UNIVERSIDAD TECNOLÓGICA DE PANAMÁ

INFORME DE VIAJE

TIPO Y NOMBRE DE LA ACTIVIDAD

Conferencia LATINCOM 2014: 6th IEEE Latin-American Conference on Communications
http://www.ieee-latincom.org/

LUGAR Y FECHA (Duración)

Cartagena de Indias, Colombia
Del 5 al 7 de noviembre de 2014 (3 días)

OBJETIVOS

- Divulgar resultados de nuestra investigación incluidos en el artículo "Implementation of Encoders and Decoding Performance Analysis of Direct Product Convolutional Codes".
- Incrementar la producción científica de la Universidad Tecnológica de Panamá.
- Establecer vínculos académicos y de investigación con miembros de otros países con el propósito de impulsar la investigación y mejorar y actualizar la educación en ingeniería.
- Participar en las conferencias magistrales, foros y tutoriales del evento, y actualizarme en temas de las áreas de telecomunicación y procesamiento de señales.

PARTICIPANTE (S)

Carlos Alberto Medina Cerrud

ASPECTOS RELEVANTES EN EL DESARROLLO DE LA ACTIVIDAD

- Alto nivel y valiosa información obtenida de los tutoriales "Trends in modern RF Spectrum Management" por Dr. Carlos E. Caicedo Bastidas, PhD (Syracuse University, U.S.A), y "Future of Optical Communication Systems and Networks" de Fabio Pittalá, Jorge López Vizcaíno, Huawei Technologies Duesseldorf GmbH, European Research Center.
- Participación en las diversas actividades de la conferencia, lo cual permitió un alto grado de interacción con colegas y estimuló discusiones de posibles colaboraciones futuras.

RESULTADOS OBTENIDOS (Contacto con futuros expositores, becas, firma de convenio, etc.)

- Ponencia del tema de investigación "Implementation of Encoders and Decoding Performance Analysis of Direct Product Convolutional Codes" en la sección de conferencias.
- Publicación del artículo "Implementation of Encoders and Decoding Performance Analysis of Direct Product Convolutional Codes" en la base de datos de publicaciones indexadas IEEE Xplore y su publicación en el Proceedings de la conferencia.
- Contacto y acuerdo verbal de colaboración futura con el Sr. Darin Mosquera de la Universidad Distrital Francisco José de Caldas, Bogotá, Colombia en el tema de publicaciones e intercambio de evaluadores y articulistas para las revistas que manejan la Facultad Tecnológica de la Universidad indicada y la revista Prisma Tecnológico de la Universidad Tecnológica de Panamá.

- Promoción de nuevos temas de investigación y colaboración con colegas de otras universidades, así como material actualizado sobre temas de telecomunicación y procesamiento de señales que será usado en los cursos de pregrado y postgrado.

CONCLUSIONES

Se tuvo una participación activa en la conferencia, presentando la información incluida en el artículo y asistiendo a todas las sesiones y actividades.

En general, la conferencia fue de un buen nivel, y es importante indicar que el "Proceedings" de la misma está indexado y los artículos se publican en IEEE Xplore con visibilidad mundial.

En particular, los tutoriales "Trends in modern RF Spectrum Management" y "Future of Optical Communication Systems and Networks", así como el foro sobre administración del espectro radio-eléctrico fueron de mucho interés y con información valiosa.

RECOMENDACIONES

La participación de nuestros docentes e investigadores en este tipo de congresos internacionales es de suprema importancia y debe seguir siendo estimulada y apoyada. Le procura formación continua a los profesionales y visibilidad a la institución a nivel internacional.

ANEXOS

Fotos del evento.
Certificado de participación como expositor en el evento.
Artículo de la conferencia.

Firma y cédula del participante:

CIP 8-326-519

Fecha de entrega del informe:

27 de noviembre de 2014


IEEE
Implementation of Encoders and Decoding Performance Analysis of Direct Product Convolutional Codes


IEEE LATINCOM
6th IEEE LATIN-AMERICAN CONFERENCE ON COMMUNICATIONS
5-7 NOVEMBER 2014 • CARTAGENA DE INDIAS, COLOMBIA

IEEE
LATINCOM™
6TH IEEE LATIN-AMERICAN CONFERENCE ON COMMUNICATIONS

5-7 NOVEMBER 2014 • CARTAGENA DE INDIAS, COLOMBIA

Certifies that

Carlos Alberto Medina Cerrud

Participated as: *Author*

With the paper entitled:Implementation of Encoders and Decoding Performance Analysis of Direct Product Convolutional Codes

