2011 International Conference on Communication Engineering and Networks
IPCSIT vol.19 (2011) © (2011) IACSIT Press, Singapore

Mirror equivalent turbo-codes – part I

Horia Balta, Alexandru Isar, Dorina Isar and Maria Balta

Electronics and Telecommunications Faculty, Politehnica University of Timisoara, Timisoara, Romania
horia.balta@etc.upt.ro

Abstract. the present paper extends the concept of (convolutional) codes equivalence to the turbo codes (TCs). The concept of equivalence of TCs is useful in the exhaustive search of pairs of component codes and interleavers, unexplored yet, for the design of turbo-coding systems. This design procedure is innovative because until now only independent components optimization were considered. In accordance with this new concept of equivalence (named in the mirror), the codes for which the code words are identical by transposition (inversion) are equivalent (from the point of view of Bit Error Rate - BER's performance). The conditions for two convolutional codes or two convolutional turbo-codes respectively to be mirror equivalent are derived in this paper. The results obtained are of theoretical nature and are derived by mathematical methods. They are confirmed by simulation in a companion paper.

Keywords - convolutional codes; equivalence; generator matrix; permutation; turbo codes;

1. Introduction

The wide range of applications of TCs leaves open the concern of their design, by optimization or by investigation of new families of turbo-coding systems. If we take into account, for example, the TCs family that associates the class of memory 3 recursive systematic double binary convolutional codes (RSDBC), as component codes, with the class of Quadratic Polynomial Permutations (QPP) interleavers of length *N*=752 [1], it counts over a million components[1]. An exhaustive search of the best pairs over such a large set implies great difficulties. This complexity amplifies exponentially with the increase of the memory or if new families of TCs [2] are included in the search. The idea of eliminating some competitors, knowing that they are equivalent as performance with other elements already investigated becomes attractive for the reduction of the complexity of the search. The present paper proposes a new equivalence criterion for TCs named in the mirror. This concept of equivalence may be extended easily to all the codes families. The structure of the paper is the following. The second section contains a short presentation of the recursive systematic multi binary convolutional codes and their equivalence in the mirror is defined. This concept is extended to the case of TCs in the third section and the necessary conditions for TCs to be equivalent in the mirror are derived. The fourth section presents some conclusions.

2. Convolutional Codes

2.1. Convolutional codes – general description

For the beginning we recall some definitions useful for our goal.

Definition 1 ([3], page 721): A convolutional encoder over a finite field *F* is a *k*-input *n*-output constant linear causal finite-state sequential circuit.

Using the polynomial representation (i.e. the *D* transform of the semi-infinite sequence $x_i=\{x_{i,0}\ x_{i,1}\ \dots\ x_{i,j}\ \dots\}$ which is $x_i(D)=\sum_{j=0}^{\infty} x_{i,j}\cdot D^j$) and denoting by $x=[x_k\ \dots\ x_2\ x_1]$ and $y=[y_n\ \dots\ y_2\ y_1]$, the input and output sequences of the convolutional encoder, the encoding relation can be written as:

[1] This is an estimation of the total number of TCs of this type, without taking into account their performance;

$$y(D) = x(D) \cdot G(D), \quad (1)$$

where $G(D)$ is the generating matrix of the convolutional encoder of dimensions $k \times n$.

Definition 2 ([3] pag. 725): The code generated by a convolutional encoder G is the set of all codewords $y=x \cdot G$, where the k inputs x are arbitrary.

Definition 3 ([3] pag. 725): Two encoders are *equivalent* if they generate the same code.

