Design of Parallel Concatenated Convolutional Codes

Sergio Benedetto, *Senior Member, IEEE* and Guido Montorsi, *Member, IEEE*

***Abstract*—A parallel concatenated convolutional coding scheme consists of two constituent systematic convolutional encoders linked by an interleaver. The information bits at the input of the first encoder are scrambled by the interleaver before entering the second encoder. The codewords of the parallel concatenated code consist of the information bits followed by the parity check bits of both encoders. Parallel concatenated codes (turbo codes), decoded through an iterative decoding algorithm of relatively low complexity, have recently been shown to yield remarkable coding gains close to theoretical limits. In this paper, we characterize the separate contributions that the interleaver length and constituent codes give to the overall performance of the parallel concatenated code, and present some guidelines for the optimal design of the constituent convolutional codes.**

***Index Terms*— Iterative decoding, concatenated codes, turbo codes.**

I. Introduction

THE RECENTLY introduced "turbo codes" [1]–[3] have raised great interest in the coding community with their astonishing performance. They are *parallel concatenated convolutional codes* (PCCC's) whose encoder is formed by two or more *constituent* systematic recursive convolutional encoders joined by an interleaver. The input information bits feed the first encoder and, after having been scrambled by the interleaver, enter the second encoder. The codeword of the parallel concatenated code consists of the information bits followed by the parity check bits of both encoders.

Since the interleaving length is normally very large, maximum likelihood decoding would be of astronomical complexity and is, thus, out of question. The proposed suboptimal decoder [1] implements an iterative algorithm whose central core is a maximum *a posteriori* symbol-by-symbol decoder. By increasing the number of iterations, a bit error probability of 10^{-4} has been obtained by simulation at E_b/N_0 as low as -0.15 dB [4].

Since the successful proposal of turbo codes, neither a good theoretical explanation of the code behavior and performance nor an adequate comprehension of the role and relative importance of the PCCC ingredients (constituent codes and interleaver) have yet appeared.

In [5] and [6], we have proposed for the first time a method to obtain analytical upper bounds to the bit error probability of maximum-likelihood decoded PCCC's and showed a wide range of results that help to understand this new coding scheme.

In this paper, we will show how the interleaver and the constituent codes (CC's) contribute to the good performance of PCCC's and propose design guidelines to find "optimum" CC for a given memory. We will define a new parameter, called *effective free distance*, that strongly influences the performance of a PCCC, show how to maximize it and give a table of the best rate—1/2 CC's with number of states ranging from 2–32. The optimization criterion is the minimization of the bit error probability.

In Section II, we will briefly recall the results from [6] that are necessary to understand what follows. In Section III, we will prove that in order to achieve large interleaver gains, PCCC's need recursive convolutional encoders. Section IV shows analytically, and with the support of simulations, that PCCC performance is strictly related to the effective free distance of the CC's, and presents a theorem on how to obtain a large value for it. The results are then applied to find the "best" CC and to present bit error probability bounds for the optimized PCCC. Finally, a summary of the main results of the paper will be presented in Section V.

II. Summary of Known Results

The block diagram of a PCCC is shown in Fig. 1. Two equal linear systematic convolutional encoders with rate 1/2 and memory ν (so that the number of states is $M = 2^\nu$) are linked through an interleaver of length N so that every block of N information bits entering the second encoder is just a permuted version of the block that entered the first encoder. The PCCC codeword is then formed by adding to the input information bit the parity-check bits generated by the first and second encoder. In this case, the PCCC is a rate 1/3 linear PCCC.[1]

In [5], [6], and [8], we have introduced the notion of *uniform interleaver* of length N, defined as a probabilistic device that maps a given input sequence of length N and weight w into all distinct $\binom{N}{w}$ permutations of it with equal probability $1/\binom{N}{w}$. The uniform interleaver has proven to be a very useful tool, in that it makes independent the weight distributions of the parity-check bits generated by the first and second encoder and permits a relatively easy evaluation of the weight distribution of the PCCC. Moreover, it permits estimation of an "average" interleaver gain, independent of the particular interleaver used in a practical scheme and, thus, decouples the roles played

Paper approved by T. Aulin, the Editor for Coding and Communications Theory of the IEEE Communications Society. Manuscript received March 20, 1995; revised June 13, 1995 and September 27, 1995. This work was supported by Agenzia Spaziale Italiana. This paper was presented in part at GLOBECOM'95, Singapore, November 1995.

The authors are with the Dipartimento di Elettronica, Politecnico di Torino, 10129 Torino, Italy.

Publisher Item Identifier S 0090-6778(96)03351-X.

[1] Several generalizations are possible, like having more than two CC, having them different, increasing the overall rate by puncturing, etc. [7]. We will consider here only the case of two identical CC's.

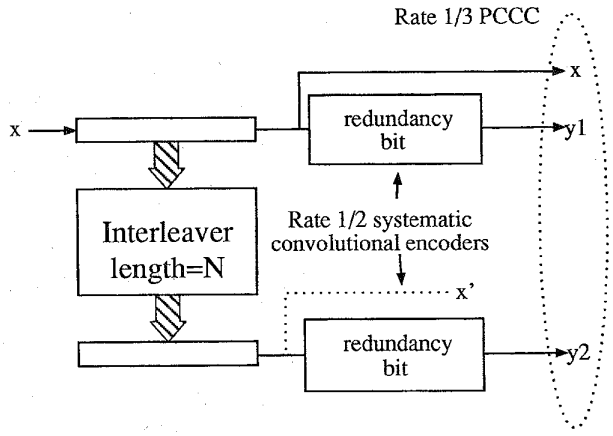


Fig. 1. Encoder structure of a PCCC.

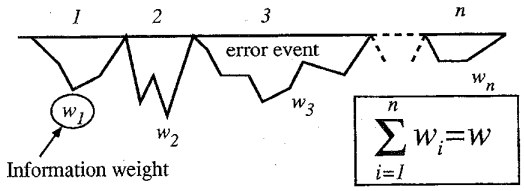


Fig. 2. Example of a sequence belonging to $A(w, Z, n)$.

by the interleaver and by the CC in determining the overall PCCC performance.

We will enumerate here without derivations the results from [6] that will be needed in the following. When not explicitly stated, they refer to PCCC using a uniform interleaver.

