An efficient missing tag identification approach in RFID collisions

Jian Su, *Member, IEEE*, Zhengguo Sheng, *Senior Member, IEEE*, Alex. X. Liu, *Fellow, IEEE*, Zhangjie Fu, *Member, IEEE*, and Chenxi Huang

Abstract—Radio frequency identification technology has been widely used to verify the presence of items in many applications such as warehouse management and supply chain logistics. In these applications, the challenge of how to timely identify the missing tags (namely tag searching or missing tag identification) is a key focus. Existing missing tag identification solutions have not achieved their full potentials because collision slots have not been well explored. In this paper, we propose an approach named collision resolving based missing tag identification (CR-MTI) to break through the performance bottleneck of existing missing tag identification protocols. In CR-MTI, multiple tags are allowed to respond with different binary strings in a collision slot. Then, the reader can verify them together by using the bit tracking technology and particularly designed string, thereby significantly improve the time efficiency. CR-MTI also reduces the number of messages transmitted by the reader using customized coding. We further explore the optimal parameter settings to maximize the performance of our proposed CR-MTI. Extensive simulation results show that our proposed CR-MTI outperforms prior art in terms of time efficiency, total executive time and communication complexity.

Index Terms—RFID, missing tag identification, missing rate, time efficiency.

1 Introduction

1.1 Background and Problem Statement

RADIO frequency identification is an important automatic identification technology that enables electromagnetic fields to identify and track objects attached with RFID tags [1]. It has been widely used in various applications including logistics management [2-4], inventory control [5-6] and localization [7-8]. A typical RFID system is composed of a single reader and multiple low-cost tags, where the reader broadcasts a request to tags, and the tags within the reader's coverage feed back with pre-stored IDs. Compared to traditional barcode technology, RFID has overwhelming advantages in many aspects including non-light-of-sight (NLOS), long reading distance, stain-resistant and bulk reading, etc [9]. Tag searching is a key issue in RFID-based applications [10]. According to the report [11], the US retail industry lost as much as $42 billion in 2013 due to theft, administrative errors and supplier fraud, which has become the main reasons of financial losses in the retail industry. The tag searching can help administrators to find out the missing items and to obtain the associated information such as quantity and price. Therefore, it is crucial to apply an efficient tag searching protocol in the reader to timely and accurately locate the missing tags.

The existing tag searching protocols fall into two categories, namely ID-collection [12-18] protocols and missing tag identification [19-22] protocols. The reader maintains an inventory list or a database to store tag IDs of all the items. In an ID-collection protocol, the reader attempts to identify the given set of tags in its vicinity and obtain all the tag IDs (named acquired IDs), and then to find out missing IDs by comparing the acquired IDs with ones in the database. In ID-collection protocols, the reader requires the tags to respond slot by slot and then obtain the corresponding ID in a singleton slot. Many previous work [12-18] have also focused on reducing the collisions and improving the slot efficiency.

Missing tag identification protocols [19-22] have been proposed to accelerate the tag identification. In such protocols, the reader has the prior knowledge of all tag IDs and requires the tags to reply using a framed slotted Aloha protocol. Based on the known IDs, the reader can pre-compute the expected status (singleton, collision or empty) of each tag and form a bitmap. By comparing the expected bitmap and the actual bitmap, the reader can identify the missing tags. For example, if the reader does not receive any response in an expected singleton slot, the corresponding tag can be regarded as missing.

1.2 Limitations of Prior Art

The key limitation of the existing ID-collection protocols is that they are inefficient in verifying the presence of a tag

Manuscript received December 13, 2020; revised April 26, 2021; accepted May 26, 2021. This work is supported by the Natural Science Foundation of China (No.61802196, 61872082, 61472184); the Natural Science Foundation of Jiangsu Province (No.BK20180791); the Natural Science Foundation of Jiangsu Higher Education Institution of China (No.17KJB510036). This work is also supported in part by Engineering Research Center of Digital Forensics, Ministry of Education. (Corresponding authors: Zhengguo Sheng; Chenxi Huang; Alex. X. Liu)

J. Su is with the School of Computer and Software, Nanjing University of Information Science and Technology, Jiangsu 210044, China (e-mail: sj890718@gmail.com).
Z. Sheng is with the Department of Engineering and Design, University of Sussex, Brighton BN1 9RH, U.K. (e-mail: z.sheng@sussex.ac.uk).
Alex. X. Liu is with School of Information Engineering, Huzhou University, Zhejiang 313000, China (e-mail: alexliu360@gmail.com).
Z. Fu is the School of Computer and Software, Nanjing University of Information Science and Technology, Jiangsu 210044, P.R.China (e-mail: wwwfzj@126.com).
C. Huang is the School of Informatics, Xiamen university, Fujian 361003, P.R.China (e-mail: supermonkeyxi@xmu.edu.cn).
Digital Object Identifier xxxx

in a given set. In ID-collection protocols, each tag needs to transmit its ID to the reader, and the length of ID is usually 96 or 128 bits long. Therefore, when the tag cardinality (the number of tags left to read) is large, the total execution time of ID-collision protocols is extremely high because each tag needs to transmit its full ID at least once. Moreover, serious tag collisions significantly may increase the time cost in missing tag identification process. Although many missing tag identification methods have been proposed to reduce the execution time, there is still a large room for performance improvement. Most of the existing protocols focus on increasing the proportion of singleton slots in a frame and reducing the data transmission of tags to improve time efficiency. However, collision slots have not been well explored to further improve time efficiency in missing tag identification. Moreover, the time consumed by reader commands is ignored. Such time consumption is extremely critical, which increases the entire execution time during the missing tag identification process.

1.3 Proposed Approach

In this paper, we make the full use of collided signals to verify missing tags and improve time efficiency. Specifically, we allow more than one tags to simultaneously feed back to the reader with different length w of bit strings based on Manchester coding, so that each collision bit can represent only one tag's presence. Based on the mixed signals from tags, the reader can recover them and verify which tags are present. The following example explains the basic idea of our proposed approach. As shown in Fig. 1, there are four candidate tags waiting to be verified for their presence in an RFID system. The reader uses an initial frame length with $F = 4$ to probe four tags. Tag C is virtually mapped to the first slot (namely singleton slot) and tags A, B and D are virtually mapped to the third slot (namely collision slot). The expected string received at the reader side in the third slot is "xxx", where "x" represents the collided bit. Because the actual string received by the reader is "001", the reader can determine that the tags B and D are missing tags. As observed in Fig. 1, the reader can identify more than one missing tags with only one query in a slot. Thus, the proposed approach can significantly reduce the total execution time over the prior art.

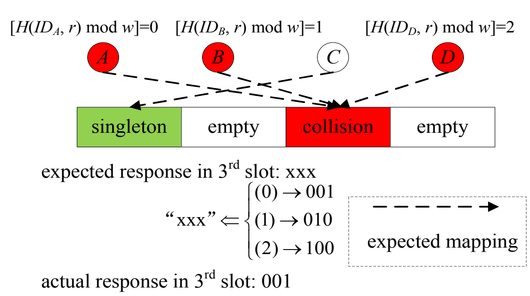

Fig. 1. An illustrative example of the proposed approach

1.4 Technical Challenges and Solutions

The first technical challenge is on allowing multiple tags to reply in a collision slot with different binary strings where each binary string can be used to solely indicate whether a tag presents or not. We address this challenge by mapping a hash value to a binary string containing only one "1", and the string is encoded with Manchester encoding. Thus, the reader can locate the collided bits and identify the corresponding missing tags using bit tracking technology.

The second technical challenge is on maximizing the performance of our proposed protocol. We address this challenge by theoretically analyzing the performance limit of our proposed approach and optimizing the parameter settings to achieve the maximal time efficiency. We adopt customized coding to further reduce the amount of messages transmitted by the reader. Leveraging the binary string design and encoding mechanism, our proposed approach can significantly reduce the total execution time over the existing missing tag identification protocols.

