Idle-slots elimination based binary splitting (ISE-BS) anti-collision algorithm for RFID

Jian Su, Zhengguo Sheng, Liangbo Xie, and Guangjun Wen, *Senior member, IEEE*

Abstract—Tag collision avoidance is critical to the success of data communications in radio frequency identification (RFID) system. This paper presents an efficient idle-slots elimination based binary splitting (ISE-BS) algorithm to improve the performance of RFID system. In ISE-BS, by introducing 1 bit random number Q and 16-bits random number serial identifier (SID) which are transmitted before data exchange, tag collisions can be informed and unnecessary data exchange between reader and tags can be further eliminated. Moreover, ISE-BS exploits Q to separate conflicting tags into '0-1' subsets randomly. Specifically, the tags in subset '0' will start to transmit in the next period, where the success flag signal reflects the immediate data transmission. The tags in subset '1' will wait in the pipeline. In such a way, the idle slots introduced by conventional binary splitting anti-collision algorithms can be removed with schedule of ISE-BS. Extensive simulation results show that ISE-BS outperforms the existing proposed algorithms.

Index Terms—RFID, anti-collision, BS, time efficiency.

I. INTRODUCTION

FAST identification is an urgent demand for modern radio frequency identification (RFID) systems especially in a dense tags environment, such as warehouse and supply chain management. Therefore, an efficient anti-collision algorithm for identifying multiple tags is required. Recently, many state-of-art algorithms have been proposed, which can be classified into probabilistic [1-5] and deterministic algorithms [6-7]. Technically, query tree (QT) [6-7] based algorithms are considered as a deterministic method derived from a collision bit identification and tracking technique. However, along with the increasing number of tags, the position of collision bit cannot be detected efficiently since the wide deviation of received signals at a reader [8-9]. Therefore, it is difficult to implement QT-based algorithms of UHF RFID in real-time systems.

In this paper, we focus on probabilistic algorithms, which can be divided into two categories, Aloha-based [1-4] and BS-based [5]. Aloha-based algorithms, such as dynamic framed slotted Aloha (DFSA), reduce the probability of tag collision by estimating tag backlog (unread tags) and optimizing frame

size. The existing DFSA algorithms include Maximum a posteriori (MAP) [1] estimation method, improved linearized combinatorial model (ILCM) [2] based algorithm, grouped dynamic frame slotted Aloha (GDFSA) [3], and fast anti-collision algorithm (FACA) [4] etc. However, Aloha-based algorithms cannot prevent all potential collisions and hence they may suffer from the 'tag starvation problem' in which a specific tag is not identified for a long time. In BS-based algorithms, concurrent tags usually form a single set. If a collision happens in one set, the reader will split it into two subsets and attempts to recognize them in turns. Although BS-based algorithms do not induce the tag starvation, the latency of tag identification is significant because the initialization of splitting set contains all the tags in one network. One representative of BS-based algorithms is adaptive binary splitting (ABS) [5] algorithm.

To improve the identification performance, we propose an anti-collision algorithm named idle-slots elimination based binary splitting (ISE-BS) for a UHF RFID system. Specifically, ISE-BS adopts 1 bit Q signal and 16-bits SID to assist tags splitting. The Q signal is transmitted before data transmission to potentially indicate whether the tag splitting is fulfilled. The idle slots introduced by conventional binary splitting anti-collision algorithms can be removed with the schedule of ISE-BS. Moreover, since all of collisions can be informed by Q and SID, respectively, the unnecessary data exchange between reader and tags can be further eliminated. The simulation results show that the proposed scheme can achieve much better performance than the other algorithms.

