IET Communications

Research Article

ISSN 1751-8628
Received on 4th November 2016
Revised 31st December 2016
Accepted on 27th January 2017
E-First on 11th April 2017
doi: 10.1049/iet-com.2016.1289
www.ietdl.org

Modified tree-based identification protocols for solving hidden-tag problem in RFID systems over fading channels

Chuyen T. Nguyen[1] ✉, Anh Tuan H. Bui[1], Van-Dinh Nguyen[2], Anh T. Pham[3]

[1]*School of Electronics and Telecommunications, Hanoi University of Science and Technology, Hanoi, Vietnam*
[2]*School of Electronic Engineering, Soongsil University, Seoul, Korea*
[3]*Computer Communications Laboratory, The University of Aizu, Aizu-Wakamatsu, Japan*
✉ *E-mail: chuyen.nguyenthanh@hust.edu.vn*

Abstract: Hidden-tag problem is one of the most important issues in the implementation of radio-frequency identification (RFID) systems. Due to effects of imperfect wireless channels, RFID tags can be hidden during the identification process by either another tag or an unsuccessful detection. The former is known as the capture effect (CE) while the latter is the detection error (DE). This study newly proposes two modified tree-based identification protocols, namely tweaked binary tree (TBT) and tweaked query tree (TQT), which are able to tackle the hidden-tag problem caused by both the CE and DE. The performance of the proposed TBT and TQT protocols, in terms of the average number of slots required to detect a tag, and the tag-loss rate, is evaluated in comparison with that of previously proposed ones. Computer simulations and numerical results confirm the effectiveness of the proposed protocols.

1 Introduction

Radio-frequency identification (RFID) is an emerging technology, which allows an automatic identification of objects over radio frequency (RF) channels. The technology promises to revolutionise smart supply chain, medical tracking, inventory management, and many other applications in the Internet of Things (IoT) paradigm [1–5]. An RFID system generally consists of one RFID reader and tags, where the main task of the reader is to detect all tags fast, completely, and reliably. To do so, the reader first broadcasts a request message to initialise the communication. Each tag then transmits a response message with its unique identity (ID) back to the reader upon receiving the request. Due to sharing the same RF channel, signal collision occurs when multiple tags respond to the reader simultaneously. Consequently, received signals may be corrupted, which makes it erroneous to recover the signals from the tags at the reader.

To tackle the signal collision problem, time division multiple access (TDMA)-based protocols, which allocate each tag a distinguished discrete time slot to reply, are widely implemented in RFID standards thanks to their robustness and simplicity [6–9]. Practically, TDMA-based protocols are classified into two main approaches: tree-based [10–14] and aloha-based [15–20]. Aloha-based protocols, on the one hand, use a frame composed of time slots, and each tag randomly responds in one of the time slots. The key issue in aloha-based protocols is to optimise the selection of the frame size, which is difficult in cases of unknown, large-scale RFID population and/or there is a significant impact from imperfect RF channels [21]. Tree-based protocols, on the other hand, counter the tag collision problem by continuously and randomly splitting a set of collided tags into two subsets and assigning dissimilar re-transmission time slots for the subsets. Depending on the splitting mechanisms, tree-based protocols are further expanded into binary tree (BT) and query tree (QT) ones. In BT protocols, colliding tags continuously split randomly into two subsets: one will respond immediately in the next reading cycle while the other will have to wait for one cycle. In QT protocols, tags first compared their IDs with a binary sequence queried by the reader, and then respond only when the comparison result is true.

Tree-based protocols are usually studied under an assumption of no impairments from physical channels, i.e. the signal-to-noise ratio (SNR) is always sufficient for tag detection, and signal collision is the only performance degrading factor. Nevertheless, this assumption is not practically realistic due to effects of channel fading, noise, and co-channel interference. Indeed, in [22–24] the physical layer effects such as noise and multi-path fading were implemented and reported to significantly degrade the performance of commercially available readers and tags. Moreover, when the effects are taken into consideration, the received SNR of a particular tag's response at the reader may fall below a detection threshold. This phenomenon is called ‘detection error’ (DE) resulting in an unsuccessful detection of the tag [25–27]. The corresponding time slot is consequently observed as in empty state, and the unsuccessfully detected tag is hidden from the identification process. In addition, a tag might be still recognised even in the midst of collision (i.e. signal interference from other tags) when its signal-to-interference-plus-noise ratio (SINR) at the reader is sufficient for a successful detection. The corresponding time slot is observed as in successful state, however, other not-detected-yet tags involving in the collision therefore are hidden. This phenomenon is called ‘capture effect’ (CE) and its impact on the system performance has been extensively studied in the literature both in theoretical [28–33] and experimental aspects [34, 35].

Over the years, several solutions for the hidden-tag problem have been proposed in RFID systems employing tree-based protocols, including both QT [36] and BT [37]. The common idea of these solutions is to require the reader to send an acknowledgement (ACK) message indicating the identified ID (IID) of the recognised tag. On the one hand, in two QT protocols proposed in [36], namely general QT 1 (GQT1) and general QT 2 (GQT2), the reader continually expands a successful query. As a result, while the recognised tag receives the ACK message, and thus can keep silent, hidden tags by the CE can take an opportunity to respond to the expanded queries. More specifically, the GQT1 expands the successfully query to one more bit, i.e. to have two additional queries, while in the GQT2 protocol, whose the performance is better than that of GQT1, the reader just uses the same successful query one more time. On the other hand, in the general BT-based (GBT) protocol proposed in [37], tags hidden by the CE, upon receiving the ACK, can be aware of their status (of not being recognised). Then, by using additional flags, which will be described later in this paper, hidden tags are allowed to respond for future requests by the reader.

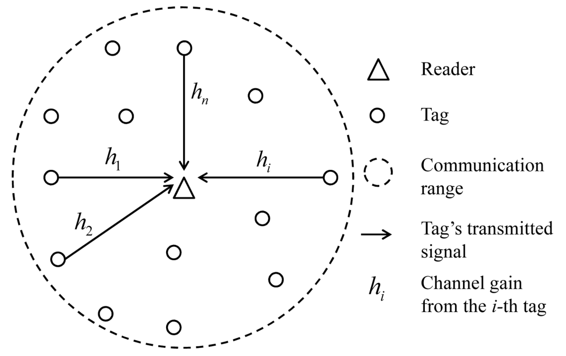

Fig. 1 *Model of an RFID system with one reader and multiple tags*

While previously proposed solutions are effective in coping with the hidden-tag problem, only the CE has been considered. In large-scale RFID systems, impacts of imperfect channels, especially channel fading and noise, could result in a considerable DE probability. This would significantly affect the identification process [38, 39]. The purpose of this paper is therefore to propose two modified tree-based protocols, which aim at tackling the hidden-tag problem caused by both the CE and DE more efficiently. The two protocols, namely tweaked BT (TBT) and tweaked QT (TQT), are developed based on the original GBT and GQT2, respectively; and the key idea is to have additional mechanisms for empty states, i.e. states that possibly have tags hidden by DE effects. It is noteworthy to mention that this paper is an extension of our previously published conference papers in [38, 39] on BT and QT protocols. Two important extensions are (i) the discussion on the impact of physical layer to illustrate the impacts of CE and DE and (ii) the additional results and insightful discussion to highlight the merits of the proposed protocols. In addition, this paper would provide, for the first time, a systematic discussion on the performance of tree-based protocols under the impact of both CE and DE. The advantages of proposed protocols are then confirmed by computer simulations in terms of the average number of slots for a successful detection of a tag, and the tag-loss rate (which is defined as the ratio between the number of missing tags and the whole tag cardinality. It is seen that they could achieve a shorter identification delay while maintaining a remarkably lower tag-loss rate compared with conventional ones.