1.5 Summary of Simulation Results

We evaluate the proposed CR-MTI through a series of extensive Monte Carlo simulations. The CR-MTI significantly improves the identification performance over the existing tag searching protocols including ID-collection protocols and missing tag identification protocols [21, 23, 28, 29, 31]. Experimental results are supplemented to indicate our proposed CR-MTI protocol outperforming the best prior missing tag identification protocols in total execution time, average communication complexity and time efficiency by an average of 43.8%, 83.7%, and 97.3%, respectively, when the number of candidate tags varies from 1000 to 20000. The experimental results also show that our proposed CR-MTI protocol reduces the total execution time of the best prior ID-collection protocol [23] by an average of 83.7% when the number of candidate tags is fixed as 10000 and the missing rate of candidate tags varies from 0 to 0.95. In essence, the CR-MTI makes full use of partial collision slots to verify the presence of multiple tags simultaneously in a slot, which can significantly reduce the total execution time and communication overhead in tag identification process. In addition, our proposed CR-MTI uses a customized coding to further reduce the amount of messages transmitted by the reader.

2 Related Works

2.1 ID-collection Protocols

To tackle the tag searching issue in RFID-based applications, many approaches have been proposed to identify the tags in dense RFID systems. We classify the existing approaches into two categories: ID-collection [23-25] protocols and missing tag identification [26-30] protocols. Herein, existing ID-collection protocols can be directly applied to tag searching. Existing ID-collection protocols can be divided into two categories: Aloha-based [15-18] [23-24] and tree-based [12-14][25] protocols. In Aloha-based protocols, the reader requires the tags within its coverage to reply by sending a request command. Both frame size f and a random number seed r are contained in the command. After receiving the command sent by the reader, each tag calculates $H(ID, r) \bmod f$ to determine which time slot it should respond to the reader. $H()$ is a hash function that is uniformly distributed and assembled on each tag. For a given time slot, it has three outcomes, the first is a singleton slot with only one tag response, the second is an empty slot with no tag response, and the third is a collision slot with multiple tag responses. In a singleton slot, the reader can obtain the corresponding ID of tag. After reading a frame, the reader uses slot statistics to optimize the frame size and launch the following ID collection round. The above iterative process does not terminate until all the tags are collected. In tree-based protocols, the reader essentially subdivides the colliding tags into smaller subsets until a single tag response is successfully identified by the reader. Specifically, the reader queries tags with a binary string (called query prefix), and the tags whose IDs match the query prefix will respond. If detecting a collision, the reader will queries tags again after appending the previous query prefix by 0 and 1, respectively. This query-and-append loop continues until all tags are successfully identified. When the reader successfully reads all the tags by using ID collection protocols, it can gather all the IDs and compare them with the IDs pre-stored in the database to find out the missing tags. Obviously, it is very time-consuming to directly use ID collection protocols to address the tag searching issue.

2.2 Probabilistic Detection Protocols

The missing tag identification protocols can be further categorized into two types: probabilistic [19][22][26-27] detection and deterministic [21][28-30] identification protocols. Probabilistic detection protocols aim to discover the missing event (missing tag number exceeds threshold) with a predefined reliability, but does not pay attention to their IDs. The probabilistic detection problem is firstly defined in literature [19], i.e., given a set of N tags, all tag IDs are being known to the reader and stored in the database. Normally, if the stock quantity of items does not change, the set of tag IDs collected by the reader and that in the database should be consistent, otherwise the missing tags are detected. Because f, r, hash function $H()$, and IDs are known in advance, the reader can pre-build an expected bitmap to represent the expected mapping relationship between the tags and the frame. After reading the full frame, the reader can discover the missing tags by comparing the difference between the actual bitmap and expected bitmap. The TRP [19] protocol focuses on optimizing the frame size, and shortens the time taken for the missing tag detection under the condition of satisfying a given detection accuracy. To fully utilize empty slots to implement missing tag detection, the authors in [22] proposed a multi-seed missing tag detection (MSMD) protocol, which allows more single tags to be mapped into the frame through multi-round hash operations, thereby improving the utilization of frame slots. The authors in [27] proposed a bloom filter-based missing tag detection (BMTD) to discover missing tags when there are unexpected tags. In BMTD, the main function of the bloom filter is to inactivate unexpected tags during the missing tag detection process, thereby weakening their interference with detection. However, the tag needs to repeatedly retransmit the data to the reader, resulting in an extended detection time. In summary, the probabilistic detection protocols cannot exactly pinpoint which tags are missing and thus fail to obtain their IDs.

2.3 Deterministic Identification Protocols

Deterministic identification protocols aim to collect all IDs of missing tags. The MAC layer mechanism of deterministic protocols is also based on Aloha-based protocol. The ID-free protocol (IIP) [21] Firstly uses the framed slotted Aloha (FSA) communication mechanism to identify missing tags. Specifically, the IIP protocol introduces a guide vector in the original FSA communication mechanism, so that the tags involved in the collision slot can retransmit with a probability of 0.5. Benefiting from the introduction of guide vector, the frame utilization of IIP can be increased to 0.52 [21]. The authors in literature [28] leverage multi-round hash operations in missing tag identification (MMTI) protocol to further optimize the frame utilization. The MMTI [28] seeks to balance the transmission cost of bitmap and the frame utilization, and derives the optimal rounds of hash operation. The literature [29] presented a slot filter-based missing tag identification (SFMTI) protocol to improve identification efficiency by reconciling expected collision slots with only 2 or 3 tags into singleton slots. In [30], the authors presented a deterministic protocol namely pair-wise collision resolving missing tag identification (PCMTI) to allow a pair of tags to concurrently respond in a same slot. By assigning a specific hash seed, each tag replies with a different value, i.e., 0 or 1, so that the reader can determine whether any tags are missing. However, neither SFMTI nor PCMTI can make full use of k-collisions ($k > 2$). By analyzing the changes in missing rate during the identification process, the authors in [31] proposed a coarse-grained inventory list based stocktaking (CLS) protocol, which utilizes partial collision slots to verify the existence of missing tags. Specifically, the reader attempts to identify more than one missing tags when an expected collision slots turn out to be an empty slot in fact. Moreover, the CLS protocol uses Huffman coding to reduce the transmitted message during the identification process. Most of the previous missing tag identification protocols [19-22][26-29] focus on efficiently improving the frame utilization and collision information will be directly discarded. This undoubtedly wastes a lot of useful information and restricts the further improvement of the efficiency of the missing tag identification protocols.

3 System Model And Problem Description

3.1 System Model

Consider an RFID-based scenario where a large number of items are attached with RFID tags in a warehouse, an RFID reader is equipped in the center of the warehouse to monitor the tags. This scenario is depicted in Fig. 2 where there is a single reader, a back-end server and a number of tags. For clarity, we first consider CR-MTI protocol in the single-reader scenario and then discuss the extension of CR-MTI protocol in multiple readers scenario. Each tag has an unique ID with 96 bits. S_c denotes the set of candidate tags to be monitored. The reader has prior knowledge of all IDs of candidate tags.

3.2 Communication Overview and Model

The RFID system contains two communication data links: downlink and uplink. Herein, the downlink represents communication links from the reader to tags, while the uplink

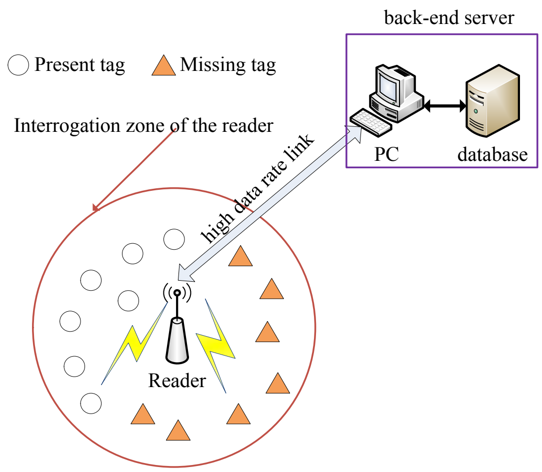

Fig. 2. An identification scenario with missing tags in an RFID system

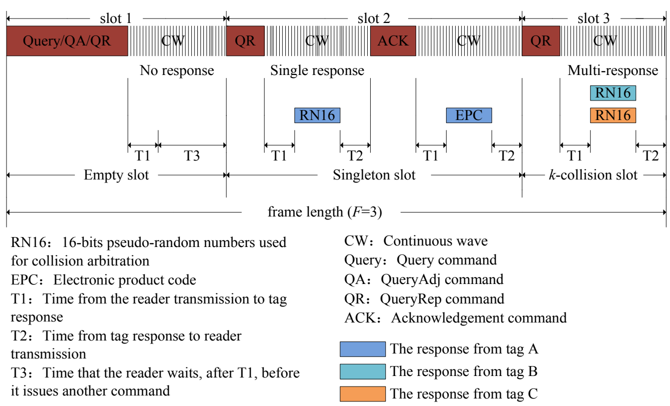

Fig. 3. The link timing of Framed slotted Aloha protocol

represents communication links from the tags to the reader. The link timing specified by EPC C1 Gen2 [32] UHF RFID standard is illustrated in Fig. 3 [1]. The communication links between the reader and tags are assumed reliable and follow the Reader-Talk-First mode [23][29], and the reader adopts framed slotted Aloha (FSA) protocol to interrogate the tags within its coverage. The reader initiates a communication process by constructing a frame and broadcasting the frame size (f) and random number seed (r). Each tag picks a slot to reply by calculating the result of $H\left(ID,\ r\right)$ mod f, where $H()$ is a uniformly distributed hash function and assembled in each tag. Note that $H()$ can be easily implemented in COTS RFID systems based on analog on-tag hash primitives [33].

