This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing

On Missing Tag Detection in Multiple-group Multiple-region RFID Systems

Jihong Yu, Lin Chen, Rongrong Zhang, Kehao Wang

***Abstract*—We formulate and study a missing tag detection problem arising in multiple-group multiple-region radio frequency identification (RFID) systems, where a mobile reader needs to detect whether there is any missing event for each group of tags. The problem we tackle is to devise missing tag detection protocols with minimum execution time while guaranteeing the detection reliability requirement for each group. By leveraging the technique of Bloom filter, we develop a suite of three missing tag detection protocols, each decreasing the execution time compared to its predecessor by incorporating an improved version of the Bloom filter design and parameter tuning. By sequentially analysing the developed protocols, we gradually iron out an optimum detection protocol that works in practice.**

***Index Terms*—RFID, missing tag detection, time-efficient.**

I. Introduction

Recent years have witnessed an unprecedented development of the radio frequency identification (RFID) technology. As a promising low-cost technology, RFID is widely used in various applications ranging from inventory control [1], supply chain management [10], [13] and logistics [23] to object tracking and location [22], [7]. An RFID system typically consists of one or several readers and a large number of tags. A reader is a device equipped with a dedicated power source and an antenna and can collect and process information sent by tags in its interrogation region. A tag, on the other hand, is a low-cost microchip labeled with a unique serial number (ID) to identify an object and can receive and transmit radio signals.

Detecting missing tags is one of the most important RFID applications. According to the statistics presented in [21], inventory shrinkage, a combination of shoplifting, internal theft, administrative and paperwork error, and vendor fraud, resulted in 44 billion dollars in loss for retailers in 2014. In this context, RFID provides a promising technology to reduce the loss by monitoring products due to its low cost and non-line-of-sight communication pattern. In this regard, efficiently detecting missing tag events is of fundamental importance and has attracted extensive research attention (cf. Section II on related work on missing tag detection).

We investigate a different problem in this paper motivated by the increasing application of mobile reader [12], [29] and the following practical settings.

- *Multiple groups of tags.* Tags are usually attached to objects belonging to different groups: e.g., different brands of the goods with the high-end brands order-of-magnitude more valuable than their low-end peers. Therefore, the missing tag events are characterized by asymmetrical threshold and reliability requirement across groups.
- *Multiple interrogation regions.* Tags may be unevenly located in multiple interrogation regions: e.g., tags may be located in several rooms or different corners or regions of a large warehouse. Hence, a reader may need to move several times to cover all monitored tags and complete the missing tag detection process.

The problem we consider is to devise missing tag detection protocol with minimum execution time while guaranteeing the detection reliability requirement for each group of tags in multiple-region scenario. In the considered multiple-group multiple-region scenario, all existing missing tag detection protocols cannot work effectively due to the following two reasons. First, existing approaches require the full coverage of tags when executing the detection algorithms, which clearly does not hold in the considered multiple-region scenario. Secondly, existing work does not take into account the heterogeneity among groups and thus either cannot meet the individual reliability requirement, or suffers extremely long detection delay.

To solve this challenging problem, we deliver a comprehensive analysis on the missing tag detection problem in the above multiple-group multiple-region environment and investigate how to devise optimum missing tag detection algorithms. Note that when there are only one group and all tags are with one interrogation region, our problem degenerates to the classical missing tag detection problem studied in the literature.

To design missing tag detection algorithms in the multiple-region multiple-group case, we leverage a powerful technique called *Bloom filter* which is a space-efficient probabilistic data structure for representing a set and supporting set membership queries [2] to detect a missing event. Specifically, we develop a suite of three missing tag detection protocols, each decreasing the execution time compared to its predecessor by incorporating an improved version of the Bloom filter design and parameter tuning. By sequentially analysing the developed protocols, we gradually iron out an optimum detection protocol that works in practice.

The rest of the paper is organised as follows. Section II gives a brief overview of related work. In Section III, we introduce the system model and formalize the multiple-group multiple-region missing tag detection problem. In Section IV to VI, we develop and analyse three detection approaches and compare their performance in terms of execution time. In Section VII, we discuss implementation issues of our proposed algorithms.

J. Yu and L. Chen are with Lab. Recherche Informatique (LRI-CNRS UMR 8623), Univ. Paris-Sud, 91405 Orsay, France, {jihong.yu, chen}@lri.fr. R. Zhang is with Lab. Informatique, Univ. Paris Descartes, France, rongrong.zhang@parisdescartes.fr. K. Wang is with Dept. Inform. Eng., Wuhan University of Technology, China, kehao.wang@whut.edu.cn.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing

Simulation analysis is presented in Section VIII. Finally, we conclude our paper in Section IX.

II. Related Work

A. Missing Tag Detection

Existing missing tag detection protocols can be classified into probabilistic protocols and deterministic protocols, summarized as below.

Probabilistic protocols detect a missing tag event with a predefined probability. Tan *et al.* initiate the study of probabilistic detection and propose a solution called Trusted Reader Protocol (TRP) in [28]. TRP detects a missing tag event by comparing slots in the pre-computed bitmap with those actually collected from the response of the tags in the population. If an expected busy (singleton or collision) slot turns out to be an empty slot, then the missing event is detected. Because the chance for a collision slot to have only missing tags is very small when missing tag size is small, collision slots are less useful than the singleton ones. Given the importance of singleton slots, follow-up works [18] [19] employ multiple seeds to turn empty and collision slots to singleton slots, which increases the detection probability and thus achieves better performance. The latest probabilistic protocol called RUN is proposed in [26] which reduces multiple seeds in one frame in [18] to one seed. The main difference from previous works lies in that RUN considers the influence of unexpected tags and can effectively work in the environment with unexpected tags.

Deterministic protocols, on the other hand, is able to exactly identify which tags are absent. Li *et al.* develop a series of deterministic protocols in [11] to reduce the identification time step by step by reconciling 2-collision slots and iteratively deactivating the tags of which the presence has been verified, respectively. Subsequently, Zhang *et al.* propose identification protocols in [31] which store and compare the bitmaps of tag responses in all rounds and observe the change among the corresponding bits among all bitmaps to determine the present and absent tags. But how to configure the protocol parameters is not theoretically analyzed. Subsequently, Liu *et al.* [15] essentially combine the multi-seed method in [18] with the deactive-based method in [11] to improve the identification performance. More recently, Liu *et al.* [16] further enhance the prior work by reconciling both 2-collision and 3-collision slots and filtering the empty and unreconcilable collision slots to improve time efficiency. To that end, they design a new vector called appended vector and relocate each tag confined in the reconcilable collision slot into a dedicated singleton slot in this vector.

In the considered multiple-group multiple-region scenario, all existed missing tag detection protocols cannot work effectively, because they require the full coverage at each moment, i.e., the reader or a back-end server always exactly know which tags are in the current interrogation region, which does not hold in the considered application scenario. Furthermore, the existing work ignores the heterogeneity among groups and thus cannot meet the individual reliability requirement.

In a broader context, tag identification and tag population estimation protocols sometimes can also be used to detect missing tags. Specifically, tag identification protocols (e.g., [9], [25]) identify all tags in the interrogation region. To detect missing tags, they can be executed to obtain the IDs of the tags present and then missing tags can be found out by comparing the collected IDs with those recorded in the database. However, tag identification protocols are usually time-consuming [11] as they are designed to identify all tags. Moreover, they fail to work when it is not allowed to read the IDs of tags due to privacy concern. Tag estimation protocols (e.g., [32], [3], [27]), on the other hand, estimate the number of tags in the interrogation region. If more than a certain number of tags are absent in RFID systems, a missing tag event can be detected by comparing the estimation and the number of expected tags stored in the database. However, estimation error may be misinterpreted as missing tags and cause detection error, especially when there are only a few missing tags.

Compared to the state-of-the-art development, we formulate the missing tag detection problem in the multiple-group multiple-region scenario, which has not been addressed before. We provide a comprehensive analysis on this new problem and investigate how to devise optimum detection algorithms.