The remaining of this paper is organised as follows: In Section 2, the considered RFID system using tree-based protocols, hidden-tag problem, and previous studies are described. Proposed protocols, and simulation results and discussions are presented in Sections 3 and 4, respectively. Finally, conclusions are drawn in Section 5.

2 RFID system model and hidden-tag problem

2.1 System model

We consider a typical RFID system consisting of a reader and n tags within its communication range, as illustrated in Fig. 1. Each tag is assigned a unique ID in the form of a binary sequence. The reader employs a tree-based protocol (either BT or QT) by which after each reader's request, responses of tags are limited within a time slot. Depending on the decoding result by the reader, a slot can be either *success*, *collision*, or *empty* (i.e. no tag responds). After each decoding, the reader will broadcast an ACK message so that all tags will be aware of the status of the slot (success/collision or empty).

Without loss of generality, the received signal in the jth slot at the reader can be written as

$$y_j = \sum_{i \in \mathscr{K}_j} h_i x_i + w_j, \tag{1}$$

where x_i is the responding signal from the ith tag. h_i is the channel gain between the reader and the ith tag, which can be assumed to be Rayleigh distributed [40] where its probability density function (pdf), denoted by $f_{h_i}(h)$, is written as

$$f_{h_i}(h) = 2h\,\mathrm{e}^{-h^2}, \quad h > 0. \tag{2}$$

It is noted that the channel gain in this work is normalised such that $E[|h_i|^2] = 1$ for further convenience. w_j is the received additive white Gaussian noise (AWGN) with zero-mean and σ^2-variance. Therefore, we can define the received instantaneous SNR of the ith tag ($\mathrm{SNR}_i = h_i^2/\sigma^2$, where the reader's transmit power is assumed to be 1, while the signal path-loss is ignored since it is negligible for indoor RFID with short range transmissions [41]. $\mathscr{K}_j$ represents a set of tags that respond to the jth query, and we also denote $|\mathscr{K}_j|$ as the number of elements in $\mathscr{K}_j$. When no physical channel impairments are considered, a time slot should be detected as in empty, successful, or collision state when $|\mathscr{K}_j| = 0$, $|\mathscr{K}_j| = 1$, or $|\mathscr{K}_j| \geq 2$, respectively.

Under impacts of channel fading and noise, DE is the phenomenon when a tag is not successfully detected even when there is no collision. Specifically, the DE can happen in the jth slot with only one transmission when the received SNR is lower than a predefined threshold γ [32]. We denote by β_j the DE probability of the jth slot. Since tags randomly respond and the path-loss is ignored, the DE probability can be supposed to be constant in every jth slot with $\mathscr{K}_j = 1$. We denote this probability as β and it can be expressed as

$$\begin{aligned}\beta = \beta_j &= \Pr\left(\mathrm{SNR}_i \leqslant \gamma \,|\, \mathscr{K}_j = 1, \quad i \in \mathscr{K}_j\right)\\ &= \Pr\left(\frac{h_i^2}{\sigma^2} \leqslant \gamma \,|\, \mathscr{K}_j = 1, \quad i \in \mathscr{K}_j\right).\end{aligned} \tag{3}$$

Furthermore, CE happens when a specific tag's SINR is large enough for the reader's detection even in the presence of the response from one or more other tags, i.e. for a specific slot jth with $\mathscr{K}_j \geq 2$, the corresponding received SINR is higher than the threshold γ. Mathematically, the probability of CE in the jth slot, denoted by α_j, can be given as

$$\begin{aligned}\alpha_j &= \Pr\left(\max_{i \in \mathscr{K}_j} \{\mathrm{SINR}_i\} > \gamma \,|\, \mathscr{K}_j \geqslant 2\right)\\ &= \Pr\left(\max_{i \in \mathscr{K}_j} \left\{\frac{h_i^2}{\sum_{k \in \mathscr{K}_j, k \neq i} h_k^2 + \sigma^2}\right\} > \gamma \,|\, \mathscr{K}_j \geqslant 2\right),\end{aligned} \tag{4}$$

where SINR_i is defined as the received SINR of the ith tag. It is observed that α_j decreases as $\mathscr{K}_j$ increases because of the higher interference and the lower probability of the capture effect. For the sake of simplicity, we also assume that the probability of CE of the RFID system, denoted as α, is constant in all slots regardless of the value of $\mathscr{K}_j$. Obviously, α is upper bounded by α_j^* with $\mathscr{K}_j = 2$ as

$$\alpha \leq \alpha_j^* = \Pr\left(\max\left\{\frac{h_i^2}{h_k^2 + \sigma^2}, \frac{h_k^2}{h_i^2 + \sigma^2}\right\} > \gamma \,\middle|\, \mathscr{K}_j = 2\right). \tag{5}$$

In the following discussion and analysis, $\alpha = \alpha_j^*$ is used, and this will result in the lower bound on system performance. Due to the effects of both the CE and DE, related tags may be ignored (hidden) in the identification process, which is also known as the hidden-tag problem. In the following section, the brief of BT, QT, and impacts of the hidden-tag problem on the RFID system performance will be discussed in detail.

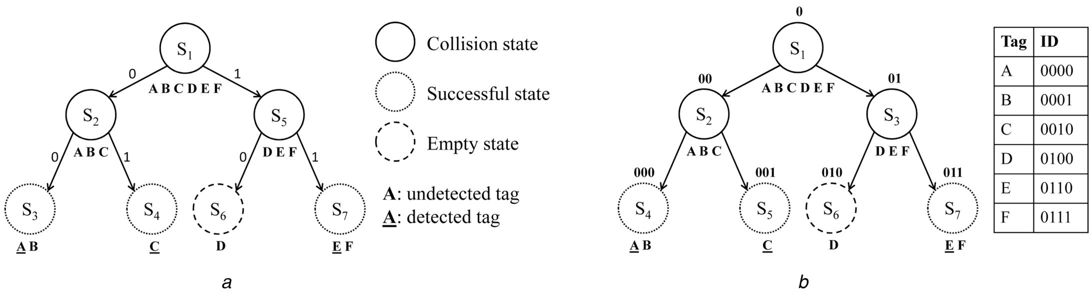

Fig. 2 *(a) BT and (b) QT protocols operation*

Table 1 The identification process of BT-based and QT-based protocols used in Fig. 2

Slot	BT								QT		
	RC	TC_A	TC_B	TC_C	TC_D	TC_E	TC_F	State	Query list	Query	State
S_1	0	0	0	0	0	0	0	collision	0	0	collision
S_2	1	0	0	0	1	1	1	collision	00, 01	00	collision
S_3	2	0	0^a	1	2	2	2	successful	01, 000, 001	01	collision
S_4	1	−1	−1	0	1	1	1	successful	000, 001, 010, 011	000	successful (A, B^a)
S_5	0	−1	−1	−1	0	0	0	collision	001, 010, 011	001	successful (C)
S_6	1	−1	−1	−1	0^b	1	1	empty	010, 011	010	empty (D^b)
S_7	0	−1	−1	−1	−1	0	0^a	successful	011	0011	successful (E, F^a)

[a, b]Respectively denote tags hidden by CE, DE.

2.2 Tree-based protocols and hidden-tag problem

In the case of the BT protocol, the reader and tags maintain their own counters, denoted as RC for the reader counter and TC for a tag counter. Tags are allowed to transmit when TC = 0, and initially, the value of all counters is set to be zero. When a slot is a collision one, all collided tags (i.e. tags that responds to the recent request from the reader) are randomly split into two subsets with equal probabilities. Tags in one subset will keep the same TC (i.e. TC = 0) while others increase their TC by one (i.e. TC = 1). At the same time, the reader and other tags that do not involve in the collision increase their RC and TC by one, respectively. If the slot is success or empty, the reader and all tags decrease their counters by one. When the related tag is successfully read as TC = −1, this tag will keep silent in subsequent requests. RC = −1 indicates that all tags have been successfully read, and therefore the reader can terminate the identification process.

