Energy Efficient Tag Estimation Method for ALOHA-Based RFID Systems

Petar Šolić, Joško Radić, and Nikola Rožić

***Abstract*— Radio frequency identification (RFID) technology has become an important tool for items identification and tracking. Its rapid deployment in dynamic industry areas enables variety of new applications in behavioral analysis and processes automation. In such systems it is crucial to identify all RFID tags before they leave the interrogation area. To accomplish fast identification, it is necessary to estimate a number of tags which are in the reader interrogation area. In this paper, we revisit the problem of tag quantity estimate and adapting frame size of dynamic frame slotted ALOHA widely used as RFID medium access control mechanism. The main disadvantage in the implementation of the state-of-the-art estimation algorithms includes the number of required computations, along with the temporary storage of large numbers which appear before estimation. As a consequence, such algorithms are energy inefficient and may require specific computer architecture to support the calculus. In order to address stated issues, we present improved linearized combinatorial model algorithm for optimal frame size selection which, due to linear property of the estimator, allows significant reduction in the estimate computation with acceptable tradeoff in accuracy. Simulations analyze the required number of slots to identify tag population and the floating point operation costs required to compute estimate. In addition, to emphasize the importance of reducing computational cost, we give a case study which compares energy consumed by the mobile RFID reader processor to compute the estimate and its energy equivalent of the required radio front-end energy to identify tags.**

***Index Terms*— Dynamic frame slotted ALOHA (DFSA), tag estimate method, optimal frame size selection, energy efficient methods.**

I. INTRODUCTION

RFID technologies based on the wireless communication between reader and tag present the great innovation in items identification and tracking. Such technology provides the edge infrastructure for the Internet of Things (IoT) final deployment. Regarding tags battery presence or absence, RFID can be divided into battery-powered active, battery-assisted passive (BAP), and battery free passive RFID technology [1].

Active RFID enables tag reading range up to 100 m, but due to tag robustness, price of about 100 USD, and limited battery lifetime reduces its spectra of usage. To reduce tag size and price, but to remain with greater reading distances of up to 40 m, one can consider BAP RFID usage. Battery in BAP tags is used to power tag circuitry, while tags communication with the reader is same as in the battery free passive RFID. Passive tags communicate and power themselves using same RF waves radiated by the reader antenna. An overview of pros and cons for different RFID systems can be found in [2].

With a size of about 15 cm in width and 1 cm in height, at price of about 0.1 USD per tag and reading range of up to 10 m, passive Ultra High Frequency (UHF) RFID has become the most popular technology in terms of price/performance ratio. One of its typical application includes a portal equipped with RFID reading antennas and passing pallet of products with passive RFID tags to be identified as soon as possible. Moreover, its rapidly introduction to the industry fields applications such as supply chains [3], localization techniques [4], [5], or indoor robot navigation [6], require all tags identification as fast as possible (e.g. increase tag reading rate) before they leave the interrogation area [7].

For the sake of reducing equipment manufacturing costs and making reader-tag communications worldwide available, EPCglobal [8] defines the set of standards that enables passive UHF RFID infrastructure. Inside the set of standards, Gen2 [9] specifies physical and Medium Access Control (MAC) settings to support reader-tag communication. Since passive tags do not own a battery, it is necessary that reader delivers enough power to energize tags and enable them to respond with the required information. Energy levels that tags can harvest are small, and thus they cannot afford themselves energy inefficient Medium Access Control Schemes (MAC).

Generally, MAC in RFID is random based, and there are two popular approaches for its realization: binary-tree and ALOHA-based protocols [1]. In binary tree [10], [11], reader can through consecutive YES/NO interrogation identify targeted tags, while ALOHA based algorithm initiates tag's data transmission in the time when reader requests. Among ALOHA-based protocols, the most often used due to its maximal efficiency is Dynamic Framed Slotted ALOHA (DFSA), where Gen2 specifies its concrete application. To maximize its throughput and thus identify all tags as fast as possible, it is necessary to correctly estimate the number of interrogating tags and set size of the next frame accordingly.

In this paper we describe DFSA implementation in RFID systems, and the state of art methods based analysis in tag estimate along with algorithms for their efficient implementation. Such algorithms usually include the number of calculations prior tag estimate, and due to necessity for

Manuscript received April 11, 2014; revised June 3, 2014; accepted June 5, 2014. Date of publication June 12, 2014; date of current version August 29, 2014. The associate editor coordinating the review of this paper and approving it for publication was Prof. Kiseon Kim.

The authors are with the University of Split, Split 21000, Croatia (e-mail: psolic@fesb.hr; radic@fesb.hr; rozic@fesb.hr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSEN.2014.2330418

factorial operations, temporary storage of large numbers. As a consequence, estimators are energy inefficient and time cost for its calculations may cause delay in tags identification. Significant reduction in the required computations is crucial for the RFID mobile readers in the terms of energy saving. To overcome disadvantages of state of the art methods, we provide Improved Linearized Combinatorial Model (ILCM), which uses only a modest calculation operations, and can be easily implemented and applied as tag estimate method. Linearization of such complex counting formulas is possible due to linear dependency of the tag estimate and the number of successful slots per frame of a fixed size, for the fixed number of collisions. Results on the number of slots required to identify tags and Floating Point Operations (FLOP) costs required to compute the estimate are provided. The impact on required FLOP cost is further discussed in terms of required energy to compute the estimate. To show its importance we provide the timing details of Gen2 protocol, which allows us to compute the total time for tags identification. Using given time, and the reader output power we are able to compare required radio front-end energy to identify tags with required energy for the estimate computation on designated hardware.

This paper is structured as follows: in the next Section we describe DFSA access control along with problem definition, state of art tag estimate methods and analysis on DFSA efficient implementation. In Section III. we present our tag estimate method called Improved Linearized Combinatorial Model (ILCM). Section IV. provides simulation results analysis, and in Section V. we provide some concluding remarks and directions for the future work.

II. DFSA Random Access in RFID and Related Works

In DFSA, reader-tag communication is divided into the frames that are further divided in the time slots. In real RFID DFSA implementation [9], reader announces the size of frame by broadcasting parameter Q, limiting frames to the sizes of $L = 2^Q$. When tags receive Q, they set their slot counters to the random value between 0 and 2^Q-1. Tag(s) with the slot counter set to zero respond back to the reader. As the next step, reader issues command to decrement tags slot counters by 1. Again, all tags with the slot counter set to zero respond back to the reader. Number of the slot counter decrease commands being broadcast by the reader is 2^Q-1, thus interrogating all slots of the frame. Regarding time slot occupancy, there are three possible scenarios:

- no response within the slot (empty slot),
- single response within the slot (successful slot),
- multiple response within the slot (collision slot).

Figure 1 shows an example of the interrogation frame.

DFSA system throughput is given by [12]:

$$U(n, p) = np(1-p)^{(n-1)} \tag{1}$$

where $p = 1/L$ is the probability of finding a tag within the slot of the frame, where frame size is $L = 2^Q$ and n represents total number of tags being interrogated. Maximum throughput follows from (1), when its first derivative equals to zero, resulting in:

$$\frac{dU(n,p)}{dp} = n(1-p)^{(n-2)}((1-p)p(n-1)) = 0 \tag{2}$$

Maximum is obtained for $p = 1/n$, i.e. when the number of tags equals frame size ($n = L$). Such case yields maximum throughput of DFSA, given as $1/e = 0.368$. Therefore, to maximize the throughput, it is necessary to estimate the number of tags ($\hat{n}$) and then set the frame size to round $(\log_2(\hat{n}))$. Throughput $U(n, p)$ for different frames is depicted in Figure 2, while Q-bins for achieving maximum throughput are given in Table I.

In real scenarios, reader can cancel the interrogation of given slot if there is no response within, or send another command when collision occurs in order to require the lost information from collided tags. Such scenarios gives different duration of different timeslots, which means that (1) should be modified accordingly. With aim to simplify frame adaptation model without of loosing generality, in our observations we

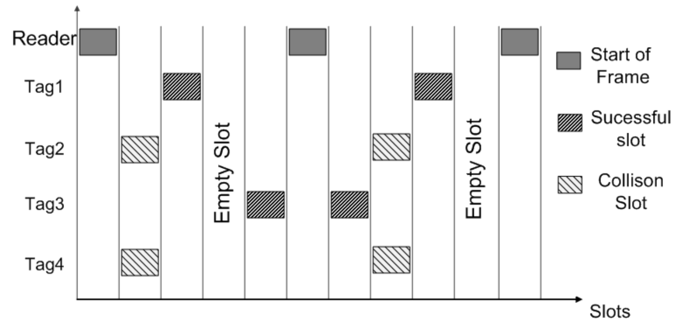


Fig. 1. Example of RFID reader-tag interrogation round with time frame sizes of 4, and with 4 tags in the interrogation area. Both frames ends up with one collision, one empty and two successful slots.

