An Improved Q-Learning Algorithm Integrated into the Aloha Anti-Collision Protocol for Energy-Efficient RFID Systems

Van-Hoa Le
School of Hospitality and Tourism
Hue University, Vietnam
levanhoa@hueuni.edu.vn

Duc-Nhat-Quang Nguyen
Faculty of Electricity, Electronics and Material Technology, University of Sciences, Hue University, Vietnam
ndnquang@hueuni.edu.vn

Viet-Minh-Nhat Vo
Institute for Educational Testing and Quality Assurance, Hue University
Vietnam
vvmnhat@hueuni.edu.vn

Abstract: *The collision in Aloha-based Radio Frequency Identification (RFID) systems is inevitable due to the random medium access nature of the Aloha protocol and the unknown number of tags within the reader's coverage. Various Aloha anti-collision protocols have been proposed, and reducing collisions has always been the top priority. However, merely reducing collisions can increase the number of idle slots, the number of interrogation epochs, and bandwidth usage. This article proposes an approach to integrating Q-learning into the Aloha anti-collision protocol, in which Interrogation Efficiency (IE), resulting in Energy Efficiency (EE), is the top priority. Two cases of fixed and dynamic frame sizes are considered. Experimental results show that the Q-learning-integrated Aloha anti-collision protocols achieve the highest IE, in which the number of collision slots, idle slots, and interrogation epochs are reduced. The dynamic-frame Q-learning-integrated Aloha anti-collision protocol achieves the best IE thanks to its ability to adjust frame size dynamically*.

Keywords: *RFID system, aloha protocol, anti-collision, Q-learning, energy-efficiency*.

Received July 11, 2024; accepted October 24, 2024
https://doi.org/10.34028/iajit/22/1/5

1. Introduction

Radio Frequency Identification (RFID) is one of the foundational technologies of the Internet of Things that has attracted much attention from academia and industry. In many practical applications, such as real-time inventory detection or automatic product identification in supply chain management, there are situations where multiple tags, known as appearing tags are within the coverage of a reader. In response to a reader's interrogation, the passive tags backscatter the signal coming from the reader. However, when multiple tags backscatter simultaneously, a collision occurs. In principle, anti-collision algorithms in wireless networks can be applied to resolve collisions in RFID systems. However, with the high asymmetry in which the reader is rich in resources and the passive tag has minimal storage and computing capabilities, the reader performs most of the processing. Given this reality, only basic anti-collision protocols are recommended to be implemented in RFID systems [5, 13].

This paper focuses on Aloha anti-collision protocols. The operating principle of Aloha protocols is that the maximum backoff time (frame) is divided into 2Q time slots, where Q is an integer assigned to the tags by the reader through a communication channel between the reader and the tags. Upon receiving a request from the reader, each tag independently and randomly selects a time slot marked by an integer in the range [0, 2Q-1]. In the Electronic Product Code-global-Class1-Generation2 (EPC-C1G2) protocol [16], a tag first backscatters its chosen integer, called the tag handle, when the time reaches the tag's chosen time slot. If the reader receives only one tag handle, it sends an ACKnowledgment (ACK code) signal to notify that the tag can backscatter more of its identification (tag ID). If the reader receives two or more tag handles, a collision occurs, and the tags are not acknowledged. These tags then independently and randomly reselect a time slot in the next round of interrogation. The identification process is repeated until the reader has identified all tags.

The Aloha anti-collision protocol works well when the frame size matches the number of appearing tags. However, the protocol's efficiency worsens as the number of tags increases while the frame size is fixed. Therefore, many dynamic Aloha anti-collision algorithms have been proposed to improve system efficiency, in which the frame size is dynamically adjusted according to the estimated number of tags, as in [17, 18]. The difference in the above protocols lies in the different methods of estimation. However, they require multiple rounds of interrogation before the identification process can optimize the frame size.

An alternative approach is to adjust the frame size based on a learning process in which the knowledge is acquired from the environment through tag identification results. The learning method can be

reinforcement learning, such as Q-learning [9, 10, 14]. Because an RFID system's computing and storage capacity is concentrated at the reader, Q-learning is especially suitable for the learning model based only on current information computed at the reader; tags only backscatter data in their selected time slot.

The paper presents an improved Q-learning integrated into the Aloha anti-collision protocol for energy-efficient RFID systems. Specifically, we suggest an energy-efficient Q-learning algorithm for adjusting frame size by learning from tag interrogation results. In most previous studies, anti-collision algorithms aim to reduce the number of collision slots, which often increases the number of idle slots, causing a waste of bandwidth occupied by idle slots. The result is reducing the overall performance of the entire system. The paper proposes an energy-efficient Q-learning algorithm in which the Interrogation Efficiency (IE) is used to determine the reward. For each reader, improving IE increases Energy Efficiency (EE). The cases of fixed and dynamic frame sizes are also considered and analyzed.

Contributions to the paper include:

- Proposing an improvement of the Q-learning algorithm, called the energy-efficient Q-learning algorithm, which adjusts frame size based on EE;
- Integrating the energy-efficient Q-learning algorithm into the Aloha anti-collision protocol with the fixed frame, called the Fixed-frame Q-learning-integrated Aloha Anti-Collision (FQAAC) protocol, and with the dynamic frame, called the Dynamic-frame Q-learning-integrated Aloha Anti-Collision (DQAAC) protocol; and
- Implementing and evaluating the efficiency of FQAAC and DQAAC by comparing them with Framed Slotted Aloha (FSA) and DFSA protocols.