1) Performance evaluation:

a) For N much larger than the memory of the CC, the performance of a PCCC is almost identical to that of the *equivalent* $(3N, N - 2\nu)$ parallel concatenated block code C_P whose codewords are sequences of the PCCC of length $3N$ that start from and end at the zero states of both CC's.

b) Defining the conditional weight enumerating function of the equivalent block code as

$$A^{C_P}(w, Z) = \sum_j A^{C_P}_{w,j} Z^j$$

where Z is a dummy variable and $A^{C_P}_{w,j}$ is the number of codewords with parity-check weight j and information bit weight w, an upper bound to the bit error probability of a PCCC for transmission over additive Gaussian noise channels and maximum likelihood decoding is

$$P_b(e) \le \sum_{w=1}^{N} \frac{w}{N} W^w A^{C_P}(w, Z)\Big|_{W=Z=e^{-R_c E_b/N_0}} \quad (1)$$

where R_C is the rate of the code.

c) The conditional weight enumerating function $A^{C_P}(w, Z)$ is obtained simply from the conditional weight enumerating functions $A^C(w, Z)$ of the two $(2N, N - \nu)$ constituent block codes equivalent to the constituent CC as

$$A^{C_P}(w, Z) = \frac{[A^C(w, Z)]^2}{\binom{N}{w}}. \quad (2)$$

d) Finally, the conditional weight enumerating function $A^C(w, Z)$ of the equivalent constituent block code can be derived from the standard transfer function of the CC.

2) Over a wide range of signal-to-noise ratios (SNR's), say, from 1.5 dB down to what can be simulated, the upper bound (1) for maximum likelihood decoding and uniform interleaving can be approached very closely by a PCCC making use of the same CC, of a randomly selected fixed interleaver and suboptimum iterative decoding.

3) For interleaver length N significantly larger than the CC memory, the interleaver performance gain is a reduction by a factor $1/N$ of the bit error probability.

III. Role of the Interleaver and Constituent Codes

In [6], we have shown how to compute the conditional weight enumerating function $A^{C_P}(w, Z)$ in (1) exactly. In this section, we will develop an approximation valid for large N, which will permit pushing the analysis further so we can draw conclusions useful for code design.

Consider a rate 1/2 convolutional code with memory ν, and its equivalent $(2N, N - \nu)$ block code whose codewords are all sequences of length $2N$ of the convolutional code starting from and ending at the zero state.[2] The codewords of the equivalent block code are concatenations of error events of the convolutional code.

Let

$$A(w, Z, n) = \sum_j A_{wjn} Z^j \quad (3)$$

be the parity-check enumerating function of the sequences of the convolutional code generated by concatenating n error events with total information weight w (see Fig. 2). In (3), A_{wjn} is the number of codewords with information bit weight w, parity-check weight j and number of concatenated error events n. For N much larger than the memory of the convolutional code,[3] the conditional weight enumerating function $A^C(w, Z)$ of the equivalent block code can be approximated by

$$A^C(w, Z) \sim \sum_{n=1}^{n_{\max}} \binom{N}{n} A(w, Z, n) \quad (4)$$

where $n_{\max}$, the largest number of error events generated by a weight w information sequence, is a function of w which depends on the encoder.

[2]For simplicity, we limit ourselves to PCCC's that employ the same rate 1/2 convolutional code for both CC's. The extension to rate $1/n$ and possibly different CC's is straightforward.

[3]This assumption permits neglecting the length of the error events compared to N, and assuming that the number of ways n information sequences producing n error events can be arranged in a register of length N is $\binom{N}{n}$.

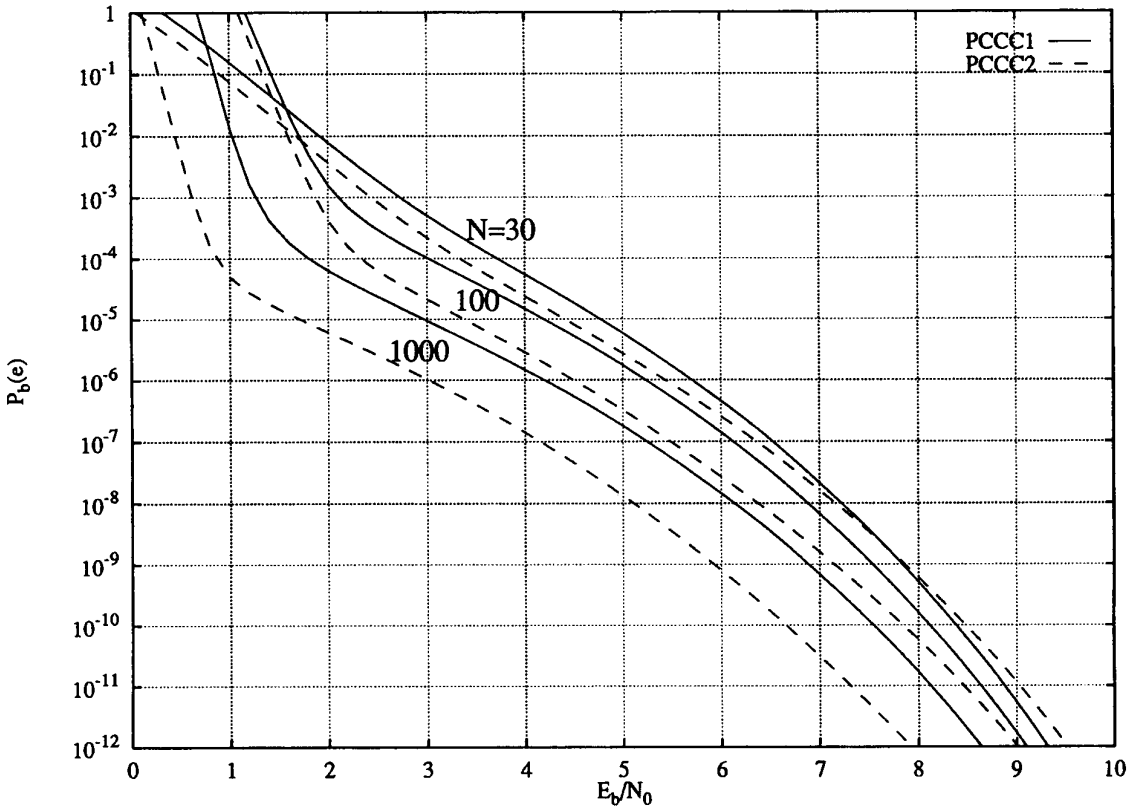


Fig. 3. Bit error probability bounds for PCCC1 and PCCC2 of Example 1 for interleaver lengths $N = 30$, 100, and 1000.

