ORIGINAL ARTICLE

Energy aware improved least and most significant bit arbitration algorithm for WORM tags

Katheeja Parveen [a],*, Sheik Abdul Khader [a], Munir Ahamed Rabbani [b]

[a] School of Computer and Information Sciences, B.S. Abdur Rahman University, Chennai, India
[b] College of Computer, Qassim University, Saudi Arabia

Abstract Passive Radio Frequency Identification systems have gained enormous attention and popularity especially after its adoption in time and data critical systems. Theoretically, these systems have the potential to read over 100 tags per second in applications which are well insulated from RF noise. Nevertheless, this may not be the case in practical systems, as tag collision is one of the major deterrents affecting the recognition rate. This paper exhaustively analyses the existing probabilistic, deterministic and hybrid algorithms on collision resolutions. In probabilistic algorithms, tags send their entire ID to the RFID reader in respective slots while tags in deterministic algorithms respond bit by bit based on the RFID reader's query. To minimize identification delay, tag communication overhead and high energy consumption, a new energy efficient collision resolution strategy named Improved Least and Most Significant Bit Algorithm (LaMSBA) is introduced to effectively singulate a tag and increase the identification efficiency in changing tag population even when the bits in tag ID's are randomly or uniformly distributed. Extensive simulation studies

* Corresponding author.
E-mail addresses: katheeja.parveen@bsauniv.ac.in (K. Parveen), psakhader@bsauniv.ac.in (S.A. Khader), marabbani@rediffmail.com (M.A. Rabbani).

show that LaMSBA can be chosen as better alternatives for dense time and data critical RFID enabled systems. In addition, M/G/1 Queuing model is suitably identified and the the analytical results concluded that LaMSBA is able to maintain the steady state condition even when Class 1 tags arrive at the rate of 15 tags/second in the reader's interrogation zone.

1. Introduction

Radio Frequency Identification (RFID) is an intriguing technology that has garnered a great deal of research interest. The latest RFID forecasts from ABI Research indicate that the value of the overall market is expected to reach $5.3 billion by the end of 2011, a year-over-year growth of more than 16% and will pass the $6 billion mark next year [1]. One of the areas, where RFID systems significantly impact the overall performance of an application, is supply chain management (SCM). While there have been many successful rollouts of RFID technology, only a handful of them are associated with the SCM. Although the results of the number of surveys show appreciation towards the usability and acceptability of RFID systems in the SCM, it is found that there is still reluctance in the business community to invest large amounts of capital in such a new technology that is yet to prove it fool proof in the long-term.

2. Problem definition

Tag collision is a major problem deferring tag identification in RFID systems. This issue primarily occur due to the backscatter of simultaneous response from tags present in the region to the RFID reader. The RFID reader in such a scenario fails to differentiate the data received from multiple tags and eventually, produces erroneous reading as seen in Fig. 1. [2]. Due to cost constraints, it is often observed that vendors prefer to have a single reader to read accurately varied tag population which will also ensure to minimize number of collisions, identification time/delay, number of bits transmitted by the reader, number of bits transmitted by the tag, occurrence of idle responses, energy consumed by the reader while scanning, energy
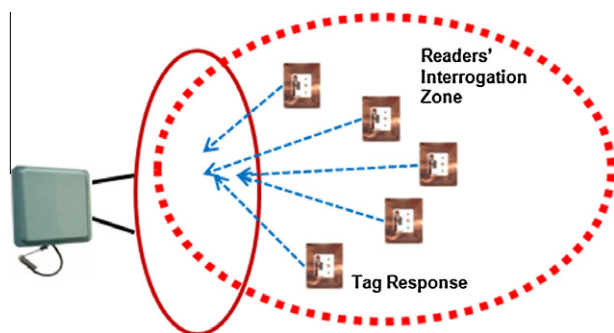
consumed by the reader while idle, bit transmissions, time spent during idle and over all identification time.

A good deal of literature is available pertaining to tag collision that has resulted in the reduction of identification delay. However, it is felt that there is a need for research enhancements and development of algorithms that are attuned to an environment with unpredictable changes. This paper aims to develop energy efficient collision resolution strategies for Class 1 passive tags in dense passive RFID systems.

Section 3 details the existing approaches available and Section 4 proposes a novel algorithm, Least and Most Significant Bit Arbitration Algorithm. Detailed simulation set up is presented in Section 5. To validate the accuracy of simulation results, M/G/1 queueing model is suitably identified and its detailed narration is mentioned in Section 6. The results of the simulation experiments conducted on existing algorithms and proposed algorithm are detailed in Section 7. Section 8 concludes the paper.

3. Existing tag collision algorithms

Broadly, tag anti collision algorithms are categorized into Space Division Multiple Access (SDMA), Code Division Multiple Access (CDMA), Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA) [3–6]. SDMA protocols are expensive and require complex antenna designs, while CDMA, apart from being expensive, is also in need of high energy source. On the other hand FDMA requires a complex RFID reader as tags talk on one of the predefined frequency channels. On the other hand, TDMA constitute large group of anti collision algorithms. TDMA algorithms can either be Reader Driven (i.e., Reader Talk First) or Tag Driven (i.e., Tag Talk First). These algorithms can further be classified as Probabilistic (Aloha based), Deterministic (Tree-based) and Hybrid (combination of probabilistic and deterministic) algorithms. Probabilistic algorithm requires tags to respond randomly in an asynchronous manner, while most tree algorithms operate by grouping the responding tags into subsets [7,8,2,9,10].

