EURASIP Journal on Embedded Systems
a SpringerOpen Journal

RESEARCH Open Access

Performance analysis of optimal schedulers in single channel dense radio frequency identification environments

Javier Vales-Alonso*, Francisco Javier Parrado-García and Juan J Alcaraz

Abstract

Schedulers in radio frequency identification dense environments aim at distributing optimally a set of *t* slots between a group of *m* readers. In single-channel environments, the readers within mutual interference range must transmit at different times; otherwise, interferences prevent identification of the tags. The goal is to maximize the expected number of tags successfully identified within the *t* slots. This problem may be formulated as a mixed integer non-linear mathematical program, which may effectively exploit available knowledge about the number of competing tags in the reading zone of each reader. In this paper, we present this optimization problem and analyze the impact of tag estimation in the performance achieved by the scheduler. The results demonstrate that optimal solutions outperform a reference scheduler based on dividing the available slots proportionally to the number of tags in each reader. In addition, depending on the scenario load, the results reveal that there exist an optimum number of readers for the topology considered, since the total average number of identifications depend non-linearly on the load. Finally, we study the effect of imperfect tag population knowledge on the performance achieved by the readers.

1 Introduction

Passive radio frequency identification (RFID) is increasingly being used to identify and trace objects in supply chains, in manufacturing process, and so forth. These environments are characterized by a large number of items with attached tags which flow on conveyor belts, inside pallets or boxes, and the like, entering and leaving facilities. In large realistic installations, several readers are commonly deployed; these are the so-called *dense reader environments*, comprising multiple readers within a mutual range.

In these scenarios, the rate of tags identified per reader is limited by the reader collision problems, namely:

- Reader-to-tag interferences (RTI) occur when two or more readers, irrespectively of the working frequency, transmit at the same time, overlapping their read ranges (reader-to-tag range) and powering the same tags. For instance, in Figure 1, if readers R and R' are feeding tag A simultaneously, tag is not able to produce a correct response to any of the readers.
- Reader-to-reader interferences (RRI) occur when two or more readers working at the same frequency are in mutual range, that is, one reader that powers a tag within its reader-to-tag range can receive stronger signals from other readers, ruining the weaker signal from the tag. For example, in Figure 1, tag B cannot be read by R if at the same time R' tries to read the tag C.

Another kind of interference is tag-to-tag interference which is internal to the reader's cell and is produced among tags competing to be identified by the reader. This latter type occurs even with a single reader, whereas the former ones (external) are only present with more than one reader. Indeed, the way of addressing internal and external interferences is completely different and independent. External ones are addressed by reserving (in real-time or with a preconfigured scheduling) resources to particular readers. Then, the reader uses these resources (time, frequency, power, etc.) to execute some algorithm to solve the tag-to-tag interference problem, as the static frame slotted ALOHA (static-FSA). Later in Section 2.1, we analyze the way how static-FSA and its derivate

*Correspondence: javier.vales@upct.es
Department of Information Technologies and Communications, Technical University of Cartagena, Cartagena,Spain

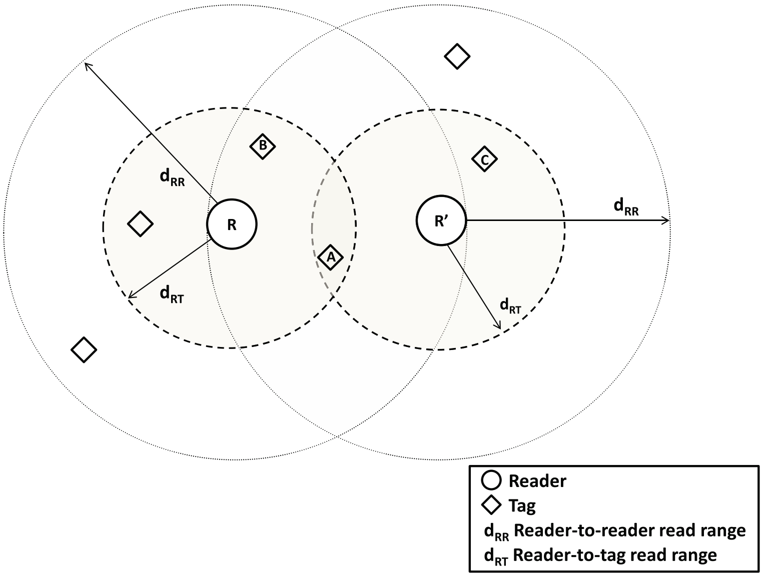
d_RR
B
C
R
R'
d_RR
A
d_RT
d_RT
○ Reader
◇ Tag
d_RR Reader-to-reader read range
d_RT Reader-to-tag read range

Figure 1 Interferences in dense reader environments.

dynamic-FSA work, since their operation is relevant to decide the readers' scheduling.

External interferences are directly related to the readers' output power, which delimit the interference range. For example, in Europe, output power may reach up to 2 W and guarantees a reader-to-tag range up to 10 m, while this may cause interferences with readers up to some hundreds of meters typically, determining interference ranges:

- If two or more readers are within two times the reader-to-tag range (d_{RT}), either part or the whole reading area overlaps, preventing tag operation. Hence, both RTI and RRI interferences are present. In this case, reader operation should be allocated at different working times.
- If the distance among the readers is between d_{RT} and the maximum distance determined by the RRI (d_{RR}), only RRI appears. The reader operation can be multiplexed either in frequency or in time.
- If the distance among the readers is larger than the maximum RRI distance, they do not suffer interferences.

Table 1 summarizes the restrictions applying to reader operation for dense reader environments.

Table 1 Reader operation restrictions versus d

	= Freq	≠ Freq
= Time	$d > d_{\mathrm{RR}}$	$d > d_{\mathrm{RT}}$
≠ Time	$d > 0$	$d > 0$

Therefore, in dense reader environments, the problem is how to distribute the reading resources available among the readers to perform optimally. The main parameters involved in this problem are the following:

- The number of readers, m.
- The number of available frequency channels, F.
- The number of time-slots available in each frequency, t.
- The topology of the readers.
- The implemented identification procedure in each reader (e.g., static-FSA, dynamic-FSA, Query-Tree protocols, etc.).
- The characteristics of the traffic of tags (e.g., static tags vs. tag flow, random vs. deterministic number of tags, etc.).

Current standards (see Section 2) propose some solutions to reduce collision issues but exclusively focused on minimizing RRI. On the other hand, a number of papers (see also Section 2) deal with minimization of the RTI but without considering reader-to-reader interferences.

In a previous paper [1], a particular simplified problem with two readers $m = 2$ in reader-to-reader range (*dual reader environment*) is addressed. Besides, in our previous paper [2], the scheduling problem for single-channel environments is firstly introduced, that is, we consider the case of any arbitrary number of readers (m) and for any particular network topology. Attending to the restrictions given above, in this case, the readers cannot transmit simultaneously if the reader-to-reader interferences are present, that is, if the distance between them is less than d_{RR} (note that this case also comprises reader-to-tag interferences).

In addition to [2], in this work, we provide insight on the impact of the schedulers derived from the knowledge of the tag population associated to each reader. To the best of our knowledge, all previous optimization models (see Section 2) have largely ignored the availability of this information. This information can be effectively exploited to construct a scheduler with the goal of maximizing the number of identifications in the whole interrogation period. In this work, we assume that this information is known and show how it can be used to develop an optimal scheduler. Moreover, we analyze the improvement obtained when this information is available and the effect on the expected performance when errors occur in tag estimation.

This resource allocation problem is addressed both for static and dynamic frame length identification procedures (which are described later in Section 2.1) and that the tags remain in coverage of their corresponding reader at least during the whole period of identification (t time-slots). The goal is to maximize the expected number of identified tags in the whole network.

The rest of the paper is organized as follows: In Section 2, the most relevant research proposals are shown. Section 2.1 describes the identification procedures commonly used in RFID readers. Section 3 describes the optimization model. Section 4 shows the performance results achieved by the optimal algorithm. Section 5 deals with the analysis of the impact of tag population estimation in the scheduler. Finally, Section 6 concludes and describes future works.

2 Related work

A number of proposal for coordinating dense reader environments have been presented in the literature; most of them are based on heuristic approaches and are, thus, suboptimal by nature. A summary of these works is contained in [3]. Besides, a number of papers [4-10] propose different system models and schedulers based on the optimization of some metric, defined upon the corresponding model.

Choi and Lee [4] propose a mixed integer linear program to minimize the reader interference problem as well as other performance metrics by selecting channel, timeslots, and output power for each reader. Their strategy is based on achieving a minimal signal-to-interference-plus-noise ratio for the signal received from tags, as well as on maximizing network utilization and minimizing power consumption. However, they neglect the availability of information about the number of tags present in the reading area of each reader and the operation of the underlying reading protocols, which are major factors determining the performance of the reading process.

