

Recovery of Collided RFID Tags With Frequency Drift on Physical Layer

Junzhi Li, Haifeng Wu, and Yu Zeng

Abstract—In a passive ultra-high frequency (UHF) radio frequency identification (RFID) system, the recovery of collided tag signals on a physical layer can enhance identification efficiency. However, frequency drift is very common in UHF RFID systems, and will have an influence on the recovery on the physical layer. To address the problem of recovery with the frequency drift, this paper adopts a radial basis function (RBF) network to separate the collision signals, and decode the signals via FM0 to recovery collided RFID tags. Numerical results show that the method in this paper has better performance of symbol error rate (SER) and separation efficiency compared to conventional methods when frequency drift occurs.

Index Terms—Frequency drift, radial basis function (RBF), radio frequency identification (RFID), separation efficiency, tag collision.

I. INTRODUCTION

In a passive ultra-high frequency (UHF) radio frequency identification (RFID) system, collision will happen when tags transmit their signals to a reader simultaneously [1]–[4]. Most of traditional anti-collision methods are on a media access control (MAC) layer [5], [6], but the collision signals could be separated and recovered on a physical layer to enhance tag identification efficiency [7]–[20]. For UHF RFID system, separation methods may be different from traditional ones. First, the tag signals symbol period is time-variant due to its frequency drift. In the EPC C1 Gen2 standard [21], the symbol frequency drifts within the range of nominal frequency, $\pm 22\%$. The separation has to be performed under an asynchronous mode. Second, the separation could be considered as a cluster problem. For supervised cluster methods, good channel information is required [16] but difficult to be estimated with the frequency drift. On the other hand, unsupervised methods will have higher complexity.

The constellation mapping (CM) algorithm [17] can adopt an unsupervised cluster method to enhance the accuracy of the channel estimation, but the computational complexity

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

This work was supported by the National Natural Science Foundation of China (61762093), the 17th Batches of Young and Middle-aged Leaders in Academic and Technical Reserved Talents Project of Yunnan Province (2014HB019), the Key Applied and Basic Research Foundation of Yunnan Province (2018FA036), and the Program for Innovative Research Team (in Science and Technology) in University of Yunnan Province. Recommended by Associate Editor Giovanni Pau. (*Corresponding author: Haifeng Wu*)

Citation: J. Z. Li, H. F. Wu, and Y. Zeng, “Recovery of collided rfid tags with frequency drift on physical layer,” *IEEE/CAA J. Autom. Sinica*, DOI: 10.1109/JAS.2019.1911720.

All the authors are with the School of Electrical and Information Technology, Yunnan Minzu University, Kunming 650500, China (e-mail: 1050643124@qq.com; whf5469@gmail.com; yv.zeng@gmail.com).

Digital Object Identifier 10.1109/JAS.2019.1911720

increases with the number of the collided tags. The single antenna zero forcing (SAZF) algorithm [19] is a supervised method and needs the channel information to determine the centers of clusters. However, the SAZF can estimate the channel with only two collided tags. Collision signal separation in the successive interference cancellation (SIC) technique [18] and the least-square channel estimate (LCE) [20] also requires channel information, which is estimated via the period and delay of a preamble. Likewise, the frequency drift leads to a time-variant period and poor channel estimation. Besides, some recent algorithms in [9], [22] also focus their works on physical-layer tag collision separation in an RFID system. However, [22] does not take into account the effect of the time-variant period, i.e., the frequency drift on the separation. Reference [9] focuses more on the use of compression sensing to monitor the missing tags in real time, and still does not discuss the frequency drift more.

Another concern this paper addresses is signal decoding. Matched filter [19] and Viterbi algorithm [18] are popular decoding algorithms for an RFID system. They are also proposed to decode collided RFID signals. Due to the frequency drift, however, an invariant waveform of the matched filter is difficult to guarantee decoding under the condition of maximum signal to noise ratio (SNR). Viterbi algorithm is a searching method with fewer searches for an optimal path. The algorithm may also be introduced to decode RFID collided tag signals. As we know, Viterbi needs to find the minimum distance between each possible path and an expected path. When frequency drift occurs, the distance between an optimal path and an expected path may not be a minimum value. This will result in poor performance of symbol error rate (SER). Although [23] considers signal decoding, it unfortunately does not discuss FM0 or Miller code and is difficult to apply to an RFID system for EPC C1 Gen2, where FM0 and Miller has been specified as a UHF RFID coding standard [8], [14], [22]–[25]. The methods in [8], [23], [25] consider FM0 decoding directly from collision signals and can be applied to the frequency shift among tags, but it assumes that the frequency drift does not occur.

In this paper, an adaptive radial basis function (RBF) neural network [26] is introduced to separate the collision tag signals in the UHF RFID system. Owing to a prior known preamble, the introduced RBF could complete a supervised cluster for the collision signals well even though the exact channel information is not known. Then, an FM0 decoding technique in our previous work in [27] is adopted to cancel separated errors. The technique could detect edge transitions to decode

the separated signals regardless of the frequency drift. From numerical results, our method for the three collided tags performs better than conventional methods when frequency drift occurs.

In the rest of this paper, we give the system model and the proposed separation algorithm in Section II. In Section III, we describe a decoding method for frequency drift environments. We give simulation results to show the performance of the separation in Section IV. Section V is our discussion. The last final section draws a conclusion.

II. SYSTEM MODEL AND COLLISION SEPARATION WITH FREQUENCY DRIFT

A. System Model

When a reader identifies UHF RFID tags, it sends a continuous carrier signal with energy for the RFID tags [7], [22]. Given I transmitting tags in a slot, the reader will receive the superposition of their signals and then down convert the receiving signals to baseband. Hence, the complex-valued baseband signal at the receiving antenna is [18]–[20]

$$\bar{z}(t) = \sum_{i=1}^{I-1} h_i e_i(t) + \zeta(t) + \Xi \quad (1)$$

where h_i is assumed to be a flat fading linear time invariant channel in a cycle of a tag ID identification [18], [19]. $\zeta(t)$ is an additive white Gaussian noise added at the reader. $e_i(t) = \sum_{k=0}^{K-1} \pi_{i,k} g_{p_i,k}(t - kp_{i,k} - q_i)$ realizes an on-off key. $p_{i,k}$ and q_i represent the symbol period and delay respectively, where generally $p_{i_1,k} \neq p_{i_2,k}$, when $i_1 \neq i_2$. $\pi_{i,k} \in \{0, 1\}$ denotes the transmitted symbol [18]. k is the length of symbol block [19], [20]. $g_{p_i,k}(t)$ denotes the pulse modulation signal.