B. Bloom Filter

A Bloom filter is a randomized data structure that is originally from database contexts [2], [24] but has attracted much research attention in networking applications [8]. Specifically, given a null bit array B, Bloom filter records the members of a set $A = \{a_1, a_2, \cdots, a_n\}$ by hashing each member a_i to k positions in B through k hash functions $h_1, h_2, \cdots, h_k$ and setting each position $B[h_v(a_i)] := 1$ for $1 \leq v \leq k$. When a membership query is asked for an element b, every bit $B[h_v(b)]$ is checked for $1 \leq v \leq k$. If all of k bits are set to 1, the Bloom filter asserts $b \in A$; otherwise, $b \notin A$.

III. System Model and Problem Formulation

A. System Model

We consider a grouped RFID system composed of a mobile reader and G groups of tags distributed in R ($R \geq 1$) interrogation regions (e.g., R rooms), concisely referred to as regions. In case where a tag may be physically located in two regions, i.e., regions may overlap one with another, the tag only responses to reader queries regarding to the first region when it is interrogated. In this sense, we can treat the regions as non-overlapping ones.

We use $\mathbb{E}$ to denote the set of the tags which are expected to be present and we denote its cardinality (i.e., the number of expected tags) by $|\mathbb{E}|$. The reader knows the IDs of all tags in $\mathbb{E}$ but does not know the set of tags in each region. For presentation conciseness, we set the ID of group g ($1 \leq g \leq G$) to its index g. We assume every tag knows its group ID through a grouping protocol, e.g. [14]. We also assume the reader knows the approximate number of tags of each group g actually present in each region r ($1 \leq r \leq R$), denoted by n_{gr}. The estimation of n_{gr} can be achieved by the reader by deactivating all tags not belonging to group g (using the ID of group g) and then using any state-of-the-art tag population estimation algorithm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing

To make our analysis generic, we do not impose any physical constraints on tags, which can be either battery-powered active tags or lightweight passive ones energized by radio waves emitted by the reader. We follow the standard Listen-before-talk communication protocol [6] between the reader and tags: the reader initiates communication first by sending commands and broadcasting the parameters to tags, such as the frame size, random seeds, and then each tag responds in its chosen time slot. Consider an arbitrary time slot, if no tag replies in this slot, it is called an *empty slot*; otherwise, it is called a *nonempty slot*. Only one bit is needed to distinguish an empty slot from a nonempty slot: 0 for an empty slot and 1 for a nonempty slot. During the communication, the tag-to-reader transmission rate and the reader-to-tag transmission rate may differ with each other. In practice, the former is either $40-640$kb/s in the FM0 encoding format or $5-320$kb/s in the modulated subcarrier encoding format, while the later is normally $26.7-128$kb/s [5].

Table I summaries main notations used in the paper.

TABLE I
MAIN NOTATIONS

Symbols	Descriptions
G	Number of groups
g	Group index and group ID
R	Number of interrogation regions
r	Region index
$\mathbb{E}$	Set of target tags that need to be monitored
n_{gr}	Number of tags of group g in region r
m_g	Number of missing tags in group g
M_g	Threshold of group g
P_{dg}	Probability of detecting a missing event of group g
α_g	System requirement on the detection reliability for group g
f	Length of Bloom filter in B-detect
k	Number of hash functions in B-detect
s	Hash function seed
P_{fp}	False positive rate of Bloom filter in B-detect
T_B	Execution time of B-detect
f_r	Bloom filter vector size in region r in AB-detect
k_g	Number of hash functions for group g in AB/GAB-detect
$P_{fp,g}$	False positive rate of Bloom filter for g in AB/GAB-detect
T_{AB}	Execution time of AB-detect
f_{gr}	Bloom filter vector length for group g in r in GAB-detect
T_{GAB}	Execution time of GAB-detect

B. Problem Formulation

We are interested in detecting missing tag event for each group g. Let m_g denote the number of missing tags in group g which is of course not known by the reader. Let M_g denote the threshold of group g. A missing event of group g denotes the event where there are at least M_g tags of group g missing in the system. Let P_{dg} denote the probability that the reader can detect a missing event of group g, we formulate the optimum missing tag detection problem as follows.

Definition 1 (Optimum missing tag detection problem). *The optimum missing tag detection problem is to devise an algorithm of minimum execution time which can detect a missing event for each group g with probability $P_{dg} \geq \alpha_g$ if $m_g \geq M_g$, where α_g is the requirement on the detection reliability for group g. When there is only one group in the system, the problem degenerates to the classical missing event detection problem.*

C. Design Rational

To design missing tag detection algorithms in the multiple-region multiple-group case, we leverage a powerful technique called *Bloom filter* which is a space-efficient probabilistic data structure for representing a set and supporting set membership queries [2] to detect a missing event. In our design, we explore the following three natural ideas, each corresponding to a proposed missing tag detection protocol detailed in the next three sections.

Baseline approach. To enable missing tag detection in the multiple-region multiple-group case, we let the reader use the same Bloom filter parameters in each region for each group of tags and construct the Bloom filter based on the responses from the tags to perform missing event detection. This approach, termed as *B-detect*, is a direct application of Bloom filter to solve our problem.

Adaptive approach. In the baseline approach B-detect, the reader uses the same parameters in each region, which may not be optimum in the case when tags are not evenly distributed across regions. Motivated by this observation, we develop an adaptive approach, named *AB-detect*, which enables the reader to use different parameters based on the number of tags in the interrogation region the reader queries. Specifically, for each region r, the reader executes one query, to which tags of all the groups in the region respond. The reader constructs a Bloom filter B_r for each region containing the response and aggregates B_r ($1 \leq r \leq R$) to form a virtual Bloom filter B^{AB}, based on which it detects missing event for each group.

Group-wise approach. We go further by developing a group-wise approach, referred to as *GAB-detect*. In GAB-detect, the reader executes G group-wise queries for each region r. Only tags of group g ($1 \leq g \leq G$) in the interrogation region respond to the g-th query. The reader then constructs a Bloom filter B_{gr}^{GAB} for each group g and aggregates B_{gr}^{GAB} ($1 \leq r \leq R$) to form a virtual Bloom filter B_{g*}^{GAB} using the technique in AB-detect, based on which it detects missing event for group g.

By sequentially analysing the above three approaches and mathematically comparing their performance in terms of execution time, we gradually iron out an optimum detection protocol that works in practice.

IV. The Baseline Approach

In the B-detect design to enable missing tag detection in the multiple-region case, we let the reader use the same parameters in each region and construct the Bloom filter based on the responses from the tags to perform missing event detection. Specifically, B-detect consists of two phases, detailed as below.

A. Protocol Description

Phase 1: Query and feedback collection. The reader performs a query in each region r with the same parameter setting (f, k, s), where f is the length of the Bloom filter

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing

vector, k is the number of independent hash functions used to construct the Bloom filter vector, and s is the seed of the hash functions which is identical for all groups and regions. How their values are chosen is analysed in Sec. IV-B on parameter optimisation. Upon receiving the request, each tag in region r, regardless of the group to which it belongs, selects k slots $(h_v(ID) \mod f)$ $(1 \leq v \leq k)$ in the frame of f slots and replies in these slots. The reader then constructs a Bloom filter vector B_r with the responses from the tags in each region r as follows. Note there are two types of slots: empty slots and nonempty slots. According to the responses from tags, if slot i $(1 \leq i \leq f)$ is empty, the reader sets $B_r(i) = 0$, otherwise it sets $B_r(i) = 1$.

Phase 2: Virtual Bloom filter construction and missing event detection. After interrogating all R regions, the reader combines the Bloom filter vectors B_r $(1 \leq r \leq R)$ to a virtual Bloom filter B by ORing each bit of them, i.e., $B(i) = B_1(i) \vee \cdots \vee B_R(i)$. The reader then performs membership test. For each tag in $\mathbb{E}$, the reader maps its ID into k bits at positions $(h_v(ID) \mod f)$ $(1 \leq v \leq k)$. If all the corresponding bits in B are 1, then the tag is regarded as present. Otherwise, the tag is considered to be missing. The reader reports a missing event in group g if the number of missing tags is at least M_g and no missing event otherwise.

