Season: Shelving Interference and Joint Identification in Large-scale RFID Systems

Lei Yang*, Jinsong Han*, Yong Qi*, Cheng Wang§, Tao Gu¶, Yunhao Liu†‡
* Department of Computer Science and Technology, Xi'an Jiaotong University, China
† TNLIST, School of Software, Tsinghua University, China
‡ Department of Computer Science and Engineering, HKUST, Hong Kong
§ Tongji University
¶ University of Southern Denmark

***Abstract*—Prior work on anti-collision for Radio Frequency IDentification (RFID) systems usually schedule adjacent readers to exclusively interrogate tags for avoiding reader collisions. Although such a pattern can effectively deal with collisions, the lack of readers' collaboration wastes numerous time on the scheduling process and dramatically degrades the throughput of identification. Even worse, the tags within the overlapped interrogation regions of adjacent readers (termed as contentious tags), even if the number of such tags is very small, introduce a significant delay to the identification process. In this paper, we propose a new strategy for collision resolution. First, we shelve the collisions and identify the tags that do not involve reader collisions. Second, we perform a joint identification, in which adjacent readers collaboratively identify the contentious tags. In particular, we find that neighboring readers can cause a new type of collisions, cross-tag-collision, which may impede the joint identification. We propose a protocol stack, named Season, to undertake the tasks in two phases and solve the cross-tag-collision. We conduct extensive simulations and preliminary implementation to demonstrate the efficiency of our scheme. The results show that our scheme can achieve above 6 times improvement on the identification throughput in a large-scale dense reader environment.**

***Index Terms*—RFID, Tag Collision, Reader Collision, Season**

I. Introduction

Radio Frequency Identification systems have been deployed in a variety of application domains, such as logistic and supply chain management [1], access control [2], theft detection [3], and tracking [4]–[8], *etc*. An RFID system typically consists of a large number of readers and tags. RFID tags are attached to products and targeted to enable the identification of those objects. Tags usually have no energy and can only be activated when they are within the electromagnetic field of a reader. The reader interrogates the tags and collects their IDs via RF signals, without the need of keeping in sight or touch. In contrast to the conventional barcode system, RFID systems have many advantages, such as non-optical proximity, long transmission range, and quick identification. Therefore, the promising RFID technology is expected to be widely used in the near feature.

The signal collision is one of the most challenging issues when implementing the RFID technology. There are three types of RFID signal collisions. The first type of collision occurs when more than one tag responds simultaneously. In this situation, the signals coming from multiple tags may interfere with each other and prevent the reader from resolving any tag's ID. We call the first collision as "tag collision", as shown in Fig. I(a). The second type of collision occurs in a multi-reader environment, as illustrated in Fig. I(b). In this example, reader r_1 and r_2 share an overlapped interrogation region (In this paper, we define such a region as 'contentious region', the tags within contentious regions as 'contentious tags', and other tags as 'non-contentious tags'). If there are some tags in this region, they cannot resolve the commands from r_1 or r_2 when two readers concurrently broadcast their commands. We call this type of collisions as 'reader collision'. The third type of collision is termed as reader-tag collision, which occurs when one reader is in another reader's interrogation region, as shown in Figure I(c), reader r_1 is located in r_2's interrogation region. Tag t_1's response will be 'drowned' by the commands from reader r_2, and resulting r_1 is unable to receive t_1's ID.

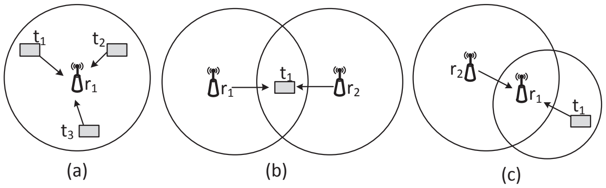


Fig. 1. Collisions in RFID systems. (a) Tag collision; (b) Reader collision; (c) Reader-Tag collision.

Clearly, avoiding collisions is a crucial task in RFID systems, especially when readers are densely deployed. The algorithms to resolve the aforementioned collisions are known as anti-tag-collision, anti-reader-collision, and anti-reader-tag collision algorithm, respectively. As a cost-effective and source-limited device, the RFID tag cannot afford the relatively complicated anti-collision algorithms adopted in traditional wireless networks, such as CSMA, CDMA, FDMA, *etc*. Existing RFID anti-collision algorithms mainly employ Time Dividing Multiple Accesses (TDMA), which allows tags and readers to send signals in different time slots. For example, Framed Slotted ALOHA (FSA) [9], [10], [26]–[28] , which is a dominant anti-tag-collision protocol, requires tags to respond in randomly chosen time slot.

Unfortunately, existing anti-collision works are inefficient to combat RFID signal collisions, especially the reader collision. They usually adopt an exclusive scheduling strategy to avoid reader collision [12]–[15]. Namely, neighboring readers that share some contentious regions must be activated in sequence. For instance, Colorwave [13], one of the most popular anti-reader-collision protocols, pre-schedules neighboring readers to work in different time slots. Those approaches may suffer from two drawbacks, low throughput and large identification delay. According to the well-known RFID standard ISO-18000 [10], the average identification throughput of Framed Slotted ALOHA protocols only archive 100 tags per second [11]. The exclusive scheduling among readers will further degrade the throughput. For example, one of experiments performed in a warehouse scenario indicates that the reader's throughput degrades to 52 tags per second on average due to the interferences among four neighboring readers, as shown in Section IV. As a result, it will spend almost half an hour to inventory 78,606 products. On the other hand, the identification delay of tags is an import metric in real-time RFID applications, such as the theft detection [3], object tracking [4], *etc*. Our experimental results show that Colorwave requires six exclusive rounds at least to schedule six mutually-interfered readers when identifying 1000 tags for each. In this case, the maximum delay introduced to each tag is up to 63 seconds. That means the moving speed of tags must be slower than $10cm$ per second in the readers' monitoring region where the range of the reader equals $3m$. Such a speed cannot well support fast identification in real-time RFID applications which have a rigid time limit on the processing speed.

