Received July 3, 2018, accepted July 31, 2018, date of publication August 6, 2018, date of current version September 7, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2863565

A Novel Anti-Collision Algorithm in RFID for Internet of Things

BAI ZHI[1,2], WANG SAINAN[1], AND HE YIGANG[2]
[1]Hunan Institute of Technology, Xiangtan 411101, China
[2]College of Electrical and Information Engineering, Hunan University, Changsha 410082, China

Corresponding author: Wang Sainan (410361802@qq.com)

This work was supported in part by the Outstanding Young Investigator Award under Grant 50925727, in part by the Natural Science Foundation of China under Grant 60876022, in part by the Youth Science Foundation–Natural Science Foundation of China under Grant 51107034, in part by the National Defense Pre-Research Foundation of China through a project under Grant C1120110004, and in part by the Scientific Research Fund of Hunan Provincial Education Department through a project under Grant 11C0479 and Grant 15C1288.

ABSTRACT This paper studies a novel anti-collision algorithm that is proposed in view of the problem, i.e., label reading collision in radio-frequency identification for Internet of Things. Based on the theoretical foundation of label grouping, the algorithm introduces the Mahalanobis distance and density function to traditional fuzzy C-means clustering grouping algorithm by using EPC code and effectively solves the problem of isolated points of clustering and the optimization problem of initial clustering center. Then, the algorithm realizes effective grouping of labels and distributing identification serial numbers to labels upon the distance from interior labels to the center of clustering. Meanwhile, the efficiency of algorithm can be improved through dynamically setting the frame slot time of readers upon the grouping condition to prevent collision. This paper analyzes the throughput rate theoretically in detail. The simulation results of throughput capacity, throughout rate, and slot efficiency of the algorithm manifest that the algorithm is superior to the most commonly used dynamic binary-tree algorithm and current dynamic ALOHA algorithm in performance.

INDEX TERMS Radio frequency identification sensors system, Internet of Things (IOT), anti-collision algorithm, tag grouping, Mahalanobis distance, density function.

I. INTRODUCTION

As a very common wireless automatic identification technology in the Internet of things system, radio frequency identification (RFID) in sensors system has a lot of incomparable advantages compared with the traditional barcode identification technology. It is pollution-free, rapid, automatic, and has strong penetrability, high security and high storage capacity, etc., and it is widely used in various kinds of areas, such as industry, logistics, production control, transportation and so on [1]–[4].

In the era of Internet of things, some researchers put forward a new technical scheme, which combines RFID for IOT, and establishes an enhanced RFID for IOT, where wireless RFID tag reading network is capable of processing data from both sensors and RFID tags. This technology has lots of advantages and it has been widely used in different fields, such as forest fire and the elderly home care. But these technologies have some shortcomings, one of the major drawbacks is the power efficiency of the network nodes. In the power efficiency field, one of an important influential factors is the Tag identification, which is the wireless RFID tag anti-collision algorithm in RFID for IOT.

The most common tag in RFID for IOT is the passive electronic tag (i.e. the operating mode is passive), which can only work when served with the energy transmitted from the reader, and the tags are unable to communicate with each other. If there are too many tags in the working channel of the reader, the reader will fail to read the data of the tags accurately, even can not to identify the tags. Therefore, it is very necessary to research Administrator new type of anti-collision algorithm so as to avoid the occurrence of such circumstance.

According to the search of references in the recent years, statistics show that there are two main relatively common anti-collision algorithms for RFID: ALOHA algorithm and binary tree algorithm. The principle of ALOHA algorithm is relatively simple, i.e. all the tags at the initial phase will compete for a communication channel at the same time, and only which is distributed can enter into the working mode, while other tags will wait for a period of time randomly before entering the next competition process. Obviously, the disadvantage of this approach is that the workload of the system is too large, especially under the condition of too many labels, the system will be overloaded [6]–[8]. Therefore, Hush [9] proposed an improvement, which enabled the ALOHA algorithm to adjust time slot adaptively according to the unidentified tag population, thus the optimum value of the system throughput is achieved. The principle of the binary tree algorithm is easy to understand as well, i.e. to enable the tags to make choice between 0 and 1 constantly, so as to achieve the purpose of ranking the tags [10], [11]. Although with such algorithm, the problem of anti-collision is solved to some extent, it has the same weakness as that of the ALOHA algorithm. When there are too many tags, too many interactive processes will occurred, giving rise to low identification efficiency on the whole. Wu *et al.* [12] proposed an improvement method against the deficiency of the binary tree algorithm. With the regressive binary-tree search, this method has improved the recognition efficiently. Deng and Liu [13] has proposed a new type of anti-collision identification algorithm by combining the advantages of the above two algorithms. With this algorithm, firstly all the tags are grouped and then identified with the ALOHA algorithm. Although with such algorithm, good effect has been achieved, it is affected by the tag grouping result. Therefore, the algorithm is only applied in some cases.

Based on the above analysis, this paper has proposed a new type of anti-collision algorithm for RFID. The core idea of this algorithm is to use the fuzzy means clustering algorithm to group the tags. In the algorithm, the Markov distance and density function are introduced to improve the clustering analysis results in the tag grouping process, and to adjust and optimize the frame time slot dynamically to achieve the best recognition results. The rest of this paper is outlined as follows. In Section 2, an overview of the brief analysis of tag anti-collision algorithm will be presented. In Section 3, a new tag grouping algorithm model uses the density function and the Mahalanobis distance in the FCM algorithm, the performance and features of the proposed algorithm will be analyzed. Finally, conclusions will be presented in Section 5.

II. THE BRIEF ANALYSIS OF TAG ANTI-COLLISION ALGORITHM

The innovation of integrated circuit technology promotes the reduction of the expense on the application of the RFID chips, and speeds up the application of RFID technology in different fields and industries [14]–[18]. In order to promote the improvement of the functions of the RFID, experts from the field of global radio frequency (RF) technology, as well as scholars and enthusiasts, have done a lot of meaningful theoretical researches and experimental work, especially the research on the RFID anti-collision algorithm [19], [20]. Anti-collision algorithm's essence is to put forward the corresponding strategy, and make the read/write device identify tags one by one fast and efficiently. The phenomena of low system throughput and recognition rate are common in the process of tag identification, especially outstanding in passive UHF RFID [21], [22]. From a certain sense, tag collision problem has become a big obstacle to the popularity of RFID technology application. According to the existing literature, the RFID anti-collision algorithm is mainly based on the random algorithm of ALOHA protocol and the determination algorithm is based on the binary tree algorithm [23]–[26].

The search algorithm based on the binary system mainly contains four processes, including requesting the serial number, choosing the serial number, reading data and choosing. By traversal search, the reader will send commands to the transponder for many times [27]. Each time the reader will group the feedback information of RFID tags, and then identify the bit in collision quickly and effectively and finally identify all tags successfully. In the process of grouping, tag ID information is stored in a binary way. Therefore, it is also called "binary tree". The model diagram of binary tree search algorithm is shown as below.

The basic idea of the algorithm is to divide the tags in conflict into two subsets of 0 and 1; then query the subset of 0 first, if there is no conflict, the tag is identified correctly, otherwise, continue to divide the subset of 0 into the subset of 00 and 01. The process continues until all the subsets of 0 in the tags are identified. Then query the subset of 1 with this step. Considering the existing problems of the basic binary tree algorithm, scholars put forward some improved algorithms, such as dynamic binary tree and retreating binary tree. Binary search algorithm adopts a top-down traversal search strategy in which EPC sequence is transmitted many times, during which the algorithm efficiency is reduced with the increase of the tags to be identified, and meanwhile the time overhead is increased [22], [28].

ALOHA algorithm is a random algorithm, and the label can send data to the reader at all the times. In a period of time, label which enters into the reader identification range has three cases: a label, no label or multiple labels. When a new label enters into the recognition space of the reader, the label will take the initiative to send the data to the reader. Introducing slot in the ALOHA algorithm and dividing the transmission time into a plurality of discrete time slot. The length is not less than one frame (ID tag) transmission time. To the reader identification tag within its range, in a certain time slot, if there is only a label sends data to the reader, then the label is to be successfully recognized. Therefore, each time slot may have three kinds, namely success identification, tag collision and idle time slot, as it shows in Figure 2.

In the graph, S denotes the time slot which is successfully identified, C denotes the collision time slot, and I denotes the idle time slot.

The representatives of algorithm based on of the ALOHA algorithm include ALOHA, slotted ALOHA, dynamic frame-slotted ALOHA, advanced framed slotted ALOHA (AFSA) and EDFSA. Throughput of pure ALOHA algorithm is as follows:

$$S = GP_e = Ge^{-2G} \tag{1}$$

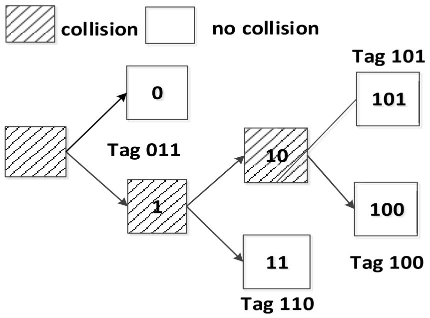

FIGURE 1. The theory of binary tree algorithm.