Consider now, the PCCC obtained by using as CC's the previously considered convolutional code and a uniform interleaver of length N. Inserting (4) into (2), we obtain the conditional weight enumeration function of the parallel concatenated block code equivalent to the PCCC as

$$A^{C_P}(w, Z) \sim \sum_{n_1=1}^{n_{\max}} \sum_{n_2=1}^{n_{\max}} \frac{\binom{N}{n_1}\binom{N}{n_2}}{\binom{N}{w}} \cdot A(w, Z, n_1)A(w, Z, n_2).$$

Using now for the binomial coefficient the asymptotic approximation

$$\binom{N}{n} \sim \frac{N^n}{n!}$$

we get

$$A^{C_P}(w, Z) \sim \sum_{n_1=1}^{n_{\max}} \sum_{n_2=1}^{n_{\max}} \frac{w!}{n_1! \cdot n_2!} N^{n_1+n_2-w} \cdot A(w, Z, n_1)A(w, Z, n_2) \quad (5)$$

which, for large N, can be approximated by the terms in the summations having the highest power of N, namely, those with $n_1 = n_2 = n_{\max}$

$$A^{C_P}(w, Z) \sim \frac{w!}{n_{\max}!^2} N^{2n_{\max}-w}[A(w, Z, n_{\max})]^2. \quad (6)$$

Inserting (6) into (1), we obtain the following asymptotic (in N) bound for the bit error probability:

$$P_b(e) \lesssim \sum_{w=w_{\min}}^{N} w \cdot \frac{w!}{n_{\max}!^2} N^{2n_{\max}-w-1} \cdot W^w[A(w, Z, n_{\max})]^2|_{W=Z=e^{-R_c E_b/N_0}} \quad (7)$$

where $w_{\min}$ denotes the minimum information weight in the error events of the CC.

A. Recursive and Nonrecursive Constituent Encoders

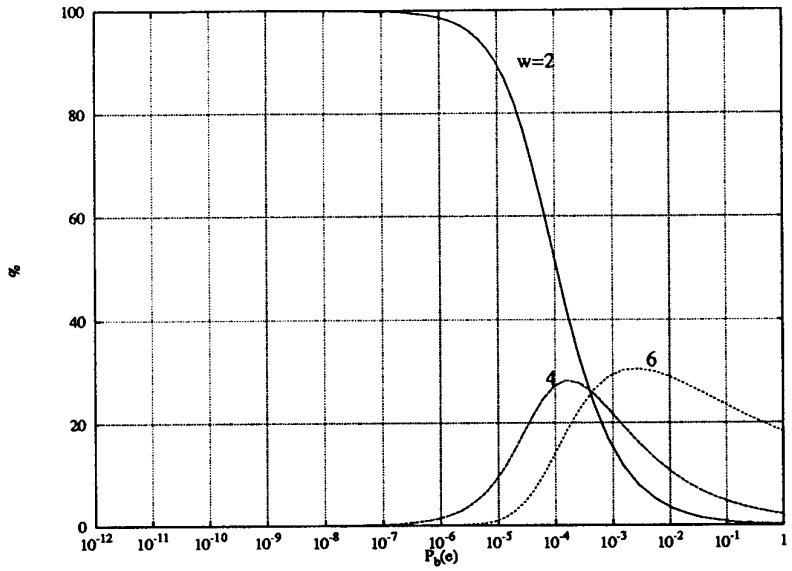
For nonrecursive convolutional constituent encoders, with reference to (7), we have $w_{\min} = 1$ and $n_{\max} = w$. In this case, in fact, every information sequence with weight one generates a finite-weight error sequence of length $2(\nu + 1)$, so that an information sequence with weight w can generate, at most, w error events corresponding to the concatenation of w error events of weight one. Taking now into account that

$$A(w, Z, w) = A(1, Z, 1)^w$$

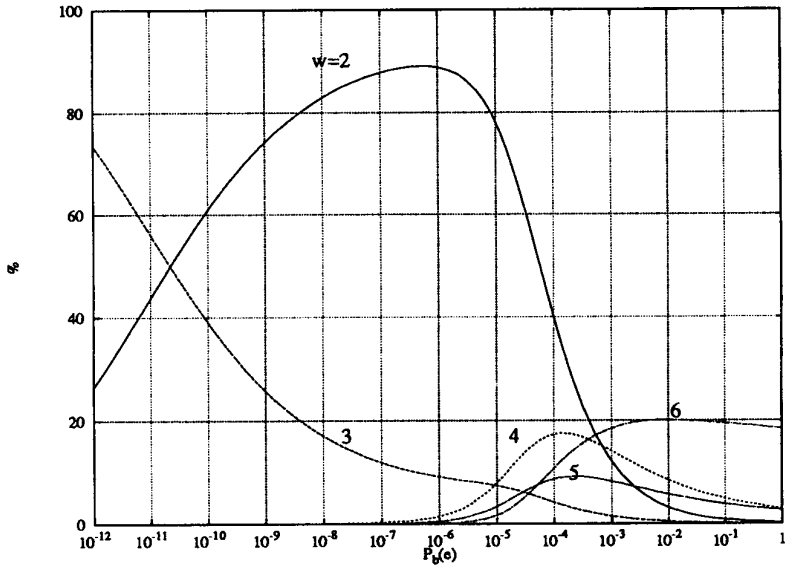
equation (7) particularizes to

$$P_b(e) \lesssim \sum_{w=1}^{N} \frac{N^{w-1}}{(w-1)!} \times W^w[A(1, Z, 1)]^{2w}|_{W=Z=e^{-R_c E_b/N_0}}. \quad (8)$$

Equation (8) shows that, in the most favorable case ($w = 1$), the bit error probability is independent of N, so that no



(a)



(b)

Fig. 4. Contributions in percent of error events with increasing information weight w to the overall bit error probability as functions of the bit error probability; (a) PCCC1; (b) PCCC2.

interleaving gain is possible. It is worthwhile to mention that the same conclusions apply to parallel concatenated codes using block codes as CC, since also for block codes $w_{\min} = 1$ and $n_{\max} = w$.

On the other hand, for recursive encoders, $w_{\min}$ is always greater than one. In particular, for recursive encoders, the following theorem holds.

Theorem 1: The parameter $w_{\min}$ is equal to two for recursive convolutional encoders. ∇

Proof: For simplicity, we shall prove the theorem for constituent codes of rate 1/2, with generator matrix $G = [1, n(D)/d(D)]$. The extension to the general case of rate k/n is straightforward.

