Adaptive Joint Estimation Protocol for Arbitrary Pair of Tag Sets in a Distributed RFID System

Qingjun Xiao, *Member, IEEE, ACM*, Shigang Chen, *Fellow, IEEE*, Min Chen, Yian Zhou, Zhiping Cai, *Member, IEEE, ACM*, and Junzhou Luo, *Member, IEEE, ACM*

***Abstract*— Radio frequency identification (RFID) technology has been widely used in Applications, such as inventory control, object tracking, and supply chain management. In this domain, an important research problem is called *RFID cardinality estimation*, which focuses on estimating the number of tags in a certain area covered by one or multiple readers. This paper extends the research in both temporal and spatial dimensions to provide much richer information about the dynamics of distributed RFID systems. Specifically, we focus on estimating the cardinalities of the intersection/differences/union of two arbitrary tag sets (called *joint properties* for short) that exist in different spatial or temporal domains. With many practical applications, there is, however, little prior work on this problem. We will propose a joint RFID estimation protocol that supports adaptive snapshot construction. Given the snapshots of any two tag sets, although their lengths may be very different depending on the sizes of tag sets they encode, we design a way to combine their information and more importantly, derive closed-form formulas to use the combined information and estimate the joint properties of the two tag sets, with an accuracy that can be arbitrarily set. By formal analysis, we also determine the optimal system parameters that minimize the execution time of taking snapshots, under the constraints of a given accuracy requirement. We have performed extensive simulations, and the results show that our protocol can reduce the execution time by multiple folds, as compared with the best alternative approach in literature.**

Manuscript received September 16, 2016; revised April 5, 2017; accepted April 27, 2017; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor A. X. Liu. This work was supported in part by the National Natural Science Foundation of China under Grant 61502098, Grant 61632008, and Grant 61320106007, in part by the Jiangsu Provincial Natural Science Foundation of China under Grant BK20150629, in part by the National Science Foundation of United States under Grant CNS-1409797, in part by the Jiangsu Provincial Key Laboratory of Network and Information Security under Grant BM2003201, in part by the Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grant 93K-9, and in part by the Collaborative Innovation Center of Novel Software Technology and Industrialization. The work of Z. Cai was supported by the National Science Foundation of China under Grant 61379145. The preliminary version of this paper titled "Temporally or Spatially Dispersed Joint RFID Estimation Using Snapshots of Variable Lengths" was published in proceedings of the ACM MobiHoc (International Symposium on Mobile Ad Hoc Networking and Computing), pp. 247–256, 2015. *(Corresponding author: Qingjun Xiao.)*

Q. Xiao and J. Luo are with the School of Computer Science and Engineering, Southeast University, Nanjing 211189, China (e-mail: csqjxiao@seu.edu.cn; jluo@seu.edu.cn).

S. Chen is with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: sgchen@cise.ufl.edu).

M. Chen and Y. Zhou were with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611 USA. They are now with Google Inc., Mountain View, CA 94043 USA (e-mail: minchen@google.com; yianzhou@google.com).

Z. Cai is with the College of Computer, National University of Defense Technology, Changsha 410073, China (e-mail: zpcai@nudt.edu.cn).

Digital Object Identifier 10.1109/TNET.2017.2709979

***Index Terms*— RFID, adaptive ALOHA protocol, cardinality estimation, set difference size, set intersection size.**

I. INTRODUCTION

RFID (radio-frequency identification) technology has been widely used in various commercial applications, including inventory control, object tracking, supply chain management and auto-payment, etc [1], [2]. RFID tags (each carrying a unique identifier) are attached to merchandise at retail stores, equipments at hospitals, or goods at warehouses, allowing an RFID reader to quickly identify products, access properties of each individual item, or collect statistical information about a group of items.

An important system function is called the *RFID cardinality estimation* [3]–[13], which is to estimate the number of tags in a particular region covered by one or multiple readers. This basic function can be used to monitor the inventory level in a warehouse, the sales in a retail store, and the popularity of attractions in tourism [11]. It can also serve as a pre-processing step to make other functions (such as tag identification [14]–[16]) more efficient. RFID estimation takes much less time to perform than a full system scan that collects all tag IDs. This makes it valuable since RFID systems operate at low wireless rates and the execution time has been the key performance metric in system design. Moreover, it does not identify any tags, which avoids the privacy issue, particularly in scenarios where the party performing the operation (such as warehouse or port authority) does not own the tagged items.

Motivation: This paper extends RFID estimation in both temporal and spatial dimensions to provide much richer information about the dynamics of a distributed RFID system. We use two applications to explain the problems of temporally dispersed RFID estimation and spatially dispersed estimation, respectively. In the first application, we consider to monitor the dynamics of the inventory in a warehouse over time. We are interested in the amount of goods moving in (i.e., the number of new tags) and the amount moving out (i.e., the number of departure tags) between *any two* reference time points, without identifying the tag IDs, where the reference time points may be evenly spaced by time intervals of a certain length. The problem cannot be easily solved simply by estimating the number of tags in the warehouse after each time interval by traditional approaches [3]–[11]. For instance, if the number of tags at time 1 is estimated to be 1000 and so does the number at time 2, we will not be able to know whether

no new tag has moved in or 1000 new tags have moved in while all old tags have moved out. To handle this problem, we need to take *snapshots* with more detailed information about existing tags, such that from any two snapshots taken at different time points, we will be able to estimate the joint properties of the corresponding two tag sets, such as their union, intersection and difference, which provide information for stocking dynamics about product inflow and outflow.

In the second application, we may consider the supply chain management in a large logistics network with multiple locations. As tagged products are shipped from location (component factory, assembly line, warehouse, port, or other storage/retail facility) to location, if each location takes periodic snapshots of its tag set and keeps a series of such snapshots over time, we will be able to make queries for joint estimation between *any two* snapshots, which may be taken from different locations or from the same location at different times. For instance, by jointly analyzing two snapshots at different locations, we can know the volume of a product delivery flow from a source warehouse to another destination warehouse in a supply chain network, which constitute the so-called *traffic matrix*. In the domain of traffic measurement for IP backbone networks, traffic matrix (i.e., the volume of traffic that flows between all possible pairs of source and destination routers) has been pervasively used, as it is very valuable to a wide variety of traffic engineering tasks including load balancing, billing, anomaly detection, and routing protocols configuration [17], [18]. Similarly, we envision that the automatic measurement of traffic matrix will in future become crucial for the efficient operation of a RFID-enabled supply chain network. Such joint estimation between each pair of locations, when performed over time across the network, gives a network-wide view about how goods flow within a logistics network. For one application, this information can help diagnose erratic shipments by identifying unexpected volumes that move over supply chains with significant deviation from a pre-established business plan — it has been reported that, due to such logistic errors, more than 65% of the inventory records did not match the physical inventory. Without any automatic tools, we will have to resort to manual inventory check to discover the errors, which is laborious, expensive and slow, especially when such inventory task needs to be performed at daily basis.

Moreover, comparing with traditional RFID estimation [3]–[11] (which were designed to operate at a single time and a single place), the ability to jointly consider any two temporally or spatially dispersed snapshots will enable us to expand the applications mentioned earlier, for example, by providing more detailed information about changes in inventory and sales, by monitoring the flows of tourists moving from place to place in a theme park, or by serving as a pre-processing step to make some sophisticated functions such as continuous tag monitoring [19] more efficient.

Problem, Challenge and Prior Art: We abstract the problem of *joint RFID estimation* from the above applications, which is to estimate the joint properties of two arbitrary sets of tags at different times or different locations in a large distributed RFID system. The joint properties include the cardinalities of the union, intersection and difference of the two sets.

The key challenge is that when a snapshot is taken for one tag set, we do not know which other set (at different time/location) the joint estimation will be made with. In fact, the snapshot can be paired with any other snapshot taken in the past or future in the system.

There is little prior work on this practically interesting problem. Directly related is the differential estimation method (DiffEstm) [20], which focuses on the difference between two sets and adopts a different problem model. It uniformly sets the sizes of all snapshots based on the worst-case situation so that any two can be paired. This is very inefficient because the tag sets in a system can have very diverse sizes. For example, in the previous logistics network application, a warehouse may sometimes be almost empty, while carrying tens of thousands of items at other times. Suppose the largest set the system can handle is 50,000. Even if a tag set at a certain time is down to hundreds, the size of its snapshot will still have to be set according to 50,000 in [20].

Our Contributions: First, we propose a new solution for the generalized joint RFID estimation problem based on a simple yet versatile snapshot construction. It adopts a two-phase protocol that needs only two passes of communication between a reader and tags to construct the snapshot of a given tag set. The size of the snapshot is roughly proportional to the size of the tag set, instead of being fixed to a large worst-case value. Given the snapshots of any two tag sets, although their sizes may be very different, we propose a way to combine their information and more importantly derive formulas to extract the joint properties of the two sets from the combined information.

Second, we analyze the means and variances of the estimated properties computed from the formulas. We show that the formulas produce asymptotically unbiased results, and estimate the joint properties with an absolute (probabilistic) error bound that can be set arbitrarily. We also derive the formulas for determining the optimal system parameters that minimize the execution time of taking snapshots, under the constraints of a given accuracy requirement for joint estimation.

Third, we perform extensive simulations to complement the theoretical analysis. The results show that by allowing the snapshots to have variable sizes, our solution significantly outperforms the existing method. For example, under the same accuracy requirement, our solution achieves about 240% improvement in execution time as compared with DiffEstm [20].

The remainder of this paper is organized as follows. Section II defines the problem of joint property estimation for any pair of tag sets. Section III discusses the related work. Section IV proposes our JREP protocol for encoding each tag set into a data structure called a snapshot and then recovering the joint property information from any pair of such snapshots. Section V analyzes this estimation protocol's mean and variance. Section VI optimizes the parameters of JREP protocol for minimizing its estimation variance. Section VII explains how to apply our protocol to the multi-reader and

multi-antenna environments. Section VIII evaluates the performance of our protocol by simulations. Section IX draws the conclusion.

II. Problem Definition

Consider a large distributed RFID system such as a supply-chain network, consisting of multiple locations, where tagged objects are shipped from location to location. At any time and any location, there is a set of tags. Consider two arbitrary sets of tags, N and N', at different locations or at the same location but different times. We study the joint properties of the two sets, including their intersection, union and difference.

Let $n = |N|$, $n' = |N'|$, $u = |N \cup N'|$, $m = |N \cap N'|$, $d = |N \setminus N'|$, and $d' = |N' \setminus N|$. Without loss of generality, we assume N is a larger set than N', and hence $n \geq n'$. The *joint RFID estimation problem* is to provide estimations $\hat{u}$, $\hat{m}$, $\hat{d}$, and $\hat{d'}$ for u, m, d and d' respectively, such that the following pre-defined accuracy requirements are met:

$$\text{Prob}\{\hat{u} - \theta \leq u \leq \hat{u} + \theta\} \geq 1 - \delta \quad (1)$$

$$\text{Prob}\{\hat{d} - \theta \leq d \leq \hat{d} + \theta\} \geq 1 - \delta \quad (2)$$

$$\text{Prob}\{\hat{m} - \theta \leq m \leq \hat{m} + \theta\} \geq 1 - \delta \quad (3)$$

$$\text{Prob}\{\hat{d'} - \theta \leq d' \leq \hat{d'} + \theta\} \geq 1 - \delta, \quad (4)$$

where δ is a probability value, and θ is a probabilistic error bound. For example, when $\delta = 5\%$ and $\theta = 100$, it requires that the absolute error of each estimation should be within the range of $\pm\theta$ at a probability of at least 95%.

An alternative way of specifying the estimation accuracy is based on a relative error $\epsilon \in (0, 1)$.

$$\text{Prob}\{\hat{u}\,(1-\epsilon) \leq u \leq \hat{u}\,(1-\epsilon)\} \geq 1 - \delta \quad (5)$$

$$\text{Prob}\{\hat{d}\,(1-\epsilon) \leq d \leq \hat{d}\,(1-\epsilon)\} \geq 1 - \delta \quad (6)$$

$$\text{Prob}\{\hat{m}\,(1-\epsilon) \leq m \leq \hat{m}\,(1-\epsilon)\} \geq 1 - \delta \quad (7)$$

$$\text{Prob}\{\hat{d'}\,(1-\epsilon) \leq d' \leq \hat{d'}\,(1-\epsilon)\} \geq 1 - \delta \quad (8)$$

where the probabilities for the relative estimation errors $\frac{\hat{u}-u}{\hat{u}}$, $\frac{\hat{d}-d}{\hat{d}}$, $\frac{\hat{m}-m}{\hat{m}}$ and $\frac{\hat{d'}-d'}{\hat{d'}}$ to fall in the range of $\pm\epsilon$ are least $1-\delta$.