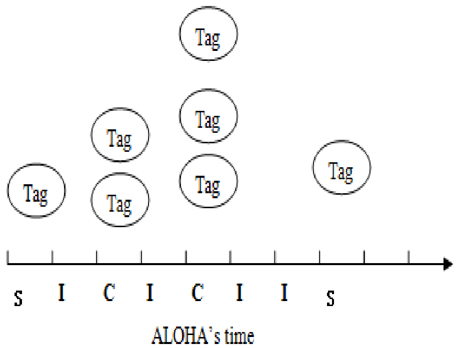

FIGURE 2. The slots of ALOHA.

When $G = 0.5$, throughput S reaches the maximum 0.184. When $G > 0.5$, the functions of systems is deteriorated dramatically, and it enters a state of instability.

III. THE NEW TAG GROUPING ALGORITHM

Aiming at all sorts of problems in the process of tag identification, and considering the uniqueness characteristic of RFID encoding, a new type of tag grouping anti-collision algorithm will be discussed in this paper, with a reference to the principle in FCM clustering of grouping the RFID tags. Its core principle is to group tags first, and then to set the frame time slot according to the result of grouping, and to assign a unique serial number for each tag in the group, and finally identify them in turn according to the serial number. Advanced FCM algorithm and tag grouping strategy will be introduced respectively below. In this chapter, we will introduce the optimization of FCM algorithm and the grouping strategy of tags in detail.

A. ESTIMATION OF TAG NUMBER

In the recognition process, RFID tags exchange information (mainly information about changes in the state of a label) with the reader. The label contains five states: sleep, activation, identified, waiting and conflict. Labels in the state of activation and sleep represent the success of slot choosing. And only a tag in the state of activation can exchange information and data with the reader. If the label is successfully identified, then the label in this round of the recognition process will not response any more. Tags in the state of sleep, waiting or conflict can have the opportunity to be activated by the reader [29].

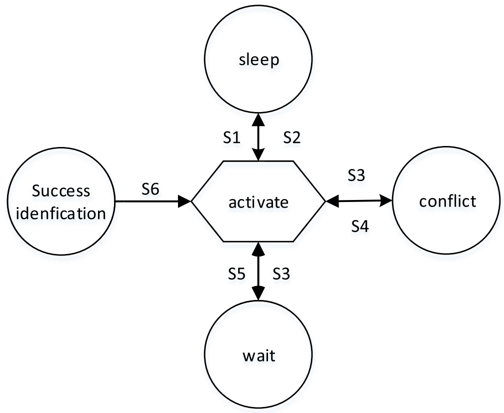

FIGURE 3. Schematic diagram of state transition of tag.

RFID tag exchanges information and data with the reader, and the change process of the label state is shown below. All the labels are active at the beginning of recognition. Groups that are not selected to be recognized enter the waiting state, and the labels selected remain active after its grouping. The latter transfers into a sleep state until the reader wakes up tags to start to recognize after the completion of slot selection. All labels in a poll after the reader response identification request to return into activation state and start again until the selected slot is successfully read in response to the REQ Kill request to enter the identified state. Other states which keep waiting will be wakened up by the reader on condition that all tags in the identified group move into the identified state [30], [31].

In Figure 3, *S1* - *S6* present the status of the RFID tag, set
S1=REQ_WAKEUP,
S2=REQ_SLEEP,
S3=REQ_START,
S4=REQ_COLLISION,
S5=REQ_STOP, *S6*=REQ_KILL.

In the process of tag identification, tag grouping algorithm is favored by many researchers because of its own incomparable advantages. These researchers proposed some representative tag grouping algorithms and confirmed the feasibility of the algorithm theoretically. But in practical situations, the number of tags in a time slice is usually unknown. We must accurately estimate the number of tags in order to determine the optimal frame length.

1) ANALYSIS OF DFSA ALGORITHM

Dynamic frame slot algorithm (DFSA) is the progress of the FSA algorithm. When the number of tags is excessive or inadequate, FSA algorithm has two shortcomings: increasing collision frequency and wasted time slots. While DFSA overcomes them by estimating the number of labels and dynamically adjusting the frame length according to the number of tags to improve the system's throughput. In the recognition process of DFSA, any random time slot can only belongs to one of these three cases: idle time slot, a success slot and a collision slot. Idle time slot– no tags select to reply in this time slot; the success time slot–only one tag in the current time slot replays and the reader successfully recognize labels; collision slot– two and more labels reply in the time slot to make the data collision. Firstly, we analyze the DFSA algorithm model, when there are too many tags in the systems, system throughput will be reduced obviously. Suppose that total number of effective tags in the systems is n, the time slot number required is L, it is known from the statistics law that: the probability of one time slot selected by r tags shall be:

$$B_{n,1/L}(r) = \binom{n}{r}\left(\frac{1}{L}\right)^{r}\left(1-\frac{1}{L}\right)^{n-r} \tag{2}$$

After the identification cycle of a frame of data, expected value of the time-slot number where tags are successfully identified, empty time-slot number and time-slot number where collision occurs are shown respectively as follows:

$$a_1^{L,\,n} = L \times B_{n,\,1/L}(1) = n\left(1-\frac{1}{L}\right)^{n-1} \tag{3}$$

$$a_0^{L,\,n} = L \times B_{n,\,1/L}(0) = L\left(1-\frac{1}{L}\right)^{n} \tag{4}$$

$$a_k^{L,\,n} = L - a_0^{L,\,n} - a_1^{L,\,n} \tag{5}$$

Under this circumstance, throughput rate $S = \frac{a_1^{L,\,n}}{L}$ for systems can be derived, namely:

$$\begin{aligned}\frac{dS}{dn} &= \frac{1}{L}\left(1-\frac{1}{L}\right)^{n-1} + \frac{n}{L}\left(1-\frac{1}{L}\right)^{n-1}\ln\left(1-\frac{1}{L}\right)\\ &= \frac{1}{L}\left(1-\frac{1}{L}\right)^{n-1}\left[1+n\ln\left(1-\frac{1}{L}\right)\right] = 0\end{aligned} \tag{6}$$

Via Formula (7), frame length is solved as follows:

$$L = \frac{1}{1-e^{\frac{1}{n}}} \tag{7}$$

When there is enough tag quantity, expand L in a Taylor's series, and obtain:

$$L \approx \frac{1+\frac{1}{n}}{1+\frac{1}{n-1}} = n+1, \quad n \gg 1 \tag{8}$$

Formula (8) shows that, when the throughput rate for systems is maximum, the total number of tags in the systems is approximately equal to the frame length.

2) ESTIMATION OF TAG NUMBER

In the dynamic frame slotted ALOHA algorithm, the reader dynamically generates a frame composed of a plurality of slots. Each tag randomly selects a time slot to transmit information. And the reader shall estimate the number of tags when the RFID for IOT begins to recognize tags each time. This process plays a vital role in the whole recognition process. If there is a big gap between estimated number and the actual number of labels, it is not conducive to the reader to identify the label and it seriously affects the efficiency of the whole system. As to estimate methods of RFID, Vogt-II algorithm proposed by H. Vogt is a kind of dynamic estimation method that causes a great influence at the time. It considers the mathematical expectation value and actual observation value. It can get more accurate label estimation value and effectively reduce the calculation error in a larger number of tags, which previously tag estimation methods cannot have. The algorithm consists of two spatial vectors (E_1, E_2, E_3) and (C_1, C_2, C_3). And select label number in the shortest distance between the two vector spaces as the tag estimation value. Among them, three component vectors of (E_1, E_2, E_3) respectively represent mathematical expectation value of idle time slot, success slot and collision slot in a frame, while vector (C_1, C_2, C_3) represents observation value of idle time slot, success slot and collision slot in a frame.

$$\varepsilon_{vd}(n, C_1, C_S, C_c) = \min_{\tilde{n}}\left|\begin{pmatrix}E_1\\E_2\\E_3\end{pmatrix} - \begin{pmatrix}C_1\\C_2\\C_3\end{pmatrix}\right| \tag{9}$$

Vogt-II algorithm also has some shortcomings. For example, the calculation amount is larger, which increases the design cost of the tags. Combining RFID tag estimation algorithm based on 0-1 distribution proposed by Lin *et al.* [32], the paper optimizes it on the basis of Vogt- II algorithm and classifies idle slot and collision slot as failure time slot. Meanwhile, by increasing two counters to count the number of success time slot and failure time slot in a read cycle in the design of the reader side in order to facilitate the description. The processes of estimation methods of label number and steps of the reader in a read cycle are as follows:

Step 1: Initialize the counter on the end of the reader. Bestow zero upon success time slot counter and failure time slot counter. Success slot count value represented by *CS* and failure time slot counter value *CF*.

Step 2: The reader sends a Query (with the length of L) command to the tag within its working range.

Step 3: Label feedback information. It randomly selects an integer within $[0, L-1]$ as its own time slot number and feedback the time slot information to the reader after receiving Query command sent by the reader.

