



A distributed multi-channel reader anti-collision algorithm for RFID environments



Haidar Safa*, Wassim El-Hajj, Christine Meguerditchian

American University of Beirut, Department of Computer Science, P.O. Box: 11-0236, Riad El-solh, Beirut 1107 2020, Lebanon

ARTICLE INFO

Article history:

Received 15 July 2013

Received in revised form 9 December 2014

Accepted 22 January 2015

Available online 3 February 2015

Keywords:

RFID

Tag

Reader

Collision

ABSTRACT

In Radio Frequency Identification environments, several readers might be placed in the same area to scan a large number of tags covering a wide distance range. The placement of the RFID elements may result in several types of collisions. This paper proposes a multi-channel algorithm to solve the reader collision problems in a dense or sparse RFID environment. We adapt a distributed approach that avoids the need of costly extra hardware for centralized control. In addition, the proposed approach does not require global synchronization in the RFID network. It introduces a multi-channel notification protocol to make RFID readers aware of the network resources utilization. We have evaluated the performance of the proposed approach using NS3 and compared it to several anti-collision solutions such as NFRA, Dica and McMac. Results show that the proposed algorithm reduces the time needed for tags' identification, thus increasing the rate of successful interrogations while minimizing the network overhead.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Radio Frequency Identification (RFID) started in 1973, as part of the “Automatic Identification and Data Capture” group, to replace the traditional use of bar codes. RFID enables wireless interaction over certain frequencies of RFID readers with a network system, to uniquely identify, track and capture the status of tagged objects within packages, animals or people at varying distances without the need of human intervention. As shown in Fig. 1, an RFID network is composed of four main elements: (1) RFID tags, (2) RFID readers, (3) the air interface, and (4) edge servers. Typically, RFID readers emit radio-frequency signals that RFID tags would detect if present in the reader's transmission range. RFID tags respond to the reader's queries by emitting radio waves back with the data stored in the chip [11]. In recent years, several major supply chain companies, such as Wal-Mart and Tesco, mandated the use of RFID systems in their warehouses. In this RFID environment (Fig. 1), hundreds of readers might be placed in the same area (i.e. in a building) to scan a large number of tags for a desired coverage range. Such a dense network exhibits high number of collisions that lead to reduction in data collection throughput, increase in identification delay, and degradation in network efficiency and reliability.

Three types of RFID collisions exist: (1) tag to tag collisions, (2) reader to reader interference (RRI), and (3) reader to tag interference (RTI). A tag to tag collision occurs when a reader broadcasts a message to tags, which, as a result of the message, transmit their IDs simultaneously to the reader [9]. Several tags anti-collision protocols exist and can be used to resolve tag collisions [16,30,31]. These protocols are generally ALOHA-based or tree-based protocols but they commonly focus on reducing the required time until a single reader completely recognizes the tags in the reader's identifying range.

As for the 2 reader collision problems (RRI and RTI), it is essential to differentiate between the transmission range of a reader and its interference range as shown in Fig. 2. This figure contains two readers, R1 and R2, and two tags, T1 and T2. The transmission/read range is the coverage area of the reader, which may reach 10 m when the reader is operating with an output power of 2 W [10], while the interference range is the area that the reader causes interference on, which may reach 1000 m [18]. RRI occurs when many readers are working at the same frequency within an interference range. In Fig. 2a, RRI occurs when R1 attempts to read data from T1 using a channel with frequency f_1 while R2 is trying to read data from tags in its transmission range (example T2) using the same channel with frequency f_1 . The signal sent from R2 to read T2 will interfere with the reply signal sent from T1 to R1 as shown in Fig. 2b. RRI can be avoided by having the readers operate at different frequencies or different time slots [16].

* Corresponding author.

E-mail addresses: haidar.safa@aub.edu.lb (H. Safa), wassim.el-hajj@aub.edu.lb (W. El-Hajj), ckm03@aub.edu.lb (C. Meguerditchian).

On the other hand, two types of RTI exist. The first type occurs when multiple readers, independently of the working frequency, try to simultaneously read the same tag located in their common reading range as shown in Fig. 3a. In this figure, RTI occurs when R1 and R2 attempt to read T1 simultaneously as shown in Fig. 3b. The tag T1 will not be able to decode the commands of both readers and consequently will not be able to reply. This type of collision can be avoided by having the readers operate at different time slots [16].

The second type of RTI occurs when two readers are operating at the same frequency, where a tag is located in the read range of one reader, and in the interference range of another. In Fig. 4a for example, RTI will occur when R1 and R2 attempt simultaneously to read T1 and T2 respectively using frequency f_1 . Since T1 is in the interference range of R2 and the read range of R1, both signals will reach T1 and collision will occur at the tag (Fig. 4b). This type of collision can be avoided by having the readers operate at different frequencies or at different time slots.

In this paper, which is an extension to our work in [21], we propose a new distributed multi-channel anti-collision algorithm, referred to as DiMCA, for RFID networks. The proposed DiMCA aims to solve all types of reader collisions: RRI, and two types of RTI. It is distributed and introduces a multi-channel notification protocol to distribute network resources among readers. When compared to the state-of-the-art collision avoidance protocols (NFRA, Dica and McMac), our proposed DiMCA reduced the total time needed for tag identification and increased the rate of successful interrogations in the network. The remainder of this paper is organized as follows. In Section 2, we survey the related work. In Section 3, we present our new approach. In Section 4, we evaluate the performance of the proposed approach and compare it with other well-known algorithms. Conclusion is presented in Section 5.

2. Related work

Many approaches were recently proposed to reduce the impact of RFID collisions, minimize interference, and maximize the read range [1,4,6–10,12,14,15,17,19,20,22,24–29]. ETSI 302 208 [10] is a standard that presents a regulation to govern the operation of RFID readers. It applies “Listen before talk” (LBT) or Carrier Sense

Multiple Access (CSMA). It states that prior to transmission, the interrogator must listen for the presence of another signal within its intended sub-band of transmission. The listening time comprises a fixed period of 5 ms plus a random time chosen from 0 ms to 5 ms in 11 steps. If the sub-band is free, the random time shall be set to 0 ms. The sub-band is then used for 4 s by a reader after selection and then freed for at least 100 ms. The frequency band of 865–868 MHz (UHF) is allocated for RFID deployment. The band is divided into 15 sub-bands, each spanning 200 kHz. The main disadvantage of this standard is that readers might be placed in a way where they cannot see each other according to their located angle positions, hence resulting in unsolved collisions.

EPC Class 1 Gen 2 [9] is a standardization effort, based on frame slotted aloha [24], proposed by EPC Global. It is applied to UHF and used for supply chain. It uses techniques like frequency hopping or frequency agile systems. The allocated frequency band is divided into 10 sub-bands. A reader uses only one channel, not the entire band. Readers randomly change bands every 0.4 s according to the Frequency Hopping Spread Spectrum (FHSS) technique which aims to minimize collision probabilities. Readers operate in even numbered channels whereas tag backscatter using odd-numbered channels, so the readers are left with only 5 channels available. This standard suffers from the reservation of a single channel by a certain reader in a large area, preventing others from using it. In addition, the study conducted in [22] has shown that the current EPC Class 1 Gen 2 standard, under ideal conditions, theoretically adds 10% overhead in terms of delay to the basic frame slotted aloha protocol.

An approach was proposed in [19] for synchronization among readers through a central control unit. It uses fine tuning methods, dynamic channel assignment and optimized spectrum management. All readers start listening at the same time, and then synchronously talk at the same time. The same channel is assigned for readers that are very far from each other. It also suggests reducing radiating power, which allows reducing the minimum distance between two antennas using the same channel. Some ideas have been mentioned but not applied like: reducing talking time, using sensors to turn readers on and off, using RF opaque and absorbing materials which are very expensive solutions.

Colorwave [27,28] is a distributed TDMA based algorithm, where each reader chooses one of the time slotted colors in [0,

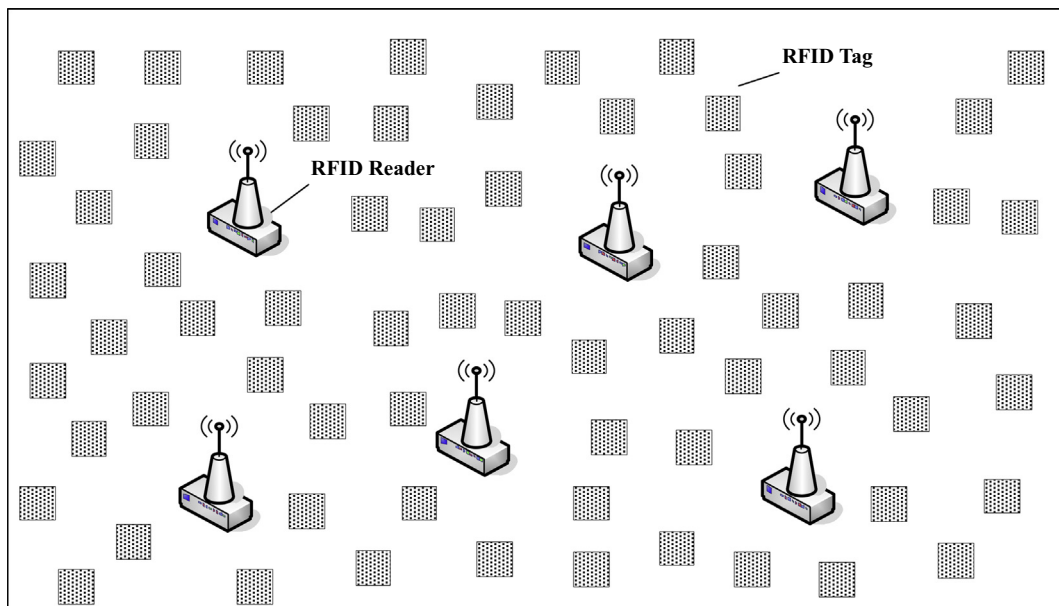


Fig. 1. An example of a dense RFID environment.