II. THE PROPOSED ISE-BS ALGORITHM

We assume that the channel is non-impaired [1-5] and a reader performs identification process repeatedly for object tracking and monitoring. In the ISE-BS, a reader maintains a counter R_c. Wherein $R_c<0$ means all of tags are identified by the reader during an identification process. ISE-BS also requests each tag to maintain an integer counter T_c, a random binary number generator (RBNG) Q, and a binary flag indicator F. At the beginning of tag identification, T_c, Q, F and R_c are initialized to 0. A tag transmits when $T_c = 0$. If $F = 0$, the tag is transmitting Q signal, otherwise SID or ID. Where SID is a 16-bits random generated number used for collision arbitration. The tags with the same value of T_c form a set. Tag collision happens when multiple tags choose the same set. Depend on the feedback, tags act as follows.

1) Q-success: If all of responding tags generate either $Q = 1$ or $Q = 0$, the reader successfully receiving a successful Q

TABLE I
AN IDENTIFICATION EXAMPLE OF BS ALGORITHM

Slot	Tag A (T_c, Q)	Tag B (T_c, Q)	Tag C (T_c, Q)	Action
1	(0, 1)	(0, 1)	(0, 1)	Collided
2	(1, x)	(1, x)	(1, x)	Idle
3	(0, 0)	(0, 1)	(0, 1)	Collided
4	(0, x)	(1, x)	(1, x)	A is identified
5	(-1, x)	(0, 1)	(0, 1)	Collided
6	(-1, x)	(1, x)	(1, x)	Idle
7	(-1, x)	(0, 0)	(0, 1)	Collided
8	(-1, x)	(0, x)	(1, x)	B is identified
9	(-1, x)	(-1, x)	(0, x)	C is identified

will broadcast a 'Q-success' feedback. The tags with $T_c = 0$ will perform: $F = 1$ and transmit its SIDs to the reader immediately.

2) *Q-collision:* If the reader receives a collided Q, it will send a feedback 'Q-collision'. The tags with $T_c = 0$ act $T_c = T_c + Q$, and $F = 0$. The tags with $T_c > 0$ perform $T_c = T_c + 1$. The reader will increase its counter R_c by 1.

3) *SID-success:* If the reader receives SID without collision, it will issue feedback 'SID-success'. The tags with $T_c = 0$ will perform: $F = 1$ and transmit its ID to the reader immediately.

4) *SID-collision:* If the reader receives a collided SID, it will issue feedback 'SID-collision'. The tags with $T_c = 0$ will act $F = 0$.

5) *ID-success:* If the reader receives a successful ID, it will issue feedback 'ID-success'. The counter of involved tag will decrement, and have $T_c = T_c - 1$. The reader will perform $R_c = R_c - 1$.

Fig. 1 shows the flowchart of ISE-BS algorithm. At the end of the identification process, the reader has a negative value of its R_c. The link timing between a reader and tags of the above slot types is depicted in Fig. 2. As can be observed, there are three slot types, i.e., successful, collision, and assisted collision. Where a successful slot contains a query command and three feedbacks: 'Q-success', 'SID-success', and 'ID-success'. A collision slot contains a query command and two feedbacks: 'Q-success' and 'SID-collision', and an assisted collision slot contains a query command and a 'Q-collision' feedback. To explain the process of the proposed ISE-BS algorithm, Tabs. I and II show the illustrative examples of ISE-BS and BS when there are three tags and all tags have initial T_c, Q, and F as 0 before tag identification. To explain the process of the proposed ISE-BS algorithm, Fig. 3 shows an illustrative example of ISE-BS when there are three tags and all of them have initial T_c, Q, and F as 0 before tag identification. 'x' represents no operation at current status. As can be observed from the example, the ISE-BS algorithm can eliminate the idle slots in traditional BS algorithm. Moreover, the total collision arbitration time is saved by adopting 1 bit Q signal, since its time duration is significantly less than that of ID. Further improvement can be achieved when the number of tags increases.