3.1. Probabilistic algorithms

Pure Aloha (PA) is a simple TDMA algorithm where a tag begins to transmit its identifier (ID) as soon as it is energized by the RFID reader. This mechanism is referred to as "Tag-Talks-First" (TTF) behavior [10–12]. Pure aloha has a number of variants namely muting, slow down, fast mode, fast mode and muting, and fast mode and slow down.

Slotted-Aloha (SA) algorithm is an improvement of PA. In SA, time is divided into discrete time intervals, called slots. A

Figure 1 Tag collision.

tag is allowed to transmit only after a slot delimiter is observed. Extensions to SA are muting and early end.

Frame-Slotted Aloha (FSA) algorithm is built by enhancing Slotted-Aloha and the discrete time division to one step further by grouping several slots into frames where in each frame can be divided to N slots. Even though the slot architecture remains the same as seen in SA, tags in FSA, are allowed to transmit exactly once every frame in a randomly selected slot within each frame.

Dynamic Frame Slotted Aloha (DFSA) algorithm is introduced to resolve the limitation of FSA. According to the number of collisions, idle responses and number of successful slots in the current frame, the RFID reader decides on the frame length to be broadcasted in the next frame [11,13–15].

In Improved DFSA (IDFSA), the tags are divided into groups in different frequency channels to enhance the identification efficiency and to save the time of the command of the EDFSA. Simulation results of Geng et al. [15] have confirmed that the system identification of IDFSA is higher than the traditional method. On the other hand, implementation of IDFSA is not economical to implement when compared with EDFSA [15].

3.2. Deterministic algorithms

In deterministic algorithms, the RFID reader repeatedly splits the tags into two groups, until a group contains only one tag. Collision resolution algorithms based on tree have been introduced through the work of Hayes [3], Tsybakov and Mikhailov [16], Capetanakis [17], and Hush and Wood [18].

3.2.1. Polling
It requires all tag identification numbers that can possibly occur in the RFID reader's range before hand. In this approach, tags instead of transmitting their entire serial number in one burst, respond to signals by revealing one bit at a time [17].

3.2.2. Binary tree walking algorithm
In this algorithm, the RFID reader initiates the session by randomly choosing the bit "0" or "1" and broadcasting the query to the tags [18].

3.2.3. Query tree
Law et al. [19] proposed Query Tree (QT) algorithm to augment the efficiency of BTWA. To speed up the identification process, QT requests tags to transmit from $(k + 1)$th bit to the end bit of tag IDs, as against in BTWA where matching tags transmit $(k + 1)$th bit alone [19]. Implementation of QT requires very less change in tag circuitry. However, when the number k bit responses from the tags is in rise, the QT will end up in heavy collision. A number of variants were proposed for QT namely QT shortcutting, QT Aggressive and Categorization, QT-short long [20], Query Tree Improved [19], Randomized Hashing Query Tree [21], QT with Reversed, Bit by Bit [22], and Modified bit by bit binary tree algorithm [23].

3.2.4. Adaptive query splitting
Myung et al. [24] proposed a algorithm called Adaptive Query Splitting (AQS), where the RFID reader is required to maintain a queue "Q" that operates similarly to the stack in the QT algorithm. In addition, the RFID reader is required to maintain a candidate queue (CQ) for storing queries sent in past identification rounds [24]. AQS helps in the reduction of collisions, but additionally produces some idle cycles. To guarantee the recognition of all tags, the RFID reader uses not only the queries of the readable cycles but also the queries of idle cycles of the last frame.

3.2.5. Optimal distance based clustering
It works by dividing the interrogation zone into "k" sized clusters based on the distance to the RFID reader. Partitioning the interrogation zone can be achieved by controlling the RFID reader's antenna power level. The tags in different clusters are interrogated separately. To read tags in one cluster, the RFID reader and tags in that cluster can adopt any one of the anti collision resolution algorithms and get identified by surpassing through several cycles [25].

3.2.6. Hybrid algorithms
Ryu et al. [26] combined QT algorithm with a slotted random back-off mechanism. He also proposed an enhanced HQT algorithm, which uses the slotted back-off with the AQS algorithm [26]. Shin et al. [27] proposed two algorithms that use a combination of QT and Framed Aloha protocols: Framed Query Tree algorithm and Query Tree ALOHA algorithm [27].

3.2.7. Spread partial-Q slot count
The algorithm is based on a slotted ALOHA CDMA technique along with tag set partitioning. The algorithm partitions the tags into a certain number of groups by taking advantage of different power levels received from the continuous wave form of the RFID reader to power-up passive tags, which then backscatter accordingly with varied signal strengths [28,29]. Nevertheless the implementation of SPSC, in turn raises the cost of system bandwidth and complexity.

