Collision Resolution with FM0 signal separation for short-range random multi-access wireless network

Haifeng Wu, Xiaogang Wu, Yi Li and Yu Zeng

***Abstract*—In internet of things, a central node to multiple sensor nodes is usually a short-range communication network, which generally uses random multiple access to resolve collision at a media access control (MAC) layer. In this case, the communication efficiency is not optimal. On the other hand, its efficiency can be improved in separated collision resolution, where collision signals can be separated and then decoded. For the resolution, good coding techniques can improve the performance of collision resolution. However, the conventional coding techniques usually need the signal fading coefficient estimated or be known in advance. The condition is not always satisfied in the short-range network. This paper proposes a finite-symbol separation Viterbi decoder (FSVD) which does not need to know the fading coefficient. Through the estimation of a dictionary matrix, the uncertainty of signal separation can be eliminated, and thus obtain more accurate likelihood distances and improve the decoding accuracy. In experiments, we establish two ultra-high frequency (UHF) short-range wireless networks based on an FM0 code, using a simulation and a software radio platform, respectively. The experimental results show that the throughput of FSVD reaches about 0.61, which is about 0.25 higher than that of a traditional ALOHA network.**

***Index Terms*—short-range network, collision separation, FM0**

I. INTRODUCTION

INTERNET of Things (IoT) establishes a ubiquitous connection of objects, people, and processes through the physical information collected by various sensors, and thereby form an internet where all things can be interconnected. The physical information of IoT needs to be collected by sensors before it can enter the internet to be exchanged. Short-range wireless communication is the main pattern for sensor communication in IoT, such as NFC, WSN, RFID, Wi-Fi, and Bluetooth [1,2]. The short-range communication is usually a wireless network, where a central node, like a reader in item identification, access control or person positioning [3] will have the ability of collecting the information from distributed senor nodes. In order to improve the real-time performance of the short-range wireless sensor communication, it is necessary for the central code to have the ability to collect multiple information in a short time. The wireless communication uses a shared channel, so multiple nodes transmitting signal simultaneously will cause collision. Usually, the collision is resolved via random multiple access on a media access control (MAC) layer, such as ALOHA and tree resolution [4-9]. In the methods, signal packets are transmitted randomly. When a collision occurs, the packets are retransmitted randomly until successfully received. Although the random collision resolution is simple, the retransmission also increases when the nodes increases. This necessarily leads to a reduction in communication efficiency. In fact, the collision signals are the superposition of the signals and can be directly separated to decode. The separated collision resolution [10-14] no longer considers the collision signals as invalid ones, reduces signal retransmission, and thus advances communication efficiency. So, it has always received more attention.

The short-range wireless communication networks often use signal encoding and decoding technology to reduce interference. Since the essence of the collision resolution is a kind of interference cancellation technology, good encoding can significantly improve the performance of the collision resolution. In wireless communication, signal frequency drift is a common phenomenon. Delay codes like FM0 and Miller codes [15,16] are also often used in short-range wireless communication to combat the frequency drift. Different from a matched filter, the delay codes can decode a symbol 0 or 1 via the change of signal phase instead of the period of signal, i.e. the frequency. Since the phase of the signal does not vary with the frequency, the delay codes can better solve the decoding in

This work was supported by the Natural Science Foundation of China under Grant 61762093, by the Key Applied and Basic Research Foundation of Yunnan Province under Grant 2018FA036, and by the Program for Innovative Research Team (in Science and Technology), University of Yunnan Province. (Corresponding author: Haifeng Wu.)

The authors are with the Department of Electrical and Information Engineering, Yunnan Minzu University, Kunming 650500, China (email: whf5469@gmail.com; wuxiaogang9527@gmail.com; 1325675214@qq.com; yv.zeng@gmail.com).

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. Citation information: DOI 10.1109/TSIPN.2021.3093000, IEEE Transactions on Signal and Information Processing over Networks

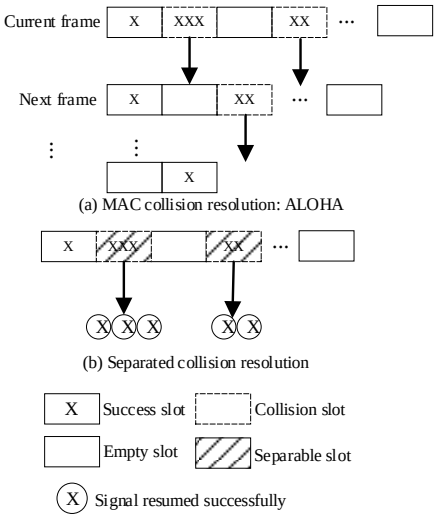


Fig.1. MAC-layer collision resolution and separated collision resolution.

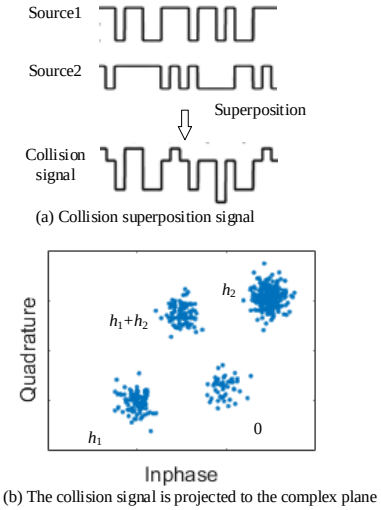


Fig.2. An example for a collision signal with two signal sources projected to a complex plane.

the frequency drift. In the separated collision resolution, it will be very difficult to decode only through the change of the phase because the decoding signal are collision superimposed signal. A simple decoding method is to judge the phase change after separating the superimposed signals. An unsupervised k-means clustering method can be used for the signal separation [17]. When the number of signal sources increases, however, there are uncertainties for signal clusters. In order to solve the uncertainty, supervised separation can be used. The fading coefficients of the signal sources can be estimated and thus the cluster center points can be determined [18]. And then, the signals will be separated. However, how to estimate the fading coefficients is also difficult, because either pilot or blind [19-21] estimation will increase the complexity of the system. Another collision resolution method is to directly decode the source signals from the collision ones instead of separating the signal, such as Viterbi and continuous interference cancellation [22]. The method uses a maximum likelihood criterion to find symbols most similar to the observed ones as decoded symbols. However, this method needs to calculate distance to find the maximum likelihood. If there is no prior information of the fading coefficients, the calculation of the maximum likelihood will probably fail.

For the problems in the separated collision resolution, this paper proposes a finite-symbol separation Viterbi decoder (FSVD). From EPC C1 Gen2 standard [23], we use a simulation and a software radio platform to establish UHF short-range wireless networks based on an FM0 code [24] in experiments, where the collision resolution embedded into the ALOHA protocol is evaluated. The experiment results show that the proposed FSVD has a decoding efficiency of nearly 100% when a signal-to-noise ratio exceeds 15 dB and the collision sources are 2, 3 and 4. Moreover, FSVD's throughput is close to decoding algorithms with known fading coefficients, and is better than traditional algorithms with unknown fading coefficients.

II. Related Work

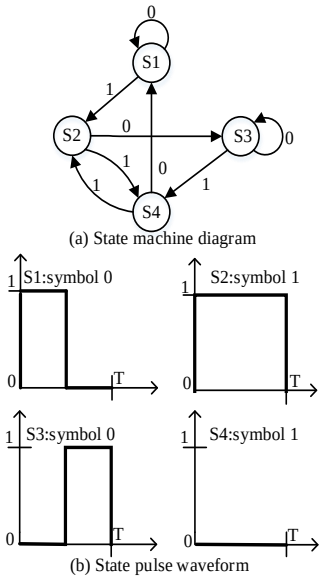
Traditional MAC-layer collision resolution uses a random multiple access network, such as ALOHA shown in Fig. 1(a), where signal packets select random slots. If a collision occurs, the packets will be retransmitted in the next frame. However, the MAC-layer method treats the collision slot as invalid, and the communication efficiency is not high. Separated collision resolution is shown in Fig. 1(b), where collision signals can be separated and then decoded. Since it considers the collision slot to be effective, the communication efficiency will be improved.

The collision signals are actually the superimposed signal of each source signal. One separated collision resolution is to firstly separate the collision signals and then decode it. Fig. 2(a) shows how two sources are superimposed and become collision signals. After received digital signals are Inphase-Quadrature (IQ) demodulated, generally, its sampling points can be projected onto a complex plane, where I is for a real axis and Q for an imaginary one. If the digital signals are represented by unipolar waves, then the sampling points of the collision signals will show $M = 2^N$ clusters on the complex plane, where N is the number of signal sources. In this way, the set composed of the centroids of the clusters can be expressed as [17].

$$\begin{aligned}\mathcal{H} = \{ & 0,\ h_1,\ h_2,\ \ldots,\ h_N,\\ & h_1+h_2,\ h_1+h_3,\ \ldots,\ h_{N-1}+h_N,\\ & \ldots,\ h_1+h_2+\ldots+h_N\}\end{aligned} \tag{1}$$

where h_1, h_2, …, h_N are the fading coefficients of signal sources. After clustered, any sampling points can be separated into an N-dimension vectors from (1), and the set of separation results is expressed as

$$\begin{aligned}\mathcal{B} = \{ & \underset{N}{[00\ldots0]},\ \underset{N}{[10\ldots0]},\underset{N}{[01\ldots0]},\ \ldots,\underset{N}{[00\ldots1]},\\ & \underbrace{[110\ldots0]}_{N},\ \underbrace{[1010\ldots0]}_{N},\ \ldots,\underbrace{[0\ldots011]}_{N},\\ & \ldots,\underset{N}{[11\ldots1]}\}\end{aligned} \tag{2}$$



(a) State machine diagram

(b) State pulse waveform

Fig.3. FM0 state machine and its waveform.

where an element of $\mathcal{B}$ will one-to-one correspond to that of $\mathcal{H}$. For example, a point belonging to the cluster with centroid 0 will separated into [00…0]. Fig. 2(b) shows an example, where a collision signal with two signal sources are projected to four clusters on a complex plane.