In the **6th Latin-Amerian Conference on Communications**

November 5th to 7th 2014 in Cartagena de Indias, Colombia

Jose David Cely Callejas
Conference General Chair

IEEE

Colombia
www.ieee.org.co

IEEE COMMUNICATIONS SOCIETY

Implementation of Encoders and Decoding Performance Analysis of Direct Product Convolutional Codes

Carlos A. Medina C.[1] and Mayteé Zambrano N.[1]
[1]School of Electrical Engineering, Universidad Tecnológica de Panamá, Panamá
{carlos.medina, maytee.zambrano}@utp.ac.pa

***Abstract*—Direct product convolutional codes are the result of a special concatenated scheme based on the well-known method of direct product for combining block codes. The approach taken for its construction is to consider convolutional codes as block codes over the field of rational functions *F*(*D*). This relatively new codes has been previously defined and investigated by the author, but in this instance, further properties and aspects about the product convolutional codes are considered. The new aspects studied are: some properties and physical realization of encoders, considerations on convergence of decoding based on EXIT charts and the effect of the blocking factor on BER performance in AWGN channel. In addition, the construction of multidimensional product convolutional codes is considered.**

***Keywords*—blocking factor, direct product convolutional codes, EXIT chart, multidimensional code, physical realization of encoders.**

I. Introduction

Code combining or concatenation is a design method to create new error detection and correction codes based on other well-known codes with desirable properties. This allows for new codes with beneficial characteristics, such as: larger distances, better random and burst error correction capabilities, low density parity check matrices and the use of low complexity iterative decoding methods.

Among the many combining methods proposed for block codes, one of the earliest is the *direct product* proposed by Elias in 1954 [1]. On the other hand, for convolutional codes, the concatenation of two or more short constituent codes was first proposed by Forney [2]. Later, Blokh and Zyablov described a *generalized code concatenation* [3], and then several concatenation schemes followed: *parallel concatenated convolutional codes* (PCCC), known as turbo codes were introduced by Berrou et al. [4]; *serially concatenated convolutional codes* (SCCC) [5], which use the same components of PCCC; and *woven convolutional codes* (WCC) proposed in [6], which are the result of a further development of SCCC. In addition, Lodge et al. in [7][8], described a method similar to the direct product of block codes for convolutional codes.

Looking for new concatenated schemes for convolutional codes, the method of direct product convolutional codes (DPCC) was introduced in [9]. DPCCs are based on two main concepts: *i*) the classic algebraic description by Forney of convolutional codes as block codes over the finite field of rational functions $F(D)$, and *ii*) the direct product method for block codes.

Multiple characteristics of DPCCs are investigated in [9] – [15]. Furthermore, in [12] a general method for combining convolutional codes, based on the indicated concepts, was also proposed.

In this work we present other aspects regarding encoding and decoding of DPCCs, which are not considered in the previous references.

The paper starts with some definitions and properties of DPCCs, showing how these codes result from the direct product of two component codes. Then, some properties of the generator and parity check matrices are considered, as well as the physical implementation of encoders. In addition, two iterative decoding methods for DPCCs, serial and product decoding, are compared using extrinsic information transfer (EXIT) charts, and the effect of a blocking factor on the performance of a DPCC over an AWGN channel is also considered. Finally, we address the construction of multidimensional DPCCs.

II. Definitions and Properties

The algebraic description of convolutional codes and their close similarity to block codes allow us to: apply the direct product method, defined for block codes, to convolutional codes; define the new code, and determine its generator and parity check matrices, in an equivalent form to block codes.

Let F be a Galois field, $F = \mathrm{GF}(q)$ (where q is a prime power $q = p^k$, with p a prime number and k a positive integer), and $F(D)$ the field of rational functions over F in the undetermined D. Then, each non-zero element $a(D)$ of $F(D)$ can be represented as a ratio of polynomials over F, $a(D) = p(D)/q(D)$, where $p(D)$ and $q(D)$ are relatively prime. Thus, we consider the following definitions of a convolutional code and its generator and parity check matrices, similar to a block code.

978-1-4799-7162-6/14/$31.00 ©2014 IEEE

***Definition* 1:** An (n,k) q-ary convolutional code C is defined as a k-dimensional subspace of $[F(D)]^n$, where $F = \mathrm{GF}(q)$, i.e., as a subspace of the vector space of n-tuples over the rational field $F(D)$. The rate R of the code C (n,k) is defined as $R = k/n$. Examples and simulations in this work consider only binary convolutional codes, i.e., $F = \mathrm{GF}(2)$.

***Definition* 2:** A $k \times n$ matrix $G(D)$ over $F(D)$ whose rows form a basis of an (n,k) convolutional code is called a generator matrix of the code.