An example of the BT protocol is illustrated in Fig. 2*a* and Table 1. After the first reading (in slot 1, named as S_1), six tags {A, B, C, D, E, and F} are split into two subsets of {A, B, C} and {D, E, F}. TCs of the first subset are zero while they are one for D, E, and F. In the second slot, collision happens again (with tags A, B, and C). TC increases in D, E, and F, while the splitting process is again applied for the {A, B, C} subset. Tag C will be then successfully read in slot 4.

In this example, two CEs happen in slots 3 and 7 where tags B and F are hidden by A and E, respectively. DE happens in slot 6 where tag D cannot be by read (hidden) due to its low SNR. After slot 7, the reader terminates the identification process since RC = −1. However, we can see that three hidden tags, i.e. B, D, and F are missed. The problem clearly affects the performance of the BT protocol.

In the case of the QT protocol, as illustrated in Fig. 2*b* and also in Table 1, the reader sends a query containing a bit string to probe tags. Then, tags whose first bits of their IDs match with the string respond to the reader. To implement this protocol, the reader maintains a query list storing possible queries for next readings. When the query list becomes empty, the process is terminated. If a query results in a collision (slot 1 with a query of 0), it is expanded into two new queries by adding one more binary bit (0 or 1) to the end of the string, i.e. 00 and 01. Otherwise (a successful or an empty slot is observed), the query is removed from the query list and the next query in the list will be called. We can see this performance via slots 4 and 5, where queries 000 and 001, respectively, are removed. After slot 7, the query is empty and thus, the identification process is completed.

Similar to the BT protocol, the hidden-tag problem also happens in the QT protocol under the impacts of imperfect channels. In our example, tags B, F, and D are hidden in slots 4, 7, and 6 due to the CE and DE, respectively. They also remain undetected after the identification process.

3 Conventional solutions for hidden-tag problem

3.1 General BT

GBT protocol [37] is modified from the original BT to especially find hidden tags caused by the CE. In this protocol, the identification process is divided into multiple cycles where each cycle corresponds to a BT. Tags hidden in a cycle can be re-called in the next one.

There are two key modifications in GBT. First, instead of broadcasting only the status of the recent slot, the reader also includes (i) the IID of the successfully detected tag and (ii) the current value of its counter (RC) in the ACK message after each reading. Upon receiving the ACK message of a success slot, a tag could verify whether or not it is the detected one. As a result, tags, which are possibly hidden by the CE, can realise their status and therefore, will not decrease their TC. Instead, they will set their TC = RC.

Second, the reader maintains an additional Boolean parameter, namely *extension flag* (*EF*) to indicate that there are tags that might be hidden and another cycle is needed. Specifically, at the beginning of each cycle the flag is *False* by default (and also we have RC = 0). In case of a successful slot and EF = *False*, EF = *True* is set. However, different from the original BT protocol, RC is kept unchanged. When EF = *True*, similar to the original BT protocol, RC decreases by one for every successful slot. For example, as shown in Fig. 3 and Table 2, in slot 3, EF turns to *True* and RC is kept unchanged. Tag B hidden in slot 3 sets TC_B = RC, and re-transmits in the next cycle where it is successfully read in slot 9. In slot 4, EF is kept unchanged while RC decreases by one. The EF flag turns back to *False* when RC reaches zero, which indicates a new cycle. This happens in slot 8. The whole identification process terminates when EF = *False* at RC = 0 and an empty slot is recorded. This happens at slot 11 (which can also be considered as the third cycle).

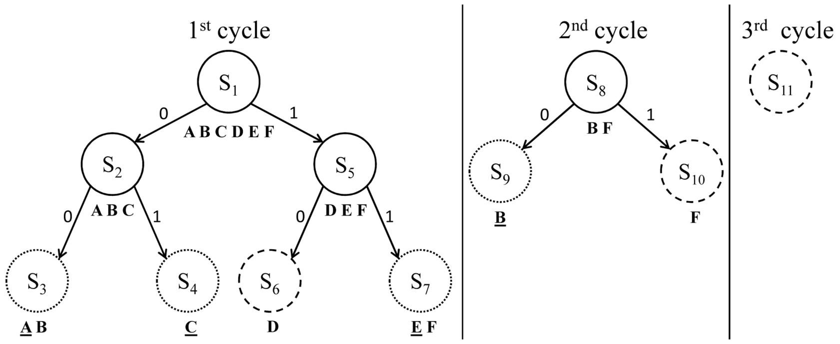

Fig. 3 *GBT protocol with hidden tags caused by the CE and DE*

Table 2 The identification process of GBT protocol used in Fig. 3

Cycle	Slot	Parameters								Feedback
		RC	EF	TC_A	TC_B	TC_C	TC_D	TC_E	TC_F	
First	1	0	F	0	0	0	0	0	0	collision
	2	1	F	0	0	0	1	1	1	collision
	3	2	F	0	0[a]	1	2	2	2	successful
	4	2	T	−1	2	0	1	1	1	successful
	5	1	T	−1	1	−1	0	0	0	collision
	6	2	T	−1	2	−1	0[b]	1	1	empty
	7	1	T	−1	1	−1	−1	0	0[a]	successful
Second	8	0	F	−1	0	−1	−1	−1	0	collision
	9	1	F	−1	0	−1	−1	−1	1	successful
	10	1	T	−1	−1	−1	−1	−1	0[b]	empty
Third	11	0	F	−1	−1	−1	−1	−1	−1	empty

[a], [b] Respectively denote tags hidden by CE, DE.

T, F, respectively, denote *True*, *False*.

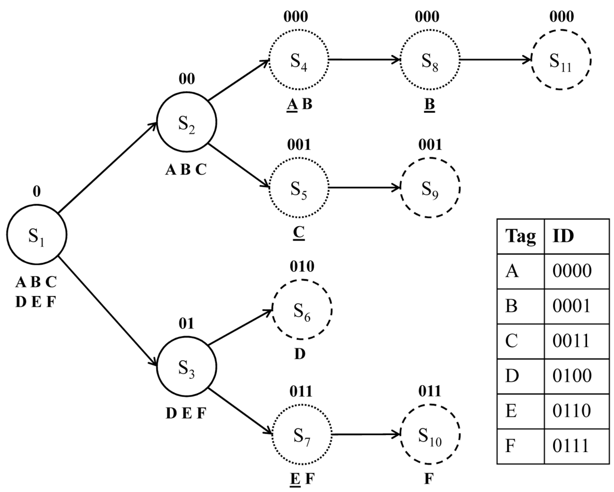

Fig. 4 *GQT2 protocol with hidden tags caused by the CE and DE*

It is clearly seen that the GBT protocol, which requires additional slots, gives tags hidden in successful slots (B and F) an opportunity of re-transmission in the next cycle where they might be detected (B). This confirms the effectiveness of GBT in solving the hidden-tag problem caused by CE. However, tags are hidden by DE cannot be recognised by GBT. As seen in slots 6 and 10, tags D and F are hidden due to the DE and they are not detected after the completion of the identification process.

3.2 Generalised QT 2

GQT1 and GQT2 protocols [36] are also modified from the original QT to solve the hidden-tag problem. Note that due to the superiority of GQT2 over GQT1, we only consider CQT2 in the comparison with our proposals.

The main idea of GQT2 protocol is that it re-reads successful queries multiple times to ensure that no tags are hidden, which is illustrated via an example in Fig. 4 and Table 3. In particular, if a query results in a successful slot (query 000 in slot 4), it is moved to the end of the query list to be re-called later. Thanks to this mechanism, tag B hidden by the CE in slot 4 is re-read at slot 8 and is successfully re-cognised. In addition, each tag maintains a special Boolean flag parameter (*F*), which is initially set to be *False* at the beginning of the identification process. After each successful read, similar to GBT protocol, the reader is required to send an ACK message containing the IID of the successfully detected tag. The detected tag, upon receiving the ACK, turns its *F* to *True* and thus, keeps silent afterwards (e.g. A, C, E in slots 4, 5, 7, respectively).