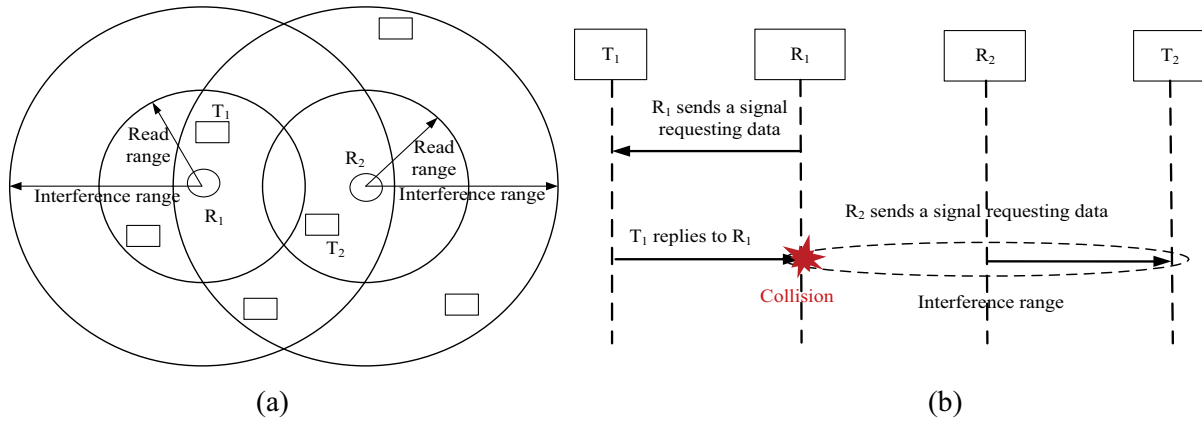


Fig. 2. (a) RRI Collision and (b) sequence diagram.

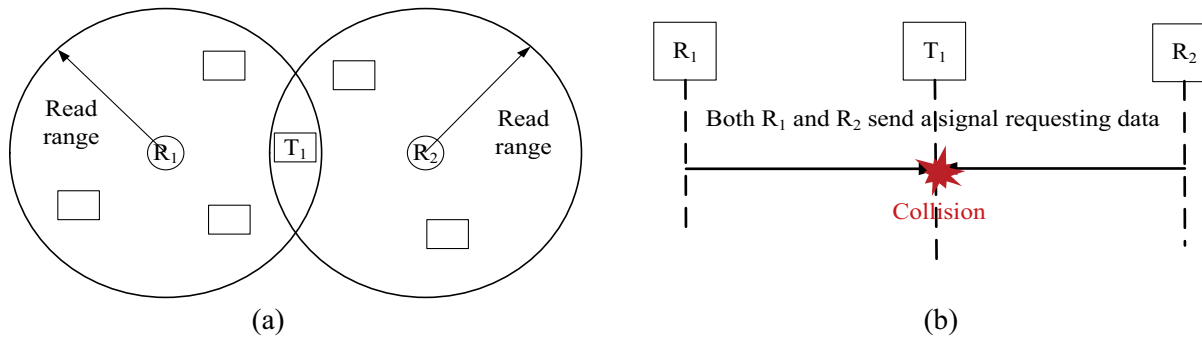


Fig. 3. (a) First type of RTI and (b) sequence diagram.

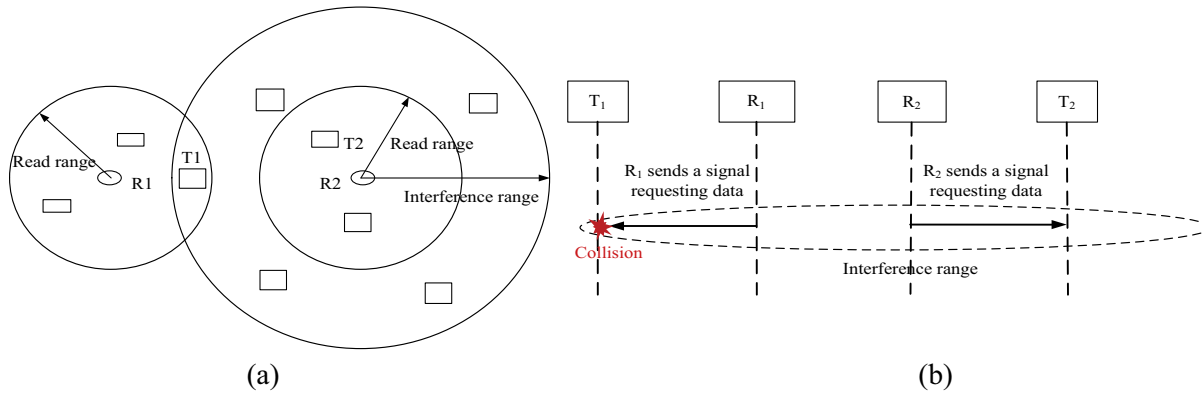


Fig. 4. (a) Second type of RTI and (b) sequence diagram.

maxColors] randomly to transmit. If the transmission collides, the transmission request is discarded and the reader randomly selects a new timeslot and sends a kick packet to all its neighbours to indicate selection of new timeslot. If any neighbour has the same color, it chooses a new color and sends a kick (small control packet) and this continues. If the percentage of successful transmission goes below certain threshold, the maxColors is incremented. While if the percentage increases beyond certain threshold, the maxColors is decremented. Since the change after a collision from the previous timeslots to a random timeslot can produce new collisions and in order to decrease the resulting next collisions, the authors in [12] proposed to introduce into collision resolution of time division protocols an additional parameter p , representing the

probability to change a timeslot after a collision. Colorwave require tight time synchronization between readers. Therefore, the overhead of time-slot reselection continuously increases when network topology is changed by reader mobility. Furthermore, Colorwave assumes that a reader is able to detect collisions in the network without being aware of a tag. However it may not be practical for a reader alone to detect the collisions that happen at the tags [4,14].

A distributed solution for collision avoidance (Dica) was proposed in [14]. In Dica, each reader repetitively contends with other readers through a control channel. The winner of the contention can start using the data channel for tag interrogation while the loser must wait till the channel is idle. In this approach, a reader

Table 1
Dense RFID anti-collision solutions.

	PULSE [4]	DAPC [6]	MCMAC [7]	Dica [14]	RAMP [15]	NFRA [8]	Proposed DiMCA
RRI	X		X		X	X	X
RTI	X	X		X		X	X
Centralized		X				X	
Distributed	X	X	X	X	X		X
Multiple data channels			X		X		X
Use of control channel	X		X	X	X	X	X
Suitable for mobile readers	X		X	X	X	X	X
FDMA			X				X
TDMA			X			X	X
CSMA	X		X	X	X	X	X

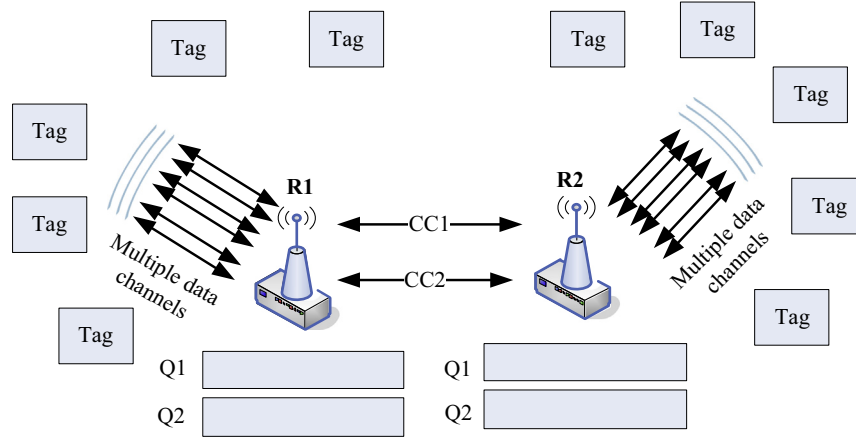


Fig. 5. Proposed architecture of the RFID network.

sends a “BRD_WHO” packet to identify operating readers in its vicinity. A “BUSY” packet is sent as a reply to the “BRD_WHO” message to indicate that a reader is currently reading tags. A “BRD_END” packet is sent to notify neighboring readers that the interrogation is finished and the channel is idle. To avoid hidden and exposed terminal problems the control channel range is set to double the transmission range of a reader. Nevertheless, Dica suffers from ignoring RRI collisions and uses only one data channel.

