Transactions Papers

On the Performance of High-Rate TPC/SPC Codes and LDPC Codes Over Partial Response Channels

Jing Li, Krishna R. Narayanan, *Member, IEEE*, Erozan Kurtas, *Member, IEEE*, and Costas N. Georghiades, *Fellow, IEEE*

***Abstract*—This paper evaluates two-dimensional turbo product codes based on single-parity check codes (TPC/SPC) and low-density parity check (LDPC) codes for use in digital magnetic recording systems. It is first shown that the combination of a TPC/SPC code and a precoded partial response (PR) channel results in a good distance spectrum due to the interleaving gain. Then, density evolution is used to compute the thresholds for TPC/SPC codes and LDPC codes over PR channels. Analysis shows that TPC/SPC codes have a performance close to that of LDPC codes for large codeword lengths. Simulation results for practical block lengths show that TPC/SPC codes perform as well as LDPC codes in terms of bit error rate, but possess better burst error statistics which is important in the presence of an outer Reed–Solomon code. Further, the encoding complexity of TPC/SPC codes is only linear in the codeword length and the generator matrix does not have to be stored explicitly. Based on the results in the paper and these advantages, TPC/SPC codes seem like a viable alternative to LDPC codes.**

***Index Terms*—Data storage system, density evolution, iterative decoding, low-density parity check codes, message-passing decoding, partial response channels, precoding, turbo product codes.**

I. INTRODUCTION

ENCOURAGED by the near Shannon-limit performance of turbo codes and low-density parity check (LDPC) codes over additive white Gaussian noise (AWGN) channels, concatenated schemes and iterative decoding are being seriously considered for application in future digital magnetic recording systems. After being precoded, filtered, and equalized to some simple partial response (PR) target, the magnetic recording channel appears much like an intersymbol interference (ISI) channel to an outer code and, hence, many of the techniques used in concatenated coding can be adopted. In particular, the observation that an ISI channel can be effectively viewed as a rate-1 convolutional code leads to the natural format of a serial concatenated system where the ISI channel is considered as the inner code and the LDPC code or punctured convolutional code is the outer code. With reasonable complexity, iterative decoding and equalization can be used to obtain good performance gains.

Several researchers have shown that turbo codes based on punctured recursive systematic convolutional codes and LDPC codes can provide about 4–5 dB of coding gain over uncoded systems at bit error rates (BERs) of around 10^{-5} or 10^{-6} [1]–[9]. However, the actual BERs that are of interest in magnetic recording applications are of the order of 10^{-15}. The performance of these codes cannot be easily evaluated at such low BERs and, hence, significant coding gains cannot be guaranteed at such low BERs. Therefore, a t-error correcting Reed–Solomon error correction code (RS-ECC) is typically assumed in addition to the LDPC code or turbo code. In this situation, it is important to ensure that the output of the LDPC or turbo decoder will not contain more than t byte errors that may cause the RS-ECC decoder to fail.

Due to the high decoding complexity of turbo codes, current research focuses on lower complexity solutions that are easily implementable in hardware. Iterative decoding of turbo product codes (TPCs), also referred to as block turbo codes (BTCs) [10]–[13], and LDPC codes in particular [5]–[9], seem to be potential solutions. An LDPC code exhibits similar performance to that of a turbo code, yet with considerably less decoding complexity (about 1/10 that of a turbo decoder). A randomly constructed LDPC code has quadratic encoding complexity in the length N of the code ($O(N^2)$). It has been shown [14] that several greedy algorithms can be applied to triangulate matrices (preprocessing) to reduce encoding complexity, where the required amount of preprocessing is of order at most $N^{3/2}$. With the exception of a few LDPC codes that have cyclic or quasi-cyclic structures [7], large memory is generally required (for storage of generator and/or parity check matrices), which is a big concern in hardware implementation. Furthermore, errors tend to occur in long bursts for LDPC codes [8], [9], which may cause failure of the outer RS-ECC code.

Paper approved by W. E. Ryan, the Editor for Modulation, Coding and Equalization of the IEEE Communications Society. Manuscript received January 16, 2001; revised July 24, 2001, and October 24, 2001. This work was supported by the Texas Higher Education Coordination Board through an ATP grant and the National Science Foundation under Grant CCR-0073506. The work of J. Li was also supported in part by TxTEC. This work was presented in part in the International Conference on Communications (ICC) in Helsinki, Finland, June 2001, and in part in the International Symposium on Information Theory (ISIT) in Washington, DC, June 2001.

J. Li, K. R. Narayanan, and C. N. Georghiades are with the Electrical Engineering Department, Texas A&M University, College Station, TX 77843-3128 USA (e-mail: krishna@ee.tamu.edu).

E. Kurtas is with Seagate Technology, Seagate Research, Pittsburgh, PA 15203 USA.

Publisher Item Identifier S 0090-6778(02)05109-7.

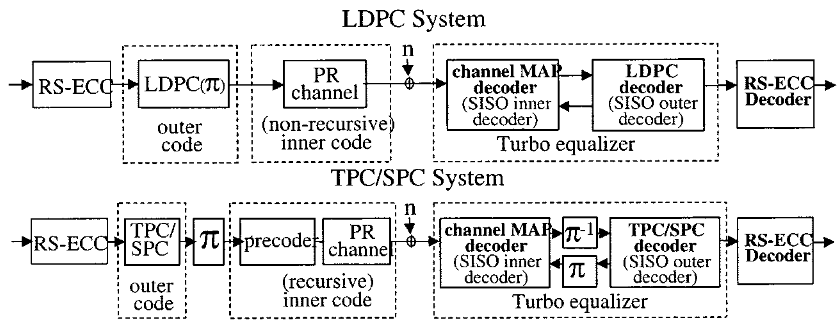

Fig. 1. System model of LDPC and TPC/SPC codes over PR channels.

Single-parity check turbo product codes (TPC/SPC) are a very simple class of TPCs which possess many desirable properties for data storage systems, such as high-rate, linear encoding/decoding complexity and a highly parallelizable encoding/decoding process. While turbo codes and LDPC codes have been under extensive investigation for use in digital magnetic recording, little has been reported about TPC/SPC codes in this area. In this paper, we undertake a comprehensive study of the properties of high-rate TPC/SPC codes and their applicability to digital magnetic recording using precoded PR channels.

We first show that, although TPC/SPC codes have a very small minimum distance, if several codewords are combined and used with an interleaver and a *precoded* PR channel, the distance spectrum improves significantly due to the interleaving gain. This makes the performance of TPC/SPC codes comparable to LDPC codes of the same rate while maintaining the advantage of a slightly lower decoding complexity and linear encoding complexity. Next, we compute the thresholds for iterative decoding of LDPC codes and TPC/SPC codes using density evolution (DE) [15]–[19]. Finally, we study the distribution of errors at the output of the decoder (i.e., at the input to the RS-ECC decoder) and show that TPC/SPC codes have better error distribution, making them better candidates for magnetic recording systems in the presence of an outer RS-ECC code.

The rest of the paper is organized as follows. The system model is presented in Section II, followed by a brief introduction to TPC/SPC codes. Section III analyzes the distance spectrum of the TPC/SPC system. Section IV calculates the thresholds of both TPC/SPC and LDPC systems using density evolution with Gaussian approximation (DE/GA). Certain interesting issues in the optimization of the decoding process are addressed in Section V. Section VI evaluates the performance of both systems, including BER and bit/byte error statistics. Finally, Section VII concludes with a discussion of future work in this area.

II. System Model

As shown in Fig. 1, in the system under study, the data are first encoded using a Reed–Solomon code, which is referred to as the error correction code (ECC). The output of the RS-ECC code is encoded using an outer code. We consider TPC/SPC codes, LDPC codes, and punctured convolutional codes as outer codes. The reason for referring to these codes as outer codes is that we consider the ISI channel as the inner code in the concatenated scheme. When TPC/SPC codes or punctured convolutional codes are considered as the outer codes, the outer codewords are interleaved and then encoded by a rate-1 recursive precoder before being recorded onto the disk. The random interleaver in the above systems works to break the correlation among neighboring bits, to eliminate error bursts, and (in conjunction with the precoder) to improve the overall distance spectrum by mapping low-weight error events to high-weight ones (spectrum thinning). Since the LDPC codes we investigated are constructed randomly using the computer (i.e., there is an embedded random interleaver within the code), an explicit random interleaver is thus not necessary. (Although not shown, simulations show that adding a random interleaver does not improve the performance of our LDPC systems.) Further, for LDPC codes which have quite good distance spectrum, no effective spectrum thinning results by concatenating a rate-1 inner code. As has been shown in [20], no precoding represents the best case for LDPC codes. In fact, the use of precoder results in about 1 dB loss on EPR4 channels with the suboptimal iterative decoding. The channel is modeled as an ISI channel with AWGN. The impulse response of the ISI channel is assumed to be a partial response polynomial with additive white Gaussian noise (Fig. 1). That is,

$$r_k = \sum_{i=0}^{L-1} h_i x_{k-i} + n_k. \tag{1}$$

In this study, we primarily consider the PR4 channel [whose channel polynomial is $H(D) = 1 - D^2$] and the EPR4 channel $(H(D) = 1 + D - D^2 - D^3)$.