By reconsidering the solution of reader-collision in another perspective, we find that it is not necessary to constrain neighboring readers in a strictly sequential processing pattern for the purpose of anti-collision. Usually, the majority of tags are non-contentious in common RFID applications. They can be concurrently identified by multiple readers because there is no reader collision in those tags. Hence, we propose to identify tags in two phases. In the first phase, we simply allow multiple readers to identify the non-contentious tags simultaneously, while shelving the reader collisions. In this way, the identification throughput of non-contentious tags will be significantly improved. In the second phase, we design efficient protocols to identify the contentious tags. We find that a reader, if it just passively monitors, can facilitate the identification responses from contentious tags that are interrogated by another reader. This observation motivates us to enable collaboration among neighboring readers to enormously reduce the identification delays of contentious tags.

In this paper, we propose a novel scheme, Season, to improve the efficiency for anti-collision based RFID identification. The Season protocol works in two phases. In the first phase, we propose the Season-I protocol in which all readers ignore the reader collisions and concurrently identify non-contentious tags. Season-I extends the existing anti-tag-collision algorithms by adaptively tuning the size of frame to improve the throughput of identification. In the second phase, neighboring readers jointly identify contentious tags. Different from existing approaches, our scheduling protocol, named Season-II, just selects only one reader from neighboring readers to perform the interrogation and let the others passively collect data from contentious tags. Thus, neighboring readers are able to collaborate with each other in the identification of contentious tags, and save vast time consumed in scheduling. Adopting joint identification, we find that the collaborative readers may face an emerging collision, termed as cross-range collision. We develop another anti-tag-collision protocol, Season-III, to combat the cross-range collision and achieve fast identification.

The rest of this paper is organized as follows. We introduce preliminary knowledge about RFID systems and the system model in Section II. We present the design of Season in Section III. In Section IV, we examine the performance of Season via preliminary implementation and extensive simulation based on real traces from a large-scale logistics system. At last, we review related works in Section V and conclude this paper in Section VI.

II. Preliminaries

In this section, we first briefly review the three types of collisions in RFID systems mentioned in the previous section, and then introduce our system model.

A. Tag Collision

The most common collision in RFID systems is tag collision, and it occurs when multiple tags in the interrogation region of a reader and transmit their IDs at the same time, as shown in Fig. I(a). A popular anti-tag-collision algorithm is Framed Slotted ALOHA (FSA) [9], [10], [26]–[28]. The design of our protocols is partially based on FSA. In FSA, the reader first divides a detecting procedure into several frames. Each frame contains f slots with equal length. At the beginning of one frame, the reader broadcasts the f to all tags and each tag randomly chooses a slot counter from 0 to $f - 1$. The reader then sequentially scans slots in the frame with the 'query' command. In each slot, if a tag's slot counter equals zero, it will backscatter its ID immediately. Otherwise, the tag decreases its slot counter by one. From the reader's perspective, there are three types of slots, 'idle', 'single', and 'collided' slots. In idle slots, no tag responds, the reader continues to scan the next slot. In single slots, only one tag replies, the reader can successfully receive the tag's ID. The reader then sends an acknowledgement of success 'ACKS' to notify the tag to keep silent in the left identification procedure. In collided slots, more than one tag responds such that the reader cannot identify any tag. The reader then sends an acknowledgement of failure 'ACKF' to indicate these tags to reply in the next frame. If there is any collided slot in the current frame, the reader renews a new frame until all tags are identified.

B. Reader Collision

This collision occurs at these tags located within the contentious regions covered by multiple readers. Engels [12] *et al.*

find that when two readers attempt to communicate with those tags at the same time, the signals cannot be correctly resolved and considered as environment noises by the tags. Meanwhile, each reader is unaware of the existence of interference from other readers. Therefore, those tags fail to be identified by any reader. To resolve the signal interference, existing approaches convert the potential reader collision to an undirected graph, named as Reader Conflict Graph (RCG). In RCG, a node represents an individual RFID reader and an edge represents a collision constraint: if two nodes are connected with an edge in RCG, the related readers will collide if they transmit the command at the same time.

Note that the edge does not represent the communication between readers but only the potential reader collision. Indeed, all readers in the network are often linked through LAN. The existing approaches, *e.g.*, Colorwave [13], utilize the graph coloring algorithm to solve reader collisions. Those solutions thereby are similar to finding the smallest number of total colors for coloring the RCG such that any two adjacent vertices are in different colors. Indeed, a color is corresponding to a periodic reservation for collision-free transmissions. The signal interference can be addressed by well arranging neighboring readers to send commands exclusively. However, as we discussed before, the exclusive scheduling incurs a poor throughput of identification.

C. Reader-Tag Collision

To resolve reader-tag collision, EPCglobal Gen II standards [9] specify two separated frequencies for the reader's query and tag's response, respectively, if readers are densely deployed. In fact, we can boil down reader-tag collisions to reader collisions as long as we synchronize all readers' behaviors. For example, if we schedule (r_1,r_2) in sequence, we can only consider the other two types of collisions.

D. System Model

In our model, we use *slotted channel* as the communication model between readers and tags. The transmissions happen within predefined and equally spaced intervals, termed as *slots*. The reader guarantees the slot synchronization via energizing probe/request. Obviously, the time required to identify tags is proportional to the number of tags. All readers are connected by wired or wireless networks which enable them to communication with each other at high speed.

Consider a set of readers $\mathcal{R} = \{r_1, \cdots, r_m\}$ are deployed at a region. An *identification procedure* is the procedure to identify all the tags within the region at a time. In this paper, we use the terms 'collect a tag', 'collect data from a tag', and 'identify a tag' interchangeably. For simplicity, we assume a unit disk model for the interrogation region of a reader. Note that our scheme is not constrained by this assumption. We denote $T_i = \{t_1, \cdots, t_n\}$ as the tag set. The tags in T_i are located in the interrogation region of reader r_i. $T = \cup_{r_i \in \mathcal{R}} T_i$ denotes the set of all tags. The neighboring reader set of r_i is denoted as $\Gamma(r_i)$. The tags in $T_i^C = \cup_{r_j \in \Gamma(r_i)}(T_i \cap T_j)$ are the *contentious tags* of reader r_i. They are located in the overlapped regions between r_i and its neighboring readers. On the contrary, the tags in $T_i^N = \cup_{r_j \in \Gamma(r_i)}(T_i \backslash T_j)$, where $T_i \backslash T_j = \{t | t \in T_i \ \& \ t \notin T_j\}$, are *non-contentious tags* of reader r_i. They are only covered by reader r_i.