Another separated collision resolution is to decode each source signal directly from the collision signal according to a code, such as FM0 and Miller code [15]. As delay codes, FM0 and Miller code can not only resist frequency drift through phase changes, but also use their finite state machine to resolve the collision with a Viterbi algorithm [22]. Fig. 3(a) is FM0's state transition diagram and its waveform. From the figure, FM0 code can be regarded as a finite state machine, and the grid diagram shown in Fig. 4 can be drawn from the machine. The Viterbi algorithm in FM0 decoding actually finds a maximum likelihood path from the grid graph and then use a successive interference cancellation technique to resolve the collision.

III. Problem Description

For the separated collision resolution, the signal separation needs to know the centroids of clusters, but there is uncertainty in them. As shown in Fig. 2(b), the centroids are arranged clockwise from 0 to h_1 , h_1+h_2 and h_2 .Since the fading coefficients' values and phases are unknown in advance, however, the arrangement may also be anticlockwise and even in other directions.

If there are M centroids, then there may be A_M^M permutations for the centroids. When M is large, the number of permutations will be a huge value. If the centroid arrangement cannot be determined, the signal separation will be difficult. The most direct solution is to estimate the signal fading coefficients for the arrangement. However, there are still difficulties in the estimation. If a pilot method [21] is selected, the pilot needs to be designed. This will increase the complexity of the system. Moreover, the pilot will reduce the communication efficiency.

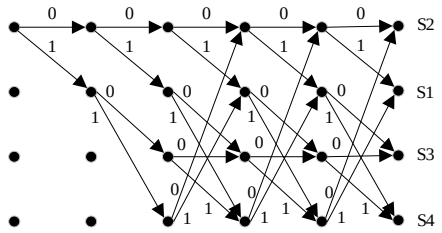
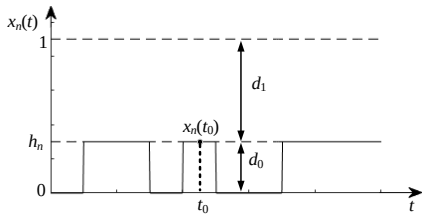


Fig.4. Grid for Viterbi algorithm in FM0 coding.



If h_n is small and d_0<d_1, then $x_n(t_0)$ is misjudged as 0.

Fig.5. An example of misjudgment for a symbol.

If a blind estimation [20] is selected, on the other hand, the complexity of the system will also increase due to its higher computation.

For the direct separated collision resolution, it needs to calculate a distance between observed signals and searched ones to find a maximum likelihood path. If the signal fading coefficients are unknown, however, a misjudgment will also produce. Fig. 5 shows a unipolar wave with a fading coefficient of h_n which is much smaller than 1. In the example, a symbol 1 will be easily misjudged as 0 and thus a wrong survival path will be found. For the problem, we can estimate h_n and normalize the unipolar wave. As mentioned above, however, the estimation will bring the complexity of the system.

The separated collision resolution proposed in this paper will solve the cluster uncertainty with lower complexity. The collision signals are uniquely separated to a unipolar wave with normalized 0 and 1. And then, a likelihood distance is calculated to reduce the error rate.

IV. System Model

The model of the short-range wireless network studied in this paper is shown in Fig. 6. Several nodes in the model use shared wireless channels to send data to a central node. Similar to the ALOHA algorithm, firstly, the nodes randomly select slots to send signals to the central node. If multiple signals collide in a slot, however, the separated collision resolution instead of re-transmitting is performed. Only when the separated signal is not decoded correctly, does it need to be retransmitted. Assume that in such a time slot, there are K binary symbols $u_{n,k} \in \{0,1\}$ n=1, 2, … N , k=1, 2, … K sent by N nodes and the symbols are encoded into sequences $v_{n,k'} \in \{0,1\}$, n=1, 2, … N , k'=1, 2, … K'. In general, $K \neq K'$. For example, Fig. 3 shows that a symbol 0 will be FM0 encoded into two symbols 1 and 0, or 0 and 1 where $K = K'/2$ [23]. If L is expressed as carrier leakage, then a signal $y(t)$ after IQ modulated at a receiving central node is expressed as

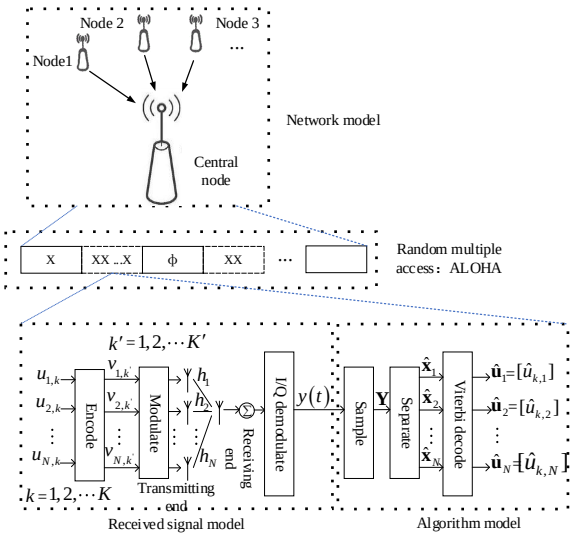


Fig.6. System model for a short-range wireless network.

$$y(t) = \sum_{n=1}^{N} h_n x_n(t) + \eta(t) + L \tag{3}$$

where

h_n is a complex fading coefficient from the n-th node to the receiving node, and in short-range communication, the fading coefficient can be regarded as a linear time-invariant flat fading channel in a short communication time [22],

$x_n(t) = \sum_{k'=1}^{K'} v_{n,k'} g_{a_n}(t - k'a_n - b_n)$ represents the n-th node signal,

a_n and b_n are the symbol period and symbol delay respectively,

$g_{a_n}(t)$ is the modulated square wave, i.e. $g_{a_n}(t)$ =1 when $0 \le t \le a_n$, otherwise 0,

$\eta(t)$ is an additive noise signal.

When decoding the received signal, we firstly sample the received signal. Suppose T is a sampling period, and I samples can produce a received signal vector $\mathbf{Y} = [y_i] \in \mathbb{C}^{1\times I}$, where $\mathbb{C}$ is a complex number set and $y_i = y(iT)$, i =1, 2, … I. Then, a minimum variance criterion is used to separate the vector $\mathbf{Y}$ into N source signal vectors $\hat{\mathbf{x}}_n$, n =1, 2, … N. Finally, the symbol vectors $\hat{\mathbf{u}}_n = [\hat{u}_{k,n}] \in \{0,1\}^{K\times 1}$, n=1, 2, … N are obtained from $\hat{\mathbf{x}}_n$, via a Viterbi decoder.

V. FSVD Algorithm

A. Overview of Separation

This sub-section gives the overview of the signal separation and Fig. 7 shows the flow chart of the separation. First, the received signal $\mathbf{Y}$ is projected on a complex plane. After clustered in the plane, M centroids $c_m \in \mathcal{H}$, m=1, 2, … M can be obtained. Next, a dictionary matrix $\hat{\mathbf{D}}$ is estimated from the clustered centroids. After y_i clustered, in addition, there will be $y_i \in \mathcal{C}_m$ where $\mathcal{C}_m$ denotes a set of points in the m-th cluster. Finally, a separation function $\mathbf{f}_{dict}(\cdot)$ can be determined from the dictionary matrix and the clustering result, and thus the separated source signal vector $\hat{\mathbf{x}}_n = [\hat{x}_{i,n}]$ would be obtained from the function.

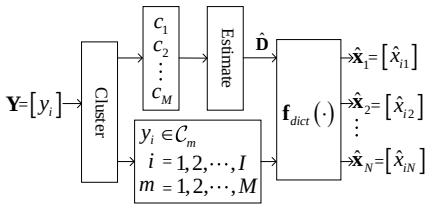


Fig.7. Flow chart of separation algorithm.

