Arbitrarily Accurate Approximation Scheme for Large-Scale RFID Cardinality Estimation

Wei Gong*, Kebin Liu*, Xin Miao*, Haoxiang Liu*†
*School of Software, TNLIST, Tsinghua University, China
†Department of Computer Science and Engineering, Hong Kong University of Science and Technology
Email:{gongwei, kebin, miao, haoxiang}@greenorbs.com

***Abstract*—One important issue of RFID applications is to estimate the cardinality of large-scale RFID tags in the interested region. From a practical perspective, we require: (i) the estimate can be arbitrarily accurate, and (ii) its time cost should be scalable with the tags size, regardless of the tags distribution. Existing solutions, however, either assume the use of hash functions with ideal random properties, or impose unacceptable computation/storage overhead for tags. More importantly, those approaches only give asymptotic results and fail to provide rigorous bounds for the rate of convergence. In this paper, we propose a new scheme, Arbitrarily Accurate Approximation (A^3), to reliably estimate the number of tags with any desired accuracy. In particular, for a given requirement of (ε, δ), we show that A^3 achieves $\mathcal{O}((\log\log n + \varepsilon^{-2})\log\delta^{-1})$ time efficiency. Results show that A^3 significantly outperforms previous designs under various distributions of tags.**

I. Introduction

In recent years, driven by the rising application demands of radio-frequency identification (RFID) technology, RFID tags are widely deployed in many industries and real life applications, such as object tracking [1][2], inventory control [3][4][5] and activity monitoring [6][7]. Owning to the easy deployment and low manufacturing cost, tags are utilized for massive object management and are dramatically growing in number all over the world, in about an order of 10 billion [8].

Estimating the cardinality of the tag set is of great importance in many RFID applications, such as warehouse management, tag identification and privacy sensitive RFID systems. Imagine a huge warehouse of large retailer like Wal-Mart, thousands of mobile phones, ipods, apparel and other office supplies are intensively piled [9]. A recent report indicates that employee theft, customer shoplifting, administrative error and vendor error cost retailers in the United States and Europe nearly $70 billion dollars annually [10]. Therefore, it is tempting to quickly and accurately estimate the number of those tagged objects for daily or weekly inventory reports, instead of laborious and unreliable humanly counting. Important applications also exist in other settings, such as counting the number of tourism or conference attendees with RFID tickets/cards. Furthermore, a common prerequisite in tag identification algorithms [11][12][13] is to estimate the cardinality of tags to be identified, which is used to set the optimal frame size. In some privacy sensitive scenarios, such as those used at the custom for continuously monitoring the number of people in several interested area. The unique identification information on tag, such as driver licenses [14] and e-passport [15], cannot be revealed. Therefore readers have to use the non-identifiable information sent by tags to compute the cardinality.

978-1-4799-3360-0/14/$31.00 © 2014 IEEE

Most existing estimation schemes for RFID systems can be classified into two categories, identification-based approaches and probabilistic approaches. In identification-based algorithms [16][17][18], the reader has to schedule the contending tags and ensure that each tag transmits in its own slot without collisions. Although those anti-collision schemes can successfully authenticate each tag and count the exact cardinality, the time of identifying all tags grows linearly with the number of tags. They are clearly not scalable for large-scale RFID systems. As a matter of fact, knowing the approximate cardinality of tags with desired accuracy is adequate in many applications. Therefore, probabilistic approaches are introduced to estimate the number of tags [12][13][19][20].

Let n be the number of tags. Qian et al. [19] achieve estimation efficiency with $\mathcal{O}(\log n)$ time slots. An even faster $\mathcal{O}(\log\log n)$ algorithm is introduced by Zheng et al. [20]. While recent work on this problem mainly focus on time-efficiency improvements, the accuracy of estimators is not well investigated. Unfortunately, since most prior work rely on Central Limit Theorem (CLT) to derive bounds for accuracy requirements, there are two deficiencies in their bounds. The one problem is that their estimation results are only asymptotic, therefore they fail to specify how quickly the estimation process converges. The other issue is that their estimations apply only to deviations on the order of the deviation of original estimator (one-time estimator). But actual system or user specified accuracy requirements call for arbitrary deviations. Meanwhile, those work have following shortcomings from other perspectives. First, they assume a random oracle that [21][22] can generate random uniform distribution as needed. A random oracle is a mathematical function mapping every possible query to a random response from its output domain. Nevertheless, random oracles are merely a mathematical abstract and cannot be implemented by any real function. Second, some of existing work resort to MD5/SHA-1 hash functions that can generate approximate uniform distribution as an alternative to random oracle. Unfortunately, this may impose heavy computation or preloaded storage overhead on resource limited tags, especially passive tags. The last but not the least, some schemes improperly assume the magnitude of cardinality as a prior knowledge.

To address above issues, we propose a new mechanism, Arbitrarily Accurate Approximation (A^3), to reliably estimate the number of tags with arbitrary accuracy requirement. For a

given requirement of accuracy level ε and error probability δ, A^3 achieves $\mathcal{O}((\log\log n+\varepsilon^{-2})\log\delta^{-1})$ estimation efficiency, as well as achieves arbitrary accuracy level with probability at least $\frac{11}{20}$ in a single round. Based on A^3 we design a scalable general protocol for large-scale RFID systems.

To the best of our knowledge, we are the first to propose arbitrarily accurate estimation scheme for large-scale RFID systems. The major contributions of this work are as follows.

1) We propose an arbitrarily accurate approximation scheme for cardinality estimation in large-scale RFID systems. For a given requirement of accuracy, it acheives sub-linear time-efficiency.
2) Compared with previous approaches, A^3 significantly reduces computation overhead and requires no extra preloaded storage on RFID tags.
3) We design a general protocol for readers and tags to accurately approximate the cardinality of tags. We validate the effectiveness and performance of A^3 through meticulous theoretical analysis and extensive simulations.

II. System Model

A. Problem and Assumption

The RFID system typically consists of several RFID readers and a number of RFID tags. Each tag is attached with a unique identification information (tagID) and can perform simple computations as well as communications by backscattering reader's RF signals. Consider there are N tags in the interested area. The aim of approximating the cardinality of tags is to acquire the quantity of RFID tags in the interested region, meeting arbitrary specified accuracy requirement. In particular, accuracy requirement contains two essential parameters, relative error ε, and error probability δ. Assume the approximation result is $\hat{N}$, then the relative error is derived as $\frac{|\hat{N}-N|}{N}$. We define that an (ε,δ) approximation scheme for N is a probabilistic process that, given any $0<\varepsilon<1$ and $0<\delta<1$, the result estimate $\hat{N}$ is within a relative error ε with probability at least $1-\delta$. This definition can also be rewritten as

$$\Pr[|\hat{\mathrm{N}}-\mathrm{N}|\leq\varepsilon\mathrm{N}]\geq 1-\delta.$$

For example, if the exact quantity of RFID tags is 1000 and user sets relative error ε as 0.01 and error probability δ as 0.01, then the output estimate of a (ε,δ) scheme, $\hat{N}$, should be between 990 to 1010 with more than 0.99 probability.

