

A Simple and Fast Anti-collision Protocol for Large-scale RFID Tags Identification

Xiaolin Jia^{1,2*}, Yuhao Feng³ and Yajun Gu¹

¹ School of Computer Science and Technology, Southwest University of Science and Technology
Mianyang, Sichuan, 621010, CN
[e-mail: my_jiaxl@163.com]

² Key Laboratory of Testing Technology for Manufacturing Process, Ministry of Education,
Southwest University of Science and Technology, Mianyang 621010, CN
[e-mail: my_jiaxl@163.com]

³ School of Social & Behavior Science, University of Arizona
Tucson, AZ, 85721-0108, USA
[e-mail: yuhaofeng@email.arizona.edu]

*Corresponding author: Xiaolin Jia

*Received May 28, 2018; revised February 2, 2020; accepted February 10, 2020;
published April 30, 2020*

Abstract

This paper proposes a novel anti-collision protocol for large-scale RFID tags identification, named Bi-response Collision Tree Protocol (BCT). In BCT, two group of tags answer the reader's same query in two response-cycles respectively and independently according to the bi-response pattern. BCT improves the RFID tag identification performance significantly by decreasing the query cycles and the bits transmitted by the reader and tags during the identification. Computation and simulation results indicate that BCT improves the RFID tag identification performance effectively, e.g. the tag identification speed is improved more than 13.0%, 16.9%, and 22.9% compared to that of Collision Tree Protocol (CT), M-ary Collision Tree Protocol (MCT), and Dual Prefix Probe Scheme (DPPS) respectively when tags IDs are distributed uniformly.

Keywords: RFID tag identification, anti-collision protocol, bi-response pattern, degenerate collision tree, stability

This research was supported by the National Natural Science Foundation of China under grant 61471306, 61601381, 61872304, the Foundation of China Scholarship Council (CSC) under grant 201709390005, the National Defense Basic Research Projects of China under grant JCKY2017404C004, the Science and Technology Foundation of Sichuan Province under grant 2018GZ0179, 19JCQN0108, the Scientific Research Fund of Sichuan Provincial Education Department under grant 19ZD2101, 18ZD1109, 16ZD3104, the fund of Key Laboratory of Testing Technology for Manufacturing Process of Ministry of Education National Natural Science under grant 14TDZK02, and the Researching Foundation of SWUST under grant 17XNZD09.

1. Introduction

Radio frequency identification system (RFID) is one of the most important technologies used for identifying data stored in a tag's microchip using RF waves by wireless channel automatically [1]. Due to their limitless possibilities and low cost, RFID systems are powerful objects identification tools well-suited in daily applications, and become the primary components of the infrastructure of Internet of Things (IOT) and Industrial Internet of Things (IIOT) which enables every object in daily life and industrial production to be located, traced and managed [2][3]. As one of the ten emerging technologies that will change the world, RFID systems are increasingly and ubiquitously digitizing the physical environment by attaching tiny tags to objects and people. Therefore, effective and reliable identification of multiple tags or objects is very important in RFID systems and IOT applications.

However, when the reader and tags communicate or transmit data on the same channel simultaneously, collision happens, which delays tag identification and data collection, increases communication overhead, wastes bandwidth, and even causes loss of their usefulness. Therefore, an effective anti-collision protocol is very crucially in RFID system. The collisions in RFID system have been divided into tag collision and reader collision [4][5]. This paper focuses on the tag collision and the anti-collision protocol for solving it.

In RFID, the anti-collision protocol solving tag collision, also called tag identification protocol, can be classified into two categories: ALOHA-based protocol, and tree-based protocol [5][6][7]. The ALOHA-based protocols are represented by some classical protocols, such as Pure ALOHA Protocol (PA) [5], Slotted ALOHA Protocol (SA) [5], Frame Slotted ALOHA Protocol (FSA) [6], and Dynamic Frame Slotted ALOHA Protocol (DFSA) [6]. The tree-based protocols also represented by some classical protocols, such as Query Tree Protocol (QT) [7], Binary Tree Protocol (BT) [6], Binary Search Algorithm (BS) [5], and Dynamic Binary Search Algorithm (DBS) [5]. Most of the classical protocols have been used in many international standards as shown in Table 1 [8-12]. But the best identification efficiency of them generally is only about 34% - 36.8%.

Table 1. Anti-collision protocols and related standards [8]

Anti-collision Protocol	Standard Adopting the Protocol
Query tree protocol (QT)	ISO/IEC 18000-3 Mode 1
Binary tree protocol (BT)	ISO/IEC 18000-6 Type B, EPCglobal Class 0 EPCglobal Class 1
Dynamic binary search algorithm (DBS) Q Protocol [11][12]	ISO 14443-3 Type-A ISO/IEC 18000-6 Type C EPCglobal Class 1 Generation 2
Pure ALOHA protocol (PA)	ISO/IEC 18000-3 Mode 1 Extension
Slotted ALOHA (SA)	ISO/IEC 18000-3 Mode 2
Framed slotted ALOHA (FSA)	ISO/IEC 18000-3 Mode 1 Extension ISO/IEC 18000-6 Type A EPCglobal Class 1 EPCglobal Class 1 Generation 2
Dynamic framed slotted ALOHA (DFSA)	ISO/IEC 18000-3 Mode 1 ISO 14443-3 Type-B

Fortunately, the classical protocols provide a good research foundation for new technologies and methods for RFID tags identification. Many improved or hybrid protocols based on these classical ALOHA-based or tree-based protocols are proposed [12-24]. Most of the proposed protocols improves the RFID tags identification performance in some sense, such as increasing the identification efficiency, decreasing the energy consumption. However, the cost for that is generally increasing the complexity of protocol itself, which limited the practicable of the protocols, especially in a passive RFID tag system.

Reference [25] presents an efficient anti-collision protocol for RFID tags identification, named Collision Tree Protocol (CT), which improves the RFID tag identification efficiency up to 50% with a simple and straightforward method. Importantly, as discussed in [26], CT is a stable anti-collision protocol, i.e., the identification performance of CT only relies on the number of tags to be identified and the average performance is a constant. Therefore, the identification speed, system throughput, and energy consumption of the RFID system adopting CT could be calculated in advance.

In [27], a novel anti-collision protocol is proposed, named Dual Prefix Probe Scheme (DPPS), which decrease the reader's queries by sending three parameters to the tags, and the tags assemble two new prefixes and choose one sub-cycle to respond the reader's query according to the new prefixes. But DPPS increases idle cycle or empty cycle (in which no tag answers the reader's query) during tags identification, which affect the identification performance of it.

In [28], another novel and state-of-the-art RFID tags identification protocol is introduced, called M-ray Collision Tree Protocol (MCT), which improve the identification performance of it and reduce the energy cost efficiently. MCT constructs a M-ary tree structure via two or three collided bits in the tags' response. The tags generates new prefixes and chooses one branch of the tree according to the prefixes to reply the reader's query. Similarly, MCT increase the empty cycle or idle slot also, which restrict the identification performance of it especially when tags IDs are distributed uniformly.

This paper puts forward a simple and fast anti-collision protocol for RFID tag identification, named Bi-response Collision Tree Protocol (BCT). In BCT, the response-cycle of tags is divided into two sub-cycles, in which two group of tags answer the same query of the reader respectively and independently according to the bi-response pattern. Both theoretical and simulation results indicate that BCT improves the RFID tags identification performance significantly and is the fastest RFID tags identification protocol, which will be discussed in detail in section 4 and section 5.

The rest of the paper is organized as follows. Section 2 exhibits the related previous protocols including Collision Tree Protocol (CT), Dual Prefix Probe Scheme (DPPS), and M-ray Collision Tree Protocol (MCT). Section 3 introduces the Bi-response Collision Tree Protocols (BCT), and compares it to CT, DPPS and MCT by an identification example. Section 4 analyzes the main identification performance of BCT such as time complexity, identification efficiency, communication complexity, and identification speed. Section 5 is the simulation results of BCT when tags IDs distribute uniformly and continuously. Finally, the conclusions of this paper are presented in Section 6.