Kim et al. [5] propose the TPC-CA algorithm based on a power control approach. It consists of controlling the reader output power optimally to reduce reader-to-reader collisions. Optimality criterion is related to minimize the area where interferences among readers occur.

Chui-Yu et al. introduces GA-BPSO in [6] a scheduler based on genetic algorithm and swarm intelligence meta-heuristics for single-channel environments. These schedulers aim at minimizing the overall sum of transaction times. However, these times are provided as parameters for the scheduler and are not based on the impact of resource allocation on the reading protocols.

Deolalikar et al. derive in [7] optimal scheduling schemes for readers in RFID networks for four basic configurations. As in our work, the authors aim at maximizing the number of identification within the scheduling period(t), but they model the performance of the reading process with an approximation: the number of tags identified increases linearly up to a saturation point. From that point on, the number of identifications remains constant. As we demonstrate in Section 2.1, this approach is not realistic for different tag-to-tag anti-collision protocol configurations (e.g., in static-FSA, there is a drop on the throughput). As in our work, only reader-to-reader interference (and thus single-channel) environments are considered.

The study of Mohsenian-Rad et al. [8] is the work more closely related to ours. The authors design two optimization-based distributed channel selection and randomized interrogation algorithms for dense RFID systems: FDFA (which is fully distributed and achieves a local optimum) and SDFA (semi-distributed and reach to the global optimum). In addition, the authors realistically assume that the reader may operate asynchronously. Similarly to our work, they consider a FDMA/TDMA scheduler, where the medium access control layer of the readers complies with EPCglobal Class-1 Gen-2 standard (therefore, a reader may allocate a number of interrogation frames within its allocated time). In this work, the authors focus on the probability that a reader starts an interrogation interval without experiencing either reader-to-reader or reader-to-tag collisions. The goal is to achieve max-min fairness in the network; as a result, the processing load is evenly distributed among all readers. However, this paper does not consider the knowledge about the number of contending tags in range of each reader. This information allows us to formulate the optimization problem in terms of reading efficiency (maximizing the number of tags identified in the overall time period). An additional contribution of [8] is to develop a protocol to construct the topology (i.e., reader-to-reader and reader-to-tag constraints) by exchanging some messages in three control channels.

This protocol may be implemented in other schedulers (like ours) to determine the network topology in real time.

Tanaka and Sasase [9] also determine an interference model which they apply later to formulate constraints in a binary integer linear program aimed at maximizing the ratio of total time where readers can successfully communicate with the tags and total interrogation time of the readers. As in our model, the goal is selecting suitable timeslot and channels for each reader. They also propose two heuristics (one distributed and one centralized) to solve the allocation problem efficiently.

Seo and Lee [10] propose a FDM/TDM scheduler (RA-GA) based on a reader-to-reader interference model, which seeks to maximize a utility function depending upon the operating time slots. This problem is solved using a genetic algorithm meta-heuristic.

As many of the previous works, neither in [9] nor in [10] the reading protocol or the current load (unidentified tags) of each cell is considered. Summarizing, to the best of our knowledge, all previous optimization models ignore the availability of information about the number of tags within the range of each reader. This information can be very effectively exploited to construct a scheduler with the goal of maximizing the number of identifications in the interrogation period. Besides, most previous works assume a model view from the physical layer perspective and are usually aimed at minimizing interference. This view has notable limitations since tag identification performance, and thus scheduling, heavily depends on the underlying tag-to-tag anti-collision protocol, as discussed in the next Section.

2.1 Tag identification procedure

The identification process involves communications between the reader and the tags and takes place in a shared wireless channel. Basically, the reader *interrogates* tags nearby by sending a *Query* packet (the exact format of this packet depends on the particular standard). The tags are energized by the reader's signal and respond to this request with their identification. When several tags answer simultaneously, a tag-to-tag collision occurs, and the information cannot be retrieved. Therefore, a tag-to-tag anti-collision mechanism is required when multiple tags are in range. ALOHA-based protocols, also called probabilistic or random access protocols, are the most prevalent in the UHF band. They are designed for situations in which the reader does not know exactly how many tags will cross its checking area. The most common ALOHA RFID protocol is FSA, a variation of slotted ALOHA. As in slotted ALOHA, time is divided into time units called *slots*. However, in FSA, the slots are subject to a super-structure called a 'frame'. Two options of the FSA are commonly used in the RFID technology:

1. *static frame length FSA (static-FSA).* The reader starts the identification process with an identification frame by sending a Query packet with information about the frame length (k slots) to the tags. The frame length is kept unchanged during the whole identification process. At each frame, each unidentified tag selects a slot at random from among the k slots to send its identifier to the reader. FSA achieves reasonably good performance at the cost of requiring a central node (the reader) to manage slot and frame synchronization. FSA has been implemented in many commercial products and has been standardized in the ISO/IEC 18000-6C [11], ISO/IEC 18000-7 [12], and EPCGlobal Class-1 Gen-2 (EPC-C1G2) standards [13].
2. *dynamic frame length FSA (dynamic-FSA).* When the tags outnumber the available slots, the identification time increases considerably due to frequent tag-to-tag collisions. On the other hand, if the slots outnumber the tags, many slots will be empty in the frame, which also leads to long identification times. Dynamic-FSA protocols were conceived to address this problem. They are similar to static-FSA, but the number of slots per frame is variable. In other words, parameter k may change from frame to frame in the Query packet to adjust the frame length. Dynamic-FSA operation is optimal in terms of reading throughput (rate of identified tags per slot) when the frame length equals the number of contenders [14]. Therefore, to maximize throughput, the reader should ideally know the actual number of competing tags and allocate that number of slots to the next frame. Different dynamic-FSA algorithms have been proposed to estimate the number of competing nodes based on the collected statistical information. The most relevant ones have been studied in depth in our previous papers [15,16].

In the next Section, both algorithms (static-FSA and dynamic-FSA) are considered in order to propose an optimal slot distribution for the single channel environment. In the case of static-FSA, the frame length is k for all readers; in the case of dynamic-FSA, we are assuming that each reader j actually knows the number of competing nodes at frame i ($n_{\mathrm{j,i}}$) and that the reader is adjusting $k_{\mathrm{j,i}} = n_{\mathrm{j,i}}$ if the number of the remaining available slots is greater than $n_{\mathrm{j,i}}$.

3 Optimal time distribution

Recall from the introduction that a dense-reader environment with the limitation of a single frequency channel

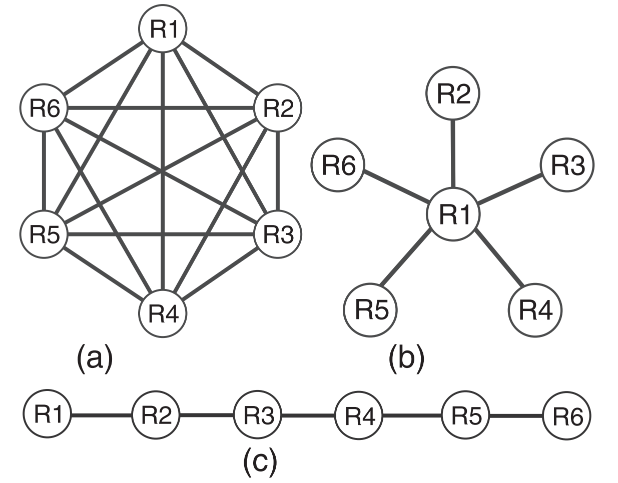

Figure 2 Example scenarios for $m = 6$. (**a**) Full-mesh topology. (**b**) Star topology. (**c**) Line topology.

Table 3 Star scenario

Number of tags	Φ	R1	R2	R3	R4
10	40.000	78	74	76	77
20	80.000	126	115	155	116
30	119.999	128	128	128	128
40	159.427	128	128	128	128
50	186.355	128	128	128	128
60	189.195	94	140	139	139
70	210.000	0	512	512	512
80	240.000	0	512	512	512
90	270.000	0	512	512	512
100	300.000	0	512	512	512

Optimal assignment of slots for the dynamic-FSA protocol.

F=1, m readers, and t slots available in the channel is assumed. In addition, for each reader $j = 1, \ldots, m$, let us denote

- n_{j}, the tags unidentified in the range of the reader j
- t_{j}, the number of slots assigned to reader j.

Let us remark that the methods used in dynamic-FSA tag-to-tag anti-collision protocols to determine the number of contenders can be directly applied in our case to estimate n_{j} in real-time (see [15] and [16] for details). Besides, topological dependencies among readers are defined by an $m \times m$ matrix $A = (a_{\mathrm{jj}'})$, the elements of which are 1 if reader j and j' cannot operate at the same time, and 0 otherwise.