Typically there are two types of tags: (1) active tag, which often has its own rechargeable batteries for power supply and has a reading distance between 150 feet to 300 feet; (2) passive tag, which captures the energy in the reader's RF signal and has a communication range less than 20 feet. Nearly all deployed RFID systems around the world involve at least one or more types of tags described above. In order to support the management of large quantities of goods (often in thousands), an array of readers (or called multiple-reader) is introduced to cover a large area, extending much more longer operation range than a single reader [23]. In this paper, we treat an array of those coordinated readers as a virtual reader.

B. ALOHA Model in RFID

we assume frame-slotted ALOHA model in the RFID system similar to previous work [2][13][12]. We adopt the *Reader Talks First* mode, which is widely used in many applications [24][20]. In this model, the reader first initializes the communication and then wait for tags' responses in each slot. If there is no response in this slot, the slot is called *empty slot*. And if any tag responds in this slot, the slot is called *non-empty slot*. Obviously, the reader need only one bit to encode this simple response, using '1' for busy signal and '0' for idle. Furthermore, in some situations the reader may need to distinguish *singleton slot*, that only receives one tag's reply, from *collision slot* that contains responses from more than one tag. Long-bit response thus can be used to discern these two types of non-empty slot. To design a time efficient estimation scheme, short response is preferred. In the design and evaluation of our A^3 scheme, we therefore mainly focus on empty-slot and non-empty slot.

III. Motivation

A. Prior Art

Cardinality estimation is an important research problem in RFID systems. Many efforts have been made from several aspects, such as time-efficiency[20], energy-efficiency [25] and privacy preserving [11]. The accuracy issue, however, has not received much attention. Most existing RFID estimation schemes share a similar pattern as follows. Without any loss of generality, there is a well-designed estimator X to approximate N tags, with expectation $E[X]=\mu=N$ and variance $Var[X]=\delta^2$. Usually the result of one-time estimation can not satisfy the user specified accuracy requirements ε_u and δ_u. Consequently, they define the random process $\bar{X}=\frac{1}{f}\sum_{i=1}^{f}X_i$ in which $\bar{X}$ stands for the average value of f independent and identically distributed (i.i.d) estimates. Therefore, they can have

$$\mu'=E[\bar{X}]=\mu,\delta'^2=Var[\bar{X}]=(\frac{\delta}{\sqrt{f}})^2.$$

We can see that the advantage of this random process is that the expectation value is keeping unchanged but the standard deviation is reduced by $\sqrt{f}$ times. Then by Central Limit Theorem, they derived that as $f\to\infty$, $\frac{\bar{X}-\mu'}{\delta'}$ approaches standard normal distribution $N(0,1)$. That is to say, when $f\to\infty$, for any fixed $\tau>0$, we have

$$Pr[|\bar{X}-\mu'|>\tau\delta']\to\frac{1}{\sqrt{2\pi}}\int_{\tau}^{\infty}e^{-\frac{t^2}{2}}dt\approx\frac{1}{\sqrt{2\pi}\tau}e^{-\frac{\tau^2}{2}}.$$

Therefore, they put the accuracy requirement ε_u and δ_u into above equation to deduce how many rounds of estimation are needed, i.e., the value of f.

Unfortunately, there are two major deficiencies in their bounds meeting the desired accuracy. First, the above result is only asymptotic [26], i.e., the above result holds only as f approaches infinity. But in practical applications, we need to deal with finite f. And the above result says nothing about the rate of convergence for the estimator. Second, the above result fails to apply to arbitrary deviation, as the deviation δ' has to be the order of δ.

TABLE I. MAIN NOTATIONS

Symbols	Descriptions
N	exact cardinality of RFID tags
$\hat{N}$	estimated cardinality of RFID tags
S	RFID tag set
$[D]$	tagID domain $\{1, ..., D\}$
C	constant factor estimation
Z_{tagID_i}	leading zeros number of hashing $tagID_i$
k	constant approximation factor
f	# of independent rounds
m	# of t-wise independent hash functions

B. Design Principles

There are several essential principles of an excellent approximation scheme for large-scale RFID cardinality estimations. First, the accuracy requirement should be rigorously guaranteed. It means that the relative error of the approximation scheme should be arbitrarily small and theoretically bounded by the error probabilities, instead of asymptotic results. Second, implementable and practical hash functions should be accounted into the theoretical analysis of the estimator. Although the random oracle model is convenient for analysis, but it also incurs discrepancies between practical implementation and pure abstract model. Third, the estimation process should be as efficient as possible, that means the estimation time should not increase too much as the quantity of RFID system drastically scales. The last but not the least, the burden of tags should be as lightweight as possible.

IV. A^3 DESIGN

The A^3 design mainly consists of two steps. The first step is to obtain $C = \Theta(N)$, that is constant approximation ratio estimation. The second step is to refine C to $(1 \pm \varepsilon)$ approximation. In section IV.A, we present a constant factor approximation algorithm and its theoretical analysis. Then a $(1 \pm \varepsilon)$ approximation scheme and corresponding theoretical bounds are detailed in section IV.B. The section IV.C presents how to boost the success probability. The general communication protocols used by reader and tags are described in Section IV.D. We have a short discussion in section IV.E. Table I summarizes notations used across this paper.

A. Constant factor approximation algorithm

The cores of methods adopted in [19][20] are based on FM-Sketch [27]. First we briefly describe the FM-Sketch approximation algorithm and then give theoretical analysis to prove that all FM-Sketch based estimations are constant factor approximations.

Let S be a set of N tags, $\{tagID_1, ...tagID_N\}$. Each tag is associated with its own tagID. For simplicity, we assume that there is an ideal uniformly hash function $h : [0, D] \rightarrow [0, 2^w - 1]$. Let $\lceil w = \log N \rceil$, that is $2^w \geq N$. We use Z_{tagID_i} to denote the number of leading zeros (leftmost zeros) in the binary form of $h(tagID_i)$ and use Z^{max} to denote maximum leading zeros of hash values for all i in S . Therefore, the basic algorithm approximates the cardinality of S as

$$C = 2^{Z^{max}}. \tag{1}$$

For example, we assume that a tag set $S = \{2, 4, 6, 8\}$, $w = 2$, and hash values are $\{0, 1, 2, 3\}$. According to that $h(2) = 0 = (00)_2$, we can obtain $Z_2 = 2$. Likewise, $Z_4 = 1, Z_6 = 0, Z_8 = 0$, hence $Z^{max} = 2$. Finally this estimation result is $C = 2^{Z^{max}} = 4$.

Now, we give detailed theoretical analysis to show that the output C is off by N at most a constant factor.

Definition Let r be an integer between 0 and w. And k is a positive integer, r_1 is the smallest r such that $2^r > kN$, and r_2 is the smallest r such that $2^r \geq \frac{N}{k}$.

Lemma 1: $Pr[Z_{tagID_i} \geq r] = 2^{-r}$.