III. PERFORMANCE ANALYSIS

To illustrate the advantages of the proposed ISE-BS algorithm, we first analyze the total number of identification
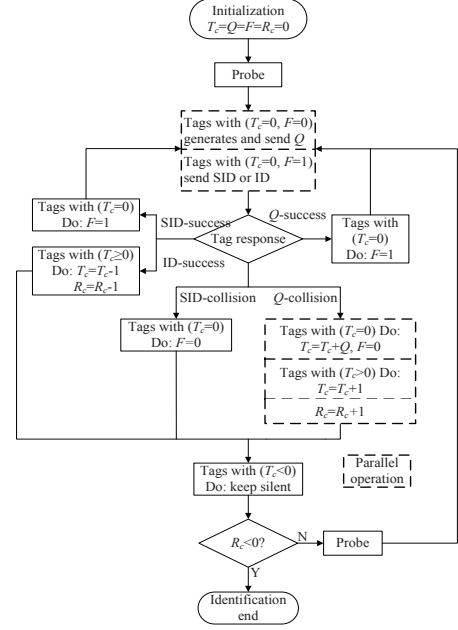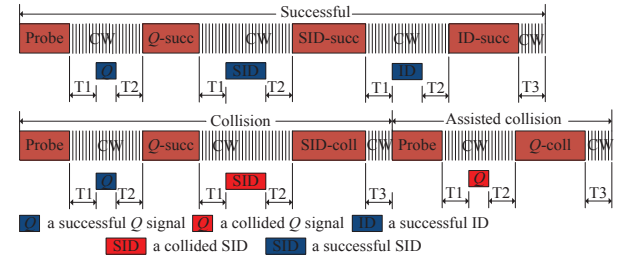

Fig. 1.   The flowchart of the proposed ISE-BS algorithm

Fig. 2.   Link timing between a reader and tags

TABLE II
AN IDENTIFICATION EXAMPLE OF ISE-BS ALGORITHM

Slot	Tag A (T_c, Q, F)	Tag B (T_c, Q, F)	Tag C (T_c, Q, F)	Action
1	(0, 1, 1)	(0, 1, 1)	(0, 1, 1)	Collided
2	(0, 0, 0)	(0, 1, 0)	(0, 1, 0)	Assisted Collision
3	(0, 0, 1)	(1, x, 0)	(1, x, 0)	A is identified
4	(-1, x, x)	(0, 1, 1)	(0, 1, 1)	Collided
5	(-1, x, x)	(0, 0, 0)	(0, 1, 0)	Assisted Collision
6	(-1, x, x)	(0, 0, 1)	(1, x, 0)	B is identified
7	(-1, x, x)	(-1, x, x)	(0, 0, 1)	C is identified

slots for recognizing all tags. Next, the system throughput and time efficiency of ISE-BS are provided. System throughput is defined by a ratio between the number of successful slots and that of total slots. The identification process of ISE-BS algorithm is similar with a binary tree protocol [10], where the total slots of ISE-BS are derived.

lemma 1. *Let $C(n)$, $I(n)$ be the number of collision nodes and idle nodes caused by the binary tree protocol traversing n tags. The total slots consumed by ISE-BS algorithm for identifying n tags is*

$$T_{slots}(n) = C(n) + n \qquad (1)$$

Proof: Tag identification of the binary tree protocol can be represented by a full binary tree because it splits the collided tags into two subsets. Therefore, all the internal nodes in the tree correspond to collisions and all the leaf nodes correspond to either idle slots or successful slots. However, the idle slots are eliminated by ISE-BS algorithm, the lemma1 is obtained. □

lemma 2. *For any n,*

$$T_{slots}(n) = n + \sum_{k=0}^{\infty} 2^k \left[1 - p(k)^n - n \cdot 2^{-k} p(k)^{n-1} \right] \quad (2)$$

where $p(k) = 1 - 2^{-k}$

Proof: Let $I(n,k)$, $S(n,k)$ and $C(n,k)$ denote the number of idle slots, successful slots and collision slots, respectively, in the depth k of the tree generated by the binary tree protocol for identifying n tags. $C(n)$ can be written as

$$C(n) = \sum_{k=0}^{\infty} C(n, k) \quad (3)$$

Let $p(k)$ denote the probability that a node in the depth k of binary tree is idle, then we have

$$I(n, k) = 2^k p(k)^n, \quad S(n, k) = n \cdot p(k)^{n-1} \quad (4)$$

Hence, $C(n, k)$ can be derived as

$$\begin{aligned} C(n, k) &= 2^k - I(n, k) - S(n, k) \\ &= 2^k \left[1 - p(k)^n - n \cdot 2^{-k} p(k)^{n-1} \right] \end{aligned} \quad (5)$$