Further on, we will denote with $\mathcal{C}_G$ the code generated by G. We will notice that the equivalence of two codes $\mathcal{C}_{G1}$ and $\mathcal{C}_{G2}$, according to the third definition, doesn't supposes the identity of the sets $\{(x,y) | y=x \cdot G_1, \forall x \in F[D]\}$ [2] and $\{(x,y) | y=x \cdot G_2, \forall x \in F[D]\}$ [3]. It is sufficient the identity of the sets $\{y | y=x \cdot G_1, \forall x \in F[D]\}$ and $\{y | y=x \cdot G_2, \forall x \in F[D]\}$. In other words, according to definition 3, two encoders are equivalent only if they generate the same code words, even if a certain word y may be generated by two different input sequences, $y = x_1 \cdot G_1 = x_2 \cdot G_2$, with $x_1 \neq x_2$.

Also note that a canonical realization for an (convolutional) encoder, [3], is „one with a number of memory elements equal to the dimension of the abstract state space" dimension (the encoder memory, m) which[4] is equal to the maximum degree of polynomials that form[5] the G matrix.

2.2. Convolutional codes used in turbo-codes

Convolutional codes used in turbo-codes [4] present some more particularities than those ones previously defined:

- Encoding and decoding operate over some sequences of finite dimension, N.
- Almost exclusively recursive and systematic convolutional codes are used. The attribute „systematic" supposes that the generator matrix has the form $G(D) = [I_k \quad G'(D)]$, in which I_k is the unitary matrix of rank k and $G'(D)$ is a matrix of dimension $k \times (n-k)$. The recursion is given by the fact that the matrix $G'(D)$ is a *rational* one, i.e. its elements are polynomials ratios of a dimension smaller or even to the encoder memory. Further on we will narrow the discussion to the case $n=k+1$. This condition is, on one hand, ordinary in TCs, and, on the other hand, it doesn't restrict the generality, the results presented further being easily generalized to the case $n>k+1$. If $n=k+1$, the coding rate of a convolutional component code is $k/(k+1)$ and the coding rate of the corresponding TC is $k/(k+2)$, supposing that the TC is composed of two convolutional component codes[6].
- We will suppose that both trellises of the TC (corresponding to the two convolutional component encoders) are closed and the dimension of the input data block is $k \cdot N$. More specifically, we suppose that the trellises closing is done through interleaved dual termination (IDS), [5]. In this case the TC's rate is diminished to the value: $(k \cdot N - 2 \cdot m)/((k+2) \cdot N)$, as m bits are necessary to determine a state into the states space[7].

Without restraining the generality, we will suppose that the implementation of the equation (1) is accomplished by the circuit from Fig. 1. Denoting with $g_i(D) = \sum_{j=0}^{m} g_{i,j} \cdot D^j$, $0 \leq i \leq k$, the generator matrix of this encoder is:

$$G(D) = \left[I_k \left| \begin{matrix} g_k(D)/g_0(D) \\ \cdots \\ g_2(D)/g_0(D) \\ g_1(D)/g_0(D) \end{matrix} \right. \right]. \quad (2)$$

So, if $x(D) = [x_k(D) \quad \ldots \quad x_2(D) \quad x_1(D)]$ is an input sequence, i.e. $x_i(D) = \sum_{j=0}^{N-1} x_{i,j} \cdot D^j$, $1 \leq i \leq k$, then the

---

[2] $F[D]$ is defined in [3] as being the ring of polynomials with coefficients in F;
[3] Actually, the identity of sets would suppose the *equality* of codes;
[4] At least for the involved encoders in the present study;
[5] For more details see [3], Apendix II;
[6] All the results presented may be easily generalized to the case of TCs with more than two component codes, but these ones, at least till present, have a less practical importance;
[7] The m bits are (theoretically) enough to close the trellis, i.e. to take it from whatever state to a zero one;

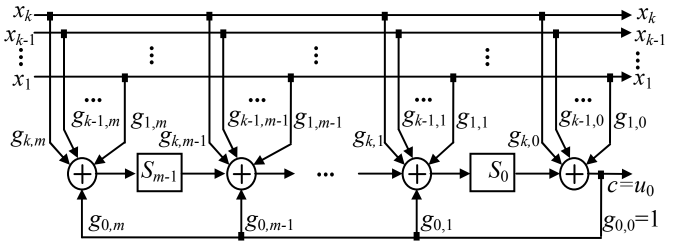

Fig 1 The general structure of a recursive and systematical multi-input convolutional encoder, with $k/(k+1)$ rate; the observer canonical form.

