BLIND DETECTION OF INTERLEAVER PARAMETERS

Guillaume Sicot and Sbastien Houcke

Dept Signal and Communication. ENST-Bretagne
TAMCIC (CNRS 2658) email: {guillaume.sicot} {sebastien.houcke}@enst-bretagne.fr

ABSTRACT

Interleaving is a key component of many digital communication systems involving error correction schemes. It provides a form of time diversity to guard against bursts of errors. Recently, interleavers have become an even more integral part of the code design itself, if we consider for example turbo and turbo-like codes. In a non-cooperative context, such as passive listening, it is a challenging problem to estimate the interleaver parameters. In this paper we propose an algorithm that allows us to estimate the parameters of the interleaver at the output of a binary symmetric channel and to locate the codewords in the interleaved block. This gives us some clues about the interleaving function used.

1. INTRODUCTION AND NOTATION

Error correcting codes are usually good at correcting randomly distributed errors but generally offer inferior performance when the errors occur in bursts. For bursty errors, interleaving the coded sequence is commonly used [1]. It distributes data bits in a different order from the one in which they are generated.

On the receiver side, the signal is demodulated, frame synchronized and deinterleaved. After those operations the receiver is able to decode the coded sequence and to correct the transmission errors.

In a non-cooperative context, we need to blindly estimate the different parameters of the interleaver and the coding scheme in order to perform the reverse operations. In such a context, the intercepted sequence may be severely corrupted (high bit error rate). In particular we are not able at that point to take advantage of the coded gain. Therefore we clearly understand the importance of developing a method that is robust with respect to high bit error rate. The method proposed in [2] deals with the case of perfect transmission. [2] provides an interesting way to estimate the size of the interleaver block, the code rate and realize the blind synchronization of the interleaver blocks. This detection is based on linear algebra theory. The method is developed in the case of perfect transmission (no transmission errors). In this paper we develop an algorithm, based on the same concept as in [2], that blindly estimates the characteristics of the interleaver from a **block coded** and **interleaved** binary sequence corrupted by a high bit error rate.

In the same way as in [2], we are able to identify the interleaver size, to synchronize the interleaver blocks and to estimate the code rate. Furthermore our algorithm allows us to estimate the position of the codewords in the interleaved block.

The paper is organized as follows. In section 2, we recall the principle of the method developed in [2]. Our method is exposed in section 3. Finally section 4 gives simulation results that show the excellent performance of our method.

1.1. Notation

A block encoder is defined by a full-rank generator matrix G that transforms each block of k_c information bits into n_c encoded bits ($k_c < n_c$). Representing the i^{th} information block and the i^{th} encoded block by vectors b_i and y_i, we have: $y_i = b_i G$. y_i is called a codeword. The ratio $r = k_c/n_c$ is called the code rate. The interleaver can be modeled by a permutation matrix P of size SS where S is called the interleaver size. This means that the interleaver performs a permutation within each block of S encoded bits. In almost all systems, the interleaver size is a multiple of the size of the codeword and we have: $S = Nn_c$ with N being the number of codewords within the interleaved block. The transmitted sequence $\mathbf{X}$ is composed of M interleaved blocks. Let us denote by $\mathbf{Z}$ the intercepted sequence of $\mathbf{X}$. $\mathbf{Z}$ is a delayed replica of $\mathbf{X}$ (by t_0 bits) that has been passed through a binary symmetric channel. Let us denote P_e the error probability of the channel. Without loss of generality we assume that the restitution delay t_0 is smaller than the size S of the interleaver.

2. LINEAR ALGEBRA TO ESTIMATE INTERLEAVER CHARACTERISTICS

In this section, we assume that the channel introduces no error (*i.e.* $P_e = 0$). Burel *et al.* [2] propose to build a matrix $H(n_a, d)$ by skipping the first d bits of $\mathbf{Z}$ and then dividing the remaining interleaved stream into M analysis blocks of an arbitrary size n_a ($M \geq n_a$). Those blocks form the lines of $H(n_a, d)$. They examine the behavior of the ratio $\rho(n_a, d)$ defined as :

$$\rho(n_a, d) = \frac{\text{rank}(H(n_a, d))}{n_a} \qquad (1)$$

for different values of n_a and d. They noticed that $\forall d$, $\rho(n_a, d)$ is equal to 1, except when n_a is a multiple of the interleaver size. Indeed, in that particular case, some columns are linear combinations of each other. As illustrated in figure 1, this property is due to the redundancy introduced by the code. Let us consider a redundant bit represented by the shaded box in figure 1. This bit is a linear combination of other bits located in the same block. If $n_a = S$, this relation is also satisfied for the next line and thus for the whole column. Thus the shaded column is a linear combination of other columns. This is not the case whenever $n_a \neq \alpha S$.