After a frame is completely executed, the reader can count the status of each slot and classify these status into three categories: empty, singleton, and k-collision. An empty slot means that no tag is responding, singleton slot means that only one tag is responding, and k-collision slot means that k tags are responding. Because expected mapping and actual mapping are involved in the missing tag identification protocol, we can further distinguish them to avoid confusion. We name the empty slots selected by no candidate tags as *expected empty slots*, and the empty slots in the actual identification phase as *actual empty slots*. Similarly, we have *expected singleton slots*, *actual singleton slots*, *expected k-collision slots*, and *actual k-collision slots*. A slot can also be divided into three categories depending on their duration: 1) short-reply slot, i.e., tag only responds with 1-bit message, 2) long-reply slot, tag responds with 10-bit message, 3) tag-reply slot, i.e., tag responds with 96-bit ID information. The duration of a short-reply slot, a long-reply slot and a tag-reply slot are denoted as T_{short}, T_{long} and T_{tag}. Following the previous parameter setting in [11][21][27-29][31], T_{short}, T_{long} and T_{tag} are set to 0.4 ms, 0.8 ms, and 2.4 ms, respectively.

1. It is noted that although the proposed protocol in this paper is not a pure anti-collision protocol, it still satisfies the timing rules except the returned RN16 being replaced by other data.

3.3 Problem Description

Due to theft, administrative errors or some other reasons (e.g., moving out of the reader's interrogation region), some tags may be missed (named missing tags) in a large-scale RFID system. The set of all missing tags can be denoted as S_m. The cardinality of missing tags is M. The problem addressed in this paper is to effectively and completely identifying all missing tags. The execution time is a critical concern to identify missing tags. It is noted that the reader does not know the quantity of missing tags and their IDs in advance. Therefore, the missing tag identification problem can be summarized as follows: Given a candidate tag set S_c whose cardinality N is known by the reader, the reader requires to identify M missing tags with the minimal time cost. The used notations are summarized in Table 1.

TABLE 1
NOTATIONS USED IN THIS PAPER

Symbol	Description
S_c	The set of candidate tags
N	The cardinality of S_c
S_m	The set of missing tags
M	The cardinality of S_m
P_{miss}	The missing rate given by M/N
f	The length of filter vector
r_i	The random number seed that is reset in each round
$H\left(*\right)$	The hash function with a uniform distribution
V_i	The indicator filter vector
w	The length of binary string in each resolvable collision slot
λ	The load factor given by N/f

4 The Proposed CR-MTI Protocol

4.1 Bit tracking technology

Bit tracking technology is commonly based on FM0 [34] or Manchester coding, in which the encoding of each data bit is either low-to-high or high-to-low with equal time interval. In RFID systems, bit tracking technology can be used to locate the positions of collided bits in a vector, so it is widely used in ID-collection protocols and unknown tag identification protocols [14][17][25][35]. In such protocols, a tag transmits data through Manchester coding. As illustrated in Fig. 3, it is feasible to trace individual collided bits by bit tracking technology. In Fig. 4, the transmitted strings of tags A, B and C are "0001", "0010", and "1000", respectively. When tags A, B, and C choose a same slot to respond with strings using Manchester coding method, the mixed signal at the reader side is "x0xx", where "x" is a collided bit. The example shows a collision in 1-st, 3-rd and 4-th bits. By using the bit tracking technology, the reader can easily locate the positions of collided bits and determine that there are at least three tags fall into the slot. Bit-wise synchronization is an important premise to implement our proposed CR-MTI protocol. We consider the bit-wise synchronization in the proposed CR-MTI protocol for the following reasons. The existing literature [36] pointed out that the synchronization offset for commercial RFID tags is normally no more than 1 µs, whereas the transmission duration of 1-bit data is about 18.8 µs . Hence, the 1 µs offset only about 5.3% of a bit duration. In addition, there are many works focusing on further decreasing BLF offset. Moreover, the authors in literature [37] verified the feasibility of bit-wise synchronization in RFID systems by carrying out the practical experiments with USRP and WISP tags. In summary, the BLF offset does not have significant impact on the proposed CR-MTI protocol.

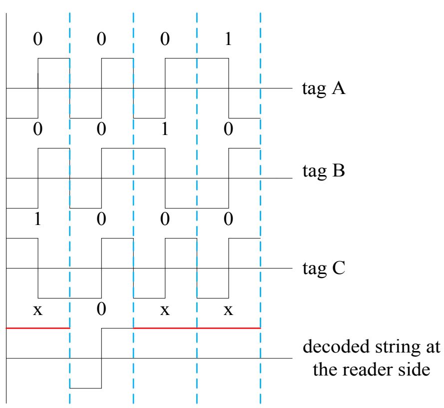

Fig. 4. An illustrated example of bit tracking technology

4.2 Protocol Design

We propose a protocol named Collision Resolving based Missing Tag Identification (CR-MTI) to solve the formulated problem. CR-MTI possesses three novelties: (1) resolvable collision slots, and identifying more than one tags in a resolvable collision slot; (2) skipping the excepted empty slots and irresolvable collision slots; (3) encoding the filter vector with customized coding to further decrease communication complexity, i.e., the number of transmitted bits to be sent to tags. The workflow of our proposed CR-MTI contains multiple rounds, each of them further includes four phases: (1) Main filter vector construction phase, in which the reader builds a vector based on the expected status of all tag IDs; (2) collision slot reconciling phase, in which the reader builds a string of a given length so that different tags can be mapped to different positions in the string and builds a changed vector; (3) presented tag verifying phase, in which the reader broadcasts an indicator filter vector, and the tags that respond in the expected singleton slot with short-reply or respond in the resolvable collision slot with encoded bits to announce their presence, respectively. In each round, the reader can verify the presence of some tags. CR-MTI protocol continues to execute until all tags are successfully verified. In the following sections, we describe the proposed CR-MTI in details.

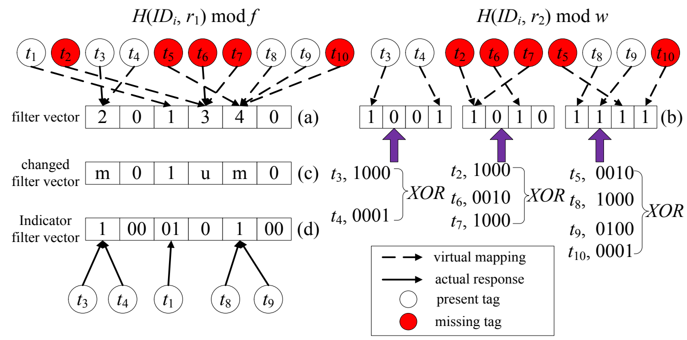

Fig. 5. An identification example of CR-MTI

4.2.1 Main filter vector construction phase

As mentioned above, the reader knows the full knowledge of the IDs of candidate tags. In main filter vector construction phase, the reader constructs a bitmap (named filter vector) using a randomly generated seed r_1 and a frame size f. Specifically, it first uses a uniform hashing function $H(*)$ to pre-compute the expected time slots according to r_1, f, and N. Based on the system model described in section 3, the time slot index selected by i-th tag is calculated by $H(ID_i,\ r_1)$ mod f. Then, the reader builds the main filter vector V_m with f bits. The i-th element $V_m(i)$ in the vector V_m means the number of tags contained in the i-th slot where $V_m(i) \in [0,\ N]$. For example, the value "0" that corresponds to an empty slot, the value "1" that corresponds to a singleton slot, the value k ($k > 1$) that corresponds to a k-collision slot with k tags. The reader uses V to store the detailed values of the main filter vector V_m where $V(i)$ contains the information of all tags in i-th slot. Fig. 5 (a) depicts the expected mapping between candidate tags and time slots and the construction of the corresponding main filter vector.