Nevertheless, we can see that tags D and F are hidden due to the DE in slots 6 and 10, respectively. Evidently, these hidden tags are not recognised because of the fact that GQT2, as the same as the GBT, does not consider the hidden-tag problem caused by DE.

4 Proposed protocols and analysis

In this section, we describe the two newly proposed TBT and TQT protocols to further improve the GBT and GQT2 in solving the hidden-tag problem. Similar examples as in the previous section will be used to illustrate the effectiveness of the proposed solutions.

4.1 TBT protocol

The TBT protocol adopts all features of the conventional GBT as mentioned in the previous section, i.e. the Boolean parameter EF and the modification of RC in case of successful slot. Nevertheless, three different/additional schemes are proposed in TBT:

- Firstly, when an empty slot is observed and the status of EF is *False*, EF also turns *True* and the RC is kept unchanged (the same as the case of a successful slot in the conventional GBT).

Table 3 The identification process of GQT2 protocol used in Fig. 4

Slot	Query list	Query	Parameters						State
			F_A	F_B	F_C	F_D	F_E	F_F	
1	0	0	F	F	F	F	F	F	collision
2	00, 01	00	F	F	F	F	F	F	collision
3	01, 000, 001	01	F	F	F	F	F	F	collision
4	000, 001, 010, 011	000	T	F[a]	F	F	F	F	successful
5	001, 010, 011, 000	001	T	F	T	F	F	F	successful
6	010, 011, 000, 001	010	T	F	T	F[b]	F	F	empty
7	011, 000, 001	011	T	F	T	F	T	F[a]	successful
8	000, 001, 011	000	T	T	T	F	T	F	successful
9	001, 011, 000	001	T	T	T	F	T	F	empty
10	011, 000	011	T	T	T	F	T	F[b]	empty
11	000	011	T	T	T	F	T	F	empty

[a, b]Respectively denote tags hidden by CE, DE.

T, F, respectively, denote *True*, *False*.

Reader

```
RC = 0, EC=1
Send start command
While(RC≥0)
If(RC=0)
      EF = false
Listen to signal:
If no signal
    If(RC>0)
        If(EF=false)
            EF = true
        Else
            RC = RC - 1
    Else
        If(EC>0)
            EC = EC - 1
        Else
            RC = - 1 †
    Respond: (empty,RC)
Else
    Try to decode ID from signals:
    If an ID is decoded
        If(EF = false)
            EF = true
        Else
            RC = RC - 1
        Respond: (successful,IID,RC)
    Else
        RC = RC + 1
        Respond: (collision)
```

Tag

```
TC = 0
Received start command
While(TC ≥0)
If(TC=0)
    Send its ID
    Received feedback (f,IID,RC)
    If(f=successful)
        If(ID = IID)
            TC = TC - 1
        Else
            TC = RC
    Else
        If(f=collision)
            TC = TC + rand(0,1)
        Else
            TC = RC
Else
    Received feedback (f,IID,RC)
    If(f=collision)
        TC = TC + 1
    Else
        TC = TC - 1
```

† The identification process is terminated

Fig. 5 *Pseudo-codes for the reader and tags using TBT protocol*
(†The identification process is terminated)

- Secondly, an additional *extra cycle* (EC) parameter at the reader is proposed. The initial value of EC is set by user and it determines the number of additional cycles before the termination of the identification process. The purpose of this parameter is to additionally prevent the missing of the last tag due to multiple DEs. This phenomenon will be specifically described in the example afterwards.
- Thirdly, different from the conventional GBT, tags involving in the current reading are required to set their TC = RC upon receiving an ACK of an empty state from the reader. Fig. 5 presents the pseudo-codes reflecting the operations of the reader and tags of the proposed TBT protocol.

To illustrate the operation of the proposed TBT protocol, the similar example as shown in Fig. 2*a* and Table 1 is illustrated in Fig. 6 and Table 4. As it is seen, tags B, F are hidden by CE and tag D is hidden by the DE after the first cycle. Thanks to the additionally proposed schemes, although tag D is hidden in slot 6 due to the DE (and thus the corresponding slot is empty), tag D sets TC_D = RC. As a result, tag D has one more chance to be called; and so do tags B and F, in the second cycle starting from slot 8. In

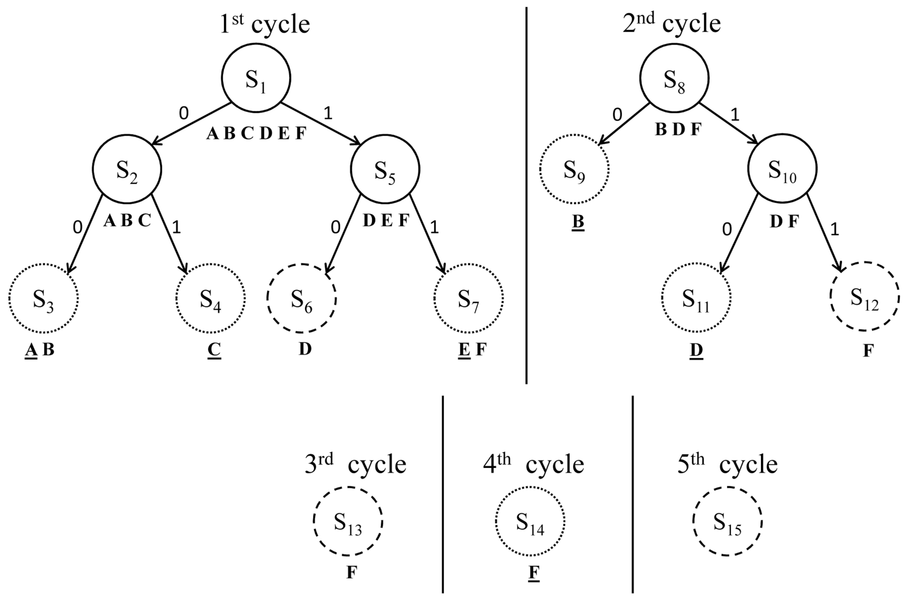

Fig. 6 *TBT protocol with hidden tags caused by the CE and the DE*

Table 4 The identification process of TBT protocol used in Fig. 6

Cycle	Slot	Parameters									Feedback
		RC	EF	EC	TC_A	TC_B	TC_C	TC_D	TC_E	TC_F	
first	1	0	F	1	0	0	0	0	0	0	collision
	2	1	F	1	0	0	0	1	1	1	collision
	3	2	F	1	0	0^a	1	2	2	2	successful
	4	2	T	1	−1	2	0	1	1	1	successful
	5	1	T	1	−1	1	−1	0	0	0	collision
	6	2	T	1	−1	2	−1	0^b	1	1	empty
	7	1	T	1	−1	1	−1	1	0	0^a	successful
second	8	0	F	1	−1	0	−1	0	−1	0	collision
	9	1	F	1	−1	0	−1	1	−1	1	successful
	10	1	T	1	−1	−1	−1	0	−1	0	collision
	11	2	T	1	−1	−1	−1	0	−1	1	successful
	12	1	T	1	−1	−1	−1	−1	−1	0^b	empty
third	13	0	F	1	−1	−1	−1	−1	−1	0^b	empty
fourth	14	0	F	0	−1	−1	−1	−1	−1	0	successful
fifth	15	0	F	0	−1	−1	−1	−1	−1	−1	empty

[a, b]Respectively denote tags hidden by CE, DE.

