On the Performance of Turbo Product Codes and LDPC Codes over Partial-Response Channels

Jing Li*, Erozan Kurtas†, Krishna R. Narayanan, and Costas N. Georghiades
Electrical Engineering Department,
Texas A&M University, College Station, TX 77843-3128
{jingli, krishna, georghia}@ee.tamu.edu, erozan.m.kurtas@seagate.com

Abstract – **We investigate the performance of low density parity check (LDPC) codes, single-parity turbo product codes (TPC/SPC) and multi-parity turbo product codes (TPC/MPC) over various partial-response (PR) channels encountered in magnetic and magneto-optical (MO) recording systems, like PR4/EPR4 and PR1/PR2 channels. The codes have similarity in structures and can be decoded using simple message-passing algorithms. We show that the combination of a TPC/SPC code and a precoded PR channel results in good distance spectrum due to interleaving gain. Density Evolution is then used to compute the thresholds for TPC/SPC and LDPC codes over PR channels. Through analysis and through simulations, we show the three types of codes yield comparable bit error rate performance with similar complexity, but they exhibit quite different error statistics, which in turn may result in sharp differences in block failure rate after the Reed-Solomon error correction code (RS-ECC).**

I. INTRODUCTION

Recent work on the performance of turbo codes based on recursive systematic convolutional codes over partial-response (PR) channels, such as those found in high density magnetic recording systems [1]-[3], show that a 4-5 dB coding gain is achievable if the comparison is done prior to the outer Reed-Solomon error correction code (RS-ECC). Due to the huge decoding complexity of turbo codes, current research efforts focus on iterative decoding of block codes [4]-[5], and specifically low density parity check (LDPC) codes over PR channels used in magnetic and magneto-optic (MO) recording systems [6, 7].

This paper focuses on the performance analysis of turbo product codes (TPC) [5] and LDPC codes over various PR channels, such as PR4/EPR4 targets for magnetic recording systems and PR1/PR2 targets for MO recording systems. We are primarily interested in the simplest type of TPC codes which are formed from parity check codes, namely, single-parity turbo product codes (TPC/SPC) and multi-parity turbo product codes (TPC/MPC), which are similar to LDPC codes in code construction and decoding algorithm. Motivated by the inherent characteristic of TPC/SPC and TPC/MPC having high code rates as well as simple and effective soft-in soft-out (SISO) decoding algorithms, we conduct a comprehensive evaluation of their potential in data storage applications.

The rest of the paper is organized as follows. Section 2 gives an overview of TPC/SPC, TPC/MPC and LDPC codes in a comparative manner. Section 3 describes the system model and addresses related issues on serially concatenated schemes, including distance spectrum and precoding. Section 4 calculates the thresholds of both TPC/SPC and LDPC systems using density evolution with Gaussian approximation. Section 5 focuses on a comprehensive performance evaluation, including bit error rate (BER), complexity, effect of precoding in terms of analytical and numerical results and, in particular, bit/byte error statistics which are crucial in data storage systems to exploit the capacity of the outer RS-ECC code. Section 6 concludes the paper.

II. TURBO PRODUCT AND LDPC CODES

This section outlines the necessary background on TPC and LDPC codes used in the rest of the work. Emphasis is given to the comparison of their code structures and decoding strategies.

A low density parity check code is characterized by its randomly constructed parity-check matrix H in nonsystematic form. The practical decoding algorithm, known as the message-passing algorithm, is an instance of Pearl's belief propagation algorithm applied to the bipartite code graph, which converges to the maximum likelihood (ML) estimate if the code graph has no cycles. In a real construction however, although small cycles can be avoided, it is impossible to enforce an overall loop-free condition. Nevertheless, simulation results have shown suboptimal decoding works very effectively.