Let $\varphi(n, t)$ denote the expected number of identified tags when n tags contend in t slots, and let us define Φ as the whole expected number of identified tags in the network, that is,

$$\Phi = \sum_{j=1}^{m} \varphi(n_{\mathrm{j}}, t_{\mathrm{j}}). \tag{1}$$

Then, the optimization problem can be stated as solving

$$\max_{\substack{t_{\mathrm{j}} \\ j=1,\ldots,m}} \; \Phi. \tag{2}$$

Subject to

$$t_{\mathrm{j}} \geq 0 \tag{3}$$

and

$$t_{\mathrm{j}} + I_{t_{\mathrm{j}}} \sum_{j'=1, j' \neq j}^{m} t_{\mathrm{j}'} a_{\mathrm{j}'\mathrm{j}}, \leq t \; \text{ for all } j = 1, \ldots, m, \tag{4}$$

where $I_{t_{\mathrm{j}}}$ is 1 if t_{j} is greater than 0, and 0 otherwise.

Constraint (3) expresses a basic limiting condition on the values assigned to the number of assigned slots. The key in our problem formulation is constraint (4) which

Table 2 Full-mesh scenario

Number of tags	Φ	R1	R2	R3	R4
10	40.000	128	128	128	128
20	80.000	126	115	155	116
30	119.999	128	128	128	128
40	159.427	128	128	128	128
50	186.355	128	128	128	128
60	189.195	94	140	139	139
70	189.113	110	70	166	166
80	188.949	80	162	190	80
90	188.992	0	212	89	211
100	188.905	157	157	99	99

Optimal assignment of slots for the dynamic-FSA protocol.

Table 4 Line scenario

Number of tags	Φ	R1	R2	R3	R4
10	40.000	85	86	165	221
20	80.000	118	118	124	248
30	120.000	153	153	171	171
40	160.000	173	170	169	173
50	199.913	174	169	169	174
60	236.153	176	168	168	176
70	254.481	188	162	162	188
80	263.693	215	172	125	215
90	273.451	242	180	90	242
100	287.462	256	256	0	453

Optimal assignment of slots for the dynamic-FSA protocol.

establishes local conditions to regulate the spatial reuse of the resources in our network. This condition states that the number of slots assigned to a reader j plus those assigned to its neighbors can not surpass the number of available slots. I_{t_j} is included since readers without slots assigned should be considered as disconnected, and no constraints have to be applied to that particular readers.

The former constraint guarantees that enough slots are available for each node in each neighborhood (set of nodes bonded with topological constraints, i.e., $a_{j'j} = 1$) to obey with the limit of t slots among all neighbors. Note that it does not guarantee that these slots can be allocated consecutively. However, this is not an issue since tags do not proceed with the next slot until a *QueryRep* packet arrives from the reader. Hence, even if the slots are not consecutively allocated, the tags perceive continuity and the identification can be performed seamlessly.

We must remark that this set of constraints produces feasible solutions regardless of the considered topology. However, in some cases (as we will show in the next section), the constraint is too strict and may lead to suboptimal solutions since space reutilization is limited. If the network graph has a large density (i.e., the number of edges is close to the maximal number of edges), the results provided by solving problem (1) will be close to the optimal solution with maximal space resource reutilization. Whereas, for sparse network graphs, the space reutilization will be small. The first kind of scenario will likely occur (due to the large reader-to-reader interference range) in facilities with non-screened readers; thus, the solutions found will be realistic.

3.1 $\varphi(n_j, t_j)$ computation for static-FSA

Finally, in order to solve the optimization problem, the expected number of identifications $\varphi(n, t)$ must be computed. The next sections deal with its computation both for static-FSA and for dynamic-FSA.

In this case, the reading process for each reader j consists of several consecutive reading frames of length k until all the t_j reading slots are eventually exhausted. It

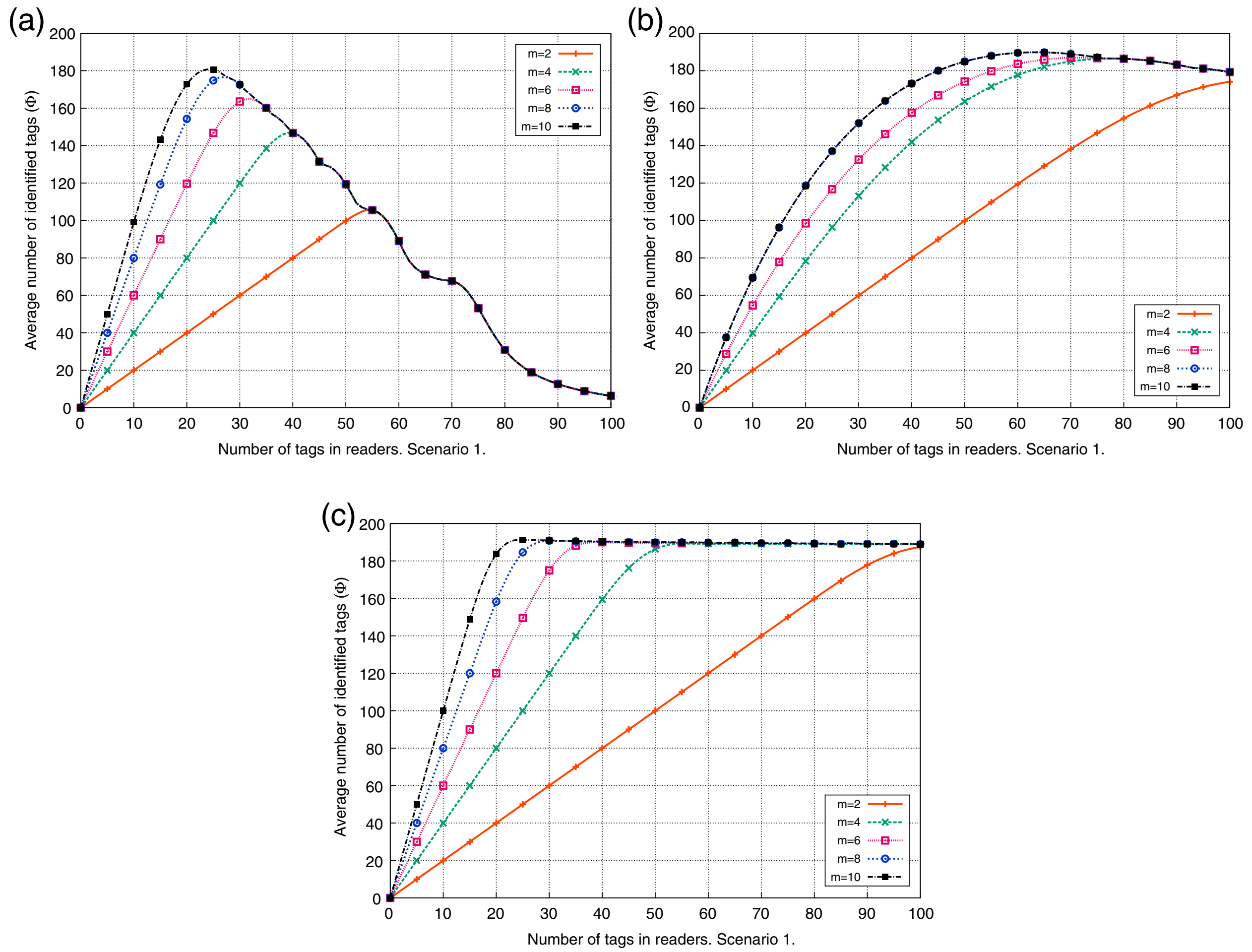

Figure 3 Expected number of identifications (Φ) versus n: Full-mesh scenario. (**a**) Static-FSA $k = 16$. (**b**) Static-FSA $k = 64$. (**c**) Dynamic-FSA.

is assumed that $t_j = ka$, being a a positive integer or zero. Given the last condition, and since expectation is a linear operator, $\varphi(n_j, t_j)$ can be computed as the sum of the average number of tags identified in the first frame $(\varphi(n_j, k))$ plus those identified in the remainder process $(\varphi(n_j - \eta, t_j - k))$, where η denotes the random number of tags identified in the first frame.