We use undirect graph $G = (\mathcal{R}, E)$ to denote RCG. Reader r_i and r_j,where $r_i, r_j \in \mathcal{R}$, are adjacent in G if their corresponding nodes are connected by an edge, *i.e.*, $(r_i, r_j) \in E$. The degree $d(r_i)$ of reader r_i is the number of edges connected to r_i in RCG. The maximum degree of graph G is defined as $\Delta(G) = max_{r \in \mathcal{R}} d(r)$. We assume the readers can be well synchronized through a global clock and some synchronization protocols [16].

III. Season

In this section, we first present three important observations that motivate our design. We then present the design of Season and describe the three protocols we propose.

A. Observations

We observe three intuitive but important facts in practice:

Observation 1: Majority of tags are non-contentious due to the well advanced deployment of readers. If we allow the readers concurrently interrogate non-contentious tags, we can improve the identification throughput. Hence, a key step of improving the identification throughput is to enable the concurrency for neighboring readers in identifying non-contentious tags. This observation motivates us to handle the non-contentious and contentious tags separately.

Observation 2: The minor contentious tags indeed cause the major delay during the identification. Sometimes, only one contentious tag may incur large delay. As shown in Fig. 2(a), there is one contentious tag in the contentious region between r_1 and r_2, while no tag is in the contentious region between r_1 and r_3. However, both reader r_1 and r_3 are not aware of this situation since they have no knowledge about the locations of tags. These two readers have to be activated exclusively if utilizing prior works [12]–[14]. Therefore, we seek to design new identification pattern for contentious tags.

Observation 3: The signals from the contentious tags can be received by the readers that cover these tags. For instance, the responding signal generated by the contentious tag t_1 will be received by two neighboring reader r_1 and r_2, as shown in Fig. 2(b). If we can deliberately arrange one reader to interrogate the contentious tags while its neighboring readers passively listen to the responses from these tags, the system is able to retrieve the data from the contentious tag even there is a potential reader collisions in RCG. Unfortunately, existing approaches do not facilitate this feature.

B. Overview

Motivated from the above observations, we split an identification procedure in two phases. In Phase-I, the system identifies all non-contentious tags. We term this phase as *Shelving Interference*. In Phase-II, neighboring readers jointly and collaboratively identify contentious tags. We call this phase as *Joint Identification*. Hence, we propose our anti-collision

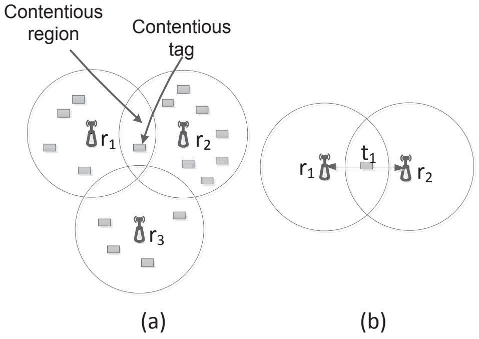


Fig. 2. (a) Potential reader collisions incur significant delay; (b) The signal from tag t_1 will be received by reader r_1 and r_2.

scheme, Season, to undertake the tasks of these two phases. Season is a protocol stack comprising of three protocols. Season-I is designed to collect data from non-contentious tags in Phase-I. Phase-II includes multiple rounds. In each round, we first employ Season-II to determine appropriate readers for actively interrogating contentious tags while keeping other readers passively listening. Then we conduct Season-III to collect data from contentious tags. The iterative execution of Season-II and Season-III continues until the system collects data from all contentious tags.

C. Season-I

In Phase-I, Season-I allows neighboring readers to concurrently identify tags in spite of the signal interference occurred at contentious tags. Although those contentious tags cannot correctly resolve the readers' query commands, this treatment helps us to naturally distinguish all non-contentious tags from contentious tags. The non-contentious tags transmit their IDs and then transfer into the silent state. Our approach can guarantee that the majority of tags can be identified after Phase-I, if most tags are located in non-contentious regions, and hence significantly improve the identification throughput. In an idea case, there are no contentious tags and all tags can be identified after Phase-I.

Season-I is a state based anti-tag-collision protocol. Similar to FSA, it divides the identification procedure into many frames and each frame contains several equivalent time slots. Different from FSA based approaches, Season-I adaptively tunes the length of frame to optimize the identification latency. That is, the reader will terminate the frame once it successfully receives a tag's ID and then start a new frame.

Obviously, each tag independently transmits its ID with a probability of $1/f$ in each slot. One important goal of Season-I is to choose appropriate f so as to minimize the expected identification time. Not surprisingly, the optimal choice of f is $|T_i^N|$. The problem of choosing an optimal f for ALOHA based approaches has been widely studied in the literature [17] [18]. But the challenge is that we usually do not know the number of tags in advance. Fortunately, a number of recent works [19]–[22], effectively estimate the number. We adopt USE [19] in Season. We require the reader to estimate number $|T_i^N|$ before identification. In detail, the reader maintains a variable k to record the number of tags that have been collected so far. Initially, $k = 0$. To minimize the identification time, we dynamically adjust the frame to $|T_i^N| - k$ after the k-th tag is collected.

D. Season-II

After all readers finish their identification of non-contentious tags, the system enters into Phase-II. Based on our third observation, we design joint identification protocols to identify contentious tags. For a group of neighboring readers, we only let one of them become active to interrogate the tags while others stay in silence and just passively listen to the signals from contentious tags. Joint identification has two clear advantages: (1) A joint identification can avoid reader collisions among neighboring readers since only one of them sends query commands, (2) Reduce the identification delay significantly because the readers concurrently receive the IDs of continuous tags.

For easy illustration, we term the reader being responsible for interrogating tags as *active reader* and the reader staying in listening state as *passive reader*. Hence, the first task in Phase-II is to select appropriate active readers from a group of neighboring readers. We propose *Season-II*, which is a distributed algorithm for determining proper active readers.

