RESEARCH ARTICLE

Efficient Tag Grouping RFID Anti-Collision Algorithm for Internet of Things Applications Based on Improved K-Means Clustering

NNAMDI HENRY UMELO[1,2], **(Member, IEEE),**
NOR KAMARIAH NOORDIN[1,2], **(Senior Member, IEEE),**
MOHD FADLEE A. RASID[1,2], **(Member, IEEE), KIM GEOK TAN**[3], **(Member, IEEE),**
AND FAZIRULHISYAM HASHIM[1,2], **(Member, IEEE)**

[1]Department of Computer and Communication Systems Engineering, Faculty of Engineering, University Putra Malaysia (UPM), Serdang, Selangor Darul Ehsan 43400, Malaysia
[2]Wireless and Photonic Networks Research Centre of Excellence (WiPNET), Faculty of Engineering and Technology, University Putra Malaysia (UPM), Serdang, Selangor Darul Ehsan 43400, Malaysia
[3]Faculty of Engineering and Technology, Multimedia University, Bukit Beruang, Melaka 75450, Malaysia

Corresponding authors: Kim Geok Tan (kgtan@mmu.edu.my) and Nor Kamariah Noordin (nknordin@upm.edu.my)

ABSTRACT Dynamic Frame Slotted ALOHA (DFSA) is a de facto algorithm in the EPC Global Class-1 Generation-2 protocol for Radio Frequency Identification (RFID) tag collision problem. DFSA fails when the UHF RFID tag deployment becomes dense like in Internet of Things (IoT). Existing works do not provide readers prior tag estimates. Most algorithms assume a collision slot means two tag collision. But in dense IoT applications, much more than two tags can constitute a collision slot. Moreover, research proves collision slot might occur due to other reasons such as error-prone channel. This paper proposes a RFID anti-collision algorithm, kg-DFSA that equips the reader with prior information on accurate tag estimate. In kg-DFSA, tag identification is divided into two stages – initialization and identification. In the initialization stage, the reader uses improved K-means clustering running concurrently with a tag counter algorithm to cluster tags into K groups using tags' RN16 while the counter returns an accurate tag number estimate. In the identification stage, the tags are read only in frame chunks that match their group IDs while a new frame size look up table is developed to boost efficiency. Variants of the proposed kg-DFSA, traditional DFSA and another grouping based DFSA algorithm (FCM-DFSA) were implemented in MATLAB. Extensive Monte Carlo simulation shows the proposed kg-DFSA edges DFSA in terms of success rate 50%, system efficiency 65% and identification time 28%. The proposed model is useful in enhancing the existing MAC protocol to support dense IoT deployment of RFID.

INDEX TERMS Dynamic frame slotted aloha, Internet of Things, k-means clustering, RFID anti-collision algorithm, tag grouping, frame size table.

I. INTRODUCTION

Radio Frequency Identification (RFID) has become one inevitable enabling technology for the realization of the vision of Internet of Things (IoT) [1], [2], [3], [4]. Recently, passive ultra-high frequency (UHF) RFID has gained much popularity and applications as it has become a sine qua non enabling technology for the IoT. Passive UHF RFID (unlike active and semi-passive RFIDs), because their tags do not require batteries, are very cheap to be deployed on a billion scale. Hence, they are suitable for the promise of IoT of tagging and making just anything in the world to be wirelessly identifiable [1], [5], [6] and data exchangeable. Because of its low cost and low energy consumption, passive RFIDs are so

The associate editor coordinating the review of this manuscript and approving it for publication was Lei Shu.

popular that they are often regarded as RFID itself. Therefore in this paper RFID is often used in place of passive RFID. Passive RFID application areas ranges from pharmaceutical, agricultural, library, military, supply chain, highway toll collection, e-passports, healthcare, ware-house [7] etc.

A typical passive UHF RFID system consists of a reader (equipped with a transceiver, processor, memory and antenna), a backend host system and a set of passive tags. These tags use backscatter modulation to respond to reader queries while the reader read the EPC of the tags. Tag collision Problem (TCP) occurs when more than one tag try to respond to reader query at same time. This collision seriously affects the efficiency of the RFID system and efforts to mitigate it is called RFID tag anti-collision algorithm or protocol [8]. The EPC Class-1 Generation-2 protocol (EPC C1G2) [9] was standardized in 2005 to effectively mitigate tag collision of few tag density. EPC C1G2 which is dynamic frame slotted ALOHA (DFSA) based is a global industry standard for passive UHF RFIDs operating within the frequency range of 860 MHz to 960 MHz. However, when the tag density becomes much like in IoT scenario EPC CIG2 fails woefully [4]. In this paper dense RFID environment is used as defined by [8] which is 100 to 1000 tags per reader. Dense tag environment in RFID research is when RFID tagged objects are too many within a given area in such a manner that tag collision becomes inevitable.

Enhancing DFSA so as to be applicable to IoT has become a very important research area [4], [5], [10]. Approaches in literature towards addressing TCP are categorized into: Deterministic (Tree-based) approach [11], [12], [13], [14], Probabilistic (ALOHA-based) approach [3], [5], [15], [16], [17], [18], [19], [20], [21], [22] and Hybrid protocols [23]. Unlike Tree-based (deterministic) protocols, DFSA (the latest variant of ALOHA) gives tags fairness in accessing the shared channel [24] and improving it will mean support for a very dense deployment of RFID [5]. Furthermore, DFSA has the advantage of dynamically adjusting its frame size based on the estimate of number of tags within its read range. DFSA uses a frame structure to read data from tags. A frame is a collection of times slots.

It follows that the frame size of a DFSA based protocol is a direct factor to the efficiency of such protocol [16]. When the frame size is too large, there will be too many idle slots and when it's too small, there will be too many collision slots. Both scenario does not favor the efficiency of the RFID system. Efficiency is maximized when success slots are maximized at the expense of both the collision slots and ideal slots. It is proven that the efficiency of an RFID system is optimal when the number of unidentified tags equals the frame size. Most works in literature either use collision slots [16] or combine the value of collision slots and idle slots in estimating the number of unidentified tags per query cycle. Whereas, the works of [14], [25], [26], [27] strongly suggest that not all collision slots are due to collision but might be due to an erroneous channel. More so, most of these works that use the collision slot for tag estimation
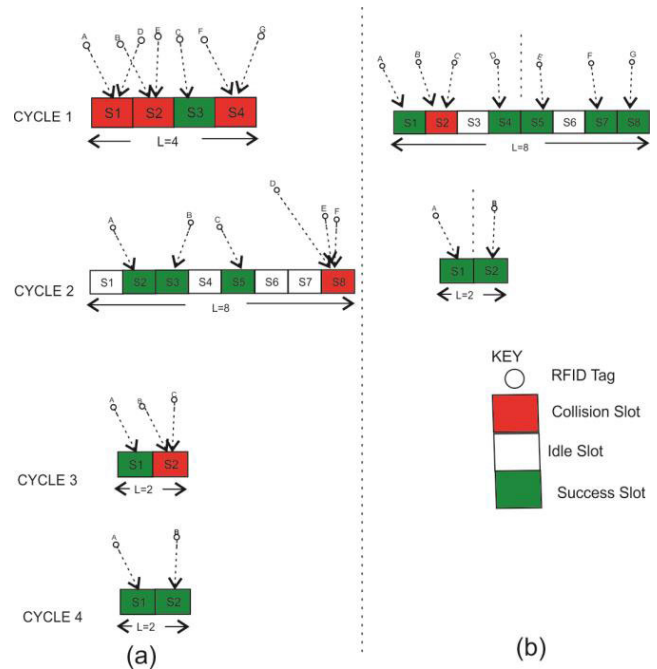

FIGURE 1. A illustrative example of the proposed scheme.

developed their algorithms based on the assumption that a collision slot means two tags have collided. However, in a dense tag deployment like IoT, a collision slot could mean much more than two tag collision.

Furthermore, in the current RFID anti-collision protocol and efforts in literature towards enhancing DFSA for RFID anti-collision, the reader does not have prior information about the number of unidentified tags within its read range [28]. The existing algorithms are such that after a query cycle (read round), the reader uses information about collision slots to estimate the unidentified tags and predict a frame size for the next query cycle. Recently, tag grouping is becoming an interesting method towards enhancing DFSA [3], [5], [6], [8], [17], [28]. Tag grouping guarantees the possibility that tags access the shared channel at the same time with minimal collision [3]. Hence, a good grouping technique should enhance the efficiency of DFSA algorithm.

To reduce the possibility of collision and enhance the efficiency of the DFSA-based RFID system for IoT scenario, this paper proposes an accurate tag number estimation and grouping scheme named K-means grouping based dynamic frame slotted Aloha (kg-DFSA). The proposed kg-DFSA algorithm is in conformity with EPC C1G2. Since in the current RFID system, information about tags are not known to the reader prior to tag reading, therefore, an anti-collision scheme that gives readers prior estimate of tags within their read range is developed. As demonstrated in Fig 1(b), given 7 tags to be identified, the reader is able to make a precise prediction of frame size of 8 slots, with virtual grouping of the tags into 2. This makes the proposed scheme utilize only 10 slots to identify all the 7 tags in just 2 cycles unlike the traditional DFSA (Fig 1(a) that used 16 slots in 4 cycles.

In this paper, tag grouping is considered a semi-supervised clustering problem while the unique EPC code (RN16) of the tags are used as basis for the clustering of tags. After grouping, each tag is assigned with a group ID, making it possible for tags of same group to randomly select numbers available only within their group into their counters for channel access. Hence, collision probability is minimized. Because of the limitations of the traditional K-means clustering of too much reliance on initial centroid [5], Euclidean distance measure as well as largest minimum distance were used to develop an improved K-means clustering algorithm that effectively groups tags.

This paper makes the following major contributions.

- It developed a new tag grouping model where tags are prefixed with group IDs apart from their unique EPC, aiming at minimizing the probability of tag collision while enhancing the probability of success slots in a dense RFID application;
- It developed a tag counter algorithm that runs concurrently with the tag grouping at the initialization stage, aiming at returning an accurate tag number estimate to the RFID reader ab initio before the tag identification stage starts. This aids the reader to predict a precise frame size. Thereby, minimizing the probability of both idle and collision slots while enhancing the efficiency of the high density RFID system;
- It proposes the integration of a frame size adjustment look up table that may cost a few kb of readers' memory but would enable the reader save time used in computing appropriate frame sizes per query cycle.
- Based on EPC C1G2 specifications, extensive simulation was conducted to evaluate the performance of our proposed kg-DFSA algorithm starting from a low density to a very high density network of tags, then compared with a variant of kg-DFSA, the de facto protocol (EPC C1G2) [9] and another grouping based algorithm FCM-DFSA [5]. Hence, our proposed algorithm is compatible with the EPC standard and does not require additional parameters or instructions.

