A Blind Recognition Algorithm of Scrambler after Convolutional Encoder

Zhongfang Wang 1,2,* , Liuqun Zhai 1,2 and Jingwen Fu 1,2

1 School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
2 Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China

Corresponding author: wangzhongfang@iie.ac.cn

Abstract. In a wireless communication system, the data is usually scrambled after channel encoding. However, existing research on blind recognition often ignores the pseudo-random scrambling of channel encoding data, which is not in line with the actual situation. To solve this problem, this paper proposes an algorithm to recognize the scrambler parameters with convolutional codes. First, we transform this problem into a cognition problem of cancelling scrambling sequence by using the properties of the polynomial of scrambler. Then, we put forward a fast judgment method after the cancellation of scrambling. The proposed method is based on the conditional entropy and can effectively identify the scrambler after the convolutional encoder. The method has a better performance of resisting error comparing to the traditional method and saves computing resources. Simulation results show that the parameters of the scrambler can still be determined effectively when the bit error rate is 6% in the case of sufficient data.

1. Introduction

Signal recognition is the main method for attackers to reversely analyse the unknown communication system, and it is also an important research direction in the field of modern signal processing and wireless link information security. In common communication systems, data scrambling should be done after channel coding. On the other hand, the blind recognition of scrambler needs to be done before the blind recognition of channel encoder.

In the current research, the analysis of the scrambler mainly uses the statistical characteristics of the source data sequence before scrambling, i.e., the unbalance of 0/1. Cluzeau proposed to construct variable Z by using the scrambled data, and then conduct traversal search in the sparse polynomial set based on the statistical test of the absolute value of the variable Z to generate the multiplier of the scrambling polynomial in literature [1]. When two different sparse multiples of the generating polynomial are found, the maximum convention of the two multiples is calculated as the generating polynomial of the scrambled data. This algorithm is also known as the Cluzeau algorithm. When the unbalance degree of the source sequence is unknown, the author deduced how to obtain the unbalance degree through the variable Z and the number of bits needed to reconstruct the scrambler N, and then Cluzeau algorithm is used to identify the generate polynomial in literature [2]. Literature [3] is an improvement of Cluzeau algorithm, which solves the problem of identification error when the second sparse multiplier searched is the multiplier of the first sparse multiplier. At the same time, it is proved that the improved Cluzeau algorithm is still applicable when the data passes through noisy channels.
However, after the convolutional encoding of the source data, the unbalance degree of the data may be greatly reduced [4]-[5]. In the case of the scrambling after convolutional code encoding, the reliability of the recognition results of the above algorithm will be greatly reduced. Literature [6] proposed to recognize the generate polynomials through the null space of the received data matrix to solve the above problems. However, the recognition reliability of this algorithm is insufficient in the case of noise.

In this paper, we first convert the scrambler generate polynomial recognition problem into the problem of scrambler elimination. Secondly, according to the scrambler elimination algorithm, the properties of the transformed sequence are analysed by using conditional entropy. At last, a fast scrambler polynomial discriminant method is proposed to solve the problem of scrambler recognition after convolutional codes with high bit error rate.

2. System overview
The blind recognition model of scrambler after channel encoding to be solved in this paper is shown in Figure 1.

![Figure 1. Communication system model](image)

In Figure 1., the scrambling process of the scrambler is to add the input sequence \(\{c_i\}\) and the LFSR sequence \(\{s_i\}\) with modulo two operation to obtain the scrambler output sequence \(\{y_i\}\). When the input sequence is \(\{c_i\}\), the relation between the output sequence and satisfies Equation (1), where the second symbol on the right side of Equation (1) \(\Theta(\cdot)\) represents the modulo two operation of \(n\) data.

\[
y_i = c_i \oplus s_i = c_i \oplus (\Theta(s_{i-n}))
\]

3. Blind recognition of scrambler after convolutional coding

3.1. Algorithm to eliminate the influence of the scrambling
In traditional method, the identification method of the generate polynomials is based on the sparse multinomial search of the generate polynomials, and the generated polynomials \(f(x)\) are determined according to the maximum convention of the two sparse multinomials obtained by the search. When \(f(x) = 1 + C_1x + \ldots + C_nx^n\), the generated LFSR sequence \(\{s_i\}\) is:

\[
y_i = c_i \oplus s_i = c_i \oplus (\Theta(s_{i-n}))
\]

It can be seen from Equation (2) that the scrambling output at each moment has a linear relationship with the scrambling output at the previous \(n\) moments. Equation (2) can be rewritten as:

\[
s_{i+n} \oplus C_n s_i \oplus C_{n-1} s_{i+1} \oplus \cdots \oplus C_1 s_{i+n-1} = 0
\]