2. Related Work

Besides the anti-collision protocols solving tag collision in RFID system as introduced above, there are many other protocols to detect or search missing tags, such as tag IDs collection protocols [29][30], missing tag detection protocols [31-36], tag researching protocols [37][38].

Many missing tag detection or searching protocols only identify the IDs of the missing tags instead of all tag IDs, and terminate the search process when the search result meets the accuracy requirement [31][38]. Therefore, anti-collision protocol is the foundation of missing tag detection protocols.

Because CT is used as the contrast protocol to analyze and compare the performance of BCT, this section introduce CT in brief, and the characteristic and performance of CT have been analyzed in detail in [25][26].

The most important characteristic of Collision Tree Protocol (CT) is that it uses the first collided bit to split the collided tags into two sub-groups. In CT, both generating new prefixes and splitting tags are according to the first collided bit. For example, to a query $p_1p_2 \cdots p_j$ and a response $r_1r_2 \cdots r_{c-1}r_c$, where $p_i, r_i \in \{0,1\}$ and r_c is the first collided bit, the reader generates two new prefixes: $p_1p_2 \cdots p_jr_1r_2 \cdots r_{c-1}0$ and $p_1p_2 \cdots p_jr_1r_2 \cdots r_{c-1}1$. At the same time, the tags which match $p_1p_2 \cdots p_j$ are split into two sub-groups. The tags in one of them match $p_1p_2 \cdots p_jr_1r_2 \cdots r_{c-1}0$; the tags in the other one match $p_1p_2 \cdots p_jr_1r_2 \cdots r_{c-1}1$. Initially, the reader queries the tags with an empty string ε as the initial prefix, and all tags response the query with their IDs. If there is no collision in the response, the reader identifies a tag. If collision occurred, the reader generates two new prefixes and queries with new prefix repeatedly, until all tags are identified.

Fig. 1(a) illustrates the identification process of CT used to identify six RFID tags with 0010010, 0110001, 0110110, 1001110, 1010101, and 1100110 as their IDs. **Fig. 1(b)** is the collision tree corresponding to the identification process of this example. **Table 2** is the work process of CT in this example in detail. In this example, CT identifies the six tags through 11 query-response cycles without any idle cycle. And 26 and 130 binary bits are transmitted by the reader and tags respectively during the identification.

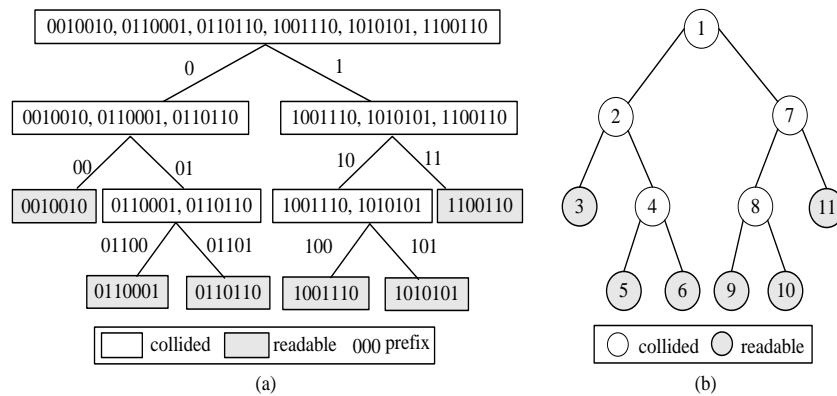


Fig. 1. Example of CT identifying tags 0010010, 0110001, 0110110, 1001110, 1010101, and 1100110: (a) identification process, (b) collision tree.

Table 2. The work process of CT to identify tags:
0010010, 0110001, 0110110, 1001110, 1010101, and 1100110.

Cycle	Prefix	ReceivedID	Channel State	New Prefix	Prefix in Stack
1	ϵ	xxxxxxx	collided	0,1	0,1
2	0	x10xxx	collided	00,01	00,01,1
3	00	10010	tagID:0010010	---	01,1
4	01	10xxx	collided	01100,01101	01100, 01101, 1
5	01111	01	tagID:0110001	---	01101, 1
6	01100	10	tagID:0110110	---	1
7	1	xxx1xx	collided	10,11	10,11
8	10	xx1xx	collided	100,101	100,101,11
9	100	1110	tagID:1001110	---	101,11
10	101	0101	tagID:1010101	---	11
11	11	00110	tagID:1100110	---	NULL

Here, ϵ represents empty string, --- means no new prefix is generated, NULL denotes the stack is empty.

3. Bi-response Collision Tree Protocol

This section introduces the Bi-response Collision Tree Protocol (BCT) in detail. Before that, several principles and definitions used in BCT are introduced.

3.1 Principles and Definitions

- Manchester code: which has been used in DBS [4] and ISO/IEC 14443 type A [28], is adopted in this paper also to encode the tags' response message. Manchester code, which is introduced in detail in [4][25][28], provides a method to trace a collision to an individual bit and find where the collided bit is because it enables bit level synchronization among tag responses [28]. Therefore, the reader can distinguish the correct and collided bits in the tags' response and generate new prefixes according to the collided bit.

- Duality and definiteness principle: According to the coding rule of EPC (electronic product code), every tag ID has a unique value in the world. At the same time, each bit in tag ID has two statues or values definitely, say 0 and 1. And each bit of tags IDs can divide the tags into exactly two sub-groups: in one of them, the value of the bit is 0; in the other one, the value of the bit is 1. This relation and regulation is called duality and definiteness principle.

- Characteristic bit: which is used by the tags to decide and choose one sub-cycle to transmit their IDs according to the bi-response pattern if their IDs match with the prefix in the query command. For example, assuming $p_1p_2 \dots p_k$ is the prefix send by the reader with the Query command, and $b_1b_2 \dots b_k b_{k+1} \dots b_m$ is a tag ID, if $b_1b_2 \dots b_k = p_1p_2 \dots p_k$, then b_{k+1} is the characteristic bit. Therefore, the location of the characteristic bit changes dynamically during the tags identification.

- Cycle and Sub-cycles: In RFID tag identification, an identification cycle is the time interval during which the reader and the tags could complete a whole communication, e.g. the reader sends query command and receives the response from the tags (correspondingly, the tags receive the reader's command and respond the query). Generally, the cycles are divided into two sub-cycles. One is query-cycle (Q-cycle) during which the reader transmits query command with a parameter to inquire the tags, and the tags in the reader's field receive the command. The other one is response-cycle (R-cycle) during which the tags answer the reader's query, and the reader receives the tags' responses.

In this paper, the identification cycle is divided into three sub-cycles: Q-cycle, R0-cycle, and R1-cycle. In Q-cycle, the reader sends query command with a prefix to inquire the tags, and the unidentified tags get the prefix and compare it with their IDs. In R0-cycle, which is just after the Q-cycle, the tag whose ID matches the prefix and the characteristic bit equals 0 answer the reader's query. In R1-cycle, which is just after the R0-cycle, the tag whose ID matches the prefix and the characteristic bit equals 1 answer the reader's query.

- Bi-response pattern: As discussed above, according to the value of the characteristic bit, the tags can choose different response sub-cycles, R0-cycle or R1-cycle, to answer the reader's query. In other words, two group of tags answer the reader's same query concurrently in the same identification cycle (but in different response sub-cycles: R0-cycle or R1-cycle). This querying and responding process in RFID tags identification is named bi-response pattern.

