Blind Parameter Estimation of a Rate k/n Convolutional Code in Noiseless Case

A. Gholizadeh, H. Khaleghi Bizaki
Electrical and Electronic Engineering Department,
Malek-Ashtar University of Technology, Tehran, Iran
Email: hbizaki@gmail.com

***Abstract*- This paper concerns to blind identification of a convolutional code with desired rate in a noiseless transmission scenario. To the best of our knowledge, blind estimation of convolutional code based on only the received bitstream doesn't lead to a unique solution. Hence, without loss of generality, we will assume that the transmitter employs a non-catastrophic encoder. Moreover, we consider a complete synchronous scenario in which one can extract separate codewords from received sequence. This assumption is valid in many practical communication systems because, the frame preambles allow us to identify the beginning of each codewords. In this paper, we examine the blind identification problem for rate$1/n$ and rate k/n convolutional codes, respectively. For rate $1/n$, we propose an iterative method that uses three steps in each iteration to test the validity of a possible value of n. We show that this method can identify the parameters of a rate $1/n$ convolutional code from only two different noiseless received codewords. Afterwards, we generalize this method for a rate k/n convolutional code in which each iteration is composed of seven successive steps. We show that this method requires at least $k+1$ different codewords to identify all parameters of a rate k/n code.**

***Index Terms*-** Blind estimation, convolutional code, cognitive radio, non-catastrophic encoder, minimal-basic encoder.

I. INTRODUCTION

In recent years, many new communication standards have been developed to improve the bit error rate and quality of service in communication systems. To support this development, the transreceiver structure should be continuously updated to remain compatible with all standards used. Moreover, the standard multiplicity imposes a new compatibility issue on communication systems with different standards.The multi-mode receiver can be a practical and effective solution to this problem. Such receiver indeed is an intelligent system which first extracts transmitter parameters from received signals and then adjusts its parameters according to them. The application of such systems, so called blind receiver, in the cognitive radio is obvious. In general, A typical blind receiver consists of: a blind channel decoder that first decides on the code existence, then recognizes the type of channel

Manuscript received 15-Dec.-2016 and revised 14-March-2018, *P- ISSN: 2322-4088*
Accepted on 23-April-2018 *E- ISSN: 2322-3936*

code parameters, and finally decodes the received stream by an appropriately adapted decoder.

Convolutional code is one of the widely used channel codes in modern communication systems. Hence, the blind identification of its parameters is one of the most important issues in blind channel decoder design. In [1-2], the well-known Euclidean algorithm is used to recover the generator matrix of a rate $1/2$convolutional encoder in the noiseless scenario. Despite of limited scope of [1-2] which can be used only for rate ½ convolutional codes, their algorithm has a low complexity and also is independent of the received codewords length. Authors in [3] exploited the algebraic properties of an optimal convolutional code to identify the second convolutional encoder of a turbo code. Also, in [4], this method is generalized to blind recovery of a rate k/n optimal convolutional code in noiseless case. Moreover, in [5], the same method is used in an iteration manner to estimate a rate $(n-1)/n$ convolutional code from noisy observations. Authors in [6] extended this method to rate k/n convolutional codes. This method can estimate the code parameters in a very low-noise scenario. However, this algorithm suffers from relatively high complexity. It also requires a long bitstream even in a noiseless environment.

In this paper, we propose a new low-complexity and fast iterative method to identify the convolutional code parameters from a complete synchronous bitstream in a noiseless transmission scenario. The *complete synchronous* means that both the beginning and the end of each received codeword are known. Of course, this level of synchronization is quite practical, since the most modern communication systems often transmit the codewords in separate frames. Moreover, in most of the practical systems the frame header is not encoded. As a result, a cooperative deframer, or even a blind one, can extract the codewords from the frame header.

On the other hand, any linear code can be generated by many equivalent encoders. Therefore, it is not possible to identify the main generator matrix only based on received codewords. Hence, we need more information about the applied encoder to limit the number of possible generator matrices. Since the identification of code parameters is desired, without loss of generality, we assume that the transmitter employs a special type of encoder, i.e., non-catastrophic encoders. Based on this assumption, we examine the blind recovery of convolutional codes for rates $1/n$ and k/n, respectively. In each scenario, we propose a blind method based on the non-catastrophic convolutional encoder property [7-9]. Our method for blind identification of a rate $1/n$ code is an iterative algorithm that uses three simple steps to verify the validity of a possible value for n. Then, we generalize this method to blind identification of rate k/n codes. This generalized method is composed of seven successive steps, in which the first three steps give an initial estimation of the parameters n and k, and then, the last four steps ensure the validity of these values. Moreover, we will determine the minimum number of codewords required for each method to work.

The remainder of this paper is organized as follows. Section 2, briefly introduces the convolutional code and also the properties of some special encoders. In section 3, the proposed methods for rate $1/n$

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

and k/n convolutional codes are explained in two separate subsections. Moreover, some examples are provided to confirm the proposed methods. Finally, section 4 concludes the paper with some prospects.

II. THE MATHEMATICAL DESCRIPTION OF CONVOLUTIONAL CODES

Consider a rate k/n binary convolutional code $\mathcal{C}(n,k,\nu)$, where ν denotes the overall constraint length. This code can be described by a $k \times n$ full-rank rational generator matrix $G(D)$ over the field of binary rational functions $\mathbb{F}_2(D)$, denoted by:

$$G(D) = \begin{bmatrix} g_{11}(D) & \dots & g_{1n}(D) \\ \vdots & \ddots & \vdots \\ g_{k1}(D) & \dots & g_{kn}(D) \end{bmatrix}, \quad (1)$$

where the entry $g_{ij}(D) \in \mathbb{F}_2(D)$ specifies the rational transfer function of the i'th encoder input to the j'th encoder output in the delay operator D. In this paper, we denote the matrices with uppercase letters and a vectors with boldface lowercase letters.