The remainder of the paper is organized as follows. Section 2 introduces the Aloha protocol for RFID systems and the Q-learning algorithm. A review of related works related to the application of Q-learning in the Aloha anti-collision protocol is analyzed in section 3. An improvement of the Q-learning algorithm integrated into the Aloha anti-collision protocol is presented in section 4, where the EE model in tag interrogation and the energy-efficient Q-learning algorithm are discussed. Simulation results are compared and evaluated in section 5. Finally, the conclusion is provided in section 6.

2. Background

2.1. Aloha Anti-Collision Protocols

Aloha is a data link layer multiple access protocol that describes how multiple terminals can access a transmission channel at random intervals. Due to the random and independent nature of access, collisions are unavoidable. An improvement of Aloha, named FSA [12], is proposed to reduce collisions for multiple access. In FSA, a frame is divided into multiple time slots, and a tag randomly chooses one to transmit data. When a collision occurs, the tag randomly reselects another time slot and waits for the next frame to transmit its data.

The FSA has a static frame size, which means the frame size is determined from the beginning and does not change during the tag interrogation process. However, determining the correct initial frame size is difficult because the number of appearing tags in the reader's coverage area is unknown. The Dynamic FSA protocol (DFSA) [3], is proposed to improve the system efficiency, where the frame size is adaptively adjusted based on the estimate of the number of remaining (uninterrogated) tags in the reader coverage. Several techniques for estimating the number of tags have also been proposed to reduce collisions and enhance system efficiency. Recently, reinforcement learning-based approaches have also been introduced [9, 10, 14] that can adjust the frame size without estimating the number of tags. This paper focuses on the reinforcement learning-based approach.

2.2. The Q-Learning Algorithm

The reinforcement learning model considered in this paper is Q-learning [7], which is, in essence, a memoryless reinforcement learning algorithm that learns by interacting with the environment and determining the Q-value for a state-action pair, $Q(s_t, a_t)$, where st and at is the state and the action at time t. The Q-value represents the priority of choosing the action at over other available actions when the system is in the state s_t. Figure 1 describes the principle diagram of the Q-learning algorithm.

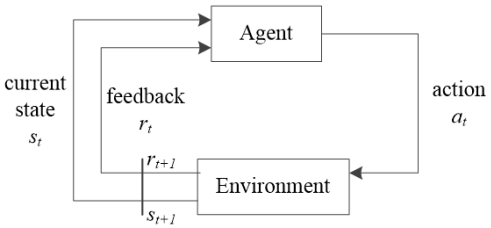


Figure 1. Principle diagram of the Q-learning algorithm [7].

Formally, for each state $s_t \in S$ and action $a_t \in A$, the Q-value is determined in Equation (1),

$$Q_{t+1}(s_t, a_t) = (1-\alpha)Q_t(s_t, a_t) + \alpha(r_t + \gamma \max_{a \in A} Q_t(s_{t+1}, a)) \quad (1)$$

where α is the learning rate, γ is the discount factor, and r_t is the delayed reward.

The value $\alpha \in [0, 1]$ controls the rate at which learning occurs, and $\gamma \in [0, 1]$ controls the readiness or delay of the reward r_t. The goal of the reward r_t is to guide the agent to a goal by rewarding or punishing the agent for the action at performed in the state s_t. The reward

function needs to be carefully defined to guide the Q-learning algorithm to convergence in a reasonable time.

The agent's learning goal is to map each state s_t to an action at that maximizes the reward. However, learning often chooses the known optimal action without occasionally probing whether a globally optimal solution can be reached. This approach can guide the agent to a local optimal solution. Therefore, several exploration-exploitation strategies have been proposed in the literature to address the problem. Our study chooses the epsilon-greedy strategy [15] to balance exploitation and exploration. Following the epsilon-greedy strategy, the agent sometimes chooses an action with a lower Q-value with the probability ε.

3. Related Works

There have been many studies on applying Q-learning to the Aloha anti-collision protocol, but mainly for wireless sensor or ad hoc networks [2, 4, 15], where the nodes involved in data transmission have massive computing and storage. For RFID systems where tags have poor computing and storage capacity, research on applying Q-learning to the Aloha anti-collision protocol is quite limited. The following are analyses and evaluations of some applications of Q-learning to the Aloha anti-collision protocol in RFID systems.

Xu and Yang [14] proposed an algorithm based on Q-learning, in which the existing environmental conditions and those of the following n states are considered. The Q-value is then determined by

$$Q_{t+1}(s_t,a_t)=\begin{cases} Q(s_t,a_t)+\alpha\left(c'_t+\sum_{j=1}^{n-1}(\lambda\gamma)^j c_{t+1}\right) & \text{if } s\neq s_t \text{ and } a\neq a_t \\ Q(s_t,a_t) & \text{otherwise} \end{cases} \quad (2)$$

where, $c_t=r_t+\gamma V_{t-1}(s_{t+1})-V_{t-1}(s_t)$, $c'_t=r_t+\gamma V_{t-1}(s_{t+1})-Q(s_t,a_t)$ and $V_t(s_t)$ is a merit function.