According to (1), (3), and (5), the lemma2 can be yielded. □

Through lemma1 and lemma2, the system throughput of ISE-BS algorithm can be deduced as $U = n/T_{slots}(n)$. Considering the disparity between slot durations, the system throughput metric may be ineffective in terms of identification time to evaluate the performance of anti-collision algorithms. Therefore, a novel performance metric of time efficiency is presented in this paper. The collided slot of ISE-BS can be divided into two categories: ID-collision (collided slot) and Q-collision (assisted collision slot). The time efficiency can be defined as

$$\eta = \frac{n \cdot T_{ID}}{(n \cdot T_{succ} + N_{coll} \cdot T_{coll} + N_{a-coll} \cdot T_{a-coll})} \quad (6)$$

where T_{ID} is the time duration of transmitting tag's ID. n, N_{coll}, and N_{a-coll} are the number of successful slots, collided slots, and assisted collision slots, respectively. T_{succ}, T_{coll}, and T_{a-coll} are the time duration for above three slot types. The parameters in Eq. (6) are counted and measured by the reader during the identification process.

lemma 3. *For any n, the number of assisted collision slots in ISE-BS algorithm is*

$$N_{a-coll} = n - 1 \quad (7)$$

Proof: According to the principle of ISE-BS algorithm, an assisted collision slot represented tag set will be divided into two subsets, and has:

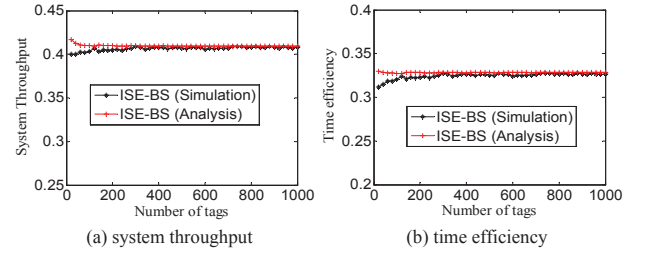$$1 + 2 \cdot N_{a-coll} + N_{coll} = n + N_{coll} + N_{a-coll} \quad (8)$$

Fig. 3. Comparison of analysis and simulation results for ISE-BS algorithm
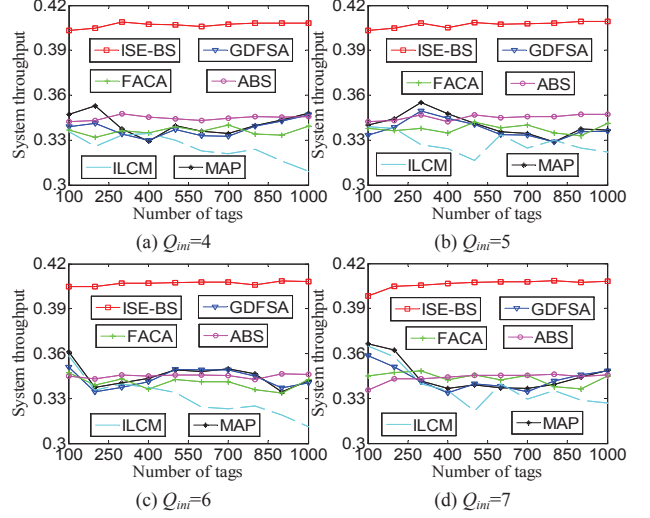