The generator matrix of a rate 1/2 systematic recursive encoder with memory ν has the form

$$G = \left[1, \frac{n(D)}{d(D)}\right]$$

where $d(D)$ is a polynomial of degree ν. For this encoder, finite-weight error events are produced by the polynomial multiples of $d(D)$. Now, every polynomial $d(D)$ divides a polynomial of the form $1+D^i$, where i is the period of a linear feedback shift register with connection polynomial $d(D)$. On the other hand, since $d(D)$ has the form $1+\cdots+D^\nu$, it cannot divide a polynomial of the form D^i for any i. ■

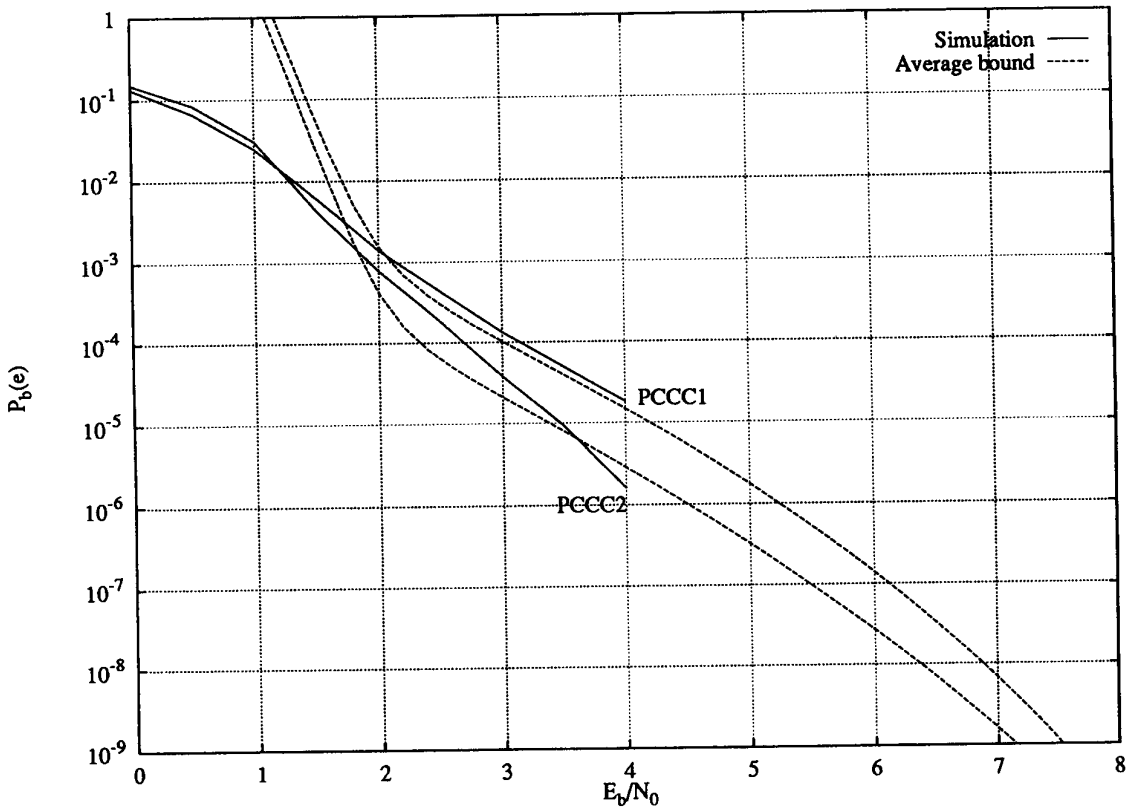


Fig. 5. Simulation results for the codes PCCC1 and PCCC2 (solid lines) compared to ML upper bounds. The simulated suboptimum iterative decoding algorithm used at most 20 iterations.

In the case of a recursive CC, with reference to (7), we have, thus, $w_{\min} = 2$ and $n_{\max} = \lfloor w/2 \rfloor$, where $\lfloor x \rfloor$ means "integer part of x." For convenience, we distinguish terms in the sum in (7) with odd and even w. For odd values of w, namely, $w = 2k + 1$, we have

$$W^{2k+1}(2k+1)(k+1)\binom{2k+1}{k}N^{-2}[A(2k+1, Z, k)]^2 \quad (9)$$

whereas, for even $w = 2k$

$$W^{2k}2k\binom{2k}{k}N^{-1}[A(2, Z, 1)]^{2k} \quad (10)$$

where we have made use of the equality

$$A(2k, Z, k) = A(2, Z, 1)^k \quad (11)$$

since the codewords described by $A(2k, Z, k)$ can only be the concatenation of k error events with information weight two.

Comparing (9) and (10) shows that terms with odd w are negligible, since they depend on N as N^{-2}.

To proceed further, we notice that the weight enumerating function of paths with weight two for the constituent code can be easily shown to be

$$A(2, Z, 1) = Z^{z_{\min}} + Z^{2z_{\min}-2} + Z^{3z_{\min}-4} + \cdots = \frac{Z^{z_{\min}}}{1 - Z^{z_{\min}-2}} \quad (12)$$

where $z_{\min}$ is the minimum weight of the parity check bits in error events generated by information sequences with $w = 2$.

Substituting (12) into (10) and then into (7) yields the asymptotic expression of the upper bound

$$P_b(e) \lesssim \sum_{k=1}^{\lfloor N/2 \rfloor} 2k\binom{2k}{k}N^{-1} \cdot \frac{(H^{2+2z_{\min}})^k}{(1 - H^{z_{\min}-2})^{2k}}\bigg|_{H=e^{-R_c E_b/N_0}} \quad (13)$$

where we have set $W = Z = H$.

Equation (13) leads to two important conclusions. First, it explains why the interleaver gain [see Section II, item 3] for PCCC's employing recursive constituent encoders goes as $1/N$.

Second, it makes explicit the most significant parameter through which the CC influences the PCCC performance, namely, $z_{\min}$, the lowest weight of the parity-check bits in error events of the CC generated by information sequences of length two. We define the lowest exponent of H in (13) as the *effective free distance* of the PCCC

$$d_{\text{free,eff}} = 2 + 2z_{\min} \quad (14)$$

since it plays a role similar to that of the free distance for convolutional codes. From the above considerations, we can conclude that the constituent encoders must be recursive, and that they should be chosen to maximize $z_{\min}$ and, hence, $d_{\text{free,eff}}$.

