Blind recovery of k/n rate convolutional encoders in a noisy environment

Mélanie Marazin, Roland Gautier, Gilles Burel

EURASIP Journal on Wireless Communications and Networking
a SpringerOpen Journal

RESEARCH Open Access

Blind recovery of *k/n* rate convolutional encoders in a noisy environment

Melanie Marazin[1,2], Roland Gautier[1,2*] and Gilles Burel[1,2]

Abstract

In order to enhance the reliability of digital transmissions, error correcting codes are used in every digital communication system. To meet the new constraints of data rate or reliability, new coding schemes are currently being developed. Therefore, digital communication systems are in perpetual evolution and it is becoming very difficult to remain compatible with all standards used. A cognitive radio system seems to provide an interesting solution to this problem: the conception of an intelligent receiver able to adapt itself to a specific transmission context. This article presents a new algorithm dedicated to the blind recognition of convolutional encoders in the general *k/n* rate case. After a brief recall of convolutional code and dual code properties, a new iterative method dedicated to the blind estimation of convolutional encoders in a noisy context is developed. Finally, case studies are presented to illustrate the performances of our blind identification method.

Keywords: intelligent receiver, cognitive radio, blind identification, convolutional code, dual code

1 Introduction

In a digital communication system, the use of an error correcting code is mandatory. This error correcting code allows one to obtain good immunity against channel impairments. Nevertheless, the transmission rate is decreased due to the redundancy introduced by a correcting code. To enhance the correction capabilities and to reduce the impact of the amount of redundancy introduced, new correcting codes are always under development. This means that communication systems are in perpetual evolution. Indeed, it is becoming more and more difficult for users to follow all the changes to stay up-to-date and also to have an electronic communication device always compatible with every standard in use all around the world. In such contexts, cognitive radio systems provide an obvious solution to these problems. In fact, a cognitive radio receiver is an intelligent receiver able to adapt itself to a specific transmission context and to blindly estimate the transmitter parameters for self-reconfiguration purposes only with knowledge of the received data stream. As convolutional codes are among the most currently used error-correcting codes, it seemed to us worth gaining more insight into the blind recovery of such codes.

In this article, a complete method dedicated to the blind identification of parameters and generator matrices of convolutional encoders in a noisy environment is treated. In a noiseless environment, the first approach to identify a rate $1/n$ convolutional encoder was proposed in [1]. In [2,3] this method was extended to the case of a rate k/n convolutional encoder. In [4], we developed a method for blind recovery of a rate k/n convolutional encoder in turbocode configuration. Among the available methods, few of them are dedicated to the blind identification of convolutional encoders in a noisy environment. An approach allowing one to estimate a dual code basis was proposed in [5], and then in [6] a comparison of this technique with the method proposed in [7] was given. In [8], an iterative method for the blind recognition of a rate $(n\text{-}1)/n$ convolutional encoder was proposed in a noisy environment. This method allows the identification of parameters and generator matrix of a convolutional encoder. It relies on algebraic properties of convolutional codes [9,10] and dual code [11], and is extended here to the case of rate k/n convolutional encoders.

This article is organized as follows. Section 2 presents some properties of convolutional encoders and dual codes. Then, an iterative method for the blind identification of

* Correspondence: roland.gautier@univ-brest.fr
[1]Université Européenne de Bretagne, Rennes, France
Full list of author information is available at the end of the article

convolutional encoders is described in Section 3. Finally, the performances of the method are discussed in Section 4. Some conclusions and prospects are drawn in Section 5.

2 Convolutional encoders and dual code

Prior to explain our blind identification method, let us recall the properties of convolutional encoders used in our method.

2.1 Principle and mathematical model

Let C be an (n, k, K) convolutional code, where n is the number of outputs, k is the number of inputs, K is the constraint length, and $C^{\perp}$ be a dual code of C. Let us also denote by $G(D)$ a polynomial generator matrix of rank k defined by:

$$G(D) = \begin{bmatrix} g_{1,1}(D) & \cdots & g_{1,n}(D) \\ \vdots & \ldots & \vdots \\ g_{k,1}(D) & \cdots & g_{k,n}(D) \end{bmatrix} \quad (1)$$

where $g_{i,j}(D)$, $\forall i = 1,..., k$, $\forall j = 1,..., n$, are generator polynomials and D represents the delay operator. Let μ_i be the memory of the ith input:

$$\mu_i = \max_{j=1,...,n} \deg g_{i,j}(D) \quad \forall i = 1, ..., k \quad (2)$$

where deg is the degree of $g_{i,j(D)}$. The overall memory of the convolutional code, denoted μ, is

$$\mu = \max_{i=1,...,k} \mu_i = K - 1 \quad (3)$$

If the input sequence is denoted by $m(D)$ and the output sequence by $c(D)$, the encoding process can be described by

$$c(D) = m(D).G(D) \quad (4)$$

In practice, the encoder used is usually an optimal encoder. An encoder is optimal, [10], if it has the maximum possible free distance among all codes with the same parameters (n, k, and K). This is because the error correction capability of such optimal codes is much higher. Furthermore, their good algebraic properties [9,10] can be judiciously exploited for blind identification.

To model the errors generated by the transmission system, let us consider the binary symmetric channel (BSC) with the error probability, P_e, and denote by $e(D)$ the error pattern and by $y(D)$ the received sequence so that:

$$y(D) = c(D) + e(D) \quad (5)$$

Let us also denote by $e(i)$ the ith bit of $e(D)$ so that: $Pr(e(i) = 1) = P_e$ and $Pr(e(i) = 0) = 1 - P_e$. The errors are assumed to be independent.