This rank deficiency property of $H(S, d)$ allows

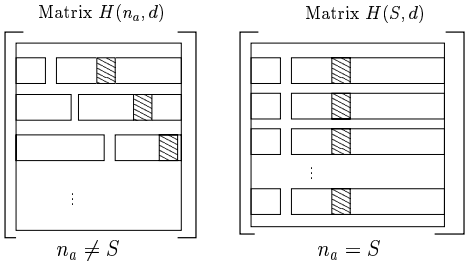


Fig. 1. Matrix $H(n_a, d)$

the authors to estimate S. Once the size of the interleaver is estimated, the minimum of $\rho(S, d)$ with respect to d allows them to estimate t_0. This can be explained by the fact that the maximum of dependent columns in H is obtained for $d = t_0$. An estimation of the code rate is given by the ratio between the number of dependent columns and the estimated interleaver size.

As shown in [2], this method gives good results for estimating the parameters of the interleaver. However in a passive listening context, the intercepted sequence may be highly corrupted, which transforms the $H(S, d)$ matrix into a full rank matrix and thus the previously algorithm can not be used.

In the following we propose an algorithm based on similar concepts. It allows us to estimate those parameters when the interleaved sequence is delayed and passed through a **binary symmetric channel**. Furthermore, our method is able to locate the position of the bits belonging to the same codeword in the interleaved block.

3. THE PROPOSED ALGORITHM

As in [2], the matrix $H(n_a, d)$ is built from $\mathbf{Z}$. The basic idea of the method is to find "almost dependent columns" in $H(n_a, d)$. This is realized using an adaptation of the well known Gauss Jordan Elimination Through Pivoting algorithm [3]. We now briefly remind this method adapted to the binary case. The goal of this algorithm is to convert $H(n_a, d)$ into a lower triangular matrix noted $L(n_a, d)$. At the end of the algorithm, the presence of all-zero columns in $L(n_a, d)$ indicates a rank deficiency of $H(n_a, d)$.

We start with the first column of matrix H, and set $i = 1$.

1. If the i^{th} element of the i^{th} column is a zero, we permute this column with the first column i' ($i' > i$) that has a one on its i^{th} element.
2. If there is no column that has a one on its i^{th} element, we permute the i^{th} row with the first row i' ($i' > i$) that has a one on its i^{th} element.
3. We add (modulo 2) this column to any column on its left that has a one on its i^{th} row. This clears the i^{th} row.
4. If we get a column with zero, it is a dependent column.
5. We repeat the procedure with the next column (*i.e.* set $i = i + 1$ and go back to 1.)

Thus, the Gauss Jordan Elimination Through Pivoting is a linear application that can be written as:

$$A_1 H(n_a, d) A_2 = L(n_a, d). \qquad (2)$$

Since $H(n_a, d)$ and $L(n_a, d)$ are binary matrices, A_1 and A_2 take their values in the binary field $\mathbb{F}_2$. Matrix A_1 reports all rows permutations performed during the algorithm whereas matrix A_2 reports all column permutations and additions performed on $A_1 H(n_a, d)$ to obtain $L(n_a, d)$.

3.1. Identification of the interleaver size and blind synchronization

$H(n_a, d)$ can be modeled as $H(n_a, d) = \tilde{H}(n_a, d) + E$ where $\tilde{H}(n_a, d)$ is the error free matrix and E contains all transmission errors.

For $n_a = \alpha S$, there exists many linear combinaisons of columns in $\tilde{H}(n_a, d)$ and for each one, we define a set of column positions

$$\mathfrak{I}_j^{(n_a,d)} = \{i_1^{(j)}, \ldots, i_{p_j}^{(j)}\}$$

such that:

$$C_{i_1^{(j)}}^{\tilde{H}(n_a,d)} + \ldots + C_{i_{p_j}^{(j)}}^{\tilde{H}(n_a,d)} = 0. \qquad (3)$$

Where C_i^H is the column i of the matrix H. Let us also define $\mathcal{D}_{n_a,d} = \{\mathfrak{I}_1^{(n_a,d)}, \ldots, \mathfrak{I}_{Q(n_a,d)}^{(n_a,d)}\}$ a basis of all sets $\mathfrak{I}_j^{(n_a,d)}$. Its cardinal $Q(\alpha S, d)$ is non zero and its maximum is reached at $d = t_0$.