Let $\boldsymbol{u}(D) = [u_1(D), \dots, u_k(D)]$ be the k-tuple input vector, where $u_i(D)$ denotes the ith input sequence of the encoder. Then, the corresponding n-tuple code vector, denoted by $\boldsymbol{c}(D) = [c_1(D), \dots, c_n(D)]$, can be obtained from the following encoding process:

$$\boldsymbol{c}(D) = \boldsymbol{u}(D).G(D) \quad (2)$$

After encoding, the n output sequences $c_j(D)$, $j = 1, \dots, n$, are multiplexed to construct the codeword $c(D)$ as follows:

$$c(D) = c_1(D^n) + Dc_2(D^n) + \cdots + D^{n-1}c_n(D^n) \quad (3)$$

In general, a typical convolutional code $\mathcal{C}(n,k,\nu)$ can be generated with several generator matrices, so-called equivalent encoders. Two equivalent generator matrices generate the same code in some different mapping orders. The set of all equivalent generator matrices of code $\mathcal{C}(n,k,\nu)$is denoted by$\mathbb{G}_{\mathcal{C}}$.

Theorem 1: *The full-rank $k \times n$ generator matrices $G(D)$ and $G'(D)$ are equivalent if and only if there exists a $k \times k$ nonsingular matrix T(D) such that [5]:*

$$G'(D) = T(D).G(D) \quad (4)$$

In practice, some encoders are avoided and some are preferred because of practical considerations, such as the complexity of encoding and decoding. For instance, system designers avoid an encoder with a catastrophic property. A catastrophic encoder maps at least an infinite length message to a finite length codeword. In this case, if an error pattern alters this finite length codeword to a zero codeword, the decoder (with the minimum distance criterion) decodes the received sequence as a zero

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

message. Thus, this error pattern produces an infinite decoding error or, equivalently, an infinite bit error rate. The following theorem is a simple test to verify whether anencoder is catastrophic or not [8, 9].

Theorem 2: *The $k \times n$ generator matrix $G(D)$ is non-catastrophic if and only if:*

$$GCD\left(\Delta_G^i(D); i = 1, \dots, \binom{n}{k}\right) = D^l \tag{5}$$

where $\Delta_G^i(D), i = 1, \dots, \binom{n}{k}$ are the $k \times k$ submatrix determinants (minors) of $G(D)$, GCD(.) gets the greatest common divisor (GCD), and l denotes the encoder delay.

The practical encoders are usually zero-delay,i.e.$l = 0$. Hence, we can simplify the condition (5) as follows:

$$GCD\left(\Delta_G^i(D); i = 1, \dots, \binom{n}{k}\right) = 1 \tag{6}$$

This relation implies that the $k \times k$ minors of a zero-delay non-catastrophic encoder are relativity prime.

A $k \times n$ generator matrix $G(D) \in \mathbb{G}_{\mathcal{C}}$ can be realized by a different sequential circuit which consists of k inputs, n outputs and at least ν memories. The below theorem gives a way to calculate the parameter ν from any generator matrix in $\mathbb{G}_{\mathcal{C}}$.

Theorem 3: *The overall constraint length ν is an invariant of convolutional code $\mathcal{C}(n, k, \nu)$ (i.e., it is not dependent on the encoder type) and can be calculated from any generator matrix $G(D) \in \mathbb{G}_{\mathcal{C}}$ as follows [9]:*

$$\nu = \max\left(\deg\left(\Delta_G^i(D)\right)\ ;\ i = 1, \dots, \binom{n}{k}\right) \tag{7}$$

The number of realization memories determines the encoder and decoder complexity, and the minimal memory is equal to the overall constraint length[6-8]. Hence, it is a practical interest to employ a so-called minimal encoder that can be minimally realized with ν memories.

Definition 1: *The polynomial generator matrix $G_{mb}(D) \in \mathbb{G}_{\mathcal{C}}$ is minimal-basic, if it can be minimally realized in a feedforward controller canonical form[8].*

In the controller canonical realization, every row of a $k \times n$ generator matrix $G(D)$ minimally realizes as a separate $1/n$ encoder and then these k outputs simply sum to generate the overall encoder output. Note that, if $G(D)$ is a polynomial, the i'th row can be minimally realized with μ_i memory elements, where μ_i is the constraint length of the$i$'th input and is defined as follows:

$$\mu_i = \max_{j=1,\dots,n} \deg\left(g_{ij}(D)\right), \quad i = 1, \dots, k \tag{8}$$

Subsequently, in a minimal-basic encoder we have:

$$\nu = \sum_{i=1}^{k} \mu_i \tag{9}$$

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

Below, we give a definition that is used in the next section.

Definition 2: *Demultiplexing at depth r breaks the polynomial $x(D) = x_0 + \cdots + x_d D^d$down into r separate polynomials of $x_i^{(r)}(D), i = 1, \dots, r$, as follows:*

$$x_i^{(r)}(D) = x_{i-1} + x_{i+r-1}D + \cdots + x_{i+\lfloor \frac{d}{r} \rfloor - 1} D^{\lfloor \frac{d}{r} \rfloor} \tag{10}$$

where $\lfloor . \rfloor$ denotes the floor operation. Putting into vector notation, we denotes these polynomials by $x^{(r)}(D) = [x_1^{(r)}(D), \dots, x_r^{(r)}(D)]$. Note that the demultiplexing at depth n indeed is the inverse of the multiplexing in equation (3), i.e., $c^{(r)}(D) = \boldsymbol{c}(D)$.