Step 4: Judgment of reader slot collision. Reader maps the time slot sequence number and label when it receives all label feedback information. If the time slot sequence number and label is one-to-one mapping relationship, then the time slot is a success slot, otherwise it is a failure time slot. According to this principle, to modify the success slot count and failure time slot count. If it is a success slot, then the counter *CS* plus 1, however when they collide, then *CF* plus 1.

Step 5: The values of *CS* and *CF* are obtained at the end of a read cycle.

Step 6: Repeat the implementation of Step 2 to Step 5 several times to calculate the average value of the number of the success time slot and failure time slot and estimate the number of tags according to it.

The tag estimation process is shown in the following figure.

B. ANALYSIS OF THE OPTIMAL FCM ALGORITHM

In 1973, on the basis of predecessors' work, Bezdek introduced the concept of membership degree and fuzzy weighted into c average clustering algorithm. He used membership degree to measure the corresponding relationship between a sample and categories, thus, the currently well-known fuzzy c mean value cluster algorithm was formed. The main principle of the algorithm is to calculate the corresponding weight between all the samples and fuzzy rules based on the objective function, meanwhile guarantee that the objective function is converged to the minimum value, the essence of which is a problem of global goal optimization [33].

1) THE FUZZY C-MEANS CLUSTER ALGORITHM

Fuzzy clustering analysis is a clustering analysis method by establishing the fuzzy similar relation, which is based on the characteristics, the closeness degree and similarity between the objective things. It is an in-depth analysis of the data of the objective things and is widely applied in data mining, image processing, and other fields. The fuzzy c-means cluster algorithm (FCM) is an improved and optimized method on the basis of fuzzy clustering algorithm, by which the implied relationship between the sample data will be found out by calculating the membership of any single sample data inside the sample, and the similar sample date will be classified into the same classification as much as possible. The FCM algorithm converts the cluster of sample data into a nonlinear optimization problem to find out the optimal solution, which has become to an important branch of unsupervised pattern recognition at present. The core idea of FCM algorithm is to first divide n vectors ($X = \{x_1, x_2, \cdots x_n\}$) into c fuzzy sets ($S_1, S_2, \cdots S_n$), and set $V = \{v_1, v_2, \cdots v_n\}$ as c clustering centers, and also to calculate the cluster center of each group to minimize the objective function values. The objective function is defined as follows:

$$L \approx \frac{1+\frac{1}{n}}{1+\frac{1}{n-1}} = n+1, \quad n \gg 1 \tag{10}$$

In which, $U = [u_{ik}]$ is a fuzzy classification matrix, u_{ik} means x_k belongs to class s_i, and u_{ik} requires to meet $\sum_{i=0}^{c} u_{ik} = 1$ and $u_{ik} \in [0, 1]$. In the $V = [v_i]$, v_i represents the i cluster center ($i = 1, 2, \cdots c$), m is weighted index (i.e. the fuzzy index), whose scope is [1.9, 2.0]. d_{ik} represents the distance between the i cluster center and x_k, calculated by the Euclidean distance Formula $d_{ik} = ||x_k - v_i||$. $J(U, V, X)$ represents the weighted sum of squares from all samples to cluster center. The fuzzy membership degree u_{ik} and cluster center v_i can be calculated by using the Formula (11) and the Formula (12).

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} (\frac{d_{ik}}{d_{jk}})^{\frac{2}{m-1}}} \tag{11}$$

$$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m} \tag{12}$$

FCM algorithm mainly contains three aspects. The first is to initialize the data of c clustering centers and the subordinate function value matrix, and meanwhile the result of data initialization needs to satisfy the above constraints; The second is to use Formula (11) and (12) to calculate the new membership matrix and the cluster center of different types; The third is to compare the clustering center distance of the two iterations and threshold parameter value, until the cluster center meeting the requirements is discovered. In the FCM algorithm, the optimization of the objective function can be achieved by updating the membership degree matrix through the continuous iteration. However, the application of Euclidean distance Formula in the FCM algorithm may lead to the phenomena of the local extremum and isolated points of the objective function, resulting in the deviation of the clustering analysis result. As a result, the algorithm has certain limitations.

2) THE OPTIMIZED FCM ALGORITHM

Because KFC algorithm takes advantage of the Euclidean distance formula to calculate the similarity of samples, the phenomena of poor stability of clustering effect and isolated points in the objective function will appear if there is a linear inseparable problem on the sample data boundary. After the further analysis focusing on the existing problem of FCM algorithm, the existing problems can be solved by the optimization from two aspects. First, as for the isolated points, Mahalanobis distance will be used to replace Euclidean distance formula to solve the problem in calculating the similarity of samples. Second, the problem that the stability of the FCM algorithm in clustering center is not strong which would be solved by introducing a density function to optimize the initial clustering center selection.

a: THE OPTIMIZATION OF THE INITIAL CLUSTERING CENTER SELECTION

FCM algorithm is sensitive to the selection of the initial clustering center, so the different initial clustering centers will lead to different results, which makes the clustering result often turn into a local optimal solution, rather than a global optimal solution. Therefore, the density function is introduced. Thus, the initial clustering center will be calculated after referring to the density range of the sample distribution. The optimization method of the initial clustering center selection, which is based on the density function, is as follows:

Suppose that a data of sample space is: where p is all the objects which are involved in clustering analysis in the FCM algorithm, and m is the individual numbers of the sample space. Sample density reflects the distribution degree of individual sample data, and has an inevitable relation with the effective radius (r_d) of neighborhood density of the individual sample.

Step 1: As for each sample point in P, calculate its density according to Formula (13), is the distance between individual sample and clustering center.

$$d_i = \sum_{k=1}^{s} \frac{r_d^2}{r_d^2 + 4||x_i - x_k||^2} \tag{13}$$

In which, $r_d = \frac{1}{2} \times \sqrt{\frac{1}{m(m-1)} \sum_{k=1}^{m} \sum_{i=1}^{m} ||x_i - x_k||^2}$ is the effective radius of sample individual neighborhood.

Step 2: For the data set P, if the density function of value of p is greater than the density value of all data points, and then set the point P as the cluster center x_1^*. And suppose x_1^* is the clustering center after the first calculation and D_1^* is the corresponding density function value. Modify the density function value of each sample according to the Formula (14). To compare the modified value, and pick out the point of maximum value x_1^*, which will be a new clustering center.

$$D_i^* = D - D_1^* \exp[\frac{-||x_i - x_1^*||^2}{0.25\delta_a}] \tag{14}$$

Step 3: Determine whether the selection of the new clustering center is feasible according to the Formula (15). If the new clustering center meets the requirement, the algorithm will be stopped. Otherwise, to return to step 3.

$$D_i^*/D_1^* < \eta \tag{15}$$

In the Formula (15), η is the pre-set value. Parameters (δ_a) limits the detection neighborhood based on x_i. Data points excluded by the parameter value make few effects on the density function value of data to be calculated. The smaller the value of δ_a is, the smaller its limited scope is, but the more categories will be gotten. In turn, the greater the value is, the greater its limited scope is, and the fewer categories will be obtained. How to choose parameters δ_a involved in the density clustering is based on the Formula (16).

$$\delta_a = \frac{1}{2} \min_k \{ \max_i \{ ||x_i - x_k|| \} \} \tag{16}$$

Through the above three steps, the density function has been introduced to the selection of initial clustering center, then the Fuzzy *C-Means* Based on Density(FCMBD algorithm) is formed.

b: THE OPTIMIZATION OF THE DISTANCE METRIC

At first, the FCM algorithm focuses on solving the clustering problems between individual samples in the spatial data, calculating the distance from samples to the cluster center with the Euclidean distance formula, and reflecting the membership degree between samples and the cluster center. For a few specific data (i.e. concave type data), the limitations of this algorithm, which has been discussed in the previous chapter, will result in the flaws in the results of cluster analysis. Euclidean distance function formula [33], [35] is as follows:

$$dist(X, Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{17}$$

Applying Mahalanobis distance to the FCM algorithm can make up the defects of FCM algorithm, improve the accuracy of the similarity calculation on the two unknown sample sets, and obtain the result that conforms to the actual data greatly. To compare with the Formula (16), Mahalanobis distance is equipped with some great features. It will not be affected by dimensions, however it can eliminate the interference of the correlation between variables. In Mahalanobis distance formula, the use of covariance matrix inverse in calculating the distance between the samples makes the space distance of the sample convex, which avoids the happening of isolated points.

For a sample data of N consistent with Mahalanobis distance formula, if d_{ij} represents the distance between the i sample and the j sample, X_i and X_j respectively stand for the vector formed by M index of the i sample and the j sample. S is the general covariance matrix of the sample.

$$d_{ij} = ||c_i - x_j|| = \sqrt{(X_i - X_i)S^{-1}(X_i - X_i)^{-1}} \tag{18}$$

c: THE ALGORITHM DESCRIPTION OF FCMBMD

To sum up, combine the density function and the advantage of Mahalanobis distance in FCM algorithm, and to apply it into the FCM algorithm, thus, Fuzzy C-Means Based on Mahalanobis and Density (FCMBMD) are concluded. FCMBMD algorithm finds out the sample point in high density with the aid of the density function, and makes use of Mahalanobis distance to calculate the similarity among the samples, which not only avoids instability caused by the random selection of the initial value, but also avoids the problem in the connections between the various features could not be taken into consideration when Euler's formula is used to compute the similarity between samples. The algorithm makes the space distance calculated by the sample similarity convex, which can avoid the possible existence of some isolated points. Steps of algorithm are as follows:

Step 1: Calculate the density value of each data point according to Formula (12). Compare the density value of all the data points, pick out the sample point with the largest density function, and find the initial center in the appropriate amount.