*Corresponding author
†E. Kurtas is with Seagate Technology, Seagate Research, 2403 Sidney St., Pittsburgh, PA 15203

A typical regular LDPC parity check matrix H satisfies: (1) Uniform column weight of $s = 3$ or above; (2) No two columns can have weight overlapping of more than one. Condition (1) becomes a standard setting because it is shown that the minimum distance of the code (averaged over the code ensemble) increases linearly with the block size provided $s \geq 3$ [4], which guarantees excellent asymptotic performance. Condition (2) is enforced to eliminate length 4 loops in its bipartite graph and, hence, to reduce unwanted correlation among messages which will impair the performance of the message-passing algorithm. Irregular LDPC codes, which do not constrain the columns to have uniform weights, have been reported to out-perform regular ones at moderate code rates. However, at very high rates and for relatively short block sizes, such as those used in the data storage systems, regular LDPC codes are shown to be slightly better [7]. Hence we focus on regular LDPC codes with column weight 3 in this work.

Turbo product codes [5, 10] are composed of a multi-dimensional array of codewords from linear block codes, such as parity check codes, Hamming codes and BCH codes. The simplest type of TPC codes, single-parity turbo product codes (TPC/SPC) and multi-parity turbo product codes (TPC/MPC), can be decoded using an SISO message-passing algorithm similar to that of LDPC codes. Further, unlike trellis-based codes which usually need puncturing to achieve high rates, TPC/SPC and TPC/MPC codes are intrinsically of very high rate, suitable for future high-density recording systems.

Let $\mathcal{C}_1 \sim (n_1, k_1, d_1)$ and $\mathcal{C}_2 \sim (n_2, k_2, d_2)$ denote two linear binary block codes, where $n_i, k_i, d_i, i = 1, 2$ are the codeword length, user data block length and minimum distance, respectively. A 2-dimensional (2-D) turbo product code $\mathcal{C} = \mathcal{C}_1 \otimes \mathcal{C}_2$ has parameters (n_1n_2, k_1k_2, d_1d_2), and its generator matrix is the Kronecker product of the generator matrices of its component codes: $G = G_1 \otimes G_2$. In the case of TPC/SPC, each row and each column satisfies a single-parity check, and the minimum distance for an m-dimensional TPC/SPC is 2^m. TPC/MPC codes are not much different from TPC/SPC codes except that there are more than one parity bits row-wise or column-wise, which lends more flexibility in code structure, code rate and code length. Since this work targets at their application in data storage systems, unless otherwise stated, all TPC/SPC and TPC/MPC codes mentioned are 2-dimensional for the sake of higher rates. This is important, since the code rate loss in data storage systems is $10\log_{10}(R^2)$ rather than $10\log_{10}(R)$ as in AWGN channels (R being the code rate) [9]. Further, both component codes in a TPC code are chosen the same to save hardware cost in a real implementation.

A turbo product code is generally viewed as a serial concatenation of its component codewords. However, from the perspective of LDPC codes, TPC/SPC and TPC/MPC codes can be treated as a special type of LDPC codes and represented in a similar bipartite graph as LDPC codes. In a 2-D TPC/SPC code, each bit has degree 2, smaller than the typical bit degree, 3, of an LDPC code, and its checks have degree n_0, also (much) smaller than that of an LDPC code, $3/(1-R)$. In other words, a bit participates in fewer checks in TPC/SPC than in LDPC codes, and a check involves less bits in TPC/SPC than in LDPC codes. The structure of TPC/MPC codes is somewhere in between.The minimum distance of a randomly constructed LDPC code is hard to determine, but it is usually quite large. Hence, a LDPC code possesses good error detection ability and almost never converges to a wrong codeword (undetected error). On the other hand, the minimum distance of a TPC/SPC or a TPC/MPC code is always 4 regardless of block size and rate. There are many such closely-neighbored codewords, so the possibility of undetected error is high. This difference in distance spectrum seems to have much impact on their error statistics, as we will see later.

Similarity in code graphs implies the possibility of a unified decoding algorithm to be applied for all three codes. Soft extrinsic information in LLR (log likelihood ratio) form is calculated and exchanged between bits and checks. An LDPC code follows a batch mode in its message exchange, such that all checks are updated simultaneously followed by all-bit-update. For TPC/SPC and TPC/MPC codes, since checks can be easily differentiated into two groups pertaining to component codes $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively, they can take a semi-sequential mode where half of the checks get updated first, followed by all-bit-update, and then the other half checks get updated, again followed by all-bit-update, as illustrated in Fig. 1 (black dots denote checks and white dots denote bits). There is no major difference between the two decoding strategies, except that the latter is expected to converge a bit faster. Hence, the behavioral difference of TPC/SPC, TPC/MPC and LDPC codes does not stem from their decoding strategies. Rather, it is rooted in the difference in their structural properties, like distance spectrum and loops. Exact descriptions of their decoding algorithms can be found, for example, in [4] for LDPC codes and in [8, 10] for TPC/SPC codes.