4. Proposed algorithm

This section proposes "Improved Least and Most Significant Bit Algorithm" (LaMSBA), a proficient anti collision algorithm which is appropriate for dense passive RFID deployments. In an effort to reduce the number of bit transmissions from tags and idle responses, the algorithm requests matching tags to respond collided bits from left and right of the query simultaneously. This approach of querying two bits at a time has greatly reduced the time spent on tag singulation and has also minimized the tag population responding to the query broadcasted. Simulation experiments are carried out in static and dense environments to analyze its capability over the existing AQS.

4.1. Methodology

The RFID reader begins the cycle by informing tags in the region that a reading cycle is about to begin. Tags in the region backscatter their ID bit by bit. Based on the response, the reader builds the initial tag response query. LaMSBA looks out for a common prefix pattern as followed by B-Bit Pooling and informs the tags, if a prefix match is found. If the length of the initial tag response query is received without

collision and its length is equal to the length of the tag ID, the RFID reader broadcasts that the tag is identified and concludes the reading cycle. During rounds where collision has occurred, a query is divided into "least prefix query" and "most prefix query". The "least prefix query" takes into account of those bits which have their bit positions from one to the half of the length of the Tag ID while the bits positions from one more the half of the length of the tag ID to the length of the tag ID are taken as the "most prefix query". For understanding, 8 bit query "10001111" is considered. In this, query "1000" is assigned to the least prefix query (bit position from 1 to 4 (both 1 and 4 inclusive)) and query "1111" is assigned to the most prefix query (bit position 5–8 (both 5 and 8 inclusive)). Also, least bit indicate "1" (first unidentified bit from the left of the least prefix query) and "1" for most bit (first unidentified bit from the right of the most prefix query). Instead of interrogating the collided bits from the left to the right as in Query Tree algorithm [31] or from the right to the left as performed by QT – Reversed algorithm [32], queries in LaMSBA are built by considering the collided bits of the least prefix query and the most prefix query. This methodology of splitting a query into least prefix and most prefix is done to ensure that a tag is singulated at a faster pace and also to bring down the matching tag density. The tags matching with the least and most prefix query backscatters their least bit and most bit. For understanding, tag T1 with its ID "10010010" is considered. When the reader broadcast the least prefix query as "10" and most prefix query as "10", the matching tag T1 responds its least bit as "0" (3rd bit position) and most bit as "0" (6th bit position) to the RFID reader. The reader upon receiving the query successfully requests tag T1 to reply its next and least and most bit. This procedure is repeated until the ID is framed successfully by the RFID reader. In cases where a collision occurs at least bit alone, most bit alone or at least bit and most bit, when more than one matching tag responds the reader builds queries as illustrated in the example herewith. Consider tags T1 (100100) and T2 (111011). When the reader requests tags to reply its first least bit and most bit values, tag T1 responds as "1" and "0" while T2 replies as "1" and "1". Since collision has occurred while receiving most bit, LaMSBA builds queries as "1XXXX0" and "1XXXX1" pushes to the stack, where "X" denotes an unidentified bit. Similarly if the tags would have faced a collision during the transmission of least bit alone, in such a scenario, LaMSBA builds queries "0XXXX" + "received most bit" and "1XXXX" + "received most bit". In cases where collision occurs at both bits, LaMSBA construct queries as "0XXXX0", "0XXXX1", "1XXXX0", and "1XXXX1". When observing closely, it can be found that, due to the consideration of least and most bit, tag T1 is singulated in the very first step itself. When the stack pops the first query and broadcasts it, Tag 1 alone responds its least bit ("0") and most bit ("0"), which in case with other algorithms could have called for more number of rounds.

Since LaMSBA has a deterministic approach, the occurrence of an idle response is limited, thus saving appreciable energy and identification time. This increase in the identification time also contributes to the benefit of avoiding missed tag reads. The pseudo code for Least and Most Significant Bit Arbitration algorithm is presented herewith.

Activity
R *inform beginning of the reading cycle*
T *respond ID bit by bit*
R *initial_tag_response ← add bits received*
R *find prefix_query from initial tag response*
R **if** *len(prefix_query) = len(TAG_ID)* **then** // single tag
mute the identified tag
END
end if
R *b ← len(initial_prefix_query)*
if *len(prefix_query) != 0* **then**
inform prefix found to tags
// push the prefix query to the stack
push(prefix_query)
else
// since the first bit has collided
//push queries "1" and "0" to the stack
push(1)
push(0)
end if
while *stack is not empty*
R // remove prefix query from the query popped
broadcast_query ← pop(probable_query)
if *len(broadcast_query)*
= len(TAG_ID) − len(prefix_query) **then**
mute the identified tag
goto step 7
end if
R // find least bit
least_bit ← 0
for *index ← 1 to length (TAG_ID)/2*
if *broadcast_query[index] = unidentified* **then**
least_bit ← index
end if
end for
R // find most bit
most_bit ← 0
for *index ← len(TAG_ID) to len(TAG_ID)/2 + 1*
if *broadcast _query[index] = unidentified* **then**
most_bit ← index
end if
end for
R *broadcast broadcast_query, least bit and most bit to tags*
T *Matching tags respond bit values of least bit and most bit*
R **if** *! matching tags* **then**
goto step 7
end if
R **if** *received(least_bit) = collision &&*
received(most_bit) = collision **then**
push(1, 1, least_bit, most_bit)
push(1, 0, least_bit, most_bit)
push(0, 1, least_bit, most_bit)
push(0,0, least_bit, most_bit)
else if *least_bit = collision* **then**
push(1, received(most_bit), least_bit, most_bit)
push(0, received(most_bit), least_bit, most_bit)
else if *most_bit = collision* **then**
push(received(least_bit), 1,
least_bit, most_bit)
push(received(least_bit), 0, least_bit, most_bit)
else
push(received(least_bit), received(most_bit), least_bit, most_bit)
end if
end while
R – Reader T – Tag len-length

