International Journal of RF Technologies 2 (2010) 135–154
DOI 10.3233/RFT-2010-005
IOS Press

Analysis of Tag Loss Ratio in dynamic RFID systems

J. Vales-Alonso*, E. Egea-López, M.V. Bueno Delgado
and J.J. Alcaraz-Espín
Technical University of Cartagena, Plaza del Hospital, Cartagena, Spain

Abstract. Radio Frequency Identification (RFID) systems promise a revolution in logistics and inventory applications. By means of wireless communication, a master device can detect and identify nearby electronic tags. Traditionally, the performance of RFID systems and their identification protocols has been analyzed for static configurations, that is, without considering incoming or outgoing tags, but just a fixed number of initially unidentified tags. However, many real scenarios cannot be consistently modeled that way. In this work we introduce a Markov model which allows us to study a dynamic RFID tag scenario, where a flow of tags is considered. This model can be used to compute the Tag Loss Ratio, which measures the ratio of outgoing unidentified tags to the incoming tags in the system, which is a critical metric in dynamic configurations. The analysis is carried out for two families of protocols used as medium access control in RFID, Framed Slotted Aloha and non-persistent CSMA. With the aim of validating the analysis predictions, we get simulation results, by means of a simulator. We evaluate a scenario similar to a real application, i.e. a mail control system based on RFID.

Keywords: Radio Frequency Identification (RFID), tag traffic, Markov analysis, Tag Loss Ratio

1. Introduction

Radio Frequency Identification (RFID) allows remote identification and tracking of items by means of wireless communications. Its use has been foreseen for a wide range of applications which spans from replacement of bar-code systems to location of containers in large cargo vehicles. Moreover it is considered one of the enabling technologies for the ubiquitous computing paradigm (Stanford, 2004).

A basic RFID cell consists of a *reader* device (also known as *master* or *interrogator*) and a (potentially large) set of RFID tags, which reply to the queries and execute the commands from the interrogator (see Fig. 1). The system operates as follows: Periodically the master transmits a *collection command* requesting the identification of tags in range. This command is answered only by tags not identified yet. When an identification round ends the master acknowledges all the correctly received identifications, making these tags quit from the rest of the identification process. This scheme is conceptually simple, but performance may be poor in case of tag collisions: When multiple tags receive simultaneously a collection command (as in

*Corresponding author: J. Vales-Alonso, Technical University of Cartagena, Plaza del Hospital No. 1, 30202 Cartagena, Spain. E-mail: javier.vales@upct.es.

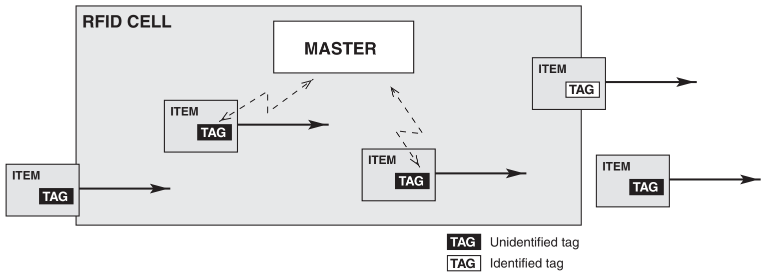

Fig. 1. RFID cell with tag traffic.

the example of Fig. 1), reply messages may collide and cancel each other, preventing identifications. Therefore, this procedure is often complemented with Medium Access Control (MAC) mechanisms that avoid tag collisions and improve performance, for instance, a Framed Slotted Aloha (FSA) protocol, a Carrier Sense Multiple Access (CSMA) contention algorithm, or some more sophisticated selection procedure for the identification cycle duration (Shih, Sun, Yen & Huang, 2006).

The reader must identify and communicate with the tags as quickly and reliably as possible, ensuring that all tags have been identified. As far as the authors know, such a problem has been studied for static scenarios, that is, a RFID cell with an initial population of unidentified tags is considered, and the goal is to analyze the time required for identifying the entire population. This topic has already been addressed in several works (Vogt, 2002; Egea-Lopez, Vales-Alonso, Martinez-Sala, Bueno-Delgado & Garcia-Haro, 2007; Bueno-Delgado & Vales-Alonso, 2010; Oh, Dighero, Thomas & Veeramani). However, such a static assumption is not realistic for a broad set of RFID applications. For example, consider a conveyor belt carrying industrial items to be identified, in such a system, there is a *flow of tags* (input/output traffic): While unidentified tags contend for identifying themselves, new items arrive and older ones leave the system (see Fig. 1). In these dynamic systems, the most important issue is to analyze the Tag Loss Ratio (henceforth, TLR), that is, the ratio of outgoing unidentified tags to the incoming tags in the system. So far, this problem has been studied by simulation (Floerkemeier & Wille, 2006). In the RFID context, the terms static and dynamic are commonly used to indicate that the anti-collision protocol changes adaptively some of its parameters to cope with changes in the load. On the contrary, we use those terms in this paper to indicate that the tags move in the coverage area but the MAC parameters remain fixed.

In this paper we analytically address this problem and propose a Markov model for a dynamic RFID system which allows us to compute the TLR, given a tag input traffic characterized by an arrival distribution. With this model we compute the TLR for both the FSA and CSMA medium access control approaches. To the best of the authors knowledge, this is the first approximation to the analysis of dynamic RFID systems. We validate the model, by means of simulation, considering a real scenario, i.e. a postal mail system based on RFID.

The remainder of this paper is organized as follows. In Section 2 we briefly review the related work in the area. In Section 3 the system under study is thoroughly described, for both FSA and CSMA configurations. Then, in Section 4 we derive a markovian model of such a system, and develop expressions for the tag loss probability. Section 5 provides a set of scenarios where the use of the former expressions is exemplified. Section 6 shows the simulation results of the real scenario considered. Finally, Section 7 concludes and outlines possible future works.

2. Related work

RFID devices are classified according to the source of energy of the tags: **Passive** ones do not have a power source and obtain the energy from the reader signal (by electromagnetic induction), whereas **active** ones incorporate their own battery. In general, the tag identification problem deals with identifying multiple objects with *(i)* minimal delay and power consumption, *(ii)* reliability, *(iii)* line-of-sight independence, and *(iv)* scalability.

