A simplified decoding algorithm for multi-CRC polar codes

YANG Haifen*, YAN Suxin, ZHANG Hao, REN Yan, HU Xiangdong, and LIN Shuisheng

School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Abstract: Polar codes represent one of the major breakthroughs in 5G standard, and have been proven to be able to achieve the symmetric capacity of binary-input discrete memoryless channels using the successive cancellation list (SCL) decoding algorithm. However, the SCL algorithm suffers from a large amount of memory overhead. This paper proposes an adaptive simplified decoding algorithm for multiple cyclic redundancy check (CRC) polar codes. Simulation results show that the proposed method can reduce the decoding complexity and memory space. It can also acquire the performance gain in the low signal to noise ratio region.

Keywords: polar code, successive cancellation list (SCL), cyclic redundancy check (CRC), adaptive decoding.

DOI: 10.21629/JSEE.2020.01.02

1. Introduction

Polar codes have attracted widespread attention since it was first introduced by Arikan [1,2]. Its encoding and decoding complexity are $O(N \log N)$ and $N$ is the length of polar codes. In 2016, polar codes have been selected as the channel coding scheme of control channel in 5G [3]. How to further improve the decoding performance and reduce complexity to satisfy the low delay and high reliable communication scenario has become the focus issue.

On the code design, construction methods of the polar codes for the additional white Gaussian noise (AWGN) and Rayleigh fading channel based on the Gaussian approximation algorithm were investigated in [4] and [5]. Zhao et al. [6] proposed a puncturing method based on the decoding reliability. To improve the error floor performance of parallel concatenated systematic polar codes, a three-dimensional code scheme was given [7]. In [8], a low complexity coding scheme was proposed called asymmetric polar coding that allows for any arbitrary block length. Details on the generator matrix and frozen set design, etc., for flexible polar code lengths were shown in [9]. The corresponding distance properties and convergence analysis were given in [10,11].

The successive cancellation (SC) algorithm is widely used in polar decoding [12,13]. Permutation decoding was proposed in [14] with the latency of SC decoders. However, the SC algorithm is not ideal for moderate code length. In [15], it incorporated the Fano sequential decoding into the standard SC decoding to address the inherent drawback of SC decoding, e.g., one wrong-decision will surely lead to a wrong codeword. After that, some kinds of decoding algorithms were provided, such as the belief propagation (BP) polar decoder [16–19]. However, the existing stochastic BP decoders suffer from a relatively long decoding latency, resulting in low hardware efficiency. Meanwhile, sphere-decoding tree searches [20–22] and other fast decoding algorithms [23,24] were proposed, respectively. Deep learning was applied to decoding a polar code with respect to numerous configurations: the number of hidden layers, the number of nodes for each layer, and activation functions [25]. A successive cancellation list (SCL) decoder was presented in [26,27], where $L$ is the size of the list. Empirically, the use of several decoding paths yields an error probability comparable to that under the optimal maximum a posteriori (MAP) decoding with practical values of the list size. In addition, by adding only a few extra bits of cyclic redundancy check (CRC) precoding, the resulting performance is comparable with state of the art low-density parity-check (LDPC) codes [28].

SCL decoding can significantly improve the error correction performance of polar codes. However, it needs to keep $L$ best candidate paths at the same time, and this comes at the cost of a lower throughput and higher area occupation when implemented in the hardware, which greatly increases the amount of computation. In particular, the high area requirement of SCL decoding is mostly dominated by its memory usage. Several techniques were proposed to speed up the SCL decoder in [29–32]. However, the performance gains and the complexity cannot be simul-
CRC-aided decoding schemes were proposed to improve the performance of polar codes [33–37]. The unified description of SC decoding and its improved version with list or stack were provided and the CRC-aided SCL stack decoding schemes were proposed. Simulation results in the binary-input AWGN channel show that the SC decoding algorithm can provide significant gain of 0.5 dB over the turbo codes. Moreover, the time complexity of its decoder is much lower than that of the turbo decoder and can be close to that of the SC decoder in the high signal to noise ratio (SNR) region.