3.2 Bi-response Collision Tree Protocol

The working process of Bi-response Collision Tree Protocol (BCT) is illustrated in Fig. 2. Initially, tagID=NULL, and ε is pushed into the stack. The reader pops a prefix from the stack (prefix pool) and sends a query command with the prefix, i.e., $p[0, \dots, k-1]$, as parameter, as shown from lines 01 to 06 in Fig. 2(a). The unidentified tags in the reader's field compare their IDs with the received prefix and answer the query according to the bi-response pattern and the value of their characteristic bit as shown in Fig. 2(b). If the compared result is true, i.e., $p[0, \dots, k-1] = ID[0, \dots, k-1]$, and the characteristic bit, say $ID[k]$, is "0," the tags immediately transmit their IDs except for the part same with the prefix, i.e., $ID[k, \dots, n-1]$, to respond the reader's query in R0-cycle, as shown from lines 08 to 11 in Fig. 2(b). Similarly if the compared result is true and the characteristic bit is "1," the tags wait a response-cycle (R0-cycle) and respond the reader's query in R1-cycle with their remainder IDs, i.e., $ID[k, \dots, n-1]$, as shown from lines 12 to 15. Initially, an empty string ε is used as the first prefix, and all tags transmit their whole IDs in the R0-cycle or R1-cycle according to the value of their first bit in their ID respectively.

If collision happens in the received bit string, notated as receivedID in Fig. 2(a), the reader generate one new prefixes and pushes it into the stack. For example, assuming $p_0 p_1 \dots p_{k-1}$ is the prefix sent by the reader and $p_k \dots p_{k+j-1} p_{k+j} \dots p_{n-1}$ is the bit string received by the reader, here $p_i \in \{0,1\}$, $0 \leq i < n$, n is the length of tag ID, $p_k p_{k+1} \dots p_{k+j-1}$ is the correct bits and p_{k+j} is the first collided bit, the reader generates only one new prefix $p_0 p_1 \dots p_{k-1} p_k \dots p_{k+j-1}$, and pushes it into the stack, as shown from lines 09 to 16 in Fig. 2(a). If there is no collision in the received bit string, the reader identifies one tag, whose ID is the concatenation of the prefix and the received bit string, i.e., prefix+receivedID and $p_0 p_1 \dots p_{k-1} p_k \dots p_{n-1}$, as shown from lines 17 to 20. After a tag is identified, the reader send command ACK(tagID) to inform the tag and not to answer the reader's following query. The reader repeats this process until all tags are identified.

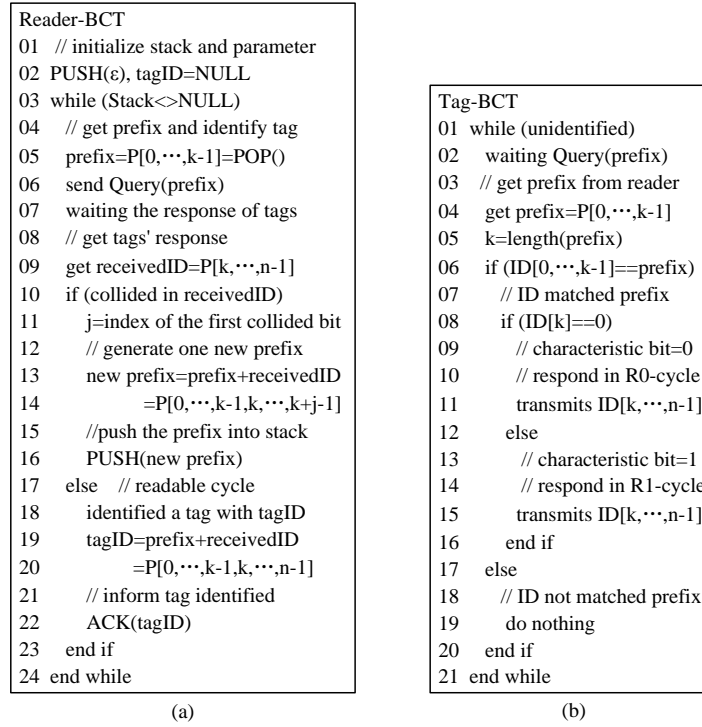


Fig. 2. Working process of BCT: (a) for reader, (b) for tags.

Fig. 3(a) gives an example of the identifying process of BCT when identifying the same six tags 0010010, 0110001, 0110110, 1001110, 1010101, and 1100110. In each node, the two group of tags answer the reader's same query in two sub-cycles: R0-cycle and R1-cycle, respectively by transmitting their remainder IDs if their IDs match the received prefix. If there has no collision in a sub-cycle, the tag can be identified by the reader, such as the tags in the sub-cycles with bold font in **Fig. 3(a)**. If collision occurred in a sub-cycle, which means that at least two tags answered the reader's query, the reader generates a new prefix and pushes it into the stack, such as the sub-cycles: R1 and R0 in node 2 and 4, respectively. **Fig. 3(b)** is the degenerate collision tree, which will be discussed later in this section, depicting the identifying processing of BCT.

The identification process and communication between the reader and tags in this example are listed in **Table 3**. In each cycle of BCT, the reader sends the query command with a prefix, and two group of tags answer the same query in different sub-cycles respectively. In each collided sub-cycle, the reader only generates one new prefix and pushes it in the stack. At the same time, in BCT, the reader need not to deal with the collided bit directly and assign the value to the collided bit to generate two new prefixes like that in the other anti-collision protocols.

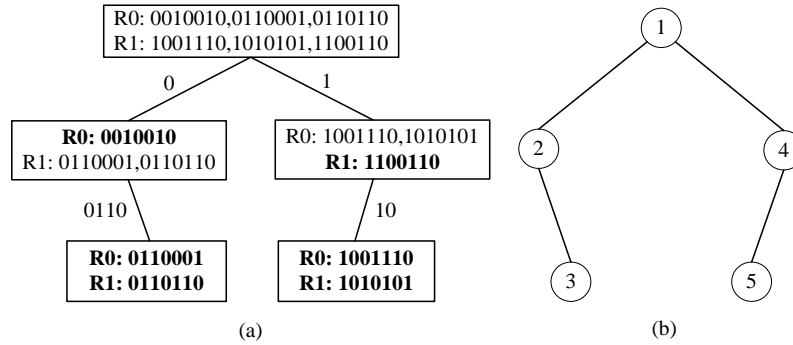


Fig. 3. Example of BCT identifying tags 0010010, 0110001, 0110110, 1001110, 1010101, and 1100110: (a) identification process, (b) degenerate collision tree. Here, R0 and R1 denote R0-cycle and R1-cycle respectively.

Table 3. The identification process of BCT used to identify tags:

0010010, 0110001, 0110110, 1001110, 1010101, and 1100110.

Cycle	Prefix	ReceivedID	Channel State	New Prefix	Prefix Stack
1	ϵ	0x10xxx 1xxx1xx	collided collided	0 1	0 0,1
2	0	010010 110xxx	tagID:0010010 collided	--- 0110	1 0110,1
3	0110	001 110	tagID:0110001 tagID:0110110	--- ---	1 1
4	1	0xx1xx 100110	collided tagID:1100110	10 ---	10 10
5	10	01110 10101	tagID:1001110 tagID:1010101	--- ---	NULL NULL

Here, ϵ represents empty string, --- means no new prefix is generated, NULL denotes the stack is empty.

3.3 Degenerate Collision Tree

Comparing Fig. 1, Fig. 3, Table 2, and Table 3, it can be found that each query-cycle in BCT is corresponding to three query-cycles in CT. For instance, as listed in Table 3 and Table 2, cycle 1 in BCT is corresponding to cycles 1, 2, 7 in CT, cycle 2 in BCT is corresponding to cycles 2, 3, 4 in CT, cycle 3 to cycles 4, 5, 6, cycle 4 to cycles 7, 8, 11, and cycle 5 to cycles 8, 9, 10, respectively.

