

Punctured Binary Turbo-Codes with Optimized Performance

I. Chatzigeorgiou, M. R. D. Rodrigues, I. J. Wassell
Laboratory for Communication Engineering
Computer Laboratory, University of Cambridge
{ic231, mrd3, ijw24}@cam.ac.uk

R. Carrasco
Communications and Signal Processing Group
Dept. of EE&C Engineering, University of Newcastle
r.carrasco@ncl.ac.uk

Abstract—Although rate-1/3 turbo codes achieve an outstanding performance in additive white Gaussian noise channels, their performance degrades for higher code rates. In this paper it is illustrated that puncturing both the systematic as well as the parity bits can result in a better performance than puncturing the parity bits only. In addition we suggest an interleaver design rule for punctured partially systematic turbo codes, which can further improve performance, without increasing the encoding and decoding complexity.

Keywords- turbo codes; punctured codes; interleaving.

I. INTRODUCTION

When Berrou *et al.* [1, 2] introduced binary turbo codes, they considered puncturing of the parity bits in order to increase the code rate. Hagenauer *et al.* [3] suggested a number of puncturing patterns whereas Açikel *et al.* [4] as well as Kousa and Mugaibel [5] presented a set of guidelines for constructing good puncturing vectors. However these authors focused on puncturing parity bits only. Babich *et al.* [6] claimed that schemes where parity bits are punctured only perform better than schemes where parity and systematic bits are punctured but Land and Hoehner [7] suggested that performance can be improved by puncturing both parity and systematic bits, in the case of rate-1/2 turbo codes.

In this paper, we first describe the turbo encoder and decoder when puncturing is applied. A binary input additive white Gaussian noise (AWGN) channel is assumed and we focus on punctured turbo codes having rates of 1/2 and 2/3. We provide expressions for the calculation of their input-redundancy enumerating function and we illustrate with a specific example why performance is improved by puncturing both the systematic and parity bits as opposed to puncturing only the parity bits. Furthermore, we propose a design rule for the interleaver of partially systematic turbo codes, which further boosts the performance of the punctured turbo code.

II. PUNCTURED BINARY TURBO CODES

The binary turbo encoder is a parallel concatenation of two recursive systematic convolutional (RSC) encoders of rate 1/2, as described in [1, 2]. The source bits are input to the first constituent encoder, while the second encoder is fed by an interleaved version of the original data. The interleaver is assumed to be pseudorandom and has a size of N bits. The output of the turbo encoder consists of the systematic bits of the first encoder, the parity bits of the first encoder and the parity bits of the second encoder.

In order to increase the code rate of 1/3, puncturing is applied. A value of 0 in a puncturing vector implies that the corresponding bit is punctured. Each puncturing vector of length

N is based on a pattern of length ℓ , which is repeated N/ℓ times. The systematic stream, the parity stream of the first encoder and the parity stream of the second encoder are punctured using puncturing vectors, based on patterns \mathbf{P}_u , \mathbf{P}_p and $\mathbf{P}_{p'}$ respectively, and the resultant codewords are transmitted over the channel.

The turbo decoder consists of two soft-input soft-output decoders. Both decoders use a-priori information to produce soft estimates of the source bits by processing the log-likelihood ratios (LLRs) of the receive systematic and parity bits. When the code rate is higher than 1/3, zero values are inserted at the positions where puncturing took place.

The first decoder uses a-priori information to produce soft estimates of the transmitted bits by processing the LLRs of the receive systematic bits and the LLRs of the receive parity bits of the first RSC encoder. Extrinsic information is extracted from the soft estimates of the source bits and acts as a-priori information for the second decoder. Similarly, the second decoder processes the LLRs of the received interleaved systematic bits as well as the LLRs of the received parity bits of the second RSC encoder, to produce better estimates of the source bits as well as extrinsic information, which will be used as a-priori information by the first decoder at the next iteration.

Decoding algorithms are usually based either on the Maximum A-Posteriori algorithm, also known as the BCJR algorithm [8], or the soft output Viterbi algorithm (SOVA) [9].

III. PARTIALLY SYSTEMATIC TURBO CODES

In this work we focus on turbo codes in the form of parallel concatenated convolutional codes (PCCCs). Both constituent codes are terminated and the termination bits are also punctured.

