This paper was presented as part of the Mini-Conference at IEEE INFOCOM 2011

Efficient Missing Tag Detection in RFID Systems

Wen Luo† Shigang Chen† Tao Li† Shiping Chen‡
† Department of Computer & Information Science & Engineering, University of Florida, Florida, USA
‡ Network Information Center, University of Shanghai for Science and Technology, Shanghai, China

***Abstract*—RFID tags have many important applications in automated warehouse management. One example is to monitor a set of tags and detect whether some tags are missing — the objects to which the missing tags are attached are likely to be missing, too, due to theft or administrative error. Prior research on this problem has primarily focused on efficient protocols that reduce the execution time in order to avoid disruption of normal inventory operations. This paper makes several new advances. First, we observe that the existing protocol is far from being optimal in terms of execution time. We are able to cut the execution time to a fraction of what is currently needed. Second, we study the missing-tag detection problem from a new *energy* perspective, which is very important when battery-powered active tags are used. The new insight provides flexibility for the practitioners to meet their energy and time requirements.**

I. Introduction

RFID (radio frequency identifier) technologies promise to revolutionize future inventory management [1], [2], [3], [4]. They are used to replace the barcode system such that the ID that identifies an object can be accessed wirelessly over a distance. Today's passive RFID tags harvest energy from a reader's radio waves and use such minute amount of energy to power their circuits and deliver information back to the reader. They operate in a range of several feet to tens of feet. However, obstacles that are pervasive in a typical warehouse environment will shorten the range. The semi-passive tags carry batteries to power the circuit, but rely on backscattering to transmit information. The active tags use internal power to transmit, and consequently do not need any energy supply from the reader. They operate at a much further distance, making them particularly suitable for automated inventory management in a large warehouse, where one or a few RFID readers are installed to access all tagged objects and carry out various management functions automatically.

Most existing work on RFID systems is to design protocols that read the IDs from the tags [5], [6], [7], [8], [9], [10], [11], [12], [13]. Other work designs time-efficient protocols to estimate the number of tags in a large RFID system [14], [15], [16], [11], [17]. This paper studies a much less explored function that is very useful in inventory management. Consider a warehouse that stores a large number of commercial products or a military base that stockpiles a large quantity of guns and ammunition packages. Suppose each object (e.g., a microwave oven, a rifle, or a bullet magazine) is attached with a RFID tag. Now, if some objects are stolen from the warehouse, how to timely detect such events? Without any automatic tools, we have to resort to manual inventory walk-through, which is laborious, expensive, and slow. Such operations cannot be performed frequently, and hence will not help us detect the theft in time in order to catch the thieves. However, if RFID tags are installed, the missing-tag detection can be automated and performed frequently so that any major theft is swiftly reported.

Tan, Sheng and Li recently carried out the pioneer research on automatic RFID-based detection of missing-tag events [18]. The paper focuses on reducing the execution time of their missing-tag detection protocol. A follow-up protocol [19] allows a RFID reader to pinpoint exactly which tags are missing but it requires more execution time. In this paper, we make three major contributions. First, we observe that the protocol in [18] is far from being optimal in terms of time efficiency. We are able to cut the execution time to a fraction of what it needs. Our solution keeps the online operations very simple, which is important for practical RFID systems.

Second, we observe that time efficiency should not be the sole performance consideration when designing a missing-tag detection protocol. The energy cost may be an even more critical concern if active tags are used. Due to limited operational distances, passive tags are mostly used for small-range applications such as fast checkout. For future automatic inventory management functions that cover a very large area, active tags are likely to be the choice. Active tags use their own power to transmit. A longer range can be achieved by transmitting at a higher power. They are also richer in resources for implementing advanced functions. Their price becomes less of a concern if they are used for expensive merchandizes (such as refrigerators) or reused many times as goods moving in and out of the warehouse. But active tags also have a problem. They are powered by batteries. Recharging batteries for tens of thousands of tags is a laborious operation, considering that the tagged products may stack up, making tags not easily accessible. To prolong the tags' lifetime and reduce the frequency of battery recharge, all functions that involve large-scale transmission by many tags should be made energy-efficient. The prior work has studied energy-efficient protocols for estimating the number of tags in a RFID system [20], or energy-efficient anti-collision protocols that minimize the energy consumption of a mobile reader when the reader is used to collect the IDs of the tags [21], [22]. We believe this paper is the first to study energy-efficient solutions for the missing-tag detection problem.