Accordingly, each node in the degenerate collision tree of BCT is also corresponding to three nodes in the collision tree of CT, such as node 1 in the degenerate collision tree is corresponding to nodes 1, 2, 7 in the collision tree, node 2 is corresponding to nodes 2, 3, 4, and so on, as illustrated in Fig. 1 and Fig. 3.

This is because that the responses in both children nodes in CT are converged to their parent node when BCT is adopted. Therefore, the degenerate collision tree is consisted with the internal nodes (parent nodes) of the corresponding collision tree, as shown in Fig. 1(b) and Fig. 3(b). And this is why it is named degenerate collision tree.

Moreover, degenerate collision tree is no longer a full binary tree (collision tree is always a full binary tree). But the characteristics of degenerate collision tree can be deduced according to the relationships between it and the collision tree. This paper only discusses the number of nodes in a degenerate collision tree, which will be used to analyze the performance of BCT in the next section.

Assuming n , n_i , and n_c are the number of leaf nodes, internal nodes, and total nodes in a collision tree, according to the definition and characteristics of collision tree defined and deduced in [25][26], the total number of nodes in collision tree is

$$n_c = 2n - 1 \quad (1)$$

and

$$n_i = n - 1. \quad (2)$$

Here, n is the number of tags identified by CT.

Because degenerate collision tree is consisted with the internal nodes of the corresponding collision tree as analyzed above, the number of nodes in a degenerate collision tree is equal to that of the internal nodes in the corresponding collision tree. Therefore, the total number of nodes in the degenerate collision tree, say n_d , is

$$n_d = n_i = n - 1. \quad (3)$$

Here, n is the number of tags identified by BCT.

4. Performance Analysis of BCT

This section analyzes the performance of BCT including time complexity, identification efficiency, communication complexity, and identification speed based on the characteristic of the degenerate collision tree.

4.1 Time Complexity of BCT

The time complexity of anti-collision protocol is the number of cycles required by the reader for all tags identification. Tag identifying process is coincident with a tree search starting at the root of the tree to finding readable nodes in the leaf nodes of the tree. Hence, the time complexity of BCT is equal to the number of nodes in the degenerate collision tree.

Therefore, from equation (3), the time complexity of BCT is

$$T(n) = n_d = n - 1. \quad (4)$$

And the average time complexity (the average required cycles for one-tag identification) of BCT is

$$T_{avg}(n) = T(n) / n = 1 - 1/n. \quad (5)$$

Here, $n > 1$. If $n = 1$, $T(n) = 1$ and $T_{avg}(n) = 1$.

Furthermore,

$$\lim_{n \rightarrow \infty} T_{avg}(n) = 1. \quad (6)$$

4.2 Identification Efficiency of BCT

The identification efficiency of anti-collision protocol is the ratio between the number of identified tags and the number of query cycles used to identify them. Let $E(n)$ be the identification efficiency of BCT, then

$$E(n) = n / T(n) = n / (n - 1). \quad (7)$$

Moreover,

$$\lim_{n \rightarrow \infty} E(n) = 100\% \quad (8)$$

and

$$E(n) > 100\%, \quad (9)$$

for n is a integer and $n > 1$.

4.3 Communication Complexity of BCT

The communication complexity is the number of bits transmitted by the reader and the tags during RFID tag identification, which represents the energy cost during the communication process. The communication complexity could be classified into two parts: one is the reader communication complexity which is the bits transmitted by the reader; the other one is the tag communication complexity which is the bits transmitted by the tags.

Assuming $C(n)$, $C_R(n)$, $C_T(n)$ are the communication complexity, the reader communication complexity, and the tag communication complexity of BCT respectively. They satisfy

$$\begin{aligned} C(n) &= C_R(n) + C_T(n) \\ &= \sum_{i=1}^{T(n)} (l_{com} + l_{pre.i}) + 2 * \sum_{i=1}^{T(n)} l_{rep.i} \end{aligned} \quad (10)$$

Here, $T(n)$ is the time complexity of BCT, l_{com} is the length of the query command send by the reader, $l_{pre.i}$ is the prefix transmitted by the reader in cycle i , and $l_{rep.i}$ is the responding bits transmitted by the tags in cycle i .

Though $l_{pre.i}$ and $l_{rep.i}$ are changed in different query-response cycle, they are exactly the two different parts of the tag ID. Hence,

$$l_{ID} = l_{pre.i} + l_{rep.i}, \quad (11)$$

here, l_{ID} is the length of tag ID.

Therefore, equation (10) can be rewritten as

$$\begin{aligned} C(n) &= \sum_{i=1}^{T(n)} (l_{com} + l_{pre.i} + l_{rep.i}) + \sum_{i=1}^{T(n)} l_{rep.i} \\ &= \sum_{i=1}^{T(n)} (l_{com} + l_{ID}) + \sum_{i=1}^{T(n)} l_{rep.i} \\ &= (n-1)(l_{com} + l_{ID} + \overline{l_{rep}}) \end{aligned} \quad (12)$$

here, $\overline{l_{rep}}$ is the average value of $l_{rep.i}$.

And so, the average communication complexity of BCT is

$$C_{avg}(n) = C(n) / n = \frac{n-1}{n} (l_{com} + l_{ID} + \overline{l_{rep}}). \quad (13)$$

According to statistical probability, $\overline{l_{rep}}$ could be replaced with $1/2 l_{ID}$. And so,

$$C_{avg}(n) = C(n) / n = \frac{n-1}{n} (l_{com} + 3/2 l_{ID}), \quad (14)$$

and

$$\begin{aligned} \lim_{n \rightarrow \infty} C_{avg}(n) &= \frac{n-1}{n} (l_{com} + 3/2 l_{ID}) \\ &= l_{com} + 3/2 l_{ID} \end{aligned} \quad (15)$$

4.4 Identification Speed of BCT

Generally, in traditional anti-collision protocols, the identification cycle is composed of two

sub-cycles: Q-cycle and R-cycle. In Q-cycle, the reader transmits query command with a prefix to inquire the tags. In R-cycle, the tags answer the reader's query by transmitting their IDs. For convenient description and comparison, CT is selected as an example in this part because the cycles definition of QT and many other protocols are similar to those of CT. In BCT, the identification cycle is composed of three sub-cycles: Q-cycle, R0-cycle, and R1-cycle. Q-cycle is same with that of CT. In R0-cycle, the tags whose characteristic bit equals 0 answer the reader's query. In R1-cycle, the tags whose characteristic bit equals 1 answer the reader's query.

Assuming the length of Q-cycle, R-cycle, R0-cycle, and R1-cycle are represented by L_Q , L_R , L_{R0} , and L_{R1} respective, we have

$$L_R = L_{R0} = L_{R1}. \quad (16)$$

And the length of the identification cycle in CT and BCT, say L_{CT} and L_{BCT} , are

$$L_{CT} = L_Q + L_R \quad (17)$$

and

$$\begin{aligned} L_{BCT} &= L_Q + L_{R0} + L_{R1} \\ &= L_Q + 2L_R \end{aligned} \quad (18)$$

respectively.

In RFID tag identification, the identification speed is the average number of tags identified per second by the identification protocol. So the identification speed of BCT, say N_{BCT} , could be written as

$$\begin{aligned} N_{BCT} &= 1/(T_{avg}(n) * L_{BCT}) \\ &= n/((n-1) * (L_Q + 2L_R)) \end{aligned} \quad (19)$$

here, $T_{avg}(n)$ is the average time complexity of BCT as shown in equations (5).

At the same time, from (19), the limitation of N_{BCT} is

$$\lim_{n \rightarrow \infty} N_{BCT} = 1/(L_Q + 2L_R). \quad (20)$$

4.5 Stability of BCT

As defined in [26], the stability of RFID tag identification protocol is the characteristic that the identification performance of the identification protocol is only decided by the number of tags to be identified, and the limitation of the average identification performance converges to a constant.

