Some Extended Results on the Design of Punctured Serially Concatenated Convolutional Codes

Massimiliano Laddomada and Bartolo Scanavino

Abstract— **The aim of this paper is twofold. On one hand, it presents the results of the search for good punctured systematic recursive convolutional encoders suitable for application in serially concatenated convolutional codes (SCCCs) operating in two different target regions: at low-to-moderate signal-to-noise ratios (SNRs), i.e., in the so-called waterfall region, and at high SNRs. On the other hand, it provides some useful design guidelines for choosing the constituent encoders in an SCCC.**

The results of the search for good SCCCs operating in the waterfall region rely upon an effective algorithm, based on density evolution technique, first proposed in a companion paper. Good punctured SCCCs were obtained through considerations deduced by the behaviour of the bit error probability of an SCCC for high values of both SNR and interleaver length, i.e., through asymptotic considerations.

The mother codes in the serial concatenation are rate $\frac{1}{2}$ recursive convolutional encoders (RCC) found by an exhaustive search for encoders tailored to SCCC schemes, using two different selection criteria. Extensive tables of optimized puncturing patterns for various mother codes and SCCCs are presented along with sample simulation results.

Index Terms— **Convolutional codes, high rate, punctured, serially concatenated convolutional codes, turbo codes.**

I. INTRODUCTION

THE idea of serial concatenation of interleaved codes dates back to 1966 upon a seminal work of Forney [1]. Such codes have been in use for a while in many different applications from deep space to compact discs for the music industry. The recent explosion of research activities has largely surrounded Parallel Concatenated Convolutional Codes (PCCCs) and SCCCs [2]-[4] and iterative soft decoding. A block diagram of a typical SCCC with some useful notation that will be used throughout the paper is depicted in Fig. 1.

In wireless band-limited channels spectral efficiency calls for high-rate channel codes with rate $\frac{k}{k+s}$ to limit the bandwidth expansion inherent in channel coding. Cain *et al.* in [5] first suggested puncturing a mother code of lower rate for obtaining higher rate codes with simplified Viterbi decoding. Since then, many papers have addressed the problem of obtaining optimal punctured convolutional codes, but most of the literature has been devoted to finding optimal puncturing patterns (PPs) for non-recursive convolutional encoders. The recent research activities related to Turbo codes suggest that

there is a great need for suitable high-rate *recursive* convolutional encoders to be used in such applications. Among the various papers in this field, several are of relevance in connection with the current work. The survey of the literature we present in the following is by no means exhaustive and is meant to simply provide a sampling of the literature in this fertile area.

Classical papers [6]-[17] deal with the problem of obtaining optimal PPs for non-recursive convolutional encoders. Another interesting approach to puncturing was introduced by Hagenauer in 1988. In detail, in [18] Hagenauer introduced the so called Rate-Compatible Punctured Convolutional Codes (RCPCCs). In RCPCC, a low rate $1/n$ mother convolutional code is punctured periodically with period P. RCPCCs are essentially a class of punctured convolutional codes (CCs) with rate $\frac{k}{k+s}$, whereby the obtained puncturing table is such that all code bits of higher rate codes are used by the lower rate codes. The concept of RCPCC has been extended in [19]-[21] to both PCCCs and SCCCs. Papers [21]-[25] deal with the design of high-rate SCCCs, while articles [26]-[28] address the design of good constituent codes for turbo codes applications. In particular, in [23] the authors proposed a design algorithm for SCCCs aiming at minimizing the effects of both weigth-2 and weigth-3 input error events on the SCCC distance spectra. Also related to the current work are papers [29]-[31], in which the authors proposed techniques to analyze the convergence behavior of parallel and serially concatenated convolutional codes.

This paper presents the results of an extensive search for SCCC schemes constituted by punctured encoders optimized for two different SCCC target SNR operating regions, i.e., at low-to-moderate SNRs and at high SNRs.

Before proceeding, let us review the mathematical notation on punctured CCs used throughout the paper. Puncturing is obtained by regularly deleting some output bits of a mother code with rate $1/n$. As a result of puncturing, the trellis of the punctured code becomes periodically time-varying. In the case of concatenated schemes, there is no loss of generality if we consider a rate $1/2$ systematic mother encoder, specified by a 1×2 generator matrix $G(D) = \left[1, \frac{g_1(D)}{g_0(D)} \right]$ composed of two polynomials $g_1(D)$ and $g_0(D)$ specifying the connections of the finite state encoder. In this formula, $g_i(D) = g_o^i + g_1^i D + \ldots + g_\nu^i D^\nu$, where $i = 0, 1$, $g_l^i \in \{0, 1\}$ and $l = 0, \ldots, \nu$. The variable ν specifies the memory of the convolutional encoder (the encoder constraint length is $1 + \nu$). The form of the generator matrix $G(D)$ expresses the fact that

Authors are with the Dep. of Electronics, Politecnico di Torino, Italy. (e-mail: {laddomada, scanavino}@polito.it).
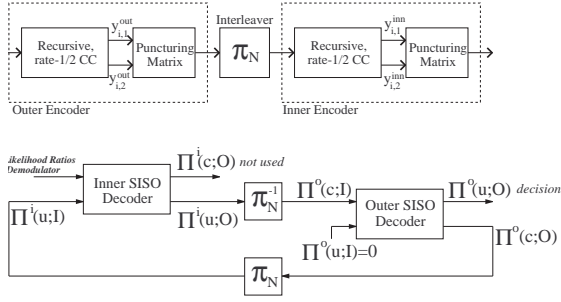
Fig. 1.   Block diagrams of the encoder and iterative decoder for SCCCs.

the considered encoders are recursive (i.e., we have ratio of polynomials).

The rest of the paper is organized as follows. Section II is devoted to the formulation of the problem and presents the code search technique we have employed for the design of asymptotically good punctured SCCCs composed by punctured constituent encoders of rate $\frac{k}{k+s}$, for $k = 2, \ldots, 8$ and $s = 1, \ldots, k-1$. In section III, we deal with the technique for the design of good SCCCs, operating in the low-to-moderate SNR region, designed through an effective algorithm first proposed in a companion paper [32]. In section IV, we present the results of our code search and propose their potential applications to SCCCs. Conclusions are drawn in section V.

II. PROBLEM FORMULATION AND CODE SEARCH TECHNIQUE FOR ASYMPTOTICALLY GOOD SCCCS

This section deals with the design techniques adopted in our search for punctured encoders to be used as outer and inner encoders for asymptotically good SCCCs. For clarity of exposition, we consider the design of good constituent codes for SCCCs separately from the design of the related puncturing patterns.

A. Design of Constituent Encoders for Asymptotically Good SCCCs

In this section, the emphasis is on the design of the constituent encoders in SCCCs.

Based on the asymptotic analysis accomplished in [4], we can state that the inner encoder in an SCCC should be a recursive convolutional encoder (no matter it is systematic or not) in order for the interleaver to yield a coding gain, while the outer encoder is only required to have maximum free distance, regardless of whether it is recursive or not.

The design approach pursued in this work is strictly related to the average asymptotic bit error probability of SCCCs, that, for interleaver sizes N very large and for high values of $\frac{E_b}{N_o}$, are given by [4]:

$$P_{b_{SCCC}} \tilde{<} C_e N^{-\frac{d_f^o}{2}} Q\left(\sqrt{d_f^o d_2 R_s \frac{E_b}{N_o}}\right) \quad (1)$$

for even values of d_f^o, and

$$P_{b_{SCCC}} \tilde{<} C_o N^{-\frac{d_f^o+1}{2}} Q\left(\sqrt{\left[\left(d_f^o - 3\right) d_2 + 2d_3\right] R_s \frac{E_b}{N_o}}\right) \quad (2)$$

for odd values of d_f^o. In both equations, C_e and C_o are two terms which do not depend on the interleaver length N, d_f^o is the free distance of the outer encoder, d_2 is the effective free distance of the inner encoder (i.e., the minimum weight of the inner codewords generated by weight-2 input sequences), R_s is the rate of the SCCC, d_3 is the minimum weight of the inner code codewords generated by weight-3 input sequences, and $Q(.)$ is the Gaussian integral function defined by $Q(t_o) = \frac{1}{\sqrt{2\pi}} \cdot \int_{t_o}^{\infty} e^{-\frac{t^2}{2}} dt$.

Equations (1) and (2) suggest that the outer encoder should be a convolutional encoder with maximum free distance: the interleaver gain goes asymptotically as $N^{-\frac{d_f^o+1}{2}}$ for odd values of d_f^o, and as $N^{-\frac{d_f^o}{2}}$ for even values of d_f^o. In particular, compatibly with the desired rate R_s of the SCCC, it is better to choose outer encoders with odd values of d_f^o. It is not necessary for the outer encoder to be neither recursive, nor systematic. Since a large body of literature has addressed the design of good non-recursive convolutional encoders, in this paper we only focus on good punctured codes obtained from recursive, systematic convolutional encoders.

Based on observations deduced above, the code search strategy adopted for obtaining good outer encoders consists in minimizing the required SNR for an appropriately chosen target bit error rate (BER). In fact, as suggested by Lee in [14]-[16], the criteria to only maximize the minimum distance of a code may be valuable if the target BER is extremely small. When on the other hand the target BER is in the moderate range, it is insufficient to only consider the first distance terms of the code's distance spectrum. Furthermore, minimization of the BER based on the first few terms of the distance spectrum would require the knowledge of the CC operating SNR, which is not always given or known specially if the CC must be embedded in a PCCC or SCCC scheme. Given these considerations, it is preferable to minimize the required SNR for a given target BER. This is indeed the strategy adopted for the code search presented in this paper. Notice that, from a practical point of view, the minimization of the required SNR for a small value of the target BER often yields codes which give distance spectra with maximum free distance. This is essentially due to the fact that the codeword distances appear in the $Q(\cdot)$ function defining the BER probability as shown in (3). The main difference between our approach and that of maximizing only the minimum distance, is that in our approach the codeword multiplicities and the distance profile has a major impact on the choice of the codes.

As a cost function for BER minimization, we employed the upper-bound on the performance of a convolutional code of rate $R_c = \frac{k}{n}$ under soft-decision maximum likelihood Viterbi decoding, defined as follows:

$$P_b \leq \frac{1}{k} \sum_{d=d_{free}}^{\infty} w_d Q\left(\sqrt{2\frac{E_b}{N_o} R_c d}\right) \quad (3)$$

where d_{free} is the minimum distance of the CC, also called free distance, w_d is the cumulative input Hamming weight associated with all the incorrect paths (also so-called error events) in the trellis departing from the correct path at a given

node and merging with the correct path later, whose Hamming weight of the coded sequence is d, and $Q(.)$ is the Gaussian integral function defined above.