In our system model, the edges of a reader in RCG represent the contentious regions that the reader shared with its neighbors. In RCG, we determine active readers according to two conditions as follows: (1) they are able to cover edges as most as possible; (2) these active readers will not incur signal interference among themselves if they are concurrently activated. Namely, the selected active readers are not adjacent in RCG. Clearly, these two conditions are the Necessary Conditions for the optimal selection of active readers. We thereby convert the problem of selecting active readers to finding the Maximal Weighted Independent Set (MWIS) in an undirected graph.

Given an undirected graph G. A independent set of V is a subset $S \subseteq V$ such that no any two nodes $n, v \in S$ are neighbors in V, and every node $w \notin S$ has at least one neighbor in S. The MIS of V is the maximal independent set generated from V. A natural variant of MIS is the maximal weighted independent set (MWIS), where each node is associated with a weight. Solving MWIS is to find a MIS with the maximal total weight of its nodes.

In our problem, we set the weight of each node as the number of its edges since we attempt to employ the minimum nodes (active readers) to cover the maximum edges (contentious regions) in RCG.

We adopt a MWIS solution proposed by [23] to determine active readers. For example, as illustrated Fig. 4 (a), the set of active readers is $\mathcal{A}_1 = \{r_2, r_6, r_7\}$ while others are considered as passive readers, *i.e.*, the set of passive readers is $\mathcal{P}_1 = \{r_1, r_3, r_4, r_5\}$. By activating the active readers in $\mathcal{A}_1$ and keeping the readers in $\mathcal{P}_1$ listening, the contentious regions corresponding to the edges that are connected to the nodes in

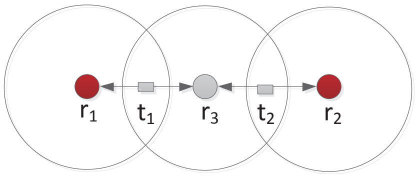
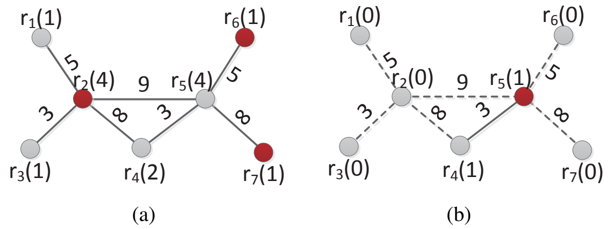


Fig. 3. Cross-range tag collision



(a)

(b)

Fig. 4. The number on the edge represents the real number of contentious tag. Active readers are shown in highlight. The number in bracket denotes the weight of the reader. (a) In the first round, active reader set is $\mathcal{A}_1 = \{r_2, r_6, r_7\}$; (b) In the second round, active reader set is $\mathcal{A}_2 = \{r_5\}$

$\mathcal{A}_1$ are covered by active readers. Thus, the contentious tags in those regions can be powered and successfully collected. However, only one round of finding MWIS is insufficient to cover all the contentious regions. From Fig. 4(a), we find that the tags in the contentious regions corresponding to edges (r_4, r_5) cannot be powered by any active readers because both r_4 and r_5 are passive readers.

To cover all contentious regions, we start the next scheduling round. In this round, we first let each node mark the edges that have covered in previous scheduling rounds and modify the node's weight as the number of left unmarked edges connected to this node. If the weight of a node equals zero, this node does not involve in this scheduling round. In this way, the active reader set becomes $\mathcal{A}_2 = \{r_5\}$ and passive reader set is $\mathcal{P}_2 = \{r_4\}$ in RCG, as shown in Fig. 4-(b). After the second round, all the contentious regions are covered. In practice, Season-II will be executed iteratively until all nodes' weights become zero.

E. Season-III

Season-III is designed to tackle a new tag collision. Assume readers r_1 and r_2 are chosen as active readers, as shown in Fig. 3. A tag collision happens at reader r_3 when tag t_1 and t_2 are interrogated by r_1 and r_2, respectively. In this case, reader r_1 can correctly receive the ID of t_1 and reader r_2 can retrieve the ID of t_2. However, reader r_3 cannot collect any ID because of the collision from two tags. We define such a tag collision as *cross-range* tag collision. Furthermore, both of reader r_1 and reader r_2 have no knowledge about whether reader r_3 has collected data from all contentious tags. This leads to a confusion from readers r_1 and r_2: when they should stop powering contentious tags? The above issue indicates that Season-I cannot be directly applied to collect data from contentious tags. Therefore, we propose a randomized protocol, Season-III, to allow active and passive readers to identify contentious tags collaboratively.

Given that the set of active readers is $\mathcal{A}$ and the set of passive readers is the $\mathcal{P}$ in the current scheduling round. Season-III works as follows.

On one hand, for active readers:

1) Each active reader $r_i \in \mathcal{A}$ starts a special frame to estimate the number of its contentious tags in its contentious regions using USE [19]. The number is denoted as n_i^1, which can be approximated to $|T_i^C|$. For instance, $n_6^1 \approx 5, n_2^1 \approx 3+5+8+9 = 25$ as shown in Fig. 4(a).
2) Reader r_i divides the procedure into several frames. Each frame contains n_i^1 time slots, where n_i^1 is a constant. In each frame, every contentious tag in r_i's contentious regions randomly selects a slot to transmit its ID. Namely, each tag independently transmits its ID with the probability of $1/n_i^1$ in each time slot.
3) Reader r_i always sends an ACKF feedback to the tag even if it successfully receives the tag's ID. This treatment is to force contentious tags always transmit its ID in each frame. In this way, we can guarantee every contentious tag has a chance to be identified by either active or passive readers.
4) After collecting data from all the contentious tags within its contentious regions, reader r_i still keeps the tags in the active state by powering the tags in this round because its neighboring passive readers may miss some tags due to the cross-range tag collision. This is in contrast to Season-I which immediately forces a tag to enter the silent state if the tag is collected in a slot. Until it receives "FINISH" messages from all its neighboring passive readers, the active reader ends its job in the current identification procedure. Note that once a reader becomes an active reader, it will quit Season after the current round.
5) Before ending its job, reader r_i broadcasts a "SILENCE" command to its contentious tags to force them to enter the silent state in the following scheduling rounds. The reader also sets its weight to zero in RCG.