4.2.2 Collision slot reconciling phase

In collision slot reconciling phase, the reader generates a new random number seed r_2 and sets a string W of length w ($k \leq w$) for individual k-collision slots. Recall that k represents the number of tags involved in a collision slot. For each k-collision slot, the reader takes advantage of r_2 to calculate $H(ID,\ r_2)$ mod w for each tag involved in the slot, the hashing result R is from 0 to $w-1$. According to value of R, the reader can set the $R+1$-th bit of the string W to 1, and the rest to 0. In other words, the tag can be mapped to a bit (this bit is set to 1) of the string W. If the R value of each tag is unique between 0 and $w-1$, the k-collision slot is regarded as a resolvable collision slot. Otherwise, the slot is regarded as an irresolvable collision slot. The above collision reconciling process is depicted in Fig. 5 (b). For example, t_3 and t_4 are mapped to a 2-collision slot in Fig. 5

(a). By performing $H\left(ID,\ r_2\right)$ mod $w = 4$, t_3 with $R = 3$ is hashed to the 4-th bit of W, t_4 with $R = 0$ is hashed to the 1-st bit of W. Because they are hashed to different bits of W, this 2-collision slot is a resolvable collision slot. Another example: tags t_2 and t_7 involved in a 3-collision slot are hashed to the same bit of W by calculating $H\left(ID,\ r_2\right)$ mod 4, this 3-collision slot is irresolvable. Different from the literature [29], this paper takes the reconcilement of k ($k \geq 4$)-collision slots into account. And the reader does not need to relocate the collided tags using extra appending vector.

In the collision slot reconciling phase, the reader updates the values of elements in the vector V_m, where an element corresponding to a resolvable collision slot is set to m, and an element corresponding to an irresolvable collision slot is set to u. For example, Fig. 5 (c) depicts an updated filter vector of V_m corresponding to that in Fig. 5 (a). In the following, the reader constructs a synthetic indicator filter vector V_i by using customized coding. In the V_i, m, u, data-0, and data-1 are coded as "1", "0", "00" and "01", respectively. The reason why the reader adopts customized coding is that the coding can be easily decoded at the tag side. In addition, when the number of missing tags is large, k-collision slot accounts for a large proportion in the entire frame. Thus, the reader uses a 1-bit length to encode the resolvable and irresolvable collision slots and a 2-bit length to encode other types of slots. The advantage of such encoding is that it can reduce the communication complexity in missing tag identification process. An example of indicator filter vector is illustrated in Fig. 5 (d).

In the proposed CR-MTI protocol, the irresolvable collision slot will be directly discarded. The reader cannot identify any missing tags in the irresolvable slot. Recently, there are many works proposed to directly decode the collided RFID signals. They focus on how to recover collided tags signals based on specialized instruments like USRP. For example, Bigroup and FlipTracer [38-39] are proposed to achieve highly reliable parallel decoding by using observed transition probabilities between signals' combined states. To further improve the efficiency of missing tag identification, the above mentioned parallel decoding techniques can be embedded in CR-MTI. However, the parallel coding technique also introduces new challenges, such as high hardware costs, high SNR requirements.

4.2.3 Present tag verifying phase

In present tag verifying phase, the reader broadcasts a query command with parameters including r_1, r_2, f, w, and the indicator filter vector V_i to the tags within its vicinity. After receiving the command from the reader, each presented tag t_i obtains an index by calculating $s = H\left(ID_{t_i},\ r_1\right)$ mod f. It is noted that similar to the previous work, the expected empty slots that correspond to "00" in the filter vector are directly filtered out (i.e., the expected empty slots are skipped in verifying phase). Therefore, the tag t_i decodes V_i and determines its slot index that it should respond by counting the number of non-empty slots preceding s in V_i. It can be seen from Fig. 4 (d) that tag t_1 will respond to the reader in the 2-nd slot, because there are a k-collision slot and an empty slot before its index, and the empty slot is filtered out. And tags t_8 and t_9 will respond in the 4-th slot because there are two k-collision slots, a singleton slot and an empty slot preceding their indexes. It is worth noting that in our proposed CR-MTI protocol, the response behavior of the tag is determined by the slot type. Specifically, for a given resolvable collision slot, a tag t_i will respond to the reader with a w-bit string which contains only one bit data-1. According to our communication model, the time duration for w-bit message transmission is t_w. For a given irresolvable collision slot, a tag t_i will respond to the reader with a 1-bit message whose time duration is T_{short}.

Assuming that a tag t_i allocated in the i-th (i.e., $i = H\left(ID_i,\ r_1\right)$ mod f) slot of a frame, it will perform different operations according the values in V_i.

1) If $V_i(i)$="01" or $V_i(i)$="1", which means that the tag t_i falls into a singleton slot or resolvable collision slot, it will switch its state to silent and not respond to the reader in the following identification process.

2) If $V_i(i)$="0", which means that the tag t_i falls into an irresolvable collision slot, it will not change its state and continue to respond to the reader in the following identification process.

Because the parameters and hashing function are known to both the reader and tags, the reader can predict the actions of the tags in advance. By comparing the difference between the expected response and the actual response, the reader is able to discover and identify the missing tags. Let k denote the number of response in the i-th slot, we can derive the conditions as follows.

1) If $V_m(i) = 1$ and $k = 1$, the tag allocated in the slot is regarded as present.

2) If $V_m(i) = 1$ and $k = 0$, the tag allocated in the slot is regarded as missing.

3) If $V_m(i) = m$, $w - k$ tags allocated in the slot are identified as missing.

It is noted that all the tags identified as present or missing will be removed from the candidate tag set N by the reader. The above missing tag identification process is repeated until N is null which means all candidate tags are verified as missing or present.

4.3 Parameter optimization

Obviously, the execution time of our proposed CR-MTI protocol is determined by the length of f and w. Therefore, we need to determine its optimal value to maximize the time efficiency. As described above, N denotes the cardinality of the candidate tag set.

Considering that N candidate tags are randomly distributed in f slots, the probability that k tags choose a same slot can be expressed as

$$P_k = \binom{N}{k} \left(\frac{1}{f}\right)^k \left(1 - \frac{1}{f}\right)^{N-k} \quad (1)$$

Accordingly, $P_{k=1}$, $P_{k=0}$, and $P_{k>1}$ are the corresponding probabilities that a slot is empty, singleton, and collision, respectively, which are given as

$$P_{k=0} = \left(1 - \frac{1}{f}\right)^N \approx e^{-\frac{N}{f}} \quad (2)$$

$$P_{k=1} = \left(\begin{array}{c} N \\ 1 \end{array} \right) \left(\frac{1}{f}\right) \left(1-\frac{1}{f}\right)^{N-1} \approx \frac{N}{f} \left(\frac{f}{f-1}\right) e^{-\frac{N}{f}} \quad (3)$$

$$P_{k>1} = \sum_{k=2}^{N} \left(\begin{array}{c} N \\ k \end{array} \right) \left(\frac{1}{f}\right)^{k} \left(1-\frac{1}{f}\right)^{N-k} \quad (4)$$

According to (2)-(4), the number of expected empty slots, expected singleton slots and expected k-collision slots are calculated as

$$n_{k=0} = f \times P_{k=0} \approx f \times e^{-N/f} \quad (5)$$

$$n_{k=1} = f \times P_{k=1} \approx N \times \left(\frac{f}{f-1}\right) e^{-\frac{N}{f}} \quad (6)$$

$$n_{k>1} = f \times P_{k>1} \quad (7)$$

In CR-MTI, there are two cases to verify whether a tag is present or not. First, by detecting the actual response in an expected singleton slot, it is determined whether a tag is present or missing. Second, $w-k$ tags allocated in a resolvable k-collision slot can be identified as missing. Let P_s denote the probability that a tag can be successfully identified, which is expressed as

$$P_s = P_{k=1} + P_{k>1}^{s} \quad (8)$$

where $P_{k>1}^{s}$ indicates the probability that k tags allocated in a same slot with different strings (i.e., the probability that a k-collision slot is resolvable). To calculate $P_{k>1}^{s}$, we first calculate the probability $P_{k/w}$ that k tags choose k different values from 0 to $w-1$. $P_{k/w}$ can be written as

$$P_{k/w} = \left\{ \begin{array}{ll} \frac{w!}{w^k (w-k)!}, & if\ w \geq k \\ 0, & others \end{array} \right. \quad (9)$$

