Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2010, Article ID 956578, 15 pages
doi:10.1155/2010/956578

Research Article

Bit Level Synchronized MAC Protocol for Multireader RFID Networks

Vinod Namboodiri and Ravi Pendse

Department of Electrical Engineering and Computer Science, Wichita State University, Wichita, KS 67260, USA

Correspondence should be addressed to Vinod Namboodiri, vinod.namboodiri@wichita.edu

Received 27 July 2009; Revised 24 June 2010; Accepted 12 July 2010

Academic Editor: Christian Hartmann

The operation of multiple RFID readers in close proximity results in interference between the readers. This issue is termed the reader collision problem and cannot always be solved by assigning them to different frequency channels due to technical and regulatory limitations. The typical solution is to separate the operation of such readers across time. This sequential operation, however, results in a long delay to identify all tags. We present a bit level synchronized (BLSync) MAC protocol for multi-reader RFID networks that allows multiple readers to operate simultaneously on the same frequency channel. The BLSync protocol solves the reader collision problem by allowing all readers to transmit the same query at the same time. We analyze the performance of using the BLSync protocol and demonstrate benefits of 40%–50% in terms of tag reading delay for most settings. The benefits of BLSync, first demonstrated through analysis, are then validated and quantified through simulations on realistic reader-tag layouts.

1. Introduction

A Radio Frequency Identification (RFID) system consists of one or more *readers* whose goal is to interact with small, low-cost *tags* in their field. Each tag has an ID (a bit string) which uniquely identifies it. Tags are typically attached to objects of interest, and a reader detects the presence of these objects based on an available mapping between IDs and objects. RFID tags can be *passive* or *active*, depending on whether they are powered by the reader or through a battery for activation and communication. Semiactive tags also exist that are activated through power from a battery but rely on the reader for energy to respond back. RFID technology is replacing the traditional bar code system and is envisaged to revolutionize supply chain management, distribution systems, banking, and more by providing services like object identification, tracking, and automation. These systems provide contactless operation with no line of sight required for operation.

As RFID technology matures, and pervades different aspects of our lives, the operation of multiple readers in proximity to each other is expected to grow significantly. From a technology perspective, tracking of individual items rather than just pallets or containers will eventually drive massive growth in the number of RFID readers required. The interrogation or read range of a reader depends on factors like antenna design, presence of obstacles, surrounding interfering sources, and tag characteristics. Often, one single reader's range is not enough to cover a region of interest. Because of their limited reading range, multiple RFID readers will be needed for coverage at retailing, manufacturing, or shipping/sorting sites. Thus many locations with RFID systems will likely require dense deployment of these readers which will coordinate among each other directly or through a centralized controller to process the information gained from identified tags.

The deployment of multiple readers in close proximity brings forth new issues as well. The issue of identifying multiple tags in the field of a single reader is termed as the *tag collision problem*. Several protocols based on the Aloha protocol or the binary tree scheme have been proposed to tackle this arbitration issue [2–4]. Two new problems that emerge in multireader deployments are the *reader-tag* collision problem and the *reader-reader* collision problem which are jointly referred to as the *reader* collision problem. The reader-tag collision problem arises when one reader's

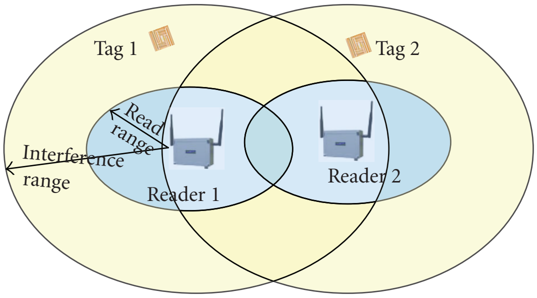


Figure 1: Reader-Tag Collision Problem: responses of a tag to a reader when queried are drowned out by the interfering signal from another concurrently operating reader in the vicinity. In this example, Tag 2 is within the read range of Reader 2. Reader 2 is also in the interfering range of Reader 1. After Reader 2 queries Tag 2 for its ID, the response of Tag 2 at Reader 2 is interfered with by the signal from Reader 1 which was querying Tag 1 at the same time. [1].

signal drowns out the relatively weak response signal of a tag to another reader in the vicinity. This happens due to the large interfering range compared to the reading range in RFID systems. The reader-reader collision problem arises where tags in the read range of multiple readers (referred to as overlapped tags) may not be reliably read. In this case, both the reception as well as response of a tag are affected due to the greater strength of the interfering signal in the reader's read range. We illustrate these problems through examples in Figures 1 and 2.

Previous work has found ways to solve the reader-tag collision problem but the solution to the reader-reader collision problem remains open [5]. The only effective combined solution for the set of both reader collision problems is to use time division multiplexing to schedule the readers to operate one after the other. This approach, however, increases the delay to identify all tags due to the sequential nature of reader operations. Thus, there is a need to develop approaches that solve both problems at the same time in an efficient manner that reduces the delay in reading all tags.

In this paper, we present a novel approach to solve the set of reader-collision problems by synchronizing the conflicting readers' timing as well as forcing them to use the same query within the framework of the Frame-Slotted Aloha tag anticollision protocol. With identical queries, the bits sent out by each reader are identical and do not collide with each other when received by tags. An analogy is the presence of multiple audio speakers in large areas like airports and railway terminals broadcasting the same announcements at the same time. This allows people to comprehend the information without the announcements from different speakers interfering with each other. To any tag in the range of multiple readers, it appears that there was only one reader sending a query, which makes the query-response process simple and collision-free. This allows simultaneous or parallel operation of the readers minimizing the total tag identification delay. This bit level synchronized query approach works with RFID because of the nature of communication involved between multiple readers and tags. The main goal for all readers is to identify all tags, and not voice or data communication typical of wireless networks.

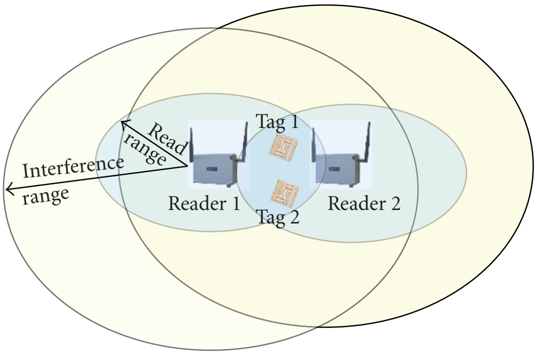


Figure 2: Reader-Reader Collision Problem: tags fall within the read range of multiple readers (called overlapped tags), and the query-response sequence with one reader is interfered with by the simultaneous query-response sequence with other readers. In this example, Tag 1 and Tag 2 are overlapped tags within the read range of both Reader's 1 and 2. When both readers operate at the same time, each interferes with the query-response sequence of the other with any of the overlapped tags. This type of interference is stronger (as compared to that of Figure 1) as the affected tag is also within the read range of the interfering reader and affects both the reception as well as response of signals to/from the tag.

Our contributions, other than the proposal of the Bit Level Synchronized (BLSync) MAC protocol, are to analyze and demonstrate its potential benefits to multireader RFID deployment scenarios. We analyze the benefit of using this approach compared to the sequential approach where all conflicting readers operate one after the other. It is demonstrated that for low to moderate overlap between the reader ranges, our protocol reduces the total time to identify all tag IDs by 40%–50% for most settings. We further reinforce our analysis through simulations where we consider the practicality of the protocol and study its performance for various realistic reader-tag topologies under varying levels of time synchronization constraints. Finally, we study the parallel operation of RFID readers from a more fundamental perspective and demonstrate and analyze an additional benefit (which we term as the *assist effect*) of the BLSync protocol where neighboring readers with mutually overlapped tags assist each other in reading tags.

The paper is organized as follows. In Section 2 we present in detail the application area under consideration and the motivation for our work providing more details about the set of reader collision problems from the perspective of previous research in the area. Our bit level synchronized approach to solve these problems is described in Section 3 as the BLSync protocol. An analysis of the reduction in time required to identify all tags with our BLSync protocol is done in Section 4. Section 5 presents simulations carried out for multiple realistic topologies quantifying the performance gains using BLSync protocol and also serves to validate our analysis. We analyze the assist effect (mentioned above) of the BLSync protocol in Section 6 to provide a more fundamental understanding of the benefits of synchronized parallel operation of RFID readers. Concluding remarks are made in Section 7.

2. Background and Problem Definition

In this section we describe our target application area in detail and define the problem to be solved. We motivate the need for our BLSync protocol by describing the set of reader collision protocols in detail including previous work in the area. We also point out the assumptions made about the type of deployments we consider.

2.1. Application Area. In this paper, our focus is on applications which require multiple readers to service an area with RFID tags. For such deployments a centralized controller exists (could be one of the readers) which can communicate with all the readers using a Local Area Network (LAN) technology like Ethernet or Wireless LAN and can also serve as a database or data collection point. The purpose of the reader deployment may be to identify the tags as they move along on a conveyor belt, to gather information about the location of objects in a supply chain, or just inventory the items in the area. For example, pallets loaded with tagged items may move through docks in a warehouse to be scanned. Multiple such docks are typically located in close proximity, with readers continuously scanning to identify tags and send collected information for processing to a centralized database. In these scenarios, it is preferable to minimize the delay in reading tags so that each reader gets to scan as often as possible so as not to miss identifying any tags. This also allows the pallets to pass through as fast as possible, speeding up the entire operation. For these applications we can define the *total tag reading delay* as the time required to collectively read/identify all tags in the interrogation zone of the readers. Our main interest in this paper is to propose a protocol to solve the set of reader collision problems in a way that minimizes total tag reading delay.