In this article, the noise is modeled by a BSC. This BSC can be used to model an AWGN channel in the context of a hard decision decoding algorithm. Indeed, the BSC can be seen as an equivalent model to the set made of the combination of the modulator, the true channel model (AWGN by example) and the demodulator (Matched filter or Correlator + Decision Rule). Furthermore, in mobile communications, channels are subject to multipath fading, which leads, in the received bit stream, to burst errors. But, a convolutional encoder alone is not efficient in this case. Therefore, an interleaver is generally used to limit the effect of these burst errors. In this context, after the deinterleaving process, on the receiver side, the errors (so the equivalent channel including the deinterleaver) can also be modeled by a BSC.

2.2 The dual code of convolutional encoders

The dual code generator matrix of a convolutional encoder, termed a parity check matrix, can also be used to describe a convolutional code. This $((n - k) \times n)$ polynomial matrix verifies the following property:

Theorem 1 *Let $G(D)$ be a generator matrix of C. If an $((n - k) \times n)$ polynomial matrix, $H(D)$, is a parity check matrix of C, then:*

$$G(D).H^T(D) = 0 \quad (6)$$

where $.^T$ is the transpose operator.

Corollary 1 *Let $H(D)$ be a parity check matrix of C. The output sequence $c(D)$ is a codeword sequence of C if and only if:*

$$c(D).H^T(D) = 0 \quad (7)$$

The parity check matrix is an $((n - k) \times n)$ matrix such that:

$$H(D) = \begin{bmatrix} h_{1,1}(D) & \cdots & h_{1,k}(D) & h_0(D) & & \\ \vdots & \ldots & \vdots & & \ddots & \\ h_{n-k,1}(D) & \cdots & h_{n-k,k}(D) & & & h_0(D) \end{bmatrix} \quad (8)$$

where $h_0(D)$ and $h_{i,j}(D)$ are the generator polynomials of $H(D)$, $\forall i = 1,..., n - k$ and $\forall j = 1,..., k$.

Let us denote by $\mu^{\perp}$ the memory of the dual code. According to the properties of a dual code and convolutional encoders [9,11], this memory is defined by

$$\mu^{\perp} = \sum_{i=1}^{k} \mu_i \quad (9)$$

The polynomial, $f(D) = \sum_{i=0}^{\infty} f(i).D^i$, is a delayfree polynomial if $f(0) = 1$. According to [12], if the polynomial $h_0(D)$ is a delayfree polynomial, then the convolutional encoder is realizable. It follows that the generator polynomial, $h_0(D)$, is such that

$$h_0(D) = 1 + h_0(1).D + \cdots + h_0(\mu^\perp).D^{\mu^\perp} \tag{10}$$

Let us denote by H, the binary form of $H(D)$ defined by

$$H = \begin{pmatrix} H_{\mu^\perp} \cdots & H_1 & H_0 & & \\ & H_{\mu^\perp} \cdots & H_1 & H_0 & \\ & & H_{\mu^\perp} \cdots & H_1 & H_0 \\ & & & \ddots\ \ddots & \ddots\ \ddots \end{pmatrix} \tag{11}$$

where H_i, $\forall i = 0,..., \mu^\perp$, are matrices of size $((n - k) \times n)$ such that

$$H_i = \begin{bmatrix} h_{1,1}(i) & \cdots & h_{1,k}(i) & h_0(i) & \\ \vdots & \cdots & \vdots & & \ddots \\ h_{n-k,1}(i) & \cdots & h_{n-k,k}(i) & & h_0(i) \end{bmatrix} \tag{12}$$

The parity check matrix (11) is composed of shifted versions of the same $(n - k)$ vectors. These vectors of size $n.(\mu^\perp + 1)$ and denoted by $\mathbf{h}_j$ ($\forall j = 1,..., n - k$) are defined by

$$h_j = \left(H_{\mu^\perp}^{(j)}\ H_{\mu^{\perp-1}}^{(j)} \cdots H_1^{(j)}\ H_0^{(j)} \right) \tag{13}$$

where $H_i^{(j)}$, which correspond to the jth row of H_i, is a row vector of size n such that

$$H_i^{(j)} = \left(h_{j,1}(i) \cdots h_{j,k}(i)\ \mathbf{0}_{j-1}\ h_0(i)\ \mathbf{0}_{n-k-j} \right) \tag{14}$$

In (14), $\mathbf{0}_l$ is a zero vector of size l.

In the case of a rate k/n convolutional encoder, each vector $\mathbf{h}_j$ (13) is composed of $(n - k - 1).(\mu^\perp + 1)$ zeros. In this configuration, the system given in (7) is split into $(n - k)$ systems:

$$\begin{aligned} &\left[c_1(D) \ \cdots \ c_k(D)\ \ c_{k+s}(D) \right] \cdot \begin{bmatrix} h_{s,1}(D) \\ \vdots \\ h_{s,k}(D) \\ h_0(D) \end{bmatrix} \\ &= \sum_{i=1}^{k} c_i(D).h_{s,i}(D) + c_{k+s}(D).h_0(D) = 0, \end{aligned} \tag{15}$$

$\forall s = 1,...,(n - k)$. Thus, the $(n - k)$ vectors (13), called parity checks, are such that

$$h_s = \left(H_{\mu^\perp}^{(s)}\ H_{\mu^{\perp-1}}^{(s)} \cdots H_0^{(s)} \right) \tag{16}$$

where $H_i^{(s)}$ is a row vector of size $(k + 1)$ defined by:

$$H_i^{(s)} = \left(h_{s,1}(i) \cdots h_{s,k}(i)\ h_0(i) \right) \tag{17}$$

Let us denote by S the size of these parity checks of the code (16) such that

$$S = (k + 1).\left(\mu^\perp + 1\right) \tag{18}$$

It follows from (16) and (10) that the $(n - k)$ parity checks, $\mathbf{h}_s$, are vectors of degree $(S - 1)$.

