A Novel Fast Tag Estimate Method for Dynamic Frame Length Aloha Anti-collision Algorithms in RFID System

Shuai Wang, Weijun Hong, Liang Yin, ShuFang Li
Beijing University of Posts and Telecommunications
Beijing, China
{chancy, hongweijun , lisf } @bupt.edu.cn

Abstract—**In an Radio Frequency Identification (RFID) system, Dynamic Frame Length Aloha (DFSA) has been widely used in anti-collision protocol due to its high efficiency. A fast and accurate estimate of tag population is critical to the performance of DFSA. In this paper, we propose a closed-form tag estimator which reduces greatly the computational complexity while maintains high accuracy. Also we propose a two-step estimate method which can further improve the estimate performance. Simulations show that the error of proposed tag estimator is less than 4%. Furthermore, the proposed tag estimator has significant advantage in lower complexity over existing estimate methods.**

Keywords- Radio Frequency Identification (RFID); anti-collision; Dynamic Frame Length Aloha (DFSA); tag estimator;

I. INTRODUCTION

Radio-frequency identification (RFID) is a automatic identification technology to identify objects wirelessly without line-of-sight [1]. Recently, RFID technology has been widely used in retail, inventory checking, e-passport and supply-chain management. An RFID system consists of readers and a certain amount of tags to be retrieved data from. Collisions will occur when multiple tags simultaneously reply their data to the reader. An efficient anti-collision protocol is of great importance to save bandwidth, energy and reduce identification delays in RFID system. Up to now, many anti-collision protocols have been proposed. These protocols can be classified into two groups: tree-based protocols and aloha-based protocols. Tree-based protocol divided a set of tags into two subsets in each step until there is only one tag left. Tree-based protocol is more suitable for passive RFID systems, for many reader queries and tag responses are needed which results in higher power consumption. On the other hand, aloha-based protocol is more popular for its simple to implement especially in passive RFID system with limited tag capabilities.

Aloha-based protocol has a series of variants such as Pure Aloha (PA), Slotted Aloha (SA), Framed Slotted Aloha (FSA) and Dynamic framed slotted Aloha (DFSA). Among those, DFSA is the most efficient one for that it tries to make the system operate in an optimal way by dynamically adjusting the frame size prior to each reading cycle. It has been proven that [2], for DFSA, maximum throughput can be achieved only when the frame size is equal to the number of tags. Hence an accurate tag estimate is required to improve identification efficiency.

There have been many studies on tag estimate. The simpler ones is less accurate, such as lower bound[3], S+2.39C[4], Zhen[5] and collision ratio[6]. These estimates compute the number of tags by predefined fix factors. As a result, the wider the tag range, the larger the error produced. The more accurate ones, such as Khandelwal[12], Vogt[7] and Chen[8], have better estimate performances than above mentioned. Vogt proposed an estimate function to minimize the distance between the read result and the expected value vector. Chen estimates the number of remaining tags in the RFID system after each interrogation using Bayes theory and searching for the number of tags which can maximize the posteriori probability. Chen has improved the performance of tag estimate greatly as shown in [8, Fig. 4-8].

While a more accurate tag estimate is expected, its computation complexity should not be neglect. Higher computation load often means higher costs and power consumption which is strictly constrained in real applications, especially mobile ones. We note that, either in Vogt's or Chen's estimate, the optimal value is acquired by searching in the whole range of the number of tags other than being calculated directly by an analytical formula. When the tag range is large, computation complexity will increase sharply. In this paper, we propose a closed-form tag estimator based on LMMSE so that the number of tags can be computed straightly instead of extremum searching. Then we propose a two-step estimate method which further improves the estimate performance by narrow the tag range. Furthermore, we compare the proposed tag estimator with existing ones.

The reminder of this paper is organized as follows. Section II presents a dynamic framed slotted aloha RFID system. Section III proposes a LMMSE tag estimator and derives its expression. Also a two-step estimate method is proposed in this section. Section IV provides simulations to demonstrate the performance of the proposed estimate. Finally, conclusions are drawn in Section V.

II. DYNAMIC FRAMED SLOTTED ALOHA PROTOCOL

In this section we briefly introduce the protocol of DFSA. In DFSA, the reader first initiates a read cycle include a request command with frame size parameter L, i.e. L time slots in the frame. Each tag randomly select one and only one slot to respond. For a given slot, there are only three possible outcomes: idle, success and collision, which means no tags, only one tag and two or more tags response in a slot respectively. These outcomes can be used to estimate the number of tags present at the beginning of current frame. If one or more collision slots exist, another frame is required to separate collision tags. The optimal size of the next frame can be determined by subtracting the number of successful slots from the estimate. Repeat this procedure in each cycle until there is no collision slot, i.e. all tags have been identified.