2.2. Motivation for BLSync. The set of reader collision problems arise mainly due to the huge asymmetry between the reading range and interference range of an RFID reader. An RFID reader typically uses high powered signals so that there is enough energy for passive tags to energize themselves as well as respond back. These high powered signals can interfere with tags operating hundreds of meters apart even though the same signal has only enough energy to allow reading tags within a few meters from the reader. As shown in Table 1, for small-to moderate-sized areas of operation (these are areas no more than 1000–3000 sq. meters with maximum separation between readers no more than few hundreds of meters. For those unfamiliar with the metric system, 3000 sq. meters is equal to 32292 sq. feet and is approximately equal to three-fourth of an acre), all readers will mutually conflict with each others operations by drowning out tag responses to neighboring readers when they operate (i.e., the reader-tag problem exists). Moreover, regulations in some countries prohibit a centralized coordinator to assign frequency channels to multiple readers. Even if readers can be operated on different channels, the channels need to be significantly apart from each other as shown in Table 1. This is not always easy due to only a small, fixed number of channels available for use. The problem is especially evident at lower frequencies like 13.56 MHz where only a single channel is available [6]. In such a scenario, the reader-tag collision problem affects even large-scale deployments.

Table 1: Minimum required distance between readers [5].

Channel Difference	Antenna projecting horizontally		
	Front (m)	Side (m)	Back (m)
0	1400	350	210
1	180	45	30
2	130	25	15
3	95	20	10

Colorwave was one of the initial solutions to the reader collision problem where readers were assigned different time slots to operate by casting it as a graph coloring problem [7]. This solves both the reader-tag and reader-reader collision problems. Further improvements and variants to this approach are described in [2, 6, 8, 9]. Readers, if scheduled to operate in a time-multiplexed manner one after the other in different time slots (i.e., sequentially), will have a tag identification delay dependent on the number of interfering readers in the vicinity. The greater the delay to finish operation of all readers in one round (or sequence), the greater the interoperation interval for each reader. This is because the next scan of each can begin only after all readers in the previous round have finished reading tags in their interrogation zone.

An analogous solution based on separating conflicting readers by frequency channels instead of time slots was developed in [10]. This solution, however, does not solve the reader-reader collision problem as overlapped tags may be queried by multiple readers on different channels. A tag is indifferent to any specific channel as it responds to queries from any channel and hence can still be affected. Techniques at tags to differentiate channels, as in the dense reading mode of the EPCGlobal Gen 2 standard [11], require relatively sophisticated tag technology making them very expensive compared to existing low-cost tags.

The authors of [5] proposed to synchronize reader operations to solve the reader-tag collision problem. All readers send signals to read their tags at the same time and then await tag responses together so that one reader's signal does not drown out tag responses to other readers. Again, however, the reader-reader collision problem remains unsolved when there are overlapped tags between different sets of readers in a deployment.

Overlapped tags are a reality in most deployments regardless of whether they are well planned or not. If the deployments are not well-planned, there might exist many overlapped tags in the area which will suffer from the reader-reader collision problem. On the other hand, in well-planned deployments, some overlap might be still desirable to increase reliability since reader ranges may vary due to changing channel conditions in the wireless environment. The chances of overlapped tags is exacerbated by the presence of heterogeneous tags in the area. For example, the planning stage may minimize overlap between readers considering only one type of tags, say the passive tags which tags rely on the reader signal to power them up and respond back. However, semiactive tags and active tags with much larger communication range than passive tags may also appear in the area. These tags require much lesser or no energy at all from the readers signal to power up and respond back, and hence, are more likely to be in the range of multiple readers.

Thus, as discussed above, the only solution to the set of both reader collision problems is to operate conflicting readers at different times. By conflicting readers we refer to the set of readers that are affected by either the reader-reader or reader-tag collision problems. For the small-to moderate-sized deployments we consider the interfering signals are large enough to affect all readers. Such a *sequential* nature of reader operation may result in significant total tag identification delay. It is in this context that we propose in the following section our bit level synchronized MAC protocol that allows multiple readers to operate at the same time without interference. The resulting *parallel* operation of readers will be shown to minimize the total tag reading delay for most settings compared to the *sequential* operation.

2.3. Assumptions. We will limit our protocol description and performance analysis in subsequent sections to only application deployments where all the readers can mutually interfere with each other based on the affected distances shown in Table 1. These distances apply to small to moderate size deployments if reader signals can be separated over multiple channels, or to large deployments if only one channel is available. The extension of the protocol description (and analysis of benefits) to include the coordination required between readers for deployments where some readers can operate without conflict is left for future work. This could, for example, use schemes like those presented in [9] that allows nonconflicting readers to operate together. However, our understanding is that most application deployments will not exceed the dimensions in Table 1 which requires large amounts of space and supporting infrastructure.

3. BLSync Protocol

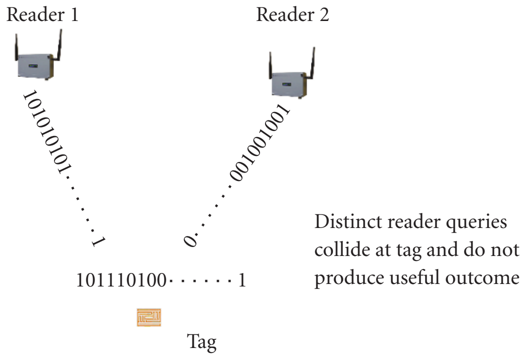
We present our BLSync Protocol in this section. We begin by giving a quick overview of the existing Frame Slotted Aloha protocol which is used as the underlying tag identification protocol with BLSync as well.

3.1. Frame Slotted Aloha. Tag anticollision or arbitration protocols are typically classified into those based on the Aloha protocol or the binary tree protocol [2–4]. The former protocol separates tag responses by asking them to use different time slots while the latter separates them out by querying their IDs in a bit by bit fashion. The Frame Slotted Aloha protocol is widely used due to its simplicity and is the protocol specified for the Class 1 Gen 2 EPC standard [11]. The reader query to tag consists of, among other things, the frame size or number of slots for which the reader will listen for tag responses. Tags randomly pick one of these slots to respond to the reader with their ID. Since multiple tags could pick the same slot, collisions may happen in slots. Thus, a reader generally needs to query for multiple rounds before all tags are read. To prevent successfully read tags from replying again in the same read cycle, they are acknowledged before the next round begins which puts them in a “sleep” state.

3.2. Advantage of Bit Synchronized Reader Queries. Generally, each reader queries tags in its range independently of any other readers in the vicinity. Thus, it is free to choose the frame size it deems fit based on its estimate of the number of tags in its interrogation zone. Apart from the frame size, another significant part or field of a query is the session or read cycle number initiated by a reader. This is used to ensure, among other things, that tags read once, will not be read again until certain conditions are satisfied. This is also usually independent of what other readers have in their queries. Thus, when two readers in vicinity of each other send their distinct queries at the same time, their signals collide with each other at tags which receive both these queries as shown in Figure 3 (such tags in the interrogation zone of one or more readers will be referred to as overlapped tags). This could delay the identification of these tags or prevent identification altogether.

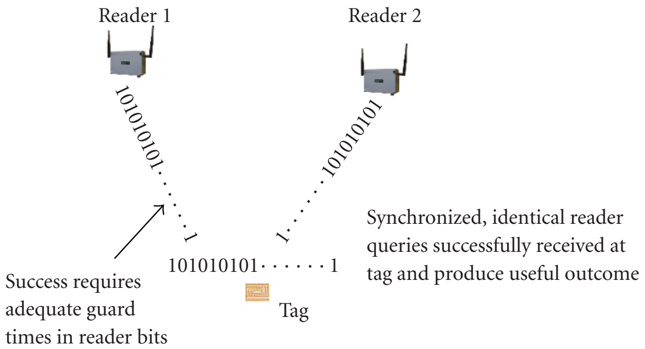
The basic idea of the Bit Level Synchronized (BLSync) MAC protocol is to utilize the fact that *the common goal of all readers is to identify tags, and not individually communicate with tags.* This allows for a protocol where all readers are performing the same queries (i.e., identical bits), and hence, can synchronize and transmit simultaneously without causing collisions at the tags. The identical bits (and hence their signals) used mean that interference from neighboring readers does not result in collisions, and the received signal level remains high or low depending on whether the transmitted bit by the readers is a “1” or a “0” as illustrated in Figure 4. We assume that diversity combining techniques commonly used at the physical layer can be used at the analog level. Such techniques are commonly used in applications like cooperative relays where the same information from multiple sources has to be correctly decoded at the receiver. We do not go into these details in the paper as our paper focuses more on higher-layer aspects. Further reading on diversity combining and related literature can be found in [12–14] and the references therein. Tag diversity combining has been briefly described in [15].