B. Performance Optimisation and Parameter Tuning

The execution time of B-detect, defined as T_B in number of slots, can be written as

$$T_B = R(t_1 + f\delta) \simeq Rf\delta, \tag{1}$$

where t_1 denotes the time for the reader to broadcast the query parameters and δ denotes the slot duration which we normalise to 1 for notation conciseness. In a large RFID system, it holds that $f \gg t_1$, so we ignore t_1. In this subsection, we derive the optimum value of f that minimizes T_B.

It is well-known that there is no false negative in the Bloom filter membership test and the false positive rate P_{fp} for an arbitrary group g can be calculated as follows [2]:

$$P_{fp} = \left[1 - \left(1 - \frac{1}{f}\right)^{(|\mathbb{E}|-m)k}\right]^k \approx (1 - e^{-(|\mathbb{E}|-m)k/f})^k, \tag{2}$$

where $m = \sum_{g=1}^{G} m_g$ denotes the total number of missing tags in all groups.

By rearranging (2), we can express the Bloom filter size as

$$f = \frac{-(|\mathbb{E}| - m)k}{\ln(1 - P_{fp}^{\frac{1}{k}})}. \tag{3}$$

The following theorem derives the optimal values of f and k in the sense of minimising the execution time.

Theorem 1. *The optimum size of the Bloom filter and the optimum number of hash functions in B-detect, denoted by f^* and k^* respectively, that minimize the execution time while satisfying the detection reliability requirement for each group g regardless of m_g, are as follows:*

$$f^* = (|\mathbb{E}| - M) \cdot \frac{k^*}{-\ln(1 - X_{g^*}^{\frac{1}{k^*}})}, \tag{4}$$

$$k^* = \frac{\ln\left(1 - \alpha_{g^*}^{\frac{1}{M_{g^*}}}\right)}{\ln\frac{1}{2}}, \tag{5}$$

where $M = \sum_{g=1}^{G} M_g$, $X_g \triangleq 1 - \alpha_g^{\frac{1}{M_g}}$, and $g^* = \arg\min_g X_g$.

Proof: Recall the definition of a missing event in group g that at least M_g tags are missing, the probability that a missing event can be detected in group g by the reader, defined as P_{dg}, can be computed as

$$P_{dg} = \sum_{i=M_g}^{m_g} \binom{m_g}{i} (1 - P_{fp})^i P_{fp}^{m_g - i}, \tag{6}$$

and P_{dg} has the following property for any $m_g \geq M_g$:

$$\begin{aligned} P_{dg} &= (1 - P_{fp})^{M_g} \sum_{i=M_g}^{m_g} \binom{m_g}{i} (1 - P_{fp})^{i-M_g} P_{fp}^{m_g-i} \\ &= (1 - P_{fp})^{M_g} \sum_{j=0}^{m_g-M_g} \binom{m_g}{j + M_g} (1 - P_{fp})^j P_{fp}^{m_g-M_g-j} \\ &\geq (1 - P_{fp})^{M_g} \sum_{j=0}^{m_g-M_g} \binom{m_g - M_g}{j} (1 - P_{fp})^j P_{fp}^{m_g-M_g-j} \\ &\geq (1 - P_{fp})^{M_g}, \end{aligned} \tag{7}$$

where the first inequality holds due to the inequality below

$$\frac{\binom{m_g}{j+M_g}}{\binom{m_g-M_g}{j}} = \prod_{i=0}^{M_g-1} \frac{m_g - i}{M_g + j - i} \geq 1, \ \forall j \in [0, m_g - M_g],$$

where the equality holds when $m_g = M_g$.

Hence, to ensure the system requirement $P_{dg} \geq \alpha_g$ regardless of m_g, we must ensure the following inequality:

$$(1 - P_{fp})^{M_g} \geq \alpha_g, \text{ or } P_{fp} \leq (1 - \alpha_g^{\frac{1}{M_g}}). \tag{8}$$

Moreover, since P_{fp} is monotonically decreasing and thus $(1 - P_{fp})^{M_g}$ is monotonically increasing with respect to the number of missing tags m_g, meaning that $m_g = M_g$ makes the detection hardest and any m_g larger than M_g will ease the hardness, we thus consider the case where $m_g = M_g$ for $1 \leq g \leq G$ to meet the detection reliability regardless of m_g.

Injecting (8) into (3) with $m_g = M_g$ leads to

$$f \geq \frac{-(|\mathbb{E}| - M)k}{\ln\left[1 - \left(1 - \alpha_g^{\frac{1}{M_g}}\right)^{\frac{1}{k}}\right]},$$

where $M = \sum_{g=1}^{G} M_g$. For clarity, let $X_g \triangleq 1 - \alpha_g^{\frac{1}{M_g}}$. Because f needs to be set such that the required detection reliability for any group is achieved and k is identical for all groups, we have:

$$f = \frac{(|\mathbb{E}| - M)k}{-\ln[1 - (\min_{1 \leq g \leq G} X_g)^{\frac{1}{k}}]}. \tag{9}$$

Without loss of generality, let $g^* = \arg\min_g X_g$ and let the derivative of the right hand side of (9) with respect to k be 0, we can derive that

$$k^* = \frac{\ln \min_g X_g}{\ln\frac{1}{2}} = \frac{\ln\left(1 - \alpha_{g^*}^{\frac{1}{M_{g^*}}}\right)}{\ln\frac{1}{2}}.$$

It can be easily checked that f achieves its minimum as (4) at k^*. The theorem is thus proved. ∎

Remark 1. *Given the practical meaning of k^* and f^*, both of them should been further rounded to the smallest integers not smaller than themselves.*

V. The Adaptive Approach

In B-detect, the reader uses the same parameters in each region, particularly the length of the Bloom filter, which may not be optimum in the case when the tags are not evenly distributed across interrogation regions. Motivated by this observation, we develop another missing tag detection protocol, named *AB-detect*, which enables the reader to use different parameters based on the number of tags in the region the reader queries.

A. Protocol Description

Phase 1: Query and feedback collection. The reader performs a query in each region r with the parameter $(f_r, \{k_g\}_{g=1}^{G}, s)$ where f_r is the length of the Bloom filter vector used in region r, k_g is the number of hash functions used by tags in group g, s is the hash seed which is identical for all groups and regions. There are two differences compared to the baseline approach. First, f_r may be different across different regions but identical across groups; Second, k_g may be different across different groups but identical across regions. We require f_r to be a power-multiple of two, i.e., $f_r = 2^{b_r}$, $(b_r \in \mathbb{N})$. As in B-detect, the reader constructs an f_r-bit Bloom filter vector B_r with the responses from the tags in each region r. Without loss of generality, we assume that $f_1 \le f_2 \le \cdots \le f_R$.

Phase 2: Virtual Bloom filter construction and missing event detection. After interrogating all R regions, the reader first expand B_r to an f_R-bit padded Bloom filter by repeating B_r $\frac{B_R}{B_r}$ times. Denote the padded Bloom filter as PB_r. The reader then combines PB_r $(1 \le r \le R-1)$ and B_R to a virtual Bloom filter B^{AB} by ORing each bit of them, i.e., $B^{AB}(i) = PB_1(i) \vee \cdots \vee PB_{R-1}(i) \vee B_R(i)$ $(1 \le i \le f_R)$, as illustrated in Fig. 1. The reader then performs membership test. For each tag in group g, the reader maps its ID into k_g bits at positions $(h_v(ID) \mod f_R)$ $(1 \le v \le k_g)$. If all the corresponding bits in B^{AB} are 1, then the tag is regarded as present. Otherwise, the tag is considered to be missing. The reader reports a missing event for group g if the number of missing tags in the group g is at least M_g and no missing event otherwise.

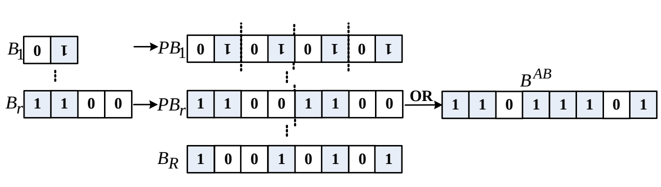


Fig. 1. An illustrative example of constructing virtual Bloom filter.