3 Blind recovery of convolutional code

This section deals with the principle of the proposed blind identification method in the case where the intercepted sequence is corrupted. Only few methods are available for blind identification in a noisy environment: for example, an Euclidean algorithm-based approach was developed and applied to the case of a rate 1/2 convolutional encoder [13]. At nearly the same time, a probabilistic algorithm based on the Expectation Maximization (EM) algorithm was proposed in [14] to identify a rate $1/n$ convolutional encoder. Further to our earlier development of a method of blind recovery for a convolutional encoder of rate $(n - 1)/n$ [8], it appeared to us worth extending it, here, to the case of a rate k/n convolutional encoder. Prior to describing the iterative method in use, which is based on algebraic properties of an optimal convolutional encoder [9,10] and dual code [11], let us briefly recall the principle of our blind identification method when the intercepted sequence is corrupted.

3.1 Blind identification of a convolutional code: principle

This method allows one to identify the parameters (n, k, and K) of an encoder, the parity check matrix, and the generator matrix of an optimal encoder. Its principle is to reshape columnwise the intercepted data bit stream, $\mathbf{y}$, under matrix form. This matrix, denoted R_l, is computed for different values of l, where l is the number of columns. The number of rows in each matrix is equal to L. If the received sequence length is L', then the number of rows of R_l is $L = \left\lfloor \frac{L'}{l} \right\rfloor$, where $\lfloor . \rfloor$ stands for the integer part. This construction is illustrated in Figure 1.

If the received sequence is not corrupted ($y = c \Rightarrow e = 0$), for $\alpha \in \mathbb{N}$, we have shown in [8] that the rank in Galois Field, $GF(2)$, of each matrix R_l has two possible values:

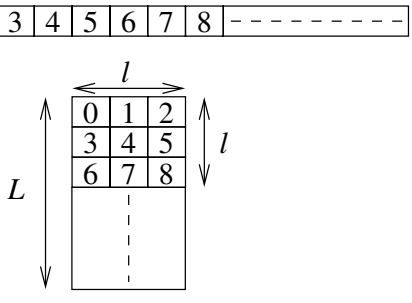

Figure 1 Example of matrix R_l. An example of the received data bit stream reshape under matrix form.

• If $l \neq \alpha.n$ or $l < n_a$

$$\text{rank}(R_l) = l \tag{19}$$

• If $l = \alpha.n$ and $l \geq n_a$

$$rank(R_l) = l.\frac{k}{n} + \mu^{\perp} < l \tag{20}$$

where n_a is a key-parameter which corresponds to the first matrix R_l with a rank deficiency. Indeed, in [8], for a rate $(n - 1)/n$ convolutional encoder, this parameter proved to be such that

$$n_a = n.\left(\mu^{\perp} + 1\right) \tag{21}$$

In this configuration, n_a is equal to the size of the parity check (S). But, what is its value in general for a rate k/n convolutional encoder?

For a rate k/n convolutional encoder, we show in Appendix A that the size of the first matrix which exhibits a rank deficiency, n_a, is equal to

$$n_a = n.\left\lfloor \frac{\mu^{\perp}}{n-k} + 1 \right\rfloor \tag{22}$$

From (22), it is obvious that the parameter, n_a, is not equal to the size of the $(n - k)$ parity check (16) of the code. In Appendix B, a discussion about the value of a rank deficiency of matrix R_{n_a} is proposed.

3.2 Blind identification of convolutional code: method

A prerequisite to the extension of the method applied in [8] to the case of a rate k/n convolutional encoder is the identification of the parameter, n. Then, a basis of dual code has to be built to further deduce the value of n_a that corresponds to the size of the parity check with the smallest degree. Using both this parameter and (22), one can assume different values for k and $\mu^{\perp}$ Then, the $(n - k)$ parity check (16) and a generator matrix of the code can be estimated.

To identify the number of outputs, n, let us evaluate the likely-dependent columns of R_l. Then, the values of l at which R_l matrices seem to be of degenerated rank are detected by converting each R_l matrix into a lower triangular matrix (G_l) through use of the Gauss Jordan Elimination Through Pivoting adapted to $GF(2)$:

$$G_l = A_l.R_l.B_l \tag{23}$$

where A_l is a row-permutation matrix of size $(L \times L)$ and B_l is a matrix of size $(l \times l)$ that describes the column combination. Let $N_l(i)$ be the number of 1 in the lower part of the ith column in the matrix, G_l. In [15,16], this number was used to estimate an optimal threshold (γ_{opt}), which allows us to decide whether the ith column of the matrix R_l is dependent on the other columns. This optimal threshold is such that the sum of the missing probabilities is as small as possible. The numbers of detected dependent columns, denoted as $Z(l)$, are such that

$$Z(l) = \text{Card}\left\{ i \in \{1, ..., l\} \,|N_l(i) \leq \frac{(L-l).\gamma_{\text{opt}}}{2} \right\} \tag{24}$$