Recursive mother encoders selected as constituent encoders in the serial concatenation have been obtained through a search for good constituent recursive, systematic, rate $1/2$ convolutional encoders by minimizing the SNR for a given target BER [14]-[16]. In order to resolve potential ties, in a second phase, between all the encoders yielding similar performances we have chosen the encoder with the best d_2, and, subsequently, the best d_3.

The mother encoders obtained by minimization of the SNR can be used as outer codes in SCCC schemes, since the optimization approach has the goal to maximize the minimum distance of the punctured code and minimize the overall weight associated with the input patterns yielding the minimum distance, followed by maximization of the successive low distance terms and minimization of their corresponding input weights for the first six minimum distance terms used in the formulation of the cost function in (3). Extensive analysis suggested us the use of the target BER=10^{-6}: such a small target value makes low-weight codewords to dominate the cost function in (3). In the following, we will identify this criterion with the acronym C_1.

In connection with the design of inner encoders, we used as objective function the maximization of the effective free distance d_2. In a second phase, among the encoders yielding the same d_2 (if several), we have chosen the one requiring the minimum SNR for achieving a predefined target BER, and then the one with maximum d_3. In the following, we will identify this criterion with the acronym C_2.

The results of this search are shown in Table I: the table contains the best rate $1/2$, systematic, recursive constituent encoders with memory ν equal to 2, 3, 4 and 5. Each row presents the optimization results for a given memory ν. In any given row, we show two encoders found using the two different objective functions presented above. In particular, in the first line of a given row we present the best encoder found maximizing d_2, while in the second line we present the best encoder found by minimizing the required SNR for achieving a predefined target BER. The second column shows the generator matrices of the optimal encoders, the third column lists the first few terms of the code distance spectra (each triplet represents the Hamming weight of the codewords d_i, the multiplicity m_i of all the input patterns with overall weight w_i leading to codewords with weight d_i, and the overall input weight w_i) and the last column shows the effective free distance d_2 and the distance generated by weight-3 sequences denoted d_3 (the entry $d_3 = \infty$ is used to signify the fact that there are no weight-3 input patterns leading to error events in the trellis of the examined encoder).

The results obtained for the 32-state, systematic, recursive encoders are slightly different. During our search, we found an encoder satisfying both objective functions mentioned above (listed in the lower line of Table I for $\nu = 5$ encoders). For this reason, in the upper line of the same row we show the best encoder having maximum d_3 while simultaneously satisfying the minimum SNR requirement and achieving maximum d_2.

This encoder can be useful as inner code in an SCCC when the outer code is punctured so that its minimum distance is equal to 3 or lower. In this case, because of the absence of inner code codewords generated by weight-3 input patterns, the overall SCCC could yield better performance.

For the sake of clarity, we show an example of how the generator matrices in Table I have to be interpreted. For example, the 16-state systematic encoder whose generator matrix expressed in octal form in Table I is $G(D) = \left[1, \frac{23}{35}\right]$ has the equivalent representation $G(D) = \left[1, \frac{1+D^3+D^4}{1+D+D^2+D^4}\right]$.

In relation to the values achievable by the effective free distance, d_2, in [27] it is shown that the maximum d_2 achievable with a recursive, systematic, rate $1/2$ encoder with generator matrix $G(D) = \left[1, \frac{g_1(D)}{g_o(D)}\right]$ is defined by:

$$d_2 \leq 4 + 2^{\nu-1}. \tag{4}$$

In particular, equality is achieved when the denominator polynomial $g_o(D)$ is primitive and under two additional conditions which require that $g_o(D) \neq g_1(D)$ and that $deg\,[g_i(D)] \leq \nu$ for $i = 0, 1$.

B. Design of Punctured Outer Encoders for Asymptotically Good SCCCs

In this section, the emphasis is on the design of punctured convolutional encoders suitable as outer encoders in asymptotically good SCCCs.

The design criteria relies upon the considerations deduced above from Equ.s (1) and (2). Indeed, the code search strategy adopted for obtaining good PPs for outer encoders consists in minimizing the required SNR for the target BER=10^{-6}. We searched for the best PPs with this criterion, by using the BER upper bound as expressed in (3), but, for complexity reasons, based on the first four dominant terms of the distance spectrum. Furthermore, we have verified experimentally that the minimization of the required SNR for this target BER practically yields PPs which give distance spectra with maximum free distance. The main difference between our approach and that of maximizing only the minimum distance, is that in our approach the codeword multiplicities and the distance profile has a major impact on the choice of the PPs. In order to resolve potential ties, in a second phase, between all the PPs yielding similar performances we have chosen the PP with the best d_2 and subsequently the best d_3. In the following, we will identify this criterion with the acronym C_1 when applied to PPs.

The mother encoders to which puncturing is applied are the best codes found under criterion C_1 discussed in the previous section. We have used these codes as mother codes by following the generally accepted rule that good mother codes lead to good punctured codes. Indeed, practical documented results show that PPs with maximum possible d_{free} are derived from mother encoders having maximum d_{free} [24], [25].

C. Design of Punctured Inner Encoders for Asymptotically Good SCCCs

In this section, the emphasis is on the design of punctured convolutional encoders suitable as inner encoders in asymp-

totically good SCCCs.

Equations (1) and (2) suggest that good punctured inner encoders in an SCCC are the ones which maximize the effective free distance d_2. In other words, the effective free distance d_2 of the inner encoder acts as the free distance of the outer encoder, d_f^o, in the case in which d_f^o is even. In the case in which d_f^o is odd, we should also maximize the d_3 in (2) (d_3 is the minimum weight of the inner code codewords generated by weight-3 input sequences). Note that these considerations are valid for the design of both mother encoders and related PPs.

Furthermore, the mother encoders designed by maximizing the effective free distance can also be used as constituent encoders in parallel concatenated convolutional codes (PCCCs), since it is known that, asymptotically, the upper-bound on the BER of a PCCC can be expressed as:

$$P_{b_{PCCC}} \leq \frac{1}{N} Q \left(\sqrt{2 \frac{E_b}{N_o} R_c d_2} \right) \qquad (5)$$

where R_c is the code rate of the PCCC, and N is the interleaver length. Since this paper is devoted to the design of good punctured encoders for SCCC schemes, we will not mention PCCCs any further in what follows. The interested readers can refer to [27], [33] for details on this topic.

The search for good PPs for inner encoders has been accomplished by choosing the PP with the best d_2, and, subsequently, the best d_3. In order to resolve potential ties, in a second phase, between all the PPs yielding similar performances in terms of d_2 and d_3, we have chosen the PP that minimizes the SNR to achieve the target BER= 10^{-6} as expressed in (3), but based on the first four dominant terms of the distance spectrum. Obviously, this approach yields asymptotically good PPs for inner codes in SCCCs operating at high SNRs. In the following, we will identify this criterion with the acronym C_2.

The mother encoders to which puncturing is applied are the best codes found under criterion C_2 discussed above. Practical documented results show that PPs with maximum possible d_2 are derived from mother encoders having maximum d_2 [24], [25].

III. DESIGN OF GOOD SCCCs THROUGH DENSITY EVOLUTION CODE MATCHING

The design of good SCCCs operating in the low-to-moderate SNR range (i.e., in the waterfall region before the error floor shows its effects) cannot be based on the asymptotic considerations deduced by (1) and (2). During the search for good SCCCs, we recognized that an effective design algorithm for punctured inner encoders in SCCCs operating in the low-to-moderate SNR range could be based on the density evolution technique [29], [30], [34], whose main merit is to highlight the convergence behavior of the iterative decoder. In a companion paper [32], we proposed an effective algorithm for matching both outer and inner encoders in an SCCC scheme in such a way as to obtain good punctured SCCCs operating in the so-called waterfall region. In order to understand the rationales beyond the proposed algorithm, we will briefly recall some results presented in [32]. In particular,
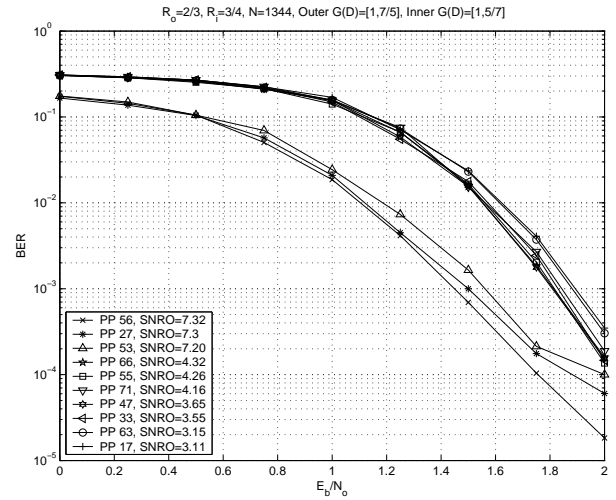

Fig. 2. Simulation results for various punctured SCCCs constituted by different puncturing patterns for the inner encoder $G(D) = [1, 5/7]$. All SCCCs have the same spread-36 interleaver of size 1,344. The legend shows, for each SCCC, the PP of the inner encoder and the $SNRO$ provided by the proposed algorithm after 10 iterations. The title of the figure shows the rate R_o of the outer code, the rate R_i of the inner encoder, the interleaver size, and the generator matrices of both outer and inner encoders.

first we will describe a reasonable model for characterizing the extrinsic information exchanged during iterative decoding of SCCCs, and, then, we will discuss a nonlinear model for describing the behaviour of the iterative decoder.

Consider the iterative decoder scheme of an SCCC shown in Fig. 1, whereby the extrinsic information, denoted by $\Pi(\cdot)$, is passed from one Soft-In Soft-Out (SISO) decoder to the other. Under the assumption of using large, random interleavers, the extrinsic information messages $\Pi(\cdot)$ can be considered as statistically independent and identically distributed random variables approaching Gaussian-like distributions $p_\Pi(\xi)$ with increasing number of iterations of the iterative decoder. As shown in [31], the probability density function $p_\Pi(\xi)$ is symmetric and, as a consequence, the statistical mean $\mu_\Pi = E(\Pi)$ represents the discrimination between the two probability density functions $p_\Pi(\xi)$ and $p_\Pi(-\xi)$. Based on these assumptions, the logarithmic version of the SISO input/output can be modelled as [29]:

$$\Pi = \log \frac{P(\cdot|1)}{P(\cdot|0)} = \mu_\Pi \cdot a + n \qquad (6)$$

where a is a binary antipodal symbol (± 1), n is a Gaussian noise with zero mean and variance σ_Π^2, and $\mu_\Pi = \frac{\sigma_\Pi^2}{2}$ is the mean of Π.