In this paper we analyze two families of MAC protocols for RFID: FSA and CSMA. FSA is the more extended solution for both passive (Shih, Sun, Yen & Huang, 2006; Vogt, 2002) and active tags (ISO-IEC-180007, 2004; Class 1 Generation 2 UHF). However, we also consider CSMA since in (Egea-Lopez, Vales-Alonso, Martinez-Sala, Bueno-Delgado & Garcia-Haro, 2007) it is shown that the use of non-persistent CSMA as anti-collision mechanism for active RFID tags improves performance and scalability. In that paper we supported this solution by studying analytically the performance of quasi-optimal non-persistent CSMA as an anti-collision mechanism in a static scenario. In this paper, we provide a model for the dynamic case for both families and again it is shown that CSMA provides better scalability. The trade off is that CSMA usually requires the tags use batteries, whereas FSA can be implemented by passive RFID tags.

Both CSMA and framed slotted ALOHA have been extensively studied (Kleinrock & Tobagi, 1975; Wieselthier, Ephremides & Michaels, 1988), but in the context of classical MAC protocols, focusing on the channel utilization and access delay. In RFID, on the contrary, the appropriate performance metrics are the identification delay and TLR. Although the method proposed by Wieselthier (Wieselthier, Ephremides & Michaels, 1988) can be adapted to compute the TLR, it only holds for FSA whereas our model is not specifically focused on FSA, and can be extended to other MAC protocols.

Few analysis of RFID protocols performance can be found in the literature. Splitting algorithms are addressed in (Hush & Wood, 1998). Since this class of protocols is deterministic, performance is evaluated in terms of the average number of time slots needed to complete the process. Vogt (2002) analyzes the identification process of framed slotted ALOHA as a Markov chain but only the static case is considered. In this paper we use a Markov model, but, unlike Vogt, we do consider tag arrivals and departures.

Finally, let us remark again that our analysis is applied to dynamic *scenarios* instead of protocols. Most of the RFID literature is focused on the evaluation of adaptive (dynamic) algorithms for anti-collision protocols, that is, on how to adapt the protocol parameters to cope with the number of tags in coverage. An extensive review of this topic can be found in (Bueno-Delgado, Vales-Alonso & Gonzalez-Castaño, 2009). In our case, the protocol parameters are fixed. To further clarify it we propose the following classification pairs: Movement and Protocol (M-P). Then, for movement, a scenario where tags move and enter and leave the coverage area, will be dynamic, whereas if tags remain in coverage until all are identified, it will be static.

For protocols, if the protocol parameters does not change it is called fixed, whereas if they change, it is called adaptive. In this paper we address the dynamic-fixed case according to this classification, whereas most of the literature is focused on the static-adaptive case.

3. System operation

In this work we consider the RFID scenario depicted in Fig. 1. We assume an incoming flow of tags entering the coverage area of a reader (RFID cell), moving at the same speed (e.g. modeling a conveyor belt). Therefore, all tags stay in the coverage area of the reader during the same time. Let us remark, that even though the speed is fixed, the number of tags that enter the coverage are is random, which models that tags are randomly located on the belt. Every tag not identified during that time is considered as lost. As stated in the introduction, a tag is identified when its identification number (ID) is correctly received and acknowledged by the reader. Once acknowledged, a tag withdraws from the identification process, that is, does not try to send its ID again. The reader periodically requests IDs sending a *collection* packet. Tags try to send its ID after receiving a query packet. The actual operation depends on the protocol in use, as we will discuss next.

3.1. Carrier sense multiple access

The operation of the identification protocol when using CSMA is as follows: After receiving a collection command from the reader all tags listen to the channel for k *micro-slots*, where k is randomly chosen in the interval $[1, \ldots, K]$. K denotes the maximum number of micro-slots, and it is a configuration parameter of the system. If the channel remains idle after k micro-slots, the tag sends its ID. Otherwise, it withdraws until the next collection command (next cycle). Notice that collision is possible if two or more tags choose the same k. If there is no collision, the reader sends an ACK-Collection command, which indicates the tag identified and asks for more IDs. The remaining nodes start the process again. Figure 2 illustrates this mechanism. Note that in this case, in a given collection cycle, there can only be zero or one identifications.

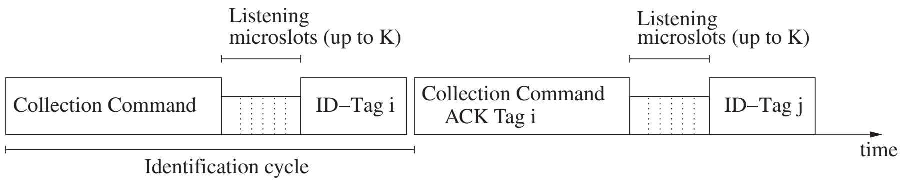

Fig. 2. Anti-collision procedure with CSMA.

3.2. *Framed slotted ALOHA*

Both ISO 18000-7 and EPC "Gen 2" (ISO-IEC-180007, 2004; Class 1 Generation 2 UHF) define a similar anti-collision procedure that we generically call Framed slotted ALOHA (FSA). In both cases, a population of tags start the identification process after receiving a collection command from the interrogator. At this moment, nodes randomly select a slot with a uniform distribution and transmit their ID in the selected slot. Let us denote the number of possible slots to choose as *frame length*, K. If two or more nodes select the same slot, a collision occurs. For each slot with a single reply, the interrogator sends an ACK packet which enforces the tag to sleep, preventing it from participating again in the identification process. Thus, the acknowledged tags (already identified) withdraw from contention in the following rounds. Figure 3 illustrates this process. We refer to a collection command plus the