The following lemma proves that there is no false negative in AB-detect.

Lemma 1. *There is no false negative in AB-detect.*

Proof: It suffices to prove that if a tag is present, it holds that

$$B^{AB}(h_v(a) \bmod f_R) = 1, \ 1 \le v \le k,$$

where a denotes the ID of the tag.

Without loss of generality, assume that the tag a is located in region r. Consider any $v \le k$, let

$$h_v(a) = x + y f_r, \ x, y \in \mathbb{N}, x < f_r.$$

Let $c = \frac{f_R}{f_r}$. By definition of B_r, PB_r and B^{AB}, we have

$$B^{AB}(x + y' f_r) = PB_r(x + y' f_r) = B_r(x) = 1, \tag{10}$$

for $\forall y' \in \mathbb{N}, y' < c$. On the other hand, we have

$$h_v(a) \bmod f_R = x + y f_r \bmod (c f_r) = x + (y \bmod c) f_r.$$

It then follows from (10) that

$$B^{AB}(h_v(a) \bmod f_R) = 1.$$

The proof is thus completed. ∎

B. Performance Optimisation and Parameter Tuning

In this section, we investigate how to tune the parameters in AB-detect to minimise the execution time while ensuring the reliability requirement of each group. We first formulate the false positive rate for each group g, defined as $P_{fp,g}$. Recall the construction of B^{AB} in AB-detect, the probability that any bit in B^{AB} is zero is $\prod_{r=1}^{g}\left(1 - \frac{1}{f_r}\right)^{\sum_{g=1}^{G} k_g n_{gr}}$. The false positive rate for group g can then be derived as

$$P_{fp,g} = \left[1 - \prod_{r=1}^{R}\left(1 - \frac{1}{f_r}\right)^{\sum_{g=1}^{G} k_g n_{gr}}\right]^{k_g}$$
$$\approx \left(1 - e^{-\sum_{r=1}^{R}\sum_{g=1}^{G}\frac{k_g n_{gr}}{f_r}}\right)^{k_g}. \tag{11}$$

The following theorem derives the optimal values of f_r and k_g that minimize the execution time while ensuring the group-wise reliability requirement.

Theorem 2. *The optimum Bloom filter vector size for the region r and the number of hash functions for the group g, denoted as f_r^* and k_g^*, that minimize the execution time while satisfying the detection reliability requirement for each group g regardless of m_g, are as follows:*

$$f_r^* = \frac{\sqrt{\sum_{g=1}^{G} k_g^* n_{gr}} \cdot \sum_{r=1}^{R}\sqrt{\sum_{g=1}^{g} k_g^* n_{gr}}}{\min_g Y_g^*}, \tag{12}$$

$$k_g^* = \frac{\ln(1 - \alpha_g^{\frac{1}{M_g}})}{\ln\frac{1}{2}}, \tag{13}$$

where $Y_g^* \triangleq -\ln[1 - (1 - \alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g^*}}]$. The minimum execution time under the above setting, defined as T_{AB}^*, is:

$$T_{AB}^* = \frac{1}{\min_{1 \le g \le G} Y_g^*}\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} k_g^* n_{gr}}\right)^2. \tag{14}$$

Proof: By the same analysis as the proof of Theorem 1, we need to ensure the following inequality:

$$P_{dg} \ge (1 - P_{fp,g})^{M_g} \text{ or } P_{fp,g} \le (1 - \alpha_g^{\frac{1}{M_g}}). \tag{15}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing

Injecting (11) into (15) leads to

$$\sum_{r=1}^{R}\sum_{g=1}^{G}\frac{k_g n_{gr}}{f_r} \le -\ln[1-(1-\alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g}}],\ 1\le g\le G.$$

For clarity, let $Y_g \triangleq -\ln[1-(1-\alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g}}]$. The above inequality is readily transformed to the following inequality:

$$\sum_{r=1}^{R}\sum_{g=1}^{G}\frac{k_g n_{gr}}{f_r} \le \min_g Y_g.$$

Without loss of generality, let $g_m = \arg\min_g Y_g$. It can be checked that

$$k_g \ge k_{g_m}\frac{\ln(1-\alpha_g^{\frac{1}{M_g}})}{\ln(1-\alpha_{g_m}^{\frac{1}{M_g}})},\ 1\le g\le G. \tag{16}$$

Next we derive the execution time of AB-detect, defined as T_{AB}. We can write T_{AB} as

$$T_{AB} = R\cdot G\cdot t_1' + \sum_{r=1}^{R} f_r \simeq \sum_{r=1}^{R} f_r,$$

where t_1' denotes the time for the reader to broadcast protocol parameters including the group ID for each group. In a large RFID system, it holds that $f_r \gg t_1'$. As RGt_1' is constant, finding the optimum k_g and f_r is equivalent to solving the following optimisation problem:

$$\text{Minimize: } T'_{AB} = \sum_{r=1}^{R} f_r \tag{17}$$

$$\text{Subject to: } \sum_{r=1}^{R}\sum_{g=1}^{G}\frac{k_g n_{gr}}{f_r} \le Y_{g_m}. \tag{18}$$

The corresponding Lagrange function can be defined as

$$\mathcal{L}(f_r,\lambda) = \sum_{r=1}^{R} f_r + \lambda\left(\sum_{r=1}^{R}\sum_{g=1}^{G}\frac{k_g n_{gr}}{f_r} - Y_{g_m}\right).$$

Solving $\nabla_{f_r,\lambda} = 0$ yields the following optimum for f_r:

$$f_r^* = \frac{\sqrt{\sum_{g=1}^{G} k_g n_{gr}}\cdot\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} k_g n_{gr}}}{Y_{g_m}}.$$

T'_{AB} thus achieves its minimum with respect to f_r as below:

$$T'^{*}_{AB} = \frac{\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} k_g n_{gr}}\cdot\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} k_g n_{gr}}}{Y_{g_m}} = \frac{1}{Y_{g_m}}\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} k_g n_{gr}}\right)^2.$$

It can be checked that T'_{AB} is monotonously increasing in k_g. Recall (16), it holds that T'_{AB} achieves its minimum as below when the equality in (16) holds:

$$T'^{*}_{AB} = \min_{k_{g_m}} \frac{k_{g_m}\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G}\frac{\ln(1-\alpha_g^{\frac{1}{M_g}})}{\ln(1-\alpha_{g_m}^{\frac{1}{M_g}})}n_{gr}}\right)^2}{Y_{g_m}}. \tag{19}$$

In the above equation, $\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G}\frac{\ln(1-\alpha_g^{\frac{1}{M_g}})}{\ln(1-\alpha_{g_m}^{\frac{1}{M_g}})}n_{gr}}\right)^2$ is a constant. Hence, T'_{AB} is minimized when $\frac{Y_{g_m}}{k_{g_m}}$ is maximized. By performing straightforward algebraic analysis, we can derive that when $k^*_{g_m} = \frac{\ln(1-\alpha_{g_m}^{\frac{1}{M_{g_m}}})}{\ln\frac{1}{2}}$, $\frac{Y_{g_m}}{k_{g_m}}$ is maximized. Hence, T'_{AB} is minimized at $k^*_g = \frac{\ln(1-\alpha_g^{\frac{1}{M_g}})}{\ln\frac{1}{2}}$ for $1\le g\le G$. Injecting k^*_g into (19) completes our proof. ■

Remark 2. *As k^*_g needs to be an integer and f_r a power-multiple of two, they need to be rounded to the smallest integer and power-multiple of two not smaller than themselves.*

C. Performance Comparison: B-detect vs. AB-detect

Theorem 3. *Given the optimum parameters in both B-detect and AB-detect, the following relationship between the minimum execution time of B-detect T^*_B and that of AB-detect T^*_{AB} holds: $\frac{1}{R} \le \frac{T^*_{AB}}{T^*_B} \le 2$.*