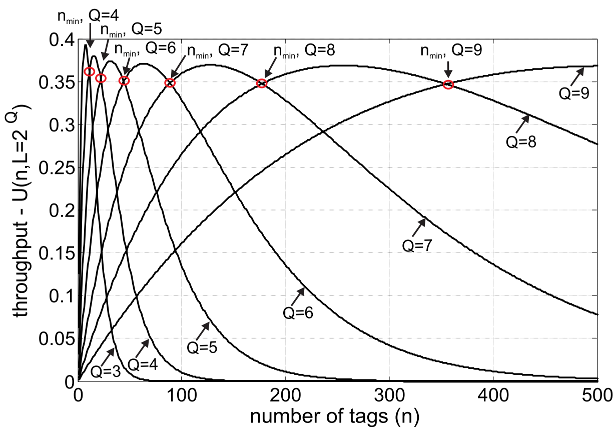


Fig. 2. DFSA throughput and minimum number of tags for appropriate frame sizes $L = 8$ to 512.

TABLE I

Minimum and Maximum Number of Tags for Q-Bins

Q	0	1	2	3	4	5	6	7	8	9
n_{min}	0	2	4	7	12	23	45	90	178	356
n_{max}	1	3	6	11	22	44	89	177	355	710

consider choosing frame size equal to the tag number ($L = n$), while the impact on different timeslot durations is presented in simulation results section.

Many published works address the issue of the accurate tag number estimate based on the number of Empty (E), Successful (S) and Collision (C) slots collected from the previous frame. Further, it is important to note that some of them are not implementable due to errors in the formulation. For example, the algorithm in [13] searches for log(0), or [14] may return negative probabilities, both noticed in [15]. To compare with the state of the art, we look for the implementable algorithms along with pseudo-codes of their efficient implementation, required to evaluate the number of Floating Point Operations (FLOP) to compute the estimate. FLOP costs are important due to its direct relationship with the required energy to compute estimate.

A. Tag Estimate Methods

Vogt proposal [16] for the tag estimate models the slot occupancy by binomial distribution, where the probability of finding r tags in one slot is given by:

$$B_{n,1/L}(r) = \binom{n}{r}\left(\frac{1}{L}\right)^r\left(1-\frac{1}{L}\right)^{n-r} \tag{3}$$

where L is the frame size, and n is the number of tags. According to (3), empty slot probability is $p_e = B_{n,1/L}(0)$, successful slot is given by $p_s = B_{n,1/L}(1)$, and probability of collision slot is $p_c = B_{n,1/L}(\geq 2) = 1 - p_e - p_s$. Using p_e, p_s, p_c, the expected values of number of empty slots $a_0 = \mathcal{E}(B_{n,1/L}(0)) = LB_{n,1/L}(0)$, successful slots $a_1 = \mathcal{E}(B_{n,1/L}(1)) = LB_{n,1/L}(1)$ and collision slots $a_{\geq 2} = \mathcal{E}(B_{n,1/L}(\geq 2)) = LB_{n,1/L}(\geq 2)$ can be calculated. Note that $\mathcal{E}$ stands for the expectation operator. If the probabilistic observation is correct, then the expected values should be near its realization. Hence, Vogt used it to calculate Mean Square Error (MSE) tag estimate:

$$\epsilon_{vd} = \min_n \left| \begin{pmatrix} a_0 \\ a_1 \\ a_{\geq 2} \end{pmatrix} - \begin{pmatrix} E \\ S \\ C \end{pmatrix} \right| \tag{4}$$

Equation (4) was implemented in Phillips I-code RFID systems [17], as an anti-collision scheme. To implement such estimate, one has to go into distribution and calculate MSE until it reaches minimum. Since, the number of tags is at least $n_{min} = S + 2C$, calculation can be started from n_{min}. An efficient implementation of Vogt tag estimate is shown in Algorithm 1.

However, Vogt's assumption that tags are identically distributed in slots is not generally correct. For larger tag sets, such assumption yields to an error in tag estimate.

Another approach provided by Chen [18], assumes that tags in the frame are multinomial distributed:

$$P(E, S, C) = \frac{L!}{E!S!C!} p_e^E p_s^S p_c^C \tag{5}$$

To calculate p_e, p_s and p_c Chen used the same Vogt's binomial model (3). Once the frame is realized, i.e. the E, S and C are

Algorithm 1 Vogt's Estimate Efficient Implementation

Require: E, S, C.
Ensure: $\hat{n}_{vogt}$.

$n = S + 2C$
$dist_n = -1$
$dist_p = 0$
while $dist_n < dist_p$ **do**
  $p_e = (1 - (1/L))^n$
  $p_s = (n/L)(1 - (1/L))^{(n-1)}$
  $p_c = 1 - p_e - p_s$
  $e = Lp_e; s = Lp_s; c = Lp_c$
  $dist_p = dist_n$
  $dist_n = (e - E)^2 + (s - S)^2 + (c - C)^2$
  if $n = S + 2C$ **then**
    $dist_p = dist_n + 1$
  end if
  $n = n + 1$
end while
$\hat{n}_{vogt} = n - 1$

obtained, *a posteriori* distribution is given by:

$$P(n|E,S,C) = \frac{L!}{E!S!C!} \times \left[\left(1-\frac{1}{L}\right)^n\right]^E \times \left[\frac{n}{L}\left(1-\frac{1}{L}\right)^{(n-1)}\right]^S \times \left[1-\left(1-\frac{1}{L}\right)^n - \frac{n}{L}\left(1-\frac{1}{L}\right)^{(n-1)}\right]^C \tag{6}$$

When $P(n|E, S, C)$ distribution is calculated, Chen suggests finding maximum of the given probability distribution, and setting the tag estimate equal to a maximum, i.e. $\hat{n} = \arg\max_n P(n|E, S, C)$. An efficient implementation of the Chen's estimate is described with Algorithm 2.

Algorithm 2 Chen's Estimate Efficient Implementation

Require: E, S, C.
Ensure: $\hat{n}_{chen}$.

$L = E + S + C$
$n = S + 2C$
$next = 0$
$previous = -1$
while $previous < next$ **do**
  $p_e = (1 - (1/L))^n$
  $p_s = (n/L)(1 - (1/L))^{(n-1)}$
  $p_c = 1 - p_e - p_s$
  $previous_n = next_n$
  $next = (L!/(E!S!C!))p_e^E p_s^S p_c^C$
  $n = n + 1$
end while
$\hat{n}_{chen} = n - 2$

Chen in his paper did the mistake in the modelling of the problem [19], where he considered mutual independence of the number of E, S and C slots.

Improved Chen version which includes mutual dependence of different slot types is provided in closed form formulation

[19] given by:

$$P(E,S,C) = \frac{L!}{E!S!C!} P_1(E) P_2(S|E) P_3(C|E,S) \tag{7}$$

where $P_1(E)$, $P_2(S|E)$, and $P_3(C|E,S)$ are given with

$$\begin{aligned} P_1(E) &= \left(1 - \frac{E}{L}\right)^n \\ P_2(S|E) &= \binom{n}{S} \frac{(L-E-S)^{n-S}}{(L-E)^n} S! \\ P_3(C|E,S) &= \sum_{k=0}^{C} \sum_{\upsilon=0}^{C-k} (-1)^{(k+\upsilon)} \binom{C}{k} \binom{C-k}{\upsilon} \\ &\quad \times \frac{(n-S)!}{(n-S-k)!} \frac{(C-k-\upsilon)^{(n-S-k)}}{C^{(n-S)}} \end{aligned} \tag{8}$$

An efficient algorithm for implementation of (8) is described with Algorithm 3.

Algorithm 3 Vahedi's Estimate Efficient Implementation

Require: E, S, C.
Ensure: $\hat{n}_{vahedi}$.