III. SYSTEM MODEL

Fig. 2 describes the system model used in this work. The outer code is one of the three modulation codes: TPC/SPC, TPC/MPC and LDPC codes. The inner code is a (precoded) PR channel such as those commonly used in magnetic recording (PR4: $1{-}D^2$, EPR4: $1{+}D{-}D^2{-}D^3$) and MO recording (PR1: $1{+}D$, PR2: $1{+}2D{+}D^2$) systems.

We would like to bring attention to the difference between the TPC and LDPC concatenation schemes

(Fig. 2). In the case of TPC/SPC and TPC/MPC codes, after user data get encoded, several codewords are combined and interleaved, then precoded, and finally passed to the PR channel. In the case of LDPC codes however, the interleaver and the precoder are not in use. This is because:

Interleaver – Since an LDPC code implicitly incorporates a random interleaver, no explicit interleaver is needed. It is worth emphasizing that in our approaches with TPC/SPC and TPC/MPC code on precoded PR channels, several codewords are combined before interleaving (inter-codeword interleaving). This is crucial because, as shown in [11], the reduction in word error rate is proportional to the number of codewords that are combined for interleaving. That is, the interleaving gain here is related to the number of codewords combined, rather than the interleaver block size as one would typically expect from Benedetto *et al*'s analysis [12].

Precoder – The precoder works to make the PR channels appear recursive to the outer code. Recursiveness is in general a required feature for the inner code in order to obtain interleaving gain [12]. The goal is to map low-weight error events to high-weight error events. Put another way, a recursive inner code will hopefully reduce the multiplicity (the number of the codewords) of low-weight error events. This effect is known as *spectral thinning*, which is recognized as the main contribution to the *interleaving gain* in a serially concatenated system. This is why although TPC/SPC and TPC/MPC codes are weak in AWGN channels, they are capable of high performance in precoded channels. For a randomly constructed LDPC code, since there are not many low-weight error events (large minimum distance), no noticeable spectrum thinning is obtained. On the other side, the presence of the precoder actually impairs channel conditions and consequently incurs performance loss especially in the first few iterations [13]. Simulations have shown that precoding causes a 0.5 to 1 dB loss for a rate 0.89 LDPC code over PR4 channels. So while the majority codes, including TPC/SPC and TPC/MPC codes, benefit from the recursiveness of the inner code, LDPC codes perform better with non-recursive inner codes.

Data storage systems require extremely low error rates, like 10^{-15} or lower. Since most codes have already reached error floors well before that, such low error rate is only achieved by using another Reed-Solomon error correction code (RS-ECC) after the modulation code to clear up residual errors. (RS-ECC is not shown in our system model.) An RS-ECC usually works on a symbol level (a symbol typically comprises of 8 or 10 consecutive bits) and is capable of correcting up to t symbol errors per block, where t ranges from 10 to 20 for a 4K bit block with single interleave. Therefore a modulation code is applicable only if its error statistics would not cause frequent RS-ECC failure. This error statistics criterion is unfortunately much neglected in most research works and in reality is the only relevant performance measure that needs to be investigated. For this reason, in our performance evaluation later, in addition to BER performance, we will pay close attention to error distributions, for it solely determines the block failure rate.

It is worth pointing out that there might exist modulation codes that have error floors below 10^{-15}. In that case, no RS-ECC is needed and therefore error statistics are not a concern. One possible instance of such codes might be LDPC codes with which no error floor is observed above 10^{-8}. However, since it is impractical to simulate down to 10^{-15}, no convincing evidence would indicate where the error floors appear. Hence in this work, we would still superimpose an RS-ECC in the LDPC case and check for its error statistics.