code word generated by the encoder is:

$$y(D) = [x_k(D) \quad \dots \quad x_2(D) \quad x_1(D) \quad x_0(D)], \tag{3}$$

where:

$$x_0(D) = c(D) = \sum_{i=1}^{k} \frac{g_i(D)}{g_0(D)} \cdot x_i(D) = \sum_{j=0}^{N-1} x_{0,j} \cdot D^j \ . \tag{4}$$

Implicitly, the equation (4) restricts the class of sequences $x(D)$. To generate a code word for the code $\mathcal{C}_G$ (given by the matrix G), the polynomial $\sum_{i=1}^{k} g_i(D) \cdot x_i(D)$ must be divisible with $g_0(D)$. Hence, the sequences $x(D)$ cannot be chosen randomly. For this reason we propose the following definition.

Definition 4. The recursive and systematic convolutional block code of memory m, $\mathcal{C}_{G,N}$, generated by the matrix G, having k inputs and k+1 outputs, of length N, is given by the vector $y(D) = [x_k(D) \quad \dots \quad x_1(D) \quad x_0(D)]$ where the sequences $x_i(D) = \sum_{j=0}^{N-1} x_{i,j} \cdot D^j$, $0 \le i \le k$, satisfy the equation:

$$x_0(D) \cdot g_0(D) = \sum_{i=1}^{k} x_i(D) \cdot g_i(D) . \tag{5}$$

The polynomials $g_i(D) = \sum_{j=0}^{m} g_{i,j} \cdot D^j$, $0 \le i \le k$, in the last equation define the generator matrix G. Certainly, we will take into account the binary field, so $x_{i,j}, g_{i,j} \in F = \{0,1\}$. For the sake of simplicity, we will use in the following the denotation:

$$G = [g_k \quad \dots \quad g_1 \quad g_0], \tag{6}$$

where g_i is the decimal transposition of the binary sequence ($g_{i,m}$ $g_{i,m-1}$ ...$g_{i,1}$ $g_{i,0}$), and $g_{i,m}$ is the most significant bit.

2.3. Mirror equivalent convolutional codes

Using the same arguments, we will define the mirror equivalence.

Definition 5 Two encoders, G and G_d are mirror equivalent, if the generated code words are mirror-imaged:

$\forall$ $y \in \mathcal{C}_{G,N}$, then $y_d \in \mathcal{C}_{Gd,N}$, where $y_d(D) = y(D^{-1}) \cdot D^{N-1}$, and
$\forall$ $y_d \in \mathcal{C}_{Gd,N}$, then $y \in \mathcal{C}_{G,N}$, where $y(D) = y_d(D^{-1}) \cdot D^{N-1}$.

The next theorem shows the conditions in which two encoders can be mirror equivalent. It is formulated for k=2 for easier expression and demonstration, but the generalization is obvious.

Theorem 1. If the polynomials g_0, g_1, g_2, g_3 satisfy the conditions:

$$g_2(D) = D^m \cdot g_0\left(D^{-1}\right) \text{ and } g_3(D) = D^m \cdot g_1\left(D^{-1}\right) \tag{7}$$

then the codes generated by the matrices G=[g_2 g_1 g_0] and G_d=[g_0 g_3 g_2] are equivalent in mirror.