Each reader is assumed to have at least a Wireless LAN (WLAN) interface (typically Wi-Fi) which can operate separately and simultaneously with tag reading operations if required. This is a standard interface on many RFID readers in the market these days and is used primarily for communication with the centralized controller to process the information from identified tags. Having an Ethernet interface is also common and will only improve the way



Distinct reader queries collide at tag and do not produce useful outcome

Figure 3: Figure showing that distinct reader query bits at an overlapped tag result in a corrupted query reception.



Success requires adequate guard times in reader bits

Synchronized, identical reader queries successfully received at tag and produce useful outcome

Figure 4: Figure showing that the same reader query bits at an overlapped tag result in correct query reception.

readers can participate in the BLSync protocol. The steps of the BLSync protocol are outlined as Algorithm 1.

3.3. Protocol Description. It is possible for all readers to use identical queries, because for the Frame Slotted Aloha (FSA) protocol, the readers initially send the frame size and session number and await tag responses in slots of the frame. If the frame size and session number chosen by all readers are identical, the queries could be made identical, and hence, noninterfering.

Any tag in the range of multiple readers, on receiving the common query, sends only one response. This response will be received by both readers at the same corresponding slot since the readers' queries are synchronized in time and use the same frame size. Tags read by readers are acknowledged in a sequential manner after the query frame so that no other reader is transmitting at the same time (i.e., ACKs are scheduled sequentially unlike queries). Acknowledged tags go to sleep for this tag read cycle or session and do not respond to any more queries with same session number. Overlapped tags thus go to sleep mode after the first reader that ACKs them. The other associated readers of these tags need not ACK them again as they can be notified about the status of these tags by the earlier reader in the ACK sequence (for example through a common WLAN network) which read these tags.

Once all tags read in the first round are acknowledged by all readers, the second round of synchronous querying begins similarly with all readers using the same frame size, which could be different than what was used in the first round. The tags read are acknowledged sequentially as before, and rounds continue till all tags are read.

The key aspect of the protocol is the use of the same frame size by all readers when querying. Rough estimates of the tag population may be available on an individual reader basis [16], or collectively due to knowledge of possible tag distribution patterns. If an estimate on the tags in the range of each reader is available, the controller sets the frame size based on the reader with the most estimated tags in its range. If only the collective tag estimate of the area is available and a uniform distribution is expected, all tags will use a frame size equal to m'/n, where m' is the tag estimate and n is the number of readers. If tag estimate is unavailable or the tag distribution unknown, all readers can pick a uniform frame size and increase/decrease the size together as required based on FSA. If some readers have much fewer tags in range, they should still use the same frame size as others as long as tags remain to be identified in their field. When all their tags are read, they can stop querying while other readers continue till their tags are read. Note that after first round of FSA by all readers, it is necessary for the readers to decide on the common frame size for the next round and so on. To avoid delaying the process, this communication with the controller can begin immediately on the WLAN network after the reader receives all tag responses in the frame and before it acknowledges any resolved tags. As ACKs to tags in the BLSync protocol are sequentially sent by one reader after another, each reader first acknowledges tags in its range before it is the turn of the next reader and so on. During the turn of a reader to acknowledge its tags, it can at the same time send the number of tags identified to the centralized controller. Channel access contention should not be an issue here as each reader will not be using its WLAN interface until its turn to acknowledge tags. Thus, the WLAN medium is itself accessed sequentially.

The prospect of multiple readers in an area operating in synchronized fashion using the same queries raises the question: Why not just use one reader *with large enough power to read all tags in the area?* The answer is that there are restrictions on the maximum output power of the readers that limit their interrogation zones, requiring multiple readers to cover the area [17].

3.4. Example. In Figures 5 and 6, we show working examples of identification delay for both the sequential operation of readers and parallel operation of readers utilizing the BLSync protocol. The scenario has three readers R1, R2, and R3 and seven tags numbered from T1 to T7. For the sake of this description, assume that each reader knows the number of tags in its range and can find the number of unidentified tags after each round by subtracting the number of identified tags from the total number.

(1) Centralized controller collects estimated count of tags in range of each reader
(2) Centralized controller determines the common frame size to be used by all readers and sets the common query to be used including the session number.
(3) The readers are synchronized in time.
(4) All readers send their queries at the same time in parallel and receive tag responses for this round based on the Frame Slotted Aloha (FSA) protocol.
(5) Based on responses received, each reader determines how many tags were read, and how many will have to be read in the next round of FSA.
(6) All resolved tags are acknowledged sequentially and put to sleep by the readers, operating in a pre-determined order. At the same time, in parallel, the number of tags to be read in the next round is sent to the centralized controller on the WLAN network by each reader.
(7) The protocol continues until all tags are read.

Algorithm 1: BLSync Protocol.

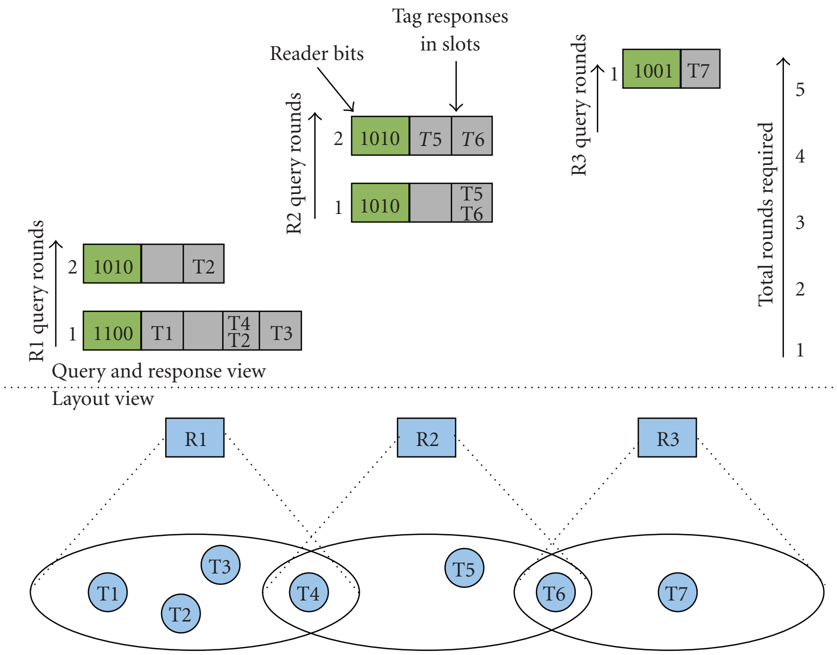


Figure 5: Working example of sequential operation of three readers. The lower part of figure shows the layout of readers and tags within the read range of each. The upper part shows in detail the progression of reading tags as the readers operate one by one in the order R1, R2, and then R3. In a query round, each reader sends a query of bits and awaits responses over a frame of time slots. Each reader chooses its query independently based on its estimate of tags in its reading range. All tags are read after a total of 5 rounds of querying requiring 11-tag response time slots (4,2,2,2,1 in those rounds) in all.

3.4.1. Sequential Frame Slotted Aloha (FSASeq). In the sequential operation readers in Figure 5, reader R1 begins operation first trying to identify the four tags in its range by sending a query with four time slots. For the example, the first bit of each reader query is the session number which is arbitrarily taken as 1 in this case. The next three bits are used to signify the number of slots to be used in the frame. A frame size of 4 requires the last three bits to be 100. In an actual deployment, the protocol can use more number of bits to specify frame sizes. In the first round R1 identifies two tags and puts them to sleep. R1 requires another round of querying with two time slots before it identifies the remaining tags. When R2 begins operation next, T4 which was an overlapped tag for R1 and R2 has already been identified and put to sleep by R1. So R2 uses only a frame size of two time slots and ends up requiring two rounds as well. When R3 begins operation, only T7 remains to be identified which is done with a frame of one time slot. Thus, a total of five rounds of querying were required with a total of 11 tag response time slots. This count ignores the smaller time slots used by readers to send query bits. The figure is not drawn to scale. Time slots in which tags respond are much larger as they need to send their whole ID which is typically 128 bits long. The time slots on which readers send queries would be smaller as it mainly requires only the bit representation of number of tags in the field which would typically be no more than 10 bits.

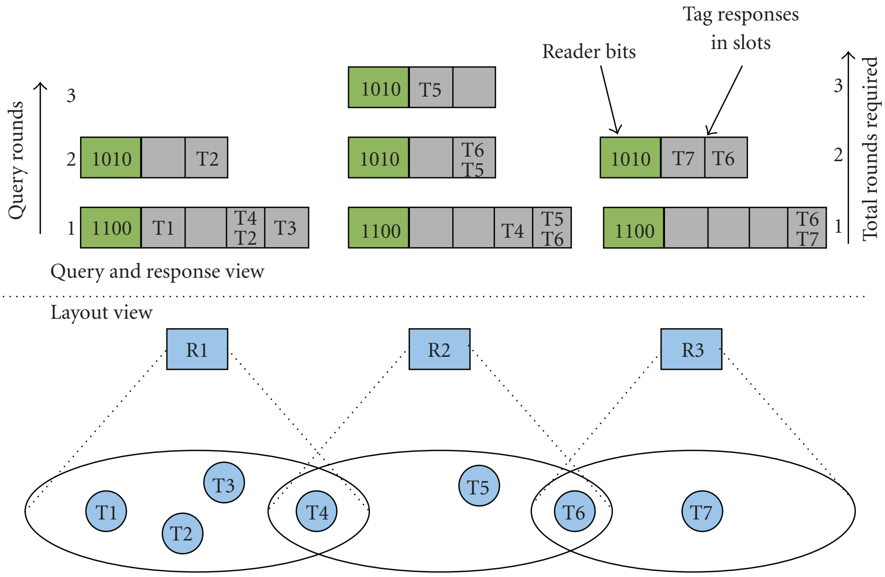


Figure 6: Working example of BLSync Protocol. The lower part of figure shows the layout of readers and tags within the read range of each. The upper part shows in detail the progression of BLSync over multiple rounds. In each round, all readers send a common query of bits and await responses over a frame of time slots. In each round, the query bits of all readers are the same. All tags are read after a total of 3 rounds of querying requiring 8-tag response time slots (4,2,2 in those rounds) in all.