We do not adopt this model because it is very expensive or even impossible to achieve as the values of m, d and d' can be very small (down to zero). Consider $m = 0$. In this case, we will have to make sure that $\hat{m} = m = 0$ in order for (7) to be met, which means *precise measurement* of the empty intersection, i.e., $Var(\hat{m}) = 0$. Because $\hat{m}$ is indirectly derived from the two snapshots, these snapshots cannot carry any positive variance in their information based on which $\hat{m}$ is computed; note that the snapshots are independent due to $N \cap N' = \emptyset$ when $m = 0$. Recording precise information (such as IDs of all tags) is very expensive; all existing RFID estimation methods collect imprecise information from tags with non-zero variance to save time for information collection.

A critical problem is that at the time when the snapshot is taken for any tag set, we do not know which other snapshot it will be paired with for joint estimation. Because it is possible to pair with another snapshot with no common tag, we will have to make the snapshot precise (thus expensive) due to the requirement (7).

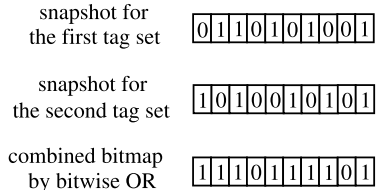

Fig. 1. DiffEstm estimates the difference and intersection between two tag sets by combining their bitmap snapshots, which must have the same length and the same sampling probability.

Finally, it arguably makes more sense to specify absolute error bound in some practical scenarios. Consider the logistics network application. Suppose we want to monitor the volume of products flowing from a number of factories to a number of assembly plants. Further suppose the volume from a particular factory to a particular plant may range from zero to ten of thousands in each pair of snapshots. To get a rough idea about the volume, we may specify the accuracy requirement as an absolute error bound of ± 50 items with 95% confidence. If the actual volume is 10, even though the relative error will be large, it does not change the fact that the estimated volume remains very small, giving correct assessment. On the contrary, if we specify a relative error of 1% and the actual volume is 10, it will require the estimation to have an absolute error of ± 0.1 item, which is not only expensive to achieve but also unnecessary. Note that in this example we estimate small intersection from snapshots of two large sets, not estimating the cardinality of one tag set from its snapshot (e.g., bitmap) as the traditional RFID estimation does.

III. Prior Work

A. Differential Estimation

DiffEstm [20] gives a relative error model similar to (6)-(8) but does not prove that its estimation results meet those requirements. In fact, DiffEstm cannot always meet the relative error bound because it has positive variance in its snapshots, whereas the relative error model requires snapshots to carry precise information as we have argued previously.

We give a simplified description of DiffEstm's snapshot construction: A reader makes a request $(f, p, \ldots)$ to tags in its coverage zone. After a tag receives the request, it will respond with a probability p, in a time slot randomly selected from an Aloha frame of size f. The reader will turn the time frame into a bitmap snapshot of length f, with each busy slot being 1 and each idle slot being 0. In fact, in the original paper, the request carries a frame size F and a parameter f. Each tag transmits in a randomly chosen slot, and the reader only listens to the first f slots. This approach is equivalent to a frame size of f with a sampling probability $p = \frac{f}{F}$.

Figure 1 illustrates how DiffEstm works. After two bitmap snapshots (on the top of the figure) are taken for two tag sets, they are bitwise-ORed to produce a combined bitmap (at the bottom). The difference and intersection of the two sets will then be derived from the information in the three bitmaps, which must all contain a sufficient number of zeros to ensure estimation accuracy [20].

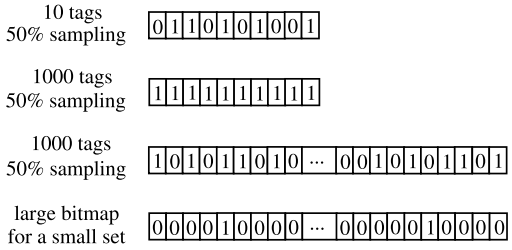

Fig. 2. Large snapshots for small tag sets.

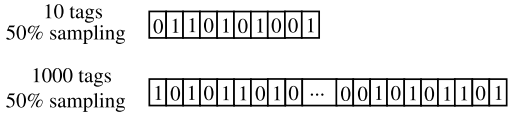

Fig. 3. Snapshots of variable sizes.

To support bitwise-OR, DiffEstm requires that all snapshots must have the same length and the same sampling probability. For any small tag set, if the common sampling probability is very small, too few or even no tag will be sampled for snapshot construction. Hence, the sampling probability will have to be reasonably large, as illustrated by the top bitmap in Figure 2, where 10 tags are recorded with 50% sampling probability. However, for a large set, a significant sampling probability will cause all bits to be set as ones (the second bitmap in the figure), unless the length of the bitmap is sufficiently large (the third bitmap in the figure). Now because the same large length has to be applied to all snapshots, it becomes a great waste for small tag sets (the fourth bitmap). Since each bit takes one time slot to get, a large bitmap length means a long time for taking a snapshot, even for a very small tag set.

Naturally, it is desirable to let each snapshot have a different size, depending on the size of the tag set it records, as shown in Fig. 3. This will require us to develop new methods of combining two snapshots (or two bitmaps) with variable sizes. The real difficulty is not at how to combine two bitmaps per se; there are simple ways to combine them. The real difficulty comes after the combination — how to perform analysis on the information combined from non-uniform snapshots, how to use that information for joint estimation, and most importantly how to ensure the accuracy requirements in (1)-(4). These are the tasks that have not been investigated in the literature.

B. RFID Estimation and Union Estimation

There is a rich set of literature that estimate the cardinality n of a single tag set [3]–[11], typically giving an estimate $\hat{n}$ with a relative error model of

$$Prob\{\hat{n}(1-\epsilon) \leq n \leq \hat{n}(1+\epsilon)\} \geq 1-\delta, \tag{9}$$

which is different from the model of joint estimation where the intersection/difference between two sets are estimated. The execution time is a function of the relative error bound ϵ and the probability δ. For example, the time cost of LOF is $\mathcal{O}(\frac{1}{\epsilon^2}\log n_{\max}) \cdot \log(\frac{1}{\delta})$ [5], the time overhead of PET is $\mathcal{O}(\frac{1}{\epsilon^2}\log\log n_{\max}) \cdot \log(\frac{1}{\delta})$ [7], and the time cost of ZOE is $\mathcal{O}(\frac{1}{\epsilon^2} + \log\log n_{\max}) \cdot \log(\frac{1}{\delta})$ [10], where $n_{\max}$ is the upper bound of the cardinalities of all tag sets. The recent work named simple two-phase RFID counting (SRC) [11] has the best performance to date.

When the tag set cannot be covered by a single reader, deployment of multiple readers is needed, each covering a subset. Many of the RFID estimation solutions can be easily extended for estimating the size of the union of the subsets. Among them, SRC_M [11] performs the best, achieving a reduction in execution time by up to 300%, when comparing with others. SRC_M also uses bitmaps. For each subset, it creates multiple bitmaps, each under a different sampling probability such as $1, 1/2, 1/4, 1/8, \cdots$. It then identifies the best sampling probability and combines the bitmaps of that probability from different subsets using bitwise OR. The combined bitmap records all tags in the union and can thus be used to estimate the union cardinality with the method in [3].

What makes SRC_M efficient is that as it scans one subset after another, it leverages the information learned from the previous subsets to reduce the number of bitmaps (different sampling probabilities) it needs for each subsequent subset. However, this method cannot be extrapolated to temporally/spatial dispersed joint estimation where we do not know which tag sets will be jointly estimated beforehand and thus cannot leverage one set's information to help reduce the overhead for the other.

If we nevertheless want to apply SRC_M to joint estimation, we may use a common sampling probability that is optimized for the worst-case scenario such that an error bound for the union estimation will always be met. In this case, SRC_M will become DiffEstm except that the former considers only union while the latter also addresses difference and intersection (which are more difficult to estimate and analyze).

IV. Joint RFID Estimation Protocol

This section presents our joint RFID estimation protocol (JREP). Our protocol consists of two components: an online encoding component for measuring the information of each tag set and storing it in a bitmap called *snapshot*, and an offline data analysis component for estimating the joint properties of two arbitrary sets such as intersection/union/difference cardinalities, using their snapshots. We use an asymmetric design to push most complexity to the offline component, while keeping the online component as efficient as possible.

A. Online Encoding

Consider a snapshot taken at an arbitrary location and an arbitrary time in a large RFID system of multiple locations. Let N be the set of tags existing at the location and the time when the snapshot is taken, and n be the number of tags in N. The reader that performs the snapshot will first get a rough estimation for the value of n by using an existing cardinality estimation protocol [5]–[7]. It determines a frame size f for the snapshot as follows:

$$f = \min_{p\in(0,1]} \{2^{\lceil \log_2(\frac{np}{\omega})\rceil}\}, \tag{10}$$

where p is a sampling probability, and ω is the frame's load factor (i.e., number of responding tags divided by frame size):

$$\omega = -\frac{3}{4} + \frac{\sqrt{3}}{4}\sqrt{4p\Big(\frac{\theta^2}{n_{\max}{Z_\delta}^2} + 2\Big) - 5}. \qquad (11)$$

We use Z_δ to denote the $1 - \frac{\delta}{2}$ percentile for standard Gaussian distribution $\mathcal{N}(0, 1)$, whose expected value is zero and whose variance is one. For example, when $\delta = 5\%$, we have $Z_\delta = 1.96$, because the probability for a random variable (following standard normal distribution) to carry a value smaller than 1.96 is about $1 - \frac{\delta}{2} = 97.5\%$. Later we will formally derive the above formulas that minimize the time overhead of online encoding and the storage overhead of the snapshot in the worst case, under the constraints of (1)-(4). Let p^* be the sampling probability that minimizes (10). The value of p^* only depends on $n_{\max}$, θ, and δ. Hence, it is predetermined for a system once these parameters are set.

For the ease of understanding, we further explain the above parameter configuration process. Firstly, we could use the prior knowledge of $n_{\max}$, θ and δ to determine the optimal sampling probability p^*, which is able to minimize the frame length f in (10) or equivalently maximize the load factor ω/p in (11). Secondly, we apply the sampling probability p^* into (11) to obtain the target load factor ω, which can satisfy the (θ, δ) accuracy constraint even for the largest tag set $n_{\max}$. Thirdly, with the known p^* and ω, we further use (10) to determine the length of ALOHA frame f that scans a particular tag set N. Note that (10) needs coarse knowledge of the tag set size n, which can be obtained by scanning the tag set N with a low-cost protocol, e.g., GMLE [6], LoF [5], or PET [7].

The process of encoding the tag set N into a bitmap is described as follows: The reader broadcasts an encoding request to start an ALOHA frame with parameters f and p^*. Upon receipt of the request, each tag decides with probability p^* whether it will participate in the encoding, which is called *tag sampling*. If a tag decides to participate, it will select a slot uniformly at random and transmit a pulse during that slot to the reader; if it chooses not to participate, it will keep silent throughout the frame transmission. The reader monitors the status of each slot — with the detection of a pulse, it records the slot as a bit '1'; otherwise, it records a bit '0'. In the case that multiple tags choose to respond in a common time slot, the reader will detect the overlapped waveform of multiple pulses, and it will record the slot as a bit '1'. After transmitting the entire frame, the reader has a bitmap B consisting of zeros and ones, which will be stored and used later for joint estimation. We call this bitmap B as a snapshot of the tag set N.

We acknowledge that the above tag set encoding protocol is only partially compliant with the EPCglobal C1G2 standard [21] used by commercial devices. The standardized protocol, which is similar to our protocol and also based on slotted ALOHA, only supports the parameter f in the frame header. It does not support tag sampling with probability p. We will explain our implementation of tag sampling in the next paragraph. Moreover, as specified by EPC C1G2 [21], each tag sends a 16-bit random number in its chosen time slot as its response to reader, in order for the reader to detect tag collisions in time slots. By contrast, we don't need the information of tag collision, and we want to save protocol time cost by shortening the length of each tag's response to only 1 bit or just a pulse (for example, by shortening the RN16 response of tags to just one bit in EPC C1G2 [21]). This will need the modification of communication protocol stack both on the reader side [22] and on the tag side [23].

TABLE I
NOTATIONS USED

n, n'	number of tags in the tag sets N and N', respectively
u, d, m	number of tags in union, difference and intersection of N and N'
B, B'	bitmaps that encode the tag sets N and N', respectively
f, f'	lengths of the bitmaps B and B', respectively, assuming $f \geq f'$
p	sampling probability controlling the fraction of tags giving replies
B^*	expanded OR bitmap of B and B' as in Fig. 4, and its length is f
ω^*	load factor of expanded OR bitmap B^*, with $\omega^* = \frac{pd}{f} + \frac{pn'}{f'}$
ω, ω'	load factor of the bitmaps B and B', respectively
Z_j	probabilistic event that the jth bit of B^* is equal to zero
X_j, Y_j	probabilistic event that the jth bit in B or B' is equal to zero
1_{X_j}	indicator function of X_j, whose value is one when the event X_j happens or is zero otherwise; 1_{Y_j} and 1_{Z_j} have similar definitions
V^*	fraction of zero bits in the bitmap B^*, i.e., $V^* = \frac{1}{f}\sum_{j=1}^{f} 1_{Z_j}$
V, V'	fraction of zero bits in the bitmaps B and B', respectively

We present an implementation of tag sampling which does not need the tags equipped with simple circuits to manipulate float numbers. Other implementations of sampling won't affect the correction of our conclusion. Let M be a large integer. The reader broadcasts an integer $\lfloor p^* M \rfloor$ instead of a floating number p^*. A tag computes a pseudo-random value $H(id)$, where id is the tag's identifier and H is a pseudo-random hash function. The tag is sampled if $H(id) \bmod M < \lfloor p^* M \rfloor$.