Third, we reveal a fundamental energy-time tradeoff that is controlled by a pair of parameters in our protocol design. Better energy efficiency can be achieved at the expense of longer execution time, or vice versa. In fact, the protocol in [18] is a special case of our protocol under a specific parameter assignment. It has the worst energy cost among all parameter assignments. It is not time-efficient, either. We can choose parameter values that result in far smaller energy cost as well as smaller execution time.

II. System Model

A. Missing-tag Detection Problem and Assumption

Consider a large RFID system of n tags. Each tag carries a unique ID and has the capability of performing certain computations and communicating with the RFID reader wirelessly. The problem is to design efficient protocols for a RFID reader to detect whether some tags are missing. We have the same *detection requirement* as [18] for designing a protocol to solve this problem: A single execution of the protocol should detect a missing-tag event with a probability at least α if m or more tags are missing, where α and m are two system parameters.

We assume that the RFID reader has access to a database that stores the IDs of all tags. This assumption is necessary, and it is also made in [18]. We cannot simply collect the IDs directly from the tags at the moment when we execute a missing-tag detection protocol. RFID systems use a slow communication channel. Collecting the IDs of all tags is time-consuming. Moreover, the missing tags will not send over any information. Without any prior knowledge of their existence, we will have no way to know their absence.

The above assumption can be easily satisfied if the tag IDs are read into a database when new objects are moved into the system and they are removed from the database when the objects are moved out — this is what a typical inventory management procedure will do. Even if such information is lost due to a database failure, we can recover the information by executing an ID-collection protocol [5], [6], [12], [13] to read the IDs from the tags. In this case, we will not detect missing-tag events that have already happened. However, now that we have the IDs of the remaining tags, we can detect the missing-tag events after this point of time, not through an expensive ID-collection protocol but through a more efficient protocol to be proposed in this paper.

B. Performance Metric 1: Protocol Execution Time

Imagine that a large retailer has a warehouse in its distribution center that regularly stores tens of thousands of electronics, furniture, apparel, shoes, pallets, cases, etc. A missing-tag detection protocol is expected to be executed frequently (e.g., once every 15 minutes) in order to timely raise alarms upon unexpected removal of objects from the warehouse. However, false detection of missing tags may occur if normal operations remove objects during the time when the protocol is executing. In a busy warehouse, as goods constantly move in and out, false alarms may happen even if the protocol's execution time is a number of seconds. Hence, it is highly desirable that the execution time is kept as small as possible in order to minimize the disruption to normal inventory operations.

The execution time is affected by two major factors. The first factor is how stringent the system requirement is. A protocol such as [18] will take far more time to detect one missing tag with 99.9% probability than to detect 100 missing tags with 95%. Hence, in order to control the protocol's execution time, a practical system may be configured with $m = 100$ and $\alpha = 95\%$. It means that a single protocol execution will detect the missing-tag event with 95% if $m = 100$. Because the protocol is executed periodically, after the ith execution, the detection probability becomes $1 - (1 - 95\%)^i$, which rapidly approaches to 100% when i becomes large. Even if the number of missing tags is smaller than m and thus the detection probability of one protocol execution is smaller than α, the missing-tag event will eventually be detected after a sufficient number of executions.

The second factor that has a major impact on execution time is the protocol design. In this paper, we show that a better protocol design can reduce the execution time considerably without any significant increase in the complexity of the online operations.

C. Performance Metric 2: Energy Cost

In order to support advanced management functions that cover a large area, battery-powered active tags are a better choice because they have much longer transmission ranges. If passive tags were used, one would have to take the RFID reader and move around the whole area, collecting information from location to location, or else a dense reader array has to be installed to extend the coverage. Active tags allow one or a few readers to collect information from a large area.