PULSE [4] is a CSMA based anti-collision protocol that uses separate channels for the data and control packets in the system. In this algorithm, the reader starts by waiting for T_{min} time, then contends and senses the control channel for a certain amount of time. If it is idle, it sends a beacon and starts reading. If the control channel is not idle, it waits and senses it again. During the interrogation time of a reader of its zone, it keeps on broadcasting a beacon message on a control channel which prevents other readers from contending or reading in the same time. During contention, CSMA, or reading, if a reader receives a beacon on the control channel, it backs off and starts waiting to contend again. The communication range for the control channel is made in a way that two readers having a common interference range can communicate through the control channel, and that is done by making the readers use a higher power for the transmission on the control channel than on the data channel. The disadvantage is that beacons can collide on the control channel, which creates a new collision problem. Moreover, the solution suffers from the hidden and exposed terminal problems. Reader anti-collision MAC protocol (RAMP) for Dense Reader RFID [15] improves PULSE by allowing the use of multiple data channels and a control channel. It proposes a random back-off time for accessing the control channel by multiple readers, and a channel hopping algorithm based on the density of readers in the interrogation zone and the channel utilization of the particular channel. The solution does not exactly specify how the beacons for

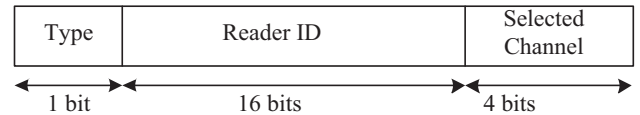


Fig. 6. Packet format.

different data channels are transmitted on the control channel. In addition beacon messages may collide.

A multi-channel MAC protocol for RFID networks (MCMAC) was proposed in [7]. MCMAC also uses a control channel to exchange control packets between readers, but unlike Dica, its communication range is set such that, any two readers that can interfere with each other on the data channel, are able to communicate on the control channel. It distributes the data channels among the readers using a random access algorithm. It defines three stages a reader must perform before starting the interrogation. During the listening stage, a reader must listen to the control channel for $T_{min} + i$ time, where i belongs to interval $[0, CW]$ where CW is the contention window. If it receives a control message, it analyzes the message to find which channel has been occupied and whether there is an idle channel to utilize. If there are idle channels, the reader selects one and sends out a control message to denote that the channel has been occupied. If there are no idle channels to use, the reader loses the cycle and waits for the next cycle. During the transmission control stage, the reader sends a control packet to notify others of the selected channel. During the reading stage, the reader keeps sending periodically control packets to neighboring readers to notify them that it is still using the channel for reading tags. This approach completely solves RRI and the second type of RTI, but does not address the first type of RTI. Therefore, two readers can be using different data channels, but still cause

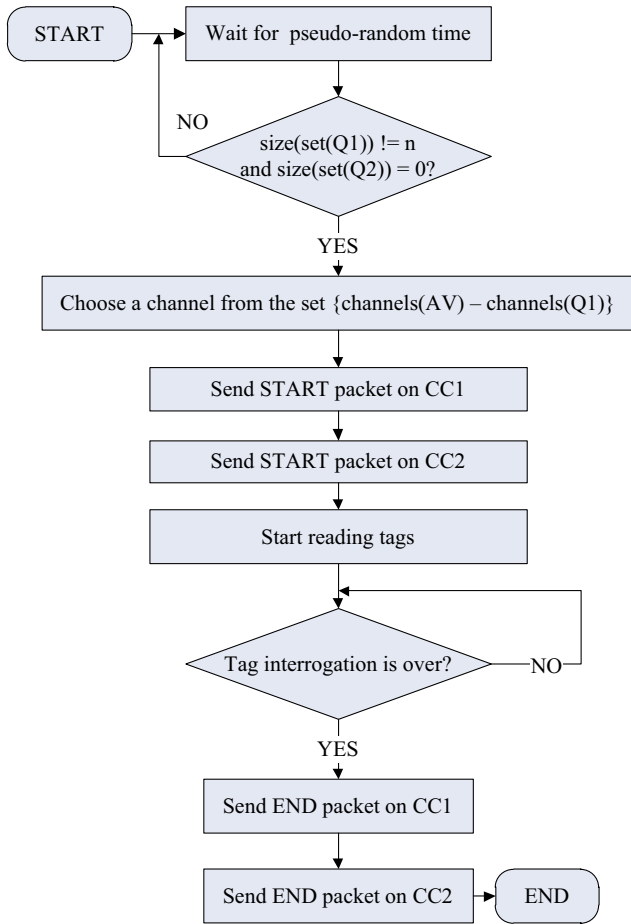


Fig. 7. Contention and interrogation of the proposed approach.

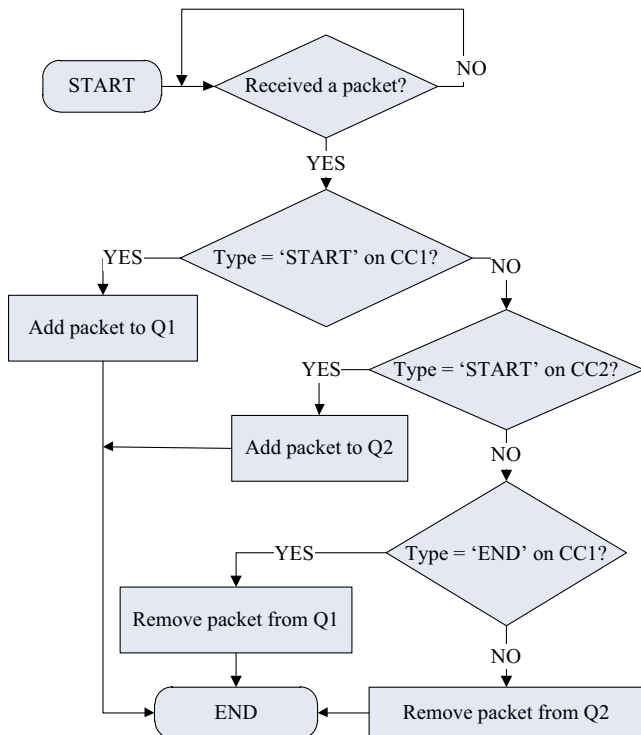


Fig. 8. Receive of the proposed approach.

collisions if they are reading the same tag simultaneously. Also, MCMAC does not address collision between control packets.

An adaptive power control with hardware implementation for wireless sensor and RFID reader networks (DAPC) was proposed in [6]. It is a decentralized solution that takes into consideration output power and signal to noise ratio by using adaptive power update and selective back-off to improve coverage. It relies on the degree of interference measured at each reader to dynamically adjust its transmission power. Its disadvantage is that it assumes that a centralized server knows the location of the readers. Also changing the output power if the location of the readers is constantly changing can be hard to manage.

A neighbor friendly anti-collision solution (NFRA) was proposed in [8]. It solves both RRI and RTI through a centralized algorithm for fixed and mobile readers. A polling server broadcasts an arrangement command (AC) which includes the range of random numbers. The readers that receive the AC generate their own random numbers. Then the server issues an ordering command (OC). Each reader compares its random number with the value in the OC. If they are the same, the reader issues a beacon to determine whether a collision occurs or not. After the beacon frames, if some readers do not detect any collisions, they send overriding frame (OF) to the neighboring readers. The OF prevents the neighboring readers from receiving the next OC from the server. The neighboring readers that do not identify the next OC due to the OF, or that detect a collision of beacons do not conduct identification of tags until the next AC. NFRA, like Dica, assumes the use of only one data channel, and it does not mention how the collision between the beacons is detected by the readers.

We compare in Table 1 the characteristics of some of the approaches found in the literature. Briefly, they can be classified as centralized or distributed, some of which use a single data channel, while others use multiple ones. Our proposed anti-collision algorithm makes use of multiple data channels for the communication between the readers and the tags and two control channels for the notification mechanism. It is distributed, hence avoiding the need of extra costly hardware for centralized control and synchronization. It is also suitable for mobile readers which are deployed for dynamic data capture replacing the fixed static readers.

3. The proposed distributed multi-channel anti-collision algorithm

In this section, we present our proposed *distributed multi-channel anti-collision* (DiMCA) strategy which solves reader collisions while reducing the *identification delay*, reducing the *collision probabilities*, and minimizing the *amount of overhead* in the network.