B. Dictionary Matrix Estimation

Let $\mathbf{D} = [\mathbf{d}_m] \in \{0,1\}^{M\times N}$ be a dictionary matrix composed of row vectors $\mathbf{d}_m$, where $\mathbf{d}_m \in \mathcal{B}$ and $\mathbf{d}_m \ne \mathbf{d}_n$ when $m \ne n$. Then, the cluster center vector $\mathbf{C} = [c_m] \in \mathbb{C}^{M\times 1}$ can be expressed as

$$\mathbf{C} = \mathbf{DH} \tag{4}$$

where $\mathbf{H} = [\, h_1, h_2, \ldots h_N \,]^{\mathrm{T}}$. If the dictionary matrix $\mathbf{D}$ is determined, a mapping relationship $\mathbf{f}_{sep}(\cdot)$ between points in $\mathcal{C}_m$ and $\mathbf{d}_m$ can be obtained from (4), expressed as

$$\mathbf{d}_m = \mathbf{f}_{sep}(y_i), \text{ if } y_i \in \mathcal{C}_m \tag{5}$$

For example, if $\mathbf{d}_2$ =[0 1 0 1] and y_i are clustered into $y_i \in \mathcal{C}_2$, it can be decomposed into four sampling points 0, 1, 0 and 1. To determine the function of equation (5), the dictionary matrix $\mathbf{D}$ needs to be estimated. Establish a cost function as

$$f(\mathbf{C} \mid \mathbf{D}) = \| \mathbf{DD}^{\dagger}\mathbf{C} - \mathbf{C} \|_2 \tag{6}$$

and the following estimates were obtained (the proof seen in Appendix)

$$\hat{\mathbf{D}} = \arg\min_{\mathbf{D}\in\mathcal{D}} \| \mathbf{DD}^{\dagger}\mathbf{C} - \mathbf{C} \|_2 \tag{7}$$

where $\mathcal{D}$ is a set of the matrix $\mathbf{D}$ with row vectors $\mathbf{d}_m \in \mathcal{B}$, and $(\cdot)^{\dagger}$ denotes a pseudo-inverse. Since $\mathcal{B}$ is a finite set, the set $\mathcal{D}$ of $\mathbf{D}$ is also a finite set. Thus, the estimation in (7) can be performed through a brute search. Note that the estimation in (7) does not need to know the signal fading coefficient $\mathbf{H}$ in advance.

C. Low-complexity Estimation

From (7) and (2), we can see that $|\mathcal{D}| = A_M^M = M!$. A brute search for (7) will result in higher complexity. We give the following two methods.

1) $N < 4$

A cluster centroid [00...0] can be determined in advance since it corresponds to the carrier leakage L during a silent period. In this case, $\mathbf{C}$ and $\mathbf{D}$ can be updated as

$$\mathbf{C} = [c_m] \in \mathbb{C}^{(M-1)\times 1}, \qquad \varepsilon(c_m) \in \mathcal{H} - \{0\} \tag{8}$$

$$\mathbf{D} = [\mathbf{d}_m] \in \{0,1\}^{(M-1)\times N}, \quad \mathbf{d}_m \in \mathcal{B} - \{[0,0,\ldots 0]\} \tag{9}$$

From (8-9), $|\mathcal{D}|$ can be reduced to $(M-1)!$. Substitute it into (6-7), we have the number of searches reduced to 3! =6 and 7!=5040 when N is 2 and 3. Moreover, (6) can be rewritten as

$$\hat{\mathbf{D}} = \arg\min_{\mathbf{D}'\in\mathcal{D}'} \| \mathbf{D}'\mathbf{C} \|_2 \tag{10}$$

where $\mathcal{D}'$ is a matrix set composed of $\mathbf{D}' = \mathbf{DD}^{\dagger} - \mathbf{I}$, $\mathbf{D} \in \mathcal{D}$ and can be pre-calculated and stored in memory. Therefore, the search in (10) can be performed in a set of the pre-stored matrices to reduce computational complexity.

TABLE I
FSVD ALGORITHM STEPS

Inputs:
Received signal vector $\mathbf{Y}=[y_i] \in \mathbb{C}^{1\times I}$
Outputs:
Source symbol vectors $\hat{\mathbf{u}}_n=[\hat{u}_{k,n}] \in \{0,1\}^{K\times 1}$, $n=1,2,\dots N$
Known conditions:
Carrier leakage L
Viterbi decoding function $\mathbf{f}_{vit}(\cdot)$

Steps:
①**Clustering**: get a centroid vector $\mathbf{C}=[c_m] \in \mathbb{C}^{M\times 1}$
②**Dictionary matrix estimation**: get $\hat{\mathbf{D}}$ from (10) or (11)
③Signal **separation**: the signal separation matrix $\hat{\mathbf{X}}=[\xi_i] \in \{0,1\}^{I\times N}$ is obtained from (14) and (5)
④**Decoding**: get symbol vectors $\hat{\mathbf{u}}_n=[\hat{u}_{k,n}] \in \{0,1\}^{K\times 1}$, $n=1, 2, \dots N$ from (15)

TABLE II
PARAMETERS FOR SIMULATION DATA

Parameters	Description
Link frequency	150 kHz
Symbol length K	16
Sampling Frequency	7.5 MHz
Symbol period tolerance	±7%
Symbol delay tolerance	±2.46 μs
Iterations in Kemans	50
Encoding	
Type	Single type FM0
Initial state	S_1
Preamble	1,0,1,0,v[1],1

Note: 1. "v" is a symbol breaking the encoding [23].

2) $N \geq 4$

When the number of nodes N in a collision slot is large, the number of searches will increase significantly. For example, when $N=4$, the number of searches is 15! ≈7e5 and (10) will result in huge computational cost. A genetic algorithm [25] can be used to reduce the computational cost if the problem is expressed as a constrained integer nonlinear programming problem, i.e.

$$\min f(\alpha_1,\alpha_2,\dots\alpha_{M-1})$$
$$\text{subject to} \begin{cases} \alpha_m \in \mathbb{Z} \\ 1 \leq \alpha_m \leq M-1 \quad , m=1,2,\dots M-1 \\ \alpha_m \neq \alpha_n, \text{if } m \neq n \end{cases} \tag{11}$$

where

$$f(\alpha_1,\alpha_2,\dots\alpha_{M-1}) = \| (\mathbf{D}\mathbf{D}^{\dagger}-\mathbf{I})\mathbf{C} \|_2 \tag{12}$$
$$\mathbf{D}=[\mathbf{f}_{d2b}(\alpha_m,N)] \in \{0,1\}^{(M-1)\times N} \tag{13}$$

$\mathbf{f}_{d2b}(\alpha_m,N)$ is a function of a decimal number converted into a N-dimension binary vector (e.g. $\mathbf{f}_{d2b}(5,4)$ = [0 1 0 1]). We adopt a genetic algorithm function in MATLAB to realize the optimization. Since the MATLAB function can only realize the optimization with decimal integer constraints, the function $\mathbf{f}_{d2b}(\cdot)$ is used to convert decimal numbers to binary numbers. If $\hat{\alpha}_1,\hat{\alpha}_2,\dots\hat{\alpha}_{M-1}$ is a group of solutions for the program, then substituting it into (13) will have the estimated dictionary $\hat{\mathbf{D}}$.

D. Separation and Viterbi Decoding

First, we will separate the collision signals. From the functional in (5), let $\xi_i=\mathbf{f}_{dict}(y_i)$. Then, the collision signal vector $\mathbf{Y}$ can be separated into a signal matrix $\hat{\mathbf{X}}=[\xi_i] \in \{0,1\}^{I\times N}$ with row vectors ξ_i, $i=1, 2, \dots I$. The separation can be expressed as

$$\hat{\mathbf{X}}=\mathbf{f}_{sep}(\mathbf{Y}) \tag{14}$$

Here, write the signal matrix $\hat{\mathbf{X}}$ into $\hat{\mathbf{X}}=[\hat{\mathbf{x}}_n] \in \{0,1\}^{I\times N}$ with column vectors $\hat{\mathbf{x}}_n=[\hat{x}_{i,n}] \in \{0,1\}^{I\times 1}$,where $\hat{x}_{i,n}=\hat{x}_n(iT)$ is the i-th sampling point of the n-th source.

Next, we will decode the separated signals. A Viterbi algorithm $\mathbf{f}_{vit}(\cdot)$ for an FM0 (shown in Fig. 4) or a Miller code rule will decode the n-th source vector $\hat{\mathbf{x}}_n$ to

$$\hat{\mathbf{u}}_n=\mathbf{f}_{vit}(\hat{\mathbf{x}}_n) \tag{15}$$

where $\hat{\mathbf{u}}_n=[\hat{u}_{k,n}] \in \{0,1\}^{K\times 1}$. Note that the separated vector $\hat{\mathbf{x}}_n$ is composed of a sequence of binary 0 and 1. When we calculate a likelihood path, therefore, the misjudging case in Fig. 5 will not occurs because the fading coefficient h_n has normalized to 1.

Finally, the steps of FSVD algorithm are given in Table I.

VI. EXPERIMENT SETUP

In this experiment, we adopt the parameters in EPC C1 Gen2 [20] to establish UHF short-range wireless networks to evaluate the performance of the proposed algorithm. In the networks, a reader acts as a central receiving node and several sources act as other transmitting nodes. The reader allows the sources to randomly select time slots to send their signals. If there is a collision in a slot, the collision signal in the slot will be decoded. The experiment gives results through simulation data and measured data, respectively. The setup of the experiment is as follows.