In the EPC C1 Gen2 standard, a reader can only discover the carrier leakage when all tags are in their absorb state. We can utilize this period to estimate the carrier leakage ff [18], [20]. Hence, we make

$$z(t) = \bar{z}(t) - \Xi.$$

B. RBF Separation Algorithm

From the analysis in Section 1, the performance of collision signal separation would degrade due to the frequency drift. Here, we introduce a RBF network against the frequency drift. The basic idea of a RBF neural network is to map a nonlinear cluster in a low dimensional space to a linear cluster in a high dimensional space. Fig. 1 shows the architecture of the RBF network for the separation, where an input vector $X(n)$ is expressed as

$$x(n) = [x_1(n), x_2(n)]^T = [\text{Im}[z(t_n)], \text{Re}[z(t_n)]]^T \quad (2)$$

in which, $t_n = n\Delta t$ where n denotes a sampling index and Δt denotes a sampling period. The function $F(\cdot)$ in the RBF is defined as

$$F[x(n)] = \sum_{m=1}^M w_m \varphi_m[x(n)] \quad (3)$$

where w_m denotes a weight, $M = 2^I$ is the number of centers,

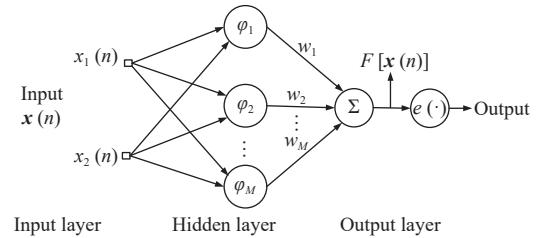


Fig. 1. An architecture of RBF network for separation.

and I denotes the number of collided tags and could be estimated through the method in [8], [17]. If the number of tags is unknown, excessive chosen centers will lead to higher computational complexity and too few may not be sufficient to complete a training network. The details of the chosen centers will be described in Section II-C. A Gauss function is chosen as the radial basis function φ_m and shown as

$$\varphi_m[x(n)] = \exp\left(-\frac{\|x(n) - c_m\|^2}{2\sigma_m^2}\right) \quad (4)$$

where c_m denotes the vector of the m th clusters central coordinate in RBF, and σ_m is the width of the Gauss function centered on c_m . The activation function $e(\cdot)$ is defined as

$$e[F[x(n)]] = \begin{cases} 1, & F[x(n)] \geq 0 \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

All tags preamble signals in front of IDs are the same and their value is known to a reader. The preambles period can be estimated via algorithms in [18], [20]. Thus, the preamble information could train the RBF networks center point c_m and weight w_m . The algorithms in [18], [20] do not consider the time-variant symbol period and the acquired preamble will have some errors. Hence, a RBF trained from preambles with errors may not be very accurate. After separation, however, our method could cancel the separating errors via FM0 decoding. The details are included in Section III.

Next, we will describe training the RBF network. If $n \in [1, N]$ and N denotes preamble signal maximum sampling value, the center point can be obtained as $c_m = \hat{c}_m(N)$. We calculate c_m by an adaptive method as

$$\begin{cases} \hat{c}_m(n) = \hat{c}_m(n-1) + \mu(x(n) - \hat{c}_m(n-1)), & \text{if } m = m' \\ \hat{c}_m(n) = \hat{c}_m(n-1), & \text{if } m \neq m' \end{cases} \quad (6)$$

where μ denotes an update rate and m' is

$$m' = \arg \min_{1 \leq m \leq M} \|x(n) - c_m(n-1)\|^2. \quad (7)$$

An initial value $c_m(0)$ could be obtained via an estimate of a histogram projection, seen in Section II-D. Different from the direct channel estimation in SAZF and LCE, (6) would obtain the centers via the adaptive training. σ_m can be estimated by

$$\sigma_m = \sqrt{\frac{\sum_{\tilde{n}_m=1}^{\tilde{N}_m} \|x'_m(\tilde{n}_m) - c_m\|^2}{\tilde{N}_m}} \quad (8)$$

where \tilde{N}_m denotes the number of sampling signals which belong to the m th cluster, and $x'_m(\tilde{n}_m)$ denotes the \tilde{n}_m th sampling signal in the m th cluster. Finally, a recursive least squares algorithm for weight training is given by

$$\hat{w}(n) = \hat{w}(n-1) + R(n)\Phi(n)\alpha(n) \quad (9)$$

where an initial value $w_m(0) = 0, m = 1, 2, \dots, M$

$$\hat{w}(n) = [\hat{w}_1(n), \hat{w}_2(n), \dots, \hat{w}_M(n)]^T \quad (10)$$

$$R(n) = R(n-1) - \frac{R(n-1)\Phi(n)\Phi^T(n)R(n-1)}{1 + \Phi^T(n)R(n-1)\Phi(n)} \quad (11)$$

$$\Phi(n) = [\varphi_1, \varphi_2, \dots, \varphi_M]^T. \quad (12)$$

The priority estimation error $\alpha(n)$ is

$$\alpha(n) = d(n) - \hat{w}^T(n-1)\Phi(n) \quad (13)$$

where $d(n) \in \{0, 1\}$ is the expected output value of the network. The center points and weights in the RBF network can be obtained after training. Substituting the real and imaginary part of the collided signal as input data into (2)–(5), we can separate the collided signals. Note that the RBF network needs to be trained for every collided tag. An example for the separation of the three collided tags via the RBF is shown in Fig. 2, where the left part gives the separated signal waves and the right part gives classified lines in the I/Q planes. Signals on one side of the line are classified as 1 and on the other side are 0. The RBF network is a clustering method, where the power gains of channels h_i decides the position of the cluster centers. Similar to [6]–[9], RBF network also requires the power gains be different and more details would be discussed in Section II-E.

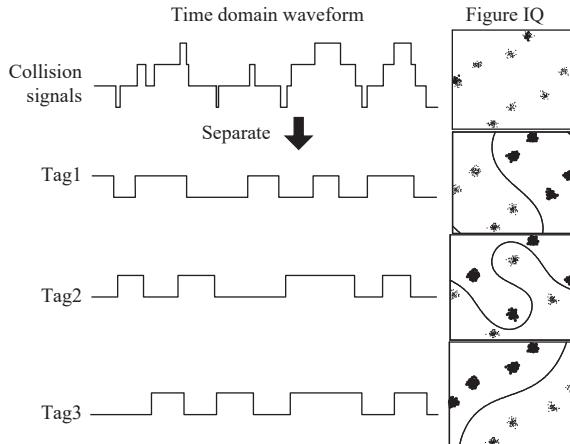


Fig. 2. RBF networks separate three collided tags.

Moreover, RBF is a supervised method, where the centers are trained via the preamble and the adaptive algorithm in (6). In fact, the symbol value of the preamble is already known since the EPC C1 Gen2 has specified it. What cannot be known is the symbol period in the preamble. How to acquire the period has been proposed in [18]. And, a previous work in [20] also introduced it. Of course, [18] and [20] do not consider the frequency drift. Thus, the preamble acquired by [18] and [20] will have some errors. However, the principle of the algorithm in [18] and [20] is to search a maximum value in the inner products of the mother functions and collided preambles, and searching is in a range of 22% that the EPC has specified. Thus, the maximum value for searching is in fact an average value of the symbol periods and will not be

beyond the range of 22%. Via an initial center value chosen by a method in [17] (the details can be seen in the Section II-D), then, our RBF network will adopt the adaptive algorithm in (6) to enhance the accuracy of center estimation. After separation by the RBF, our method can cancel the RBF separating errors via FM0 decoding. Even if there is a small amount of error in the training data, it will not have a great impact on the separated signals because it can also be corrected by our FM0 decoding, which ensures lower SER. Next, we will discuss other points which may have an impact on the performance of this proposed collision recovery.

C. The Number of Centers

From (3)–(12), where the number of centers and the number of tags will have the relationship $M = 2^I$, our algorithm must estimate the number of tags if we want to know how many centers should be used. The cluster-based methods in [8], [17]–[20] could be used to estimate the number of tags. For example, 3 gives the IQ plan of three collision tag signals. From the figure, the number of center points is 8, which can be obtained from $M = 2^3$. That is, the number of tags can determine the number of center points. The details of the RFID system setting in Fig. 3 can be see in Section IV-A.