Therefore, if the data sequence \(\{s_i\}\) used for scrambling satisfies Equation (4):

\[
s_i \oplus \sum_{j=1}^{d} s_{i-j} = 0, 0 < i_1 < i_2 < \cdots < i_{d-1}
\]
It can be proved that if a polynomial \( Q(x) = 1 + \sum_{j=1}^{d-1} x^j \) is a multiplier of the scrambler polynomial \( f(x) \), the above equation still equal to 0. Therefore, as long as the equation satisfying equation (4), the multiplier \( Q(x) \) of \( f(x) \) can be found and \( f(x) \) can be determined. In this section, the recognition of the scrambler makes use of this property to eliminate the influence of scrambling on information flow.

To eliminate the influence of scrambler, this paper proposes an algorithm to eliminate scrambler by using the idea of convolution. For the observation sequence \( y_i = \{y_1, y_2, \ldots, y_n\} \), if we convolve the sequence \( y_i \) with the sequence \( h = \{h_1, h_2, \ldots, h_N\} \) of finite terms, then the new sequence \( v_i \) is obtained. This process can be written as \( v_i = y_i * h \). The generation of the finite sequence \( h \) can be expressed in polynomial form \( h(x) = 1 + \sum_{j=1}^{N-1} x^j \), and the sequence \( i = \{i_1, i_2, \ldots, i_N\} \) is the position of a non-zero term of \( h(x) \). We can eliminate the scrambling code by creating new sequence \( v_i \).

Since the convolution of observation sequence \( y_i \) is equivalent to the sequence \( y_i \) being shifted and then superimposed according to \( h \), sequence after the \( i \)th shift can be denoted as \( y_{i}' = \{y_1, y_2, \ldots, y_{i+1}, \ldots, y_n\} \). \( y_{i}' \) represents the new sequence formed after modulo two of the data of the \( i \)th and \( j \)th shift, namely:

\[
y_{i}' \oplus y_{j}' = \{y_{i}', y_{i+1}', \ldots, y_{i+j}', \ldots\}
\]

For the generation polynomial of the scrambler \( f(x) \), we can get the sequence \( v_i \) by taking modulus two of every \( y_{i}' \), where \( y_{i}', y_{i+1}', \ldots, y_{i+j}' \) is selected according \( h(x) \). \( y_{i}' \) represents a left shift of \( y_i = \{y_1, y_2, \ldots, y_n\} \) with length of \( i_j (1 \leq j \leq N) \). Then the sequence \( v_i \) can be expressed as:

\[
v_i = y_{i_1}' \oplus y_{i_2}' \oplus \cdots \oplus y_{i_N}' \quad 0 < i_1 < i_2 < \cdots < i_N
\]

Sequence \( y_i \) is the modulo-two sum of the code word \( c_i \) and the scrambling data \( s_i \), i.e. \( y_i = c_i \oplus s_i \). Since \( y_i = c_i \oplus s_i \), sequence \( v_i \) can be rewritten as:

\[
v_i = y_{i_1}' \oplus y_{i_2}' \oplus \cdots \oplus y_{i_N}'
= \{c_{i_1} \oplus s_{i_1}, c_{i_2} \oplus s_{i_2}, \ldots, c_{i_N} \oplus s_{i_N}\}
\]

Where \( c_{i_1} \) is the modulo-two sum of N code word data, and \( s_{i_0} \) is the modulo-two sum of N scrambling data, namely:

\[
c_{i_1} = (c_{i_1} \oplus c_{i_2} \oplus \cdots \oplus c_{i_N})
\]

When \( s_{i_0} = 0 \), which means \( s_{i_1} \oplus s_{i_2} \oplus \cdots \oplus s_{i_N} = 0 \), the sequence \( v_i \) is \( [c_{i_1}, c_{i_2}, \ldots, c_{i_N}] \). It is composed of pure convolutional encoding data. According to the properties of the scrambling polynomial in Equation (4), \( h(x) = 1 + \sum_{j=1}^{N-1} x^j \) is a multiplier of the generation polynomial of \( f(x) \).

When \( s_{i_0} \neq 0 \), the sequence after the superposition \( v_i \) is still a scrambled sequence, which means \( h(x) = 1 + \sum_{j=1}^{N-1} x^j \) is not the multiplier of the generated polynomial of \( f(x) \).

Therefore, it is possible to judge whether a polynomial \( h(x) \) is a multiplier of the scrambling polynomial \( f(x) \), according to whether the sequence \( v_i \) is a superposition of pure code word data. In Section 3.2, a fast judgment method for the superposition sequence of pure code words is presented.
3.2. Algorithm to eliminate the influence of the scrambling

After the encoding sequence goes through the scrambler eliminating algorithm, the constrained of codewords grows, but the code rate remains unchanged. In this section, a fast judgment algorithm based on conditional entropy is proposed to determine whether the codeword after the eliminating algorithm is pure codeword superposition.