Proof. A word generated by the matrix G has the form $y(D)$=[$x_2(D)$ $x_1(D)$ $x_0(D)$], where $x_i(D) = \sum_{j=0}^{N-1} x_{i,j} \cdot D^j$, $i = 0,1,2$, with the restriction:

$$x_0(D) = \frac{g_2(D)}{g_0(D)} \cdot x_2(D) + \frac{g_1(D)}{g_0(D)} \cdot x_1(D), \tag{8}$$

or:

$$x_0(D)\cdot g_0(D)=x_2(D)\cdot g_2(D)+x_1(D)\cdot g_1(D). \tag{9}$$

Defining:

$$x_{2d}(D)=D^{N-1}\cdot x_2\left(D^{-1}\right) \text{ and,}$$
$$x_{1d}(D)=D^{N-1}\cdot x_1\left(D^{-1}\right), \tag{10}$$

it results from (7):

$$\begin{aligned} x_2(D)\cdot g_2(D)&=D^{N+m-1}\cdot x_{2d}\left(D^{-1}\right)\cdot g_0\left(D^{-1}\right)\\ x_1(D)\cdot g_1(D)&=D^{N+m-1}\cdot x_{1d}\left(D^{-1}\right)\cdot g_3\left(D^{-1}\right)\end{aligned}. \tag{11}$$

Using (9) and (11) it results:

$$\begin{aligned} x_0(D)\cdot g_0(D)=\ &D^{N+m-1}\cdot\left[x_{2d}\left(D^{-1}\right)\cdot g_0\left(D^{-1}\right)+x_{1d}\left(D^{-1}\right)\cdot g_3\left(D^{-1}\right)\right]=\\ &D^{N+m-1}\cdot x_{0d}\left(D^{-1}\right)\cdot g_2\left(D^{-1}\right),\end{aligned} \tag{12}$$

from where, taking into account (7), it results:

$$x_{0d}(D)=D^{N-1}\cdot x_0\left(D^{-1}\right), \tag{13}$$

in which $x_{0d}(D)$ is defined like the redundant output for the encoder generated by the matrix G_d:

$$x_{0d}(D)=\frac{g_0(D)}{g_2(D)}\cdot x_{2d}(D)+\frac{g_3(D)}{g_2(D)}\cdot x_{1d}(D). \tag{14}$$

The result obtained in equation (13) states that if $x=[x_2\ x_1]$ is an input sequence for the encoder defined by G, and $y=[x_2\ x_1\ x_0]$ is the corresponding sequence at its output, then the sequence $x_d=[x_{2\mathrm{d}}\ x_{1\mathrm{d}}]$ (the mirror image of the sequence x through the encoding performed by the matrix G_d) will generate the sequence $y_d=[x_{2d}\ x_{1d}\ x_{0d}]$, i.e. the mirror image of y. (q.e.d.)

Using the notation $\mathfrak{M}(u)$ for the mirror image of the sequence u, Theorem 1 states that $G=[\mathfrak{M}(g_0)\ g_1\ g_0]$ and $G_d=[g_0\ \mathfrak{M}(g_1)\ \mathfrak{M}(g_0)]$ generates mirror equivalent codes. A generalization of theorem 1 is the following.

Theorem 1 (*generalization*): Codes $\mathcal{C}_{G,N}$ and $\mathcal{C}_{Gd,N}$ generated through encoders having the matrices $G=[g_k\ ...\ g_{i+1}\ \mathfrak{M}(g_0)\ g_{i-1}\ ...\ g_0]$ and $G_d=[\mathfrak{M}(g_k)\ ...\ \mathfrak{M}(g_{i+1})\ g_0\ \mathfrak{M}(g_{i-1})\ ...\ \mathfrak{M}(g_0)]=\mathfrak{M}(G)$ are mirror equivalent, with $1\leq i\leq k$.

So, the mirror equivalence works on the class of systematic encoders as well. If we apply the definition 3 on this class then the general equivalence will automatically lead to equality. In contrast, mirror equivalence can involve systematic encoders that are not equal.

In the next section, we will investigate the possibility that two turbo-codes to be equivalent. Due to the obtrusion that the input sequence must be segmented in blocks of N bits, we will search only the mirror equivalence.