Finally, we note that the information exchange between the decoders is established via turbo equalization. The extrinsic information from one decoder is used as *a priori* information by the other. In this work, the inner decoder is a MAP (maximum *a posteriori*) decoder implementing the BCJR algorithm, and the outer decoder is either a TPC/SPC, TPC/MPC or LDPC decoder.

IV. THRESHOLD ANALYSIS

Precoding results in good distance spectrum for TPC/SPC and TPC/MPC systems, which guarantees good performance under optimal decoding. However, it is highly desirable to conduct an analysis which takes into consideration of the suboptimal nature of the iterative decoding algorithm. This section calculates the thresholds of TPC/SPC and LDPC systems using density evolution (DE), which is known for its power in the analysis and design of (irregular) LDPC codes [14, 15]. Due to the limitation on space, we show only the critical steps here. For a detailed analysis on thresholds computation and some related optimization issues, readers are referred to [11].

A. Problem Formulation

The idea of density evolution is to track the pdf (probability density function) of the message (in LLR form) during the q_{th} iteration, denoted by $f_{L_o^{(q)}}(x)$. If the bit s_k is zero, then a decoding success is achieved at q_{th} iteration when $f_{L_o^{(q)}}(x) = 0, \forall x < 0$ as $q \to \infty$. Since it is quite difficult to analytically evaluate $f_{L_o^{(q)}}(x)$ for all q, we approximate pdf's to be Gaussian distributed. Gaussian approximation (GA) has been used by Chung *et al* [15], and has been shown to significantly reduces the complexity with very minor loss in accuracy. Applying the consistency condition, $f_{L_o^{(q)}}(x) = f_{L_o^{(q)}}(-x) \cdot e^x$, 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. Hence the mean of the messages,

$m_o^{(q)}$ serves as the sufficient statistic of the message density. Under the assumption of an infinite block size and a perfect interleaver, the problem of finding the threshold can be formulated as (N is the block size):

$$\mathbf{C} = \inf_{SNR} \left\{ SNR : \lim_{q\to\infty} \lim_{N\to\infty} m_o^{(q)} \to \infty \right\}. \quad (1)$$

B. Message Flow within the Channel MAP Decoder

To evaluate the concatenated systems using density evolution, 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[1] Since there is no simple derivation available to analytically compute the pdf's in a MAP decoder (equalizer), Monte-Carlo simulations are used to determine a relationship as $m_i^{(q+1)} = \gamma_i(m_o^{(q)})$. $m_i^{(q+1)}$ 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 non-linear, the input sequence is not assumed to be all zeros, rather a sequence of i.i.d. bits.

C. Message Flow within the Outer Decoder

LDPC decoder – The LDPC decoder itself is an iterative decoder which uses L iterations to update extrinsic information passed between bits and checks. We use superscript (q) and (l) to denote quantities during the q_{th} iteration of turbo equalization (big loop) and l_{th} iteration within the LDPC decoder (local loop). Following much the same line of derivation as in [15], we have the messages flow within LDPC decoder as (s is the degree of a bit, t is the degree of a check):

$$\text{bit-to-check:} \quad m_b^{(q,l)} = m_o^q + (s-1)\cdot m_c^{(q,l-1)}, \quad (2)$$

$$\text{check-to-bit:} \quad m_c^{(q,l)} = \psi^{-1}\left(\left[\psi(m_b^{(q,l-1)})\right]^{t-1}\right), \quad (3)$$

where m_c and m_b decode the mean of messages passed from check to bit and bit to check, respectively, and $m_c(q,0) = 0, \forall q$. $\psi(x)$ calculates the expected value of $\tanh(\frac{u}{2})$, where u follows a Gaussian distribution with mean x and variance $2x$. $\psi(x)$ is given by:

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

Finally, after L_{th} local iterations, the message passed to inner MAP decoder is given by:

$$\text{LDPC-to-MAP:} \quad m_o^{(q)} = s \cdot m_c^{(q,L)}. \quad (5)$$

TPC/SPC decoder – A TPC/SPC code can be viewed as a special type of LDPC code, yet it is different in that it has small minimum distance of only 4, or equivalently, short cycles of length 8 in its bipartite graph, even if the block size goes to infinity. Since density evolution requires all messages exchanged to be independent, it has to operate on a cycle-free subgraph of the code. 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 update for each component code which yield the upper bound[2] of the threshold [11]. The procedure is similar to that of LDPC codes, but the check-to-bit update is done in two steps (K_1 and K_2 are the user data size of component code $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively):

$$\text{bit-to-check:} \quad m_{b_1}^{(q)} = m_i^{(q)}; \quad (6)$$

$$\text{check-to-bit } (\mathcal{C}_1): \quad m_{c_1}^{(q)} = \psi^{-1}([\psi(m_{b_1}^{(q)})]^{K_1}); \quad (7)$$

$$\text{bit-to-check:} \quad m_{b_2}^{(q)} = m_i^{(q)} + m_{c_1}^{(q)}; \quad (8)$$

$$\text{check-to-bit } (\mathcal{C}_2): \quad m_{c_2}^{(q)} = \psi^{-1}([\psi(m_{b_2}^{(q)})]^{K_2}); \quad (9)$$

$$\text{TPC/SPC-to-MAP:} \quad m_o^{(q)} = m_{c_1}^{(q)} + m_{c_2}^{(q)}; \quad (10)$$

D. Thresholds

Fig. 3 shows the thresholds of rate-0.89 and 0.94 TPC/SPC and LDPC codes over PR4/EPR4 channels. As can be seen, the upper bound of TPC/SPC is about 0.5 dB (or less) away from that of LDPC codes. This shows that the performance of TPC/SPC is expected to be within a few tenths of a dB from that of LDPC code. This is confirmed by simulation results, which are also shown for comparison purpose.

V. PERFORMANCE EVALUATION

This section presents analytical and numerical results. Although it has become customary just to show BER vs SNR plots for performance analysis, we take a different approach in the following subsections to shed light into the less understood aspects of iterative schemes.

A. Effect of Precoding

Recently, several independent studies [13, 16] were conducted from different perspectives towards the role a precoder plays on ISI channels. The *precoder weight gain* [16], and the *effective channels* [13] are used for evaluating the performance and/or convergence of precoded ISI channels. These investigations have cast great insight into the general understanding of precoders. However, the role of a precoder is best evaluated under the specific system setting and application requirements.

A typical precoder in magnetic recording channels takes the form of $1/(1 \oplus D^2)$, where $\oplus$ denotes modulo-2 addition. In [8], the effect of different precoders

[1]Subscript "o" stands for outer code and "i" for inner code.

[2]By upper bound, we mean an exact threshold should be better than this, ie, for a given rate, the required SNR could be smaller.

for TPC/SPC over PR4/EPR4 channels is studied and $1/(1 \oplus D^2)$ is shown through simulations to be the best among others, which well matches the analysis in [13].

Here, we focus the study of precoding on MO recording channels. Although the conventional choice is $1/(1 \oplus D)$ for PR1 channel and $1/(1 \oplus D^2)$ for PR2, our study shows that $1/(1 \oplus D^2)$ for PR1 and $1/(1 \oplus D)$ for PR2 are a better choice for TPC/SPC and TPC/MPC systems. Several important criteria are evaluated to justify our choice.

Simulations of differently precoded PR2 channels coded with a rate 0.89 TPC/SPC code are plotted in Fig. 4. The digit number beside each curve denotes the precoder in its octal form. For example, 3 denotes $1/(1 \oplus D)$ and 1 means no-precoding. The bit error rate after 3, 5, 10 and 15 turbo iterations are shown in different subplots. Fig. 5 shows the bit-error statistics (the number of erroneous bits in a block vs the number of such blocks) of the same setting after 10 turbo iterations[3]. Channel conditions with SNR=6.25 dB (left column) and 6.5 dB (right column) are evaluated. When given a target BER at around 10^{-5} and 10^{-6} (as is the typical target in the research in this area) and an up-limit of the delay and complexity to be not exceeding 10 iterations, and in particular, when error bursts are to be avoided as much as possible, $1/(1 \oplus D^2)$ seem to be a worse choice than either $1/(1 \oplus D)$ or $1/(1 \oplus D \oplus D^2)$. It is worth mentioning that, since recording channels are unstable and subject to fluctuations, it is desirable for a code to be able to work well in the range of $x \pm \Delta x$ rather than a single point of x dB. Similar results are observed for rate-0.94 TPC/SPC codes on PR2 channels. But this time, $1/(1 \oplus D)$ seems to perform slightly better than $1/(1 \oplus D \oplus D^2)$. Hence, $1/(1 \oplus D)$ is chosen for PR2 channels with outer TPC codes.