For $n_a \neq \alpha S$, $\alpha \in \mathbb{N}$, the columns of $\tilde{H}(n_a, d)$ are all independent (see figure 1). Hence the cardinal $Q(n_a, d)$ of $\mathcal{D}_{n_a,d}$ is zero. Thus, the maximisation of $Q(n_a, d)$ w.r.t (n_a, d) allows us to estimate S and t_0. In the following, we present a way to estimate the cardinal of $\mathcal{D}_{n_a,d}$ from $H(n_a, d)$. This is done by using the matrix $L(n_a, d)$.

First of all, let B_i be the number of ones in the lower part[1] of column i in the matrix $L(n_a, d)$. Note that the sum modulo 2 of independent columns gives an independent column. Thus for an independent column i, B_i is Binomial distributed its mean is: $m_B = \frac{1}{2}\left(\left\lfloor \frac{MS}{n_a} \right\rfloor - n_a\right)$.

For a fixed number of analyzed bits (*i.e.* M is fixed), m_B decreases when n_a increases. Let us define $\phi(k)$ as:

$$\phi(k) = \frac{B_k}{m_B} \qquad (4)$$

For $n_a \neq \alpha S$, it is easily shown that:

$$\forall k \in \{1, \ldots, n_a\}, \lim_{M\to\infty} \phi(k) \xrightarrow{\mathcal{P}} 1.$$

with $\xrightarrow{\mathcal{P}}$ meaning the convergence in probability.

Now for $n_a = \alpha S$ with $\alpha \in \mathbb{N}$, assuming that there is no error on and over the main diagonal of $A_1 H(n_a, d)$, there is, for each element $\mathfrak{I}_j^{(n_a,d)}$ of $\mathcal{D}_{\alpha S,d}$, one column of $L(\alpha S, d)$ at position $k_j \in \mathfrak{I}_j^{(\alpha S,d)}$ such that :

$$\lim_{M\to\infty} \phi(k_j) \xrightarrow{\mathcal{P}} 2 - 2\sum_{i=0}^{\lfloor \frac{p_j}{2} \rfloor} \binom{p_j}{2i} P_e^{2i}(1-P_e)^{p_j-2i} \qquad (5)$$

with P_e the probability of error of the binary symmetric channel and p_j the cardinal of $\mathfrak{I}_j^{(\alpha S,d)}$. And for the other columns $k \in \mathfrak{I}_j^{(\alpha S,d)}$, $k \neq k_j$, we have:

$$\lim_{M\to\infty} \phi(k) \xrightarrow{\mathcal{P}} 1$$

Even with a finite M, the gap between these two behaviors of $\phi(.)$ is significant as long as $\{p_{k_j}\}_j$ and P_e are not too large. The existence of this gap allows us to estimate $Q(n_a, d)$:

$$\hat{Q}(n_a, d) = \text{Card}(\{k \in \{1, \ldots, n_a\} / \phi(k) < \beta\})$$

with β a well defined threshold. Simulations show that it is quite easy to dissociate the two populations and the choice of the threshold does not have a great influence on the performance of the algorithm.

Furthermore an estimation of the code rate is obtained by $Q(\hat{S}, \hat{t_0})/\hat{S}$. As it will be explained in the next section, we have another possibility to check if we have correctly estimated the block synchronization and code rate.

3.2. Position of the codewords within the interleaved sequence.

In order to locate the bits belonging to the same codeword on the interleaver block, we need to estimate $\mathcal{D}_{S,t_0}$. For each column k of matrix $L(S, t_0)$ satisfying $\phi(k) < \beta$, we estimate one $\mathfrak{I}_j^{(S,t_0)}$ using matrix A_2 (see (2)). Indeed A_2 represents transformations that are performed on columns of $H(S, t_0)$ and we are able to identify the columns $i_1^{(k)}, \ldots, i_{p_k}^{(k)}$ of $H(S, t_0)$ such that :

$$C_{i_1^{(k)}}^{H(S,t_0)} + \ldots + C_{i_{p_k}^{(k)}}^{H(S,t_0)} = C_k^{L(S,t_0)}$$

In other words, $\hat{\mathfrak{I}}_j^{(S,t_0)} = \{i_1^{(k)}, \ldots, i_{p_k}^{(k)}\}$ is an estimator of one element of $\mathcal{D}_{S,t_0}$. It means that in one interleaved block, bits at positions $\{i_1^{(k)}, \ldots, i_{p_k}^{(k)}\}$ are linearly dependent and therefore belong to the same codeword.

Note that if the synchronization was wrong, we would not be able to find to which codeword belongs the first or last bit of the block. This algorithm may also be used to perform the blind frame synchronization.