Proof: Recall (4), (5), (13), (14) and Y^*_g in Theorem 2, with some algebraic operations, it can be known that $-\ln(1-X_{g^*}^{\frac{1}{k^*}})$ in (4) is equal to $\min_g Y^*_g$ and $k^* \ge k^*_g$ for $\forall g$. We then have

$$T^*_{AB} \le \frac{k^*}{\min_g Y^*_g}\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} n_{gr}}\right)^2.$$

Let $\overline{T}^*_{AB} \triangleq \frac{k^*}{\min_g Y^*_g}\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} n_{gr}}\right)^2$ and further recall (1), we have

$$\frac{\overline{T}^*_{AB}}{T^*_B} = \frac{\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} n_{gr}}\right)^2}{R*\sum_{r=1}^{R}\sum_{g=1}^{G} n_{gr}}.$$

Expanding $\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} n_{gr}}\right)^2$ leads to

$$\begin{aligned}&\left(\sum_{r=1}^{R}\sqrt{\sum_{g=1}^{G} n_{gr}}\right)^2\\ &= \sum_{r=1}^{R}\sum_{g=1}^{G} n_{gr} + \sum_{i=1}^{R-1}\sum_{r=i+1}^{R} 2\sqrt{\sum_{g=1}^{G} n_{gi}\cdot\sum_{g=1}^{G} n_{gr}}\\ &\le \sum_{r=1}^{R}\sum_{g=1}^{G} n_{gr} + \sum_{i=1}^{R-1}\sum_{r=i+1}^{R}\left(\sum_{g=1}^{G} n_{gi} + \sum_{g=1}^{G} n_{gr}\right)\\ &\le R\sum_{r=1}^{R}\sum_{g=1}^{G} n_{gr}.\end{aligned}$$

To guarantee that f_r is power-multiple of two, we need to at most double it. It thus holds that $\frac{\overline{T}^*_{AB}}{T^*_B} \le 2$. On the other hand, the low bound of the ratio $\frac{\overline{T}^*_{AB}}{T^*} = \frac{1}{R}$ occurs if all tags are located in only one region. It can also be noted that $T^*_{AB} = \overline{T}^*_{AB}$ when both M_g and α_g are identical across all groups. Therefore, it holds that $\frac{1}{R} \le \frac{T^*_{AB}}{T^*_B} \le 2$. ■

Theorem 3 leads to the following engineering implications.

- In the worst case, AB-detect doubles the execution time compared to B-detect;
- In a large asymmetric system where the number of regions R is large, AB-detect can achieve significant performance gain.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing

VI. The Group-wise Approach

In AB-detect, the reader constructs one Bloom filter that contains the response bits of tags of all groups in the interrogation region. Mixing responses from tags of different group may cause "interference" among groups and thus may increase the detection time for certain groups. Motivated by this observation, we develop a group-wise approach, termed as *GAB-detect*, in which the reader queries one group each time and constructs group-wise Bloom filters to eliminate the inter-group interference.

A. Protocol Description

Phase 1: Query and feedback collection. The reader performs G queries in each region r. In the g-th query ($1 \le g \le G$), the reader broadcasts a tetrad (g, k_g, f_{gr}, s) where g is the group ID of group g, k_g is the number of hash functions used by group g tags, f_{gr} is the Bloom filter size used in region r for group g, s is the hash seed which is identical for all regions and groups. Again, we require f_{gr} to be a power-multiple of two. Without loss of generality, we assume that $f_{g1} \le f_{g2} \le \cdots \le f_{gR}$. When receiving the query, each tag compares its group ID with g. If the tag does not belong to the group being queried, it keeps silent and waits for the next query. Otherwise, the tag selects k_g positions $(h_v(ID) \mod f_{gr})$ $(1 \le v \le k_g)$ in the frame of f_{gr} slots and transmits a short response at each of the k_g slots. The reader then constructs a Bloom filter for each group g and each region r, denoted by B_{gr}^{GAB}.

Phase 2: Virtual Bloom filter construction and missing event detection. After interrogating all R regions, the reader combines B_{gr}^{GAB} $(1 \le r \le R-1)$ to a virtual Bloom filter B_{g*}^{GAB} for each group g by using the expansion and combination technique in AB-detect. The reader then performs membership test for each group g by using B_{g*}^{GAB}.

B. Performance Optimisation and Parameter Tuning

In this section, we investigate how to tune protocol parameters in GAB-detect to minimise the execution time while ensuring the reliability requirement of each group. We first derive the false positive rate of GAB-detect for any group g, defined as $P_{fp,g}$. Recall the construction of B_{g*}^{GAB}, the probability that any bit in B_{g*}^{GAB} is zero is $\prod_{r=1}^{R}\left(1-\frac{1}{f_{gr}}\right)^{k_g n_{gr}}$. Hence, the false positive rate for group g can be derived as

$$P_{fp,g} = \left[1-\prod_{r=1}^{R}\left(1-\frac{1}{f_{gr}}\right)^{k_g n_{gr}}\right]^{k_g} \approx \left(1-e^{-\sum_{r=1}^{R}\frac{k_g n_{gr}}{f_{gr}}}\right)^{k_g}. \quad (20)$$

The following theorem derives the optimal values of f_{gr} and k_g that minimize the execution time while ensuring the group-wise reliability requirement.

Theorem 4. *The optimum Bloom filter vector size and number of hash functions for group g in region r, denoted as f_{gr}^* and k_g^*, that minimize the execution time while satisfying the detection reliability requirement for each group g regardless of m_g, are:*

$$f_{gr}^* = \frac{\sqrt{n_{gr}}\cdot\sum_{r=1}^{R}\sqrt{n_{gr}}}{Z_g^*}, \quad (21)$$

$$k_g^* = \frac{\ln(1-\alpha_g^{\frac{1}{M_g}})}{\ln\frac{1}{2}}, \quad (22)$$

The minimum execution time under the above setting, defined as T_{GAB}^*, is:

$$T_{GAB}^* = \sum_{g=1}^{G}\frac{\left(\sum_{r=1}^{R}\sqrt{n_{gr}}\right)^2}{Z_g^*}, \quad (23)$$

where $Z_g^* \triangleq \frac{\ln[1-(1-\alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g^*}}]}{-k_g^*}$.

Proof: By the same analysis as the proof of Theorem 1, we need to ensure the following inequality:

$$P_{fp,g} \le (1-\alpha_g^{\frac{1}{M_g}}). \quad (24)$$

Injecting (20) into (24) leads to

$$\sum_{r=1}^{R}\frac{k_g n_{gr}}{f_{gr}} \le \frac{-\ln[1-(1-\alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g}}]}{k_g}.$$

For clarity, let $Z_g \triangleq \frac{-\ln[1-(1-\alpha_g^{\frac{1}{M_g}})^{\frac{1}{k_g}}]}{k_g}$.

Furthermore, the execution time of GAB-detect, defined as T_{GAB}, can be derived as follows

$$T_{GAB} = RCt_1' + \sum_{g=1}^{G}\sum_{r=1}^{R} f_{gr} \simeq \sum_{g=1}^{G}\sum_{r=1}^{R} f_{gr}.$$

Finding the optimum f_{gr} and k_g is equivalent to solving the following optimisation problem:

$$\text{Minimize:} \ T_{GAB}' = \sum_{g=1}^{G}\sum_{r=1}^{R} f_{gr} \quad (25)$$

$$\text{Subject to:} \ \sum_{r=1}^{R}\frac{n_{gr}}{f_{gr}} \le Z_g, \ 1 \le g \le G. \quad (26)$$

The above optimization problem can be further decomposed to G sub-problem where sub-problem g $(1 \le g \le G)$ is specified as below:

$$\text{Minimize:} \ \sum_{r=1}^{R} f_{gr}$$

$$\text{Subject to:} \ \sum_{r=1}^{R}\frac{n_{gr}}{f_{gr}} \le Z_g.$$

We use the method of Lagrange multiplier to solve each sub-problem g. The Lagrange function can be defined as

$$\mathcal{L}(f_{gr}, \lambda_g) = \sum_{r=1}^{R} f_{gr} + \lambda_g\left(\sum_{r=1}^{R}\frac{n_{gr}}{f_{gr}} - Z_g\right). \quad (27)$$

Solving $\nabla_{f_{gr},\lambda_g} = 0$ yields the following optimum:

$$f_{gr} = \frac{\sqrt{n_{gr}}\cdot\sum_{r=1}^{R}\sqrt{n_{gr}}}{Z_g^*},$$

where Z_g^* is the maximum of Z_g achieved at $k_g^* = \frac{\ln(1-\alpha_g^{\frac{1}{M_g}})}{\ln\frac{1}{2}}$. Injecting k_g^* into T_{GAB} yields the optimum of T_{GAB} and completes the proof. ■

C. Performance Comparison: AB-detect vs. GAB-detect

In this section, we compare the execution time of AB-detect and GAB-detect.