III. BLIND IDENTIFICATION OF CONVOLUTIONAL CODES

In this section, we propose a method for blind identification of convolutional code parameters based on a complete synchronous errorless received stream. In general case, the parameters to be estimated for a typical convolutional code $\mathcal{C}(n, k, \nu)$ are constructed of: n, k, ν and a generator matrix $G(D) \in \mathbb{G}_{\mathcal{C}}$.

As argued before, it is not possible to estimate the desired encoder of the transmitter only from the received codewords, because there are many equivalent encoders that generate the same code. Due to this equivalency, without loss of generality, we can assume that the transmitter employs a non-catastrophic convolutional encoder. Thus, we use the non-catastrophic condition (6) to blindly extract the convolutional code parameters. In the following subsections, we represent our methods for blind identification of rate $1/n$ and k/n convolutional codes, respectively.

a) Blind identification of a rate $1/n$ convolutional code

Assume that the transmitter encodes a message sequence $u(D)$ into a codeword $c(D)$ by the following $1 \times n$ non-catastrophic polynomial generator vector:

$$\boldsymbol{g}(D) = [g_1(D), \dots, g_n(D)] \tag{11}$$

For this encoder, equation (6) can be simplified as follows:

$$GCD(g_1(D), \dots, g_n(D)) = 1 \tag{12}$$

We exploit this relation to recognize the parameter n. Let we have two different complete synchronous errorless codewords $c_1(D)$ and $c_2(D)$ in the receiver. Also, let $u_i(D)$ and $\boldsymbol{c}_i(D)$ be the message and code vector of corresponding to $c_i(D)$, respectively. Using this notation, the encoding process can be written as:

$$\boldsymbol{c}_i(D) = u_i(D).\boldsymbol{g}(D) \quad ; \quad i = 1, 2 \tag{13}$$

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

Let $\hat{n}$ denote the estimation of n. To identify the parameter n, we propose an iterative method for $\hat{n} = 2,3, \dots$. The relation (6) is valide only when $\hat{n} =$ n is admitted. Thus, the hypothesis of $\hat{n} =$ n is examined in the $(\hat{n} - 1)$'th iteration by the following three steps:

Step 1: For $i = 1,2$, demultiplex $c_i(D)$ at depth $\hat{n}$ as $\boldsymbol{c}_i^{\hat{n}}(D) = [c_{i1}^{\hat{n}}(D), \dots, c_{i\hat{n}}^{\hat{n}}(D)]$.

Step 2: For $i = 1,2$, calculate GCD of the entries of $\boldsymbol{c}_i^{\hat{n}}(D)$, denoted by $u_i^{\hat{n}}(D)$.

Step 3: For $i = 1,2$, divide the vector $\boldsymbol{c}_i^{\hat{n}}(D)$ by $u_i^{\hat{n}}(D)$ to obtain the vector of $\boldsymbol{g}_i^{\hat{n}}(D)$.

Note that if we have $\hat{n} = n$, then for $i = 1,2$, demultiplexing of the codeword $c_i(D)$ at depth $\hat{n}$ (or n) results the code vector $\boldsymbol{c}_i(D)$, i.e. $c_i^{(n)}(D) = \boldsymbol{c}_i^n(D) = \boldsymbol{c}_i(D)$. Thus, according to (11) and (12), one can simply verify that, for $i = 1,2$, $u_i^n(D)$ is indeed equal to massage sequence $u_i(D)$. Therefore, from (13), we have $\boldsymbol{g}(D) = \boldsymbol{g}_1^{\hat{n}}(D) = \boldsymbol{g}_2^{\hat{n}}(D)$. In other word, if $\hat{n} = n$, then the vectors of $\boldsymbol{g}_1^{\hat{n}}(D)$ and $\boldsymbol{g}_2^{\hat{n}}(D)$ are equal. However, this equality is not true for any $\hat{n} < n$, since in this case two demultiplexed vectors $\boldsymbol{c}_i^{\hat{n}}(D)$, $i = 1,2$, do not necessarily belong to the same code, and so the vectors $\boldsymbol{g}_1^{\hat{n}}(D)$ and $\boldsymbol{g}_2^{\hat{n}}(D)$ would be unequal. This property implies that the iteration should be continued until the vectors in step (3) become equal. As a result, if this equality is satisfied for a $\hat{n}$, then we conclude that the hypothesis $\hat{n} = n$ is true. This means that both vectors $\boldsymbol{g}_1^{\hat{n}}(D)$ and $\boldsymbol{g}_2^{\hat{n}}(D)$ are equal to non-catastrophic encoder $\boldsymbol{g}(D)$.

So far, we have recognized the parameter n and also a non-catastrophic basis $\boldsymbol{g}(D)$ for the convolutional code $\mathcal{C}(n, 1, \nu)$. Since a $1 \times n$ non-catastrophic encoder is minimal-basic, from(9), one can calculate the overall constraint length ν as follows:

$$\nu = \max(g_1(D), \dots, g_n(D)) \tag{14}$$

The detail of the proposed method, is reperesented as a pseudo-code in Algorithm1. In step 2 of this algorithm, we require $\hat{n}$ iterations to calculate the polynomial $u_1^{\hat{n}}(D)$ (or $u_2^{\hat{n}}(D)$), where GCD of only two polynomials must be found in every iteration. This can be accomplished by a well-known Euclidean algorithm.The complexity of the proposed algorithm is highly depended on the received codeword lengths.

Let d_{max} denotes the maximum degree of two received codewords. Then, for iteration $\hat{n}$, every GCD in step 2 can be calculated using the Euclidean algorithm with the complexity $\tilde{\mathcal{O}}(\frac{d_{max}^2}{\hat{n}^2})$. However, benefit a newer GCD algorithm can decrease the computational complexity.