Lemma 1: If we treat \((n, k, K)\) convolutional encoding, scrambling and scrambler eliminating algorithm as a whole process of encoding, the eliminating method makes the constraint length of the codeword grows to \(K'\). Let \(r_t\) be the output codeword after the whole encoding process at moment \(t\).

Conditional entropy of the codeword is \(H(r_t | r_{t-1}, r_{t-2}, \ldots, r_{t-K+1}) = k\), and it doesn’t depend on the constraint length \(K'\). The process of convolutional code scrambling and descrambling is shown in Figure 2.

![Figure 2. Communication system model](image)

In Figure 2, we let \(m_t\) pass through convolutional encoder, the output of the model goes into the scrambler and we get \(y_t\), finally through the eliminator we get \(r_t\).

Proof: For superposition scramble eliminating encoding, let \(m_t\) represents the codeword input to the convolutional encoder at time \(t\). And \(m_t\) is a codeword of \(k\) bits. There is a total of \(2^k\) possibilities for the values of \(m_t\), namely, \(m_t \in \{0,1,2,\ldots,2^k\}\). \(m_t\) is a random input with equal probability, that is:

\[
p(m_t) = \frac{1}{2^k}, m_t \in \{0,1,2,\ldots,2^k\}
\]

Let \(r_t\) be the output codeword of the encoder at moment \(t\) with \(n\) bits, \(r_t \in \{0,1,2,\ldots,2^n\}\). According to the encoding processing, the code word \(r_t\) is not only related to the input message of the current moment \(m_t\), but also related to the input message of the previous \((K' - 1)\) moment, indicating that the input message of \(K'\) moment jointly determines the value \(r_t\) of the current moment, namely:

\[
r_t \leftarrow (m_t, m_{t-1}, m_{t-2}, \ldots, m_{t-K'+1})
\]

Where \(f\) determined by the structure of the encoder, it represents the correspondence between the codewords \((m_t, m_{t-1}, m_{t-2}, \ldots, m_{t-K'+1})\) and the codeword \(r_t\) at each moment.

The hidden Markov model is a double stochastic process, as shown in Figure 3, \(X = \{X_1, X_2, \ldots, X_t\}\) represents a Markov transition process, the state of which is called implicit state; \(O = \{O_1, O_2, \ldots, O_t\}\) represents the observation process, and the state of the process is the observation state. Among them, the transition between implicit states is a random event, and the transition between implicit states and observed states is a random process.
Figure 3. Schematic diagram of hidden Markov model

Comparing the coding process with the Hidden Markov Model, the sequence of encoded code words \( \{ r_1, r_2, \ldots, r_T \} \) can be regarded as the observation process sequence and the code words \( r_t \) as the observation state, then the state \( q_t \) of the coded \( K \)-level registers can be regarded as the hidden state, \( q_t = (m_1, m_2, \ldots, m_{K-1}) \). According to Equation (10), the observed state \( r_t \) can be uniquely determined by the hidden state \( q_t \), that is, the transition probability between the hidden state and the observed state is:

\[
\Pr(O_t = r | X_t = q_t) = 1
\]  

(11)

Markov process \( Q = \{ q_1, q_2, \ldots, q_T \} \) is a first-order Markov model, that is:

\[
p(X_t = q_t | X_{t-1} = q_{t-1}, X_{t-2} = q_{t-2}, \ldots, X_1 = q_1) = p(X_t = q_t | X_{t-1} = q_{t-1}) = \frac{1}{2^t}
\]  

(12)

Therefore, the observation process sequence \( \{ r_1, r_2, \ldots, r_T \} \) meets the following requirements:

\[
p(O_t = r_t | O_{t-1} = r_{t-1}, O_{t-2} = r_{t-2}, \ldots, O_{t-K+1} = r_{t-K+1}) = \frac{1}{2^t}
\]  

(13)

Equation (13) shows that, when the observed state \( O_t = r_t \) at the moment \( t \) is known, the probability of the observed state \( O_t = r_t \) at the previous \((K-1)\) moment of the current moment is \( \frac{1}{2^t} \). Therefore, the conditional entropy can be calculated as \( H(r_t | r_{t-1}, r_{t-2}, \ldots, r_{t-K+1}) \):

\[
H(r_t | r_{t-1}, r_{t-2}, \ldots, r_{t-K+1}) = -\sum_{i=1}^{t} p(r_t | r_{t-1}, r_{t-2}, \ldots, r_{t-K+1}) \log_2 (p(r_t | r_{t-1}, r_{t-2}, \ldots, r_{t-K+1})) = k
\]  

(14)

Lemma 1 was proved.