The former part can be computed if the distribution of the random variable η is known; so, let us denote $P(a|n_j, k)$ as the probability that a tags are identified if n_j tags contend in k slots. Then

$$\varphi(n_j, k) = \sum_{a=0}^{n_j} aP(a|n_j, k). \tag{5}$$

The second part, can be computed given the joint probability of identifying a tags in the first frame and a' in the remainder process if n_j tags contend in t_j slots, which we will denote as $P(a, a'|n_j, t_j)$. We obtain

$$\varphi(n_j - \eta, t_j - k) = \sum_{a=0}^{n_j} \sum_{a'=0}^{n_j} a' P(a, a'|n_j, t_j). \tag{6}$$

But, clearly $P(a, a'|n_j, t_j) = P(a|n_j, k)P(a'|n_j - a, t_j - k)$, and this leads to

$$\varphi(n_j - \eta, t_j - k) = \sum_{a=0}^{n_j} \varphi(n_j - a, t_j - k)P(a|n_j, k) \tag{7}$$

Appendix 1 demonstrates that the value of $P(a|n_j, t_j)$ is given by (where the technique in [17] was used to compute the probability $P(a|n, t)$)

$$P(a|n_j, t_j) = \frac{n_j!}{t_j^{n_j}} \binom{t_j}{a} \sum_{c=0}^{n_j - a} (-1)^c \binom{t - a}{c} \frac{(t_j - a - c)^{n_j - a - c}}{(n_j - a - c)!}. \tag{8}$$

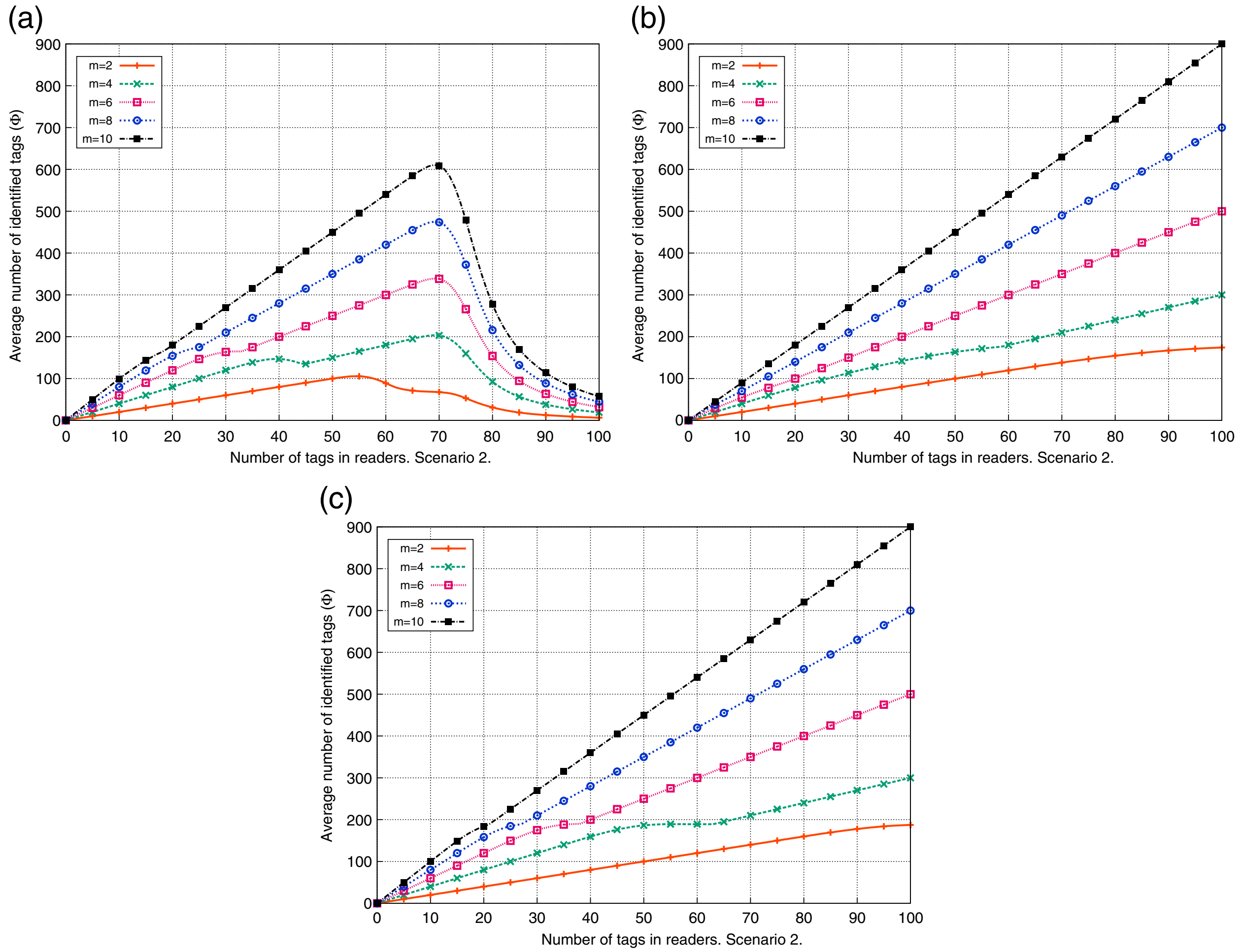

Figure 4 Expected number of identifications (Φ) versus *n*: Star scenario. (**a**) Static-FSA $k = 16$. (**b**) Static-FSA $k = 64$. (**c**) Dynamic-FSA.

Thus, the following recursive equation results:

$$\varphi(n_j,t_j)=\begin{cases}\sum_{a=0}^{n_j}(a+\varphi(n_j-a,t_j-k))P(a|n_j,k), \text{ if } t_j\geq k\\ 0, \text{ otherwise}\end{cases} \tag{9}$$

3.2 $\varphi(n_j, t_j)$ computation for dynamic-FSA

In this second case, the reading process for each reader j also consists of several reading frames but of variable length $k_{j,1}, k_{j,2}, \ldots$, until all the t_j reading slots are exhausted. Besides, denote the number of contenders in each frame as $n_{j,1}, n_{j,2}, \ldots$. Since the dynamic-FSA operation is used (see Section 2.1), the reader seeks to maximize reading throughput and allocates the optimal number of slots in each frame. That is, as much slots as the number of contending tags ($k_{j,i} = n_{j,i}$). This is possible while $n_{j,i} < t_j - \sum_{c=1}^{i-1} k_{j,c}$, that is, if the remainder number of slots is greater that the number of contenders. Otherwise, we assume that a last frame is allocated with all the remaining slots ($k_{j,i} = t_j - \sum_{c=1}^{i-1} k_{j,i}$).

Like in the previous case $\varphi(n_j, t_j)$ can be described through a recursive equation:

$$\varphi(n_j,t_j)=\begin{cases}\varphi(n_j,n_j)+\sum_{a=0}^{n_j}\varphi(n_j-a,t_j-n_j)P(a|n_j,n_j), \text{ if } n_j<t_j\\ \varphi(n_j,t_j) \text{ if } n_j\geq t_j.\end{cases}$$

From Equation (5),

$$\varphi(n_j,n_j)=\sum_{a=0}^{n_j} aP(a|n_j,n_j),$$

and

$$\varphi(n_j,t_j)=\sum_{a=0}^{n_j} aP(a|n_j,t_j), \text{ if } n_j\geq t_j.$$

Hence,

$$\varphi(n_j,t_j)=\begin{cases}\sum_{a=0}^{n_j}(a+\varphi(n_j-a,t_j-n_j))P(a|n_j,n_j) \text{ if } n_j<t_j\\ \sum_{a=0}^{n_j} aP(a|n_j,t_j) \text{ if } n_j\geq t_j.\end{cases} \tag{10}$$