where Card{x} is the cardinal of x. So, the gap between two non-zero cardinals, $Z(l)$, is equal to the estimated codeword size ($\hat{n}$). Let $\mathcal{I}$ be a set of l-values where the cardinal is non-zero. From the matrix, $B_i, \forall i \in \mathcal{I}$, one can build a dual code basis. Let $\mathcal{I}$ be a $((L - i) \times i)$ matrix composed of the last $(L - i)$ rows of R_i. If $\mathbf{b}_j$, $\forall j = 1,..., i$, represents the jth column of B_i, $\mathbf{b}_j$ is considered as a linear form close to the dual code on condition that:

$$d\left(R_i^1.b_j\right) \leq (L-i).\gamma_{opt} \tag{25}$$

where d(x) is the Hamming weight of x. Let us denote a set of all linear forms by $\mathcal{D}$. Within the set of detected linear forms, the one with the smallest degree is taken and denoted, here, by $\hat{\mathbf{h}}$, and its size by $\hat{n}_a$. From (22), one can make different hypotheses about k and $\mu^{\perp}$ values. This algorithm is summed up in Algorithm 1.

For a rate $(n - 1)/n$ convolutional encoder with $\hat{\mathbf{h}}$ as parity check, solving the system described in Property 1 (see Section 2) enables one to identify the generator matrix. One should, however, note that with a rate k/n convolutional code, a prerequisite to the identification of the generator matrix, $G(D)$, is the identification of the $(n - k)$ parity check, $\mathbf{h}_j$ of size S (see (16) and (18)).

Algorithm 1: Estimation of k and $\mu^{\perp}$

Input: Value of $\hat{n}$ and $\hat{n}_a$
Output: Value of $\hat{k}$ and $\hat{\mu}^{\perp}$
for $k' = 1$ *to* $\hat{n} - 1$**do**
  for $Z = 1$ *to* $\hat{n} - k'$**do**
    $\hat{\mu}^{\perp} = \left[\hat{\mu}^{\perp} \;\; \hat{n}_a.\left(1 - \frac{k'}{n}\right) - Z\right]$;
    $\hat{k} = \left[\hat{k} \;\; k'\right]$;
  end
end

It is done by building $(\hat{n} - \hat{k})$ row vectors denoted by x_s so that

$$x_s = \left(\gamma_1(t) \;\; \cdots \gamma_k(t) \;\; \gamma_{k+s}(t) \;\; \cdots\right), \tag{26}$$

$\forall s = 1,..., \forall s = 1, ..., \left(\hat{n} - \hat{k}\right)$. For each vector, x_s, a matrix, R_l^s, is built as previously done for R_l. Then, for each matrix R_l^s, a linear form of size S has to be estimated. This algorithm is summed up in Algorithm 2 where $\hat{\mathbf{h}}_s$ refers to the identified $\hat{n} - \hat{k}$ parity check.

Identification of the generator matrix from both these $(\hat{n} - \hat{k})$ parity checks and the whole set of the code

parameters can be realized by solving the system described in Property 1.

In [15,17], a similar approach, based on a rank calculation, is used to identify the size of an interleaver. In this article, an iterative process is proposed to increase the probability to estimate a good size of interleaver. The principle of this iterative process is to perform permutations on the R_l matrix rows to obtain a new virtual realization of the received sequence. These permutations increase the probability to obtain non-erroneous pivots during the Gauss Elimination process (23). Our earlier identification of a convolutional encoder relied on a similar approach [8]. Indeed, at the output of our algorithm, either: (i) the true encoder, or an optimal encoder, is identified or (ii) no optimal code is identified. But in case (ii), the probability of detecting an optimal convolutional encoder is increased by a new iteration of the algorithm.

The average complexity of one iteration of the process dedicated to the blind identification of convolutional encoder is $\mathcal{O}\left(l_{max}^4\right)$. Indeed, our blind identification method is divided into three steps: (i) identification of n, (ii) identification of a dual code basis, and (iii) identification of parity checks and a generator matrix. Each step consist of maximum (l_{max} - 1) process of Gaussian eliminations on R_l matrices of size ($L \times l$)

Algorithm 2: Estimation of $(\hat{n} - \hat{k})$ parity check.

Input: y, $\hat{n}$, $\hat{k}$ and $\hat{\mu}^{\perp}$

Output: $(\hat{n} - \hat{k})$ parity check

for $s = 1$ *to* $(\hat{n} - \hat{k})$ **do**

$x_s = \left(y_1(t) \quad \cdots y_{\hat{k}}(t) \quad y_{\hat{k}+s}(t) \quad \cdots\right)$;

for $l = \left(\hat{k} + 1\right).\left(\hat{\mu}^{\perp} + 1\right)$ *to* l_{max} **do**

Build matrix R_l^s of size ($L \times l$) with $\mathbf{x}_s$;

$R_l^s \rightarrow T_l = A_l.R_l^s.B_l$

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

if $N_l(i) \leq \frac{L-l}{2}.\gamma_{opt}$ **then**

if $\deg b_i^l = \left(\hat{k} + 1\right).\left(\hat{\mu}^{\perp} + 1\right)$ **then**

$\hat{h}_s = b_i^l$;

end

end

end

end

end

where $L = 2.l_{max}$. Thus, the average complexity is such that

$$\mathcal{O}\left(L.\sum_{l=2}^{l_{\max}} l^2\right) = \mathcal{O}\left(2.l_{\max}.l_{\max}^3\right) = \mathcal{O}\left(l_{\max}^4\right) \tag{27}$$

Thereby, the average complexity of the iterative process is

$$\mathcal{O}\left(nb_{\text{iter}}.l_{\max}^4\right) \tag{28}$$

where nb_{iter} is the number of iterations realized.