Let $\mathbf{c}(D) = [c_1(D) \ \ldots \ c_n(D)]$ be a code-word of a convolutional code C (n,k), and $\mathbf{u}(D) = [u_1(D) \ \ldots \ u_k(D)]$ an information word, both in $F(D)$. Then, the coding rule, similar to block codes, is

$$\mathbf{c}(D) = \mathbf{u}(D)G(D) \tag{1}$$

***Definition* 3:** For any convolutional code C (n,k), a parity check matrix $H(D)$ of the code is a $r \times n$ matrix in $F(D)$ with rank $r = n - k$, such that $\mathbf{c}(D)$ is a code-word of C if and only if

$$\mathbf{c}(D)H^{\mathrm{T}}(D) = \mathbf{0} \tag{2}$$

where T denotes the transposition operator.

With these definitions and applying the direct product method we have the definition of a direct product convolutional code (DPCC):

***Definition* 4:** Let C^h (n^h,k^h) and C^v (n^v,k^v) be convolutional codes with code length n^h y n^v, and free distances d^h and d^v, respectively. Then, the direct product code $C^{\otimes} = C^h \otimes C^v$ is defined to be an $(n^h n^v, k^v k^v)$ code whose codewords consist of all $n^v \times n^h$ arrays in which the columns belong to C^v and the rows to C^h. The rate of the direct product code is given by $(k^v k^h / n^v n^h) = R^v R^h$. Thus, this code is a $k^h k^v$ subspace of $[F(D)]^{n^h n^v}$.

Now, let us consider the generator and parity check matrices and the encoding process of the DPCC.

Let G^h and G^v be generator matrices of the horizontal and vertical codes C^h (n^h,k^h) and C^v (n^v,k^v), respectively. Consider an information matrix $U(D)$ $k^v \times k^h$ over $F(D)$. To obtain a code-word (matrix) the encoding can be done in two orders: row (horizontal)–column (vertical) or column–row, resulting in the same code-word (matrix). In the first case, each row of $U(D)$ is first encoded with the matrix $G^h(D)$, and then each column of the resulting matrix $V(D) = U(D)G^h(D)$, is encoded using the matrix $G^v(D)$, resulting the code-word

$$C(D) = G^{v\mathrm{T}}(D)[U(D)G^h(D)] \tag{3}$$

o with the column–row similar process,

$$C(D) = [G^{v\mathrm{T}}(D)U(D)]G^h(D) \tag{4}$$

Both encoding processes are illustrated in Fig. 1. Note that, in any case, the code-word matrix is given by

$$C(D) = G^{v\mathrm{T}}(D)U(D)G^h(D) \tag{5}$$

This expression (of the encoding procedure) can be rewritten for the DPCC in the traditional form, expression (1),

$$C(D) = U(D)G(D) \tag{6}$$

where the generator matrix $G(D)$ is given by

$$G(D) = G^v(D) \otimes G^h(D) \tag{7}$$

and $\otimes$ indicates the Kronecker product.

Since generator matrices $G^v(D)$ and $G^h(D)$ have elements in $F(D)$, i.e., they are rational matrices, from expression (7) results that $G(D)$ is also a rational matrix $k^h k^v \times n^h n^v$ with full rank. Thus, the code C generated by (6) is the direct product of the codes C^h and C^v, and it is a convolutional code with dimension $(n^h n^v, k^h k^v)$.

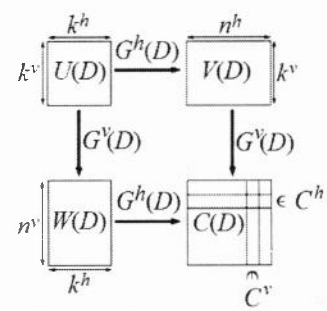

Figure 1. Encoding of a direct product convolutional code.

Let $H^v(D)$ and $H^h(D)$ be parity check matrices of the component codes C^v and C^h, respectively. From Definition 4 we have:

$$\mathbf{c}(D)H^{\mathrm{T}}(D) = \mathbf{0} \tag{8}$$

where the parity check matrix $H(D)$ is given by

$$H(D) = \begin{bmatrix} I_{n^v} \otimes H^v(D) \\ H^h(D) \otimes I_{n^h} \end{bmatrix} \tag{9}$$

where I_m is an identity matrix $m \times m$. Note that $H(D)$ has redundant rows, and therefore it can be reduced. In addition, if n^v or n^h is large, matrix $H(D)$ could be sparse and iterative decoding methods for LDPC (low density parity check) codes could be used.