The investigation of the case for PR1 channels is shown in Fig. 6. As expected, the non-precoding case has flat curves, and the conventional precoder $1/(1 \oplus D)$ significantly steepens the curves without adding extra complexity. However, if one is willing to increase the states in the channel MAP decoder from 2 to 4, $1/(1 \oplus D^2)$ is capable of better performance. Since a 4-state MAP is reasonable in complexity, we use $1/(1 \oplus D^2)$ for PR1 channels.

In spite of our evident reasons for choosing the precoder, we are cautious in proposing it for all applications. As mentioned before, the choice is best evaluated against specific system requirements.

B. Code Specification and Decoding Complexity

Tab. 1 specifies the codes studied in this paper. The effective data block size, L, refers to the total number of user data bits combined for interleaving. In TPC/SPC systems, 4 codewords of rate $(32/33)^2 = 0.94$ code and 16 codewords of rate $(16/17)^2 = 0.89$ code are combined, respectively, to obtain an L=4K bits. This equivalence in effective block size is needed for a fair performance comparison with respect to delay, memory occupation and overhead.

Tab. 2 compares the complexity per decoded bit per iteration for the subject codes as well as for channel MAP decoding (assuming $\log\tanh(\frac{x}{2})$ is implemented using table-lookup). Apparently, the decoding complexity mainly come from the channel MAP decoder. Further, unlike LDPC codes which have quadratic encoding complexity ($O(N^2)$), TPC/SPC and TPC/MPC codes are linear time encodable.

Table 1: Code specification

Code	Rate (R)	Avg. degrees per bit (s)	Effective data block size (L)
TPC/SPC	$(\frac{32}{33})^2 = .94$	2	4K bits
	$(\frac{16}{17})^2 = .89$	2	4K bits
TPC/MPC	$(\frac{64}{66})^2 = .94$	2.16	4K bits
	$(\frac{64}{68})^2 = .89$	2.26	4K bits
LDPC	$\frac{16}{17} = .94$	3	4K bits
	$\frac{8}{9} = .89$	3	4K bits

Table 2: Complexity per decoded bit per iteration.

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

s: average connections per bit, see Table 1.
m: memory size of the convolutional code

C. Bit Error Rate

Fig. 7 shows the performance of TPC/SPC, TPC/MPC and LDPC codes over magnetic recording channels (PR4/EPR4). The error floors in TPC/SPC codes are possibly due to the existence of many neighboring codewords at minimum distance. A TPC/MPC code also has minimum distance of 4, but the number of such neighboring codewords is smaller. Fig. 8 shows the performance of TPC/SPC and LDPC codes over MO recording channels (PR1/PR2). All the curves shown in the figures are after 8 iterations. As can be seen, TPC/SPC and TPC/MPC achieve performance comparable to that of LDPC codes on PR4/EPR4 channels, with gains of 4.5 to 5 dB over uncoded systems at BER of 10^{-5}. On PR1 channels, LDPC codes are slightly better, but on PR2 channels, TPC/SPC codes outperform LDPC codes by 0.4 dB. Compared to uncoded systems, which require 10.2 dB for a PR1 channel and 11.7 dB for a PR2 channel to reach BER of 10^{-5}, a rate-0.94 TPC/SPC code gains 4.3 dB and 4.8 dB, re-

[3]Since random interleavers might affect the error statistics, we experimented with several randomly chosen interleavers and observed similar results every time.

spectively, and a rate-0.89 TPC/SPC code gains 5.2 dB and 5.7 dB, respectively.