We consider a maximum complexity $\tilde{\mathcal{O}}(\frac{d_{max}^2}{\hat{n}^2})$ for every polynomial division in step 3. The $\hat{n}$'th iteration in Algorithm 1 is composed of $\hat{n} - 1$ polynomial GCD calculations in step 2, and $2\hat{n}$ polynomial divisions in step 3.

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

Alg. 1. Blind identification of code $\mathcal{C}(n,1,\nu)$

Input: Two different codewords $c_1(D), c_2(D)$

Output: n, ν and a minimal-basic encoder $\boldsymbol{g}(D)$

$f \Leftarrow 0$

$\hat{n} \Leftarrow 2$

while $f \neq 1$ **do**

  for $i = 1$ to 2 **do**

    $\boldsymbol{c}_i^{\hat{n}}(D) \Leftarrow c_i^{(\hat{n})}(D)$; Step 1

    $u_i^{\hat{n}}(D) \Leftarrow c_{11}^{\hat{n}}$

    for $j = 2$ to $\hat{n}$ **do**

      $u_i^{\hat{n}}(D) \Leftarrow GCD\left(u_i^{\hat{n}}(D), c_{1j}^{\hat{n}}\right)$; Step 2

    end

    $\boldsymbol{g}_i^{\hat{n}}(D) \Leftarrow \frac{\boldsymbol{c}_i^{\hat{n}}(D)}{u_i^{\hat{n}}(D)}$; Step 3

  end

  if $\boldsymbol{g}_1^{\hat{n}}(D) \neq \boldsymbol{g}_2^{\hat{n}}(D)$ **then**

    $\hat{n} \Leftarrow \hat{n} + 1$

  else

    $f \Leftarrow 1$

  end

end

$\boldsymbol{g}(D) \Leftarrow \boldsymbol{g}_1^{\hat{n}}(D)[or\ \ \boldsymbol{g}_1^{\hat{n}}(D)]$

$n \Leftarrow \hat{n}$

$\nu \Leftarrow \max(g_1(D), \dots, g_n(D))$

As a result, we can approximate the complexity order with $\tilde{\mathcal{O}}(d_{max}^2)$. As a final remark to this method, it is worth to note that only two different nonzero complete synchronous errorless codewords are sufficient and also necessary for this method. Also, for this algorithm is no matter to codeword lengths.

Example 1: Assume that the transmitter employs the following non-catastrophic encoder to generate the convolutional code $\mathcal{C}(3,1,3)$:

$$\boldsymbol{g}(D) = [1 + D \quad 1 + D + D^2 \quad 1 + D^3]$$

The transmitter encodes the following message sequences:

$$u_1(D) = 1 + D \quad , \quad u_2(D) = D$$

where the corresponding received errorless codewords are as follows:

$$c_1(D) = 1 + D + D^2 + D^5 + D^6 + D^{10} + D^{11} + D^{14}$$

$$c_2(D) = D^3 + D^4 + D^5 + D^6 + D^7 + D^{10} + D^{14}$$

As first iteration of the proposed algorithm, we examine the hypothesis of $\hat{n} = 2$ and demultiplex the received codewords in depth 2 as following vectors (step 1):

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

$$\boldsymbol{c}_1^2(D) = [1 + D + D^3 + D^5 + D^7 \quad 1 + D^2 + D^5]$$
$$\boldsymbol{c}_2^2(D) = [D^2 + D^3 + D^5 + D^7 \quad D + D^2 + D^3]$$

Now, for $i = 1,2$, we calculate theGCD of two entries from $\boldsymbol{c}_i^2(D)$ by Euclidean algorithm, where the results are as follows (step 2):

$$u_1^2(D) = 1 \quad , \quad u_2^2(D) = D$$

Then, calculating $\boldsymbol{c}_1^2(D)/u_1^2(D)$ and $\boldsymbol{c}_2^2(D)/u_2^2(D)$, we find (step 3):

$$\boldsymbol{g}_1^2(D) = [1 + D + D^3 + D^5 + D^7 \quad 1 + D^2 + D^5]$$
$$\boldsymbol{g}_2^2(D) = [D + D^2 + D^4 + D^6 \quad 1 + D + D^2]$$

Since these vectors are unequal, i.e. $\boldsymbol{g}_1^2(D) \neq \boldsymbol{g}_2^2(D)$, we conclude that the hypothesis $\hat{n} = 2$ is false. In the next iteration, we examine the hypothesis $\hat{n} = 3$ and demultiplex the received codewords in depth 3 as following vectors (step 1):

$$\boldsymbol{c}_1^3(D) = [1 + D^2 \quad 1 + D^3 \quad 1 + D + D^3 + D^4]$$
$$\boldsymbol{c}_2^3(D) = [D + D^2 \quad D + D^2 + D^3 \quad D + D^4]$$

Then, for $i = 1,2$, we calculate GCD of entries in $\boldsymbol{c}_i^2(D)$ by twice use of Euclidean algorithm, where the results are as follows (step 2):

$$u_1^3(D) = 1 + D \quad , \quad u_2^3(D) = D$$

Note that the resulted polynomials are equal to message sequences. As final step, calculating$\boldsymbol{c}_1^3(D)/u_1^3(D)$ and $\boldsymbol{c}_2^3(D)/u_2^3(D)$, we obtain (step 3):

$$\boldsymbol{g}_1^3(D) = \boldsymbol{g}_2^3(D) = [1 \quad 1 + D + D^2 \quad 1 + D^3] \tag{15}$$

Since these vectors are equal, we conclude that the hypothesis $\hat{n} = 3$ is true. Finally, we introduce the either vectors $\boldsymbol{g}_1^3(D)$ or $\boldsymbol{g}_2^3(D)$ as a non-catastrophic encoder for code $\mathcal{C}$ and immediately, from (14), we find $\nu = 3$.