Theorem 5. *When f_r^* in (12) and f_{gr}^* in (21) are power-multiples of two, it holds that $T_{AB}^* \geq T_{GAB}^*$.*

Proof: Recall Y_g in Theorem 2 and Z_g^* in Theorem 4 and let $x_{gr} \triangleq k_g^* n_{gr}$, we can rearrange (23) as

$$T_{GAB}^* = \sum_{g=1}^{G} \frac{\left(\sum_{r=1}^{R} \sqrt{k_g^* n_{gr}}\right)^2}{Y_g^*} \leq \frac{\sum_{g=1}^{G}\left(\sum_{r=1}^{R} \sqrt{k_g^* n_{gr}}\right)^2}{\min_g Y_g^*}$$
$$= \frac{1}{\min_g Y_g^*}\left(\sum_{r=1}^{R}\sum_{g=1}^{G} x_{gr} + 2\sum_{i=1}^{R-1}\sum_{r=i+1}^{R}\sum_{g=1}^{G}\sqrt{x_{gi}x_{gr}}\right).$$

On the other hand, we can expand (14) as

$$T_{AB}^* = \frac{1}{\min_g Y_g^*}\left(\sum_{r=1}^{R}\sum_{g=1}^{G} x_{gr} + 2\sum_{i=1}^{R-1}\sum_{r=i+1}^{R}\sqrt{\sum_{g=1}^{G} x_{gi}\sum_{g=1}^{G} x_{gr}}\right).$$

Moreover, let $\beta_{ir} = \sqrt{\sum_{g=1}^{G} x_{gi}\sum_{g=1}^{G} x_{gr}}$ and $\phi_{ir} = \sum_{g=1}^{G}\sqrt{x_{gi}x_{gr}}$, we have:

$$\phi_{ir}^2 = \sum_{g=1}^{G} x_{gi}x_{gr} + \sum_{g=1}^{G-1}\sum_{w=g+1}^{G} 2\sqrt{x_{gi}x_{gr}\cdot x_{wi}x_{wr}} \quad (28)$$

$$\beta_{ir}^2 = \sum_{g=1}^{G} x_{gi}x_{gr} + \sum_{g=1}^{G-1}\sum_{w=g+1}^{G} (x_{gi}x_{wr} + x_{gr}x_{wi}), \quad (29)$$

It follows from $x_{gi}x_{wr} + x_{gr}x_{wi} \geq 2\sqrt{x_{gi}x_{gr}\cdot x_{wi}x_{wr}}$ that $\phi_{ir}^2 \leq \beta_{ir}^2$. We then have

$$(T_{AB}^* - T_{GAB}^*)\min_g Y_g^* = 2\sum_{i=1}^{R-1}\sum_{r=i+1}^{R}(\beta_{ir} - \phi_{ir}) \geq 0.$$

The proof is thus completed. ■

VII. Discussion

In this section, we discuss some implementation issues of our proposed missing tag detection algorithms.

A. Estimating Tag Population

In our algorithms, the reader needs to estimate the number of tags in n_{gr} in each region and for each group. This may lead to extra overhead prior to missing tag detection. However, this overhead can be limited as the estimation can be achieved in $O(\log n_{gr})$ time using state-of-the-art estimation approaches. Specifically, we can apply two types of methods to estimate n_{gr}: single-group estimator and multi-group estimator. In the single-group estimator, when staying at region r the reader queries with the group ID g and only the tags from g respond. Then it operates like a single-group system. n_{gr} can be estimated by the methods in [3]. On the other hand, multi-group estimator estimates multiple group sizes simultaneously by employing the maximum likelihood estimation method as in [20], which is time-efficient.

Despite the extra overhead due to estimation of n_{gr}, this estimation phase enables the pre-detection of missing tags if the number of missing tags is important (e.g., due to unexpected loss or accidents). More specifically, the reader can achieve pre-detection by comparing the bitmaps constructed by the tag feedbacks and computed a priori by the reader. If a bit that is 1 in the pre-calculated bitmap by reader but turns out to be 0 in the bitmap of the feedbacks, the reader can identify the absence of tags mapped into this slot. If the number of missing tag for a given group exceeds the threshold, a missing event is reported for the group. Consequently, the reader may not need to execute the fine-grained detection algorithms as developed in the last three sections since missing tag events have already been detected in the estimation phase, thus reducing the time cost.

We may wonder whether existing tag estimation algorithms can be used to detect the missing tag event. When the detection requirement is not stringent, e.g., there are a large number of missing tags and the reader only needs to detect a small number of them so as to report a missing event, estimating the number of tags may be used. However, when the detection requirement is stringent, estimating the number of tags is not efficient as it either requires long execution time or cannot satisfy the detection requirement. To demonstrate this, we have conducted more experiments by comparing our approach with the estimation of tag numbers. Under the same detection reliability requirement, the estimation algorithm spends over $48 - 72.6$ times as much time as our algorithms. Therefore, in our approach, we perform a coarse estimation on the tag population for two reasons: 1) our algorithms need a coarse estimation of tag population to configure parameters; 2) in case when the detection requirement is not stringent, this phase allows the reader to quickly detect a missing event.

B. Presence of Unknown/Unexpected Tags

Unknown and unexpected tags can be interpreted as the tags that have not been identified by the reader [17], such as newly arrived products, on which the reader does not have any knowledge. During the interrogation, the unknown tags will respond together with the known tags, which results in the interference to the detection of missing known tags and thus degrades the performance [26] [30]. Note that the malicious behavior, such as DoS and cloning attacks, is out of scope of this paper.

Fortunately, two of our proposed algorithms, AB-detect and GAB-detect, are resistant to the interference caused by unknown tags. The reason is as follows. The unknown tags have not been identified by the reader, so they do not have their individual group IDs [14] such that no group ID in the interrogation messages matches with theirs. Therefore, unknown tags stay silent during the whole detection process.

VIII. Numerical Results

In this section, we evaluate the performance of the proposed approaches in terms of execution time and investigate tradeoffs under different parameter settings.

A. Simulation Settings

We conduct the experiment under both symmetric and asymmetric scenarios under different settings of R, G and M_g.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing

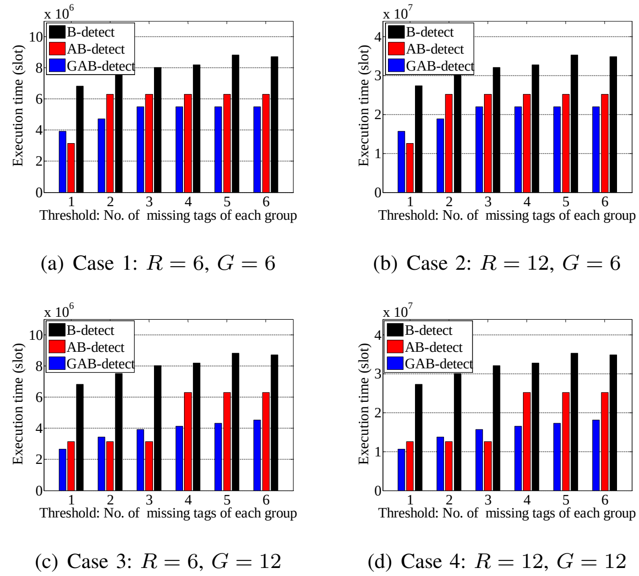
By symmetric/asymmetric, we mean that tag population size in each region r is identical/different. Moreover, we set the same M_g for all group g but vary α_g for each group. Moreover, we use the symmetric transmission rate as in [4] [19] in the numerical analysis and set the transmission time for one bit to be one slot, i.e., $\delta = 1$. The length of group ID is set to $\lceil \log_2 G \rceil$ bits as in [14]. We simulate the optimum parameters settings derived in (4) (5) for B-detect, (12) (13) for AB-detect, and (21) (22) for GAB-detect.