If a tag is sampled for response, it will choose one time slot in the ALOHA frame to send a pulse to reader. The slot selection could also leverage the random function H: The tag computes a hash value $H(id \,|\, r) \bmod f$, where r is a randomly-chosen constant pre-configured with all tags, to make the values of $H(id \,|\, r)$ and $H(id)$ independent of each other. With the implementations of tag sampling and slot selection, we have the following property established.

Property 1 (Pseudorandom Sampling and Time Slot Selection): Consider an arbitrary common tag in the two sets N and N', whose frame sizes are f and f' respectively, with $f \geq f'$. A tag is either sampled for encoding in both frames or neither. If the tag is sampled and does not select the $(j \bmod f')$th slot in the frame of f', then nor will it select the jth slot in the frame of f for $\forall j \in [0, f)$.

Proof: The tag will either be sampled for both frames or be sampled for neither, since the same pseudo-random value $H(id)$ is used for sampling in both frames. Suppose the tag does not select the $(j \bmod f')$th slot in the frame of f', i.e., $H(id \,|\, r) \neq j \mod f'$. Because both f and f' are the powers of two as in (10) and we assume $f \geq f'$, the length of shorter frame f' must be able to divide the length of longer frame f, or say, frame size f a multiple result of frame size f'. Hence, we have $H(id \,|\, r) \neq j \mod f$, which means the jth slot in the longer frame of length f is not selected. ■

We summarize the notations used in Table I.

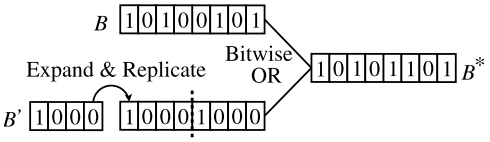

Fig. 4. Expanded OR of two bitmaps B and B'.

B. Offline Joint RFID Estimation

Given two arbitrary snapshots, B and B', which may be taken at different locations or at the same location but different time points, we give the formulas for estimating their difference, intersection and union.

1) Expanded OR: Let f and f' be the lengths of B and B', respectively. Without losing generality, we assume the length of the first bitmap is no smaller than the length of the second: $f \geq f'$. If it is not the case, we can simply swap the two bitmaps and relabel them as B and B' which can satisfy $f \geq f'$. According to (10), we know that both f and f' are the powers of two, i.e., the numbers of the form 2^k where k is an integer. The reason for them to be powers of two is to support the following operation that combines the information from the snapshots of two arbitrary tag sets for joint estimation.

We introduce an auxiliary bitmap, which is called the *expanded OR* between B and B', and is denoted as B^*. The expanded OR, which has been illustrated in Fig. 4, is defined as follows: We know that f is a multiple result of f', because both f and f' are the powers of two as in (10) and we have assumed $f \geq f'$. For example, in Fig. 4, we have illustrated a bitmap B with $f = 8$ bits and another shorter bitmap B' with $f' = 4$ bits. So the size of the former doubles the size of the latter. We will replicate the shorter bitmap B' for $\frac{f}{f'}$ times, such that it is expanded to the same length with the longer bitmap B. We then perform bitwise OR to combine them, and the resulting bitmap B^* is of f bits long.

The operation of expanded OR may appear to be trivial, but the implication of replicating the information of one bitmap when combining with another bitmap is not obvious at all. It requires rigorous analysis for its impact on estimation accuracy as the technique was never used in RFID estimation before.

Although the expanded OR operation in Fig. 4 is designed for jointly analyzing two tag sets, it can be easily extended towards multiple tag sets. However, there is a serious drawback when implementing joint property analysis for multiple tag sets. It is clear that, when jointly analyzing multiple tag sets, we need to apply the similar expanded OR operation to multiple bitmaps and combine them together. Then, the problem is that, as more bitmaps are involved, the resultant OR bitmap will contain a larger ratio of one bits, which may result in bad estimation accuracy. In order for the OR bitmap not to become overly crowded with too many ones, the load factor of each bitmap (i.e., number of tags divided by bitmap length) needs to be smaller, which will cause higher communication cost. By contrast, if we insist that only two tag sets can be involved in the joint property analysis, then the upper bound on each bitmap's load factor could be much larger and more acceptable in practice. In a word, the joint analysis of two tag sets is the most economic form of joint property analysis. Another benefit of the joint analysis of two tag sets is quite useful in practice. With this tool in hand, we can know the volume of product transportation flows between each pair of warehouses, which constitute the so-called traffic matrix and can give us a global view about a logistic supply chain network.

Let N be the original tag set that is encoded by bitmap B. The size of N is denoted as n. Let X_j, $0 \leq j < f$, be the event that the jth bit in B remains zero after the n tags are randomly sampled and encoded into this bitmap.

$$Prob\{X_j\} = (1 - \frac{p}{f})^n \tag{12}$$

Let V be a random variable for the fraction of bits in B that are zeros (We can also measure an instance value of V from the snapshot B. This instance value will be used in the estimator derived later). We have

$$V = \frac{1}{f} \sum\nolimits_{i=0}^{f-1} 1_{X_j},$$

where 1_{X_j} be the indicator variable of X_j, whose value is 1 when the event X_j occurs and 0 otherwise. Clearly, $E(1_{X_j}) = Prob\{X_j\}$. Hence,

$$\begin{aligned} E(V) &= \frac{1}{f} \sum_{i=0}^{f-1} E(1_{X_j}) \\ &= \frac{1}{f} \sum_{i=0}^{f-1} Prob\{X_j\} = (1 - \frac{p}{f})^n. \end{aligned} \tag{13}$$

Let N' be the tag set encoded by B', n' be the set size, and Y_j, $0 \leq j < f'$, be the event that the jth bit in B' is zero.

$$Prob\{Y_j\} = (1 - \frac{p}{f'})^{n'} \tag{14}$$

The above equation is also true for any bit in the expanded B' (for producing B^*). Let V' be the proportion of zero bits in B', which satisfies $V' = \frac{1}{f'} \sum_{i=0}^{f'-1} 1_{Y_j}$. Similar to (13),

$$E(V') = (1 - \frac{p}{f'})^{n'} \tag{15}$$

Let Z_j, $0 \leq j < f$, be the event that the jth bit in B^* is zero. Since this bit is OR of the jth bit in B and the $(j \bmod f')$th bit in B', the event Z_j happens when both events X_j and $Y_{j \bmod f'}$ happens. Hence, we have

$$\begin{aligned} Prob\{Z_j\} &= Prob\{X_j \wedge Y_{j \bmod f'}\} \\ &= Prob\{X_j \,|\, Y_{j \bmod f'}\} \cdot Prob\{Y_{j \bmod f'}\} \\ &= (1 - \frac{p}{f})^d (1 - \frac{p}{f'})^{n'}, \end{aligned} \tag{16}$$

where $Prob\{Y_{j \bmod f'}\} = (1 - \frac{p}{f'})^{n'}$ is from Eq. (14), and we have $Prob\{X_j \,|\, Y_{j \bmod f'}\} = (1 - \frac{p}{f})^d$, because the condition $Y_{j \bmod f'}$, according to Property 1, suggests that all common tags of N and N' won't select the jth slot in the frame of f, and consequently only the d tags in $N \backslash N'$ may select this slot.

Let V^* be the proportion of zero bits in B^*, which satisfies $V^* = \frac{1}{f}\sum_{i=0}^{f-1} 1_{Z_j}$. By a similar process of deriving (13),

$$E(V^*) = (1-\frac{p}{f})^d(1-\frac{p}{f'})^{n'}. \tag{17}$$

In the following, we give the estimators for the joint properties of N and N', including d, d', m, and u.

2) Estimator of Set Difference $d = |N \setminus N'|$: Replacing the second term $(1-\frac{p}{f'})^{n'}$ in (17) by $E(V')$, we have

$$E(V^*) = E(V')(1-\frac{p}{f})^d,$$
$$d = \frac{\ln E(V^*) - \ln E(V')}{\ln(1-\frac{p}{f})}. \tag{18}$$

When the numbers of bits in B^* and B' are large enough, we can approximate $E(V^*)$ and $E(V')$ by their instance values V^* and V' measured from B^* and B'. Then, replacing $E(V^*)$ and $E(V')$ with V^* and V' in (18), we have the estimator $\hat{d}$.

$$\hat{d} = \frac{\ln V^* - \ln V'}{\ln(1-\frac{p}{f})} \tag{19}$$

Such replacement of expected values with instance values will produce asymptotically unbiased estimator of d, because both V^* and V' are the average values of a large number of independent observations (e.g., at least 32 time slots). We briefly explain this point as follows, based on central limit theorem and multivariate delta-method [24]. Since V' (or V^*) is the arithmetic mean of a large number of independent random variables $V' = \frac{1}{f'}\sum 1_{Y_j}$ (or $V^* = \frac{1}{f}\sum 1_{Z_j}$), by the central limit theorem, it approximates a Gaussian distribution, and its variance is inversely proportional to the number of random variables f' (or f). Further consider that $\hat{d}$ in (19) is a function of V' and V^* with continuous first partial derivatives. We can apply the delta-method, and conclude that the estimation $\hat{d}$ approximates a Gaussian distribution, whose expected value is $E(\hat{d}) \approx \frac{\ln E(V^*) - \ln E(V')}{\ln(1-\frac{p}{f})}$. Therefore, by combining it with (18), we have the asymptotic approximation $E(\hat{d}) \approx d$, when f and f' are large enough. Later, we will derive an approximated formula of $E(\hat{d})$ with better accuracy.

Below we use a similar approach to derive the estimators of d', m and u.

3) Estimator of Set Difference $d' = |N' \setminus N|$: By the definitions of d and d', we know that $d = n + d' - n'$, where $n + d'$ is the number of tags in $|N \cup N'|$. Applying it to (17),

$$\begin{aligned} E(V^*) &= (1-\frac{p}{f})^{n+d'-n'}(1-\frac{p}{f'})^{n'} \\ &= E(V)(1-\frac{p}{f})^{d'}E(V') \,/\, (1-\frac{p}{f})^{n'} \\ &= E(V)(1-\frac{p}{f})^{d'}E(V') \,/\, E(V')^{\ln(1-\frac{p}{f})/\ln(1-\frac{p}{f'})}. \end{aligned}$$

Solving the above equation for d', we have

$$d' = \frac{\ln E(V^*) - \ln E(V) - \ln E(V')}{\ln(1-\frac{p}{f})} + \frac{\ln E(V')}{\ln(1-\frac{p}{f'})}.$$

Similar to the previous estimator, we can substitute the expected values $E(V)$, $E(V')$ and $E(V^*)$ by their instance values V, V' and V^* measured from B, B' and B^*, and have an asymptotically unbiased estimator $\hat{d'}$ of approximately Gaussian distribution for the intersection cardinality d'.

$$\hat{d'} = \frac{\ln V^* - \ln V - \ln V'}{\ln(1-\frac{p}{f})} + \frac{\ln V'}{\ln(1-\frac{p}{f'})} \tag{20}$$

4) Estimator of Set Intersection $m = |N \cap N'|$: We rewrite the expected value of V^* in (17) as

$$E(V^*) = (1-\frac{p}{f'})^{n'}(1-\frac{p}{f})^{n-m} = E(V')E(V)(1-\frac{p}{f})^{-m}$$
$$m = \frac{\ln E(V) + \ln E(V') - \ln E(V^*)}{\ln(1-\frac{p}{f})}.$$

By substituting the expected values $E(V)$, $E(V')$ and $E(V^*)$ with their instance values V, V' and V^*, we have an asymptotically unbiased estimator $\hat{m}$ of approximately Gaussian distribution for the intersection cardinality m.

$$\hat{m} = \frac{\ln V + \ln V' - \ln V^*}{\ln(1-\frac{p}{f})} \tag{21}$$

5) Estimator of Set Union $u = |N \cup N'|$: Multiplying both sides of Eq. (17) with $(1-\frac{p}{f})^{n'}$, we have

$$\begin{aligned} &E(V^*)(1-\frac{p}{f})^{n'} \\ &\quad = (1-\frac{p}{f})^{d+n'}E(V') \\ &E(V^*)E(V')^{\ln(1-\frac{p}{f})/\ln(1-\frac{p}{f'})} \\ &\quad = (1-\frac{p}{f})^{u}E(V') \\ &u = \frac{\ln E(V^*) - \ln E(V')}{\ln(1-\frac{p}{f})} + \frac{\ln E(V')}{\ln(1-\frac{p}{f'})}. \end{aligned}$$

By substituting the expected values $E(V')$ and $E(V^*)$ with their instance values V' and V^*, we have an estimator $\hat{u}$ of approximately Gaussian distribution for the union cardinality.

$$\hat{u} = \frac{\ln V^* - \ln V'}{\ln(1-\frac{p}{f})} + \frac{\ln V'}{\ln(1-\frac{p}{f'})} \tag{22}$$

6) Traditional Estimator of $n = |N|$ and $n' = |N'|$: We estimate n and n' based on the classical work in [25]:

$$\hat{n} = \frac{\ln V}{\ln(1-\frac{p}{f})} \qquad \hat{n'} = \frac{\ln V'}{\ln(1-\frac{p}{f'})}, \tag{23}$$

where $\hat{n}$ and $\hat{n'}$ denote the estimated values.