b) Blind identification of a rate k/n convolutional code

In this scenario, we assume that the transmitter employs a $k \times n$ non-catastrophic polynomial generator matrix to generate the convolutional code $\mathcal{C}(n, k, \nu)$, where $G(D)$ is defined as (1). Let we have L different complete synchronous errorless codewords $c_i(D), i = 1, \dots, L$, in the receiver. Also, let $\boldsymbol{u}_i(D) = [u_{i1}(D), \dots, u_{ik}(D)]$ and $\boldsymbol{c}_i(D) = [c_{i1}(D), \dots, c_{in}(D)]$ be the massage vector and the code vector of corresponding to the i'th codeword $c_i(D)$, respectively. We can summarize the encoding process using of the message matrix $U(D) = [\boldsymbol{u}_1(D)^T, \dots, \boldsymbol{u}_L(D)^T]^T$ and the code matrix $C(D) = [\boldsymbol{c}_1(D)^T, \dots, \boldsymbol{c}_L(D)^T]^T$, as follows:

$$C(D) = U(D).G(D) \tag{16}$$

Note that, since the generator matrix $G(D)$ is full rank, we have $rank(U(D)) = rank(C(D))$. Moreover, if the binary transmitted messages $\boldsymbol{u}_i(D)$ are considered with uniform distribution entries,

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

it is expected that, for values of $L \gg k$, the rank of matrix $U(D)$ (and subsequently, the rank of matrix $C(D)$) will be equal to k. We exploit this property along with non-catastrophicity condition (6) to blindly identify the parameters of convolutional code $\mathcal{C}(n,k,\nu)$. To this end, we propose an iterative method for $\hat{n} = 2,3,\dots$. The relation (6) is valide only when $\hat{n} = \mathrm{n}$ is admitted. The following steps are involved in every iteration to investigate the hypothesis of $\hat{n} = n$:

Step 1: For $i = 1,\dots,L$, demultiplex $c_i(D)$ at depth $\hat{n}$ as $\boldsymbol{c}_i^{\hat{n}}(D) = [c_{i1}^{\hat{n}}(D),\dots,c_{i\hat{n}}^{\hat{n}}(D)]$.

Step 2: Construct the matrix $C^{\hat{n}}(D) = [\boldsymbol{c}_1^{\hat{n}}(D)^T,\dots,\boldsymbol{c}_L^{\hat{n}}(D)^T]^T$.

Step 3: Calculate the rank of the matrix $C^{\hat{n}}(D)$denoted by $\hat{k}$.

Note that if $\hat{n} = n$, then $C^{\hat{n}}(D)$ is equal to $C(D)$. In this case, if we have $rank(U(D)) = k$ (i.e.L is sufficiently large), then the rank of $C^{\hat{n}}(D)$ is equal to k, i.e., $\hat{k} = k$. However, this is not true for any $\hat{n} < n$ because, in this case, the demultiplexed vectors $c_i^{(\hat{n})}(D), i = 1,\dots,L$ that can be seen as random vectors, do not necessarily belong to the same code. Hence, if L is sufficiently greater than $\hat{n}$, the rank of $C^{\hat{n}}(D)$ is equal to $\hat{n}$. As a result, for any $\hat{n} < n$, we have $\hat{k} = \hat{n}$. According to these statements, we must continue the iteration until a matrix $C^{\hat{n}}(D)$ with rank deficiency is found (i.e.$\hat{k} < \hat{n}$). If such matrix is found for $\hat{n}$, we conclude $n = \hat{n}$ and $k = \hat{k}$.

In another case, when L is not sufficiently greater than $\hat{n}$, it is probable that we encounter rank deficiency for some $\hat{n} < n$. To address this problem, we add the following four steps to ensure the initial guess, in which the validity of the guesses $\hat{n}$ and $\hat{k}$, corresponding to a matrix $C^{\hat{n}}(D)$ with rank deficiency, are tested:

Step 4: Select two $\hat{k} \times \hat{n}$ full-rank submatrices of $C^{\hat{n}}(D)$, denoted by $C_1^{\hat{n}}(D)$ and $C_2^{\hat{n}}(D)$.

Step 5: For $i = 1,2$, calculate the $\hat{k} \times \hat{k}$ minors of submatrix $C_i^{\hat{n}}(D)$, denoted by $\Delta_{C_i^{\hat{n}}}^{j}(D), j = 1,\dots,\binom{\hat{n}}{\hat{k}}$, and construct the vector $\boldsymbol{\Delta}_{C_i^{\hat{n}}}(D) = [\Delta_{C_i^{\hat{n}}}^{1}(D),\dots,\Delta_{C_i^{\hat{n}}}^{\binom{\hat{n}}{\hat{k}}}(D)]$, which we call as the minor vector of $C_i^{\hat{n}}(D)$.

Step 6: For $i = 1,2$, calculate GCD of the entries of $\boldsymbol{\Delta}_{C_i^{\hat{n}}}(D)$, whichis denoted by $\Delta_{U_i^{\hat{n}}}(D)$.

Step 7: For $i = 1,2$, divide the vector $\boldsymbol{\Delta}_{C_i^{\hat{n}}}(D)$ by $\Delta_{U_i^{\hat{n}}}(D)$ to obtain the vector $\boldsymbol{\Delta}_{G_i^{\hat{n}}}(D)$.