As described in [7], turbo codes can be systematic, partially systematic (some of the source bits are punctured) or non-systematic. Hagenauer *et al.* [3], considered systematic turbo codes and suggested a number of puncturing patterns in order to achieve various code rates. We consider partially systematic turbo codes whose puncturing pattern is homogeneous, i.e., the puncturing bits are uniformly distributed among the systematic and the parity streams of the turbo encoder.

A. Weight Enumerating Functions

Before proceeding to the puncturing of rate-1/3 turbo codes, we first present expressions describing their performance [10, 11]. These are applicable to terminated turbo codes, where the memory size is much smaller than the interleaver size. The constituent convolutional codes are assumed to be identical.

In order to analyze the performance of turbo codes, the expressions that describe the constituent convolutional codes

need to be modified to describe the equivalent block codes. The conditional weight enumerating function (CWEF) of the equivalent block code can be expressed as:

$$A^c(w, Z) = \sum_j A_{w,j} Z^j \quad (1)$$

where $A_{w,j}$ denotes the number of codewords generated by an input information word of Hamming weight w , whose parity bits have Hamming weight j . The CWEF of the PCCC encoder using a uniform interleaver of length N is given by:

$$A^{PCCC}(w, Z) = \frac{(A^c(w, Z))^2}{\binom{N}{w}} \quad (2)$$

From (2) we obtain the input-redundancy weight enumerating function (IRWEF) of the PCCC code:

$$A^{PCCC}(W, Z) = \sum_w W^w A^{PCCC}(w, Z) \quad (3)$$

from which the bit error rate (BER) can be derived. By setting $W=H$ and $Z=H$, the weight enumerating function (WEF) assumes the form:

$$B^{PCCC}(H) = A^{PCCC}(W=H, Z=H) = \sum_h B_h H^h \quad (4)$$

where B_h is the number of codewords with Hamming weight h ($h=w+j$). The WEF is related to the word error probability.

In the case of a terminated turbo code of rate $1/3$, an input stream of N bits generates three outputs of length $N+v$ bits each, where v is the memory size of the constituent encoder. Expressions (1) and (2) do not provide accurate results, since they assume that the length of each output is equal to the interleaver size N . Therefore we need to extend these expressions to take into account the terminating bits.

The conditional weight enumerating function (CWEF) of the equivalent block code for the case of a terminated convolutional code can be rewritten as:

$$A^c(w, D, Z) = \sum_d \sum_j A_{w,d,j} D^d Z^j \quad (5)$$

where $A_{w,d,j}$ denotes the number of codewords generated by an input information word of Hamming weight w , whose systematic bits (including the systematic terminating bits) have Hamming weight d whereas the parity bits (including the parity terminating bits) have Hamming weight j . The reason for the introduction of variable D is due to the fact that the weight of the systematic output of the code is not necessarily equal to the weight of the input information word. The additional v terminating bits could further increase the weight of the systematic output ($w \leq d$).

The PCCC encoder transmits the $N+v$ systematic bits of the first constituent code only (see fig.1). For rate- $1/3$ turbo codes, the CWEF of the first constituent code $A^{C1}(w, D, Z)$ is given by (5). If we set $D=1$ in $A^{C1}(w, D, Z)$ we obtain the CWEF of the second constituent code, i.e., $A^{C2}(w, Z) = A^{C1}(w, D=1, Z)$. However, in the case of punctured turbo codes, the CWEF of the second constituent code also needs to be calculated.

The CWEF, IRWEF and WEF of the PCCC encoder, using a uniform interleaver of length N , assume the form:

$$A^{PCCC}(w, D, Z) = \frac{A^{C1}(w, D, Z) \cdot A^{C2}(w, Z)}{\binom{N}{w}} \quad (6)$$

$$A^{PCCC}(W, D, Z) = \sum_w W^w A^{PCCC}(w, D, Z) \quad (7)$$

$$B^{PCCC}(H) = A^{PCCC}(W=1, D=H, Z=H) = \sum_h B_h H^h \quad (8)$$

where the codeword Hamming weight h is equal to $d+j$.