Simulation results show that although the proposed algorithm is more complex than the traditional Q-learning algorithm, it significantly reduces collisions. However, the efficiency of the new algorithm depends on the n-state parameter, but how to determine n has not been analyzed. As the number of states increases, the computational complexity explodes.

Loganathan *et al*. [10] and Loganathan *et al*. [9] suggested a Reinforcement Learning-based Dynamic Aloha anti-collision (RL-DFSA) protocol to provide better time efficiency while saving energy by reducing the overhead of control messages. RL-DFSA includes a policy for the reader to adjust the frame size between different estimates of the number of tags. The estimate is calculated based on the inference that the number of collision tags in a time slot can only be equal to or greater than two. Therefore, RL-DFSA divided the action space of the Q-learning algorithm into 11 levels to increase the smoothness of adjusting the frame size while not excessively increasing the algorithm's complexity.

Anti-collision solutions proposed for RFID systems must consider the asymmetry, where readers are rich in computing and storage resources while passive tags only backscatter their ID and do not know anything about the surrounding environment. Reducing the complexity of the implemented anti-collision protocol is a priority that needs attention. The paper proposes an improvement to the Q-learning algorithm, in which frame size adjustment is mainly based on IE. Details of our proposal are presented in the next section.

4. An Improvement of Q-Learning Based on Energy Efficiency

4.1. The Energy Efficiency Model in Tag Interrogation

The tag interrogation model considered in the paper corresponds to EPC-C1G2 standards, where the communication time between the reader and tags is divided into slots [16]. Figure 2 shows the time associated with three types of collision, success, and idle slots. The reader begins transmission by sending a command during a time t_R. The reader maintains the downlink carrier, also known as Continuous Wave (CW), so that tags can utilize its power to backscatter their data. After each command, there is a time T_1 required for the tag to generate its response and a time T_2 required for the reader to receive the tag data. A slot is considered idle when the reader waits for an unresponse in a time T_3. Tag data is responded to in a time t_T.

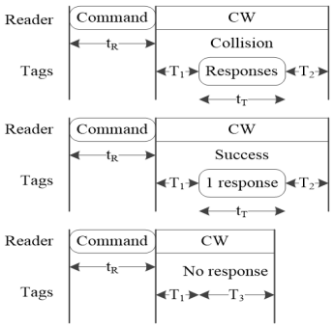


Figure 2. Example of a collision/success/idle slot in the tag interrogation model [8].

The energy consumption model considered in the paper is mainly the energy consumed by the reader, which is a function of the data transmission and reception time. The consumed energy thus includes the energy for the reader to transmit command, the energy to maintain CW to power passive tags (P_{tx}), and the energy to receive data from the tag (P_{rx}). Therefore, the total energy consumed during an interrogation round (E) is expressed in Equation (3).

$$E=E_c+E_s+E_i=\sum_{j=1}^{c+s}\left(P_{tx}\times\left(t_{Rj}+T_1+t_{Tj}+T_2\right)+P_{rx}\times t_{Tj}\right)+\sum_{j=1}^{l}\left(P_{tx}\times\left(t_{Rj}+T_1+T_3\right)\right) \quad (3)$$

where, E_c, E_s and E_i are the energy consumed in collision, success and idle slots, respectively. C, S and I are the number of collision, success and idle slots, respectively.

EE is determined by the ratio of the energy consumed for success slots (E_s) to the total energy to identify all tags (E_c+E_s+E_i). We assume that the times of collision, success, and idle slots are approximately equal, i.e., T; EE is thus equivalent to IE as in Equation (4).

$$EE = \frac{E_S}{E_c + E_s + E_i} = \frac{\sum_{j=1}^{S}\left(P_{tx} \times T + P_{rx} \times t_{Tj}\right)}{\sum_{j=1}^{C+S}\left(P_{tx} \times T + P_{rx} \times t_{Tj}\right) + \sum_{j=1}^{I}\left(P_{tx} \times T\right)} \approx \frac{S}{C+S+I} = IE \quad (4)$$

In other words, we can determine the EE of the Aloha-based anti-collision protocol by evaluating the IE. The EE increases if the number of collision (and idle) slots decreases. As shown in Figure 2, the idle slot size is always smaller than the collision slot size, so the more the collision slots reduce, the more the EE increases.

4.2. The Energy-Efficient Q-Learning Algorithm

The energy-efficient Q-learning algorithm is run on the reader. A Q-table is created where each row corresponds to each tag, and the number of columns corresponds to the number of slots in a frame. Assuming there are n appearing tags and the frame has L slots, each cell (i, j) of the Q-table carries a Q-value, $Q(s_t, a_t)$, representing the priority to select the slot i, i=1..L, by tag j, j=1..n. The Q-value is initialized at 0 and is updated by an EE function as in Equation (5), where depending on the current interrogation state (s_t), an action at is chosen so that the IE does not decrease. In addition, a reward r_t is also used to navigate the IE function as in Equation (6).

$$f_{EE}(s_t, a_t) = r_t + \gamma \max_{a \in A} Q_t(s_t, a) \quad (5)$$

$$r_t = \begin{cases} S/(S+C+I) & \text{if success} \\ 0 & \text{if idle} \\ -S/(S+C+I) & \text{if collision} \end{cases} \quad (6)$$

Equation (7) shows the updating of Q-value.

$$Q_{t+1}(s_t, a_t) = (1-\alpha)Q_t(s_t, a_t) + \alpha f_{EE}(s_t, a_t) \quad (7)$$

Two cases of integrating Q-learning into the Aloha anti-collision protocol, fixed frame FQAAC and dynamic frame DQAAC, are considered, respectively. Algorithms (1) and Algorithm (2) describe the workflow of the FQAAC and DQAAC algorithms.