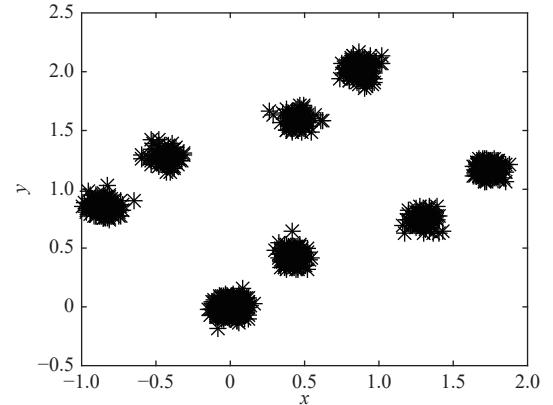


Fig. 3. Clusters for three collided tags in an I/Q plane.

Another method in [8] may also estimate the number of tags, which is to judge how many transition edges in a V signal [21] of a preamble. The V indicates an FM0 violation, which means that a phase inversion should have occurred but did not. An example is illustrated in Fig. 4, where the V signal signed in an oval has three edges and the number of tags is three.

Even if we do not know the number of tags, and hence do not know the actually number of centers, there will be only the following two cases.

1) When the number of chosen centers is more than the actual number of clusters, the network training can be completed. As shown in Fig. 5, the number of clusters is eight and the number of chosen centers marked by circles is fourteen. From the figure, a curve can effectively classify signals 0 and 1. That is, the signals on one side of the curve are 0 and on the other side are 1. The results are consistent with [27]. However, the complexity of network training will

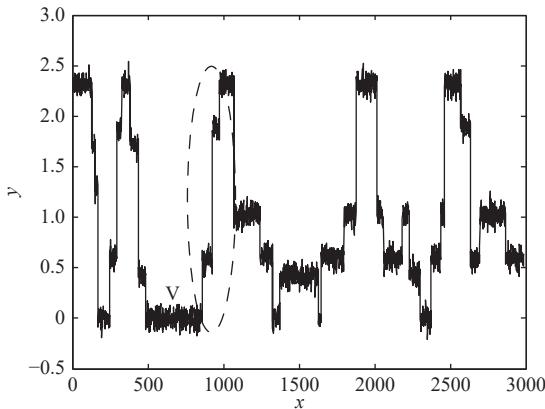


Fig. 4. Estimation for the number of tags via V signal in a preamble.

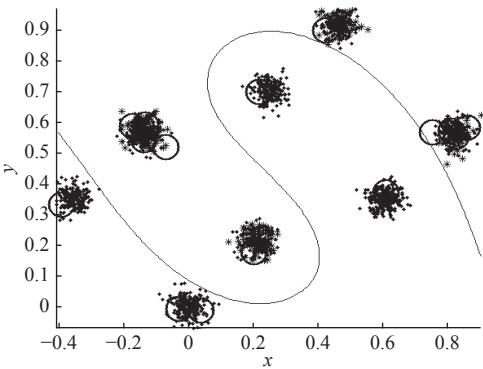


Fig. 5. More chosen centers than actual centers in classification.

increase due to excessive centers.

2) When the number of chosen centers is less than the number of clusters, the trained network sometimes cannot separate collided tags. An example is shown in Fig. 6, where the number of clusters is eight and the number of chosen centers is six. From the figure, a curve has crossed one of clusters.

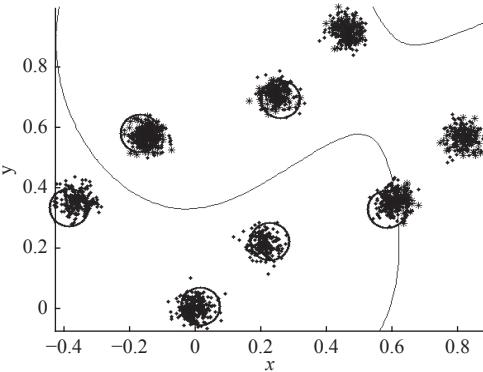


Fig. 6. Fewer chosen center than actual centers in classification.

From the analysis above, if we do not know the number of tags, we have to choose as many centers as possible despite this leading to a higher computational complexity. However, for a dynamic frame length protocol, the average number of

tags in a collision slot is about 2.33. Hence, we would consider the case for two or three collision tags in the protocol. It is not necessary to choose excessive centers which are greater than eight clusters.

D. Initial Value of Centers

Since we adopt an adaptive method in (6) to obtain the value of the centers, how we choose an initial value will be key for the estimation. One of the methods is to select a value from the result of conventional estimates, like [18]–[20]. Due to frequency drift, however, conventional estimates do not work well and thus the initial values from them may not be good candidates. Note that the centers of clusters are time-invariant even with the frequency drift, as long as the channel information is time-invariant. Therefore, we could adopt an unsupervised cluster method in [17], where the k -means algorithm is used to accurately estimate the centers. However, the unsupervised method has higher computational complexity.

Another simpler method is also proposed in [17]. Here, Fig. 7 gives the steps of the methods. First, let an I/Q plane be paved with grids. Then, the number of points falling into each grid are counted and plotted on a histogram. Finally, project the histogram to the x axis or y axis, and find each cluster peak. When we find the grid where the peak is located, we will take the center of the grid as the initial value. Via the adaptive iteration in (6), our RBF network could find more accurate centers.

E. Power Gains of Channels

Our algorithm is also related to the power gains of channels. Different kinds of channels may produce different performances. Our channel at least guarantees the two following conditions.

1) The power gains of channels should be time-invariant or slowly variant. Similar to [17], [19], [20], our algorithm is actually a cluster-based method. If the power gains change, the centers of the clusters will also change. This will have an influence on the performance of clustering and classification. Consider an UHF RFID system with link frequency 500 kHz and 96 bit-ID tags. The duration of transmitting a 96 bit ID is only about 0.192 ms. UHF RFID is a type of near-field communication system. A one to three-meter field is a usual communication distance. In this case, the condition that the channels during an identification cycle are flat-fading and time-invariant condition are guaranteed. It is this reason that this paper assume that the power gains of channels are invariant, like [17]–[20].

2) The power gains and orientation of channels should be different among tags. Consider the following special cases. If two or three tags channels are all identical, the original 22 and 23 clusters will decrease to 3 or 4 clusters, as shown in Figs. 8 and 9. In this case, it is difficult to separate the collided tags. Therefore, less difference of channels among tags will result in worse performance of separation, and [6]–[9] require that separation should be performed under different power gains of channels. In practice, the position and direction of tags may be different. Thus, the fading coefficient and orientation of channels will be different.