A. Data Generation

1) Simulation Data

In this simulation, a random multiple access method uses ALOHA, where a frame length and sources are both 128. Since the two are equal, the average number of collision sources in a collision slot is about 2.33 [9]. Thus, we mainly simulate the collision signals of 2, 3, and 4 sources in a collision slot. The simulated collision signal is from (3), i.e. a complex baseband signal added a Gaussian white noise. The main parameters are from EPC C1 Gen2 standard and the details are shown in Table II.

In addition, Table III gives signal fading coefficients for 2, 3 and 4 sources, respectively. The performance of the traditional Viterbi [22] is related to the signal coefficients. Some coefficient values will make the Viterbi fail to search a survival path. The fading coefficients given in Table III will distinguish the performance of the traditional Viterbi and the proposed FSVD.

2) Measured Data

For measured data, we use a universal software radio peripheral (USRP) to establish a UHF short-range wireless network, whose parameters is still from EPC C1 Gen2. Its software is implemented with GNU Radio [24], and its code

TABLE III
SIGNAL FADING COEFFICIENT SETTING

Tag	2	3[1]		4
h_1	$0.3e^{\pi i/4}$	$0.2e^{\pi i/8}$	$0.02e^{\pi i/8}$	$0.2e^{\pi i/8}$
h_2	$0.5e^{\pi i/8}$	$0.3e^{\pi i/4}$	$0.03e^{\pi i/11}$	$0.4e^{3\pi i/8}$
h_3		$0.6e^{3\pi i/8}$	$0.05e^{3\pi i/8}$	$0.3e^{2\pi i/9}$
h_4				$0.1e^{\pi i/7}$

Note: 1. The coefficients in the first column are used in Fig. 10 and 13, and the second column used in Fig. 15.

TABLE IV
PARAMETERS FOR USRP

Parameter	Description
Motherboard	USRP N200
Daughter board	RXF900
Antenna	
Quantity	2
Type	Circularly polarized
Gain	7 dBic
Distance	0.5-1.5 m
Link frequency	40 kHz
Maximum queries	1000
Encoding	FM0
Transmission power	17.8 dBm
Emission amplitude	0.1
Topology	Tags are randomly placed in front of a reader, as long as they are not fully overlapped.

TABLE V
EVALUATED ALGORITHM SETTING

Algorithm	Description
FSVD	
Clustering Method[3]	Kmeans, Grid clustering
Nonlinear Programming	Genetic algorithm[1]
Decoding	Viterbi
Channel	Unknown
C-Viterbi[2]	
Decoding	SIC Viterbi decoding
Channel	Assuming known
H-Viterbi[2]	
Decoding	Viterbi decoding
Channel	Assuming known
FS-Jump	
Clustering Method[3]	Kmeans[3], Grid clustering[3]
Decoding	Phase hopping decoding
Channel	Unknown

Note: 1. GA algorithm code from MATLAB.
2. Viterbi use Euclidean distance to calculate the path likelihood, in instead of a hard decision like Hamming distance.
3. Kmeans for simulation data and grid clustering [22] for measured one.

download address is https://github.com/nkargas/Gen2-UHF-RFID-Reader. The USRP platform uses a USRP N200 motherboard and RXF900 daughter board to act as a reader, i.e. a central receiving node, and several EPC C1 Gen2 sources act as other transmitting nodes. Besides, the daughter board is equipped with two circularly polarized antennas, one for sending and the other for receiving. Table IV gives some parameters in the software radio. Finally, the measured data can be captured by the USRP reader and collision signals will come from the segment of collision detect RN16 [23], as shown in Fig. 8.

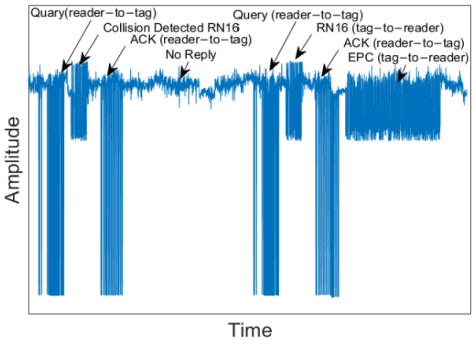


Fig.8. Wave captured by USRP reader from EPC C1 Gen2.

B. Evaluated Algorithm

In order to evaluate the performance, this paper compares four algorithms. The first is the proposed FSVD, and its steps are given in Table I. The second is the traditional Viterbi algorithm [22], denoted by C-Viterbi, The Viterbi algorithm can use Hamming distance or Euclidean distance to calculate the likelihood of each path when decoding. If Hamming distance is used, it is a hard decision and the result is not optimal. Here, we use Euclidean distance to calculate the likelihood distance. The algorithm uses successive interference cancellation to decode each collision signal source, respectively. The third is the Viterbi algorithm with known fading coefficients (e.g. adopting the method in [11, 26] to estimate them), denoted by H-Viterbi. The algorithm separates collision signals by known fading coefficients [10-14], and then performs Viterbi decoding on the separated signals. The fourth is phase-jump decoding by FSVD algorithm without known fading coefficients, denoted by FS-Jump (the decoding is based on the encoding rule of FM0 [15,16] and does not require searching for an optimal path. After the separated signal is passed through the matched filter, it is detected whether there is a jump in each symbol period. If there is a jump, it will be decoded as 0, otherwise it will be decoded as 1). The algorithm firstly separates the collision signals by FSVD, and then phase-jump decode the separated signals. The details in the above algorithm are given in Table V. Besides, it is noted that FSVD and FS-Jump both use two clustering methods, K-means and grid clustering, one for simulated data and the other for measured data. The reason is that K-means has better clustering performance for simulated data with lower SNR, but has higher complexity. Grid clustering [17] has lower complexity, but has good performance only for real data with higher SNR.

C. Experiment Metric

In this experiment, we will use the following metrics to evaluate the algorithms.

- Bit error rate (BER) is defined as the number of decoded symbols with error N_S above the total number N_T, shown as

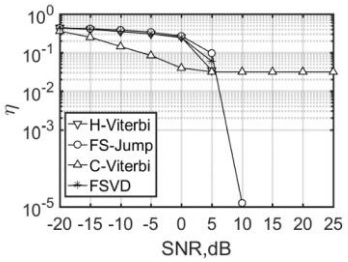


Fig.9. BER with 2 collision sources.

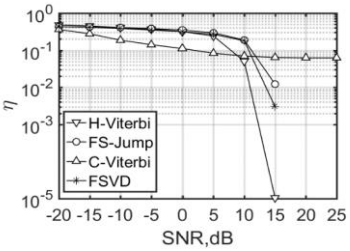


Fig.10. BER with 3 collision sources.

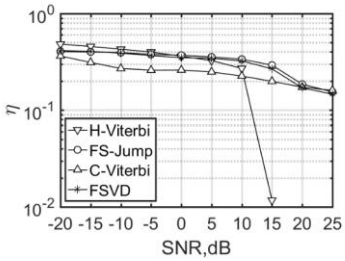


Fig.11. BER with 4 collision sources.

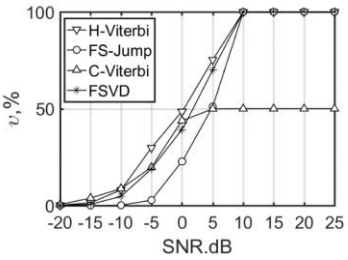


Fig.12. Decoding efficiency with 2 collision sources.

$$\eta = \frac{N_S}{N_T} \tag{16}$$

- Decoding efficiency is defined as the average number of successfully decoded source packet K_S in a collision slot above the number of collision sources K_T, shown as

$$\upsilon = \frac{K_S}{K_T} \tag{17}$$

where a symbol packet will be considered as a successfully decoded packet if and only if all symbols in the packet are successfully decoded.

- Throughput is defined as the average number of successfully decoded symbol packets in a frame L_S above the frame length L_T, shown as

$$\gamma = \frac{L_S}{L_T} \tag{18}$$

• Decoding time T_D is defined as the average time that each algorithm spends to decode each collision signal in a collision time slot. The metric is to measure the complexity of the algorithm. All data processing is performed under an ASUS FL8000U PC computer with a CPU, Intel Core i7-8550U.

VII. Experimental Results and Analysis

In this experiment, we mainly compare the four algorithms of h-Viterbi, FSVD, FS-Jump and c-Viterbi. H-Viterbi assumes fading coefficients to be known and thus perform supervised collision signal separation. FSVD performs unsupervised collision signal separation with unknown fading coefficients. Since both of the algorithms use Viterbi decoding, the unsupervised separation performance of FSVD will be close to that of the supervised separation with known fading coefficients if FSVD's performance is close to H-Viterbi. FS-Jump and FSVD both use the collision separation with unknown fading coefficients. The former uses phase-jump decoding and the latter uses Viterbi decoding. If FSVD performs better than phase-jump decoding, it means that Viterbi performs better than phase-jump. C-Viterbi directly decodes sources from the collision signal instead of separating first. Under some fading coefficients, C-Viterbi will misjudge a survival path. This simulation uses the fading coefficients and let c-Viterbi compared with the other methods to show the misjudgment.