When active tags are used, we must conserve their battery power in order to prolong the tags' lifetime before they have to be recharged. The tagged goods (such as apparel) may stack in piles, and there may be obstacles, such as racks filled with merchandize, between a tag and the reader. We expect the active tags are designed to transmit with significant power that is high enough to ensure reliable information delivery in such a demanding environment. The energy consumed by the RFID reader is of less concern because its batteries can be easily recharged or it may even use an external power source. We assume that the reader transmits at sufficiently high power.

D. Time Slots

Communications between the reader and the tags are time-slotted. The reader's signal will synchronize the clocks of the tags. In some protocols, the communication is driven by the reader in a request-and-response pattern. The reader issues a request, which is followed by a time frame consisting of f slots, during which the tags may respond by transmitting some information.

A slot is said to be *empty* if no tag responds (transmits) in the slot. It is called a *singleton slot* if exactly one tag responds. It is a *collision slot* if more than one tag responds. A singleton or collision slot is also called a *busy slot*. The Philips I-Code system [23] requires a slot length of 10 bits in order to distinguish singleton slots from collision slots. On the contrary, one bit is enough if we only need to distinguish empty slots from busy slots — '0' means empty and '1' means busy. Hence, tag responses will be much shorter (or consume much less energy) if a protocol only needs to know empty/busy slots, like the one in this paper does.

A frame takes $f \times t_s$ time, where t_s is the time of a slot. When f is large, the time it takes the RFID reader to transmit its request, which is a small constant, can be ignored. For time efficiency, we should minimize the frame size f, subject to the detection requirement in Section II-A. For energy efficiency, we should minimize the total number of responses from all tags. Because there are empty slots and collision slots, the number of responses is not the same as the number of time slots in the frame. As we will see later, reducing the number of responses may require us to increase the number of time slots in order to meet the detection requirement.

Let t_{id} be the time it takes to transmit a tag ID. Obviously, $t_{id} > t_s$ because it takes a longer time to transmit multiple bits in an ID than one-bit information in a tag response. Based on the specification of the Philips I-Code system [23], we determine that $t_{id} = 2.4$ms for a 96-bit tag ID and $t_s = 0.4$ms, after the required waiting times (e.g., gap of idle time between transmissions) are included.

III. Prior work

The most related work we find in the literature is the Trusted Reader Protocol (TRP) by Tan, Sheng and Li [18]. For security reasons, their system consists of a server and a RFID reader. The former stores the tag IDs and performs the computation, while the latter communicates with the tags. We assume the reader is trusted. To simplify the protocol description, we logically combine the server and the reader into a single entity, still called the reader.

A design goal of TRP is to reduce the time of the detection process. To initiate the execution of the protocol, a RFID reader broadcasts a polling request, asking the tags to respond in a time frame of f slots. The polling request has two parameters, the frame size f and a random number r. Each tag maps itself to a slot in the frame by hashing its ID and r. It then transmits during that slot. The reader records which slots are busy and which are empty. This is binary information where each slot carries either '1' or '0'. Multiple readers may be used to extend the coverage. In this case, all readers will simultaneously monitor the time slots in the frame. A slot is considered to be busy if any reader records that the slot is busy.

Because the reader knows the IDs of all tags, it knows which tags are mapped to which slots. More specifically, it knows which slots are expected to be singletons, i.e., one and only one tag is mapped to each of them. If an expected singleton slot turns out to be empty, the tag that is mapped to this slot must be missing. Not all tags are mapped to singleton slots. When two or more tags are mapped to the same slot (a collision slot), if only one of the tags is missing, the slot will remain busy and thus the missing-tag event will not be detected.

Obviously, if we increase the frame size, collision will be less likely and there will be a larger number of singleton slots, which means the probability for any tag to map to a singleton is greater. In the event of missing tags, the probability that an expected singleton turns out to be empty is also greater, and hence the probability of detecting the missing-tag event is greater. The requirement is that we must detect the missing-tag event with a probability of at least α if m or more tags are missing. TRP is designed to minimize its execution time by using the smallest frame size that ensures a detection probability of α.