3.1. Basic concepts and notations

The proposed DiMCA builds a notification mechanism between the readers to make them aware of the resources, channels, and time allocations being used in the network. As shown in Fig. 5, for the communication between readers, DiMCA reserves two control channels CC1 and CC2. Readers that can avoid interfering with each other by using different frequencies can communicate on CC1, and readers that can avoid interfering with each other by using different time slots can communicate on CC2. Thus, CC1 is used to reach readers that have in their reading range some tags that are also in the interference range of the transmitter. While CC2 is used to reach readers that have in their reading range some tags that are also in the reading range of the transmitter. For the communication between the readers and the tags, multiple data channels are used. To describe further the proposed approach, we first define the following notations. Let R_i denote a reader i . For every R_i we define r_i

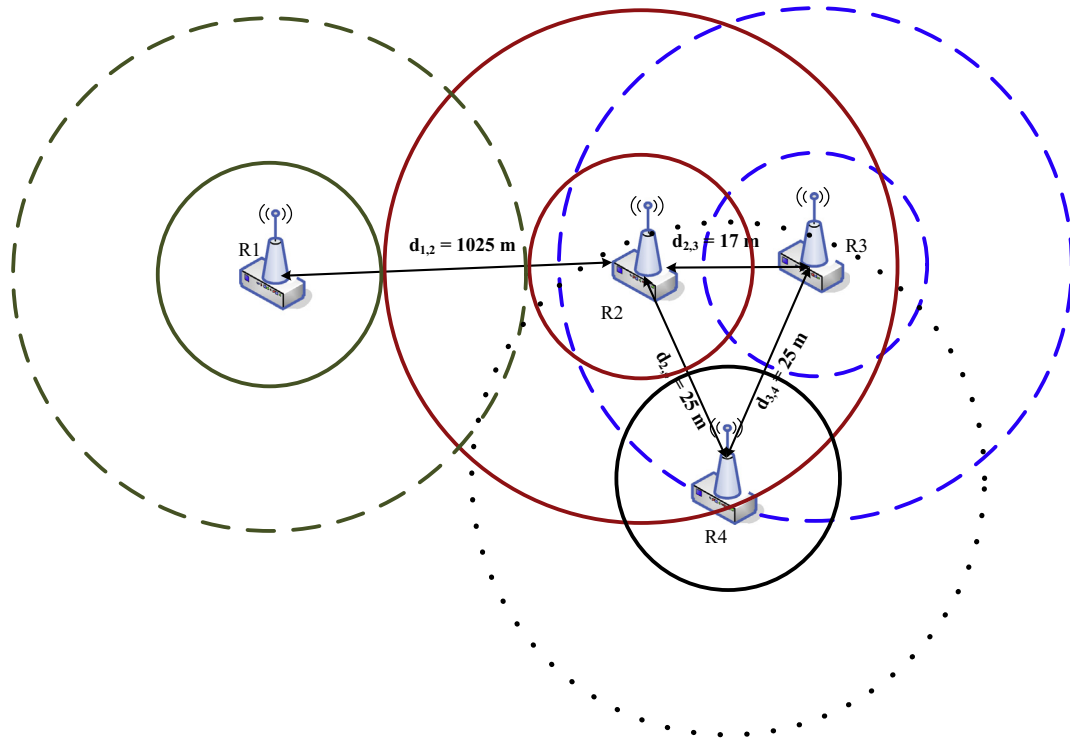


Fig. 9. Architecture of the illustrated example.

Table 2
Distance between the readers of Fig. 7.

Distance (m)	R_1	R_2	R_3	R_4
R_1	0	1025	1044	1039
R_2	1025	0	17	25
R_3	1044	17	0	25
R_4	1039	25	25	0

to be R_i 's read range (i.e., the distance range for which R_i can identify tags), I_i to be R_i 's interference range (i.e., the distance range where the R_i 's signal can affect another reader's signal), and $d_{i,j}$ to be the distance between reader R_i and R_j .

Assuming that all the readers have the same read range, then for two readers R_i and R_j , RTI occurs when:

$$d_{i,j} < r_i + r_j = 2r_i \quad (1)$$

In this case the readers R_i and R_j must operate at different time slots to avoid collisions. Both RRI and RTI occur when:

$$d_{i,j} < r_i + I_j \quad (2)$$

In this case the readers R_i and R_j must operate at different frequencies to avoid collisions. In the proposed approach, the control channel CC1 has a transmission range equal to $r + I$ to be able to reach readers that have in their interference range some tags situated in the reading range of the transmitter. The control channel CC2 has a transmission range equal to $2r$ to reach readers that have in their reading range some tags that are in the reading range of the transmitter.

Also, each reader R maintains two queues that are used to decide whether the interrogation can start or not (i.e., a reader can start sending message to identify tags in its range), and which data channel can be used. The first queue, Q1, stores information such as the IDs of the operating readers that interfere with R if the same frequency is used along with the corresponding data channels. The second queue, Q2, stores information about readers

that are currently interrogating tags and that interfere with their operation if they operate in the same time slot.

3.2. Operational flowcharts of the proposed DiMCA algorithm

Every reader R can receive on any control channel while using the data channel for tag interrogation. The control information received on CC1 will be stored in Q1 (i.e., packets received from readers that interfere with R if the same frequency is used), and the control information received on CC2 will be stored in Q2 (i.e., packets received from readers that interfere with R_j if they operate simultaneously).

The control packet's format is shown in Fig. 6. It is composed of three fields: (1) the *type*, which can be 'START' or 'END', (2) the *Reader Id*, which is the unique identifier of each reader, and (3) the *Selected Channel*, which is the channel chosen by the transmitter of the tag interrogation. During the transmission on CC2, the *Selected Channel* field is left empty, because neighbors situated at distance $2r$ from the transmitter do not need information on the selected channel. These neighbors only need to identify the readers operating during the same time slot. The *Selected Channel* field is also left empty for the *END* packets, because to remove a certain *START* packet from the queue, the reader ID is enough to identify a reader.

The proposed DiMCA algorithm is composed of two parts, the operation of the reader (i.e., Contention and interrogation) and the receiving part of the reader as shown in Figs. 7 and 8 respectively. The number of available data channels AV is considered to be n . Initially, all data channels in $AV = \{c_1, \dots, c_n\}$ are available. When a reader wants to start an interrogation, it first waits for pseudo-random time, and then starts by checking both its queues: Q1 and Q2. If the size of the set of Q1 is not equal to n (i.e., not all the data channels are being used by the neighbor readers that interfere with it on the data channel), and if Q2 is empty (i.e., none of the neighboring readers, that interfere with it is currently interrogating tags), it selects an available data channel, sends a *Start*

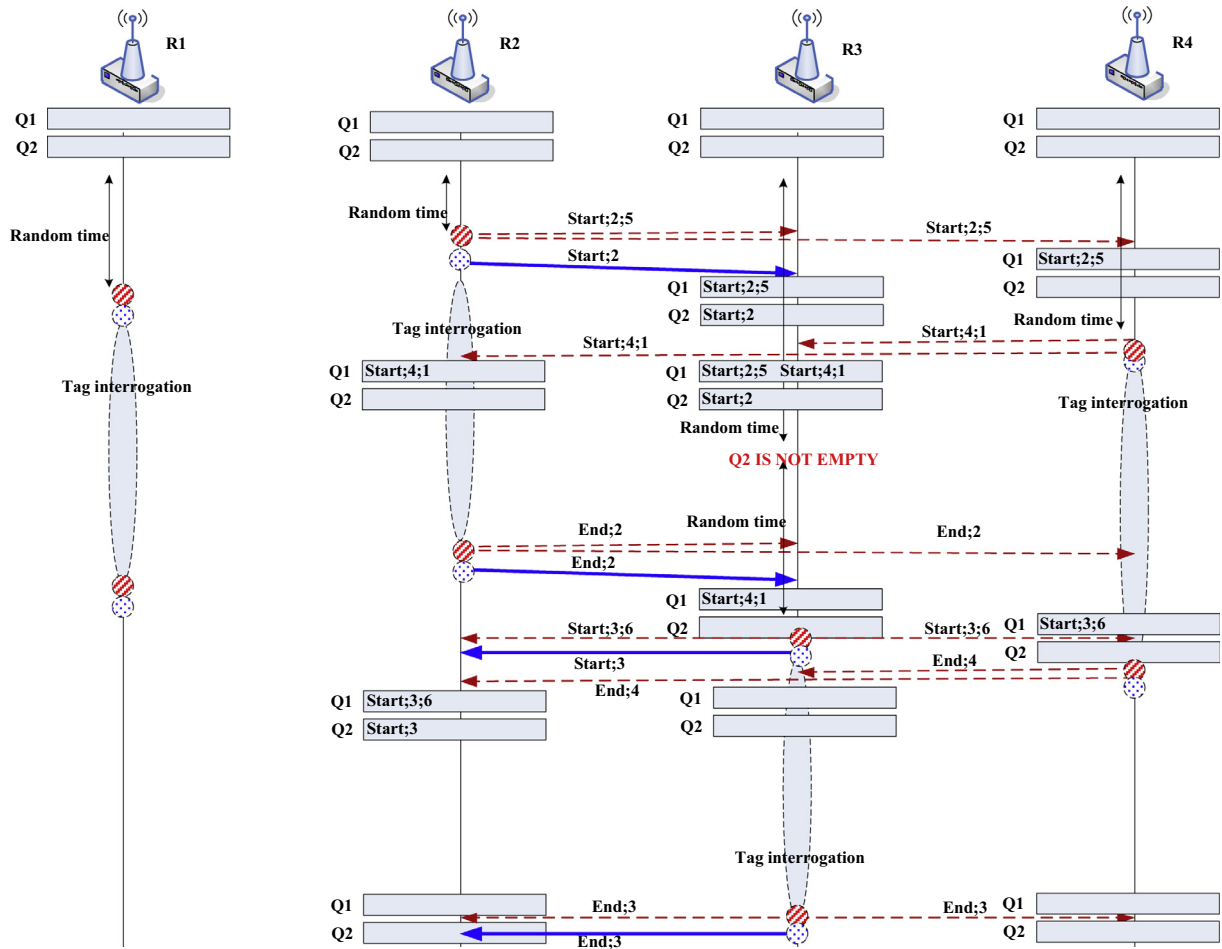


Fig. 10. Example to illustrate the algorithm.