The random variable Π can be analyzed by specifying a signal-to-noise ratio defined as $SNR = \frac{\mu_\Pi^2}{\sigma_\Pi^2}$. The latter approach was first proposed in [30]. Under the assumption that $p_\Pi(\xi)$ is both Gaussian and symmetric, it is possible to observe that $\sigma_\Pi^2 = 2 \cdot \mu_\Pi$. The previous relation yields $SNR = \mu_\Pi/2$.

The iterative decoder shown in Fig. 1 can be described by a nonlinear dynamical feedback system, involving two SISO decoders whose behaviour is described by two nonlinear functions defined as follows. The nonlinear function accomplished by the inner decoder on its input variables, namely $SNRI_i$

TABLE I
OPTIMIZED RATE $\frac{1}{2}$, RECURSIVE CONSTITUENT ENCODERS FOR CONCATENATED CHANNEL CODES.

$R = \frac{1}{2}$	$G(D) = \left[1, \frac{g_1}{g_o}\right]$	Distance Spectra–d_i, m_i, w_i	d_2, d_3
$\nu = 2, d_2$	$G(D) = \left[1, \frac{5}{7}\right]$	$5, 1, 3 - 6, 2, 6 - 7, 4, 14 - 8, 8, 32 - 9, 16, 72$	$d_2 = 6, d_3 = 5$
$\nu = 2, snr$	$G(D) = \left[1, \frac{7}{5}\right]$	$5, 1, 2 - 6, 2, 6 - 7, 4, 14 - 8, 8, 32 - 9, 16, 72$	$d_2 = 5, d_3 = \infty$
$\nu = 3, d_2$	$G(D) = \left[1, \frac{17}{13}\right]$	$6, 1, 4 - 7, 3, 9 - 8, 5, 20 - 9, 11, 51 - 10, 25, 124$	$d_2 = 8, d_3 = 7$
$\nu = 3, snr$	$G(D) = \left[1, \frac{15}{17}\right]$	$6, 1, 2 - 7, 3, 12 - 8, 5, 20 - 9, 11, 48 - 10, 25, 126$	$d_2 = 6, d_3 = \infty$
$\nu = 4, d_2$	$G(D) = \left[1, \frac{35}{23}\right]$	$7, 2, 8 - 8, 3, 12 - 9, 4, 16 - 10, 16, 84 - 11, 37, 213$	$d_2 = 12, d_3 = 7$
$\nu = 4, snr$	$G(D) = \left[1, \frac{23}{35}\right]$	$7, 2, 6 - 8, 3, 12 - 9, 4, 20 - 10, 16, 76 - 11, 37, 194$	$d_2 = 7, d_3 = \infty$
$\nu = 5, d_3$	$G(D) = \left[1, \frac{71}{53}\right]$	$8, 3, 12 - 10, 16, 84 - 12, 68, 406 - 14, 860, 6516 - 16, 3812, 30620$	$d_2 = 12, d_3 = \infty$
$\nu = 5, snr, d_2$	$G(D) = \left[1, \frac{67}{51}\right]$	$8, 2, 7 - 10, 20, 110 - 12, 68, 398 - 14, 469, 3364 - 16, 2560, 20864$	$d_2 = 20, d_3 = 8$

TABLE II
PUNCTURING TABLE FOR 4-STATE RECURSIVE CONSTITUENT ENCODERS.

$G(D) = \left[1, \frac{5}{7}\right]$	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.13, (3,1,3)	p.56, (3,4,10)	p.253, (2,4,10)	p.1253, (2,2,4)	p.3253, (2,4,10)	p.25253, (2,7,14)	p.125253, (2,9,18)
	$d_2 = 4, d_3 = 3$	$d_2 = 3, d_3 = 3$	$d_2 = 2, d_3 = 3$	$d_2 = 2, d_3 = 3$	$d_2 = 2, d_3 = 3$	$d_2 = 2, d_3 = 3$	$d_2 = 2, d_3 = 3$
$d_{2,BER,opt}$	p.7, (3, 1, 3)	p.27, (2, 1, 4)	p.067, (2, 4, 13)	p.527, (2, 6,26)	p.3525, (2, 84,2693)	p.12527, (2, 15,74)	p.72525, (2, 153,5216)
	$d_2 = 5, d_3 = 3$	$d_2 = 4, d_3 = 3$	$d_2 = 4, d_3 = 2$	$d_2 = 4, d_3 = 2$	$d_2 = 4, d_3 = 2$	$d_2 = 4, d_3 = 2$	$d_2 = 4, d_3 = 2$

$G(D) = \left[1, \frac{5}{7}\right]$	$R = \frac{3}{5}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.57, (4,3,8)	p.257, (4,23,66)	p.277, (4,3,10)	p.1273, (3,4,10)	p.1277, (4,11,30)	p.1377, (4,2,6)	p.5656, (3,4,10)
	$d_2 = 4, d_3 = 4$	$d_2 = 4, d_3 = 4$	$d_2 = 5, d_3 = 4$	$d_2 = 3, d_3 = 3$	$d_2 = 4, d_3 = 4$	$d_2 = 5, d_3 = 4$	$d_2 = 3, d_3 = 3$
$d_{2,BER,opt}$	p.57, (4,3,8)	p.167, (3,1,3)	p.277, (4,3,10)	p.573, (3,7,23)	p.577, (3,3,11)	p.1377, (4,2,6)	p.3173, (3,12,42)
	$d_2 = 4, d_3 = 4$	$d_2 = 4, d_3 = 3$	$d_2 = 5, d_3 = 4$	$d_2 = 4, d_3 = 3$	$d_2 = 5, d_3 = 3$	$d_2 = 5, d_3 = 4$	$d_2 = 4, d_3 = 3$

$G(D) = \left[1, \frac{5}{7}\right]$	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.5673, (3,1,3)	p.5757, (4,3,8)	p.5777, (4,2,6)	p.25273, (2,1,2)	p.25657, (3,4,10)	p.25757, (4,18,52)	p.27677, (4,6,18)
	$d_2 = 4, d_3 = 3$	$d_2 = 4, d_3 = 4$	$d_2 = 5, d_3 = 4$	$d_2 = 2, d_3 = 3$	$d_2 = 3, d_3 = 3$	$d_2 = 4, d_3 = 4$	$d_2 = 4, d_3 = 4$
$d_{2,BER,opt}$	p.3567, (3,1,3)	p.3577, (4,3,8)	p.5777, (4,2,6)	p.12573, (2,2,6)	p.14757, (3,7,23)	p.12777, (2,1,3)	p.16777, (3,1,3)
	$d_2 = 5, d_3 = 3$	$d_2 = 5, d_3 = 3$	$d_2 = 5, d_3 = 4$	$d_2 = 4, d_3 = 2$	$d_2 = 4, d_3 = 3$	$d_2 = 5, d_3 = 2$	$d_2 = 5, d_3 = 3$

$G(D) = \left[1, \frac{5}{7}\right]$	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.27777, (4,2,6)	p.125273, (2,1,2)	p.127273, (3,8,20)	p.127657, (4,23,66)	p.127677, (4,15,43)	p.137677, (4,3,10)	p.137777, (4,2,6)
	$d_2 = 5, d_3 = 4$	$d_2 = 2, d_3 = 3$	$d_2 = 3, d_3 = 3$	$d_2 = 4, d_3 = 4$	$d_2 = 4, d_3 = 4$	$d_2 = 5, d_3 = 4$	$d_2 = 5, d_3 = 4$
$d_{2,BER,opt}$	p.27777, (4,2,6)	p.52573, (2,4,13)	p.63673, (3,11,35)	p.73567, (3,1,3)	p.73577, (3,2,6)	p.137677, (4,3,10)	p.137777, (4,2,6)
	$d_2 = 5, d_3 = 4$	$d_2 = 4, d_3 = 2$	$d_2 = 4, d_3 = 2$	$d_2 = 5, d_3 = 3$	$d_2 = 5, d_3 = 3$	$d_2 = 5, d_3 = 4$	$d_2 = 5, d_3 = 4$

$G(D) = \left[1, \frac{7}{5}\right]$	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.15, (3,1,2)	p.33, (3,6,18)	p.147, (2,1,2)	p.647, (2,2,4)	p.3247, (2,4,8)	p.15247, (2,6,12)	p.65247, (2,9,18)
	$d_2 = 3$	$d_2 = 3$	$d_2 = 2$	$d_2 = 2$	$d_2 = 2$	$d_2 = 2$	$d_2 = 2$
$d_{2,BER,opt}$	p.15, (3,1,2)	p.33, (3,6,18)	p.133, (2,1,4)	p.527, (2,2,10)	p.2533, (2,4,18)	p.12533, (2,7,36)	p.52527, (2,9,54)
	$d_2 = 3$	$d_2 = 3$	$d_2 = 3$	$d_2 = 3$	$d_2 = 3$	$d_2 = 3$	$d_2 = 3$

$G(D) = \left[1, \frac{7}{5}\right]$	$R = \frac{3}{5}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.37, (4,3,8)	p.137, (4,23,92)	p.177, (4,3,6)	p.765, (3,4,12)	p.775, (4,11,36)	p.777, (4,2,4)	p.3333, (3,6,18)
	$d_2 = 4$	$d_2 = 4$	$d_2 = 4$	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$	$d_2 = 3$
$d_{2,BER,opt}$	p.37, (4,3,8)	p.137, (4,23,92)	p.177, (4,3,6)	p.765, (3,4,12)	p.775, (4,11,36)	p.777, (4,2,4)	p.3333, (3,6,18)
	$d_2 = 4$	$d_2 = 4$	$d_2 = 4$	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$	$d_2 = 3$

$G(D) = \left[1, \frac{7}{5}\right]$	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.6735, (3,1,2)	p.3737, (4,3,8)	p.3777, (4,2,4)	p.15333, (2,1,2)	p.13567, (3,5,14)	p.13737, (4,18,56)	p.17577, (4,6,14)
	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$	$d_2 = 2$	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$
$d_{2,BER,opt}$	p.6735, (3,1,2)	p.3737, (4,3,8)	p.3777, (4,2,4)	p.13336, (2,1,4)	p.13567, (3,5,14)	p.13737, (4,18,56)	p.17577, (4,6,14)
	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$	$d_2 = 3$	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$

$G(D) = \left[1, \frac{7}{5}\right]$	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.17777, (4,2,4)	p.65553, (2,1,2)	p.56567, (3,8,26)	p.57537, (4,23,92)	p.57577, (4,15,44)	p.77577, (4,3,6)	p.77777, (4,2,4)
	$d_2 = 4$	$d_2 = 2$	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$	$d_2 = 4$	$d_2 = 4$
$d_{2,BER,opt}$	p.17777, (4,2,4)	p.55533, (2,1,4)	p.56567, (3,8,26)	p.57537, (4,23,92)	p.57577, (4,15,44)	p.77577, (4,3,6)	p.77777, (4,2,4)
	$d_2 = 4$	$d_2 = 3$	$d_2 = 3$	$d_2 = 4$	$d_2 = 4$	$d_2 = 4$	$d_2 = 4$

and E_b/N_o, can be described as:

$$SNRI_o = G_{inn}\left(SNRI_i, \frac{E_b}{N_o}\right) \qquad (7)$$

whereby $SNRI_i$ measures the quality of the extrinsic information exchanged between the two SISO decoders during the iterative decoding algorithm, while E_b/N_o measures the quality of the channel symbols normalized per information bit.