Step 2: Set the fuzzy index m, m > 1.

Step 3: Select stop parameter h.

Step 4: Set the value of δ_a, that is, the initial center obtained by the parameters set according to the Formula (13).

Step 5: Suppose the times of the clustering number's iterations is k ($k = 0, 1, 2 \cdots$). For the k iteration, calculate the distance value d_{ij} between the user and each cluster center.

Step 6: Calculate the membership degree matrix U^k based on clustering center distance and distance function.

Step 7: Calculate clustering center matrix $C^{(k)}$.

Step 8:Make the user into a cluster $\forall x_i$ according to the membership degree matrix U^k. C_k is the k user cluster. When $C_k\{k|u_{ki} = \max_j(u_{ki}), 1 \leq j \leq c\}$, $x_i \in C_k$ exists, the i user will be assigned to the k user cluster.

C. TAG GROUPING ALGORITHM

1) THE TAG GROUPING STRATEGY

EPC is a new generation of product coding system launched by the International Bar Code Organization. And the RFID electronic tag coding follows the coding rules of EPC. Electronic tag code contains at least four parts, including header, general manager number, object class and serial number. Header defines the length and structure of coding, and the header is usually fixed for certain applications; General Manager Code identifies an organizational entity; Object classification code is used by the EPC management entity to identify the types of an item; the serial number within each object classification is unique. EPC coding system provides the theoretical support for tag grouping. In the process of the RFID identification, the tag within the recognition scope will be grouped effectively, and it could be identified in sequence, which finally achieves the goal to reduce the occurrence rate of conflict.

Tag grouping means to classify plenty of tags according to a certain characteristic of the tags. Since information of the tags is unforeseen beforehand, tag grouping can be considered as a typical unsupervised clustering problem. To realize tag grouping, first of all, a characteristic that can represent the tags shall be selected. According to the generative rule of the modern electronic tags, for whichever type of tag, for the purpose of verification, a cyclic redundancy check code, namely the commonly referred CRC code, which is generated from the data written in will be written in simultaneously, generally placed at the last few bits of the data package. Thus, when the reader reads the tag information, CRC code of the tag will be obtained naturally.

With the development of the communication technology and micro-electronic technology, length of the current CRC code can be up to 16 bits, namely, the recurrence probability of CRC code of the two tags is 1/216.The extremely low probability has guaranteed the uniqueness of the tags. However, as the CRC code consists of the binary system of 0 and 1, there exists great similarity. For example: although the CRC code 11111111 11111110 and 11111111 11111101 are different, they are similar, and are similar to 11111111 11111011 and 11111111 11110111, etc. Therefore, there is great similarity probability for the CRC code of different tags. Since the CRC code has low repeatability and high similarity, it is selected by us, so as to be served as the basis of tag classification.

After the objects of classification are determined, we will consider what method to use to classify the CRC codes of the tags.

As outlined above, since it is a typical unsupervised clustering problem, the K-mean algorithm will be considered to be applied first [34].

Nevertheless, subjecting to the selection of the initial cluster center, different initial cluster centers have greater influence on the final cluster result. Besides, the binary structure of the CRC code enables the similar concepts between the two samples to be fuzzy, because the traditional K-mean algorithm is based on the distance function. Thus, it is easy to cause the same similarity of several tags in the group, unfavorable for the next-step time slot allocation [36]. To solve this problem, the paper selects the fuzzy c mean value cluster algorithm, equivalent to formulating an artificial rule by introducing a membership matrix. Thus, not only the cluster effect is guaranteed, but also the similarity differentiation of the tags in the cluster is guaranteed as well [3], [37].

Suppose that a data set to be clustered is: $X = \{x_1, x_2, \cdots, x_n\}$, in which n is the classifiable number contained by the data, $x_k = \left\{x_{k1}, x_{k2}, \cdots, x_{kp}\right\}$, p is the number of band. Suppose that the data are required to be classified into c classes, the degree of membership of x_k to class i is u_{ik}, define the membership matrix to be $U = [u_{ik}]_{C\times V}$; meanwhile, the degree of membership is required to meet the following constraint conditions:

$$
\begin{aligned}
&u_{ik} \in [0, 1] \quad 1 \leq i \leq c,\ 1 \leq k \leq n\\
&\sum_{i=1}^{c} u_{ik} = 1 \quad 1 \leq k \leq n\\
&\sum_{i=1}^{n} u_{ik} > 0 \quad 1 \leq i \leq c
\end{aligned}
$$

Then, the objective function of the fuzzy c mean value cluster algorithm is:

$$J = \sum_{i=1}^{n}\sum_{k=1}^{c}(u_{ik})^m \left\|x_k - v_i\right\| \tag{19}$$

In which, m is the fuzzy weighted coefficient.

Here, a very important issue must be considered. Since it is the unsupervised clustering, it is unknown beforehand that, how many groups to be classified are the most appropriate. If the quantity of the cluster center is selected improperly, it is easily to cause too many or too few tag quantity for a group, thus giving rise to waste or shortage of the frame length. The paper is intended to solve this problem by setting a reasonable frame length range. If after the initial classification, tag quantity of a group exceeds this range, make corresponding adjustment as per the rule, and then make clustering again, until the conditions are met. The reason to adopt this method is that, the time-slot numbers set by the reader are usually the definite values 20∼28, i.e. the frame length range is 1∼256, we can choose a reasonable section thereof to have it serve as the frame length range of the algorithm convergence. Thus, the cluster result is guaranteed as relatively optimal.

The concrete realization process of the tag grouping algorithm based on fuzzy c mean value cluster is shown below:

Step 1: Initialize the cluster center v, number of class c, fuzzy weighted index m and frame length range $[L_{\min} L_{\max}]$;

Step 2: Calculate the fuzzy membership degree matrix u according to formula (20);

$$u_{ij} = \begin{cases} \left[\sum_{k=1}^{c} \frac{\|x_i - v_j\|^{\frac{2}{m-1}}}{\|x_i - v_k\|^{\frac{2}{m-1}}} \right]^{-1} & \|x_i - v_k\| \neq 0 \\ 1 & \|x_i - v_k\| = 0 \& k = j \\ 0 & \|x_i - v_k\| = 0 \& k \neq j \end{cases} \tag{20}$$

In which, u_{ij} is the fuzzy membership degree of the condition where individual x_i belongs to class j, v_j is the cluster center of class j.

Step 3: Update each cluster center v according to formula (21);

$$v_j = \frac{\sum_{i=1}^{n} u_{ij}^m x_i}{\sum_{i=1}^{n} u_{ij}^m} \tag{21}$$

In which, n is the number of primary data vector.

Step 4: Calculate the target value of the cluster according to Formula (22). If the minimum conditions of the objective function set could not be met, return to Step 2, otherwise, continue;

Step 5: Make statistics to tag quantity L of each group. If one of which falls beyond the reasonable range, i.e. $l_i \notin [L_{\min} \quad L_{\max}]$, adjust cluster center quantity according to the rule of Formula (14), and return to Step 1, otherwise, the algorithm will be stopped.

$$c = \begin{cases} c-1 & l_i < L_{\min} \\ c+1 & l_i > L_{\max} \\ c & L_{\min} \leq l_i \leq L_{\max} \end{cases} \tag{22}$$

2) DYNAMIC TIME SLOT ALLOCATION STRATEGY BASED ON OPTIMIZED FCM

Suppose that after clustering, all the tags are classified into L groups, and tag quantity of each group is l, where $L_{\min} \leq l \leq L_{\max}$. Under this circumstance, time-slot number required by each group can be determined as l at most. So, set the frame length to be l. Since tag quantity of each group is different, frame length of each group is different as well. Nevertheless, numbers of each tag in the group has not been determined yet. Since the determination of numbers is very important, the state of collision will occur only when number of each tag is assured to be unique. The method adopted by this paper is as follows: first of all, calculate the mean value of the CRC codes in each group; then calculate the distance from the CRC code of each tag to the mean value in the group; finally determine the number of each tag according to the distance, i.e. the nearer the distance, the smaller the number will be. If the circumstance where the distances are identical occur, the tag serial number calculated first shall be numbered

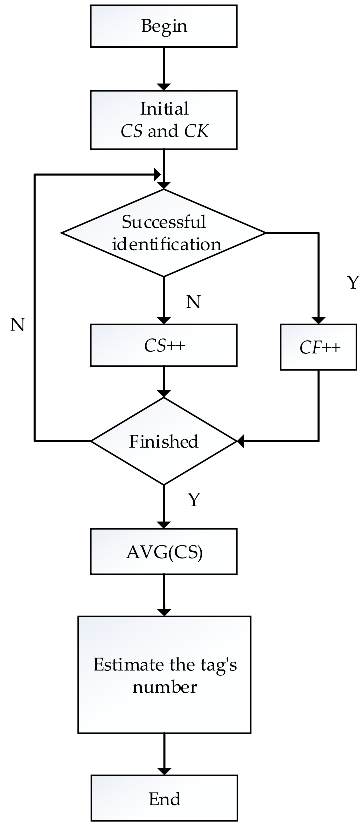

FIGURE 4. Tag number estimation flow-process chart.

firstly. Thus, tags of each group can be allocated with a unique number. Then, the reader will identify the tags of each group in a proper order.