3. Mirror Equivalent Turbo-Codes

3.1. Turbo-codes, general features

In fig. 2 is presented the general scheme of a TC. C1 and C0 are the component encoders which code the k input sequences, marked $x_{2\div(k+1)}$ and generate a turbo-coded word in the form $y_{TC}(D)=[x_{k+1}(D)\ \ ...\ \ x_2(D)\ \ x_1(D)\ \ x_0(D)]$. After multiplexing and modulation (operations that are not figured in Fig. 2) this word crosses the channel. The channel output consists of the soft sequences: $v_{0\div(k+1)}=x_{0\div(k+1)}+w_{0\div(k+1)}$, resulted by summing the noise (which will be considered as AWGN in the following) with the emitted sequence. L_{a0} and L_{a1} represent the a priori information obtained by interleaving and re-interleaving the extrinsic information L_{e1} and L_{e0} respectively, generated by the decoders DEC1 and DEC0; L_1 is the soft output of the TC, which can be the logarithmic likelihood ratio (LLR) in case k=1 [6], or a priori probability (APP) for the case $k\geq 2$ and symbol wise decoding [7].

Regarding the equivalence, the essential difficulty raised by the TC is the interleaving which must be performed before the decoding made by the second encoder.

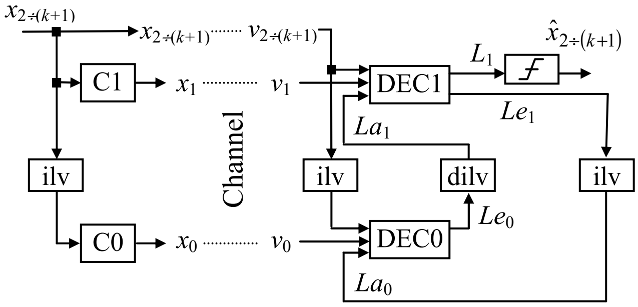

Fig. 2. The general scheme of the TC. The turbo encoder composed by C1, C0 and ilv (left part). AWGN communication channel (middle part). The turbo decoder composed by DEC1, DEC0, ilv and dilv (right part).

This interleaving has an implicit effect on the last output of the turbo-encoder, marked x_0 in Fig 2. If the TC performs inner-symbol interleaving then, in the case $k>1$, this operation affects all sequences that contribute to the calculation of the second encoder output.

For a mathematical formulation of the TCs' equivalence problem, we propose first the following definition.

Definition 6. The turbo-code, denoted by $\mathcal{TC}_{G,N,\pi}$, built by the component encoders that implement the generating matrix $G=[g_k \dots g_1\, g_0]$ and by the interleaver which corresponds to the permutation $\pi:\mathcal{K}\times\mathcal{T}\rightarrow\mathcal{K}\times\mathcal{T}$, where $\mathcal{K}=\{1,2, ..., k\}$ and $\mathcal{T}=\{0,1,2, ..., N\text{-}1\}$, is constituted from the group of words of the form $y_{TC}=[x_{k+1} \dots x_2\, x_1\, x_0]$, satisfying the following equations:

$$\begin{cases} x_1(D)=\sum\limits_{i=2}^{k+1}\dfrac{g_{i-1}(D)}{g_0(D)}\cdot x_i(D) \\ x_0(D)=\sum\limits_{i=2}^{k+1}\dfrac{g_{i-1}(D)}{g_0(D)}\cdot x_{\pi i}(D) \end{cases},\ x_i(D)=\sum_{j=0}^{N-1}x_{i,j}\cdot D^j\ , \tag{15}$$

where $x_{i,j}\in\{0,1\}$, $0\le i\le k+1$, and

$$x_\pi(D)=\lfloor x_{\pi,i}(D)\rfloor_{k+1\ge i\ge 2}=\pi\{x\}(D)=\pi\{\lfloor x_i\rfloor_{k+1\ge i\ge 2}\}(D)\,, \tag{16}$$

is the input sequence interleaved by the function π.

In definition 6, we considered both symbol interleaving (which operates on symbols with index from 0 to N-1) and inner-symbol interleaving (which operates on the k bits of a symbol).