Fig. 4. Comparison of system throughput for various algorithms

According to Eq. (8), the lemma 3 can be yielded. □

Therefore, combing the above lemmas 1, 2 and 3, the expected time efficiency of ISE-BS algorithm can be expressed as

$$\eta_{\exp} = \frac{n \cdot T_{ID}}{n \cdot T_{succ} + (n-1) \cdot T_{a-coll} + (T_{slots} - 2n + 1) \cdot T_{coll}} \quad (9)$$

Fig. 3 shows the comparison between simulation and theoretical results of ISE-BS algorithm in system throughput and time efficiency. As can be seen, the analysis results are highly accurate and closed to the simulations.

IV. SIMULATION RESULTS

To validate the proposed ISE-BS algorithm, simulation results are compared with the existing state-of-art anti-collision algorithms including GDFSA [3], MAP [1], FACA [4], ILCM [2] and ABS [5]. During the simulations, a reader and a varying number of tags from 20 to 1000 in step of 100, are adopted in each simulation. Same as in [1-5], the wireless channel has no capture effect. The tag's ID is 96 bits, data rate of reader-to-tag and tag-to-reader are 80 kbps and 160 kbps, respectively. The magnitude of T1, T2, T3, and T4 are assumed as 25, 25, 50, and 100 microsecond (μs). All experimental results are averaged after 500 iterations.

The system throughput of various anti-collision algorithms with different initial frame sizes is described in Fig. 4. As can be seen, the performance of Aloha-based algorithms i.e.,
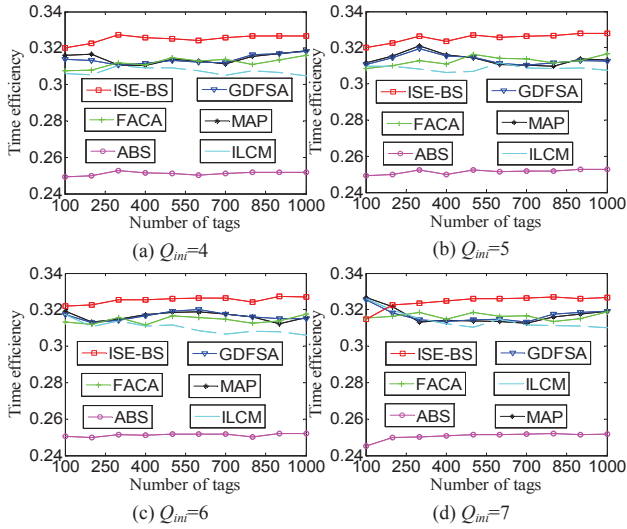
Fig. 5. Comparison of time efficiency for various algorithms

Method	R_U	Improvement	R_η	Improvement
MAP	6.86%	34.7%	3.59%	31.1%
GDFSA	5.30%	49.5%	3.26%	37.4%
FACA	4.46%	57.5%	1.57%	69.9%
ILCM	10.5%	-	5.21%	-
ABS	0.84%	92.0%	0.64%	87.8%
ISE-BS	0.59%	94.4%	0.90%	82.7%

With a smaller R, a better reliability can be achieved. Tab. III summarizes the fluctuation ratio of various algorithms in terms of system throughput and time efficiency. Where R_U and R_η denote the fluctuate ratio of system throughput and time efficiency, respectively. As can be observed from Figs. 4-5 and Tab. III, the proposed ISE-BS algorithm can achieve the best overall performance. Also, since our proposed algorithm is based on the same hardware platform of BS algorithm, it will not bring in extra challenges compared to other algorithms

V. CONCLUSION

In this paper, an efficient anti-collision algorithm has been proposed to overcome the tags collision problem. Our scheme is based on binary splitting by introducing Q signal to assist tags splitting. Since the idle slots are removed and collision arbitration time is saved, the performance can be improved. The simulation results has been shown that ISE-BS outperforms other probabilistic anti-collision algorithms.