The remainder of the paper is organized as follows. Section II gives a background to RFID TCP and exposes the challenges to DFSA-based EPC C1G2 protocol. Section III presents a new RFID anti-collision scheme. We described the framework of our pro-posed model and showed the improvement we did on K-means clustering as well as showed how our new algorithm is developed. Using simulation, section IV shows the performance evaluation of the proposed algorithm in comparison with tradition DFSA, kg-DFSA variant and another grouping based DFSA algorithm. Section V concludes the paper.

II. BACKGROUND TO RFID ANTI-COLLISION
This section details the problem this paper addresses which is TCP of RFID systems. The workings of the existing protocol was explored. We also discussed the challenges with the existing methods as well highlighted some state-of-the-art
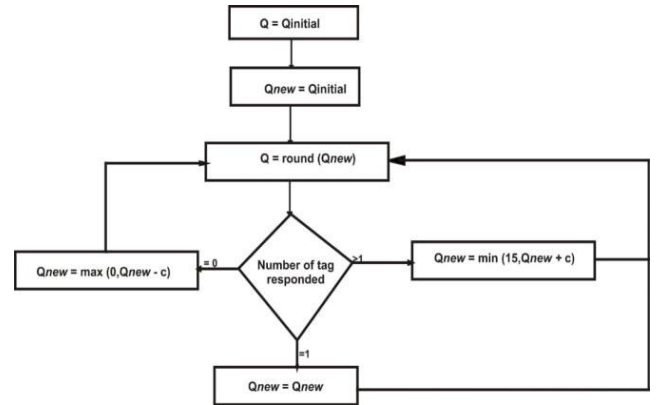

FIGURE 2. Q-algorithm of EPC C1G2.

methods in literature. Then an overview of the proposed method was done.

A. TAG COLLISION PROBLEM (TCP)
Basically there are three types of collision that can occur in a RFID system [23]. Reader-to-tag collision occurs when a tag receives an initial electromagnetic wave (query) from a given reader R1, and in trying to respond to the query from the reader R1, another reader R2 queries same tag causing signals collision and the tag responds to none of the queries. The second type of collision referred to as reader-to-reader collision happens when the electromagnetic wave from a particular reader R2 distorts reader R1 from successfully reading a given tag's data [29]. The last which seem to be most common type of collision in UHF RFID systems is the tag-to-tag collision. Tag-to-tag collision occurs when two or more tags signals in a bid to reflect a wave query from a reader, collide before their signals is read by the reader.

The tag-to-tag type of collision is called Tag Collision Problem (TCP) of RFID systems [23] and efforts to mitigate it is referred to as Tag Anti-collision MAC Protocol. EPC C1G2 [9] is the existing protocol from EPCglobal that addresses TCP. However, current research is towards enhancing EPC C1G2 to be applicable to IoT [5], [24].

B. EPC C1G2 PROTOCOL
The EPC C1G2 is the existing protocol for passive RFID systems working within the UHF of 860MHz and 960MHz. This is based on DFSA and a Q algorithm (described in Fig. 2) used for frame size adjustment per query cycle. The following represents the procedure of EPC C1G2:

The reader initiates a Query command that consists of Q parameters representing the frame size which ranges between [0, 15];

The tags have random number generator as well as a slot counter. The tags randomly generate numbers between $[0, 2^Q - 1]$ and store same in their slot counter;

The random number that each tag generates is the particular slot with which the tag would reply to reader query;

The tag(s) with random number zero immediately replies the query with its 16-bit random number (RN16);

The random number $[0, 2^Q - 1]$ represents access probability to the frame which represents the communication channel.

Other tags back off and decrease their counter by 1 while waiting for their turn;

Depending on the random number generated by the tags, further querying will result to either success slot, idle slot or collision slot;

The reader uses an ACK command to acknowledge the tag in success slot while the tag replies with its EPC;

The reader responds with a QueryRep to use the same Q parameter and frame size for the next query cycle or a QueryAdjust to change Q and the frame size as in Fig. 2;

With the QueryAdjust command and another C parameter which ranges from [0.1, 0.5], the reader changes the Q value depending if there is collision slot or idle slot. As seen in flowchart of Q-algorithm in Fig. 2, the C value is added to the Q, when there is collision slot and subtracted from the Q, when there is idle slot, else the Q value remains the same and goes with the QueryRep.

C. CHALLENGES TO DFSA

The advantage DFSA has over the Tree or deterministic approach in RFID anti-collision effort is the fact that DFSA guarantees equal opportunities (fairness) to tags in the random access of the shared channel [24]. This advantage makes DFSA the protocol of choice for IoT application of RFID.

However, DFSA as implemented in EPC C1G2 has a hand full of challenges like poor tag number estimation, a bogus frame size adjustment policy as well as channel error issues [24]. The focus of this paper which seem to be the biggest of all the challenges is on poor tag number estimation which has a direct relationship with the frame size the reader predicts for the next query cycle. Too large frame size means too many idle slots while too little frame size means too many collision slots. Both situations cause degraded efficiency of the RFID system [18], [19], [30].

Applying RFID to IoT demands dense deployment of tags. Hence, an accurate tag number estimation algorithm is imperative [6], [17]. In the existing tag number estimation algorithm, the RFID reader is not equipped with prior information of tag number before the actual tag identification but relies on broadcasting queries and checking the value of slots. Most estimation algorithms assume a collision slot means two tags have collided. Unfortunately, protocols developed for IoT with such assumption could cause a total system breakdown as collision could mean much more than two tags replied the reader using same time slot.

D. RELATED WORKS

The traditional DFSA algorithm is designed based on a lower bound shown in (1). In the traditional DFSA, collision slot is assumed to imply that 2 tags have collided and after each query round the algorithm computes the unidentified tags estimate by removing the value success slots while

multiplying the value of collision slots by 2.

$$\hat{n} = S_s + 2S_c \tag{1}$$

where $\hat{n}$ is tag estimate, S_s is success slot, S_i is idle slot and S_c is collision slot.

Consequently, a few authors proposed an improvement to the lower bound using a method referred to as Schoute estimation method [31]. In Schoute's method, there is an assumption that the number of tags that choose a given times slot (garbled slot) is given by the posteriori distribution probability:

$$\sum_{k=2}^{\infty} k \cdot \rho_k = \frac{\sum_{k=2}^{\infty} e^{-1} \frac{1}{(k-1)!}}{1 - 2e^{-1}} = \frac{e-1}{e-2} \approx 2.39 \tag{2}$$

where k is the collision slot.

Therefore, with (2) Schoute [31] was able to give an estimate that is based on a collision slot s_c value of 2.39. Hence, Schoute gives a better estimation algorithm than the lower bound. Schoute estimate is expressed in (3). Whereas, lower bound and Schoute are very simple and easy to implement, their high estimation error becomes a huge drawback when a high density of tags are to be identified [20].

$$n = s_s + 2.39 s_c \tag{3}$$

Another popular estimation method was given by Vogt [32]. Vogt proposed two methods DFSAVI and DFSAVII. In DFSAVI, Vogt first assumed that all collision means two tags have collided and gave a tag estimation formula that is same as the lower bound equation in (1). Afterwards, Vogt discovered that estimation error became too much with increase in tag density using DFSAVI. Hence, he used Chebyshev's inequality to develop another tag estimation algorithm called DFSAVII. Vogt's second algorithm DFSAVII computes the difference between an actual read result vector $< S_s, S_i, S_c >$ and a theoretical expected result vector $< E[S_s], E[S_i], [S_c] >$. DFSAVII gave a higher performance than DFSAVI given high tag density. DFSAVII is expressed in equation (4)

$$\hat{n}_{vogt} = \min_n \left| \begin{pmatrix} E[s_i] \\ E[s_s] \\ E[s_c] \end{pmatrix} - \begin{pmatrix} s_i \\ s_s \\ s_c \end{pmatrix} \right| \tag{4}$$

Another estimation method was proposed by Chen [22]. Chen derived a method of determining the probabilities for success slot, idle slot and collision slots (S_s, S_i, and S_c). Chen modelled S_s, S_i, and S_c as a multinomial with frame size L independent trials.

$$\rho(S_s, S_i, S_c) = \frac{L!}{S_s! S_i! S_c!}$$

The decision rule of Chen is to use the tag estimates n that maximizes the probability $\rho(n|S_s, S_i, S_c)$.
where:

$$\rho(n \mid S_s, S_i, S_c) = \frac{L!}{S_s! S_i! S_c!}$$
$$\times [1 - \left(1 - \frac{1}{L}\right)^n - \frac{n}{L}(1 - \frac{1}{L})^{n-1}]^{S_c} \tag{5}$$

Since Chen did not consider that the values of frame size L per query cycle depends on the values of the slots and number of tags n, it's often regarded as impracticable [33].

Another method was proposed by Eom and Lee [21] that gives a better estimation result by properly determining the number of tags for any collision slot. They formulated the number of tags per collision slot with two types of equations - (6) which is the observed collision detection results for each query cycle and (7) for the expected values.

$$\beta_k = \frac{L}{\gamma_{k-1} . s_c + s_s} \tag{6}$$

$$\gamma_k = \frac{1 - e^{-\frac{1}{\beta_k}}}{\beta_{k-1} \left(1 - \left(1 + \frac{1}{\beta_k}\right) e^{-\frac{1}{\beta_k}}\right)} \tag{7}$$

where β_k = ratio of frame size to tag number after kth iteration and γ_k = number of tags per collision slots after kth iteration.

Eom and Lee further combines the two equations into an iterative algorithm and concludes that their corresponding convergence is the number of tags per collision slot. Then they multiplied the γ_k with the collision slot and add to number of successful slots to have an estimation for tag number as seen in (8). The Eom estimation method is given by:

$$\hat{n}_{Eom} = \gamma_k \cdot s_c + s_c \tag{8}$$

Eom estimation method is accurate when the frame size is large but, as the frame size decreases, its estimation accuracy decreases [18].