Proof: In above lemma, $Z_{tagID_i} \geq r$ means that hash value $h(tagID_i)$ of $tagID_i$ is between 0 and $2^{w-r} - 1$. Since the hash value $h(tagID_i)$ is uniformly distributed in range of $[0, 2^w - 1]$, we can get

$$Pr[Z_{tagID_i} \geq r] = \frac{2^{w-r}}{2^w} = 2^{-r}. \tag{2}$$

■

Definition Given any specific r, for each $tagID_i \in S$, we define

$$x_{tagID_i}(r) = \begin{cases} 1 & \text{if } Z_{tagID_i} \geq r \\ 0 & \text{if } Z_{tagID_i} < r \end{cases}$$

and $X(r) = \sum_{tagID_i \in S} x_{tagID_i}(r)$.

By Lemma 1, we know that $x_{tagID_i}(r)$ takes 1 with probability 2^{-r}, hence the expectation and variance of it are as follows

$$E[x_{tagID_i}(r)] = 2^{-r} \tag{3}$$

$$Var[x_{tagID_i}(r)] = 2^{-r}(1 - 2^{-r}). \tag{4}$$

Lemma 2: $Pr[X(r_1) > 0] < \frac{1}{k}$.

Proof: By definition of r_1 and (3),

$$E[X(r_1)] = \sum_{tagID_i \in S} E[x_{tagID_i}(r_1)] = N \cdot 2^{-r_1} < \frac{1}{k}.$$

Therefore, by Markov inequality, we have

$$Pr[X(r_1) > 0] = Pr[X(r_1) \geq 1] \leq E[X(r_1)] < \frac{1}{k}$$

. ■

Lemma 3: $Pr[X(r_2) = 0] < \frac{2}{k}$.

Proof: Likewise, we can obtain

$$E[X(r_2)] = N2^{-r_2}.$$

Since $X(r_2)$ is the sum of independent variables and each of which has variance $2^{-r}(1 - 2^{-r})$, we know

$$Var[X(r_2)] = N \cdot 2^{-r_2} \cdot (1 - 2^{-r_2}) < N2^{-r}.$$

Hence, by Chebyshev inequality, we know

$$\begin{aligned} Pr[X(r_2)=0] &= Pr[E[X(r_2)]-X(r_2)=E[X(r_2)]] \\ &\le Pr[|E[X(r_2)]-X(r_2)|=E[X(r_2)]] \\ &\le Pr[|E[X(r_2)]-X(r_2)|\ge E[X(r_2)]] \\ &\le \frac{Var[X(r_2)]}{(E[X(r_2)])^2} \\ &< \frac{N2^{-r_2}}{(N2^{-r_2})^2} \\ &= \frac{2^{r_2}}{N}. \end{aligned}$$

By definition of r_2, we know that $2^{r_2} < 2 \cdot \frac{N}{k}$. Otherwise, r_2 cannot be the smallest r satisfyting $2^r \ge \frac{N}{k}$. Combining this and above inequality proves that $Pr[X(r_2)=0] < \frac{2}{k}$. ■

Theorem 1 (Constant Approximation Bound): For any $k > 3$, $Pr[\frac{1}{k} \le \frac{C}{N} \le k] \ge 1 - \frac{3}{k}$

Proof: First, we show that if $X(r_1) = 0$ and $X(r_2) \ne 0$ the above theorem is correct. If $X(r_1) = 0$, it means that there is no $tagID_i \in S$ can give $Z_{tagID_i} \ge r_1$, and thus $Z^{max} < r_1$. Likewise, if $X(r_2) \ne 0$, it means that there is at least one $tagID_i \in S$ can satisfy $Z_{tagID_i} \ge r_2$ and thus $Z^{max} \ge r_2$. And also, according to the definition of r_1, r_2 and Z^{max}, we can derive that if $r_2 \le Z^{max} < r_1$, the above theorem is correct.

By lemma 2 and lemma 3, we know $X(r_1) \ge 1$ can happen with probability at most $\frac{1}{k}$, whereas $X(r_2) = 0$ can happen with probability at most $\frac{2}{k}$, thus the union bound of two events happening is at most $\frac{3}{k}$. Therefore, the probability of having '$X(r_1) = 0$ and $X(r_2) \ne 0$' is at least $1 - \frac{3}{k}$. ■

It is also worth noting that there are many FM-Sketch based counting solutions which had made several variations and extensions, such as in [19] the ideal hash function was substituted by MD5 hash function, and in [20] longest matching prefix of specific pattern replaced leading zeros counting. However, as their similarity to core parts of above basic constant factor approximation algorithm, the accuracy of those approaches is still rigorously bounded by theorem 1. Their corresponding proofs are omitted due to the space constraints.

B. Arbitrarily accurate approximation algorithm

From former sub-section, we obtain a constant factor estimation $C = \Theta(N)$. By introducing a "balls and bins" approach, we design an arbitrarily accurate approximation algorithm to refine C to $(1 \pm \varepsilon)$ accuracy level. The key intuition is that when randomly hashing N balls into C bins, the probability that the specific one bin (such as the first bin) is empty, is high concentrated about its expectation. Thus we form this expectation as a function of N and then inverting provides a good estimation of N with high probability.

Assume that we have a completely random hash function $h_c : [D] \to [C]$. If we randomly hash N balls into C bins, the probability that the first bin is non-empty should be

$$q = Pr[h_c^{-1}(0) \cap S \ne \emptyset] = 1 - (1 - \frac{1}{C})^N. \quad (5)$$

Then the following lemma shows that if we can obtain a good estimate $\hat{q}$ that is close enough to real q, then inverting equation 5 can produce an ε approximation of N.

Lemma 4: Let $k > 3$ and $\varepsilon > 0$. And suppose C and N are such that $\frac{1}{2k} \le \frac{N}{C} \le \frac{1}{2}$. Then if $|q - \hat{q}| \le \lambda = min(\frac{1}{e} - \frac{1}{3}, \frac{\varepsilon}{6k})$, then a new estimate $\hat{N}$, defined as

$$\hat{N} = \frac{\ln(1-\hat{q})}{\ln(1-\frac{1}{C})} \quad (6)$$

satisfies $|\hat{N} - N| \le \varepsilon N$

Proof: We prove this by using some well-known bounds and a little calculus. As $C \ge 2N$, hence $C \ge 2$ and $\frac{1}{C} \le \frac{1}{2}$. Also we have $(1-x) \ge e^{-2x}$ when $x \le \frac{1}{2}$. Therefore

$$1 - \frac{1}{C} \ge e^{-\frac{2}{C}} \Rightarrow q = 1 - (1 - \frac{1}{C})^N \le 1 - e^{-\frac{2N}{C}} \le 1 - \frac{1}{e}.$$