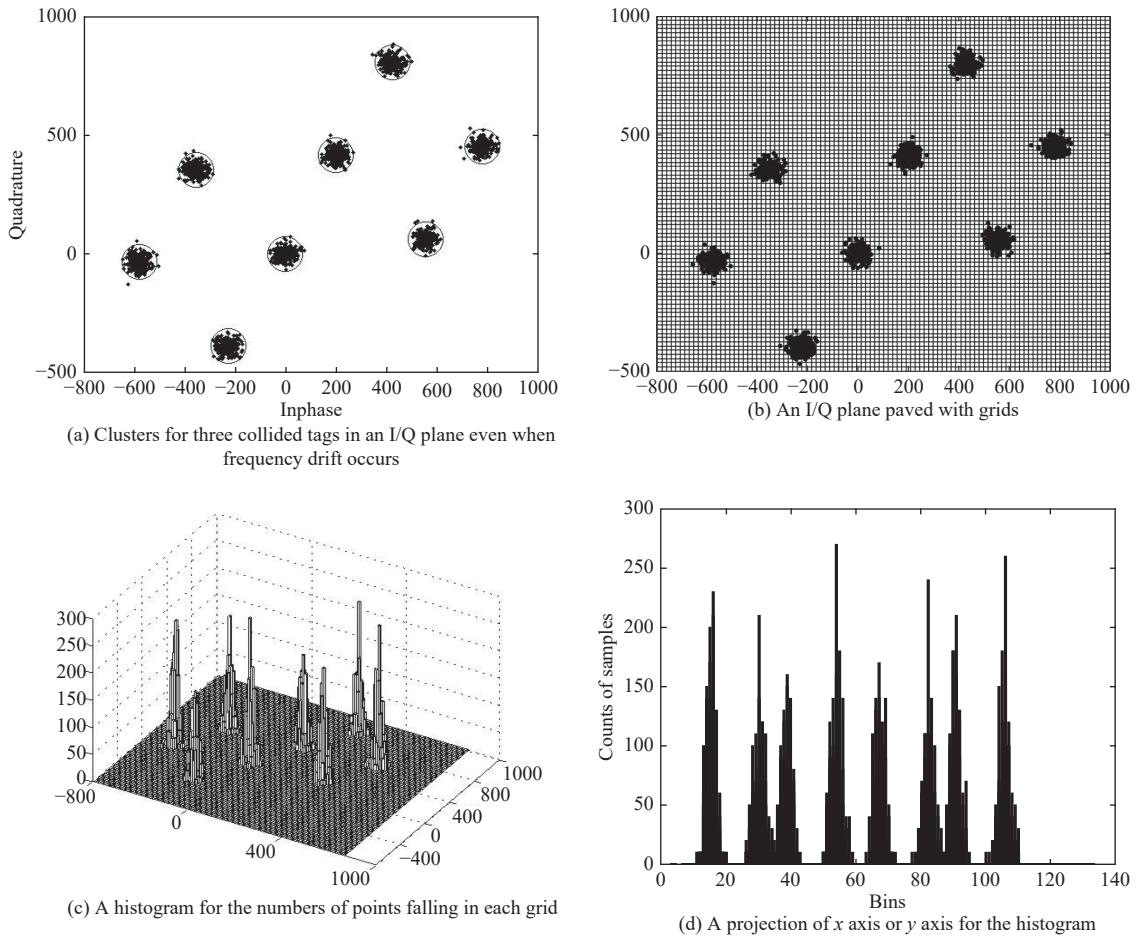


Fig. 7. Gain the initial value of centers.

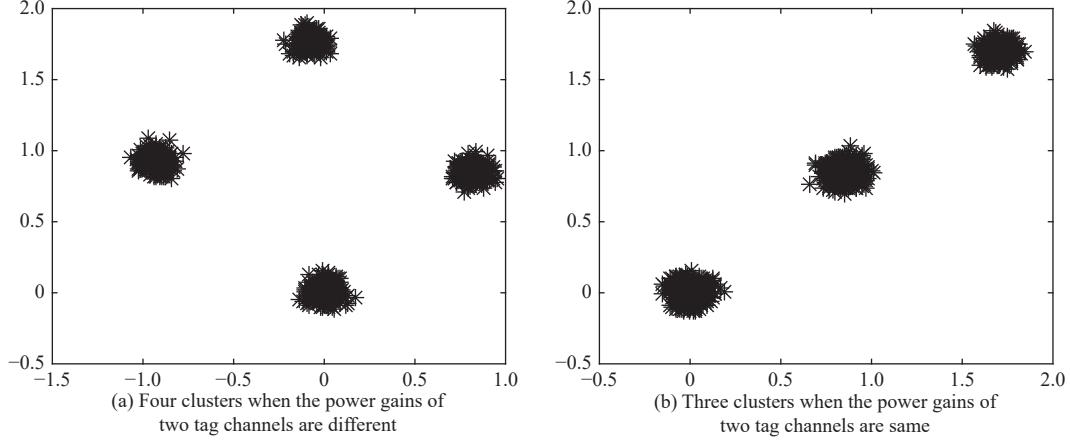


Fig. 8. Influence of two channels gain.

III. SIGNAL DECODING WITH FREQUENCY DRIFT

Due to noises, inferences, and drifts, signals separated by the RBF network will have some errors. To remove these separated errors, we introduce our FM0 decoding method under the condition of the frequency drift. In the EPC C1 Gen2 standard, FM0 is very popular for RFID tag signals [21], [25]. Fig. 10 illustrates an example of FM0 ($M = 1$) code, where there must be an edge (1 to 0 or 0 to 1) between two symbols. Moreover, a symbol 0 means that there is another edge in the middle, while 1 means no edges. From the

example, FM0 can be decoded where the edges are. Due to the frequency drift, however, FM0 will fail to be decoded only through the edges because the symbol period is time-variant. Next, we will describe how to decode separated signals under the condition of frequency drift.

A. FM0 Decoding With Frequency Drift

From the example in Fig. 10, the general rule for FM0 decoding is: a symbol will be decoded as 0 if there is a positive or negative edge during the symbol, and a symbol

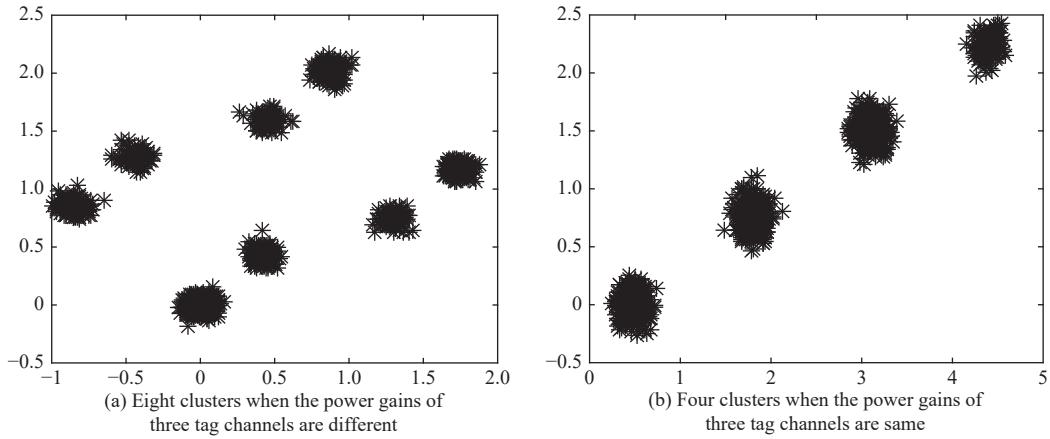


Fig. 9. Influence of three channels gain.

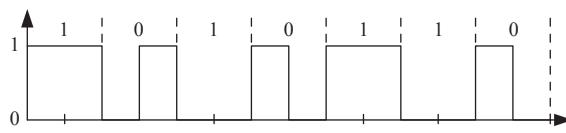


Fig. 10. An example for FM0 code.

will be decoded as 1 if there are no edges during the symbol. Further, there must be an edge between two symbols. Furthermore, the periods of symbols are time-variant due to frequency drift. Thus, the time when the edges occur will also be uncertain and will be in a range. Of course, FM0 is also considered as a hidden Markov chain where a Viterbi algorithm may be adopted [18]. However, when the symbol frequency is time-variant, computing the distance between two paths is very difficult. Matched filters with maximum-SNR criterions [19] are also not appropriate. This is because a time-variant symbol frequency is difficult to match to an appropriate filter.