Assume that the initial guess is true, i.e. we have $\hat{n} = n$ and $\hat{k} = k$. In this case, the demultiplexed matrix of $C^{\hat{n}}(D)$ is equal to the code matrix $C(D)$. Let $U_1(D)$ and $U_2(D)$ be two $k \times k$ full-rank submatrices from the message matrix $U(D)$ whose rows are selected corresponding to the submatrices $C_1^n(D)$ and $C_2^n(D)$. From (16), we have:

$$C_i^n(D) = U_i(D).G(D); \quad i = 1,2 \tag{17}$$

Subsequently, for $i = 1,2$, we can immediately conclude:

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

$$\Delta^{j}_{C_i^n}(D) = |U_i(D)|.\Delta^{j}_{G}(D)\,; \quad j = 1, \dots, \binom{n}{k} \tag{18}$$

where $|.|$ denotes the matrix determinant. We can rewrite equation (18), in vector notation, as follows:

$$\boldsymbol{\Delta}_{C_i^n}(D) = |U_i(D)|.\boldsymbol{\Delta}_{G}(D) \tag{19}$$

where $\boldsymbol{\Delta}_{G}(D) = [\Delta^{1}_{G}(D), \dots, \Delta^{\binom{n}{k}}_{G}(D)]$ is the minor vector of matrix $G(D)$. According to (6), the entries of $\boldsymbol{\Delta}_{G}(D)$ are relatively prime, since we assumed that $G(D)$ is a zero-delay non-catastrophic encoder. As a result, one can simply conclude form (19) that for $i = 1,2$, $\Delta_{U_i^n}(D)$ (calculated in step 6) is equal to $|U_i(D)|$. Therefore, the calculated minor vectors in the final step are equal to $\boldsymbol{\Delta}_{G}(D)$. In summary, if $\hat{n} = n$ and $\hat{k} = k$, then the two minor vectors $\boldsymbol{\Delta}_{G_1^{\hat{n}}}(D)$ and $\boldsymbol{\Delta}_{G_2^{\hat{n}}}(D)$ are equal. However, this equality is not true for any $\hat{n} < n$, since in this case, the demultiplexed vectors $\boldsymbol{c}_i^{\hat{n}}(D),\ i = 1, \dots, L$ do not necessarily belong to the same code and can be seen as random vectors. Thus, the above statements are not true for any $\hat{n} < n$, and so the two resulted minor vectors in step 7 would be unequal. As a result, if two minor vectors are equal, we conclude that the initial guesses $\hat{n} = n$ and $\hat{k} = k$ are true.

In step 4, we restricted $C_1^{\hat{n}}(D)$ and $C_2^{\hat{n}}(D)$ to be two $\hat{k} \times \hat{n}$ submatrices of $C^{\hat{n}}(D)$, only to simplify the description of the proposed method. However, these can be constructed from any $\hat{k}$ independent vectors chosen from the row space of $C^{\hat{n}}(D)$.

So far, we have identified the parameter n and k. Now, according to (17), we can divide $C_i^n(D)$ (from step 4) by $\Delta_{U_i^n}(D)$ (from step 6 that is equal to $|U_i(D)|$), for $i = 1$ or 2, to obtain a non-catastrophic encoder for the code space $\mathcal{C}(n, k, \nu)$. The resulted generator matrix is denoted by $\hat{G}(D)$. Then, from theorem 1, we get:

$$\nu = \max\left(\deg(\Delta^{1}_{\hat{G}}(D), \dots, \Delta^{\binom{n}{k}}_{\hat{G}}(D))\right) \tag{20}$$

The proposed method is presented as a pseudo-codein Algorithm2. In the complexity viewpoint, the $\hat{n}$'th iteration of this algorithm is composed of a polynomial matrix rank calculation in step 3, two different $\hat{k}$ independent vectors finding process in step 4, $2\binom{\hat{n}}{\hat{k}}$ polynomial matrix determinant calculation in step 5, $2\binom{\hat{n}}{\hat{k}}$ polynomial GCD calculations in step 6, and $2\hat{n}$ polynomial divisions in step 7. Let d_{max} denote the maximum degree of the received codewords. Then steps 3 and 4 can be executed by Gauss elimination algorithm with maximum complexity $\tilde{O}(L\hat{n}d_{max})$. Also, every $\hat{k} \times \hat{k}$ minor in step 5 can be calculated with complexity $\tilde{O}(\frac{\hat{k}^3 d_{max}}{\hat{n}})$.

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

Alg. 2. Blind identification of code $\mathcal{C}(n,k,\nu)$

Input: Different codewords $c_1(D),\dots,c_L(D)$

Output: $n,\ k,\ \nu$ and a non-catastrophic encoder $\hat{G}(D)$

$f \Leftarrow 0$

$\hat{n} \Leftarrow 2$

while $f \neq 1$ **do**

  for $i = 1$ to L **do**

    $c_i^{\hat{n}}(D) \Leftarrow c_i^{(\hat{n})}(D)$; Step 1

  end

  $C^{\hat{n}}(D) \Leftarrow [c_1^{\hat{n}}(D)^T,\dots,c_L^{\hat{n}}(D)^T]^T$; Step 2

  $\hat{k} \Leftarrow rank(C^{\hat{n}}(D))$; Step 3

  if $\hat{k} \prec \hat{n}$ **then**

    $\hat{n} \Leftarrow \hat{n} + 1$

  else

    for $i = 1$ to 2 **do**

      $C_i^{\hat{n}}(D) \Leftarrow$ find $\hat{k}$ independent vectors from row space $C^{\hat{n}}(D)$; Step 4

      for $j = 1$ to $\binom{\hat{n}}{\hat{k}}$ **do**

        $\Delta_{C_i}^{j} \Leftarrow$ calculate the j'th minor of $C_i^{\hat{n}}(D)$; Step 5

      end

      $\boldsymbol{\Delta}_{C_i^{\hat{n}}}(D) \Leftarrow [\Delta_{C_i^{\hat{n}}}^{1}(D),\dots,\Delta_{C_i^{\hat{n}}}^{\binom{\hat{n}}{\hat{k}}}(D)]$