The distance calculation formula is based on Mahalanobis distance, as is shown below:

$$d_{ij} = ||c_i - x_j|| = \sqrt{(X_i - X_j)S^{-1}(X_i - X_j)^{-1}} \quad i = 1, 2, \cdots, c; \quad j = 1, 2, \cdots, n \tag{23}$$

3) POLLING IDENTIFICATION PROCESS

To guarantee the smooth-going of the polling identification, we have set up 2 soft counters in the program of the reader: *rs* counter and *rus* counter, respectively for making statistics to the number of tags successfully read and read unsuccessfully. Although, theoretically, the tag grouping algorithm which mentioned previously by the paper enables each tag to have one unique serial number code, which will be successfully read so long as the systems work normally, the reality is that even if there is only one tag occupying the channel, there is the possibility of identification failure due to environment interference, etc. Therefore, statistics shall be made to the number of tags failing to be identified. In addition, since the tag grouping algorithm of the paper enables the number of frame slotted and tags to be equal, the circumstance of empty time slot does not exist theoretically. So, statistics of such state will not be made.

After the completion of tag grouping and time slot allocation for the tags in the group, the reader will empty and initialize the 2 soft counters, and start to carry out polling against a group of tags. After the polling, the reader will make statistics of the quantity of unidentified tags of the current group according to the conditions of the two soft counters. Next, the reader will continue to carry out polling to the next group, till polling of all the groups are completed. Afterwards, put all the unidentified tags of all the groups together to re-group and carry out the next polling. Since there is few quantity of tags remaining unidentified after one time of polling, before the next grouping, the frame length range $[L_{\min}, L_{\max}]$ can be adjusted appropriately, which is beneficial for speeding up grouping and identification. If total number of the remaining tags is less than $L_{\max}$, there is no need to regroup, however just carry out time slot allocation directly. Flow chart of grouping algorithm is shown as follows:

The process of one-time polling is shown as follows:

Step 1: The reader will send a REQ_Query request, including the time-slot numbers to be read by the reader;

Step 2: After successfully identifying the tags, the reader will receive an ACK_Query response, and turn to Step 3; if the reader fails to receive the ACK_Query response, it means identification failure, and turn to Step 4;

Step 3: The reader will send an REQ_Kill request, lock the tags successfully read, *rs* counter will plus 1; judge whether or not time slots of the group have been read completely, if so, turn to Step 5, otherwise, return to Step 1;

Step 4: *rus* counter will plus 1, judge whether or not time slots of the group have been read completely, if so, turn to Step 5, otherwise, return to Step 1;

Step 5: Judge whether or not polling for all groups have completed, if so, the polling shall be finished; otherwise, empty all the counters, prepare to read the next group of tags and return to Step 1.

D. ANALYSIS OF ALGORITHM PERFORMANCE

In the RFID for IOT, RFID tag anti- collision algorithm performance has a variety of analysis methods. The paper analyzes the algorithm from the following two angles: the number of time slots consumed by tags and the actual total data transmitted effectively.

Assume within the given time, M labels randomly move into the reader to recognize space. All tags are evenly divided into N groups, N in the range [0, t). To any group of labels, let T(i) and S(i) respectively represent the number of slots and transmission total data (0 < I < T) in recognizing all tags in the 'i' group, then the total number of queries is times of labels of 't' group:

$$T(M) = T(t-1) + T(t-1) + \cdots T(0) = \sum_{i=0}^{t-1} T(i) \tag{24}$$

The total transmission data is transmission data used by the 't' group labels:

$$S(M) = S(t-1) + S(t-1) + \cdots S(0) = \sum_{i=0}^{t-1} S(i) \tag{25}$$

If the label satisfies three conditions:

(1) Label ID distributes equally

(2) The number of tags is large enough

(3)The random numbers generated by the tag is uniformly distributed, that is, the probability that each label emerges in the interval $[0, t)$ is ${}^{1}/_{t}$.

Then the expectation of the number of tags in each group is ${}^{M}/_{T}$, which approximately is $T(0) = T(1) \cdots = T(t-1)$, $T(M)$ can be expressed as:

$$T(M) = tT(t-1) = tT(t-2) = \cdots = tT(0) \tag{26}$$

The knowledge of probability and mathematical statistics can prove that the overall performance of the algorithm is optimal when the number of slots consumed per packet is the smallest and the data transmission is the least.

IV. THE SIMULATION AND THE ANALYSIS

In this paper, the simulation experiment is carried out from two aspects; one is the simulation on the result of clustering center and the times of the iteration, which is obtained through FCMBMD, FCM and KFCM algorithm; the other is the simulation on checking the algorithm based on optimal FCM tag grouping.

A. THE SIMULATION EXPERIMENT OF FCMBMD

FCMBMD algorithm, together with FCM algorithm, KFCM algorithm, uses machine learning data sets on the official website of the California University to perform the contrast experiment. All algorithms use MATLAB tools and the MATLAB official interface documents to write the clustering algorithm of FCM, KFCM and FCMBMD and to perform the simulation experiment. To make the contrast analysis of simulation results easy, some small data sets of IRIS and Balance Scale are selected, which is because the properties of these data sets are small relatively. ISIR data is divided into four dimensions of data samples, including 150 data in three different categories, and 50 samples in each category. In the experiment, FCM and KFCM adopt Gaussian function, while FCMBMD uses density function and Mahalanobis distance. The experiment mainly simulates the result of clustering center, the times of iterations, the convergence of iteration and so on.

1) COMPARISON OF THE RESULTS OF CLUSTERING CENTER

The data of clustering center is tested according to the Balance Scale. In accordance with the relevant references, there are three clustering center values in total, and their standard actual values respectively are $V_1 = (3.52, 4.25, 2.36, 3.12)$, $V_2 = (2.18, 2.98, 3.75, 4.85)$ and $V_3 = (3.42, 4.18, 4.21, 3.37)$. FCM, KFC and FCMBMD

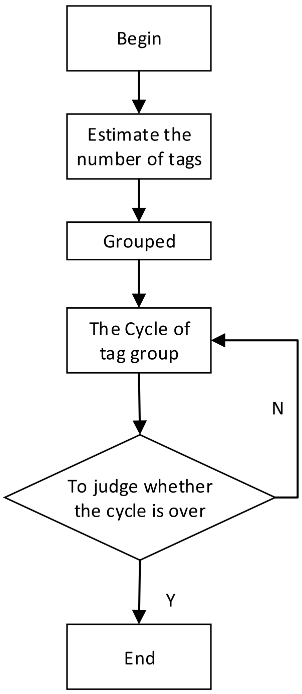

FIGURE 5. Flow chart of grouping algorithm.

TABLE 1. The clustering center results by different clustering algorithm.

Algorithm name	Cluster centers
FCM	$V_1 = (4.03, 3.65, 2.80, 3.68)$ $V_2 = (2.48, 3.65, 4.85, 4.42)$ $V_3 = (2.78, 3.66, 3.57, 2.46)$
KFCM	$V_1 = (3.78, 4.37, 2.75, 3.25)$ $V_2 = (1.88, 3.37, 3.71, 4.66)$ $V_3 = (3.61, 4.35, 4.17, 3.31)$
FCMBMD	$V_1 = (3.46, 4.27, 2.20, 3.29)$ $V_2 = (1.98, 2.86, 3.72, 4.79)$ $V_3 = (3.40, 4.07, 4.18, 3.35)$

algorithm will be used for clustering analysis of the BS data set respectively. The results are shown in the following table.

From the table above, a conclusion can be obtained that the clustering center gotten through the three kinds of fuzzy clustering analysis algorithm is located near the actual center of the standard data. However, the result gotten through the optimal algorithm based on density function and Mahalanobis distance is closer to the actual center.

2) COMPARISON OF THE TIMES OF ITERATIONS AND CONVERGENCE

The precision rate of clustering results only reflects one aspect of the algorithm's performance, so the degree of its precision rate does not always mean that it is the optimal algorithm. In terms of clustering algorithm, the iterative computation efficiency of the objective function and the convergence speed of the objective function are also important evaluation indexes. In the process of RFID tag identification, under the situation that a lot of tag data to be the large identified will be grouped, the iteration times of the algorithm and the convergence of the algorithm need to be considered. Otherwise, the algorithm could not meet the requirements.