Algorithm 1: FQAAC Algorithm.

Input: ε, α, γ // exploration-exploitation rate, // learning rate, discount factor
Output: C, I, S // collision, idle and success slots
Process
1. Initiate t=0 and Q-table: Q(s_t, a_t), s_t ∈S and a_t∈A
2. Initiate C=0, I=0, S=0
3. while t<max_iteration or n>0 do
4. The reader sends a command to the tags in its coverage
5. Select an action: Each tag chooses a slot in its row based on the epsilon-greedy strategy: if ε=0.1, a slot is selected randomly. Otherwise, the highest Q(s_t, a_t) is selected.
6. Update the reward r_t by Equation (6)
7. Update the Q-value by Equation (7)
8. Update C, I, S
9. t=t+1
10. end while

Algorithm 2: DQAAC Algorithm.

Input: ε, α, γ // exploration-exploitation rate, // learning rate, discount factor
Output: C, I, S // collision, idle and success slots
Process:
1. Initiate t = 0 and Q-table: Q(s_t, a_t), s_t∈S and a_t∈A
2. Initiate C=0, I=0, S=0
3. Estimate n and initiate L //estimate the number of tags // and initiate the frame size
4. while t<max_iteration or n>0 do
5. The reader sends a command to the tags in its coverage
6. Select an action: Each tag chooses a slot in its row based on the epsilon-greedy strategy: if ε=0.1, a slot is selected randomly. Otherwise, the highest Q(s_t, a_t) is selected.
7. Update the reward r_t by Equation (6)
8: Update the Q-value by Equation (7)
9. Update C, I, S
10. Update n=S+2.39C // estimate the remaining tags
11. Update the frame size L value based on n as in Table 1
12. t=t+1
13. end while

One limitation of FQAAC is the fixed frame size, so collision occurrence is still significant, especially when tag density is high. DQAAC improves on FQAAC by dynamically adjusting the frame size. Note that the number of unrecognized tags (n) reduced at the t+1th interrogation epoch equals the number of success slots (S) at the t^{th} interrogation epoch. Therefore, the frame size at the t+1th interrogation epoch can be reduced by the number of success slots at the tth interrogation epoch plus a variance (v). The value L is updated depending on the estimated number of remaining uninterrogated tags (Lines 10 and 11 in Algorithm (2)) and two upper/lower thresholds, as described in Table 1. Accordingly, the Q-table is updated.

Table 1. The upper and lower thresholds corresponding to the frame size.

Frame size	1	4	8	16	32	64	128	256
Lower	-	-	-	1	10	17	51	112
Upper	-	-	-	9	27	56	129	∞

With DQAAC, because the frame size changes, the dimension of the Q matrix also changes. There are two cases to be considered:

1. The Q matrix is reset, meaning the Q-values are reset to 0 every time the frame size changes.
2. The Q matrix continues to inherit the learning results of the previous Q matrix.

In fact, the Q matrix size will become smaller over time as the number of remaining uninterrogated tags decreases. Inheriting the knowledge learned from the previous Q matrix will help the algorithm converge faster and thus reduce the resources needed by the system.

The computational complexity of FQAAC and DQAAC is $O(m)$, where m is the number of iterations required to identify all n tags. The complexity is also equal to that of FSA and DFSA. However, FQAAC and DQAAC have an additional operation of updating Q-table, so the actual complexity of FQAAC and DQAAC is $O(n*L)$.

4.3. The Operation Model of FQAAC and DQAAC Protocols

Implementing the FQAAC and DQAAC algorithms requires the device to have memory and computing capacity. Due to the asymmetric nature of RFID systems, where readers are resource-rich, while tags have limited memory and computing capacity, the implementation of FQAAC and DQAAC algorithms takes place on the reader. Current readers can compute like a minicomputer with memory of 128MB or more [1]. For passive tags, in addition to the ability to match the prefix received from the reader with its ID, some tags also have memory (i.e., EPC memory) to maintain some variables and perform some simple calculations to update the values for these variables [1]. Accordingly, the FQAAC and DQAAC protocols are implemented as follows.

First, the reader initializes the Q-table with the cell values equal to 0. Depending on the status of the tag responses in each interrogation cycle, the cell values are updated by Q-values (as in Equation (7)). The packet command sent from the reader carries a reference list in which each tag registers a cell index, and the cell value is the time slot at which the tag responds to its data. The reference list is carried in the data field, as shown in Figure 3.