$L = E + S + C$
$n = 2C + S$
$next = 0$
$previous = -1$
while $previous < next$ **do**
  $P_1 = (1 - (E/L))^n$
  $P_2 = (n!/(S!(n-S)!)((L-E-S)^{n-S}/(L-E)^n)S!$
  $P_3 = 0$
  for $k = 0 \to C$ **do**
    for $\upsilon = 0 \to C - k$ **do**
      $P_3 = P_3 + (-1)^{k+\upsilon}(C!/(k!(C-k)!)((C-k)!/(\upsilon!(C-k-\upsilon)!)((n-S)!/(n-S-k)!)((C-k-\upsilon)^{n-S-k}/C^{n-S})$
    end for
  end for
  $previous_n = next_n$
  $next = (L!/(E!S!C!))P_1 P_2 P_3$
  $n = n + 1$
end while
$\hat{n}_{vahedi} = n - 2$

As an alternative to [19] (yielding same estimate) by using the exponential generating functions counting technique is given by Floerkemeier's frame by frame estimation [12], [20]. Probability of finding n tags in the realized frame is:

$$P(E,S,C|n) = \frac{L!}{E!S!C!} \frac{T(E,S,C,n)}{L^n} \tag{9}$$

where $T(E,S,C,n)$ is given with the $x^n/n!$ coefficient within the expansion of exponential generating function:

$$\begin{aligned} G(x) &= \left(\frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots\right)^C x^s \\ G(x) &= \left(e^x - (1+x)\right)^c x^s \end{aligned} \tag{10}$$

As it can be concluded, all presented algorithms involves entering into probability distributions, and estimate through finding their minimums or maximums. In order to avoid such complex calculations, Gen2 standard suggests usage of simple frame adaptation scheme called Q-Selection algorithm, which performance we analyse in the next subsection.

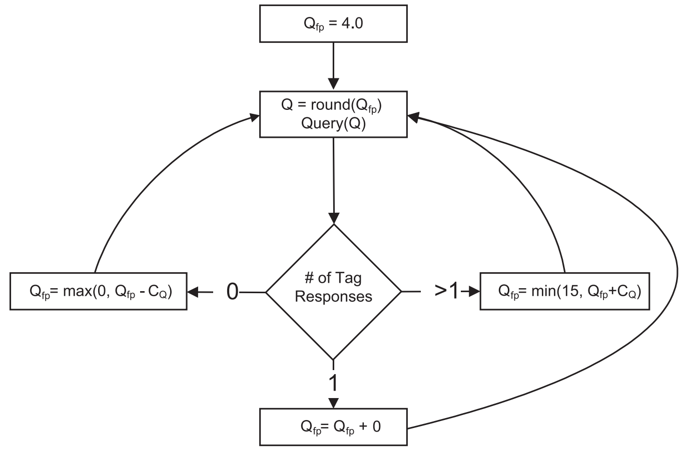


Fig. 3. Q-Selection algorithm suggested in [9], where $0.1 \le C_Q \le 0.5$.

B. Q-Algorithm

The Q-algorithm used in Gen2 standard is depicted in Figure 3. As it can be seen from the flow graph, Q-algorithm for frame adaptation uses only modest math operations which adds or subtracts constant value C_Q in collision or empty slot scenario, respectively. Reader starts the interrogation round by broadcasting Q_{init}, and at the end updates new value $Q = \text{round}(Q_{\text{fp}})$, broadcast to collided tags in order to identify them. The main advantage is its simplicity in implementation, but the main disadvantage is the way how to choose constant value C_Q, which is not a part of the standard specification. Some papers have been written with aim to optimize Q-algorithm, such as [21], where C_Q can be optimized if number of tags is known, otherwise authors suggests usage two different constants, C_c which is chosen arbitrarily and added to Q_{fp} in collision scenario, and $C_e = (e-2)C_c$ added to Q_{fp} in case of empty slot. Our simulations show that the latter scheme, where tag number is unknown, puts a little influence on the tag reading rate, due to pick of C_c which should be different for different tag number (n). In [20], authors compare their developed frame adaptation method with the tuned Q-algorithm, where the constant is set to $C_Q = \left(0.8/\log_2 L\right)$. Such observation improves tag reading rate in the case of small number of tags, but for large n (over 100) the algorithm misses the frame size and causes great identification delay. Recently, authors in [22] used other approach in Q-algorithm optimization (denoted with Wu-Q), where they supposed that Q_{init} should be equal 4, $C_Q = 0.5$ for $0 \le Q \le 2$, and $C_Q = 0.1$ for $Q \ge 10$, or $C_Q = 1/Q$ otherwise. Wu-Q approach provides the best results among proposals, but for other Q_{init} the algorithm misses frame size.

The impact on the time delay (expressed by the mean number of slots required to identify all tags) for different fixed C_Q values, and Wu-Q is shown in Figure 4. Results are provided for both $Q_{\text{init}} = 4$ and $Q_{\text{init}} = 6$. As it can be seen, C_Q should be picked out depending on the number of tags, which is generally unknown. In addition, by using $Q_{\text{init}} = 6$,

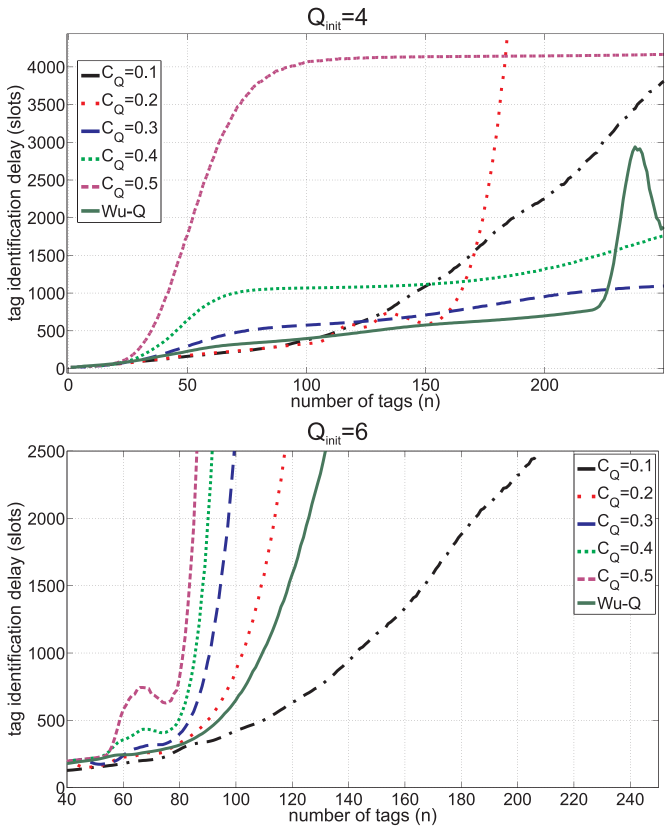


Fig. 4. Q-Selection algorithm performance for different C_Q values.

frame will be missed for any value of C_Q which causes great identification delay.

In the next section we provide a novel, and efficient tag estimate method called Improved Linearized Combinatorial Model (ILCM).

III. Improved Linerized Combinatorial Model (ILCM)

A. Preliminaries

Derivation of ILCM is based on the calculation of the combinatorial model [23], which gives the conditional probability that the frame will be realized with E empty, S successful and C collision slots, if there are n tags. This probability is given with

$$p(E, S, C \mid n) = \frac{L!}{E!S!C!} \frac{N_S(n, S) N_C(n, S, C)}{L^n} \tag{11}$$

where frame size $L = E + S + C$; $N_S(n, S) = n!/(n - S)!$ represents the number of ways to distribute n tags among S successful slots, and $N_C(n, S, C)$ stands for the number of ways to distribute remaining $n - S$ tags in C collision slots. To provide $N_C(n, S, C)$, the exponential generating function $G(x) = (e^x - (1 + x))^C$ and its Maclaurin expansion $G(x) = (x^2/2! + x^3/3! + x^4/4! + x^5/5! + \cdots)^C$ is used. Then, $N_C(n, S, C)$ is given by the coefficient of the term $x^{(n-S)}/(n - S)!$ in the expansion of $G(x)$. Multinomial coefficient $L!/E!S!C!$ denotes the number of ways to permute positions of E, S, and C slots, and the denominator L^n denotes the total number of ways to distribute n tags in L slots. The decision on the expected number of tags $\hat{n}$ is taken where $p(E, S, C \mid n)$ is maximum, i.e.

$$\hat{n} = \arg\max_n \{p(E, S, C \mid n)\} \tag{12}$$

To better understand using of (11), let us consider an example when $n = 8$ tags distribute in the frame $L = 4$, and the reader collects the information of zero empty, two successful, and two collision slots, i.e. $E = 0, S = 2, C = 2$. Then, $N_S(8, 2) = 56$, and $N_C(8, 2, 2)$ is given by the coefficient of the term $x^6/6!$ in Maclaurin expansion of the exponential generating function $(e^x - 1 - x)^2 = x^2 - 2xe^x + 2x + e^{2x} - 2e^x + 1$. Expansion in terms of $x^6/6!$ becomes $x^2 - 2(1/5!)6! + 2x + (2^6/6!)6! - 2(1) + 1 = x^2 + 2x + 1 + 50$, i.e. there are $N_C(8, 2, 2) = 50$ ways of distributing 6 tags in 2 collision slots. Finally, $p(E = 0, S = 2, C = 2 \mid n = 8) = 0.2563$. To find tag estimate $\hat{n}$, it is necessary to find n when $p(E = 0, S = 2, C = 2 \mid n)$ becomes maximal. Since the calculus of exponential generating functions for larger n and C is non-trivial, to calculate $p(E, S, C \mid n)$ we consider usage of the open source math tool SAGE (www.sagemath.org), due to simple handling and operations with generating functions. To simplify calculations, below we provide the tag estimate called ILCM.