A. Simulation Data

In the simulation, we use MATLAB software to evaluated the performance of the above algorithms.

When SNR ranges from -20 to 25 dB, we give BER curves for 2, 3, and 4 sources, shown in Fig 9, 10, and 11. As can be seen from Fig.9, when SNR is between -5 dB and 5 dB, FSVD's BER curve is between H-Viterbi and FS-Jump. That is, the performance of the FSVD algorithm is close to H-Viterbi with known fading coefficient, and is better than FS-Jump with the same separation algorithm. The result shows that FSVD's separation performance is close to the supervised clustering performance. Since FSVD's separation algorithm is the same as FS-Jump, the result also shows that Viterbi decoding is better phase-jump decoding. In addition, when SNR is between 5 dB and 25 dB, C-Viterbi's curve tends to be approximately horizontal. The result shows that its BER cannot be reduced even if SNR is increased. The reason is that, the fading coefficients in Table III make C-Viterbi search for an invalid survivor path and thus not decode more sources. The result in Fig. 10 are similar to that in Fig. 9. FSVD's BER curve is still between H-Viterbi and FS-Jump, and C-Viterbi cannot decode more sources. In Fig. 11, the best performance is still H-Viterbi with known fading coefficients, while FSVD and FS-Jump's BER curves decrease with SNR. Note that the BER in Fig. 11 is the average result of four sources, so the BER does not occur on every source evenly. From the separation efficiency in Fig. 14, it can be seen that although the BER is about 10^{-1}, the separation efficiency can exceed 50%. The result shows that when SNR increases, the three algorithms can decode more

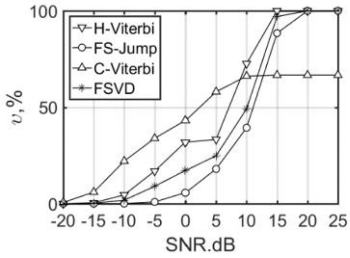


Fig.13. Decoding efficiency with 3 collision sources.

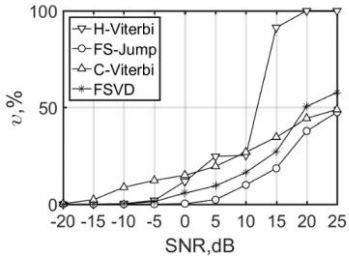


Fig.14. Decoding efficiency with 4 collision sources.

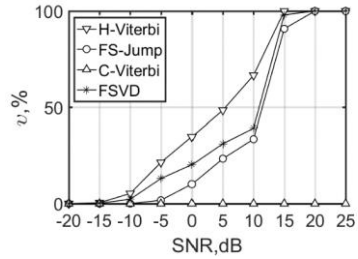


Fig.15. Decoding efficiency with 3 collision sources for another group of fading signal coefficients.

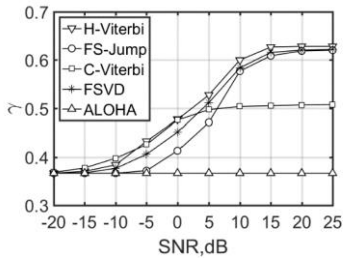


Fig.16. Throughput of separated collision resolutions in ALOHA system.

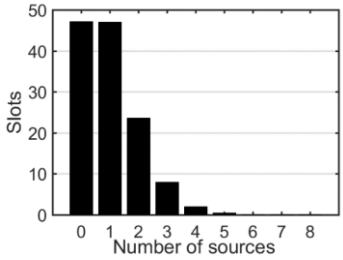


Fig.17. The number of collision slots with 0 to 8 sources in ALOHA system where a frame length and collision time slot are both 128.

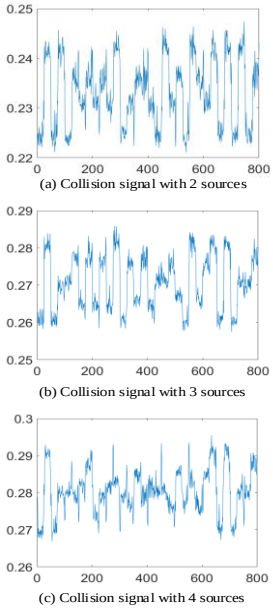


Fig.18. Module of complex collision signal captured by USRP.

sources. However, their BERs increase, compared with Fig 9 and 10 because the clustering accuracy under 4 sources is not as good as that 2 or 3 sources. For C-Viterbi, although its BER decreases with SNR, due to the fading coefficients, when SNR increases, the algorithm still cannot decode more sources.

When SNR ranges from -20 to 25 dB, we give decoding efficiency curves with 2, 3 and 4 sources, as shown in Fig. 12,13, and 14. Similar to the BER curves, FSVD's efficiency curve is between H-Viterbi and FS-Jump, and the decoding efficiency of C-Viterbi cannot reach 100% even under higher SNR. The result shows that the decoding efficiency of FSVD is affected by SNR or the number of collision sources. When SNR increases or the number of sources decreases, FSVD's efficiency will increase. On the other hand, the performance of C-Viterbi is related to the fading coefficients of sources. Given the coefficients in Table III, it will make C-Viterbi not to correctly decode some sources. Fig. 15 shows another decoding efficiency curve of C-Viterbi, where this group of fading coefficients is also given in Table III. Compared with Fig. 13, the decoding efficiency of C-Viterbi changes into 0. Even if SNR increases, its decoding efficiency cannot be advanced and no sources can be decoded under the group of coefficients.

When SNR ranges from -20 to 25 dB, we give the throughput of the above algorithms embedded in ALOHA of the short-range network, as shown in Fig. 16. In addition, the figure shows the throughput of an ALOHA system without separated collision resolutions, which is close to a theoretical value of 0.367. From the figure, when SNR increases, the throughput of the separated collision resolutions will be greater than that of the ALOHA system without separated resolutions. This indicates that the separated collision resolution algorithm can indeed increase the throughput of ALOHA. Like the results of the BER and decoding efficiency, FSVD's throughput is between H-Viterbi and FS-Jump and reach about 0.6 at SNR of 15, while C-Viterbi 's throughput can only reach about 0.5. In particular, when SNR=5dB, the decoding efficiency of FSVD

TABLE VI
RESULTS OF EVALUATED ALGORITHMS FOR MEASURED DATA[1]

Collided number	Index	FSVD	FS-Jump	C-Viterbi
	η	0	0	0.59
2	υ (%)	100	100	0
	T_D (ms)	19	14	6.5
	η	0	0.021	0.63
3	υ (%)	100	67	0
	T_D (ms)	48	41	9.8
	η	0.35	0.39	0.55
4	υ (%)	6.1	3.7	0
	T_D (ms)	9.9E4	9.9E4	11
/	γ	0.61	0.59	0.37

Note: 1.Setup of the experimental results can be seen in Table IV.

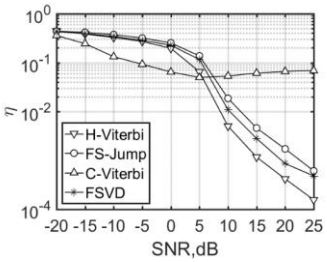


Fig.19. BER of the 2 collision sources in Ricean quasi-static fading channel.

algorithm is 60% for the case of 2 collision sources and 55% for 3 collision sources, and the throughput is 0.53.

In addition, note that Fig. 9-14 give the simulation results only for 2, 3, and 4 collision sources, because there are rarely more than 4 sources in a collision slot if a frame length and the number of collision sources are equal. Fig. 17 gives the number of collision slots with 0 to 8 sources, respectively when the frame length and collision time slots are both 128. As shown in Fig. 17, the number of collision slots with more sources is less. In particular, the number of slots where the collision sources exceed 4 is already very small.

B. Measured Data

The measured data in this subsection is generated by the USRP platform. The collision signals with 2, 3 and 4 sources, are given in Fig. 18, where the signal amplitudes are the module of their complex signals from I and Q channels. Table VI gives the decoding results for the measured signals, the decoding efficiency of the FSVD algorithm is 100% for the case of 2 tags and 3 tags, and the throughput is 0.61. Note that since the fading coefficients in the measured data is difficult to be obtained in advance, the experiment results of H-Viterbi are not given. From the table, FSVD has the best BER, decoding performance and throughput. On the other hand, C-Viterbi fails to decode the collision signal, so its throughput can only approach an ALOHA system without separation. However, it can also be seen from the table that FSVD's time for decoding 2 and 3 sources is about 3 times and 5 times that of C-Viterbi, respectively, and decoding 4 sources will take more time. In Fig. 17, when a frame length is equal to the number of sources, the number of slots with more than 3 collision sources does not account for much and thus the case of decoding 4 sources will

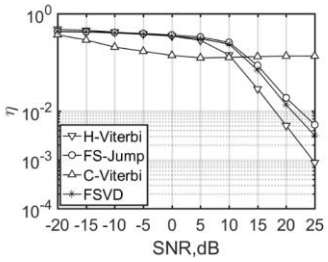


Fig.20. BER of the 3 collision sources in Ricean quasi-static fading channel.