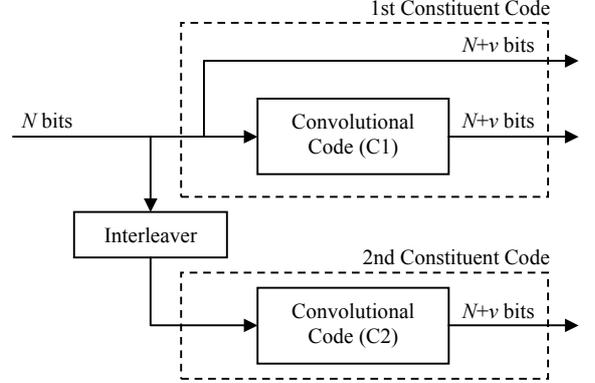


Figure 1. Schematic of a turbo code in the form of PCCC.

B. Example: Puncturing a parallel concatenation of RSCs(1, 5/7)

We illustrate the effect of puncturing by an example. The PCCC of our example uses two four-state, rate- $1/2$, RSC codes with recursive polynomial 7 and forward polynomial 5, i.e. RSC(1, 5/7). For this example we assume that the interleaver is only 4 bits long for simplicity. Similar results are observed for larger interleaver sizes.

Initially, we consider a rate- $1/3$ systematic PCCC. The CWEF of the equivalent rate- $1/2$ block code for the first constituent code is:

$$\begin{aligned} A^{C1}(0, D, Z) &= 1 \\ A^{C1}(1, D, Z) &= 2D^2Z^4 + D^3(Z^2 + Z^4) \\ A^{C1}(2, D, Z) &= D^2Z^4 + D^3(Z^2 + 3Z^4) + D^4Z^2 \\ A^{C1}(3, D, Z) &= 2D^3Z^2 + D^4(Z^2 + Z^4) \\ A^{C1}(4, D, Z) &= D^6Z^4 \end{aligned} \quad (9)$$

It is observed that the free distance of the convolutional code is 5. By setting $A^{C2}(w, Z) = A^{C1}(w, D=1, Z)$ and substituting into equations (6)-(8) we obtain the WEF of the PCCC code:

$$\begin{aligned} B^{PCCC}(H) &= \\ &= 1 + 2.083 \cdot H^7 + 1.917 \cdot H^8 + 3.167 \cdot H^9 + 3.833 \cdot H^{10} + \\ &= 2.750 \cdot H^{11} + 0.250 \cdot H^{12} + H^{14} \end{aligned} \quad (10)$$

whose free distance is equal to 7.

Hagenauer *et al* [3], used the puncturing patterns $\mathbf{P}_u=[11]$, $\mathbf{P}_p=[10]$ and $\mathbf{P}_p=[01]$, to obtain a systematic turbo code of rate $1/2$. Patterns \mathbf{P}_u and \mathbf{P}_p define the code rate of the first constituent code, which is equal to $2/3$. Taking into account puncturing, the CWEF of the rate- $2/3$ equivalent block code for the first constituent code is:

$$\begin{aligned} A^{C1}(0, D, Z) &= 1 \\ A^{C1}(1, D, Z) &= 2D^2Z^2 + D^3(1 + Z^2) \\ A^{C1}(2, D, Z) &= D^2Z^2 + 4D^3Z^2 + D^4 \\ A^{C1}(3, D, Z) &= D^3(1 + Z^2) + 2D^4Z^2 \\ A^{C1}(4, D, Z) &= D^6Z^2 \end{aligned} \quad (11)$$

We observe that the free distance of the code is reduced from 5 to 3. The CWEF of the equivalent block code for the second constituent code is:

$$\begin{aligned} A^{C^2}(0, Z) &= 1 \\ A^{C^2}(1, Z) &= 4Z^2 \\ A^{C^2}(2, Z) &= 1 + 5Z^2 \\ A^{C^2}(3, Z) &= 2 + 2Z^2 \\ A^{C^2}(4, Z) &= Z^2 \end{aligned} \quad (12)$$

Consequently the WEF of the rate 1/2 systematic PCCC code is:

$$\begin{aligned} B^{PCCC}(H) &= \\ &1 + 0.5 \cdot H^3 + 0.333 \cdot H^4 + 2.667 \cdot H^5 + 4.667 \cdot H^6 + \\ &4.833 \cdot H^7 + H^8 + H^{10} \end{aligned} \quad (13)$$

so puncturing has reduced the free distance of the systematic turbo code from 7 to 3.