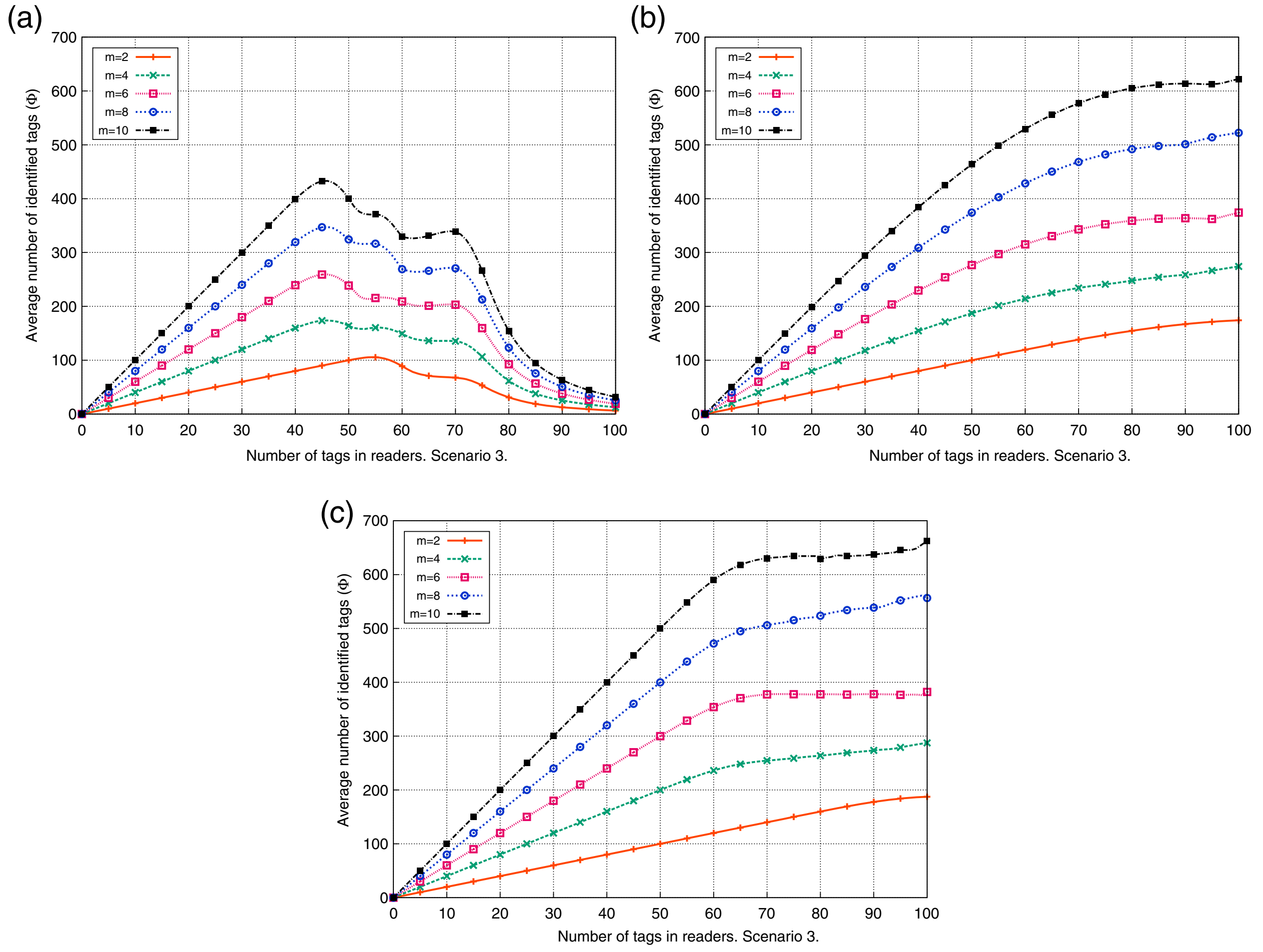

Figure 5 Expected number of identifications (Φ) versus *n*: Line scenario. (**a**) Static-FSA $k = 16$. (**b**) Static-FSA $k = 64$. (**c**) Dynamic-FSA.

4 Results

The optimal assignment has been computed in static-FSA and dynamic-FSA cases using the recursive formulas described in the previous section. Three representative scenarios (see Figure 2) have been selected. The edges (i.e., connecting lines) represent the existence of interference between two vertices (readers). On the first scenario, a full-mesh topology of m readers has been selected. It is a typical configuration in facilities, since the RRI distance is large (in the order of hundreds of meters) as discussed in the introduction. On the other hand, the star topology of m readers selected for scenario two represents another practical case, where readers are confined to some areas (e.g., by screening the reading area), and interferences are restricted to some particular pairs, exclusively between R1 and the other readers in this example. Finally, the line scenario is representative of an assembly line, where neighbor readers are in range.

Besides, the following parameters have been considered:

- $t = 512$,
- n tags to be identified at each reader, from 1 to 100 tags,
- $m = 2, 4, 6, 8$, and 10,
- and for static-FSA $k = 16$ and 64.

Our optimization algorithm has been implemented using the *General Algebraic Modeling System*, a high-level modeling system for mathematical programming and optimization, and AlphaECP, a MINLP (Mixed-Integer Non-Linear Programming) solver based on the extended cutting plane method. It allowed us to define our optimization problem directly from the mathematical description provided in Section 3.

Tables 2, 3, and 4 show the optimal configurations (slots assigned to each reader) for the dynamic-FSA protocol in all scenarios with $m = 4$. Let us remark that the optimal solutions are non-trivial, that is, can be obtained through an educated guess. Clearly, this solution is not unique: a circular permutation of the optimal solution, replacing the slots from Rj to R$(j+1)$ if $j < m$, and from Rm to R1 is also an optimal solution for the full-mesh scenario. The

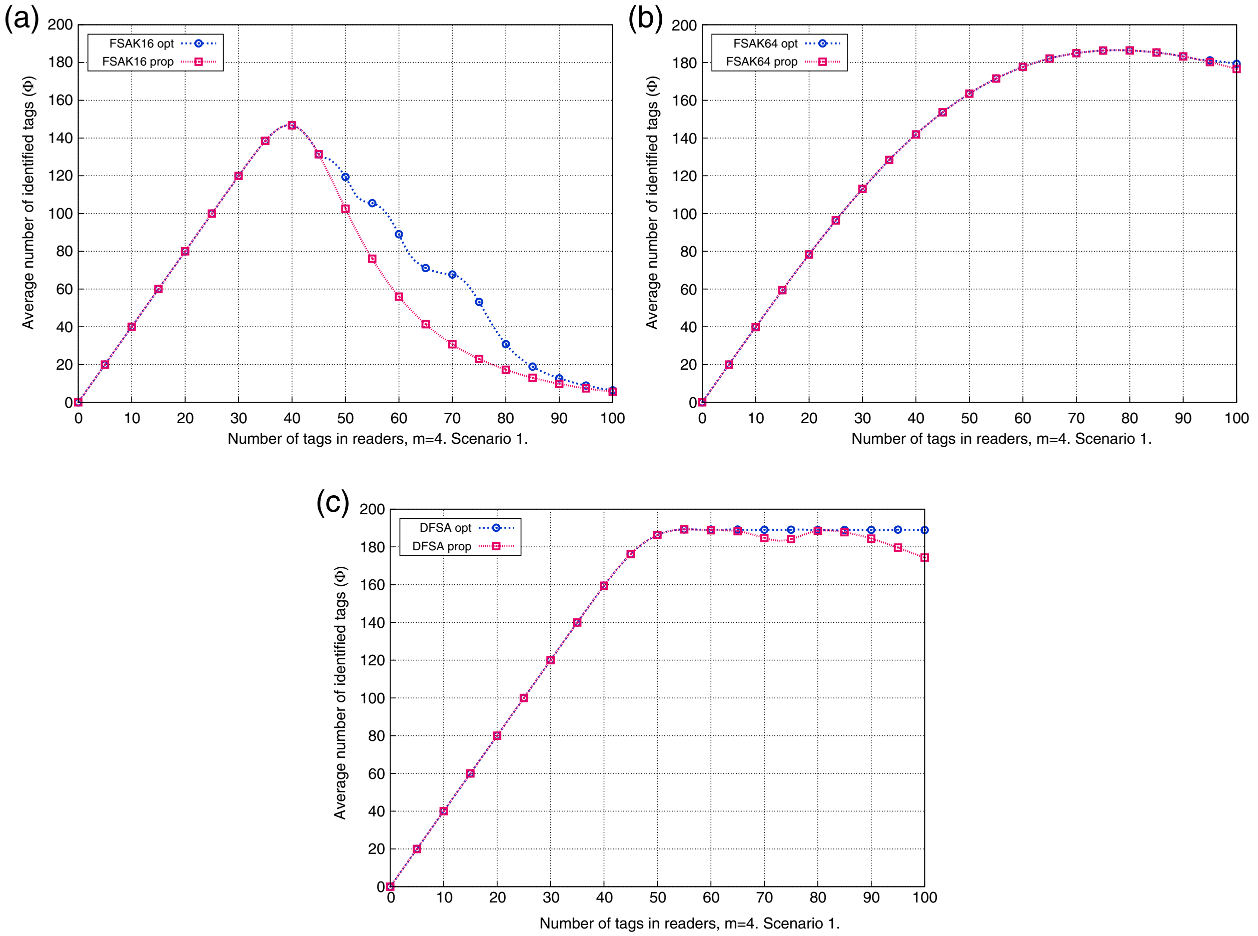

Figure 6 Optimal vs. proportional allocation for full-mesh topology, *m* = 4. (**a**) Static-FSA $k = 16$. (**b**) Static-FSA $k = 64$. (**c**) Dynamic-FSA.

same applies to the star scenario if the slots in R1 are kept constant while any permutation is applied to the rest of the readers and to the line scenario replacing the slots from Rj to R$(m-j)$.

In addition, note that the results obtained for the star scenario (Table 3) can be improved. For example, for $n=30$, after assigning 128 slots to R1, it would be possible to assign 384 to all remainder readers, which will provide a solution better than that obtained by solving problem (1). As discussed in Section 3, this is caused by the strict resource reutilization obtained by applying the set of constraints given by Equation (4). This problem does not appear for networks characterized by a dense graph, as the full-mesh scenario.

Besides, Figures 3, 4, and 5 show the expected number of tags identified (Φ) for all the possible values of m using the optimal assignments. Note that the resources available ($t=512$) are the same for all the configurations; however, the performance clearly varies. This illustrates how the underlying reading protocol determines the final system performance. Dynamic-FSA performs better than static-FSA assignment for both configurations of k (16, 64), as can be expected. This is reasonable since dynamic-FSA achieves an optimal reading throughput frame-by-frame while the number of available slots is at least equal to the number of contenders.