4. Simulation and analysis

4.1. Feasibility of conditional entropy algorithm
In the last chapter, we introduce a fast method to judge the encoded sequence, which is based on conditional entropy. In order to verify the feasibility of this method, we test the results of encoded sequence with same scrambler with no error and with error respectively. The simulation is done in the scene with error code and without error code.

In the case of bit errors, we test the performance of coding sequence judgment method based on conditional entropy. In this simulation, the data length is 1 million, and the bit error rate is set at 0.5%.
8%. The simulation results are shown in Figure 4. It can be seen that with the increase of bit error rate, the conditional entropy of the sequence is more and more close to the entropy of the random sequence. Most of the codes in the figure do not approach the conditional entropy of random sequences until the bit error rate is 8%. In other words, when the bit error rate is less than 8%, the judgment method using conditional entropy can effectively distinguish the coding sequence from the random sequence.

**Figure 4.** Convolutional code sequence judgment based on entropy rate in case of error code

### 4.2. Performance of the blind recognition algorithm

The following simulation verifies the effectiveness of the blind recognition algorithm. In the simulation experiment, two different scrambling polynomials were used for performance comparison with reference [6], as shown in Table 1. The convolutional codes adopted are convolutional encoder (3,1,6), same as reference [6], and their generation polynomials are shown in Table 1.

| Convolution code | Encoding generates polynomial matrices | Perturbation polynomial |
|------------------|----------------------------------------|------------------------|
| (3,1,6)          | $G = \begin{bmatrix} 1 + x^3 + x^4 + x^5 \\ 1 + x^4 + x^5 \\ 1 + x^2 + x^3 + x^4 \end{bmatrix}$ | $f_1(x) = x^3 + x^4 + 1$ |
|                  |                                        | $f_2(x) = x^{11} + x^2 + 1$ |

The accuracy of the proposed blind recognition algorithm is compared under different bit error rate. In order to compare with the recognition results in literature [6], the convolutional code selected is the convolutional code (3,1,6) in Table 1, the encoding generation polynomial matrix is $G$, and the scrambler polynomial is $f_1(x) = x^3 + x^4 + 1$ and $f_2(x) = x^{11} + x^2 + 1$. For each bit error rate, Monte Carlo simulation was conducted for 1000 times, and the recognition results obtained were shown in Figure 5. Literature [6] has poor performance in scrambler recognition at a high bit error rate. As can be seen from Figure 5, at the same bit error rate, the correctness of the proposed algorithm is higher than that of the literature [6]. Literature [6] realized the recognition of scrambler polynomials based on the null space of the received data matrix. This method has a low tolerance to noise, so the accuracy of scrambler recognition is low. The algorithm in this paper turns the problem of scrambler recognition into the conditional entropy problem of judging the transformed sequence, so the correct rate of scrambling recognition in the case of noise is higher than that in reference [6].
Figure 5. Comparison of the accuracy of scrambling recognition between the algorithm in this paper and the algorithm in literature [6]

5. Conclusion
In this paper, we propose a blind scrambler recognition method after convolutional encoding. Firstly, we transform the problem of scrambler recognition into the problem of eliminated scrambling sequence recognition. Then a fast judgment algorithm based on conditional entropy is proposed and proved. Finally, the proposed algorithm is simulated and compared with previous algorithm. Simulation results show that the proposed algorithm can distinguish the random sequence from the coding sequence when the bit error rate is less than 6%, and can correctly identify the parameters of the scrambler.

References
[1] Cluzeau M 2007 Reconstruction of a linear scrambler Ieee T Comput 56 1283-91
[2] Yu M, Li-Min Z and Hao-Tong W 2015 Reconstructing Synchronous Scrambler With Robust Detection Capability in the Presence of Noise IEEE Transactions on Information Forensics and Security 10 397-408
[3] He W J 2015 Reconstructing the feedback polynomial of a linear scrambler with the method of hypothesis testing Jet Communications 9 1044-7
[4] Liu X B, Koh S N, Chui C C and Wu X W 2013 A Study on Reconstruction of Linear Scrambler Using Dual Words of Channel Encoder Ieee Transactions on Information Forensics and Security 8 542-52
[5] Liu X B, Koh S N, Wu X W and Chui C C 2012 Reconstructing a Linear Scrambler With Improved Detection Capability and in the Presence of Noise Ieee Transactions on Information Forensics and Security 7 208-18
[6] Han S N and Zhang M 2018 A Method for Blind Identification of a Scrambler Based on Matrix Analysis Ieee Commun Lett 22 2198-201