This paper proposes a multi-CRC aided adaptive simplified SCL (multi-CRC AD-SSCL) polar decoding scheme. Different from the existing multi-CRC aided SCL algorithms, a new partitioning principle is proposed based on the simplified SCL algorithm in this paper. The redundant computation in the proposed algorithm is removed by defining simplified nodes, which reduces the complexity. Compared with the conventional CRC polar scheme, the proposed algorithm divides the information sequence into multiple sub-sequences and then each sub-sequence is checked by a CRC. Furthermore, in order to ensure the decoding performance, the proposed algorithm adaptively adjusts the size of the list L until at least one path from sub-sequence can pass CRC. Different from the conventional CRC algorithm which adjusts the list value and re-decoding after the overall decoding is completed, the average list value required for the proposed algorithm will be effectively reduced especially at a low SNR. Theoretical analysis and simulation show that the proposed decoding algorithm can significantly reduce the computational complexity and memory overhead. In addition, it can acquire a certain performance gain in the low SNR region.

The rest of this paper is organized as follows. The coding and decoding of polar codes is given in Section 2. Section 3 introduces a construction of polar codes and gives the corresponding decoding algorithm. Simulation results are given and analyzed in Section 4. Finally, Section 5 concludes the entire paper.

### 2. Preliminaries

Polar codes can achieve the symmetric capacity of the binary-input discrete memoryless channel (B-DMC) through channel polarization, which consists of channel combining and channel splitting. We can construct a polar code (the length \( N = 2^n \)) as the following:

\[
x_1^N = u_1^N G_N = u_1^N B_N F^\otimes n
\]

where \( G_N \) is the generation matrix of the order \( N \), \( B_N \) is a bit-reversal permutation matrix, and \( F^\otimes n \) is the \( n \)th Kronecker product of \( F = [1 \; 0 \; \; 1 \; 1] \). \( x_1^N \) and \( u_1^N \) are the encoded and information vectors, respectively. According to the channel polarization theory, \( u_1^N \) can be divided into two subsets \( u_A \) and \( u_{A^c} \). \( u_A \in X^K \) is information bits, and \( u_{A^c} \in X^{N-K} \) is frozen bits.

Supposing there is the received vector \( y_1^N \), the SC decoding algorithm generates its decision \( \hat{u}_1^N \) by computing (2).

\[
\hat{u}_i = \begin{cases} 0, & i \in A^c \text{ or } a^{u_i} \geq 0 \\ 1, & \text{otherwise} \end{cases}
\]

where \( a^{u_i} \) represents the logarithmic likelihood ratio (LLR) of \( u_i \):

\[
a^{u_i} = \ln \frac{\Pr(y_1^N, \hat{u}_i^{-1} | u_i = 0)}{\Pr(y_1^N, \hat{u}_i^{-1} | u_i = 1)}.
\]

SC decoding can be represented as a binary tree \( T_n \) traversing, which starts from the root node and advances recursively from left to right as shown in Fig. 1(a). During the traversal process, SC involves two kinds of messages. Taking the node \( v \) at the stage \( s \) as an example, one is soft information (LLR) that is \( a^{vl} = [a_1, \ldots, a_{2^s}] \) and the other is hard decision estimates \( \beta^v = [\beta_1, \ldots, \beta_{2^s}] \). The message passing process of the node \( v \) is shown in Fig. 1(b), where \( a^{vl} \) and \( \beta^v \) can be approximated as in (4) and (5) for \( 1 \leq i \leq 2^s \).

\[
\alpha_i^{vl} = \text{sgn}(a_i^v) \text{sgn}(a_{i+2^{s-1}}^v) \min(|a_i^v|, |a_{i+2^{s-1}}^v|)
\]

\[
\alpha_i^{vr} = a_i^v + (1 - 2\beta_i^{vl})a_i^v
\]

where \( \beta^v \) can be calculated by

\[
\beta^v = [\beta^{vl} \oplus \beta^{vr} \beta^{vr}]
\]

![Binary tree](Image)

Fig. 1 Binary tree \( T_n \) and message passing in binary tree

For a leaf node \( u \), \( \beta^v \) is the hard decision. The computational complexity of the SC algorithm is \( O(N \log N) \).
Fig. 2 gives the bit error rate (BER) and frame error rate (FER) performance comparison with different code lengths at the code rate 0.5. It can be seen that the code with a longer length can bring more performance gains with the SNR increasing, while in the low SNR region the difference is ignorable.

It divides the information sequence \( S \) into multiple sub-sequences \( S_i \) and each sub-sequence is checked by a short CRC \( r_i \). Thus, compared with the conventional CRC polar encoding, multi-CRC requires a precoding module. In addition, segmenting in Fig. 3 should be carried out according to the segmentation principle as follows.

3. Multi-CRC simplified decoding scheme

Fig. 3 shows the encoding structure of the multi-CRC.

