

Non-Binary Turbo Codes Interleavers

Maria KOVACI, Horia BALTA

University "Polytechnic" of Timișoara, Faculty of Electronics and Telecommunications,
Postal Address, 30223 Timișoara, ROMANIA, E-Mail: maria.kovaci@etc.utt.ro,
horia.balta@etc.utt.ro

Abstract: In this paper the particularities of the interleavers used in the case of the Recursive Systematic Convolutional Code (RSC), $n/(n+2)$ rate, with $n=2$, unpunctured non-binary turbo codes (NBTC) were described. The interleavers: block, pseudo-random and new interleaving methods are presented. These new interleaving methods were elaborated starting from the particular form of the date accepted by non-binary turbo codes and they constitute an extension of the classic methods. The AWGN (Additive White Gaussian Noise) and the BPSK (Binary Phase Shift Keying) modulation were employed. The turbo decoder was used 15 iterations for each block. A Log Likelihood Ratio (LLR) stop criterion was selected.

The BER (Bit Error Rate) and FER (Frame Error Rate) performances for small interleaver lengths ($N=448$) and for high interleaver lengths (more than 1000 bits) were obtained. In the case of FER performances, for each interleaving length, the Shannon's limits were computed, too.

Keywords: - non-binary turbo codes, interleaver

I. Introduction

The turbo codes, TC, are a class of error correcting codes that operates on approaching of the Shannon limit. The scheme of the TC with parallel configuration is shown on Fig. 1, [1]. The information sequence, noted u , is coded by the coder $C1$, resulting the parity sequence x_1 . The same bits sequence, u , is provided to the coder $C2$, but in a different order with the aid of the one interleaver, I . The coder $C2$ generates de parity sequence, x_2 . The resulted sequences $x_0=u$, x_1 and x_2 , by multiplexing and modulation (operations omitted on Fig.1) constitute the TC output, signal which will be emitted on channel. At the output of the channel, by demodulation and demultiplexing, it results the corresponding received (soft) sequences, y_0 , y_1 and y_2 .

Each decoder computes the Log Likelihood Ratio, LLR, for each bit from u :

$$LLR(u_i) = \ln \frac{p(u_i = 1|y)}{p(u_i = 0|y)}, \quad (1)$$

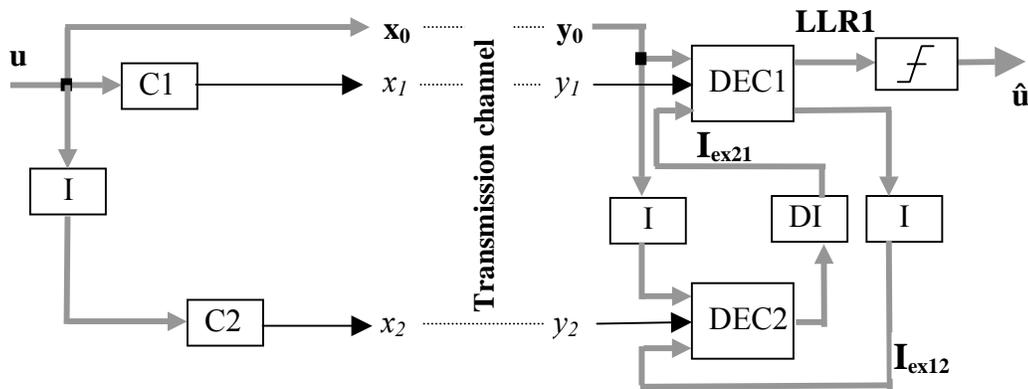


Fig. 1. Turbo code - general scheme.

Only the LLR of the first decoder, noted $LLR1$ and also the extrinsic information for the second decoder are shown on Fig. 1.

Each decoder receives extrinsic information. With the aid of this extrinsic information and of the received sequences from channel (y_0 and y_1 , respectively y_0 interleaved and y_2) each decoder provides, on his turn, extrinsic information. This process is repeated iteratively with a certain number (given or computed, varying with turbo code type). After the effectuation of the all imposed iterations, a hard decision of the LLR, generated on the last iteration by one of the two decoders (on Fig.1 is choosed LLR1), is done. The resulted sequence, from the hard decision operation, constitutes the TC output.