According to (4) and (9), $P_{k>1}^{s}$ is calculated as

$$\begin{array}{l} P_{k>1}^{s} = P_{k>1} \times P_{k/w} \\ \approx \sum\limits_{k=2}^{N} \frac{1}{k!} \times \left(\frac{N}{f}\right)^{k} \times e^{-\frac{N}{f}} \times \frac{w(w-1)\ldots(w-k+1)}{w^k} \end{array} \quad (10)$$

Then, the number of such slots can be written as

$$n_{k>1}^{s} = f \times P_{k>1}^{s} \quad (11)$$

Let n_u denote the number of slots that can be used to verify the presence of the tags in each round, which is expressed as

$$\begin{array}{l} n_u = n_{k=1} + n_{k>1}^{s} \\ = f\lambda e^{-\lambda} + f \sum\limits_{k=2}^{N} \frac{1}{k!} \lambda^k e^{-\lambda} \frac{w(w-1)\ldots(w-k+1)}{w^k} \end{array} \quad (12)$$

where $\lambda = \frac{N}{f}$. Because k tags can be identified in a resolvable k-collision slot, the total number of identified tags in each round can be expressed as

$$\mathcal{N}_{total} = n_{k=1} + k \times n_{k>1}^{s} \quad (13)$$

Let T_{round} denote the execution time in each identification round. T_{round} is composed of two parts: one is the time required for the reader to send a query command to the tags (denoted as T_Q), and the other is the time required for the tags to respond to the reader (denoted as T_R). Therefore, T_{round} can be calculated as

$$T_{round} = T_Q + T_R \quad (14)$$

Our proposed CR-MTI protocol uses customized coding to encode the types of expected slots, in which 2-bit is used to encode expected empty slots and expected singleton slots, and 1-bit is used to encode k-collision slots including both resolvable and irresolvable k-collision slots. Therefore, T_Q can be expressed as

$$\begin{array}{l} T_Q = \frac{t_{tag}}{96} \times [2 \times (n_{k=0} + n_{k=1}) + n_{k>1}] \\ \quad + t_{long-cmd} + (f-1)\, t_{short-cmd} \end{array} \quad (15)$$

in which $t_{long-cmd}$ and $t_{short-cmd}$ means the time duration taken by the reader to launch the first query command and the rest $(f-1)$ query commands during the frame, respectively. Referring to the existing UHF RFID standard, $T_{long-cmd}$ and $T_{short-cmd}$ can be set as 0.55 and 0.1 ms. T_R is expressed as

$$T_R = n_{k=1} \times t_{short} + n_{k>1}^{s} \times t_W \quad (16)$$

where t_W denote the time required for the tags to respond the string W to the reader in a resolvable k-collision slot. The time efficiency η is defined as

$$\begin{array}{l} \eta = \frac{\mathcal{N}_{total} \times t_{short}}{T_{round}} = \frac{n_{k=1} + k \times n_{k>1}^{s}}{T_Q + T_R} \times t_{short} \\ \approx \frac{\left[\lambda e^{-\lambda} + \sum\limits_{k=2}^{N} \frac{1}{(k-1)!} \lambda^k e^{-\lambda} P_{k/w}\right] \times t_{short}}{(1+e^{-\lambda}+\lambda e^{-\lambda}) \frac{t_{tag}}{96} + \lambda e^{-\lambda} \times t_{short} + t_w \times \sum\limits_{k=2}^{N} \frac{1}{(k)!} \lambda^k e^{-\lambda} P_{k/w} + \alpha} \end{array} \quad (17)$$

herein α is expressed as

$$\begin{array}{l} \alpha = \frac{t_{long-cmd} + (f-1) t_{short-cmd}}{f} \\ \approx \frac{0.55 + 0.1 \times (f-1)}{f} \end{array} \quad (18)$$

For CR-MTI protocol, the values of f and w should be appropriately set. The reasons are as follows. According to the principle of our proposed CR-MTI protocol, it is intuitive to know that if f is set too large, the number of expected k-collision slots will become small. In this case, if the value of w is too large, it will cause a lot of redundant bits and thus increase the communication complexity. If f is set too small, then there will be a large number of tags falling in the same slot. In this case, a small w will lead to the decrease of the number of resolvable k-collision slots. According to the previous parameter setting, i.e., t_{short}=0.4 ms, t_{tag}=2.4 ms, we know that the date rate between the reader and tags is about 40 kbps [28-31]. Therefore, t_w can be calculated as

$$\begin{array}{l} t_w = t_{short} + (w-1) \times 0.025 \\ \quad = 0.4 + (w-1) \times 0.025 \end{array} \quad (19)$$

Although we cannot directly derive a closed-form solution of λ that satisfies the maximal η due to the existence of w, it can be easily solved through numerical calculation method. Intuitively, when the value of λ is large, the number of k-collision slots will be large. In this case, setting w to a large value helps to better resolve the k-collision slots, thereby reducing the total execution time. When the value of λ is small, the number of k-collision slots will be small. In

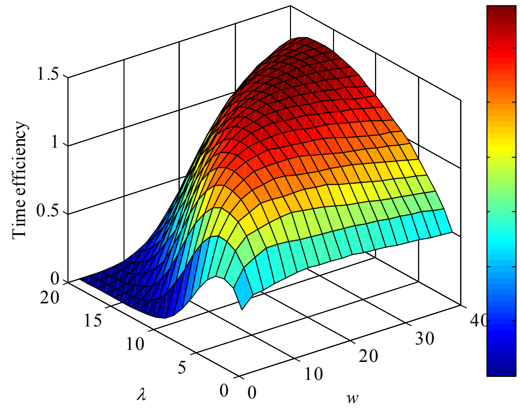

Fig. 6. Time efficiency of CR-MTI with different λ and w

this case, setting w to a small value helps to reduce the communication complexity in collision slot reconciling phase, thereby reducing the total execution time and improving the time efficiency. By using the numerical method, we can search the optimal frame size f_{opt} and string size w_{opt} offline according to the given candidate tag set. We conduct a series of experiments under the following cases: the number of candidate tags is set as 1000. λ varies from 1 to 20 in steps of 1, and w is ranging from 2 to 40 in steps of 2. We can observe from Fig. 6, the optimal time efficiency depends on the combination of λ and w. In Fig. 6, the time efficiency of our proposed CR-MTI peaks when $\lambda = 15$, $w = 34$ and the number of candidate tags is 1000. Although the optimal combination of λ and w may vary with the number of tags in a large scale, it is not necessary to maintain the optimality in a real-time way due to the costly overhead. In the simulation section, we will discuss the impact of these two parameters on performance of our proposed CR-MTI.

4.4 Discussion: Multiple Readers And Channel Errors

4.4.1 Multiple readers

Limited by the size of the transmitted power and the communication range of tags, a single reader cannot cover the entire area of a large retail or warehouse center. Thus, such applications with a huge number of tags requires multiple readers with overlapping regions. The use of multiple readers in a large-scale RFID system introduces several new technical challenges such as reader-to-reader collisions. Because many prior reader-scheduling protocols [40-41] have been presented to determine the running order of each reader, such collisions in the overlapping region can be well handled. In our proposed CR-MTI protocol, we require all the readers to equip the same parameters including f, r, w and V_i. After receiving the responses from the tags, the readers send them to the back-end server for data fusion. Therefore, by using the well-known Bloom filter technologies [28-29], our proposed CR-MTI protocol can be easily extended to multi-readers scenarios.

4.4.2 Channel errors

In the above discussions, we assume the communication channel between the reader and tags is error-free. However, because some physical factors such as path loss and multipath effect occur in practical application scenarios, the communication channel is error-prone [21, 28, 42], which will increase the bit error rate, thereby degrading the performance of missing tag identification protocol. Therefore, we need to discuss the performance of our proposed CR-MTI protocol under unreliable channels.