Some existing methods can directly recover the collided tag via decoding, e.g. the method in [25]. However, when the frequency drifts, i.e. with a time-variant frequency, these methods do not work well. The method in [8] can be applied to the frequency shift among tags, but it assumes that the frequency in a tag will not drift, i.e. a tag whose frequency is time-invariant. Thus, an FM0 symbol will be decoded as 0 if there are edges during the symbol, and decoded as 1 if there are no edges. For the example of an one-order ($M = 1$) FM0 code shown in Fig. 11 from [8], a symbol even in collision will be decoded as 0 as long as there is an edge on half of its period times that are odd numbers, i.e., $(2n+1)T/2$ where T is a symbol period and n is an integer. However, when the frequency drifts, the transition edges position may be time-variant. Determining when this transition happens is difficult. Reference [8] judges whether there is an edge through a threshold. When the value of the transition is greater than the threshold, the transition will be valid. If there are noises or interferences, however, the value of the transition may be affected. Thus, the use of this threshold may lead to misjudgment.

In this paper, our algorithm is divided into two stages: separation and decoding. On one hand, since the signals

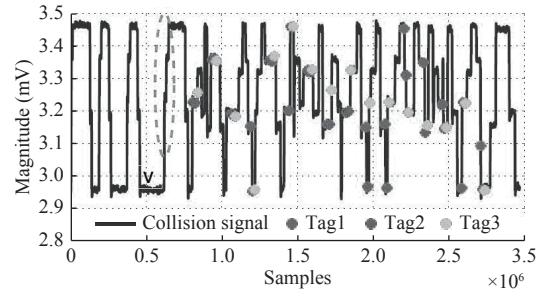


Fig. 11. Collision separation in [8].

separated by a RBF cover a single tag and appear either 1 or 0, it is easy to know when the transition happens as long as a 1 changed into 0 or 0 changed into 1. Also, because the frequency drift is within 22%, the EPC specifies can eliminate invalid transitions. Thus, it is not necessary to know the exact period.

On the other hand, due to noise and interferences, the separated signals by RBF networks will have separated errors, as shown in Fig. 12. The signals shown by black boxes are the separation errors and the transition edges are invalid edges. In this case, recovery of collision cannot be performed well only through direct FM0 decoding, because FM0 believes that a symbol is decoded as 0 as long as transition edges happen during the symbol. In our FM0 decoding, a frequency beyond 22% will be considered as an invalid edge. Therefore, edges in

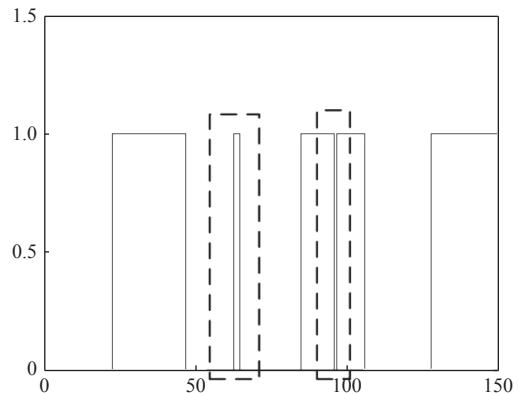


Fig. 12. Separated errors shown in black boxes.

the black boxes would be canceled.

In a word, the separated signals by RBF networks have only two cases: 0 or 1. Thus, where 0 changes into 1 or 1 changes into 0, a transition edge occurs. More complex detection theory methods, such as setting a threshold or seeking a peak via correlation functions seem to be unnecessary.

B. Steps of FM0 Decoding

From the EPC C1 Gen2 standard, the drift between the actual frequency and a nominal one f_{lp} is not beyond 22%. Hence, the actual frequency should drift within the range of $[0.78f_{lp}, 1.22f_{lp}]$. Here, we will use a nominal symbol period T_{lp} as a standard level width. Hence, an actual width between two symbols will have the following cases.

First, there will be a true edge in the middle of a symbol that is decoded as 0 if both of two continuous level widths are within $1/2 [T_{lp}/1.22, T_{lp}/0.78]$, shown in Fig. 13(a).

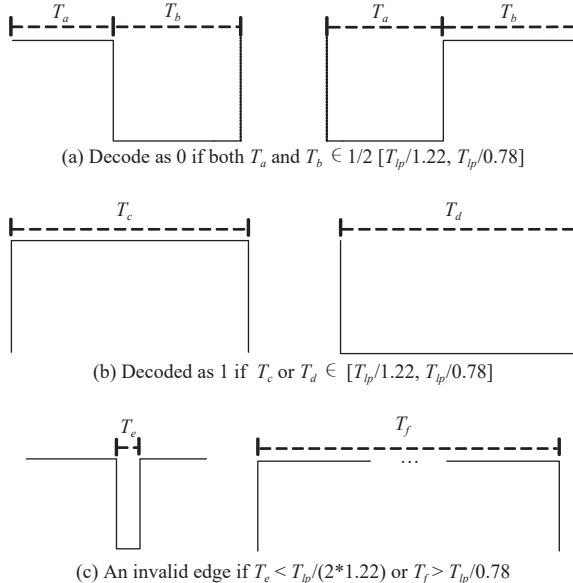


Fig. 13. Steps of FM0 decoding.

Second, a symbol will be decoded as 1 if its level width between two edges is within $[T_{lp}/1.22, T_{lp}/0.78]$, shown in Fig. 13(b).

Third, there will be a false edge if a level width l_i between the two edges is $[l_i < T_{lp}/(2 * 1.22) \text{ or } l_i > T_{lp}/0.78]$, shown in Fig. 13(c).

From the analysis above, we will give the FM0 decoding steps under the condition of the frequency drift, shown in Algorithm 1.

Algorithm 1 Algorithm of decoding

- 1) Record all edges position t_i , $i = 1, 2, \dots, N$, where N denotes the total number of edges;
- 2) Calculate the level width $l_i = t_{i+1} - t_i$;
- 3) If
 - a) l_i and $l_{i+1} \in 1/2[T_{lp}/1.22, T_{lp}/0.78]$, there will be a valid edge during a symbol and the symbol is decoded as 0.

Thus, $ID_i = 0$ and $i = i + 1$;

b) $l_i \in [T_{lp}/1.22, T_{lp}/0.78]$, there will be no edges during a symbol and the symbol is decoded as 1. Thus, $ID_i = 1$;

c) $l_i < T_{lp}/(2 * 1.22)$ or $l_i > T_{lp}/0.78$, there will be an invalid edge during a symbol. Thus, $t_i = t_{i+1}, t_{i+1} = t_{i+2}, \dots$;

4) $i = i + 1$. If $i < N$, go to step 3).

IV. RESULTS

A. System Setting

In this section, we give numerical results to verify the performance of the tag collision recovery. In the numerical experiments, we consider a scenario with a single-antenna reader and some passive tags. The final results are the average of 3000 independent experiment results.

The readers need to use a cross-layer approach to identify tags. The Aloha protocol is used on the MAC layer. When the tags in the MAC layer collide, the proposed method recovers the collision at the physical layer. Some system parameters in the experiments are referenced to the EPC C1 Gen2 standard [21], and the others are referenced to [9], [17]–[20]. The detailed parameters are as follows.

Channel: $h_1 = 0.7e^{j\pi/6}$, $h_2 = 0.3e^{j\pi/4}$, $h_3 = 0.2e^{j2\pi/3}$, $h_4 = 0.5e^{j3\pi/4}$.

Nominal link frequency: $f_{lp} = 500$ kHz [19]–[21].

Sampling frequency for baseband signal: 7500 kHz.

Symbol frequency and delay: each tag symbol frequency $1/p_{n,k}$; deviates up to $\pm 22\%$ from the nominal frequency f_{lp} ; the symbol frequency deviation among tags deviates up to $\pm 22\%$; each tag's symbol delay q_n is less than 24 μs [18], [20].