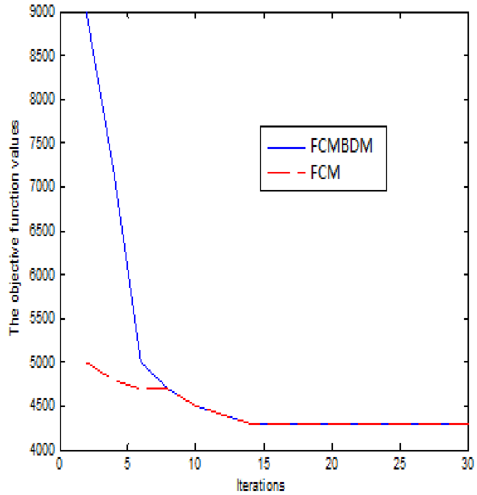

FIGURE 6. Comparison of the convergence rate of the Balance Scale in FCMBDM and FCM.

In the convergence simulation experiment, data attributes and data categories of samples, which are the average values after 30 iteration tests in the Balance Scale and the IRIS data set, are completely different. The contrast algorithms are FCMBD and KFCM algorithm. For the same data set, when the times of iterations are decreased gradually, the change curve on the value of objective function in two kinds of algorithms is shown in Figures 6-7.

The results of the experiment show that: no matter for what sample data, although the value of the objective function is very close in the FCMBD algorithm and the KFCM algorithm when algorithm ends, the convergence rate in FCMBD algorithm is obviously faster than that in other two algorithms, and the times of iterations are less than that in other two algorithms. Thus, FCMBD algorithm is better than KFCM algorithm in stability and convergence.

B. OPTIMAL SIMULATION AND ANALYSIS BASED ON TAG GROUPING

In order to better verify a new performance of RFID anti-collision algorithm based on tag grouping in the RFID for IOT, two simulation experiments will be performed in the paper.

1) ANALYSIS THE ACCURACY RATE OF CONTRAST ANALYSIS

The simulation software used by this paper is MATLAB, the CPU configured by the computer is the Inter dual-core

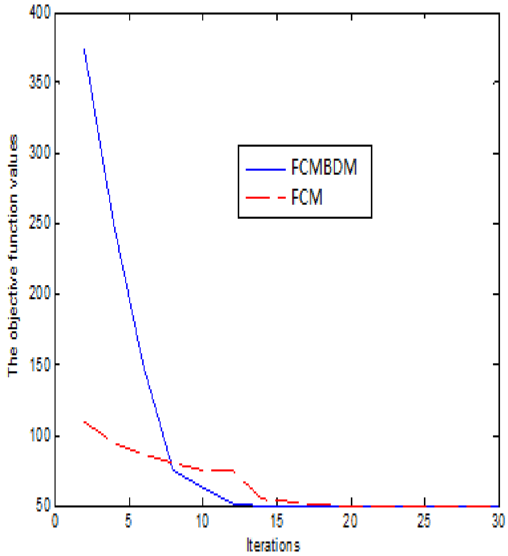

FIGURE 7. Comparison of the convergence of the IRIS data in FCMBDM and FCM.

2.66GHz CPU, with RAM of 3GB. As to the algorithm employed by the paper, the 16-bit CRC code is used to realize tag grouping. So, at the time of simulation, 1,000 groups of 16-bit binary data are generated at random with the MATLAB software to simulate the CRC code. In order to evaluate the performance of adaptive frame slotted algorithm based on density function and Fuzzy c-means dynamic grouping of Mahalanobis distance, we mainly study two indexes: success rate of tag identification and time consumption of the algorithm, and compare it with the dynamic binary tree algorithm and dynamic ALOHA algorithm. Where, accuracy rate of identification = the number of tags successfully identified/ total number of tags. (Specific to Problem 4 of the first reviewer) Since it is the simulation environment, time consumption of the algorithm could not be calculated actually, it is replaced with total time slot number. The greater the total time slot number, the longer the algorithm time will be consumed. Tag quantity will be started from 0, increased progressively by 100 in sequence, till up to 1000. The simulation results are respectively shown in Figure 8 and Figure 9.

Figure 8 shows obviously that, in the beginning, under the condition of small tag quantity, the success rates of identification of the 3 kinds of algorithms are almost identical, which are high. With the increase of the tag quantity, the success rates of identification of the 3 kinds of algorithms begin to descend. However, the descending process of algorithm adopted by this paper is relatively gentle, when the tag quantity reaches the maximum value 1000, the success rate of identification remains 92.4%, and the average success rate of identification of the whole process reaches up to 96.8%. Nevertheless, the success rates of identification of the other two kinds of algorithms descend faster, when the tag quantity reaches the maximum value, their success rates are respectively 71.5% and 63.2%. The reason for this is, according to the algorithm employed by this paper, during grouping, each tag is allocated with a unique serial number, greatly lowering the occurrence probability of collision. Therefore, the success rate of identification will always maintain a higher level.

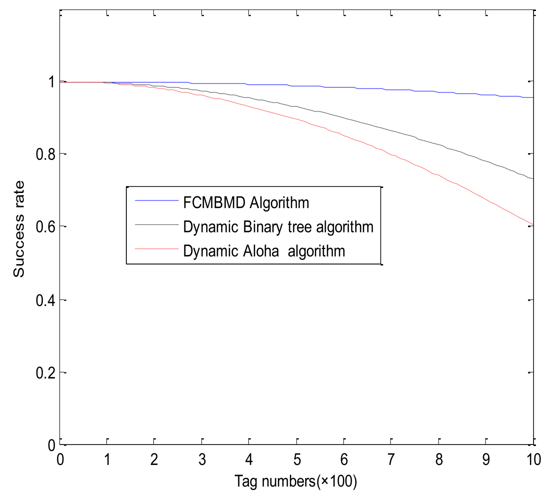

FIGURE 8. Comparison of accuracy rate.

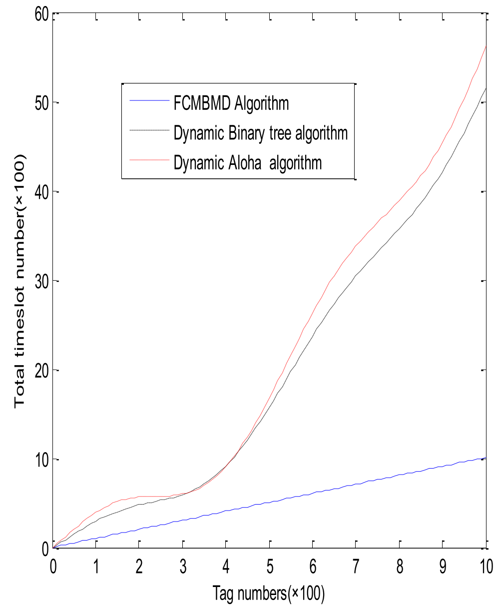

FIGURE 9. Comparison of algorithm time consumption.

The simulation results of Figure 9 show that, time consumption of the algorithm in this paper is far less than that of the dynamic binary tree algorithm and dynamic ALOHA algorithm. Since with these two traditional algorithms, collision could not be well reduced, a large number of extra time slots are required to guarantee the identification of all the tags. Therefore, the total number of time slots is far higher than the total number of tags, and the throughput for the systems is extremely poor. While the algorithm employed by this paper has guaranteed the equivalent of the total number of tags to the frame length, namely the time-slot number of each group. Therefore, even if the first-time polling fails to identify all the tags, it can fulfill the identification via

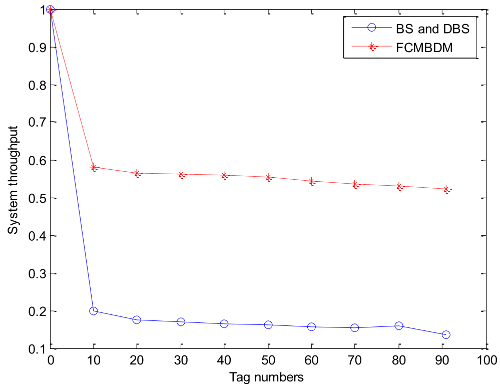

FIGURE 10. Comparison of the throughput in BLBO and in BS, DBS algorithm.

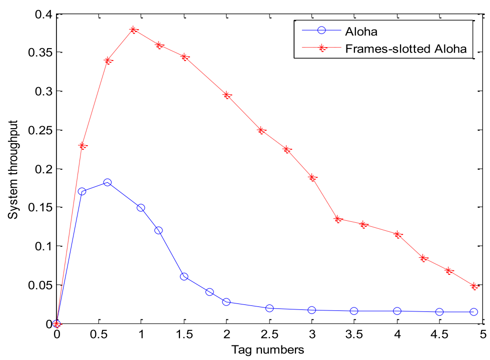

FIGURE 11. Comparison of the throughput in BLBO algorithm other algorithms.

additional 1~3 times or so, and the total time-slot number will not be too much different from the total number of tags. Thus, the time consumption of the algorithm in this paper is fully assured to be not too long.