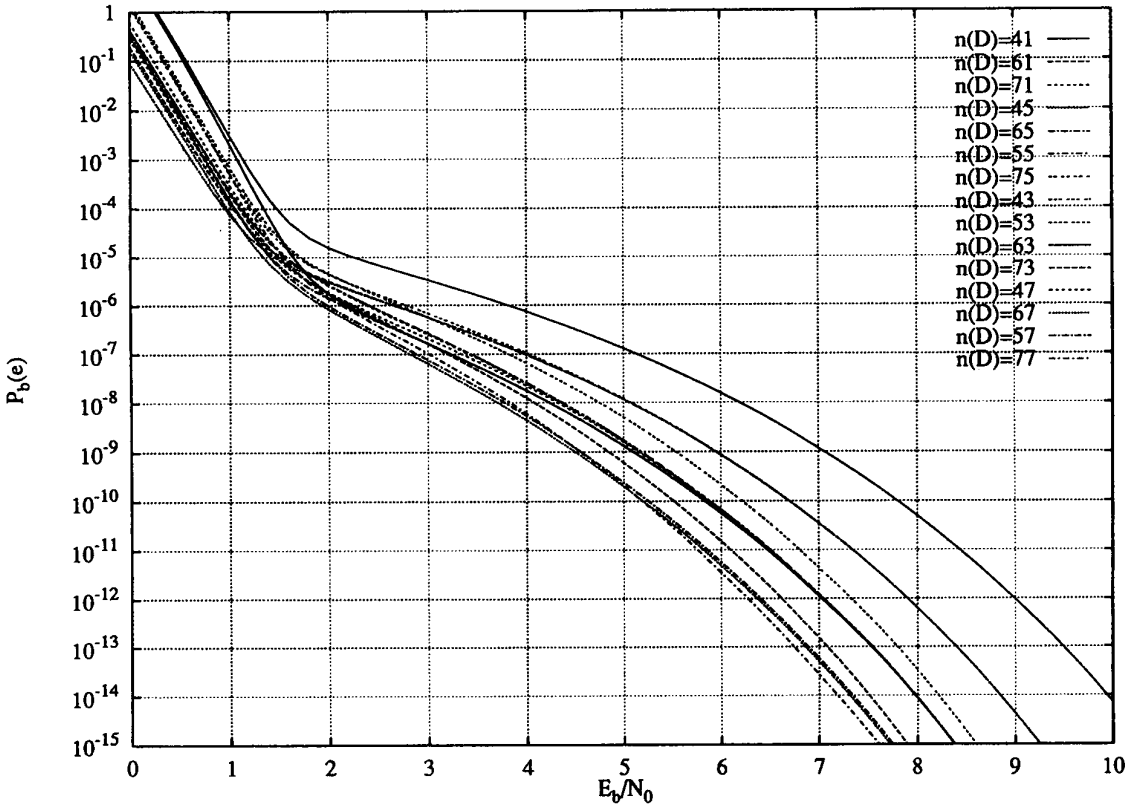


Fig. 6. Bit error probability bounds for rate 1/3 PCCC with interleaving length $N = 100$ employing as CC, the 32-state encoders yielding the value of $z_{\min}$ of Theorem 2.

IV. Optimization of the Recursive Constituent Encoders

Although the behavior of PCCC's for very low values of the SNR is not well understood yet, we believe that in this region, below the cut-off rate, the interleaver gain dominates the code performance and, thus, the effects induced by changing the CC will generally not be very great.

Nonetheless, for SNR's above 1.5–2 dB, the choice of the encoder matters. The following example can help to clarify the previous results.[4]

Example 1: Consider two PCCC's using uniform interleavers with the same length N and two different four-state systematic recursive convolutional CC's, identified as CC1 and CC2. The generator matrices are

$$G_1 = \left[1, \frac{1+D+D^2}{1+D^2}\right]$$

for CC1 and

$$G_2 = \left[1, \frac{1+D^2}{1+D+D^2}\right]$$

for CC2.

The two CC's are identical up to interchange of the two outputs, and have the same free distance, namely, five.

Let us embed CC1 and CC2 into two PCCC's (PCCC1 and PCCC2). For CC1 the information sequence $1+D^2$ generates a code sequence $[1+D^2, 1+D+D^2]$ with $z_{\min} = 3$, which in turn leads to the PCCC1 sequence $[1+D^2, 1+D+D^2, 1+D+D^2]$ with $d_{\text{free,eff}} = 2+2z_{\min} = 8$. On the other hand, for CC2 the information sequence $1 + D^3$ generates a code sequence $[1+D^2, 1+D+D^2+D^3]$ with $z_{\min} = 4$, and this leads to the PCCC2 sequence $[1+D^2, 1+D+D^2+D^3, 1+D+D^2+D^3]$ with $d_{\text{free,eff}} = 10$.

Therefore, the effective free distance for PCCC2 is 10, as opposed to eight for the PCCC1. Note also that $d_{\text{free}} = 8$ for PCCC1, whereas, for PCCC2 the free distance is obtained from the information sequence $1 + D + D^2$ which generates the code sequence $(1+D+D^2, 1+D^2, 1+D^2)$ with weight $d_{\text{free}} = 7$. However, since this information sequence has odd weight $w = 3$, a large enough N will make its contribution to the bit error probability insignificant with respect to that of $d_{\text{free,eff}}$.

These conjectures will be supported now by analytical results based on the evaluation of (1). They are reported in Fig. 3, where we plot the bit error probabilities for PCCC1 and PCCC2 for interleaver lengths $N = 30$, 100, and 1000. We see that the improvement yielded by PCCC2 over PCCC1 increases progressively with increasing N and that the performance curves cross only for large values of the SNR's.[5]

[4]This example was suggested to the authors by D. Forney, upon reading a draft version of [6].

[5]This crossing is due to the lower d_{free} of PCCC2.