D. Error Statistics

While BER performance is important, it is insufficient in estimating the block failure rate in data storage systems. Hence error statistics are examined to facilitate the investigation. Fig. 9, 10 and 11 plot the number of errors in each block vs the number of such blocks, for a rate 0.94 TPC/SPC, TPC/MPC and LDPC code on EPR4 channels, respectively. The left column plots bit error statistics and the right byte error statistics (a byte contains 8 consecutive bits). The statistics are made from observation of more than 160,000 blocks of size 4K bits. For clarity, the number of error-free blocks is not shown. As expected, the error statistics are similar for TPC/SPC and TPC/MPC codes and are quite different from LDPC codes. That TPC/SPC and TPC/MPC codes have many neighboring codewords and that neighbors do not differ in many bit positions may explain why they tend to have many block errors with very few bits in error in each block. On the other side, LDPC codes have much bursty error patterns. Although the data collected are not sufficient to draw convincing conclusions, they indicate that TPC/SPC and TPC/MPC would be more compatible with data storage systems where an outer RS-ECC code is expected to clear up the residue errors.

VI. CONCLUSION

Three types of graph-based block codes, TPC/SPC, TPC/MPC and LDPC, are investigated for their potential in data storage applications. Through a comprehensive evaluation which includes the effect of precoding, complexity, error rate and error statistics, we propose TPC/SPC to be a more promising candidate than LDPC codes for future recording systems. Some of the conclusions in this work are highlighted below:

1. Of the three types of codes considered, TPC/SPC seem to perform the best in terms of complexity, bit error rate and error statistics. In spite of the possible error floors, they are likely to work in more harmony with the outer RS-ECC than LDPC codes to ensure overall good performance. Their encouraging performance on PR channels should inspire further investigation into more realistic channel models, such as Lorentzian channels in magnetic recording systems.

2. Precoding affects the performance in many ways, including BER, convergence, error floor and error statistics. The merit of a precoder is case-dependent, and requires a careful judgment with respect to specific system settings and application needs. LDPC codes perform better without precoding whereas TPC/SPC and TPC/MPC obtain interleaving gain through the use of precoders.

3. Density evolution is a useful tool in the analysis of iterative procedures. The thresholds computed for TPC/SPC and LDPC systems indicate the capacity of the system performance, which match quite well with the actual simulations. A more in-depth study might lead to a better understanding of the message-passing algorithm as well as cast insight into the design of better codes.

References

[1] T. M. Duman and E. Kurtas, "Performance of turbo codes over EPR4 equalized high density magnetic recording channels", *Proc. Global Conf. Comm.*, 1999

[2] T. Souvignier and *et al*, "Turbo decoding for PR4: Parallel vs. serial concatenation", *Proc. Intl. Conf. Comm.*, pp.1638-1642, June, 1999

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

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

[5] R. M. Pyndiah, "Near-optimum decoding of product codes: block turbo codes", *IEEE Trans. Comm.*, vol.46, No.8, Aug, 1998

[6] J. L. Fan, A. Friedmann, E. Kurtas and S. McLaughlin, "Low density parity check codes for magnetic recording", *Proc. 37th Annual Allerton Conf. Comm., Control and Computing*, 1999

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

[8] J. Li, E. Kurtas, K. R. Narayanan and C. N. Georghiades, "On the Performance of Turbo Product Codes over Partial Response Channels",*8th Joint Magnetism and Magnetic Materials (MMM)-Intermag Conf.*, Jan. 2001, also to appear *IEEE Trans. Magnetics*, July, 2001

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

[10] J. Hagenauer, E. Offer and L. Papke, "Iterative decoding of binary block and convolutional codes", *IEEE Trans. Info. Theory*, vol.42, No.2, Mar., 1996

[11] J. Li, K. R. Narayanan, E. Kurtas and C. N. Georghiades, "On the Performance of High-Rate TPC/SPC and LDPC Codes over Partial Response Channels", submitted to *IEEE Trans. Comm.*

[12] S. Benedetto, D. Divsalar, G. Montorsi and F. Pollara, "Serial concatenation of interleaved codes: Design and performance analysis", *IEEE Trans. Info. Theory*, vol.42, pp.409-429, Apr., 1998