      $\Delta_{U_i^{\hat{n}}}(D) \Leftarrow 1$

      for $j = 1$ to $\binom{\hat{n}}{\hat{k}}$ **do**

        $\Delta_{U_i^{\hat{n}}}(D) \Leftarrow GCD\left(\Delta_{U_i^{\hat{n}}}(D), \Delta_{C_i^{\hat{n}}}^{j}(D)\right)$; Step 6

      end

      $\boldsymbol{\Delta}_{G_i^{\hat{n}}}(D) \Leftarrow \frac{\boldsymbol{\Delta}_{C_i^{\hat{n}}}(D)}{\Delta_{U_i^{\hat{n}}}(D)}$; Step 7

    end

    if $\boldsymbol{\Delta}_{G_1^{\hat{n}}}(D) \neq \boldsymbol{\Delta}_{G_2^{\hat{n}}}(D)$ **then**

      $\hat{n} \Leftarrow \hat{n} + 1$

    end

    $f \Leftarrow 1$

  end

end

$\hat{G}(D) \Leftarrow \frac{C_1^{\hat{n}}(D)}{\Delta_{U_1^{\hat{n}}}(D)}$ *or* $\frac{C_2^{\hat{n}}(D)}{\Delta_{U_2^{\hat{n}}}(D)}$

$n \Leftarrow \hat{n}$

$k \Leftarrow \hat{k}$

$\nu \Leftarrow \max\left(\deg(\Delta_{\hat{G}}^{1}(D),\dots,\Delta_{\hat{G}}^{\binom{n}{k}}(D))\right)$

Manuscript received 15-Dec.-2016 and revised 14-March-2018, Accepted on 23-April-2018

P- ISSN: 2322-4088
E- ISSN: 2322-3936

Note that the matrices $\boldsymbol{\Delta}_{C_i^{\hat{n}}}(D)$, $i = 1,2$, have degree $\frac{\hat{k}d_{max}}{\hat{n}}$ at the worst case. Thus, every GCD in step 6 or every division in step 7 can be calculated with the maximum complexity $\tilde{\mathcal{O}}(\frac{\hat{k}^2 d_{max}^2}{\hat{n}^2})$. Since the values of n, k and L are much smaller than the maximum received codeword length, we can express the complexity only in term of d_{max}. As a result, we can approximate the complexity of Algorithm2 with $\tilde{\mathcal{O}}(d_{max}^2)$.

As a final remark on this method, note that the steps 1 to 3 require at least $k+1$ different codewords to introduce the values $\hat{n} = n$ and $\hat{k} = k$ as an initial guess for next steps. The $k+1$ codewords are sufficient when the $(k+1) \times k$ message matrix $U(D)$ is full-rank, or equivalently the rank of $C(D)$ is equal to k. In this case, the $(k+1) \times n$ matrix $C^n(D)$ has a rank deficiency, and so $\hat{n} = n$ is detected as a valid guess, and also the rank of $C^n(D)$ is equal to k, i.e., $\hat{k} = k$. Moreover, step 4 requires at least $k+1$ different codewords to find two different $k \times k$ full-rank submatrix from $(k+1) \times k$ matrix $C^n(D)$. Note that the $k+1$ codewords are sufficient if at least two $k \times k$ submatrices from $C^n(D)$ are full-rank. Therefore, the $k+1$ different codewords are sufficient for the proposed blind method if it is possible to find at least two full-rank $k \times k$ submatrices from the received code matrix $C(D)$. Finally, we again emphasize that the codeword lengths is no matter to this algorithm.

Example 2: Assume that the transmitter employs the following non-catastrophic encoder $G(D)$ to generate the convolutional code $\mathcal{C}(3,2,3)$:

$$G(D) = \begin{bmatrix} 1 & D & 1+D^2 \\ D & 1+D & 1 \end{bmatrix}$$

Consider the four message sequences as:

$$u_1(D) = [D \quad 1+D], \quad u_2(D) = [1+D^2 \quad D]$$
$$u_3(D) = [0 \quad D^2+D^3], \quad u_4(D) = [1 \quad 1+D^2]$$

After encoding and transmitting the above messages, the corresponding received errorless codewords will be:

$$c_1(D) = D + D^2 + D^6 + D^{11}$$
$$c_2(D) = 1 + D^2 + D^5 + D^7 + D^{10} + D^{14}$$
$$c_3(D) = D^7 + D^8 + D^9 + D^{11} + D^{12} + D^{14}$$
$$c_4(D) = 1 + D + D^3 + D^7 + D^9 + D^{10}$$

In the first iteration, we examine the hypothesis of $\hat{n} = 2$ and demultiplex the codewords in depth 2 as following matrix (step 2):

$$C^2(D) = \begin{bmatrix} D+D^3 & 1+D^5 \\ D+D^5+D^6 & 1+D^2+D^3 \\ D^4+D^6+D^7 & D^3+D^4+D^5 \\ 1+D^5 & 1+D+D^3+D^4 \end{bmatrix}$$

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

One can simply verify that the rank of $C^2(D)$ is equal to 2 (step 3). Thus, we conclude that the hypothesis of $\hat{n} = 2$ is false, since in this case we have $\hat{k} = \hat{n} = 2$. In the next iteration, we examine that the hypothesis of $\hat{n} = 3$ and demultiplex the codewords in depth 3 as following matrix (step 2):

$$C^3(D) = \begin{bmatrix} D^2 & 1 & 1+D^3 \\ 1 & D^2+D^3 & 1+D+D^4 \\ D^3+D^4 & D^2+D^4 & D^2+D^3 \\ 1+D+D^2 & 1+D^2+D^3 & 0 \end{bmatrix}$$