In the same manner, we can identify the input-output relation for the outer SISO decoder by the following equation:

$$SNRO_o = G_{out}\left(SNRO_i\right), \qquad (8)$$

and the input-output relation between the input variable $SNRO_i$ and the $SNRO$ evaluated on the soft output, $\Pi^o(u; O)$, of the information bits, as follows:

$$SNRO = G_{dec}(SNRO_i). \qquad (9)$$

Since mathematical expressions for characterizing the functions $G_{inn}(\cdot)$, $G_{out}(\cdot)$, and $G_{dec}(\cdot)$ are very hard to define, in the proposed algorithm these functions are empirically evaluated by Monte Carlo simulation.

Starting from the nonlinear model of the iterative decoder, the design of good SCCCs operating in the waterfall region

TABLE III

PUNCTURING TABLE FOR 8-STATE RECURSIVE CONSTITUENT ENCODERS.

$G(D) = [1, \frac{17}{13}]$	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.13, (4,3,10) $d_2 = 4, d_3 = 5$	p.33, (4,29,128) $d_2 = 4, d_3 = 4$	p.172, (3,5,18) $d_2 = 4, d_3 = 3$	p.1253, (3,15,40) $d_2 = 3, d_3 = 3$	p.5253, (2,1,2) $d_2 = 2, d_3 = 3$	p.15256, (2,2,6) $d_2 = 3, d_3 = 2$	p.125253, (2,4,8) $d_2 = 2, d_3 = 3$
$d_{2,BER,opt}$	p.7, (3, 1, 4) $d_2 = 7, d_3 = 4$	p.27, (3, 3, 14) $d_2 = 6, d_3 = 4$	p.135, (2, 23,598) $d_2 = 6, d_3 = 4$	p.527, (2, 3,18) $d_2 = 6, d_3 = 4$	p.2527, (2, 4,20) $d_2 = 6, d_3 = 4$	p.12527, (2, 7,42) $d_2 = 6, d_3 = 4$	p.56525, (2, 83,2892) $d_2 = 6, d_3 = 4$

$G(D) = [1, \frac{17}{13}]$	$R = \frac{3}{5}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.57, (4,1,3) $d_2 = 5, d_3 = 4$	p.257, (4,3,10) $d_2 = 4, d_3 = 5$	p.277, (4,1,4) $d_2 = 6, d_3 = 6$	p.1257, (4,19,59) $d_2 = 4, d_3 = 4$	p.1277, (4,2,7) $d_2 = 5, d_3 = 4$	p.1377, (5,4,14) $d_2 = 6, d_3 = 5$	p.3333, (4,29,128) $d_2 = 4, d_3 = 4$
$d_{2,BER,opt}$	p.37, (4,1,4) $d_2 = 7, d_3 = 5$	p.167, (3,1,4) $d_2 = 7, d_3 = 4$	p.177, (4,1,4) $d_2 = 7, d_3 = 5$	p.557, (3,2,6) $d_2 = 6, d_3 = 3$	p.577, (4,5,21) $d_2 = 7, d_3 = 4$	p.777, (5,5,19) $d_2 = 7, d_3 = 5$	p.2727, (3,3,14) $d_2 = 6, d_3 = 4$

$G(D) = [1, \frac{17}{13}]$	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.5673, (4,3,10) $d_2 = 4, d_3 = 5$	p.5377, (4,1,3) $d_2 = 5, d_3 = 4$	p.5777, (5,3,11) $d_2 = 6, d_3 = 5$	p.15273, (3,7,20) $d_2 = 3, d_3 = 3$	p.15557, (4,20,70) $d_2 = 4, d_3 = 4$	p.25377, (4,3,9) $d_2 = 5, d_3 = 4$	p.27377, (5,12,39) $d_2 = 5, d_3 = 5$
$d_{2,BER,opt}$	p.3567, (3,1,4) $d_2 = 7, d_3 = 4$	p.3737, (4,1,4) $d_2 = 7, d_3 = 5$	p.3777, (5,5,19) $d_2 = 7, d_3 = 5$	p.12577, (2,1,4) $d_2 = 6, d_3 = 4$	p.13567, (3,3,12) $d_2 = 6, d_3 = 4$	p.17177, (4,8,31) $d_2 = 6, d_3 = 4$	p.17776, (4,1,4) $d_2 = 6, d_3 = 5$

$G(D) = [1, \frac{17}{13}]$	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.27777, (5,2,8) $d_2 = 6, d_3 = 6$	p.75172, (3,5,18) $d_2 = 4, d_3 = 3$	p.125657, (3,2,4) $d_2 = 3, d_3 = 4$	p.135673, (4,3,10) $d_2 = 4, d_3 = 4$	p.125777, (4,2,6) $d_2 = 5, d_3 = 4$	p.135777, (5,10,34) $d_2 = 4, d_3 = 4$	p.137777, (5,2,8) $d_2 = 7, d_3 = 6$
$d_{2,BER,opt}$	p.27777, (5,2,8) $d_2 = 6, d_3 = 6$	p.52567, (2,2,8) $d_2 = 6, d_3 = 4$	p.57167, (3,6,25) $d_2 = 6, d_3 = 3$	p.73567, (3,1,4) $d_2 = 7, d_3 = 4$	p.73737, (4,6,26) $d_2 = 7, d_3 = 5$	p.76777, (4,1,4) $d_2 = 7, d_3 = 5$	p.137777, (5,2,8) $d_2 = 7, d_3 = 6$

$G(D) = [1, \frac{15}{17}]$	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.7, (4,3,12) $d_2 = 4$	p.33, (4,29,126) $d_2 = 4$	p.136, (3,5,16) $d_2 = 3$	p.656, (3,15,50) $d_2 = 3$	p.3256, (2,1,2) $d_2 = 2$	p.15652, (2,2,4) $d_2 = 2$	p.65256, (2,3,6) $d_2 = 2$
$d_{2,BER,opt}$	p.7, (4,3,12) $d_2 = 4$	p.33, (4,29,126) $d_2 = 4$	p.127, (2,1,6) $d_2 = 4$	p.527, (3,15,92) $d_2 = 4$	p.2527, (2,1,6) $d_2 = 4$	p.12565, (2,37,1518) $d_2 = 4$	p.52527, (2,4,36) $d_2 = 4$

$G(D) = [1, \frac{15}{17}]$	$R = \frac{3}{5}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.57, (4,1,2) $d_2 = 5$	p.167, (4,3,12) $d_2 = 4$	p.177, (4,1,2) $d_2 = 4$	p.537, (4,19,98) $d_2 = 4$	p.677, (4,2,6) $d_2 = 4$	p.1377, (5,5,12) $d_2 = 5$	p.3333, (4,29,126) $d_2 = 4$
$d_{2,BER,opt}$	p.37, (4,1,4) $d_2 = 5$	p.167, (4,3,12) $d_2 = 4$	p.177, (4,1,2) $d_2 = 4$	p.537, (4,19,98) $d_2 = 4$	p.737, (4,3,12) $d_2 = 5$	p.1377, (5,5,12) $d_2 = 5$	p.3333, (4,29,126) $d_2 = 4$

$G(D) = [1, \frac{15}{17}]$	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.3567, (4,3,12) $d_2 = 4$	p.3776, (4,1,2) $d_2 = 4$	p.3777, (5,3,8) $d_2 = 5$	p.15656, (3,8,22) $d_2 = 3$	p.15557, (4,20,78) $d_2 = 4$	p.15776, (4,5,12) $d_2 = 4$	p.15777, (5,13,42) $d_2 = 5$
$d_{2,BER,opt}$	p.3567, (4,3,12) $d_2 = 4$	p.3377, (4,1,4) $d_2 = 5$	p.3777, (5,3,8) $d_2 = 5$	p.12766, (3,7,34) $d_2 = 4$	p.15557, (4,20,78) $d_2 = 4$	p.12777, (4,3,12) $d_2 = 5$	p.15777, (5,13,42) $d_2 = 5$

$G(D) = [1, \frac{15}{17}]$	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.17777, (5,2,4) $d_2 = 5$	p.57136, (3,5,16) $d_2 = 3$	p.57656, (3,3,6) $d_2 = 3$	p.73567, (4,3,12) $d_2 = 4$	p.57737, (4,2,4) $d_2 = 4$	p.67777, (5,11,32) $d_2 = 5$	p.77777, (5,2,4) $d_2 = 5$
$d_{2,BER,opt}$	p.17777, (5,2,4) $d_2 = 5$	p.52733, (3,14,78) $d_2 = 4$	p.55733, (3,2,8) $d_2 = 4$	p.52777, (3,1,4) $d_2 = 5$	p.53777, (4,2,8) $d_2 = 5$	p.67777, (5,11,32) $d_2 = 5$	p.77777, (5,2,4) $d_2 = 5$

can be accomplished in two steps. The first step consists in choosing a punctured outer encoder which maximizes the SCCC interleaver gain: the constituent outer encoder and the related PP for achieving a specific code rate, both of them satisfying the criterion C_1, suit well this purpose. The second step aims at matching the outer and the inner encoders (the latter ones belonging to the codes found by applying criterion C_2) by choosing the PP for the inner encoder which minimizes the BER after a target number of iterations N_{it}. The choice of the inner encoder PP is done by tracing for each PP (belonging to the set of candidate PPs guaranteeing the desired inner code rate) the SNR exit-charts described by (7) and (8), and, then, by choosing the PP yielding the maximum value of $SNRO(N_{it})$ after $N_{it} = 10$ iterations. A key observation here is the fact that maximizing the $SNRO(N_{it})$ after an arbitrary number of iterations N_{it} of the iterative decoding algorithm is such that the probability density function $p_\Pi(\xi)$ can be easily discriminated from $p_\Pi(-\xi)$, potentially yielding lower bit error probabilities after the iterative decoding [30]. Consider the block diagram of the iterative decoder shown in Fig. 1. The soft output $\Pi^o(u; O)$ on the source bits at the output of the outer decoder can be expressed as follows:

$$\Pi^o(u; O) = \widetilde{\Pi}^o(c; I) + \widetilde{\Pi}^o(c; O) \qquad (10)$$

for systematic outer encoders (the subscript $\widetilde{\Pi}$ indicates that the considered Π take into account only the systematic bits).