We observe that the free distance of the turbo code is mainly determined by the free distance of the first constituent code, i.e. $d_{free}=3$, since puncturing of the second constituent code leads to 3 parity words with zero Hamming weight. We also observe that the puncturing rate of the first constituent code is 1/4, i.e. 1 in every 4 bits is punctured ($\mathbf{P}_u=[11]$, $\mathbf{P}_p=[10]$), while the puncturing rate of the second constituent code is 1/2 ($\mathbf{P}_{p'}=[01]$).

We now consider the case of partially systematic turbo codes. If the puncturing patterns $\mathbf{P}_u=[110]$, $\mathbf{P}_p=[101]$ and $\mathbf{P}_{p'}=[011]$ are used to obtain a turbo code of rate 1/2, the first constituent code has a code rate of 3/4 and its CWEF has the form:

$$\begin{aligned} A^{C^1}(0, D, Z) &= 1 \\ A^{C^1}(1, D, Z) &= Z^3 + D^2(2Z^2 + Z^3) \\ A^{C^1}(2, D, Z) &= 2D^2(Z + Z^2 + Z^3) \\ A^{C^1}(3, D, Z) &= D^2(Z + 2Z^2) + D^4Z \\ A^{C^1}(4, D, Z) &= D^4Z^4 \end{aligned} \quad (14)$$

Although the puncturing rate of the first constituent code was increased from 1/4 (systematic punctured code) to 1/3 (partially systematic code), the free distance of the code did not change.

Furthermore, as seen in table I, the puncturing rate of the first constituent code has been increased by 1 bit every 12 codeword bits but the puncturing rate of the second code has been reduced by 2 bits every 12 parity bits. Therefore, it is more likely that the number of parity words with zero weight, generated by the second encoder, is reduced or even eliminated.

Indeed, the CWEF of the equivalent block code for the second constituent code is:

$$\begin{aligned} A^{C^2}(0, Z) &= 1 \\ A^{C^2}(1, Z) &= Z + 3Z^3 \\ A^{C^2}(2, Z) &= 5Z^2 + Z^4 \\ A^{C^2}(3, Z) &= 3Z + Z^3 \\ A^{C^2}(4, Z) &= Z^2 \end{aligned} \quad (15)$$

which contains no parity words of zero weight. The WEF of the rate 1/2 partially systematic PCCC code is equal to:

$$\begin{aligned} B^{PCCC}(H) &= \\ &1 + H^4 + 3.667 \cdot H^5 + 3.667 \cdot H^6 + 4 \cdot H^7 + \\ &1.333 \cdot H^8 + 0.333 \cdot H^9 + H^{10} \end{aligned} \quad (16)$$

The free distance of the partially systematic turbo code is 4, which is equivalent to a better word error performance, compared to the rate 1/2 systematic turbo code.

The puncturing rate of each constituent code for the case of a rate k/n systematic or partially systematic PCCC obtained from a rate 1/3 PCCC code, are illustrated in table II.

TABLE I. PUNCTURING RATES OF THE CONSTITUENT CODES OF A RATE-1/2 PCCC

Constituent Code	Puncturing Rate		Variation
	Rate 1/2 Systematic PCCC	Rate 1/2 Part. Sys. PCCC	
First	$\frac{1}{4} \left(= \frac{3}{12} \right)$	$\frac{2}{6} \left(= \frac{4}{12} \right)$	$+\frac{1}{12}$
Second	$\frac{1}{2} \left(= \frac{6}{12} \right)$	$\frac{1}{3} \left(= \frac{4}{12} \right)$	$-\frac{2}{12}$

TABLE II. PUNCTURING RATES OF THE CONSTITUENT CODES OF A RATE- k/n PCCC

Constituent Code	Puncturing Rate		Variation
	Rate k/n Systematic PCCC	Rate k/n Part. Sys. PCCC	
First	$\frac{3k-n}{4k}$	$\frac{3k-n}{3k}$	$+1 \cdot \left(\frac{3k-n}{12k} \right)$
Second	$\frac{3k-n}{2k}$	$\frac{3k-n}{3k}$	$-2 \cdot \left(\frac{3k-n}{12k} \right)$