Consider a dynamic framed slotted aloha system with n tags to be read and a read cycle with a frame length L. For a given slot, the number of tags allocated in the slot is a binomial distribution with n Bernoulli experiments and $1/L$ occupied probability. The probability of finding r tags in the slot is therefore given by

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

The probabilities of idle, successful, and collision for the slot is

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

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

$$p_c = 1 - p_e - p_s \quad (4)$$

III. LMMSE TAG ESTIMATOR

A. Problem Formulation

We denote n, $\hat{n}$, L the true number of tags, the corresponding estimator and the frame size respectively. Let X_1, X_2, X_3 denote the number of slots with outcomes of idle, success and collision respectively in current frame. We assume that $\hat{n}$ has linear relationships with X_1, X_2, X_3 as follows

$$\hat{n} = c_1X_1 + c_2X_2 + c_3X_3 \quad (5)$$

Where c_1, c_2 and c_3 are undetermined coefficients. Since X_1 contribute nothing to $\hat{n}$, it is reasonable to set c_1 to 0. Also, c_2 should be set to 1 because each success slot contains only one tag. For X_3 equal to $L - X_2 - X_1$, (5) can be rewritten as

$$\hat{n} = X_2 + aX_3 = (1-a)X_2 - aX_1 + aL \quad (6)$$

In which a are undetermined coefficient.

Note that we have formulated tag estimate as a linear estimate problem which is easy to analysis. In order to get optimal $\hat{n}$ from X_1 and X_2, we need to determine the values of a using some statistical criterion. A natural one is MSE (mean square error), which simultaneously takes into account the bias and variance of the estimator. However, adoption of MSE generally leads to an unrealizable estimator due to the dependence between the optimal value and the unknown true value [9]. Here we use MMSE (minimize mean square error) instead, which avoids that problem by using of Bayes theory. We are interested in the following problem.

Definition: For the linear estimate denoted in (6), select a to minimize the following MMSE function

$$B_{mse}(\hat{n}) = E[(n-\hat{n})^2] \quad (7)$$

Where $E[.]$ is mathematical expectation operator.

In general, the linear model assumption is reasonable when the number of collisions slots X_3 is not very large. Ideally, we can expect a perfect linear model when X_3 =0 which means n $= X_2$, i.e. there are only success slots. Whereas when $X_3 = L$, the estimate performance will degrade greatly. This property also holds true for other estimate methods for hardly can we get any information about n when all slots are in collision state. In practice we can set the initial L to an appropriate value based on the range of n.

B. A general LMMSE estimator of the number of tags

We need to solve the problem defined in (7) and find the optimal value of $\hat{n}$. Substituting (6) into (7) we have

$$B_{mse}(\hat{n}) = a^2E[(L-X_1-X_2)^2] + 2aE[(L-X_1-X_2)(X_2-n)] + E[(X_2-n)^2] \quad (8)$$

Take the first partial derivative of a and let the derivative equal to zero, we have

$$a^{\#} = \frac{LE[n]-LE[X_2]-E[nX_1]+E[X_1X_2]-E[nX_2]+E[X_2^2]}{L^2+E[X_1^2]+E[X_2^2]-2LE[X_1]-2LE[X_2]+2E[X_1X_2]} \quad (9)$$

Substituting (9) into (6) we have

$$\hat{n} = X_2 + a^{\#}X_3 \quad (10)$$

We have got a general LMMSE estimator of the number of tags. As shown in (10) $\hat{n}$ is in closed-form. Therefore $\hat{n}$ can be directly calculated by statistics of n and $\vec{X}$ instead of taking a time consuming search.

C. Derivation of the proposed LMMSE estimator

The LMMSE estimator is based on Bayesian theory, which considers n as a random variable, not a deterministic one. Though it is hard to find the true distribution of n, its range or upper bound is usually known in practice or can be set to an empirical value. In fact, tag estimates [7][8] are also based on a limited range and search for the optimal value within it. With knowledge only the range and no inclination as to whether n should be in any particular value, using the same way [9] the author has suggested when denote MMSE, we assign a $U(n_{\min}, n_{\max})$ PDF to the random variable n. In (9), $E(X_1)$, $E(X_2)$, $E(nX_1)$, $E(nX_2)$, $E(X_1^2)$, $E(X_2^2)$ and $E(X_1X_2)$ need to be derived under the uniform assumption. To simplify the following derivation, we define $\Phi_1(x)$, $\Phi_2(x)$, $\Phi_3(x)$, Ψ, Ω, α, β, Γ and Θ in appendix A.