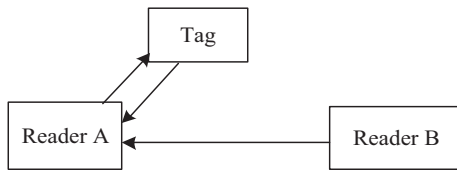


Fig. 11. Interference at reader A caused by reader B.

packet on CC1 and CC2 and starts the tag interrogation. If one of the two conditions is not satisfied, the reader waits for a pseudo-random time and tries again. During the contention and tag interrogation phase, if the reader receives a *START* control packet, it extracts the information about the reader and the data channel and queues them in the corresponding queue depending on the control channel over which it received them as shown in Fig. 8. The idea behind setting the transmission range of reader R on CC1 to $r + l$ is to notify neighbor readers causing RRI or the second type of RTI with R about its use of a certain data channel. Hence they can select a different data channel if any is available as shown in Fig. 7. On the other hand, the transmission range of CC2 is set to $2r$ to notify neighbor readers, who cause the first type of RTI with R, that R is operating and consequently they cannot operate simultaneously as shown in Fig. 8. This setting ensures the avoidance of the hidden and exposed terminal problems; i.e., readers interfering with each other can communicate on the control channels. When a reader finishes interrogating tags, it sends *End* packets on both CC1

Table 3
Simulation parameters.

Parameter	Value
Number of tags for each reader	100
Time to identify 100 tags	0.46 s
Type of the reader antenna	Omni-directional
Data rate	1 Mbps
Comparing algorithms	EPC C1G2, Dica, Mcmac, NFRA
Read range of a reader (r)	10 m
Interference range of a reader (l)	1000 m
Transmission range of CC1	$r + l = 1010$ m
Transmission range of CC2	$2r = 20$ m
Number of runs	10
<i>Sparse reader environment scenarios</i>	
Number of readers	5, 10, 15, 20, 25
Simulation range	3 km ³
Number of data channels	6
<i>Dense reader environment scenarios</i>	
Number of readers	25, 75, 150, 250, 350, 1000, 2000
Simulation range	2 km ² with a height of 20 m
Number of data channels	10

and CC2 to notify them that it is over, and the receiving readers remove the corresponding *Start* packets from their queues.

3.3. Illustrative example

To illustrate how the algorithm works, an example is shown in Fig. 9 in which four readers (R1, R2, R3, and R4) are placed at vary-

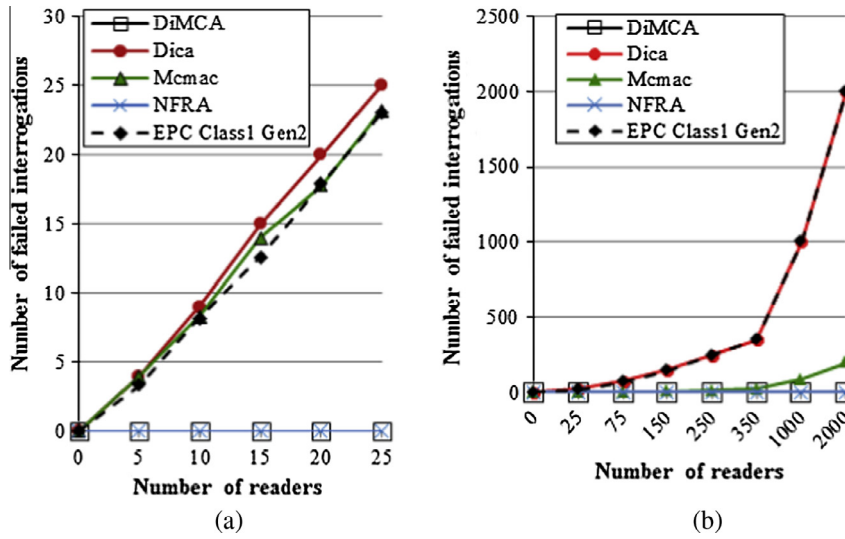


Fig. 12. Number of failed interrogations vs. the number of readers (a) sparse environment and (b) dense environment.

ing distances from each other as indicated in Table 2. R1’s read range does not intersect with any other reader’s interference or read range. R2 and R3 have their read ranges and consequently interference ranges intersecting; thus these readers cannot interrogate tags simultaneously. R4’s read range does not intersect with any other reader’s read range; nevertheless it is in the interference range of the readers R2 and R3. Hence, to avoid interfering with R2 and R3 operations, R4 should use a different frequency channel.

Fig. 10 shows how the DiMCA algorithm works. We assume that the reading range of readers is 10 m (justified later). Consequently, we set the transmission ranges of CC1 (dashed arrows between readers) and CC2 (solid arrows between readers) to 1010 m and 20 m respectively. Initially, both queues Q1 and Q1 of all readers are empty. We assume that all readers want to start tag interrogation simultaneously. First, every reader waits for a pseudo-random time to avoid collision of control packets on the control channels. In the figure, R2 chose the smallest random time, therefore it starts the contention first. Since R2’s Q1 and Q2 are empty, it selects the data channel 5 and broadcasts a START packet on CC1.

R3 and R4 receive this control packet since $D_{2,3} = 17$ m and $D_{2,4} = 25$ m. R3 and R4 add the received information on CC1 to their Q1. R2 then broadcasts a START packet on CC2. Only R3 receives this packet since $D_{2,3} = 17$ m and adds the information to its Q2. R2 then starts the tag interrogation. In the meantime, R1’s random time expires, and since R1’s queues are still empty, R1 broadcasts START packets on both CC1 and CC2, but no one receives them because the distance from R1 to any other reader is greater than 1025 m. So R1 starts the tag interrogation independently of other readers. When the random time of R4 expires, it checks its queues. Since there are available data channels left, and Q2 is empty, R4 selects a data channel other than 5 (i.e., channel 1), which is being used by R2, and broadcasts START packets on both CC1 and CC2. R2 and R3 receive the packet sent on CC1, yet none of the readers receive the packet sent on CC2 because all the readers are more than 20 m away from R4. Then R4 starts the tag interrogation using data channel 1. During the tag interrogation of R4, R3’s random time expires, and starts contending by checking its queues. Since Q2 is not empty, it cannot start its tag interrogation. It chooses a

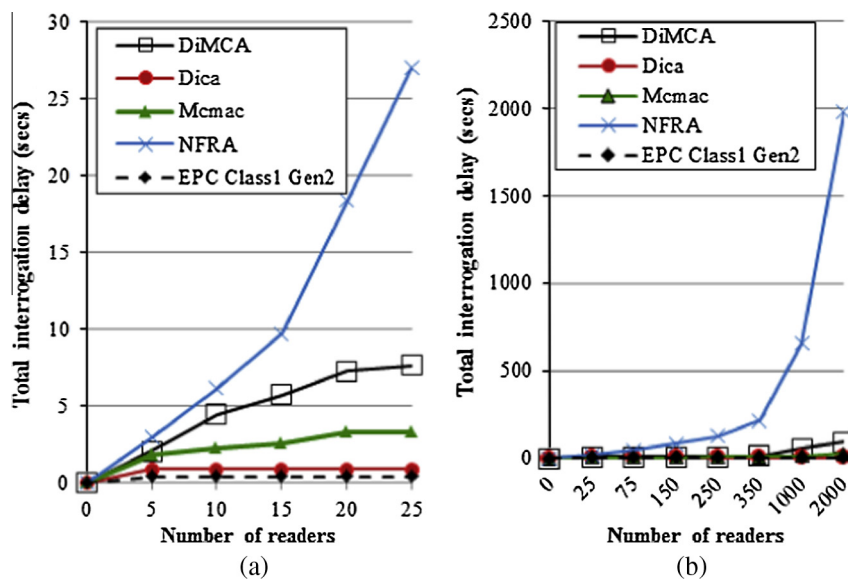


Fig. 13. Total interrogation time vs. the number of readers (a) sparse environment and (b) dense environment.

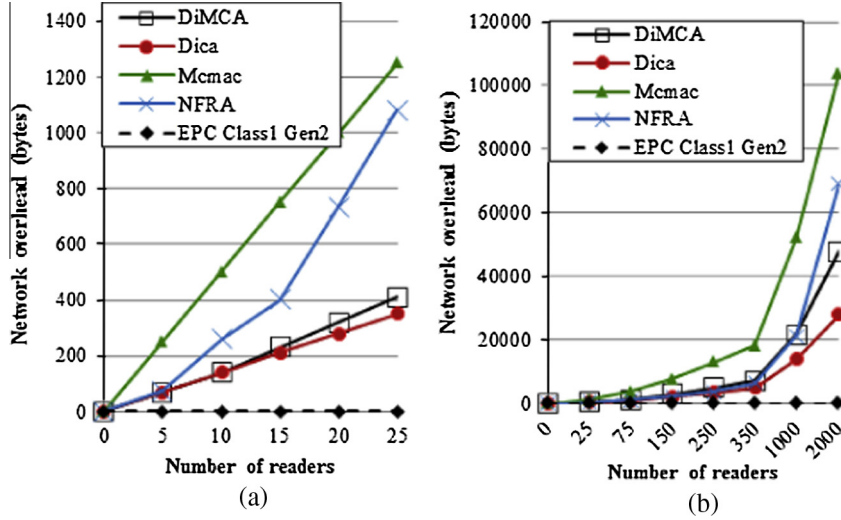


Fig. 14. Network overhead vs. the number of readers (a) sparse environment (b) dense environment.

new pseudo-random time to wait. When R2 finishes its tag interrogation phase, it sends *END* packets on CC1 and CC2. R3 and R4 receive the *END* packet sent on CC1, while only R3 receives the *END* packet sent on CC2. Upon receiving the *END* packet, each reader removes the corresponding information from the queues based on the *ReaderId* field. R3's second pseudo-random time expires and checks its queues again. It finds that its Q2 is empty and can start its tag interrogation. It selects data channel 6 and sends *START* packets on CC1 and CC2 and starts its tag interrogation. All readers, after finishing the interrogation of tags, send *END* packet on both CC1 and CC2 for the other readers to remove the corresponding information from their queues, hence freeing the data channels and allowing them to start their tag interrogations if needed.