7) Reduction to DiffEstm [20] and PZE [3]: It is interesting to see that when we set $f = f'$, the estimators (19), (20) and (21) are reduced to the estimators of DiffEstm. If we further set the sets identical, i.e., $N = N'$, the union estimator (22) becomes the PZE estimator in [3], similar to those in (23) for a single set. Hence, DiffEstm and PZE are special cases of our estimators. Note that PZE repeats a small frame with a small sampling probability many times to reduce estimation variance. Here we use a larger bitmap (with a larger sampling probability) to reduce variance. The two approaches are equivalent [11].

The most fundamental difference in (19), (20), (21) and (22) from the prior art is the generalized semantics of $V*$, which is

the fraction of bits that are zeros in the bitmap B^* combined from two bitmaps of variable sizes, f and f', where each bit in the smaller bitmap has to be used multiple times in order to enable bitwise OR. However, it is not intuitive why this multiple use of the same bits will work in estimation until we formally analyze the estimation accuracy under such a maneuver of combining information from non-uniform bitmaps.

V. Theoretical Analysis

In this section, we theoretically analyze the accuracy of the joint property estimations $\hat{d}$, $\hat{d'}$, $\hat{m}$ and $\hat{u}$.

A. Preliminaries

In order to derive the variances of joint properties, we need the mean and variance of $\hat{n}$ (or $\hat{n'}$), which can be found in [25]:

$$E(\hat{n}) \approx n + \frac{1}{2p}(e^{\omega} - \omega p - 1) \tag{24}$$

$$Var(\hat{n}) \approx \frac{f}{p^2}(e^{\omega} - \omega p - 1), \tag{25}$$

where $\omega = \frac{pn}{f}$ is called the *load factor* of bitmap B;

$$E(\hat{n'}) \approx n' + \frac{1}{2p}(e^{\omega'} - \omega' p - 1) \tag{26}$$

$$Var(\hat{n'}) \approx \frac{f'}{p^2}(e^{\omega'} - \omega' p - 1), \tag{27}$$

where $\omega' = \frac{pn'}{f'}$ is the load factor of bitmap B'. Note that the two terms $\frac{1}{2p}(e^{\omega} - \omega p - 1)$ and $\frac{1}{2p}(e^{\omega'} - \omega' p - 1)$ in (24) and (26) are negligibly small, making $\hat{n}$ and $\hat{n'}$ roughly unbiased.

The analysis in the paper afterwards utilizes the following three approximations.

Lemma 1: If p, n, f and f' are constants, for large f and f',

$$\big(1 - \frac{p}{f}\big)^n \approx e^{-\frac{pn}{f}}$$

$$\big(1 - \frac{2p}{f}\big)^n - \big(1 - \frac{p}{f}\big)^{2n} \approx -\frac{p^2 n}{f^2} e^{-\frac{2pn}{f}}$$

$$\big(1 - \frac{p}{f} - \frac{p}{f'}\big)^n - \big(1 - \frac{p}{f}\big)^n \big(1 - \frac{p}{f'}\big)^n \approx -\frac{p^2 n}{ff'} e^{-n(\frac{p}{f} + \frac{p}{f'})}.$$

Proof: The first approximation is well-known and its poof is omitted. We prove the second approximation: $\big(1 - \frac{2p}{f}\big)^n - \big(1 - \frac{p}{f}\big)^{2n} = \big(1 - \frac{2p}{f}\big)^n - \big(1 - \frac{2p}{f} + \frac{p^2}{f^2}\big)^n$. Using the Taylor series $(x + \epsilon)^n \approx x^n + n x^{n-1}\epsilon$, it is roughly $-n\big(1 - \frac{2p}{f}\big)^{n-1}\frac{p^2}{f^2}$. Because n is large, it further approximates $-n\big(1 - \frac{2p}{f}\big)^n \frac{p^2}{f^2} \approx -\frac{p^2 n}{f^2} e^{-\frac{2np}{f}}$. We can prove the third approximation similarly using Taylor series: $\big(1 - \frac{p}{f} - \frac{p}{f'}\big)^n - \big(1 - \frac{p}{f}\big)^n \big(1 - \frac{p}{f'}\big)^n = \big(1 - \frac{p}{f} - \frac{p}{f'}\big)^n - \big(1 - \frac{p}{f} - \frac{p}{f'} + \frac{p^2}{ff'}\big)^n \approx -\frac{p^2 n}{ff'} e^{-n(\frac{p}{f} + \frac{p}{f'})}$. ∎

Directly from Lemma 1, we can derive two approximations.

$$(1 - \frac{2p}{f})^d \approx (1 - \frac{p}{f})^{2d} - \frac{p^2 d}{f^2} e^{-\frac{2pd}{f}} \approx (1 - \frac{p^2 d}{f^2}) e^{-\frac{2pd}{f}}$$

$$(1 - \frac{p}{f} - \frac{p}{f'})^m \approx (1 - \frac{p}{f'})^m (1 - \frac{p}{f})^m - \frac{p^2 m}{ff'} e^{-pm(\frac{1}{f} + \frac{1}{f'})}$$

$$\approx (1 - \frac{p^2 m}{ff'}) e^{-pm(\frac{1}{f} + \frac{1}{f'})} \tag{28}$$

B. Probabilistic Distribution of $\hat{n^*}$

By observing the equations (19), (20), (21) and (22) for estimators $\hat{d}$, $\hat{d'}$, $\hat{m}$ and $\hat{u}$, respectively, we find that they have a term in common, which is denoted by $\hat{n^*}$ as follows.

$$\hat{n^*} = \frac{\ln V^*}{\ln(1 - \frac{p}{f})} \tag{29}$$

The physical meaning of $\hat{n^*}$ is the estimation of the number of (either true or replicated) tags in the expanded OR bitmap B^*. As a special case, when $f = f'$, the OR bitmap B^* is equivalent to an encoding of the union of the two tag sets, and we can see that (29) and (22) are equivalent, showing $\hat{n^*} = \hat{u}$. For $\hat{n^*}$ in (29), we give out its mean value and variance.

Theorem 1 (Probabilistic Distribution of $\hat{n^*}$): The cardinality estimation $\hat{n^*}$ defined in (29) approximates a Gaussian distribution, whose expected value and variance are

$$E(\hat{n^*}) \approx n^* + \frac{1}{2p}(e^{\omega^*} - \omega^* p - 1) \tag{30}$$

$$Var(\hat{n^*}) \approx \frac{f}{p^2}(e^{\omega^*} - \omega^* p - 1), \tag{31}$$

where ω^* can be regarded as the load factor of B^*:

$$n^* = d + n'\frac{f}{f'}, \quad \omega^* = \frac{pd}{f} + \frac{pn'}{f'}. \tag{32}$$

Proof: Recall that Z_i, $0 \le i < f$, is the event that the ith bit in bitmap B^* is zero, and 1_{Z_i} is its corresponding indicator variable. Because V^* is the fraction of zero bits in B^*, the number of zero bits in B^* is fV^*, with $fV^* = \sum_{i=0}^{f-1} 1_{Z_i}$. For reducing the complexity of analyzing $Var(f\hat{V^*})$ later, we approximate the bits in B^* as independent, which is true especially when the times of self-replication of B' for generating B^* (see Fig. 4) is a small value as compared with the length f' of B'. Due to $fV^* = \sum_{i=0}^{f-1} 1_{Z_i}$, using the central limit theorem, V^* approximates a Gaussian distribution. Because $\hat{n^*}$ in (29) is a differentiable function of V^*, $\hat{n^*}$ also approximates a Gaussian distribution according to the delta-method [24].

Since $\hat{n^*}$ is a function of V^*, we firstly investigate the mean value and the variance of fV^*. Using $Prob\{Z_i\}$ in (16),

$$\begin{aligned} E(fV^*) &= E(\sum_{i=0}^{f-1} 1_{Z_i}) = \sum_{i=0}^{f-1} E(1_{Z_i}) = \sum_{i=0}^{f-1} Pr\{Z_i\} \\ &= f(1 - \frac{p}{f})^d (1 - \frac{p}{f'})^{n'} \approx f e^{-\frac{pd}{f}} e^{-\frac{pn'}{f'}} = f e^{-\omega^*}. \end{aligned} \tag{33}$$

The variance of fV^* is as follows.

$$\begin{aligned} &Var(fV^*) \\ &= E\big((\sum_{i=0}^{f-1} 1_{Z_i})^2\big) - \big[E(fV^*)\big]^2 \\ &= \big(\sum_{i \ne j} E(1_{Z_i} 1_{Z_j}) + \sum_{i=j} E(1_{Z_i} 1_{Z_j})\big) - \big[E(fV^*)\big]^2 \\ &= \big(\sum_{i \ne j} Prob\{Z_i \wedge Z_j\} + \sum_{i=j} Prob\{Z_i \wedge Z_j\}\big) \\ &\quad - \big[E(fV^*)\big]^2 \end{aligned} \tag{34}$$

When $i = j$, the probability $Prob\{Z_i \wedge Z_j\}$ is

$$Prob\{Z_i \wedge Z_j\} = Prob\{Z_i\} = (1 - \frac{p}{f})^d (1 - \frac{p}{f'})^{n'}$$

otherwise, it is approximately

$$\begin{aligned} &Prob\{Z_i \wedge Z_j\} \\ &\quad \approx (1 - \frac{2p}{f})^d \big(\frac{1}{f'}(1 - \frac{p}{f'})^{n'} + \frac{f'-1}{f'}(1 - \frac{2p}{f'})^{n'}\big), \end{aligned} \tag{35}$$

where $(1 - \frac{2p}{f})^d$ is the probability that neither the ith slot nor the jth slot in B^* contain any of the tags in $N \setminus N'$, whose total number is d, and $\frac{1}{f'}(1 - \frac{p}{f'})^{n'} + \frac{f'-1}{f'}(1 - \frac{2p}{f'})^{n'}$ is the probability that neither slots contain any of the tags in N', whose number is n'. The latter is further explained as follows. When $i = j \mod f'$, the conditional probability of containing no tags from N' is $(1 - \frac{p}{f'})^{n'}$; otherwise, it approximates $(1 - \frac{2p}{f'})^{n'}$. Since $f' \gg 1$, (35) can be simplified as

$$\begin{aligned} &Prob\{Z_i \wedge Z_j\} \\ &\quad \approx (1 - \frac{2p}{f})^d \big(\frac{1}{f'}(1 - \frac{p}{f'})^{n'} + (1 - \frac{2p}{f'})^{n'}\big) \\ &\quad \approx (1 - \frac{2p}{f})^d (1 - \frac{2p}{f'})^{n'} \approx (1 - \frac{2p}{f})^d (1 - \frac{2p}{f})^{\frac{f}{f'}n'}. \end{aligned}$$

Applying the two cases of $Prob\{Z_i \wedge Z_j\}$ to (34), we have

$$\begin{aligned} &Var(fV^*) \\ &\quad \approx f(f-1)(1 - \frac{2p}{f})^d (1 - \frac{2p}{f})^{\frac{f}{f'}n'} \\ &\qquad + f(1 - \frac{p}{f})^d (1 - \frac{p}{f})^{\frac{f}{f'}n'} - f^2 (1 - \frac{p}{f})^{2d} (1 - \frac{p}{f})^{2\frac{f}{f'}n'} \end{aligned}$$

Because $n^* = d + \frac{f}{f'}n'$, and $\omega^* = \frac{pn^*}{f}$,

$$\begin{aligned} &Var(fV^*) \\ &\approx f^2\big((1 - \frac{2p}{f})^{n^*} - (1 - \frac{p}{f})^{2n^*}\big) - f(1 - \frac{2p}{f})^{n^*} + f(1 - \frac{p}{f})^{n^*} \\ &\approx -p^2 n^* e^{-2\frac{pn^*}{f}} - f e^{-2\frac{pn^*}{f}} + f e^{-\frac{pn^*}{f}} \\ &= f e^{-\omega^*}(1 - (1 + \omega^* p) e^{-\omega^*}). \end{aligned} \tag{36}$$

By a similar process, we can derive that

$$Var(fV) \approx f e^{-\omega}(1 - (1 + \omega p) e^{-\omega}) \tag{37}$$

$$Var(f'V') \approx f' e^{-\omega'}(1 - (1 + \omega' p) e^{-\omega'}), \tag{38}$$

which can also be found in literature [25].