On the other hand, for passive readers:

1) During the estimate phase of active readers, each passive reader $r_j \in \mathcal{P}$ listens to the responses from tags and estimates the number n_j^1 of contentious tags within the contentious regions between it and its neighboring active readers. The n_j^1 estimated by passive readers may be less than $|T_j^C|$, since there may exist contentious regions among passive readers. After estimation, $n_j^1 = |\bigcup_{r_i \in \Gamma(r_j) \& r_i \in \mathcal{A}} (T_i^C \cap T_j^C)|$. For example, $n_4^1 = 8$ and $n_5^1 = 5 + 8 + 9 = 22$ in Fig. 4(a).
2) Reader r_j passively listens to the responses from contentious tags during its neighboring active readers' interrogation. After collecting these tags, it sends a "FINISH" message to its neighboring active readers.
3) If reader r_j has no neighboring passive readers in this round, it ends its job in current identification procedure and sets its weight to zero in RCG. Otherwise, it still executes the Season protocols in the next scheduling

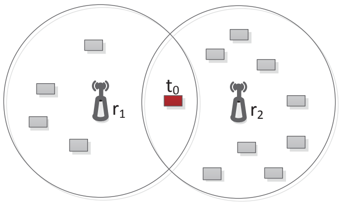


Fig. 5. Unbalanced loads of readers

round.

The role of readers may change during the scheduling round. Assume the scheduling sequence of active readers is $\{\mathcal{A}_1, \mathcal{A}_2\}$, where $\mathcal{A}_1 = \{r_2, r_6, r_7\}$ and $\mathcal{A}_2 = \{r_5\}$ as illustrated in the example shown in Fig. 4. Reader r_5 is a passive reader in the first round but it becomes an active reader in the second round. Once a reader becomes an active reader in one round, its weight will become zero and then finishes identification process.

To illustrate execution of Season, we give an example shown in Fig. 4. At the beginning of the first round shown in Fig. 4(a), active reader r_2 estimate the number of its contentious tags $n_2^1 \approx 3 + 5 + 8 + 9 = 25$. At the same time, the passive reader r_5 estimates the number of its contentious tags in the current round $n_5^1 \approx 9 + 5 + 8 = 22$. Reader r_2 continues to power tags until it collects the 25 tags and also receives the "FINISH" messages from r_1, r_3, r_4 and r_5. Concurrently, r_5 listens to the tags' replies. After successfully collecting its 22 tags, it sends a "FINISH" message to r_2, r_6 and r_7. At the end of the first round, all of the readers adjust their weights. Reader r_1, r_2, r_3, r_6, and r_7 set their weights to zero and report their collections. In the second round as shown in Fig. 4(b), there are only r_4 and r_5's weights not equaling to zero in RCG. Reader r_5 is selected as the active reader. It starts to power tags and r_4 listens to the tags' replies. The procedure ends when r_4 sends a "FINISH" message to r_5.

F. Discussion

1) Unbalanced Loads of Readers: The load of reader is defined as the number of tags located in its integration range. In Season, neighboring readers may have unbalanced loads. For example in Fig.5, reader r_2 have more tags in its interrogating regions than reader r_1. At the beginning of Phase-I, two readers cannot collect tag t_0 due to the reader collision. However, reader r_1 complete running Season-I earlier than r_2 and then stopping interrogating. Then the reader r_2 is able to collect t_0 since it is still running Season-I. In this case, some contentious tags may be collected in Phase-I and cause confusion to the joint identification in Phase-II.

We introduce session number to solve this problem. Each tag contains a session number with the initialized value as zero. At the beginning of Season, each reader randomly generates a non-zero session number and broadcasts it. If a tag can resolve a session number, it must be non-contentious. Then the tag changes its session number to what it receives. In Phase-I, the reader sends query commands with the non-zero session number. Each tag only replies the query command with same session number as it holds. In this way, the readers can only collect non-contentious tags during Phase-I. In Phase-II, the reader broadcasts query commands with a unified session number of zero. Because the contentious tag's session number has not changed during Phase-I, they will reply to the query command. Another consequence produced by unbalanced loads of readers is that the readers with lower loads will wait for the ends of the readers with higher loads during Phase-I. We can simply switch Phase-I and Phase-II to shorten such delay. Namely, readers first jointly identify contentious tags and then identify their own non-contentious tags.

2) Source Sensitive and Insensitive: RFID application can be summarized into two categories. One is source-insensitive, in which the source, namely the ID of reader that detected the tag, is not concerned. The user may only want to confirm that all tags can be collected, for example in warehouse monitoring. Another is source-sensitive, in which tags must be exactly reported multiple times, for example the object tracking. In the second type of applications, a tag can be approximately located by recording the readers that collect the tag. Duplicate reports of a tag from neighboring readers can also help the administrator to re-deploy readers for better coverage. Season can well support both the source insensitive and sensitive applications. For source-insensitive application, Season allows passive readers to immediately send a "FINISH" message to their neighboring active readers without identifying its contentious tags.

IV. Performance Evaluation

We now evaluate Season using real-world logistics and tracking traces.

A. Evaluation Methodology

1) Testbed and deployment: To validate the feasibility of joint identification, we use a NI PXI-1044 RFID testing tool with PXI 5600 receiver as our passive reader. We uniformly set the power of antenna as 20 dBm which supports around an interrogation range of $2m$. We also deploy five readers in a logistics enterprise, Xi'an postal processing center in Shaanxi, China. The center is the one of the seven largest postal processing centers in China. It covers an area of about 16,128m^2 and contains 30 importing/exporting gates. Fig.7 shows the architectural plans of the center. We attach more than 100 passive tags into pouches and find that the percentage of contentious tags is less than 10% for a stable and full coverage.

2) Simulating Real RFID Applications: For simulation, we use two typical application scenarios and three random reader topologies described as follows.

Warehouse: According to our measurement results in Xi'an postal center, we simulate a total of $12 * 6 = 72$ readers for covering the entire center in a square-grid formation. Each reader is located at one vertex in the grid. Each reader has an interrogating range of $7m$, which has 126 contentious regions.