For a frame with L slots, number the slots in sequence from 1 to L. We define two random variables for l_{th} slot, i.e. $i_l \in \{0(not\ \ idle), 1(idle)\}$ and $s_l \in \{0(not\ \ success), 1(success)\}$.

The conditional expectation of i_l、 s_l are

$$E[i_l \mid n] = 0 + 1 \times P(i_l = 1 \mid n) = P_e = (1 - 1/L)^n \tag{11}$$

$$E[s_l \mid n] = 0 + 1 \times P(s_l = 1 \mid n) = P_s = (n/L)(1 - 1/L)^{n-1} \tag{12}$$

For $n \sim U(n_{\min}, n_{\max})$, its expectation is

$$E[n] = (n_{\max} + n_{\min})/2 = \Omega/2 \tag{13}$$

The expectation of X_1 can be derived as

$$\begin{aligned} E[X_1] &= \sum_{n=n_{\min}}^{n_{\max}} E[X_1 \mid n] P(n) \\ &= (1/\Psi) \sum_{n=n_{\min}}^{n_{\max}} \sum_{l=1}^{L} E[i_l \mid n] \\ &= (L/\Psi) \sum_{n=n_{\min}}^{n_{\max}} E[i_l \mid n] \\ &= L\Phi_1(\alpha)/\Psi \end{aligned} \tag{14}$$

Similarly, we have

$$E[X_2] = (1/\Psi) \sum_{n=n_{\min}}^{n_{\max}} \sum_{l=1}^{L} E[s_l \mid n] = \Phi_2(\alpha)/\Psi \tag{15}$$

$$E[nX_1] = \sum_{n=n_{\min}}^{n_{\max}} nE[X_1 \mid n] P(n) = L\Phi_2(\alpha)\alpha/\Psi \tag{16}$$

$$E[nX_2] = \sum_{n=n_{\min}}^{n_{\max}} nE[X_2 \mid n] P(n) = (\Phi_3(\alpha)\alpha + \Phi_2(\alpha))/\Psi \tag{17}$$

$$\begin{aligned} E[X_1^2] &= \sum_{n=n_{\min}}^{n_{\max}} E[X_1^2 \mid n] P(n) \\ &= (1/\Psi) \sum_{n=n_{\min}}^{n_{\max}} E[(i_1 + i_2 + \ldots + i_L)^2 \mid n] \\ &= (1/\Psi) \sum_{n=n_{\min}}^{n_{\max}} (LE[i_1^2 \mid n] + L(L-1)E[i_u i_v \mid n]) \quad (u \neq v) \end{aligned} \tag{18}$$

In which

$$E[i_1^2 \mid n] = 1^2 \times P(i_1 = 1 \mid n) = P(i_1 = 1 \mid n) \tag{19}$$

$$E[i_u i_v \mid n] = P(i_u = i_v = 1 \mid n) = P(i_v = 1 \mid i_u = 1, n) P(i_u = 1 \mid n) \tag{20}$$

We note that $P(i_v = 1 \mid i_u = 1, n)$ is the probability of observing one idle slot conditioned to having observed one idle slot. So it is equal to

$$P(i_v = 1 \mid i_u = 1, n) = (1 - 1/(L-1))^n \tag{21}$$

Then we have

$$E[X_1^2] = (L/\Psi)(\Phi_1(\alpha) + (L-1)\Phi_1(\beta)) \tag{22}$$

In the same way, we have

$$E[X_2^2] = (1/\Psi) \sum_{n=n_{\min}}^{n_{\max}} (LE[s_1^2 \mid n] + L(L-1)E[s_u s_v \mid n]) \tag{23}$$