Next, with $E(V^*)$ and $Var(V^*)$, we derive the mean and variance of $\hat{n^*} \approx -\frac{f}{p}\ln V^*$. We expand $\ln V^*$ by its Taylor series about q^*, which denotes $E(V^*) \approx e^{-\omega^*}$ in (17).

$$\hat{n^*} = \frac{f}{p}\big(-\ln q^* - \frac{V^* - q^*}{q^*} + \frac{(V^* - q^*)^2}{2\,{q^*}^2} + \dots\big)$$

By truncating the Taylor series after the third term,

$$\begin{aligned} E(\hat{n^*}) &\approx \frac{f}{p}\big(\omega^* - \frac{E(V^*) - q^*}{q^*} + \frac{Var(V^*)}{2\,{q^*}^2}\big) = n^* + \frac{f}{p}\frac{1}{2\,{q^*}^2} \\ &\quad \cdot \frac{q^*}{f}\big(1 - (1 + \omega^* p) q^*\big) = n^* + \frac{1}{2p}\big(e^{\omega^*} - \omega^* p - 1\big). \end{aligned}$$

Similarly, by truncating the Taylor series after the second term,

$$\begin{aligned} Var(\hat{n^*}) &= (\frac{f}{p})^2 Var\big(-\ln q^* - \frac{V^* - q^*}{q^*}\big) = (\frac{f}{p})^2 Var(\frac{V^*}{q^*}) \\ &= \frac{f}{p^2 q^*}\big(1 - (1 + \omega^* p) q^*\big) = \frac{f}{p^2}(e^{\omega^*} - \omega^* p - 1). \end{aligned}$$

Therefore, the equations (30) and (31) have been proved. ■

C. Covariances Among $\hat{n^*}$, $\hat{n}$ and $\hat{n'}$

The estimators equations for $\hat{d}$, $\hat{d'}$, $\hat{m}$ and $\hat{u}$ in (19), (20), (21) and (22) can all be rewritten as the linear combinations of $\hat{n^*}$, $\hat{n}$ and $\hat{n'}$. For example, $\hat{d}$ in (19) can be approximated as $\hat{n^*} - \frac{f'}{f}\hat{n'}$ (to explain in (42) later). Until now, we have derived the probabilistic distributions of of $\hat{n^*}$, $\hat{n}$ and $\hat{n'}$. We however are still not ready to begin the analysis of the variance of the joint property estimators, since $\hat{n^*}$, $\hat{n}$ and $\hat{n'}$ are not mutually independent, and we need to analyze their covariances.

We firstly analyze the covariance of $\hat{n^*}$ and $\hat{n'}$ in Theorem 2, and then analyze the covariance of $\hat{n^*}$ and $\hat{n}$ in Theorem 3.

Theorem 2 (Covariance of $\hat{n^*}$ and $\hat{n'}$): For the cardinality estimations $\hat{n^*}$ in (29) and $\hat{n'}$ in (23), their covariance is

$$Cov(\hat{n^*}, \hat{n'}) \approx \frac{f}{f'} Var(n'). \tag{39}$$

Proof: Due to the page limit, we have to omit the detailed proof, which could be similar to the proof of Theorem 1. ■

Theorem 3 (Covariance of $\hat{n^*}$ and $\hat{n}$): For the cardinality estimations $\hat{n^*}$ in (29) and $\hat{n}$ in (23), their covariance is

$$Cov(\hat{n^*}, \hat{n}) \approx Var(\hat{n}) + (\frac{f}{f'} - 1)\frac{f}{p^2}(e^{\frac{mp}{f}} - \frac{m}{f}p^2 - 1). \tag{40}$$

Proof: Due to the page limit, we have to omit the detailed proof, which could be similar to the proof of Theorem 1. ■

Using a similar procedure of analyzing $Cov(\hat{n^*}, \hat{n'})$ and $Cov(\hat{n^*}, \hat{n})$, we can derive $Cov(\hat{n}, \hat{n'})$. To save space, we present the analysis result directly and omit the detailed proof.

$$Cov(\hat{n}, \hat{n'}) = \frac{f}{p^2}(e^{\frac{mp}{f}} - \frac{m}{f}p^2 - 1) \tag{41}$$

Therefore, we have fully understood the mean values and variances of $\hat{n^*}$, $\hat{n}$ and $\hat{n'}$, and also their mutual covariances.

D. Mean and Variance of $\hat{d}$

In the subsequent subsections, we will analyze the probabilistic properties of joint property estimations $\hat{d}$, $\hat{d'}$, $\hat{m}$ and $\hat{u}$. We begin from $\hat{d}$. Using the definitions of $\hat{n^*}$ in (29) and $\hat{n'}$ in (23), we can rewrite the formula of $\hat{d}$ in (19) as $\hat{d} = \hat{n^*} - \hat{n'} \cdot \ln(1 - \frac{p}{f'})/\ln(1 - \frac{p}{f})$. Using the approximation that $\ln(1 - x) \approx -x$ if x is sufficiently small,

$$\hat{d} \approx \hat{n^*} - \frac{f}{f'}\hat{n'}. \tag{42}$$

Since $\hat{d}$ is a linear combination of $\hat{n^*}$ and $\hat{n'}$, both of which follow Gaussian distributions, $\hat{d}$ also approximates a

Gaussian distribution. Based on $E(n^*)$ in (30) and $E(n')$ in (26), we obtain the expected value of $\hat{d}$ as follows.

$$E(\hat{d}) = d + \frac{1}{2p}(e^{\omega^*} - \omega^* p - 1) - \frac{f}{f'}\frac{1}{2p}(e^{\omega'} - \omega' p - 1)$$

The variance of $\hat{d}$ is derived as follows.

$$\begin{aligned} Var(\hat{d}) &\approx Var(\hat{n^*} - \frac{f}{f'}\hat{n'}) \\ &= Var(\hat{n^*}) + \frac{f^2}{f'^2}Var(\hat{n'}) - 2\frac{f}{f'}Cov(\hat{n^*}, \hat{n'}) \end{aligned}$$

The covariance $Cov(\hat{n^*}, \hat{n'})$ is approximated by (39). Then,

$$Var(\hat{d}) \approx Var(\hat{n^*}) - \frac{f^2}{f'^2}Var(\hat{n'}). \tag{43}$$

E. Mean and Variance of $\hat{u}$

Using $\hat{n'}$ in (23) and $\hat{n^*}$ in (29), by the fact that $\ln(1-x) \approx -x$ when x is sufficiently small, we can rewrite $\hat{u}$ in (22) as $\hat{u} \approx \hat{n^*} - \frac{f-f'}{f'}\hat{n'}$. Hence,

$$\begin{aligned} E(\hat{u}) &\approx E(\hat{n^*}) - \frac{f-f'}{f'}E(\hat{n'}) \\ &= u + \frac{1}{2p}(e^{\omega^*} - \omega^* p - 1) - \frac{f-f'}{f'}\frac{1}{2p}(e^{\omega'} - \omega' p - 1) \\ Var(\hat{u}) &\approx Var(\hat{n^*}) + \frac{(f-f')^2}{f'^2}Var(\hat{n'}) - 2\frac{f-f'}{f'}Cov(\hat{n^*}, \hat{n'}). \end{aligned}$$

Since $Cov(\hat{n^*}, \hat{n'}) \approx \frac{f}{f'}Var(n')$ as in (39), we have

$$Var(\hat{u}) \approx Var(\hat{n^*}) - \frac{f^2 - f'^2}{f'^2}Var(\hat{n'}). \tag{44}$$

F. Mean and Variance of $\hat{m}$

From (21), (23), (29), we have $\hat{m} \approx \hat{n} + \frac{f}{f'}\hat{n'} - \hat{n^*}$. Hence,

$$\begin{aligned} &E(\hat{m}) \\ &\approx E(\hat{n}) + \frac{f}{f'}E(\hat{n'}) - E(\hat{n^*}) \approx m + \frac{1}{2p}(e^{\omega} - \omega p - 1) \\ &\quad + \frac{f}{f'}\frac{1}{2p}(e^{\omega'} - \omega' p - 1) - \frac{1}{2p}(e^{\omega^*} - \omega^* p - 1) \\ &Var(\hat{m}) \\ &\approx Var(\hat{n}) + \frac{f^2}{f'^2}Var(\hat{n'}) + Var(\hat{n^*}) \\ &\quad + 2\frac{f}{f'}Cov(\hat{n}, \hat{n'}) - 2Cov(\hat{n^*}, \hat{n}) - 2\frac{f}{f'}Cov(\hat{n^*}, \hat{n'}). \end{aligned} \tag{45}$$

We have derived that $Cov(\hat{n^*}, \hat{n'})$ in (39), $Cov(\hat{n^*}, \hat{n})$ in (40), and $Cov(\hat{n}, \hat{n'})$ in (41), equation (45) becomes

$$\begin{aligned} &Var(\hat{m}) \\ &\approx Var(\hat{n}) + \frac{f^2}{f'^2}Var(\hat{n'}) + Var(\hat{n^*}) \\ &\quad + 2\frac{f}{p^2}(e^{\frac{mp}{f}} - \frac{m}{f}p^2 - 1) - 2Var(\hat{n}) - 2\frac{f^2}{f'^2}Var(\hat{n'}) \\ &= Var(\hat{n^*}) - Var(\hat{n}) - \frac{f^2}{f'^2}Var(\hat{n'}) \\ &\quad + 2\frac{f}{p^2}(e^{\frac{mp}{f}} - \frac{m}{f}p^2 - 1). \end{aligned} \tag{46}$$

G. Mean and Variance of $\hat{d'}$

From (21), (23), (29), we have $\hat{d'} \approx \hat{n^*} - \hat{n} - \frac{f-f'}{f'}\hat{n'}$. Hence,

$$\begin{aligned} E(\hat{d'}) &\approx E(\hat{n^*}) - E(\hat{n}) - \frac{f-f'}{f'}E(\hat{n'}) \\ &\approx d' + \frac{1}{2p}(e^{\omega^*} - \omega^* p - 1) - \frac{1}{2p}(e^{\omega} - \omega p - 1) \\ &\quad - \frac{f-f'}{f'}\frac{1}{2p}(e^{\omega'} - \omega' p - 1) \\ Var(\hat{d'}) &\approx Var(\hat{n^*}) + Var(\hat{n}) + (\frac{f-f'}{f'})^2 Var(\hat{n'}) \\ &\quad + 2\frac{f-f'}{f'}Cov(\hat{n}, \hat{n'}) - 2Cov(\hat{n^*}, \hat{n}) \\ &\quad - 2\frac{f-f'}{f'}Cov(\hat{n^*}, \hat{n'}). \end{aligned}$$

The three covariances are known. Therefore,

$$\begin{aligned} Var(\hat{d'}) &= Var(\hat{n^*}) + Var(\hat{n}) + (\frac{f-f'}{f'})^2 Var(\hat{n'}) \\ &\quad - 2Var(\hat{n}) - 2\frac{f-f'}{f'}\frac{f}{f'}Var(\hat{n'}) \\ &= Var(\hat{n^*}) - Var(\hat{n}) - \frac{f^2 - f'^2}{f'^2}Var(\hat{n'}). \end{aligned} \tag{47}$$

H. Tight Upper Bound of Variances of $\hat{d}$, $\hat{d'}$, $\hat{m}$ and $\hat{u}$

Because the estimators approximate Gaussian distributions, the accuracy requirements of (1)–(4) can be turned into a set of constraints on bounded $Var(\hat{d})$, $Var(\hat{d'})$, $Var(\hat{m})$, and $Var(\hat{u})$. The following property shows that these constraints can be turned into a single one on $Var(\hat{n^*})$.

Property 2 (Variance Tight Upper Bound): $Var(\hat{n^*})$ is approximately a tight upper bound of the variance of joint property estimations $Var(\hat{d})$, $Var(\hat{d'})$, $Var(\hat{m})$, and $Var(\hat{u})$.

Proof: From (43), we know that $Var(\hat{d}) \leq Var(\hat{n^*})$, and the two sides are equal when $n' = 0$, which makes $Var(\hat{n'})$ zero as shown by (27). It can be seen from (44) that $Var(\hat{u}) \leq Var(\hat{n^*})$, and the two sides are equal when $f = f'$. From (47), we know that $Var(\hat{d'}) \leq Var(\hat{n^*})$, and the two sides are equal when $n = 0$ and $f = f'$. Since (43), (44) and (47) are approximations, we only state in Property 2 that $Var(\hat{n^*})$ is an approximate tight upper bound of $Var(\hat{d})$, $Var(\hat{u})$, $Var(\hat{d'})$.