C. Simulation Results

In fig.2, the performances of rate 1/2 systematic and partially systematic PCCCs are compared. Both use a random interleaver of size 1000 bits. The systematic PCCC uses the puncturing patterns $\mathbf{P}_u=[11]$, $\mathbf{P}_p=[10]$ and $\mathbf{P}_{p'}=[01]$, whereas the partially systematic PCCC applies the puncturing patterns $\mathbf{P}_u=[110]$, $\mathbf{P}_p=[101]$ and $\mathbf{P}_{p'}=[011]$. The increase of the puncturing rate of the first encoder causes degradation to the performance of the first decoder, which affects the overall PCCC performance. However, as the number of iterations increases, the turbo decoder approaches the performance of a maximum likelihood (ML) decoder and the benefit of reducing the puncturing rate of the second encoder is noticeable.

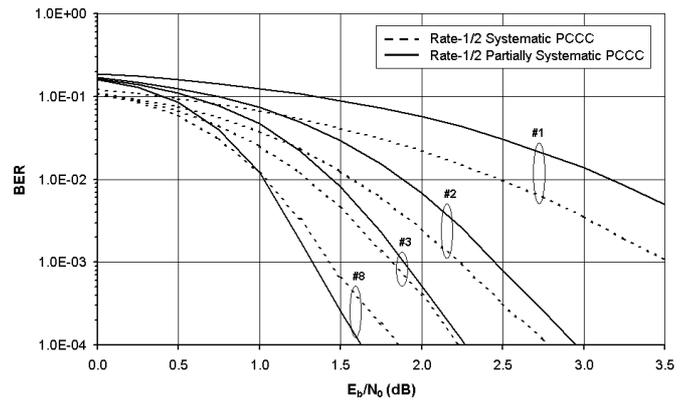


Figure 2. Bit error rate of rate 1/2 systematic and partially systematic PCCCs after 1, 2, 3 and 8 iterations. The interleaver size is $N=1,000$.

In fig.3, the performance of rate 2/3 systematic and partially systematic PCCC after 8 iterations has been included. The systematic code uses the puncturing patterns $\mathbf{P}_u=[1111]$, $\mathbf{P}_p=[1000]$ and $\mathbf{P}_{p'}=[0010]$, whereas the partially systematic code applies the puncturing patterns $\mathbf{P}_u=[1001]$, $\mathbf{P}_p=[1010]$ and $\mathbf{P}_{p'}=[0101]$. The rate 2/3 systematic PCCC performs better at low E_b/N_0 values, however the partially systematic PCCC finally outperforms the systematic PCCC when the E_b/N_0 ratio is greater than 2.65 dB.

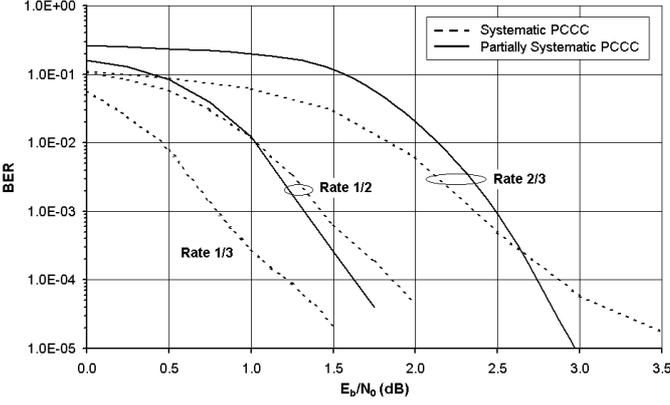


Figure 3. Bit error rate of rate 1/2 and 2/3 systematic and partially systematic PCCCs after 8 iterations. The interleaver size is $N=1,000$.

IV. INTERLEAVER DESIGN

The turbo decoder consists of two soft-input soft-output decoders that exchange information in order to produce better estimates of the transmitted bits. The performance of a constituent decoder depends on the IRWEF of the corresponding constituent code, which is a function of both the systematic and the parity weight $A^C(W,D,Z)$, and consequently the WEF $B^C(H) = A^C(W=I, D=H, Z=H)$.