B. ILCM

ILCM derivation is possible due to the property that tag estimate $\hat{n}$ and the number of successful slots S are linearly dependent for the fixed number of collision slots C. To provide $\hat{n}$ with a linear model $\hat{n} = f_{C_j}(S)$, it is necessary to calculate $p(E, S, C \mid n)$ for all possible outcomes of E, S, C, n in the frame L, and use maximums to construct lines, where index j, $0 \leq j \leq L - 1$, stands for the number of collision slots. Note that in estimation and interpolation procedures we do not consider cases that result in all collisions, i.e. $j = L$, because in that case n can be any large number. Figure 5 shows an example of exploiting linearity (with lines connecting $\hat{n}$ by changing S for fixed C_j, i.e. $\hat{n} = f_{C_j}(S)$) for the frame $L = 16$.

Since the functions $f_{C_j}(S)$ within the frame L are lines, $\hat{n}$ in L can be written as $\hat{n} = f_{C_j}(S) = kS + l$, where slope $k = f(C)$ and $\hat{n}$-intercepts $l = f(C)$ should be calculated. In addition, the issue is that different L have different k and l, and our goal is to provide model for adapting any frame size, i.e. to find functions which can interpolate k and l reliably for any L. Values of k and l in respect to L can be calculated in different ways which can yield good or less good estimates. One of its attempts is presented in LCM model [24], where k and l are described with quadratic functions ($\hat{n} = f(C, L) = a(L)C^0 + b(L)C^1 + c(L)C^2$) for frames $L = 8, 16$, and 32. To encompass multiple frames, functions $a(L)$, $b(L)$, and $c(L)$ are interpolated linearly, resulting in

$$\begin{aligned} k &= (1.0569 + 0.0115L) + (-0.172 + 0.0022L)C \\ &\quad + (0.0441 - 0.0013L)C^2 \\ l &= (-3.8 + 0.2336L) + (0.9633 - 0.0314L)C \\ &\quad + (0.2825 - 0.0053L)C^2 \\ \hat{n} &= kS + l \end{aligned} \tag{13}$$

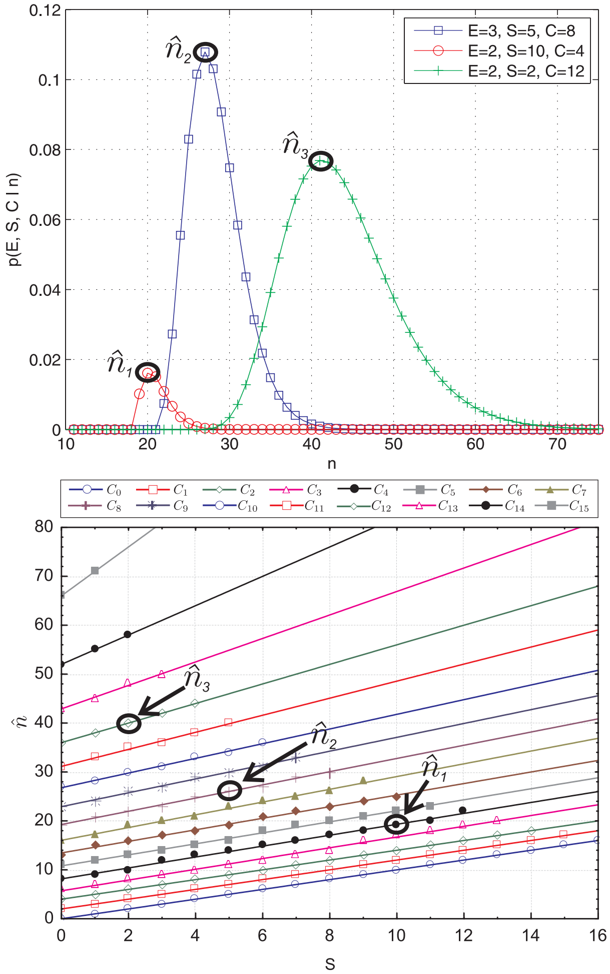


Fig. 5. Upper figure presents examples of $p(E, S, C \mid n)$ distributions for different frame realizations if $L = 16$, calculated from (11). Tag estimate is given by: $\hat{n} = \arg\max_n\{p(E, S, C \mid n)\}$. Lower figure depicts linear dependency of the tag estimate $\hat{n}$ and the number of successful slots S in for the fixed number of collision slots C in the frame $L = 16$. Examples of estimates $\hat{n}_1, \hat{n}_2$, and $\hat{n}_3$ are depicted in the both upper and lower figure.

However, LCM interpolation does not work well for smaller frame sizes, when it can yield negative results and the estimate for larger frame sizes becomes inaccurate.

In the following subsection we provide functions for the interpolation, where tag estimation can compete with state-of-art methods.

C. ILCM Design

To accomplish efficient tag estimation, we interpolated $\hat{n}$ for $L = 4, 8, 16, 32$, and 64. Using acquired lines we have interpolated the slope k with $k = g(C, L) = C/(F(L) + G(L)C) + H(L)$, while l can be reliably modelled as $l = f(C, L) = A(L)\tan(B(L)C)$. In addition we found that $A(L)$ and $F(L)$ behave linearly, i.e. $A(L) = 1.2592 + 1.513L$, $F(L) = 4.344L - 16.28$. $B(L)$ is exponential, $B(L) = 1.234L^{-0.9907}$, $G(L) = L/(-2.282 - 0.273L)$, and $H(L)$ is logarithm, $H(L) = 0.2407\ln(L + 42.56)$. Preview of ILCM interpolating functions is given in Figure 6.

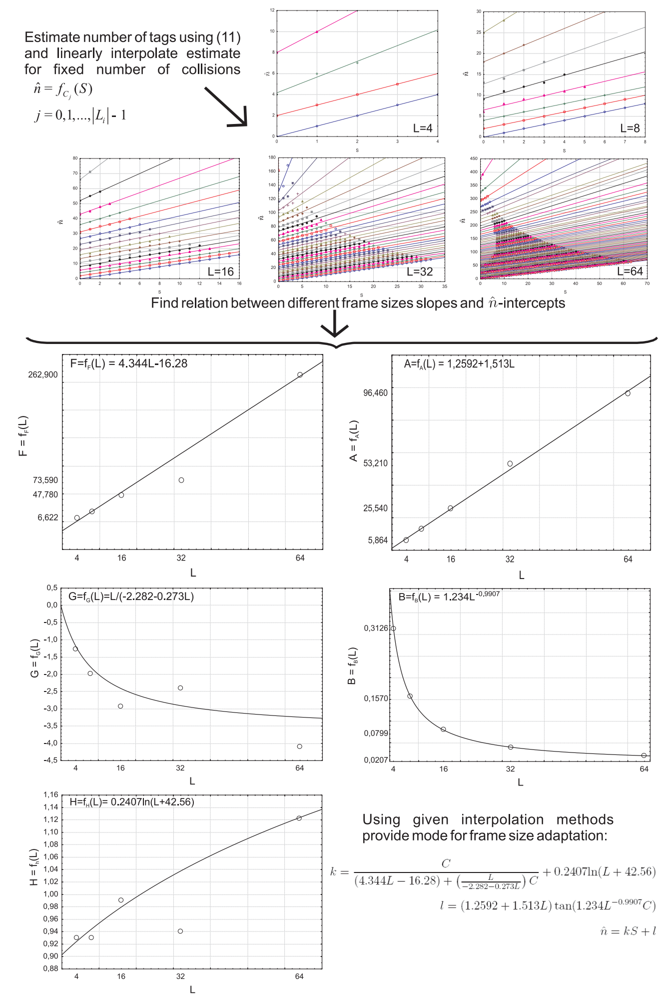


Fig. 6. Interpolation functions used for deriving improved linearized combinatorial model (ILCM).

Finally, such interpolation method yields following tag estimate method:

$$\begin{aligned} k &= \frac{C}{(4.344L - 16.28) + \left(\frac{L}{-2.282-0.273L}\right)C} \\ &\quad +0.2407\ln(L + 42.56) \\ l &= (1.2592 + 1.513L)\tan(1.234L^{-0.9907}C) \\ \hat{n} &= kS + l \end{aligned} \tag{14}$$

ILCM estimate should be bounded in two scenarios:

- because the interpolation is given from $L = 4$ slots, for smaller frame sizes k may be lesser than 0. Such scenarios should return $k = 0$,
- no collision scenarios ($C = 0$) involves an error in estimate. Such scenarios should set $\hat{n} = S$.