3.4. Computation of interference range

The interference range can be calculated using the path loss model given in formula (3):

$$PL(\text{DB}) = \begin{cases} 32 + 25 \log\left(\frac{d}{1}\right) & 0 \leq d \leq 8 \text{ m} \\ 23 + 35 \log\left(\frac{d}{1}\right) & d \geq 8 \text{ m} \end{cases} \quad (3)$$

where d is the distance from the reader. We assume that the readers and the tags are located as shown in Fig. 11. Moreover, since the reader coverage area depends on the reader's output power, we consider that readers are operating with an output power of 2 W, which is the one reached in Europe [14]. So the transmission range of the readers resulted from the 2 W output power is 10 m [5]. Consequently, we use Eq. (3) to calculate the path loss from reader A to the tag and the reverse path loss from the tag to reader A, given that the distance between the reader and the tag is 10 m:

$$PL(\text{DB}) = 23 + 35 \log\left(\frac{d}{1}\right) = 23 + 35 \log(10) = 58 \text{ dB} \quad (4)$$

On the other hand, according to [13], an input power of 13 dB is required to activate a passive tag, so total losses are: $58 + 58 + 13 = 129$ dB. Consequently, to receive the backscattered signal from the tag successfully at reader A, the path loss of the interference signal from reader B to reader A must be greater than 129 dB. The distance needed from reader A to reader B to avoid the reader collision is then calculated as follows:

$$PL(\text{DB}) = 23 + 35 \log\left(\frac{d}{1}\right) = 129 \Rightarrow d \approx 1068 \text{ m} \quad (5)$$

This justifies our choice of the reader interference range, which we assumed as 1000 m when the read range is 10 m.

4. Simulation results

In this section, we study the performance of the proposed DiMCA algorithm and compare it with the standard EPC Class1 Gen2 [9] and several notification based anti-collision mechanisms such as Dica [14], Mccmac [7] and NFRA [8]. We have integrated these algorithms in the network simulator NS3 using the *RangePropagationLossModel* and the 802.11 Wifi model of NS3 [23] (disabling the RTS/CTS and SIFS/DIFS on the MAC layer as done in NS2 [3]). The simulation parameters are shown in Table 3. In our implementation, collisions between control packets are handled by the CSMA function of the MAC layer, while the possibility of reader collisions during the tag interrogation time is detected by the system model described in Section 3. Because tags do not participate in the reader anti-collision solution, we only implement the behavior of the readers in the network. However, like in [15], we assume that each reader has to identify 100 tags in its range, the time needed to read 100 tags is 0.46 s, and all readers might attempt to read tags simultaneously.

Two scenarios have been investigated, one for a sparse reader environment (i.e., small number of readers) and one for dense reader environment (i.e., large scale number of readers). In the sparse environment, we manually located 5, 10, 15, 20, and 25 readers in a 3D area of 3 km^3 such that all types of collisions can frequently occur. In the dense reader environment, we randomly distributed 25, 70, 150, 250, 350, 1000 and 2000 readers in a 3D area of 2 km^2 with a height of 20 m, assuming that the transmission range of the readers is 10 m and the interference range is 1000 m.

We have measured metrics such as total number of failed interrogations, total interrogation delay, rate of successful interrogations, system's efficiency, and network overhead. To ensure credibility of our simulations, we have used the central limit theorem [2] to determine the number of runs required to achieve 90% confidence level, with \pm precision value of 4%.

4.1. Number of failed interrogations

An interrogation is considered to be failed if its query collides with another reader's query in the network. Fig. 12a shows that in the sparse environment scenario, the number of failed

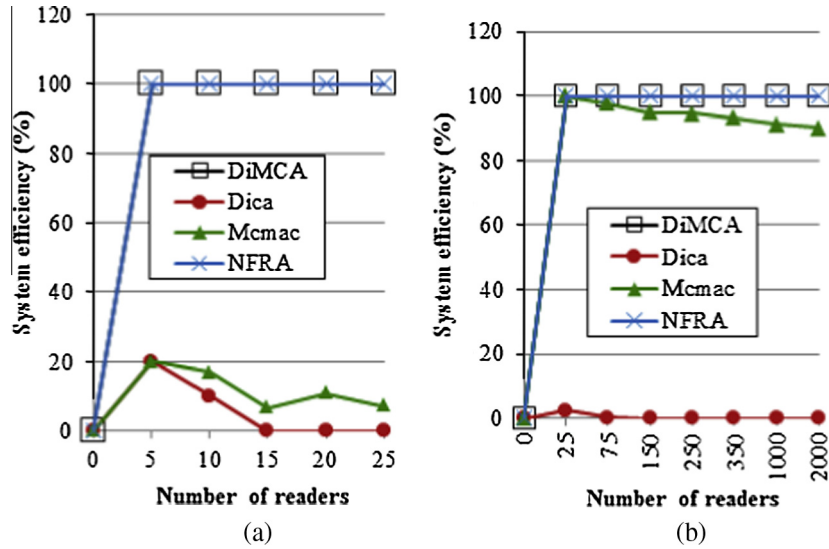


Fig. 15. System efficiency (%) vs. the number of readers (a) sparse environment and (b) dense environment with $r = 10$ and $I = 1000$.

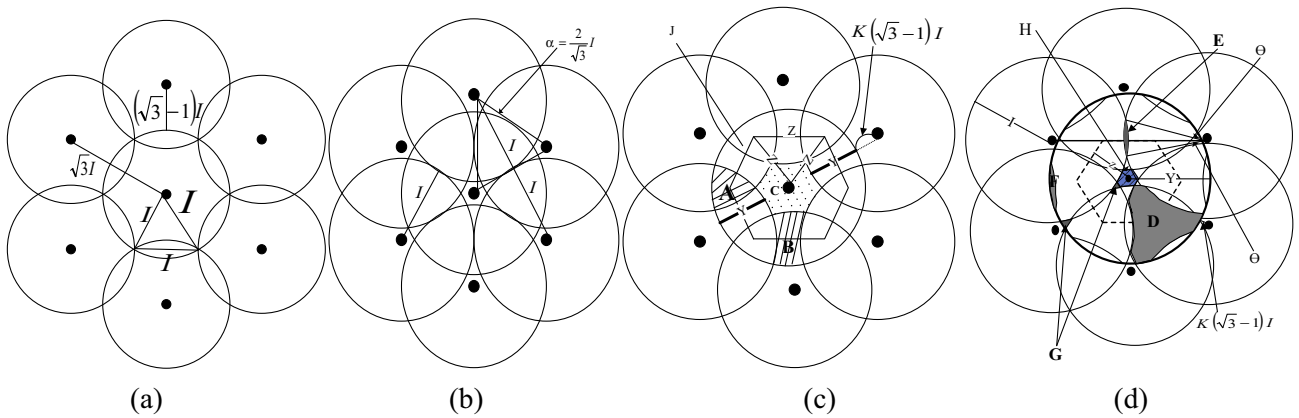


Fig. 16. (a) Readers when $\gamma \leq K \leq 1$ and the distances between them are maximal, (b) readers when $\gamma \leq K \leq 1$ and the distances between them are in-between, (c) readers when $\gamma \leq K \leq 1$ and the distances between them are minimal and (d) system readers when $0 < K < \gamma$.

interrogations increases as the number of readers increases in Dica, Mcmac and EPC Class 1 Gen2. This is because Dica solves RTI only, while Mcmac and EPC Class 1 Gen 2 solve RRI only. Using the proposed DiMCA, no failed interrogations were encountered for any number of readers, just like in NFRA, which also solves RTI and RRI, however using a centralized approach. On the other hand, in a dense environment, Mcmac has a better performance than in a sparse environment, as shown in Fig. 12b, because it solves the second type of RTI, but it ignores the first type. EPC Class1 Gen2’s performance is degraded, because its number of available channels is always 5, which is very small compared to the number of readers causing RRI to occur. The number of failed interrogations in Dica keeps increasing exponentially, because of the unhandled RRI collisions. However, both NFRA and our approach still have no failed interrogations, solving both RRI and RTI.