If, the component codes ($C1$ and $C2$) have r inputs, then TC is named non-binary turbo code NBTC, [2] or multi –binary turbo code, MBTC. NBTC works identically like a TC classic, with the difference that all the processing which concern the information sequence will be made on r bits. This fact is marked on Fig. 1 by thickened lines, which symbolize thoroughfares of r bits. In the same way, the attached vectors become matrixes with r lines. The labels of these matrixes are written with no-italic and bold form, for example: \mathbf{x}_0 .

Therefore, the interleavers for NBTC must be capable to operate with matrix and not with vectors. In this paper, a study about the construction problems of such interleavers is presented. Also, the BER performances obtained with these interleavers are shown on the paragraph 4. The interleavers construction for NBTCs, presented on the third paragraph, is made from the interleavers types, already known for TC classics, which are succinct presented on the following paragraph. In the last paragraph are presented the results obtained by simulation of the NBTC with the proposed interleavers.

II. The classical interleavers

A. Random interleaver

The random interleaver has a simple design, [3], that provides a good original sequence spreading, but it generally has the minimum distance $d_{min} = 2$, i.e. the smallest possible value. The construction of a random interleaver is the following. Knowing the interleaving length N_r , the ensemble $A = \{1, 2 \dots N_r\}$ is built. We choose in a randomly way a number $n_1 \in A$. Then, the allocation $\pi(1) = n_1$ is made and this value (n_1) is eliminated from A . The process is repeated while A has no elements. So, the following random interleaver function:

$$\pi_r(i) = rand(i), \quad \forall i \in I = \{1, 2 \dots N_r\}, \quad (2)$$

is used.

A main disadvantage for the random interleaver is the irreproducibility of the generation process of mapping π_r : it should be memorized for reproduction, after its construction.

B. The S-interleaver

The S-interleaver is a random type interleaver, [3]. However, unlike the pure random interleaver, by construction is forced a minimum interleaving distance equal with S . The interleaving mapping construction algorithm is the following. We select a possible future position for the current bit. This position is compared with the last S positions already selected. If the condition:

$$|\pi_s(i) - \pi_s(j)| > S \text{ for } i \text{ and } j \text{ with } |i - j| < S, \quad (3)$$

is satisfied we go further. If the condition is not true, we select another position for current bit, which will be also tested. The process will be repeated up to the moment when all the positions

for the N_s bits were found. The simulations demonstrated that, if $S < \sqrt{N_s/2}$, then the process will converge. The design of this interleaver is difficult because of the difficulty accomplishment of the condition increases with the increasing of the number of bits already tested.

C. The block interleaver

The block (rectangular) interleaver presents the simplest structure, [3]. To obtain a block interleaver function is necessary to factorize its length.

$$N_b = X \times Y, \quad (4)$$

where X si Y are closed natural numbers, [2]. The block interleaver function is:

$$\begin{aligned} \pi_b(i+j*X+1) &= i*Y+j+1, \\ \forall i \in I = \{0, 1 \dots X-1\} \text{ si } \forall j \in J = \{0, 1 \dots Y-1\}. \end{aligned} \quad (5)$$

Any two bits, initially situated at a distance less then $d_{min} = \min(I, J)$, will be situated, after interleaving, at a distance superior to d_{min} .

D. The pseudo-random interleaver

The pseudo-random interleaver is recommended by ‘‘CCSDS Recommendation for Telemetry Channel Coding’’, [4]. For this interleaver type there is a controlled spreading.

It is a high performance interleaver, which combines the advantages of the random and block interleavers, i.e. it presents a good spreading at enough large minimum distance.

The recommended permutation for each block of length N_p is given by a particular reordering of integers: $1, 2, \dots N_p$. It is generated by the following algorithm.