Since an overall maximum-likelihood (ML) decoding and equalization of the system is prohibitively complex, the practical yet effective way is to use turbo equalization to iterate soft outputs between the outer decoder and the equalizer, and then feed the hard decision decoding to the RS-ECC code.

A. Introduction to TPC/SPC Codes

A TPC [10] is composed of a multi-dimensional array of codewords from linear block codes like Hamming codes, BCH codes, and parity check codes (see Fig. 2 for the code structure). A two-dimensional (2-D) TPC $\mathcal{C}$, formed from component codes $\mathcal{C}_i \sim (n_i, k_i, d_i, G_i)$, $i = 1, 2$, has parameters $(n_1 n_2, k_1 k_2, d_1 d_2, G_1 \otimes G_2)$, where n, k, d, and G denote the codeword size, user data size, minimum distance, and generator matrix, respectively, and $\otimes$ denotes the Kronecker product. It has been recognized that very simple (almost useless) com-

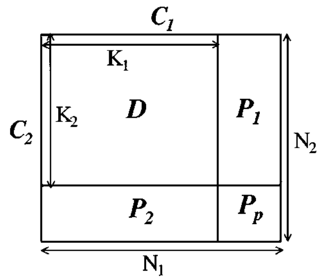

Fig. 2. Structure of a 2-D TPC code (D: user data, P_1: parity from component code $\mathcal{C}_1$, P_2: parity from component code $\mathcal{C}_2$, P_p: parity on parity).

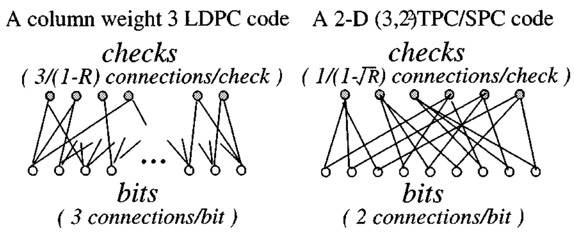

Fig. 3. Bipartite graph representation of LDPC and 2-D TPC/SPC codes.

TABLE I
DECODING COMPLEXITY (NUMBER OF OPERATIONS PER BIT PER ITERATION)

Operations	TPC/SPC	LDPC	MAP
addition	$3d$	$4s$	$15 \cdot 2^m + 9$
min/max			$5 \cdot 2^m - 2$
table lookup	$2d$	$2s$	$5 \cdot 2^m - 2$

d: dimensions of TPC/SPC code

s: average column weight of LDPC code

m: memory size of the convolutional code

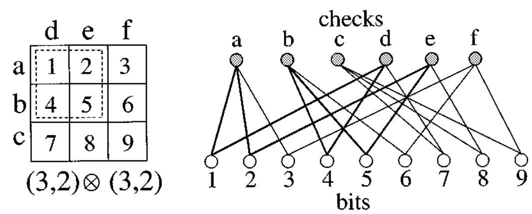

Fig. 4. Illustration of minimum distance for a TPC/SPC code.

ponent codes can result in an overall powerful TPC code. Particularly of interest in this paper are TPC codes formed from single-parity check (SPC) component codes, namely, TPC/SPC codes. Since the encoding operation involves adding a single-parity check bit in each row and column, it is extremely simple and a dense generator matrix need not be explicitly stored as for LDPC codes.

A TPC/SPC code can be interpreted from different perspectives. One particular viewpoint is to model it as a serial concatenation of its component codes with a linear block interleaver in between. From the graph-based point of view, it can also be viewed as a special type of structured regular LDPC code where each row in each dimension satisfies a check (Fig. 3). Each bit node has degree s and each check node has degree K_0+1 for a $(K_0+1, K_0)^s$ s-dimensional TPC/SPC code. Magnetic recording systems require a high code rate since for recording systems code rate loss (in decibels) is of the order of $10\log_{10}(R^2)$ rather than $10\log_{10}(R)$ as in an AWGN channel [21] (R is the code rate). Hence, in this work, we only focus on 2-D TPC/SPC codes with code rate $R=(K_0/(K_0+1))^2$. However, it should be noted that the above properties, as well as the decoding algorithm, are readily extendible to the multidimensional case.

B. Decoding Algorithm of TPC/SPC Codes

While the general treatment of decoding a TPC code is via the Chase algorithm [12], a TPC/SPC decoder can adopt a simple and effective message-passing decoding algorithm. Since each row and column of a TPC/SPC represents a single-parity check, it forms a special case of LDPC codes and, hence, a message-passing decoding algorithm can be used. As explained in [8], the way messages are exchanged in the decoding process is essentially the same as that of LDPC codes except that for LDPC codes all checks are simultaneously updated, whereas for TPC/SPC codes checks are grouped in to two groups (corresponding to component codes $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively) and updated in turn. This "serial" update is expected to converge a little faster than "parallel" update. (The exact decoding steps can be found in [30].)

Table I compares the complexity of TPC/SPC, LDPC, and MAP decoders implementing the BCJR algorithm in the log domain [22] [assuming that $\log(\tanh(x/2))$ and its reverse function $2\tanh^{-1}(e^x)$ are implemented through table lookup, and that multiplications are converted to additions in the log-domain]. We can see that the decoding algorithm for a 2-D TPC/SPC code requires about 2/3 the complexity and about 1/3 the storage space of the decoding algorithm for a regular column-weight-3 LDPC code in each decoding iteration. The decoding algorithm for a punctured convolutional code is considerably higher, although the actual number of iterations needed would be lower.

III. ANALYSIS OF THE DISTANCE SPECTRUM

A. Distance Properties of TPC/SPC Codes (AWGN Channel)

The minimum distance of a randomly constructed regular LDPC code with column weight $j \geq 3$ depends on the actual construction and is hard to determine, but with high probability increases linearly with block length N, especially for large N. Hence, it possesses good error detection capability and the decoding algorithm rarely converges to a wrong codeword. On the other hand, the distance spectrum of a TPC/SPC code is characterizable. It can be seen that a 2-D TPC/SPC code has a minimum distance of 4 and therefore encounters many undetectable errors (Fig. 4). In particular, all rectangular error patterns are undetectable. Therefore, TPC/SPC codes by themselves are quite weak compared to LDPC codes of the same rate. There have been attempts to improve the performance of TPC/SPC codes [23]. Most of these approaches involve adding extra parity checks in more dimensions, thereby reducing the rate of the code. Here, we propose a different modification to the

TPC/SPC code structure which results in significant improvement in performance without loss in data rate. The idea is to group P TPC/SPC codewords together and interleave them before encoding by the precoder. In the following sections, we analyze the properties of such a system.[1]

B. Distance Properties of TPC/SPC Systems (PR Channels)

In this section, we compute the distance spectrum of a TPC/SPC system with a precoded PR4 channel using the ideas in [24]–[26]. Since the precoder is a rate-1 recursive convolutional code, the combination of the ISI channel and the precoder is a recursive ISI channel. Our approach is to consider the overall system as the concatenation of the outer code and the precoded ISI channel (which acts as a recursive inner code). Then, we can compute the distance spectrum of such a system over the ensemble of all possible interleavers such as in [24], [25]. We show that caution should be exercised in extrapolating the results of Benedetto *et al.*, since the results are somewhat unexpected. Hence, it is worth pursuing this exercise.