By definition, $\lambda \le \frac{1}{e} - \frac{1}{3}$, thus $q + \lambda \le \frac{2}{3}$, we can obtain

$$\frac{1}{1-(q+\lambda)} < 3. \quad (7)$$

And $\ln(1-x) + x < 0$ when $x < 1$, as $C > 1$ we have

$$-\frac{1}{\ln(1-\frac{1}{C})} \le C. \quad (8)$$

The calculus we use is that for any continuous function there is $|f(x) - f(\bar{x})| \le \varepsilon |sup_{y \in (x,\bar{x})} f\prime(y)|$ if $\bar{x}$ is close to x. Hence, for $f(x) = \ln(1-x)$, we know that

$$|\ln(1-x) - \ln(1-\bar{x})| \le \frac{|x-\bar{x}|}{max(1-x, 1-\bar{x})}. \quad (9)$$

Combing (7), (8), and (9), it gives that

$$\begin{aligned} |\hat{N} - N| &= \frac{|\ln(1-q) - \ln(1-\hat{q})|}{-\ln(1-\frac{1}{C})} \\ &\le C \cdot \frac{|q-\hat{q}|}{max(1-q, 1-\bar{q})} \\ &\le C \cdot \frac{|q-\hat{q}|}{1-(q+\lambda)} \le 3R\lambda \\ &\le 3 \cdot 2kN \cdot \frac{\varepsilon}{6k} = \varepsilon N. \end{aligned}$$

■

According to lemma 4, we know that approximating q is a good way to estimate the cardinality of tags. However, the the ideal hash function h_c are not known to be constructible efficiently. Therefore we choose to employ t-wise independent hash functions to generate desired approximation of q. Let $\mathcal{H}$ be a family of t-wise independent hash functions from $[D]$ into $[C]$, and $p = Pr_{h \in \mathcal{H}}[h^{-1}(0) \cap S \ne \emptyset]$. Next we will show if the t is large enough, then p can be arbitrarily close to q.

Lemma 5: Let t be $\lceil \frac{\log \frac{2}{\lambda}}{\log 5} \rceil$, then $|p - q| \le \frac{\lambda}{2}$ where $\lambda = min(\frac{1}{e} - \frac{1}{3}, \frac{\varepsilon}{6k})$.

Proof: Let $\mathcal{H}_i \subseteq \mathcal{H}$ be the subset of hash functions that map i-th element of S into 0. As p is to count the percentage of the number of hash functions, that map some element to 0, to

all hash functions, so $p = \frac{|\bigcup_{i=1}^{N} \mathcal{H}_i|}{|\mathcal{H}|}$. By inclusion-exclusion, we have

$$p = \sum_i Pr_{h\in\mathcal{H}}[h \in \mathcal{H}_i] - \sum_{i<j} Pr_{h\in\mathcal{H}}[h \in (\mathcal{H}_i \cap \mathcal{H}_j)] + \cdots.$$

We use T_l to denote the l-th term in above equation. Therefore, for any odd $t > 0$, we can get

$$\sum_{l=1}^{t-1}(-1)^{l+1}T_l \le p \le \sum_{l=1}^{t}(-1)^{l+1}T_l.$$

Since the hash functions are t-wise independent, the probabilities of all $\binom{N}{l}$ subsets can multiple together. Therefore we obtain

$$\sum_{l=1}^{t-1}(-1)^{l+1}\binom{N}{l}C^{-l} \le p \le \sum_{l=1}^{t}(-1)^{l+1}\binom{N}{l}C^{-l}. \quad (10)$$

At the same time, by using binomial expansion we can change the expression q into

$$q = 1 - (1 - \frac{1}{C})^N = \sum_{l=1}^{N}(-1)^{l+1}\binom{N}{l}C^{-l}$$

and for odd t, we have

$$\sum_{l=1}^{t-1}(-1)^{l+1}\binom{N}{l}C^{-l} \le q \le \sum_{l=1}^{t}(-1)^{l+1}\binom{N}{l}C^{-l}. \quad (11)$$

Since both (10) and (11) are sandwiched, we know that if t is sufficiently large, the difference between two terms q and p can be arbitrarily small. As derived by (10) and (11), the interval of width is $\binom{N}{t}C^{-t}$ and t is $\lceil\frac{\log\frac{2}{\lambda}}{\log 5}\rceil$, we have

$$|p - q| \le \binom{N}{t}C^{-t} \le (\frac{eN}{tC})^t \le (\frac{1}{5})^t \le \frac{\lambda}{2}.$$

■

Now we show that p is at most $\frac{\lambda}{2}$ far from q, so if we can get an estimation $\hat{p}$ of p satisfying $|p-\hat{p}| \le \frac{\lambda}{2}$, then $|q-\hat{p}| \le \lambda$ can hold.

Definition Let $\mathcal{H}^m = \{h_1, ...h_m\}$ be a subset of a family $\mathcal{H}$ of t-wise independent hash functions from $[D]$ into $[C]$. And for each $h_i \in \mathcal{H}^m$ we define variable

$$x_{h_i}(\mathcal{H}^m) = \begin{cases} 1 & \text{if } h_i^{-1}(0) \cap S \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

and the estimation $\hat{p}$ as

$$\hat{p} = X(\mathcal{H}^m) = \frac{1}{m}\sum_{h_i\in\mathcal{H}^m} x_{h_i}(\mathcal{H}^m) = \frac{1}{m}|\{i|h_i^{-1}(0)\cap S \neq \emptyset\}|.$$

Lemma 6: Let m be $\frac{756k^2}{\varepsilon^2}$, then $Pr[|X(\mathcal{H}^m) - p| > \frac{\lambda}{2}] \le \frac{1}{21}$

Proof: By definition, we know that $x_{h_i}(\mathcal{H}^m)$ takes 1 with probability p. Hence, we can derive $E[x_{h_i}(\mathcal{H}^m)] = p$ and $Var[x_{h_i}(\mathcal{H}^m)] = p(1-p)$. As $X(\mathcal{H}^m)$ is the sum of m independent variables, we know that

$$E[X(\mathcal{H}^m)] = \frac{1}{m}\sum_{h_i\in\mathcal{H}^m} E[x_{h_i}] = \frac{1}{m}\cdot mp = p$$

$$Var[X(\mathcal{H}^m)] = \frac{1}{m^2}\sum_{h_i\in\mathcal{H}^m} Var[x_{h_i}] = \frac{p-p^2}{m}.$$

Then by Chebyshev inequality, we obtain

$$\begin{aligned} Pr[|X(\mathcal{H}^m) - p| > \frac{\lambda}{2}] &\le \frac{Var[X(\mathcal{H}^m)]}{(\frac{\lambda}{2})^2} \\ &= \frac{(p-p^2)}{m}\cdot\frac{144k^2}{\varepsilon^2} \\ &= (p-p^2)\cdot\frac{\varepsilon^2}{756k^2}\cdot\frac{144k^2}{\varepsilon^2} \\ &= ((p-\frac{1}{2})^2 + \frac{1}{4})\cdot\frac{144}{756} \le \frac{1}{21}. \end{aligned}$$

■

Theorem 2 (Epsilon Approximation Bound): Let k be 7, t be $\lceil\frac{\log\frac{2}{\lambda}}{\log 5}\rceil$, and the size of subset $\mathcal{H}^m$ be $\frac{37044}{\varepsilon^2}$, then $Pr[|\hat{N} - N| \le \varepsilon N] \ge \frac{11}{21}$.