Segmentation principle  We define three special nodes in the binary tree \( T_n \) to reduce complexity: rate-0 nodes (with no information bits), rate-1 nodes (with no frozen bits), rep nodes (with one information bit in rightmost leaf). The structure of a special node cannot be destroyed in the segmenting process. If the node \( v \) at the stage \( s \) is a special node, then the length of the sub-information vector \( U_i \) corresponding to the node \( v \) should satisfy \( \min \text{length}(U_i) = 2^s \). This segmentation principle can reduce the calculation of LLR at least \( 2^s \) times for the sub-information vector \( U_i \) during the decoding process.

Proof  The length of the sub-information sequence \( \text{length}(U_i) < 2^s \) indicates that the special node is destroyed. The special node will degenerate to the ordinary node as shown in Fig. 4. The circle represents the common node and the triangle represents the special node. From Fig. 4, the node \( v \) that degenerates into an ordinary node needs to respectively pass LLR messages \( a^{vl} \) and \( a^{vr} \) to their sub-nodes \( v^l \) and \( v^r \). Since \( a^{vl} \) and \( a^{vr} \) are \( 1 \times 2^{s-1} \) vectors, compared with the node \( v \) based on the segmentation principle, the degenerated node \( v \) needs to compute \( 2^s \) more LLR values. Similarly, if the sub-node \( v^l \) or \( v^r \) of the special node \( v \) is destroyed, the sub-node \( v^l \) or \( v^r \) needs to calculate \( 2^{s-1} \) more LLR values.
The decoding process is shown in Algorithm 1. Generally, the path metric (PM) [31] can be computed as
\[ PM_{j,l} = PM_{j-1,l} + \sum_{i=1}^{2^s} \ln(1 + e^{-\alpha_i \beta_i}) \] (7)

Aimed at the special node \( v \), we can calculate the PM as follows.
Rate-0 nodes:
\[ PM_{j+2^s-1,l} = PM_{j-1,l} + \sum_{i=1}^{2^s} \ln(1 + e^{-\eta_i \alpha_i}) \] (8)
Rate-1 nodes:
\[ PM_{j+2^s-1,l} = PM_{j-1,l} + \sum_{i=1}^{2^s} \ln(1 + e^{-\eta_i \alpha_i}) \] (9)
Rep nodes:
\[ PM_{j+2^s-1,l} = PM_{j-1,l} + \sum_{i=1}^{2^s} \ln(1 + e^{-\eta_i \alpha_i}) \] (10)

where \( \eta_i = 1 - 2\beta_i (1 \leq i \leq 2^s) \), \( l \) is the path index and \( \hat{\alpha}_j \) is the estimate of bit \( j \) at path.

Algorithm 1  Multi-CRC AD-SSCL decoding algorithm.

1) Initiation decoder: \( L = 1 \)
2) For \( i = 1, 2, \ldots, M \)
   // aimed at every input sub-information sequence \( U_i \)
   3) for \( j = 1, 2, \ldots, N_i \)
      // aimed at every bit \( j \) of sub-information sequence \( U_i \)
      4) SSCL decoding via traverse the corresponding part of \( T_n \)
      5) if \( s \neq 0 \) // \( s = \text{stage}(v) \)
         6) if \( v \) is special node:
            7) calculate the PM by (8), (9) and (10), \( j = j + 2^s \),
            return to 3
         8) else
            9) traverse the next corresponding node \( v \), return to 4
         return to 3
      11) else // \( s = 0 \) that is \( v = u_j \)
       12) calculate the PM referring to (7), let \( j = j + 1 \),
       return to 3
      13) end if
      14) if \( j = K_i \)
      15) perform CRC detection to the current sub information bit \( i \), select and save one path by:
      16) Case 1  Save the path which fulfills CRC and has the lowest PM, let \( L = L \)
      17) Case 2  If no paths pass CRC:
      18) if \( 2L \leq L_{\max} \) //recalculate \( U_i \)
      19) let \( L = L_{\max}, i = i \), return to 2
      20) else if \( L = L_{\max} \) //calculate \( U_{i+1} \)
5) end if
6) else //recalculate \( U_i \)
7) let \( L = L_{\max}, i = i \), return to 2
8) end if
9) end for
10) end for

4. Simulation and results

In this section, we compare the performance of the proposed algorithm and the conventional CRC aided adaptive SCL. The simulation is based on binary phase shift keying (BPSK) modulation and the AWGN channel. The information index \( A \) adopts the 3rd generation partnership project (3GPP) standard in [3] and \( L_{\max} \) is set to 16 in simulation.