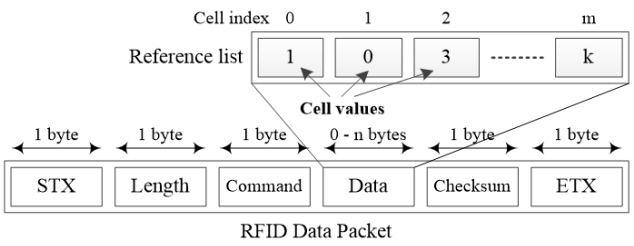


Figure 3. Integrating the reference list in RFID Data packet structure [11].

Each tag maintains a pointer variable (p) that indicates its cell position in the reference list that is carried in the packet command. Upon receiving a command, the tag checks the cell value at pointer p in the reference list. If this value is zero, the tag is determined to have been successfully identified and is silent. Otherwise, the cell value is the suggested slot for the tag to respond. Figure 4 illustrates how the FQAAC or DQAAC protocol works.

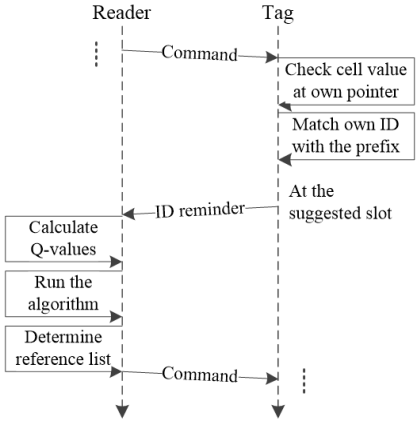


Figure 4. The operation model of FQAAC or DQAAC protocols.

5. Simulation and Analysis

Four algorithms, FSA, DFSA, FQAAC and DQAAC, are implemented on a PC with 2.4 GHz Intel Core 2 CPU, 8G RAM, in Python. The simulation parameters are described in Table 2.

Metrics to evaluate the algorithms include:

- *IE*: determined by the ratio of the number of success slots to the total number of collision, success and idle slots as in Equation (8).

$$IE = S / (C + S + I) \tag{8}$$

- Runtime: the total time needed to interrogate all tags.
- Bandwidth Efficiency (*BE*): calculated by the amount of bandwidth reduced due to the reduced frame size. In the paper, the BE is the ratio of the number of slots reduced ($framesize_{DQAAC}$-$framesize_{DFSA}$) to the number of slots normally needed ($framesize_{DFSA}$) as in Equation (9).

$$BE = \frac{framesize_{DQAAC} - framesize_{DFSA}}{framesize_{DFSA}} \tag{9}$$

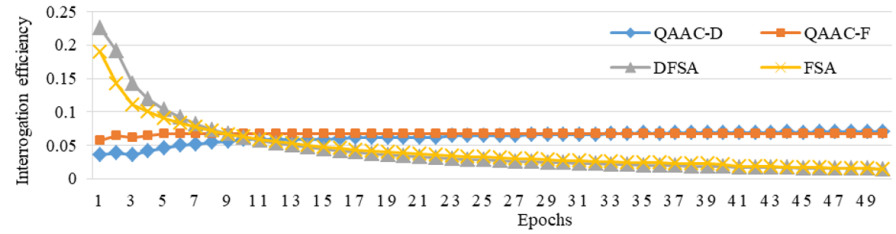
Table 2. Simulation parameters.

Parameters	Value
Number of uninterrogated tags (n)	from 10 to 100 tags
Frame size (L)	from 16
Maximal interation ($max\_interation$)	1000
Exploration-exploitation rate (ε)	0.1
Learning rate (α)	0.1
Discount factor (γ)	0.9

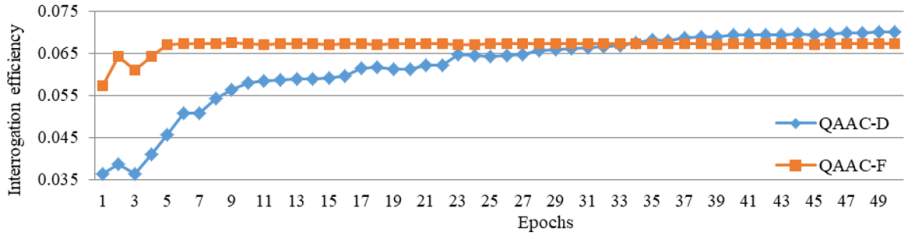
5.1. Interrogation Efficiency (IE)

The IE over epochs is shown in Figure 5-a), where initially, FSA and DFSA have high IE (in which DFSA is better than FSA), but their IE gradually decreases over time (epochs). The reason is that, with an initial frame size of 64 and 100 tags randomly distributed in the reading area, the probability of a tag successfully accessing an empty slot is relatively high. In the early stages, DFSA's IE is better than FSA's IE because DFSA adjusts the dynamic frame size from 64 to 128

(Figure 9), and as a result, the number of success slots achieved is high. However, later on, as the number of remaining unrecognized tags decreased, the number of collision slots also decreased, but the number of idle slots remained high. According to Equation (4), IE is therefore reduced.



a) The interrogation efficiency over epochs of FSA, DFSA, FQAAC, and DQAAC.



b) A comparison of the interrogation efficiency over epochs between FQAAC and DQAAC.