A serious limitation of TRP is that it only considers time efficiency. It is not energy-efficient because all tags must transmit during the time frame. Moreover, what is less obvious is that TRP is not optimal in terms of execution time, either. In the following sections, we propose a new solution that drastically reduces energy consumption, as well as execution time.

IV. An Intermediate Protocol

We present an energy-efficient protocol that serves as an intermediate design step towards our final solution in the next section.

A. Protocol Description

It is well known that a small group of people much fewer than 365 can have a high probability to contain two who celebrate their birthdays on the same day. This is called the *birthday paradox*. Similarly, two relatively small subsets of the tags in a large RFID system can also have a large probability of sharing a common tag. Let M be the set of m missing tags, and K be a subset of k tags that the reader randomly selects from the inventory list N of n tags currently in the system. The reader performs a simple operation to verify the presence of these k tags. It transmits the IDs of these tags one after another. After transmitting an ID, the reader waits for a short period and listens for a response. When a tag receives its ID, it will acknowledge its presence by sending a response. If the reader does not receive any response back for an ID, it reports the missing tag event. The idea is that if k is reasonably large, K and M will have a good chance to share at least one common tag. In other words, the reader will find that the presence of at least one tag in K cannot be positively confirmed. Hence, the missing-tag event is detected. This intermediate protocol is energy-efficient because there are overall $k(\ll n)$ tag responses, instead of n responses required by TRP. Note that the energy consumption by the reader for transmitting the IDs is a secondary concern.

B. Limitations

We observe that the intermediate protocol has much room for improvement. First, it is not time-efficient. Although only a subset of tag IDs is selected, it takes a considerable amount of time to verify the presence of each selected tag. According to the parameters of the Philip system [23], it takes about 2.4 ms to transmit a tag ID of 96 bits, and it takes 0.4 ms for a tag to respond with one bit acknowledgement after receiving its ID. In comparison, it takes just 0.4 ms for a RFID reader to identify each empty or busy slot in the frame used by TRP.

Second, although only a subset of tags transmit and each of them only transmits once, they have to listen to the channel for their IDs, which means that their circuits have to be continuously powered to receive up to k IDs. It is true that transmitting is likely to consume much more power than receiving if the same number of bits are involved. However, in our intermediate protocol, each selected tag makes just one short transmission, but it has to receive a large number of bits, which makes the aggregate receiving energy significant.

In our next protocol, for better time efficiency, we make sure that no tag IDs are transmitted. For better energy efficiency, we make sure that each tag only needs to receive at most one polling request.

V. Efficient Missing-tag Detection Protocol (EMD)

A. Protocol Description

We propose our final solution that addresses the limitations of the intermediate protocol in the previous section. The RFID reader initiates the protocol execution by broadcasting a polling request. Upon receipt of the request, each tag decides with a sampling probability p whether to participate in the polling. If it decides to participate, it will randomly select a slot in the subsequent frame to respond. If it decides not to participate, it will simply enter the sleep mode and wake up at the next schedule time for the protocol execution. All decisions are made pseudo-randomly and predictable by the reader.

The polling request consists of three parameters: a frame size f, a random number r, and an integer $x = \lceil p \times X \rceil$, where X is a large, pre-configured constant (e.g., 2^{16}). After receiving the request, a tag performs a hash $H_1(id, r)$, where id is the tag's ID and H_1 is a hash function whose range is $[0, X)$. If $H_1(id, r) < x$, the tag will participate in the polling; otherwise, it won't.

If a tag decides to participate in the polling, it performs another hash $H_2(id, r)$ in the range of $[0, f)$ to determine in which slot of the time frame it will transmit. The tag will then transmit at the $H_2(id, r)$th slot.