Firstly, the FER performance comparison of the conventional CRC and the three multi-CRC schemes are shown in Fig. 5. The generator polynomials of CRC are
\[ g_4(x) = x^4 + x^3 + 1, \]
\[ g_8(x) = x^8 + x^7 + x^6 + x^4 + x^2 + 1, \]
\[ g_{12}(x) = x^{12} + x^{11} + x^3 + x^2 + 1, \]
\[ g_{24}(x) = x^{24} + x^{23} + x^6 + x^5 + x + 1. \]

![Fig. 5](image-url) FER of multi-CRC vs. conventional CRC

It is shown in Fig. 5 that compared with the traditional algorithm, the three schemes can achieve certain performance gains at a low SNR. This is because the proposed algorithm uses the idea of segment decoding. The decoder can remove the paths which cannot timely pass the CRC from the path list to make the branches of the paths which can pass the CRC preserved as more as possible. Besides,
the proposed algorithm adaptively adjusts the value of the list \( L \) for SNR to improve the performance.

Fig. 6 gives the complexity of the multi-CRC simplified SCL and the conventional algorithm. The complexity of the SCL class mainly depends on the calculation units of LLR which is also called processing elements (PE). The PE computation of the proposed algorithm and the conventional algorithm can be calculated by (11) – (13). It is shown in Fig. 6 that the multi-CRC simplified SCL algorithm can efficiently reduce the complexity compared with the conventional algorithm. For example, the three methods of the proposed algorithm are only 32.8%, 32.9% and 39.2% of the conventional algorithm at 2 dB, respectively.

\[
\text{PE}_{\text{proposed}} = \sum_{i=1}^{M} \text{PE}_i
\]

(11)

\[
\text{PE}_i \approx L'_i \cdot [N_i \log(N_i) - \text{PE}_{\text{save}}(i)]
\]

(12)

where \( \text{PE}_i \) represents the PE size of the \( i \)th sub-information vector in (11). \( L'_i = L^\text{initial}_i + \ldots + L^\text{final}_i \) denotes the sum of all the lists that have been used in decoding the sub-information vector. In addition, \( \text{PE}_{\text{save}}(i) \) denotes to save PE size of the \( i \)th sub-information vector by simplified SCL decoding.

\[
\text{PE}_{\text{conventional}} = L' \cdot N \log N
\]

(13)

where \( L' = 1 + 2 \cdots + L \) means the sum of all the lists that have been used.

Furthermore, we calculate the memory overhead of two algorithms according to (14) and (15) respectively.

\[
M_{\text{proposed}} = \lceil K_i \rceil \cdot \lceil L_i \rceil + K, \quad 1 \leq i \leq M
\]

(14)

where \( L_i \) represents the final list size of the \( i \)th decoding sub-information vector in (14). \( K_i \) means the length of the sub-sequence \( C_i \).

\[
M_{\text{convention}} = K \cdot \bar{L}
\]

(15)

where \( \bar{L} \) represents the final list size of the conventional algorithm.

From Fig. 7 we see that the conventional algorithm needs \( M_{\text{convention}} \) memory cells to save the survival paths, while the proposed algorithms use memory multiplexing to reduce memory cells. For example, the three methods of the proposed algorithm are only 77.3%, 75.7% and 74.1% of the conventional algorithm at 1 dB, respectively.

In order to analyze the complexity of the proposed algorithm under different code lengths and code rates, Table 1 gives detailed numerical results. It can be seen that under all the situations, the proposed algorithm can reduce the decoding complexity.

| \( N \) | \( R \) | \( \text{List} \) | \( \text{PE}_{\text{conventional}} \) | \( \text{PE}_{\text{proposed}} \) | Ratio/% |
|---|---|---|---|---|---|
| 256 | 0.5 | 4 | 4 880 | 3 344 | 68.52 |
| 512 | 0.3 | 4 | 9 792 | 6 720 | 68.63 |
| 512 | 0.5 | 4 | 10 576 | 7 504 | 70.95 |
| 512 | 0.7 | 4 | 10 992 | 7 920 | 72.05 |
| 512 | 0.5 | 16 | 42 304 | 72 064 | 70.95 |
| 1 024 | 0.5 | 4 | 24 432 | 18 288 | 74.85 |

5. Conclusions

In this paper, we propose a multi-CRC simplified SCL polar decoding scheme which can reduce the computational complexity and memory cells in the decoding. The construction of polar codes based multi-CRC is given. The simulation results show that compared with conventional methods, the proposed methods can not only significantly reduce the computational complexity but also acquire a certain performance in the low SNR.