For non-systematic outer encoders, we can only state that $\Pi^o(u; O)$ is a function, say $f(\cdot)$, of $\Pi^o(c; I) + \Pi^o(c; O)$.

Based on the assumptions deduced above, both soft information $\widetilde{\Pi}^o(c; I)$ and $\widetilde{\Pi}^o(c; O)$ are Gaussian distributed. As a consequence, the outer decoder soft output $\Pi^o(u; O)$ is supposed to be Gaussian distributed with variance $\sigma^2_{\Pi^o(u;O)}$ and mean value $\mu_{\Pi^o(u;O)} = \frac{\sigma^2_{\Pi^o(u;O)}}{2}$. The bit error probability P_b can then be evaluated as follows:

$$P_b \approx \frac{1}{2} erfc \left(\frac{1}{\sqrt{2}} \frac{\mu_{\Pi^o(u;O)}}{\sigma_{\Pi^o(u;O)}} \right) = \frac{1}{2} erfc \left(\frac{\sqrt{\mu_{\Pi^o(u;O)}}}{2} \right) =$$
$$= \frac{1}{2} erfc \left(\sqrt{\frac{SNRO}{2}} \right) .$$
$$(11)$$

This equation shows that the $SNRO(N_{it})$ can be used as a cost function the maximization of which can lead to lower BER.

Punctured SCCCs designed with this technique, present a low error-floor due to the good distance spectra of the outer encoders embedded in the serial concatenation, supported by early converging punctured inner encoders.

Before concluding the section, let us discuss the parameters employed for the proposed coding search technique. The block size used for tracing exit-charts was set to 200.000: such a high value is needed for guaranteeing the hypotheses discussed above on the statistical behaviour of the extrinsic messages $\Pi(\cdot)$. Notice that such values of block size have been also proposed by ten Brink [29] in relation to the density evolution

TABLE IV

PUNCTURING TABLE FOR THE 16-STATE RECURSIVE CONSTITUENT ENCODERS.

$G(D) = [1, \frac{23}{35}]$

	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.13, (4,1,2); $d_2=4$	p.17, (4,29,150); $d_2=4$	p.156, (3,3,12); $d_2=3$	p.636, (3,5,20); $d_2=3$	p.2746, (3,6,26); $d_2=3$	p.13632, (3,13,52); $d_2=3$	p.65166, (3,24,114); $d_2=3$
$d_{2,BER,opt}$	p.7, (4,1,4); $d_2=6$	p.27, (3,1,6); $d_2=5$	p.127, (3,4,26); $d_2=5$	p.725, (3,2,10); $d_2=5$	p.6525, (3,12,94); $d_2=5$	p.12527, (2,1,10); $d_2=5$	p.52527, (2,1,10); $d_2=5$

$G(D) = [1, \frac{23}{35}]$

	$R = \frac{3}{5}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.37, (5,1,4); $d_2=6$	p.273, (4,1,2); $d_2=4$	p.177, (5,1,4); $d_2=6$	p.736, (4,4,14); $d_2=4$	p.757, (5,9,30); $d_2=5$	p.777, (5,1,4); $d_2=6$	p.3347, (4,18,80); $d_2=4$
$d_{2,BER,opt}$	p.37, (5,1,4); $d_2=6$	p.167, (4,1,4); $d_2=4$	p.177, (5,1,4); $d_2=6$	p.537, (4,6,26); $d_2=5$	p.577, (4,1,4); $d_2=6$	p.777, (5,1,4); $d_2=6$	p.2537, (4,19,110); $d_2=5$

$G(D) = [1, \frac{23}{35}]$

	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.5673, (4,1,2); $d_2=4$	p.3737, (5,1,4); $d_2=6$	p.3777, (6,8,28); $d_2=6$	p.13746, (3,1,2); $d_2=3$	p.16736, (4,4,14); $d_2=4$	p.17357, (4,1,2); $d_2=4$	p.16777, (5,4,12); $d_2=5$
$d_{2,BER,opt}$	p.3567, (4,1,4); $d_2=6$	p.3737, (5,1,4); $d_2=6$	p.3777, (6,8,28); $d_2=6$	p.12766, (3,1,4); $d_2=5$	p.13567, (4,5,22); $d_2=5$	p.16737, (4,1,4); $d_2=5$	p.16777, (5,4,12); $d_2=5$

$G(D) = [1, \frac{23}{35}]$

	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.27777, (5,1,2); $d_2=5$	p.53746, (4,96,622); $d_2=4$	p.73572, (4,12,46); $d_2=4$	p.135673, (4,1,2); $d_2=4$	p.75677, (5,10,32); $d_2=5$	p.76777, (5,1,4); $d_2=6$	p.77777, (6,7,24); $d_2=6$
$d_{2,BER,opt}$	p.27777, (5,1,2); $d_2=5$	p.52747, (3,2,8); $d_2=5$	p.56747, (4,11,58); $d_2=5$	p.73567, (4,1,4); $d_2=6$	p.73737, (5,9,40); $d_2=6$	p.76777, (5,1,4); $d_2=6$	p.77777, (6,7,24); $d_2=6$

$G(D) = [1, \frac{35}{23}]$

	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.13, (4,1,3); $d_2=6, d_3=4$	p.17, (4,29,152); $d_2=6, d_3=4$	p.266, (3,2,6); $d_2=5, d_3=3$	p.1352, (3,2,6); $d_2=4, d_3=3$	p.5346, (3,7,24); $d_2=4, d_3=3$	p.13266, (3,13,59); $d_2=5, d_3=3$	p.53346, (3,24,129); $d_2=6, d_3=3$
$d_{2,BER,opt}$	p.7, (4,1,5); $d_2=11, d_3=5$	p.27, (3,1,5); $d_2=10, d_3=4$	p.325, (2,1,8); $d_2=10, d_3=4$	p.1525, (2,2,10); $d_2=10, d_3=4$	p.6525, (2,6,42); $d_2=10, d_3=4$	p.16525, (2,56,1765); $d_2=10, d_3=4$	p.152525, (2,7,50); $d_2=10, d_3=4$

$G(D) = [1, \frac{35}{23}]$

	$R = \frac{3}{5}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.56, (5,1,3); $d_2=8, d_3=5$	p.273, (4,1,3); $d_2=6, d_3=4$	p.277, (5,1,3); $d_2=8, d_3=5$	p.637, (4,4,15); $d_2=6, d_3=4$	p.1357, (5,8,30); $d_2=8, d_3=4$	p.1377, (5,1,4); $d_2=10, d_3=6$	p.5257, (4,19,63); $d_2=4, d_3=4$
$d_{2,BER,opt}$	p.37, (5,4,17); $d_2=10, d_3=5$	p.167, (4,1,5); $d_2=11, d_3=5$	p.177, (5,2,8); $d_2=11, d_3=5$	p.567, (3,1,5); $d_2=10, d_3=5$	p.737, (4,1,5); $d_2=10, d_3=5$	p.1377, (5,1,4); $d_2=10, d_3=6$	p.2727, (3,1,5); $d_2=10, d_3=4$

$G(D) = [1, \frac{35}{23}]$

	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.3673, (4,2,6); $d_2=7, d_3=4$	p.5656, (5,1,3); $d_2=8, d_3=5$	p.5777, (6,8,30); $d_2=9, d_3=6$	p.25273, (3,1,3); $d_2=4, d_3=3$	p.15673, (4,4,14); $d_2=6, d_3=4$	p.27357, (4,1,3); $d_2=7, d_3=4$	p.27677, (5,4,16); $d_2=7, d_3=5$
$d_{2,BER,opt}$	p.7653, (4,1,5); $d_2=11, d_3=5$	p.3577, (4,1,5); $d_2=11, d_3=5$	p.3777, (5,2,10); $d_2=11, d_3=6$	p.13535, (3,4,20); $d_2=10, d_3=4$	p.13537, (3,1,5); $d_2=10, d_3=4$	p.16737, (4,2,10); $d_2=11, d_3=5$	p.16777, (4,1,5); $d_2=11, d_3=6$

$G(D) = [1, \frac{35}{23}]$

	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.27777, (5,1,5); $d_2=9, d_3=6$	p.63257, (4,96,475); $d_2=4, d_3=4$	p.67267, (4,10,38); $d_2=6, d_3=4$	p.67373, (4,1,3); $d_2=7, d_3=4$	p.135757, (5,9,31); $d_2=7, d_3=5$	p.137377, (5,1,3); $d_2=8, d_3=5$	p.137777, (6,7,26); $d_2=10, d_3=6$
$d_{2,BER,opt}$	p.17777, (5,1,5); $d_2=11, d_3=6$	p.52567, (2,1,5); $d_2=10, d_3=4$	p.52767, (3,4,23); $d_2=10, d_3=4$	p.156735, (4,1,5); $d_2=11, d_3=5$	p.73577, (4,1,5); $d_2=11, d_3=5$	p.77577, (5,2,8); $d_2=11, d_3=5$	p.77777, (6,10,42); $d_2=11, d_3=6$

technique applied to the analysis of the iterative decoder of PCCCs. In a practical setting, the decorrelation between the extrinsic messages is guaranteed by the interleaver in the serial concatenation. The value of E_b/N_o used for code design was set to 0.5dB from the Shannon channel capacity assuming BPSK modulation and AWGN channel. This value was chosen based on extensive tests. Suffice to say that it guarantees early converging SCCCs. We invite the interested reader to refer to [32] for the details of the algorithm along with the set of parameters used for code search.

In the following, the results obtained by applying this algorithm will be denoted by criterion C_3.

IV. CODE SEARCH RESULTS AND APPLICATIONS

The results of the search for good PPs are presented in Tables II-IX. In particular, Tables II-V contain the PPs obtained by applying the criteria C_1 and C_2. Each table collects the results related to encoders with the same number of states. Generator matrices are specified in the upper leftmost corner of each table. For any encoder, we show the best PPs specified in the appropriate column yielding a punctured encoder with rate $\frac{k}{k+s}$ (with $k \in \{2,\ldots,8\}$ and $s \in \{1,\ldots,k-1\}$) satisfying the design criteria C_1 and C_2 discussed above. In particular, the upper row specifies the best PPs obtained with the criterion C_1. The lower row, identified by $d_{2,BER,opt}$, specifies the best PP obtained with the criterion C_2. Next to each PP listed, we show the minimum distance d_m, the number of nearest neighbors M_m yielding the minimum distance d_m, and the total weight W_m of these input patterns as the triplet (d_m, M_m, W_m). In the lower part of any given row, we specify the effective free distance d_2, and the weight-3 distance d_3, of the punctured encoders. Where d_3 is not specified, it means that there are no weight-3 input patterns leading to error events of the punctured encoder.