Let N denote the length of each codeword (effective block size) formed by grouping P TPC/SPC codewords of length $(N/P) = (K_0+1)^2$ each and interleaving them. This length-N codeword is then passed through a precoded PR channel. Each TPC/SPC code has $\sqrt{N/P}$ rows and columns. Let A_l^o denote the number of outer codewords (TPC/SPC) of output Hamming weight l, and $A_{l,\,d_E}^i$ denote the number of inner codewords (precoded PR channel) of input Hamming weight l and output Euclidean weight d_E. Assuming a uniform interleaver, the average number of codewords of Euclidean weight d_E, $A_{d_E}^c$, over the ensemble of interleavers is

$$A_{d_E}^c = \sum_{l \geq 4,\, l\,\text{even}} \frac{A_l^o \times A_{l,\,d_E}^i}{\binom{N}{l}}. \tag{2}$$

The lower limit for the sum is $l = 4$ because the minimum distance of the TPC/SPC code is 4 and only even terms are considered since all codewords of the TPC/SPC are of even weight. The average word error rate is upper-bounded by the union bound

$$P_w \leq \sum_{d_E} A_{d_E}^c \cdot Q\left(\frac{d_E}{2\sigma}\right) \tag{3}$$

where σ^2 is the variance of the noise. To argue that TPC/SPC codes are capable of interleaving gain on precoded PR channels, we need to show that $A_{d_E}^c$ for small d_E decreases with an increase in interleaver size, which in turn provides a reduction in error rate. Since a precoded PR channel is in general nonlinear, the all-zeros codeword cannot be treated as the reference codeword. However, a full compound of error events pertaining to $A_{l,\,d_E}^i$ is prohibitively complex for an exact analysis. To simplify this, Öberg and Siegel have made the assumption that the input to the precoded channel is an independent and identically distributed (i.i.d.) sequence [25]. This assumption

[1] As a clarification of our notation, we use "TPC/SPC codes" to mean plain TPC/SPC codes (with respect to AWGN channels), and use "TPC/SPC systems" to mean the combination of TPC/SPC codes and PR channels (which forms a serial concatenated code). Similar terms hold for "LDPC codes" and "LDPC system."

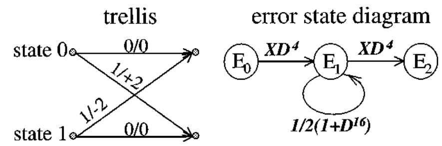

Fig. 5. Equivalent trellis for even/odd bits of precoded PR4 channels $((1 - D^2)/(1 \oplus D^2))$.

makes it easier to compute the transfer function of the precoded channel, since an i.i.d. sequence of zeros and ones can be treated as the reference sequence. In the following, we use this assumption to analyze the distance spectrum of the combination of the TPC/SPC outer code and the precoded channel.

Let us consider a precoded PR4 channel as an example. The equivalent trellis corresponding to odd/even bits of the precoded PR4 channel $((1 - D^2)/(1 \oplus D^2))$ is shown in Fig. 5. Following similar derivations as in [26], the average error enumerating function, where the average is taken over all possible input sequences, is given by

$$\begin{aligned} T(X,\, D) &= \frac{X^2 D^8}{1 - \frac{1}{2}(1 + D^{16})} \\ &= X^2 D^8 \left[1 + \frac{1}{2}(1 + D^{16}) + \frac{1}{2^2}(1 + D^{16})^2 \right. \\ &\qquad \left. + \cdots + \frac{1}{2^k}(1 + D^{16})^k + \cdots \right] \\ &= X^2 D^8 \left[\left(1 + \frac{1}{2} + \frac{1}{2^2} + \cdots \right) \right. \\ &\qquad \left. + D^{16} \left(\frac{1}{2} + \frac{2}{2^2} + \frac{3}{2^3} + \cdots \right) + O(D^{32}) \right] \\ &= X^2 D^8 [2 + 2D^{16} + O(D^{32})] \end{aligned} \tag{4}$$

where the exponent of X is the input Hamming weight of the error sequence, and the exponent of D is the output squared Euclidean distance of the error sequence. The fractional terms in the branch weight enumerator such as $1/2(1 + D^{16})$ (Fig. 5) are a direct consequence of the assumption that the input corresponding to that branch can be a 0 or 1 with equal probability 1/2 [26].

For the precoded PR4 channel, the independent (i.e., unconcatenated) input error sequence always has input weight 2. This can be seen from the transfer function since every term corresponds to X^2. Specifically, all input error sequences of the form $1 + D^{2j}$ result in an error event. The minimum Euclidean distance over all such error events occurs when $j = 1$ and the minimum Euclidean distance is 8 (assuming i.i.d inputs). Every finite weight codeword is the concatenation of k weight-2 input error events for some k. For large N, let $T_N(X^{2k},\, D)$ denote the truncated weight enumerator truncated to length N, where each error event is the result of k input error sequences each of weight-2. Then

$$T_N(X^{2k},\, D) \propto \binom{N}{k} X^{2k} D^{8k} [2 + 2D^{16} + O(D^{32})]^k \tag{5}$$

since there are approximately $\binom{N}{k}$ ways to arrange k error events in a block of length N. For the least nonzero l in the

TPC/SPC system, namely $l = 4$ [i.e., $k = 2$ in (5)], we see that $A^i_{l=4,\,d_E=4} \approx 4\binom{N}{2}$, and $A^o_{l=4} \approx P[\binom{\sqrt{N/P}}{2}]^2$ (there are $[\binom{\sqrt{N/P}}{2}]^2$ ways in which we can arrange a block of weight 4 within a TPC/SPC and there are P blocks in a codeword of length N). Substituting them into (2) and using the approximation $\binom{N}{n} \approx N^n/n!$ for large N, we have

$$A^c_{d_E=4} \propto \frac{\frac{N^2}{P} \cdot N^2}{N^4} \propto P^{-1}. \qquad (6)$$

It should be observed from (6) that the reduction in word error rate is proportional to the number of blocks P of the TPC/SPC that form a codeword, rather than N, as what would be expected from Benedetto *et al.*'s analysis [24]. This is especially important for finite block lengths, since this means that an interleaving gain is limited to the number of codewords of the outer code that are concatenated. Although we have only discussed the error event corresponding to the least nonzero l (i.e., $l = 4$), it can be shown that, for other values of small l, similar arguments hold. Similar results can be shown for EPR4 or other ISI channels also. To handle ISI channels with a larger number of states, it is convenient to consider the precoder separately from the channel. That is, we treat the concatenation of the TPC/SPC and the precoder as a code whose codewords are passed through the ISI channel. Since the interleaving gain is dependent only on the recursive nature of the inner code, an interleaving gain will result regardless of the type of ISI channel. This idea will be further addressed later for the optimization of TPC/SPC systems.

It is important to note that the fact that the least nonzero l is 4 (i.e., $d_{\min} = 4$ for the outer code) is crucial to the result in (6). It is shown by Benedetto *et al.* [24] that the outer code should have a $d_{\min}$ of at least 3 in order to obtain an interleaving gain in the word error rate. *The key advantage of TPC/SPC codes is that for any rate and any codeword length, $d_{\min} = 4$, which enables an interleaving gain.* On the contrary, for punctured convolutional codes of high rate (0.9 or higher, such as what is of interest), the constraint length of the code must be very large to obtain a minimum distance of 3 or higher. For example, even a 16-state punctured convolutional code with generator polynomials $(31, 33)_8$ of rate 0.9 has a $d_{\min}$ of only 2. The obvious disadvantage is that the decoding complexity increases exponentially with the constraint length. Therefore, TPC/SPC codes are a computationally efficient choice for constructing a good class of high-rate outer codes which guarantee an interleaving gain.

IV. Threshold Analysis Using Density Evolution

A. Introduction to Density Evolution and Gaussian Approximation

Although distance spectrum analysis shows that TPC/SPC codes concatenated with precoded ISI channels possess good distance spectra, the analysis is useful only if a maximum likelihood decoder is used. Since an iterative decoder is used in practice, it would be more convincing if the analysis takes into consideration the suboptimal nature of iterative decoding. The recently developed technique of density evolution (DE) [15]–[19] permits analysis of iterative decoding. This section goes through the critical points in the application of DE to TPC/SPC and LDPC systems. For comparison purposes, we extend it to include serial turbo systems (with punctured convolutional codes) also.

B. Problem Formulation

The systems under investigation have a unified architecture in that the (precoded) PR channel is modeled as an inner rate-1 convolutional code, with the outer code being an LDPC code, a TPC/SPC code or a (punctured) convolutional code. A turbo equalizer is used to iterate messages between the inner and outer decoders. During the qth iteration, the outer decoder generates extrinsic information on the jth coded bit a_j, denoted by $L_o^{(q)}(a_j)$, and passes it to the inner decoder. The inner MAP decoder then uses this extrinsic information (treated as *a priori*) with the received signal and generates extrinsic information, $L_i^{(q+1)}(a_j)$. The extrinsic information $L_o^{(q)}(a_j)$ is a random variable and, for an infinite block size, the random variables $L_o^{(q)}(a_j)$ are i.i.d. $\forall j$.