Proof: By theorem 1, let $k = 7$, constant estimation algorithm gives an error probability of $\frac{3}{7}$ at most, as $\frac{N}{k} \le C\prime \le kN \Rightarrow 2k\cdot\frac{N}{k} \le 2k\cdot C\prime \le 2k\cdot kN \Rightarrow 2N \le C \le 2k^2N$ with probability of $1-\frac{3}{7}$. Combing lemma 5 and lemma 6, we know that estimation $\hat{p}$ of q gives error probability of at most $\frac{1}{21}$, since $|p-q| \le \frac{\lambda}{2}$ holds when $t = \lceil\frac{\log\frac{2}{\lambda}}{\log 5}\rceil$and $|\hat{p}-p| \le \frac{\lambda}{2}$ with probability at least $\frac{20}{21}$. Therefore, the union bound that the probability of as least one of the two events happening is at most $\frac{3}{7} + \frac{1}{21} = \frac{10}{21}$. This is sufficient to establish theorem 2. ■

C. Boosting the success probability

The theorem 2 shows that an approximation estimation $\hat{N}$ can be given with probability at least $\frac{11}{21}$. But the success probability $\frac{10}{21}$ does not seem very impressive. To meet the requirements of some high standard applications, it may need to be able to succeed with a probability arbitrarily close to 1, i.e., where δ can be arbitrarily small.

We independently select f hash subsets $\mathcal{H}^{m_i}$ $(1 \le i \le f)$ from a family $\mathcal{H}$ of t-wise independent hash functions. And Let $\hat{N}_i$ be the estimate result for each subset $\mathcal{H}^{m_i}$. Then we use $\hat{N}$ to denote the median of $\hat{N}_1, ..., \hat{N}_f$. Thus, we can define random variable

$$x(\mathcal{H}^{m_i}) = \begin{cases} 0 & \text{if } |\hat{N}_i - N| \le \varepsilon N \\ 1 & \text{otherwise} \end{cases}$$

and $X = \sum_{i=1}^{f} x(\mathcal{H}^{m_i})$.

Theorem 3 (Delta Approximation Bound): For any δ between 0 and 1, there is an $f = \mathcal{O}(\log\delta^{-1})$ ensuring that $Pr[|\hat{N} - N| \le \varepsilon N] \ge 1 - \delta$.

Proof: From theorem 2, we know that $x(\mathcal{H}^{m_i})$ takes 1 with probability at most $\alpha = \frac{10}{21}$. And it is easy to derive that $E[x(\mathcal{H}^{m_i})] = \alpha < \frac{1}{2}$ and $E[X] = f\alpha$. If X is less than $\frac{f}{2}$,

Algorithm 1 A^3 algorithm for RFID tags in constant estimation phase

1: Receive the random estimating path p and $pathmask$.
2: **if** $tagID \wedge pathmask = p \wedge pathmask$ **then**
3: Respond instantly.
4: **else**
5: Keep silent.
6: **end if**

Algorithm 2 A^3 algorithm for RFID readers in constant estimation phase

1: $low \leftarrow 1$, $high \leftarrow 32$;
2: Randomly select a number in $[0, 2^{32}]$ as p; Broadcast p;
3: **while** $low < high$ **do**
4: $mid \leftarrow \lceil \frac{low+high}{2} \rceil$;
5: Set high mid bits of $mask$ to 1; Broadcast $mask$.
6: **if** there is no response in following time slot **then**
7: $high \leftarrow mid - 1$;
8: **else**
9: $low \leftarrow mid$;
10: **end if**
11: **end while**
12: **return** $C = 2^{low}$;

we can see that $|\hat{N}_i - N| \leq \varepsilon N$ definitely holds since $\hat{N}$ is the median of $\hat{N}_1, ..., \hat{N}_f$. Thus, if the event $X \geq \frac{f}{2}$ happens with probability at most δ, the argument in above theorem is correct. Towards this, by Chernoff bound we have

$$\begin{aligned} Pr[X \geq \frac{f}{2}] &= Pr[X - E[X] \geq \frac{f}{2} - E[X]] \\ &\leq Pr[|X - E[X]| \geq \frac{f}{2} - E[X]] \\ &= Pr[|X - E[X]| \geq \frac{f}{2} - f\alpha] \\ &= Pr[|X - E[X]| \geq \frac{\frac{1}{2} - \alpha}{\alpha} \cdot f\alpha] \\ &\leq 2e^{-\frac{(\frac{1}{2}-\alpha)^2}{3\alpha^2} \cdot f\alpha} \leq \delta. \end{aligned}$$

Therefore, if we set $f = \lceil \frac{3\alpha}{(\frac{1}{2}-\alpha)^2} \ln \frac{2}{\delta} \rceil = 2520 \ln \frac{2}{\delta} = \mathcal{O}(\log \delta^{-1})$, we can make $Pr[X \geq \frac{f}{2}] \leq \delta$, and then the complement event $X < \frac{f}{2}$ happens with probability at least $1 - \delta$. ■

D. General protocol

In this section, we formally give the general communication protocol and algorithm for both tags and readers. In our protocol, we assume that the cardinality of tags does not exceed 2^{32}.

Algorithm 1 defines behaviors of tags in the constant estimation phase. The task assigned to the tag is very simple. The tag receives the estimating path p and $pathmask$ (both p and $pathmask$ are 32-bit integers, see details in [20]) from reader (line 1), and then compare the prefix of $tagID$ and estimating path p. If they are the same, the tag respond instantly, otherwise it keeps silent (line 2-6). The tag receives

Algorithm 3 A^3 algorithm for RFID tags in (ε, δ) estimation phase

1: Receive the c_t coefficients.
2: Compute linear hash function $\mathcal{H}(tagID, c_t)$.
3: **if** $\mathcal{H}(tagID) == 0$ **then**
4: Respond instantly.
5: **else**
6: Keep silent.
7: **end if**

Algorithm 4 A^3 algorithm for RFID readers in (ε, δ) estimation phase

1: $f \leftarrow 2520 \ln \frac{2}{\delta}$;
2: **for** $i \leftarrow 1$ to f **do**
3: Acquire C in constant estimation phase (Algorithm 2).
4: $C \leftarrow C \cdot 2 \cdot 7$, $\lambda \leftarrow min(\frac{1}{e} - \frac{1}{3}, \frac{\varepsilon}{42})$;
5: $t \leftarrow \lceil \frac{\log \frac{2}{\lambda}}{\log 5} \rceil$, $m \leftarrow \frac{37044}{\varepsilon^2}$, $X \leftarrow 0$;
6: **for** $j \leftarrow 1$ to m **do**
7: Randomly select a t-wise independent hash function with coefficients c_t; Broadcast coefficients c_t.
8: **if** there is a response in following time slot **then**
9: $X \leftarrow X + 1$;
10: **end if**
11: **end for**
12: $X \leftarrow \frac{X}{m}$;
13: $\hat{N}_i = \frac{\ln(1-X)}{\ln(1-\frac{1}{C})}$;
14: **end for**
15: **return** the median of $\{\hat{N}_1, ..., \hat{N}_f\}$.

five $pathmask$s at most in this phases, as the reader adopts a binary search.