As an example of how to read the table entries, consider the rate 2/3 PP 13 shown in Table II satisfying the criterion C_1 obtained by using the 4-state mother encoder with generator matrix $G(D) = [1, \frac{5}{7}]$. This PP leads to a encoder whose minimum distance 3 is due to one input pattern with weight 3. The effective free distance d_2 of the encoder is 4 and the distance d_3 is 3.

The PPs are represented in octal notation. A given PP should be read from right to left by collecting k-pairs of systematic-parity bits. As an example, the PP in Table II which yields a code with rate 2/3 for the 4-state encoder, should be interpreted as follows: $p = 13_8 = 1011_2 = <x_1, y_1, x_2, y_2>$ (the subscript denotes the base of the numbers). In this case the PP leaves the encoder systematic and deletes the first parity bit associated with every two input bits.

Tables VI-IX show the best PPs for inner encoders obtained by density evolution (criterion C_3) as explained in section III. The tables are organized as follows. In the first column, we

TABLE V

PUNCTURING TABLE FOR 32-STATE RECURSIVE CONSTITUENT ENCODERS.

$G(D) = [1, \frac{71}{53}]$	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.13, (6,15,60) $d_2 = 6$	p.36, (4,1,4) $d_2 = 8$	p.133, (4,28,192) $d_2 = 8$	p.1272, (4, 108,406) $d_2 = 4$	p.5253, (3,3,6) $d_2 = 3$	p.13172, (3,4,18) $d_2 = 5$	p.64752, (3,11,50) $d_2 = 4$
$d_{2,BER,opt}$	p.7, (5,3,16) $d_2 = 11$	p.65, (4,2,12) $d_2 = 10$	p.325, (2,1,8) $d_2 = 10$	p.1525, (3,3,30) $d_2 = 10$	p.6525, (2,2,16) $d_2 = 10$	p.32525, (2,27,864) $d_2 = 10$	p.152525, (2,6,42) $d_2 = 10$

$G(D) = [1, \frac{71}{53}]$	$R = \frac{3}{6}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.57, (6,3,12) $d_2 = 8$	p.273, (6,15,60) $d_2 = 6$	p.277, (6,2,8) $d_2 = 8$	p.756, (4,1,4) $d_2 = 6$	p.757, (5,3,12) $d_2 = 8$	p.777, (6,3,12) $d_2 = 10$	p.3636, (4,1,4) $d_2 = 8$
$d_{2,BER,opt}$	p.37, (5,1,4) $d_2 = 10$	p.167, (5,3,16) $d_2 = 11$	p.177, (6,3,12) $d_2 = 11$	p.1565, (4,6,34) $d_2 = 10$	p.737, (5,4,16) $d_2 = 10$	p.777, (6,3,12) $d_2 = 10$	p.6565, (4,2,12) $d_2 = 10$

$G(D) = [1, \frac{71}{53}]$	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.5673, (6,15,60) $d_2 = 6$	p.5757, (6,3,12) $d_2 = 8$	p.3777, (6,1,4) $d_2 = 11$	p.25273, (4,10,36) $d_2 = 4$	p.17356, (5,17,72) $d_2 = 6$	p.27357, (5,2,8) $d_2 = 7$	p.27677, (6,8,32) $d_2 = 7$
$d_{2,BER,opt}$	p.3567, (5,3,16) $d_2 = 11$	p.7765, (5,1,8) $d_2 = 11$	p.3777, (6,1,4) $d_2 = 11$	p.12727, (4,116,4788) $d_2 = 10$	p.13567, (4,12,362) $d_2 = 10$	p.33735, (5,5,22) $d_2 = 11$	p.17677, (6,10,50) $d_2 = 11$

$G(D) = [1, \frac{71}{53}]$	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.17777, (6,1,4) $d_2 = 11$	p.75256, (4,17,68) $d_2 = 4$	p.74756, (4,2,8) $d_2 = 7$	p.135673, (6,15,60) $d_2 = 6$	p.135677, (6,18,74) $d_2 = 6$	p.137677, (6,2,8) $d_2 = 8$	p.137777, (6,1,4) $d_2 = 10$
$d_{2,BER,opt}$	p.17777, (6,1,4) $d_2 = 11$	p.52567, (3,3,24) $d_2 = 10$	p.52737, (4,8,48) $d_2 = 10$	p.73567, (5,3,16) $d_2 = 11$	p.73577, (5,4,20) $d_2 = 11$	p.77577, (6,3,12) $d_2 = 11$	p.77777, (6,2,8) $d_2 = 11$

$G(D) = [1, \frac{67}{51}]$	$R = \frac{2}{3}$	$R = \frac{3}{4}$	$R = \frac{4}{5}$	$R = \frac{5}{6}$	$R = \frac{6}{7}$	$R = \frac{7}{8}$	$R = \frac{8}{9}$
BER	p.13, (5,2,7) $d_2 = 9, d_3 = 5$	p.53, (4,2,7) $d_2 = 7, d_3 = 4$	p.127, (4,22,191) $d_2 = 18, d_3 = 5$	p.536, (4,31,446) $d_2 = 13, d_3 = 4$	p.3352, (3,6,21) $d_2 = 4, d_3 = 3$	p.12672, (3,5,22) $d_2 = 9, d_3 = 3$	p.124672, (3,11,36) $d_2 = 5, d_3 = 3$
$d_{2,BER,opt}$	p.7, (5,2,10) $d_2 = 19, d_3 = 6$	p.65, (4,7,51) $d_2 = 18, d_3 = 5$	p.325, (4,22,191) $d_2 = 18, d_3 = 5$	p.1525, (3,3,21) $d_2 = 18, d_3 = 5$	p.6525, (3, 4,34) $d_2 = 18, d_3 = 5$	p.32525, (3, 14,135) $d_2 = 18, d_3 = 5$	p.152525, (3, 23,299) $d_2 = 18, d_3 = 5$

$G(D) = [1, \frac{67}{51}]$	$R = \frac{3}{5}$	$R = \frac{4}{6}$	$R = \frac{4}{7}$	$R = \frac{5}{7}$	$R = \frac{5}{8}$	$R = \frac{5}{9}$	$R = \frac{6}{8}$
BER	p.57, (6,6,24) $d_2 = 12, d_3 = 6$	p.273, (5,2,7) $d_2 = 9, d_3 = 5$	p.177, (6,1,4) $d_2 = 19, d_3 = 7$	p.747, (4,1,3) $d_2 = 14, d_3 = 4$	p.737, (5,1,3) $d_2 = 19, d_3 = 5$	p.1377, (6,2,7) $d_2 = 15, d_3 = 6$	p.5353, (4,2,7) $d_2 = 7, d_3 = 4$
$d_{2,BER,opt}$	p.75, (5,1,4) $d_2 = 19, d_3 = 6$	p.167, (5,2,10) $d_2 = 19, d_3 = 6$	p.177, (6,1,4) $d_2 = 19, d_3 = 7$	p.537, (4,2,11) $d_2 = 18, d_3 = 5$	p.737, (5,1,3) $d_2 = 19, d_3 = 5$	p.1737, (6,2,7) $d_2 = 19, d_3 = 6$	p.2537, (4,5,32) $d_2 = 18, d_3 = 5$

$G(D) = [1, \frac{67}{51}]$	$R = \frac{6}{9}$	$R = \frac{6}{10}$	$R = \frac{6}{11}$	$R = \frac{7}{9}$	$R = \frac{7}{10}$	$R = \frac{7}{11}$	$R = \frac{7}{12}$
BER	p.5673, (5,2,7) $d_2 = 9, d_3 = 5$	p.5757, (6,6,24) $d_2 = 12, d_3 = 6$	p.5777, (7,10,36) $d_2 = 15, d_3 = 7$	p.17256, (4,10,39) $d_2 = 7, d_3 = 4$	p.16756, (5,10,47) $d_2 = 11, d_3 = 6$	p.17357, (5,1,4) $d_2 = 12, d_3 = 6$	p.17757, (6,3,11) $d_2 = 15, d_3 = 6$
$d_{2,BER,opt}$	p.3567, (5,2,10) $d_2 = 19, d_3 = 6$	p.3577, (5,1,4) $d_2 = 19, d_3 = 6$	p.3777, (6,1,4) $d_2 = 19, d_3 = 7$	p.12735, (4,14,108) $d_2 = 18, d_3 = 5$	p.13765, (4,1,7) $d_2 = 18, d_3 = 6$	p.12777, (5,3,19) $d_2 = 19, d_3 = 6$	p.32777, (6,5,26) $d_2 = 19, d_3 = 7$

$G(D) = [1, \frac{67}{51}]$	$R = \frac{7}{13}$	$R = \frac{8}{10}$	$R = \frac{8}{11}$	$R = \frac{8}{12}$	$R = \frac{8}{13}$	$R = \frac{8}{14}$	$R = \frac{8}{15}$
BER	p.27777, (7,9,31) $d_2 = 16, d_3 = 7$	p.67652, (4,23,85) $d_2 = 6, d_3 = 4$	p.127273, (4,2,6) $d_2 = 7, d_3 = 4$	p.135673, (5,2,7) $d_2 = 9, d_3 = 5$	p.73757, (5,1,3) $d_2 = 16, d_3 = 5$	p.77577, (6,1,4) $d_2 = 19, d_3 = 7$	p.77777, (7,7,25) $d_2 = 19, d_3 = 7$
$d_{2,BER,opt}$	p.32777, (7,7,25) $d_2 = 19, d_3 = 7$	p.53527, (4,22,191) $d_2 = 18, d_3 = 5$	p.52577, (3,1,7) $d_2 = 18, d_3 = 6$	p.73567, (5,2,10) $d_2 = 19, d_3 = 6$	p.73577, (5,1,4) $d_2 = 19, d_3 = 6$	p.77577, (6,1,4) $d_2 = 19, d_3 = 7$	p.77777, (7,7,25) $d_2 = 19, d_3 = 7$

show the rate $R_o = \frac{k_O}{k_O+1}$ of the punctured outer encoder along with the respective PP, both of them obtained upon application of criterion C_1. The intersection between a given row indexed by the value of the rate of the punctured outer encoder (i.e. $\frac{k_O}{k_O+1}$ where $k_O \in \{1, \ldots, 8\}$), and a column, indexed by the value $R_i = \frac{k_I}{k_I+1}$, indicates the optimal PP which yields an inner encoder with rate $\frac{k_I}{k_I+1}$ where $k_I \in \{2, \ldots, 8\}$. In this case, by neglecting the effects of the encoder terminations, the overall SCCC presents a rate $R_s = \frac{k_O}{k_O+1} \cdot \frac{k_I}{k_I+1}$.