3.4.2. BLSync Protocol. Figure 6 shows a working example of the BLSync protocol. All three readers start out with the same frame size of four slots as R1 decides to use that size for the four tags in its range. T1 and T3 are read in the slotted frame of R1 while T2 and T4 collide in slot 3. T4 is, however an overlapped tag, and its response is collision-free for R2. Hence, T4 is acknowledged and put to sleep by R2. Other tags similarly are either acknowledged and put to sleep by readers or remain to be read in future rounds. In round two, R2 requires a frame size of two, and hence all readers use that frame size even though R1 had to read only one tag (R1 does not know that T4 has been put to sleep by R2). T6 collides in R2's frame but is clear in the frame of R3 and hence put to sleep. After round two, since only R2 has not any tags left to read, the other readers do not send any more queries. Note that R2 still uses a frame size of two as it does not know that T6 was identified and acknowledged in round two. Through the example, the first bit of each reader query is the session number which is arbitrarily taken as 1 in this case. The next three bits are used to signify the number of slots to be used in the frame. For the example, frame size of 4 and 2 require the last three bits to be 100 and 010, respectively.

The important result of the BLSync MAC protocol is that it solves the reader collision problem. With all readers sending the same query bits, any overlapped tag can be accessed as if only a single query was sent to it. Tight time synchronization of queries can be achieved using the centralized controller that oversees all the readers and communicates to them directly. In practice, the use of extraguard time (of the order of time synchronization error) is required to ensure that tags receive each bit from readers correctly. Thus, the protocol bit transmission from readers would look like as that shown in Figure 7.

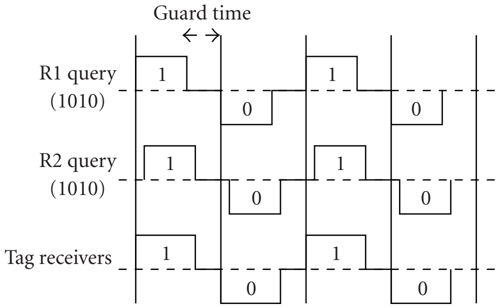


Figure 7: Bit level synchronization with guard time to make up for synchronization errors between readers. The above example is shown for the case when all readers want to send queries with bits 1010 to tags.

4. Analysis of Tag Reading Delay with BLSync

We mentioned how each reader uses the Frame Slotted Aloha (FSA) protocol to identify tags in its interrogation zone. When these readers operate in sequential manner, one after the other, we term the collective protocol as FSASeq as termed in Section 3.4 as well. Here we will perform a simple average case analysis of the reduction in tag reading delay achieved by using the BLSync protocol at all readers over the FSASeq protocol. An overview of these protocols and their relationships are shown in Figure 8.

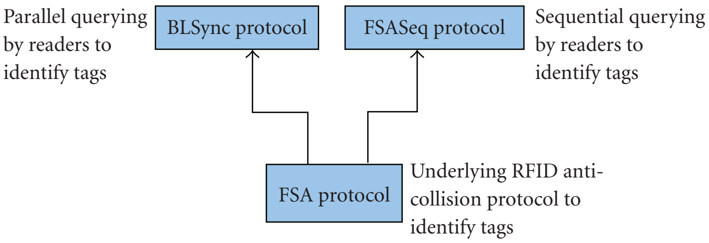

Parallel querying by readers to identify tags
BLSync protocol
FSASeq protocol
Sequential querying by readers to identify tags
FSA protocol
Underlying RFID anti-collision protocol to identify tags

Figure 8: Relationships between the BLSync and FSASeq Protocols in terms of the FSA Protocol.

We consider scenarios with n readers that cannot operate together and need to be scheduled sequentially one after the other as explained in Section 2. There are m tags to be read in range of the readers (at the instant before any reader begins operation), and it is assumed that this number m and the number of tags in the interrogation zone of each reader i, m_i, are known. This enables efficient operation of the FSA protocol, and for this reason, the estimation of tag count is often used as a preliminary step before the actual arbitration process [16, 18, 19]. This assumption also allows comparisons without the inefficiencies of the Aloha protocol when the number of tags to read is unknown. Note that, this assumption favors the FSASeq protocol, since wasted time slots due to suboptimal frame sizes need to be accounted for each reader individually (and in additive fashion). In contrast, the BLSync protocol uses only one frame in parallel at all readers.

We begin our analysis with the FSA protocol which is used by both FSASeq and BLSync protocols. This is followed by developing expressions for the time slots required to read all tags when using the FSASeq and BLSync protocols. We will subsequently compare these two based on the amount of overlapped tags between readers and the type of tag distribution in the layout under consideration. The analysis in this section does not take into account the use of guard times in protocol timing to handle time synchronization errors. We present those results in the following section where we compare FSASeq to BLSync for practical deployments. Further, we ignore the relatively small time taken by the readers to transmit query bits as explained in Section 3.4 as well. Our notion of time to identify all tags only looks at the number of time slots required. We will consider the time to send query bits when we modify our results to include guard times in the following section.

4.1. Analysis of Frame Slotted Aloha (FSA) Protocol. We begin by analyzing the number of time slots required to read m_i tags in the range of a reader i. It is well known that the optimal number of slots per frame to maximize the number of tags read per frame using Frame Slotted Aloha is equal to the number of tags to be read [20]. Further, the throughput of the Frame Slotted Aloha protocol is approximately equal to $1/e$ when the size of the frame equals the number of tags responding [20, 21]. So, with $1 - 1/e$ fraction of tags in each round remaining un-identified, in r_i rounds we would have $(1 - 1/e)^{r_i-1}m_i$ tags left to be identified from the initial m_i tags. We want to find out the number of rounds required such that the expected number of un-identified tags is less than or equal to $1 - \gamma$, where γ is the fraction of tags to be identified. Since we seek the minimum number of such rounds, we require

$$\left(1 - \frac{1}{e}\right)^{r_i-1} m_i = (1 - \gamma)m_i \qquad (1)$$

or

$$r_i = 1 + \frac{\log(1 - \gamma)}{\log(1 - 1/e)}. \qquad (2)$$

By tuning γ to a large value, we can approximately find the expected number of rounds of FSA required to read all tags. Note that this value of number of rounds required is independent of the number of tags to identify. Thus we have $r_i = R$, for all i, where R is constant for a given value of γ.

The number of time slots needed to read all tags m_i in the range of reader i then is simply the sum of un-identified tags (or frame size) per round

$$t_i(m_i) = \sum_{j=0}^{r_i} m_i \left(1 - \frac{1}{e}\right)^j = m_i \sum_{j=0}^{R} \left(1 - \frac{1}{e}\right)^j, \qquad (3)$$

for $\gamma = 0.99$, $R = 10$, and the time to read m_i tags is $2.685m_i \approx e \cdot m_i$. This can also be obtained from the fact that for large γ, R is large which makes $\sum_{j=0}^{R}(1 - 1/e)^j = (1 - (1 - 1/e)^{R+1})/(1 - 1 + 1/e) = e[1 - (1 - 1/e)^{R+1}] \approx e$. In summary, to read m_i tags, a reader needs $e \cdot m_i$ time slots with FSA.

4.2. Analysis of Sequential Frame Slotted Aloha (FSASeq) Protocol. As explained before, when multiple readers interfere with each other, they are scheduled by the controller to operate at different times, often one immediately after the other to scan all tags in their area. Thus, the time to read all tags with FSASeq protocol then would be the sum of individual time required by each reader i to read its tags. For simplicity, we ignore the small fraction of tags $1-\gamma$ (typically ≤ 0.01) which may remain unread. Let x_i be the number of tags already read from the m_i tags in range of a reader i by readers operating before reader i in the sequence. Using our previous notations and (3), the time to read all tags, m, is

$$t_{\text{seq}}(m) = \sum_{i=1}^{n} t_i(m_i - x_i) = e\sum_{i=1}^{n}(m_i - x_i) + m, \qquad (4)$$

where the final term adds the number of acknowledgment slots needed to put all tags to sleep once they are identified.

Lemma 1. *The expected time to read all tags in the FSASeq protocol is $e \cdot m + m$ regardless of the degree of overlap among readers. That is, in* (4), $\sum_{i=1}^{n}(m_i - x_i) = m$.