Figure 5. The IE of 4 algorithms over epochs (with 100 tags and frame size of 64).

For FQAAC and DQAAC, their IE is initially low but then increases over time and is better than the IE of BFSA and DFSA, as shown in Figure 5-a). To clarify, Figure 5-b) depicts the comparison between FQAAC and DQAAC, in which DQAAC initially has a lower IE than FQAAC but later gets better and surpasses FQAAC. The reason is as follows. With FQAAC, initially, the frame size is initialized to 64, and the tags choose empty slots randomly. The knowledge at this time is not enough to help each tag accurately determine the empty slot. Over time, learning brings more knowledge, and with the number of remaining unrecognized tags decreasing, the IE of algorithms with Q-learning improves more clearly. As for FQAAC, its IE quickly saturates because the frame size does not change, so the number of idle slots increases even though the number of collision slots decreases. According to Equation (4), the IE value remains constant. For DQAAC, adaptively adjusting the frame size (Figure 9) reduced the number of idle slots. Along with the decrease in the number of collision slots, the IE of DQAAC gradually increases (Figure 5-b)) according to Equation (4).

Regarding the efficiency comparison of the four algorithms when the number of tags increases, Figure 6 shows that the Q-learning integrated Aloha anti-collision protocols, FQAAC and DQAAC, consistently achieve better results than the traditional Aloha protocols, BFSA and DFSA. Specifically, when the number of tags is sparse, there is a significant difference in the IE between the Q-learning integrated and traditional algorithms. Calculating Q-values based on the knowledge of the system state helps the tags choose slots with low collision probability. However, when the tag density increases, the frame size limit (Table 1) does not allow the infinite expansion of the number of slots, so the IE gradually decreases. Despite the decrease in efficiency, DQAAC is always better than FQAAC, thanks to the policy of adjusting the number of time slots, which reduces the number of collisions. The convergence of the IE of the algorithms reflects the existence of an efficiency threshold limited by the tag density. Beyond this threshold, Q-learning-based collision prevention algorithms are no longer practical.

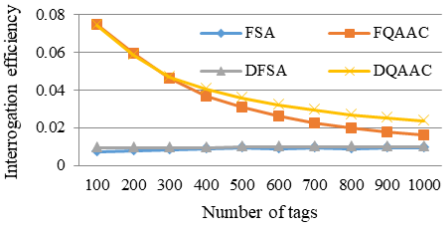


Figure 6. The IE gradually decreases as the number of tags increases.

5.2. Runtime

In order to achieve high IE, Q-learning-integrated Aloha anti-collision protocols suffer from loss in runtime. Figure 7 shows a rapid increase in runtime as tag density increases. When the tag density is too high, the loss in runtime is too significant compared to the gain in IE. There is a need to find a threshold in the tag density at which a compromise is reached in IE and runtime.

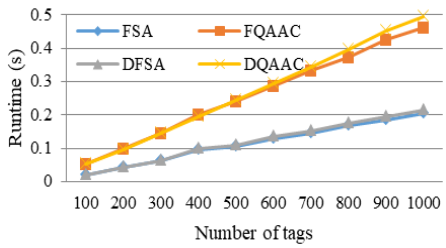
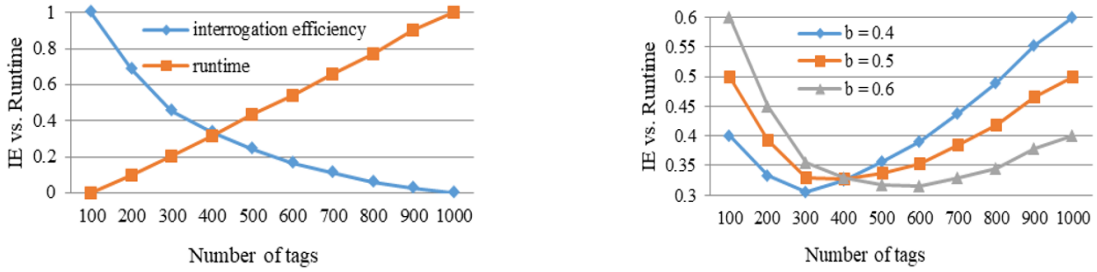


Figure 7. Runtime increases rapidly as tag density increases.

Let β be the compromised weight, $\beta \in [0, 1]$, and IE and runtime are normalized to the interval [0, 1], Figure 8-a) shows the graph of the normalized IE and runtime where the best compromise is achieved when the threshold in the number of tags is 40, corresponding to β=0.5. A survey of different compromised weights β is performed in Figure 8-b), showing that the threshold in the number of tags changes as the compromised weights change. Thus, the IE of the two algorithms, FQAAC and DQAAC, is only good when the number of tags is smaller than the threshold in tag density.

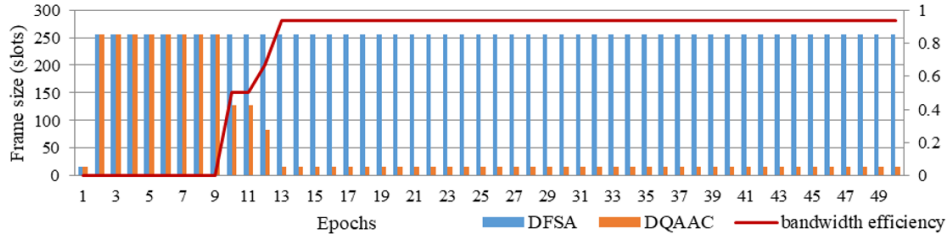


a) There exists a compromise point between interrogation efficiency and runtime. b) The compromise point found is at 400 tags with different compromised weights.