5. Simulation experiment

A simulator is designed in C++ to evaluate the performance of the existing and proposed algorithms. The RFID reader is modeled, based on the design features of SkyeTek's M1 – Mini RFID reader. The environment follows single channel, Poisson arrivals, exponential service, infinite population (since the RFID reader is unaware of the number of tags), service in random order and follows any arbitrary probability distribution. The following assumptions are made,

- A noise free channel is considered, i.e., packet losses occur only due to collision.
- The tag's antenna is never at 90°, i.e., all tags are in alignment with the RFID reader's antenna.
- The RFID reader is allowed to transmit energy until all tags are read.
- All tag IDs are of same bit length.
- The RFID reader has the knowledge on the number of bits present in a tag ID.
- The RFID reader is unaware of the number of tags.
- Tags are scattered at random location inside the terrain.
- Although tags are energized at the same time, the energy consumption is estimated only after the reading process has started.
- The delay associated with energizing tags, propagation and processing omitted.

5.1. Simulation set up in static environment

Table 1 show the simulation set up of the RFID system in static environment.

The metrics namely collision, idle responses, bit transmission (reader and tag), energy consumption (scanning, receiving and idle), time spent (bit transmission, idle time) and total identification time spent are noted down after the identification of every 200th tag for evaluation.

6. M/G/1 retrial queueing model

This paper is the first of its kind to identify a single reader retrial M/G/1 queueing model for passive RFID systems to validate the accuracy of the simulation results obtained in the existing algorithm and proposed algorithms. This queueing model also helps to predict the behavior of RFID systems that

Table 1 Simulation set up in static environment.

Terrain	10 m × 10 m
Number of UHF RFID reader	1
Number of class 1 tags	1000
Length of tag ID	96 bits
Time spent for a RFID reader to determine a idle response	0.5 ms
Duration of a silence command	0.19 ms
Number of bits transmitted to issue a silence command	5 bits
Data rate as per ISO 15693	26 kbps
Power consumed by the RFID reader during	
Scanning	180 milli watts
Idle	30 milli watts
Sleep	50 milli watts

attempt to provide service for randomly arising demands. In classical queuing theory it is usually assumed that any customer (i.e., tag), who cannot get service (i.e., identified) instantly from the server (i.e., RFID reader) upon arrival either joins a waiting line or leaves the system forever. In real time applications, this may often not be the case. The tags instead of leaving the service area (interrogation zone) completely return after a random back off time to observe if a service can be obtained now. The standard queueing models do not take into account the phenomenon of retrials and hence a queuing model is derived to suit to RFID system environments.

To predict results close to real time environments, tags instead of being in fixed locations are allowed to move around or leave the interrogation zone. The average set up time taken by the RFID reader after the commencement of every identification cycle is also considered. Further new tags arrive to the interrogation zone according to a compounded Poisson process with rate λ [30]. Tags which find the RFID reader busy may enter into orbit (inactive state) or wait till the RFID reader summons in accordance with Service in Random Order (SIRO) discipline. Also the system capacity is determined to be infinite as the RFID reader is unaware about the number of tags in the interrogation region and hence an upper bound cannot be defined. Successive inter-retrial times of any tag are governed by an arbitrary probability distribution function $A(x)$, with corresponding density function $a(x)$ and Laplace–Stieltjes transform $\gamma^*(\theta)$. Successive service times are independent with common probability distribution function $S(x)$, density function $s(x)$, Laplace–Stieltjes transform $S^*(\theta)$ and first two moments are defined as $E(S)$ and $E(S^2)$. Similarly the successive set up times are independent and identically distributed with probability distribution function $R(x)$, density function $r(x)$, Laplace–Stieltjes transform $R^*(\theta)$ and first two moments $E(R)$ and $E(R^2)$.

Besides, an identified tag may either join the retrial group again for another service (be a staying tag) with probability p or to leave the system (be a departing tag) forever with probability $q (1 - p)$. It is assumed that the probability of successful commencement of service is δ for a new tag (when the tag finds the RFID reader idle and when there are no other tags in the orbit) and for all other new and returning tags the probability is set as α. Inter arrival times, retrial times, service times and breakdown times are assumed to be mutually independent.

Hence, the Kendall notation can be defined as $M^X/G/1$ denoting a simple queueing system consisting of one reader, in which tags arrive in batches in accordance with a time-homogeneous compound Poisson process and the distribution of service is general with system capacity being infinite, and following SIRO queue discipline.

6.1. Analytical evaluation

In a mobile environment, tags may continuously arrive, depart or stay randomly. To obtain an in depth knowledge about the efficiency of the system and in identifying an approximate identification time of an arriving tag, it is vital to have an understanding on the number of tags who are waiting to be identified.