An interesting property of the new scheme is that the generator matrix of the DPCC inherits the properties of the component codes. Consequently, if a DPCC is built with constitutive codes whose generator matrices are minimal, the resulting generator matrix will be a non-catastrophic generator matrix, the canonical control form of the matrix is a minimal encoder and the corresponding Forney trellis of the code is a minimal trellis. Likewise, if the generator matrices of the component codes are systematic, then the DPCC matrix is also systematic. In addition, it is important to consider the memory m and the constraint length ν of the DPCC [9], which are given by:

$$m = m^v + m^h \tag{10}$$

$$v = k^h v^v + k^v v^h \qquad (11)$$

where m^v, m^h, v^v y v^h are the memories and constraint lengths of the component codes C^v and C^h, respectively.

Concerning the free distance d_f of the DPCC, which is a fundamental parameter related to the performance of the code, it can result smaller than the product of the free distances of the component codes, $d^h_f \times d^v_f$. To estimate d_f and determine restrictions that guarantee the product distance,

$$d_f \geq d^h_f d^v_f \qquad (12)$$

a block distance of a code has been defined [9]:

***Definition* 5:** The block distance d_B of an (n,k) convolutional code C is defined to be

$$d_B(C) = \min_{\substack{\mathbf{c}_1,\mathbf{c}_2 \\ \mathbf{c}_1 \neq \mathbf{c}_2}} \{D_{F(D)}(\mathbf{c}_1, \mathbf{c}_2)\} \qquad (13)$$

where $D_{F(D)}(\mathbf{c}_1,\mathbf{c}_2)$ is the Hamming distance between two vectors $\mathbf{c}_1$ and $\mathbf{c}_2$ in $F(D)$, i.e., the number of positions in which vectors differ. Since the code is linear, d_B is also equal to the minimum Hamming weight over all nonzero code-words $\mathbf{c}(D)$, over $F(D)$, of C.

For a given convolutional code C (n,k) we have

$$d_B \leq n \text{ and } d_B \leq d_f \qquad (14)$$

Thus, for a DPCC, with component codes C^v and C^h, the resulting free distance is:

$$d_f \geq d_B = d^v_B \times d^h_B \qquad (15)$$

where $d^v_B = d_B(C^v)$ and $d^h_B = d_B(C^h)$. In addition,

$$d_f \geq max\ \{d^v_B\, d^h_f, d^h_B\, d^v_f\} \qquad (16)$$

Hence, if the block distance of at least one of the component code equals its free distance, the product distance (12) is guaranteed.

In general, short codes do not reach their free distance due to (14). But this can be overcome by using a *blocked* convolutional code. The *blocking procedure* does not change the convolutional code (n,k) as the set of symbol sequences over the base field F. Blocking a code by a factor M only gather the blocks symbols of length n in larger blocks of length Mn, resulting in a code $C_{[M]}(Mn,Mk)$, which is basically the original code. The block distance of a convolutional code grows with the M factor until it reaches the code free distance, see (14), resulting

$$d_B(C_{[M]}) = d_f \qquad (17)$$

for a given M. The necessary value of M for the block distance to reach the free distance of the code can be calculated using the active distances (column and reverse column) of the considered convolutional code [9].

III. Encoding Aspects

The physical implementation of a DPCC encoder can be done in several forms. In this section, two possible structures, which are straightforward, are considered. In the first one, the encoder is built from individual encoders of the component codes. Independently of the blocking factor used for any of the constituent codes, the encoders correspond simply to the original (not blocked) codes. The total number of encoders in this structure is determined by the encoding procedure, row–column or column–row, and the blocking factor M used for one of the component codes. The general structure of this type of encoder is shown in Fig. 2, where x and y denote *horizontal* (h) or *vertical* (v), depending on the encoding order of the component codes.

Consider, for example, a DPCC with component codes C^h (n^h,k^h) and C^v (n^v,k^v), and a blocking factor M for the horizontal code. Then, the encoder for this DPCC, $C^{\otimes} = C^v \otimes C^h_{[M]}$, consists of:

- *Row-Column encoder*: $l_x = k^v$ encoders for C^h and $l_y = Mn^h$ encoders for C^v. In this case, $x \equiv$ "h" and $y \equiv$ "v".
- *Column-Row encoder*: $l_x = Mk^h$ encoders for C^v and $l_y = n^v$ encoders for C^h. In this case, $x \equiv$ "v" and $y \equiv$ "h".

Note that the output sequence c of both encoders contains n^v sequences c_i^v, $1 \leq i \leq n^v$, from the horizontal code C^h and Mn^h sequences c_j^v, $1 \leq j \leq n^h$, of the vertical code C^v.