Another important result shown in these figures is the existence of saturation points in the system. That is, in some cases, the throughput does not increase when the load is increased. For dynamic-FSA, in all cases, the throughput never decreases; this is caused by the flexibility of dynamic-FSA to adapt to different loads. For static-FSA $k=64$, the effect is almost similar to the dynamic-FSA case, except in the full-mesh scenario where the throughput slightly decreases when the tag is beyond $n=60$. However, for all static-FSA $k=16$ cases, the effect of the load in the throughput is dramatic, with a throughput minima and a step decreasing performance. This is of considerable importance, since static-FSA $k=16$ is the default configuration of many readers in the

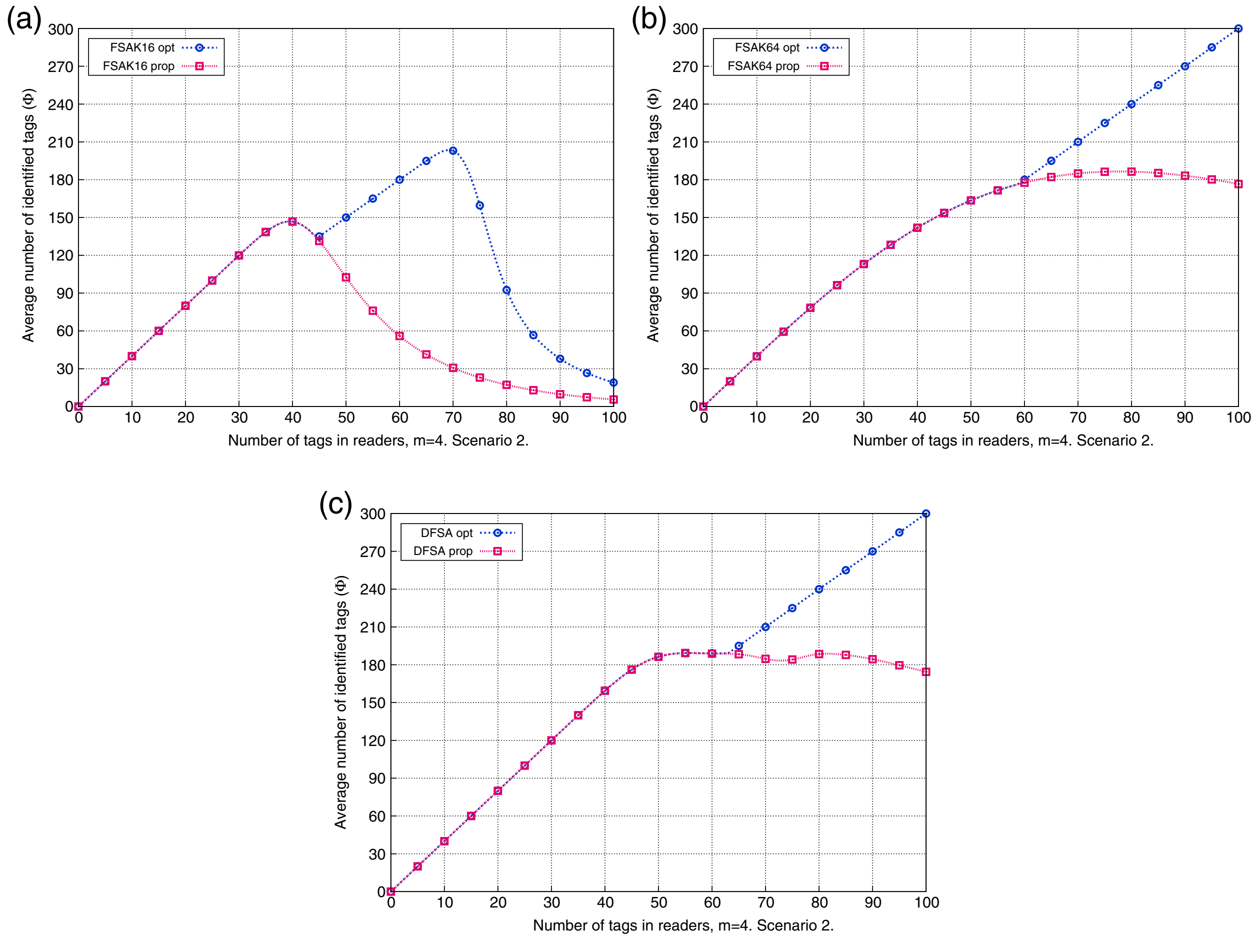

Figure 7 Optimal vs. proportional allocation for star topology, $m=4$. (**a**) Static-FSA $k=16$. (**b**) Static-FSA $k=64$. (**c**) Dynamic-FSA.

market, and this configuration leads to poor collective performance.

In addition Figures 6, 7, and 8 show, for $m = 4$, the performance of the optimal allocation versus a non-optimal allocation scheme selected for comparison, namely, using $\frac{1}{m}$ of time allocated to each reader ('proportional' resource sharing), that is, $t_1 = ... = t_4 = 128$. This heuristic is a natural choice, since the number of tags in the range of each reader is the same; therefore a good performance could be expected. In fact, the proportional scheme achieves in a range of n a performance nearly equal to the optimal one, as can intuitively be expected, in some cases (e.g., static-FSA $k = 64$ and dynamic-FSA in the full-mesh topology). However, for some cases, the allocation is clearly suboptimal (e.g., star and line scenarios for $n > 60$). Noteworthy, in the star scenario, there is a point ($n \geq 70$) where the best option is directly to disconnect the central reader. In this case, without restrictions in the network, the remainder readers can be allocated each all the 512 slots. A similar behavior occurs for the line scenario, disconnecting some readers when there is a high load (e.g., see Table 4; if $n = 100$ the third reader is disconnected).

5 Tag estimation impact on scheduler performance

The aim of this section is twofold:

(1) Quantify the improvement achieved in the scheduler when tag instant population estimation is available.
(2) Quantify the impact of tag population estimation errors on the performance achieved by the scheduler.

As stated in Section 2, previous works do not assume knowledge about the tag population and are mostly based on minimizing interferences. To establish a comparison between our model and a reference model that do not use population information, at least, we must focus on the same performance metric, i.e., the expected number of identifications (which can also be viewed as throughput).

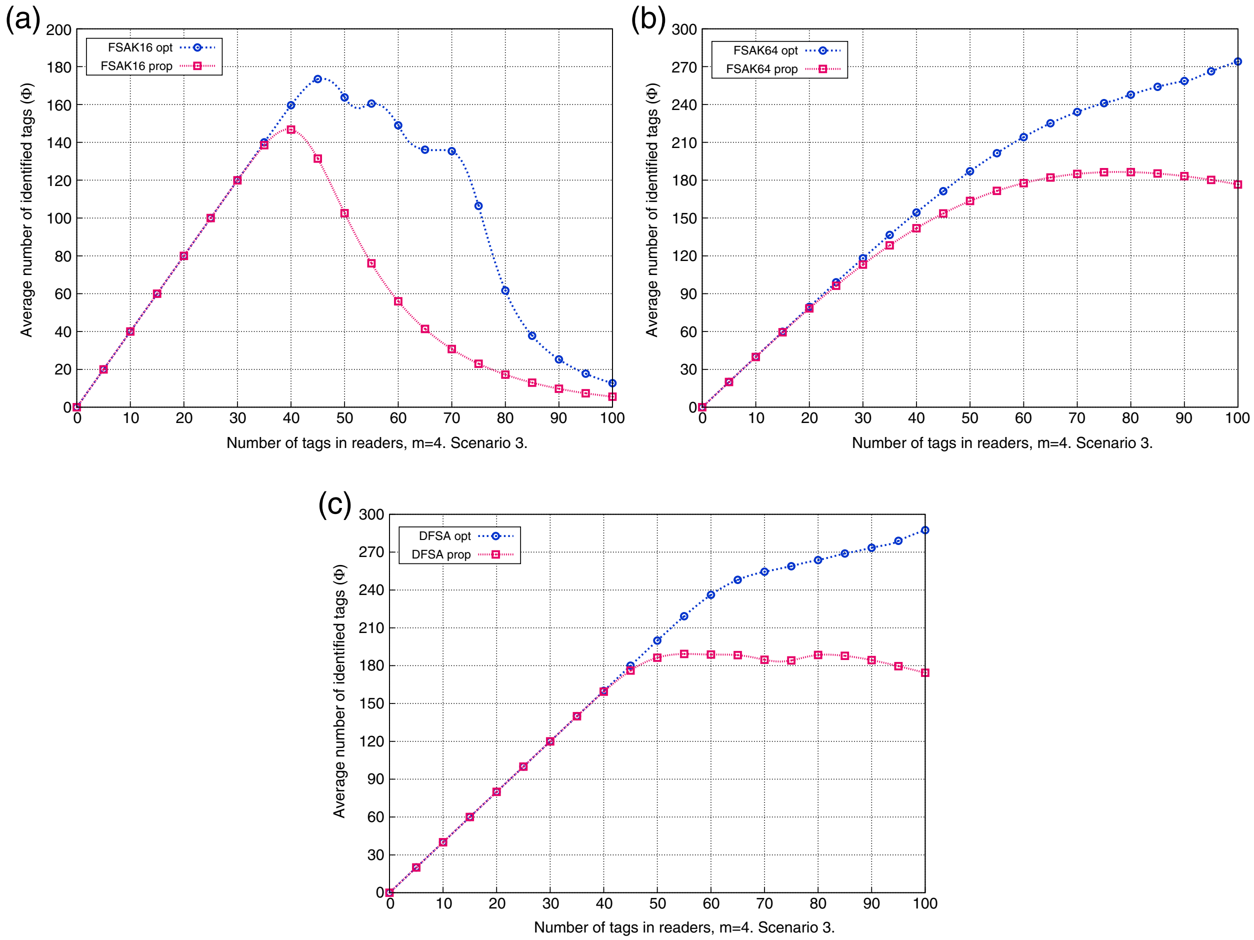

Figure 8 Optimal vs. proportional allocation for line topology, $m = 4$. (**a**) Static-FSA $k = 16$. (**b**) Static-FSA $k = 64$. (**c**) Dynamic-FSA.