If an anti-collision protocol for RFID tag identification is a stable protocol, many technical indicators or parameters could be estimated and controlled before tags (objects) are identified. This is very important for many RFID application systems, such as stream-line production, auto-control system, real-time data collection, and so on.

From equations (4) and (5), the time complexity and the average time complexity of BCT are only dependent on the number of tags n ; and from (6), the average time complexity of BCT converges to 1, a constant. So, according to the definition of stability, the time complexity of BCT is stable.

According to equations (7) and (8), the identification efficiency of BCT is only dependent on the number of tags also, and converges to 100%. That is the identification efficiency of BCT is stable. Furthermore, the identification efficiency of BCT is always above 100%, as

shown in equation (9).

The communication complexity of BCT is dependent on the number of tags, and affects by l_{com} , l_{ID} , and $\overline{l_{rep}}$ as depicted in equations (12) and (13). But in a specific RFID system, both l_{com} and l_{ID} are constants. Though $\overline{l_{rep}}$ is affected by the location of collision during tags identification and the distribution of tags IDs, $\overline{l_{rep}}$ approaches to $\frac{1}{2}l_{ID}$ according to the statistical probability. And so, the limitation of the average communication complexity of BCT converges to a constant also, as shown in equation (15). Therefore, the communication complexity of BCT is stable.

As indicated in equations (19) and (20), the identification speed of BCT is stable also. This is because that the identification speed of BCT is only dependent on the number of tags n , and the limitation of identification speed of BCT is a constants, for both L_Q and L_R are constants in a specific RFID system.

5. Simulation and Results

The performance of Collision Tree Protocol (CT) has been discussed in [25][26] compared to some classical anti-collision protocols such as QT, BT, BS, DBS, FSA, and DFSA. This section compares the performance of BCT to some new and similar tag identification protocols including that of Collision Tree Protocol (CT), M-ray Collision Tree Protocol (MCT) ($m=4$ in this paper), and Dual Prefix Probe Scheme (DPPS).

First of all, because there are 1, 2, 2, and 4 tags' response cycles or slots in each identification cycle of CT, BCT, DPPS, and MCT respectively, this section uses the normalized time complexity and normalized identification efficiency to compare and analyze the performance of them. The normalized time complexity of anti-collision protocol is the average response cycles required by the tags for one-tag identification. The normalized identification efficiency of anti-collision protocol is the ratio between the number of tags and the number of tags' response cycles during the tags identification. For example, if one query cycle in an anti-collision protocol is corresponding to k response cycles, the average time complexity and identification efficiency of the protocol are normalized to $kT_{avg}(n)$ and $E(n)/k$. This normalization is just to make it easier to compare and understand, and not affect the tags identification process and the actual performance of the protocols.

5.1 Simulation Settings

For keeping the continuity of the research and making it easy to compare and understand the related protocols, this paper continues the similar simulation environment of that in papers [25][26].

Two necessary commands are used in the simulations. One is Query command (22 bits), which is used during RFID tags identification to start a query and transmit the parameters (like prefix or slots number) to the tags. The other one is ACK command (18 bits), which is used after a tag identified successfully to inform the identified tag to sleep and not to answer the following command. In actual RFID systems, the ACK command could be replaced by other commands, such as Read, Write, according to the requirement of the systems.

Two scenarios, i.e., scenario S1 and scenario S2, are established in the simulations according to the distribution of tags IDs. In both scenarios, the tags IDs are composed with 96 binary bits and the number of tags increase from 8 to 4096.

Scenario S1: uniform distribution, all tags IDs are generated randomly, and each ID have a unique value, i.e., none two tags have the same ID.

Scenario S2: continuous distribution, all tag ID is generated by increasing 1 on the former one except the first one which is generated randomly.

At the same time, this paper assumes that the communication process between the reader and tags is based on ISO 18000-6 specification [17][28][39] as shown in Fig. 4. Therefore, the reader's query cycles (Q-cycle) last $800\mu\text{s}$, and the tags' response cycles (R-cycle, R0-cycle, and R1-cycle) last $3100\mu\text{s}$.

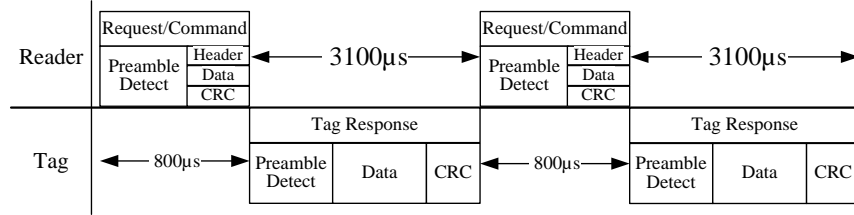


Fig. 4. the communication process between the reader and tags. [39]

5.2 Simulation Results

The simulation results of CT, BCT, MCT, and DPPS in scenario S1 and S2 are illustrated in Fig. 5, Fig. 6 and Fig. 7. Fig. 5 includes the average time complexity, identification efficiency, and identification speed of them. Fig. 6 illustrates the communication complexity and energy consumption of them. Fig. 7 shows the effect of idle cycle on the identification performance of anti-collision protocol: MCT and DPPS. It is worth to note that CT and BCT are not presented in Fig. 7 because both of them have eliminated the idle cycle during RFID tag identification, i.e., no idle cycle exist in them.

From Fig. 5(a), CT and BCT require less than 2 response cycles for one-tag identification in both scenario S1 and S2. And MCT requires average 2.39 and 2.13 response cycles in scenario S1 and S2 respectively, while average 2.36 and 2.20 response cycles are required by DPPS to identify one tag. This is because BCT and CT eliminate the idle cycle during tags identification, but MCT and DPPS increase the idle cycles in this process.

Accordingly, the identification efficiency of CT and BCT are always above 50% , while that of MCT and DPPS are between 40% and 47% as shown in Fig. 5(b). At the same time, the performance of MCT and DPPS in scenario S1 are lower than that in scenario S2, which means that much more idle cycles are appeared when tags IDs distribute uniformly (scenario S1), and the identification efficiency of MCT and DPPS are affected by the distribution of tags IDs.

Fig. 5(c) is the identification speed of CT, BCT, MCT and DPPS. Obviously, BCT is the fastest tags identification protocol in both scenario S1 and S2. MCT is the second faster one when tags IDs are distributed continuously (scenario S2) because more query cycles are reduced by MCT in this situation. The identification speed of MCT and DPPS in scenario S1 are even lower than that of CT because they increased many idle cycles during tags identification.

Fig. 5(d) shows the relative identification speed (%) of BCT, MCT and DPPS compared to that of CT in scenario S1 and S2. In both scenario S1 and S2, the identification speed of BCT is up to 112% of that of CT. The identification speed of MCT in scenario S1 and S2 are 98% and 110% of that of CT respectively, while that of DPPS are 94% and 101% respectively.