Simulation results of the optimized punctured codes have been obtained for various SCCCs with the coding structure depicted in Fig. 1. The proposed scheme is composed of a serial concatenation of two rate $1/2$ recursive mother encoders that are punctured in order to achieve a encoder with rate $R_s = R_oR_i$. The encoders are separated by a spread-36 interleaver π_N of length $N = 1,344$ designed in according to [35]. This interleaver size has been chosen in order to accommodate various rates for both codes. Puncturing is performed at the output of the outer and the inner codes in the SCCC (see Fig. 1). All simulations have been conducted by counting 100 erroneous frames employing 10 iterations of the MAX*-Log-MAP algorithm with 3-bit quantization as specified in [36]. The assumed modulation format is BPSK.

Consider the simulation results shown in Fig. 2 for the ten rate-$1/2$ SCCCs obtained by concatenating a rate-$2/3$ outer encoder (with generator matrix $G(D) = [1, 7/5]$ and PP

15) with ten rate-$3/4$ inner encoders (with generator matrix $G(D) = [1, 5/7]$). Table VI shows that the best PP for the rate-$3/4$ inner encoder concatenated with the rate-$2/3$ outer encoder punctured with the PP 15, is 56. Simulation results shown in Fig. 2 where the best PP found by criterion C_3 is compared to other candidate PPs yielding a rate-$3/4$ inner encoder, confirm the effectiveness of the design approach C_3. In the legend we show the $SNRO$ obtained after 10 iterations for each PP examined. We invite the interested reader to refer to [32] for further simulation results on SCCCs designed with the criterion C_3.

V. CONCLUSIONS

In this paper we presented extensive optimized puncturing pattern tables for designing serially concatenated convolutional codes (SCCCs) operating in two different objective regions: in the waterfall region and in the so-called error-floor region.

The optimization has been conducted using three different techniques each one of which is suited to a certain application in connection with the design of SCCCs.

We have further conducted exhaustive search for mother encoders of rate $1/2$ to be used for puncturing using two different selection criteria. These encoders have then been used as mother encoders to which puncturing is applied.

Sample simulation results were conducted to verify the functionality of the designed encoders in the concatenated coding context.

TABLE VI

PUNCTURING TABLES FOR SCCC SCHEMES COMPOSED BY 4-STATE RECURSIVE CONSTITUENT ENCODERS.

$G_i(D) = [1, \frac{5}{7}]$ $G_o(D) = [1, \frac{5}{7}]$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=6, d_3=5$	15	65	325	535	4565	25253	152525
$\frac{2}{3}$, p.13, (3,1,3); $d_2=4, d_3=3$	15	47	67	1721	4467	23466	110575
$\frac{3}{4}$, p.56, (3,4,10); $d_2=3, d_3=3$	15	27	135	175	2627	32427	72165
$\frac{4}{5}$, p.253, (2,1,2); $d_2=2, d_3=3$	7	33	332	671	4635	24467	55227
$\frac{5}{6}$, p.1253, (2,2,4); $d_2=2, d_3=3$	7	47	266	671	2653	23546	116331
$\frac{6}{7}$, p.3253, (2,4,10); $d_2=2, d_3=3$	7	33	332	1332	3632	16646	164631
$\frac{7}{8}$, p.25253, (2,7,14); $d_2=2, d_3=3$	7	47	351	1236	3632	14647	117231
$\frac{8}{9}$, p.125253, (2,9,18); $d_2=2, d_3=3$	7	47	153	1172	7232	14647	133146
$G_i(D) = [1, \frac{7}{5}]$ $G_o(D) = [1, \frac{7}{5}]$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=5, d_3=\infty$	13	53	253	1352	5351	25652	125253
$\frac{2}{3}$, p.15, (3,1,2); $d_2=3$	13	56	354	1057	7311	7236	164233
$\frac{3}{4}$, p.33, (3,6,18); $d_2=3$	13	66	73	1271	4754	7233	164532
$\frac{4}{5}$, p.147, (2,1,2); $d_2=2$	13	72	272	1253	7232	7253	127056
$\frac{5}{6}$, p.647, (2,2,4); $d_2=2$	13	72	256	1253	7232	7253	127056
$\frac{6}{7}$, p.3247, (2,4,8); $d_2=2$	13	72	256	1253	7232	7253	127056
$\frac{7}{8}$, p.15247, (2,6,12); $d_2=2$	16	56	253	1253	7232	25352	125253
$\frac{8}{9}$, p.65247, (2,9,18); $d_2=2$	13	72	253	1272	7252	25352	125253
$G_i(D) = [1, \frac{7}{5}]$ $G_o(D) = [1, \frac{5}{5}]$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=5, d_3=\infty$	13	56	346	1370	3346	25263	163146
$\frac{2}{3}$, p.15, (3,1,2); $d_2=3$	16	17	266	276	6172	15274	64770
$\frac{3}{4}$, p.33, (3,6,18); $d_2=3$	15	17	247	663	7234	6633	163432
$\frac{4}{5}$, p.147, (2,1,2); $d_2=2$	16	72	156	1256	5631	25256	114671
$\frac{5}{6}$, p.647, (2,2,4); $d_2=2$	13	72	156	1256	4671	25256	114671
$\frac{6}{7}$, p.3247, (2,4,8); $d_2=2$	13	72	156	1253	5631	25352	114671
$\frac{7}{8}$, p.15247, (2,6,12); $d_2=2$	13	72	156	1272	4671	25352	114671
$\frac{8}{9}$, p.65247, (2,9,18); $d_2=2$	13	56	256	1256	5631	35252	114633

TABLE VII

PUNCTURING TABLE FOR THE 32-STATE RECURSIVE CONSTITUENT INNER ENCODER WITH GENERATOR MATRIX $G(D) = [1, \frac{67}{51}]$. THE OUTER ENCODER IN THE SCCC HAS GENERATOR MATRIX $G(D) = [1, \frac{67}{51}]$.

	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=20, d_3=8$	13	56	256	1316	5253	25256	127252
$\frac{2}{3}$, p.13, (5,2,7); $d_2=9, d_3=5$	7	27	37	137	5272	27252	127252
$\frac{3}{4}$, p.53, (4,2,7); $d_2=7, d_3=4$	6	56	272	1274	7252	21276	25672
$\frac{4}{5}$, p.127, (4,22,191); $d_2=18, d_3=5$	6	56	272	1073	1257	5273	21676
$\frac{5}{6}$, p.536, (4,31,446); $d_2=13, d_3=4$	6	72	272	1073	7252	27252	121276
$\frac{6}{7}$, p.3352, (3,6,21); $d_2=4, d_3=3$	6	53	253	1073	1257	5656	125546
$\frac{7}{8}$, p.12672, (3,5,22); $d_2=9, d_3=3$	6	72	253	1073	5253	5656	37342
$\frac{8}{9}$, p.124672, (3,11,36); $d_2=5, d_3=3$	13	72	272	1073	6272	5656	114656

TABLE VIII

PUNCTURING TABLES FOR SCCC SCHEMES COMPOSED BY 8-STATE RECURSIVE CONSTITUENT ENCODERS.

$G_i(D) = [1, \frac{15}{17}]$ $G_o(D) = [1, \frac{15}{17}]$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=6, d_3=\infty$	13	56	352	1352	1273	27252	65352
$\frac{2}{3}$, p.7, (4,3,12); $d_2=4$	16	72	247	1316	7312	27416	123236
$\frac{3}{4}$, p.33, (4,29,126); $d_2=4$	16	72	172	353	673	1357	5357
$\frac{4}{5}$, p.136, (3,5,16); $d_2=3$	16	72	247	1456	3354	15670	26356
$\frac{5}{6}$, p.656, (3,15,50); $d_2=3$	16	56	156	273	6172	3356	23273
$\frac{6}{7}$, p.3256, (2,1,2); $d_2=2$	16	36	227	1631	7151	17226	134436
$\frac{7}{8}$, p.15652, (2,2,4); $d_2=2$	16	36	227	1227	7151	27231	127107
$\frac{8}{9}$, p.65256, (2,3,6); $d_2=2$	13	36	227	1631	7131	35131	74433
$G_i(D) = [1, \frac{17}{13}]$ $G_o(D) = [1, \frac{15}{17}]$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=6, d_3=\infty$	13	72	272	1272	5253	21653	127252
$\frac{2}{3}$, p.7, (4,3,12); $d_2=4$	16	72	362	1313	1752	5333	27233
$\frac{3}{4}$, p.33, (4,29,126); $d_2=4$	13	56	274	273	1273	1373	23372
$\frac{4}{5}$, p.136, (3,5,16); $d_2=3$	13	72	274	372	1273	3353	22747
$\frac{5}{6}$, p.656, (3,15,50); $d_2=3$	13	72	153	276	277	3353	22753
$\frac{6}{7}$, p.3256, (2,1,2); $d_2=2$	13	55	351	1331	3332	35132	115456
$\frac{7}{8}$, p.15652, (2,2,4); $d_2=2$	16	33	153	1233	6651	17270	164662
$\frac{8}{9}$, p.65256, (2,3,6); $d_2=2$	13	55	172	1331	6646	33151	153162

TABLE IX

PUNCTURING TABLES FOR SCCC SCHEMES COMPOSED BY 16-STATE RECURSIVE CONSTITUENT ENCODERS.