Based on the results of Krishnakumar and Pavai Madheswari [30], the mean number of tags in the system (L_S) under the steady state conditions is given by the Eq. (1):

$$P_S(z) = P_0\{\{[z + (1-z)\gamma^*(\lambda)]zR^*(w(z))[1 - pS^*(w(z))]$$
$$+ pzS^*(w(z)) - z\}\frac{(\alpha - \delta) + (1-z)\gamma^*(\lambda)[S^*(w(z))\alpha p - \alpha]\}}{(z - H^*(w(z)))}$$

where

$$P_0 = \frac{\lambda^*(\lambda) - \alpha(p + \lambda E(S)) - \overline{\alpha}(1 + \lambda E(R))}{\delta q\gamma^*(\lambda) + (\alpha - \delta)q(1 + \lambda E(R))}$$

$$L_s = \left(\frac{d}{dz}P_S(z)\right)_{z=1} \tag{1}$$

where $P_S(z)$ is the total probability generating function of the number of tags in the system.

$$(\alpha - \delta)\{\lambda^2 E(R^2) + 2E(R)\lambda + 2(1 + \lambda E(R))$$
$$= \frac{[1 - \gamma^*(\lambda) + \lambda E(S)]\} + 2\lambda E(S)\delta\gamma^*(\lambda)}{2[(\alpha - \delta)(1 + \lambda E(R)) + \delta\gamma^*(\lambda)]}\lambda^2(\alpha E(S^2) + \overline{\alpha}E(R^2))$$
$$+ 2\lambda(p\alpha E(S) + \overline{\alpha}E(R))$$
$$+ \frac{2(1 - \gamma^*(\lambda))\{\alpha(\lambda E(S) + p) + \overline{\alpha}(1 + \lambda E(S))\}}{2[\gamma^*(\lambda) - \alpha(\lambda E(S) + p) - \overline{\alpha}(1 + \lambda E(R))]}$$

Traffic intensity is measured to evaluate the average occupancy of a reader during a specified period of time. It helps in determining the steady state condition of a reader, and also predicts when the system is probable to reach an explosive state. Since tags may arrive and depart randomly in Poisson fashion, it is highly necessary to analyze the system capability. The probability that the reader is busy is given by Eq. (2):

$$P_{busy} = \underset{z \to 1}{lt} Q^*(z, 0)$$

$$P_{busy} = \frac{\lambda E(S)}{q} \tag{2}$$

Let W_S denote the average time a tag spends in the system under steady state. Then due to Little's formula, W_S is given by Eq. (3),

$$W_s = \frac{L_S}{\lambda} \tag{3}$$

Since it is assumed that the reader emits waves until all tags are identified, the moments $E(R)$ and $E(R^2)$ is assigned as 0. In addition, due to successful commencement and the assumption of non-break down (repair) state of the reader, α and δ are assigned as 1. With reference to $\gamma^*(\lambda)$, the value depends on the retrial rate from the orbit (inactive state).

6.2. Simulation set up in mobile environment

The simulation experiment is conducted on a single reader and with tags arriving at the rate of 5, 6, 7, 8 and 9 tags per second. Table 2 illustrates the environment for the conduct of the experiment. Further, the assumptions listed for static environment is also taken into account.

Further to simulate the scenario of staying and departing tags, tags may stay and depart with the probability $p = 0.6$ and $q = 0.4$, respectively. To understand the simulation of staying and departing tags, presence of 12 tags in the reader's interrogation zone is considered. The simulator randomly chooses five tags (value rounded to whole number) to depart the zone (whether identified or unidentified) and seven tags (values rounded to whole number) are considered as staying tags.

Table 2 Simulation set up in mobile environment.

| Parameter	Value
Terrain	5 m × 5 m
Number of UHF RFID reader	1
Tag's arrival rate	5, 6, 7, 8, and 9 tags/s
Length of tag ID	96 bits
Probability that a tag may stay (p)	0.6
Probability that a tag may depart (q)	0.4
Time spent for a RFID reader to determine a idle responses	0.5 ms
Duration of a silence command	0.19 ms
Number of Bits transmitted to issue a silence command	5 bits
Data rate as per ISO 15693	26 kbps
Power consumed by the RFID reader during	
Scanning	180 milli watts
Idle	30 milli watts
Sleep	50 milli watts

Table 2 demonstrates the simulation set up in mobile environment.

Besides, the arriving tag may either wait in the orbit or participate in the ongoing identification cycle based on the query. Similarly, the departing tag may either be an unidentified or an identified tag.

7. Simulation results

The simulation set up is done as mentioned in Section 5.1. Tags in the simulation environment are designed to have randomly distributed Ids. The metrics namely collision, idle response, bit transmission, energy spent and identification time are noted down after the identification of every 200th tag.