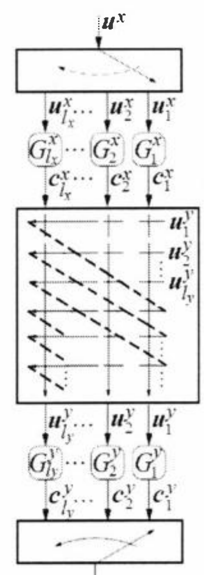

Figure 2: General structure of a DPCC encoder.

The second structure for the encoder of the DPCC can be implemented directly from the resulting generator matrix $G(D)$ given by (7). This construction could result in a structure less complex in terms of the total number of delay elements compared with the previous encoder.

Another possible structure is given in [7] and [8], but the original codes are modified and there are delay elements to implement time division interleaving in the encoders of the component convolutional codes.

In Fig. 3 there is an example of a column-row encoder for a DPCC given by $C^{\otimes} = C^{v} \otimes C^{h}_{[2]}$, where the component codes are $[5,7]_8$ binary convolutional codes with generator matrices given by $G(D) = (1{+}D^2 \;\; 1{+}D{+}D^2)$. The blocking factor is $M = 2$ and the resulting code rate is $R = 2/8$. Hence, the generator matrix of the resulting DPCC is

$$G(D) = G^{v}(D) \otimes G^{h}_{[2]}(D) \tag{18}$$

$$G(D) = (1 + D^2 \quad 1 + D + D^2) \otimes \begin{pmatrix} 1+D & 1+D & 0 & 1 \\ 0 & D & 1+D & 1+D \end{pmatrix} \tag{19}$$

$$G^{\mathrm{T}}(D) = \begin{pmatrix} 1+D+D^2+D^3 & 0 \\ 1+D+D^2+D^3 & D+D^3 \\ 0 & 1+D+D^2+D^3 \\ 1+D^2 & 1+D+D^2+D^3 \\ 1+D^3 & 0 \\ 1+D^3 & D+D^2+D^3 \\ 0 & 1+D^3 \\ 1+D+D^2 & 1+D^3 \end{pmatrix}$$

We wrote the transpose of the matrix $G(D)$ only for (printing) convenience.

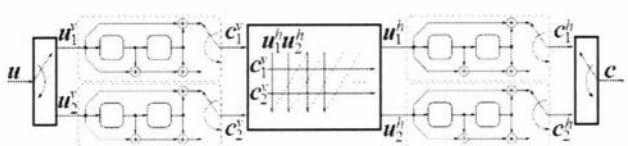

Figure 3: Column–Row encoder for $C^{\otimes} = C^{v} \otimes C^{h}_{[2]}$ using individual encoders.

An encoder for the same DPCC, $C^{\otimes} = C^{v} \otimes C^{h}_{[2]}$, but using the equivalent generator matrix $G(D)$ is illustrated in Fig. 4.

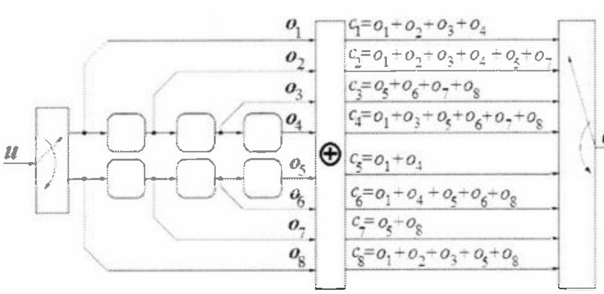

Figure 4: Encoder for $C^{\otimes} = C^{v} \otimes C^{h}_{[2]}$ using the DPCC generator matrix.

IV. Convergence Analysis of Decoding Methods

From the encoding process, DPCCs result to be a special form of serially concatenated convolutional codes, which suggests the application of some sort of iterative decoding method. These methods are based on algorithms that use reliability information as input and generate reliability information at the output (SISO – *soft input soft output*). In general, these methods use some algorithm from the family of MAP algorithms (*maximum a posteriori probability*), and in this paper, the decoding algorithms use SISO modules based on the MAP family implemented in the software developed by W. Schnug and described in [16].

For concatenated codes, the decoding method is adapted to reflect the construction of the code, so that in the case of our DPCC with two component codes, two SISO decoders are used. These decoders interchange reliability information corresponding to the component codes.

The encoding procedure and the structure of the DPCCs allow using an iterative decoding method for serially concatenated codes, or an iterative decoding called *product decoding*, which is also known as MAP filtering [7]. In [13], block diagrams for an iterative serial decoder and for a product decoder are illustrated. Both have the same structure, which is shown in Fig. 5, but they differ in the information they process and share between modules. In both cases, the code-word is used as input but in each method the information exchanged between the SISO modules is different. In serial decoding, only reliability values of one of the component codes are exchanged, whereas in product decoding, values of whole code-words (both component codes) are used.