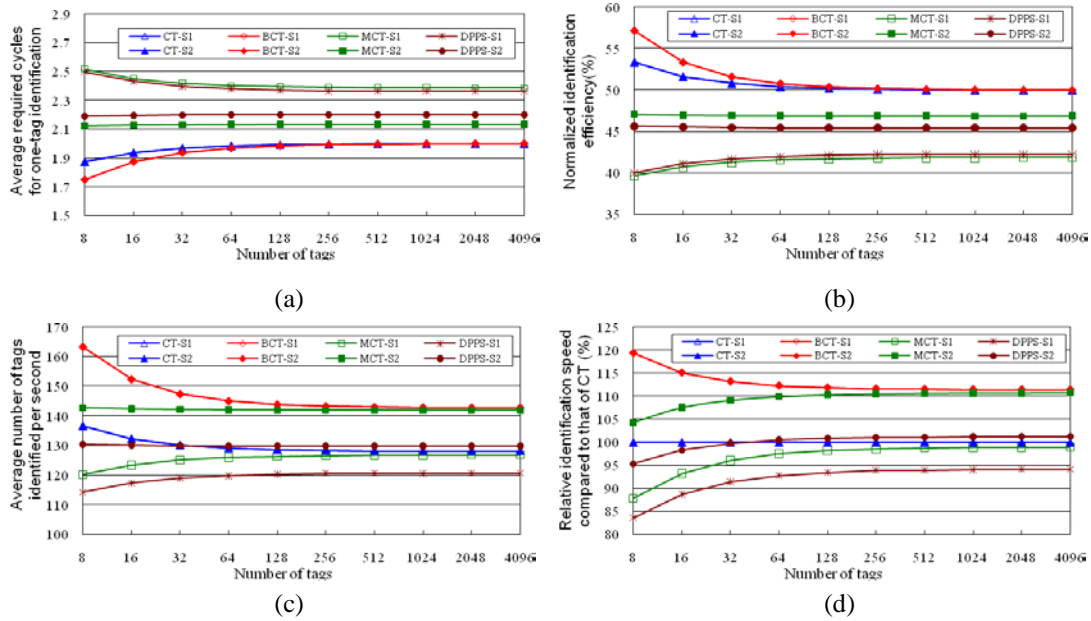


Fig. 5. Performance of CT, BCT, MCT, and DPPS in scenario S1 and S2: (a) average time complexity, (b) identification efficiency (%), (c) identification speed, (d) relative identification speed compared to that of CT (%).

Frankly, the communication complexity of CT is the worst one among these protocols in both scenario S1 and S2 as shown in [Fig. 6\(a\)](#), which transmitted average about 455 bits for one tag identification. MCT is the best one in both scenario S1 and S2, only about 370 bits and 285 bits are transmitted by the reader and tags. The communication complexity of BCT, MCT and DPPS in scenario S2 are better than that in scenario S1, because much fewer bits are transmitted by the tags of them in scenario S2 as shown in [Fig. 6\(c\)](#).

Because the prefixes in scenario S2 (tags IDs are distributed continuously) are much longer than those in scenario S1 (tags IDs are distributed uniformly), i.e., more bits are transmitted by the reader in scenario S2, the reader communication complexity of CT, BCT, MCT and DPPS in scenario S2 are higher than that in scenario S1 as indicated in [Fig. 6\(b\)](#). Contrarily, the tag communication complexity of CT, BCT, MCT and DPPS in scenario S2 are much lower than that in scenario S1 as shown in [Fig. 6\(c\)](#). At the same time, when tags IDs are distributed continuously (scenario S2), tag communication complexity of these protocols are close to each other.

The number of bits transmitted by the reader and tags represents the energy consumption of the RFID system for RFID tags identification [28]. The energy consumption of CT, BCT, MCT and DPPS compared to that of CT are illustrated in [Fig. 6\(d\)](#). From the figure, the power consumption of BCT, MCT and DPPS are all lower than that of CT. MCT is the best protocol in energy consumption in both scenario S1 and S2, which needs fewer energy than other protocols particularly in scenario S2. The energy consumption of MCT are 84.5% and 64.7% of that of CT in scenario S1 and S2 respectively, while that of BCT are 93.2% and 74.3% of that of CT.

Although MCT is the best protocol in communication complexity and energy consumption, the identification efficiency and identification speed of it are not as good as its expectation as illustrated in [Fig. 5](#). Only when tags IDs are distributed continuously, the identification speed of MCT is close to that of BCT. This is because MCT increases idle cycles during tags

identification, while decreasing the reader's queries. Essentially, it cannot ensure that all branches always have tags to respond the reader's query. The same things happened on DPPS also.

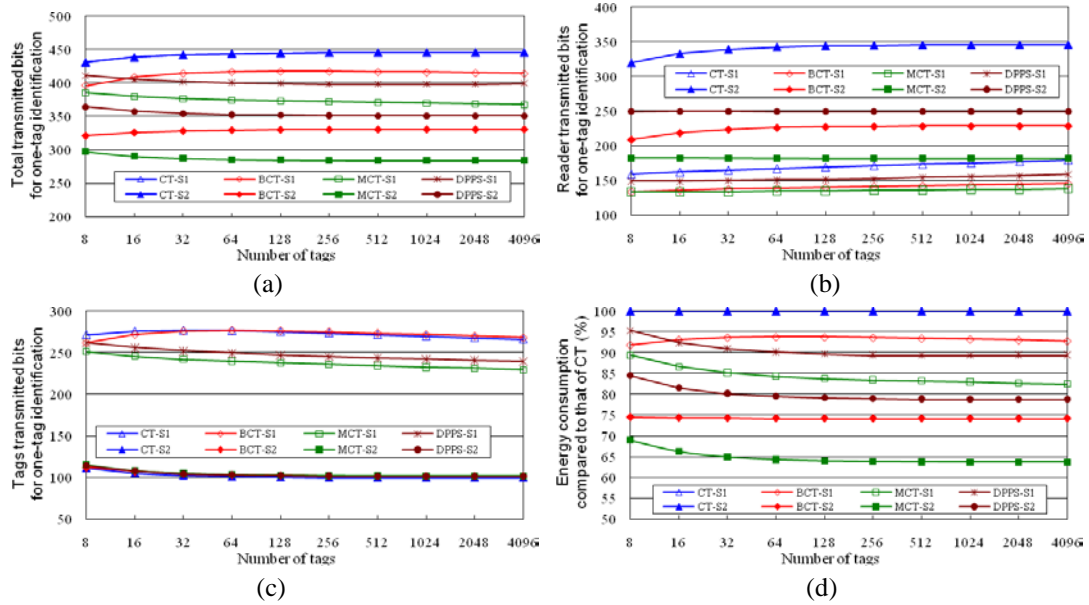
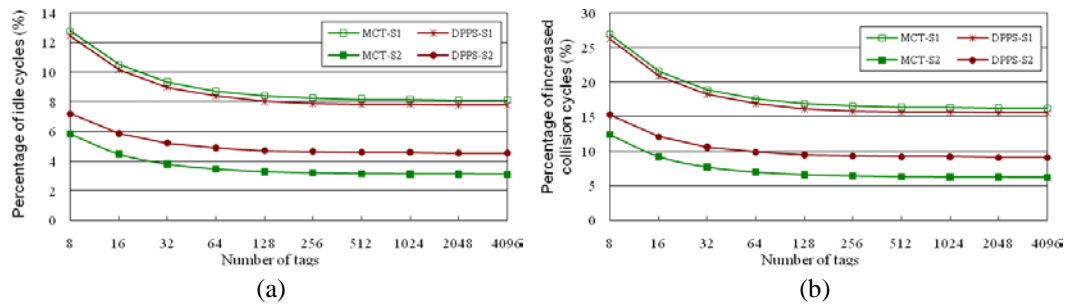


Fig. 6. Performance of CT, BCT, MCT and DPPS in scenario S1 and S2: (a) communication complexity, (b) reader communication complexity, (c) tag communication complexity, (d) energy consumption compared to CT (%).

Fig. 7 depicts the idle cycle and its impact on the performance of MCT and DPPS. Above 8.1% and 7.7% cycles in MCT and DPPS are idle cycle in scenario S1, and above 3.2% and 4.5% cycles are idle cycle in scenario S2 as shown in **Fig. 7(a)**. Due to the idle cycles, MCT increases collisions 16.4% and 6.3% in scenario S1 and S2, and DPPS increases collisions 15.2% and 9.6% as illustrated in **Fig. 7(b)**. Accordingly, the identification efficiencies of MCT are decreased 19.5% and 6.7% in scenario S1 and S2 respectively, and those of DPPS are decreased 18.5% and 10.2% as shown in **Fig. 7(c)**.