The idea in DE is to examine the probability density function (pdf) of $L_o^{(q)}(a_j)$ during the qth iteration, denoted by $f_{L_o{}^{(q)}}(x)$. Let us assume that the overall code is linear and, hence, the all-zero sequence is transmitted. If the sign of $L_o^{(q)}(a_j)$ is positive $\forall j$, then the decoding algorithm has converged to the correct codeword. The probability that $L_o^{(q)}(a_j) < 0, \forall j$, is $\Pr(L_o^{(q)}(a_j) < 0) = \int_{-\infty}^{0} f_{L_o{}^{(q)}}(x)\,dx$. The key is to find the SNR value above which $\Pr(L_o^{(q)}(a_j) < 0) \to 0$. This SNR value is referred to as the threshold or capacity of the system. Under the assumption of an infinite block length, the pdf of $L_o^{(q)}(a_j)$ is the same for all j and, hence, we drop the dependence on j. The threshold is then given by

$$\mathbf{C} = \inf_{SNR} \left\{ SNR\colon \lim_{q\to\infty} \lim_{N\to\infty} \int_{-\infty}^{0} f_{L_o{}^{(q)}}(x|\overline{\mathbf{y}}_N, SNR)) \to 0 \right\} \qquad (7)$$

where $\overline{\mathbf{y}}_N$ denotes the observed sequence of length N, superscript (q) denotes the qth iteration, and subscripts i and o denote quantities pertaining to the inner and outer code, respectively.

Since it is quite difficult to analytically evaluate $f_{L_o^{(q)}}(x)$ for all q, simplification can be made by approximating $f_{L_o^{(q)}}(x)$ to be Gaussian. This is the same approximation that Wiberg *et al.* [27], Chung *et al.* [17], and El Gamal *et al.* [28] have used to analyze concatenated codes. Further, Richardson and Urbanke [16] have shown that, for binary input–output symmetric channels, a consistency condition is preserved under DE for all messages, such that the pdfs satisfy the condition $f_{L_o{}^{(q)}}(x) = f_{L_o{}^{(q)}}(-x) \cdot e^x$. Imposing this constraint to the approximate Gaussian densities at every step leads to $(\sigma_o^{(q)})^2 = 2m_o^{(q)}$, i.e., the variance of the message density equals twice the mean. Under i.i.d. and Gaussian assumptions, the mean of the messages $m_o^{(q)}$ then serves as the sufficient statistic of the message density. The problem thus reduces to

$$\mathbf{C} = \inf_{SNR} \left\{ SNR\colon \lim_{q\to\infty} \lim_{N\to\infty} m_o^{(q)} \to \infty \right\}. \qquad (8)$$

C. Message Flow Within the Channel MAP Decoder

To evaluate the concatenated systems using DE, we need to examine the message flow within the outer decoder, the inner decoder, as well as in between the two. Specifically, we need to evaluate $m_o^{(q)}$ as a function of $m_i^{(q)}$ and *vice versa*. For the inner MAP decoder (equalizer), since it is not straightforward to derive $m_i^{(q+1)}$ as a function of $m_o^{(q)}$, Monte Carlo simulations are used to simulate the behavior of the MAP decoder and determine a relationship between $m_i^{(q+1)}$ and $m_o^{(q)}$, denoted by

$$m_i^{(q+1)} = \gamma_i(m_o^{(q)}). \tag{9}$$

The mean of the message $m_i^{(q+1)}$ is evaluated at the output of the inner MAP decoder given the input *a priori* information is i.i.d. and Gaussian with mean $m_o^{(q)}$ and variance $2m_o^{(q)}$. Since ISI channels are generally nonlinear, the input sequence is not assumed to be all zeros, rather a sequence of i.i.d. bits. Detailed description and figures of Monte Carlo simulation technique for computing γ_i can be found in [20].

D. Message Flow Within the Outer Code

This section describes how to compute $m_o^{(q)}$ as a function of $m_i^{(q)}$ for different outer codes.

1) LDPC Codes: The LDPC decoder itself is an iterative decoder which uses L iterations to update extrinsic information passed between bits and checks. Since turbo equalization is also an iterative process, we use superscript (q) and (l) to denote quantities during the qth iteration of turbo equalization (outer loop) and lth iteration within the LDPC decoder (local loop). Let $\mathcal{E}(b_k)$ and $\mathcal{E}(c_j)$ denote the set of all checks connected to bit b_k, and the set of all bits connected to check c_j, respectively, in the LDPC code. Assuming regular LDPC codes with $|\mathcal{E}(b_k)| = t$, $\forall k$, and $|\mathcal{E}(c_j)| = s, \forall j$, we have a code rate $R = 1 - t/s$. Message flow on the code graph is a two-way procedure, namely, bit updates and check updates, which correspond to the summation in the real domain and the so-called check-sum operation or $\tanh$ rule [17], [8], [18]. After L local iterations of message exchange, the message passed over to the inner MAP decoder is the LLR of the bit in the Lth iteration after subtracting $L_i^{(q)}$ which was obtained from the inner code and was used as *a priori* information.

Under the Gaussian assumption, we are interested in tracking the means of $L_b^{(q,\,l)}$ and $L_c^{(q,\,l)}$ given by $m_b^{(q,\,l)}$ and $m_c^{(q,\,l)}$, respectively. Treating extrinsic information as independent, the means of the extrinsic information at each iteration can be shown to be [17]

bit-to-check:

$$m_b^{(q,\,l)} = m_o^q + (s-1)\cdot m_c^{(q,\,l-1)} \tag{10}$$

check-to-bit:

$$m_c^{(q,\,l)} = \psi^{-1}\left(\left[\psi(m_b^{(q,\,l-1)})\right]^{t-1}\right) \tag{11}$$

LDPC-to-MAP:

$$m_o^{(q)} = s\cdot m_c^{(q,\,L)} \tag{12}$$

where $\psi(x)$ is the expected value of $\tanh(u/2)$, and u follows a Gaussian distribution with mean x and variance $2x$. $\psi(x)$ is given by

$$\psi(x) = \begin{cases} \frac{1}{\sqrt{4\pi x}}\int_{-\infty}^{\infty}\tanh\left(\frac{u}{2}\right)e^{-(u-x)^2/4x}\,du, & x>0 \\ 0, & x=0. \end{cases} \tag{13}$$

$\psi(x)$ is continuous and monotonically increasing on $[0,\infty)$ with $\psi(0)=0$ and $\psi(\infty)=1$. The initial condition is $m_b^{(q,\,0)} = m_c^{(q,\,0)} = 0$. When x is large (corresponding to low error probability), $(1-\psi(x))$ is shown to be proportional to the error probability [17]. The above derivation is essentially an extension of Chung *et al.*'s work [17] for the case of turbo equalization. For more detailed and thorough understanding, readers are directed to [16]–[19] and the references therein. For the turbo equalization case, after q (big) iterations between the outer and inner decoder (where each big iteration includes L local iterations within the LDPC decoder), the capacity is evaluated as

$$\mathbf{C}_{\text{LDPC}} = \inf_{\text{SNR}}\left\{\text{SNR: } \lim_{q\to\infty} s\cdot m_c^{(q,\,L)} \to \infty\right\}. \tag{14}$$

It is instructive to note that L is to be chosen carefully, since it affects the capacity of the resulting code.[2] Of particular practical interest is to find the best tradeoff between the resulting threshold and complexity, as will be addressed in a later section.

2) TPC/SPC Codes: Although a TPC/SPC code can be viewed as a special type of LDPC code, the DE procedure cannot be applied directly. This is because DE assumes that there are no cycles in the code graph. For TPC/SPC codes, even as the length of the code becomes very large, there are always cycles of length $4(k+1)$, where k is any integer. This is due to the fact that a rectangular error pattern, as shown in Fig. 4, always results in a loop. Such a loop of length 8 is shown in Fig. 4 (in thick lines). Consequently, the assumption that messages being passed within the code are independent (loop-free operation) is no longer valid.

For this reason, we propose and discuss a slightly modified procedure. If the number of local iterations within the TPC/SPC code is restricted to be small, then, the DE method would have operated on cycle-free subgraphs of TPC/SPC codes. Put another way, the messages exchanged within TPC/SPC codes along each step are statistically independent as long as the cycles have not "closed." Here, we restrict the number of local iterations within TPC/SPC codes to be one row update and one column update. Any more updates in either direction will either pass information to its source or pass duplicate information to the same node, which is unacceptable. On the other side, due to the (perfect) random interleaver, an infinite number of turbo iterations can be performed between the inner and outer decoders if the messages within the outer TPC/SPC code are reset to zero in every new turbo iteration. In order to improve the convergence of the decoding algorithm, we consider a serial update—that is, the row update and the column update

[2]The decoding strategy is considered part of the "code," since different decoding parameters lead to varying performance.