The RFID reader has the IDs of all tags, from which it can derive all information that is needed to know which tags will participate in the polling and, if so, at which slots they will transmit. Hence, it knows exactly which slots in the frame will be empty and which are expected to be busy. At the end of the frame, if the reader finds that a slot that is supposed to be busy turns out to be empty, it knows that the tag(s) that is expected to transmit in the slot must be missing. In this case, the reader reports a missing-tag event. When multiple synchronized readers are used to extend the coverage, we treat a slot as busy if any reader records that the slot is busy.

B. Energy-time Tradeoff and TRP

The performance of the above EMD protocol is controlled by two parameters, p and f. Given a sampling probability p, we can find the optimal frame size $f^*(p)$ that minimizes the protocol's execution time. We then vary the sampling probability to investigate how the performance of EMD changes under different values of p. The results are shown by Fig. 1 in the form of an energy-time tradeoff curve, where the vertical axis is the protocol execution time and the horizonal axis is the energy cost measured by the total number of tag transmissions (which is expected to be $n \times p$). This tradeoff is controlled by the sampling probability p. TRP [18] corresponds to the special case of $p = 1$; it is represented by a black dot in the figure. Clearly, $p = 1$ is not a good choice for either energy efficiency or time efficiency. In fact, it is the worst in terms of energy cost. The sampling probability that minimizes the execution time is denoted as p_t. Due to space limitation, we omit the processes for computing $f^*(p)$ and p_t.

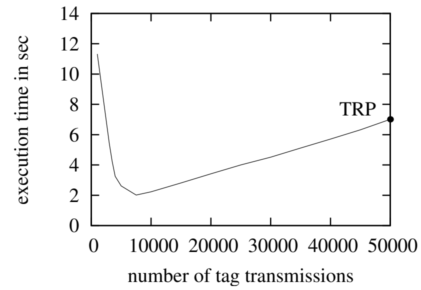

Fig. 1. Protocol execution time with respect to energy cost $n \times p$, where $n = 50,000$, and p varies from 0 to 1.

VI. Numerical Results

A. Simulation Setting

We have performed extensive simulations to study the performance of the proposed EMD and compare it with the most related work, TRP [18]. The performance evaluation is carried out under various different sets of system parameter values, including m, α, and n.

For each set of parameters, TRP will compute its optimal frame size [18]. Once the frame size f is determined, the execution time is known, which is $f \times t_s$. The energy cost of TRP is n responses. Similarly, for each set of parameters, EMD will choose a sampling probability p and compute the optimal frame size f^* which minimizes the execution time of EMD under that sampling probability. The energy cost of EMD is $n \times p$ responses, and the execution time is $f^* \times t_s$.

B. Performance Comparison

In this section, we compare the performance of EMD with a sampling probability p_t, under which the smallest execution time is achieved, and TRP under different values of m, α and n. The left plot of Fig. 2 compares the energy cost of these two protocols under different values of n when $m = 100$ and $\alpha = 90\%$. The right plot compares their execution times under the same setting. EMD uses less than one fortieth of the energy that TRP uses, while its execution time is less than one third of the time by TRP.

In Fig. 3-4, we make the same comparison under different α values. The results show that when we increase the detection probability, the energy/time gains by EMD are somewhat reduced, but remain significant, particularly for energy.

In Fig. 5-6, we keep $\alpha = 95\%$ and vary the value of m. The results show that the energy/time gains by EMD increase when m increases. However, if m is small, the gain in execution time shrinks, while the gain in energy cost remains large.

VII. Conclusion

This paper proposes a new protocol for detecting missing-tag events. In addition to improving time efficiency, the protocol

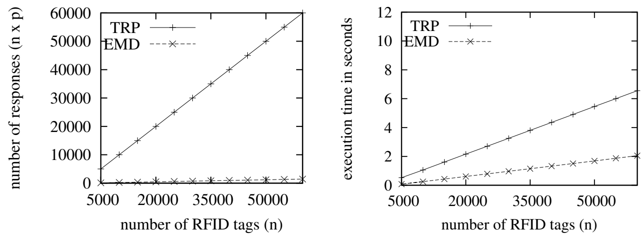

Fig. 2. • *Left plot:* The number of tag responses with respect to the number of tags, when $m = 100$ and $\alpha = 90\%$. • *Right plot:* The protocol execution time with respect to the number of tags, when $m = 100$ and $\alpha = 90\%$.