A. Puncturing Pattern of the Constituent Codes

Assume that $(\mathbf{P}_u, \mathbf{P}_p)$ and $(\mathbf{P}_u', \mathbf{P}_p')$ are two pairs of puncturing patterns that produce good punctured convolutional codes. In the case of punctured turbo codes, pair $(\mathbf{P}_u, \mathbf{P}_p)$ is applied to the first constituent code and \mathbf{P}_p' to the second constituent code. At the receiver, the receive systematic bits and the receive parity bits of the first constituent decoder conform to the puncturing pair $(\mathbf{P}_u, \mathbf{P}_p)$, whereas the interleaved receive systematic bits and the receive parity bits of the second constituent decoder conform to a puncturing pair $(\mathbf{P}_u', \mathbf{P}_p')$.

If the turbo code is systematic, $\mathbf{P}_u = [11\dots 1]$. Since no systematic bits are punctured, the second constituent decoder conforms to puncturing pair $(\mathbf{P}_u, \mathbf{P}_p)$ since $\mathbf{P}_u' = \mathbf{P}_u$, which guarantees that performance of the second constituent code will also be good. In the case of a partially systematic turbo code, the second decoder uses the parity bits selected by the puncturing pattern of the second encoder \mathbf{P}_p' , as well as the interleaved systematic bits. Although the receive systematic bits conform to pattern \mathbf{P}_u , the interleaved systematic bits conform to \mathbf{P}_u' , which is not necessarily equal to \mathbf{P}_u . Therefore, it is not guaranteed that the second decoder will converge as quickly as it would, if the puncturing pair was $(\mathbf{P}_u, \mathbf{P}_p)$.

Returning to our previous example, the WEF of the second encoder of a rate-1/2 systematic PCCC, using an interleaver of size 4 bits, is:

$$B^{C2}(H) = 1 + H^2 + 6 \cdot H^3 + 5 \cdot H^4 + 2 \cdot H^5 + H^6 \quad (17)$$

More specifically, the WEF of the second encoder of a rate-1/2 partially systematic PCCC, using an interleaver of size 4 bits is illustrated in table III. The puncturing patterns used are $\mathbf{P}_u=[110]$ and $\mathbf{P}_p'=[011]$. The random interleaver could take one of $4! = 24$ possible configurations. A configuration of the form $A_1A_2A_3A_4$ maps the first bit to position A_1 , the second bit to position A_2 , etc. According to \mathbf{P}_u , the 3rd systematic bit should be punctured, however the bit at position A_3 is eventually punctured. In our example, the six last configurations would lead to poor performance.

TABLE III. WEF OF THE 2ND ENCODER OF A RATE-1/2 PARTIALLY SYSTEMATIC PCCC USING AN INTERLEAVER OF SIZE 4

Interleaver Configurations	Weight Enumerating Function
1234, 1432, 2134, 2431, 4132, 4231, 1324, 1423, 2314, 2413, 3124, 3214, 3412, 3421, 4123, 4213, 4312, 4321	$1 + H^2 + 6H^3 + 5H^4 + 2H^5 + H^6$
1243, 1342, 2143, 2341, 3142, 3241	$1 + H + 4H^3 + 7H^4 + 3H^5$

B. Interleaver Design Rule

In order to tackle this problem, the interleaver needs to be redesigned in such a way that pattern \mathbf{P}_u eventually punctures the same positions of the systematic and the interleaved systematic streams. This can be achieved if the interleaver of size N restricts bit u_k to lie in positions that differ by multiples of ℓ from position k , i.e., $k \pm \ell, k \pm 2\ell$ and so on, where ℓ is the length of the puncturing pattern. We call this interleaver an ℓ -bit periodic random interleaver.