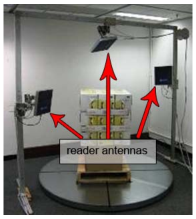


Fig. 6. PAMF

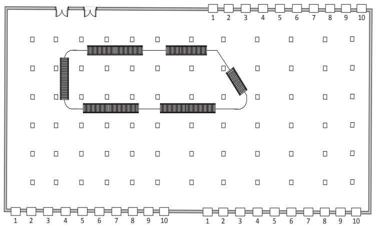


Fig. 7. Floor-map of the postal center

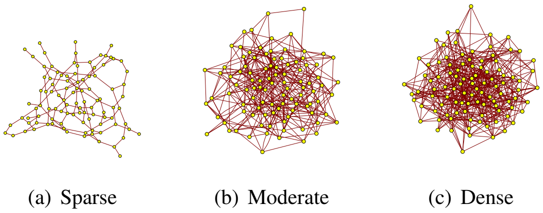

(a) Sparse
(b) Moderate
(c) Dense

Fig. 8. Random RCGs

We employ the real EMS trace of this center, which deliveries 2,456 items, including express mails, parcels, and boxes, to a medium-size city each day on average. We collect the delivery records in the month of December, 2009 as our basic dataset. The dataset contains 78,606 records.

Object tracking: We collect the tracking dataset from the RFID Ecosystem project [24]. The deploying map is shown in [25]. There are 30 readers (or antennas) in the deployment area. The tracking dataset has 1653 records. Each record includes the tag locations, source, and identification time.

Random Topologies: Without losing generality, we also randomly generated 3 separate RCGs with 100 readers, labeled with "Sparse", "Moderate", and "Dense", respectively. They have different maximum degrees to reflect the three deploying topologies, as summarized in Table I. The topologies of these three RCGs are also illustrated in Fig.8.

3) Performance Metrics: Assume the set of time slots consumed by a single reader r_i is $\mathcal{I}(r_i)$, then the identification time of this reader equals $I(r_i) = |\mathcal{I}(r_i)|$. Besides the identification itme, we also measure the following four matrices:

(i) Throughput: It is defined as the ratio of total number of tags to the overall identification time, denote as λ. Namely, $\lambda = \frac{|T|}{|\bigcup_{r_i \in \mathcal{R}} \mathcal{I}(r_i)|}$

(ii) Average Delay: The delay of tag t, denoted as $D(t)$, is defined as the expected number of time slots consumed by tag t in waiting for its identification. The average delay is defined as $D_{avg} = \frac{\sum_{t_i \in T} D(t_i)}{|T|}$

(iii) Read Rate: In practice, the reader cannot accurately collect all the tags in their interrogation region even if there is no reader collisions due to environment noise, multi-path, signal attenuation, and other factors. We use *read rate*, defined as the ratio of the number of correctly collected tags to the total number of tags in the interrogation region, to measure the feasibility of our approach.

(iv) Scheduling Round: We also evaluate the efficiency of anti-reader-collision by using the total number of scheduling rounds.

TABLE I
SUMMARY OF APPLICATIONS

Scenarios	# of readers	# of max degree	# of edges	# of tags
Warehouse	72	4	126	2,456
Tracking	30	3	29	1,653
Sparse	100	3	133	50,000
Moderate	100	8	343	50,000
Dense	100	16	495	50,000

B. Implementation Results

For testing joint identification, we employ a NI PXI-1044 testing tool with a PXI 5600 receiver as the passive reader. We also employ an Alien reader as the active reader. The tags are put in the middle between these two readers, with a distance $2.5m$ in between. The CDF of read rate of our passive reader is shown in Fig. 9. We can observe that the passive reader achieves a read rate of 0.73 in 60% of testing cases. The average value of its read rates is up to 0.71, which is nearly as good as that in the single-reader deploying scenario.

C. Simulation Results

1) Identifying tags without reader collisions: We first simulate the environment of deploying a single reader to show the performance of identifying non-contentious tags. We compare Season-I with prior anti-tag-collision protocols with number of tags ranging from 1 to 1000. The identification time of three types of protocols, FSA based approach, Balanced Tree (BT) based approach, and Season, is shown in Fig.10. From the figure, we observe that the identification time of each protocol is proportional to number of tags. Among them, Season-I is much faster than both FSA and BT based approaches. Especially when number of tag is above 100, Season-I has 30.6% and 42.2% time saving on average than BT and FSA, respectively. Furthermore, we also evaluate the throughput of these anti-tag-protocols as illustrated in Fig.11. The results show that Season-I is the best anti-tag collision protocol whose maximum throughput is up to 0.4 and 60% of the cases has a throughput higher than 0.37. However, the throughput of FSA and BT is typically lower than 0.29. We also observe that BT is the most stable protocol, 90% of the cases keeps around 0.25 to 0.26.

2) Identifying tags with reader collisions: In the experiment, we simulate multi-reader environments to show the performance of Season under reader-collision. We compare Season with DCS and Colorwave via the number scheduling rounds needed for anti-reader-collision. DCS and Colorwave employ the graph coloring method to schedule the readers. The results are shown in Fig.12. Season has the least number all the time compared with other anti-reader-collision protocols in the five scenarios. For example, Season only needs 4 scheduling rounds in the 'warehouse' scenario where the maximum degree

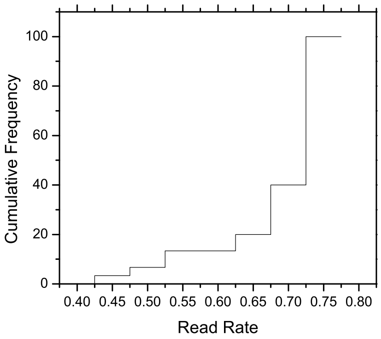


Fig. 9. CDF of read rate

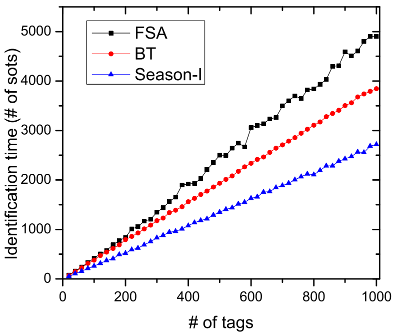


Fig. 10. Identification time

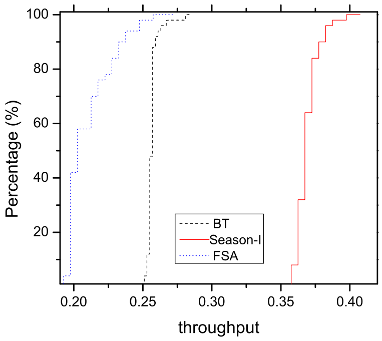


Fig. 11. Throughput

of RCG is 4. However, DCS and Colorwave require 27 and 28 rounds due to the high probability of collision among their randomly chosen colors in RCG. DCS is better than Colorwave since DCS knows the maximum degree of RCG in advance and this information helps DCS to reduce the probability of color collision.