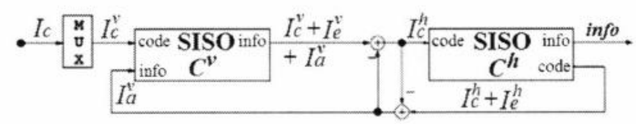

Figure 5: Block diagram of a general iterative decoder for a direct product convolutional code.

In the following, the convergence analysis for both, serial and product, decoding methods is described using the extrinsic information transfer (EXIT) charts.

The method of the EXIT charts [17] allows the analysis of the convergence behavior of iterative decoders. This tool is used to compare the convergence of the serial and product decoding methods of the binary DPCC $C^{\otimes} = C^{v} \otimes C^{h}_{[2]}$ generated by (19) to transmit information over a memoryless AWGN channel using BPSK modulation.

Basically, an EXIT chart describes the flow of extrinsic information through the SISO decoders in the decoding process of a concatenated scheme. Two curves, corresponding to the transfer characteristics (*extrinsic information* vs *a priori information*) of the inner and outer decoders are combined in one plot to obtain an EXIT diagram of the concatenated code. One interesting characteristic of DPCCs is that, because any code-word can be decomposed in code-words of any of the component codes independently of the encoding process, then any of the component codes can be decoded first or second in the decoding process. In the following explanation, we refer to inner and outer codes and decoders to indicate which code is decoded first (inner) and which second (outer) in the serial and product decoding, without the terms being related to the encoding process because what has been explained above.

The inputs to the inner decoders are the channel values and the a priori values I_{ai} of the inner code. I_{ai} correspond to information bits for serial decoding and to code bits for product decoding. The decoder generates extrinsic I_{ei} and channel I_c information. Similar to I_{ai}, I_{ei} is reliability of information bits for serial decoding or code bits for product decoding. In order to obtain the transfer characteristic of the inner decoders, we consider one decoder with inputs from the cannel I_c at a particular E_b/N_0 ratio, a priori values I_{ai} from a source of Gaussian distributed values with variance σ^2_{ai} and mean value μ_{ai}. The distribution p_{ei} of the extrinsic output information is then computed by means of Monte Carlo simulation for a specific parameter combination (I_{ai}, E_b/N_0) similar to [17]. Simulations are based on the following considerations: *i*) it is only necessary to simulate one inner decoder and one outer decoder – for a product convolutional code, all individual decoders corresponding to the inner code are the same and all individual decoders of the outer code are the same; *ii*) for serially concatenated codes in AWGN channel, the extrinsic values coming from the outer decoders are almost Gaussian distributed and the interleavers keep the a priori values fairly uncorrelated over many iterations [17].

The transfer characteristic of the outer decoder is the relationship between the outer extrinsic information I_{eo} and the outer a priori information I_{ao}. Note that I_{eo} does not depend on the channel E_b/N_0. The transfer characteristic of the outer decoder is obtained by simulating the outer decoder with a Gaussian source at the input for different values of the variance and then the distribution of the extrinsic output p_{eo} is calculated using Monte Carlo simulation.

The curves of the EXIT chart in Fig. 6 correspond to the inner and outer decoders of the serial and product decoding schemes for the DPCC $C^{\otimes} = C^{v} \otimes C^{h}_{[2]}$ previously considered, with an $E_b/N_0 = 1.0$ dB.

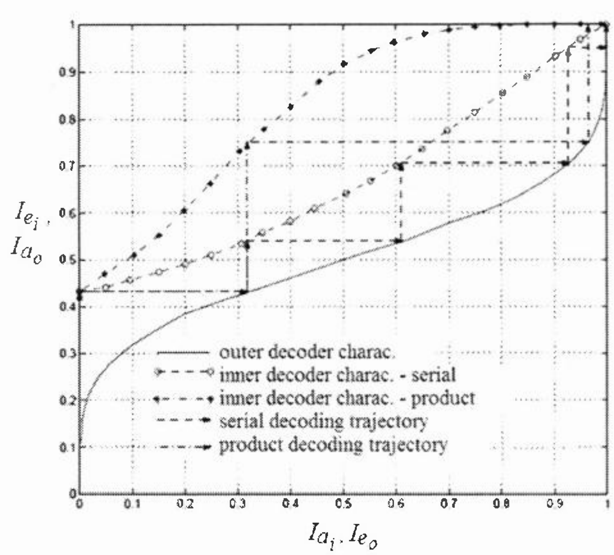

Figure 6: EXIT chart with iterative decoding trajectories at $E_b/N_0 = 1.0$ dB.