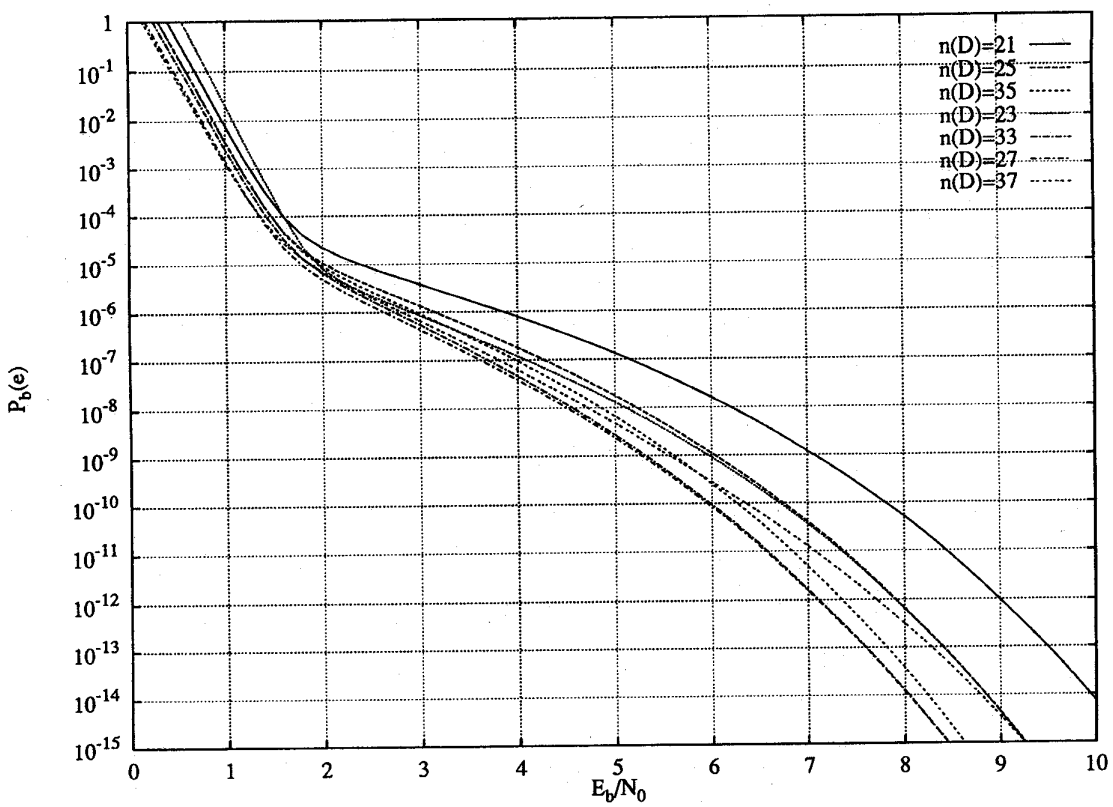


Fig. 7. Bit error probability bounds for rate 1/3 PCCC with interleaving length $N = 100$ employing as CC the 16-state encoders yielding the value of $z_{\min}$ of Theorem 2.

To obtain further insight into the significance of the parameter $z_{\min}$, we have plotted in Fig. 4, the relative contributions in percent to the overall bit error probability of the error events of PCCC1 and PCCC2 with interleaver length $N = 1000$ due to input sequences with increasing weights w as functions of the bit error probability $P_b(e)$. The curves in Fig. 4(b) give clear evidence that the codeword with information weight $w = 2$ is the dominant one for bit error probabilities ranging from 10^{-3} down to 10^{-10}.

Finally, to show that the improvement yielded by PCCC2 over PCCC1 under uniform interleaving is preserved with real, fixed interleavers, we have simulated the two PCCC with interleavers of length $N = 100$ chosen at random. The decoder uses an iterative algorithm described in [10] with a number of iterations equal to 20. The results are reported in Fig. 5, where we have also redrawn the upper bounds previously obtained.$\diamond$

The design objective for the constituent recursive convolutional encoders is to obtain as large $z_{\min}$ as possible. The following theorem shows what can be achieved with a CC of rate $1/n$, given the memory and using a primitive feedback polynomial.

Theorem 2: A rate $1/n$ recursive convolutional encoder with memory ν and generator matrix

$$G = \left[1, \frac{n_1(D)}{d(D)}, \frac{n_2(D)}{d(D)}, \cdots, \frac{n_{n-1}(D)}{d(D)}\right]$$

can achieve the following value for $z_{\min}$:

$$z_{\min} = (n-1)(2^{\nu-1} + 2). \tag{16}$$

Indeed, any check generators $n_i(D)/d(D)$, where $d(D)$ is any primitive polynomial of degree ν and $n_i(D)$ is any monic polynomial of degree ν except $d(D)$ [with the $(n-1)$ numerator polynomials $\{n_i(D), 1 \leq i \leq n-1\}$ having greatest common divisor 1] will achieve this value of $z_{\min}$.$\triangledown$

Proof: For simplicity, we will prove the theorem for a CC of rate 1/2, with generator matrix $G = [1, n(D)/d(D)]$. The extension to the general case of rate $1/n$ is straightforward.

The polynomial $n(D)$ must have degree $\deg[n(D)] \leq \nu$. We first prove that all polynomials $n(D)$ with degree $d < \nu$ yield a value of $z_{\min}$ strictly less than the right-hand side of (16). To this end, let $M = 2^{\nu}$ and note that $d(D)$ is the generator polynomial of an $(M-1, M-\nu-1)$ Hamming code [11]. Moreover, since $d(D)$ is a primitive polynomial, the minimum-degree polynomial of the form $1 + D^i$ which is a multiple of $d(D)$ is $(D^{M-1} + 1)$. The quotient $q(D)$ obtained from the division of $(D^{M-1} + 1)$ by $d(D)$ is the generator polynomial of an $(M-1, \nu)$ cyclic maximal length shift-register code [11], so that the products $q(D)n(D)$ are codewords of this code, which is known to have all words (except the zero codeword) with the same weight $z = M/2$. This completes the first part of the proof of (16).

To increase the value of $z_{\min}$, the only possibility consists in increasing the degree of $n(D)$ to ν. We prove now that all