In which

$$E[s_1^2 \mid n] = 1^2 P(s_1 = 1 \mid n) = P(s\ \ = 1 \mid n) \tag{24}$$

$$E[s_u s_v \mid n] = P(s_v = 1 \mid s_u = 1, n) P(s_u = 1 \mid n) \tag{25}$$

$$P(s_v = 1 \mid s_u = 1, n) = ((n-1)/(L-1))(1 - 1/(L-1))^{n-2} \tag{26}$$

Then we have

$$E[X_2^2] = (1/\Psi)(\Phi_2(\alpha) + \alpha\Phi_3(\beta)) \tag{27}$$

The last one is $E[X_1X_2]$, which can be written as

$$\begin{aligned} E[X_1X_2] &= \sum_{n=n_{\min}}^{n_{\max}} E[(i_1 + i_2 + \ldots + i_L)(s_1 + s_2 + \ldots + s_L) \mid n] \\ &= (1/\Psi) \sum_{n=n_{\min}}^{n_{\max}} L(L-1)E[i_u s_v \mid n] \qquad (u \neq v) \end{aligned} \tag{28}$$

In which

$$\begin{aligned} E[i_u s_v \mid n] &= P(s_v = 1 \mid i_u = 1, n) P(i_u = 1 \mid n) \\ &= (n/L)(1 - 2/L)^{n-1} \end{aligned} \tag{29}$$

So we have

$$E[X_1X_2] = (L-1)\Phi_2(\beta)/\Psi \tag{30}$$

Finally, using (8), (9) ~ (12), (17), (22), (25), a is acquired as

$$a = \Gamma/\Theta \tag{31}$$

D. A two-step estimate method

As defined in (7), $\hat{n}$ is the optimal value which can minimize the mean square error throughout the range of n. Intuitively, if we can narrow the range, a more accurate estimator will be achieved since more errors are excluded from the averaging process. From a large simulation, we find that the maximum error is no more than $2*\sqrt{B_{mse}(\hat{n})}$. So we can adopt a two-step estimate method as follows.

Step 1: Estimate the number of tags in its initial range $(n_{\min}, n_{\max})$ *and get the first estimator* $\hat{n}_f$.

Step 2: Narrow the range to $(\hat{n}_f - 2*\sqrt{B_{mse}(\hat{n})}, \hat{n}_f + 2*\sqrt{B_{mse}(\hat{n})})$ *and perform estimate once more to get the final estimator* $\hat{n}$.

It is a natural idea that three or more narrowing steps could do better. However, computational complexity also increases. The two-step estimate method is accurate enough as shown in section IV.

E. Computational complexity analysis

We analysis the computational complexity of the proposed estimate step by step in this section. Suppose $n \sim U(0, n_{\max})$, $\alpha^{n_{\max}}$, $\beta^{n_{\max}}$ need to be computed prior to others, for they are the key components in other factors. Clearly the total necessary multiplications for them are $2n_{\max}$. After simple calculation, we obtain approximately the number of necessary multiplications is, respectively, 4, 7, 20, 9 and 11 for $\Phi_1(x)$, $\Phi_2(x)$, $\Phi_3(x)$, Γ and Θ. The total number of necessary multiplications in step 1 is $2n_{\max}$ +51. For the second step $2(n_f + 2*B_{mse}(\hat{n}))$ +51 multiplications are needed. Since $B_{mse}(\hat{n}) << n_f$ and $n_f \approx n_{\max}/2$ on average, we have $n_f + 2*B_{mse}(\hat{n}) \approx n_{\max}/2$ averagely. So the total number of necessary multiplications in step 2 is $n_{\max}$ +51 approximately. Finally the total necessary multiplications are approximately $3n_{\max}$+101. In addition, the total number of additions is nearly 40, which can be omitted at all.

The MAP estimate proposed in [8] by Chen may be the most accurate one among the existing methods. It has

$$P(n \mid E,S,C) = (L!/E!S!C!)p_e^{\ E} p_s^{\ S} p_c^{\ C} \qquad (32)$$

In which E,S,C is the number of slots of idle, success and collision respectively, and p_e, p_s, p_c has been defined in (2), (3) and (4). The algorithm finds $\hat{n}$ which can maximize $P(n \mid E,S,C)$. Obviously, (32) needs to be calculated $n_{\max}$ times in whole searching process. Omitting the calculation of p_e, p_s, p_c and the fix factor $E!S!C!$, it still needs nearly $E+S+C=L$ multiplications in each searching. Hence totally it needs $L \times n_{\max}$ multiplications. Since $0 < L \le n_{\max}$, it needs $n^2{}_{\max}/2$ multiplications averagely. Its complexity is much higher than that of proposed when $n_{\max} > 30$. When $n_{\max}$ is larger, the advantage of lower computation complexity of our proposed estimator will be more significant.