Most of the works on RFID tag number estimation in literature focus on estimating tag number using query cycles [21], [22], [31], [32], [34], [35] and the value of collision and idle slots, only a handful have tried to think outside the box. A few works that seem to estimate the number of tags before the querying process starts was done in 2018 by [5] and 2019 by [4]. In [5], fuzzy-C means clustering was used to group tags afterwards, DFSA was deployed to read tags. They effectively grouped the tags such that the efficiency of the RFID system is made to be dependent on grouping conditions. Their model is such that for every data point (tag), their algorithm computes the membership degree for all centroid since a tag could belong to more than one group, causing too much computational space. They used (9) in computing the fuzzy membership degree of each tag. Moreover, they employed Mahalonobis density function to achieve result which add further to the memory weight of the algorithm. Besides, as seen in their flowchart of Fig. 3, their tag estimation still used the value of collision slots CK and success slot CS per cycle. This reasons further places more doubt on its practicability in terms of efficiency and time delay for dense IoT application.

$$u_{ik} = \frac{1}{\sum\limits_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{m-1}}} \tag{9}$$
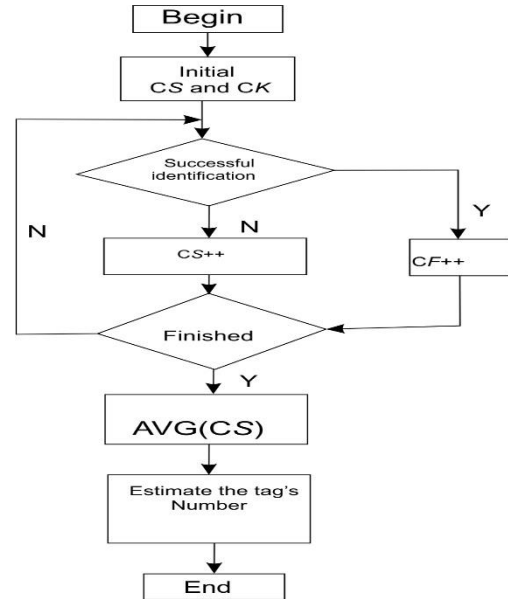

FIGURE 3. Flowchart of tag estimation in [5].

where m is the fuzzy idex of each tag, d_{ik} is the distance of the tags to the centroid.

Reference [4] developed their tag estimate using results from sub-frames after query cycles, while [6] did a performance evaluation of a few grouping based RFID anti-collision algorithms. Whereas [6] only analysed and evaluated existing grouping based algorithms without proffering a different method, [4] gave a tag estimate after query cycles. However, this paper develops a new algorithm that estimates tag number priori the actual tag reading. Thereby, giving the RFID reader the vantage of predicting a precise frame size right from the first reading cycle and possibly minimizing the number of query cycles needed to identify or read all tags.

E. OVERVIEW OF TRADITIONAL K-MEANS CLUSTERING
Clustering is the process of grouping a set of physical or virtual objects into clusters of similar objects [36]. It is the grouping or partitioning of data or data points into K distinct clusters based on their similarity using a distance measure. K-means clustering is an unsupervised machine learning tool and has been used in signal processing, data mining and pattern recognition to classify patterns using a measure of similarity or difference of such pattern [37], [38]. Recently, K-means have been used in electrical energy load balancing [39], students' result grouping so as to focus more attention on students with special needs [40] and in [41] another machine learning technique was used in RFID supply chain to detect false positive readings and filter out same so that it's not registered on a database. The K in the name represents the fact that the algorithm looks for a given number of clusters which are defined in terms of how close data points are to each other. Below are the six major steps involved in K-means clustering:

Step 1: choose the number of clusters, i.e. assign value for k

Step 2: Randomly select data points to be initial centroids for each cluster.

Step 3: Measure the distance between each data point to the centroids

Step 4: Assign each data point to a cluster that it is nearest to its centroid.

Step 5: Calculate the mean of each cluster and update same as the new centroid

Step 6: Repeat steps 3 – 5 with the new centroid until convergence (no more changes) or to a maximum number of iteration

K-means clustering is always done with the aid of a distance function and this paper uses K-means to achieve a clear task of tag grouping. Among clustering techniques, K-means has the advantage of easy implementation, less complexity, celerity and efficiency [37]. However, it has a weakness which is too much reliance on the initial clustering center [5]. Therefore, this paper improves K-means clustering by employing largest minimum distance together with Euclidian distance measure shown in (10). The X and Y represents the different tags (data points) whose distances are to be measured and clustered. While n is the tag number estimate.

$$dis(x.y) = \sqrt{\sum_{i=1}^{n}(x_1 - y_1)^2} \qquad (10)$$

III. METHODOLOGY
In this section we first did a theoretical analysis and a simulation validation of the efficiency of the RFID system. This is very important as it is the basis of the method this paper adopts. Then we described the basic idea in our new model, showed our improvement on K-means clustering and demonstrated how the new algorithm performs tag grouping and counting using tags' RN16 and finally described the enhancement we made on the existing frame size adjustment algorithm.

A. THEORETICAL ANALYSIS OF DFSA FOR TCP
DFSA as implemented in EPC C1G2 protocol can easily support a few tags. However, the efficiency of the RFID system degrades with increase in tag density [4]. Much of the reason for this degrade in efficiency has been proven to be because either the frame size is too large causing too many idle slots or the frame size is too small causing too many collision slots. The only determinant factor for what the frame size should be per query cycle is the estimated number of unidentified tags.

In projecting the estimated tag number n in relation to the frame size L the reader uses to read tags, we make the assumption that tags have equal opportunities in accessing the shared channel. Therefore by binomial distribution, the probability that there are q number of tags in a slot is given by:

$$p_q = \binom{n}{q}\left(\frac{1}{L}\right)^q\left(1 - \frac{1}{L}\right)^{n-q}$$

Therefore, the probability that there exist only one tag in a slot is expressed as:

$$p_1 = n\left(\frac{1}{L}\right)^q\left(1 - \frac{1}{L}\right)^{n-1}. \qquad (11)$$

Also the probability that there exist no tag in a slot is expressed as:

$$p_0 = \left(1 - \frac{1}{L}\right)^n. \qquad (12)$$

Consequently, given frame size as L, in a query cycle the number of success slots is expressed as:

$$S_s = L \times p_1 = n\left(1 - \frac{1}{L}\right)^{n-1}. \qquad (13)$$

While the number of idle slots are:

$$S_i = L \times p_0 = L\left(1 - \frac{1}{L}\right)^n. \qquad (14)$$

Hence, the number of collision slots is:

$$S_c = L - S_s - S_i$$
$$= L - n\left(1 - \frac{1}{L}\right)^{n-1} - L\left(1 - \frac{1}{L}\right)^n. \qquad (15)$$

Meanwhile efficiency of the RFID system can be computed as:

$$\eta = \frac{S_s}{L}. \qquad (16)$$

To maximize efficiency we calculate:

$$\frac{dS_s}{dL} = \left(1 - \frac{1}{L}\right)^{n-1} + n\left(1 - \frac{1}{L}\right)^{n-1}\ln\left(1 - \frac{1}{L}\right)$$
$$= \left(1 - \frac{1}{L}\right)^{n-1}\left[1 + n\ln\left(1 - \frac{1}{L}\right)\right] = 0. \qquad (17)$$

When n is known:

$$L = \frac{1}{1 - e^{\frac{1}{n}}} = \frac{e^{\frac{1}{n}}}{e^{\frac{1}{n}} - 1}. \qquad (18)$$

And when the value of n is very large:

$$L = \frac{1 + \frac{1}{n}}{1 + \frac{1}{n} - 1} = n + 1. \qquad (19)$$

Theoretically, it is proven that the closer the frame size is to the estimated tag number, the higher the efficiency of the RFID system. Furthermore, we performed a simulation validation of this fact using MATLAB software. Different frame size L was used in a Monte Carlo iteration for varying density of tags ranging from zero then 50 and to 300. Fig. 4 shows an efficiency result obtained from the simulation and it is clearly seen that for given frame sizes L, efficiency of the RFID system is maximum when the number of tags were nearest or approximately equal to the frame sizes. The validation was also necessary to test the accuracy of our simulation tools.
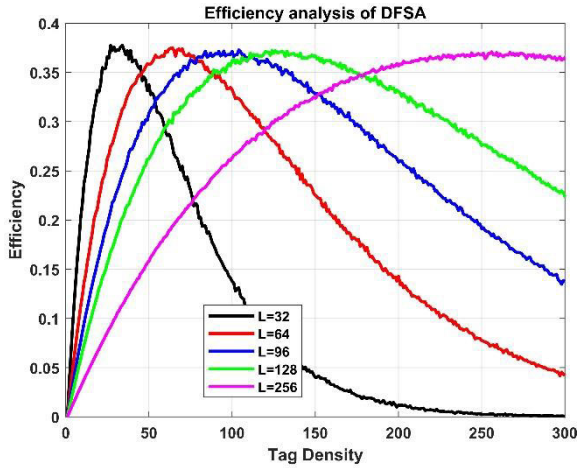
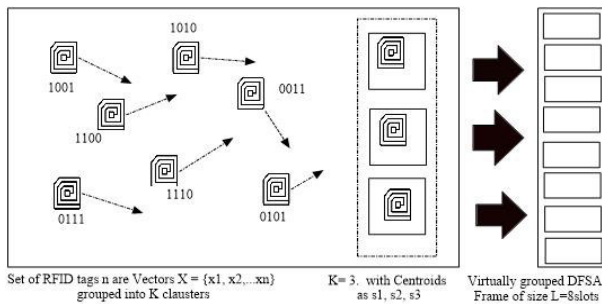**FIGURE 4.** Efficiency analysis of DFSA for RFID systems.

FIGURE 5. A scenario of RFID tag identification using the proposed model where tags only access the channel (frame) using group ID.

B. SYSTEM MODEL

In the proposed model, the tag identification effort is divided into two – initialization and the identification stages. In the initialization stage, the reader uses queries to group and count tags before passing control to the identification stage. We assume that all RFID tags within the identification range of the reader $\{x_1, x_2, \ldots x_n\}$ must be identified and to identify each and every tag must belong to a group $\{1, 2, \ldots k\}$ and have a group ID, using tags RN16. The K-means function with the aid of its distance measure, performs series of distance measure and changing of centroid of each group till the algorithm reaches a convergence when all the tags have their groups, evenly distributed and are ready to respond to reader queries with their EPC and group ID so as to be identified in the next stage.

The idea shown in Fig. 5 paints a scenario where tags' signals given their random nature could collide in attempting to respond to reader queries (reader frame represented in an 8 slot size). However, using their binary RN16 (represented in 4 bit binary numbers), the tags can be clustered into groups and given their group IDs such that the random process becomes a bit orderly and without unfairness to any tag. The grouping problem is to effectively label all RFID tags in the same group with same prefix ID. Tags with same group ID

randomly access the channel using the chunk of slots (within the DFSA frame) assigned to their group. It should also be noted that whereas the RN16 of RFID tags are unique binary numbers, this paper uses a binary to decimal converter to assign each RFID tag, a unique integer value for its RN16 as well as its group number both of which are the basis for the clustering. This is different from [2] where integers were assigned to already identified tags separating them from the unidentified tags.

At the identification stage, the unidentified tag number estimate n_{uid} is updated using

$$n_{uid} = n_{est} - n_s \qquad (20)$$

where n_{est} is the tag number estimate derived from the initialization stage which afterwards, is replaced by the n_{uid} identification stage and n_s expressed in (21) is number of successfully read tags after the current query cycle.

$$n_s = \sum_{i=1}^{n_{uid}} S_s \qquad (21)$$

where S_s is success slot

C. RFID TAG GROUPING USING IMPROVED K-MEANS CLUSTERING

In order to overcome the limitation of traditional K-means too much reliance on initial cluster centre (centroid) [5], [38], we propose a distance measure (Euclidean) as in (10), that is not only less complex but also because tag reading is a wireless communication task. Euclidean measure needs no outliers like Manhattan distance measure. More so, in selecting the centroid, we employ largest minimum distance which is proven to enhance the speed, stability and precision of clustering [37].