TABLE II
SUMMARY OF DE PROCEDURE FOR TPC/SPC SYSTEMS (UPPER BOUND)

Initialization:

$m_o^{(0)} = 0;$

Density Evolution:

for $q = 1, 2, \cdots$

compute $m_i^{(q)} = \gamma_i(\bar{\mathbf{r}}, m_o^{(q-1)});$

$m_{c_2}^{(q)} = 0;$

row-wise: bit to check: $m_{b_1}^{(q)} = m_i^{(q)};$

check to bit: $m_{c_1}^{(q)} = \psi^{-1}(\,[\psi(m_{b_1}^{(q)})]^{K_1});$

col-wise: bit to check: $m_{b_2}^{(q)} = m_i^{(q)} + m_{c_1}^{(q)};$

check to bit: $m_{c_2}^{(q)} = \psi^{-1}(\,[\psi(m_{b_2}^{(q)})]^{K_2});$

$m_o^{(q)} = m_{c_1}^{(q)} + m_{c_2}^{(q)};$

end;

Target:

$$\mathbf{C}_{TPC/SPC} = \inf_{SNR}\,\{SNR : \lim_{q\to\infty} m_o^{(q)} \to \infty\}$$
$$= \inf_{SNR}\,\{SNR : \lim_{q\to\infty} m_{c_1}^{(q)} + m_{c_2}^{(q)} \to \infty\}$$

are not performed simultaneously. Rather, the row update is performed first and the extrinsic information from the row checks is passed to bits and later used in the column updates. The resulting procedure to compute the densities can then be summarized as in Table II.

For an exact threshold, the density evolution procedure should, in addition to avoiding looping messages, also ensure completeness in the sense that every bit should have utilized all the messages (through dependencies) from all the checks. The procedure discussed in the previous paragraph and tabulated in Table II, although stemming naturally from the decoding procedure, is unfortunately not complete. This is because only one row update followed by one column update is performed, which is not sufficient to exploit the information from all the checks. For example, let us consider the extrinsic messages on bit 1 in Fig. 4. After the row update, the decoding procedure has utilized the dependency on check a, and then after the column update, the dependency of check d. The vertical update also utilizes the dependencies on check b and c from the previous row update. However, check e and f, also bear information about bit 1 indirectly since e and f provide information about bits 2 and 3, which can improve the estimate of 1. However, checks e and f are not fully exploited in updating the information on bit 1. Hence, the resulting threshold is an upper bound.[3]

3) Serial Turbo System: In the serial turbo system, the outer code is a punctured convolutional code with a moderate constraint length. Treating it much the same way as we treat the inner convolutional code (the PR channel), a MAP decoder implementing the BCJR algorithm is used and the same Monte Carlo method is adopted to track the mean of the extrinsic information (of the outer code), $m_o^{(q)}$, during the qth iteration. The capacity is computed using

$$\mathbf{C}_{\text{serial}} = \inf_{SNR}\left\{SNR\colon \lim_{q\to\infty} m_o^{(q)} \to \infty\right\}. \qquad (15)$$

[3]By upper bound, we mean that the exact thresholds of TPC/SPC system should be better than this. In other words, for a given decibel, the achievable code rate (bandwidth efficiency) could be higher or, equivalently, for a given rate, the required SNR could be smaller.

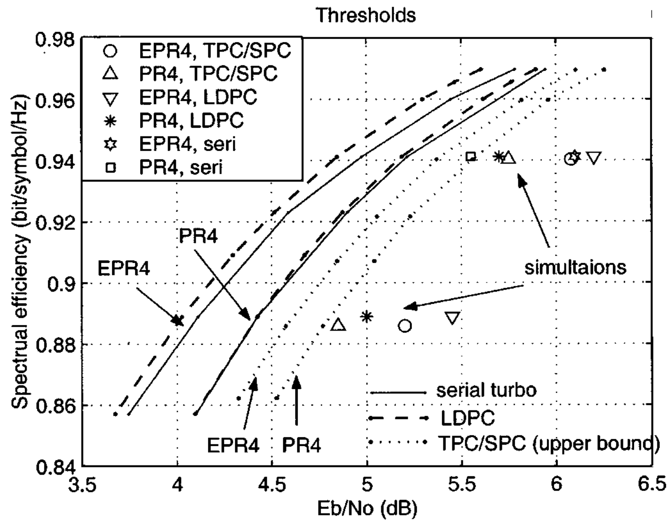

Fig. 6. Thresholds of TPC/SPC, LDPC, and serial turbo systems over ideal PR channels [the outer code in serial turbo system is a systematic recursive convolutional code $(31,\ 33)_{oct}$].

E. Thresholds

The upper bound on the threshold for TPC/SPC codes and the thresholds for LDPC codes and punctured convolutional codes are shown in Fig. 6 for PR4 and EPR4 channels. We consider regular LDPC codes with column weight 3, since regular LDPC codes have a slight advantage over irregular LDPC codes for short block sizes and high rates as in data storage applications [29]. It can be seen that the *upper bound* for TPC/SPC codes is about 0.5 dB away from that of LDPC codes for a code rate of 0.94. This shows that the performance of TPC/SPC is expected to be within a few tenths of a decibel from that of LDPC codes. Further, the thresholds for LDPC codes are comparable to those of a serial concatenated code with a 16-state convolutional code. Since the decoding complexity of LDPC codes and TPC/SPC codes is significantly lower than that of 16-state convolutional codes for high rates, there seems to be little advantage in using punctured convolutional codes.

Also presented are the corresponding simulation results that are evaluated at a BER of 10^{-5}, with block size of 4K user data bits. It can be seen that for practical block sizes the performance of TPC/SPC codes is actually slightly better than that of LDPC codes for EPR4 channels and is comparable to that of LDPC codes for PR4 channels. Due to the finite block size, the simulations are around 0.5–1 dB away from the bounds. Nevertheless, this presents a reasonable match and indicates that DE is a useful tool in the threshold analysis of LDPC codes, TPC/SPC codes, and serial concatenated codes for PR channels.

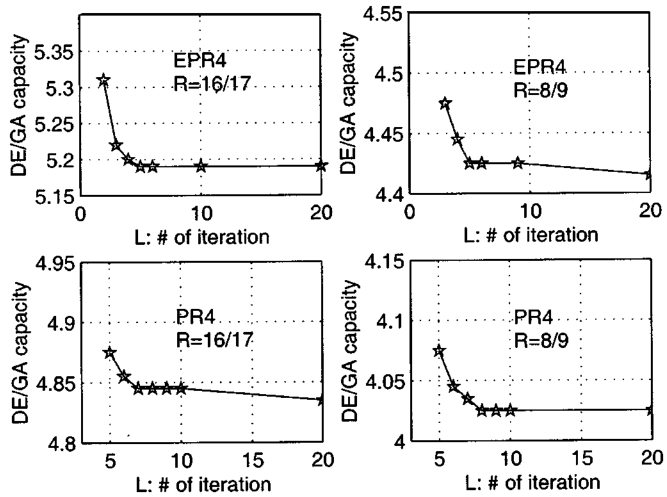

Fig. 7. Thresholds versus L in LDPC systems.

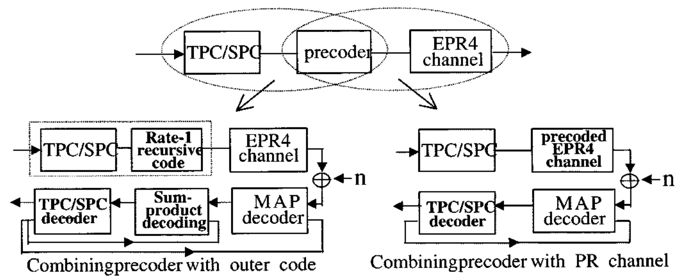

Fig. 8. Different views of the serial concatenated system.

V. Optimization of the Decoding Process

A. LDPC Systems

Each turbo iteration (outer loop) involves a pass of forward–backward decoding of the inner MAP decoder (BCJR algorithm) followed by L rounds of bit-check/check-bit updates (small loop) of the LDPC decoder. As mentioned above, with the assumption of an infinite block size and a perfect random interleaver, the girth (shortest cycles) of an LDPC code is unbounded, and thus L can be infinitely large. Perceivably, the resulting thresholds are nondecreasing with L, but overly large L is computationally inefficient. Hence, it would be of practical interest to investigate how the value of L affects performance and, in particular, to find an optimal balancing point where best performance is achieved at the least decoding complexity. This can be done by calculating the thresholds of LDPC systems using DE with different values for L. We examined a rate-16/17 and a rate-8/9 regular LDPC code (column weight 3) over PR4 and EPR4 channels, respectively. As shown in Fig. 7, increasing L beyond a point brings only marginal improvement in the thresholds. Further, it is interesting to observe that the optimal value of L is slightly different with different channel coefficients. Whereas L =4 or 5 seems to be a good tradeoff on EPR4 channels, L =7–8 seems better for PR4 channels. Extensive simulation experiments show that somewhere around 5–8 seems to be a good choice for L, corroborating this result. It is also worth mentioning that the above results are for the LDPC code ensemble where the column weights are uniformly 3 and the row weights follow the concentration rule (as uniform as possible). The optimal value of L might differ slightly for different designs of LDPC codes, but the difference should be small. Further, for a fixed complexity, the value of L may be lower than the ones reported (which is for unconstrained complexity).