Another accurate estimate is the minimum squared error (MSE) by Vogt [7]. The estimation function used by Vogt is as follows:

$$\hat{n} = \arg\min((a_0 - E)^2 + (a_1 - S)^2 + (a_c - C)^2) \qquad (33)$$

Where a_0, a_1, a_c are the expected values of the number of empty slots, singly occupied slots, and collision slots, respectively. They can be calculated as follows:

$$a_0 = L \cdot p_e = L(1-1/L)^n$$

$$a_1 = L \cdot p_s = n(1-1/L)^{n-1}$$

$$a_c = L \cdot p_c = L - L(1-1/L)^n - n(1-1/L)^{n-1} \qquad (34)$$

The estimate in (33) also finds the optimal value of $\hat{n}$ in the range of n. It needs about 7 multiplications in each search and $7n_{\max}$ multiplications in whole. Although it has the same level of complexity to the proposed estimate, its computation load will be much higher when $n_{\max} > 100$. Moreover, our proposed estimate provides more precise performance than the Vogt method as shown in section IV.

IV. SIMULATION

In this section we show the simulations performed to verify the proposed estimator. In Fig.1, frame length was set to L =128 slots. The estimate errors of S+0.39C, Khandelwal, Vogt, LMMSE and Chen's MAP are compared in Fig.1. Results indicate that the LMMSE estimator has an error of no more than 4%. The error of Chen's MAP is a little less than the proposed. Though the computation complexity was dramatically decreased by the LMMSE, its maximum difference to Chen's MAP is no more than 1%. Vogt gives a pretty good estimate as LMMSE and Chen's when n is around 128. However, its error increases more quickly when n goes away from 128 and exceeds 6% at n =30. The error of

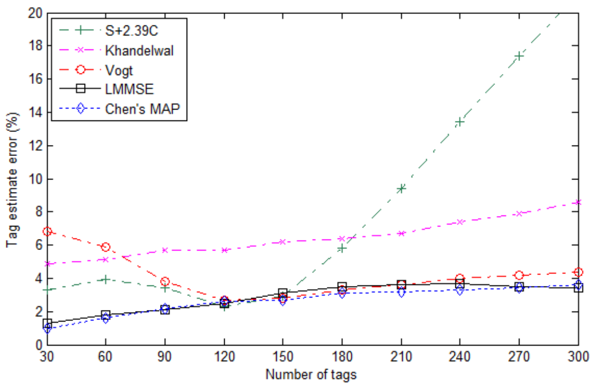

Figure 1. Simulation resultes for tag estimate error

Khandelwal is above 4% throughout the whole range of thenumber of tags. S+2.39C presents the worse performance among these five. It is shown in Fig.1 that when the number of tags exceed 100 the error of S+2.39C raise to a higher level far from others. The reason is that S+2.39C is a fix factor estimate which is not appropriate when the number of collision slots is large.

In Fig.1, LMMSE, Chen's MAP and Vogt all show their accurate and stable estimate with lower error ratio. Now we compare these methods in view of computation complexity. Fig.2 shows computation complexity for LMMSE, Chen's MAP and Vogt. For a larger $n_{\max}$, the computation complexity of Chen's MAP increase dramatically which is linearly with $n^2_{\max}$. The multiplications of Vogt estimate increase linearly with tag number and reduce the computation complexity a lot compared to Chen's MAP. However, our proposed method achieve a lower computation complexity about 50% than Vogt's. It is clearly the proposed method is more computationally efficient than the others.

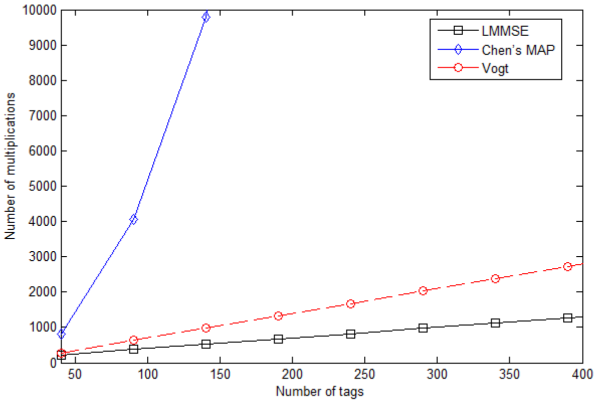

Figure 2. Simulation resultes for computation complexity

V. CONCLUSION

We have presented a general LMMSE estimator in close form. Based on it, a real LMMSE estimator is derived. In our method, the number of tags can be calculated through a closed form expression only once. It was shown that the proposed estimator has reduced significantly the computation complexity which is the main shortcoming of other estimates using statistic method. Moreover, its estimate error is less than 4% which is at the same level of performance as Chen's MAP. Therefore, the proposed estimator is particularly suitable for applications which have limited computational ability and power supply.