In this study, we consider tag grouping a semi-supervised learning problem. Semi-supervised because whereas the tags' RN16 are labelled, their group ID are not. Hence, given n number of RFID tags $\{x_1, x_2, \ldots, x_n\}$, within the read range of the RFID reader which are waiting to be identified. They can be grouped into K clusters using the following steps:

Step 1: the reader randomly chooses one among $\{x_1, x_2, \ldots, x_n\}$, to be the initial centroid (focal point)z_1, for instance we take $z_1 = x_1$

Step 2: for the second cluster, the reader chooses a datapoint which is far from z_1, then compute the distance of each tag and z_1

$$\|x_i - z_1\|, \quad i = 1, 2, \ldots, n$$
If :
$$\|x_j - z_1\| = \max \{\|x_i - z_1\|,$$
$$i = 1, 2, \ldots, n\}, \quad j = 1, 2, \ldots, n) \qquad (22)$$

We also take x_j as the centroid of cluster two $z_2 = x_j$

Step 3: We compute the distance of each tag $\{x_1, x_2, \ldots, x_n\}$ to the centroids $\{z_1, z_2,\}$ one after the other

$$d_{i1} = \|x_i - z_1\|, i = 1, 2, \ldots, n \qquad (23)$$
$$d_{i2} = \|x_i - z_2\|, i = 1, 2, \ldots, n \qquad (24)$$

Select the minimum of the two distances:

$$\min \quad (d_{i1}, d_{i2}), \quad i = 1, 2, \ldots, n$$

Step 4: We calculate the minimums of the distances of all the tags to the centroids. Then we select the maximum of all the minimums to be our third centroid z_3

If:

$$\min(d_{i1}, d_{i2}) = \text{Max} \{\text{Min}(d_{i1}, d_{i2}),$$
$$i = 1, 2, \ldots, n\}, \quad j = 1, 2, \ldots, n \quad (25)$$

Then:

$$z_3 = x_j \quad (26)$$

Assuming that we have r ($r < K$) centroids $\{z_1, i = 1, 2, \ldots, r\}$, then we need to get the $r + 1$th centroid.

Therefore if:

$\min (d_{j1}, d_{j2}, \ldots, d_{jr}) = \max \{ \min(d_{i1}, d_{i2}, \ldots, d_{ir}), i = 1, 2, \ldots, n\} j = 1, 2, \ldots, n$

then:

$$z_{r+1} = x_j$$

Step 5: Repeat, till $r + 1 = K$

Step 6: Unlike in traditional K-means clustering, initial centroids K is now known which is $z_1, z_2, \ldots, z_k$

Step 7: Going by the rule of minimizing distance, we assign tags $x_1, x_2, \ldots, x_n\}$ to one of the K clusters,

That is, if:

$$\|x - z_j(t)\| = \min \{\|x_j - z_1\|,$$
$$i = 1, 2, \ldots, K\}, j = 1, 2, \ldots, K \quad (27)$$

then:

$x \in s_j(t)$.

Where t is the number of iteration, s_j is the jth cluster and z_j is the centroid

Step 8: Calculate the values of each centroid:

$z_j(t + 1), j = 1, 2, \ldots, K$

Compute for the mean of the tags in each cluster:

$$z_j(t + 1) = \frac{1}{n_j} \sum_{x \in s_j(t)} x, j = 1, 2, \ldots, K \quad (28)$$

where n_j is the number of samples of jth cluster s_j.

Then compute the mean of tags in K clusters respectively.

Using the new means to be the new centroid would minimize the cluster criterion function J_j.

$$J_j = \sum_{x \in s_j(t)} \|x - z_j(t + 1)\|^2, \quad j = 1, 2, \ldots, K \quad (29)$$

Step 9: If: $z_j(t + 1) \neq z_j(t), j = 1, 2, \ldots, K$

then, GOTO step 7, group tags and keep repeating the grouping.

If: $z_j(t + 1) = z_j(t), j = 1, 2, \ldots, K$,

then, the algorithm reaches its convergence

TABLE 1. RFID EPC coding scheme.

EPC 96 bit ID	Version number	Domain management	Object class	Serial number
No. of bits	8	28	24	36

D. BINARY TO DECIMAL CONVERSION OF DATA

EPC is a unique ID of all RFID tags. Just like barcodes, they are universally used to uniquely identify physical object in the world. This is another reason RFID is seen as inevitable to achieving the dreams of IoT, which is to enable just anything to have connectivity. RFID unique ID (EPC) is 96-bits and is accessible in the cloud via EPCIS (EPC information service) [42]. Therefore, RFID tagged things are uniquely identifiable and when interfaced with say 5G network can achieve IoT's dream. As can be seen in Table 1, EPC is long [42] and because of space limitations, this paper utilizes only 16-bits out of the 96-bits for analysis and evaluation. As shown in Table 1, the EPC code has four parts. The first 8 bits is used to show the version of RFID tag while in the domain management, the manufacturer identifies each tag uniquely from the organizational end. The object class of 24-bit is used by the EPC management to specify the type of item while the 36-bit serial number is distinct within each object class.

This unique serial number forms the basis for our tag grouping in this paper as 16-bit binary data are generated randomly in MATLAB to simulate the tags. In our study, a binary to decimal conversion of the data was implemented as a MATLAB function that returns the decimal equivalent of each tag EPC to the clustering algorithm. For the purpose of clarity and space limitations, and considering thousands of tags need to be identified, the binary to decimal converter was introduced that converts each 16-bit binary RN16 to decimal. Hence, aids fast grouping of tags. Algorithm 1 shows the binary to decimal algorithm used to reduce the size of RN16.

Algorithm 1 To Convert Tags' Binary RN16 to Decimal

1: Receive tag's RN16 n

% Initialize variables i, int dec_equiv

      i = 0, dec_equiv = 0

2: **While** $n \neq 0$, i = 1 to n

3:     remainder = **rem** $(n, 10)$

4:     $n = n / 10$

5:     dec_equiv = dec_equiv + (remainder x 2^i)

6: **Return** dec_equiv

E. DESCRIPTION OF THE GROUPING AND COUNTING ALGORITHM (KG-DFSA)

The proposed kg-DFSA algorithm features the initialization and identification stages. In the initialization stage, a tag counter is defined. The counter increments and returns an accurate estimate of tags as the reader queries the tags and groups them based on their RN16 using an improved k-means

clustering algorithm written as a MATLAB function. Consequently, each tag has a group ID as seen in Algorithm 2. Afterwards, in the identification phase, DFSA frame L is virtually divided into chunks (few collection of timeslots) depending on the tag estimate returned by the counter. Tags with their group IDs contend for channel access allocated to their group (chunk) by randomly selecting a number from $[0, 2^Q - 1]$ and save in their slot counters. The tag with slot counter zero uses the channel to reply to the reader query. The result of which is either success slot S_s, idle slot S_i or collision slot S_c. As seen in Algorithm 2 all the querying performed by the reader in the initialization stage (described in flowchart of Fig. 6) are meant to handshake, count, group and prefix each tag with a group ID. The identification stage of our kg-DFSA in Fig. 6 shows once the grouping and counting is done, the reader uses a precise frame size to swiftly identify all tags in groups with less query cycle and fewer slots in the identification stage. It is also noteworthy that the initialization stage is performed by the reader only once when it's up.

Algorithm 2 RFID Reader Performing Tag Grouping Using Improved k-Means Clustering

---

1: Initialize L, k, n, z
2: **while** $x > 0$ **do**
3: Broadcast Query with $L = 2^Q$
4: **For** $k = 1$ **to** n
4:   randomly select $z_1, z_1 \in (x_1 x_2 \ldots .x_n)$
5:   Receive tag RN16 of x
6:   Pass to Binary to decimal function
8:   Receive dec_equiv of x RN16
6:   compute $dist_1 = ||x - z_1||$ with (10)
7: **Next** k
8: **if** $x < dist_1$, **then** group_ID $= 01$
7: select $max||x - z_1||$
8:   $x = z_2$
8: **endif**
9: **For** $k = 1$ **to** n
5:   Receive tag RN16 of x
6:   compute $dist_2 = ||x - z_2||$
7: **Next** k
8: **if** $x < dist_2$, **then** group_ID $= 10$
7:   select $min||dist_1 - dist_2||$
8:   $x = z_3$
10:   **else** group_ID $= 11$
11:   **endif**
12: **endwhile**

F. SIMULTANEOUS GROUP READING STRATEGY

After the tag grouping have been done using improved k-means method, there is also need to show how the reader reads tags within each cluster group. After clustering each tag belongs to a particular group with a group ID. In implementing this, as shown in Fig. 7, we assume for instance there are 4 groups and assign "00", "01", "10" and "11" respectively
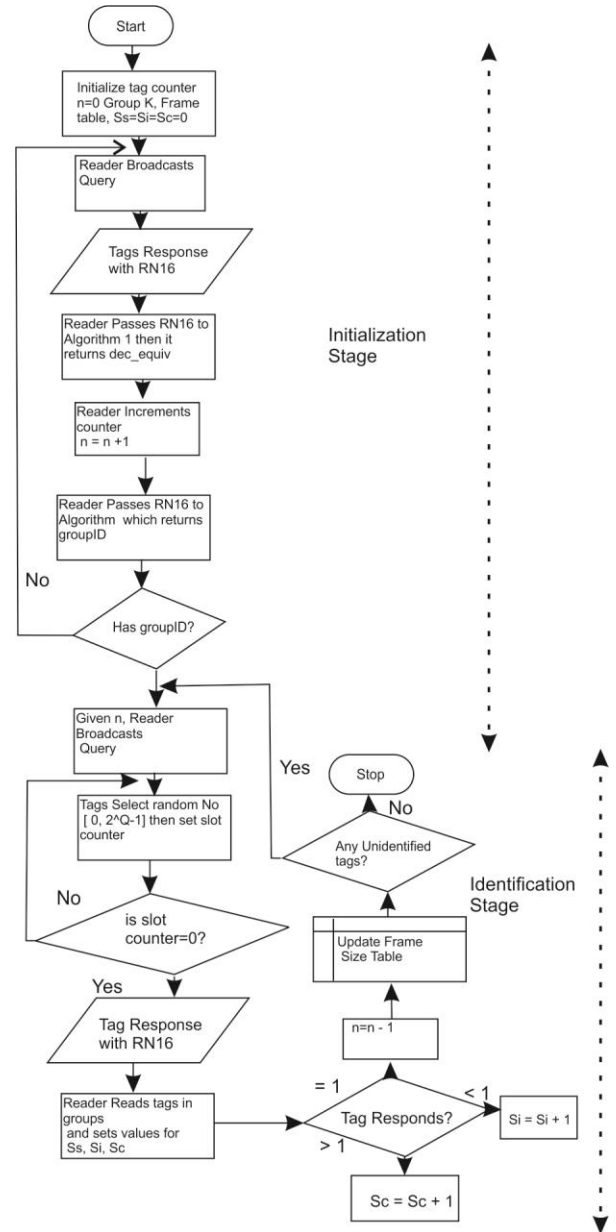

FIGURE 6. Flowchart describing Kg-DFSA grouping and counting of tags during initialization stage and reading same in identification stage.

as a prefix to each tag RN16 as the reader is forced to read tags with prefix "00" first. Afterwards the tags with prefixes "01" are read by the reader then the tags with prefix "10" and lastly the tags with prefixes "11".