Obviously, for k=1, only the symbol interleaving exists. Between the TCs with $k>1$, the ones used in practice until nowadays are the duo-binary TCs only. In their case, the inner/symbol interleaving resumes to reversing the positions of the two bits for the symbols with an even index. We did not restricted the operation of the inner-symbol interleaving only in the frame of the symbol in equation (16) but in practice this restriction is imposed for keeping a minimal level for the complexity of the turbo decoding.

In the following paragraph, we will present the conditions in which two TCs can be equivalent using the definition 6 which associate a TC with the multitude of turbo-coded words.

3.2. Mirror equivalent TCs

We can formulate now the conditions for the mirror equivalence of two TCs.

Theorem 2. Two TCs, $\mathcal{TC}_{G,N,\pi}$ and $\mathcal{TC}_{Gd,N,\pi d}$, are mirror equivalent if the component codes, $\mathcal{C}_{G,N}$ and $\mathcal{C}_{Gd,N}$, are mirror equivalent and the corresponding interleaver functions satisfy the relation:

$$\pi_d(\mathcal{M}(u))=\mathcal{M}(\pi(u)), \tag{17}$$

where u is any input sequence for the encoder defined by the matrix G.

Proof. Mirror equivalence supposes that if $y=[x_{k+1} \dots x_2\, x_1\, x_0]$ is a code word for $\mathcal{TC}_{G,N,\pi}$, then the mirror image, $\mathcal{M}(y)=[\mathcal{M}(x_{k+1}) \dots \mathcal{M}(x_2)\, \mathcal{M}(x_1)\, \mathcal{M}(x_0)]$ is a code word for $\mathcal{TC}_{Gd,N,\pi d}$. As can be seen in the left part of Fig.

2, the output of the block ilv, which implements the interleaving, is connected at the input of the secondary convolutional encoder C0 only. Hence, the interleaving does not affect the generation of the first k+1 components of the turbo-coded word. In consequence the mirror equivalence of the TCs is ensured by the mirror equivalence of the primary component encoders C1 and C1d, for these components. It remains to establish the mirror equivalence for the last component, that is the output sequence (redundancy) generated by the secondary encoders, C0 (Fig. 2) and C0d. In the case of the TC $\mathcal{TC}_{G,N,\pi}$, this sequence has the expression :

$$\begin{aligned} x_0(D) &= \sum_{i=2}^{k+1} \frac{g_{i-1}(D)}{g_0(D)} \cdot x_{\pi,i}(D) \\ &= \left[\frac{g_k(D)}{g_0(D)} \quad \cdots \quad \frac{g_1(D)}{g_0(D)} \right] \cdot \left[x_{\pi,(k+1)}(D) \quad \cdots \quad x_{\pi,2}(D) \right]^* \\ &= G_{i/0}(D) \cdot \pi\{x_{k+1 \geq i \geq 2}\}^*(D), \end{aligned} \tag{18}$$

in which * symbolizes the transposition operation and $G_{i/0} = [g_k \quad \ldots \quad g_1]/g_0$. We denoted by [$x_{\pi,k+1}$ ... $x_{\pi,3}$ $x_{\pi,2}$], the interleaved sequences set, which is obtained from the set [x_{k+1} ... x_3 x_2] through the interleaver function π (including inner-symbol). Denoting by $u = x_{k+1 \geq i \geq 2}$, the mirror image of x_0 becomes:

$$\mathcal{M}(x_0(D)) = x_0(D^{-1}) \cdot D^{N-1} = G_{i/0}(D^{-1}) \cdot \pi\{u\}^*(D^{-1}) \cdot D^{N-1} = \mathcal{M}(G_{i/0}(D)) \cdot \mathcal{M}(\pi\{u\}(D))^*. \tag{19}$$