References

[1] ARIKAN E. Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. IEEE Trans. on Information Theory, 2009, 55(7): 3051 – 3073.

[2] ARIKAN E. Channel polarization: a method for constructing capacity-achieving codes. Proc. of the IEEE International Symposium on Information Theory, 2008: 1173 – 1177.
[3] 3GPP TS 36.212. Multiplexing and channel coding. https://www.3gpp.org/ftp/specs/.

[4] ELKELESH A. EBADA M. CAMMERER S. et al. Genetic algorithm-based polar code construction for the AWGN channel. Proc. of the 12th International ITG Conference on Systems, Communications and Coding, 2019: 1 – 6.

[5] ZHOU D. NIU K. DONG C. Construction of polar codes in Rayleigh fading channel. IEEE Communications Letters, 2019, 23(11): 402 – 405.

[6] ZHAO S. SHAO Z. CHEN H. Puncturing method for systematic polar code based on decoding reliability. Journal of Southeast University, 2017, 47(1): 23 – 26. (in Chinese)

[7] LIU Z. NIU K. DONG C. et al. Adding a rate-1 third dimension to parallel concatenated systematic polar code: 3D polar code. Wireless Communications and Mobile Computing, 2018, 18: 1 – 6.

[8] CAVATASSI A. TONELLIER T. GROSS W J. Asymmetric construction of low-latency and length-flexible polar codes. Proc. of the IEEE International Conference on Communications, 2019: 1 – 6.

[9] MOROZOV R. TRIFONOV P. On distance properties of convolutional polar codes. IEEE Trans. on Communications, 2019, 67(7): 1 – 8.

[10] ZHANG Q. LIU A. ZHANG Y. Practical design and decoding of parallel concatenated structure for systematic polar codes. IEEE Trans. on Communication, 2016, 64(2): 456 – 466.

[11] LIU Z. NIU K. CHAO D. Convergence analysis and performance optimization of parallel concatenated systematic polar code. The Journal of China Universities of Posts and Telecommunications, 2018, 25(2): 1 – 9. (in Chinese)

[12] CHEN K. NIU K. LIN J R. Improved successive cancellation decoding of polar codes. Electronics Letters, 2012, 61(8): 3100 – 3107.

[13] NIU K. CHEN K. Stack decoding of polar codes. Electronics Letters, 2012, 48(12): 695 – 697.

[14] KAMENEV M. KAMENeva Y. KURMAPOULOS K. et al. Permutation decoding of polar codes. https://arxiv.org/abs/1901.05459.

[15] JEONG M O. HONG S N. SC-Fano decoding of polar codes. IEEE Access. DOI: 10.1109/ACCESS.2019.2924011.

[16] ZHANG Y. LIU A. PAN X. et al. A modified belief propagation decoder. IEEE Communications Letters, 2014, 18(7): 1091 – 1094.

[17] SUN S. CHO S G. ZHANG Z. Post-processing methods for improving coding gain in belief propagation decoding of polar codes. Proc. of the IEEE Global Communications Conference, 2017: 1 – 6.

[18] SIMSEK C. TURK K. Hardware optimization for belief propagation polar code decoder with early stopping criteria using high-speed parallel-prefix adder. Proc. of the International Conference on Telecommunications and Signal Processing, 2017: 182 – 185.

[19] HAN K. WANG J. GROSS W J. et al. Stochastic bit-wise iterative decoding of polar codes. IEEE Trans. on Signal Processing, 2019, 67(5): 1138 – 1151.

[20] NIU K. CHEN K. LIN J. Low-complexity sphere decoding of polar codes based on optimum path metric. IEEE Communications Letters, 2014, 18(2): 332 – 335.

[21] PIAO J. DAI J. NIU K. CRC-aided sphere decoding for short polar codes. IEEE Communications Letters, 2019, 23(2): 210 – 213.

[22] HUSMANN C. NIKOLAOU P C. NIKITOPoulos K. Reduced latency ML polar decoding via multiple sphere-decoding tree searches. IEEE Trans. on Vehicular Technology, 2018, 67(2): 1835 – 1839.

[23] GOELA N. KORADA S B. GASTPAR M. On LP decoding of polar codes. Proc. of the IEEE Information Theory Workshop, 2010: 1 – 5.