ILCM can be efficiently implemented using Algorithm 4.

Presented method is simple to apply and does not involve the number of calculations before providing its estimate. In the following section we provide the results obtained from DFSA tag identification simulations.

Algorithm 4 ILCM Estimate Implementation

Require: E, S, C.

Ensure: $\hat{n}_{ILCM}$.

$L = E + S + C$

$k = (1.2592 + 1.513L)\tan(1.234L^{-0.9907}C)$

$l = C/((4.344L - 16.28) + (L/(-2.282 - 0.273L)C) + 0.2407\ln(L + 42.56)$

1: **if** $k < 0$ **then**

2: $k = 0$

3: **end if**

$\hat{n}_{ILCM} = kS + l$

4: **if** $C = 0$ **then**

5: $\hat{n}_{ILCM} = S$

6: **end if**

IV. Simulation Results

Simulation results are considered for the scenario of the pallet of products with tags attached, and the pallet is not moved away from the reader until all tags get identified. The process of identification in Gen2 RFID system works through the rounds containing multiple frames within all tags get identified. After the first frame of the first round is realized, Q value is changed (according to its tag estimate, $Q = \text{round}(\log_2(\hat{n} - S))$) and broadcast to tags as an information for the size of next frame of its round. Current round is not finished until all tags get identified. Once the round is completed, reader begins another round to again identify all tags. In order to obtain convergence of results, simulations were conducted through exhaustive Monte-Carlo simulation (10000 random experiments for each number of tags) of Gen2 identification process. In our investigations we have not included radio propagation effects, but these methods can be applied to the estimate as done in [25]. Described identification process includes two specific cases which we consider in our simulations:

- To sense the environment, first frame size should be set to some value regarding the nature of the identification process. The impact on choosing a correct first frame size is investigated in [18]. Larger first frame size for the larger tag number will reduce tag identification time due to reduced number of collisions. But, if a number of tags to be identified is lesser, larger size of the first frame will cause delay due to a large number of empty slots occurred. Following [9] we set the first frame size to $Q_{\text{init}} = 4$. Additional results are provided for $Q_{\text{init}} = 6$.
- Another issue in the simulation is the all-collision scenario. In such scenario, it is not possible to extract any information without *a priori* knowledge on tag distribution in the area of interrogation. All-collisions may be caused by any large number of tags. In all-collision scenario we set next frame size to $Q_{new} = Q_{current} + 2$.

In this paper, for the all-collision scenario we consider the usage of equation (14) to provide the estimate in such scenarios, where (14) for $C = L$ becomes $\hat{n} = l = (1.2592 + 1.513L)\tan(1.234L^{-0.9907}L)$ depicted in Figure 8.

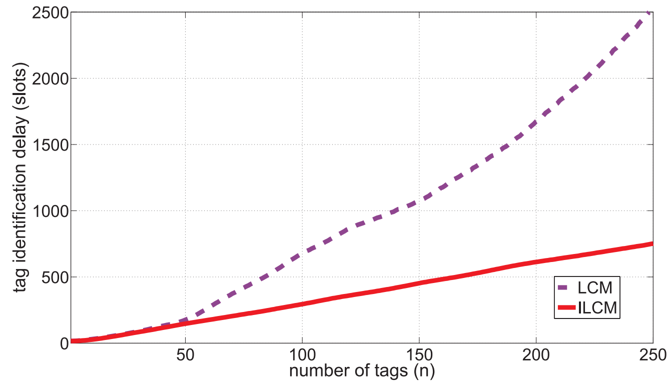


Fig. 7. Tag identification delay for ILCM and LCM [24] algorithm ($Q_{\text{init}} = 4$).

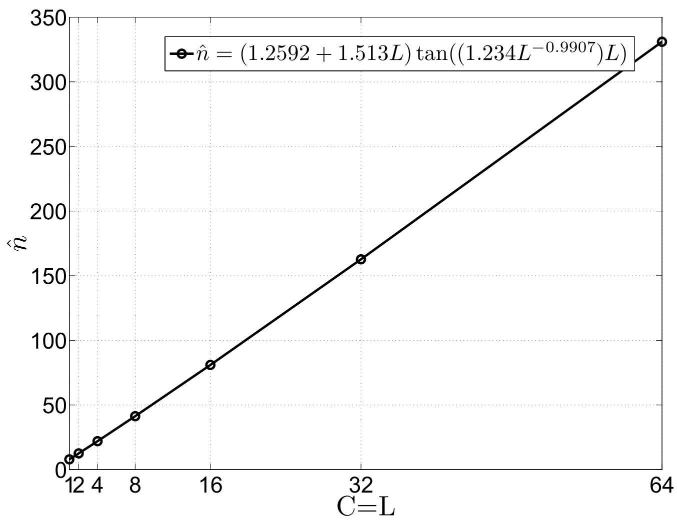


Fig. 8. ILCM tag estimate in the all-collision scenario.

More detailed performance comparison of ILCM algorithm and Q-Selection is given in [27]. Comparison between (13) and (14) for mean number of slots required to identify tags is depicted in Figure 7. LCM is designed only for $L = 8, 16$, and 32. For frames $L < 8$ it gives negative value estimates, and there has to be bounded with $Q = 3$, if $Q = \text{round}(\log_2(\hat{n} - S)) < 3$. However, where designed, LCM performance is close to the superior ILCM.

State-of-art tag estimates were implemented by algorithms provided in the Section II. Mean number of slots required to identify tags is given in Figure 9. Results are presented for Gen2 process of identification for the $Q_{\text{init}} = 4$ and $Q_{\text{init}} = 6$ including Vahedi, Chen, Vogt and ILCM estimate. Perfect estimate stands for $\hat{n} = n_c$, where n_c is known number of tags. As it can be seen from Figure 9, the mean number of slots to identify tags largely depend on the initial frame size. The more appropriate frame size is, the estimator error will be lower, which minimizes tag identification time. Hereby we do not include the comparison with Q-algorithm, since the constant C_Q should be chosen by *a priori* knowledge of the tag distribution (see Fig. 4). From the results it can be concluded that the estimator error is negligible, and the estimate result does not depend on the chosen method. This is mainly due to a relatively large span of tags which fits to the estimated Q,

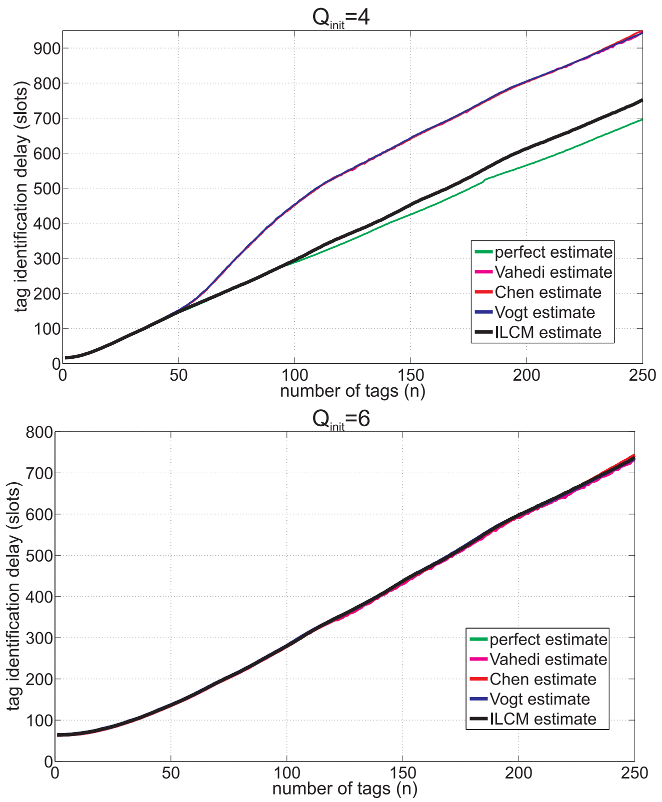


Fig. 9. Tag Identification delay in Gen2 process of identification for different algorithms (upper figure $Q_{\text{init}} = 4$, and lower $Q_{\text{init}} = 6$). The reason for the improvement in $Q_{\text{init}} = 4$ scenario is due to all-collision scenario where we provide the estimate $\hat{n} = l = (1.2592 + 1.513L)\tan(1.234L^{-0.9907}L)$.