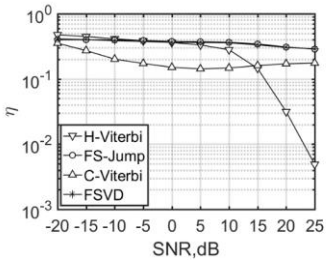


Fig.21. BER of the 4 collision sources in Ricean quasi-static fading channel.

also occur less. It is also worth noting that the decoding efficiency of C-Viterbi is 0. The reason is that after the received collision signal is removed from carrier leakage, the fading coefficients are all below 0.05. The result is similar to Fig. 15 where the searched survivor path fails. Of course, we can increase the signal transmission power or shorten the distance between sources and an antenna. However, this will increase the power consumption, or a closer distance may not always be guaranteed in some applications.

VIII. DISCUSSION

A. Random Ricean Quasi-static Channel Fading Coefficients

In the simulation experiment, it is not enough to give only the channel fading coefficient in Table III, which does not fully reflect the performance of each algorithm because the FSVD algorithm and the FS-Jump algorithm are related to clustering. If the fading coefficients of some sources are close, the number of clusters will be $M < 2^N$ and thus the performance of the algorithm will be greatly affected. Since the fading coefficient will change with the environment, the case above may occur. It should be more reasonable to evaluate the algorithms in a random channel than in a fixed channel. Here, we generate random Ricean quasi-static channel fading coefficients, and give the results of BER, decoding efficiency and throughput in such channels. The random Ricean quasi-static channel fading coefficients are generated by

$$H_{Ricean} = \sqrt{\frac{K_{fac}}{K_{fac}+1}} + \sqrt{\frac{1}{K_{fac}+1}} * H_{Rayleigh} \qquad (19)$$

where K_{fac} is Ricean factor which is set to 3.9811dB from this communication condition [27], and mean square value of H_{Ricean} is 0.8080.

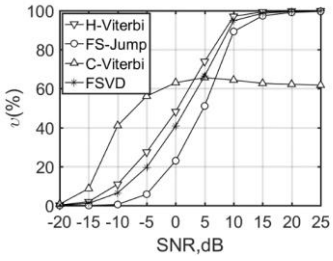


Fig.22. Decoding efficiency of 2 collision sources in Ricean quasi-static fading channel.

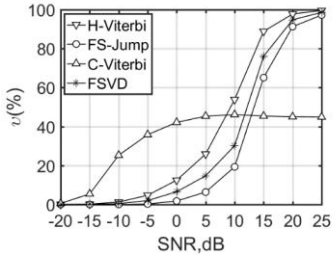


Fig.23. Decoding efficiency of 3 collision sources in Ricean quasi-static fading channel.

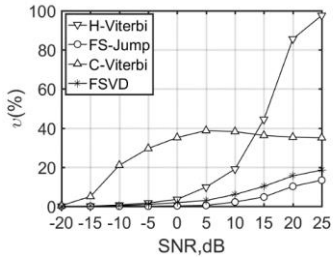


Fig.24. Decoding efficiency of 4 collision sources in Ricean quasi-static fading channel.

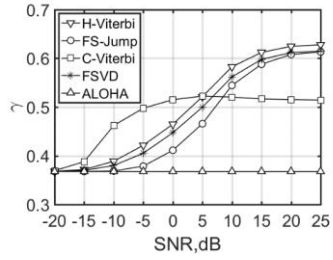


Fig.25. Throughput of separated collision resolutions in Ricean quasi-static fading channel.

Fig. 19-21 show the BER curves of 2, 3 and 4 collision sources, respectively. For 2 and 3 collision sources, except for the H-Viterbi algorithm with known channel state information, the proposed FSVD has the best BER performance. In the case of 4 collision sources, the performance of C-Viterbi algorithm is slightly better than that of FSVD, which is similar to the result of BER using a fixed channel. The reason is that FSVD adopting genetic algorithm sometimes may obtain a local optimum in the case of 4 sources, which will cause more errors. Fig. 22-24 show the decoding efficiency curves of 2, 3 and 4 collision sources, respectively. The results are consistent with

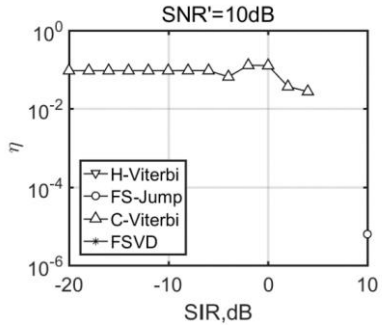


Fig.26. When *SNR*'=10 dB, BERs of 2 collision sources changes with *SIR*.

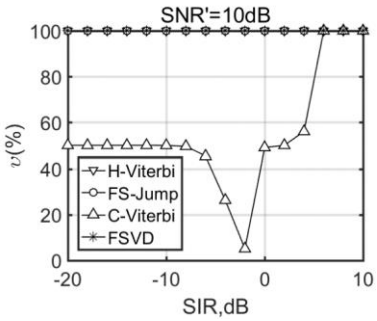


Fig.27. When *SNR*'=10 dB, decoding efficiencies of 2 collision sources changes with *SIR*.

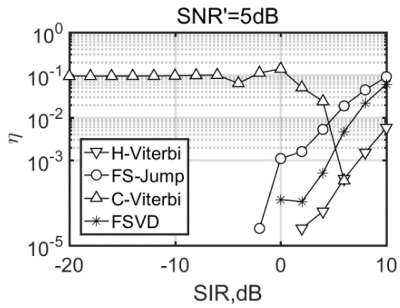


Fig.28. When *SNR'*=5 dB, BERs of 2 collision sources changes with *SIR*.

Fig. 19-21. Except for the 4 collision sources, FSVD has better decoding efficiency performance. Note that in the dynamic ALOHA system, there are fewer cases of 4 collision sources in a time slot, as shown in Fig. 17. Therefore, even if the C-Viterbi algorithm has better performance in decoding 4 collision sources, it may not have higher throughput. Fig. 25 shows the throughput curve under the random Ricean channel. When SNR is greater than 7 dB, the throughput of FSVD is greater than C-Viterbi. When SNR=7 dB, the decoding efficiency for the case of 2 collision sources and 3 collision sources are 76% and 25% respectively, and the throughput is 0.52.

B. Signal to Interference and Noise Ratio (SINR)

In the previous experiment, we give the results with varying SNR. However, the algorithm may also be related to the signal-to-interference ratio, that is, the relative power between the signal sources. Here, we consider the case for two collision sources. The case for three or more than three collision sources

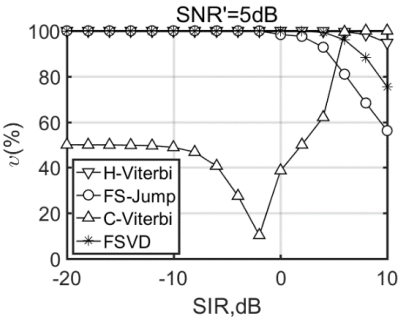


Fig.29. When *SNR'*=5 dB, decoding efficiencies of 2 collision sources changes with *SIR*.

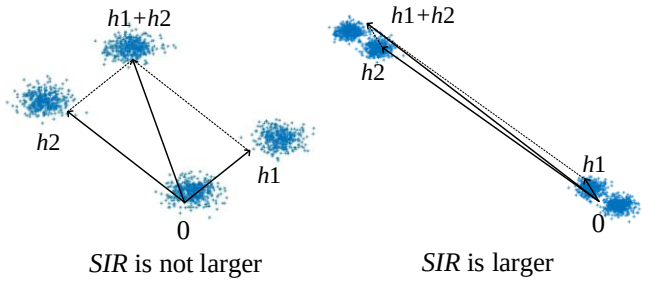


Fig.30. Clusters for 2 collision sources with different *SIR*.

are similar to the case for two ones. We define *SIR* and *SNR′* as

$$SIR = 10\lg\frac{P_2}{P_1} \tag{20}$$

$$SNR' = 10\lg\frac{P_1}{P_N} \tag{21}$$

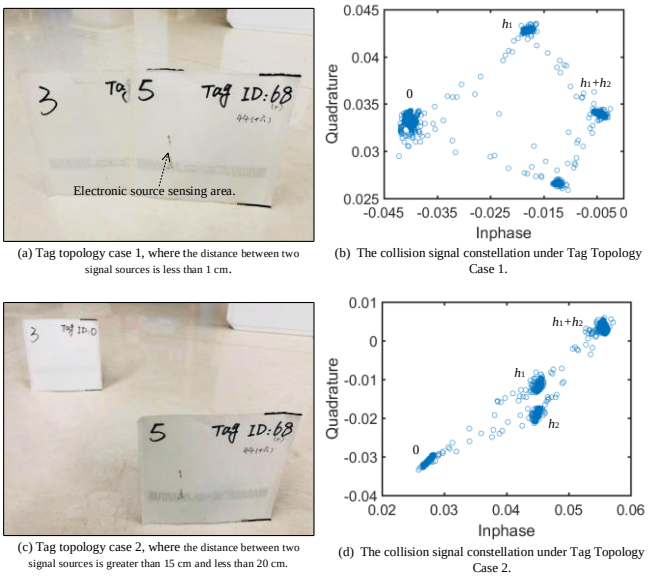
respectively, where

P_1 is the signal power of source 1, its channel fading coefficient adopts source 1 in column 1 of Table III, and its modulus and phase are both fixed,

P_2 is the signal power of source 2 and the phase of its channel fading coefficient adopts source 2 in column 1 in Table III, but its modulus will change with *SIR*,

P_N is a noise power and changes with *SNR′*.