T, F respectively denote *True*, *False*.

the second cycle, both tags B and D are detected in slots 9 and 11, respectively. This demonstrates the ability of the proposed TBT protocol to solve the hidden-tag problem caused by DE.

Tag F is also hidden by the CE in slot 7. It is then re-called in slot 12 thanks to the original feature of GBT protocol. We assume that it is again hidden by DE (and therefore missed in case of GBT protocol). Nevertheless, thanks to the setting of TC = RC in the proposed protocol, F is re-called in slot 13. In this example, we initially set EC = 1 to indicate that one additional cycle will be proceed even after RC reaches 0 and an empty slot is observed. In the case tag F suffers from multiple DEs (i.e. hidden again in slot 13) the reader still performs one more reading, by which tag F is successfully detected. The initial value of EC could be set based on the actual situation. If the system frequently experiences tag missing due to signal degradation and noise, higher value of EC should be set.

The average total number of slots T required by TBT identification process to detect n tags can be expressed as

$$T = \sum_{k=1}^{m} T(u_k) + \epsilon, \tag{6}$$

where u_k denotes by the number of undetected tags at the beginning of the kth cycle for $k = 1, 2, \ldots, m$ (m refers to the last cycle). $T(u_k)$ is the average number of slots consumed by a TBT cycle to find u_k tags. ϵ represents the average number of extra slots to deal with the last missing tag due to the DE, which lies in a range of [EC (EC + 1)]. To calculate T, we need to find u_k and $T(u_k)$. In this case, the number of tags recognised during the kth cycle, which is denoted by d_k, must also be known. According to [38], we have

$$u_k = n - \sum_{l=1}^{k-1} d_l \tag{7}$$

$$d_k = T_1(u_k) \tag{8}$$

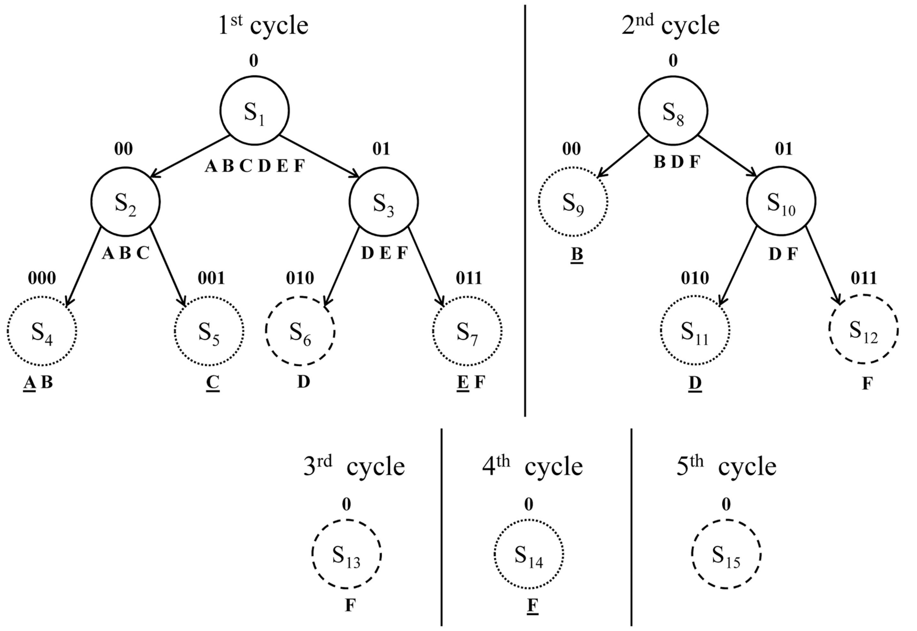

Fig. 7 *TQT protocol with hidden tags caused by the CE and the DE*

Table 5 The identification process of TQT protocol used in Fig. 7

Cycle	Slot	Query list	Query	EC	F_A	F_B	F_C	F_D	F_E	F_F	State	Storage
first	1	0	0	1	F	F	F	F	F	F	collision	∅
	2	00, 01	00	1	F	F	F	F	F	F	collision	∅
	3	01, 000, 001	01	1	F	F	F	F	F	F	collision	∅
	4	000, 001, 010, 011	000	1	F	F[a]	F	F	F	F	successful	successful{000}
	5	001, 010, 011	001	1	T	F	F	F	F	F	successful	successful{000, 001}
	6	010, 011	010	1	T	F	T	F[b]	F	F	empty	successful{000, 001} empty{010}
	7	011	011	1	T	F	T	F	F	F[a]	successful	successful{000, 001, 011} empty{010}
second	8	0	0	1	T	F	T	F	T	F	collision	∅
	9	00, 01	00	1	T	F	T	F	T	F	successful	successful{00}
	10	01	01	1	T	T	T	F	T	F	collision	successful{00}
	11	010, 011	010	1	T	T	T	F	T	F	successful	successful{00, 010}
	12	011	011	1	T	T	T	T	T	F[b]	empty	successful{00, 010} empty{011}
third	13	0	0	1	T	T	T	T	T	F[b]	empty	empty{0}
fourth	14	0	0	0	T	T	T	T	T	F	successful	successful{0}
fifth	15	0	0	0	T	T	T	T	T	T	empty	empty{0}

[a, b]Respectively denote tags hidden by CE, DE.

T, F respectively denote *True* and *False*.

$$T(u_k) = \frac{1 + 2^{1-u_k}(1-\alpha)\sum_{i=0}^{u_k-1}\binom{u_k}{i}T(i)}{1 - 2^{1-u_k}(1-\alpha)}, \quad \{T(0) = T(1) = 1\} \tag{9}$$

where $T_1(u_k)$ is the number of successful slots in a TBT cycle with u_k undetected tags, which can be calculated in the same way as $T(u_k)$ as

$$T_1(u_k) = \frac{\alpha + 2^{1-u_k}(1-\alpha)\sum_{i=0}^{u_k-1}\binom{u_k}{i}T_1(i)}{1 - 2^{1-u_k}(1-\alpha)}, \tag{10}$$

where $T_1(0) = 0, T_1(1) = 1 - \beta$. Using (7), (8), and (10) alternatively, we obtain $d_1, u_2, d_2, \ldots, u_m$ and d_m. Then, $T(u_k)$ is substituted into (6) to get T.

4.2 TQT protocol

The proposed TQT protocol also adopts key features of the GQT2, including the query list, the ACK message containing the tag IID, the state of the time slot. The Boolean flag parameter (F) of tags is also used. Nevertheless, additional parameters and operations of the reader are required for TQT, which we will introduce in Fig. 7 and Table 5 with the same above example of finding six tags A–F.

First, the reader maintains a special storage to record all queries whose the related time slot is either successful or empty in a cycle. The initial storage is an empty set denoted by ∅. For example, as shown in Table 5, at the end of the first cycle (slot 7), the storage records {000, 001, 011} for successful states, and {010} for an empty state in slot 6. It is important to note that, unlike the GBT2, the queries related to empty slots are also recorded so that tags hidden by DE will have another chance to be recalled in the next cycle.

Reader	Tag

New query = {0}, *EC*=1
Send start command
While(New query ≠ *null*)
  Clear the storage
  Query list = New query
  While(Query list is not empty)
    Send out first query *q* in Query list
    Listen to signal:
      If no signal
        Record (*q*, empty)
        Delete *q* from Query list
      Else
        Try to decode ID from signals:
        If an ID is decoded
          Record (*q*, successful)
          Delete *q* from Query list
          Broadcast the IID
        Else
          Query list = {Query list, *q*0, *q*1}
          Delete *q* from Query list
  If (number of recorded queries = 1
    & state of that query = empty)
    If *EC*>0
      EC = *EC* - 1
      New query = recorded query
    Else
      New query = *null* †
  Else
    New query = Output of bit-matching process
	F = *False*
	Received start command
	While (*F* = *False*)
	  If (query *q* = prefix bits)
	    Send its ID
	  Received feedback IID
	    If (ID = IID)
	      F = *True*

† The identification process is terminated

Fig. 8 *TQT pseudo-codes for reader and tag*
(†The identification process is terminated)

The reader starts a new second cycle from slot 8 by querying the first common bits of all bit strings in the storage (0 in this case). The storage is also reset for the new reading cycle. It is noted that we assume tags with the first bit of 0 in this example for the sake of simplicity. Practically, TQT protocol, by query messages of 0 and 1, divides all tags into two subsets and then, sequently performs the identification in each subset. The identification process terminates when the reader finds that only one query exists in its storage and that the corresponding state is empty.