In Fig. 7a, traditional DFSA method of reading tags is depicted where 20 tags randomly access the channel using a 16 slot frame. Fig. 7b on the other hand shows how in kg-DFSA, the 16 slot frame is divided virtually into chunks of 4 slots per group while each tag accesses the shared channel using its group ID of either "00", "01", "10" or "11". This is supposed to bring in some orderliness in the random process and ensure minimal query results of collision slots while maximizing success slots and system efficiency. Furthermore, since the quantity of tags in each group ("00,
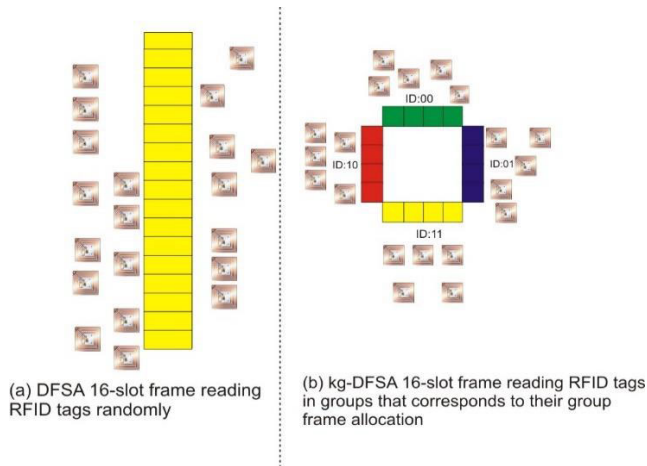
(a) DFSA 16-slot frame reading RFID tags randomly

(b) kg-DFSA 16-slot frame reading RFID tags in groups that corresponds to their group frame allocation

FIGURE 7. Group reading strategy of kg-DFSA against random reading in traditional DFSA.

01, 10 and 11'') differ and are the determinant of the number of time slots allocated to each group, the chunk of the frame size l allocated to each group from the available frame size L must differ and is expressed as follows:

$$\sum_{i=1}^{n} L_{min} \leq l \leq L_{max} \tag{30}$$

where n is quantity of tags

Therefore, with the prefix on each tag specifying its group ID and its uniqueness, the reader is able to simultaneously read tags within and beyond each group concurrently with minimal collision probability. The optimal grouping condition is when the tags in all the groups are approximately equal (fairness in tag grouping) as seen in Fig. 8 and the frame is allocated equivalently using equation (30). Therefore, the number of tags in each group does not necessarily matter. What matters is if each group has an equivalent allocation within the frame – let the group with more tags be given more slots while the group with the least number of tags be given the least number of slots.

G. NEW FRAMESIZE ALLOCATION AND UPDATING TABLE
The existing EPC C1G2 RFID protocol uses a method called Q-Algorithm (described in Fig. 2) to update the reader frame size per query cycle. The limitation of the current Q-Algorithm is not only that the reader is forced to update its frame size in multiples of two but also the reader uses a parameter C (ranges from 0.1 to 0.5) to adjust the Query frame which is always far from exact and makes a precise frame size per query cycle impossible.

To reduce time taken to achieve this and enhance system efficiency, the new kg-DFSA protocol uses tables to map and store each tag number estimate to the computed frame size per query cycle. This is done so that after the tag initialization stage of kg-DFSA protocol, the reader also pops up the previous frame size allocation table, making it accessible at the tag reading stage. And in the tag reading stage, given tag number estimate, the reader just looks up it table and selects the frame size that approximates to that tag number

estimate unlike a similar method used by [43] where they used table to store only tag number estimates. More so, their work used traditional DFSA for their experiment. In implementing this in MATLAB, an M-file is developed to store the tag number estimate n_{uid} and mapped against the predicted frame size L. We refer to this new frame size adjustment method as Q+ algorithm as it is an improvement over the existing Q-algorithm. Hence, whenever the RFID reader is up, all the M-files (DFSA, K-means clustering algorithm, Q+ algorithm) initializes and work together to attain kg-DFSA. This method only requires a few kb of the reader's memory as the expected numbers (tag number estimates and their mapped frame sizes) in the table are integers.

IV. PERFORMANCE EVALUATION/RESULTS AND DISCUSSION
The simulation experiment in this paper is carried out in two ways; one is to check the results of tag clustering (although clustering is only ''a means to an end'' in this paper), then the other which is the ultimate aim of this paper is to evaluate the performance of the proposed kg-DFSA algorithm. We vary the proposed algorithm into kg-DFSA-I (using traditional k-means clustering) and kg-DFSA-II (using improved k-means clustering). We also develop a fuzzy C-means clustering algorithm for tag grouping (FCM-DFSA) as in [5]. Reference [5] was used in comparing our proposed algorithm because the paper focuses on enhancing DFSA-based EPC C1G2 protocol by tag grouping which is also the focus of this paper, unlike [11] that focuses on Query Tree approach to RFID anti-collision. Besides, the authors in [5] used fuzzy C-means machine learning technique which is in line with the methodology this paper adopts – K-means clustering. Finally we developed traditional DFSA and carried out a performance evaluation of the four algorithms.

A. SIMULATION ENVIRONMENT AND PARAMETERS
The simulation is with one reader and numerous tags and focuses on the MAC sub-layer [15], [44]. The communication channel is assumed to be free of any interference or noise. It is also assumed that no tags enters or goes out of the RFID interrogation zone. As in IoT, we set up a high tag density RFID environment ranging from 100 to 1000 as seen in [22], [28], [34], [35], [44] and defined by [11]. Our simulation parameters are in conformity with EPC C1G2 as shown in table 2. Hence, at the time of simulation, 1,000 groups of 16-bit binary data were randomly generated with the MATLAB (8.5.0, Mathworks, Nathick Massachusetts, USA) software to simulate the RN16 code of the RFID tags. The tags were randomly deployed within the interrogation zone of one reader. Extensive Monte Carlo simulation is carried out as in [14], [16], [28], [34], [45]. The computer configuration is a CPU Intel core i7 of 2.30 GHz with a RAM of 8.00 GB running on Microsoft Windows 10 Home Edition.

B. PERFORMANCE METRICS
System efficiency (SE), success rate (identification accuracy), and identification time are used as metrics. In RFID

TABLE 2. Simulation Parameters.

Parameters	Value	Parameters	Value
Reader-to-tag data_0	1-Tari	RTcal	37.5µs
Reader-to-tag data_1	2-Tari	TRcal	50µs
Reader-to-tag rate	80kpbs	T1	62µs
Tag-to-reader rate	160kpbs	T2	62µs
Tpri	6.25µs	T3	100µs
Tari	12.5µs	Probe	4bits
Feedback	3bits	RN16	6bits
Query	22bits	ID	96bits
QueryAdj	9bits	Ack	18bits
R-T preamble	112.5µs	QueryRep	4bits
T-R preamble	37.5µs	Framesync	62.5µs

TABLE 3. Duration for different timing parameters.

Parameters	Value (µs)	Parameters	Value (µs)
T_1	62.50	T_{EPC}	37.50
T_2	62.50	T_{Ack}	50.00
T_3	50.00	T_{RN16}	62.00
T_{Querry}	412.50	T_s	62.00
$T_{QuerryAdj}$	168.75	T_i	100.00
$T_{QuerryRep}$	0.75	T_c	62.50

systems as shown in Table 3, the time duration for the three slots - S_s, S_i, and S_c are not equal [4], [27], [35]. Therefore, a good evaluation of RFID anti-collision algorithm must consider this. This study conforms to existing EPC C1G2 protocol. Hence, in the simulation and evaluation the timing parameters as in [4], [8], [9], [15], [44], [46] listed in Table 3 were used.

Like in [4], [44] we define identification time

$$T_t = S_s.T_s + S_i.T_i + S_c.T_c \qquad (31)$$

where T_s, T_i and T_c represent success slot time, idle slot time, and collision slot time respectively.

System Efficiency on the other hand is defined as:

$$SE = \frac{S_s.T_s}{L} \qquad (32)$$

where L is the frame size.

In our simulation study, we observe and compare the efficiency of the proposed kg-DFSA and its variants with different frame sizes L using different values for Q as earlier described in section II-B of this paper and as in [3], [4], [16], [28].

This paper adopts reference [16] definition of success rate (identification accuracy) as the probability of identified tags over the real number of tags. That is:

$$p\left(\hat{n} \mid n\right) = \frac{\hat{n}}{n} \qquad (33)$$

where $\hat{n}$ is number successfully identified tags and n the real number of tags.

C. RESULT OF CLUSTERING
Whereas the ultimate aim of this paper is to evaluate the performance of the proposed kg-DFSA algorithm, we consider
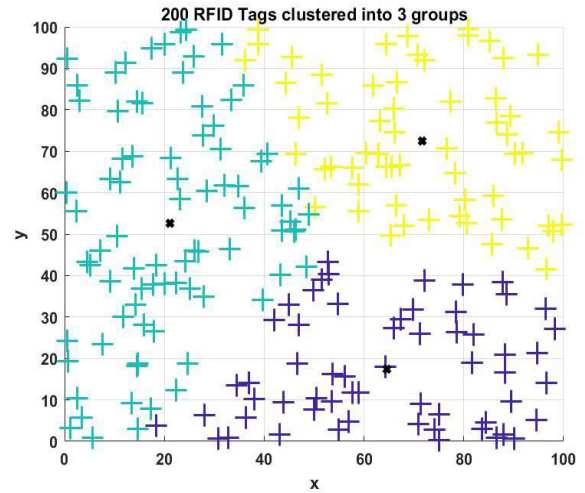

FIGURE 8. Result of grouping of 200 tags using k-means clustering. The three groups are coloured yellow, green and blue.

it imperative to first observe and analyze the results of the clustering since tag grouping itself is developed to enhance the dynamism in the kg-DFSA algorithm as efficient tag clustering could mean efficient tag reading. Earlier in this paper via Fig. 4, we showed the simulation result of an efficiency analysis of DFSA algorithm and here in testing our clustering, we use 200 tags for ease of viewing. The data for the clustering are the unique RN16 of each RFID tag randomly simulated in MATLAB in 16bits.