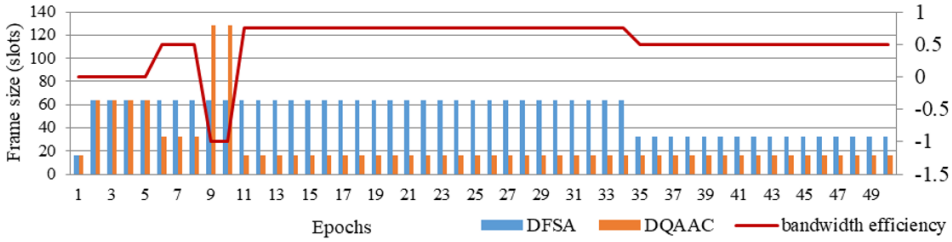
Figure 8. There exists a threshold in tag density where IE and runtime reach a compromise.

5.3. Bandwidth Efficiency

Figure 9 depicts the frame size adjusted over interrogation rounds (epochs), with a dense tag density (Figure 9-a)) and a sparse tag density (Figure 9-b)). With DFSA, due to the limit on the maximum frame size [3], the frame size still is at most the maximum threshold 256 even though the tag density increases high. However, with DQAAC, the learning algorithm helps adjust the frame size more appropriately. After several iterations with a frame size of 256, DQAAC reduces the frame size to 16 (Figure 9-a)). In the case of sparse tag density (Figure 9-b)), although the reduction in frame size of DQAAC compared to DFSA is less, DQAAC always maintains the frame size at a minimum level (16). This means that DQAAC achieves better BE than DFSA.



a) Frame size changes over epochs with sparse tag density.



b) Frame size changes over epochs with dense tag density.

Figure 9. The frame size is adjusted over interrogation rounds, where DQAAC achieves better BE than DFSA.

5.4. Discussion

Integrating Q-learning into the Aloha anti-collision protocol has significantly improved the efficiency of tag interrogation. However, the integration also complicates the processing (e.g., for the reward function and Q-value calculation). It increases the storage space (e.g., for the Q-table) and the data exchange bandwidth (e.g., for the suggested tag access slots). Security issues have also been raised for the two integrated protocols, FQAAC and DQAAC, and the suggestions in [6] can help improve the security capabilities of these two integrated protocols.

6. Conclusions

Aloha is one of the popular protocols used in natural RFID systems, where collision is a critical problem when deploying the Aloha protocol in practice. Several solutions are proposed that have only achieved a certain level of efficiency. The paper proposes a collision reduction approach based on Q-learning, in which two improvements to Q-learning are proposed to increase IE and reduce energy consumption. Specifically, FQAAC and DQAAC are two improvements of BFSA and DFSA that help determine the best slot based on a reinforcement learning process. Knowledge is learned

from the environment through identification results as collision, success or idleness. DQAAC differs from FQAAC in that it has a more flexible frame adjustment. Simulation results show that FQAAC and DQAAC achieve better interrogation and energy efficiencies than BFSA and DFSA. However, the difference in efficiency gradually decreases as tag density increases. Furthermore, runtimes of FQAAC and DQAAC are significant when tag density is high. A compromise between IE and runtime was also analyzed, which showed a compromise threshold in the number of tags corresponding to the compromise weight β, at which FQAAC and DQAAC have IE and EE can be achieved.

Acknowledgment

The authors thank The Ministry of Education and Training, Vietnam, for supporting and funding this project (Code B2023-DHH-18).

References

[1] Abdulghafor R., Turaev S., Almohamedh H., Alabdan R., Almutairi B., and Almutairi A., "Recent Advances in Passive UHF-RFID Tag Antenna Design for Improved Read Range in Product Packaging Applications: A Comprehensive Review," *IEEE Access*, vol. 9, pp. 63611-63635, 2021. DOI:10.1109/ACCESS.2021.3074339

[2] Acik S., Kosunalp S., Tabakcioglu M., and Iliev T., "Improving the Performance of ALOHA with Internet of Things Using Reinforcement Learning," *Electronics*, vol. 12, no. 17, pp. 1-15, 2023. DOI:10.3390/electronics12173550

[3] Cha J. and Kim J., "Dynamic Framed Slotted ALOHA Algorithms Using Fast Tag Estimation Method for RFID System," *in Proceedings of the 3rd IEEE Consumer Communications and Networking Conference*, Las Vegas, pp. 768-772, 2006. DOI:10.1109/CCNC.2006.1593143

[4] Chu Y., Mitchell P., and Grace D., "ALOHA and Q-Learning Based Medium Access Control for Wireless Sensor Networks," *in Proceedings of the International Symposium on Wireless Communication Systems*, Paris, pp. 511-515, 2012. DOI:10.1109/ISWCS.2012.6328420