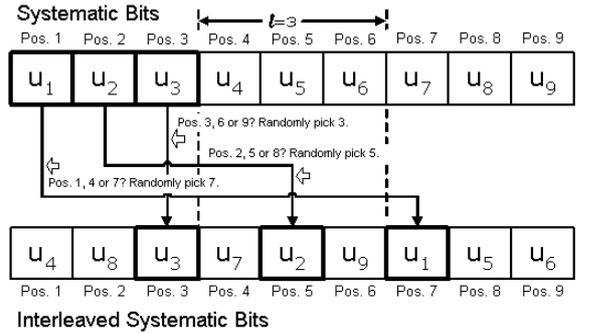


Figure 4. Example of a 3-bit periodic random interleaver

In our example, the length of each puncturing pattern is 3, therefore a 3-bit periodic random interleaver is used. In fig.4, a schematic representation of such an interleaver is given. The bits that occupy positions 1, 4 and 7 in the systematic stream are mapped to one of positions 1, 4 and 7 in the interleaved systematic stream with equal probability. The same logic applies to all positions. An interleaver following this design rule could only use configurations 1234 and 4231 (see table III).

A complete example of two rate-1/2 turbo codes is presented in fig.5. In fig.5(a) the effect of a random interleaver is shown while in fig.5(b) the effect of a 3-bit periodic random interleaver is illustrated. At the receiver, zeroes are inserted at the punctured positions of the receive systematic stream. In the case of random interleaving, zeroes in the interleaved receive systematic stream appear at random positions. In the case of periodic random interleaving, zeroes in the interleaved receive systematic stream appear in the same positions as in the receive systematic stream. Therefore it is guaranteed that pattern \mathbf{P}_u specifies the puncturing period and the puncturing positions in both the systematic and the interleaved systematic streams.

C. Simulation Results

The performance of a rate-1/2 partially systematic turbo code using a 3-bit periodic random interleaver is shown in fig.6. It can be observed that it outperforms the partially systematic turbo code using a random bit interleaver right from the first iteration. In fig.7 the performance of rate 1/3, 1/2 and 2/3 turbo codes is

presented. At a rate 1/2, the proposed scheme performs only 0.3 dB worse than the rate 1/3 turbo code at a BER=10⁻⁴.

V. CONCLUSIONS AND FUTURE WORK

We illustrated that terminated partially systematic PCCCs perform better than terminated systematic PCCCs for rates 1/2 and 2/3. Further investigation needs to be made in order to explore whether there is a rate after which the performance of partially systematic PCCCs degrades, compared to systematic PCCCs. It is suspected that the increased puncturing rate of the first encoder considerably reduces the free distance of the code. Since it is the characteristics of the first constituent code that mainly affect the overall performance of the turbo code, it is expected that, for high code rates, the additional parity bits provided by the second encoder will not suffice to keep the value of the free distance of the turbo code as high as possible.

An optimized method was also presented according to which, partially systematic turbo codes of rate 1/2 and 2/3 with improved performance are generated. The proposed method outperforms the systematic turbo codes as well as the partially systematic turbo codes, which use a random bit interleaver, after a number of iterations. Future work will include investigating optimal puncturing patterns so as to achieve higher rate codes with good performance.

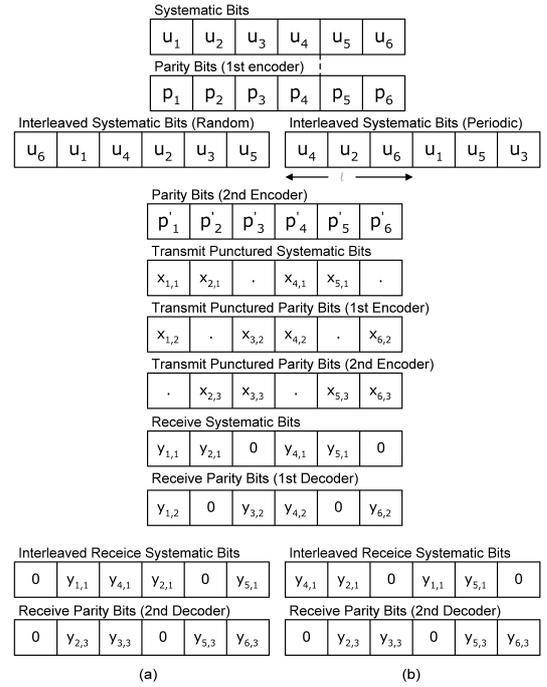


Figure 5. Effect of interleaving to the receive systematic bits of a rate 1/2 partially systematic turbo code. In (a) a random bit interleaver is used while in (b) a random 3-bit periodic interleaver is used.