- $N_p = k_1 \cdot k_2$ where k_1 is a fixed parameter, and k_2 varies in function of interleaver length;
- next do the following operations for $s=1$ to $s= N_p$ (the current position before interleaving) to obtain the permutation numbers $\pi_p(s)$:

- $m = (s - 1) \bmod 2$
- $i = [(s-1)/(2k_2)]$
- $j = [(s-1)/2] - i \cdot k_2$
- $t = (19 \cdot i + 1) \bmod (k_1/2)$
- $q = t \bmod 8 + 1$
- $c = (p_q \cdot j + 21 \cdot m) \bmod k_2$
- $\pi_p(s) = 2 \cdot (t + (N_p + 1)/2 + 1) \cdot s - m$

where $[x]$ = the largest integer less than or equal to x , ‘‘mod’’= the operation modulo and $q=1 \div 8$, $p_1=31, p_2=37, p_3=43, p_4=47, p_5=53, p_6=59, p_7=61, p_8=67$, [4].

E. The Takeshita-Costello interleaver

The Takeshita-Costello interleaver is presented in [5], and it suppose that the interleaver length, N_{TC} , to be equal with 2^j (j is a natural number). The construction of the interleaver function is the following. The vector $c_{i=1 \div N}$, is built with the relation :

$$c(i) = k \cdot i \cdot (i+1) / 2. \quad (6)$$

where k is, usually, equal with one. The interleaver function is obtained using the relation :

$$\pi(c(i)) = c(i+1), \quad \forall i \in I. \quad (7)$$

III. Interleavers for non-binary turbo-codes

Similarly to the TC classic, for NBTC can be adopted different solutions of the trellis closing. The NBTC trellis closing way alongside the inputs number per coder, the coder memory inclusive the bits numbers per block turbo coded are factors, which have an influence on the form of the interleaver. The structure of the NBTC data block in the different aspects of the trellis closing is suggestively presented on Fig.2.

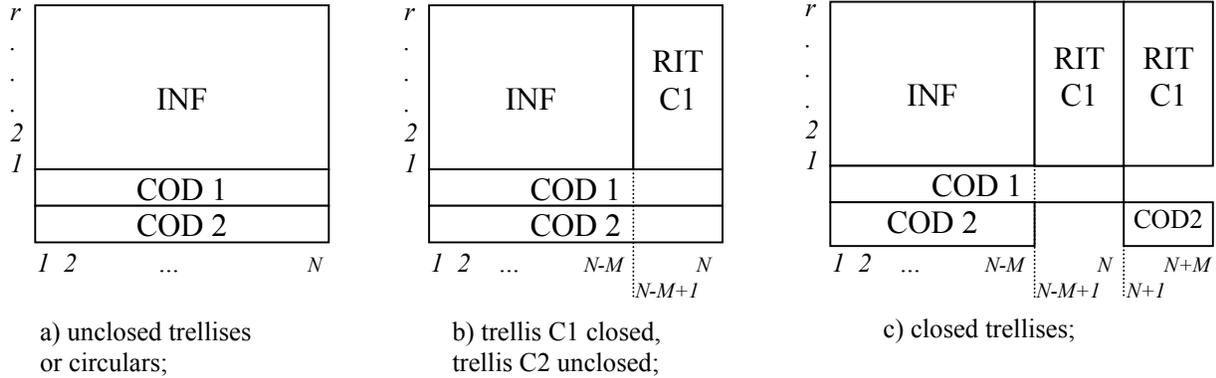


Fig. 2. The turbo-coded block structure in accordance with the NBTC trellis closing way.

The signification of the notations from Fig. 2 is the following: r – the inputs number (taps) in coder; N – the length of a coded block (of a one multi-component coder); M – the components coders memory; INF –the information bits; RIT – the redundant bits for the trellis closing; COD –the resulted bits from coding, the control bits. The interleaving is done either exclusively on INF block (the a) or c) cases) or on INF+RIT¹ blocks. Therefore, the block, which must be interleaved, has a matrix with the $r \times N$ or $r \times (N-M)$ dimensions. This matrix structure imposes that the bits, which were on the same column, must not be on the same column after interleaving (the column interleaving). In the following, we consider the interleaving block with $r \times N$ dimension. The interleaving of such block can be thought in three distinct ways:

A. The $r \times N$ block interleaving $1 \times r \cdot N$ line transformed

After the data block transformation, from the matrix form (Fig.2a)) on the vector form (Fig.3 a)), imposing the interleaver length to be multiple of r , it can be applied any methods presented in precedent paragraph. The method disadvantage consists on the column interleaving failure. To satisfy this requirement, the column matrix transformation by lines multiplexing (by a preliminary column-line interleaving) (Fig.3 b)) must be done.