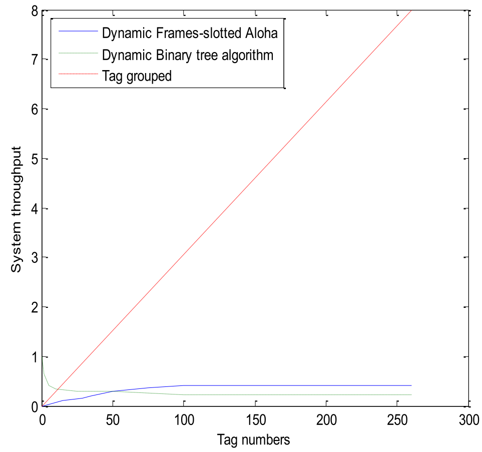

FIGURE 12. Comparison of throughput rate in different algorithms.

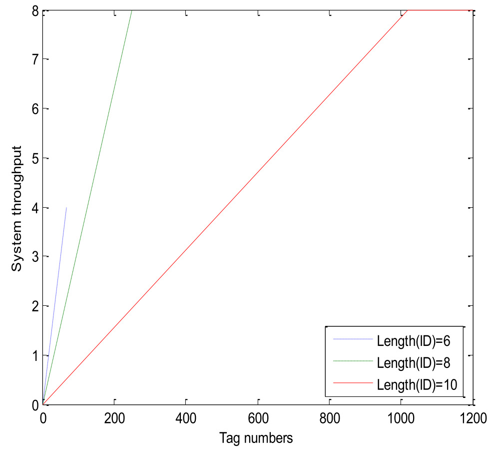

FIGURE 13. Comparison of throughput rate in different length of ID in tag grouping.

2) THE ANALYSIS OF THROUGHPUT RATE AND TIME SLOT UTILIZATION

In the RFID for IOT, suppose that tag data volume changes between 50-120 dynamically, tag ID shows normal distribution, the fixed length of the label is 128 bit, then take the average simulation value from 30 times' simulation, and analyze performance index from throughput, throughput rate and utilization rate of slot [38], [39].

Throughput is an important index to measure the degree of RFID tag's performance in the process of identification. Throughput rate is a measure of RFID data transfer rate, and it is the average time that tags use to complete communication in a period of time. By comparing the throughput rate obtained respectively by the algorithm referred in the paper with the throughout rate in DBS algorithm and Frame Slotted ALOHA algorithm, it can be concluded that time slot utilization rate is the ratio of not empty slots and total time slots. In the RFID for IOT, the time slot of anti-collision algorithm is divided into collision time slot, idle time slot and recognition time slot. Generally, only recognition time slot can identify data. Two simulations are carried out in the experiment. One group is the simulation on the time slot utilization rate in different length of ID in different tag grouping algorithm. The other group is comparison of tag grouping algorithm and some common algorithms (i.e. dynamic frame time slot algorithm (dynamic ALOHA) and dynamic binary search algorithm). The simulation result is shown in the Figure 10-14.

It can be seen from the Figure 10, the throughput obtained by the algorithm proposed in this paper is close to 50%, while the throughput of the DBS algorithm is not higher than 30% and the throughput decreases gradually with the increase of tag numbers. Therefore, it can be concluded that the throughput obtained by the algorithm proposed in this paper is higher than that in the BS search algorithm.

Through the analysis of the data in Figure 10 and Figure 11, the throughput in Frame-Slotted ALOHA algorithm is not higher than 40%, that is to say, when G = 1, the throughput obtains the maximum which is only 36.8%. After giving a theory and simulation analysis of ALOHA algorithm according to the existing data, the throughput in the pure ALOHA algorithm just gets to the maximum when G = 0.5, and the

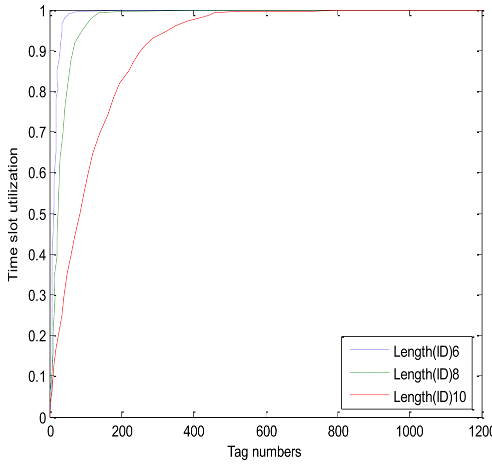

FIGURE 14. Time slot utilization in different ID length in tag grouping algorithm.

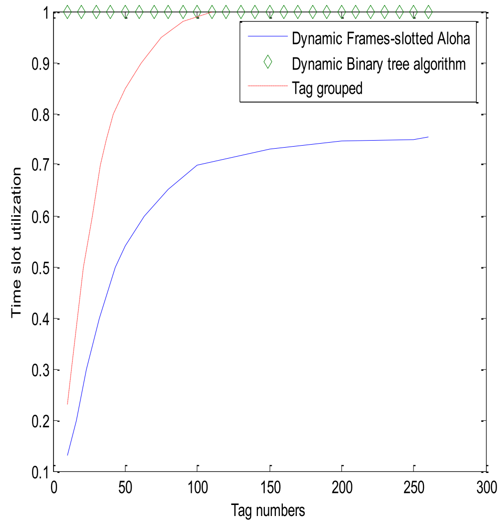

FIGURE 15. Comparison of time slot utilization rate in different algorithm.

maximum value is only 18.4%. Combining Figure 10 and Figure 11, it is clear that the throughput in dynamic grouping algorithm is higher than that in DBS and Frame Slotted ALOHA algorithm.

From Figure 12, it can also be seen that, in tag grouping algorithm, the longer is the length of the ID code, the higher the throughput will be. When the length of ID keeps the same, throughput rate will be increased with the increasing of the tag number. When the tag number reaches the maximum capacity of the system, throughput rate reaches the maximum of 2t, which shows that a single time slot can transmit 2t tags successfully at most. From the throughput rate in Figure 13, the advantages of binary search algorithm are obvious when the number of tags is small. With the increasing of tags, throughput rate decreases and tends to be stable remaining at around 0.3. It explains that in multiple tags identification in this algorithm, although no empty slots exist in the process, most of them are collision time slots that could not communicate successfully, and there are not too many time slots that can succeed in communication. In dynamic frame time slot algorithm, the throughput rate will rise with the increase of tags and remains at around 40%. It is also because collision time slots in the algorithm could not be used whose throughput rate is low. While in tag grouping algorithm, the more the tags are, the superior the performance of throughout rate are. The throughout rate in tag grouping algorithm will increase with the increase of tags, which benefits from taking advantage of its collision time slots. The more the tags are, the higher the utilization rate of the collision time slots will be, and thus more tags will be identified, and finally the throughput rate will be higher. In general, where there are more labels, the algorithm based on tags grouping is better than the other two algorithms.

It can be seen from Figure 14 that when the length of the tag ID is fixed, the time slot utilization rate is also increased with the increasing of tag number, and its maximum utilization rate reaches to 100%. The data in Figure 15 shows that time slot utilization rate is not exactly the same in different algorithms, in which the rate in the binary search algorithm is the highest, the rate in tag grouping algorithm is the second, and the rate in dynamic frame time slot algorithm is the lowest. But with the gradual increase of tag numbers, time slot utilization rate of the algorithm is higher.

In conclusion, the adaptive time slot anti-collision algorithm based on the fuzzy c mean value dynamic grouping mentioned in this paper is far better than the traditional binary tree algorithm and ALOHA algorithm no matter in terms of the success rate of identification or time consumption of algorithm, with superior overall performance.

V. CONCLUSIONS

In the RFID for IOT, anti-collision algorithm is a key research problem of the RFID field. The paper has proposed an RFID anti-collision algorithm based on tag grouping, which has effectively solved the issue of tag grouping and unique tag number allocation in the group by using the fuzzy c mean value cluster method. Thus, dynamic adjustment of frame slotted of the reader is realized. Meanwhile, the paper has also proved that the algorithm in the paper has high throughput of the systems. The experimental results also indicate clearly that, algorithm performance in this paper is superior to the traditional anti-collision algorithm. In the future we will focus on the research of the optimization of the tag grouping process, for further improvement of the execution efficiency of the algorithm.

COMPETING INTERESTS

The authors declare that they have no competing interests.

AUTHOR CONTRIBUTIONS

Bai Zhi and Wang Sainan co-conceived, co-designed and performed the data collection experiments and co-wrote the

paper under the guidance of Prof He Yigang; Sanesy Kumcr Shokla provided data for comparing technologies.