3.3. Algorithm improvement

The estimation $\hat{\mathcal{D}}_{n_a,d}$ of the basis $\mathcal{D}_{n_a,d}$ may not be completed: we could miss some basis vectors. However, another realization of $H(n_a, d)$, may be used to find the missing basis vectors. In order to obtain this new matrix we simply permute the rows of $H(n_a, d)$, yielding a "virtual new realization"

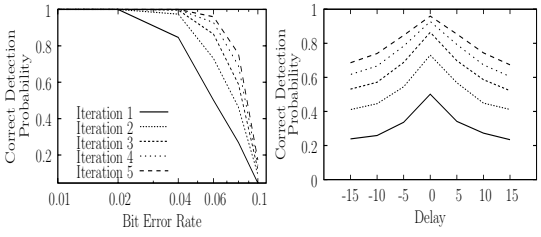
[1] from row n_a+1 to the last row $\lfloor \frac{MS}{n_a} \rfloor$ of $H(n_a, d)$, where M is the number interleaver blocks in $\mathbf{X}$.

of the transmission. By doing so, we can complete our basis and build, iteration after iteration, the whole map of the codewords in the interleaved block. This procedure can also be applied to the detection of the size of the interleaver. This allows us to significantly improve performance of our algorithm as shown in the next section.

4. SIMULATION RESULTS

Let us consider the $(7, 4)$ Hamming block code and a random interleaver of size $S = 56$. The restitution delay t_0 over the binary symmetric channel is set to 0. Matrix $H(n_a, d)$ is built from 50000 intercepted bits. For each simulation, 3000 Monte Carlo Trials are run. The threshold β is fixed to 0.6. For each trial, the coded sequence and the interleaver are randomly chosen.

Figure 2(a) presents the correct detection probability of the interleaver size versus the BER, assuming that t_0 is known (*i.e.* $d = t_0$). Figure 2(b) presents the probability of correct detection of the size of the interleaver obtained for a BER of 6% and for different values of d. Each iteration corresponds to a new virtual realization of the intercepted sequence $\mathbf{Z}$ (as explained in section 3.3). We note that performance increases significantly with the number of iterations. For this code and $\beta = 0.6$, using (5), we can prove that our algorithm fails asymptotically for $P_e \approx 0.1024$. Figure 2(a)) shows clearly this limit.

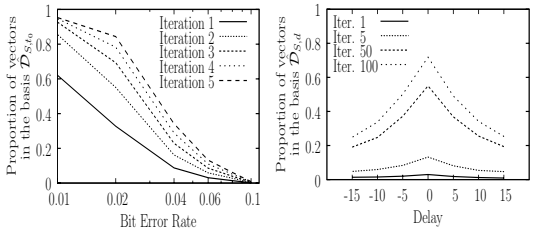


(a) versus BER (b) versus d - BER=6%

Fig. 2. Correct detection probability of S

We also note that the performance decreases when $|d|$ increases, but as we analyze the sequence off line, it is possible to estimate the size of the interleaver for different values of the offset d. However this increases the computational complexity.

We now illustrate the ability of our algorithm to estimate the position of codewords in the interleaved block. This ability is closely related to the number of vectors found in the estimate basis $\hat{\mathcal{D}}_{S,d}$. The more vectors we have in the basis, the more codewords our algorithm is able to locate in the interleaved block. Figure 3(a) shows the proportion of vectors found in the basis versus the BER assuming $n_a = \alpha S$ and $d = t_0$ for 1 to 5 iterations. For example, at BER of 2%, after 5 iterations, we are able to find 84% of the basis $\hat{\mathcal{D}}_{S,d}$. Figure 3(b) shows the proportion of vectors in the



(a) versus BER (b) versus d - BER=6%

Fig. 3. Proportion of vectors found in $\hat{\mathcal{D}}_{S,d}$

basis $\hat{\mathcal{D}}_{S,d}$ versus d after 1, 5, 50 and 100 iterations. We can see that we obtain a maximum for $d = t_0$, which allows us to estimate the delay t_0. Morever we can note again the benefit given by the iterations. As shown by figure 3(a) the benefit is much larger for the first 50 iterations than for the next 50 iterations. Indeed with a BER of 6%, and for $d = 0$, we have found 55% after 50 iterations, 72% after 100 iterations.

5. CONCLUSION

We have presented an algorithm based on linear algebra properties which, from a delayed and corrupted interleaved sequence of block coded bits, allows us to blindly estimate the interleaver size, to synchronize the interleaver blocks, to estimate the code rate and to obtain a precise idea of the kind of interleaver used.

This method exhibits excellent performance: at BER of 8%, we are able to correctly estimate the interleaver size in 76% of cases.

6. REFERENCES

[1] J.G. Proakis, *Digital Communications*, McGraw-Hill, 1995.

[2] G. Burel and R. Gautier, "Blind estimation of encoder and interleaver characteristics in a non cooperative context," *IASTED - CIIT*, November 2003.

[3] G.H. Golub and C.F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, 1989.