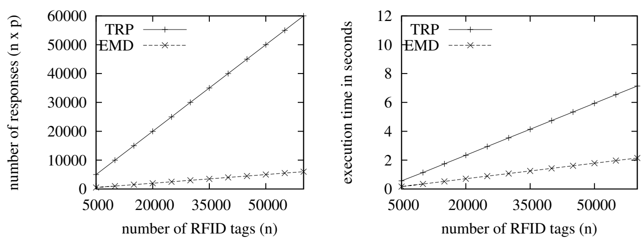

Fig. 3. • *Left plot:* The number of tag responses with respect to the number of tags, when $m = 100$ and $\alpha = 95\%$. • *Right plot:* The protocol execution time with respect to the number of tags, when $m = 100$ and $\alpha = 95\%$.

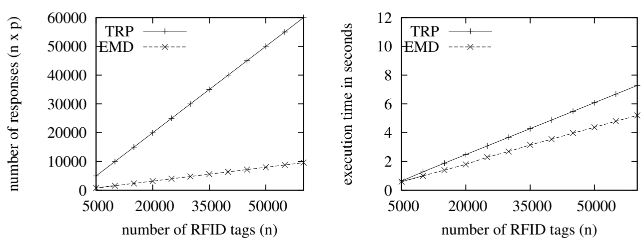

Fig. 4. • *Left plot:* The number of tag responses with respect to the number of tags, when $m = 100$ and $\alpha = 99\%$. • *Right plot:* The protocol execution time with respect to the number of tags, when $m = 100$ and $\alpha = 99\%$.

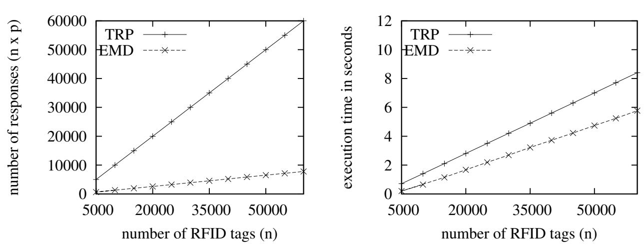

Fig. 5. • *Left plot:* The number of tag responses with respect to the number of tags, when $m = 50$ and $\alpha = 95\%$. • *Right plot:* The protocol execution time with respect to the number of tags, when $m = 50$ and $\alpha = 95\%$.

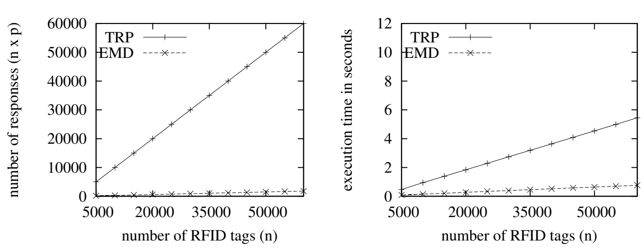

Fig. 6. • *Left plot:* The number of tag responses with respect to the number of tags, when $m = 200$ and $\alpha = 95\%$. • *Right plot:* The protocol execution time with respect to the number of tags, when $m = 200$ and $\alpha = 95\%$.

also puts energy efficiency into consideration, which is very important when active tags are used. The design of the protocol allows energy-time tradeoff, which is the first of its kind. If we choose the operating point on the tradeoff curve that achieves the minimum execution time, the new protocol will not only have a much smaller execution time than the existing work, but also have a much smaller energy cost, often by an order of magnitude.

VIII. Acknowledgments

This work was supported in part by the US National Science Foundation under grant CPS-0931969.

References

[1] L. Ni, Y. Liu, and Y. C. Lau, "Landmarc: Indoor Location Sensing using Active RFID," *Proc. of IEEE PerCom*, 2003.

[2] B. Sheng, Chiu C. Tan, Q. Li, and W. Mao, "Finding Popular Categories for RFID Tags," *Proc. ACM MOBIHOC*, May 2008.