After identifying 1000 tags, the number of collisions that occurred while implementing LaMSBA is 945, illustrating a decrease by 57.78% with AQS (Fig. 2). LaMSBA is able to decrease the collision profoundly due to the construction of a query by considering the collided bit from the left and the right simultaneously such that the number of tags responding to a query will be limited. The substantial decrease is also evident with respect to the idle response as observed in Fig. 3. LaMSBA is able to diminish the generation of idle responses as queries are constructed deterministically only based on the tags' response of the least and most bits.
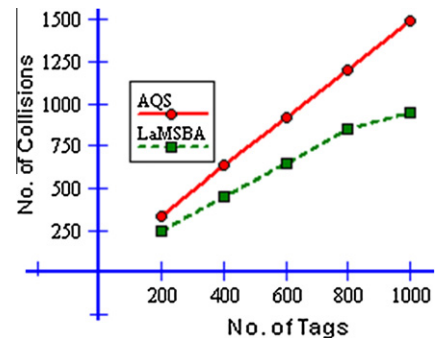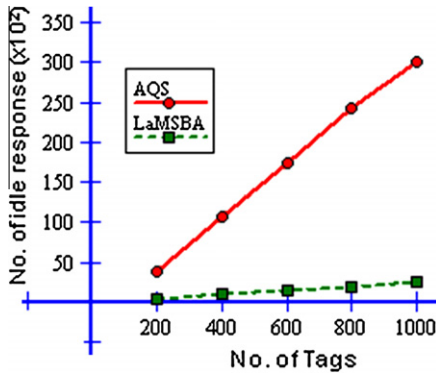

Figure 2 No. of collisions of AQS and LaMSBA.
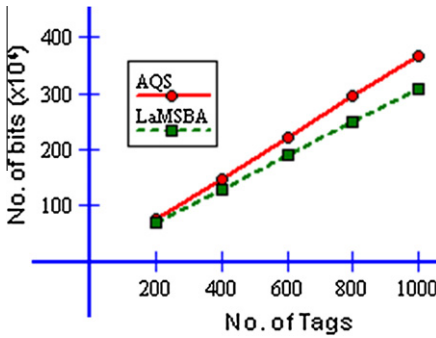
Figure 3 Idle response of AQS and LaMSBA.

Figure 4 No of bits transmitted by AQS and LaMSBA.

Figure 5 Energy spent during tag's response.
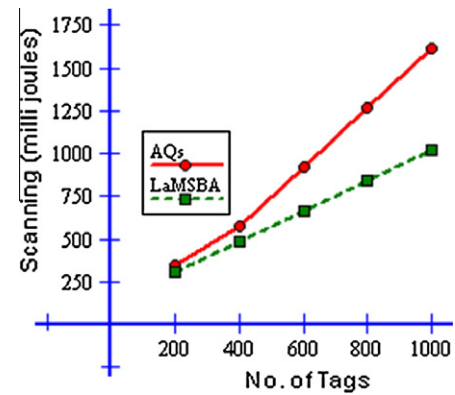
Figure 6 Energy spent by the reader during scanning.
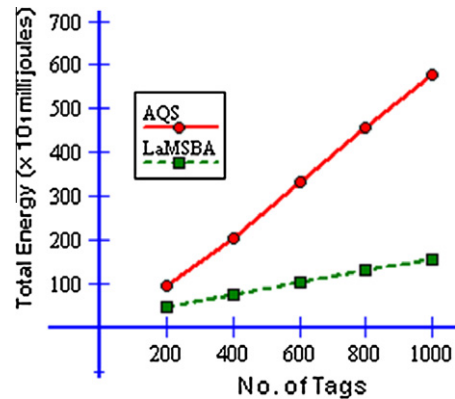
Figure 7 Total energy consumed by AQS and LaMSBA.

The overall reduction in bit transmission (Fig. 4) is found to decrease by 19.52% when compared with AQS. This decrease is possible due to the instant singulation of tags. While AQS follow the usual procedure of browsing bits from left to right, LaMSBA takes all possible efforts to ensure that the number of tags responding to a particular query is limited, so that the reader uses less number of queries to identify a certain tag in the zone.

In the process of reducing the number of bits transmitted, LaMSBA has helped in saving energy while scanning by 58.68% when compared with AQS. On the other hand, the reduction in transmission of bits from tags has allowed the reader to spend 556.01 milli joules when AQS spent 882.27 milli joules (Fig. 5). Fig. 6 shows the energy spent by the reader during scanning.

Over all LaMSBA spends 2612.26 milli joules when AQS spent 5755.95 milli joules. This momentous reduction is only made possible by the reduction of idle response during the identification process.

When observed the time taken for bit transmission, AQS needed 14.7 s however LaMSBA has reduced the need to 12.29 s owing to the reduction in the number of bits received by the reader (Fig. 7).

7.1. Analytical evaluation

Table 3 summarizes the identification rate of AQS and the proposed algorithm LaMSBA. As seen in Table 3, the number of tags identified in a second, with respect to LaMSBA is 58.51 tags, when compared to AQS whose identification rate is 33.59 tags/s.
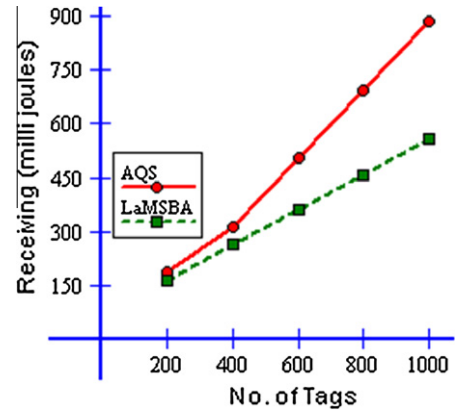
Table 3 Identification rate of LaMSBA.