To further prove $Var(\hat{m}) \leq Var(\hat{n^*})$, we rewrite (46) as

$$\begin{aligned} &Var(\hat{n^*}) - Var(\hat{m}) \\ &\approx Var(\hat{n}) + \frac{f^2}{f'^2}Var(\hat{n'}) - 2\frac{f}{p^2}(e^{\frac{mp}{f}} - \frac{m}{f}p^2 - 1) \\ &\approx \frac{f}{p^2}[(e^{\omega} - \omega p - 1) + \frac{f}{f'}(e^{\omega'} - \omega' p - 1) - 2(e^{\frac{mp}{f}} - \frac{m}{f}p^2 - 1)] \\ &\approx \frac{f}{p^2}[(1-p)(\omega + \frac{f}{f'}\omega' - 2\frac{mp}{f}) + \frac{1}{2}(\omega^2 + \frac{f}{f'}\omega'^2 - 2(\frac{mp}{f})^2)] \\ &\approx \frac{f}{p^2}\big[(1-p)\big((\omega - \frac{mp}{f}) + (\frac{f}{f'}\omega' - \frac{mp}{f})\big) \\ &\quad + \frac{1}{2}\big((\omega^2 - (\frac{mp}{f})^2) + (\frac{f}{f'}\omega'^2 - (\frac{mp}{f})^2)\big)\big] \end{aligned}$$

The third step uses the Taylor expansion of e^{ω} about $\omega = 0$. We want to show that it is greater than or equal to zero. It is a decreasing function of m. Hence, we consider the case of $m = n' \leq n$ when its value becomes the smallest. Hence,

$$\begin{aligned} &Var(\hat{n^*}) - Var(\hat{m}) \\ &\quad \geq \frac{f}{p^2}[(1-p)(\frac{f}{f'}\omega' - \omega') + \frac{1}{2}(\frac{f}{f'}\omega'^2 - \omega'^2)] \end{aligned}$$

Recall that $f \geq f'$, and thus the above expression is non-negative. Therefore, we have $Var(\hat{n^*}) \geq Var(\hat{m})$, and they are equal if $f = f'$ and $m = n' = n$ (or the tag sets N and N' are identical), which can be verified by checking (46). ■

VI. System Parameters

We have already known that the probabilistic distributions of joint estimations $\hat{d}$, $\hat{d'}$, $\hat{m}$ and $\hat{u}$ approximate Gaussian distributions. Our requirement is to bound the estimation error of each joint property by the range of $\pm\theta$ with high probability, as stated in (1)-(4). In this section, we try to set the optimal system parameters f and p to minimize the protocol execution time, subject to the accuracy requirement.

A. Load Factor

Consider the requirement (1) on the union cardinality estimation $\hat{u}$, which specifies a confidence interval of width 2θ at a confidence level of $1-\delta$. For a Gaussian distribution with $E(\hat{u}) \approx u$, the requirement on $\hat{u}$ is translated to:

$$Z_\delta \sqrt{Var(\hat{u})} \leq \theta \quad Var(\hat{u}) \leq \frac{\theta^2}{{Z_\delta}^2}, \tag{48}$$

where Z_δ is the $1-\frac{\delta}{2}$ percentile of standard Gaussian distribution. Similarly, the requirements (2)-(4) can be translated to

$$Var(\hat{m}) \leq \frac{\theta^2}{{Z_\delta}^2}, \quad Var(\hat{d}) \leq \frac{\theta^2}{{Z_\delta}^2}, \quad Var(\hat{d'}) \leq \frac{\theta^2}{{Z_\delta}^2}.$$

In order to cover all possible cases, due to Property 2, all these constraints (1)-(4) can be replaced by $Var(\hat{n^*}) \leq \theta^2/{Z_\delta}^2$.

$$\begin{aligned} \frac{f}{p^2}\left(e^{\omega^*} - \omega^* p - 1\right) &\leq \frac{\theta^2}{{Z_\delta}^2} \\ \frac{n}{p\omega}\left((1-p)\omega^* + \frac{1}{2}\omega^{*2} + \frac{1}{6}\omega^{*3}\right) &\leq \frac{\theta^2}{{Z_\delta}^2} \quad \begin{matrix}\text{Taylor}\\ \text{Expansion}\end{matrix} \end{aligned} \tag{49}$$

From (32), we have $\omega^* = \frac{pd}{f} + \frac{pn'}{f'} = \frac{pn}{f} - \frac{pm}{f} + \frac{pn'}{f'} = \omega - \frac{pm}{f} + \omega'$. If $m = 0$, then $\omega^* = \omega + \omega'$, which maximizes the left side of (49). We consider this worst-case constraint in terms of m. Hence,

$$\begin{aligned} \frac{n}{p\omega}\left((1-p)(\omega+\omega') + \frac{1}{2}(\omega+\omega')^2 + \frac{1}{6}(\omega+\omega')^3\right) &\leq \frac{\theta^2}{{Z_\delta}^2} \\ \frac{1}{\omega}\left((1-p)(\omega+\omega') + \frac{1}{2}(\omega+\omega')^2 + \frac{1}{6}(\omega+\omega')^3\right) &\leq \frac{\theta^2 p}{{Z_\delta}^2 n}. \end{aligned}$$

To satisfy the above constraint in the worst case in terms of n (which is bounded by $n_{\max}$), we have the following equation.

$$\begin{aligned} &\frac{1}{\omega}\left((1-p)(\omega+\omega') + \frac{1}{2}(\omega+\omega')^2 + \frac{1}{6}(\omega+\omega')^3\right) \\ &\quad \leq \frac{\theta^2 p}{{Z_\delta}^2 n_{\max}} \end{aligned} \tag{50}$$

In our system design, we shall set both w and w' for a system-wide optimal load factor. With $\omega' = \omega$, we have

$$\begin{aligned} \frac{1}{\omega}\Big((1-p)2\omega + \frac{1}{2}(2\omega)^2 + \frac{1}{6}(2\omega)^3\Big) &\leq \frac{\theta^2 p}{{Z_\delta}^2 n_{\max}} \\ \omega \leq -\frac{3}{4} + \frac{\sqrt{3}}{4}\sqrt{4p\big(\frac{\theta^2}{n_{\max}{Z_\delta}^2} + 2\big) - 5}. \end{aligned} \tag{51}$$

Because $w = \frac{np}{f}$, it is inversely proportional to the frame size f, which measures the protocol execution time when encoding the tags in B. Hence, we should set our target load factor as

$$\omega = -\frac{3}{4} + \frac{\sqrt{3}}{4}\sqrt{4p\big(\frac{\theta^2}{n_{\max}{Z_\delta}^2} + 2\big) - 5}. \tag{52}$$

We justify our choice of setting $\omega = \omega'$ above. The left side of (50) is an increasing function in both ω and ω'. If we allow $\omega \neq \omega'$ and still set their values to be as small as possible, then one of them will be greater than the right side of (52) and the other will be smaller. Because N and N' are arbitrary tag sets under consideration, it means that some tag sets will be encoded with their load factors greater than the right side of (52) and some others will have smaller load factors. Let N_1 and N_2 be two sets with load factors greater than (52). We should be able to perform joint estimation on any two encoded sets without violating the accuracy requirement. However, if we perform joint estimation on N_1 and N_2, because their load factors are larger than (52), the constraint of (50) will not hold.

B. Frame Size and Sampling Probability

From (51) and $w = \frac{np}{f}$, we have

$$f \geq np \Big/ \left(-\frac{3}{4} + \frac{\sqrt{3}}{4}\sqrt{4p\big(\frac{\theta^2}{n_{\max}{Z_\delta}^2} + 2\big) - 5}\right). \tag{53}$$

Recall that the value of f is set to be a power of two in order to support expanded OR between the snapshots of any two tag sets. We want to choose the optimal sampling probability that minimizes the protocol execution time by keeping the frame size f as small as possible. Hence, we have the formula for the frame size as quoted by equation (10) in Section IV-A: $f = \min_{p\in(0,1]}\{2^{\lceil \log_2(\frac{np}{\omega})\rceil}\}$, where p is a sampling probability and the load factor ω is determined by equation (52). It requires a prior knowledge of n — the number of tags in current tag set N, which can be estimated by an existing low-cost protocol such as GMLE [6], LOF [5] and PET [7]. The optimal sampling probability p^* that minimizes f depends on the pre-determined parameters $n_{\max}$, θ and δ. Hence, it can be pre-computed.

VII. Bitmap Encoding of a Large Tag Set by Multi-Reader and Multi-Antenna Systems

In this section, we discuss how to apply our joint property estimation protocol named JREP to a large-scale distributed RFID system with multiple readers and multiple antennas.

When applying our protocol to practice, a technical issue will emerge: Even though an RFID reader can be equipped

with multiple antennas (for example, Impinj R420 reader can have at most 32 antennas [26]), it is impossible to use just one reader to cover a large business place, e.g., a warehouse or a cargo port. Hence, it is common practice for industries to deploy multiple readers in a large place. The problem is how we can collect a bitmap snapshot of tags through multi-reader and multi-antenna cooperation. If a bitmap snapshot of tag set at any place is available, then plenty of useful information can be derived using our JREP protocol to analyze these snapshots, for example, we can know the number of tags moved from a warehouse to a cargo port within a time interval from t_1 to t_2, by jointly analyzing the snapshot of the warehouse at t_1 and the snapshot of the cargo port at t_2.

For a business place, where a single reader with multiple antennas is deployed, taking a snapshot of all tags is not quite complicated. For most commercial reader products, the antennas that are connected to the same reader are activated at different time intervals in a round robin fashion, such that there do not exist two antennas to simultaneously send packets and interfere with each other. Hence, for each antenna, when its time interval comes, it can broadcast commands independently. All other antennas will keep in a silent listening state and assist the activated antenna to take a bitmap snapshot of tags in its interrogation zone, i.e., if any antennas (including the activated one) senses a busy time slot, then the corresponding bit is one. Given the multiple bitmap snapshots when different antennas are activated, we can combine them by Bitwise OR as illustrated in Fig. 1, to construct a snapshot of tags covered by any antenna of the reader. Such combination of bitmaps can work well if we configure all antennas to use the same MAC layer parameters when they are activated (i.e., Aloha frame length, tag sampling probability and hash function seed).

For a business place, where multiple readers with multiple antennas are deployed, taking a snapshot of all tags is fairly sophisticated. It may appear that, if each reader can produce one bitmap as stated above, then all bitmaps can be easily bitwise ORed to construct a system-wide bitmap. However, this method holds an assumption that each reader is able to take a bitmap snapshot of tags in its interrogation zone without radio interference from other readers. Regretfully, there exist two types of radio collisions in a multi-reader system [27].

Reader-Tag Collision occurs when the signal from a reader is of sufficient strength when received at a second reader that the signal jams the response from a tag to the second reader. This type of interference has negative impact on our JREP protocol, because the signal strength of interfering reader could be multiple magnitude larger than a tag response, and it is difficult to tell from their overlap waveform whether the current time slot is occupied by any tags. Avoidance of this type of interference is mainly through FDMA (frequency-division multiple access) [27]. For example, EPC C1G2 standard recommends each reader to perform channel hopping[1] every 0.4s, so that the probability becomes smaller for nearby reader transmission to be on the same channel with the tag response [21]. Another practical FDMA method proposed by EPC C1G2 is to constrain reader transmission to occupy only a small portion of the center of each channel, and tag response is situated at the channel boundary to avoid collision [21].

Reader-Reader Collision occurs when a tag is located in the intersection part of interrogation zones of multiple readers and two readers communicate with that tag at the same time. This type of radio collision has negative impact on our JREP protocol, because that tag with very simple circuit may not correctly resolve the multiple reader commands received and behave in undesirable ways in its time slot. To avoid this type of collision, many reader scheduling algorithms were proposed in literature [27]–[29]. Most of them are based on TDMA (time-division multiple access) technique, which schedules conflicting readers to different time intervals. Take an early piece of work named Colorwave [28] as an example. It considers an "interference graph" over the readers, wherein there is an edge between two readers if they could lead to a collision when transmitting simultaneously. It attempts to randomly color the readers such that each pair of interfering readers have different colors. Readers with different colors are scheduled to different time intervals to avoid collisions.

Support Long ALOHA Frame. Besides the multiple reader scheduling problem, there is another technical issue about how to encode a large tag set into a long bitmap. According to EPC C1G2 standard [21], the number of time slots in an ALOHA frame is 2^Q, as specified by a Q parameter contained in the *Query* command, which is used by a reader to start a frame. Since by the current RFID standard, Q is in the range from 0 to 15, the length of an ALOHA frame is at most 2^{15}. However, in some situations, we may need to deal with a large tag set with tens of thousands of tags, and thus may need a long ALOHA frame whose size is larger than $2^{15} = 32768$.

To address this problem, we will divide a long ALOHA frame into multiple segments with equal length. Since the length of each segment is no more than 2^{15}, each segment can be started by the *Query* command. These segments will be transmitted one by one, which later can be concatenated to form the whole frame. Since each tag randomly chooses one segment for response, before each *Query* command that starts a segment, we will use the *Select* command to tell tags whether to participate in the segment, or not. The *Select* command may filter tags based on the value of the suffix bits of tag's EPC code, so that all tags are uniformly distributed in the segments.