Secondly, similar to TBT protocol, to further improve the performance against multiple DEs, the TQT reader also maintains an extra cycle (EC) parameter that defines the number of additional queries to be performed after the termination as mentioned above. For example, the multiple DEs happen in slots 12 and 13 of the third cycle. As EC = 1 is set, one additional query is performed in slot 14 to successfully solve the hidden-tag problem. Also, the value of EC can be increased to secure the complete identification in noisy environment. Finally, we summarise the performance of TQT in Fig. 8 with pseudo-codes for the reader and tags.

5 Simulation results and discussions

In this section, we evaluate and discuss the performance of TBT and TQT protocols in terms of the average number of slots required to detect a tag (denoted by η), and the tag-loss rate (denoted by ρ), i.e. the ratio between the number of hidden tags and the total number of tags. The obtained simulation results are compared with those of conventional protocols, including GBT and GQT2, to show the effectiveness of the proposed protocols.

First, Fig. 9*a* shows the effect of Rayleigh fading channel on the CE and DE (i.e. α and β, respectively) probabilities by simulation based on definitions in (3) and (4), given the detection threshold $\gamma = 2$. In this figure, α and β with respect to the noise variance σ^2 are presented. It is seen that, obviously, the DE increases as the noise variance increases. Nevertheless, the CE deceases as σ^2 increases. Indeed, as σ^2 increases, the SINR decreases and therefore the probability that a tag is still detected in the presence of interference and noise decreases. As the noise variance increases, the DE becomes more significant. This would result in a higher tag-loss rate in the conventional GBT and GQT protocols, in which the hidden-tag problem caused by DE is ignored.

We now compared the performance of the proposed TBT with that of GBT. First, we validate the analysis of the proposed TBT protocol by showing analytical and simulation results in Fig. 9*b*. In this figure, we present the total number of slots required by TBT to detect 200 tags for different values of probability of the CE α, given DE probability ($\beta = 0.1$) and the number of extra cycles, EC = 1. Since the number of detected tags in the *k*th cycle must be an integer, we employ three functions in (8), namely ceiling $\lceil d_k \rceil$, floor $\lfloor \boldsymbol{d}_k \rfloor$ and round $[d_k]$ (the nearest integer of d_k) to obtain the analytical results. Note that, ⌈⌉, ⌊ ⌋, and [ ] denote the ceiling, floor, and round functions, respectively. It is clearly seen that the simulation curve lies between the two analytical ones using ceiling and floor functions, while approximates the one with the round function.

Next, Fig. 10*a* compares the performance of TBT and GBT in terms of the total required slots and the average number of slots per detected tag with respect to α, when $\beta = 0.1$ or 0.2. It is seen that, the higher CE probability α results in better performance in both cases of TBT and GBT. In addition, the higher probability of DE results in the higher total number of slots required in the proposed TBT while it has no impact on the performance of GBT. For example, at $\alpha = 0.1$ and $\beta = 0.2$, more than 650 slots are required in the proposed TBT protocol while the total number of required slots in GBT is less than 550. This phenomenon is logical as the conventional GBT does not consider hidden tags caused by the DE, and because of this, it suffers pretty high tag-loss rate, as we will discuss in the next figure. Furthermore, although TBT requires more slots to complete the identification process, the average number of slots per detected tag is the same as that of GBT protocol, e.g. $\eta = 3.2$ in the mentioned example.

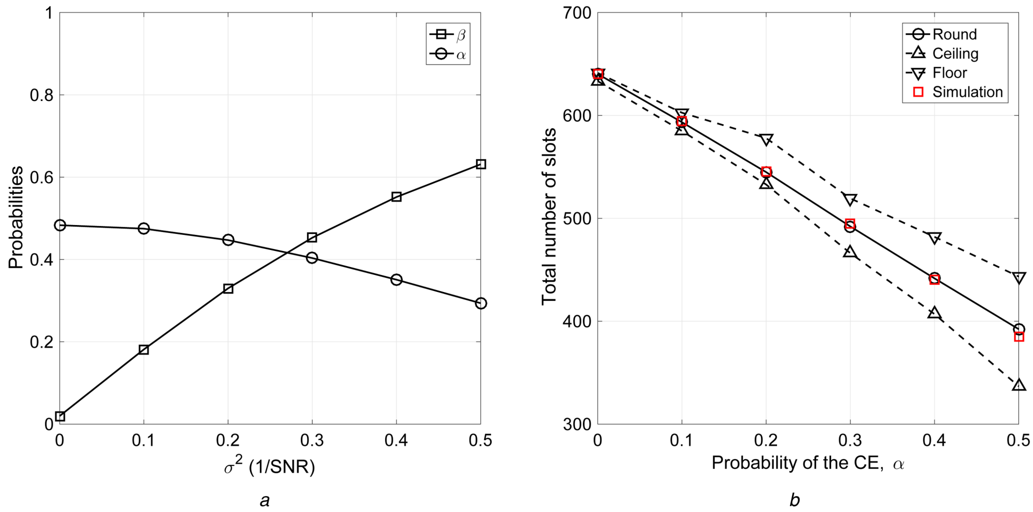

Fig. 9 ***(a)*** *CE and DE probabilities w.r.t. noise variance* σ^2 *and* ***(b)*** *total number of slots of TBT to detect 200 tags,* $\beta = 0.1$