Block length: The tag's signal length K is 16 and identical to that of RN16 specified in EPC C1 Gen2 [21].

Antenna: single receiving antenna.

B. Separation via RBF

Figs. 14(a)–(c) gives results where RBF network separates three collided tags in an I/Q plane when frequency drift occurs and $\text{SNR} = 25$ dB. We can see from Fig. 14 that, the signals on one side of the classified curves are classified as 1 and the other classified as 0. That is, the curves could separate the cluster into signals 1 and 0.

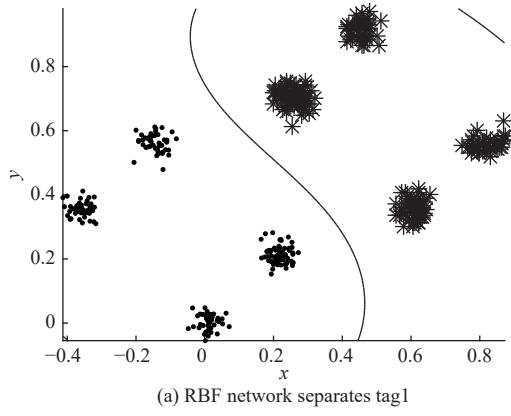
Furthermore, note that only an approximate linear line could classify the clusters for the separation of tag1 and tag3 shown in Fig. 14(a) and (c). On the other hand, the classifying curve of tag 2 shown in Fig. 14(b), requires to be nonlinear.

C. Estimation of Centers

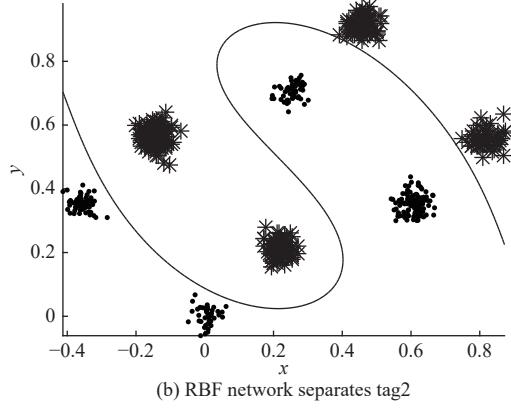
When the number of collided tags is three and frequency drift occurs, the results for center estimation through SIC [18], SAZF [19], LCE [20] and RBF is given in Fig. 15. From the figure, our method has fewer estimation errors than other conventional methods when SNR is greater than about 12 dB. It owes to an initial center value chosen by the method in [17]. Moreover, our RBF network can adopt the adaptive algorithm in (6) to further enhance the accuracy of the center estimation.

D. SER and Separation Efficiency

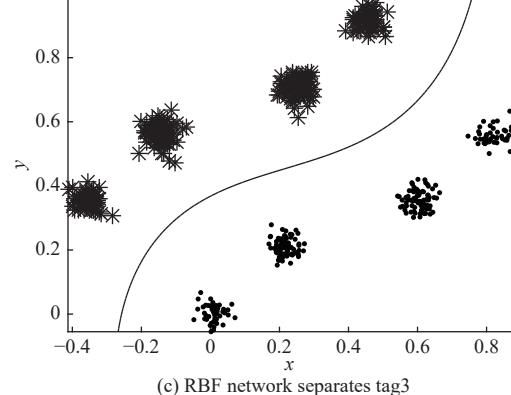
To evaluate the performance of each algorithm on



(a) RBF network separates tag1



(b) RBF network separates tag2



(c) RBF network separates tag3

Fig. 14. RBF network separates three collided tags.

separating tag signals, we consider the metric of ID SER, which is defined as

$$R_e = \frac{m_k}{m_l} \times 100\% \quad (14)$$

where m_l denotes the total number of symbols and m_k denotes the number of error symbols. Separation efficiency is defined as

$$P_e = \frac{n_b}{n_a} \times 100\% \quad (15)$$

where n_b is the number of tags successfully separated; n_a is the total number of collided tags. For the experiment of separation efficiency, the separation of a tag would fail as long as there is one ID symbol error for the decoded tag.

When the number of collided tags is three and no frequency drift occurs, the SER of SIC, SAZF, LCE and RBF for the three tags is given in Fig. 16. From the figure, the SER curve

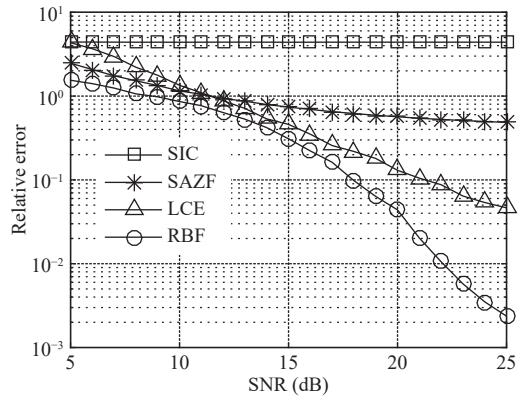


Fig. 15. Comparison results of center estimation via various algorithms.

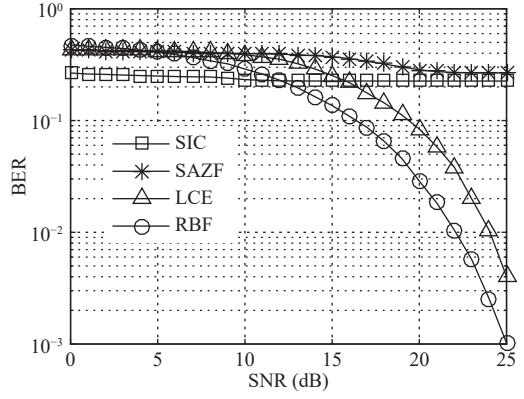


Fig. 16. Symbol error rate in (14) when the number of tags is three and no frequency drift occurs.

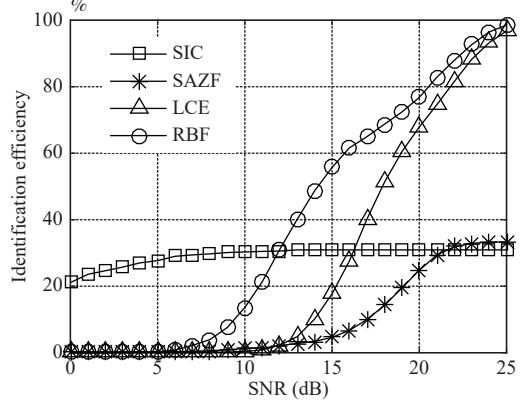


Fig. 17. Separation efficiency in (15) when the number of tags is three and no frequency drift happens.

of the RBF algorithm is below that of the others. The reason is that RBF does not need accurate channel information because it decodes the separated signal directly from the FM0 code. Fig. 17 shows the separation efficiency of the algorithms under the condition of three collided tags and no frequency drift. From the figure, the maximum separation efficiency of SIC and SAZF are 30% and 33%, respectively. The separation efficiencies of RBF and LCE algorithm increase with SNR when $\text{SNR} > 5 \text{ dB}$, and the separation

efficiencies nearly reach 98%. However, the RBF curve is still on the left of the LCE curve which means the performance of separation for RBF is a little better than that of LCE.