To identify all parameters of an encoder, it is necessary to obtain two consecutive rank deficiency matrix. So, the minimum value of $l_{\max}$ is

$$l_{\max} = n_a + n = n.\left\lfloor \frac{\mu^{\perp}}{n-k} + 1 \right\rfloor + n \tag{29}$$

Furthermore, in the literature, the parameters of convolutional encoders used take typically quite very small values. Indeed, the maximum parameters are such that

$$n_{\max} = 5, \quad k_{\max} = 4, \quad K_{\max} = 10 \tag{30}$$

A minimum value of $l_{\max}$ is given in Table 1 for three optimal encoders used in the following section dedicated to the analysis and performances study of our blind identification method.

4 Analysis and performances

In order to gain more insight into the performances of our blind identification technique, let us consider three convolutional encoders, $C(3,1,4)$, $C(3,2,3)$, and $C(2,1,7)$.

Let R_l be a matrix built from 20, 000 received bits with $l = 2, ..., 100$ and $L = 200$. It is very important to take into account the number of data to prove that our algorithm is well adapted for implementation in a realistic context. The amount of 20,000 bits is quite low with regards compared to standards. For example, in the case of mobile communications delivered by the UMTS at a data rate up to 2 Mbps, only 10 ms are needed to receive 20, 000 bits. Furthermore, the rates reached by standards in the future will be higher.

For each simulation, 1000 Monte Carlo were run, and focus was on

- the impact of the number of iterations upon the probability of detection;
- the global performances in terms of probability of detection.

In this article, the detection means complete identification of the encoders (parameters and generator matrix).

4.1 The detection gain produced by the iterative process

The number of iterations to be made is a compromise between the detection performances and the processing

Table 1 Different values of $l_{\max}$ (the minimum value of $l_{\max}$ is given for three optimal encoders)

Encoder	$l_{\max}$
$C(3, 2, 3)$	18
$C(3, 1, 4)$	9
$C(2, 1, 7)$	16

delay introduced in the reception chain (see [8]). To evaluate this number of iterations, let $P_{det}(i)$ be the probability of detecting the true encoder at the ith iteration.

The probability of detecting the true encoder, P_{det}, is called probability of detection.

- $C(3, 2, 3)$ convolutional encoder:

Figure 2 shows the probability of detecting the true encoder (P_{det}) compared with P_e for 1, 10, and 50 iterations. It shows that, for the $C(3, 2, 3)$ convolutional encoder, 10 iterations of the algorithm result in the best performances: indeed, there is no advantage in performing 50 iterations rather than 10. On the other hand, the gain between 1 and 10 iterations is huge.

- $C(3,1,4)$ convolutional encoder:

Figure 3 illustrates the evolution of P_{det} compared with P_e for 1, 10, and 50 iterations in the case of $C(3,1, 4)$ convolutional encoder. It shows that the gain between the 1st and the 50th iterations is nearly nil.

For a rate k/n convolutional code where $k \neq n - 1$, the algorithm presented in Figure 2 requires several iterations to estimate the $(n - k)$ parity checks (16). Consequently, for such codes ($k \neq n - 1$) there is no need to realize this iteration process. Indeed, the gain provided by our iterative process is not significant. But, for a rate $(n - 1)/n$ convolutional encoder, it is clear that the algorithm performances are enhanced by iterations. Moreover, it is important to note that the detection of a convolutional code depends on both the parameters of the code, the channel error probability, and the correction capacity of the code. Thus, the number of iterations needed to get the best performance is code dependent. For such a code, it would be worth assessing the impact of the required number of data. In order to achieve this, for the $C(2,1, 7)$ convolutional encoders, a comparison of the detection gain produced by the iterative process for several values of L is proposed.

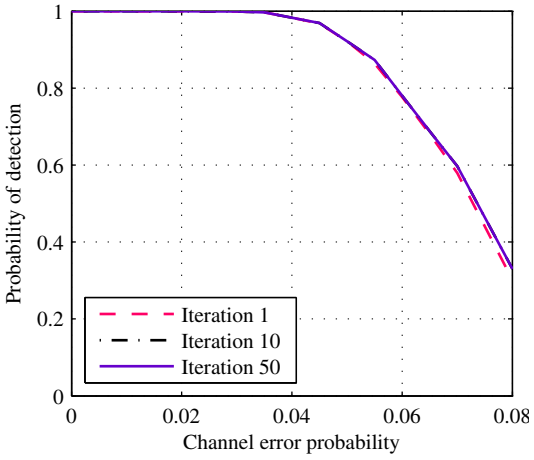

Figure 3 $C(3,1,4)$: Probability of detection compared with P_e. For the $C(3,1,4)$ encoder, the probability of detecting the true encoder is depicted compared with the channel error probability for 1, 10, and 50 iterations.