B. TPC/SPC Systems

Most work on turbo equalization of PR channels treats the combination of the precoder and the ISI channel as the inner code [1]–[3], [8]. Therefore, each iteration in the turbo equalization process involves decoding of the outer code followed by a BCJR decoder for the precoded channel. Since most of the complexity comes from the inner MAP decoder (Table I), it is desirable to reduce the number of iterations q involving the MAP decoder and, hence, to devise a decoding strategy which minimizes q with a fairly small sacrifice in performance. This is of particular interest to high-density recording systems where the appropriate PR targets correspond to 8-state (like EPR4) or even 16-state trellis [like E^2PR4 with $H(D) = 1+2D-2D^3-D^4$].

Although it is important to exploit the memory in the ISI channel and the recursiveness introduced by the precoder, the interleaving gain is dependent only on the recursiveness of the inner code. With this observation, we propose an efficient and effective modified receiver structure where the combination of precoder and the TPC/SPC code is considered as an outer code and the nonprecoded PR channel is the inner code. As such, MAP equalization need not be performed at every iteration stage. Rather, it can be done after every s iterations between the TPC/SPC code and the precoder, as illustrated in Fig. 8. The key advantage here is that the precoder is often of the form $1/(1 \oplus D)$ or $1/(1 \oplus D^2)$, which can be represented by a 2-state trellis rather than an 8- or 16-state trellis for an EPR4 or E^2PR4 channel and therefore saving considerable complexity without sacrifice in performance. The complexity can be further reduced by using the sum-product algorithm on the graph of the precoder [30]. When the precoder is of the form $1/(1 \oplus D^m)$ (m an integer), its corresponding code graph alone has no cycles and therefore sum-product decoding is optimal. In particular, using the $\tanh$ implementation of the sum-product algorithm results in approximately 1/5 the complexity of a conventional 2-state BCJR algorithm for the precoder $1/(1 \oplus D^m)$ (altogether 5 additions and 5 lookups per encoded bit) [30].

Given this setup, we now address two important questions in a practical implementation:

i) given an overall allowable complexity, what are the optimum values of s (the number of local iterations in TPC/SPC decoder) and q (the number of turbo iterations between the channel and the outer code)?

ii) what is the tradeoff in performance versus overall complexity, given that we can optimize the performance by answering question i)?

We answer these questions using DEs with some modifications. For a given q and s, let $\Delta(q, s)$ denote the overall complexity, including additions, max operations, and lookups (see Table I). Since we are interested in finite complexity and, hence, a finite number of iterations, we first reformulate the thresholds

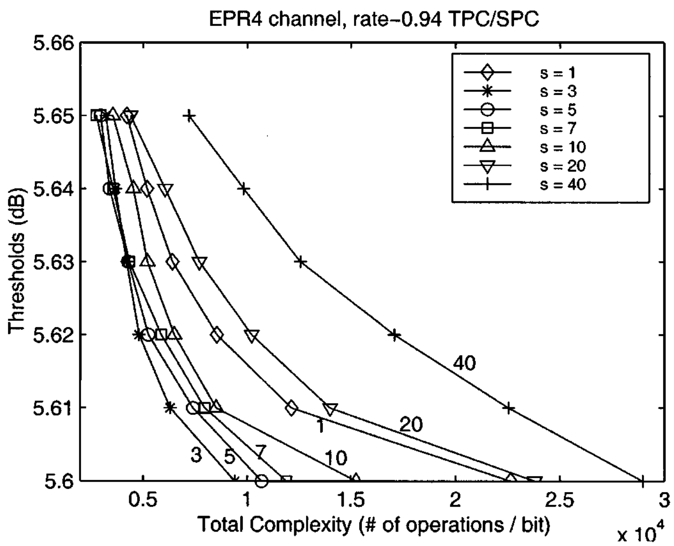

Fig. 9. Optimization of TPC/SPC systems.

as the SNR for which the mean reaches a threshold (ν_{thresh}) (a positive number serving as the practical infinite point). For a given q and s, the new threshold is thus given by

$$\mathbf{C}(q,\, s) = \inf_{SNR} \left\{ SNR \colon m_o^{(q)} \geq \nu_{\rm thresh} \right\}. \qquad (16)$$

When the value of $\nu_{\rm thresh}$ is set large enough, the difference from the actual threshold will be negligible. For a given overall complexity Δ_0, different values of q and s will produce different thresholds and we are interested in the best (least) value $\mathbf{C}^*(\Delta_0)$ given by

$$\mathbf{C}^*(\Delta_0) = \min_{q,\, s} \{ \mathbf{C}(q,\, s) \colon \Delta(q,\, s) \leq \Delta_0 \}. \qquad (17)$$

The cost function $\Delta(q,\, s)$ depends on the outer code, the actual channel and the precoder. For a TPC/SPC code on an EPR4 channel, we have (see Table I)

$$\Delta(q,\, s) = ((10+10)s + 25 \cdot 2^3 + 5)q = (20s + 205)q \quad (18)$$

where 205 is the number of operations per encoded bit for a BCJR decoding of the 8-state EPR4 channel, and 20 is the number of operations per encoded bit for one small iteration between the TPC/SPC code and the precoder.

Fig. 9 shows a plot of $\mathbf{C}^*(\Delta_0)$ versus Δ_0 for various values of s for a rate 0.94 TPC/SPC code over EPR4 channels, where $\nu_{\rm thresh} = 30$. Obvious from the figure is that, for a given Δ_0, the value of s has a significant impact on the resulting thresholds. Also seen from the figure is that setting s to be around 3 optimizes the thresholds and complexity consistently and, hence, is a good choice. This means that the equalization procedure (with respect to the channel MAP) is used only once every three iterations and, hence, results in complexity savings. It is interesting to note that the performance of $s = 40$ is quite poor. That is, for a fixed complexity, if s is increased beyond 5, due to the few stages of turbo equalization that are possible, the resulting thresholds are weak. Depending on the exact complexity Δ_0 that can be allowed, the procedure can be repeated over that range to optimize s and q.

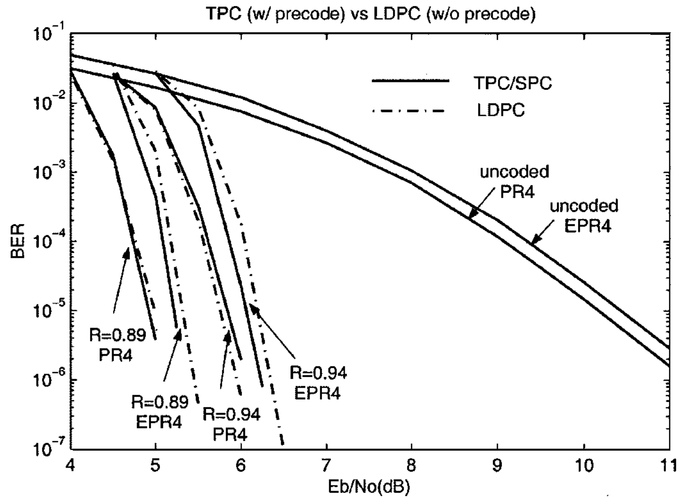

Fig. 10. Performance of TPC/SPC vs. LDPC over ideal PR channels.

VI. Simulation Results

To be applicable to present-day data storage systems, the 2-D TPC/SPC codes we investigated have rate 0.89 and rate 0.94 which are formed from (17, 16) and (33, 32) TPC codes, respectively. We combine sixteen $(17,\, 16)^2$ TPC/SPC codewords and four $(33,\, 32)^2$ TPC/SPC codewords, respectively, to form an effective data block size of 4Kbits. The channel models we test are PR4 and EPR4 magnetic recording channels. For comparison purposes, also presented are the results of a rate $8/9 = 0.89$ and $16/17 = 0.94$ regular LDPC codes with column weight 3 and data block size 4K. It should be noted here that irregular LDPC codes of such high rates have been seen to perform slightly worse than regular codes [29] and, hence, this represents the best case for LDPC codes. In all the simulations presented, there are two iterations inside the TPC/SPC decoders and four iterations inside the LDPC decoders. Although this leads to a decoding complexity of LDPC codes a bit higher than TPC/SPC codes, it is a good compromise of complexity and performance for both codes.