Furthermore, we measure the overall throughput of three protocols in the five scenarios and show the results in Fig.13. The overall throughput of Season is much higher than other twos due to the concurrent identification of non-contentious tags and joint identification of contentious tags. For example, the overall throughput of Season in ‘sparse’ scenario is 8.5, meaning 8.5 tags can be identified per slot on average. Finally, we measure the average delay of the three protocols and plot the results in Fig.14. In all the five scenarios, the average delay of Season is no more than 300 time slots, which can be negligible in practice. It also indicates that Season can be applied in mobile environments to identify high-speed tags (up to 9 m/s). On the other hand, DCS and Colorwave suffer from a longer delays, *i.e.*, the longest delay is up to 62,352 time slots, in which some tags have to wait for at least 2 minutes before the reader collects them.

V. Related Works

In the literature, RFID anti-tag-collision mechanisms comprise of two categories, Framed Slotted ALOHA (FSA) based [9], [26]–[28] and Binary Tree (BT)based algorithms [10], [29]. The well known RFID organization, EPC Global, adopts a variation of FSA, ‘Q-Adaptive’ in its protocol family, EPC Gen2 [9], which adaptively tunes the frame length according to the type of last slot. Lee *et al.* [26] find that the maximum identification throughput can be achieved within a reader's scanning field when the size of detecting frame equals to the number of tags. Sheng *et al.* [27] focus on the fundamental problems of continuously scanning in RFID systems and design their identifying algorithms based on the information gathered in the previous scanning process. Xie *et al.* [28] involves the practical conditions in the design of probabilistic model of RFID systems, such as the path loss and multi-path effect. They also utilize the real settings to efficiently identify tags on the moving conveyor. The binary tree based algorithm has been adopted by another well-known RFID protocol family, ISO 18000-6 [10]. When designing tree based algorithms, researchers usually organize the tags in a binary tree according to their IDs and identify the tags by using the tree based search technique. Myung and Lee [29] propose an adaptive binary splitting (ABS) protocol to reduce collisions and efficiently identify tags based on previous result.

For avoiding reader collisions, Colorwave [13] is one of pioneer works. Colorwave tries to color the readers randomly in a RCG such that each pair of interfering readers can gain different colors. In [30], the authors suggest k-coloring of the interference graph, where the k is the number of available channels. Recently, EPCGlobal [9] proposes a dense reading mode, in which the tag responses happen in different channels to avoid collisions. In [31], the authors design a Q-learning process to arrange channels and allocate time slots for readers with a help of a training process. In [14], the author proposes a tag-access-scheduling protocol (EGA) based on STDMA. Tang, *et al.* [15] study a challenging problem of scheduling the activation of the readers without collision such that the system can wok in a stable way in the long term.

To speed-up the identification procedure, Floerkemeier [32] suggests estimating the cardinality of tags based on the number of idle slots in the current frame. Kodialam and Nandagopal [19] propose two estimation algorithms, Unified Simple Estimator (USE) and Unified Probabilistic Estimator (UPE) with three estimators. In addition, there are a lot of security and privacy issues about the RFID system, such as [33]–[35]. In our feature work, we will more focus on these issues in our protocols.

VI. Conclusion

Anti-collision is a crucial task in RFID systems. In this paper, we propose an anti-collision protocol stack, Season, to improve the identification efficiency for densely deployed RFID systems. Our results show that Season significantly increases the identification throughput in both the single-reader and multi-reader environments, and hence dramatically reduces the identification delay for tags. In our future work, we plan to extend our scheme to mobile reader environments and explore more practical issues in the RFID identification

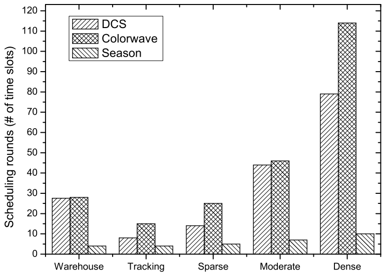


Fig. 12. Scheduling rounds

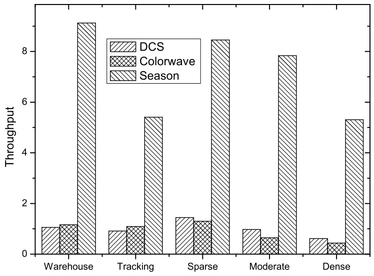


Fig. 13. Throughput

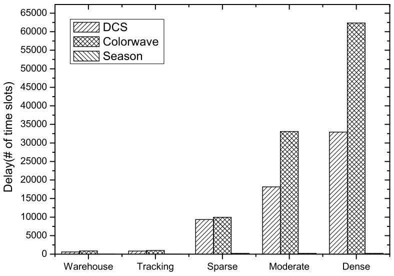


Fig. 14. Delays

procedure, such as the asynchronization, background noise, and the like.

Acknowledgment

We thank Qinsong Yao and Yiyang Zhao for their kind help in our experiment. This work is supported in part by National Natural Science Foundation of China (NSFC) (No.60933003, No.60736016, No.60873262, and No.60903155), National High Technology Research and Development Program of China (863 Program) under Grants No.2009AA01Z116, National Basic Research Program of China (973 Program) under Grants No.2011CB302705, No.2010CB328004 and No.2010CB328101, China Postdoctoral Science Foundation funded project (No.20090461298), Hong Kong Innovation and Technology Fund GHP/044/07LP and ITP/037/09LP, the Science and Technology Research and Development Program of Shaanxi Province under Grant No.2008KW-02, and IBM Joint Project. This work also Partially supported by the Danish Agency for Science, Technology and Innovation, International Network Program.