In this case, we find $rank(C^3(D)) = 2$ (step 3). Since $\hat{k} < \hat{n}$, we select $\hat{n} = 3$ and $\hat{k} = 2$ as possibly true guesses, and verify their validity by steps 4 to 7. To this end, we first find two 2×2 full-rank submatrices from $C^3(D)$ as follows (step 4):

$$C_1^3(D) = \begin{bmatrix} D^2 & 1 & 1+D^3 \\ 1 & D^2+D^3 & 1+D+D^4 \end{bmatrix}$$

$$C_2^3(D) = \begin{bmatrix} D^2 & 1 & 1+D^3 \\ D^3+D^4 & D^2+D^4 & D^2+D^3 \end{bmatrix}$$

Now, we calculate the minor vectors as follows (step 5):

$$\boldsymbol{\Delta}_{C_1^3}(D) = [1+D^4+D^5 \quad 1+D^2+D^6 \quad 1+D^2+D^3+D^4+D^5+D^6]$$

$$\boldsymbol{\Delta}_{C_2^3}(D) = [D^3+D^6D^3+D^5+D^6+D^7D^3+D^4+D^5+D^7]$$

and then, by totaly four times use of Euclidean algorithm, we find (step 6):

$$\Delta_{U_1^3}(D) = 1+D+D^3, \quad \Delta_{U_2^3}(D) = D^3+D^4$$

In final step, by$\boldsymbol{\Delta}_{C_1^3}(D)/\Delta_{U_1^3}(D)$ and $\boldsymbol{\Delta}_{C_2^3}(D)/\Delta_{U_2^3}(D)$, we obtain (step 7):

$$\boldsymbol{\Delta}_{G_1^3}(D) = \boldsymbol{\Delta}_{G_2^3}(D) = [1+D+D^2 \quad 1+D+D^3 \quad 1+D^2+D^3]$$

Since the minor vectors are equal, we conclude that the hypothesises $\hat{n} = 3$ and $\hat{k} = 2$ are true. Moreover, from (20), we simply find $\nu = 3$. Finally, we can introduce submatrices $C_1^3(D)/\Delta_{U_1^3}(D)$ or $C_2^3(D)/\Delta_{U_2^3}(D)$ as a non-catastrophic encoder for $\mathcal{C}$.

IV. CONCLUSION

This paper, a simple method is introduce to blind identification of convolutional code of any rate from complete synchronous noiseless codewords. It is assumed that the transmitter uses a non-catastrophic encoder where is accepted in practical codes. We used the non-catastrophicity properties to identify the convolutional code parameters. In rate $1/n$, we proposed an iterative algorithm with three steps in which in every iteration examine the validity of a possible value for parameter n. This method can identify the parameters n and ν and also a minimal-basic set for code $\mathcal{C}(n, 1, \nu)$. Moreover, we have shown that only two codewords are sufficient and necessary to this method. In rate k/n, we have presented an iterative algorithm with seven. The first three steps, make an initial guess about the parameters n and k, then the last four steps verify the validity of this guess. Moreover, this method identifies the parameter ν and also a non-catastrophic basis for the code space $\mathcal{C}(n, k, \nu)$. Also, it is

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936

shown that the algorithm requires at least $k + 1$ different codewords to succeed. We emphasize that although the proposed method is limited to errorless scenarios, the development for noisy environments would be done as our future work.

REFERENCES

[1] F. Wang, Z. Huang and Y. Zhou, "A Method for Blind Recognition of Convolution Code Based on Euclidean Algorithm," International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1414-1417, Sept. 2007.

[2] H. Xie, X. m. Chai, F. h. Wang and Z. t. Huang, "A Method for Blind Identification of Rate 1/2 Convolutional Code Based on Improved Euclidean Algorithm," IEEE International Conference on Signal Processing (ICSP), pp. 1307-1310, Oct. 2012.

[3] M. Marazin, R. Gautier and G. Burel, "Blind Recovery of the Second Convolutional Encoder of a Turbo-Code When its Systematic Outputs are Punctured," Military Technical Academy (MTA) Review, vol. 19, no. 2, pp. 213-232, June 2009.

[4] Y. Zrelli, M. Marazin, E. Rannou and R. Gautier, "Blind Identification of Convolutional Encoder Parameters over GF(2m) in the Noiseless Case," International Conference on Computer Communications and Networks (ICCCN), pp. 1-5, July 2011.

[5] M. Marazin, R. Gautier and G. Burel, "Dual Code Method for Blind Identification of Convolutional Encoder for Cognitive Radio Receiver Design," IEEE Globecom Workshops, pp. 1-6, Nov. 2009.

[6] Marazin, M., Gautier, R. & Burel, G., "Blind Recovery of k/n Rate Convolutional Encoders in a Noisy Environment", EURASIP Journal on Wireless Communications and Networking, vol. 2011, no. 168, pp. 1-9, 2011.

[7] Forney, J. G., "Convolutional codes I: Algebraic structure", IEEE Transactions on Information Theory, vol. 16, no. 6, pp. 720-738, 1970.

[8] R. Johannesson and Z. X. Wan, "A Linear Algebra Approach to Minimal Convolutional Encoders," IEEE Transactions on Information Theory, vol. 39, no. 4, pp. 1219-1233, July 1993.

[9] G. D. Forney, R. Johannesson and Z. X. Wan, "Minimal and Canonical Rational Generator Matrices for Convolutional Codes," IEEE Transactions on Information Theory, vol. 42, no. 6, pp. 1865-1880, Nov. 1996.

Manuscript received 15-Dec.-2016 and revised 14-March-2018,
Accepted on 23-April-2018
P- ISSN: 2322-4088
E- ISSN: 2322-3936