For a comprehensive evaluation, we simulate four cases with different combination of (R, G) in both the symmetric and asymmetric scenarios: case 1: $(6, 6)$, case 2: $(12, 6)$, case 3: $(6, 12)$, and case 4: $(12, 12)$. The required detection reliability for group g ($1 \leq g \leq G$) is set to $\alpha_g = 0.749 + 0.05(g-1)$, i.e., $0.749 \leq \alpha_g \leq 0.999$ in case 1 and case 2, and $\alpha_g = 0.44 + 0.05(g-1)$, i.e., $0.449 \leq \alpha_g \leq 0.999$ in case 3 and case 4. The total number of tags in each region is 12000 and the group size is $12000/G$ in symmetric scenario. In the asymmetric scenario, on the other hand, the total number of tags is randomly chosen from $[1000, 5000]$ in each of the first $R/2$ regions and $[10000, 20000]$ in the remaining regions, and the group size in the same region is identical. The simulation results are obtained by taking the average of 100 independent trials.

B. Performance Evaluation

1) Performance under symmetric scenario: Fig. 2 depicts the execution time of three protocols under different threshold for the four cases in the symmetric scenario. As shown in the results, globally GAB-detect achieves the best time efficiency and AB-detect outperforms B-detect, especially when the detection reliability for each group varies more significantly, i.e., $G = 12$. This can be explained as follows: The frame size in AB-detect and B-detect are set based on $\min_g Y_g^*$ in Theorem 1 and 2, which overkills the groups with larger Y_g^*. In contrast, GAB-detect addresses this limit by eliminating the inter-group interference. We can also observe that in some cases, GAB-detect has longer execution time than AB-detect. This is due to the design requirement that the Bloom filter size needs to be the power-multiple of two. However, globally speaking, GAB-detect still outperforms B-detect. Furthermore, we investigate the actual reliability of the proposed schemes. The results demonstrate that all proposed schemes can detect the missing event with probability one.

2) Performance under asymmetric scenario: Fig. 3 illustrates the execution time for the four cases with different thresholds in the asymmetric scenario. It can be seen from the four subfigures in Fig. 3 that GAB-detect outperforms AB-detect and saves execution time up to 70% in comparison to B-detect. This can be interpreted as follows: In the asymmetric scenarios, the performance gap between AB-detect and B-detect is more significant compared to the symmetric scenario because the frame size in B-detect is identical across the regions regardless of the tag size in an individual region while AB-detect distinguishes the regions with different tag sizes when setting the frame size. Furthermore, we investigate the actual reliability of the proposed schemes and the results show that all proposed schemes can detect the missing event with probability one.



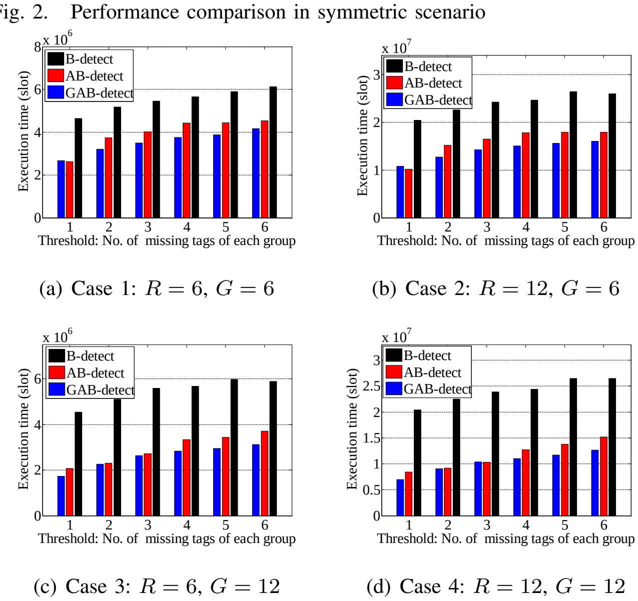
(a) Case 1: $R = 6$, $G = 6$

(b) Case 2: $R = 12$, $G = 6$

(c) Case 3: $R = 6$, $G = 12$

(d) Case 4: $R = 12$, $G = 12$

Fig. 2. Performance comparison in symmetric scenario



(a) Case 1: $R = 6$, $G = 6$

(b) Case 2: $R = 12$, $G = 6$

(c) Case 3: $R = 6$, $G = 12$

(d) Case 4: $R = 12$, $G = 12$

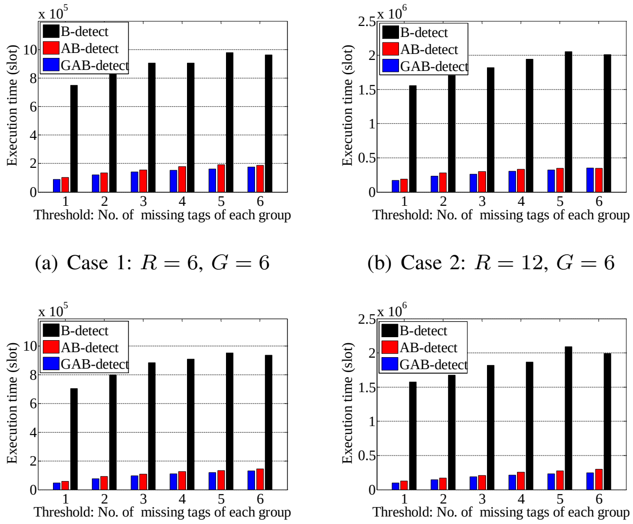
Fig. 3. Performance comparison in asymmetric scenario

To further evaluate the performance and evaluate the analytical results, we conduct a set of numerical analysis in a even more asymmetric scenario where the tag size is randomly chosen from $[50, 100]$ in each of the first $R-1$ regions and from $[5000, 10000]$ in the remaining region. As shown in the four subfigures in Fig. 4, the performance gain of GAB-detect and AB-detect over B-detect is more remarkable. Specifically, the detection time of B-detect is up to 12.6 times as much as that of GAB-detect and AB-detect.

3) Impact of nonidentical M_g: To comprehensively evaluate the performance, we conduct more numerical analysis in both symmetric and asymmetric scenarios which are same with the previous settings except that R is fixed to 6 and $M_g = g$ for group g. Moreover, we also investigate the impact of estimation error ϵ on the performance.

From the results listed in Table II where $(\cdot, \cdot, \cdot)$ represents the time needed by B-detect, AB-detect and GAB-detect, respectively, we can observe that GAB-detect significantly

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2592902, IEEE Transactions on Mobile Computing



(a) Case 1: $R = 6$, $G = 6$ (b) Case 2: $R = 12$, $G = 6$

(c) Case 3: $R = 6$, $G = 12$ (d) Case 4: $R = 12$, $G = 12$

Fig. 4. Performance comparison in more asymmetric scenario

outperforms AB-detect and B-detect when M_g is different for each group. Besides, the execution time increases by less than 11% in the worst case when ϵ varies from 0 to 10%. While on average, B-detect and GAB-detect and AB-detect spend 9% and 6% and 2.6% more time, respectively. Therefore, it is fair to allow $\epsilon = 10\%$.

TABLE II
EXECUTION TIME ($\times 10^6$) UNDER NONIDENTICAL M_g AND ϵ

Scenario	**Number of groups G**		**Estimation error ϵ**
	6	12	
Symmetric	(8.1, 6.3, 4.7)	(8.4, 6.3, 4.3)	0
	(8.9, 6.3, 4.7)	(9.3, 6.3, 4.6)	10%
Asymmetric	(6.1, 3.9, 3.2)	(6.4, 3.1, 2.8)	0
	(6.6, 4.2, 3.4)	(6.9, 3.4, 3.1)	10%

IX. Conclusion

In this paper, we have formulated a missing tag detection problem arising in multiple-group multiple-region RFID systems, where a mobile reader needs to detect whether there is any missing event for each group of tags. By leveraging the technique of Bloom filter, we develop a suite of three missing tag detection protocols, each decreasing the execution time compared to its predecessor by incorporating an improved version of the Bloom filter design and parameter tuning. In our future work, we plan to study the case where multiple mobile readers are available to detect missing tag events and design optimum missing tag detection algorithms in that context.