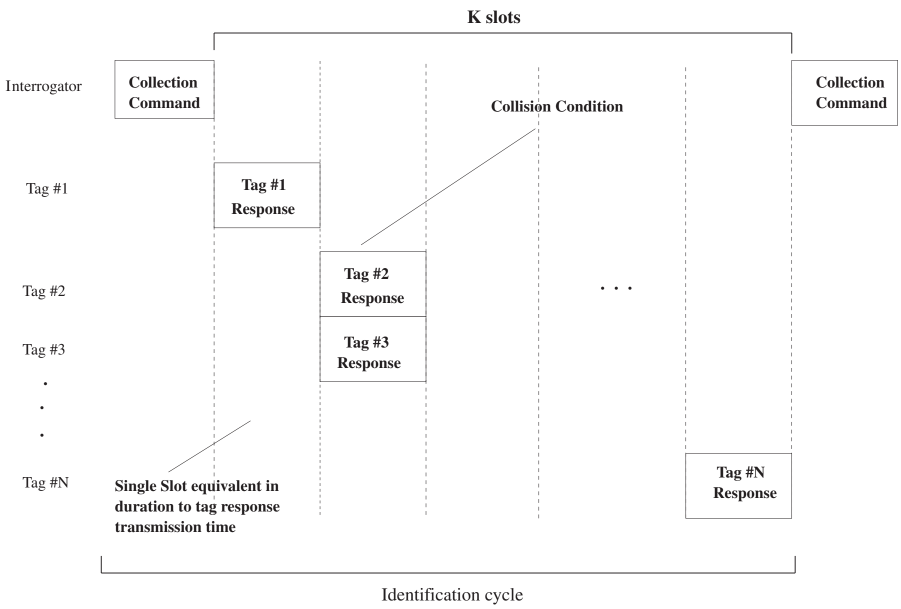

Fig. 3. Anti-collision procedure of ISO 18000-7, reproduced from (ISO-IEC-180007, 2004).

K slots as an *identification cycle*. It should be noted that unlike CSMA, with FSA an identification cycle involves several slots and more than one tag can be identified, up to K tags can be identified in one cycle.

Comparing CSMA and FSA identification time is not straightforward since the identification cycles have a different nature in both systems. Typically, several CSMA-cycles can be contained in a single FSA-cycle, since the micro-slots duration is much smaller than a slot duration in FSA. Furthermore, the cycle length in CSMA is variable, since depends on the random slot chosen, while it is fixed in FSA. Therefore, we provide performance results in terms of TLR. A fair time comparison between both access methods require a careful selection of parameters, and operating with time units instead of "cycles". Such issue is beyond the scope of this paper since here we focus on the model and TLR, interested readers may consult (Egea-Lopez, Vales-Alonso, Martinez-Sala, Bueno-Delgado & Garcia-Haro, 2007).

4. System model and analysis

Thereafter, the following notation and conventions are used:

- Probabilities are denoted as $Pr\{\text{Event}\}$.
- Random variables are denoted as ν and stochastic processes as X.
- A row vector is denoted as $\vec{V}$.
- The i-th component of a vector is denoted $(\vec{V})_i$.
- $\vec{a} \cdot \vec{b}$ is the dot product of vectors $\vec{a}$ and $\vec{b}$.
- $\vec{1}$ is a row vector (of appropriate dimension) whose components are equal to 1.

Let us define an *identification cycle*, as the interval of time of duration T_c between consecutive collection requests (independently of the underlying medium access mechanism). Zero, one or more tags may be identified during each identification cycle. Additionally, let us denote T as the time a tag remains in the RFID cell. To simplify, let us assume that T is a positive integer multiple of T_c, that is, the tags stay in the RFID cell for a given number of collection cycles. Let us denote N as the number of cycles in the coverage area plus one. Therefore, T can be expressed as $T = (N-1)T_c$. Once a tag has entered the coverage area, it should be identified in the following $N-1$ identification cycles. Otherwise (if it reaches the cycle N), it is lost. Indeed, this assumption allows us to use the expression "a tag in the i-th cycle", which refers to a tag that remains in the system for a time in the interval $[(i-1)T_c, iT_c]$, for $i = 1, \ldots, N$. To avoid confusions between the absolute number of cycles elapsed since system startup and the relative identification cycle of one tag, we will explicitly use the word *stage* henceforth to denote the relative identification cycle of one tag in the system (Vales-Alonso, Bueno-Delgado, Egea-Lopez, Alcaraz-Espin & Garcia-Haro, 2007), that is, the number of collection requests that have been sent since a tag entered the system.

In addition, note that incoming tags entering the system *after* a collection command have been issued do not participate in the current identification cycle since they do not receive the collection packet. Therefore, we can model incoming traffic in our system as a discrete arrival process. Let us assume that the arrival process for the i-th cycle is modeled by a discrete stationary stochastic process $\boldsymbol{A}(i)$. Therefore, the number of arrivals does not depend on the cycle considered (we can denote it just $\boldsymbol{A}$). Besides, assume that $\boldsymbol{A} \leq H$, for some $H \in \mathbb{N}$, i.e. in a given time-slot the maximum number of new tags is H (which may be arbitrarily high, but finite). Finally, let us denote $a(h) = Pr\{h \text{ arrivals in } T_c\}$, for each $h = 0, \ldots, H$, as the probability distribution of the arrival process $\boldsymbol{A}$.

To provide a more practical view of the model, let us note that N models the time tags spend in coverage. Therefore, in a practical situation, it can be derived from real systems parameters like speed of tags and coverage range. Similarly, H models the maximum number of tags that can enter simultaneously the coverage area, which can be derived from the belt dimensions or the allocation of tags in box, for instance.

4.1. Discrete dynamics

The former assumptions allow us to express the dynamics of our system as a discrete model, evolving cycle by cycle, such that,

- Each tag belongs to one *stage* in the set $[1, \ldots, N]$.
- After a cycle, identified tags withdraw from the identifation process, and we consider that they leave the system.
- After a cycle, each tag unidentified and previously in the k-th *stage* moves to the $(k+1)$-th *stage*.
- If a tag enters *stage* N, it is considered out of the range of the reader, and, therefore, lost.
- At the beginning of each cycle, $\boldsymbol{A}$ new tags are assigned to *stage* 1.

For any arbitrary cycle, the evolution of the system to the next cycle only depends on the current state. Thus, a Markov model can be used to study the behavior of the RFID system. Next section describes this model.