CONFLICTS OF INTEREST

The authors declare no conflict of interest

REFERENCES

[1] H. Wu, Y. Zeng, J. Feng, and Y. Gu, "Binary tree slotted ALOHA for passive RFID tag anticollision," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 1, pp. 19–31, Jan. 2013.
[2] H. J. Duan, H. Wu, Y. Zeng, and Y. Chen, "LS channel estimation and signal separation for UHF RFID tag collision recovery on the physical layer," *Sensors*, vol. 16, no. 4, p. 442, 2016.
[3] A. Moreno, "Intelligent van for the distribution of pharmaceutical drugs," *Sensors*, vol. 12, no. 5, pp. 6587–6609, Dec. 2012.
[4] J.-Q. Li, S.-P. Zhang, L. Yang, X.-H. Fu, Z. Ming, and G. Feng, "Accurate RFID localization algorithm with particle swarm optimization based on reference tags," *J. Intell. Fuzzy Syst.*, vol. 31, no. 5, pp. 2697–2706, 2016.
[5] J. Li, L. Huang, Y. Zhou, S. He, and Z. Ming, "Computation partitioning for mobile cloud computing in a big data environment," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 2009–2018, Aug. 2017.
[6] J.-Q. Li, S.-Q. He, Z. Ming, and S. Cai, "An intelligent wireless sensor networks system with multiple servers communication," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 8, p. 6, 2015.
[7] K. Cui and T. T. Zhao, "Unsaturated dynamic constitutive model under cyclic loading," *Cluster Comput.*, vol. 20, no. 4, pp. 2869–2879, 2017.
[8] L. Xie, Y.-F. Yin, X. Chen, S.-L. Lu, and D.-X. Chen, "RFID data management: Algorithms, protocols and performance evaluation," *Chin. J. Comput.*, vol. 36, no. 3, pp. 457–470, 2013.
[9] D. R. Hush and C. Wood, "Analysis of tree algorithms for RFID arbitration," in *Proc. IEEE Int. Symp. Inf. Theory*, Cambridge, MA, USA, Aug. 1998, p. 107.
[10] D. Huifang, L. Jinqiao, and D. Chunhui, "RFID multi-tags anti-collision algorithm with adaptive Q leading to the maximum throughput," in *Proc. 3rd Pacific–Asia Conf. Web Mining Web-Based Appl.*, Guilin, China, 2010, pp. 166–169.
[11] H. Wan and Y. Z. Yang, "The research and improvement of anti-collision algorithms of RFID," *Microcomput. Inf.*, vol. 25, no. 3, pp. 230–232, 2009.
[12] W. Yong, L. Qing, W. Lei, and S. Hao, "Study on anti-collision algorithms in UHF RFID system," *Instrum. Technol.*, vol. 2, pp. 16–18, 2008.
[13] J.-Q. Deng and H.-F. Liu, "Binary-matrix search in anti-collision of RFID system," *Microcomput. Inf.*, vol. 25, no. 10, pp. 4–5, 2009.
[14] W. Wei *et al.*, "Gradient-driven parking navigation using a continuous information potential field based on wireless sensor network," *Inf. Sci.*, vol. 408, pp. 100–114, Oct. 2017.
[15] G. Calis, S. Deora, N. Li, B. Becerik-Gerber, and B. Krishnamachari, "Assessment of WSN and RFID technologies for real-time occupancy information," in *Proc. 28th Int. Symp. Automat. Robot. Construct. (ISARC)*, Seoul, South Korea, Jun./Jul. 2011.
[16] J. Fernández-Salmerón, A. Rivadeneyra, F. Martínez-Martí, L. F. Capitán-Vallvey, A. J. Palma, and M. A. Carvajal, "Passive UHF RFID tag with multiple sensing capabilities," *Sensors.*, vol. 15, no. 15, pp. 26769–26782, 2015.
[17] C. Galiotto, N. Marchetti, N. Prasad, and R. Prasad, "Low access delay anti-collision algorithm for readers in passive RFID systems," *Wireless Pers. Commun.*, vol. 64, no. 1, pp. 169–183, 2012.
[18] S. S. Rao, E. G. Rajan, and K. Lalkishore, "ASAF ALOHA protocol for dense RFID systems," *Wireless Pers. Commun.*, vol. 66, no. 4, pp. 667–681, 2012.
[19] M. Shao, X. Jin, and L. Jin, "An improved dynamic adaptive multi-tree search anti-collision algorithm based on RFID," in *Proc. Int. Conf. Data Sci. Adv. Anal.*, Shanghai, China, Oct./Nov. 2014, pp. 72–75.
[20] M. Djeddou, R. Khelladi, and M. Benssalah, "Improved RFID anti-collision algorithm," *Int. J. Electron. Commun.*, vol. 67, no. 3, pp. 256–262, 2013.
[21] J. Sampe and M. Othman, "Fast detection anti-collision algorithm for RFID system implemented on-chip," *J. Appl. Sci.*, vol. 8, no. 7, pp. 1315–1319, 2008.
[22] K. C. Shin, S. Bo Park, and G. S. Jo, "Enhanced TDMA based anti-collision algorithm with a dynamic frame size adjustment strategy for mobile RFID readers," *Sensors*, vol. 9, no. 2, pp. 845–858, 2009.
[23] L. Zhang, W. Xiang, and X. Tang, "An adaptive anti-collision protocol for large-scale RFID tag identification," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 601–604, Dec. 2014.
[24] Z. Bai, Y. He, and S. Wang, "Research of RFID tag anti-collision algorithm based on binary tree," *J. Netw.*, vol. 9, no. 9, pp. 2543–2548, 2014.
[25] J. Cao, L. Shu, and Z. Lu, "Synchronous dynamic adjusting: An anti-collision algorithm for an RF-UCard system," *Netw. Syst. Sci.*, vol. 2, no. 2, pp. 8–20, 2009.
[26] Z. Li and C. He, "Optimal scheduling-based RFID reader-to-reader collision avoidance method using artificial immune system," *Appl. Soft Comput.*, vol. 13, no. 3, pp. 2557–2568, 2013.
[27] K. Shin and W. Song, "RAC-multi: Reader anti-collision algorithm for multichannel mobile RFID networks," *Sensors.*, vol. 10, no. 1, pp. 84–96, 2010.
[28] H. Safa, W. El-Hajj, and C. Meguerditchian, "A distributed multi-channel reader anti-collision algorithm for RFID environments," *J. Parallel Distrib. Comput.*, vol. 64, pp. 44–56, Jun. 2015.
[29] 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. 268, 2012.
[30] J. Vales-Alonso, F. J. Parrado-García, and J. J. Alcaraz, "Performance analysis of optimal schedulers in single channel dense radio frequency identification environments," *EURASIP J. Embedded Syst.*, vol. 2013, no. 1, pp. 16–22, 2013.
[31] R. Xu and D. Wunsch, II, "Survey of clustering algorithms," *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 645–678, May 2005.
[32] C. H. Lin, X. D. Li, and J. Jin, "DNS traffic anomaly detection based on W-Kmeans algorithm," *Comput. Eng. Des.*, vol. 34, pp. 2104–2108, 2013.
[33] M. Berget, B.-H. Mevik, and T. Næs, "New modifications and applications of fuzzy C-means methodology," *Comput. Statist. Data Anal.*, vol. 52, no. 5, pp. 2403–2418, 2008.
[34] C. Y. Wang and C. C. Lee, "A grouping-based dynamic framed slotted ALOHA anti-collision method with fine groups in RFID systems," in *Proc. 5th Int. Conf. Future Inf. Technol.*, Busan, Korea, 2010, pp. 1–5.
[35] Z. Li, J. Li, and C. He, "Artificial immune network-based anti-collision algorithm for dense RFID readers," *J. Parallel Distrib. Comput.*, vol. 41, no. 10, pp. 4798–4810, 2014.
[36] F. Z. Zhang, J. F. Chang, and Q. Q. Zhou, "Context-aware recommendation algorithm based on fuzzy C-means clustering," *J. Comput. Res. Develop.*, vol. 50, pp. 2185–2194, 2013.
[37] L. Li, H. Q. Lou, and Y. L. Yu, "An improved fuzzy C-means clustering algorithm," *Comput. Technol. Develop.*, vol. 19, pp. 71–73, 2009.
[38] A. Yang *et al.*, "Optimum surface roughness prediction for titanium alloy by adopting response surface methodology," *Results Phys.*, vol. 7, pp. 1046–1050, Jan. 2017.
[39] K. Cui, W.-H. Yang, and H.-Y. Gou, "Experimental research and finite element analysis on the dynamic characteristics of concrete steel bridges with multi-cracks," *J. Vibroeng.*, vol. 19., no. 6, pp. 4198–4209, 2017.
[40] Y. Sun, H. Qiang, X. Mei, and Y. Teng, "Modified repetitive learning control with unidirectional control input for uncertain nonlinear systems," *Neural Comput. Appl.*, pp. 1–10, Apr. 2017, doi: 10.1007/s00521-017-2983-y.

BAI ZHI was born in China in 1975. He received the M.S. and Ph.D. degrees from Hunan University, China, in 2006.

His current research interests include radio-frequency identification chip circuit design, anti-collision analysis and optimization, and software engineering. He received the Society Best Symposium Paper Award in 2011.

WANG SAINAN was born in Hunan, China, in 1980. She received the M.S. degree in mathematics from the Changsha University of Science and Technology in 2010.

Her main research interests include probability theory, mathematical statistics, and mathematical modeling.

HE YIGANG was born in Xinshao County, Hunan, China, in 1966. He is currently a Professor, a Doctoral Tutor, and the Dean of the School of Electrical Engineering and Automation, Hefei University of Technology. He was a National Science Fund for Distinguished Young winner.

His current research interests include electrical science and engineering technology, automatic test and diagnostic equipment, high-speed low-voltage low-consumption integrated circuit, intelligent and real-time information processing, smart grid, maximum scale circuit, mixed-signal circuit, and system test and diagnosis.