/SL048611.pdf.

[5] International Organization for Standardization (ISO), "Parameters for Air Interface Communications at 860 MHz to 960 MHz – Radio Frequency Identification for Item Management, ISO/IEC 18000-6," 2004.

[6] EPCGlobal, "Class 1 Generation 2 UHF Air Interface Protocol Standard," 2005.

[7] International Organization for Standardization (ISO), "Initialization and Anticollision Identification Cards – Contactless Integrated Circuit Cards – Proximity cards, ISO/IEC 14443-3," 2001.

[8] H. Vogt, "Multiple object identification with passive RFID tags," in *the Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, 2002.

[9] W.-T. Chen, "An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length ALOHA," *IEEE Transactions on Automation Science and Engineering*, vol. 6, no. 1, pp. 9–15, 2009.

[10] S.-R. Lee, S.-D. Joo, and C.-W. Lee, "An enhanced dynamic framed slotted ALOHA algorithm for RFID tag identification," in *the Proceedings of the Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, MobiQuitous'05*, pp. 166–172, 2005.

[11] G. Khandelwal, A. Yener, K. Lee, and S. Serbetli, "ASAP : A MAC protocol for dense and time constrained RFID systems," in *the Proceedings of the IEEE International Conference on Communications, ICC'06*, vol. 9, pp. 4028–4033.