Algorithm 2 depicts the task of readers in the constant estimation phase. The reader first randomly chooses a number p as estimating path and sends it to the tags (line 2). Then according to responses from tags, binary searching is conducted to find the longest prefix (line 3-11). Finally, it outputs the estimate (line 12). Here it is worth noting that when p is 0, the longest prefix is the same as maximum leading zeros. What is more, for any chosen p, the probabilities of longest prefix matching are the same. Our theorem 1 thus still holds.

Algorithm 3 describes the work of tags in (ε, δ) estimation phase. The tag just waits to receive the coefficient of t-wise independent hash function and then do the simple linear hashing (line 1-2). If the hashing result is equal to zero the tag replies immediately, otherwise it does nothing (line 3-7).

Algorithm 4 gives (ε, δ) approximating process of the reader. The reader first computes how many rounds are needed to achieve required δ (line 1). Then for a given constant estimate C, it selects m t-wise independent hash functions coefficients and broadcasts to tags. If the reader receives any signal in the following time slot, the counter X increases by one. Therefore one round estimate $\hat{N}_i$ is obtained (line 2-14). Through f rounds approximation, it outputs the estimate of cardinality with required accuracy (line 15).

As shown above, in constant estimation phase, the time slots needed is $\mathcal{O}(\log \log n)$. And in (ε, δ) estimation phase,

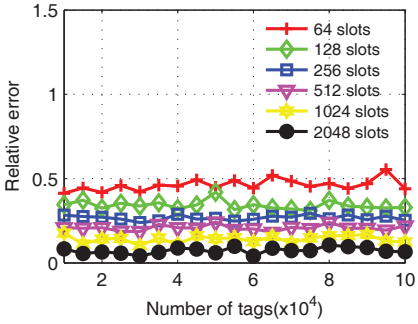


Fig. 1. Relative error of estimate $\hat{N}$

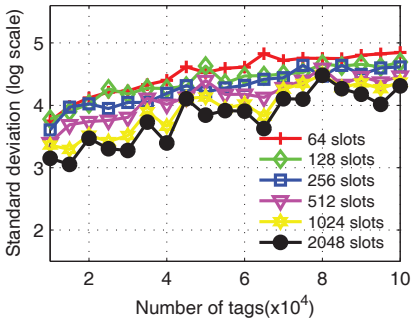


Fig. 2. Standard deviation of estimate $\hat{N}$

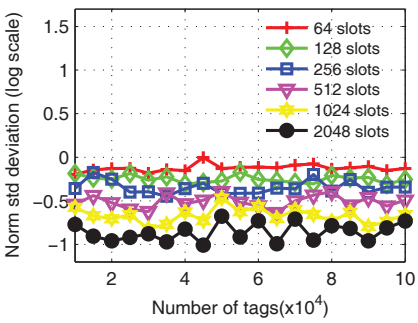


Fig. 3. Normalized std deviation of estimate $\hat{N}$

the time slots is $\mathcal{O}(\varepsilon^{-2})$. By theorem 3, it requires $\mathcal{O}(\log \delta^{-1})$ independent estimate rounds. Therefore, the total time complexity is $\mathcal{O}((\log\log n + \varepsilon^{-2})\log \delta^{-1})$, that is nearly constant for a given (ε, δ). In addition, we do not assume preloaded storage such as MD5/SHA-1 codes or require complex hashing computations that might impose unsuitable burden on the tags. More importantly, compared with previous approaches our final estimate result is a rigorously theoretical (ε, δ) approximation, instead of asymptotic result. As a result, our A^3 scheme is indeed an appropriate solution to accurately approximate the cardinality of tags for practical large-scale RFID systems.

E. Discussion

Someone may notice that the number of time slots $\frac{37044}{\varepsilon^2}$ and the number of rounds f needed in above protocols may seem not so time-efficient. There are two notations about this. First, A^3 is the first to give rigorous bound for finite number of time slots and f, differing from asymptotic results of previous methods (see details in section III). Second, this bound is the worst case to guarantee relative error within epsilon. That is, for any input, at most $\frac{37044}{\varepsilon^2}$ times slots are needed. In practice the number of time slots is much smaller than the upper bound. For example, in order to achieve relative error of 0.2, the theoretical worst case requires $37044/0.2^2 = 926,100$ time slots. While as shown in evaluation part, 512 time slots are adequate to get the same accuracy in most cases.

V. Evaluation

We evaluate the performance of A^3 under extensive simulations. First, we study the estimation accuracy with varying size of tags under various settings. Then we compare A^3 with two most recent methods LOF [19] and PET [20] in terms of estimate accuracy using standard metrics.

A. Setup and Metrics

The simulations are implemented on a DELL VOSTRA PC with Intel i3 CPU at 3.1GHz and 2GB RAM. The programming language is C#. We assume communications between tags and reader are reliable. We take 400 runs and report the average. The estimation accuracy is of great importance for approximation scheme. We adopt three standard metrics. The first metric is relative error: $relerr = |\frac{\hat{N}-N}{N}|$. The second metric is standard deviation: $\sigma = \sqrt{E[(\hat{N}-N)^2]}$. The third parameter is normalized standard deviation: $\sigma_n = \frac{\sigma}{E[N]}$. Besides accuracy, we also evaluate estimation efficiency using different frame size.

TABLE II. Actual time slots vs. worse case bound with $N = 50,000$, $f = 1$.

Relative error	0.59	0.35	0.20	0.07
Actual time slots	32	128	512	2048
Worse case time slots	106418	302400	926100	7560000
Worse case/Actual	3326	2363	1809	3691

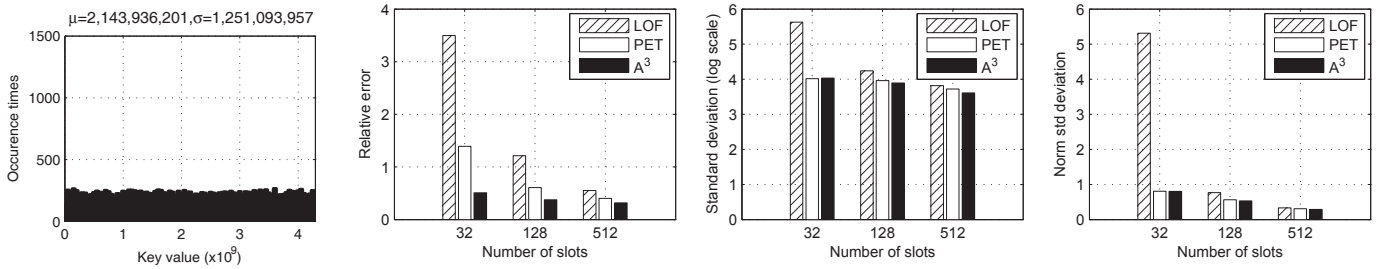
B. A^3 Investigation

First, we examine the accuracy of A^3. In Figure 1, we can see that as the frame size is increasing, relative error is getting smaller. With only 256 time slots, A^3 maintains relative error around 0.25. And relative error is reduced around 0.06 when 2048 time slots are used. Figure 1 also shows that relative error is insensitive to the number of tags. In other words, A^3 can obtain accurate estimate in near-constant time for any size of tags, without a priori knowledge about actual number of tags. We examine standard deviations and normalize standard deviations in Figure 2 and 3, respectively. Figure 2 demonstrate that more hash functions (larger frame size) effectively diminish standard deviation of estimate results. As illustrated in Figure 3, we again see that the number of tags has little influence on the normalized standard deviation. In particular, using 512 time slots, normalize standard deviation of estimate results are most between 0.2 to 0.3.