If u is an input sequence for the code $\mathcal{C}_{G,N}$, then $u_d=\mathcal{M}(u)$ is an input sequence for the code $\mathcal{C}_{Gd,N}$ (thanks to mirror equivalence). Similarly to equation (18), the redundant sequence generated by C0d is:

$$x_{d0}(D) = G_{d,i/0}(D) \cdot \pi_d\{u_d\}^*(D). \tag{20}$$

Due to the mirror equivalence of the codes $\mathcal{C}_{G,T}$ and $\mathcal{C}_{Gd,T}$, we have that $\mathcal{M}(G_{i/0}(D))=G_{d,i/0}(D)$. Finally, the identity of expressions (19) and (20) is ensured by the equality: $\mathcal{M}(\pi\{u\}(D))= \pi_d\{u_d\}(D)$. (q.e.d.)

4. Conclusions

We have defined the concept of mirror equivalence. Two codes are equivalent if their code words are pairs of mirror images. We have established the conditions for two recursive and systematic convolutional codes and for two TCs (which incorporate recursive and systematic convolutional codes) respectively to be equivalent in mirror. The condition that a convolutional code associated to the matrix G=[g_k ... g_1 g_0] to have an equivalent in mirror is that the mirror image of the generator polynomial g_0 to be found between the polynomials g_1 ... g_k (in conformity with the generalization of Theorem 1). If this condition is satisfied, then the mirror equivalent code is generated by the matrix G_d, whose generator polynomials are the mirror images of the polynomials g_0 g_1 ... g_k. The conditions which must be satisfied by two TCs to be mirror equivalents are:

i) the component convolutional codes to be equivalent in mirror and
ii) the two interleavers used to satisfy condition (17), i.e. the mirrored sequence permuted by an interleaver to be identical with the sequence permuted by the other interleaver and next mirrored.

The mirror equivalence concept can be applied for any type of correcting codes. The identification of pairs of equivalent TCs is useful for the design of TCs based on exhaustive search of pairs of good components because the number of the TCs which must to be investigated diminishes.

5. Acknowledgment

The work of Horia Balta was supported by the project "Development and support for multidisciplinary postdoctoral programs in primordial technical areas of the national strategy for research - development - innovation" 4D-POSTDOC, contract nr. POSDRU/89/1.5/S/52603, project co-funded from the European Social Fund through the Sectorial Operational Program Human Resources 2007-2013.

A part of the research which stays at the basis of this paper was developed in the framework of a grant of the Romanian Research Council (CNCSIS) with the title “Using Wavelets Theory for Decision Making”, no. 349/13.01.09.

6. References

[1] Q J. Sun and O. Y. Takeshita, “Interleavers for turbo codes using permutation polynomials over integer rings,” IEEE Transactions on Information Theory, vol. 51, no. 1, Jan. 2005, pp. 101–119

[2] C. Berrou, A. Graell i Amat, Y.Ould-Cheikh-Mouhamedou, Y. Saouter, “Improving the distance properties of turbo codes using a third component code: 3D turbo codes”, Transactions on Communications, vol. 57-9, September 2009, pp. 2505 - 2509

[3] G. D. Forney, jr, “Convolutional Codes I: Algebraic Structure”, IEEE Transactions on Information Theory, vol. IT-16, No. 6, November 1970, pp. 720–738.

[4] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: turbo-codes,” in Proc. IEEE Int. Conf. on Commun., Geneve, Switzerland, May 1993, pp. 1064–1070.

[5] P. Guinand and J. Lodge, “Trellis termination for turbo encoders,” in Proc. 17th Biennial Symposium on Communications, Kingston, Ontario, Canada, May 1994, pp. 389–392

[6] L.Hanzo, T.H.Liew, B.L.Yeap, “Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission over Fading Channels”, John Wiley & Sons Ltd, England, 2002

[7] L. R. Bahl, J. Cocke, F. Jelinek, J. Raviv, “Optimal Decoding of Linear Codes for Minimising Symbol Error Rate”, IEEE Transactions on Information Theory, vol. 20, pag. 284-287, martie 1974