Generally, channel errors can be categorized into three types: 1) channel errors in the process of transmitting query parameters including f, r_i, w, and others; 2) channel errors in the process of transmitting the indicator vector V_i; 3) channel errors in the process of tag responding data to the reader. For the former two types of channel errors, the reader can append a 10-bit checksum to the end of the query parameters or V_i. Therefore, the tags can verify the correctness of checksum by checking whether it matches the received data. If not, the reader will be required to retransmit the query parameters or V_i. For the latter type of channel error, false positive and false negative need to be considered. The false positive means that a candidate tag in an expected singleton slot is wrongly identified as a missing tag due to channel errors such as signal distortion. The false negative means that an actually missing tag is wrongly identified as a presented tag due to channel errors such as noise interference. A polling based method can be used to cope with the false positive error (i.e., a presented tag is identified as a missing one). The reader polls the potential missing tags by their IDs one by one, then the fake "missing" ones will acknowledge and can be filtered out. The reader can launch multiple rounds of missing tag identification protocol to cope with the false negative error (i.e., an actually missing tag is identified as a presented tag). Because the random number seed and communication session used by the reader in each tag identification round are different, even if a missing tag is not recognized in a round, it will be more likely to be recognized in the following rounds. Meanwhile, the reader allows the presented tag to report the ID, then the true "missing" ones can be filtered out.

5 Experimental Results

5.1 Simulation Setup

In this section, we evaluate the performance of our proposed CR-MTI protocol through a series of Monte Carlo experiments in MATLAB on an Alienware-15 laptop with an Intel 2.5GHz CPU. To provide a fair performance evaluation, we follow the same parameter setting as in [28-29, 31] and compare our proposed CR-MTI with the prior art including IIP [21], MMTI [28], SFMTI [29], CLS [31], and TES-FAS [23]. We setup a high dense RFID system with a single reader and a variety of tags. As same as in literatures [21-25][28-29][31], the communication channels between the reader and tags are assumed as no error-prone. In our simulations, to reduce the randomness and ensure the convergence, the simulation results are averaged over 1000 iterations.

5.2 Comparison of Results in Various Metrics

5.2.1 Impact of number of candidate tags

Referring to the existing literatures [11][21][28-31], the execution time is one of the most important performance metrics to evaluate the missing tag identification protocols,

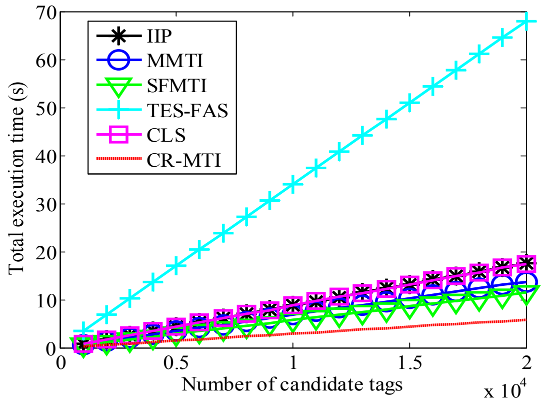

Fig. 7. Comparison of various methods in total execution time

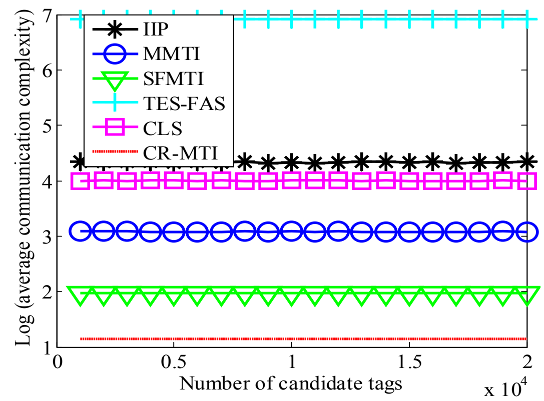

Fig. 8. Comparison of various methods in average communication complexity

which is defined as the time duration from the reader sends a query command to all tags to be verified. We vary the number of candidate tags from 1000 to 20000 in steps of 1000. *Our proposed CR-MTI is superior to other protocols and reduces the total execution time of SFMTI (2nd best) by average of 49.3%*. Fig. 7 compares the total execution time of all comparative approaches when the number of tags is from 1000 to 20000. Observed in Fig. 5, the total execution time of all methods increases with the number of candidate tags because more time slots are needed by the reader to verify the presence of the candidate tags. The experimental results also show that our proposed CR-MTI protocol outperforms TES-FAS and IIP. For example, when the number of candidate tags $N = 10000$, the total execution time of TES-FAS and IIP are 33.98 and 8.792 seconds, respectively. However, CR-MTI only takes 2.905 seconds, which shortens the time by 91.5% and 66.9% when compared with TES-FAS and IIP. When compared with the state-of-the-art SFMTI, our proposed CR-MTI can still reduce the execution time by an average of 49.3%. For example, when $N = 10000$, the total execution time of CR-MTI is 2.95 seconds and that of STMTI is 5.73 seconds. Moreover, it is noted that the total execution time required by CR-MTI is also less than the lower bound of the minimal execution time, which is defined as $N \times T_{short}$ in literature [29]. The reason is that in the existing missing tag identification protocols, the reader can only verify the presence of one tag in a slot, and in CR-MIT, the reader can make full use of partial collision slots to verify the presence of multiple tags simultaneously in a slot, which can significantly reduce the total execution time.

The average communication complexity (i.e., the average number of transmitted messages between the reader and tags) is another important performance metric which can explicitly reflect the energy consumption in missing tag identification process. *Our proposed CR-MTI protocol reduces the average amount of messages per tag verification of the best existing solution by an average of 43.8%*. Fig. 8 depicts the average communication complexity of all reference methods. As can be observed, the average communication complexity of all protocols is independent of the number of candidate tags. In other words, the average communication complexity is only related to the data rate, the length of tag ID and command. Because the TES-FAS protocol requires the tag to return the complete ID when verifying the presence of the tags, the required communication complexity is the highest. As a contrary, other missing tag identification protocols use singleton slots to verify the presence of the tags, which greatly reduces the amount of transmitted messages because the complete ID is not required to be transmitted. Compared with other reference protocols, the reasons that CR-MTI can reduce the communication complexity are two-fold. The first is that CR-MTI uses customized coding to reduce the amount of messages transmitted by the reader. The second is that by using a resolvable collision slot to verify more than one tags, CR-MTI can significantly reduce the number of commands sent by the reader and the amount of response messages sent by the tags.

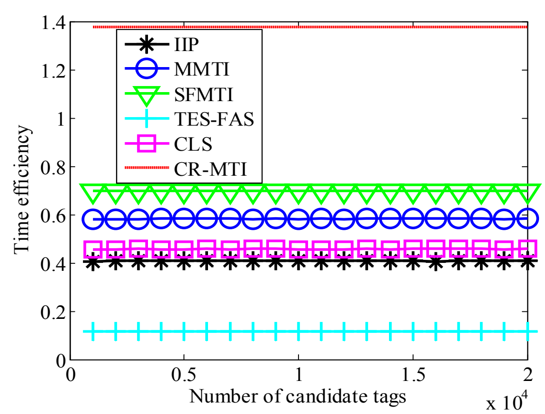

Fig. 9. Comparison of various methods in time efficiency

Our proposed CR-MTI protocol improves the time efficiency of the prior art in ID-collection protocol by an average of $10.6\times$. Fig. 9 demonstrates the time efficiency of all comparative protocols. According to the definition of time efficiency, it is highly related to the total execution time. Thus, a similar performance ranking can be observed in Fig. 9. Specifically, our proposed CR-MTI protocol improves the time efficiency by 203%, 102%, 62.9%, and 167% over IIP, MMTI, SFMTI, and CLS, respectively. The results further show evidence that CR-MTI is significantly better than the other protocols in terms of time efficiency.

5.2.2 *Impact of missing rate*

In this section, we fix $N = 10000$ and vary the missing rate to further evaluate our proposed CR-MTI protocol. The missing rate varies from 0 to 0.95 in steps of 0.05. *Our proposed CR-MTI protocol reduces the total execution time of TES-FAS by an average of 83.7%.* Fig. 10 compares the total execution time of all comparative protocols when the missing rate is vary from 0 to 0.95. As can be observed in Fig. 10, IIP, MMTI and SFMTI hold a stable performance in terms of total execution time. The reason is that all the above protocols identify one or more tags in expected singleton slot, hence their execution time is only affected by the number of candidate tags. On the contrary, the total execution time of TES-FAS and CLS gradually decreases as the missing rate increases. The reason is that as the missing rate continues to increase, more expected collision slots will turns out to be empty slots, thereby reducing the total number of slots and the amount of transmitted messages. It can still be observed that the performance of CLS is worse than SFMTI and CR-MTI in most of time. The reason is that when the missing rate is low, the probability that an expected k-collision slot turn out to be an empty slot is very low, and the detection of a large number of k-collision slots will increase the total execution time. As the missing rate increases, the total execution time of CLS quickly decreases and is gradually less than CR-MTI. However, CLS protocol can only maintain a good performance when the missing rate is very high. To improve the missing tag identification performance in scenarios that the missing rate of candidate tags is relatively low, the improved version of CLS named DLS is proposed in [31].