On that account, the idle cycle in anti-collision protocol is the most important factor except for tags splitting method, affecting the performance of RFID tags identification, increasing more collision, transmitted bits and power consumption, and decreasing the efficiency, speed of tags identification. The performance of BCT is so better thanks to it eliminates the idle cycle during tags identification.



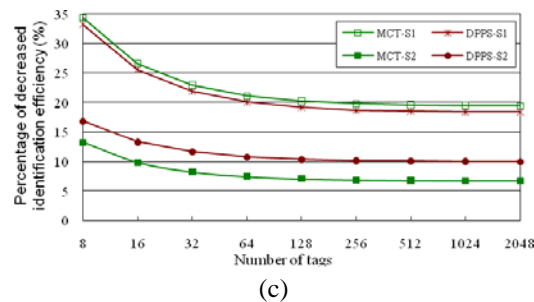


Fig. 7. Impact of the idle cycle on the performance of MCT, and DPPS in scenario S1 and S2: (a) percentage of idle cycles, (b) percentage of increased collision cycles, (c) percentage of decreased identification efficiency.

6. Conclusion

RFID is the cornerstone of IOT, and will be applied increasingly in IOT enabled applications e.g., inventory item tracking and supply chain management. But collision occurred when multiple tags communicate with reader simultaneously. This paper presents a new response pattern for RFID tags identification called bi-response pattern, and an anti-collision protocol, i.e., BCT, to solve the tag collision during RFID tags identification, which increases the tag identification speed significantly while decreases the time complexity and communication complexity effectively, and gives a new foundation for the researching new RFID tags identification protocol, such as parallel tags identification protocol. For the response of the tags in BCT is only decided by the received prefix and not dependent on the historical queries of the reader, BCT belongs to the class of memoryless anti-collision protocol. BCT can be used in passive RFID system to solve tag collision problem. Furthermore, as a simple anti-collision protocol with high performance, BCT is very suitable to be applied in auto-control system, assembly line production, and massive data collection, logistics and warehouse management, and especially used for large-scale passive RFID tags identification.

References

- [1] A.X. Liu, M. Shahzad, X.L. Liu, K. Li, *RFID Protocol Design, Optimization, and Security for the Internet of Things*, Wiley, 2017. [Article \(CrossRef Link\)](#).
- [2] I. Farris, S. Pizzi, M. Merenda, A. Molinaro, R. Carotenuto, A. Iera, "6lo-RFID: A Framework for Full Integration of Smart UHF RFID Tags into the Internet of Things," *IEEE Network*, vol.31, no.5, pp.66-73, Sep. 2017. [Article \(CrossRef Link\)](#).
- [3] M. H. Afifi, L. Zhou, S. Chakrabarty, J. Ren, "Dynamic Authentication Protocol Using Self-powered Timers for Passive Internet of Things," *IEEE Internet of Things Journal*, vol.5, no.4, pp.2927-2935, Aug. 2018. [Article \(CrossRef Link\)](#).
- [4] X.L. Jia, Q.Y. Feng, "An Improved Anti-collision Protocol for Radio Frequency Identification Tag," *International Journal of Communication Systems*, vol.28, no.3, pp.401-413, Feb. 2015. [Article \(CrossRef Link\)](#).
- [5] K. Finkenzerler, *RFID handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication*, Wiley, New York, pp.189-212, 2010. [Article \(CrossRef Link\)](#).
- [6] D.K. Klair, K.W. Chin, R. Raad, "A Survey and Tutorial of RFID Anti-collision Protocols," *IEEE Communications Surveys and Tutorials*, vol.12, no.3, pp.400-421, 2010. [Article \(CrossRef Link\)](#).

- [7] C. Law, K. Lee, K.Y. Siu, "Efficient Memoryless Protocol for Tag Identification," in *Proc. of Fourth International workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM'00)*, New York, NY, USA, pp.75–84, 2000. [Article \(CrossRef Link\)](#).
- [8] X.L. Jia, Y.J. Gu, Q.Y. Feng, L.M. Zhu, L. Yue, "Standards and Protocols for RFID Tag Identification," *Electronics, Communications and Networks IV*, CRC Press, London, pp.851-855, July. 2015.
- [9] Alien Technology, "EPCglobal Class 1 Gen 2 RFID Specifications," *Whitepaper*, 2005. [Article \(CrossRef Link\)](#).
- [10] Y. Maguire, R. Pappu, "An optimal Q-algorithm for the ISO 18000-6C RFID protocol," *IEEE Transactions on Automation Science and Engineering*, vol.6, no.1, pp.16-24, Jan. 2009. [Article \(CrossRef Link\)](#).
- [11] K. Fyhn, R.M. Jacobsen, P. Popovski, "Multi-packet Reception of Passive UHF RFID Tags: A Communication Theoretic Approach," *IEEE Transactions on Signal Processing*, vol.59, no.9, pp.4225-4237, Sep. 2011. [Article \(CrossRef Link\)](#).
- [12] X.L. Liu, K.Q. Li, G.Y. Min, Y.M. Shen, A.X. Liu, W.Y. Qu, "A Multiple Hashing Approach to Complete Identification of Missing RFID Tags," *IEEE Transactions On Communications*, vol.62, no.3, pp.1046-1057, Mar. 2014. [Article \(CrossRef Link\)](#).
- [13] L.J. Zhang, J. Zhang, X.H. Tang, "Assigned Tree Allotted Aloha RFID Tag Anti-collision Protocols," *IEEE Transactions on Wireless Communications*. vol.12, no.11, pp.5493-5505, Nov. 2013. [Article \(CrossRef Link\)](#).
- [14] Y.H. Chen, Q.Y. Feng, Z. Ma, T. Liu, "Multiple-bits-slot Reservation Aloha Protocol for Tag Identification," *IEEE Transactions on Consumer Electronics*, vol. 59, no. 1, pp.93-100, Feb. 2013. [Article \(CrossRef Link\)](#).
- [15] W.T. Chen, "Optimal Frame Length Analysis and an Efficient Anti-Collision Algorithm With Early Adjustment of Frame Length for RFID Systems," *IEEE Transactions on Vehicular Technology*, vol.65, no.5, pp.3342-3348, May 2016. [Article \(CrossRef Link\)](#).
- [16] Y.R. Chen; J.Su, W.D. Yi, "An Efficient and Easy-to-Implement Tag Identification Algorithm for UHF RFID Systems," *IEEE Communications Letters*, vol.21, no.7, pp.1509-1512, Jul. 2017. [Article \(CrossRef Link\)](#).
- [17] J. Su, Zh.G. Sheng, D.F. Hong, G.J. Wen, "An Effective Frame Breaking Policy for Dynamic Framed Slotted Aloha in RFID," *IEEE Communications Letters*, vol.20, no.4, pp.692-695, Apr. 2016. [Article \(CrossRef Link\)](#).
- [18] J. Su, Zh.G. Sheng, L.B. Xie, "A Collision-Tolerant-Based Anti-Collision Algorithm for Large Scale RFID System," *IEEE Communications Letters*, vol.21, no.7, pp. 1517-1520, Jul. 2017. [Article \(CrossRef Link\)](#).
- [19] Y.Ch. Lai, L.Y. Hsiao, B.Sh. Lin, "Optimal Slot Assignment for Binary Tracking Tree Protocol in RFID Tag Identification," *IEEE/ACM Transactions on Networking*, vol.23, no.1, pp.255-268, Feb. 2015. [Article \(CrossRef Link\)](#).
- [20] H. Landaluce, A. Perallos, E. Onieva, L. Arjona, L. Bengtsson, "An Energy and Identification Time Decreasing Procedure for Memoryless RFID Tag Anticollision Protocols," *IEEE Transactions on Wireless Communications*, vol.15, no.6, pp.4234-4247, June. 2016. [Article \(CrossRef Link\)](#).
- [21] J. Su, Z. Sheng, L. Xie, G. Li, A. Liu, "Fast Splitting Based Tag Identification Algorithm for Anti-collision in UHF RFID System," *IEEE Transactions on Communications*, vol.67, no.3, pp.2527-2538, Mar. 2019. [Article \(CrossRef Link\)](#).
- [22] J. Su, Z. Sheng, V. C.M. Leung, Y. Chen, "Energy Efficient Tag Identification Algorithms for RFID: Survey, Motivation and New Design," *IEEE Wireless Communications*, vol.26, no.3, pp. 118-124, June, 2019. [Article \(CrossRef Link\)](#).
- [23] J. Su, Z. Sheng, A. Liu, Y. Han, and Y. Chen, "A Group-based Binary Splitting Algorithm for UHF RFID Anti-collision Systems," *IEEE Transactions on Communications*, vol.68, no.2, pp.998-1012, 2020. [Article \(CrossRef Link\)](#). [on line]
- [24] J. Su, Z. Sheng, A. Liu, Z. Fu, and Y. Chen, "A Time and Anergy Saving Based Frame Adjustment Strategy (TES-FAS) Tag Identification Algorithm for UHF RFID Systems," *IEEE Transactions on Wireless Communications*, pp.1-1, 2020. [Article \(CrossRef Link\)](#). [on line]