We implemented the improved k-means clustering for 200 tag density using Euclidean distance measure of (10) while we assume K=3. From the Fig. 8, it is seen that using the RN16 of each tag, the clustering algorithm groups the tags into three then converges. The tags were evenly distributed into three groups – blue, green and yellow. The "x" indicates the last centroids when the clustering algorithm converged. We show only the result of 200 tags for easy viewing. Else with 1000 tags and above, the clustering is also achieved, though, with more iteration. The even distribution of the clustered tags is promising for fairness in terms of access to the shared channel as the size of each group determines the number of slots allocated to the group within the kg-DFSA frame. Thereby, confirming the assertion by [37] on the celerity of k-means algorithm. Moreover, with such fair tag distribution and using (30), optimal efficiency is possible.

D. KG-DFSA EVALUATION RESULT AND DISCUSSION
With the clustering algorithm integrated into DFSA and the DFSA frame divided in chunks as in (30), k-means grouped DFSA (kg-DFSA) is proposed. In simulating the kg-DFSA, we wrote the clustering algorithm and the binary-to-decimal converter as MATLAB functions that run together with the DFSA algorithm. For the purpose of evaluating the performance of kg-DFSA, an extensive Monte Carlo simulation is
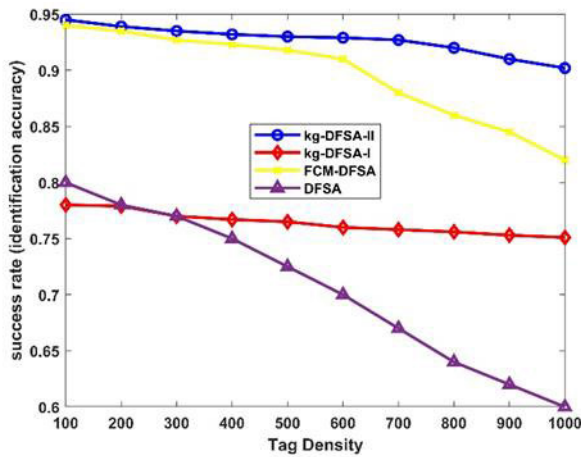
FIGURE 9. Result of success rate for the four algorithms.

FIGURE 10. System efficiency when frame size L=64.

carried out on traditional DFSA, fuzzy C-means based DFSA algorithm (FCM-DFSA) [7], traditional k-means clustering based DFSA algorithm (kg-DFSA-I) and improved k-means clustering based DFSA (kg-DFSA_II) using 1000 tags. Because this paper focuses on large scale deployment of RFID (IoT) and the behavior of the algorithms under evaluation in IoT scenario, the simulation started with 100tag density and increased all the way to 1000 as defined by [8] and like in [7], [17], [33].

From the result of success rate or identification accuracy shown in Fig. 9, the grouping based algorithms (FCM-DFSA kg-DFSA-II and kg-DFSA-I) gave better success rate than DFSA over the varying tag density. Algorithms FCM-DFSA and kg-DFSA-II initially gave almost identical results of over 93% when 100 to about 300 tags constituted the tag density.

Afterwards, kg-DFSA-II showed more stability of about 90% while FCM-DFSA fell a bit short to 82% in terms of tag identification accuracy as the tag density increased to 1000tags. On the other hand kg-DFSA-I algorithm shows a poor success rate of less than 75% within the whole tag density variation. Meanwhile traditional DFSA algorithm itself went from less than 80% accuracy with 100tags to about 60% with 1000tag density. The reason the three grouping based algorithms (kg-DFSA-I, kg-DFSA-II and FCM-DFSA) performed better than traditional DFSA in terms of success rate is because grouping of tags using their unique RN16 to a good extent minimizes the possibility of collision and maximizes success probability.

Furthermore, Kg-DFSA-I because of too much reliance on the initial centroids before effective tag clustering [5] could have been limited in its ability to evenly distribute the tags into groups in a fair manner that ensures tags' probability of colliding during the random access of the shared channel is minimized. Hence, it is deduced that effective clustering (not just clustering) is pivotal to the identification accuracy performance of any grouping based RFID anti-collision MAC protocol. More so, when the reader is pre-informed of the
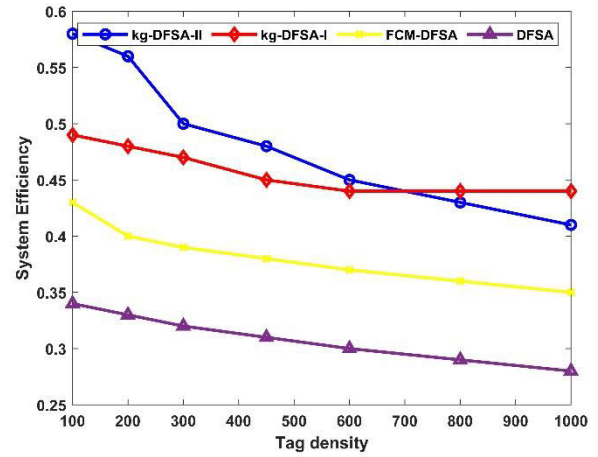
number of tags within its read range in addition to grouping as in kg-DFSA_II, the reader is able to start the reading effort by broadcasting a frame size that is precise. This minimizes the possibility of ideal slots while increasing the number of success slots.

The efficiency of DFSA-based RFID systems depends on the frame size [17]. Therefore, we performed the system efficiency (SE) evaluation with different frame sizes (64, 128, 256 and 512) and the results are as presented in Fig. 10, 11, 12 and 13 respectively. Looking at the SE results in Fig. 10 and 11, it can be seen that the SE was relatively high initially when the tag densities were smallest (100 tags) for the four algorithms. It is also seen that the proposed kg-DFSA-II edged the other three (traditional-DFSA, FCM-DFSA and kg-DFSA-I). However, the results still followed the fact expressed in our SE of equation (19) and subsequently Fig. 4, where we had earlier proven that efficiency is optimal when the tag density is approximately equal to the frame size L.

Furthermore, in the subsequent SE results of Fig. 12 and 13 especially, SE for all four algorithms increased as tag density increased from 100 to 200 tags (for Fig. 12) and to about 600 tags (for Fig. 13) and afterwards crashed downwards. Although Kg-DFSA-II maintained some edge of between 0.42 and 0.6 of SE than traditional DFSA (0.28 to 0.37), FCM-DFSA (0.37 to 0.57) and kg-DFSA-I (0.38 to 0.58) in all the four different frame size scenarios, it is important to deduce that no matter the algorithm used, the frame size is supreme. Finally, its noteworthy to state that the more stable manner with which the SE result in Fig. 13 fell with increased tag density compared to how it increased before getting to its peak of 0.55 suggests that whereas, both are undesirable, collision slots might be a bigger challenge than idle slots in RFID systems.

From the identification time result shown in Fig. 14, traditional DFSA algorithm performed best initially while tag density was less than 300. However, as the number of tags increased further to 400 and above, the three grouping based
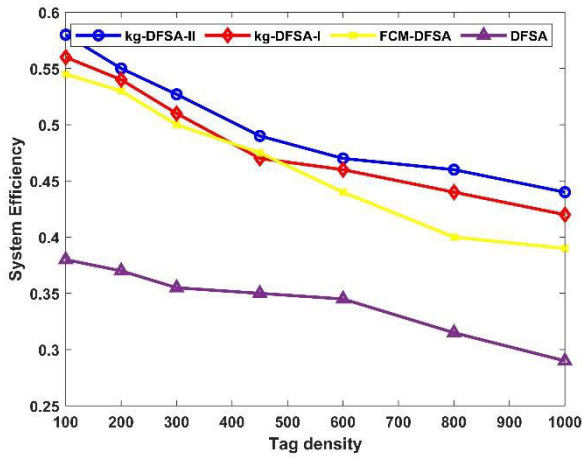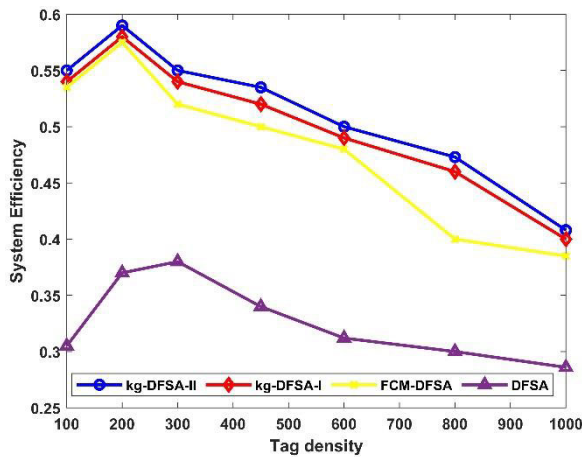
FIGURE 11. System efficiency when frame size L=128.

FIGURE 12. System efficiency when frame size L=256.
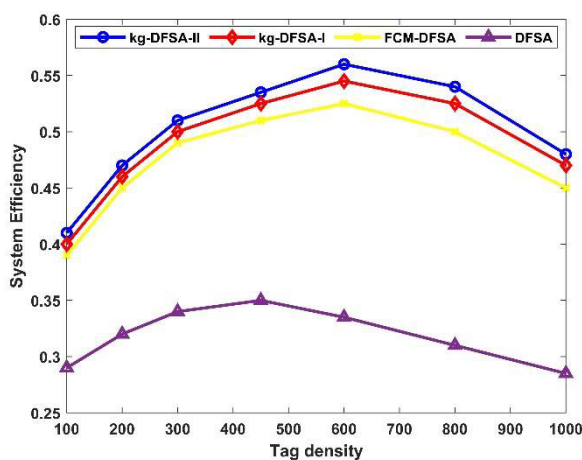

FIGURE 13. System efficiency when frame size L=512.

algorithms overtook DFSA and gave better performances. Specifically, kg-DFSA-II soared best edging the other algorithms. The initial increase in identification time can be
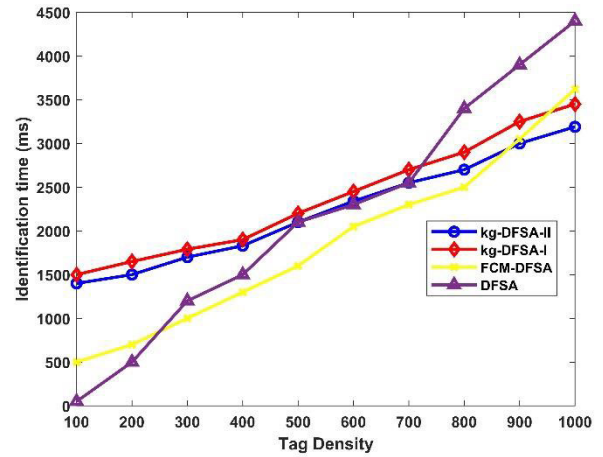

FIGURE 14. Total identification time for the four algorithms.

attributed to the initialization stages of the kg-DFSA-I and kg-DFSA-II algorithms. Recall that in kg-DFSA, the tag reading function of reader is divided into two phases – initialization phase and tag identification stages. The reader uses the initialization phase as a form of handshaking to count and group all tags within its read range. Afterwards, the tags are identified in groups. The reader utilizes some query cycle to achieve this initialization stage which takes some time. More so, the binary to decimal conversion of tags RN16 to achieve tag grouping also adds to overhead and consequently identification time of the proposed algorithm.