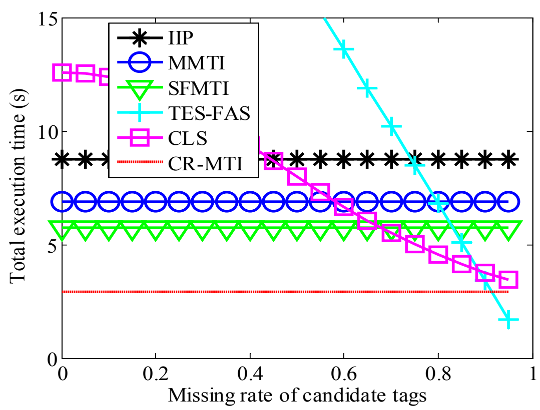

Fig. 10. Comparison of various methods in total execution time when the missing rate P_{miss} varies from 0 to 0.95

5.2.3 *Impact of load factor λ and w*

In this section, we validate the optimal combination of λ and w, which are critical parameters in our proposed CR-MTI. In the simulations, we vary the parameter λ from 1 to 20, and record the corresponding time efficiency when the w are set as 26, 28, 30, 32 and 34, respectively. As can be observed in Fig. 11, to achieve the best time efficiency, different values of w lead to different optimum values of λ. The optimum values of λ is from 12 to 16. We further evaluate the time efficiency of CR-MTI under specific pairs of λ and w when the number of candidate tags is from

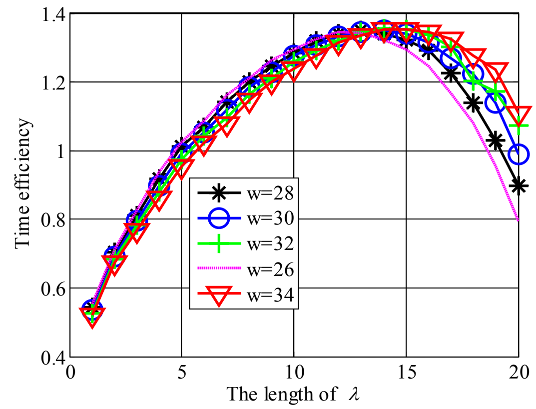

Fig. 11. The time efficiency with respect to λ when N is fixed to 1000

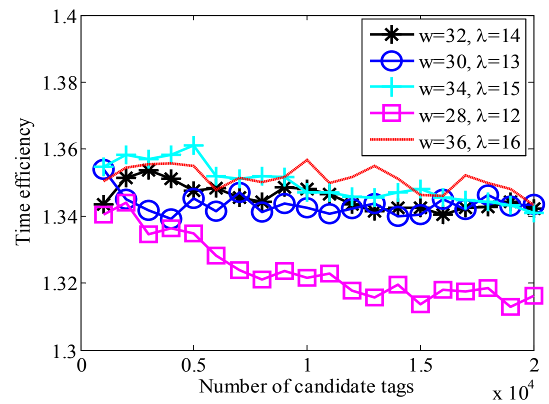

Fig. 12. The time efficiency of CR-MTI under various pairs of λ and w

1000 to 20000 in Fig. 12. The average time efficiency of three curves from highest to lowest are 1.351, 1.349, 1.345, 1.343 and 1.324, respectively. The corresponding combination are ($\lambda = 16$, $w = 36$), ($\lambda = 15$, $w = 34$), ($\lambda = 14$, $w = 32$), ($\lambda = 13$, $w = 30$) and ($\lambda = 12$, $w = 28$), respectively. We can find that no constant parameter setting can always maintain the best performance when the number of candidate tags is from 1000 to 20000. In practical deployment of RFID systems, it incurs high computational complexity to search the optimum values of λ and w and update them in a real-time manner. The experimental results in Fig. 12 indicate that a default parameter setting is preferable during the whole identification process with recommendation of $\lambda = 15$ and $w = 36$.

6 Conclusion

In this paper, we have proposed a collision resolving based missing tag identification (CR-MTI) protocol to identify the missing tags in a time-efficient way. CR-MTI maps multiple tags to different bits of a binary string and thus verify them together in a resolvable k-collision slot. Most of collision slots can contribute to the reduction of the total execution time in the missing tag identification process. A customized coding is used to reduce the amount of messages transmitted by the reader. We have further optimized the parameters of our proposed CR-MTI to maximize the performance. Extensive experimental results have shown that our proposed protocol significantly outperforms all prior missing tag identification protocols under various evaluation metrics, i.e., total execution time, communication complexity and time efficiency.

References

[1] X. Xie, X. Liu, X. Zhao, W. Xue, B. Xiao, H. Qi, K. Li, and J. Wu, "Implementation of differential tag sampling for COTS RFID systems," *IEEE Trans. Mobile Comput.*, vol. 19, no. 8, pp. 1848-1861, 2019.

[2] F. Viani, M. Salucci, F. Robol, G. Oliveri, and A. Massa, "Design of a UHF RFID/GPS fractal antenna for logistics management," *J. Electromagn. Waves Appl.*, vol. 26, no. 4, pp. 480-492, 2012.

[3] C. He, Z. J. Wang, and C. Miao, "Query diversity schemes for backscatter RFID communications with single-antenna tags," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 6932-6941, 2017.

[4] G. Wang, J. Han, C. Qian, W. Xi, H. Ding, Z. Jiang, and J. Zhao, "Verifiable smart packaging with passive RFID," *IEEE Trans. Mobile Comput.*, vol. 18, no. 5, pp. 1217-1230, 2019.

[5] X. Liu, J. Cao, Y. Yang, W. Ou, X. Zhao, K. Li, and D. Yao, "Fast RFID sensory data collection: tradeoff between computation and communication costs," *IEEE/ACM Trans. Netw.*, vol. 27, no. 3, pp. 1179-1190, 2019.

[6] V. D. Hunt, A. Puglia, and M. Puglia, *RFID: a guide to radio frequency identification*. Hoboken, NJ, USA: Wiley, 2007.

[7] P. Yang, W. Wu, M. Moniri, and C. C. Chibelushi, "Efficient object localization using sparsely distributed passive RFID tags," *IEEE Trans. Ind. Electron.*, vol. 60, no. 12, pp. 5914-5924, 2013.

[8] S. S. Saab and Z. S. Nakad, "A standalone RFID indoor positioning system using passive tags," *IEEE Trans. Ind. Electron.*, vol. 58, no. 5, pp. 1961-1970, 2011.

[9] D. M. Dobkin, *The RF in RFID: UHF RFID in practice, 2nd ed*. Newton, MA, USA: Newnes, 2012.

[10] X. Liu, B. Xiao, S. Zhang, and K. Bu, "One more hash is enough: efficient tag stocktaking in highly dynamic RFID systems," in *IProc. IEEE Int. Conf. Commun.*, 2016, pp. 1-6.

[11] X. Liu, X. Xie, K. Wang, H. Qi, J. Cao, S. Guo, and K. Li, "Pinpointing anomaly RFID tags: situation and opportunities," *IEEE Netw.*, vol. 31, no. 6, pp. 40-47, 2017.

[12] J. Myung, W. Lee, J. Srivastava, and T. K. Shih, "Tag-splitting: Adaptive collision arbitration protocols for RFID tag identification," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 6, pp. 763-775, 2007.

[13] H. Guo, C. He, N. Wang, and M. Bolic, "PSR: a novel highefficiency and easy-to-implement parallel algorithm for anticollision in RFID systems," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 1134-1145, 2016.

[14] M. Shahzad and A. X. Liu, "Probabilistic optimal tree hopping for RFID identification," *IEEE Trans. Mobile Comput.*, vol. 23, no. 3, pp. 706-809, 2015.

[15] T. F L. Porta, G. Maselli, and C. Petrioli, "Anti-collision protocols for single-reader RFID systems: temporal analysis and optimization," *IEEE Trans. Mobile Comput.*, vol. 10, no. 2, pp. 267-279, 2010.