[3] R. Das, "Global RFID Market Tops $5.5 Billion," *http://www.convertingmagazine.com/article/CA6653688.html*, April 2009.

[4] Q. Yao, Y. Qi, J. Han, J. Zhao, X. Li, and Y. Liu, "Randomizing RFID Private Authentication," *Proc. of IEEE PerCom*, 2009.

[5] J. Myung and W. Lee, "Adaptive Splitting Protocols for RFID Tag Collision Arbitration," *Proc. of ACM MOBIHOC*, 2006.

[6] N. Bhandari, A. Sahoo, and S. Iyer, "Intelligent Query Tree (IQT) Protocol to Improve RFID Tag Read Efficiency," *Proc. of IEEE ICIT*, 2006.

[7] F. Zhou, C. Chen, D. Jin, C. Huang, and H. Min, "Evaluating and Optimizing Power Consumption of Anti-collision Protocols for Applications in RFID Systems," *Proc. of ISLPED*, 2004.

[8] "Information technology automatic identification and data capture techniques C radio frequency identification for item management air interface - part 6: parameters for air interface communications at 860-960 MHz," *Final Draft International Standard ISO 18000-6*, November 2003.

[9] S. Lee, S. Joo, and C. Lee, "An Enhanced Dynamic Framed Slotted ALOHA Algorithm for RFID Tag Identification," *Proc. of IEEE MOBIQUITOUS*, 2005.

[10] J. R. Cha and J. H. Kim, "Dynamic Framed Slotted ALOHA Algorithms Using Fast Tag Estimation Method for RFID Systems," *IEEE Consumer Communications and Networking Conference (CCNC)*, January 2006.

[11] H. Vogt, "Efficient Object Identification with Passive RFID Tags," *Proc. of IEEE PerCom*, 2002.

[12] V. Sarangan, M. R. Devarapalli, and S. Radhakrishnan, "A Framework for Fast RFID Tag Reading in Static and Mobile Environments," *The International Journal of Computer and Telecommunications Networking*, vol. 52, no. 5, 2008.

[13] B. Zhen, M. Kobayashi, and M. Shimizu, "Framed ALOHA for Multiple RFID Objects identification," *IEICE Transactions on Communications*, Mar 2005.

[14] M. Kodialam and T. Nandagopal, "Fast and Reliable Estimation Schemes in RFID Systems," *Proc. of ACM MOBICOM, Los Angeles*, 2006.

[15] M. Kodialam, T. Nandagopal, and W. Lau, "Anonymous Tracking using RFID tags," *Proc. of IEEE INFOCOM*, 2007.

[16] J. Zhai and G. N. Wang, "An Anti-Collision Algorithm Using Two-functioned Estimation for RFID Tags," *Proc. of ICCSA*, 2005.

[17] C. Qian, H. Ngan, and Y. Liu, "Cardinality Estimation for Large-scale RFID Systems," *Proc. of IEEE PerCom*, 2008.

[18] C. Tan, B. Sheng, and Q. Li, "How to Monitor for Missing RFID Tags," *Proc. of IEEE ICDCS*, June 2008.

[19] T. Li, S. Chen, and Y. Ling, "Identifying the Missing Tags in a Large RFID System," *Proc. of ACM Mobihoc*, 2010.

[20] T. Li, S. Wu, S. Chen, and M. Yang, "Energy Efficient Algorithms for the RFID Estimation Problem," *Proc. IEEE INFOCOM*, March 2010.

[21] V. Namboodiri and L. Gao, "Energy-Aware Tag Anti-Collision Protocols for RFID Systems," *Proc. of IEEE PerCom*, 2007.

[22] D. Klair, K. Chin, and R. Raad, "On the Energy Consumption of Pure and Slotted Aloha based RFID Anti-Collision Protocols," *Computer Communications*, 2008.

[23] Philips Semiconductors, "I-CODE Smart Label RFID Tags," *http://www.nxp.com/acrobat_download/other/identification/SL092030.pdf*, Jan 2004.