4.2. Total interrogation delay

The total interrogation delay for a varying number of readers is defined as the total time needed for all readers to finish the interrogation of tags in their vicinity. Fig. 13 shows that EPC Class 1 Gen

2 has the lowest interrogation delay for any number of readers in both sparse environment (Fig. 13a) and dense environment (Fig. 13b). This can be justified because EPC Class 1 Gen 2 does not use any notification mechanism to avoid collisions. Also, Dica and Mcmac have a low interrogation delay in both scenarios, since they do not handle all collisions. Fig. 13a shows that, in a sparse environment, our approach outperforms NFRA (both solve all collisions) and yields a much better interrogation delay for any number of readers in the network. In the proposed approach, the delay increases linearly as the number of readers increases. In contrast to NFRA, the delay increases exponentially when the number of readers exceeds 10. Fig. 13b shows how the delay varies in a dense environment. While the delay highly increases using both our approach and NFRA when the number of readers exceeds 350, our proposed algorithm yields a much better interrogation delay than NFRA for any number of readers in the network and reaches 92 s for 2000 readers, in contrast to the NFRA, where the delay increases to reach 1975 s for 2000 readers. This is reasonable because in NFRA, the number of arrangement commands needed to finish all the interrogations increases as collisions are detected between readers during the beaconing phase, which yields to higher interrogations delays.

Table 4
Simulation parameters.

Parameter	Value
Simulation range	100 m ² with a height of 4 m
Transmission range of CC2	2r = 4 m
Number of data channels	10
<i>Scenario 1</i>	
Interference range of a reader (I)	6 m
Transmission range of CC1	r + I = 8 m
<i>Scenario 2</i>	
Interference range of a reader (I)	50 m
Transmission range of CC1	r + I = 52 m

4.3. Network overhead

To measure overheads, we count all the control packets exchanged between readers, or between the readers and the polling server, whether during the contention phase or the interrogation phase itself. Fig. 14a shows that, in sparse environment, our approach has the lowest overhead after Dica, which does not solve all collisions and EPC Class 1 Gen 2 which does not use any control packets to avoid collisions. The proposed DiMCA outperforms NFRA which solves all reader collisions, especially when the number of readers exceeds 15. Fig. 14b shows that, in dense environment, the amount of overhead using our approach keeps increasing as the number of readers increases to reach 47762 bytes for 2000 readers, while using NFRA, the overhead increases to reach 68995 bytes for 2000 readers, almost 44% increase.

4.4. System's efficiency

Finally, we define the system's efficiency as follows:

$$\text{System Efficiency (\%)} = \frac{\text{Total}_{\text{success}} \times 100}{\text{Total}_{\text{interrogations}}} \quad (6)$$

where $\text{Total}_{\text{success}}$ is the total successful interrogations by all readers and $\text{Total}_{\text{interrogations}}$ is the total number of interrogations in the network.

Fig. 15 shows that both the proposed DiMCA and NFRA achieve 100% efficiency, while Dica and Mccmac achieve lower efficiency because of the unhandled RRI and RTI collisions. These results prove that efficiency of the distributed notification mechanism we used is more efficient than all the other ones in both the sparse and dense environment scenarios.

4.5. Deriving the number of readers

To calculate the number of readers that can be placed in a certain area without interfering with neighboring readers, we adapt the model used in [8] to our environment. The readers are assumed to be regularly distributed, so six readers would be located around a certain reader, all at equal distances from it as shown in Fig. 16. The distances shown in Fig. 16a are the maximum possible distances to be able to cover all the spaces in the area. The reader's interference range is denoted by I . Since a triangle is made by the intersections of the two circles with the reader, the distance between two readers would be $\sqrt{3}I$. So other readers can be located at distance between I and $(\sqrt{3} - 1)I$ along the line connecting the readers.

A uniform random variable K is defined from zero to one to specify the location of other readers. Therefore, other operating readers can be located on $I + K(\sqrt{3} - 1)I$ distance from the reference reader. Let α be the distance between two operating readers, as shown in Fig. 16b. Consequently, because of the triangle made by the three readers and for some $K = k^*$, α can be expressed as follows:

$$\alpha = I + k^*(\sqrt{3} - 1)I = \frac{2}{\sqrt{3}}I \quad (7)$$

$$\text{Thus, } k^* = \frac{2 - \sqrt{3}}{3 - \sqrt{3}} \quad (8)$$

Assuming $\gamma = k^*$, two cases are distinguished: (1) ($\gamma \leq K \leq 1$), and (2) ($0 < K < \gamma$). When K is greater than or equal to γ , as shown in Fig. 16c, the number of shapes denoted as A is six, and A is shared by three overlapping circles. The number of the shapes denoted as B shared by two circles is also six. We want to find the actual area covered only by the reader in the center. Therefore, the reader has six A 's divided by three, and six B 's divided by two and one shape C . Thus, the area S_1 which one reader occupies is $2A + 3B + C$, and it is equal to the area of a regular hexagon J , as shown in Fig. 16c. Let Y and Z be the thickness of a biconvex lens shape overlapped by two circles and the length of a side of the regular hexagon consisting of six regular triangles, respectively. Then, the height of the regular triangle in the hexagon is half of the distance between two readers, and Z is calculated by multiplying the height by $2/\sqrt{3}$ because of a regular triangle property. Thus, the length Z and the area S_1 are derived as follows:

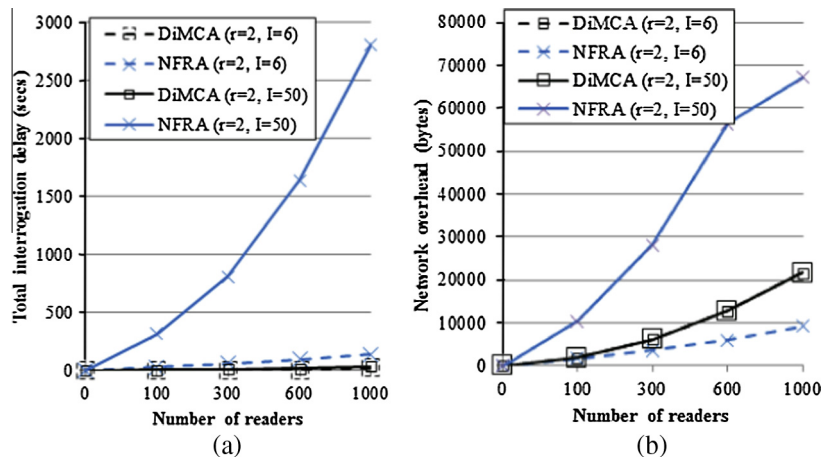


Fig. 17. (a) Total interrogation delay in seconds vs. the number of readers in a dense reader environment with $r = 2$ and $I = 6, 50$ and (b) network overhead in bytes vs. the number of readers in a dense reader environment with $r = 2$ and $I = 6, 50$.

$$Y = I - K(\sqrt{3} - 1)I$$

$$Z = \frac{2}{\sqrt{3}} \times \left(\frac{Y}{2} + K(\sqrt{3} - 1)I \right) = \frac{2}{\sqrt{3}} \times \left(\frac{I + K(\sqrt{3} - 1)I}{2} \right) \quad (9)$$

$$S_1 = 6 \times \frac{\sqrt{3}}{4} Z^2 = \frac{\sqrt{3}}{2} I^2 (1 + K(\sqrt{3} - 1))^2 \quad (10)$$

If K is smaller than γ , a model of Fig. 16d is used where the reader has a virtual area F . The area F has the same size as the area E . The area F is introduced to compute the area D with ease. Note that D becomes symmetric by separately considering the area F . In a similar manner, the numbers of the areas D , E , F , and G are 6, 6, 6, and 12, and they are shared by 3, 4, 3, and 2 overlapping circles, respectively. The center area H is not shared. The area S_2 which one reader occupies is $2D + 1.5E + 2F + 6G + H$, and it is equivalent to $2D + 3.5E + 6G + H$, since E and F are the same. The hexagon shown in Fig. 16d is composed of 2 D 's, 6 G 's, 3 E 's, and H . Therefore, the area S_2 is obtained by adding the area of the hexagon to $0.5E$. The hexagonal area can be calculated by (9). As shown in Fig. 16d, the angle θ is set to obtain the area of E . The angle Θ is a uniform random variable in $[0, \pi/6]$, since the value depends on K . The area $0.5E$ is found by the difference between the fan-shaped area and the triangle area:

$$0.5E = I^2 \theta - I^2 \sin \theta \cos \theta = I^2 \left(\theta - \frac{\sin 2\theta}{2} \right) \quad (11)$$

$$S_2 = \frac{\sqrt{3}}{2} I^2 (1 + K(\sqrt{3} - 1))^2 + 0.5E \quad (12)$$

Therefore, the effective area S occupied by one reader is found as:

$$\begin{aligned} S &= S_1 I (\gamma \leq K \leq 1) + S_2 I (0 < K < \gamma) \\ &= \frac{\sqrt{3}}{2} I^2 (1 + K(\sqrt{3} - 1))^2 + 0.5E T(0 < K < \gamma) \end{aligned} \quad (13)$$

where $T(\cdot)$ is an indicator function. Since K and θ are uniform random variables, the expected value of the area covered by a reader becomes:

$$\begin{aligned} E[S] &= \int_0^1 \frac{\sqrt{3}}{2} I^2 (1 + K(\sqrt{3} - 1))^2 dk \\ &\quad + \gamma \int_0^{\frac{\pi}{6}} I^2 \left(\theta - \frac{\sin 2\theta}{2} \right) \frac{6}{\pi} d\theta \\ \Rightarrow E[S] &= \left(\frac{3 + 4\sqrt{3}}{6} \right) I^2 + \frac{\gamma \pi I^2}{12} - \frac{6\gamma I^2}{8\pi} \end{aligned} \quad (14)$$

Thus, we can know how many readers can operate at the minimum in dense RFID reader environment by dividing the entire area by (14). We let I be the readers' interference range, respectively. Thus, the lower bound for the number of the operating readers N can be given by:

$$N > \frac{\pi I^2}{E[S]}$$

In the dense reader environment, to make the area denser, we simulated another two scenarios in which we randomly distributed 100, 300, 600 and 1000 readers in 3D area of 100 m² with a height of 4 m, assuming that the transmission range of the readers is 2 m for both scenarios, and the interference range is 6 m in one scenario and 50 m in another one. Table 4 summarizes these simulation parameters.