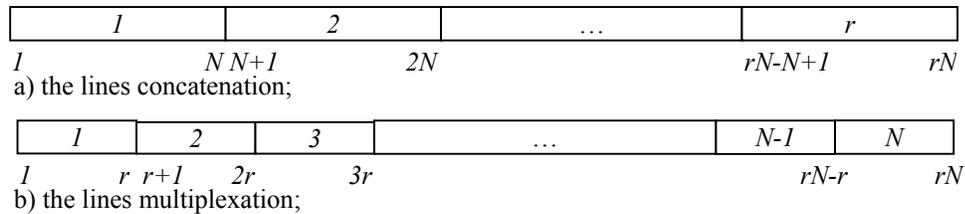


Fig. 3. The data block transformed line (vector).

¹ The discussion was limited only for parallel concatenation (pure turbo code). In the case of the serial concatenation the interleaving operates on COD1 bits, too.

B. The interleaving of the $r \times N$ block by specific methods of the matrix

These “specific methods” of the matrix implies the interleaving functions, which have two indexes (i, j). Thus, the position before and after interleaving depends of two indexes. A typical example is the block or rectangular interleaving, given by relation (5).

The directly applying of the rectangular interleaving on one block with $r \times N$ dimension, usually small r , will leads to a minimum interleaving distance, $r+1$. A differential cyclical operation of the data block, to eliminate this problem, is proposed before the interleaving, operation what is shown in Fig. 4. The differential cyclical consists in one cyclic permutation operation of the lines, with different steps from the line to line.

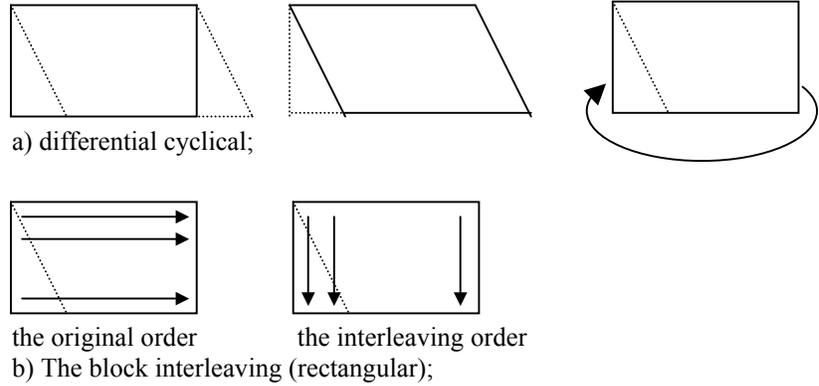


Fig. 4. The block interleaving bloc with differential cyclical.

C. The independent interleaving of the block lines

The independent interleaving of the block lines can be done with any methods presented in the precedent paragraph. If, for each line, is used the same interleaver function, the interleaving is done trough the symbols (the column is not interleaved). The advantage is the using of the N/r length interleaver. To realize the column interleaving also a preliminary differential cyclical can be used.

IV. Experimental results

A memory 3 NBTC was used on the simulations. The component codes are RSCs with generating matrix $G=[13/15, 11/15]$. For decoding, the Maximum APosteriori algorithm, MAP, was employed. The 15 iterations were proposed for each block. A LLR stop criterion was selected [6]. The AWGN channel and the BPSK modulation were used. The used interleavers are described in Table 1, and the obtained results are shown on the Fig.5 diagrams.

Table 1 The used interleavers.