REFERENCES

[1] 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, 2009.
[2] P. Solic, J. Radic, and N. Rozic, "Energy efficient tag estimation method for Aloha-based RFID systems," *IEEE Sensor J.*, vol. 14 no. 10, pp. 3637-3647, 2014.
[3] Y. Pan, Q. Peng, J. Lin, et al., "Reducing tag collision in radio frequency identification systems by using a grouped dynamic frame slotted Aloha algorithm," *Acta Phys. Sin.*, vol. 62 no. 14, pp. 148401-1-148401-08, 2013.
[4] W. T. Chen, "A fast anticollision algorithm for the EPCglobal UHF Class-1 Generation-2 RFID standard," *IEEE Commun. Lett.*, vol. 18, no. 9, pp. 1519-1522, 2014.
[5] J. Myung, W. Lee, and J. Srivastava, "Adaptive binary splitting for efficient RFID tag anti-collision," *IEEE Commun. Lett.*, vol. 10, no. 3, pp. 144-146, 2006.
[6] H. Landaluce, A. Perallos, L. Bengtssonand, and I. J. Garcia Zuazola, "Simplified computation in memoryless anti-collision RFID identification protocols," *Electron. Lett.*, vol. 50, no. 17, pp.1250-1252, 2014.
[7] J. Su, Z. Sheng, G. Wen, and V. Leung, "A time efficient tag identification algorithm using dual prefix probe scheme (DPPS)," *IEEE Signal Process. Lett.*, vol. 23 no. 3, pp. 386-389, 2016.
[8] C. Angerer, R. Langwieser, M. Rupp, "RFID reader receivers for physical layer collision recovery," *IEEE Trans. Commun.*, vol. 58, no. 12, pp. 3526- 3537, 2010.
[9] J. Su, Z. Sheng, D. Hong, G. Wen, "An effective frame breaking policy for dynamic framed slotted Aloha in RFID," *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 692- 695, 2016.
[10] J. I. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Trans. Inform. Theory*, vol. 25, no. 5, pp. 505- 515, 1979.

GDFSA, FACA, ILCM and MAP is significantly affected by the initial frame size. When the number of tags is large and the frame size is small, both Aloha-based methods are unable to adjust the appropriate frame size to fit the tag backlog (unread tags), and cause performance deterioration. In other words, the stability and scalability of these methods are poor to adapt to a wide range of tags. Compared to Aloha-based algorithm, the performance of ABS and ISE-BS are better since they are not affected by the variation of frame size. Specifically, the average system throughput of ISE-BS algorithm is about 0.4065 which is above the maximum system throughput of existing probabilistic algorithms.

Fig. 5 compares the performance of various algorithms in terms of time efficiency. As can be observed, some reference methods perform varied in terms of time efficiency. For example, although the total slots of ABS is more than that of Aloha-based algorithms, the time efficiency of ABS is poor than that of Aloha-based algorithms since disparity nature between slot durations. In Aloha-based algorithms, collision arbitration relies on 16 bits random number (RN16). In ABS algorithm, it depends on tag's ID. The transmission time duration of RN16 is more less than that of ID. Hence, the time efficiency of ABS algorithm is lowest. As a contrary, since most of collision arbitration is replaced by 1 bit Q signal in ISE-BS algorithm and idle slots are removed, the time efficiency of ISE-BS algorithm is significantly improved. ISE-BS algorithm performs better time efficiency than ABS, ILCM, FACA, GDFSA, and MAP by up to 29.3%, 4.95%, 3.61%, 3.38%, and 3.38%, respectively.

To compare the reliability [9], the fluctuate ratio of variables (e.g. system throughput or time efficiency) can be defined as

$$R = \frac{|V_{avg} - V_{\max}|}{V_{avg}} \qquad (10)$$

where V_{avg} and V_{max} denote the average and maximum value of variables when the number of tags varies from 100 to 1000 in step of 100 with an initial frame size of 16, 32, 64, and 128, respectively. R indicates the performance fluctuation.