[24] CAVATASSI A. TONELLIER T. GROSS W J. Fast decoding of multi-kernel polar codes. Proc. of the IEEE Wireless Communication and Networking Conference, 2019: 1 – 6.

[25] SEO J. LEE J. KIM K. Decoding of polar code by using deep feed-forward neural networks. Proc. of the International Conference on Computing, Networking and Communications, 2018: 238 – 242.

[26] TAL I. VARDY A. List decoding of polar codes. Proc. of the IEEE International Symposium on Information Theory Proceedings, 2011: 1 – 5.

[27] CHEN K. NIU K. LIN J. List successive cancellation decoding of polar codes. Electronics Letters, 2012, 48(9): 500 – 501.

[28] NIU K. CHEN K. CRC-aided decoding of polar codes. IEEE Communications Letters, 2012, 16(10): 1668 – 1671.

[29] ALAMDAR-YAZDI A. KSCISCHANG F R. A simplified successive-cancellation decoder for polar codes. IEEE Communications Letters, 2011, 15(12): 1378 – 1380.

[30] HASHEMI S A. CONDO C. GROSS W J. Fast simplified successive-cancellation list decoding of polar codes. Proc. of the IEEE International Symposium on Information Theory, 2016: 815 – 819.

[31] BALATSOKAS-STIMMING A. BASTANI PARIZI M. BURG A. LLR-based successive cancellation list decoding of polar codes. IEEE Trans. on Signal Processing, 2015, 63(19): 5165 – 5179.

[32] BERHAULT G. LEROUX C. JEGO C. Memory requirement reduction method for successive cancellation decoding of polar codes. Journal of Signal Processing Systems, 2017, 88(3): 425 – 438.

[33] LI B. SHEN H. TSE D. An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check. IEEE Communications Letters, 2012, 16(12): 2044 – 2047.

[34] HASHEMI S A. MONDELLI M. HASSANI S H. et al. Partitioned list decoding of polar codes: analysis and improvement of finite length performance. Proc. of the IEEE Global Communication Conference, 2017: 1 – 7.

[35] HASHEMI S A. MONDELLI M. HASSANI S H. et al. Decoder partitioning: towards practical list decoding of polar codes. IEEE Trans. on Communications, 2018, 66(9): 3749 – 3759.

[36] ZHOU H. LIANG X. LI L. et al. Segmented successive cancellation list polar decoding with tailored CRC. Journal of Signal Processing Systems, 2019, 8(1): 923 – 935.

[37] ELKELESH A. EBADA M. CAMMERER S. et al. Decoder-tailored polar code design using the genetic algorithm. IEEE Trans. on Communications, 2019, 67(7): 4521 – 4534.

Biographies

YANG Haifen was born in 1977. She received her Ph.D. degree in 2008 from the Department of Communication and Information Engineering, University of Electronic and Science Technology of China (UESTC), Chengdu. She is now an associate professor in UESTC. Her research interests include signal processing in wireless communications.

E-mail: yanghf@uestc.edu.cn
YAN Suxin was born in 1994. She received her master degree in 2016 from the Department of Communication and Information Engineering, University of Electronic and Science Technology of China (UESTC), Chengdu. Her research interests include channel coding in wireless communications. E-mail: sx_yan@126.com

ZHANG Hao was born in 1993. He received his bachelor degree in 2017 from University of Electronic Science and Technology of China (UESTC), Chengdu. He is now studying as a postgraduate at UESTC. His research interests include specific emitter identification. E-mail: zhanghao_sice@163.com

REN Yan was born in 1995. She received her bachelor degree in 2017 from the Department of Communication Engineering, Wuhan University of Technology (WHUT), Wuhan. She is currently working toward her master degree in the Department of Information and Communication Engineering at University of Electronic and Science Technology of China (UESTC). Her research interests include wireless communication and deep learning. E-mail: renyanstone@163.com

HU Xiangdong was born in 1990. He is now studying toward his M.S. degree in the Department of Information and Communication Engineering, University of Electronic and Science Technology of China (UESTC). His research interests include wireless communication and machine learning. E-mail: 379482415@qq.com

LIN Shuisheng was born in 1967. He received his Ph.D. degree in 1998 from University of Electronic and Science Technology of China (UESTC), Chengdu. He is now a professor in UESTC. His research interests include wireless communications and integrated circuits. E-mail: sslin@uestc.edu.cn