REFERENCES

- [1] C. Berrou, A. Glavieux, "Near Optimum Error Correcting Coding and Decoding: Turbo-Codes", *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261-1271, Oct. 1996.
- [2] C. Berrou, A. Glavieux, P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-Codes", in *Proc. ICC '93*, Geneva, Switzerland, pp. 1064-1070, May 1993.
- [3] J. Hagenauer, E. Offer, L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes", *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429-445, March 1996.
- [4] Ö. Açikel, W. E. Ryan, "Punctured Turbo-Codes for BPSK/QPSK Channels", *IEEE Trans. Commun.*, vol. 47, no. 9, pp. 1315-1323, September 1999.
- [5] M. A. Kousa, A. H. Mugaibel, "Puncturing Effects on Turbo Codes", *IEE Proc. Commun.* vol. 149, no. 3, pp. 132-138, June 2002.
- [6] F. Babich, G. Montorsi, F. Vatta, "Design of rate-compatible punctured turbo (RCPT) codes" *JCC 2002*, New York, vol. 3, pp. 1701-1705, 2002.
- [7] I. Land, P. Hoeher, "Partially Systematic Rate 1/2 Turbo Codes", *Proc. Int. Symp. Turbo Codes*, Brest, France, pp. 287-290, September 2000.
- [8] L. R. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal Decoding of Linear Codes for Minimising Symbol Error Rate", *IEEE Trans. Information Theory*, pp. 294-287, Mar. 1974.
- [9] J. Hagenauer, P. Hoeher, "A Viterbi Algorithm with Soft-Decision Outputs and its Applications", *Proc. Globecom '89*, vol. 3, pp. 1680-1686, November 1989.
- [10] S. Benedetto, G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes", *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 409-428, March 1996.
- [11] S. Benedetto, G. Montorsi, "Design of Parallel Concatenated Convolutional Codes", *IEEE Trans. Commun.*, vol. 44, no. 5, pp. 591-600, May 1996.

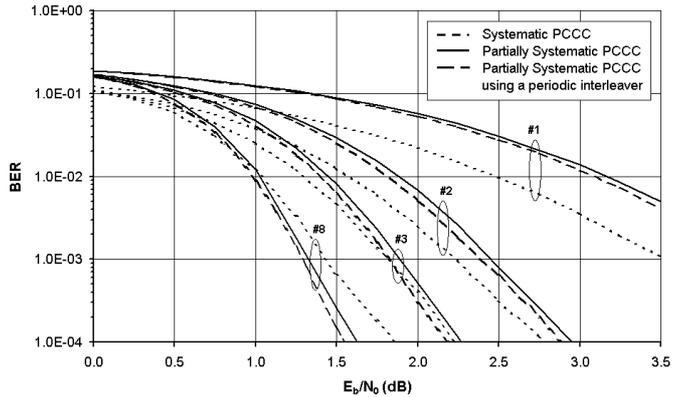


Figure 6. BER comparison of rate 1/2 PCCCs after 1, 2, 3 and 8 iterations. The interleaver size is N=1,000.

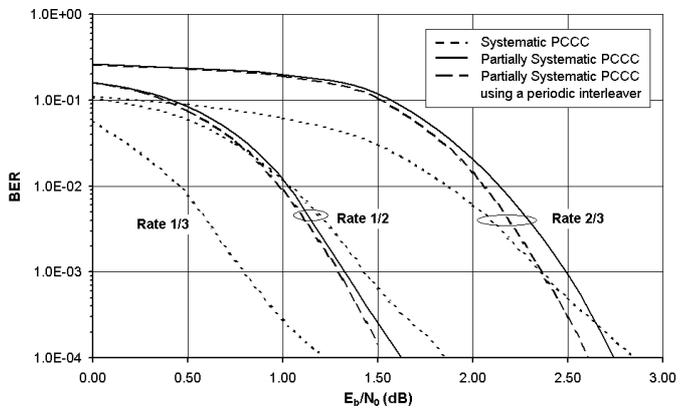


Figure 7. Performance comparison of rate 1/3, 1/2 and 2/3 turbo codes after 8 iterations. The interleaver size is N=1,000.