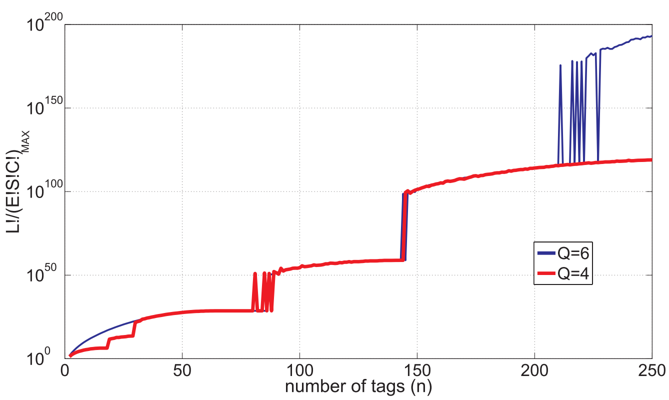


Fig. 10. Multinomial coefficient values that have to been computed for tag estimate in Vahedi and Chen method, for the first $Q_{\text{init}} = 4$ and $Q_{\text{init}} = 6$.

making the estimation error insignificant. Therefore, more attention should be given to algorithms implementation costs. Vahedi and Chen estimates include computations of factorial operations within multinomial coefficient which can yield large numbers required for temporary storage. Such large numbers which appear prior estimate are depicted in Figure 10. Real world implementation strongly depends on the reader Digital Signal Processor (DSP) architecture, and may become an issue regarding the size of processor registers and a way numbers are stored. This issue may be avoided by using the logarithmic approach described in [15]. Instead of looking for the minimum/maximum of the probability distribution, one can look for its logarithm since the minimum/maximum relation will stay the same. However, to apply this one needs to keep logarithms of factorials in the memory and call for the required value once needed. This brings up memory requirements instead of calculating factorials. Our observations did not consider this approach, since in the analysis of results we assume that the whole code has been ran within the processor cache.

TABLE II
FLOATING POINT OPERATION COSTS [15] AND [26]

Operation	FLOP cost
Addition, subtraction, multiplication	1
Comparison operation	2
Division, square root	10
Exponential, logarithmic and trigonometric function	50
Factorial	100

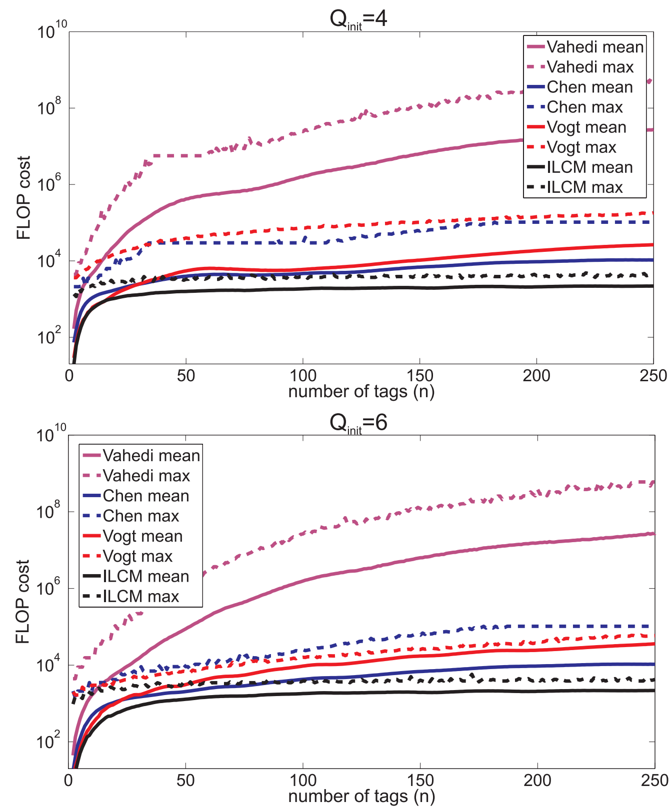


Fig. 11. Floating point operations cost to identify tags in Gen2 process of identification ($Q_{\text{init}} = 4$ and $Q_{\text{init}} = 6$).

FLOP costs actually depend on the DSP architecture. However, to evaluate total cost we used same values as in [15] and [26], given in Table II. Results of FLOP cost in process of tag identification ($Q_{\text{init}} = 4$ and $Q_{\text{init}} = 6$) are shown in Figure 11. As it can be concluded, ILCM provides the lowest cost in FLOP operation for its estimate. Due to ILCM estimate low FLOP cost, and DSP performance in Floating Point Operations Per Second per watt (FLOPS/watt), one may consider peak performance in tag identification delay along with saving the energy for the estimate computation. To elaborate what is the actual saving cost, in the next two subsections the case study on the real-world mobile RFID reader implementation is provided.

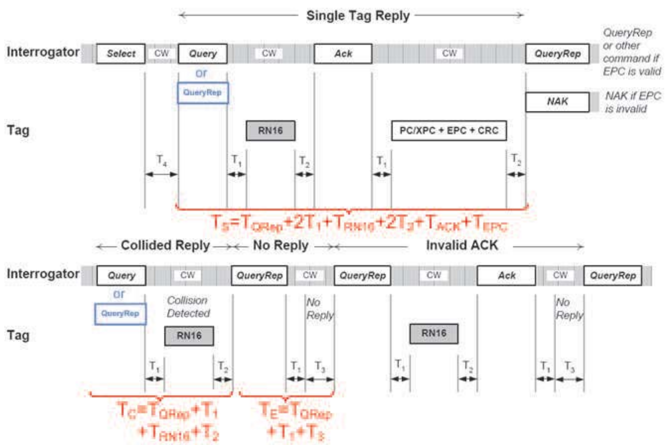


Fig. 12. Timing details for each slot type in Gen2 RFID.

A. Timing in Gen2

In Gen2, the interrogation is started by reader transmitting $Query$ command, which contains reader-tag interrogation parameters, including the size of frame Q. Upon receiving Q, tags initialize slot counters to random value between 0 and 2^Q-1, and the reader after interrogating slot decrements tags slot counters using $QRep$ command. When tags slot counters become zero, tags respond with the 16-bit random number $RN16$, which reader acknowledge with $ACKRN16$ command. After successful decoding of $ACKRN16$, tags complete their response by sending its ID, i.e. Electronic Product Code (EPC). If a collision occurs, $RN16$ could not be correctly detected and the reader it that case sends another $QRep$ to continue tag interrogation. Gen2 DFSA slots are units of time, where their exact durations are depending on the reader-tag interrogation parameters. Figure 12 gives all collision, empty and successful slots time details of Gen2 protocol. Our simulations use the 394kbps tag-reader link rate (data available in [28]), where timing details are shown in Table IV. Note that Gen2 supports 40kbps-640kbps tag-reader link rates, which has the great impact on tag identification delay (the impact on choosing different timing in terms of Q-algorithm performance is given in [29]). Chosen timing details are picked in that way due to energy saving discussion described in the next section. Using given timing, the throughput in terms of tags/s can be calculated as the ratio of successfully read tags divided with the duration of the frame

$$\text{tags/s} = \frac{N_S}{T_E \cdot N_E + T_C \cdot N_C + T_S \cdot N_S + T_{Query}} \quad (15)$$

where N_S, N_E, N_C are the number of successful, empty and collision slots, and T_S, T_E, T_C are their time durations. Gen2 throughput in terms of tags/s for timing specified in Table IV is shown in Figure 13, where it can be seen that changing the slot type duration affects the policy of choosing optimal Q. Therefore, the optimal Q values for given timing are given in Table III, and the tag identification time in Figure 14. The results are provided for Vahedi, Chen, Vogt, ILCM, perfect estimate, and Wu-Q algorithm. It can be seen that ILCM needs the minimum time to identify tags. Further, it can be seen that Wu-Q algorithm becomes unstable for $Q_{\text{init}} = 6$, which is not the case for the other described tag estimate methods (see Figure 9).

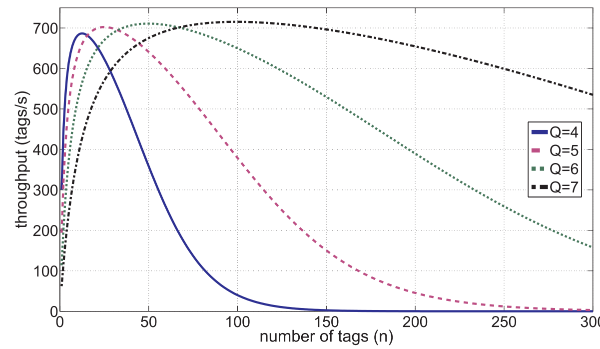


Fig. 13. Gen2 throughput in terms of tags/s for timing specification given in Table IV.