- $C(2,1,7)$ convolutional encoder:

Figure 4 depicts P_{det} compared with P_e, for 1, 5, and 50 iterations and for $L = 200$. For 1, 10, 40, and 50

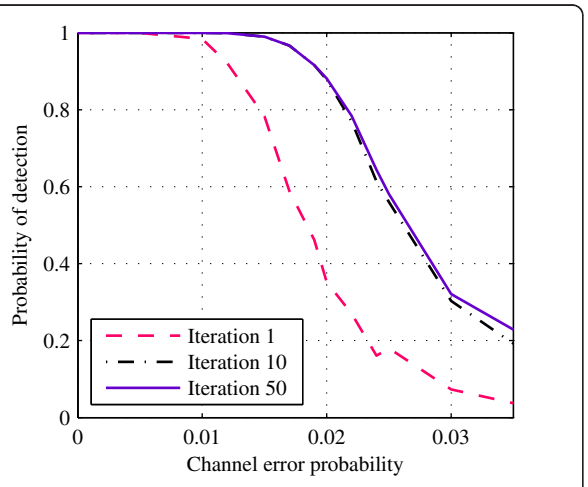

Figure 2 $C(3,2,3)$: Probability of detection compared with P_e. For the $C(3,2,3)$ encoder, the probability of detecting the true encoder is depicted compared with the channel error probability for 1, 10, and 50 iterations.

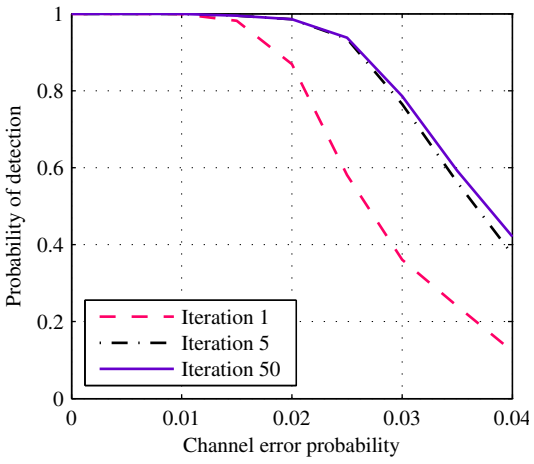

Figure 4 $C(2,1,7)$: Probability of detection compared with P_e for $L = 200$. For the $C(2,1,7)$ encoder and $L = 200$, the probability of detecting the true encoder is depicted compared with the channel error probability for 1, 5, and 50 iterations.

iterations, Figure 5 illustrates the evolution of P_{det} compared with P_e for $L = 500$. It shows that, for $L = 200$, 5 iterations permit us to identify the true encoder, whereas, for $L = 500$, the identification of the true encoder requires 40 iterations. For $L = 200$, after 5 iterations, P_{det} is close to 1 for $P_e \leq 0.02$, but after 40 iterations and $L = 500$, P_{det} is close to 1 for $P_e \leq 0.03$. It is clear that the number of received bits is an important parameter of our method. Indeed, by increasing the size of matrices R_l, the probability to obtain non-erroneous pivots increases during the iterative process. Thus, it is possible to realize more iterations of our algorithm to improve detection performances. But, for implementation in a realistic context, the required number of data has to be taken into account. In the last section, we will show that the algorithm performances are very good when $L = 200$.

4.2 Probability of detection

To analyze the method performances, three probabilities were defined as follows:

1. probability of detection (P_{det}) is the probability of identifying the true encoder;
2. probability of false-alarm (P_{fa}) is the probability of identifying an optimal encoder but not the true one;
3. probability of miss (P_{m}) is the probability of identifying no optimal encoder.

In order to assess the relevance of our results through a comparison of the different probabilities to the code correction capability, let us denote by BER_r the theoretical residual bit error rate obtained after decoding of the corrupted data stream with a hard decision [12]. Here, to be acceptable, BER_r must be close to 10^{-5}.

Figures 6, 7, and 8 show the different probabilities compared with P_e after 10 iterations and the limit of the 10^{-5} acceptable BER_r for $C(3, 2, 3)$, $C(3, 1, 4)$, and $C(2, 1, 7)$ convolutional encoders, respectively. One should note that the probability of identifying the true encoder is close to 1 for any P_e with a post-decoding BER_r less than 10^{-5}. Indeed, the algorithm performances are excellent: P_{det} is close to 1 when P_e corresponds to either $\text{BER}_r < 2 \times 10^{-4}$ for $C(3,2,3)$ convolutional encoder or $\text{BER}_r < 0.67 \times 10^{-4}$ for the C(3,1,4) encoder.

5 Conclusion

This article dealt with the development of a new algorithm dedicated to the reconstruction of convolutional code from received noisy data streams. The iterative method is based on algebraic properties of both optimal convolutional encoders and their dual code. This algorithm allows the identification of parameters and generator matrix of a rate k/n convolutional encoder. The performances were analyzed and proved to be very good. Indeed, the probability to detect the true encoder proved to be close to 1 for a channel error probability that generates a post-decoding BER_r that is less than 10^{-5}. Moreover, this algorithm requires a very small amount of received bit stream.

In most digital communication systems, a simple technique, called puncturing, is used to increase the code rate. The blind identification of the punctured code is divided into two part: (i) identification of the equivalent encoder and (ii) identification of the mother code and

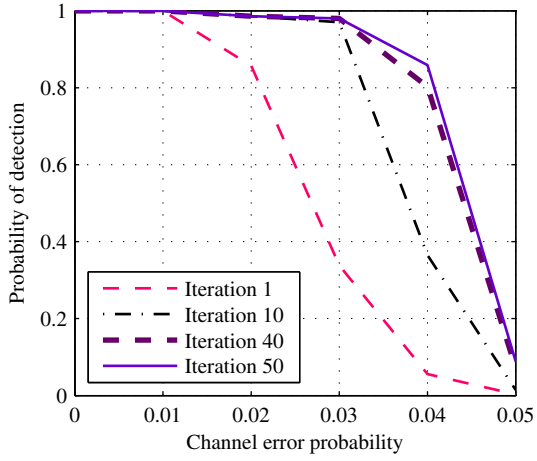

Figure 5 $C(2,1,7)$: Probability of detection compared with P_e for $L = 500$. For the $C(2,1,7)$ encoder and $L = 500$, the probability of detecting the true encoder is depicted compared with the channel error probability for 1, 10, 40, and 50 iterations.