References

[1] Barcoding Inc. How RFID works for inventory control in the warehouse. Available: http://www.barcoding.com/rfid/inventory-control.shtml.

[2] B. H. Bloom. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970.

[3] B. Chen, Z. Zhou, and H. Yu. Understanding RFID counting protocols. In *ACM MobiHoc*, pages 291–302, 2013.

[4] M. Chen, W. Luo, Z. Mo, S. Chen, and Y. Fang. An efficient tag search protocol in large-scale rfid systems. In *IEEE INFOCOM,*, pages 899–907, 2013.

[5] EPCglobal Inc. Radio-frequency identity protocols class-1 generation-2 UHF RFID protocol for communications at 860 mhz - 960 mhz version 1.0.9. [Online], 2005. Available: http://www.gs1.org.

[6] ETSI. ETSI EN 302 208-1, 2003. Available: http://www.medwelljournals.com/fulltext/?doi=rjasci.2009.57.61.

[7] J. Han, C. Qian, X. Wang, D. Ma, J. Zhao, P. Zhang, W. Xi, and Z. Jiang. Twins: Device-free object tracking using passive tags. In *IEEE INFOCOM*, pages 469–476, 2014.

[8] F. Hao, M. Kodialam, T. Lakshman, and H. Song. Fast dynamic multiple-set membership testing using combinatorial bloom filters. *IEEE/ACM TON*, 20(1):295–304, 2012.

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

[10] C.-H. Lee and C.-W. Chung. Efficient storage scheme and query processing for supply chain management using RFID. In *ACM SIGMOD*, pages 291–302, 2008.

[11] T. Li, S. Chen, and Y. Ling. Identifying the missing tags in a large RFID system. In *ACM MobiHoc*, pages 1–10, 2010.

[12] H. Liu, W. Gong, X. Miao, K. Liu, and W. He. Towards adaptive continuous scanning in large-scale rfid systems. In *IEEE INFOCOM*, pages 486–494. IEEE, 2014.

[13] J. Liu, B. Xiao, K. Bu, and L. Chen. Efficient distributed query processing in large rfid-enabled supply chains. In *IEEE INFOCOM*, pages 163–171, 2014.

[14] J. Liu, B. Xiao, S. Chen, F. Zhu, and L. Chen. Fast rfid grouping protocols. In *IEEE IFOCOM*, pages 1948–1956, 2015.

[15] X. Liu, K. Li, G. Min, Y. Shen, A. X. Liu, and W. Qu. A multiple hashing approach to complete identification of missing rfid tags. *IEEE Transactions on Communications,*, 62(3):1046–1057, 2014.

[16] X. Liu, K. Li, G. Min, Y. Shen, A. X. Liu, and W. Qu. Completely pinpointing the missing RFID tags in a time-efficient way. *IEEE Transactions on Computers*, 64(1):87–96, 2015.

[17] X. Liu, B. Xiao, S. Zhang, and K. Bu. Unknown tag identification in large rfid systems: An efficient and complete solution. *IEEE Transactions on Parallel & Distributed Systems*, 26(6):1775–1788, 2015.

[18] W. Luo, S. Chen, T. Li, and Y. Qiao. Probabilistic missing-tag detection and energy-time tradeoff in large-scale RFID systems. In *ACM MobiHoc*, pages 95–104, 2012.

[19] W. Luo, S. Chen, Y. Qiao, and T. Li. Missing-tag detection and energy–time tradeoff in large-scale RFID systems with unreliable channels. *IEEE/ACM Transactions on Networking*, 22(4):1079–1091, 2014.

[20] W. Luo, Y. Qiao, and S. Chen. An efficient protocol for rfid multigroup threshold-based classification. In *IEEE INFOCOM*, pages 890–898, 2013.

[21] National Retail Federation. National retail security survey. [Online], 2015. Available: https://nrf.com.

[22] L. M. Ni, D. Zhang, and M. R. Souryal. RFID-based localization and tracking technologies. *IEEE Wireless Communications*, 18(2):45–51, 2011.

[23] RFID Journal. The RFID application in logistics and supply chain management, 2009. Available: http://www.etsi.org/deliver/etsi_en/302200_302299/30220801/01.01.01_20/en_30220801v010101c.pdf.

[24] O. Rottenstreich, Y. Kanizo, and I. Keslassy. The variable-increment counting bloom filter. *IEEE/ACM TON*, 22(4):1092–1105, 2014.

[25] M. Shahzad and A. X. Liu. Probabilistic optimal tree hopping for RFID identification. In *ACM SIGMETRICS*, volume 41, pages 293–304, 2013.

[26] M. Shahzad and A. X. Liu. Expecting the unexpected: Fast and reliable detection of missing RFID tags in the wild. In *IEEE INFOCOM*, pages 1939–1947, 2015.

[27] M. Shahzad and A. X. Liu. Fast and accurate estimation of RFID tags. *Networking, IEEE/ACM Transactions on*, 23(1):241–254, 2015.

[28] C. C. Tan, B. Sheng, and Q. Li. How to monitor for missing RFID tags. In *IEEE ICDCS*, pages 295–302, 2008.

[29] L. Xie, Q. Li, C. Wang, X. Chen, and S. Lu. Exploring the gap between ideal and reality: An experimental study on continuous scanning with mobile reader in RFID systems. *IEEE Transactions on Mobile and Computing*, 14(11):2272–2285, 2015.

[30] J. Yu, L. Chen, and K. Wang. Finding needles in a haystack: Missing tag detection in large RFID systems. *arXiv preprint:1512.05228*, 2015.

[31] R. Zhang, Y. Liu, Y. Zhang, and J. Sun. Fast identification of the missing tags in a large RFID system. In *IEEE SECON*, pages 278–286, 2011.

[32] Y. Zheng and M. Li. Zoe: Fast cardinality estimation for large-scale rfid systems. In *INFOCOM, 2013 Proceedings IEEE*, pages 908–916. IEEE, 2013.

Jihong Yu received the B.E degree in communication engineering and M.E degree in communication and information systems from Chongqing University of Posts and Telecommunications, Chongqing, China, in 2010 and 2013, respectively, and is currently pursuing the Ph.D. degree in computer science at the University of Paris-Sud, Orsay, France. His research interests include wireless communications and networking and RFID technologies.

Lin Chen (S07-M10) received his B.E. degree in Radio Engineering from Southeast University, China in 2002 and the Engineer Diploma from Telecom ParisTech, Paris in 2005. He also holds a M.S. degree of Networking from the University of Paris 6. He currently works as associate professor in the department of computer science of the University of Paris-Sud. He serves as Chair of IEEE Special Interest Group on Green and Sustainable Networking and Computing with Cognition and Cooperation, IEEE Technical Committee on Green Communications and Computing. His main research interests include modeling and control for wireless networks, distributed algorithm design and game theory.

Rongrong Zhang received the B.Eng and M.Eng degrees in communication and information systems from Chongqing University of Posts and Telecommunications, Chongqing, China, in 2010 and 2013, respectively, and is currently pursuing the Ph.D. degree in Computer Communication in the Faculty of Mathematics and Computer Science at the University of Paris Descartes, France. Her research interests focus on Wireless Area Body Networks (WBANs) for healthcare and wireless communications.

Kehao Wang received the B.S degree in Electrical Engineering, M.s degree in Communication and Information System from Wuhan University of Technology, Wuhan, China, in 2003 and 2006, respectively, and Ph.D in the Department of Computer Science, the University of Paris-Sud XI, Orsay, France, in 2012. His research interests are cognitive radio networks, wireless network resource management, and data hiding.