Proof. No tag is read by multiple readers in the same read cycle. Once a reader reads tags in its range, it acknowledges them (before the next reader in sequence starts operation) which puts them to sleep for this read cycle. Moreover, since readers cover all tags, each tag is read at least once. Thus, each tag is read exactly once. Similarly, each tag is acknowledged exactly once. Thus, the time to read all tags is $e \cdot m + m$, independent of individual values of m_i and x_i. □

Thus, we can write

$$t_{\text{FSASeq}}(m) = e \cdot m + m. \tag{5}$$

4.3. Analysis of Bit Level Synchronized MAC (BLSync) Protocol. Here all readers operate in parallel, so the time taken to read all tags depends on the reader that has to read the maximum number of tags, say reading $m_{\max}$ tags. Thus using our earlier analysis of time taken to read a certain number of un-identified tags, the time taken by BLSync protocol can be formally expressed as

$$t_{\text{BLSync}}(m) = e \cdot m_{\max} + m, \tag{6}$$

where the second term is the slots for acknowledgments as for the sequential scheme since the acknowledgments are still sent sequentially by all readers.

The value of $m_{\max}$ depends on the overlap between readers in terms of tags as well as the distribution of tags in the area. We begin by analyzing the effect of overlap factor for a uniform distribution of tags in the area.

4.3.1. Effect of Overlap Factor. In the BLSync protocol, tags are queried simultaneously by multiple readers, and tag responses are received by any reader which has the tag in its interrogation field. We need to characterize this effect of overlap factor in terms of the number of tags responding to each reader.

Now if α be the fraction of overlapped tags among m tags with n readers, which we refer to as *overlap factor*, then the expected number of tags per reader, and hence $m_{\max}$, is $(m + \alpha\ m\ (n-1))/n$, assuming each reader has similar number of tags in its range due to a uniform distribution of tags in the area. Formally, α can be expressed as $(\sum_{i=1}^{n} m_i - m)/(m(n-1))$, that is the ratio of over counted tags (when each reader counts tags in its range) to product of original m tags and number of readers less 1.

Thus, from (6) we have

$$t_{\text{BLSync}}(m) = \frac{e(m + \alpha\ m\ (n-1))}{n} + m \tag{7}$$

with $0 \le \alpha \le 1$. The lower bound comes about for cases of no overlap between reader ranges. Even if there are no overlapped tags (which is a function of from how far tags respond to a reader), RFID readers use high powers and interfere over large distances even though they read over only a few meters. Thus, even with zero overlap, readers still cannot operate at the same time for the scenario considered as explained in Section 2. The upper bound is the value of α when there is complete overlap among the tags under each reader. We consider the no overlap scenario ($\alpha = 0$) even though by definition the reader-reader collision problem occurs only with overlapped tags. But, as explained in Section 2, it is not easy to plan for nonoverlapping readers due to the presence of active tags. Thus, without knowledge of the types of tags in the area, the readers (if not using BLSync) will need to operate in a sequential fashion at all times.

The performance benefits of using the BLSync protocol compared to FSASeq depend on the value of α. For the case where there is complete overlap, $t_{\text{BLSync}} = e\ m + m$ (where α = 1) which is the same as that of the FSASeq protocol. For the case of zero overlap ($\alpha = 0$), $t_{\text{BLSync}} = e(m/n) + m$. In general, using (5) and (7), the performance gain fraction G is given by

$$\begin{aligned} G &= \frac{(em + m) - (e(m + \alpha\ m\ (n-1))/n + m)}{em + m} \\ &= 1 - \frac{e(1 + \alpha(n-1)) + n}{n(e+1)} \end{aligned} \tag{8}$$

The plot in Figure 9 shows the variation in performance gain with respect to α and n. G is seen to increase with increase in values of n and decreases with increase in value of α, the overlap factor. Note that increase in number of readers n with α constant denotes the case where the given conflict area, where readers cannot normally operate together, is expanding (if area does not expand, increasing n will increase α), requiring more readers.

4.3.2. Effect of Tag Distribution. Next we consider the case where instead of varying overlap factor, the distribution of tags varies and is not a uniform distribution anymore but a geometric distribution. We will look at the case of $\alpha = 0$ only for our analysis here; the effect of varying overlap factor along with nonuniform tag distributions is studied in detail through simulations in the following section.

The layout consists of multiple cells as shown in Figure 12(b). The probability of a tag being in a cell is geometrically distributed with parameter β, $0 \le \beta < 1$. Ordering cells from the left, with p being the fraction of all tags that are in cell 1, the expected number of tags in cell 1 is $p\ m$, βpm in cell 2, $\beta^2 pm$ in cell 3, and so on. The probability p can be calculated as $(\beta - 1)/(\beta^n - 1)$ since the sum of tags in all cells equals m. A low value of β indicates that cell 1 is likely to have a large fraction of the tags. A high value of β implies that the tags tend to be more uniformly distributed among the cells. The number of tags in a cell are randomly deployed within the cell.

Now the maximum number of tags in any cell, $m_{\max}$ is pm with $\alpha = 0$. Thus, time to read all tags by BLSync is

$$t_{\text{BLSync}} = epm + m, \tag{9}$$

obtained analogously to our earlier analysis of the effect of overlap factor.

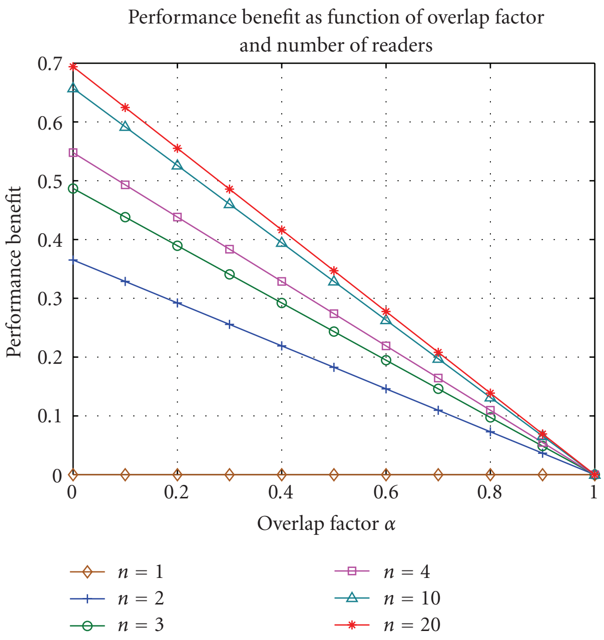


Figure 9: Performance gain through numerical evaluation in terms of decreased tag reading time with BLSync for different values of n and overlap factor α.

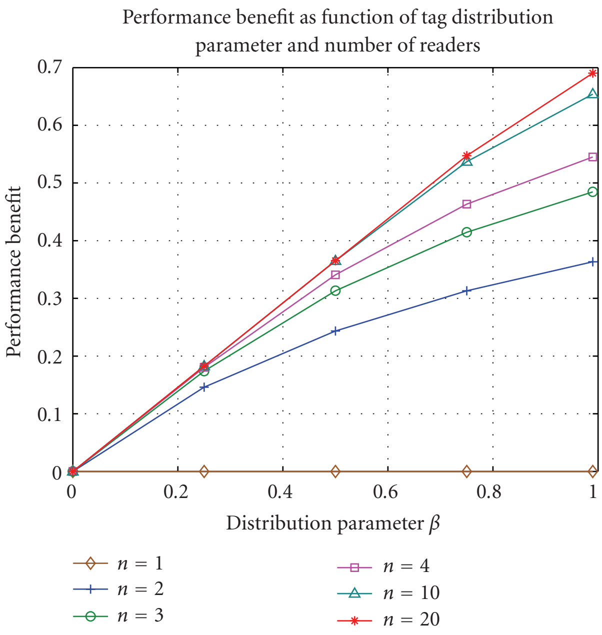


Figure 10: Performance gain through numerical evaluation for different values of n as a function of β with $\alpha = 0$.

The performance benefit G now can be characterized as

$$G = \frac{em + m - (epm + m)}{em + m} = 1 - \frac{ep + 1}{e + 1}. \qquad (10)$$

Figure 10 shows the value of G for different values of β for multiple values of n. It can be seen that for $\alpha = 0$, the tag distribution parameter has a similar effect as seen in the uniform distribution case for varying overlap factors. The greatest benefit of BLSync happens when the tags are uniformly distributed across all cells.

5. Benefits of BLSync in Realistic Settings

In this section, we intend to validate our analysis of the previous section by simulating the time required to read m tags in an area covered by n readers that conflict with each others operation. Note that these simulations do not use any analytical results of the previous section—these are meant to be an *independent* evaluation of the protocols to reinforce our analytical results. The simulator was written in C++, and the operation of frame slotted Aloha was the main functionality provided. Tags were made to randomly select slots in reader query frames over multiple rounds until each of them was identified. Both the FSASeq and BLSync protocols were simulated through sequential and parallel operations of Frame Slotted Aloha (FSA) following respective protocol guidelines presented previously. We consider two realistic tag layout settings: a uniform distribution of tags to study the effect of overlap factor on tag reading time and a geometric distribution of tags to study the effect of both the overlap factor and tag distribution on tag reading time. These can be considered realistic as one setting represents a scenario where tags could be randomly found in different locations within the interrogation zone with equal probability, while the second setting considers a possible clustered layout with tags present in groups. Varying the geometric parameter further allows consideration of various levels of clustering that may be encountered.