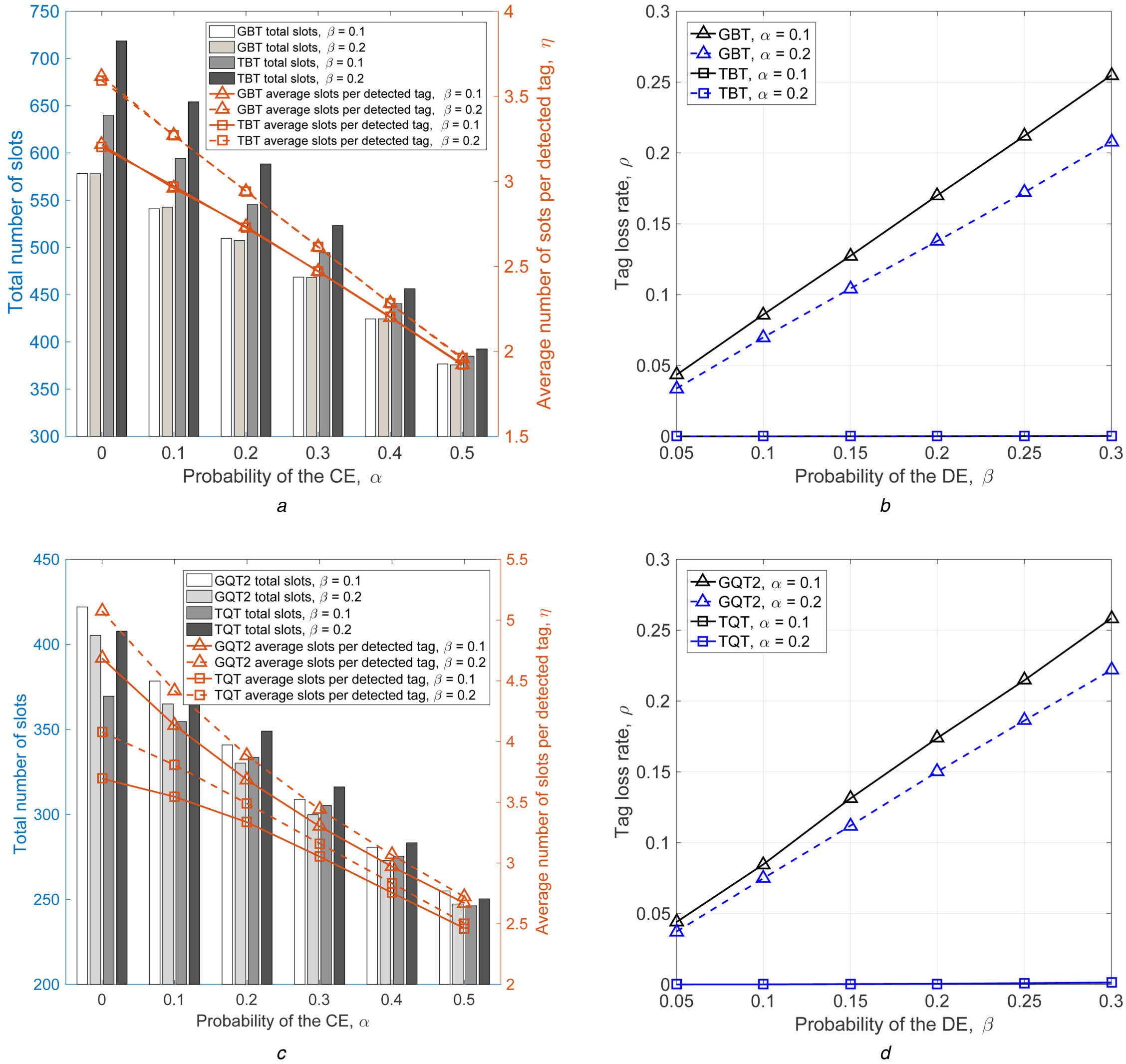

Fig. 10 ***(a)*** *Total number of slots and average number of slots for per detected tag η w.r.t. α, for* $\beta = 0.1$ *and 0.2 in GBT and proposed TBT protocols,* ***(b)*** *Tag-loss rate ρ w.r.t. β, for* $\alpha = 0.1$ *and 0.2 in GBT and proposed TBT protocols,* ***(c)*** *Total number of slots and average number of slots for per detected tag η w.r.t. α, for* $\beta = 0.1$ *and 0.2 in GQT2 and proposed TQT protocols,* ***(d)*** *Tag-loss rate ρ w.r.t. β, for* $\alpha = 0.1$ *and 0.2 in GQT2 and proposed TQT protocols*

Fig. 10*b* presents the tag-loss rates of TBT and GBT with respect to different values of probability of DE, β, when the probability of CE is fixed at $\alpha = 0.1$ or 0.2. It is clearly shown that GBT does suffer from the high loss rate as the probability of DE β increases. The proposed protocol TBT, on the other hand, achieves a significantly low tag-loss rate. As a matter of fact, the tag-loss rate in TBT protocol is almost 0% regardless of different values of α and β. This is thanks to the fact that tags hidden in a cycle will retransmit in the first slot of the next one, while TBT does not stop identification until a number of cycles with first empty slots are detected.

We now evaluate the performance of TQT and conventional GQT2 protocols with the number of tags n and EC set by 100 and 1, respectively. It is noteworthy to mention that, in QT-based schemes, each IID is assumed to take a half slot to broadcast. Moreover, since the performance of QT-based protocols significantly depends on the tag's ID distribution, we consider the most general case where each tag has a randomly generated 96-bit ID. In particular, Fig. 10*c* compares the performance of TQT and GQT2 in terms of the total number of required slots and the average number of slots per detected tag with respect to the probability of CE α, while DE probability is fixed at $\beta = 0.1$ or 0.2. Similar to the comparison of TBT and GBT protocols in Fig. 10*a*, it is logical that when α increases, the performance of both TQT and GQT2 protocols is improved. In addition, several interesting results can be observed. First, regarding the total number of slots, GQT protocol uses a higher number of slots for the identification even in the presence of DE. When the probability of DE increases, it is logical that more slots are required by our proposed TQT to complete the identification process. The conventional GQT2 protocol, however, needs fewer slots. The reason is that when GQT2 detects tags, its reader also needs an additional half slot to confirm each detected IID. Therefore, the more DE, the few slots are utilised; and the consequence is the higher number of hidden tags. Secondly, regarding the average number of slots per detected tag, our proposed TQT protocol offers a significant improvement, especially in the area of low probabilities of CE. For example, at $\alpha = 0.2$ and $\beta = 0.1$, $\eta = 3.3$ in the TQT protocol while it is 3.7 in the GQT2 even the total number of slots is higher than in the TQT. This comes from the fact that GQT2 misses hidden tags caused by DE while our proposed TQT could completely read all tags.

Finally, Fig. 10*d* validates our argument in Fig. 10*c* about the tag-loss rate performance comparison between GQT2 and the proposed TQT. We observe the tag-loss rates in both protocols with different values of β, while α is set to be 0.1 or 0.2. It is clearly seen that as the probability of DE, β increases, the tag-loss rate significantly increases in the GQT2 protocol. On the other hand, TQT detects almost all tags to maintain a perfect tag-loss rate, which approximates 0% regardless of values of α and β. This confirms the most important advantage of our proposed TQT (as well as TBT) protocol.

6 Conclusions

This paper has studied tree-based protocols for RFID systems in the effort to remedy impacts of the hidden-tag problem caused by both the CE and DE. We newly proposed two protocols of TBT and TQT, which are both based on previously proposed GBT and GQT2 protocols. The proposed protocols divided identification processes into multiple tree-based cycles. Tags hidden by both CE and DE in a cycle were re-called in the first time slot of the next one. Simulation results confirmed that the proposed protocols offered a significant performance improvement in comparison with that of conventional ones. Specifically, the obtained results showed that the tag-loss rate of proposed schemes was considerably smaller than that of conventional ones and approximated 0% regardless of probabilities of CE and DE (α and β). Meanwhile, the average number of slots required per tag could be comparable especially when α was large enough.

7 References

[1] Finkenzeller, K.: '*RFID handbook: fundamentals and applications in contactless smart cards and identification*' (Wiley & Sons, 2003, 2nd edn.)

[2] Ala, A.-F., Mohsen, G., Mehdi, M.*, et al.*: 'Internet of things: a survey on enabling technologies, protocols, and applications', *IEEE Commun. Surv. Tutor.*, 2015, **15**, (4), pp. 2347–2376

[3] Tseng, C.-W., Chen, Y.-C., Huang, C.-H.: 'Integrating transducer capability to GS1 EPC global identify layer for IoT applications', *IEEE Sens. J.*, 2015, **15**, (10), pp. 5404–5415

[4] Sara, A., Rossella, L., Sabina, M.*, et al.*: 'RFID technology for IoT-based personal healthcare in smart spaces', *IEEE Internet of Things J.*, 2014, **1**, (2), pp. 144–152

[5] Bolic, M., Simplot-Ryl, D., Stojmenovic, I.: '*RFID systems: Research trends and challenges*' (Wiley & Sons, 2010)

[6] 'EPCglobal Class 1 Generation 2 UHF Air Interface Protocol Standard Version 1.0.9', http://www.epcglobalinc.org/, 2005

[7] 'I-CODE1 Label ICs Protocol Air Interface Datasheet', http://www.semiconductors.philips.com/acrobat_download/other/identification/SL092030.pdf, January 2005

[8] Ahson, S.A., Ilyas, M.: '*RFID handbook: applications, technology, security and privacy*' (CRC Press, 2008)