TABLE III
ACHIEVING MAXIMUM THROUGHPUT - Q-BINS OBTAINED FROM THE FIGURE 13

Q	0	1	2	3	4	5	6	7	8	9
n_{min}	0	-	2	4	7	16	33	67	136	274
n_{max}	1	-	3	6	15	32	66	135	273	549

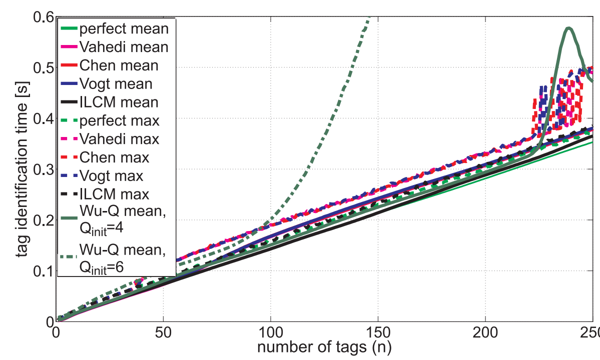


Fig. 14. Tag identification delay in Gen2 process of identification ($Q_{\text{init}} = 4$) for timing specification in Table IV.

TABLE IV
GEN2 READER INTERROGATION PARAMETERS

Parameter	**Duration**	**Parameter**	**Duration**
Tari	6.5μs	T_3	25.381μs
RTCal	16.25μs	PRT	57.594μs
BLF	394kHz	TFS	35.25μs
T_1	20.84μs	T_{Query}	236.34μs
T_2	7.61μs	T_{ACK}	181.5μs
TRext	1	T_{QRep}	67.75μs
M	1	T_S	1.1ms
T_{RN16}	126.9μs	T_C	223.11μs
T_{EPC}	695.43μs	T_E	113.97μs

B. Energy Efficiency Discussion

The importance of power saving for mobile RFID readers is elaborated in [30]. The focus is on the saving of the radio-power, since it is the main battery consumer. Authors also report that the mobile RFID reader battery drain out after 1.5 hours if the reader output power is set to the maximal value. In some scenarios the required output power can be reduced. Therefore, we consider the typical application of pallet bearing tagged products for fast identification. To consider power levels which are required in that scenario, we look for the tag sensitivity, i.e. for the transmit power level

TABLE V
TAG ESTIMATE IMPLEMENTATION COST TO IDENTIFY 250 TAGS

Algo	Order	t_c [ms]		E_c [mJ]		E_r [mJ]		$\frac{E_c}{E_c+E_r}$ [%]	
		mean	max	mean	max	mean	max	mean	max
Vogt	$O(n)$	1.3	9.1	0.97	6.8	5.71	7.67	14.52	46.99
Chen	$O(n)$	>0.53	>5.3	>0.4	>4	5.69	7.52	>6.56	>34.72
Vahedi	$O(nC^2)$	>1389	>28954	>1038	>21628	5.67	7.35	>99.46	>99.97
ILCM	$O(1)$	0.1	0.2	0.074	0.149	5.49	5.79	1.33	2.51

needed to establish reader-tag communication. According to the [28], required power level to power-up the state-of-the-art tag (Alien ALN-9640) is about -15dBm (for parameters specified in Table IV [28]). In typical pallet scenario (pallet dimension of 1.2m×1m) required reading range could be set to 2 meters in order to identify only pallet tags. Therefore, reader output power in that scenario to read tags can be calculated as [28]

$$P_{tag} = P_{reader} + G_{reader} + G_{tag} + \left(\frac{\lambda}{4\pi d}\right)^2 \text{[dBm]} \quad (16)$$

where P_{tag} is tag sensitivity, G_{reader} and G_{tag} are reader and tag antenna gains (both in dBi). $\lambda = c/f$ stands for the wave length ($c \approx 3 \cdot 10^8$m/s, f is the system frequency), and d is the distance between reader and tag. Typical Gen2 reader-tag settings include $G_{reader} = 6$dBi and G_{tag} for ALN-9640 tag is about 1.5 dBi [28]. λ for $f = 900$MHz, equals 0.33m, and the system required distance equals 2m. Then for tags which have sensitivity of -15dBm, required output power from (16) is $P_{reader} = 15.04$dBm (31.91mW), or even less for Battery Assisted Passive (BAP) RFID tags [30].

Tag estimate algorithms implementation cost details are hardware depending. To approximate total costs, we provide some numbers which allows revealing further insights in energy consumption. At first, it is important to note that ARM processors for mobile applications generally do not support floating point arithmetic, since they do not own floating point units. In such cases, calculus is usually being solved on the upper programming levels, using dedicated programming libraries. As a consequence, calculations are slow and should be minimized whenever possible. To demonstrate the algorithms feasibility in terms of power consumption we have considered Intel PXA270 microprocessor [31] (used both in mobile and fixed readers [32], [33]), which benchmarks in terms of floating point operations is online available [34], and report the time of 50.9ns (with the power consumption of 747mw [31]) to make an floating point multiplication, i.e. 1 FLOP. Further, total algorithm costs is calculated as $t = w + q$, where w stands for the FLOP cost, and q is the number of bytes of data exchanged between processor and memory [35]. To simplify the observations we consider $t = w$, since we assume that all of the code can be performed within the cache memory of the processor. This might not be always the case, but it represents the lower bound costs observation. Using described hardware performances, and the time required to identify 250 tags (Fig. 14), the energy consumption details are summarized in Table V. Results are given for PXA270 processor tag estimate implementation cost to identify $n = 250$ tags. t_c stands for the computation time of the algorithm, E_c is the energy cost for the estimate computation, and E_r stands for the reader radio front-end energy costs ($P_{\text{out}} = 31.91$mW). It can be seen that ILCM algorithm provides the lowest energy costs to compute estimate, along with the least energy required to identify tags. Note that Vahedi's and Chen's methods performances are given in lower bound terms due to multinomial coefficients which calculus is not directly supported with PXA270 32-bit architecture. This could be solved through additional software libraries, making the calculus more sophisticated. Note that Wu-Q algorithm is the most efficient in terms of requested calculus to provide the estimate. However, to use it for different Q_{init} it would need further optimization, making the estimate computation more complicated.

Approach in reducing energy costs is of the great importance in building Internet of Things (IoT) environment, where the new, energy efficient and cheap readers are to be employed for the ubiquitous sensing applications. As an example, authors in [36], report 40USD reader (using MSP430F5510 processor, allowing 25MIPS (integer operations)).

V. CONCLUSION

In this paper we analysed state of art algorithms of tag estimate in DFSA-based RFID systems. Along the analysis on their performance through efficient implementation, we provided the method called Improved Linearized Combinatorial Model (ILCM), which yields the estimate by calculating only parameter (k and l), allowing to yield an reasonable estimate even in all-collision scenario. Performance comparison shows that ILCM behaves similarly to the state of art algorithms in terms of tag identification delay (slots), but its computational complexity is lower, which is of the most interest to mobile RFID readers where reduction in FLOP cost may extend battery lifetime. Presented results shows the importance of real energy savings. Results are given through the use-case scenario for the concrete RFID Gen2 settings, where for given tag identification scenario, energy wasted on computation is comparable to the energy used by the radio front-end.

Future work will include implementation of given algorithms on Software Defined Radio Gen2 application [37] similarly as [24], including complete analysis on Gen2 system real performances (incorporating radio propagation effects).

REFERENCES

[1] D. D. Dobkin, *The RF in RFID*. Burlington, MA, USA: Elsevier, 2008.
[2] P. Šolić, J. Radić, and N. Rožić, "Software defined radio based implementation of RFID tag in next generation mobiles," *IEEE Trans. Consum. Electron.*, vol. 58, no. 3, pp. 1051–1055, Aug. 2012.

[3] W. He and L. Xu, "Integration of distributed enterprise applications: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 1, pp. 35–42, Feb. 2014.

[4] L. Catarinucci *et al.*, "Smart RFID antenna system for indoor tracking and behavior analysis of small animals in colony cages," *IEEE Sensors J.*, vol. 14, no. 4, pp. 1198–1206, Apr. 2014.

[5] A. Athalye, V. Savic, M. Bolic, and P. Djuric, "Novel semi-passive RFID system for indoor localization," *IEEE Sensors J.*, vol. 13, no. 2, pp. 528–537, Feb. 2013.

[6] B.-S. Choi, J.-W. Lee, J.-J. Lee, and K.-T. Park, "A hierarchical algorithm for indoor mobile robot localization using RFID sensor fusion," *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2226–2235, Jun. 2011.

[7] W.-J. Yoon and S.-H. Chung, "ISS-TCA: An identified slot scan-based tag collection algorithm for performance improvement in active RFID systems," *IEEE Trans. Ind. Electron.*, vol. 59, no. 3, pp. 1662–1672, Mar. 2012.