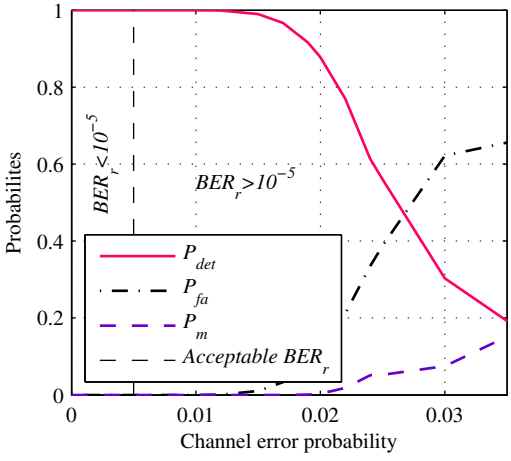

Figure 6 $C(3,2,3)$: Probability of detection, probability of false-alarm, and probability of miss compared with P_e. For the $C(3, 2, 3)$, the probability of detection, the probability of false-alarm, and the probability of miss are depicted compared with he channel error probability.

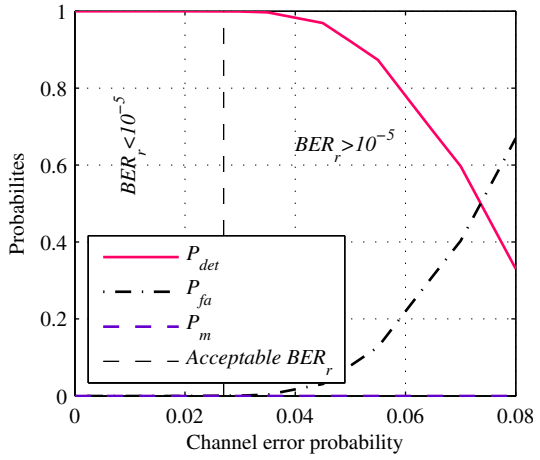

Figure 7 C(3,1,4): Probability of detection, probability of false-alarm, and probability of miss compared with P_e. For the C(3, 1, 4), the probability of detection, the probability of false-alarm, and the probability of miss are depicted compared with he channel error probability.

puncturing pattern. Our method, dedicated to the blind identification of k/n convolutional encoders, also allows the blind identification of the equivalent encoder of the punctured code. Thus, our future study will be to identify the mother code and the puncturing pattern only from the knowledge of this equivalent encoder.

A The key-parameter n_a

According to (20), the rank of the matrix, $R_{\alpha.n}$, is:

$$\text{rank}\,(R_{\alpha.n}) = \alpha.n.\frac{k}{n} + \mu^{\perp} < \alpha.n \tag{31}$$

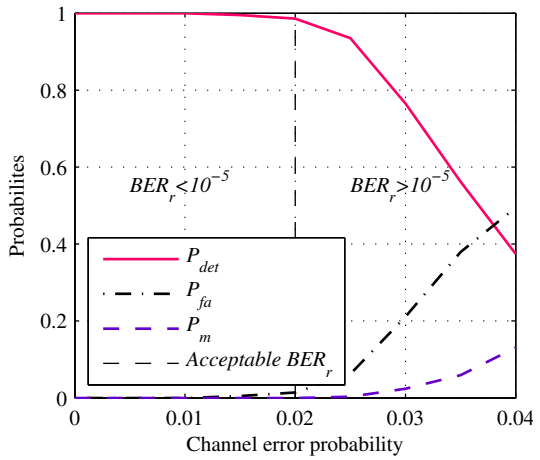

Figure 8 C(2,1,7): Probability of detection, probability of false-alarm and, probability of miss compared with P_e. For the C(2, 1, 7), the probability of detection, the probability of false-alarm, and the probability of miss are depicted compared with he channel error probability.

Let us seek n_a, when $n_a = \alpha.n$, which corresponds to the first matrix, R_{n_a}, with a rank deficiency. This corresponds to seeking the minimum value of α.

$$\alpha.n\left(1 - \frac{k}{n}\right) > \mu^{\perp} \tag{32}$$

$$\alpha.n > \frac{n}{n-k}.\mu^{\perp} \tag{33}$$

$$\alpha > \frac{\mu^{\perp}}{n-k} \tag{34}$$

So, the minimum value of α, denoted α_{min}, is such that

$$\alpha_{\min} = \left\lfloor \frac{\mu^{\perp}}{n-k} \right\rfloor + 1 \tag{35}$$

According to (35), the key-parameter n_a is such that

$$n_a = n.\alpha_{\min} = n.\left\lfloor \frac{\mu^{\perp}}{n-k} + 1 \right\rfloor \tag{36}$$

B The rank deficiency of R_{n_a}

According to (36), the rank of R_{n_a} is such that

$$\text{rank}\left(R_{n_a}\right) = k.\left\lfloor \frac{\mu^{\perp}}{n-k} + 1 \right\rfloor + \mu^{\perp} \tag{37}$$