For the simulations, our goal was to study a moderately sized deployment (an assumption explained earlier) with readers colocated, covering an area like a conveyor belt. Each data point of the results shown represents the mean of 500 runs with 95% confidence intervals shown. The number of tags in the area was kept fixed at 50. More than the number of tags, the important parameter was the overlap factor for the study under consideration. Depending on the degree of overlap, some readers had most of these 50 tags in their range which is a high enough number at a single snapshot in time. We also present results for the case where guard times are employed to handle time synchronization errors when multiple readers operate using BLSync. This takes into account more practical issues in the operation of the BLSync protocol.

5.1. Uniform Tag Distribution. In this case, we are concerned with studying the effect of overlap factor on tag reading time. Figure 11 shows the benefits of using the BLSync protocol as opposed to the FSASeq protocol for $n = 5$ and $n = 10$. The simulation results demonstrate benefits that are about 5%–10% less (for small values of α) than the analysis since the latter did not account for border effects (fewer tags in range) that arise in the two end readers in Figure 12(a). For larger values of α ($\alpha \geq 0.5$), the performance benefit is better than what was predicted by our analysis. We explain the reason for this in the next subsection. It can be seen that the

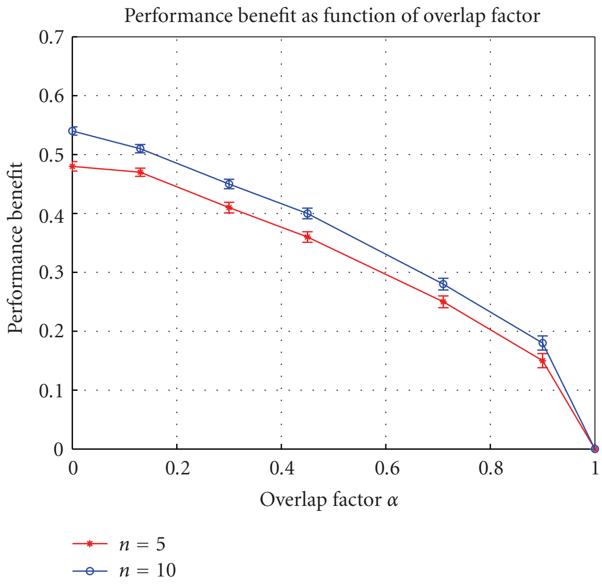


Figure 11: Performance gain through simulation for $n = 5, 10$ and overlap factor α.

benefit of using BLSync drops off slowly for overlap factors till very close to 1. Moreover, as shown previously in the analysis, there is a noticeable improvement in performance with increase in number of readers employed for the same overlap factor.

5.2. Geometric Tag Distribution. To study the effect of a nonuniform distribution of tags, we consider a scenario as shown in Figure 12(b) where there are multiple equal-sized cells with a reader in the center of each cell. Figures 13 and 14 show the performance benefits of using BLSync over FSASeq for $n = 5$ and $n = 10$, respectively. The geometric distribution parameter β is varied from 0 to 0.99, simulating conditions varying from a highly clustered cell to a uniform distribution of tags in all cells. The overlap factor could not be reduced to 0 for this scenario, since for some cases, there exist tags that are not covered by any reader. For large values of β, the results obtained are similar to that obtained for a uniform distribution of tags. For smaller values of β, however, the advantages of small overlap factors have diminished greatly. The concentration of tags in few cells for small values of β increases the maximum number of tags in the range of a reader among all readers making that reader a bottleneck reducing the benefits of parallel operation.

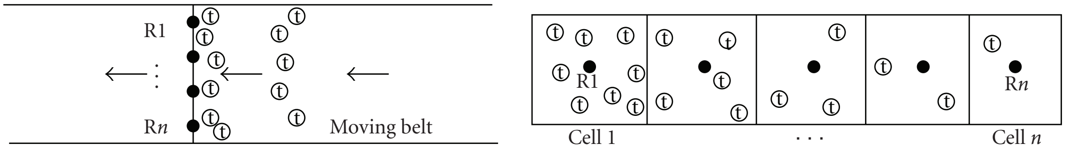
Another surprising aspect is that for $\beta = 0$, which implies that all tags are in one cell, the performance benefit is not zero as expected. We found the reason to be the help (which we call the *assist effect*) the bottleneck reader receives from neighboring readers who could put any of their mutually overlapped tags to sleep if the tag suffered no collisions at these readers. For example, in Figure 6, R2 assists R1 in round one by reading the mutually overlapped tag t4. This assistance could not be utilized in the FSASeq scheme where each reader operated on its tags before another reader began operations. In the BLSync protocol, overlapped tags enjoy a better chance of being read and acknowledged since they could be read in the frames of multiple readers at the same time. This results in all tags being read faster compared to the original protocol even if the bottleneck reader has to read the same number of tags as that of a reader in the FSASeq protocol. The assist effect plays a part not only for $\beta = 0$ or $\alpha = 1$, but in all scenarios where there are overlapped tags.

5.3. Extension of Results with Guard Time. The protocol analyzed so far was simple and basic in nature, assuming time synchronization will not be an issue since the system is centralized with the coordinator able to synchronize all other readers through their Wireless LAN interfaces. However, it has been shown in previous work that more than propagation delay, other components like operating system delay can contribute significant errors in synchronization. Hence we extend our protocol to use a guard band of 2δ for each bit received, where δ is the maximum possible synchronization error. It was reported by the authors of [22] that a one-hop synchronization of $1.5\,\mu$s is possible with wireless sensor devices. This is a bit harder to achieve in a Wireless LAN since the protocol stack is not as thin as that of a wireless sensor, and hence fine grained timestamping is more difficult. Nevertheless, the authors of [23] have experimented with Wireless LAN radios and have achieved synchronization with errors within $15\,\mu$s with older work reporting around $100\,\mu$s as possible [24].

Since this guard time is only required for bits transmitted by a reader, the time penalty depends on the number of bits transmitted by readers. As mentioned before, the main information sent by readers to tags is frame size and session numbers and slot marker bits sent by readers to count down slots. Thus, if C bits are transmitted for all frame fields and $t_{\text{BLSync}}(m)$ bits to count down all slots required for reading each tag in its field, a sum of $C + t_{\text{BLSync}}$ bits will be sent by each reader for a session where it reads all tags in its field. With a $2\delta\,\mu$s guard band, we can expect the total time to read all tags with specified guard band to be $t_{\text{BLSync}-G}(m) = 2\delta(C + t_{\text{BLSync}}(m))$. In terms of additional slots to read all tags (additional slots to read all tags are mathematically the same as using larger slots taking into account the guard time in terms of total time to read all tags), a reader would incur a delay of $\lceil 2\delta(C + t_{\text{BLSync}})/t_{\text{slot}}$ where t_{slot}, is the slot duration in milliseconds and δ specified in milliseconds as well.

Figure 15 shows results obtained by including the guard time for the uniform distribution case for $n = 5$ only. Conservative values of $t_{\text{slot}} = 1$ms (typical for 100 bit tag IDs and tags communicating at 100 kbps) and $C = 50$ were chosen. Values of δ were varied from 0 to $100\,\mu$s in steps of $20\,\mu$s to cover a wide range of possible values of clock synchronization, even though as mentioned above, $15\,\mu$s has been shown possible.

The results show that for large δ, there is a penalty now for the BLSync protocol over FSAloha when α is large; the additional delay is due to incorporation of guard time with very small benefits of parallelism when overlap is large. For other values of overlap factors and δ (which we believe will be the more common case), there is again a significant performance benefit by using BLSync and



(a) (b)

Figure 12: Simulation Scenarios. (a) Readers operating across the width of a moving belt containing tags. The readers are spaced equally apart with the width of belt deciding overlap between readers. The tags are uniformly distributed across the width of the belt. (b) Multiple cells with a reader assigned to the center of each cell. Each group of tags in a cell could be visualized as those on a mobile object, say a box or truck, that is replaced by another and so on. The number of tags in a cell has a geometric distribution starting from Cell 1. The tags in each cell are uniformly distributed.

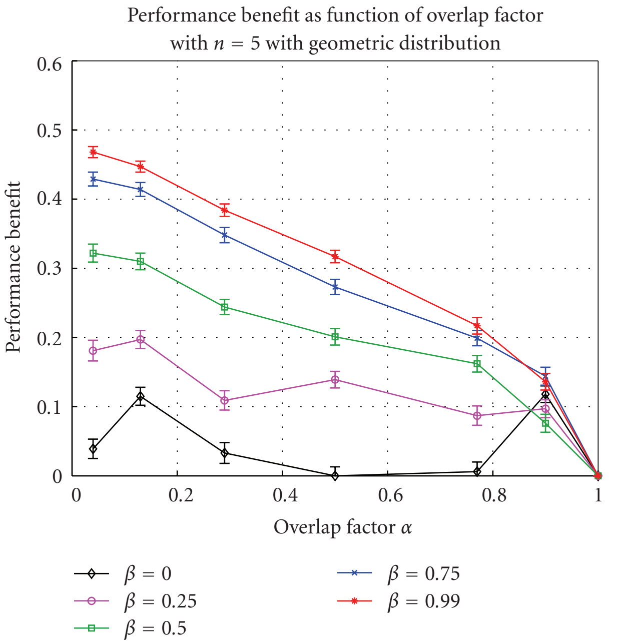


Figure 13: Performance gain through simulation for $n = 5$ for different values of geometric distribution parameter β as function of overlap factor, α.