VIII. Simulation Results

A. Simulation Setting

We evaluate the performance of the proposed JREP protocol and compare it to DiffEstm [20] for joint estimation. For the union estimation, we also want to compare with SRC_M, the best protocol among those that were originally designed to estimate the cardinality of the union of multiple tag sets covered by different RFID readers. Assuming that the optimal sampling probability is known, SRC_M becomes equivalent to DiffEstm in union estimation. Please check the Section III-B for further explanation.

[1]By FCC rules, in North America, EPC UHF RFID protocol can use 902-928MHz ISM bandwidth, which is divided into fifty 500kHz channels [26].

TABLE II

PROBABILITY FOR INTERSECTION CARDINALITY ESTIMATION (m) BY JREP ($\omega = 0.735$) TO MEET THE BOUND $\theta = 500$

$n(\times 10^3)$ \ $n'(\times 10^3)$	[0, 5)	[5, 10)	[10, 15)	[15, 20)	[20, 25)	[25, 30)	[30, 35)	[35,40)	[40, 45)	[45, 50)
[0, 5)	1.00	—	—	—	—	—	—	—	—	—
[5, 10)	1.00	1.00	—	—	—	—	—	—	—	—
[10, 15)	1.00	1.00	1.00	—	—	—	—	—	—	—
[15, 20)	1.00	1.00	1.00	1.00	—	—	—	—	—	—
[20, 25)	1.00	1.00	1.00	1.00	1.00	—	—	—	—	—
[25, 30)	1.00	1.00	1.00	1.00	1.00	1.00	—	—	—	—
[30, 35)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	—	—	—
[35, 40)	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	—	—
[40, 45)	0.99	0.99	0.99	1.00	0.99	1.00	1.00	1.00	1.00	—
[45, 50)	**0.98**	0.99	0.99	1.00	0.99	1.00	1.00	1.00	0.99	0.99

TABLE III

PROBABILITY FOR UNION CARDINALITY ESTIMATION (u) BY JREP ($\omega = 0.735$) TO MEET THE ERROR BOUND $\theta = 500$

$n(\times 10^3)$ \ $n'(\times 10^3)$	[0, 5)	[5, 10)	[10, 15)	[15, 20)	[20, 25)	[25, 30)	[30, 35)	[35,40)	[40, 45)	[45, 50)
[0, 5)	1.00	—	—	—	—	—	—	—	—	—
[5, 10)	1.00	1.00	—	—	—	—	—	—	—	—
[10, 15)	1.00	1.00	1.00	—	—	—	—	—	—	—
[15, 20)	1.00	1.00	1.00	1.00	—	—	—	—	—	—
[20, 25)	0.99	1.00	1.00	1.00	0.99	—	—	—	—	—
[25, 30)	1.00	1.00	1.00	1.00	0.99	1.00	—	—	—	—
[30, 35)	0.99	0.99	0.99	0.99	0.99	1.00	1.00	—	—	—
[35, 40)	0.98	0.98	0.99	0.99	0.97	1.00	0.99	0.99	—	—
[40, 45)	0.97	0.97	0.97	0.97	0.96	0.99	0.99	0.98	0.97	—
[45, 50)	**0.95**	0.96	0.97	0.98	0.96	0.99	0.97	0.98	0.97	0.96

We consider two performance metrics. First, when the two protocols are subject to the same average execution time, we compare *their probabilities of meeting a given error bound*. The probability is measured as the number of joint estimations that meet the error bound divided by the total number of joint estimations performed in the simulation. In favor to DiffEstm and SRC_M, we assume that they know the optimal sampling probability that maximizes their worst-case probabilities of meeting the error bound. The original paper [20] does not give a formula for this optimal sampling probability. We obtain it through exhaustive search by simulations.

Second, given an accuracy requirement as defined in (1)-(4), we compare the *execution times* of the two protocols. The execution time is measured as the number of time slots it takes the reader to encode a tag set in a snapshot bitmap, including the frame size f and other slots needed to give an initial rough estimation of n (Section VI). For JREP, we invoke GMLE [6] in the first phase to generate a raw estimation of n with a 95% confidence interval of $\pm 20\%$ error. The time cost of running GMLE in the first phase hence is roughly $1.544 \cdot {Z_{0.05}}^2/0.2^2 \approx 148$ time slots [6], which is negligibly small as compared with the cost of the second phase that uses Aloha frame to scan a tag set with a few thousands of tags.

The system model is a distributed RFID system of multiple locations, where each reader periodically takes a snapshot of its local set of tags, whose number ranges from 0 to 50000, with $n_{max} = 50000$. The average number of tags in a set is chosen to be 10000, which reflects that the normal business flow of tagged objects is smaller than the worst-case number that the system is designed to handle. The size of each tag set will be taken from a truncated normal distribution $\mathcal{N}(10000, 2000^2)$ in the range of $[0, n_{max}]$. For the accuracy requirement, we set $\theta = 500$ and $\delta = 5\%$ by default. We will also perform simulation with other values of θ and δ.

B. Estimation Accuracy Under Same Execution Time

In this subsection, we compare the estimation accuracy of JREP and DiffEstm (or SRC_M) protocols when they are given the same execution time cost. For JREP, since it uses adaptive frame size with similar load factor to scan different tag sets, its average time cost primarily depends on the average size of tag sets in the system. We choose to set the load factor of JREP to $\omega = 0.735$, which will be explained later in the next paragraph. Since the size of a tag set follows a predefined normal distribution, the average execution time of JREP is about 18405 slots in our simulation (i.e., for a large group G of tag sets whose sizes follows normal distribution $\mathcal{N}(10000, 2000^2)$, we have $\frac{1}{|G|}\sum_{n_i \in G} 2^{\lceil \log_2 \frac{n_i p^*}{\omega} \rceil} \approx 18405$). We will use the same number of time slots for DiffEstm. Hence, DiffEstm uses an ALOHA frame of 18,405 time slots to scan each tag set, no matter whether the tag set is small or large. We show the estimation accuracy of JREP in Table II and Table III, and show the estimation accuracy of DiffEstm in Tables IV and V.

We set the accuracy requirement as $\theta = 500$ and $\delta = 5\%$, i.e., the difference between the estimated values ($\hat{d}$, $\hat{d'}$, $\hat{u}$, $\hat{m}$) and the true values (d, d', u, m) must be bounded by ± 500 with 95% probability. Then, we compute the parameters of

TABLE IV

PROBABILITY FOR INTERSECTION CARDINALITY ESTIMATION (m) BY DIFFESTM WITH FIXED FRAME LENGTH 18405 TO MEET $\theta = 500$

$n(\times 10^3)$ \ $n'(\times 10^3)$	[0, 5)	[5, 10)	[10, 15)	[15, 20)	[20, 25)	[25, 30)	[30, 35)	[35,40)	[40, 45)	[45, 50)
[0, 5)	1.00	—	—	—	—	—	—	—	—	—
[5, 10)	1.00	1.00	—	—	—	—	—	—	—	—
[10, 15)	1.00	1.00	1.00	—	—	—	—	—	—	—
[15, 20)	1.00	1.00	1.00	1.00	—	—	—	—	—	—
[20, 25)	1.00	1.00	1.00	1.00	0.98	—	—	—	—	—
[25, 30)	1.00	1.00	0.99	0.98	0.97	0.94	—	—	—	—
[30, 35)	1.00	1.00	0.98	0.97	0.94	0.89	0.85	—	—	—
[35, 40)	1.00	1.00	0.96	0.96	0.91	0.89	0.82	0.76	—	—
[40, 45)	1.00	0.99	0.94	0.89	0.86	0.79	0.72	0.73	0.67	—
[45, 50)	1.00	0.96	0.92	0.88	0.82	0.77	0.74	0.68	0.61	**0.51**

TABLE V

PROBABILITY FOR INTERSECTION CARDINALITY ESTIMATION (u) BY DIFFESTM WITH FIXED FRAME LENGTH 18405 TO MEET $\theta = 500$

$n(\times 10^3)$ \ $n'(\times 10^3)$	[0, 5)	[5, 10)	[10, 15)	[15, 20)	[20, 25)	[25, 30)	[30, 35)	[35,40)	[40, 45)	[45, 50)
[0, 5)	1.00	—	—	—	—	—	—	—	—	—
[5, 10)	1.00	1.00	—	—	—	—	—	—	—	—
[10, 15)	1.00	1.00	1.00	—	—	—	—	—	—	—
[15, 20)	1.00	1.00	1.00	0.99	—	—	—	—	—	—
[20, 25)	1.00	0.99	0.99	0.98	0.94	—	—	—	—	—
[25, 30)	0.99	0.97	0.95	0.93	0.91	0.87	—	—	—	—
[30, 35)	0.96	0.94	0.90	0.88	0.85	0.81	0.75	—	—	—
[35, 40)	0.91	0.89	0.84	0.84	0.78	0.76	0.71	0.67	—	—
[40, 45)	0.86	0.82	0.77	0.73	0.70	0.66	0.62	0.61	0.56	—
[45, 50)	0.78	0.72	0.69	0.67	0.64	0.60	0.59	0.56	0.51	**0.42**

JREP to meet the accuracy requirement, i.e., compute the value of ω from (52) and then the values of f and p from (10). However, since the value of f is rounded up to the power of two to support the operation of expanded OR, these two parameters are in fact set conservatively. Alternatively, we can set their values empirically through simulations (similar to [11]). We first compute the initial value of ω from (52) and then perform bi-section search to reduce it as small as possible such that the resulting values of f and p from (10) will still satisfy the accuracy requirement. As a result, the load factor ω of JREP protocol to meet the accuracy requirement is 0.735.

Table II shows the probability for the intersection estimation $\hat{m}$ by JREP to meet the error bound ± 500. We simulate two tag sets of sizes n and n', with $n \geq n'$. The first column shows the range from which n is chosen uniformly at random. For example, the first range is $[0, 5000)$ and the last range is $[45000, 50000)$. Be ware that the numbers in the first column have a unit of 1000. The first row shows the range from which n' is chosen. Similarly its first range is $[0, 5000)$ and the last range is $[45000, 50000)$. Because we require $n \geq n'$, the combinations of n and n' above the diagonal are invalid. Each cell in the table shows the probability of meeting the error bound when n and n' are chosen from specified ranges. For example, consider the left bottom cell inside the table, where n is chosen from $[45000, 50000)$ and n' from $[0, 5000)$. The probability of meeting the error bound is 98%, which is measured from 1,000 simulation runs, each with two tag sets N and N' arbitrarily generated and the number m of common tags randomly chosen from the range $[0, n']$.

Table III shows the probability for the union estimation $\hat{u}$ by JREP to meet the error bound ± 500. All probabilities in both tables are greater than 95%, which confirms our analytical results that the proposed estimators satisfy the accuracy constraints in (1) and (3). The same is true for difference estimations $\hat{d}$ and $\hat{d}'$, which are not shown due to space limits.

We use the same number of slots for DiffEstm. In Tables IV, we show the probability for the intersection estimation $\hat{m}$ by DiffEstm to meet the error bound ± 500. This probability is as low as 51%, when performing joint estimation for two tag sets whose sizes are in the range [45000, 50000). This is because, out of its fundamental problems in protocol design, DiffEstm has to use a fixed frame size to scan all different tag sets. Since DiffEstm is configured to use the same time cost as JREP, its frame size parameter is more suitable for average-sized tag sets and becomes insufficient as compared with the largest tag sets. In this simulation, DiffEstm uses an ALOHA frame of 18405 slots to encode a large tag set whose size is between 45000 and 50000, which is surely insufficient. Thus, the accuracy of DiffEstm is much worse in Tables IV than what JREP can achieve at the same time cost in Tables II. Similar phenomenon can be observed in Table V, which shows the probability for the union estimation $\hat{u}$ by DiffEstm to meet the error bound ± 500. The probability is as low as 42%, when both n and n' are between 45000 and 50000.

C. Execution Time to Achieve Same Accuracy

Next, we fix the accuracy requirements with $\theta = 500$ and $\delta = 5\%$, and compare the execution times of JREP and DiffEstm for taking a bitmap snapshot of a tag set. Because DiffEstm is not designed for absolute error bound, there is no formula to compute its frame size. With $n_{max} = 50000$,

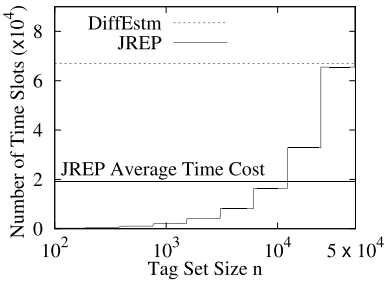

Fig. 5. Time comparison with $\theta = 500$ and $\delta = 5\%$.