Algorithm	Time taken (s)	Identification rate (tags/s)
AQS	29.77	33.59
LaMSBA	17.09	58.51

The simulation environment follows the details presented in Section 6.2 and Table 2.

Due to the increase in identification efficiency by 74.20%, arrival rates 13, 14 and 15 tags/s are also subjected for study. Using Eq. (1), the mean number of tags in the system is calculated. This mean is obtained after observing the simulation experiment for 25 min.

As observed in Table 4, the mean number of tags waiting for service in the system has significantly reduced solely due to the methodology of LaMSBA. LaMSBA is able to minimize the number of tags in the system drastically due to the singulation and identification of a tag much quicker than other algorithms. When the number of tags waiting was 15.58 tags in CBCID, LaMSBA had only six tags, in spite of the decrease in the retrial rate.

As seen in Table 4, when the arrival rate is set as 12 tags/s, LaMSBA has approximately six tags, while AQS reached the state of explosiveness as soon as the arrival rate has crossed 9 tags/s. Also, since the system is stable (Table 5) even when tags arrive at the rate of 15 tags/s, it can be seen that LaMSBA has only 71 tags in the system as against AQS.

It can be seen in Table 5 that even when the number of arriving tags goes beyond 12, the system is still observed to be in steady state, while this was not noticed in AQS. The increase in the arrival rate has increased the traffic intensity gradually, rather than drastically as noticed in AQS. This is due to the increase in identification rate of LaMSBA, as the algorithm can identify 58.51 tags in a second as against the existing and proposed algorithms. Similar observation can be viewed in Table 6.

Table 4 Analytical and simulation results of LaMSBA.

Arrival rate (tags/s)	Analytical results		Simulation results (averaged)
	Retrial rate	No. of tags	
5	22	1.519	3
6	32	1.413	4
7	36	1.848	4
8	41	2.360	4
9	48	2.814	5
10	50	4.339	5
11	62	4.539	6
12	67	6.885	9
13	81	7.719	11
14	85	16.770	18
15	93	71.340	75

Table 5 Analytical results of LaMSBA – traffic intensity.

Tags/s	Traffic intensity	
	AQS	LaMSBA
5	0.372125	0.213625
6	0.446550	0.256350
7	0.520975	0.299075
8	0.595400	0.341800
9	0.669825	0.384525
10	–	0.427250
11	–	0.469975
12	–	0.512700
13	–	0.555425
14	–	0.598150
15	–	0.640875

Table 6 Analytical results of LaMSBA – mean waiting time.

Tags/s	Traffic intensity	
	AQS	LaMSBA
5	1.654597	0.303887
6	0.939170	0.235585
7	0.809340	0.264022
8	1.127323	0.293306
9	4.325669	0.312772
10	–	0.433903
11	–	0.412706
12	–	0.573785
13	–	0.593770
14	–	1.197912
15	–	4.756400

8. Conclusion

The introduction of Improved Least and Most Significant Bit Algorithm (LaMSBA) can be considered as one of the better alternatives to minimize the number of idle responses that is experienced by the RFID systems. In conventional Query Tree based algorithms, tags respond from the left to the right of the query or from the right to the left of the query as followed where as tags in LaMSBA respond the collided bit from left and right simultaneously. This approach has brought in significant improvement in the tag identification process, which is confirmed by the extensive simulation experiments conducted. The simulations also assure that the repeated generation of idle responses is minimized, which is always found to be one of the prime causes for the increase in identification delay and energy consumption. For the identification of 1000 tags, the number of collisions incurred is found to be 945, total bits transmitted is 3,073,812 bits, and the total energy spent is 2612.26 milli joules. The total time spent on identification is 17.09 s, representing an increase in identification efficiency by 74.20% when compared to the existing AQS.

The analytical results calculated also support the efficiency found through simulation experiments conducted. The system is found to be stable even when tags arrive at the rate of 15 tags/s. In addition, the mean waiting time of a tag is found to be approximately 0.64 s and the number of tags waiting in the system is approximately 71 tags. Since the algorithm is able to identify tags in the least time (approximately 58 tags in a s), when compared to AQS (approximately 33 tags in a s), it can be concluded that LaMSBA can be considered as a desirable choice for implementation in time and data critical passive RFID systems.

The future research directions with respect to LaMSBA will be to test in a real time environment with varying tag population. With extensive avenues opened for RFID deployments, the collision resolution algorithms implemented in these systems should further be in a position to handle varying tag ID lengths.

References

[1] Yüksel Mehmet Erkan, Yüksel Asım Sinan. RFID technology in business systems and supply chain management. J Econ Social Stud 2011;1(1):53–71.

[2] Li Bo, Yang Yuqing, Wang Junyu. Anti-collision issue analysis in Gen2 protocol. Auto-ID Labs White Paper WP-HARDWARE-047; 2009. p. 1–34.

[3] Hayes JF. An adaptive technique for local distribution. IEEE Trans Commun, COM 1978;26:1178–86.

[4] Rohatgi A, Durgin GD. RFID anti-collision system using the spread spectrum technique. Technical Report; 2005, p. 1–19.

[5] Choi JH, Lee D, Lee H. Query tree-based reservation for efficient RFID tag anti-collision. IEEE Commun Lett 2007;11(1):85–7.