BER: Fig. 10 shows the performance of LDPC codes and TPC/SPC codes over PR4 and EPR4 channels. It can be seen that gains of 4.4–5 dB over uncoded PR maximum-likelihood (PRML) systems are obtained for TPC/SPC codes at a BER of 10^{-5}, comparable to those of LDPC codes. All TPC/SPC codes are precoded with $1/(1 \oplus D^2)$ which is the best for PR4/EPR4 channels, as shown analytically in [20] and as shown empirically in Fig. 11. LDPC codes are not precoded, for, as shown in [20], their performances are better without precoding. We have confirmed this through simulations also. Hence, the comparison is fair as it represents the best cases for both codes.

Error Statistics: Although both TPC/SPC and LDPC codes seem to offer significant coding gains when the average BER is of the order of 10^{-7}, it is still unclear whether LDPC codes and TPC/SPC codes may suffer from an error floor. Therefore, the conventional use of RS-ECC is still necessary to reduce the BER to 10^{-15} as is targeted for recording systems. The RS-ECC code works on the byte level, capable of correcting up to t byte errors in each data block of size 4K bits or 512 bytes (t is usually around 10–20). Hence, the maximum number of uncorrected errors left over in each block after TPC/SPC or LDPC decoding

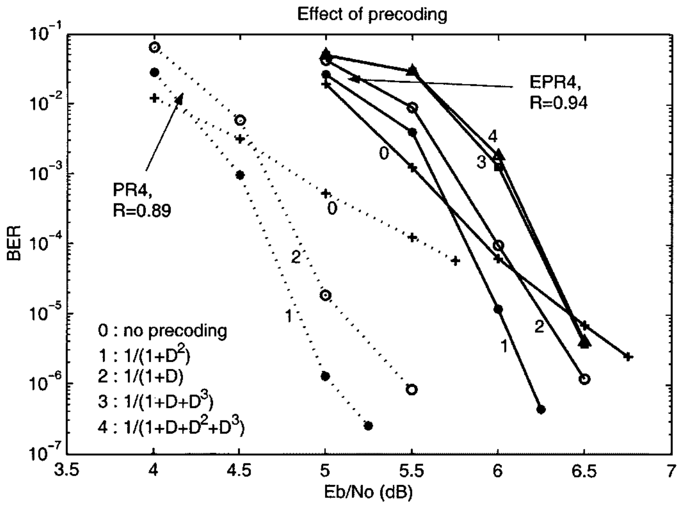

Fig. 11. Effect of precoding in TPC/SPC systems.

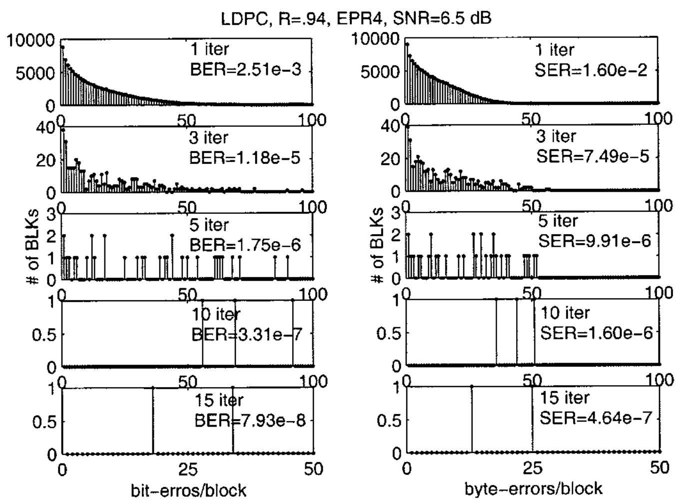

Fig. 12. Error statistics of LDPC codes over EPR4 channels (code rate = 0.94, SNR = 6.5 dB, collected over 100 000 blocks; x axis: maximum number of errors observed within a block; y axis: occurrence of such blocks).

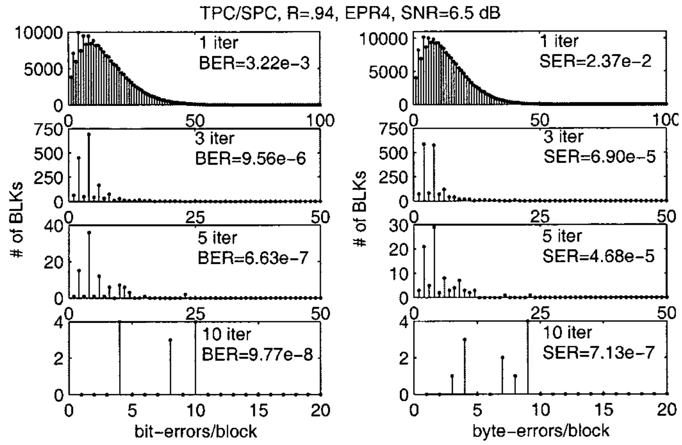

Fig. 13. Error statistics of TPC/SPC codes over EPR4 channels (code rate = 0.94, SNR = 6.5 dB, collected over 165 000 blocks).

has to be relatively small to guarantee the proper functioning of the RS-ECC code. In other words, block error statistics is crucial and closely relate to the overall system performance. Unfortunately, this has been largely neglected in most of the previous work.

Figs. 12 and 13 plot the histograms of the number of bit/byte errors for an effective block size 4K, rate 0.94 LDPC code and TPC/SPC code over EPR4 channels, respectively. The left column plots bit error histograms and the right plots byte error histograms. The statistics are collected over more than 100 000 blocks of data size 4K bits. At an SNR of 6.5 dB and after the 10th iteration (outer loop), the maximum number of symbol errors observed in a single block is less than 10 for TPC/SPC codes (which would be corrected by the RS-ECC code), but around 50 for LDPC codes. If further iterations are allowed, error bursts in LDPC codes are alleviated. Nevertheless a block containing 25 symbol errors is observed after 15 turbo iterations and this may still cause the RS-ECC code to fail. Unless a more powerful RS-ECC is employed, LDPC codes are prone to cause block failure, where all data in that block are presumed lost. It should be noted that although what we have observed suggests that TPC/SPC codes may be more compatible to magnetic recording systems than LDPC codes, the statistics are nonetheless insufficient. Ryan and Li present error statistics for the TPC/SPC codes compiled from simulating over 10^6 blocks in [32]. Due to the random interleaver as well as the suboptimal iterative decoding, hardware tests may still be needed before a convincing argument can be made.

VII. Conclusion

This paper investigates the potential of applying TPC/SPC codes to magnetic recording systems, with LDPC codes as a comparison study. The main results from this paper can be summarized as follows.

1) In the application of TPC/SPC codes to PR magnetic recording channels, considerable coding gains can be achieved by combining several blocks of TPC/SPC together (since interleaving gain is proportional to the number of TPC/SPC blocks combined in a codeword) and by choosing a proper precoder for the channel. In particular, gains of more than 4.4 dB over uncoded systems are observed on PR4 and EPR4 channels, revealing a performance comparable to that of LDPC codes.
2) While the decoding complexity is slightly smaller than that of LDPC codes, TPC/SPC codes are linear time encodable. Further, they do not require large storage for the parity check and generator matrices. The interleaving pattern should be stored—however, algebraic interleavers which can be generated "on the fly" can be used which demonstrate reasonably good "randomness" and which save precious storage in hardware implementation [30], [31].
3) In contrast to LDPC codes whose large error bursts are beyond the capacity of the outer RS-ECC codes, TPC/SPC codes demonstrate error statistics favorable to RS-ECC codes, which assures a consistent and quality performance of the whole system.
4) DE is an effective tool in the analysis of iterative decoding processes by taking into consideration both the code structure and the iterative feature of the decoding algorithm. Through its use in the calculation of thresholds

for TPC/SPC, LDPC and serial turbo systems, we demonstrate a framework under which this useful method can be exploited for a variety of concatenated systems where iterative approaches are used.

To summarize, our work has indicated TPC/SPC codes as a promising candidate in the application of future magnetic recording systems. However, further experiments need to be conducted over more realistic channel models, like Lorentzian channels and, hopefully, on real data collected in the lab. Other interesting problems include how to achieve a good compromise among iterations, performance, complexity and delay in a practical setting, as well as how to incorporate the run-length limit constraint without affecting much complexity and performance.