The decoding trajectory is drawn, as a zigzag path, between the transfer characteristic curves of the inner and outer decoders. This trajectory represents the exchange of extrinsic information between inner and outer decoders. These paths indicate approximately the number of iterations required for convergence but not necessarily the exact number. This is due to the fact that a decoding trajectory is the result of a simulation based on measurements of mutual information and the assumptions of Gaussian distribution and independency of the a priori values.

From the results shown in Fig. 6 we note that product decoding is faster and requires less iterations than serial decoding for convergence.

V. Blocking Factor and BER Performance

When the code-word (matrix) of a DPCC is formed, the serial transmission of symbols over the channel can be done row-by-row (code-words of C^h) or column-by-column (code-words of C^v). If the inner decoder (first decoder) corresponds to the component code opposite to the transmitted code, i.e., use C^v as inner code for row-by-row transmission or C^h as inner code for column-by-column transmission, then the blocking factor M plays the role of an interleaver size for the DPCC, allowing to spread the channel errors and reducing the correlation among the sequences of both decoders in the decoding stages. Hence, an improvement on the performance of the iterative decoding can be expected by increasing the blocking factor.

To evaluate this transmission-decoding approach, we plot the bit error rate (BER) curves (Fig. 7) of a system using a DPCC, $C^{\otimes} = C^{v} \otimes C^{h}_{[M]}$, for several blocking factors M (1, 5, 8, 10, 25, 50) in AWGN, BPSK modulation and product decoding with 3 iterations.

It is shown that the BER improves significantly from $M = 1$ to $M = 5$, for which the DPCC reaches its product free distance (16), i.e., $d_f = d^v_f\, d^h_B = (5)(5) = 25$. The next increments of M (8, 10) improve the BER up to a certain value after which only marginal improvements are obtained with further increments of M (25, 50). In this case, an adequate value of M to have a good performance without unnecessary complexity is 10.

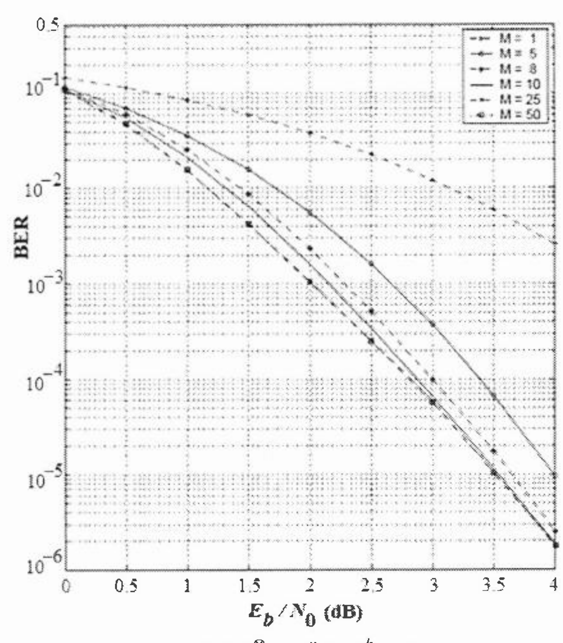

Figure 7: BER for DPCC $C^{\otimes} = C^{v} \otimes C^{h}_{[M]}$ for several values of M using product decoding.

Thus, the blocking factor M is an important parameter, since it not only allows the DPCC to have product free distance, as shown in (17) and (16), but also because greater values than the minimum needed to reach the product distance result in further improvement of the BER performance. Nevertheless, greater values of M also increase the complexity, so it is necessary to determine an adequate value of M, to increase the performance but to keep the complexity as low as possible.

In addition, greater values of M could be used to deal with other types of channel effects, such as burst errors, even though the minimum free distance of the direct product convolutional code will not increase.

VI. MULTIDIMENSIONAL CODES

So far, the DPCC has been created by combining two component convolutional codes, $C = C_1 \otimes C_2$, resulting in code-words (matrices) of 2-dimensions. If this code is used now as one of the component code of another DPCC, let say $C' = C \otimes C_3$, the resulting code will have 3-dimensional code-words. Then, if we use j component codes, the resulting DPCC will have j-dimensional code-words, which can be arranged in a hypercube of dimension j with length in each dimension given by the length of the corresponding component code.

Multidimensional codes could be useful for applications transmitting j-dimensional data, such as: pictures and video images, tomography and magnetic resonance imaging, or MIMO communication systems.

In general, a given property of the 2-dimensional code (e.g., block distance or free distance) can be extended to j-dimensional codes by induction. In addition, the encoder and decoder of a j-dimensional code can be easily derived from the structures discussed in Sections III and IV.