$G_i(D) = [1, \frac{35}{23}]$ $G_o(D) = [1, \frac{23}{35}]$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=7, d_3=\infty$	13	72	272	1253	5272	27252	125256
$\frac{2}{3}$, p.13, (4,1,2); $d_2=4$	13	72	256	1272	7612	5257	25276
$\frac{3}{4}$, p.17, (4,29,150); $d_2=4$	13	72	274	1370	1257	5276	104747
$\frac{4}{5}$, p.156, (3,3,12); $d_2=3$	16	72	272	1370	1257	5273	5357
$\frac{5}{6}$, p.636, (3,5,20); $d_2=3$	13	72	272	276	1273	1657	5357
$\frac{6}{7}$, p.2746, (3,6,26); $d_2=3$	16	72	253	1346	6352	7253	104567
$\frac{7}{8}$, p.13632, (3,13,52); $d_2=3$	13	72	352	1172	6352	1357	77142
$\frac{8}{9}$, p.65166, (3,24,114); $d_2=3$	13	72	316	1346	6352	7253	21373
$G_i(D) = [1, \frac{23}{35}]$ $G_o(D) = [1, \frac{23}{35}]$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{6}{7}$	$\frac{7}{8}$	$\frac{8}{9}$
$\frac{1}{2}$, − ; $d_2=7, d_3=\infty$	13	56	256	1352	5272	25156	125652
$\frac{2}{3}$, p.13, (4,1,2); $d_2=4$	13	56	256	1253	5712	3276	127213
$\frac{3}{4}$, p.17, (4,29,150); $d_2=4$	13	72	352	1057	1656	15370	167054
$\frac{4}{5}$, p.156, (3,3,12); $d_2=3$	13	72	352	1272	4173	33606	167054
$\frac{5}{6}$, p.636, (3,5,20); $d_2=3$	13	72	354	1351	1273	2376	15273
$\frac{6}{7}$, p.2746, (3,6,26); $d_2=3$	13	72	316	1351	5616	7741	164266
$\frac{7}{8}$, p.13632, (3,13,52); $d_2=3$	15	71	331	745	7145	13171	62556
$\frac{8}{9}$, p.65166, (3,24,114); $d_2=3$	13	72	346	1271	5616	3176	143256

REFERENCES

[1] G.D. Forney Jr.: "Concatenated Codes", Cambridge, MA: MIT Press, 1966.

[2] C. Berrou, A. Glavieux, P. Thitimajshima: *Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes*, Proc. 1993 IEEE Int. Conf. on Comm., Geneva, Switzerland, pp.1064-1070, May 1993.

[3] D. Divsalar, F. Pollara: *Turbo Codes for PCS Applications*, Proc. 1995 IEEE Int. Conf. on Comm., Seattle, WA, pp.54-59, May 1995.

[4] S. Benedetto, D. Divsalar, G. Montorsi, F. Pollara: *Serial Concatenation of Interleaved Codes: Performance Analysis, Design, and Iterative Decoding*, IEEE Trans. on Inform. Theory, vol.44, no.3, pp.909-926, May 1998.

[5] J. B. Cain, G. Clark, J. M. Geist: *Punctured Convolutional Codes of Rate $\frac{n-1}{n}$ and Simplified Maximum Likelihood Decoding*, IEEE Trans. on Communications, vol.25, no.1, pp.97-100, January 1979.

[6] K. J. Hole: *Punctured Convolutional Codes for the 1-D Partial-Response Channel*, IEEE Trans. on Inform. Theory, vol.37, no.3, pp.808-817, May 1991.

[7] D. G. Daut, J. W. Modestino, L. D. Wismer: *New Short Constraint Length Convolutional Code Constructions for Selected Rational Rates*, IEEE Trans. on Inform. Theory, vol.28, no.5, pp.794-800, September 1982.

[8] M. Kim: *On Systematic Punctured Convolutional Codes*, IEEE Trans. on Communications, vol.45, no.2, pp.133-139, February 1997.

[9] Y. Yasuda, K. Kashiki, Y. Hirata: *High-Rate Punctured Convolutional Codes for Soft Decision Viterbi Decoding*, IEEE Trans. on Communications, vol.32, no.3, pp.315-319, March 1984.

[10] D. Haccoun, G. Begin: *High-Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding*, IEEE Trans. on Communications, vol.37, no.11, pp.1113-1125, November 1989.

[11] G. Begin, D. Haccoun: *High-Rate Punctured Convolutional Codes: Structure Properties and Construction Technique*, IEEE Trans. on Communications, vol.37, no.12, pp.1381-1385, December 1989.

[12] G. Begin, D. Haccoun, C. Paquin: *Further Results on High-Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding*, IEEE Trans. on Communications, vol.38, no.11, pp.1922-1928, November 1990.

[13] K. J. Hole: *New Short Constraint Length Rate $\frac{N-1}{N}$ Punctured Convolutional Codes for Soft-Decision Viterbi Decoding*, IEEE Trans. on Inf. Theory, vol.34, no.5, pp.1079-1081, September 1988.

[14] P. J. Lee: *New Short Constraint Length, Rate 1/N Convolutional Codes which Minimize the Required SNR for Given Desired Bit Error Rates*, IEEE Trans. on Communications, vol.33, no.2, pp.171-177, February 1985.

[15] P. J. Lee: *Further Results on Rate 1/N Convolutional Code Constructions with Minimum Required SNR Criterion*, IEEE Trans. on Communications, vol.34, no.4, pp.395-399, April 1986.

[16] P. J. Lee: *Constructions of Rate $\frac{n-1}{n}$ Punctured Convolutional Codes with Minimum Required SNR Criterion*, IEEE Trans. on Communications, vol.36, no.10, pp.1171-1174, October 1988.

[17] J. Chang, D. Hwang, M. Lin: *Some Extended Results on the Search for Good Convolutional Codes*, IEEE Trans. on Inf. Theory, vol.43, no.5, pp.1682-1697, September 1997.

[18] J. Hagenauer: *Rate-Compatible Punctured Convolutional Codes (RCPC Codes) and their Applications*, IEEE Trans. on Communications, vol.36, no.4, pp.389-400, April 1988.

[19] F. Babich, G. Montorsi, F. Vatta: *Design of Rate-Compatible Punctured Turbo (RCPT) Codes*, In Proceedings of IEEE International Conference on Communications vol.3, pp.1701-1705, April-May 2002.

[20] F. Babich, G. Montorsi, F. Vatta: *Rate-Compatible Punctured Serial Concatenated Convolutional Codes*, In Proceedings of IEEE Globecom, vol.4, pp.2062-2066, December 2003.

[21] H. Kim, G. L. Stuber: *Rate Compatible Punctured SCCC*, In Proceedings of IEEE 54th VTC 2001 Fall, vol.4, pp.2399-2403, October 2001.

[22] S. S. Pietrobon: *On Punctured Serially Concatenated Turbo Codes*, In Proceedings of 35th Asilomar Conference on Signals, Systems and Computers, vol.1, pp.265-269, November 2001.

[23] O. F. Acikel, W. E. Ryan: *Punctured High Rate SCCCs for BPSK/QPSK Channels*, In Proceedings of IEEE ICC 2000, vol.1, pp.434-439, June 2000.

[24] F. Daneshgaran, M. Laddomada, M. Mondin: *An Extensive Search for Good Punctured Rate k/k+1 Recursive Convolutional Codes for Serially Concatenated Convolutional Codes*, IEEE Trans. on Inform. Theory, vol. IT-50, No. 1, pp. 208-218, January 2004.

[25] F. Daneshgaran, M. Laddomada, M. Mondin: *High-Rate Recursive Convolutional Codes for Concatenated Channel Codes*, IEEE Trans. on Communications, vol.52, No.11, pp.1846-1850, November 2004.

[26] S. Benedetto, R. Garello, G. Montorsi: *A Search for Good Convolutional Codes to Be Used in the Construction of Turbo Codes*, IEEE Transactions on Communications, Vol.46, No.9, pp.1101-1105, September 1998.

[27] D. Divsalar, F. Pollara: *On the Design of Turbo Codes*, JPL TDA Progress Report 42-123, November 1995.

[28] P. Thitimajshima: *Recursive Systematic Convolutional Codes and Application to Parallel Concatenation*, In Proceedings of Global Telecommunications Conference, 1995. GLOBECOM '95, Vol. 3 , pp.2267-2272, November 1995.

[29] S. ten Brink: *Convergence Behavior of Iteratively Decoded Parallel Concatenated Codes*, IEEE Trans. on Communications, vol.49, no.10, pp.1727-1737, October 2001.

[30] D. Divsalar, S. Dolinar, F. Pollara: *Iterative Turbo Decoder Analysis Based on Density Evolution*, IEEE Journal on Selected Areas in Communications, Vol.19, No.5, pp.891-907, May 2001.

[31] T. Richardson, A. Shokrollahi, R. Urbanke: *Design of Capacity-Approaching Irregular Low-Density Parity Check Codes*, IEEE Trans. on Inform. Theory, vol.47, no.2, pp.619-637, February 2001.

[32] M. Laddomada, B. Scanavino: *Design of Punctured Serially Concatenated Convolutional Codes*, IEEE Communications Letters, Vol.9, No.2, pp.169-171, February 2005.

[33] S. Benedetto, G. Montorsi: *Design of Parallel Concatenated Convolutional Codes*, IEEE Trans. on Communications, vol.44, no.5, pp.591-600, May 1996.

[34] S. Chung, T. Richardson, R. Urbanke: *Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation*, IEEE Trans. on Inform. Theory, vol.47, no.2, pp.659-670, February 2001.

[35] C. Fragouli, R.D. Wesel: *Semi-Random Interleaver Design Criteria*, Proc. IEEE Globecom 1999, vol.5 , pp.2352-2356, Dec. 1999.

[36] G. Montorsi, S. Benedetto: *Design of Fixed-Point Iterative Decoders for Concatenated Codes With Interleavers*, IEEE Journal on Selected Areas in Communications, Vol.19, No.5, pp.871-882, May 2001.

Massimiliano Laddomada (S'00-M'03) was born in 1973. He received the degree in Electronics Engineering in 1999, and the Ph.D. degree in Communications Engineering from Politecnico di Torino, in 2003. He is currently a research associate at Politecnico di Torino and a part-time faculty at CSU, Los Angeles.

From June 2000 to March 2001, he was a visiting researcher at CSU (California State University), Los Angeles, and a consultant engineer in Technoconcepts, Inc., Los Angeles, a start up company specializing in Software Radio. He was awarded a five-year open-ended fellowship by E.D.S.U. in recognition of his university career as an Electronics Engineer. In 2003, he was awarded with the *Premio Zucca per l'Innovazione nell'ICT* from Unione Industriale di Turin.

His research is mainly in wireless communications, especially modulation and coding, including turbo codes and, more recently, networks coding.

Massimiliano Laddomada is a member of IEEE. Currently, he is serving as a member of the editorial board of IEEE Communications Surveys and Tutorials.

Bartolo Scanavino (S'00-M'03) was born in 1974. He received the Laurea degree in electronics engineering in 1999, and the Ph.D. degree in communications engineering in 2003, both from Politecnico di Torino, Torino, Italy.

He is currently a Young Researcher at the Politecnico di Torino. From April 2002 to October 2002, he was a visiting researcher at the University of California, San Diego, and a Consulting Engineer with Titan Corporation, San Diego, CA.

His research is mainly in wireless Communications, especially modulation and coding, including turbo codes and, more recently, radio resource Management.