Therefore, the rank deficiency of R_{n_a}, denoted $Z(n_a) = n_a - \text{rank}\left(R_{n_a}\right)$, is

$$\begin{aligned} Z(n_a) &= (n-k).\left\lfloor \frac{\mu^{\perp}}{n-k} + 1 \right\rfloor - \mu^{\perp} \\ &= (n-k).\left\lfloor \frac{\mu^{\perp}}{n-k} \right\rfloor - \mu^{\perp} + (n-k) \end{aligned} \tag{38}$$

The modulo operator is equivalent to

$$(a \mod (b)) = a - \left\lfloor \frac{a}{b} \right\rfloor.b \tag{39}$$

and thus:

$$Z(n_a) = -\left(\mu^{\perp} \mod (n-k)\right) + (n-k) \tag{40}$$

The modulo operator is such that

$$0 \le (a \mod (b)) < b \tag{41}$$

Consequently, the value of ($\mu^{\perp}$- mod (n - k)) is

$$0 \le \left(\mu^{\perp} \mod (n-k)\right) < (n-k) \tag{42}$$

$$-(n-k) < -\left(\mu^{\perp} \mod (n-k)\right) \le 0 \tag{43}$$

$$0 < (n-k) - \left(\mu^{\perp} \mod (n-k)\right) \leq (n-k) \tag{44}$$

So, $Z(n_a)$ is such that

$$0 < Z(n_a) \leq (n-k) \tag{45}$$

where $Z(n_a) \in \mathbb{N}$. Therefore, the rank deficiency of the matrix, R_{n_a}, is such that

$$1 \leq Z(n_a) \leq (n-k) \tag{46}$$

Acknowledgements
This study was supported by the Brittany Region (France).

Author details
[1]Université Européenne de Bretagne, Rennes, France [2]Université de Brest; CNRS, UMR 3192 Lab-STICC, ISSTB, 6 avenue Victor Le Gorgeu, CS 93837, 29238 Brest cedex 3, France

Competing interests
The authors declare that they have no competing interests.

Received: 22 April 2011 Accepted: 14 November 2011
Published: 14 November 2011

References

1. B Rice, Determining the parameters of a rate 1/*n* convolutional encoder over gf(q), in *Proceedings of the 3rd International Conference on Finite Fields and Applications*, Glasgow (1995)
2. E Filiol, Reconstruction of convolutional encoders over GF(p), in *Proceedings of the 6th IMA Conference on Cryptography and Coding*, vol. 1355. (Springer Verlag, 1997) pp. 100–110
3. J Barbier, Reconstruction of turbo-code encoders, in *Proc SPIE Security and Defense Space Communication Technologies Symposium*, vol. 5819. (Orlando, FL, USA, 2005) pp. 463–473
4. M Marazin, R Gautier, G Burel, Blind recovery of the second convolutional encoder of a turbo-code when its systematic outputs are punctured. MTA Rev. **XIX**(2), 213–232 (2009)
5. J Barbier, G Sicot, S Houcke, Algebraic approach for the reconstruction of linear and convolutional error correcting codes. Int J Appl Math Comput Sci. **2**(3), 113–118 (2006)
6. M Côte, N Sendrier, Reconstruction of convolutional codes from noisy observation, in, in *Proceedings of the IEEE International Symposium on Information Theory ISIT 09*, Seoul, Korea, pp. 546–550 (2009)
7. A Valembois, Detection and recognition of a binary linear code. Discr Appl Math. **111**(1-2), 199–218 (2001). doi:10.1016/S0166-218X(00)00353-X
8. M Marazin, R Gautier, G Burel, Dual code method for blind identification of convolutional encoder for cognitive radio receiver design, in *Proceedings of the 5th IEEE Broadband Wireless Access Workshop, IEEE GLOBECOM 2009*, Honolulu, Hawaii, USA, (2009)
9. GD Forney, Convolutional codes I: algebraic structure. IEEE Trans Inf Theory. **16**(6), 720–738 (1970). doi:10.1109/TIT.1970.1054541
10. R McEliece, The algebraic theory of convolutional codes, in *Handbook of Coding Theory*, vol. 2. (Elsevier Science, 1998), pp. 1065–1138
11. GD Forney, Structural analysis of convolutional codes via dual codes. IEEE Trans Inf Theory **19**(4), 512–518 (1973). doi:10.1109/TIT.1973.1055030
12. R Johannesson, KS Zigangirov, *Fundamentals of Convolutional Coding*. IEEE Series on Digital and Mobile Communication (IEE Press, 1999)
13. F Wang, Z Huang, Y Zhou, A method for blind recognition of convolution code based on euclidean algorithm, in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, 1414–1417 (2007)
14. J Dingel, J Hagenauer, Parameter estimation of a convolutional encoder from noisy observations, in *Proceedings of the IEEE International Symposium on Information Theory, ISIT 07* Nice, France, pp. 1776–1780 (2007)
15. G Sicot, S Houcke, Blind detection of interleaver parameters, in *Proceedings of the ICASSP*, pp. 829–832 (2005)
16. G Sicot, S Houcke, Theoretical study of the performance of a blind interleaver estimator, in *Proceedings of the ISIVC*, Hammamet, Tunisia, (2006)
17. G Sicot, S Houcke, J Barbier, Blind detection of interleaver parameters. Elsevier Signal Process. **89**(4), 450–462 (2009)

doi:10.1186/1687-1499-2011-168
Cite this article as: Marazin *et al.*: Blind recovery of *k/n* rate convolutional encoders in a noisy environment. *EURASIP Journal on Wireless Communications and Networking* 2011 **2011**:168.