[5] Elbasani E., Siriporn P., and Choi J., *A Survey on RFID in Industry 4.0, Design, Challenges and Solutions*, Springer, 2020. https://link.springer.com/chapter/10.1007/978-3-030-32530-5_1

[6] Farzaneh Y., Azizi M., Dehkordi M., and Mirghadri A., "Vulnerability Analysis of two Ultra Lightweight RFID Authentication Protocols," *The International Arab Journal of Information Technology*, vol. 12, no. 4, pp. 340-345, 2015. https://iajit.org/PDF/vol.12,no.4/5708.pdf

[7] Jang B., Kim M., Harerimana G., and Kim J., "Q-Learning Algorithms: A Comprehensive Classification and Applications," *IEEE Access*, vol. 7, pp. 133653-133667, 2019. DOI:10.1109/ACCESS.2019.2941229

[8] Landaluce H., Perallos A., Onieva E., Arjona L., and Bengtsson L., "An Energy and Identification Time Decreasing Procedure for Memoryless RFID Tag Anticollision Protocols," *IEEE Transactions on Wireless Communications*, vol. 15, no. 6, pp. 4234-4247, 2016. https://ieeexplore.ieee.org/document/7425263

[9] Loganathan M., Sabapathy T., Elshaikh M., Osman M., Abd Rahim R., Jusoh M., Jais M., and Ahmad B., "Reinforcement Learning Based Anti-Collision Algorithm for RFID Systems," *International Journal of Computing*, vol. 18, no. 2, pp. 155-168, 2019. https://pdfs.semanticscholar.org/c192/08b7120fb037fdefe6c72dab55135a5dcc5b.pdf

[10] Loganathan M., Sabapathy T., Elshaikh M., Osman M., and Abd Rahim R., "Energy Efficient Anti-Collision Algorithm for the RFID Networks," *Bulletin of Electrical Engineering and Informatics*, vol. 8, no. 2, pp. 622-629, 2019. DOI:10.11591/eei.v8i2.1427

[11] Park W., Jang J., Kang S., Song C., Kim S., and Kim J., "Trading Card Game Exploiting RFID and 3D Graphic," *in Proceedings of the 3rd International Conference on Human-Centric Computing*, Cebu, pp. 3-8, 2010. DOI:10.1109/HUMANCOM.2010.5563357

[12] Vogt H., "Efficient Object Identification with Passive RFID Tags," *in Proceedings of the 1st International Conference* on *Pervasive Computing*, Zurich, pp. 98-113, 2002. DOI:10.1007/3-540-45866-2_9

[13] Wang L., Luo Z., Guo R., and Li Y., "A Review of Tags Anti-Collision Identification Methods Used in RFID Technology," *Electronics*, vol. 12, no. 17, pp. 1-36, 2023. DOI:10.3390/electronics12173644

[14] Xu J. and Yang S., "Research on the RFID Anti-Collision Algorithms Based on Q-Learning Algorithm," *in Proceedings of the International Conference on Computer Science and Information Processing*, Xi'an, pp. 695-698, 2012. DOI:10.1109/CSIP.2012.6308949

[15] Yau K., Goh H., Chieng D., and Kwong K., "Application of Reinforcement Learning to Wireless Sensor Networks: Models and Algorithms," *Computing*, vol. 97, no. 11, pp. 1045-1075, 2015. DOI:10.1007/s00607-014-0438-1

[16] Zhang J., Periaswamy S., Mao S., and Patton J., "Standards for Passive UHF RFID," *GetMobile: Mobile Computing and Communications*, vol. 23,

no. 3, pp. 10-15, 2020. DOI:10.1145/3379092.3379098

[17] Zhang Y. and Zhao D., "A New Dynamic Frame Slotted ALOHA-Algorithm for Anti-Collision in RFID Systems," *in Proceedings of the China-Japan Joint Microwave Conference*, Shanghai, pp. 502-504, 2008. https://ieeexplore.ieee.org/document/4772479

[18] Zheng F. and Kaiser T., "Adaptive Aloha Anti-Collision Algorithms for RFID Systems," *EURASIP Journal on Embedded Systems*, vol. 2016, no. 7, pp 1-14, 2016. DOI:10.1186/s13639-016-0029-7

Van-Hoa Le received his Ph.D. in Computer Science from Hue University, Vietnam, in 2020. He is currently a Lecturer at Hue University, Vietnam. His research interests include Optical Packet/Burst-Based Switching Networks, Mobile RFID/Sensor Systems, Fairness, Quality of Service, Smart Tourism, Neural Networks, and Soft Computing.

Duc-Nhat-Quang Nguyen received a Master's Degree in Computer Science and Information Engineering at National Cheng Kung University, Taiwan, in 2020. Currently, he is a Lecturer at the University of Sciences, Hue University, Vietnam. His research interests include Digital IC Design, Artificial Intelligence, Internet of Things, RFID System.

Viet-Minh-Nhat Vo received his Ph.D. in Cognitive Informatics from the University of Quebec in Montreal, Canada, in 2007. He is currently an Associate Professor at Hue University, Vietnam. His research interests include Optical Packet/Burst-Based Switching Networks, Mobile RFID/Sensor Systems, Soft Computing, Neural networks, and Evolutionary Computation.