4.2. Markovian model

Based on previous considerations, our system can be modeled by a homogeneous discrete Markov process $\boldsymbol{X}_c$, whose state space is described by a vector $\vec{E} = \{e_1, \ldots, e_N\}$, where each $e_j \in (0, \ldots, H)$, denotes *the number of unidentified tags in the j-th* stage *at the beginning of an identification cycle*. Figures 4 and 5 illustrates our model. It depicts the state of the system for two consecutive cycles, showing tags entering and leaving the system, in both no identification and identification scenarios (Vales-Alonso, Bueno-Delgado, Egea-Lopez, Alcaraz-Espin & Garcia-Haro, 2007).

Therefore, e_j is the number of tags which are going to start their j-th stage in coverage. Component e_1 represents the number of tag arrivals during the previous

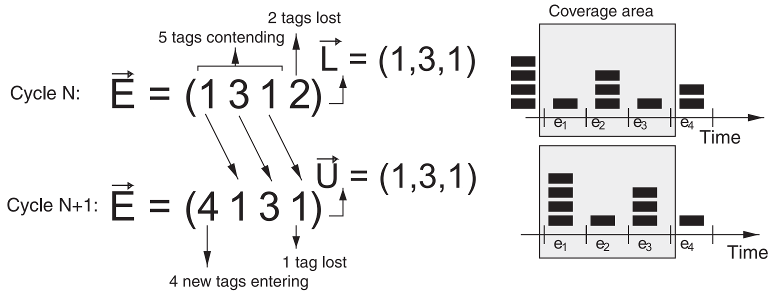

Fig. 4. Representation of a state transition. Case 1: No identification.

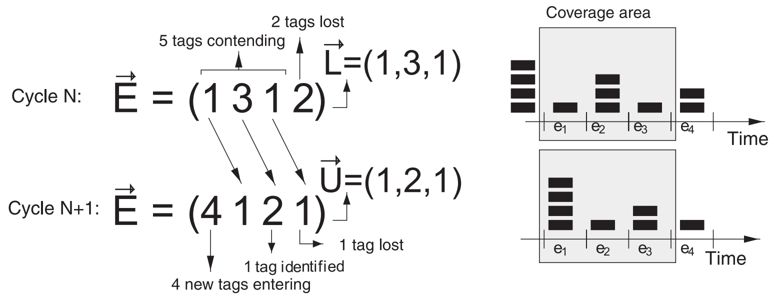

Fig. 5. Representation of a state transition. Case 2: Identification.

identification cycle (which do not contend since they had not received the collection packet). Finally, component e_N indicates the number of tags lost, since tags leave coverage area after $N-1$ cycles.

In order to enumerate the complete state space for our model let us define the mapping Ψ as a correspondence between the state vector and an enumeration of the possible number of states:

$$\Psi : [0, \ldots, H] \times^{(N)} \times [0, \ldots, H] \rightarrow [1, \ldots, (H+1)^N]$$

$$\vec{E} = \{e_1, e_2, \ldots, e_N\} \rightarrow \Psi(\vec{E}) = 1 + \sum_{j=1}^{N} e_j H^{j-1} \tag{1}$$

Obviously, Ψ is an injective mapping, and, since both sets have the same cardinality, it is a bijection mapping, and so there exists an inverse mapping Ψ^{-1}. This property allows us to define the i-th state in our model as the state whose associated vector is given by Ψ^{-1}. Let us denote $\vec{E}_i$ as the vector associated to the i-th state, i.e. $\vec{E}_i = \Psi^{-1}(i)$. Figure 6 provides an algorithm to compute the vector associated to a given state i.

```
function E_i = itovec(i,H,N)
  i = i-1; j = N;
  while j > 0 {
    e_j = i(H + 1)^{j-1};
    i = i-e_j(H + 1)^{-1};
    j = j-1
  }
```

Fig. 6. Algorithm to build the state space.

Finally, to simplify notation, the complete state space can then be represented by an $(H+1)^N \times N$ matrix E whose rows correspond to each of the $\vec{E}_i$ state vectors. Then every E matrix component e_{ij} corresponds to the j-th component of the $\vec{E}_i$ state vector.

Now, the goal is to describe the transition probability matrix P for the model, from every state i to another state j. Indeed, P depends on the underlying anti-collision protocol used. Once found, the stationary state probabilities can be computed as $\vec{\pi} = \vec{\pi}P$, where the vector $\vec{\pi}$ denotes the stationary probability distribution of the state space, $\vec{\pi} = \{Pr\{\vec{E}_1\}, \ldots, Pr\{\vec{E}_{(H+1)^N}\}\}$.

As stated in the introduction, we are mainly interested in the Tag Loss Ratio, TLR, i.e. the ratio of outgoing unidentified tags to the incoming tags in the system, that is, the probability that a tag leaves the system unidentified. Let us define λ_j as the average incoming traffic of unidentified tags in the j-th *stage*, for $j = 1, \ldots, N$. Therefore,

$$\lambda_j = \sum_{i=1}^{(H+1)^N} e_{ij}\vec{\pi}_i \tag{2}$$

Obviously, λ_1 is the average incoming traffic in the system (therefore, $\lambda_1 = \boldsymbol{A}$), and λ_N is the average number of unidentified tags leaving out of the system. That is,

$$\mathrm{TLR} = \frac{\lambda_N}{\lambda_1} = \frac{\sum_{i=1}^{(H+1)^N} e_{iN}\vec{\pi}_i}{\overline{\mathbf{A}}} \tag{3}$$

Therefore, in order to compute the TLR, we have to find the stationary state probabilities $\vec{\pi}$. So first, we have to compute the probability transition matrix (P) of the Markov model and then invert it. The transition probabilities depend on the anti-collision protocol used, since states change according to the number of identified tags. In the following section we derive the probability transition matrix (P) and then compute the associated TLR for the two families of identification protocols under study: CSMA and Frame Slotted ALOHA.

4.3. CSMA

In this case, let us first denote $s(\boldsymbol{f}, n)$ as the probability of success (one tag identified in the collection cycle) when n nodes select a contention micro-slot using probability distribution $\boldsymbol{f}$. Let us denote f_r as the probability that each contender independently picks micro-slot r, from $r = 1, \ldots, K$, i.e. $f_r = Pr\{\boldsymbol{f} = r\}$. Probability $s(\boldsymbol{f}, n)$ is computed in (Tay, Jamieson & Balakrishnan, 2004), and its expression is reproduced in equation (4).

$$s(\boldsymbol{f}, n) = n \sum_{z=1}^{K-1} f_z \left(1 - \sum_{r=1}^{z} f_r\right)^{(n-1)} \tag{4}$$

Besides, let us denote $p_{ij} = Pr\{\boldsymbol{X}_N = \vec{E}_j | \boldsymbol{X}_{N-1} = \vec{E}_i\}$, i.e. the transition probability from state i to state j.