Although our reference model does not use information about the instant population, it is rational to assume at least a coarse knowledge of the environment, typically the average number of competing tags. This allows the designer to configure the system for a standard case. Note that if this information is unavailable, the designer should guess somehow a configuration, and the performance would be lower than in the reference model.

Henceforth, let us assume that our reference model is based on the availability of information about the average tag population and that the designer is able to select the optimal scheduler configuration for this case (e.g., by solving problem (1)).

For simplicity, let us denote by $\vec{n}$ the m-dimensional vector $(n_1, \ldots, n_m)$, and by $\Phi_{\vec{n}}(\vec{n'})$ the expected number of identifications when the optimal solution to problem (1) with tag estimation parameter $\vec{n}$ is applied to the actual population $\vec{n'}$. Besides, let $\vec{n^*}$ denote the m-dimensional vector, where the jth component is the average number of tags in reader j.

Thus, if the probability distribution of tag population, i.e., $P(\vec{n})$ is known, the improvement (Δ) achieved by our scheduler over the reference scheduler is

$$\Delta = \sum_{\forall \vec{n}} (\Phi_{\vec{n}}(\vec{n}) - \Phi_{\vec{n^*}}(\vec{n}))P(\vec{n}). \tag{11}$$

By solving problem (1), both optimal slot assignments can be computed. Let us denote $\widehat{t}$ and $\widehat{t^*}$ to the optimal assignments for tag populations $\vec{n}$ and $\vec{n^*}$, respectively, and $\widehat{t_j}$ and $\widehat{t^*}_j$ the slots assigned to particular reader jth.

Then, Δ can be rewritten as

$$\Delta = \sum_{\forall \vec{n}} \left(\sum_{j=1}^{m} \varphi(n_{\mathrm{j}}, \widehat{t_{\mathrm{j}}}) - \sum_{j=1}^{m} \varphi(n_{\mathrm{j}}, \widehat{t^*}_{\mathrm{j}}) \right) P(\vec{n}), \tag{12}$$

where $\varphi(n, t)$ is computed directly with formulas (9) and (10) for static-FSA and dynamic-FSA, respectively.

In addition, it could be argued that instant tag population estimation may be subject to errors. This can be included in our computations through an error vector $\vec{\epsilon}$, where the jth component is $\epsilon_{\mathrm{j}} = \breve{n}_{\mathrm{j}} - n_j$, n_{j} the estimation, and $\breve{n}_j$ is the actual number of tags. Therefore, the real tag distribution is $\vec{n} + \vec{\epsilon}$, and Δ should be modified as

$$\Delta = E_{\vec{\epsilon}} \left\{ \sum_{\forall \vec{n}} \left(\sum_{j=1}^{m} \varphi(n_j y + \epsilon_{\mathrm{j}}, \widehat{t_{\mathrm{j}}}) - \sum_{j=1}^{m} \varphi(n_{\mathrm{j}} + \epsilon_{\mathrm{j}}, \widehat{t^*}_{\mathrm{j}}) \right) P(\vec{n}) \right\}. \tag{13}$$

Table 5 Full-mesh scenario

Number of readers	Dynamic-FSA	Static-FSA,	Static-FSA,
		***k* = 64**	***k* = 16**
2	0.0273	0	0.0889
4	0.0120	0.0008	0.2891
6	0.0423	0.0050	0.4831
8	0.0791	0.0086	0.6024
10	0.0947	0.0086	0.6731

Ratio of improvement using tag estimation. $\boldsymbol{\epsilon = 0}$.

Note that in this last case, $\widehat{t_{\mathrm{j}}}$, is still computed with the estimation vector $\vec{n}$. So, if $\vec{n}$ and $\vec{\epsilon}$ are independent, we finally reach to

$$\Delta = \sum_{\forall \vec{\epsilon}} \left[\sum_{\forall \vec{n}} \left(\sum_{j=1}^{m} \varphi(n_{\mathrm{j}} + \epsilon_{\mathrm{j}}, \widehat{t_{\mathrm{j}}}) - \sum_{j=1}^{m} \varphi(n_{\mathrm{j}} + \epsilon_{\mathrm{j}}, \widehat{t^*}_{\mathrm{j}}) \right) P(\vec{n}) \right] P(\vec{\epsilon}). \tag{14}$$

Note that we use a perfect knowledge of the average number of tags; therefore, we are assuming the least-favorable comparison case for our scheduler versus the reference model.

5.1 Numerical examples

For the sake of example, let us assume that for each reader, the number of tags is given by a uniformly distributed random variable $\mathbf{n}$ in the range $[0, 100]$. That is, $n_{\mathrm{j}} = \mathbf{n}$ for all $j = 1, \ldots, m$. Hence, $\vec{n^*} = (50, \ldots, 50)$. Tables 5, 6, and 7 show the average performance improvement achieved for the examples described in the previous section and for the dynamic-FSA and static-FSA tag-to-tag anti-collision protocols. The results are shown as the ratio of improvement (Δ) to the expected readings without using tag estimation.

The results clearly depend on the scenario and on the tag-to-tag anti-collision protocols. Improvement ranges from nearly 0% in many static-FSA $k = 64$ cases, while it may reach up to 67% for static-FSA, $k = 16$ in the full-mesh scenario. For dynamic-FSA, the improvement

Table 6 Star scenario

Number of readers	Dynamic-FSA	Static-FSA,	Static-FSA,
		***k* = 64**	***k* = 16**
2	0.0273	0	0.0889
4	0.1592	0.1702	0.2636
6	0.0509	0.0023	0.1940
8	0.0417	0.0005	0.1756
10	0.0333	0.0000	0.1683

Ratio of improvement using tag estimation. $\boldsymbol{\epsilon = 0}$.

Table 7 Line scenario

Number of readers	Dynamic-FSA	Static-FSA, $k = 64$	Static-FSA, $k = 16$
2	0.0266	0	0.0817
4	0.0331	0.0144	0.2114
6	0.0049	0.0018	0.2105
8	0.0298	0.0050	0.2073
10	0.0101	0.0013	0.2056

Ratio of improvement using tag estimation. $\epsilon = 0$.

Table 9 Star scenario

Number of readers	Dynamic-FSA	Static-FSA, $k = 64$	Static-FSA, $k = 16$
2	0.0275	0	0.0704
4	0.1541	0.1750	0.2087
6	0.0543	0.0017	0.2123
8	0.0406	-0.0002	0.1878
10	0.0302	0.0000	0.1925

Ratio of improvement using tag estimation. $\epsilon \sim U[-10, 10]$.

is between 2.7% and 21.14%, depending on the particular scenario. Let us remark again that this comparison is performed against the average tags identified when the optimal configuration is computed using as information the mean number of competing tags. Therefore, this is the *minimum* improvement ratio: non-optimal schedulers (as the reference heuristic used in Section 4) will obtain worse results.

Besides, we can consider the estimation error. In our test we have assumed for each reader an error distributed uniformly $\epsilon \sim U[-10, 10]$. Results are shown in Tables 8, 9, and 10.

Again, the results heavily depend on the configuration, but in most cases, even assuming an error in the tag number estimation, they show a positive feedback using the estimation. In some cases, in the full-mesh and line scenarios, there is a negative impact, but almost negligible. Therefore, we can conclude that even assuming errors, the utilization of tag estimators is worth to be considered.

6 Conclusions

This work introduced a novel optimal scheduler for a particular dense reader environment composed by m readers which must share a single frequency channel. The scheduler proposed exceeds in performance to heuristic algorithms, improving the average number of tags identified in an RFID facility. Besides, the effect of the reading protocols has also been studied in depth, concluding that a dynamic FSA algorithm excels static frame length ones. Indeed, the impact of using knowledge about tag population in the scheduler has been analyzed. It has been concluded that even assuming errors in the estimation, our scheduler is able to obtain a higher performance than a reference model, where the average population is perfectly known.