However, with increase in number of tags the proposed kg-DFSA-I and kg-DFSA-II algorithms overtook the traditional DFSA and FCM-DFSA by taking less time to identify all tags. This must have resulted due to less of collision slots and idle slots and more of success slots that resulted due to precise frame size the reader uses in both kg-DFSA algorithms, which equips both with information on accurate tag number. This can also be attributed to the fact that it may have taken traditional DFSA algorithm extra time in querying unidentified tags due to too many collision slots. Too many collision slots means more tags have collided and to identify all the tags, the reader using the traditional DFSA algorithm must send more query rounds. Therefore, for relatively taking too much time to identifying all tags when the tag density is not dense, and relatively lesser time to identify all tags when the tag density is much, it is concluded that the proposed grouping based kg-DFSA algorithm will be well suited for dense RFID deployment and might not be best for few tag density RFID systems.

E. PERFORMANCE SUMMARY IN RESPECT TO IoT APPLICATION

In light of IoT applications, a summary of the performance improvement of the proposed kg-DFSA, its variant, and FCM-DFSA over traditional DFSA using the highest tag
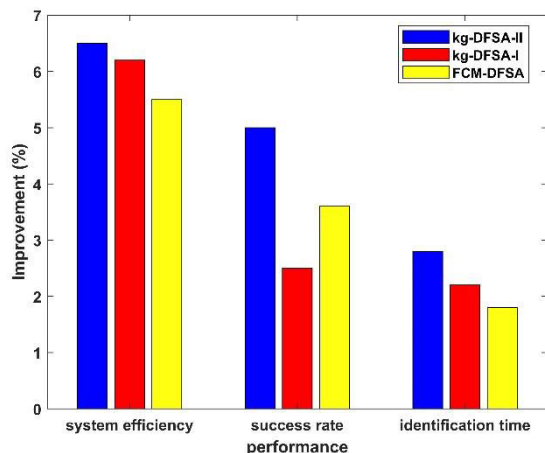
FIGURE 15. A summary of % improvement of the grouping based algorithms over traditional DFSA.

TABLE 4. Description of symbols and variables.

Variable	Description
x_i	RFID tags of a given cluster that are within the read range of the reader
x_j	RFID tags of another cluster that are within the read range of the reader
i, j	Ranges from 1 to n
n	Total number of tags
k	Number of clusters
z_1	Centroid of cluster 1
z_2	Centroid of cluster 2
z_k	Centroid of the kth cluster
n_{est}	Tag number estimate
n_{uid}	Unidentified tag number
d_{i1}	Distance of each tag to the centroid in cluster 1
d_{i2}	Distance of each tag to the centroid in cluster 2
t	Number of iterations
min	Minimum of 2 or more distances
max	Maximum of 2 or more distances
r	Imaginary centroid
Q	Query adjustment parameter ranges [0, 15]

density in our study (1000 tags) is shown in Fig. 15. It is seen from the graph that using improved k-means clustering to group RFID tags, providing the reader with prior knowledge about estimate of tags within its read range and forcing tags to reply to reader queries in timeslots that match their group ID, system efficiency, success rate and identification time are enhanced.

From the bar chart performance summary of the simulation study, it is unsurprising that all the grouping based algorithms (kg-DFSA-II, kg-DFSA-I and FCM-DFSA) especially kg-DFSA-II gave significant edge in system efficiency over traditional DFSA. Hitherto, and even in this paper, theoretical and simulation studies all agreed that the initial frame size with which a RFID reader uses to query surrounding RFID tags is a direct determinant of the efficiency of the RFID system [8], [16], [27]. System efficiency in RFID systems is the ratio of number of success slot to the frame size. The improvement in efficiency the proposed kg-DFSA algorithm yields is the direct effect of having a MAC algorithm that first gives the reader an accurate estimate of tags. Recall that these two major contributions were made in the initialization stage of kg-DFSA – tag counting and tag grouping. More so, the group reading strategy which forces tags to choose only time slots that are assigned to their group further ensures less collision and idle slots and more success slots. More success slots means improved efficiency.

In comparing the success rate (identification accuracy) of the grouping based algorithms under evaluation in this paper, the focus is on comparing their performances relative to traditional DFSA. Success rate defined in (33) measures the probability that a query cycle identifies a tag given total unidentified tags. The work of [16] shows a success rate of 0.9 when 2 and 3 tags were used but fell sharply to 0.8 when the tags were increased to 4. The authors didn't evaluate the success rate for 5 tags and above. The sharp drop in success rate from 0.9 to 0.8 just by mere increase of tag density from 3 to 4 strongly suggests that exposing their

algorithm to IoT scale of say 1000 tags (which is the scale of this paper) could cause a system failure. The kg-DFSA-II result of 0.9 success rate given 1000 tags which translates to about 50% improvement over traditional DFSA, makes kg-DFSA-II relevant in MAC protocol development for IoT applications.

Finally, from the earlier result of identification time in Fig. 14, it is seen that initially using 100 tags, all the grouping based algorithms especially kg-DFSA-II took relatively too much time to identify all tags but the situation changed with increase in number of tags. Specifically, from 500 tag density all the way to 1000, all the grouping based algorithms overtook DFSA with kg-DFSA-II giving a 28% improvement than DFSA. This clearly means that our proposed kg-DFSA algorithm is promising when large scale identification of RFID tags is involved.

V. CONCLUSION
One major research challenge to IoT application of RFID is Tag Collision Problem (TCP). This paper proposes a novel method of grouping and counting RFID tags such that the reader is equipped with information on the estimate of tags within its read range. Using improved K-means machine learning technique, this paper enhances DFSA algorithm of EPC C1G2 Protocol with more intelligence in a manner that its uses the prior estimate of tags to ab initio predict a frame size that is precise. Therefore, limiting the probability of

both idle and collision slots while enhancing the probability of success slots per query cycle. Moreover, the paper has also demonstrated through simulation that the new algorithm proposed in the paper edges traditional DFSA anti-collision algorithm and its variants in terms of identification time, system efficiency and success rate. In the future, our research work shall focus on the effect of frame adjustment policy on the efficiency of especially mobile RFID systems in view of IoT applications. Our future study would go cross-layer towards using real life RFID system in our evaluation.

APPENDIX

For the purpose of clear definition, Table 4 is a list the variables used in this paper as well as their descriptions.

ACKNOWLEDGMENT

REFERENCES

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014, doi: 10.1109/JIOT.2014.2306328.

[2] Y. Yang and X. Wang, "Fast RFID tag sorting at the edge for Internet of Things," *IEEE Access*, vol. 9, pp. 90268–90282, 2021, doi: 10.1109/ACCESS.2021.3090438.

[3] D. Zhong, "An ALOHA-based algorithm based on grouping of tag prefixes for industrial Internet of Things," *Secur. Commun. Netw.*, vol. 2022, no. 2, Apr. 2022, Art. no. 1812670.

[4] G. Zhang, S. Tao, W. Xiao, Q. Cai, W. Gao, J. Jia, and J. Wen, "A fast and universal RFID tag anti-collision algorithm for the Internet of Things," *IEEE Access*, vol. 7, pp. 92365–92377, 2019, doi: 10.1109/ACCESS.2019.2927620.

[5] B. Zhi, W. Sainan, and H. Yigang, "A novel anti-collision algorithm in RFID for Internet of Things," *IEEE Access*, vol. 6, pp. 45860–45874, 2018, doi: 10.1109/ACCESS.2018.2863565.

[6] N. H. Umelo, N. K. Noordin, M. F. A. Rasid, T. K. Geok, and F. Hashim, "Grouping based radio frequency identification anti-collision protocols for dense Internet of Things application," *Int. J. Elect. Comput. Eng.*, vol. 12, no. 6, pp. 5848–5860, 2022, doi: 10.11591/ijece.v12i6.pp5848-5860.

[7] X. Jia, Q. Feng, T. Fan, and Q. Lei, "RFID technology and its applications in Internet of Things (IoT)," in *Proc. 2nd Int. Conf. Consum. Electron., Commun. Netw. (CECNet)*, Apr. 2012, pp. 1282–1285, doi: 10.1109/CECNet.2012.6201508.

[8] J. Su, A. X. Liu, Z. Sheng, S. Member, and Y. Chen, "A partitioning approach to RFID identification," *IEEE/ACM Trans. Netw.*, vol. 28, no. 5, pp. 2160–2173, Oct. 2020, doi: 10.1109/TNET.2020.3004852.

[9] GS1. (2013). *EPC TM Radio-Frequency Identity Protocols Generation-2 UHF RFID Specification for RFID Air Interface*. [Online]. Available: http://www.gs1.org/sites/default/files/docs/epc/Gen2_Protocol_Standard.pdf%0Ahttp://www.gs1.org/sites/default/files/docs/epc/uhfc1g2_1_2_0-standard-20080511.pdf

[10] Z. Jiang, B. Li, M. Yang, and Z. Yan, "LC-DFSA: Low complexity dynamic frame slotted ALOHA anti-collision algorithm for RFID system," *Sensors*, vol. 20, no. 1, p. 228, Dec. 2019, doi: 10.3390/s20010228.

[11] J. Su, Z. Sheng, C. Huang, G. Li, A. X. Liu, and Z. Fu, "Identifying RFID tags in collisions," *IEEE/ACM Trans. Netw.*, early access, Nov. 14, 2022, doi: 10.1109/TNET.2022.3219016.

[12] B. R. System, J. Su, J. Zhou, Z. Sheng, S. Member, and A. X. Liu, "Unknown tag identification protocol based on collision slot resolution in large-scale and battery-less RFID system," *IEEE Sensors*, early access, Sep. 9, 2022, doi: 10.1109/JSEN.2022.3203708.

[13] J. Su, Y. Chen, Z. Sheng, Z. Huang, and A. X. Liu, "From M-Ary query to bit query: A new strategy for efficient large-scale RFID identification," *IEEE Trans. Commun.*, vol. 68, no. 4, pp. 2381–2393, Apr. 2020, doi: 10.1109/TCOMM.2020.2968438.

[14] A. Fahim, T. Elbatt, A. Mohamed, and A. Al-Ali, "Towards extended bit tracking for scalable and robust RFID tag identification systems," *IEEE Access*, vol. 6, pp. 27190–27204, 2018, doi: 10.1109/ACCESS.2018.2832119.

[15] J. Su, Z. Sheng, A. X. Liu, Z. Fu, and Y. Chen, "A time and energy saving-based frame adjustment strategy (TES-FAS) tag identification algorithm for UHF RFID systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 2974–2986, May 2020, doi: 10.1109/TWC.2020.2969634.