To help build the transition probability matrix P let us define the auxiliary vectors $\vec{L}_i$ and $\vec{U}_i$ as:

$$\begin{aligned} \vec{L}_i &= \{e_{i1}, \ldots, e_{iN-1}\}, \\ \vec{U}_i &= \{e_{i2}, \ldots, e_{iN}\} \end{aligned} \tag{5}$$

That is, the $\vec{E}_i$ state vector without either the last or the first component. And let us define the *outcome* vector as $\vec{O}^{ij} = (\vec{L}_i - \vec{U}_j) = \{o_1^{ij}, \ldots, o_{N-1}^{ij}\}$. Figures 4 and 5 shows that by comparing (subtracting) the components of the auxiliary vectors $\vec{L}$ and $\vec{U}$ corresponding to the states i and j for two consecutive identification cycles N and $N+1$ we can know the outcome of the identification process (the outcome vector $\vec{O}^{ij}$). This observation help us to construct the transition matrix, and specially facilitates programming it when solving it numerically, as we will describe in sect. 4.5.

First, let us define the function $id(i, j)$ that operates on an outcome vector $\vec{O}^{ij}$ and provides the number of identified tags in a transition from a state i to a state j (see Fig. 5). Then, $id(i, j)$ is provided by equation (6).

$$id(i, j) = \vec{O}^{ij} \cdot \vec{1} \tag{6}$$

for each $i, j \in [1, \ldots, (N+1)^H]$.

Notice that function $id(i, j)$ just subtracts the auxiliary vectors L and U (of two consecutive cycles) and then sum the components of the resulting outcome vector, which is obviously the number of identifications that can occur in that transition. Notice also that, when applying this function to any pair of state vectors $\vec{E}_i$ and $\vec{E}_j$, if $e_{ik} < e_{j(k+1)}$ for some $k = 1, \ldots, N-1$ such transition is impossible (new tags cannot appear in *stages* other than *stage* 1). These impossible transitions will result in $id(i, j)$ providing a negative value.

Then, we can build the transition matrix P for the Markov process $\boldsymbol{X}_N$ taking into consideration that:

- As stated in section 3, using CSMA only one identification is possible in each cycle.
- In each cycle, the number of contending tags is given by

$$M = \vec{L}_i \cdot \vec{1} \tag{7}$$

for a given state i.
- $id(i, j) = 0$ for the no identification case (represented in Fig. 5(a)). In this case, there is not success in the collection cycle.
- $id(i, j) = 1$ when there is success, since only 1 tag can be identified in each CSMA identification cycle (see Fig. 5). The probability of success is uniformly distributed between all the contenders M, but we have to weight it by the number of contenders at each particular *stage*, that is, the probability that identification occurs at the *stage* k considering the initial state i, is e_{ik}/M. In this case, only the component with index k of the outcome vector is equal to 1 and the rest are all 0. This component provides the *stage* in which occurred the identification. Let us define an auxiliary function that provides such an index,

$$\gamma(i, j) = k, \quad \text{if} \quad o_k^{ij} = 1 \tag{8}$$

Notice that since $id(i, j) = 1$ this is an injective function.
- In the two previous cases the probability of the arrival of new tags must be taken into account, which corresponds to $a(e_{j1})$. Since the arrival process in independent from the identification process, the joint probability is directly computed.
- Otherwise, the transition is impossible, and thus it has a null probability.

Summing up, equation (9) represents the transition probability P for the CSMA anti-collision protocol,

$$p_{ij} = \begin{cases} a(e_{j1})[1 - s(\boldsymbol{f}, M)], & \text{if} \quad id(i, j) = 0 \\ a(e_{j1})[s(\boldsymbol{f}, M)]\frac{e_{i\gamma(i,j)}}{M}, & \text{if} \quad id(i, j) = 1 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

4.4. Framed slotted aloha

For a given collection cycle, let us denote K as the number of available contention slots (see section 3) and M the number of contending tags. In this case, let us define the random variable $s(K, M)$ that indicates the number of contention slots being filled with exactly with 1 tag, for a given number of slots and competing tags. The mass probability function of $s(K, M)$ has been computed in (Vogt, 2002), and is reproduced in equation (10).

$$Pr\{s(K, M) = k\} = \frac{\binom{K}{k} \prod_{i=0}^{k-1}(M - i)G(K - k, M - k)}{K^M} \tag{10}$$

Besides, the auxiliary function G is defined as follows,

$$G(a, l) = a^l + \sum_{i=1}^{l} \left\{ (-1)^i \prod_{j=0}^{i-1} \{(l - j)(a - j)\} (a - i)^{l-i} \frac{1}{i!} \right\} \tag{11}$$

Henceforth let us denote $Pr\{s(K, M) = k\}$ as $s_k(K, M)$.

As stated in section 3, using FSA, up to K tags may be identified in a single collection cycle. Therefore, possible cases range from $id(i, j) = 0$ to $id(i, j) = K$. The probability of $id(i, j)$ successful identifications is again uniformly distributed between all the contenders, but we have to weight it by the number of particular tags at each *stage*. Thus, given a transition from state i to state j, we take into account the probability of all the possible ways of getting $id(i, j)$ with equation (12).

$$\nu(i, j) = \frac{\prod_{k=1}^{N-1} \binom{e^{ik}}{o_k^{ij}}}{\binom{M}{id(i, j)}} \tag{12}$$

From equations (6), (7), (10) and (12) the transition matrix P can be computed as shown in equation (13).

$$p_{ij} = \begin{cases} a(e_{j1})s_0(K, M), & \text{if} \quad id(i, j) = 0 \\ a(e_{j1})\nu(i, j)s_{id(i,j)}(K, M), & \text{if} \quad id(i, j) \in [1, \ldots, K] \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

Let us remark that in both cases, FSA and CSMA, the transition probabilities do not change over time, since they only depend on the state.