Fig. 18 gives SER of SIC, SAZF, LCE and RBF under the condition of three collided tags and frequency drift. As shown in Fig. 18, only the SER curve of RBF decreases with SNR. Fig. 19 shows the separation efficiency of the algorithms under frequency drift. As shown in Fig. 19, only the separation efficiency of RBF increases with SNR, and nearly reaches 98% when SNR = 25 dB. On the other hand, the separation efficiency of SIC, SAZF and LCE algorithm is still 0 even under higher SNR. The results of the two figures above indicate that frequency drift makes the other algorithms fail to decode the collided tags under higher SNR.

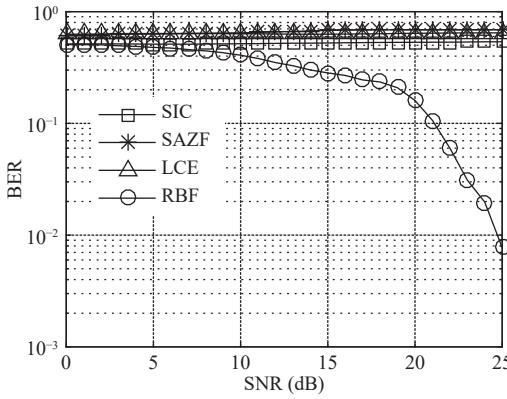


Fig. 18. Symbol error rate in (14) when frequency drift happens and the number of tags is three.

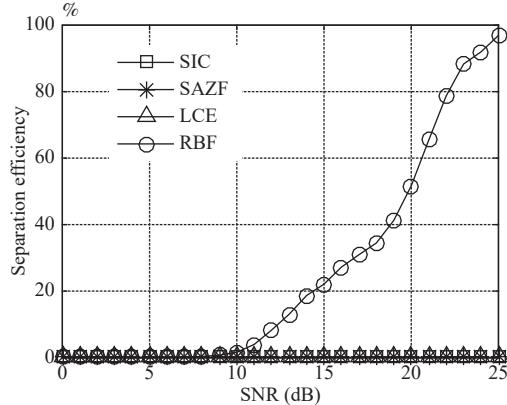


Fig. 19. Separation efficiency in (15) when frequency drift happens and the number of tags is three.

Fig. 20 gives the SER of four algorithms above under SNR = 20 dB, when frequency drift occurs and the number of tags varies from 2 to 4. Similar to Figs. 19 and 20, the curve of RBF is not horizontal regardless of the number of tags. The results indicate that two collided tag signals cannot be recovered by the other three methods as long as frequency drift occurs.

We give the comparison between our method and the method in [8] for SER performance under different drifts, shown in Fig. 21 when SNR is 20 dB. From the figure, when

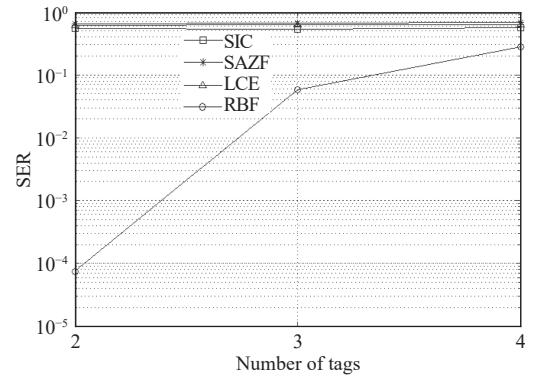


Fig. 20. Symbol error rate in (14) under SNR = 20 dB when frequency drift happens and the number of tags varies from two to four.

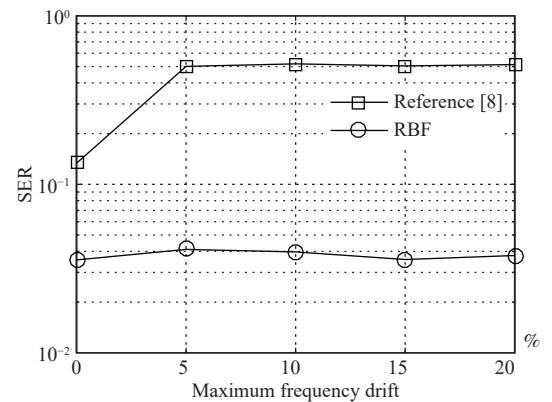


Fig. 21. Symbol error rate in (14) under SNR = 20 dB when frequency drift varies from 0 to 22% and the number of tags is three.

the frequency is time-invariant, i.e. the drift is 0%, SERs of the method in [8] and our method are both lower than 10^0 . Due to the noises having influence on the threshold judgment, [8]'s method shows higher SER. When the drift varies from 5% to 22%, however, [8] arrives at about 10^0 . On the other hand, our methods SER is between 10^{-1} and 10^{-2} .

V. DISCUSSION

From the numerical results, the proposed algorithm can separate collision tag signals with the frequency drift. Here, we will discuss some several factors that have an impact on the performance of the proposed algorithm. The first factor is chosen centers. The number of clusters for collision signals mapped to an I/Q plane is the number of tags of 2 powers. Generally, the number of centers for an RBF network may be chosen as the number of clusters. Thus, the number of centers would be known if the number of tags is known. When the number of tags is unknown, however, selecting the appropriate number of tags will be key for the RBF network. From Figs. 5 and 6, it is seen that too few centers is not used for correct separation while too many centers produce higher computational complexity of network training. As seen from the numerical results in Fig. 20, the performance of the separation is related to the number of collision tags. SER will increase with the number of collision tags, especially, when

the number of tags is 4, SER is beyond 10^{-1} . Thus, it is not necessary to separate all of collision tags regardless of the number of the collision tags. Separation for only two or three collision tags would a wise choice. For a dynamic frame length protocol, the average number of collision tags in a collision slot is about 2.33 [5]. Therefore, eight centers can cope with an RBF network for most of the collision tags.

Continuing, the preamble signal will have an impact on the RBF network. When the frequency of symbols drift, it is difficult to acquire exact information for the period and delay of symbols. The estimated preamble will have errors, which will also produce a trained RBF network with some errors. However, the drift is a low-frequency variation process, and the preamble signal only takes up about six symbol periods. The time information estimated by the existing methods will not have more errors. Moreover, the errors will also be eliminated via FM0 decoding.

The power gain and direction of channels is another key factor. In an extreme case, the exact same channel will not be able to separate collision signals because the clusters of two tags will overlap with each other. In a real environment, if the location, direction, and distance of the tags are different, the channel gain will be different, and thus the collision signals could be separated through the RBF network.

We choose three tags because the average number of collision tags in a slot is about 2.33 with the Aloha method from [3]. This indicates that the case of more than three collision tags in a slot is a low-probability occurrence. Moreover, our algorithm's separation curve makes it more difficult to classify clusters when there are too many collision tags. Thus, if more than three collision tags happen, we will not choose separation on the PHY layer.

It should be noted that the training time has an important influence on the tag separation performance of our algorithm, mainly on the RN16 stage. If the training time during RN16 is too long, the system will consider the tag as unresponsive and thus discard the subsequent identification of the tag. In fact, the training time of the proposed algorithm is related to the computing performance of hardware. Using high-performance computing chips can effectively reduce training time, but this will increase readers costs. As the performance of chips increase and the price decreases, our algorithm could be an alternative solution.

VI. CONCLUSION

The symbol frequency drift of RFID tag signal is a common phenomenon in a UHF system. This paper uses a RBF algorithm and FM0 to recover the collided tag signals with frequency drift. From experimental results, the performance of the symbol error rates and the separation efficiency for RBF is better than those of traditional algorithms when symbol frequency drift occurs.