VII. CONCLUSIONS AND FURTHER STUDIES

The algebraic description of convolutional codes allow us to treat them as block codes, and therefore use combining methods developed for block code on convolutional codes. One of these methods is the direct product, which results in the so called direct product convolutional codes. In this paper some further properties and aspects about encoding and decoding of such concatenated scheme are considered, and it is shown that these codes have good properties, such as: product free distance, which can be achieved using blocked codes; several low-complex encoder implementations; a low-complexity decoding procedure; good BER performance in AWGN, and the possibility to construct multidimensional codes by using more than two component codes.

Some other studies can be done, mainly in comparing the complexity and performance of the DPCC: in other channels (e.g., BEC and wireless channels), in applications such as MIMO systems and coded modulation schemes, where it can compete with similar concatenated schemes (PCCC, SCCC, WCC) and using decoding methods for LDPC codes when the parity check matrix results significantly sparse.

REFERENCES

[1] P. Elias, "Error Free Coding", IRE Trans. on Information Theory IT-4, 1954, pp. 29-37.

[2] G. D. Forney, *Concatenated Codes*, MIT Press, Cambridge, Mass., USA, 1966.

[3] E. Blokh and V. Zyablov, "Coding of generalized concatenated codes", Problemy Peredachi Informatsii 10, No. 2, 1974, pp. 45-50.

[4] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Sahnnon limit error-correcting coding and decoding: turbo-codes (1)", Proc. IEEE Int. Conference on Communications, Geneva, Switzerland, May, 1993, pp. 1064-1070.

[5] Benedetto, S.; Montorsi, G., "Iterative decoding of serially concatenated convolutional codes," *Electronics Letters*, vol.32, no.13, pp.1186, 1188, 20 Jun 1996.

[6] S. Höst, R. Johanesson, and V. Zyablov, "A First Encounter with Binary Woven Convolutinal Codes", Proc. of Int. Symposium on Communication Theory and Applications, Lake District, UK, July 13-18, 1997.

[7] J. Lodge, P. Hoeher, and J. Hagenauer, "The Decoding of Multidimensional Codes using Separable MAP Filters", Proc. of the 16th Biennial Symposium on Communications, Queen's University, Kingston, Ontario, Canada, pp. 343-346, May 27-29, 1992.

[8] J. Lodge, r. Young, and P. Guinand, "Separable Concatenated Convolutional Codes: The Structure and Properties of a Class of Codes for Iterative Decoding", European Trans. on Telecommunications 6, No.5, 1995, pp. 35-542.

[9] M. Bossert, C. Medina, and V. Sidorenko, "Encoding and Distance Estimation of Product Convolutional Codes", Proc. of the 2005 IEEE International Symposium on Information Theory (ISIT'05), Adelide, Australia, September, 2005, pp. 1063-1067.

[10] C. Medina, V. Sidorenko, and V. Zyablov, "Error Exponent of Product Convolutional Codes", Problems in Information Transmission, Vol. 42, No. 3, July-September 2006, pp. 167-182.

[11] C. Medina, M. Gabrowska, "Space-Time Product Convolutional Codes with Full Antenna Diversity", Proc. 2006 IEEE Int. Zurich Seminar on Communications, Zurich, Zwitzerland, Feb., 2006. pp. 118-121.

[12] V. Sidorenko, C. Medina, and M. Bossert, "From Block to Convolutional Codes using Block Distances", Proc. of the 2007 IEEE International Symposium on Information Theory (ISIT'07), Nice, France, June, 2007, pp. 2331-2335.

[13] C. Medina, V. Sidorenko. "Product Convolutional Codes: Distance Properties, Coding and Decoding" (in Spanish), Revista I+D Tecnológico, Vol. 6, No. 1 y 2, pp. 15-23, 2009, ISSN 1680-8894

[14] F. Vatta, A. Schiavi, V. Sidorenko, M. Bossert, "Termination and Tailbiting of direct product convolutional codes", IEEE Information Theory Workshop, 2009, pp. 213-217.

[15] V. Sidorenko, M. Bossert, and F. Vatta, "Properties and encoding aspects of direct product convolutional codes", Proc. of the 2012 IEEE International Symposium on Information Theory (ISIT'12), Cambridge, MA, USA, July, 2012, pp. 2351-2355.

[16] W. Schnug, *On Generalized Woven Codes*, Ph.D. Thesis, University of Ulm, Germany, 2002.

[17] S. ten Brink, "Design of Serially Concatenated Codes based on Iterative Decoding Convergence", 2nd Int. Symposium on Turbo Codes, Brest, France, Sept., 2000.