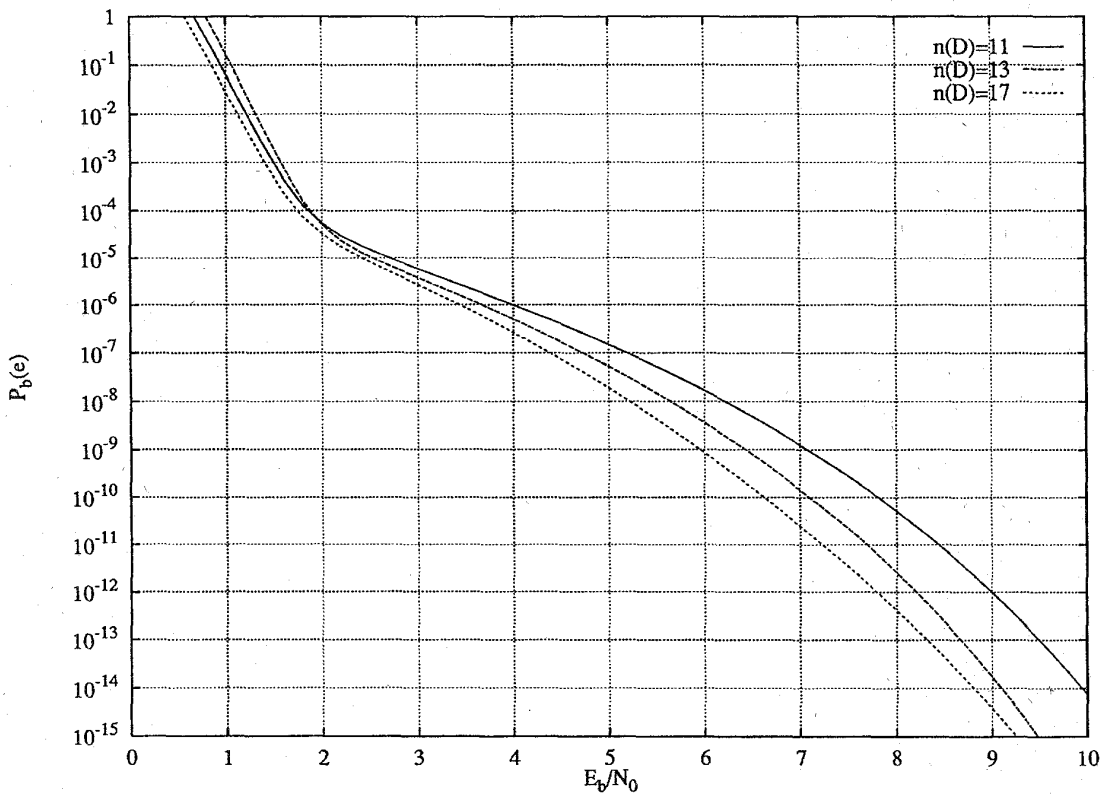


Fig. 8. Bit error probability bounds for rate 1/3 PCCC with interleaving length $N = 100$ employing as CC the eight-state encoders yielding the value of $z_{\min}$ of Theorem 2.

TABLE I
BEST RATE 1/2 CC FOR RATE 1/3 PCCC WITH INTERLEAVER LENGTH $N = 100$. THE POLYNOMIALS $d(D)$ AND $n(D)$ DEFINING THE FEEDBACK AND PARITY-CHECK CONNECTIONS OF THE ENCODERS ARE GIVEN IN OCTAL NOTATION WITH THE LEAST SIGNIFICANT BIT ON THE LEFT

ν	$d(D)$	$n(D)$	$d_{\text{free,eff}}$	d_{free}^{CP}	w_{free}^{CP}
1	3	2	4	4	2
2	7	5	10	7	3
3	15	17	14	8	4
4	31	33	22	9	5
	31	27	22	9	5
5	51	77	38	10	6
	51	67	38	12	4

polynomials $n(D)$ of the form $n(D) = D^{\nu} + \cdots + 1$ achieve (16). Split $n(D)$ as

$$n(D) = D \cdot D^{\nu-1} + n_2(D) \stackrel{\text{def}}{=} D \cdot n_1(D) + n_2(D)$$

and consider that the products $c_1(D) = q(D)n_1(D)$ and $c_2(D) = q(D)n_2(D)$ are codewords of the maximal-length shift-register code with a "one" as the least significant bit. On the other hand, the product $Dc_1(D)$ represents a (noncyclic) shift of one position to the left of $c_1(D)$. Since the maximal-length code is cyclic, and $c_1(D)$ has degree $M-1$, the binary word represented by $Dc_1(D)$ coincides with a codeword of the maximal length code except for the most significant bit (a "one" corresponding to the power D^{M-1}) and the least significant bit [a "zero" instead of the "one" which would follow from a cyclic shift of one position to the left of the codeword $c_1(D)$]. Thus, summing modulo-2 the binary words represented by $Dc_1(D)$ and $c_2(D)$ yields, for powers from D up to D^{M-2}, part of a codeword of the maximal length code with weight $M/2$.[6] As to the remaining powers, D^{M-1} contributes a "one" to the weight, and D^0 gives another "one" since $c_2(D)$ has least significant bit equal to "one," whereas, $Dc_1(D)$ has it equal to "zero." This completes the proof of (16).

The only hypotheses made on the polynomials $n(D)$ is that they be of the kind $D^{\nu} + \cdots + 1$. There are obviously M such polynomials; all work, with the exception of $n(D) = d(D)$. This completes the proof of the second part of the theorem■.

One could ask why we choose $d(D)$ as a primitive polynomial. One reason is that by doing so the quotient $(D^i + 1)/d(D)$ has the largest degree, making it easy to choose an $n(D)$ that maximizes $z_{\min}$.[7] Moreover, if D^i+1 divides $d(D)$, thus, $D^{ki} + 1$ also divides it, for k integer. Thus, the number of distinct binomials that are multiples of $d(D)$ contained in an interleaver of length N (which can originate error events of the PCCC with $w = w_{\min}$) is of the order of N/i, and there

[6]This codeword would have the least significant bit equal to zero, being the sum modulo-2 of $c_2(D)$ and the cyclic shift of one position to the left of $D^{\nu-1}q(D)$.

[7]We discussed this matter with S. W. Golomb, and, a few days after, he was so kind so as to send us a proof [9] that (16) is indeed the largest $z_{\min}$ achievable by any polynomial $d(D)$.

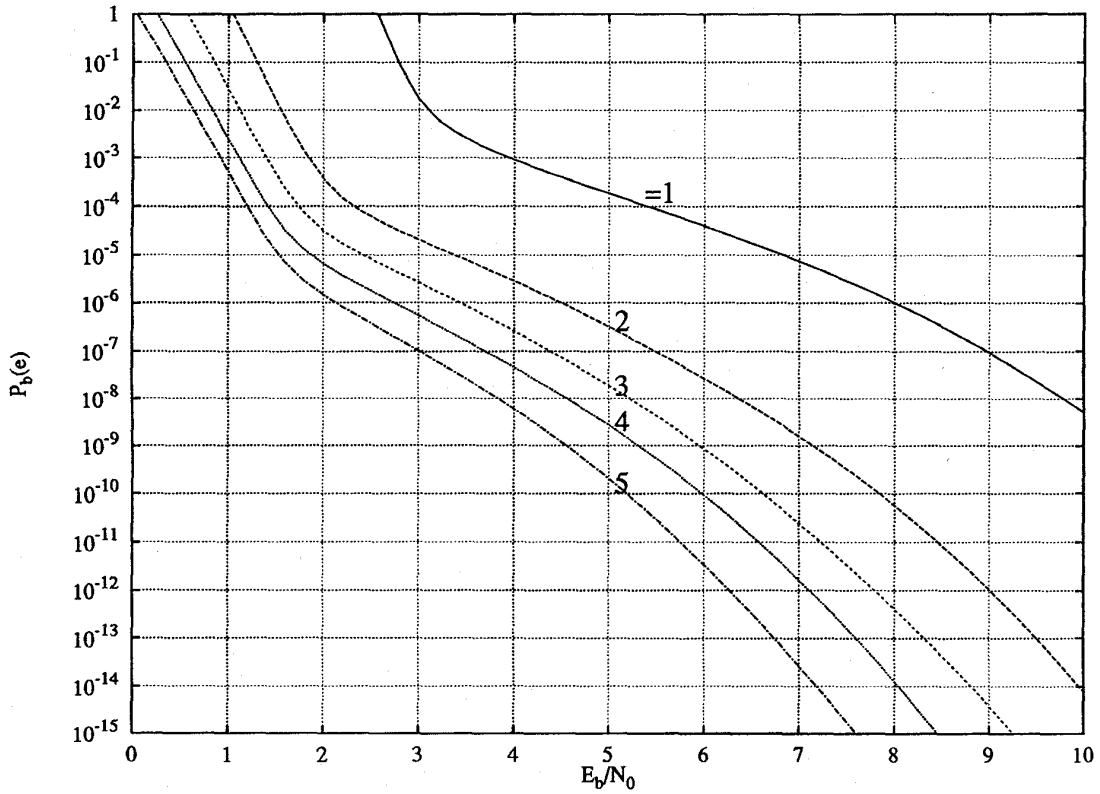


Fig. 9. Bit error probability bounds for rate 1/3 PCCC with interleaving length $N = 100$ employing as CC the best two, four, eight, 16, and 32-state encoders reported in Table I.

is an obvious reason to keep this number as low as possible by maximizing i.