4.5. *Programming the transition matrix*

To finish this section, let us remark that the previous analysis also provides an algorithm to implement a program that computes the transition matrix to obtain the stationary probabilities. First, recall that each state can be obtained from an index i with the algorithm of Fig. 6. Then, we only need two nested loops for the i and j indexes, compute each corresponding state vector with the aforementioned algorithm, extract from this the arrays $\vec{L}_i$ and $\vec{U}_j$ and compute the outcome vector, all of them being simple vector operations: Addition, substraction and comparisons.

5. Examples and results

We have computed TLR for different values of H and N and for both CSMA and FSA. For CSMA we have used the Sift approximation to the optimal distribution for $\boldsymbol{f}$ derived in (Tay, Jamieson & Balakrishnan, 2004). As shown in (Egea-Lopez, Vales-Alonso, Martinez-Sala, Bueno-Delgado & Garcia-Haro, 2007) this distribution performs and scales better than FSA in the static scenario. We have used $K = 8$ contention micro-slots. For FSA we have selected $K = 8$ slots. In both cases, for the arrival process $\boldsymbol{A}$ we have selected a truncated Poisson distribution, with parameter λT_c:

$$a(h) = \frac{(\lambda T_c)^h}{h! \sum_{i=0}^{H} \frac{(\lambda T_c)^i}{i!}}, \quad h = 0, \ldots, H \tag{14}$$

However, as discussed in sec. 3.2, CSMA identification cycles are different than FSA ones. One tag at most can be identified in a CSMA identification cycle, whereas up to 8 tags might be identified in a FSA one. Thus, to keep incoming traffic roughly comparable, λT_c range spans from 0.2 to 0.9 tags in CSMA whereas for FSA it spans from 1 to 7.

The results are shown in Figs 7–10. Let us first recall that N model the time that tags spend in coverage and H the maximum number of tags that can enter the coverage area simultaneously. The results show that, keeping fixed N, TLR increases as the

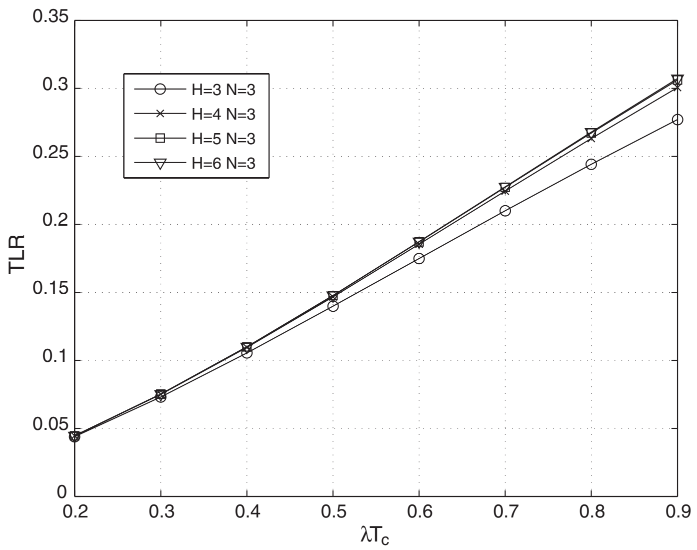

Fig. 7. TLR results for CSMA with Sift distribution (parameter $M = 32$) with 8 contention micro-slots and Poisson arrivals, $N = 3$ and $H = 3$ to $H = 6$.

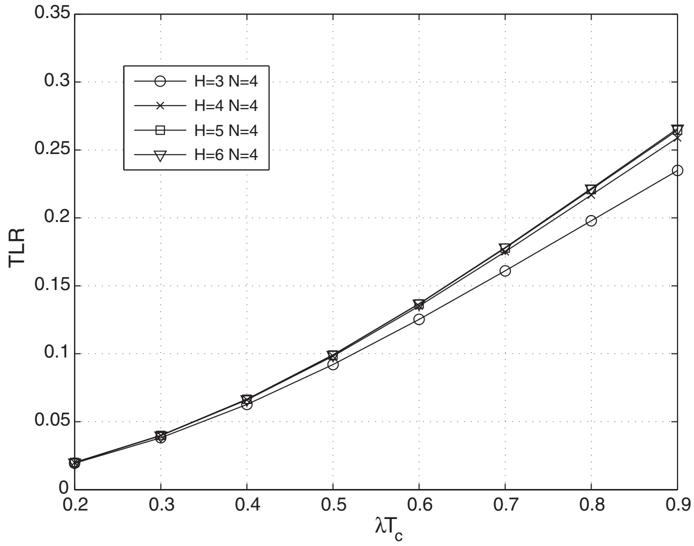

Fig. 8. TLR results for CSMA with Sift distribution (parameter $M=32$) with 8 contention micro-slots and Poisson arrivals, $N=4$ and $H=3$ to $H=6$.

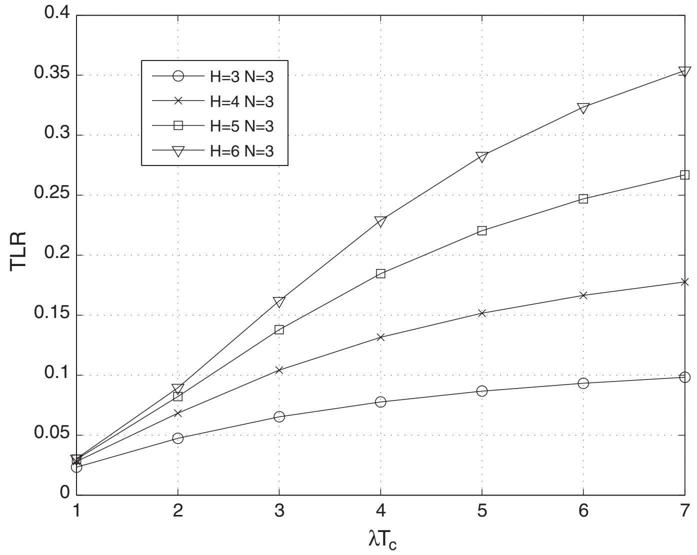

Fig. 9. TLR results for FSA with 8 slots and Poisson arrivals, $N=3$ and $H=3$ to $H=6$.