[8] (2014, Mar. 25). EPCglobal. EPCglobal Inc., Lawrenceville, NJ, USA [Online]. Available: http://www.gs1.org/epcglobal

[9] *Class1 Generation 2 UHF Air Interface Protocol Standard v1.2.0*, EPCglobal Inc., Lawrenceville, NJ, USA, Oct. 2008.

[10] C. Law, K. Lee, and K. Siu, "Efficient memoryless protocol for tag identification (extended abstract)," in *Proc. 4th Int. Workshop Discrete Algorithms Methods Mobile Comput. Commun.*, Aug. 2000, pp. 75–84.

[11] J. I. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Trans. Inf. Theory*, vol. 25, no. 5, pp. 505–515, Sep. 1979.

[12] C. Floerkemeier, "Infrastructure support for RFID systems," Ph.D. dissertation, Dept. of Comput. Sci., ETH Zurich Univ., Zürich, Switzerland, 2006.

[13] H. Wu and Y. Zeng, "Bayesian tag estimate and optimal frame length for anti-collision aloha RFID system," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 4, pp. 963–969, Oct. 2010.

[14] B. Knerr, M. Holzer, C. Angerer, and M. Rupp, "Slot-wise maximum likelihood estimation of the tag population size in FSA protocols," *IEEE Trans. Commun.*, vol. 58, no. 2, pp. 578–585, Feb. 2010.

[15] J. Vales-Alonso, V. Bueno-Delgado, E. Egea-Lopez, F. Gonzalez-Castano, and J. Alcaraz, "Multiframe maximum-likelihood tag estimation for RFID anticollision protocols," *IEEE Trans. Ind. Informat.*, vol. 7, no. 3, pp. 487–496, Aug. 2011.

[16] H. Vogt, "Efficient object identification with passive RFID tags," in *Proc. Int. Conf. Pervasive Comput.*, Zurich, Switzerland, Aug. 2002, pp. 98–113.

[17] *I-code1 Label ICS Protocol Air Interface Datasheet*, Phillips Semiconductor, Andover, MA, USA, Jan. 2005.

[18] W.-T. Chen, "An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length ALOHA," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 9–15, Jan. 2009.

[19] E. Vahedi, V. W. Wong, I. F. Blake, and R. K. Ward, "Probabilistic analysis and correction of Chen's tag estimate method," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 3, pp. 659–663, Jul. 2011.

[20] C. Floerkemeier, "Bayesian transmission strategy for framed ALOHA based RFID protocols," in *Proc. IEEE Int. Conf. RFID*, Grapevine, TX, USA, Mar. 2007, pp. 228–235.

[21] D. Lee, K. Kim, and W. Lee, "Q^+ *-algorithm*: An enhanced RFID tag collision arbitration algorithm," in *Ubiquitous Intelligence and Computing* (Lecture Notes in Computer Science), vol. 4611, J. Indulska, J. Ma, L. Yang, T. Ungerer, and J. Cao, Eds. Berlin, Germany: Springer-Verlag, 2007, pp. 23–32.

[22] H. Wu, Y. Zeng, J. Feng, and Y. Gu, "Binary tree slotted ALOHA for passive RFID tag anticollision," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 1, pp. 19–31, Jan. 2013.

[23] P. Šolić, J. Radić, and N. Rožić, "Algorithm for deriving optimal frame size in passive RFID UHF class1-gen2 standard using combinatorial model boundaries," *Automatika*, vol. 51, no. 3, pp. 255–263, Sep. 2010.

[24] P. Šolić, J. Radić, and N. Rožić, "Linearized Combinatorial Model for optimal frame selection in Gen2 RFID system," in *Proc. IEEE Int. Conf. RFID*, Orlando, FL, USA, Apr. 2012, pp. 89–94.

[25] J. Alcaraz, J. Vales-Alonso, E. Egea-Lopez, and J. Garcia-Haro, "A stochastic shortest path model to minimize the reading time in DFSA-based RFID systems," *IEEE Commun. Lett.*, vol. 17, no. 2, pp. 341–344, Feb. 2013.

[26] C. W. Ueberhuber, *Numerical Computation 1: Methods, Software, and Analysis*. Berlin, Germany: Springer-Verlag, 1997.

[27] P. Solic *et al.*, "Improved linearized combinatorial model (ILCM) for optimal frame size selection in ALOHA-based RFID systems," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Budapest, Hungary, Jun. 2013, pp. 1092–1097.

[28] P. Nikitin and K. V. S. Rao, "Effect of Gen2 protocol parameters on RFID tag performance," in *Proc. IEEE Int. Conf. RFID*, Apr. 2009, pp. 117–122.

[29] P. Šolić, M. Šarić, and M. Stella, "Tags/s-RFID reader-tag communication throughput using Gen2 Q-algorithm frame adaptation scheme," in *Proc. 5th Eur. Conf. Comput. Sci. (ECCS)*, Oct. 2013, pp. 100–105.

[30] X. Xu, L. Gu, J. Wang, G. Xing, and S.-C. Cheung, "Read more with less: An adaptive approach to energy-efficient RFID systems," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 8, pp. 1684–1697, Sep. 2011.

[31] *Intel XScale Bulverde PXA270 Processor, Electrical, Mechanical, and Thermal Specification*, Intel, Santa Clara, CA, USA, 2006.

[32] (2014, Mar. 25). *Intermec IP30/CN3E Mobile RFID Reader* [Online]. Available: http://www.intermec.com/products/ip30a/index.aspx

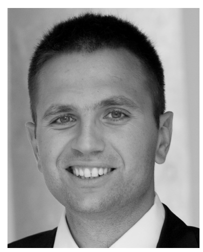
[33] (2014, Mar. 25). *Motorola FX9500 Fixed RFID Reader* [Online]. Available: http://www.motorolasolutions.com/US-EN/Business+Product+and+Services/RFID/RFID+Readers/FX9500_US-EN#specs_tab

[34] (2014, Mar. 25). *Performance Measurement on Arm* [Online]. Available: http://www.pengutronix.de/development/kernel/arm-benchmarks-20100729_en.html

[35] J. W. Choi and R. W. Vuduc, "How much (execution) time and energy does my algorithm cost?" *XRDS*, vol. 19, no. 3, pp. 49–51, Mar. 2013.

[36] P. Nikitin, S. Ramamurthy, and R. Martinez, "Simple low cost UHF RFID reader," in *Proc. IEEE Int. Conf. RFID*, Apr. 2013, pp. 126–127.

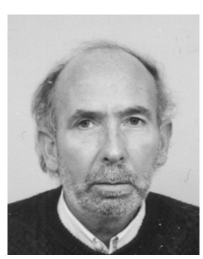
[37] M. Buettner and D. Wetherall, "A software radio-based UHF RFID reader for PHY/MAC experimentation," in *Proc. IEEE Int. Conf. RFID*, Orlando, FL, USA, Apr. 2011, pp. 134–141.



Petar Šolić received the M.S. and Ph.D. degrees in computer science from the University of Split, Split, Croatia, in 2008 and 2014, respectively. During his undergraduate study in 2007, he was a member of the national team, winner of the Imagine Cup competition representing Croatia in the Software Design category at the worldwide finals in Seoul, Korea. He is currently a Post-Doctoral Reseacher with the Department of Communication and Information Technologies, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split. His research interests include information technologies, RFID technology, and its application.

Joško Radić received the M.S. and Ph.D. degrees in communication engineering from the University of Split, Split, Croatia, in 2005 and 2009, respectively. He is currently a Post-Doctoral Researcher with the Department of Electronics, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split. His research interests include wireless communications, signal processing, multicarrier systems, and coding theory.

Dr. Radić is a member of the IEEE Communications Society, the Broadcast Technology, and Consumer Electronics and Oceanic Engineering Societies. He is also a member of the Croatian Communications and Information Society.



Nikola Rožić received the B.S.Eng. degrees in electrical engineering and electronics from the University of Split, Split, Croatia, in 1968 and 1969, respectively, and the M.S. and Ph.D. degrees from the University of Ljubljana, in 1977 and 1980, respectively. He is currently a Professor Emeritus with the Department for Electronics, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split. His research interests include information and communication theory, signal processing, source and channel coding, prediction methods, and forecasting. He is the Chief Editor of the *International Journal of Communications Software and Systems*, which is technically co-sponsored by the IEEE Communications Society ComSoc. He is a member of the IEEE Communications, Computer, Information Theory and Signal Processing Societies. He is currently the President of the Croatian Communications and Information Society, which is the sister society of the IEEE ComSoc.