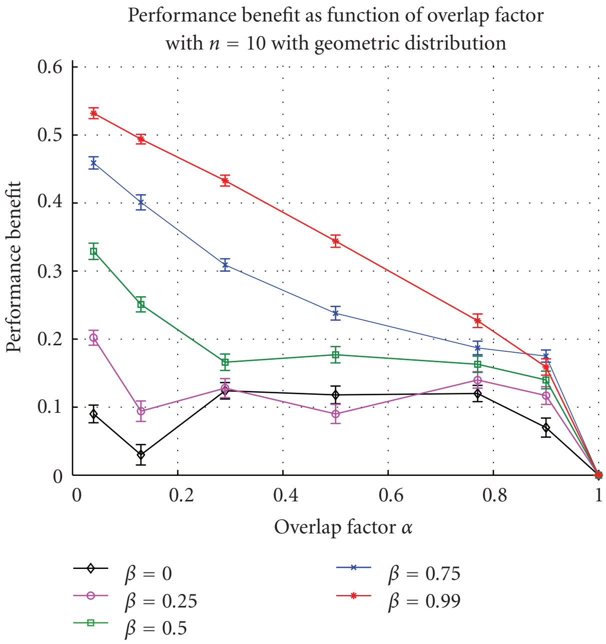


Figure 14: Performance gain through simulation for $n = 10$ for different values of geometric distribution parameter β as function of overlap factor, α.

only small reduction due to the addition of guard time. It is reasonable to expect similar results for the geometric distribution case as well. Thus, the advantages of parallel operation easily outweigh that of sequential operation even if adequate allowances are made for time synchronization constraints; for most cases greater than 25%–30% reduction in tag reading delay is possible. With improvements in hardware technology enabling tighter time synchronization in the future, we expect the performance benefits to approach those of perfect time synchronization ($\delta = 0$).

5.4. Discussion. During step 4 of the BLSync protocol in Algorithm 1, it was mentioned that each reader finds the tags remaining (to be identified) in its interrogation zone by subtracting the tags it identifies in the previous round from its original estimate. A reader is, however, unable to account for the tags read in its range by the assist effect by other readers, since it is not possible to know which of the un-identified tags were within its zone. Taking account of these tags results in more efficient frame sizes to be used in the FSA process. So, step 4 needs to be modified such that each reader reestimates the tags yet to be identified in its range every round of a read cycle. This can be done either by a short estimation procedure before each round (similar to that done in [16]) or by using the number of responses in the next round to make the estimate for the following rounds. We have used the latter approach in our simulation plots due to the smaller overhead each round, in spite of the delay of one round to account for tags read by the assist effect. Note that Figures 11, 13, and 14 already take into account this modification described for the BLSync protocol.

6. Analysis of Assist Effect

Here we present an analysis of the expected number of tags that are read through the assist effect in the BLSync protocol.

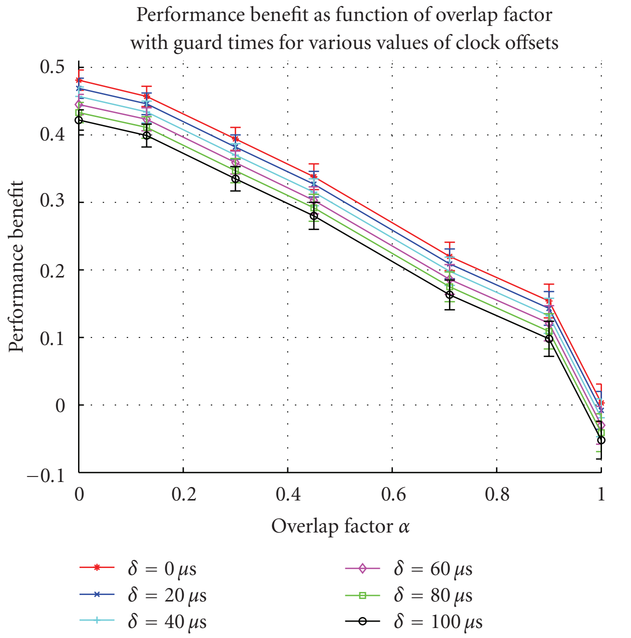


Figure 15: Performance gain through simulation for $n = 5$ for varying overlap factor α with guard times based on different δ.

This was not accounted for in our analysis in Section 4. With the protocol performance already evaluated through simulations in Section 5, the main idea of the analysis in this section is to give deeper insight into the reasons for the assist effect. We will consider the uniform distribution case only. Let κ be the reader with maximum tags in its interrogation zone. We will analyze how many of κ's tags are read on average by assistance from other readers in the first round of querying, denoted as $\overline{a}$. For further rounds, the analysis can be extended similarly taking into account the remaining tags after round 1.

Now, $\overline{a} = p_a \times$ No. of tags in range of, where p_a is the probability of a tag, say T, within range of κ receiving assistance. Consider

$$\begin{aligned} p_a &= \Pr(\text{A tag } T \text{ of } \kappa \text{ is assisted by at least one reader}) \\ &= 1 - \Pr(\text{No assistance from any reader for } T) \\ &= 1 - \prod_{j=1, j \neq \kappa}^{n} q_j, \end{aligned} \tag{11}$$

where q_j is the probability that the tag T does not receive assistance from reader j.

Thus, we can now reduce the problem of finding p_a to that of finding q_j for each reader $j, j \neq \kappa$. Let $O(\kappa, j)$ denote the set of tags that are in the overlapped region of readers κ and j, and, M_j denote the set of tags in the range of any reader j. Then,

$$\begin{aligned} q_j = \Pr(T \in O(\kappa, j)) & \\ \times \Pr\Big(& T \text{ collides with at least one tag} \\ & \text{in } \{M_\kappa - O(\kappa, j)\}\Big) \\ \times \Pr\Big(& T \text{ does not collide with any tag} \\ & \text{of } \{O(\kappa, j) - T\}\Big) \\ \times \Pr\Big(& T \text{ does not collide with any tag} \\ & \text{of } \big\{M_j - O(\kappa, j)\big\}\Big). \end{aligned} \tag{12}$$

The intuitive explanation of the above equation is that for any tag $T \in M_\kappa$ to get assistance from any reader j, it must (i) be in the overlapped region of both κ and j and (ii) must collide with at least one tag in $M_\kappa - O(\kappa, j)$, since if this does not happen, the tag will be read anyway and assistance from other readers is not required at all. (iii) The collision in (ii) should not happen with other tags in $O(\kappa, j)$ (and hence the minus $O(\kappa, j)$ term in (ii)) since these tags reply at the same slots in κ and j and cannot be separated from each other, and (iv) T should not collide with any tags in $M_j - O(\kappa, j)$ so that it is read by j, and hence, receives assistance from j.

The first term in the product, the probability of overlap, p_o, can be calculated from the layout under consideration. The remaining terms can be computed based on the size of $O(\kappa, j)$ and M_j, for all j. Let k be the size of M_κ and thus is the frame size to be used by all readers. Also, based on the assumption of a uniform tag distribution among all readers, $|M_j| = k$. Thus, for a given j, q_j can be written as

$$\begin{aligned} q_j &= p_o(j) \times \left[1 - \left\{\frac{k-1}{k}\right\}^{|M_\kappa - O(\kappa, j)|}\right] \\ &\quad \times \left\{\frac{k-1}{k}\right\}^{|O(\kappa, j) - 1|} \times \left\{\frac{k-1}{k}\right\}^{|M_j - O(\kappa, j)|} \\ &= p_o(j) \times \left[1 - \left\{\frac{k-1}{k}\right\}^{|M_\kappa - O(\kappa, j)|}\right] \times \left\{\frac{k-1}{k}\right\}^{|M_j - 1|}. \end{aligned} \tag{13}$$

This value of q_j can be substituted in the equation for p_a above to find its value. This will then give us the value of $\overline{a}$ we are looking for.

The accuracy of our analysis can be ascertained by comparing it to simulations performed on the assistance received for the first round of queries only. Figure 16 shows this comparison for the layout in Figure 12(a). It can be seen that the analysis matches the simulation results quite well. Also the assist effect increases slightly with increase in number of readers with overlap factor held constant. The benefit of assist effect on Figures 13 and 14 is to reduce the rate of decreasing performance benefit with increasing overlap factor.