maximum number of arrivals H increases, as expected, that is, the system have to identify more tags in the same time. In addition, keeping fixed H, TLR decreases as

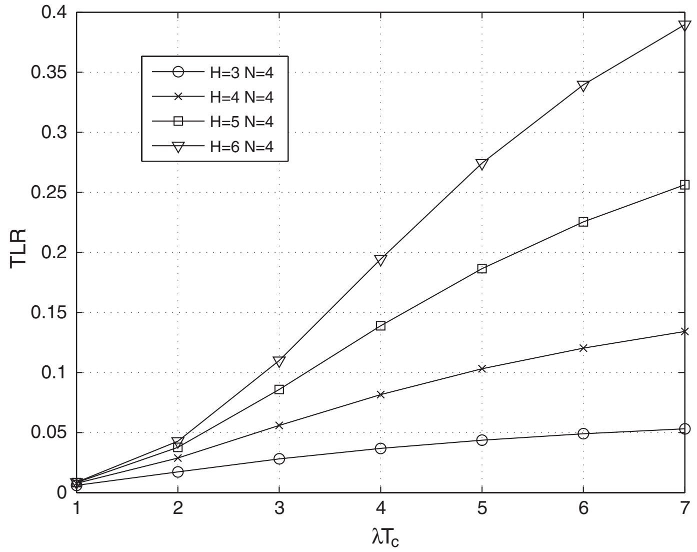

Fig. 10. TLR results for FSA with 8 slots and Poisson arrivals, $N=4$ and $H=3$ to $H=6$.

the maximum number of cycles in coverage N increases, also as expected, since it means that we have more time to identify a bounded maximum number of tags.

The fact that for a given H, TLR is higher when N is higher is because there are more tags contending simultaneously, and, therefore, there are more collisions, since the length of the frame is kept fixed. For example, for $H=6$ and $N=3$ there might be 12 tags contending at most whereas for $H=6$ and $N=4$ there might be up to 18 tags. The number of tags contending is the key to the anti-collision protocol performance. In fact, it shows that there is a trade-off between the size of the coverage area and the anti-collision protocol parameters (length of the frame). That is, one may intuitively think that if there is more time available to identify tags, the number of losses will decrease. However, in dynamic-fixed scenarios, if for any reason the protocol cannot identify tags quickly enough, due to a sudden peak, for instance, the new tags entering add to the previously ones contending and make the performance of the anti-collision protocol worse, creating an undesirable feedback that leads the system to being unable to identify any tags. If the size of the coverage area is smaller, this effect may be avoided at the cost of losing more tags at peak instants. It should be highlighted how CSMA handles better than FSA an increase in the number of new arrivals.

6. Simulation results: A mail control system

With the aim of validating the results of the analysis, a real RFID system has been simulated. We have considered a mail control system as a RFID scenario that fulfills the parameters assumed in the analytical model. The postal mail control system

cycles that tags are in coverage. According to the analytical model, we consider each RFID tagged envelop is in coverage for 3 identification cycles. We simulate FSA and CSMA protocols. First, the same scenarios proposed in the analytical model

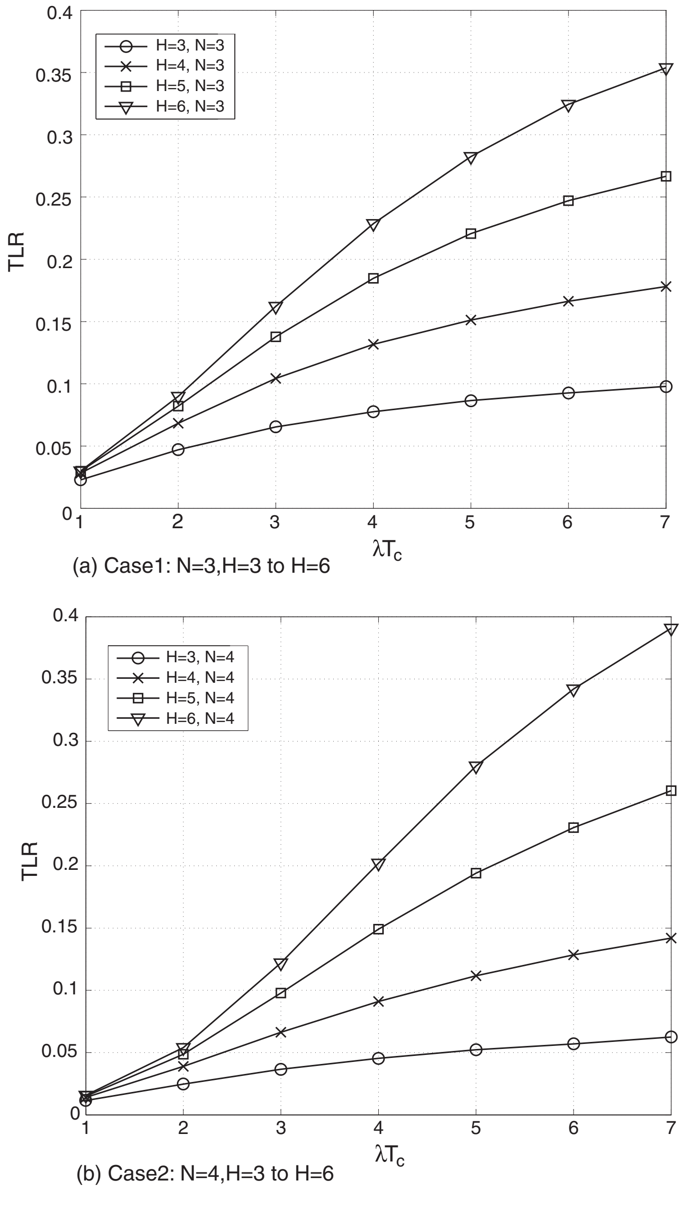

Fig. 12. Postal mail control system. TLR results for FSA with 8 slots.

have been simulated. Each identification cycle is set to $K = 8$ slots (or micro-slots in CMSA) and each slot (or micro-slot) lasts around 4 ms. Therefore the time the RFID tagged envelopes are in the reader coverage range is 100 ms. Although this short value for time in coverage may seem at first sight unrealistic, for H between 3 and 6, it may model for instance a machine that checks a maximum of 30 to 60 envelops per second, which is reasonable enough.