Fig. 26-29 show the BER and decoding efficiency changing with *SIR* under different *SNR′*. In Fig. 26, when *SNR′* =10 dB, except C-Viterbi, the three ones have no errors, so their efficiency also achieve 100% in Fig. 28. Note that the BER of C-Viterbi has a peak around *SIR* =0 dB. The reason is as follows. C-Viterbi directly decodes the strongest signal from the collision signal, and then uses successive interference cancellation (SIC) to decode the next strongest signal. Weaker signal will be considered as interference in each decoding. If *SIR* =0dB, the power of the signal and the interference are nearly equal. In this case, therefore, the algorithm cannot judge which signal is interference signal and will produce too many errors. In the USRP hardware experiment, we can calculate the channel fading coefficients of different sources. According to (20), after 10 experiments, the average SIR can be about 1.9371 dB. We subtract the expected source signal from the original



(a) Tag topology case 1, where the distance between two signal sources is less than 1 cm.

(b) The collision signal constellation under Tag Topology Case 1.

(c) Tag topology case 2, where the distance between two signal sources is greater than 15 cm and less than 20 cm.

(d) The collision signal constellation under Tag Topology Case 2.

Fig.31. Different tag topology scenarios and their corresponding collision signal constellation.

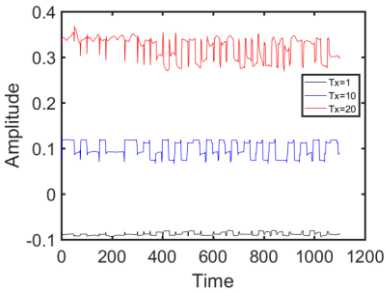


Fig.32. Waveform of USRP measured data at different Tx.

collision signal to get the noise signal. According to $SNR = 10\lg(P_{\text{signal}}/P_{\text{noise}})$, after 10 experiments, we can get an average SNR of about 12.0276 dB.

In Fig. 28 of *SNR′* =5 dB, except C-Viterbi, all other three algorithms' BERs increase with *SIR* when *SIR* >0 dB. The reason is that the performance of the three algorithms is related to the clusters from collision signals on an IQ plane. When *SIR* is not larger, as shown in Fig. 30, the distance between each cluster is larger and clustering can get better results. However, when *SIR* increases, the amplitude of one source becomes larger and the distance between clusters becomes closer, as shown in Fig. 30. In this case, the clustering effect will be poor if the noise power increases. Thus, more errors will produce. However, if *SIR* is large, the proposed method is not necessarily required. Capture effect [28, 29] will occur while the collision signal decoded, that is, the signal with the strongest power can be directly decoded. As shown in Fig. 30, it can be approximately regarded as only two clusters. No collision can be detected because the relation of the number of clusters and the number of sources, $M = 2^N$ can know N =1.

Therefore, the influence of SIR on the algorithms is as follows. When the signal and interference power are close, C-Viterbi will generate more bit errors. On the other hand, when

the signal power is much greater than the interference power, the proposed algorithm will cause bit errors, but the capture effect can be used to directly decode the collision signal.

The 2 sources placement in two different scenarios is shown in Fig. 31, where (a) is the scenario where the distance between two signal sources is less than 1 cm, (c) is the scenario where the distance between two signal sources is greater than 15 cm and less than 20 cm. However, due to the limited performance of our hardware equipment, we cannot collect collision RN16 signals when the distance between signal sources is greater than 20 cm. The reason is that when the distance is too large, the electromagnetic wave energy radiated by our antenna cannot make the signal source active. Although the channel fading coefficient of the tag is slightly different, it is still relatively close, that is, the *SIR* is close to 0 dB. The USRP platform transmitter daughterboard is SBX-400, recommended by reference [24], its open-source code on https://github.com/nkargas/Gen2-UHF-RFID-Reader does not have an adjustable transmit power parameter, so we choose the variable "self.rx_gain" in the "reader.py" file under the "Gen2-UHF-RFID-Reader/gr-rfid/apps/" directory to adjust the received signal power, and the values are 1, 5, 10, 15, 20 dBm, as shown in Fig. 32. In the experimental results, the average bit error rate of the traditional algorithm C-Viterbi is 0.3667, which still cannot be successfully decoded. However, the bit error rate of the FSVD algorithm is 0, the performance is better.

C. Number of Sources

Finally, the proposed method assumes perfect knowledge of the number of collision sources. If the number N is unknown, however, the method will be limited. Actually, some literatures [11, 26] have study the problem. The frequency and modulation method in the above references are the same as ours. They perform IQ modulation on the collision signal and then map it to a complex plane to determine the number of clusters. If the number of clusters M is determined, the number of conflicts N will also be determined, because of the relationship $M = 2^N$. Since we consider only the case of 1, 2, 3 and 4 sources, we estimate that the number of clusters can only be 2, 4, 8 and 16. The estimation methods are briefly described as follows.

Reference [11] uses the "Slot State Detection Algorithm (SSDA)". It firstly divides the samples on an IQ plane into small grids, and then calculate the number of clusters by finding the local maximum of the sample density. Reference [26] adopts the "improved frequency shift algorithm". It updates the cluster centers in a denser direction through Euclidean distance to find the densest points, and finally get the number of collision sources. Since the methods solve the estimation very well, we do not focus on the problem in this paper.

IX. Conclusion

For a short-range wireless network with an ALOHA system, this paper uses a separated collision resolution to decode collision signals to improve the network throughput. In a simulation experiment, the throughput of the proposed FSVD is about 0.62 at a SNR of 15 dB, and the throughput of the H-Viterbi algorithm with known fading coefficients is about 0.63. The difference between the two is only 0.01, but the FSVD does not need to estimate the coefficients. On the other hand, a traditional C-Viterbi algorithm will result in misjudgment of a survival path under some fading coefficients. Compared with an ALOHA without separation, the throughput of C-Viterbi has not been greatly improved. This paper also uses a software radio platform to establish an ALOHA near-field communication network from EPC C1 Gen2 standard. For the captured data from the platform, FSVD's throughput reaches 0.61 and is 0.25 higher than the traditional ALOHA system.

FSVD uses unsupervised clustering and a finite symbol optimization method to separate collision signals, and does not require signal fading information. However, it takes more time to complete clustering and optimizing. When there are 3 collision signal sources, it takes 48ms, shown in Table VI. On the other hand, the link frequency is 150kHz. Thus, that may not be necessarily real-time. The reason why it takes 48ms is that the algorithm uses a brute search and runs on a PC. Especially when separating 4 collision signal sources, the algorithm takes about 10 seconds on average. Note that in a high-efficiency ALOHA network, the frame length needs to be set to the number of signal sources. In this case, the probability of collision of 4 signal sources is about 1%. In order to save time, the collision slot with 4 sources may not be separated. Indeed, our algorithm has higher complexity, which is caused by the collision signal separation.

The algorithm code in this paper have been uploaded to GitHub. Its download address is https://github.com/monk5469/collision-separation.

Appendix

From (4)

$$\mathbf{H} = \mathbf{D}^{\dagger}\mathbf{C} \tag{A-1}$$

Substituting (A-1) into (4) has

$$\mathbf{C} = \mathbf{D}\mathbf{D}^{\dagger}\mathbf{C} \tag{A-2}$$

Let $\tilde{\mathbf{D}}$ be an estimation of the dictionary matrix $\mathbf{D}$. Then, when the sum of squared errors

$$\sum_{m=1}^{M} e_m^2 = \mathbf{e}^{\mathrm{H}}\mathbf{e} = (\tilde{\mathbf{D}}\tilde{\mathbf{D}}^{\dagger}\mathbf{C} - \mathbf{C})^{\mathrm{H}} (\tilde{\mathbf{D}}\tilde{\mathbf{D}}^{\dagger}\mathbf{C} - \mathbf{C}) \tag{A-3}$$

is minimum, the optimal estimate is obtained, expressed as

$$\hat{\mathbf{D}} = \arg\min_{\tilde{\mathbf{D}} \in \mathcal{D}} \| \tilde{\mathbf{D}}\tilde{\mathbf{D}}^{\dagger}\mathbf{C} - \mathbf{C} \|_2 \tag{A-4}$$

Simplify the symbols in (A-4), let $\mathbf{D} = \tilde{\mathbf{D}}$ and then substituting it into (A-4) will have (7).

References

[1] B. Bansal and S. Rana, "Internet of Things : Vision , Applications and Challenges," Int. J. *Eng. Trends Technol*., vol. 47, no. 7, pp. 380–384, May. 2017, DOI: 10.14445/22315381/IJETT-V47P263.