- [25] X.L. Jia, Q.Y. Feng, C.Z. Ma, "An Efficient Anti-collision Protocol for RFID Tag Identification," *IEEE Communications Letters*, vol.14, no.11, pp. 1014-1016, Nov. 2010. [Article \(CrossRef Link\)](#).
- [26] X.L. Jia, Q.Y. Feng, L.Sh. Yu, "Stability Analysis of an Efficient Anti-collision Protocol for RFID Tag Identification," *IEEE Transactions on Communications*, vol.60, no.8, pp.2285-2294, Aug. 2012. [Article \(CrossRef Link\)](#).
- [27] J. Su, Z.G. Sheng, G.J. Wen, V.C.M. Leung, "A Time Efficient Tag Identification Algorithm Using Dual Prefix Probe Scheme (DPPS)," *IEEE Signal Processing Letters*, vol.23, no.3, pp.386-389, Mar. 2016. [Article \(CrossRef Link\)](#).
- [28] L.J. Zhang, W. Xiang, X.H. Tang, Q. Li, Q.F. Yan, "A Time- and Energy-aware Collision Tree Protocol for Efficient Large-scale RFID Tag Identification," *IEEE Transactions on Industrial Informatics*, vol.14, no.6, pp. 2406-2417, June 2018. [Article \(CrossRef Link\)](#).
- [29] M. Shahzad, A.X. Liu, "Probabilistic Optimal Tree Hopping for RFID Identification," *IEEE/ACM Transactions on Networking*, vol.23, no.3, pp.796–809, June 2015. [Article \(CrossRef Link\)](#).
- [30] X. Liu, J. Cao, Y. Yang, W. Qu, X. Zhao, K. Li, D. Yao, "Fast RFID Sensory Data Collection: Trade-off Between Computation and Communication Costs," *IEEE/ACM Transactions on Networking*, vol.27, no.3, pp.1179–1191, June 2019. [Article \(CrossRef Link\)](#).
- [31] X. Liu, X. Xie, S. Wang, J. Liu, D. Yao, J. Cao, K. Li, "Efficient Range Queries for Large Scale Sensor-Augmented RFID Systems," *IEEE/ACM Transactions on Networking*, vol. 27, no. 5, pp. 1873–1886, Oct. 2019. [Article \(CrossRef Link\)](#).
- [32] M. Shahzad, A.X. Liu, "Fast and Reliable Detection and Identification of Missing RFID Tags in the Wild," *IEEE/ACM Transactions on Networking*, vol.24, no.6, pp.3770-3784, Aug. 2016. [Article \(CrossRef Link\)](#).
- [33] J. Yu, L. Chen, R. Zhang, K. Wang, "On Missing Tag Detection in Multiple-Group Multiple-Region RFID Systems," *IEEE Transactions on Mobile Computing*, vol.16, no.5, pp.1371-1381, May 2017. [Article \(CrossRef Link\)](#).
- [34] J. Yu, L. Chen, R. Zhang, K. Wang, "Finding Needles in a Haystack: Missing Tag Detection in Large RFID Systems," *IEEE Transactions on Communications*, vol.65, no.5, pp.2-36-2047, May 2017. [Article \(CrossRef Link\)](#).
- [35] H. Chen, G. Ma, Z. Wang, F. Xia, J. Yu, "Probabilistic Detection of Missing Tags for Anonymous Multicategory RFID Systems," *IEEE Transactions on Vehicular Technology*, vol.66, no.12, pp.11295-11305, Dec. 2017. [Article \(CrossRef Link\)](#).
- [36] J. Yu, W. Gong, J. Liu, L. Chen, K. Wang, R. Zhang, "Missing Tag Identification in COTS RFID Systems: Bridging the Gap between Theory and Practice," *IEEE Transactions on Mobile Computing*, vol.19, no.1, pp.130-141, Jan. 2020. [Article \(CrossRef Link\)](#).
- [37] J. Yu, W. Gong, J. Liu, L. Chen, K. Wang, "On Efficient Tree-Based Tag Search in Large-Scale RFID Systems," *IEEE/ACM Transactions on Networking*, vol.27, no.1, pp.42-55, Feb.2019. [Article \(CrossRef Link\)](#).
- [38] M. Chen, W. Luo, Z. Mo, S. Chen, Y. Fang, "An Efficient Tag Search Protocol in Large-Scale RFID Systems With Noisy Channel," *IEEE/ACM Transactions on Networking*, vol.24, no.2, pp.703-716, April 2016. [Article \(CrossRef Link\)](#).
- [39] X.L. Jia, M. Bolic, Y.H. Feng, Y.J. Gu, "An Efficient Dynamic Anti-Collision Protocol for Mobile RFID Tags Identification," *IEEE Communications Letters*, vol.23, no.4, pp. 620-623, April 2019. [Article \(CrossRef Link\)](#).



Xiaolin Jia: male, PhD, associate professor, IEEE member, CCF Senior Member, works in the School of Computer Science and Technology of Southwest University of Science and Technology (SWUST), Mianyang, China. He has been honored as the "Leading Talent of High-Tech Industry and Strategic Emerging Industry" and "Excellent Teacher" of Mianyang Government, and the "Top Ten Outstanding Youth" and "Excellent Teacher" of SWUST. He has been appointed as a member of the IEEE Communications Society Reviewers. And now, he is the reviewer of IEEE Transactions on Communications, IEEE Transactions on Wireless Communications, IEEE Transactions on Signal Processing, IEEE Transactions on Industrial Informatics, KSII Transactions on Internet and Information Systems, Transactions on Mobile Computing, IEEE Communications Letters, IEEE Signal Processing Letters, etc. His research interests include RFID technology, Internet of Things (IOT), and Computer Engineering. He can be contacted by email: my_jiaxl@163.com.



Yuhao Feng: male, is a senior undergraduate student of the School of Social & Behavior Science, University of Arizona, USA. His research interests include RFID technology, embedded system, Computer Science and large data sets etc. He can be contacted by email: yuhaofeng@email.arizona.edu.



Yajun Gu: female, lecturer, works in the School of Computer Science and Technology of Southwest University of Science and Technology (SWUST), Mianyang, China. She received the M.S. degree in Computer Science from Southwest University of Science and Technology (SWUST) in 2011. Her research interests include embedded system, computer simulation, and Internet of Things (IOT). She can be contacted by email: guyajun@swust.edu.cn.