REFERENCES

[1] T. M. Duman and E. Kurtas, "Performance of turbo codes over EPR4 equalized high density magnetic recording channels," in *Proc. Global Telecom. Conf.*, vol. 1b, 1999, pp. 744–748.

[2] T. Souvignier, A. Friedmann, M. Oberg, P. Siegel, R. Swanson, and J. Wolf, "Turbo decoding for PR4: parallel vs. serial concatenation," in *Proc. Int. Conf. Comm.*, June 1999, pp. 1638–1642.

[3] W. Ryan, L. McPheters, and S. McLaughlin, "Combined turbo coding and turbo equalization for PR4-equalized Lorentzian channels," in *Proc. Conf. Int. Sci. and Sys.*, Princeton, NJ, 1998, pp. 489–493.

[4] M. Oberg and P. H. Siegel, "Performance analysis of turbo-equalized dicode partial-response channel," in *Proc. 36th Annual Allerton Confer. Commun., Control and Computing*, Sept. 1998, pp. 230–239.

[5] R. G. Gallager, "Low-density parity check codes," *IRE Trans. Inform. Theory*, pp. 21–28, 1962.

[6] J. L. Fan, A. Friedmann, E. Kurtas, and S. McLaughlin. Low density parity check codes for magnetic recording. presented at 37th Allerton Conf.. [Online]http://www.geocities.com/~jfan.stanford/

[7] Y. Kou, S. Lin, and M. Fossorier, "LDPC codes based on finite geometries, a rediscorty and more," IEEE Trans. Inform. Theory, vol. 27, pp. 2711–2736, Nov. 1999, submitted for publication.

[8] J. Li, E. Kurtas, K. R. Narayanan, and C. N. Georghiades, "On the performance of turbo product codes and LDPC codes over partial response channels," in *Proc. Int. Conf. Commun.*, Helsinki, Finland, June 2001, pp. 2176–2183.

[9] H. Song, R. M. Todd, and J. R. Cruz, "Performance of low-density parity-check codes on magnetic recording channels," in *Proc. 2nd Int. Symp. on Turbo Codes and Related Topics*, Brest, France, Sept. 2000, pp. 395–398.

[10] P. Elias, "Error-free coding," *IRE Trans. Inform. Theory*, vol. IT-4, pp. 29–37, Sept. 1954.

[11] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inform. Theory*, vol. 42, Mar. 1996.

[12] D. Chase, "A class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory*, vol. 18, pp. 170–182, Jan 1972.

[13] R. M. Pyndiah, "Near-optimum decoding of product codes: Block turbo codes," *IEEE Trans. Commun.*, vol. 46, pp. 1003–1010, Aug 1998.

[14] T. J. Richardson and R. L. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 638–656, Feb. 2001.

[15] T. J. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans Inform.*, vol. 47, pp. 619–637, Feb. 2001.

[16] T. J. Richardson and R. Urbanke, "The capacity of low-density parity check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 599–618, Feb. 2001.

[17] S.-Y. Chung, R. Urbanke, and T. J. Richardson, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 657–670, Feb. 2001.

[18] J. Li, E. Kurtas, K. R. Narayanan, and C. N. Georghiades, "Thresholds for iterative equalization of partial response channels using density evolution," in *Proc. Int. Symp. on Inform. Theory*, Washington, DC, June 2001, p. 73.

[19] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman. Analysis of low density codes and improved designs using irregular graphs. [Online]. Available: http://www.icsi.berkeley.edu/~luby/.

[20] K. R. Narayanan, "Effect of precoding on the convergence of turbo equalization for partial response channels," *IEEE J. Select. Areas. Commun.*, vol. 19, pp. 686–698, Apr. 2001.

[21] K. Immink, "Coding techniques for the noisy magnetic recording channel: A state-of-the-art report," *IEEE Trans. Commun.*, pp. 413–419, May 1989.

[22] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. Int. Conf. Commun.*, vol. 2, June 1995, pp. 1009–1013.

[23] P.-P. Sauvé, A. Hunt, S. Crozier, and P. Guinand, "Hyper-codes: High-performance, low-complexity codes," in *Proc. 2nd Int. Symp. on Turbo Codes and Related Topics*, Brest, France, Sept. 2000, pp. 121–124.

[24] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: Design and performance analysis," *IEEE Trans. Inform. Theory*, vol. 44, pp. 909–926, May 1998.

[25] M. Öberg and P. H. Siegel, "Parity check codes for partial response channels," in *Proc. Global Telecom. Conf.*, vol. 1b, Dec. 1999, pp. 717–722.

[26] L. L. McPheters, S. W. McLaughlin, and K. R. Narayanan, "Precoded PRML, serial concatenation, and iterative (turbo) decoding for digital magnetic recording," *IEEE Trans. Magnetics*, vol. 35, pp. 2325–2327, Sept. 1999.

[27] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping University, 1996.

[28] H. El Gamal, A. R. Hammons, Jr., and E. Geraniotis, "Analyzing the turbo decoder using the Gaussian approximation," in *Proc. Int. Symp. Inform. Theory*, 2000, p. 319.

[29] D. J. C. MacKay and M. C. Davey. Evaluation of Gallager codes for short block length and high rate applications. [Online]. Available: http://www.keck.ucsf.edu/~mackay/seagate.ps.gz.

[30] J. Li, K. R. Narayanan, and C. N. Georghiades, "Product Faccumulate codes: A class of capacity-approaching, low complexity codes," IEEE Trans. Inform. Theory, submitted for publication.

[31] G. C. Clark, Jr. and J. B. Cain, *Error-Correction Coding for Digital Communications*. New York: Plenum Press, 1981.

[32] W. Ryan and H. Li. Concatenated codes and iterative decoders for magnetic recording. [Online]http://www.ece.arizona.edu/~dcsl/conmagrec.html

Jing Li received the B.S. degree in computer science (with honors) from Beijing University, China, in 1997 and the M.E. degree in electrical engineering from Texas A&M University, College Station, in 1999. She is currently working toward the Ph.D. degree in electrical engineering at Texas A&M University.

Since 1998, she has been a Research/Teaching Assistant with the Department of Electrical Engineering, Texas A&M University. She was a research intern with Seagate Research Laboratory, Pittsburgh, PA, and with Tyco/Telecommunications Laboratory, Eatontown, NJ, in the summers of 2000 and 2001, respectively. Her current research focus in on advanced coding and modulation for wireless and optical fiber communications as well as data storage systems.

Ms. Li is a member of Pinnacle National Honor Society for graduate and non-traditional students and the recipient of the Ethel Ashworth-Tsutsui Memorial Award for Research in 2001.

Krishna R. Narayanan (S'92–M'98) received the Ph.D. degree in electrical engineering from the Georgia Institute of Technology, Atlanta, in 1998.

Since then, he has been an Assistant Professor in the Electrical Engineering Department at Texas A&M University, College Station. His research interests are in coding modulation and receiver design for wireless communications and digital magnetic recording.

Prof. Narayanan is the recipient of the NSF CAREER Award in 2001. He currently serves on the editorial board of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS.

Erozan Kurtas (M'98) received the B.Sc. degree from Bilkent University, Ankara, Turkey, in 1991 and the M.Sc. and Ph.D. degrees from Northeastern University, Boston, MA, in1993 and 1997, respectively.

His research interests cover the general field of digital communication and information theory with special emphasis on coding and detection for intersymbol interference channels. He has published over 30 journal and conference papers on the general fields of information theory, digital communications, and data storage. He is currently head of the Channel Department at Seagate Technology, Seagate Research, Pittsburgh, PA.

Costas N. Georghiades (S'82–M'82–SM'90–F'98) was born in Cyprus in 1955. He received the B.E. degree with distinction form the American University of Beirut in 1980 and the M.S. and D.Sc. degrees from Washington University in 1983 and 1985, respectively, all in electrical engineering.

Since September 1985, he has been with the Electrical Engineering Department, Texas A&M University, College Station, where he is a Professor and holder of the J. W. Runyon, Jr. Endowed Professorship and Director of the Telecommunications and Signal Processing Group. His general interests are in the application of information, communication and estimation theories to the study of communication systems.

Dr. Georghiades is a member of Sigma Xi and Eta Kappa Nu and a registered Professional Engineer in Texas. He served as an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, as Publications Editor and Associate Editor for Communications for the IEEE TRANSACTIONS ON INFORMATION THEORY, as Guest-Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS and as an Associate Editor for the IEEE COMMUNICATIONS LETTERS. He has been involved in organizing a number of conferences, including as Technical Program Chair for the 1999 IEEE Vehicular Technology Conference and the 2001 Communication Theory Workshop. He currently serves as program chair for the Communication Theory Symposium within Globecom 2001.