[16] Z. Huang, R. Xu, C. Chu, Z. Li, Y. Qiu, J. Li, Y. Ma, and G. Wen, "A novel cross layer anti-collision algorithm for slotted ALOHA-based UHF RFID systems," *IEEE Access*, vol. 7, pp. 36207–36217, 2019, doi: 10.1109/ACCESS.2019.2900739.

[17] M. Chekin, M. Hossienzadeh, and A. Khademzadeh, "A rapid anti-collision algorithm with class parting and optimal frames length in RFID systems," *Telecommun. Syst.*, vol. 71, no. 1, pp. 141–154, May 2019, doi: 10.1007/s11235-018-0492-7.

[18] H. Wu, Y. Wang, and Y. Zeng, "Capture-aware Bayesian RFID tag estimate for large-scale identification," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 119–127, Jan. 2018, doi: 10.1109/JAS.2017.7510757.

[19] J. Su, Z. Sheng, D. Hong, and G. Wen, "An effective frame breaking policy for dynamic framed slotted ALOHA in RFID," *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 692–695, Apr. 2016, doi: 10.1109/LCOMM.2016.2521839.

[20] J. Choi and C. Lee, "Maximum a posteriori (MAP)-based tag estimation method for dynamic framed-slotted ALOHA (DFSA) in RFID systems," *EURASIP J. Wireless Commun. Netw.*, vol. 2012, no. 1, pp. 1–12, Dec. 2012, doi: 10.1186/1687-1499-2012-268.

[21] J. B. Eom and T. J. Lee, "Accurate tag estimation for dynamic framed-slotted ALOHA in RFID systems," *IEEE Commun. Lett.*, vol. 14, no. 1, pp. 60–62, Jan. 2010, doi: 10.1109/LCOMM.2010.01.091378.

[22] 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, doi: 10.1109/TASE.2008.917093.

[23] X. Wang, "A fast hybrid strategy-based RFID tag identification protocol," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–7, Feb. 2021, doi: 10.1155/2021/6646812.

[24] N. Cmiljanic, H. Landaluce, and A. Perallos, "A comparison of RFID anti-collision protocols for tag identification," *Appl. Sci.*, vol. 8, no. 8, p. 1282, Aug. 2018, doi: 10.3390/app8081282.

[25] Z. Zhou and B. Chen, "RFID counting over time-varying channels," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 1142–1150.

[26] Y.-H. Chen, R.-Z. Hung, L.-K. Chen, P.-T. Jan, and Y.-R. Su, "Channel-quality aware RFID tag identification algorithm to accommodate the varying channel quality of IoT environment," *Appl. Sci.*, vol. 9, no. 2, p. 321, Jan. 2019, doi: 10.3390/app9020321.

[27] J. Su, Z. Sheng, A. X. Liu, Y. Han, and Y. Chen, "Capture-aware identification of mobile RFID tags with unreliable channels," *IEEE Trans. Mobile Comput.*, vol. 21, no. 4, pp. 1182–1195, Apr. 2022, doi: 10.1109/TMC.2020.3024076.

[28] J. Su, Z. Sheng, A. X. Liu, Y. Han, and Y. Chen, "A group-based binary splitting algorithm for UHF RFID anti-collision systems," *IEEE Trans. Commun.*, vol. 68, no. 2, pp. 998–1012, Feb. 2020, doi: 10.1109/TCOMM.2019.2952126.

[29] B. Cao, Y. Gu, Z. Lv, S. Yang, J. Zhao, and Y. Li, "RFID reader anticollision based on distributed parallel particle swarm optimization," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3099–3107, Mar. 2021, doi: 10.1109/JIOT.2020.3033473.

[30] H.-W. Wang, "Efficient DFSA algorithm in RFID systems for the Internet of Things," *Mobile Inf. Syst.*, vol. 2015, pp. 1–10, Jan. 2015, doi: 10.1155/2015/942858.

[31] F. C. Schoute, "Dynamic frame length ALOHA," *IEEE Trans. Commun.*, vol. COM-31, no. no. 4, pp. 565–568, Apr. 1983. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1095854

[32] H. Vogt, "Multiple object identification with passive RFID tags," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2002, pp. 1–6, doi: 10.1109/ICSMC.2002.1176119.

[33] I. Onat and A. Miri, "A tag count estimation algorithm for dynamic framed ALOHA based RFID MAC protocols," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2011, pp. 1–5, doi: 10.1109/ICC.2011.5962792.

[34] J. Su, Z. Sheng, V. C. M. Leung, and Y. Chen, "Energy efficient tag identification algorithms for RFID: Survey, motivation and new design," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 118–124, Jun. 2019, doi: 10.1109/MWC.2019.1800249.

[35] J. Su, Z. Sheng, L. Xie, G. Li, and A. X. Liu, "Fast splitting-based tag identification algorithm for anti-collision in UHF RFID system," *IEEE Trans. Commun.*, vol. 67, no. 3, pp. 2527–2538, Mar. 2019, doi: 10.1109/TCOMM.2018.2884001.

[36] K. Kusrini, "Grouping of retail items by using K-means clustering," *Proc. Comput. Sci.*, vol. 72, pp. 495–502, Jan. 2015, doi: 10.1016/j.procs.2015.12.131.

[37] Y. Li and H. Wu, "A clustering method based on K-means algorithm," *Appl. Mech. Mater.*, vol. 25, pp. 1104–1109, Jan. 2012, doi: 10.1016/j.phpro.2012.03.206.

[38] O. Ossama, H. M. O. Mokhtar, and M. E. El-Sharkawi, "An extended K-means technique for clustering moving objects," *Egyptian Informat. J.*, vol. 12, no. 1, pp. 45–51, Mar. 2011, doi: 10.1016/j.eij.2011.02.007.

[39] H. B. Tambunan, D. H. Barus, J. Hartono, A. S. Alam, D. A. Nugraha, and H. H. H. Usman, "Electrical peak load clustering analysis using K-means algorithm and silhouette coefficient," in *Proc. Int. Conf. Technol. Policy Energy Electr. Power (ICT-PEP)*, Sep. 2020, pp. 258–262, doi: 10.1109/ICT-PEP50916.2020.9249773.

[40] D. Chi, "Research on the application of K-means clustering algorithm in student achievement," in *Proc. IEEE Int. Conf. Consum. Electron. Comput. Eng. (ICCECE)*, Jan. 2021, pp. 435–438, doi: 10.1109/ICCECE51280.2021.9342164.

[41] G. Alfian, M. Syafrudin, B. Yoon, and J. Rhee, "False positive RFID detection using classification models," *Appl. Sci.*, vol. 9, no. 6, p. 1154, Mar. 2019, doi: 10.3390/app9061154.

[42] H. Lv, X. Li, L. Jia, R. Li, J. Ding, and J. You, "Suffix-based double reader tag movement RFID anti-collision method," *J. Cloud Comput.*, vol. 10, no. 1, pp. 1–15, Dec. 2021, doi: 10.1186/s13677-021-00228-8.

[43] Y. Chen, J. Su, and W. Yi, "An efficient and easy-to-implement tag identification algorithm for UHF RFID systems," *IEEE Commun. Lett.*, vol. 21, no. 7, pp. 1509–1512, Jul. 2017, doi: 10.1109/LCOMM.2017.2649490.

[44] L. Arjona, H. Landaluce, A. Perallos, and E. Onieva, "Timing-aware RFID anti-collision protocol to increase the tag identification rate," *IEEE Access*, vol. 6, pp. 33529–33541, 2018, doi: 10.1109/ACCESS.2018.2849223.

[45] Z. He and H. Luo, "An efficient early frame breaking strategy for RFID tag identification in large-scale industrial Internet of Things," *Sci. Program.*, vol. 2021, pp. 1–6, May 2021, doi: 10.1155/2021/4603629.

[46] J. Su, Z. Sheng, D. Hong, and V. C. M. Leung, "An efficient sub-frame based tag identification algorithm for UHF RFID systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–6, doi: 10.1109/ICC.2016.7511360.

NOR KAMARIAH NOORDIN (Senior Member, IEEE) received the B.Sc. degree in electrical engineering (major in telecommunications) from the University of Alabama, Tuscaloosa, USA, in 1987, the master's degree from University Teknologi Malaysia, and the Ph.D. degree from University Putra Malaysia (UPM). She was appointed as an Associate Professor and a Professor with UPM, in 2006 and 2012, respectively. She had an experience of more than 25 years with the university, since 1988. During her tenure, she has been the Head of Department, the Deputy Dean of Academics, the Director of Corporate Strategy and Communication of the University, and the Dean of the Faculty of Engineering. She has secured more than 30 research and consultancy projects. She has published more than 200 journals and conference papers.

MOHD FADLEE A. RASID (Member, IEEE) received the B.Sc. degree in electrical engineering from Purdue University, USA, and the Ph.D. degree in electrical/electronics engineering (mobile communications) from Loughborough University, U.K. He is currently a Professor with the Department of Computer and Communications Systems Engineering, University Putra Malaysia (UPM). He is also the Director of the WSN Research Group, which focuses on wireless medical sensors in healthcare. He has been a Research Consultant to different projects, including the British Council UKIERI Project on wireless sensor medical research.

KIM GEOK TAN (Member, IEEE) received the B.E., M.E., and Ph.D. degrees in electrical engineering from the University of Technology Malaysia, in 1995, 1997, and 2000, respectively. He has been a Senior Research and Development Engineer with EPCOS, Singapore, since 2000. From 2001 to 2003, he joined DoCoMo Euro Laboratories, Munich, Germany. He is currently an Academic Staff with Multimedia University. His research interests include radio propagation for outdoor and indoor, RFID, multi-user detection technique for multi-carrier technologies, and A-GPS.

FAZIRULHISYAM HASHIM (Member, IEEE) received the M.Sc. degree from Universiti Sains Malaysia, and the Ph.D. degree in telecommunication engineering from The University of Sydney, Australia. He is currently an Associate Professor with the Department of Computer and Communication Systems Engineering, Universiti Putra Malaysia. His research interests include heterogeneous wireless communication systems and network security.

NNAMDI HENRY UMELO (Member, IEEE) received the B.Eng. degree in computer engineering from the Enugu State University of Science and Technology, Enugu, Nigeria, in 2006, and the M.Sc. degree from the Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria, in 2012. He is currently pursuing the Ph.D. degree with the WiPNET Laboratory, Department of Computer and Communication Systems Engineering, University Putra Malaysia (UPM), Malaysia. He is also an Academic Staff at the Akanu Ibiam Federal Polytechnic Unwana, Nigeria. His research interests include wireless networks, RFID for the IoT, WSN, and smart homes. He is also a COREN Registered Engineer and a Member of the Nigerian Society of Engineers.

· · ·