[6] Parveen Katheeja, Abdul Khader Sheik, Ahamed Rabbani Munir. B-bit pooling memoryless algorithm for anti-collision resolution in passive RFID tags. In: IEEE international conference communication control and computing technologies; 2010. p. 357–65.

[7] Yu S, Zhan Y, Wang Z, Tang Z. Anti-collision algorithm based on jumping and dynamic searching and its analysis. Comput Eng 2005;31:19–20.

[8] Wong CP, Feng Quanyuan. Grouping based bit slot aloha protocol for tag anti-collision in RFID systems. IEEE Commun Lett 2007;11(2):946–8.

[9] Klair Dheeraj, Chin Kwan Chu, Raad Raad. A survey and tutorial of RFID anti-collision protocols. IEEE Commun Surv Tutor 2010;12(3):400–19.

[10] Schwartz M. Telecommunication networks protocols. Modeling and analysis. USA: Addison-Wesley; 1998, p. 201–15.

[11] Finkenzeller K. RFID handbook fundamentals and applications in contactless smart cards and identification. John Wiley and Sons, Ltd.; 2003.

[12] Dheeraj Klair, Chin K-W, Raad R. On the energy consumption of pure and slotted aloha based RFID anti-collision protocols. Comp Commun 2009;32:961–73.

[13] Cha JR, Kim JH. Novel anti-collision algorithms for fast object identification in RFID system. In: The 11th International conference on parallel and distributed systems. Korea; 2005. p. 63–7.

[14] Chen Wen-Tzu. An accurate tag estimate method for improving the performance of an RFID anti collision algorithm based on dynamic frame length ALOHA. IEEE Trans Autom Sci Eng 2009;6(1):9–15.

[15] Geng Shu-qin, Wu-chen Wu, Li-gang Hou, Wang Zhang. Anti collision algorithms for multi tag RFID. Radio Freq Ident Fundamen Appl Bring Res Practice 2010:103–12.

[16] Tsybakov BS, Mikhailov VA. Free synchronous packet access in a broadcast channel with feedback. Problemy Predachi Inform (USSR) 1978;14(4):32–59.

[17] Capetanakis JI. Tree algorithms for packet broadcast channels. IEEE Trans Inform Theory 1979;25:505–15.

[18] Hush DR, Wood C. Analysis of tree algorithms for RFID arbitration. In: The IEEE international symposium on information theory. Mexico City, USA; 1998. p. 107.

[19] Law C, Lee K, Siu K-Y. 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. Toronto, CA; 2000. p. 75–84.

[20] Chen Yuan-Hsin, Horng Shi-Jinn, Run Ray-Shine, Lai Jui-Lin, Chen Rong-Jiang, Chen Wei-Chih, Pan Yi, Takao Terano. A novel anti-collision algorithm in RFID systems for identifying passive tags. IEEE Trans Indust Inform 2010;6(1):105–20.

[21] Bonuccelli MA, Lonetti F, Martelli F. Randomized hashing for tag identification in RFID networks. In: Technical report: computer communications, and networks; 2005. p. 1–8.

[22] Jacomet M, Ehrsam A, Gehrig U. Contact-less identification device with anti-collision algorithm. In: Conference on circuits systems, computers and communications. Athens, Greece; 1999. p. 1–6.

[23] Choi H-S, Cha J-R, Kim J-H. Fast wireless anti-collision algorithm in ubiquitous ID system. In: Vehicular technology conference; 2004. p. 4589–92.

[24] Myung J, Lee W, Srivastava J. Adaptive binary splitting for efficient RFID tag anti-collision. IEEE Person Commun Mag 2006;10(3):144–6.

[25] Alsalih Waleed, Ali Kashif, Hassanein Hossam. Optimal distance based clustering for tag anti collision in RFID systems. In: IEEE conference on local, computer networks; 2008. p. 266–73.

[26] Ryu J, Lee H, Seok Y, Kwon T, Choi Y. A hybrid query tree protocol for tag collision arbitration in RFID systems. In: IEEE ICC*. Scotland; 2007. p. 5981–6.

[27] Shin J-D, Yeo S-S, Kim T-H, Kim SK. Hybrid tag anti-collision algorithms in RFID systems. Berlin/Heidelberg, UK: Springer; 2007, p. 693–700.

[28] Parveen Katheeja, Khader Sheik Abdul, Rabbani Munir Ahamed. Cluster bit collision identification for recognizing passive tags in RFID system. Commun Comp Inform Sci 2011;133(2):190–9.

[29] Tseng DF, Lin ZC. Anti-collision with the aid of interference cancellation and tag set portioning in radio frequency identification systems. IET Commun 2009;3(1):143–50.

[30] Krishnakumar B, Pavai Madheswari S. MX/G/1 retrial queue with multiple vacations and starting failures. Operat Res 2003;40:115–37.

[31] Shih DH, Sun PL, Yen DC, Huang SM. Taxonomy and survey of RFID anti-collision protocols: short survey. Computer Communications 2006;29(11):2150–66.

[32] Jung-Sik Cho, Jea-Dong Shin, Sung Kwon Kim. (2008), RFID Tag anti-collision protocol: query tree with reversed IDs. In Proceeding international conference on advanced communication technology; 2008. p. 225–230.