APPENDIX A

$$\alpha = 1 - 1/L \tag{35}$$

$$\beta = 1 - 2/L \tag{36}$$

$$\Phi_1(x) = \sum_{n=n_{\min}}^{n_{\max}} x^n = \frac{x^{n_{\min}}(1 - x^{n_{\max} - n_{\min} + 1})}{1 - x} \tag{37}$$

$$\begin{aligned}\Phi_2(x) &= \sum_{n=n_{\min}}^{n_{\max}} n x^{n-1} \\ &= \frac{x^{n_{\min}} - x^{n_{\max}+1}}{(1-x)^2} + \frac{x^{n_{\max}}(n_{\max}+1) - n_{\min} x^{n_{\min}-1}}{x-1}\end{aligned} \tag{38}$$

$$\begin{aligned}\Phi_3(x) &= \sum_{n=n_{\min}}^{n_{\max}} n(n-1)x^{n-2} \\ &= \frac{n_{\max}(n_{\max}+1)x^{n_{\max}-1} - n_{\min}(n_{\min}-1)x^{n_{\min}-2}}{1-x} \\ &\quad - \frac{2(x^{n_{\min}} - x^{n_{\max}+1})}{(x-1)^3} - \frac{2(x^{n_{\max}}(n_{\max}+1) - n_{\min} x^{n_{\min}-1})}{(x-1)^2}\end{aligned} \tag{39}$$

$$\Psi = n_{\max} - n_{\min} + 1 \tag{40}$$

$$\Omega = n_{\max} + n_{\min} \tag{41}$$

$$\begin{aligned}\Gamma = \frac{\Omega L}{2} + \frac{1}{\Psi}[&-L\Phi_2(\alpha)(1+\alpha) + (L-1)\Phi_2(\beta) \\ &-\alpha\Phi_3(\alpha) + \alpha\Phi_3(\beta)]\end{aligned} \tag{42}$$

$$\begin{aligned}\Theta = L^2 + \frac{1}{\Psi}[&L(1-2L)\Phi_1(\alpha) + (1-2L)\Phi_2(\alpha) + \\ &L(L-1)\Phi_1(\beta) + 2(L-1)\Phi_2(\beta) + \alpha\Phi_3(\beta)]\end{aligned} \tag{42}$$

REFERENCES

[1] K. Finkenzeller, RFID Handbook , 2nd ed. New York: Wiley, 2003.

[2] S.-R. Lee, S.-D. Joo, and C.-W. Lee, "An enhanced dynamic framed ALOHA algorithm for RFID tag identification," in Proc. Int. Conf. Mobile and Ubiquitous Systems: Networking and Services , 2005.

[3] Parameters for Air Interface Communications at 860 MHz to 960MHz Information Technology—Radio Frequency Identification forItem Management, 2004, ISO/IEC 18000-6, Part 6.

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

[5] B. Zhen, M. Kobayashi, and M. Shimizu, "Framed ALOHA for multiple RFID objects identification," IEICE Trans. Commun. , vol. E80-B,no. 3, pp. 991–999, Mar. 2005.

[6] J.-R. Cha and J.-H. Kim, "Novel anti-collision algorithms for fast objectidentification in RFID system," in The 11th Intl. Conference on Paralleland Distributed Systems, (Korea), pp. 63–67, 2005.

[7] H. Vogt, "Efficient object identification with passive RFID tags," inProc. Int. Conf. Pervasive Computing , 2002, pp. 98–113.

[8] W. T. Chen, "An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic framelength ALOHA," IEEE Trans. Autom. Sci. Eng. , vol. 6, no. 1, pp. 9–15,2009.

[9] S. M. Kay, Fundamentals of Statistical Signal Processing: Detection Theory . Upper Saddle River, NJ: Prentice-Hall, 1993, vol. 1.

[10] L. Roberts, "Extensions of packet communication technology to a handheld personal terminal," in Proc. Spring Joint Computer Conf. , 1972,pp. 295–298.

[11] C. Floerkemeier, "Transmission control scheme for fast RFID objectidentification," in Proc. Int. Conf. Pervasive Computing Communica-tions Workshops , 2006.

[12] G. Khandelwal, A. Yener, K. Lee, and S. Serbetli, "ASAP: a MACprotocol for dense and time constrained RFID systems," in IEEE Inter-national Conference on Communications (ICC'06), (Istanbul, Turkey),2006.