Since the separation of collision signals in an I/Q plane is in fact a clustering problem, this paper proposes an algorithm to separate collision signals via a RBF neural network. In this paper, the algorithm only gives a framework for theoretical implementation of communication signal processing, and the numerical experiment is carried out on a software platform,

which is not tested in real environment. However, all parameters are set from the actual environment, such as EPC C1 Gen2, and we plan to run the algorithm in a real UHF RFID system in the future to test its performance.

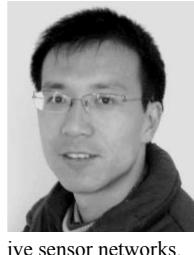
REFERENCES

- [1] Xi Tan, He Wang, Lingzhi Fu, Junyu Wang, Hao Min, Daniel W. Engels, "Collision detection and signal recovery for uhf rfid systems," *IEEE Transactions on Automation Science and Engineering*, vol. PP, no. 99, pp. 1–2, 2016.
- [2] L Zhang, W Xiang, X Tang, "An adaptive anticollision protocol for large-scale rfid tag identification," *Wireless Communications Letters IEEE*, vol. 3, no. 6, pp. 601–604, 2014.
- [3] Haifeng Wu, Yu Zeng, Jihua Feng, Yu Gu, "Binary tree slotted aloha for passive rfid tag anticollision," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 19–31, 2013.
- [4] Min Shao, Xiaofang Jin, and Libiao Jin. An improved dynamic adaptive multi-tree search anti-collision algorithm based on rfid. In *International Conference on Data Science and Advanced Analytics*, pages 72–75, 2015.
- [5] Haifeng Wu, Yu Zeng, "Bayesian tag estimate and optimal frame length for anti-collision aloha rfid system," *IEEE Transactions on Automation Science & Engineering*, vol. 7, no. 4, pp. 963–969, 2010.
- [6] Martin Mayer, Norbert Goertz, "Rfid tag acquisition via compressed sensing: Fixed vs. random signature assignment," *IEEE Transactions on Wireless Communications*, vol. 15, no. 3, pp. 2118–2129, 2016.
- [7] Jinsong Han, Chen Qian, Panlong Yang, Dan Ma, Zhiping Jiang, Wei Xi, Jizhong Zhao, "Geneprint: Generic and accurate physical-layer identification for uhf rfid tags," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 846–858, 2016.
- [8] Mohammed Benbaghdad, Belkacem Fergani, Smail Tedjini, "Toward a new phy layer scheme for decoding tags collision signal in uhf rfid system," *IEEE Communications Letters*, vol. 20, no. 11, pp. 2233–2236, 2016.
- [9] Yuanqing Zheng, Mo Li, "P-mti: Physical-layer missing tag identification via compressive sensing," *IEEE/ACM Transactions on Networking*, vol. 23, no. 4, pp. 1356–1366, 2015.
- [10] P. Ishwarya, V Nithish Kumar, and G. Lakshminarayanan. An efficient digital baseband encoder for short range wireless communication applications. In *International Conference on Electrical, Electronics, and Optimization Techniques*, pages 2775–2779, 2016.
- [11] Feng Zhu, Bin Xiao, Jia Liu, Li Jun Chen, "Efficient physical-layer unknown tag identification in large-scale rfid systems," *IEEE Transactions on Communications*, vol. 65, no. 1, pp. 283–295, 2017.
- [12] Xi Yang, Haifeng Wu, Yu Zeng, Fei Gao, "Captureaware estimation for the number of rfid tags with lower complexity," *IEEE Communications Letters*, vol. 17, no. 10, pp. 1873–1876, 2013.
- [13] J Kaitovic and M Rupp. Rfid physical layer collision recovery receivers with spatial filtering. In *IEEE International Conference on Rfid Technology and Applications*, pages 39–44, 2015.
- [14] M Benbaghdad, B Fergani, S Tedjini, and E Perret. Simulation and measurement of collision signal in passive uhf rfid system and edge transition anti-collision algorithm. In *Rfid Technology and Applications Conference*, pages 277–282, 2014.
- [15] Fabio Ricciato, Paolo Castiglione, "Pseudo-random aloha for enhanced collision-recovery in rfid," *IEEE Communications Letters*, vol. 17, no. 3, pp. 608–611, 2013.
- [16] Yuxiao Hou, Jiajue Ou, Yuanqing Zheng, Mo Li, "Place: Physical layer cardinality estimation for largescale rfid systems," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2702–2714, 2016.
- [17] Dawei Shen, G Woo, D. P Reed, and A. B Lippman. Separation of multiple passive rfid signals using software defined radio. In *IEEE*

- International Conference on Rfid*, pages 139–146, 2009.
- [18] Karsten Fyhn, Rasmus M. Jacobsen, Petar Popovski, Anna Scaglione, Torben Larsen, “Multipacket reception of passive uhf rfid tags: A communication theoretic approach,” *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4225–4237, 2016.
- [19] Christoph Angerer, Robert Langwieser, Markus Rupp, “Rfid reader receivers for physical layer collision recovery,” *IEEE Transactions on Communications*, vol. 58, no. 12, pp. 3526–3537, 2010.
- [20] Hanjun Duan, W. U. Haifeng, Zeng Yu, “Channel estimation for recovery of uhf rfid tag collision on physical layer,” *Journal of Electronics and Information Technology*, pp. 1–5, 2016.
- [21] Epc radio frequency identity protocols class-1 generation-2 uhf rfid protocol for communications at 860 mhz–960 mhz. 2013.
- [22] M. Benbaghdad, B. Fergani, S. Tedjini, “Backscatter signal model of passive uhf rfid tag. application to collision detection,” *Electronics Letters*, vol. 52, no. 11, pp. 974–976, 2016.
- [23] Jiajue Ou, Mo Li, and Yuanqing Zheng. Come and be served:parallel decoding for cots rfid tags. In *International Conference on Mobile Computing and NETWORKING*, pages 500–511, 2015.
- [24] P. Hu, P. Zhang, D Ganesan, “Laissez-faire: Fully asymmetric backscatter communication,” *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 4, pp. 255–267, 2015.
- [25] Aggelos Bletsas, John Kimionis, Antonis G. Dimitriou, George N. Karystinos, “Single-antenna coherent detection of collided fm0 rfid signals,” *IEEE Transactions on Communications*, vol. 60, no. 3, pp. 756–766, 2012.
- [26] Simon S Haykin, “Neural networks and learning machines,” *Pearson Schweiz Ag*, 2009.
- [27] Junzhi Li, Haifeng Wu, Yu Zeng, and Yong Shen. Fm0 decode for collided rfid tag signals with frequency drift. In *International Conference on Image, Vision and Computing*, pages 941–945, 2017.



Junzhi Li currently pursuing his M.S. degree in electrical engineering at Yunnan Minzu University, Kunming, China. His main research interest is radio frequency identification (RFID) technology.



Haifeng Wu received the M.S. degree in electrical engineering from Yunnan University, Kunming, China, in 2004, and the Ph.D. degree in electrical engineering from Sun Yat-Sen University, Guangzhou, China, in 2007. He is currently a Professor in the Department of Information Engineering at the Yunnan Minzu University. Prior to that, he was a postdoctoral scholar at the Kunchuan Institute of Technology from 2007 to 2009. His research interests include RF engineering, mobile communications and cooperative sensor networks.



Yu Zeng received the M.S. degree in electrical engineering from Yunnan University, Kunming, China, in 2006. She is currently an Assistant Professor in the Department of Information Engineering at the Yunnan Minzu University. Prior to that, she was an Electrical Engineer at Kunming Institute of Physics from 2006 to 2009. Her research interests include wireless network and mobile communications.