The results are shown in Figs 11 and 12. Regarding the analytical study, the simulation results match very well the analysis results. Only in Fig. 12(b), with $\lambda T_c = 7$, the TLR increases, but the difference is less than 0.01.

Finally, in order to get more information about the system considered, another example has been simulated. In this case we change the number of slots available, so the protocol is able to cope with more tags. We have implemented FSA protocol, with $K = 64$, $N = 4$ and $H = [3,6]$. Lamda has been increased from 10 to 60. The scenario proposed involves that, each RFID tagged envelop will be around 800 ms in coverage. Figure 13 shows the simulation result, where we can observe that, if we fix $H = 3$, the TLR reaches 10^{-4} and does not vary, independently the λT_c value. On the other hand, with $H = 6$, the TLR reaches up to 10^{-3}. It means that, one of each one thousand envelopes will be lost, a critical result in this kind of scenario.

From these results we can evaluate the influence of different protocol parameters, such as the number of slots, the arrival process, the time in coverage (conveyor belt

velocity), etc. As can be seen, protocol and scenario parameters can be tuned to achieve a given reliability, in terms of maximum admissible ratio of tags lost. As a final remark, the main drawback of the model is the high number of states that are involved, which cause problems to compute numerical results. In any case, it provides a valuable starting point for the design and evaluation of RFID systems.

7. Conclusions

In this paper we have shown a Markov model for the analysis of dynamic RFID scenarios, that is, where an arbitrary number of tags flows through the coverage area of the reader. To the best of our knowledge this is the first model proposed for such systems. This model has been used to derive the *Tag Loss Ratio*, TLR, which is the metric of interest for RFID systems, where the main goal is to *reliably* identify all the tagged items. Besides, the TLR has been computed for both FSA and CSMA. The main applications of this model are:

- Since it provides an exact solution for the TLR, it can be used to validate RFID simulators for more complex/realistic systems, i.e. the postal mail control system simulated in this paper.
- It can be used at design stage to evaluate the influence of different protocol parameters, such as the number of slots, in the system performance for several MAC protocols.

We leave the details on efficient implementation of the model with optimized programming libraries as future work. We are also working on new analytical models with a reduction in the number of states. Finally, we intend to apply these models and results in the design of optimized MAC protocols for RFID.

Acknowledgements

This work has been funded by the Spanish Ministerio de Innovación y Ciencia by the National project CALM TEC2010-21405-C02-02 (subprograma TCM) and also developed within the framework of "Programa de Ayudas a Grupos de Excelencia de la Región de Murcia", funded by Fundación Seneca, Agencia de Ciencia y Tecnología de la Región de Murcia (Plan Regional de Ciencia y Tecnología 2007/2010).

References

Bueno-Delgado, M.V. & Vales-Alonso, J. (2010). On the optimal frame-length configuration on passive RFID systems. *Journal of Network and Computer Applications*, In press

Bueno-Delgado, M.V., Vales-Alonso, J. & Gonzalez-Castaño, F.J. (November 2009). Analysis of DFSA anti-collision protocols in passive RFID environments, *Proc 35th Annual Conference of the IEEE Industrial Electronic Society (IECON'09)*, Porto, Portugal.

Egea-Lopez, E., Vales-Alonso, J., Martinez-Sala, A.S., Bueno-Delgado, M.V. & Garcia-Haro, J. (2007). Performance evaluation of non-persistent CSMA as anti-collision procedure for active RFID tags, *Proc of 5th International Conference on Wired/Wireless Internet Communications*, Coimbra, Portugal.

Class 1 Generation 2 UHF Air Interface Protocol Standard Version 1.0.9: Gen 2.

Floerkemeier, C. & Wille, M. (2006). Comparison transmission schemes for framed slotted ALOHA based RFID protocols. In *Proc of International Symposium on Applications and the Internet Workshops*, pp. 92–97.

Hush, D.R. & Wood, C. (1998). Analysis of tree algorithms for RFID arbitration. *Proc Int Symp on Information Theory*, p. 107.

ISO-IEC-180007 (2004). *Information technology – Radio frequency identification for item management – Part 7: Parameters for active air interface at 433 MHz*.

Kleinrock, L. & Tobagi, F. (1975). Packet switching in radio channels: Part I-carrier sense multiple access and their throughput-delay characteristics. In *IEEE Transactions on Communications*, *23*, 1400–1416.

Oh, C., Dighero, C., Thomas, S. & Veeramani, D. (2009). Tag performance evaluation and optimisation of Gen 2 reader communication for radio frequency identification deployment in supply chains. *International Journal of RF Technologies: Research and Applications*, *1*(3), 151–168, doi: 10.1080/17545730903228888, http://iospress.metapress.com/content/4745212712WXV7R1

Shih, D., Sun, P., Yen, D. & Huang, S. (2006). Taxonomy and survey of RFID anti-collision protocols. *Elsevier Computer Communications*, *29*, 2150–2166.

Stanford, V. (2004). Pervasive computing goes the last hundred feet with RFID systems. *IEEE Pervasive Computing*, *2*(2), 9–14.

Tay, Y.C., Jamieson, K. & Balakrishnan, H. (2004). Collision-minimizing CSMA and its applications to wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, *22*(6), 1048–1057.

Vales-Alonso, J., Bueno-Delgado, M.V., Egea-Lopez, E., Alcaraz-Espin, J.J. & Garcia-Haro, J. (2007). Markovian model for computation of tag loss ratio in dynamic RFID systems. *Proc of European Workshop on RFID Systems and Technologies*, Bremen, Germany.

Vogt, H. (2002). Efficient object identification with passive RFID tags. *Lectures Notes in Compute Science*, *2414*, 99–113.

Wieselthier, J.E., Ephremides, A. & Michaels, L.A. (1988). An exact analysis and performance evaluation of framed ALOHA with capture. *IEEE Transactions on Communications*, *37*(2), 125–137.