References

[1] B. Sheng, C. C. Tan, Q. Li, and W. Mao, "Finding Popular Categories for RFID Tags," in Proceedings of MobiHoc, 2008.

[2] T. Kriplean, E. Welbourne, N. Khoussainova, V. Rastogi, M. Balazinska, G. Borriello, T. Kohno, and D. Suciu, "Physical Access Control for Captured RFID Data," *Pervasive Computing*, 2007.

[3] C. C. Tan, S. Bo, and L. Qun, "How to Monitor for Missing RFID tags," in Proceedings of ICDCS, 2008.

[4] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: Indoor Location Sensing Using Active RFID," in *ACM Wireless Networks*, Volume 10, Issue 6, November 2004, Pages 701-710.

[5] M. Kodialam, T. Nandagopal, and W. C. Lau, "Anonymous Tracking Using RFID Tags", in Proceedings of INFOCOM, 2007.

[6] Y. Liu, Z. Yang, X. Wang, and L. Jian, "Location, Localization, and Localizability," in *Journal of Computer Science and Technology*, 2010.

[7] Z. Yang, and Y. Liu, "Quality of Trilateration: Confidence based Iterative Localization", in *IEEE Transactions on Parallel and Distributed Systems*, Vol. 21, No. 5, May 2010,Pages 631-640.

[8] Z. Yang, Y. Liu, X-Y. Li,"Beyond Trilateration: On the Localizability of Wireless Ad-hoc Networks", in *IEEE/ACM Transactions on Networking* (TON), Vol. 18, No. 6, December 2010, Pages 1806-1814.

[9] "EPCglobal RFID Class-1 Gen2 UHF RFID Protocol," Standard, 2005.

[10] "RFID For Item Management Air Interface. Part-6," ISO 18000-6 Standard, 2003.

[11] J. R. Cha and J. H. Kim, "Novel Anti-collision Algorithms for Fast Object Identification in RFID System," in Proceedings of ICPADS, 2005.

[12] D. W. Engels and S. E. Sarma, "The Reader Collision Problem," in Proceedings of IEEE SMC, 2002.

[13] J. Waldrop, D. W. Engels, and S. E. Sarma, "Colorwave: An Anticollision Algorithm for the ReaderCcollision Problem," in Proceedings of IEEE ICC, 2003.

[14] Z. Zhou, H. Gupta, S. R. Das, and X. Zhu, "Slotted Scheduled Tag Access in Multi-Reader RFID Systems," in Proceedings of ICNP, 2007.

[15] S. Tang, J. Yuan, X. Y. Li, G. Chen, Y. Liu, and J. Zhao, "RASPberry: A Stable Reader Activation Scheduling Protocol in Multi-reader RFID Systems," in Proceedings of ICNP, 2009.

[16] L. S. Leong, N. M. Ng, A. R. Grasso, and P. H. Cole, "Synchronization of RFID Readers for Dense RFID Reader Environments," in Proceedings of SAINT Workshops 2006.

[17] J. F. Kuros and K. W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, 2005.

[18] F. C. Schoute, "Dynamic Frame Length ALOHA," IEEE *Transactions on Communications*, 1983.

[19] M. Kodialam and T. Nandagopal, "Fast and Reliable Estimation Schemes in RFID Systems," in Proceedings of MobiCom, 2006.

[20] C. Qian, H. Ngan, and Y. Liu, "Cardinality Estimation for Large-scale RFID Systems," in Proceedings of PerCom, 2008.

[21] T. Li, S. Wu, S. Chen, and M. Yang "Energy Efficient Algorithms for the RFID Estimation Problem," in Proceedings of INFOCOM, 2010.

[22] H. Han,B. Sheng, C. Tan, Q. Li, W. Mao and S. Lu, "Counting RFID Tags Efficiently and Anonymously," in Proceedings of INFOCOM, 2010.

[23] S. Basagni, "Finding a Maximal Weighted Independent Set in Wireless Networks," *Telecommunication Systems*, 2001.

[24] E. Welbourne, K. Koscher, E. Soroush, M. Balazinska, and G. Borriello, "Longitudinal Study of a Building-scale RFID Ecosystem," in Proceedings of ACM MobiSys, 2009.

[25] http://rfid.cs.washington.edu/deployment/

[26] S. R. Lee, S. D. Joo, and C. W. Lee, "An Enhanced Dynamic Framed Slotted ALOHA Algorithm For RFID Tag Identification," in Proceedings of MobiQuitous, 2005.

[27] B. Sheng, Q. Li, and W. Mao, "Efficient Continuous Scanning in RFID Systems," in Proceedings of INFOCOM, 2009.

[28] L. Xie, B. Sheng, C. C. Tan, H. Han, Q. Li, and D. Chen, "Efficient Tag Identification in Mobile RFID Systems," in Proceedings of INFOCOM, 2009.

[29] J. Myung and W. Lee, "Adaptive Binary Splitting: A RFID Tag Collision Arbitration Protocol for Tag Identification," *Mobile Networks and Applications*, 2006.

[30] H. Gupta, Z. Zhou, S. R. Das, and Q. Gu, "Connected Sensor Cover: Self-organization of Sensor Networks for Efficient Query Execution," *IEEE/ACM Transaction on Networking*, 2006.

[31] J. Ho, D. W. Engels, and S. E. Sarma, "HiQ: A Hierarchical Q-learning Algorithm to Solve the Reader Collision Problem," in Proceedings of IEEE SAINT Workshops 2006.

[32] C. Floerkemeier and E. Zurich, "Transmission Control Scheme for Fast RFID Object Identification," in Proceedings of PerCom Workshop, 2006.

[33] L Yang, J.Han, Y. Qi, Y. Liu, "Identification-Free Batch Authentication for RFID Tags ", in Proceedings of IEEE ICNP, 2010.

[34] L. Lu, J. Han, L. Hu, Y. Liu, and L. M. Ni, "Dynamic Key-Updating: Privacy-Preserving Authentication for RFID Systems," in Proceedings of IEEE PerCom 2007.

[35] L. Lu, J. Han, R. Xiao, and Y. Liu, "ACTION: Breaking the Privacy Barrier for RFID Systems," in Proceedings of IEEE INFOCOM, 2009.