Fig. 17 shows the results comparing our approach to NFRA in denser environments. Fig. 17a shows that our approach achieves lower delays for both scenarios. This is because in NFRA, the number of arrangement commands required to finish all the interrogations will increase as collisions are detected between the readers

during the beaconing phase; this yields higher interrogation delay. We observe that the interrogation delay using our approach increases as the environment gets denser. This is because readers would have to wait for their Q2 to be empty to start their interrogations, to avoid type 1 of RTI. Fig. 17b shows the overhead using our approach and NFRA in a denser environment by varying the interference range and distributing the readers over a smaller area. It shows that our approach has the lowest overheads in the network after Dica; however Dica does not solve all the collisions. Mcmac has the highest overheads. Compared to NFRA which solves all the reader collisions, our approach has a lower overhead, especially when the number of readers exceeds 15.

5. Conclusion

In this paper, we proposed a new distributed multi-channel algorithm that solves both RTI and RRI reader collisions using a notification mechanism that is used to make RFID readers aware of the network resources utilization. We evaluated the proposed approach through simulations using a varying number of readers. We have showed that the information gathered in the queues of the readers allowed them to efficiently use the network resources, time and data channels, to solve both types of reader collisions (RTI and RRI). Moreover, we showed that our distributed strategy minimized the total interrogation delay compared to the centralized approach NFRA, even when the delays increase in dense environments. In addition, the amount of overhead in the network is found to be reasonable. Hence, for every interrogation every reader has to send a specific number of control packets (2 *START* packets and 2 *END* packets). As a result, the total number of exchanged packets remains limited, unlike Mcmac where the readers keep broadcasting control packets during their interrogation time to keep their neighbors notified of their use of the data channel. Finally, it was shown that our approach has a high rate of successful interrogations and 100% efficiency.

Acknowledgment

This work was supported in part by a grant from the University Research Board of the American University of Beirut, Beirut, Lebanon (URB-AUB-21721-11175-102725).

References

- [1] W. Alsalihi, K. Ali, H. Hassanein, A power control technique for anti-collision schemes in RFID systems, *Comput. Networks* 57 (9) (June 2013) 1991–2003.
- [2] T.R. Andel, A. Yasinac, On the credibility of manet simulations, *IEEE Comput.* 39 (7) (July 2006) 48–54.
- [3] A. Balakrishnan, S. Krushnan, Simulation of RFID platform on NS-2, *IEEE*, 2005, pp. 1–12.
- [4] S.M. Birari, S. Iyer, PULSE: A MAC Protocol for RFID Networks, 1st Int. Workshop on RFID and Ubiquitous Sensor Networks, Japan, December 2005.
- [5] M.V. Bueno-Delgado, J. Vales-Alonso, C. Angerer, M. Rupp, A comparative study of RFID schedulers in dense reader environments, in: *Proc. of the IEEE International Conference on Industrial Electronics (ICIT)*, pages 1353–1358, Viña del Mar, Chile, March 2010.
- [6] K. Cha, S. Jagannathan, Adaptive power control protocol with hardware implementation for wireless sensor and RFID reader networks, *IEEE Syst. J.* 1 (2) (December 2007) 145–159.
- [7] H. Dai, S. Lai, H. Zhu, A Multi-Channel MAC Protocol for RFID Reader Networks, in: *Proc. of IEEE Int. Conf. on Wireless Communications, Networking and Mobile Computing (WiCom 2007)*, 21–25, 2007, pp. 2093–2096.
- [8] J.-B. Eom, S.-B. Yim, T.-J. Lee, An efficient reader anticollision algorithm in dense RFID networks with mobile RFID readers, *IEEE Trans. Industr. Electron.* 56 (7) (July 2009) 2326–2336.
- [9] EPC Radio-Frequency Identify Protocol for Communications at 868–960MHz, Version 1.0.9: EPCglobal Standard Specification, 2004, <<http://www.epcglobalinc.org/standards>>.
- [10] ETSI-EN 302 208-2 v1.1.1, September 2004, CTAN. <<http://www.etsi.org>>.
- [11] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, John Wiley & Sons, 2003.

- [12] F. Gandino, R. Ferrero, B. Montrucchio, M. Rebaudengo, Probabilistic DCS: an RFID reader-to-reader anti-collision protocol, *J. Network Comput. Appl.* 34 (3) (May 2011) 821–832.
- [13] L. Harvey, RFID Design Principles, Chapter 5: Components of the RFID System, 2007. <www.artechhouse.com>.
- [14] K. Hwang, S.-S. Yeo, J.H. Park, Distributed Tag Access with Collision-Avoidance among Mobile RFID Readers, in: Proc. of Int. Conference on Computational Science and Engineering, 2009 (CSE '09), vol. 2, no., pp. 621–626, 29–31 August 2009.
- [15] G. Joshi, K. Mamun, S. Kim, A Reader Anti-collision MAC Protocol for Dense Reader RFID System, 2009 International Conference on Communications and Mobile Computing, pp. 313–316.
- [16] D.K. Klair, K.-W. Chin, R. Raad, A Survey and Tutorial of RFID Anti-Collision Protocols, *IEEE Communications Surveys & Tutorials*, vol. 12, no. 3, pp. 400–421, Third Quarter 2010. <http://dx.doi.org/10.1109/SURV.2010.031810.00037>.
- [17] Y.-C. Lai, L.-Y. Hsiao, B.-S. Lin, An RFID anti-collision algorithm with dynamic condensation and ordering binary tree, *Comput. Commun.* 36 (17–18) (December 2013) 1754–1767.
- [18] K.S. Leong, M. Ng, P. Cole, The reader collision problem in RFID systems", in: Proc. of IEEE Int. Symp. on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications, 2005, pp. 658–661.
- [19] K. Leong, M. Ng, A. Grasso, P. Cole, Synchronization of RFID readers for dense RFID reader environments, in: Proc. IEEE Int. Symp. Appl. Internet Workshops, January 2006, pp. 48–51.
- [20] Z. Li, C. He, J. Li, X. Huang, RFID reader anti-collision algorithm using adaptive hierarchical artificial immune system, *Expert Syst. Appl.* 41 (5) (April 2014) 2126–2133.
- [21] C. Meguerditchian, H. Safa, W. El-Hajj, New Reader Anti-Collision Algorithm for Dense RFID Environments, in: Proceedings of the 18th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2011), December 11–14, 2011.
- [22] V. Nambodiri, M. DeSilva, K. Deegala, S. Ramamoorthy, An extensive study of slotted Aloha-based RFID anti-collision protocols, *Comput. Commun.* 35 (16) (2012) 1955–1966. 15 September.
- [23] NS-3. <<http://www.nsnam.org/>> (accessed 16.01.11).
- [24] S. Piramuthu, S. Wochner, M. Grunow, Should retail stores also RFID-tag 'cheap' items?, *Eur. J. Oper. Res.* 233 (1) (February 2014) 281–291.
- [25] P. Pupunwiwat, P. Darcy, B. Stantic, Conceptual selective RFID anti-collision technique management, *Procedia Comput. Sci.* 5 (2011) 827–834.
- [26] C. Qian, Y. Liu, R.H. Ngan, L.M. Ni, ASAP: scalable collision arbitration for large RFID systems, *Parall. Distr. Syst., IEEE Transact.* 24 (7) (July 2013) 1277–1288.
- [27] J. Waldrop, D.W. Engles, S.E. Sarma, Colorwave: A MAC for RFID reader networks, in: Proc. IEEE Conf. Wireless Commun. Network, March 2003, pp. 1701–1704.
- [28] J. Waldrop, D.W. Engles, S.E. Sarma, Colorwave: An anticollision algorithm for the reader collision problem, in: Proc. IEEE Int. Conf. Commun., May 2003, pp. 1206–1210.
- [29] M.-K. Yeh, J.-R. Jiang, S.-T. Huang, Adaptive splitting and pre-signaling for RFID tag anti-collision, *Comput. Commun.* 32 (17) (November 2009) 1862–1870.
- [30] J.-B. Eom, T.-J. Lee, R. Rietman, A. Yener, An efficient framed slotted ALOHA algorithm with pilot frame and binary selection for anti-collision of RFID tags, *IEEE Commun. Lett.* 12 (11) (November 2008) 861–863.
- [31] J. Myung, W. Lee, J. Srivastava, Adaptive binary splitting for efficient RFID tag anti-collision, *IEEE Commun. Lett.* 10 (3) (March 2006) 144–146.