From here on, we will limit ourselves to the case of CC with rate 1/2. Generalization to rate $1/n$ constituent codes is straightforward. Theorem 2 and its proof suggest that a good procedure for the choice of the generator matrix

$$G = \left[1, \frac{n(D)}{d(D)}\right]$$

of a recursive CC with memory ν is the following:

1) Choose as $d(D)$ a primitive polynomial of degree ν.
2) For all $n(D)$ yielding a weight $z_{\min} = 2^{\nu-1} + 2$, evaluate the weight distribution and choose the best. The optimization in its simplest form can be done by choosing the $n(D)$ that maximizes the free distance of the PCCC, or, more accurately, by evaluating the analytical bounds to the bit error probability for all candidate codes and choosing the one yielding the lowest bit error probability for the desired interleaver length and range of SNR's.

An example of the design of a CC with 32 states will clarify the procedure.

Example 2: Consider a PCCC with rate 1/3 employing as its CC a 32-state rate 1/2 systematic recursive encoder. There are 15 distinct polynomials $n(D)$ yielding the highest value of $z_{\min} = 18$. We have evaluated the upper bounds to the bit error probabilities of the PCCC using these codes and a uniform interleaver of length $N = 100$. The results are reported in Fig. 6. The curves show that the differences in performance due to different CC's are significant. In fact, the dispersion of the curves is close to 3 dB at a bit error probability of 10^{-9}.[8] $\diamondsuit$

We have applied this procedure to optimize rate 1/3 PCCC using rate 1/2 constituent recursive convolutional encoders with memory ν from 1–5 and a uniform interleaver with length $N = 100$.[9] The results are reported in Table I, where we show the generator matrix through the polynomials $n(D)$ and $d(D)$, the effective free distance $d_{\text{free,eff}}$, the free distance d_{free} of the PCCC and the weight w_{free} of the information sequence which generates the error event yielding the free distance.

In Figs. 6–8, we report the bit error probabilities for all candidate codes with eight, 16, and 32 states (for two and four states, there is only one code, whose performance will be shown in Fig. 9).

Finally, in Fig. 9, the performance of the best codes for each number of states is reported. These curves give evidence that increasing the number of states can improve the performance significantly. We conclude that interleaver length and the memory of the CC have to be traded off according to

[8]The significant difference in performance between codes with the same $d_{\text{free,eff}}$, especially evident for low values of the bit error probability, suggests that other error events of the PCCC with higher w and weight lower than $d_{\text{free,eff}}$ begin to influence the performance. The significance of this phenomenon and the bit error probability where it starts to have impact depend highly on the interleaver length.

[9]The hierarchy does not change for larger values of N.

the system requirements. In fact, increasing the interleaver length increases the decoding delay at almost no expense in complexity, whereas, increasing the memory increases the complexity with only a slight increase in decoding delay. These conclusions differ somewhat from those of [1] where the role of the CC was considered unimportant. This was probably due to the SNR range considered and to the very long interleaver used.

V. Conclusions

Our main results may be summarized as follows:

- The error coefficient interleaving gain for a PCCC with large interleaving length goes as $N^{1-w_{\min}}$, where N is the interleaver length and $w_{\min}$ is the minimum number of information bits in a finite-weight error event.
- All recursive convolutional encoders have $w_{\min} = 2$, so that the interleaving gain goes as $1/N$. On the other hand, all nonrecursive convolutional encoders and block codes have $w_{\min} = 1$, so such codes are not useful in parallel concatenated codes.
- The next most important constituent code parameter is $z_{\min}$, the minimum parity-check weight in code sequences with $w = 2$. For a large range of SNR's, the behavior of the PCCC is determined by the effective free distance $d_{\text{free,eff}} = 2 + 2z_{\min}$.
- It is possible to achieve $z_{\min} = (n-1)(2^{\nu-1}+2)$ with a rate $1/n$ recursive convolutional encoder with memory ν.
- Bit error probability bounds show that there are significant performance differences between PCCC codes with the same N, ν, $z_{\min}$. A table of the best rate 1/2 convolutional codes to be used as CC in a PCCC has been presented, for $1 \leq \nu \leq 5$.

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. ICC'93*, Geneva, Switzerland, May 1993, pp. 1064–1070.

[2] C. Berrou and A. Glavieux, "Turbo codes: General principles and applications," in *Proc. 6th Tirrenia Int. Workshop Digital Commun.*, Tirrenia, Italy, Sept. 1993.

[3] S. Le Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," in *Proc. ICC '94*, New Orleans, LA, May 1994.

[4] D. Divsalar and F. Pollara, "Turbo codes for PCS applications," in *Proc. ICC '95*, Seattle, WA, June 1995.

[5] S. Benedetto and G. Montorsi, "Performance of turbo codes," *Electronics Letters*, vol. 31, no. 3, pp. 163–165, Feb. 1995.

[6] ——, "Unveiling turbo-codes: Some results on parallel concatenated coding schemes," accepted for publication in *IEEE Trans. Information Theory*, Sept. 1995.

[7] ——, "Analysis and results on parallel concatenated coding schemes with multiple interleavers," in *Proc. 3rd Int. Symp. Commun. Theory Applications*, Lake District, UK, July 1995.

[8] ——, "Average performance of parallel concatenated block codes," *Electron. Lett.*, vol. 31, no. 3, pp. 156–158, Feb. 1995.

[9] S. W. Golomb, Private communication.

[10] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Soft-output decoding algorithms in iterative decoding of parallel concatenated convolutional codes," submitted to *ICC'96*.

[11] S. Benedetto, E. Biglieri, and V. Castellani, *Digital Transmission Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

Sergio Benedetto (SM'90), for a photograph and biography, see this issue p. 590.

Guido Montorsi (M'95), for a photograph and biography, see this issue p. 590.