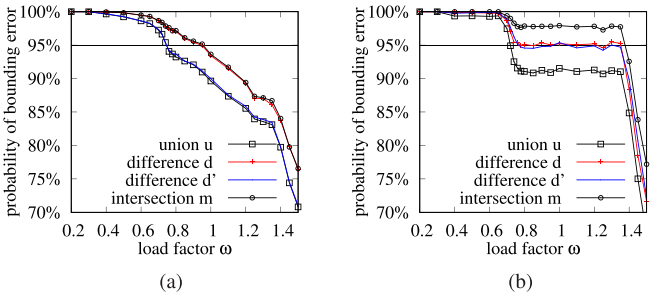

Fig. 6. Impact of load factor on estimation accuracy of joint properties. (a) A large tag set and a small tag set. (b) Two large tag sets.

we use exhaustive search by simulation to find its minimum frame size that can meet the error bound. The results are shown in Fig. 5, where the horizontal axis is the size of a tag set, which varies from 100 to 50000, and the vertical axis is the number of time slots needed to take a bitmap snapshot of the set. Due to the nature of its design, DiffEstm uses a constant frame size of 67000 slots. The frame size of JREP is variable. It is small when the tag set is small. For example, for a set of 5000 tags, the number of time slots needed by JREP is 8192, only 12% of what's needed by DiffEstm. The average time cost of JREP is shown by the solid horizontal line. Similar results are observed from simulations with different accuracy requirements (whose results cannot be included due to space).

D. Impact of Protocol Parameters

In this subsection, we evaluate the impact of protocol parameters, including load factor ω and sampling probability p. In Fig. 6, we plot the relation between estimation accuracy and load factor ω (while the sampling probability $p = 1$), and the accuracy is quantified by the probability of successfully bounding absolute estimation error within the bound ± 500. The higher the probability is, the better the accuracy we have. In Fig. 6(a), we consider the case of performing joint estimation for a large tag set $n \in [45000, 50000)$ and a small tag set $n' \in [0, 5000)$. In Fig. 6(b), we consider the case of jointly analyzing two large tag sets $n, n' \in [45000, 50000)$. Both plots demonstrate two phenomenons: the estimation accuracy of any joint property monotonically decreases as load factor ω increases; the estimation accuracy of the union u is the worst among all joint properties u, d, d' and m, which is consistent with our analysis result in Property 2. To ensure at least $1 - \delta = 95\%$ probability of keeping u's estimation

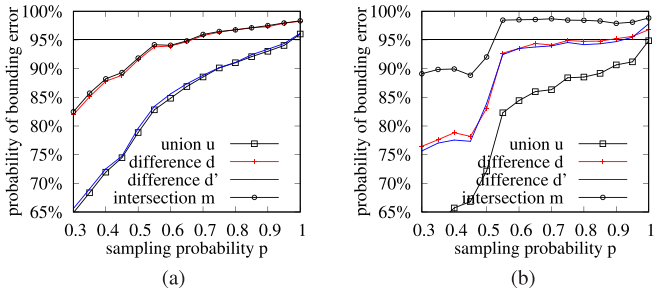

Fig. 7. Impact of sampling probability on estimation accuracy of joint properties. (a) A large tag set and a small tag set. (b) Two large tag sets.

error within ± 500, the load factor ω needs to be smaller than 0.735.

We further evaluate the impact of sampling probability p, while fixing the load factor ω to 0.735. We plot the evaluation result in Fig. 7. In plot (a), we perform joint estimation for a large tag set and a small tag set, while in plot (b), we jointly analyze two large tag sets. The two subfigures show that the probability of successfully bounding estimation error within ± 500 reduces rapidly as the sampling probability p decreases. Hence, when p is smaller than one, sampling error has a significant impact over estimation accuracy. So in most circumstances, we treat $p = 1$ to approximate the optimal configuration of sampling probability, which is consistent with the theoretical calculation of the optimal p^* in equation (10).

IX. Conclusion

This paper studies the problem of joint cardinality estimation: Given any two tag sets in a large RFID system, estimating their union cardinality, intersection cardinality, and difference cardinalities. We propose a solution called JREP that adapts its snapshot based on the size of the tag set that it records. Variable-sized snapshots are combined through an invented operation called "expanded OR" in order to support joint estimation. We derive a full set of estimators, analyze their bias and variance, and provide formulas for setting the optimal system parameters under the predefined constraints on the union/intersection/difference estimation accuracy. We use both theoretical analysis and simulation results to demonstrate that the new solution is much more efficient than the prior art.

References

[1] W. Luo, Y. Qiao, S. Chen, and T. Li, "Missing-tag detection and energy–time tradeoff in large-scale RFID systems with unreliable channels," *IEEE/ACM Trans. Netw.*, vol. 22, no. 4, pp. 1079–1091, Aug. 2014.

[2] T. Li, S. Chen, and Y. Ling, "Efficient protocols for identifying the missing tags in a large RFID system," *IEEE/ACM Trans. Netw.*, vol. 21, no. 6, pp. 1974–1987, Dec. 2013.

[3] M. Kodialam and T. Nandagopal, "Fast and reliable estimation schemes in RFID systems," in *Proc. ACM MOBICOM*, Sep. 2006, pp. 322–333.

[4] H. Han *et al.*, "Counting RFID tags efficiently and anonymously," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[5] C. Qian, H. Ngan, and Y. Liu, "Cardinality estimation for large-scale RFID systems," in *Proc. IEEE PERCOM*, Mar. 2008, pp. 30–39.

[6] T. Li, S. Wu, S. Chen, and M. Yang, "Energy efficient algorithms for the RFID estimation problem," in *Proc. INFOCOM*, Mar. 2010, pp. 1–9.

[7] Y. Zheng, M. Li, and C. Qian, "PET: Probabilistic estimating tree for large-scale RFID estimation," in *Proc. IEEE ICDCS*, Jun. 2011, pp. 37–46.

[8] V. Shah-Mansouri and V. W. S. Wong, "Cardinality estimation in RFID systems with multiple readers," *IEEE Trans. Wireless Commun.*, vol. 10, no. 5, pp. 1458–1469, May 2011.

[9] M. Shahzad and A. X. Liu, "Every bit counts: Fast and scalable RFID estimation," in *Proc. ACM MOBICOM*, Aug. 2012, pp. 365–376.

[10] Y. Zheng and M. Li, "ZOE: Fast cardinality estimation for large-scale RFID systems," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 908–916.

[11] Z. Zhou, B. Chen, and H. Yu, "Understanding RFID counting protocols," in *Proc. ACM MOBICOM*, 2013, pp. 312–327.

[12] H. Liu, W. Gong, X. Miao, K. Liu, and W. He, "Towards adaptive continuous scanning in large-scale RFID systems," in *Proc. IEEE INFOCOM*, Apr./May 2014, pp. 486–494.

[13] W. Gong, K. Liu, X. Miao, and H. Li, "Arbitrarily accurate approximation scheme for large-scale RFID cardinality estimation," in *Proc. IEEE INFOCOM*, Apr./May 2014, pp. 477–485.

[14] H. Vogt, "Efficient object identification with passive RFID tags," in *Proc. IEEE PERCOM*, 2002, pp. 98–113.

[15] J. Myung and W. Lee, "Adaptive splitting protocols for RFID tag collision arbitration," in *Proc. ACM MOBIHOC*, 2006, pp. 202–213.

[16] S.-R. Lee, S.-D. Joo, and C.-W. Lee, "An enhanced dynamic framed slotted ALOHA algorithm for RFID tag identification," in *Proc. IEEE MOBIQUITOUS*, Jul. 2005, pp. 166–172.

[17] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot, "Traffic matrix estimation: Existing techniques and new directions," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 32, no. 4, pp. 161–174, 2002.

[18] J. Cao, A. Chen, and T. Bu, "A quasi-likelihood approach for accurate traffic matrix estimation in a high speed network," in *Proc. IEEE INFOCOM*, Apr. 2008, pp. 412–420.

[19] B. Sheng, Q. Li, and W. Mao, "Efficient continuous scanning in RFID systems," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[20] Q. Xiao, B. Xiao, and S. Chen, "Differential estimation in dynamic RFID systems," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 295–299.

[21] *EPC RFID Class 1 Generation 2 UHF RFID Protocol for Communications at 860–960 MHz Version 1.2.0: Annex G. Multiple- and Dense-Interrogator Channelized Signaling*, EPCglobal, 2008.

[22] M. Buettner and D. Wetherall, "A software radio-based UHF RFID reader for PHY/MAC experimentation," in *Proc. IEEE Int. Conf. RFID*, Apr. 2011, pp. 134–141.

[23] A. P. Sample, D. J. Yeager, P. S. Powledge, A. V. Mamishev, and J. R. Smith, "Design of an RFID-based battery-free programmable sensing platform," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 11, pp. 2608–2615, Nov. 2008.

[24] G. Casella and R. L. Berger, *Statistical Inference*. Pacific Grove, CA, USA: Duxbury Press, 2002.

[25] K.-Y. Whang, B. T. Vander-Zanden, and H. M. Taylor, "A linear-time probabilistic counting algorithm for database applications," *ACM Trans. Database Syst.*, vol. 15, no. 2, pp. 208–229, Jun. 1990.

[26] *SpeedwayR Installation and Operations Guide V5.6*, Impinj, Seattle, WA, USA, 2012.

[27] D. W. Engels and S. E. Sarma, "The reader collision problem," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2002, pp. 6.

[28] J. Waldrop, D. W. Engels, and S. E. Sarma, "Colorwave: An anticollision algorithm for the reader collision problem," in *Proc. IEEE ICC*, May 2003, pp. 1206–1210.

[29] Z. Zhou, H. Gupta, S. R. Das, and X. Zhu, "Slotted scheduled tag access in multi-reader RFID systems," in *Proc. IEEE ICNP*, Oct. 2007, pp. 61–70.

Qingjun Xiao (A'10–M'12) received the B.Sc. degree from the Nanjing University of Posts and Telecommunications, China, in 2003, the M.Sc. degree from Shanghai Jiao Tong University, China, in 2007, and the Ph.D. degree from The Hong Kong Polytechnic University in 2011, all in computer science. After graduation, he joined Georgia State University and the University of Florida as a Post-Doctoral Researcher. He is currently an Assistant Professor with Southeast University, China. His current research interests include protocol and algorithm design in wireless sensor networks, RFID systems, and network traffic measurement. He is a member of the ACM.

Shigang Chen (M'02–SM'12–F'16) received the B.S. degree from the University of Science and Technology of China, Hefei, China, in 1993, and the M.S. and Ph.D. degrees from the University of Illinois at Urbana–Champaign, USA, in 1996 and 1999, respectively, all in computer science. After graduation, he was with Cisco Systems, San Jose, CA, USA, for three years before joining the University of Florida, Gainesville, FL, USA, in 2002, where he is currently a Professor with the Department of Computer and Information Science and Engineering. He served on the technical advisory board for Protego Networks from 2002 to 2003. He has authored or coauthored over 100 peer-reviewed journal/conference papers. He holds 11 U.S. patents. His current research interests include computer networks, Internet security, wireless communications, and distributed computing.

He served in the steering committee of the IEEE IWQoS from 2010 to 2013. He received the IEEE Communications Society Best Tutorial Paper Award in 1999 and the NSF CAREER Award in 2007. He is an Associate Editor of the IEEE/ACM TRANSACTIONS ON NETWORKING, *Computer Networks*, and the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.

Min Chen received the B.E. degree in information security from the University of Science and Technology of China in 2011, and the M.S. and Ph.D. degrees in computer science from the University of Florida, under the supervision of Prof. S. Chen, in 2015 and 2016, respectively. He is currently a Software Engineer with Google Inc. His current research interests include Internet of Things, big network data, next-generation RFID systems, and network security.

Yian Zhou received the B.S. degree in computer science and economics from the Peking University of China in 2010, and the Ph.D. degree in computer and information science and engineering from the University of Florida, Gainesville, FL, USA, in 2015, under the supervision of Prof. S. Chen. She is currently a Software Engineer with Google Inc. Her current research interests include traffic flow measurement, cyber-physical systems, RFID systems, big network data, and cloud computing.

Zhiping Cai (M'10) received the bachelor's, master's, and Ph.D. (Hons.) degrees in computer science and technology from the National University of Defense Technology (NUDT), China, in 1996, 2002, and 2005, respectively. He is currently a Professor with the Networking Engineering Department, College of Computer, NUDT. His current research interests include cloud computing, network virtualization and security. He is a member of the ACM. His doctoral dissertation has been rewarded with the Outstanding Dissertation Award of the Chinese PLA.

Junzhou Luo (M'07) received the B.S. degree in applied mathematics and the M.S. and Ph.D. degrees in computer network from Southeast University, Nanjing, China, in 1982, 1992, and 2000, respectively. He is a Full Professor with the School of Computer Science and Engineering, Southeast University. His research interests are next generation network architecture, network security, cloud computing, and wireless LAN. He is a member of the IEEE Computer Society and Co-Chair of the IEEE SMC Technical Committee on Computer Supported Cooperative Work in Design, and he is a member of the ACM and Chair of the ACM SIGCOMM China.