[16] W.-T. Chen, "An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length Aloha," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 9-15, 2009.

[17] W. Zhu, M. Li, J. Cao, Z. He, and R. Xie, "Multiple resolution bit tracking protocol for continuous RFID tag identification," in *Proc. IEEE Int. Conf. Mobile Ad Hoc and Sensor (MASS)*, 2019, pp. 100-108.

[18] X. Xie, X. Liu, H. Qi, and K. Li, "Fast identification of multi-tagged objects for large-scale RFID systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 4, pp. 992-995, 2019.

[19] C. C. Tan, B. Sheng, and Q. Li, "How to monitor for missing RFID tags," in *Proc. Int. Conf. Dist. Comput. Syst.(ICDC 2008)*, 2008, pp. 295-302.

[20] Y. Zheng and M. Li, "Fast tag searching protocol for large-scale RFID systems," *IEEE/ACM Trans. Netw.*, vol. 21, no. 3, pp. 924-934, 2013.

[21] T. Li, S. Chen, and Y. Ling, "Identifying the missing tags in a large RFID system," in *Proc. ACM Int. Symp. Mobile ad hoc Netw. Compt. (Mobihoc 2010)*, 2010, pp. 1-10.

[22] 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 Trans. Netw.*, vol. 22, no. 4, pp. 1079-1091, 2013.

[23] J. Su, Z. Sheng, A. X. Liu, Z. Fu, and Y. Chen, "A time and energy saving-based frame adjustment strategy (TES-FAS) tag identification algorithm for UHF RFID systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 2974-2986, 2020.

[24] S. Lee, S. Joo, and C. Lee, "An enhanced dynamic framed slotted ALOHA algorithm for RFID tag identification," in *Proc. Int. Conf. Mobile Ubiquitous Syst.: Comput. Netw. Serv.*, 2005, pp. 166-172.

[25] L. Zhang, W. Xiang, X. Tang, Q. Li, and Q. Yan, "A time- and energy-aware collision tree protocol for efficient large-scale RFID tag identification," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2406-2417, 2018.

[26] W. Luo, S. Chen, T. Li, and S. Chen, "Efficient missing tag detection in RFID systems," in *Proc. IEEE Int'l Conf. Computer Commun. (INFOCOM)*, 2011, pp. 356-360.

[27] J. Yu, L. Chen, R. Zhang, and K. Wang, "Finding needles in a haystack: missing tag detection in large RFID systems," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2036-2047, 2017.

[28] X. Liu, K. Li, G. Min, Y. Shen, and A. X. Liu, "A multiple hashing approach to complete identification of missing RFID tags," *IEEE Trans. Commun.*, vol. 62, no. 3, pp. 1046-1057, 2014.

[29] 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 Trans. Comput.*, vol. 64, no. 1, pp. 87-96, 2015.

[30] L. Zhang,W. Xiang, I. Atkinson, and X. Tang, "A time-efficient pairwise collision-resolving protocol for missing tag identification," *IEEE Trans. Commun.*, vol. 65, no. 12, pp. 5348-5361, 2017.

[31] W. Zhu, X. Meng, X. Peng, J. Cao, and M. Raynal, "Collisions are preferred: RFID-based stocktaking with a high missing rate," *IEEE Trans. Mobile Comput.*, vol. 19, no. 7, pp. 1544-1554, 2020.

[32] EPC Radio-Frequency Identity Protocols generation-2 UHF RFID Protocol for Communications at 860 MHZ-960 MHZ, EPCglobal, London, U.K., 2013.

[33] L. Yang, Q. Lin, C. Duan, and Z. An, "Analog on-tag hashing: towards selective reading as has primitives in Gen2 RFID systems," in *Proc. IEEE Int'l Conf. Computer Commun. (INFOCOM)*, 2017, pp. 301-314.

[34] Y. Kim and A. J. H. Vinck, "Anticollision algorithms for FM0 code and miller subcarrier sequence in RFID applications," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5186-5173, 2018.

[35] X. Liu, K. Li, G. Min, K. Lin, B. Xiao, Y. Shen, and W. Qu, "Efficient unknown tag identification protocols in large-scale RFID systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 12, pp. 3145-3155, 2014.

[36] J. Wang, H. Hassanieh, D. Katabi, and P. Indyk, "Efficient and reliable low-power backscatter network," in *Proc. ACM SIGCOMM*, 2012, pp. 61-72.

[37] X. Liu, X. Xie, S. Wang, J. Liu, D. Yao, J. Cao, and K. Li, "Efficient range queries for large-scale sensor-augmented RFID systems," *IEEE/ACM Trans. Netw.*, vol. 27, no. 5, pp. 1873-1886, 2019.

[38] J. Ou, M. Li, and Y. Zheng, "Come and be served: Parallel decoding for COTS RFID tags," in *Proc. MOBICOM*, 2015, pp. 500-511.

[39] M. Jin, Y. He, Y. Zheng, D. Fang, X. Chen, "FlipTracer: Practical parallel decoding for backscatter communication," *IEEE/ACM Trans. Netw.*, vol. 27, no. 1, pp. 330-343, 2019.

[40] L. Yang, Y. Qi, J. Han, C. Wang, and Y. Liu, "Shelving interference and joint identification in large-scale RFID systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 11, pp. 3149-3159, 2015.

[41] M. Ma, P. Wang, C. Chu, "Redundant reader elimination in large-scale distributed RFID networks," *IEEE Internet of Things J.*, vol. 5, no. 2, pp. 884-894, 2018.

[42] C. He, S. Chen, H. Luan, X. Chen, and Z. J. Wang, "Monostatic MIMO backscatter communications," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1896-1909, 2020.

Jian Su has been a lecturer in the School of Computer and Software at the Nanjing University of Information Science and Technology since 2017. He received his PhD with distinction in communication and information systems at University of Electronic Science and Technology of China (UESTC) in 2016. He holds a B.S. in Electronic and information engineering from Hankou university and an M.S. in electronic circuit and system from Central China Normal University. His current research interests cover Internet of Things, RFID, and Wireless sensors networking. He is a member of IEEE and a member of ACM.

Zhengguo Sheng has been a senior lecturer in the Department of Engineering and Design at the University of Sussex since 2015. He received his Ph.D. and M.S. with distinction at Imperial College London in 2011 and 2007, respectively, and his B.Sc. from the University of Electronic Science and Technology of China (UESTC) in 2006. His current research interests cover the Internet of Things (IoT), connected vehicles, and cloud/ edge computing.

Alex X. Liu received his Ph.D. degree in Computer Science from The University of Texas at Austin in 2006, and is an honorary professor of Huzhou University, School of Information Engineering. Before that, he was a Professor of the Department of Computer Science and Engineering at Michigan State University. He received the IEEE & IFIP William C. Carter Award in 2004, a National Science Foundation CAREER award in 2009, the Michigan State University Withrow Distinguished Scholar (Junior) Award in 2011, and the Michigan State University Withrow Distinguished Scholar (Senior) Award in 2019. He has served as an Editor for IEEE/ACM Transactions on Networking and an Area Editor for Computer Communications. He is currently an Associate Editor for IEEE Transactions on Dependable and Secure Computing and IEEE Transactions on Mobile Computing. He has served as the TPC Co-Chair for ICNP 2014 and IFIP Networking 2019. He received Best Paper Awards from SECON-2018, ICNP-2012, SRDS-2012, and LISA-2010. His research interests focus on networking, security, and privacy. He is an IEEE Fellow and an ACM Distinguished Scientist.

Zhangjie Fu has been currently a Professor of Computer Science and the Director of Bigdata Security Lab at Nanjing University of Information Science and Technology, China. He received his PhD degree in computer science from the School of Computer, Hunan University, China, in 2012. He was a visiting scholar of Computer Science and Engineering at State University of New York at Buffalo from March, 2015 to March, 2016. His research interests include IoT Security, Outsourcing Security, Digital Forensics, Network and Information Security. His research has been supported by NSFC, PAPD, and GYHY. Zhangjie is a member of IEEE and a member of ACM.

Chenxi Huang has been currently an assistant professor in School of Informatics, Xiamen University, China. He received the Bc. degree in computer science from the Tongji University, in 2015. His research interests include image processing, image reconstruction, data fusion, three dimensional visualization, and machine learning.