Interleaver	Block length	Component Code	Interleaver design
AN1	448	131115	C (differential cyclical) S interleaver of 224 bits
BN1	448	131115	A (lines multiplexing) S interleaver of 448 bits
AN2	1792	131115	C (differential cyclical) S interleaver of 896 bits
BN2	1784	131115	A (lines multiplexing) S interleaver of 1784 bits
CN2	1784	131115	A (lines concatenation) S interleaver of 1784 bits
DN2	1784	131115	A (lines concatenation) interleaver CCSDS
1513	1784	1513 classic punctured	S interleaver of 1784 bits

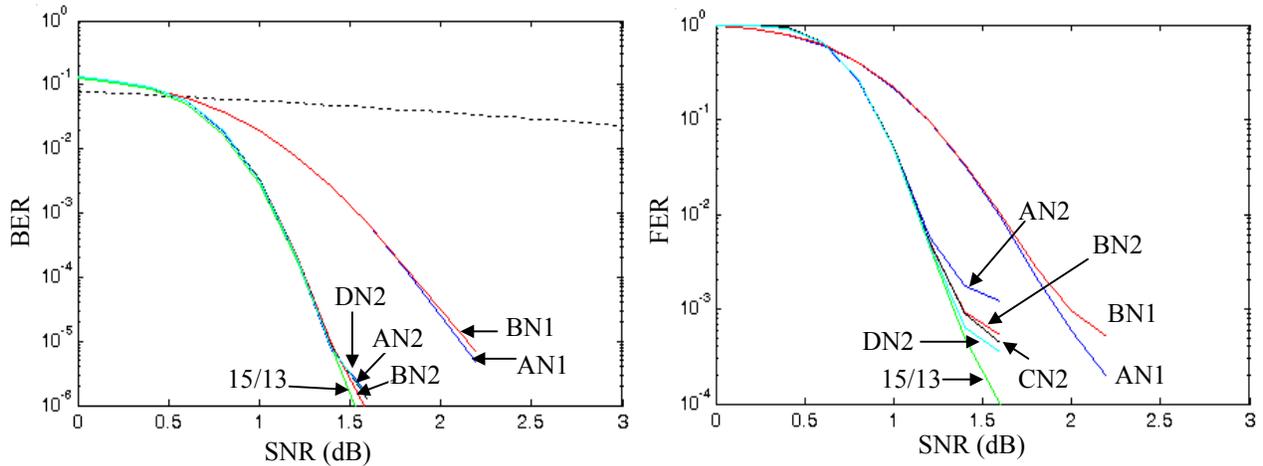


Fig. 5 The BER and FER performances of the NBTC

V. Conclusions

In this paper the particularities of the interleavers used in the case of the Recursive Systematic Convolutional Code (RSC), $n/(n+2)$ rate, with $n=2$, unpunctured non-binary turbo codes (NBTC) were described. We built interleavers for two of the three methods described in paragraph 3. The performances of these interleavers (in conjunction with a memory-3 MBTC) are presented in Fig. 5 and compared with the most performing punctured classic TC (15/13) with the same coding rate. The diagrams from Fig. 5 show that BER performances are very close to the performances of the TC classic. But, there is an essential difference in the FER performances. The most important interleaver, from BER point of view is (BN2), the interleaver obtained with method A with the lines multiplexing.

Acknowledgement

The authors want to thank their Ph.D. director, Professor Miranda Naforniță, for the suggestion to write this paper and for her continuous help.

References

- [1] C. Berrou, A. Glavieux, P. Thitimajshima - "Near Shannon limit error-correcting coding and decoding: Turbo-codes", Proc. ICC'93, Geneva, Switzerland, May 1993, pp. 1064 – 1070;
- [2] C. Berrou, M. Jézéquel, C. Douillard, S. Kerouédan, „The Advantages of Non-Binary Turbo Codes”, Information Theory Workshop ITW2001 Cairns, Australia, Sept 2-7, 2001, pp.61–63;
- [3] S. A. Bărbulescu, S. S. Pietrobon, ”Turbo Codes: a tutorial on a new class of powerful error correcting coding schemes. Part I: Code Structures and Interleaver Design”, 26 October, 1998;
- [4] Consultative Committee for Space Data Systems “Telemetry Channel Coding”, Blue Book, CCSDS 101.0-B-6, Oct 2002.
- [5] O.Y. Takeshita, D.J. Costello - “New Classes of Algebraic Interleavers for Turbo-Codes”, ISIT 1998, August 16-21, USA.
- [6] L. Trifina, H.G. Balta, A. Rușinaru, “Decreasing of the Turbo MAP Decoding Time by Using an Iterations Stopping Criterion”, paper accepted on the IEEE conference, Iasi, 14-15 July.