Table 8 Full-mesh scenario

Number of readers	Dynamic-FSA	Static-FSA, $k = 64$	Static-FSA, $k = 16$
2	0.0284	0	0.0527
4	0.0064	0.0007	0.2334
6	0.0361	0.0040	0.4107
8	0.0580	0.0101	0.5515
10	0.0747	0.0070	0.5712

Ratio of improvement using tag estimation. $\epsilon \sim U[-10, 10]$.

As future works, we aim at extending our model to multi-channel scenarios, developing a model that allow full resource reutilization and further analyze RFID realistic scenarios to propose optimal configuration strategies.

Appendix

Computation of $P(a|n, t)$

To compute the probability $P(a|n, t)$, we apply the technique in [17], where the authors formulate probabilistic transforms for urn models that convert the dependent random variables describing urn occupancies (slot occupancies in our case) into independent random variables. Due to the independence of random variables in the transform domain, it is simpler to compute the statistics of interest, and afterwards the transform is inverted to get the desired result.

Let us denote $P(a|n, t)$ as the probability of interest and $\boldsymbol{P}(\lambda, t, i)$ its transformation, with λ as parameter meaningful in the transform domain only. Indeed, there is no dependence on the number of balls (tags), n, in the transform domain.

The procedure is as follows: first, the appropriate transform for a particular urn model is selected. In our case,

Table 10 Line scenario

Number of readers	Dynamic-FSA	Static-FSA, $k = 64$	Static-FSA, $k = 16$
2	0.0442	-0.0006	0.0023
4	0.0324	0.0212	0.0991
6	0.0003	0.0019	0.1169
8	0.0279	0.0071	0.0815
10	0.0118	0.0012	0.0942

Ratio of improvement using tag estimation. $\epsilon \sim U[-10, 10]$.

both the t urns (slots) and the n balls (tags) are distinguishable. In this case, the independent random variables $Z_1, \ldots, Z_t$ describing the occupancy of an urn in the transform domain are Poisson distributed with mean λ [17]. That is, $P(Z_i = j) = e^{-\lambda}\frac{\lambda^{(j)}}{j!}$. Second, the probability of interest, $\boldsymbol{P}(\lambda, t)$, is computed in the transformed domain. In our case, given a frame of length t and taking into account the independence of Z_i, the probability of having i urns (slots) with one ball (tag) is

$$\begin{aligned}\boldsymbol{P}(\lambda, t) &= \binom{t}{i} P(Z=1)^i (1 - P(Z=1))^{t-i} \\ &= \binom{t}{i} (e^{-\lambda}\lambda)^i (1 - e^{-\lambda}\lambda)^{t-i}.\end{aligned} \tag{15}$$

Finally, the inverse transform is computed as

$$P(a|n,t) = \frac{n!}{t^n}[\,\lambda^n]\,\{e^{\lambda t}\boldsymbol{P}(\lambda, t)\}, \tag{16}$$

with $[\,\lambda^n]\,\{h(\lambda)\}$ denoting the coefficient of λ^n in the power series $\{h(\lambda)\}$. So, we have to rewrite Equation (15) as a power series in λ and extract the appropriate coefficient. We use first the binomial expansion $(a+b)^c = \sum_{k=0}^{h}\binom{h}{k}a^k b^{h-k}$:

$$P(a|n,t) = \frac{n!}{t^n}[\,\lambda^n]\left\{\binom{t}{i}\sum_{c=0}^{t-i}\binom{t-i}{c}(-1)^c e^{\lambda(t-i-c)}\lambda^{c+i}\right\}, \tag{17}$$

and using the expansion of the exponential function as a power series, the sum in Equation (17) can be rewritten as

$$\begin{aligned}&\sum_{c=0}^{t-i}\binom{t-i}{c}(-1)^c\sum_{j=0}^{\infty}\frac{(t-i-c)^j}{j!}\lambda^j\lambda^{c+i} = \\ &= \sum_{j=0}^{\infty}\lambda^{j+i}\left(\sum_{c=0}^{j}(-1)^c\binom{t-i}{c}\frac{(t-i-c)^{j-c}}{(j-c)!}\right),\end{aligned} \tag{18}$$

and extracting the coefficient of λ^n for the appropriate n value, $n = j + i$, we obtain the result in Equation (8).

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

This work has been supported by project CALM TEC2010-21405-C02 which is funded by the Spanish Ministerio de Innovación y Ciencia. It has been 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). We are also indebted to Javier Fernández-Nogueira for his help in MINLP optimization.

Received: 29 November 2012 Accepted: 22 May 2013
Published: 17 June 2013

References

1. J Vales-Alonso, MV Bueno-Delgado, JJ Alcaraz, in *Paper presented in 2011 IEEE international conference on RFID-technologies and applications (RFID-TA)*. Optimal scheduling in dual reader RFID environments (Sitges Spain, 15–16 September 2011)
2. J Vales-Alonso, FJ Parrado-Garcia, JJ Alcaraz, E Egea-Lopez, in *Paper presented in 2012 fourth international EURASIP workshop on RFID technology (EURASIP RFID)*. Optimal scheduling in single channel dense reader RFID environments (Torino, Italy, 27–28 September 2012)
3. MV Bueno-Delgado, J Vales-Alonso, C Angerer, M Rupp, in *Paper presented in IEEE international conference on industrial technology*. A comparative study of RFID schedulers in dense reader environments (Viña del Mar, Chile, 14–17 March 2010)
4. J Choi, C Lee, An MILP-based cross-layer optimization for a multi-reader arbitration in the UHF RFID system. Sensors **11**(3), 2347–2368 (2011)
5. J Kim, W Lee, E Kim, D Kim, K Suh, Optimized transmission power control of interrogators for collision arbitration in UHF RFID systems. IEEE Commun. Lett. **11**(1), 22–24 (2007)
6. C Chui-Yu, K Cheng-Hsin, KY Chen, in *Paper presented in IEEE international conference on systems, man and cybernetics*, Optimal RFID networks scheduling using genetic algorithm and swarm intelligence (San Antonio, USA, 11–14 October 2009)
7. V Deolalikar, J Recker, M Mesarina, S Pradhan, *Optimal scheduling for networks of RFID readers Paper presented at the first international workshop on RFID and ubiquitous sensor networks*. (EUC Workshops LNCS 3823), Nagasaki,Japan, 6–9 December 2005)
8. AH Mohsenian-Rad, V Shah-Mansouri, VWS Wong, R Schober, Distributed channel selection and randomized interrogation algorithms for large-scale and dense RFID systems. IEEE Trans. on Wireless Commun. **9**(4), 1402–1413 (2010)
9. Y Tanaka, I Sasase, Interference avoidance algorithms for passive RFID systems using contention-based transmit abortion. IEICE Trans. Commun. **E90-B**(11), 3170–3180 (2007)
10. H Seo, C Lee, in *Paper presented in 2010 IEEE international conference on communications (ICC)*. A New GA-Based Resource Allocation Scheme for a Reader-to-Reader Interference Problem in RFID Systems. (Cape Town, South Africa, 23–27 May 2010)
11. International Organization for Standardization, *ISO/IEC 18000-6C:2004: information technology – radio frequency identification for item management-Part 6: parameters for air interface communications at 860 MHz to 960 MHz*. (International Organization for Standardization, Geneva, Switzerland, 2010)
12. International Organization for Standardization, *ISO/IEC 18000-7:2008: information technology – radio frequency identification for item management – Part 7: parameters for active air interface communications at 433 MHz*. (International Organization for Standardization, Geneva, Switzerland, 2010)
13. EPC Global Inc., *EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz–960 MHz Version 1.2.0*. (EPC Global Inc., Brussels Belgium, 2008)
14. N Abramson, in *Proc. National Computer Conference*. Packet switching with satellites (ACM Press, New York, 1973), pp. 695–702
15. MV Bueno-Delgado, J Vales-Alonso, FJ González-Castaño, in *Paper presented in proceedings of the 35th international conference of the IEEE Industrial Electronics Society*. Analysis of DFSA anti-collision protocols in passive RFID environments (Porto,Portugal, 3–5November 2009)
16. J Vales-Alonso, MV Bueno-Delgado, E Egea-López, J Alcaraz, FJ González-Castaño, Multi frame maximum likelihood tag estimation for RFID anti-collision protocols. IEEE Trans. on Industrial Informatics. **7**(3), 487–496 (2011)
17. O Milenkovic, KJ Compton, Probabilistic transforms for combinatorial urn models. Comb. Probab.Comput. **13**(4-5), 645–675 (2004)

doi:10.1186/1687-3963-2013-11
Cite this article as: Vales-Alonso *et al.*: **Performance analysis of optimal schedulers in single channel dense radio frequency identification environments.** *EURASIP Journal on Embedded Systems* 2013 **2013**:11.