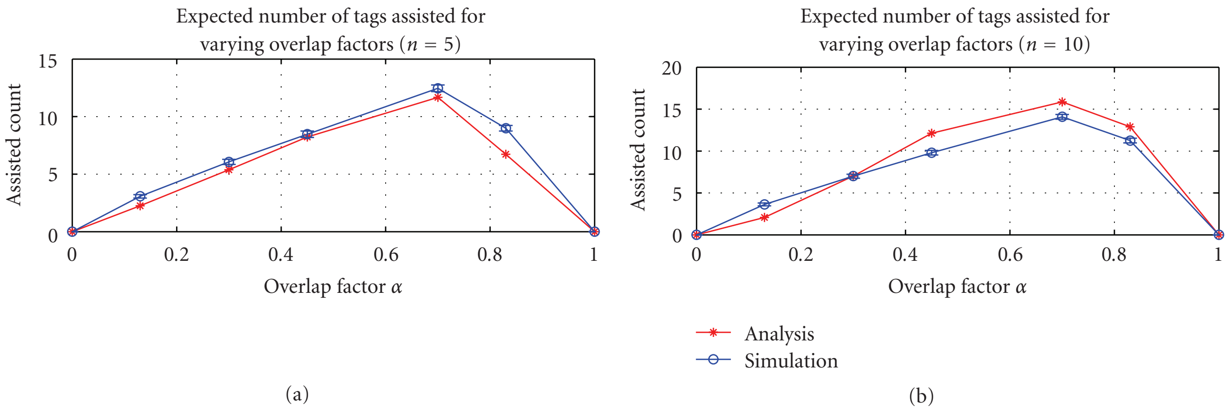


FIGURE 16: Comparison of the assist effect for the first round of queries for $n = 5, 10$ as a function of varying overlap factor, α.

The inclusion of the tags read through the assist effect in our earlier analysis in Section 4.3 will make it more accurate. The throughput per frame will now change from $(1/e)m$ to $((1/e) + p_a(i))m$, where $p_a(i)$ is the probability of a tag getting assistance in querying round i. Further analysis on the number of rounds to read all tags as done in Section 4.1 will give required analytical expressions for the time required to read all tags through BLSync protocol.

7. Conclusions and Directions for Future Work

We presented the BLSync MAC protocol for identifying RFID tags in a multireader scenario. The protocol takes advantage of the common goal of all readers to identify tags in their interrogation zone by using the same bit level queries for all readers along with synchronized timing to solve the reader collision problem. A centralized controller enables the reader synchronization through a Wireless LAN network. The resulting parallel operation provides significant reductions in tag reading delay for scenarios with low to moderate amounts of overlapped tags in the range of readers with all tag distributions except highly clustered ones.

To operate the BLSync protocol in distributed settings, there needs to be an entity that facilitates the coordination among readers to synchronize their queries. This can be done through a leader election protocol (using WLAN network) among the readers [25]. The leader acts in a capacity similar to that of the controller in the centralized case. It synchronizes the time as well as the queries of the readers. The comparatively larger operating range of WLAN technologies coupled with the dense deployment of readers allows the leader to oversee the network with very few hops of communication. Recent studies have shown that very fine grained time synchronization is possible in such settings as well[26]. Considering a relatively static scenario, an elected leader can maintain its status for long periods of time without re-election. Due to the incentive for all readers to cooperate, this distributed solution will be stable as well.

In this paper we concentrated on applications that only require identification of tags such as those specified for EPC Gen1-Class 0,1 and EPC Gen2-Class 1 tags. This work could also be extended to applications which also require writing to tags. A similar synchronized approach can be used to write to tags if the data is the same. Otherwise, the BLSync protocol needs to be modified to allow all readers to operate individually for writing data and then resuming synchronized operation, which would decrease the benefits of BLSync protocol due to reduced parallelism. Other efforts could include adaptation of BLSync to work with EPC standard protocols and associated performance evaluation in practice, moving away from the fundamental investigation done in this work.

Under practical settings, multipath and other effects on the wireless channel may impact the synchronization of signals among readers. However, the relatively small read range of RFID readers coupled with emerging techniques to reduce multipath and other effects through power control and/or directivity of reader transmissions. The combination of guard bands coupled with such techniques should mitigate possible negative impacts on protocol performance. It would be useful to develop a proof-of-concept system running the protocol to test its performance in such nonideal settings. The results achieved in this paper under more idealistic settings do suggest that there is enough performance improvements to take the next step towards more practical studies.

Finally, more work needs to be done at the physical layer to realize diversity combining at tags. Techniques commonly used at the physical layer in applications like cooperative relays where the same information from multiple sources has to be correctly decoded at the receiver could be a starting point. Special attention must be given to the complexity of any receiver design as these are for RFID tags, the passive type of which is very resource constrained.

Acknowledgment

This paper was supported in part by the U.S. Army Research Office under DEPSCoR ARO Grant W911NF-08-1-0256.

References

[1] N. Vaidya and S. Das, "RFID-based networks-exploiting diversity and redundancy," Tech. Rep., June 2006.

[2] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, John Wiley & Sons, New York, NY, USA, 2003.

[3] C. Law, K. Lee, and K. Y. Siu, "Efficient memoryless protocol for tag identification (extended abstract)," in *Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp. 75–84, ACM, 2000.

[4] B. Zhen, M. Kobayashi, and M. Shimizu, "Framed ALOHA for multiple RFID objects identification," *IEICE Transactions on Communications*, vol. 88, no. 3, pp. 991–999, 2005.

[5] K. S. Leong, M. L. Ng, A. R. Grasso, and P. H. Cole, "Synchronization of RFID readers for dense RFID reader environments," in *Proceedings of the International Symposium on Applications on Internet Workshops*, pp. 48–51, January 2006.

[6] D. W. Engels and S. E. Sarma, "The reader collision problem," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 641–646, October 2002.

[7] J. Waldrop, D. W. Engels, and S. E. Sarma, "Colorwave: an anticollision algorithm for the reader collision problem," in *Proceedings of the International Conference on Communications (ICC '03)*, vol. 2, pp. 1206–1210, May 2003.

[8] J. Ho, D. W. Engels, and S. E. Sarma, "HiQ: a hierarchical Q-learning algorithm to solve the reader collision problem," in *Proceedings of the International Symposium on Applications on Internet Workshops*, pp. 88–91, January 2006.

[9] Z. Zhou, H. Gupta, S. Das, and X. Zhu, "Slotted scheduled tag access in multi-reader rfid systems," in *Proceedings of the IEEE International Conference on Network Protocols*, pp. 61–70, Beijing, China, 2007.

[10] V. Deolalikar, M. Mesarina, J. Recker, and S. Pradhan, "Perturbative time and frequency allocations for RFID reader networks," in *Proceedings of the EUC Workshops*, 2006.

[11] EPCglobal, "Epc radio-frequency identity protocols class-1 generation-2 uhf rfid protocol for communications at 860 mhz-960 mhz version 1.0.9," 2005.

[12] D. G. Brennan, "Linear diversity combining techniques," *Proceedings of the IRE*, vol. 47, no. 1, pp. 1075–1102, 1959.

[13] S. Valentin, D. H. Woldegebreal, T. Volkhausen, and H. Karl, "Combining for cooperative WLANs—a reality check based on prototype measurements," in *Proceedings of the IEEE International Conference on Communications Workshops (ICC '09)*, pp. 1–5, June 2009.

[14] A. B. Sediq and H. Yanikomeroglu, "Diversity combining of signals with different modulation levels in cooperative relay networks," in *Proceedings of the 68th IEEE Vehicular Technology Conference (VTC '08)*, pp. 1–5, September 2008.

[15] Impinj, "Antenna orientation sensitivity," 2010, http://www.impinj.com/applications/app-notes.aspx.

[16] M. Kodialam and T. Nandagopal, "Fast and reliable estimation schemes in RFID systems," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM '06)*, pp. 322–333, ACM, September 2006.

[17] "Commission FC. Part 15—radio frequency devices," August 2006, http://www.fcc.gov/oet/info/rules/.

[18] F. C. Schoute, "Dynamic frame length Aloha," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 565–568, 1983.

[19] H. Vogt, "Efficient object identification with passive RFID tags," in *Proceedings of the International Conference on Pervasive Computing*, pp. 98–113, 2002.

[20] S.-R. Lee, S.-D. Joo, and C.-W. Lee, "An enhanced dynamic framed slotted ALOHA algorithm for RFID tag identification," in *Proceedings of the 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services*, pp. 166–172, July 2005.

[21] D. Bertsekas and R. Gallagher, *Data Networks*, Prentice-Hall, Upper Saddle River, NJ, USA, 2nd edition, 1992.

[22] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, pp. 39–49, November 2004.

[23] J. Kannisto, T. Vanhatupa, M. Hnnikinen, and T. Hmlinen, "Precision time protocol prototype on wireless lan," in *Proceedings of the 11th International Conference on Telecommunications (ICT '04)*, vol. 3124 of *Lecture Notes in Computer Science*, pp. 1236–1245, Springer, 2004.

[24] M. Mock, R. Frings, E. Nett, and S. Trikaliotis, "Continuous clock synchronization in wireless real-time applications," in *Proceedings of the 19th IEEE Symposium on Reliable Distributed Systems (SRDS '00)*, pp. 125–132, October 2000.

[25] A. S. Tanenbaum and M. V. Steen, *Distributed Systems: Principles and Paradigms*, Prentice-Hall, Upper Saddle River, NJ, USA, 2001.

[26] R. Solist, V. S. Borkar, and P. R. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," in *Proceedings of the 45th IEEE Conference on Decision and Control (CDC '06)*, pp. 2734–2739, San Diego, Calif, USA, December 2006.