Furthermore, we examine the time efficient of A^3 by comparing actual used time to their corresponding worse case bound. We fix $f = 1$, $N = 50,000$, and then run different number of time slots to acquire actual relative errors as baseline. The results shown in Table II demonstrate that the actual time slots are far more less than the worse case bounds. In particular, To achieve relative error of 0.07, the worse case time slots is 7,560,000, which is 3690 times more than the actual time slots.

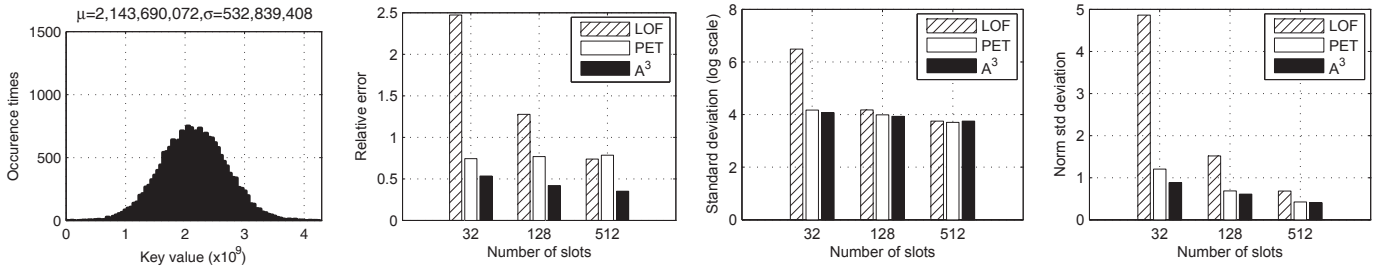
C. Performance Comparison

We compare the performance of A^3 with two state-of-the-art schemes, LOF [19] and PET [20]. As those two schemes and A^3 share the probabilistic nature that the more rounds a protocol performs, the more accurate estimate is. Therefore



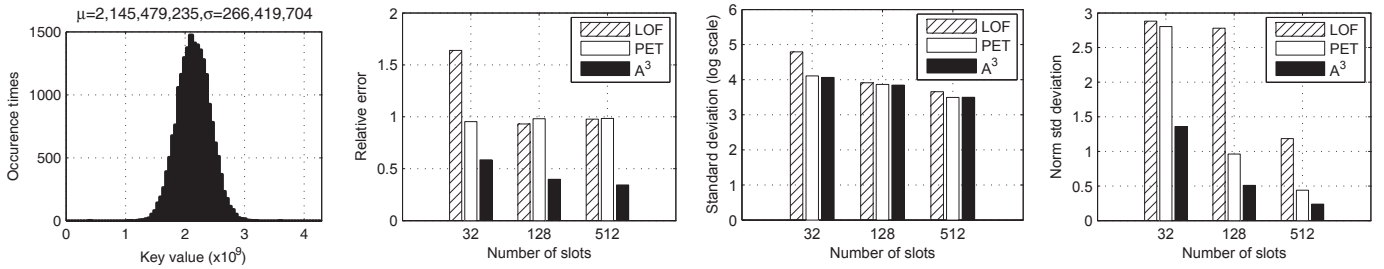
(a) TagID (key) distribution (b) Relative error (c) Standard deviation (d) Normalized std deviation

Fig. 4. Comparison under uniform distribution ($\mu = 2,143,936,201, \sigma = 1,251,093,957$)



(a) TagID (key) distribution (b) Relative error (c) Standard deviation (d) Normalized std deviation

Fig. 5. Comparison under normal distribution ($\mu = 2,143,690,072, \sigma = 532,839,408$)



(a) TagID (key) distribution (b) Relative error (c) Standard deviation (d) Normalized std deviation

Fig. 6. Comparison under normal distribution ($\mu = 2,145,479,235, \sigma = 266,419,704$)

we conduct comparisons of estimate accuracy under fixed the same amount of time settings. LOF gives an estimate in every 32 time slots, while PET in every 5 time slots. In contrast, A^3 can provides an estimate in any time slot.

We synthesize 20000 tagIDs data from different uniform distribution and normal distribution, ranging in [0,2^{32}]. First, we inspect three schemes using a uniform distribution. As illustrated in Figure 4, there are large gaps between A^3 and the other two schemes when frame size is 32. Although these gaps are narrowing with increased frame size, A^3 still outperforms LOF and PET. For poor performances of LOF and PET with 32 slots, the main reason is that LOF just performs 1 round estimation and PET conducts 6 rounds estimation in this time constraint. Thus, those two schemes are not appropriate for latency-sensitive RFID applications such as high-velocity object identification. Then we evaluate three schemes using normal distribution with different (μ, σ) parameters. As shown in Figure 5, and 6, we again observe that A^3 is advantageous over both LOF and PET all the time, at worst comparable. More importantly, we find that performances of LOF and PET are disappointing under normal distribution compared to using uniform distribution. As shown in Figure 5(a), even with frame size of 512, relative errors are 0.74 and 0.78 for LOF and PET respectively, while A^3 has relative error 0.35. An even more worst case is in Figure 6. This figure demonstrates that relative errors of LOF and PET maintains closely to 1. In contrast, relative errors of A^3 are 0.58 with 32 slots and 0.39 with 128 slots.

VI. Related Work

Most existing estimation schemes for RFID systems can be classified into two categories, identification-based approaches and probabilistic approaches. In identification-based algorithms, the problem of estimating the cardinality of RFID tags

is transformed into identifying each tagID of all tags. Since all tags share the same wireless channel, the reader needs to properly schedule their transmissions. Many researchers proposed various anti-collision schemes to solve this issue. In the slotted ALOHA protocols that are introduced in [16][17][18], if there is no collision, the tag replies immediately after receiving the reader's probe. Otherwise, the tag chooses another random time slot to transmit the response. Rather than resolving the MAC-layer collisions, our design principle of A^3 is to accurately estimate the total number of tags.