[2] I. Makhdoom, M. Abolhasan, H. Abbas, and W. Ni, "Blockchain's Adoption in IoT: The Challenges, and A Way Forward," J. *Netw. Comput. Appl*., vol. 125, pp. 251–279, Oct. 2018, DOI: 10.1016/j.jnca.2018.10.019.

[3] V. Coskun, B. Ozdenizci, and K. Ok, "A Survey on Near Field Communication (NFC)," *Wirel. Pers. Commun.*, vol. 71, no. 3, pp. 2259–

2294, Dec. 2013, DOI: 10.1007/s11277-012-0935-5.

[4] S. Ali, E. Hossain, and D. I. Kim, "LTE/LTE-A Random Access for Massive Machine-Type Communications in Smart Cities," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 76–83, Jan. 2017, DOI: 10.1109/MCOM.2017.1600215CM.

[5] D. Tung, C. Wong, and Q. Chen, "Multi-Channel Pure Collective Aloha MAC Protocol with Decollision Algorithm for Satellite Uplink," *2018 IEEE 4th World Forum Internet Things*, pp. 251–256, 2018, DOI: 10.1109/WF-IoT.2018.8355107.

[6] S. Chae, K. Kang, and Y. Cho, "A randomized adaptive neighbor discovery for wireless networks with multi-packet reception capability," J. *Parallel Distrib. Comput.*, vol. 131, pp. 235–244, Nov. 2018, DOI: 10.1016/j.jpdc.2018.11.010.

[7] I. B. Arun and T. G. Venkatesh, "Design and Performance Analysis of a MAC protocol for wireless LANs supporting Multipacket Reception," J. *Netw. Comput. Appl.*, vol. 87, pp. 223–236, Mar. 2017, DOI: 10.1016/j.jnca.2017.03.010.

[8] L. Beltramelli, A. Mahmood, P. Osterberg, and M. Gidlund, "LoRa beyond ALOHA: An Investigation of Alternative Random Access Protocols," *IEEE Trans. Ind. Informatics*, vol. 3203, pp. 1–11, May. 2020, DOI: 10.1109/TII.2020.2977046.

[9] F. Vazquez-gallego and S. Member, "Goodbye , ALOHA !," *IEEE Access*, vol. 4, pp. 2029–2044, Feb. 2016, DOI: 10.1109/ACCESS.2016.2557758.

[10] C. D. Operator-based, B. Guo, S. Peng, and X. Hu, "Complex-valued Differential Operator-based Method for Multi-component Signal Separation," *Signal Processing*, vol. 132, pp. 66–76, Sep. 2016, DOI: 10.1016/j.sigpro.2016.09.015.

[11] Z. Huang, J. Su, G. Wen, W. Zheng, and C. Chu, "A Physical Layer Algorithm for Estimation of Number of Sources in UHF RFID Anti-Collision Design," *Comput. Mater. Contin.*, vol. 61, no. 1, pp. 399–408, 2019, DOI: 10.32604/cmc.2019.05876.

[12] C. Angerer, S. Member, R. Langwieser, M. Rupp, and S. Member, "RFID Reader Receivers for Physical Layer Collision Recovery," IEEE Trans. Commun., vol. 58, no. 12, pp. 3526–3537, 2010. DOI: 10.1109/TCOMM.2010.101910.100004.

[13] A. Bletsas, J. Kimionis, S. Member, A. G. Dimitriou, and G. N. Karystinos, "Single-Antenna Coherent Detection of Collided FM0 RFID Signals," IEEE Trans. Commun., vol. 60, no. 3, pp. 756–766, 2012. DOI: 10.1109/TCOMM.2011.020612.110212.

[14] P. Hu, P. Zhang, and D. Ganesan, "Laissez-Faire: Fully Asymmetric Backscatter Communication," Comput. Commun. Rev., vol. 45, no. 5, pp. 255–267, 2015. DOI: 10.1145/2785956.2787477.

[15] Y. G. Kim, S. Member, and A. J. H. Vinck, "Anti-Collision Algorithms for FM0 Code and Miller Subcarrier Sequence in RFID Applications," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5168–5173, 2018, DOI: 10.1109/TVT.2018.2817587.

[16] J. Li, H. Wu, and Y. Zeng, "Recovery of collided RFID sources with frequency drift on physical layer," *IEEE/CAA J. Autom. Sin.*, vol. PP, pp. 1–11, 2019, DOI: 10.1109/JAS.2019.1911720.

[17] D. Shen, G. Woo, D. P. Reed, A. B. Lippman, and J. Wang, "Separation of multiple passive RFID signals using software defined radio," *2009 IEEE Int. Conf. RFID*, pp. 139–146, 2009, DOI: 10.1109/RFID.2009.4911203.

[18] B. Nguyen and B. De Baets, "Kernel-Based Distance Metric Learning for Supervised-Means Clustering," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 30, no. 10, pp. 3084–3095, Dec. 2019, DOI: 10.1109/TNNLS.2018.2890021.

[19] W. C. Chen and C. D. Chung, "Spectrally Efficient OFDM Pilot Waveform for Channel Estimation," *IEEE Trans. Commun.*, vol. 65, no. 1, pp. 387–402, Jan. 2017, DOI: 10.1109/TCOMM.2016.2616859.

[20] Q. Zhang *et al.*, "Algorithms for Blind Separation and Estimation of Transmitter and Receiver IQ Imbalances," *J. Light. Technol.*,vol.37,no.10, pp. 2201–2208, 2019, DOI: 10.1109/JLT.2019.2899833.

[21] T. Kim and S. H. Chae, "A Channel Estimator via Non-Orthogonal Pilot Signals for Uplink Cellular IoT," *IEEE Access*, vol. 7, pp. 53419–53428, May. 2019, DOI: 10.1109/ACCESS.2019.2912446.

[22] A. Scaglione, T. Larsen, and S. Member, "Multipacket Reception of Passive UHF RFID Sources : A Communication Theoretic Approach," *IEEE Trans. SIGNAL Process.*, vol. 59, no. 9, pp. 4225–4237, Sep. 2011.

[23] *EPC™ Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz-960MHz Version 2.0. 1*, EPCglobal, G. S. (2015). Inc., Brussels, BE, 2015.

[24] N. Kargas, F. Mavromatis, and A. Bletsas, "Fully-Coherent reader with commodity SDR for Gen2 FM0 and computational RFID," *IEEE Wirel. Commun. Lett.*, vol. 4, no. 6, pp. 617–620, 2015, DOI: 10.1109/LWC.2015.2475749.

[25] A. A. Rahmani Hosseinabadi, J. Vahidi, B. Saemi, A. K. Sangaiah, and M. Elhoseny, "Extended Genetic Algorithm for solving open-shop scheduling problem," *Soft Comput.*, vol. 23, no. 13, pp. 5099–5116, Apr. 2019, DOI: 10.1007/s00500-018-3177-y.

[26] Hou Y, Ou J, Zheng Y, et al. PLACE: Physical layer cardinality estimation for large-scale RFID systems[J]. IEEE/ACM transactions on networking, vol. 24, no. 5, pp. 2702–2714, 2016. DOI: 10.1109/TNET.2015.2481999.

[27] Medawar S , Handel P , Zetterberg P , "Approximate Maximum Likelihood Estimation of Rician K-Factor and Investigation of Urban Wireless Measurements,". IEEE Transactions on Wireless Communications, pp. 2545-2555, 2013, DOI: 10.1109/TWC.2013.042413.111734.

[28] Haifeng Wu, Yang Wang and Yu Zeng, "Capture-aware Bayesian RFID Tag Estimate for Large-scale Identification," IEEE/CAA J. Autom. Sinica, vol. 5, no. 1, pp. 119-127, Jan. 2018. DOI: 10.1109/JAS.2017.7510757.

[29] Haifeng Wu, Yu Zeng, "Passive RFID Tag Anti-collision Algorithm for Capture Effect," IEEE Sensors Journal, vol. 15, no. 1, pp. 218-226, Jan, 2015. DOI: 10.1109/JSEN.2014.2339653.

Haifeng Wu was born in Kunming, Yunnan Province, China in 1977. He 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 at the Department of Information Engineering in the Yunnan Minzu University. His research interests include neural signal processing, machine learning and mobile communications.

Xiaogang Wu was born in Weinan, Shaanxi Province, China in 1991. He received the B.E. degree in communication engineering from Xi'an technology and business college, Xi 'an, China, in 2013. He is currently pursuing his M. S. degree at the Department of Information Engineering at the Yunnan Minzu University. His research interests include mobile communications, wireless near field communication.

Yi Li was born in Bazhong, Sichuan Province, China in 1993. He received the B.E. degree in electronic information engineering from Sichuan Technology and Business University, Sichuan, China, in 2016. He is currently pursuing his M.S. at the Department of Information Engineering at the Yunnan Minzu University. His research interests include UHF RFID communication.

Yu Zeng was born in Kunming, Yunnan Province, China in 1981. She received the M.S. degree in electrical engineering from Yunnan University, Kunming, China, in 2006. She is currently an assistant professor at the Department of Information Engineering at the Yunnan Minzu University. Her research interests include wireless network, mobile communications.