[9] Choi, S.-Y., Lee, J., Kim, S.H.*, et al.*: 'Hybrid anti-collision method based on maximum throughput for RFID system', *Electron. Lett.*, 2010, **46**, (19), pp. 1346–1348

[10] Hush, D.R., Wood, C.: 'Analysis of tree algorithm for RFID arbitration'. Proc. of the IEEE Int. Symposium on Information Theory, Cambridge, USA, August 1998, p. 107

[11] Landaluce, H., Perallos, A., Bengtsson, L.*, et al.*: 'Simplified computation in memoryless anticollision RFID identification protocols', *Electron. Lett.*, 2014, **50**, (17), pp. 1250–1252

[12] Guo, Y., Li, S., Dou, J.*, et al.*: 'Deterministic cloned tag detection protocol for anonymous radio-frequency identification systems', *IET Inf. Sec.*, 2016, **10**, (1), pp. 28–32

[13] Cui, Y., Zhao, Y.: 'Mathematical analysis for binary tree algorithm in RFID'. Proc. of the IEEE Vehicular Technology Conf., Marina Bay, Singapore, May 2008, pp. 2725–2729

[14] 'EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 Mhz – 960 Mhz Version 1.2.0', http://www.gs1.org/sites/default/files/docs/epc/uhfc1g2_1_2_0-standard-20080511.pdf, accessed May 2016

[15] Chen, W.-T.: 'Performance comparison of binary search tree and framed aloha algorithms for RFID anti-collision', *IEICE Trans. Commun.*, 2008, **E91-B**, (4), pp. 1168–1171

[16] Šolic, P., Radic, J., Rožic, N.: 'Energy efficient tag estimation method for aloha-based RFID systems', *IEEE Sens. J.*, 2014, **14**, (10), pp. 3637–3647

[17] Chen, W.-T.: 'An accurate tag estimate method for improving the performance of an RFID anti-collision algorithm based on dynamic frame length ALOHA', *IEEE Trans. Autom. Sci. Eng.*, 2009, **6**, (1), pp. 9–15

[18] Kim, Y.-B.: 'Determination of optimal frame sizes in framed slotted ALOHA', *Electron. Lett.*, 2014, **50**, (23), pp. 1764–1766

[19] Subodh, P., Kyung, S.K., Seokjoo, S.: 'On maximising tag reading efficiency of a multi-packet reception capable radio frequency identification reader', *IET Commun.*., 2015, **9**, (5), pp. 701–706

[20] Deng, D.-J., Tsao, H.-W.: 'Optimal dynamic framed slotted ALOHA based anti-collision algorithm for RFID systems', *Wirel. Pers. Commun.*, 2010, **59**, (1), pp. 109–122

[21] Maichalernnukul, K., Zheng, F., Kaiser, T.: 'BER analysis of space-time coded RFID system in Nakagami-m fading channels', *Electron. Lett.*, 2014, **50**, (5), pp. 405–407

[22] Buettner, M., Wetherall, D.: 'An empirical study of UHF RFID performance'. Proc. of the 14th Annual Int. Conf. on Mobile Computing and Networking, USA, 2008, pp. 223–234

[23] Donno, D.-D., Ricciato, F., Tarricone, L.: 'Listening to tags: uplink RFID measurements with an open-source software-defined radio tool', *IEEE Trans. Instrum. Meas.*, 2013, **62**, (1), pp. 109–118

[24] Xie, L., Li, Q., Wang, C.*, et al.*: 'Exploring the gap between ideal and reality: an experimental study on continuous scanning with mobile reader in RFID systems', *IEEE Trans. Mob. Comput.*, 2015, **14**, (11), pp. 2272–2285

[25] Jacobsen, R.M., Fyhn, K., Popovski, P.*, et al.*: 'Reliable identification of RFID tags using multiple independent reader sessions'. Proc. of the IEEE Int. Conf. on RFID, Orlando, USA, April 2009, pp. 64–71

[26] Popovski, P., Fyhn, K., Jacobsen, R.M.*, et al.*: 'Robust statistical methods for detection of missing RFID tags', *IEEE Wirel. Commun.*, 2011, **18**, (4), pp. 74–80

[27] Fyhn, K., Jacobsen, R.M., Popovski, P.*, et al.*: 'Fast capture-recapture approach for mitigating the problem of missing RFID tags', *IEEE Trans. Mob. Comput.*, 2012, **11**, (3), pp. 518–528

[28] Delgado, M.V.B., Vales-Alonso, J.: 'Analysis of the identification process in active RFID systems with capture effect'. Proc. of the European Workshop on Smart Objects: Systems, Technologies and Applications, Ciudad, Spain, June 2010, pp. 1–6

[29] Shin, W.J., Kim, J.G.: 'A capture-aware access control method for enhanced RFID anti-collision performance', *IEEE Commun. Lett.*, 2009, **13**, (5), pp. 354–356

[30] Wu, H., Zeng, Y.: 'Passive RFID tag anticollision algorithm for capture effect', *IEEE Sens. J.*, 2015, **15**, (1), pp. 218–226

[31] Li, B., Wang, J.: 'Efficient anti-collision algorithm utilizing the capture effect for ISO 18000-6C RFID protocol', *IEEE Commun. Lett.*, 2011, **15**, (3), pp. 352–354

[32] Nguyen, C.T., Hayashi, K., Kaneko, M.*, et al.*: 'Maximum likelihood approach for RFID tag cardinality estimation under capture effect and detection errors', *IEICE Trans. Commun.*, 2013, **E96-B**, (5), pp. 1122–1129

[33] Yang, W., Haifeng, W., Yu, Z.: 'Capture-aware estimation for large-scale RFID tags identification', *Signal Process. Lett. IEEE*, 2015, **22**, (9), pp. 1274–1277

[34] Šolić, P., Maras, J., Radić, J.*, et al.*: 'Comparing theoretical and experimental results in Gen2 RFID throughput', *IEEE Trans. Autom. Sci. Eng.*, 2016, **PP**, (99), pp. 1–9

[35] Šolić, P., Blažević, Z., Škiljo, M.*, et al.*: 'Impact of tag responsiveness on Gen2 RFID throughput', *IEEE Commun. Lett.*, 2016, **20**, (11), pp. 2181–2184

[36] Wu, V.K.Y., Campbell, R.H.: 'Using generalized query tree to cope with the capture effect in RFID singulation'. Proc. of the 6th IEEE Consumer Communications and Networking Conf., Las Vegas, USA, January 2009, pp. 1–5

[37] Lai, Y.-C., Hsiao, L.-Y.: 'General binary tree protocol for coping with the capture effect in RFID tag identification', *IEEE Commun. Lett.*, 2010, **14**, (3), pp. 208–210

[38] Nguyen, C.T., Bui, A.T.H., Mai, V.V.*, et al.*: 'Tweaked binary tree algorithm to cope with capture effect and detection error in RFID systems'. Proc. of the 21th Asia-Pacific Conf. on Communications, Kyoto, Japan, October 2015, pp. 674–679

[39] Bui, A.T.H., Nguyen, C.T., Hoang, T.M., *et al.*: 'Tweaked query tree algorithm to cope with capture effect and detection error in RFID systems'. Proc. of the 3rd Int. Conf. on Computing, Management and Telecommunications, Da Nang, Viet Nam, December 2015, pp. 46–51
[40] He, C., Wang, Z.J.: 'Closed-form BER analysis of non-coherent FSK in MISO double Rayleigh fading/RFID channel', *IEEE Commun. Lett.*, 2011, **15**, (8), pp. 848–850
[41] Mayer, L.W., Wrulich, M., Caban, S.: 'Measurements and channel modeling for short range indoor UHF applications'. Proc. of the First European Conf. on Antennas and Propagation, Nice, France, November 2006, pp. 1–5