A number of probabilistic approaches are deliberately designed so as to quickly obtain the approximate cardinality of RFID tags. Kodialam et al. [12] first propose probabilistic schemes, Unified Simple Estimator (USE) and Unified Probabilistic Estimator (UPE), for RFID systems. Qian et al. [19] proposes LOF algorithms, in which the geometric distribution hash is used to itemized tags in order to fast acquire the estimation with $\mathcal{O}(\log n)$ time slots. Zheng et al. [20] further improve the efficiency of estimation to $\mathcal{O}(\log \log n)$ time slots by designing a Probabilistic Estimating Tree (PET). Li et al. [25] first observe an asymmetry in energy cost and propose energy-efficient RFID estimation algorithm. Shahzad et al. [28] introduce Average Run based Tag estimation (ART) schemes to fast cardinality estimation. But there are some limitations in those probabilistic approaches. First, [12] requires rough magnitude of cardinality of RFID tag set as prior information. Second, preloaded MD5/SHA-1 values on chips [19][20] impose unnecessary storage burdens. Moreover, the accuracy of estimators largely depends on the distribution of the candidate set of tagIDs. In contrast, our A^3 employs only simple hash functions. And there is no extra storage burden for tags. More importantly, A^3 provides arbitrarily accurate approximation with rigorous theoretical analysis.

VII. Conclusion

This paper concerns the fundamental problem of RFID estimation, providing an arbitrary accurate approximation scheme. In contrast to most prior work that provide only asymptotic results, A^3 achieves $\mathcal{O}((\log \log n + \varepsilon^{-2}) \log \delta^{-1})$ time efficiency. Extensive simulations and comparisons demonstrate the effectiveness and efficiency of A^3. In the future, we will work on adapting A^3 scheme to be compatible with C1G2 standard. The integration of energy-efficient schemes into A^3 is also considered.

Acknowledgments

We would like to thank the anonymous reviewers for valuable and insightful comments. This work is supported in part by the NSFC Young Scholar No.61303196 and 61103187, National Basic Research Program of China (973) Grant No. 2012CB316200, NSFC Major Program 61190110, and China Postdoctoral Science Foundation No. 2013M540950. We also acknowledge the support from the codes of USRP2reader from the Open RFID Lab (ORL) project [29].

References

[1] L. Yang, Y. Qi, J. Fang, X. Ding, T. Liu, and M. Li. Frogeye: Perception of the slightest tag motion. In *Proceedings of IEEE INFOCOM*, 2014.

[2] L. Yang, J. Han, Y. Qi, and Y. Liu. Identification-free batch authentication for rfid tags. In *Proceedings of IEEE ICNP*, 2010.

[3] C.H. Lee and C.W. Chung. Efficient storage scheme and query processing for supply chain management using rfid. In *Proceedings of ACM SIGMOD*, 2008.

[4] C.C. Tan, B. Sheng, and Q. Li. How to monitor for missing rfid tags. In *Proceedings of IEEE ICDCS*, 2008.

[5] L. Yang, J. Han, Y. Qi, C. Wang, T. Gu, and Y. Liu. Season: Shelving interference and joint identification in large-scale rfid systems. In *Proceedings of IEEE INFOCOM*, 2011.

[6] Y. Liu, Y Zhao, L. Chen, J. Pei, and J. Han. Mining frequent trajectory patterns for activity monitoring using radio frequency tag arrays. *IEEE Transactions on Parallel and Distributed Systems*, 23(11):2138–2149, 2012.

[7] T. Liu, L. Yang, Q. Lin, Y. Guo, and Y. Liu. Anchor-free backscatter positioning for rfid tags with high accuracy. In *Proceedings of IEEE INFOCOM*, 2014.

[8] Frost and Sullivan. Global rfid market. *Industry Report*, 2011.

[9] R.B. Freeman, A.O. Nakamura, L.I. Nakamura, M. Prudhomme, and A. Pyman. Wal-mart innovation and productivity: a viewpoint. *Canadian Journal of Economics/Revue canadienne d'économique*, 44(2):486–508, 2011.

[10] D.R. Avery, P.F. Mckay, and E.M. Hunter. Demography and disappearing merchandise: How older workforces influence retail shrinkage. *Journal of Organizational Behavior*, 33(1):105–120, 2012.

[11] M. Kodialam, T. Nandagopal, and W.C. Lau. Anonymous tracking using rfid tags. In *Proceedings of IEEE INFOCOM*, 2007.

[12] M. Kodialam and T. Nandagopal. Fast and reliable estimation schemes in rfid systems. In *Proceedings of ACM MobiCom*, 2006.

[13] H. Han, B. Sheng, C.C. Tan, Q. Li, W. Mao, and S. Lu. Counting rfid tags efficiently and anonymously. In *Proceedings of IEEE INFOCOM*, 2010.

[14] Mark Baard. Rfid driver's licenses debated, 2004.

[15] A. Juels, D. Molnar, and D. Wagner. Security and privacy issues in e-passports. In *Proceedings of IEEE SecureComm*, 2005.

[16] B. Sheng, Q. Li, and W. Mao. Efficient continuous scanning in rfid systems. In *Proceedings of IEEE INFOCOM*, 2010.

[17] R. Kumar, T.F. La Porta, G. Maselli, and C. Petrioli. Interference cancellation-based rfid tags identification. In *Proceedings of ACM MSWiM*, 2011.

[18] T.F. La Porta, G. Maselli, and C. Petrioli. Anticollision protocols for single-reader rfid systems: Temporal analysis and optimization. *IEEE Transactions on Mobile Computing*, 10(2):267 –279, feb. 2011.

[19] C. Qian, H. Ngan, Y. Liu, and L.M. Ni. Cardinality estimation for large-scale rfid systems. *IEEE Transactions on Parallel and Distributed Systems*, 22(9):1441–1454, 2011.

[20] Y. Zheng, M. Li, and C. Qian. Pet: Probabilistic estimating tree for large-scale rfid estimation. In *Proceedings of IEEE ICDCS*, 2011.

[21] R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited. *Journal of the ACM (JACM)*, 51(4):557–594, 2004.

[22] J.S. Coron, J. Patarin, and Y. Seurin. The random oracle model and the ideal cipher model are equivalent. *Advances in Cryptology–CRYPTO 2008*, pages 1–20, 2008.

[23] R. Want. Rfid: A key to automating everything. *Scientific American*, 10, 1991.

[24] Yuanqing Zheng and Mo Li. Zoe: Fast cardinality estimation for large-scale rfid systems.

[25] T. Li, S. S. Wu, S. Chen, and M. C. K. Yang. Generalized energy-efficient algorithms for the rfid estimation problem. *IEEE/ACM Transactions on Networking*, 20(6):1978 –1990, dec. 2012.

[26] A.W. Van der Vaart. *Asymptotic statistics*. Number 3. Cambridge Univ Pr, 2000.

[27] P. Flajolet and G. Nigel Martin. Probabilistic counting algorithms for data base applications. *Journal of computer and system sciences*, 31(2):182–209, 1985.

[28] Muhammad Shahzad and Alex X. Liu. Every bit counts: fast and scalable rfid estimation. In *Proceedings of ACM MobiCom*, 2012.

[29] Open rfid lab, http://pdcc.ntu.edu.sg/wands/orl.