0-7803-7097-1/01/$10.00 ©2001 IEEE

[13] K. R. Narayanan, "Effect of precoding on the convergence of Turbo Equalization for Partial Response channels", accepted by *IEEE Jour. Sele. Areas Comm.*

[14] T. Richardson and R. Urbanke, "The capacity of low-density parity check codes under message-passing decoding", submitted to *IEEE Trans. Info. Theory*

[15] S. Y. Chung, R. Urbanke and T. J. Richardson, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation", submitted to *IEEE Trans. Info. Theory*, June, 2000

[16] I. Lee, "The effect of a precoder on serially concatenated coding systems with ISI channel", *Proc. Intl. Conf. Comm.*, June, 2000

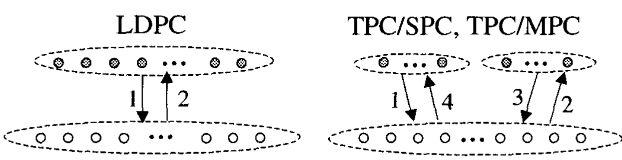

Figure 1: Bipartite graph and message flow

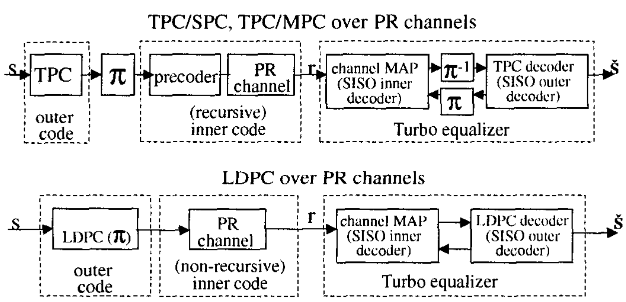

Figure 2: System Models

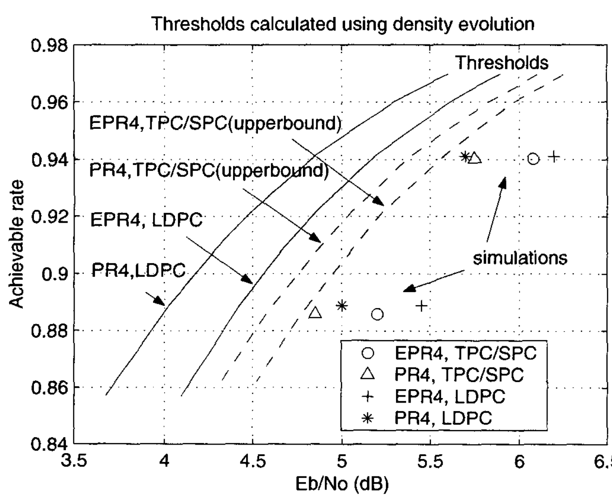

Figure 3: γ_o for the 1_{st} and 3_{rd} turbo iteration

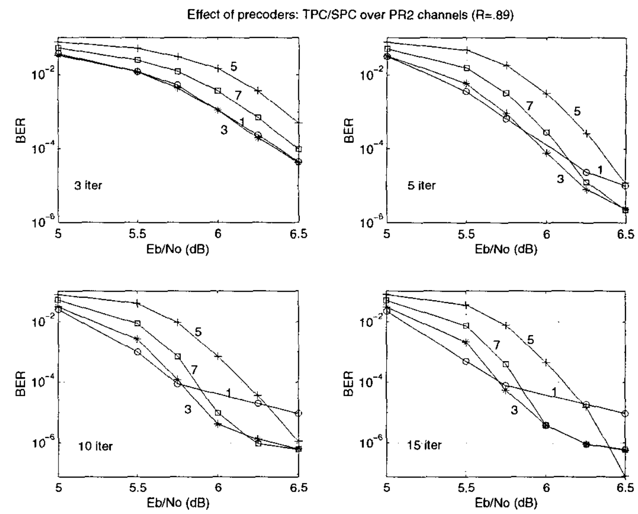

Figure 4: Effect of precoding for TPC/SPC on PR2 channels

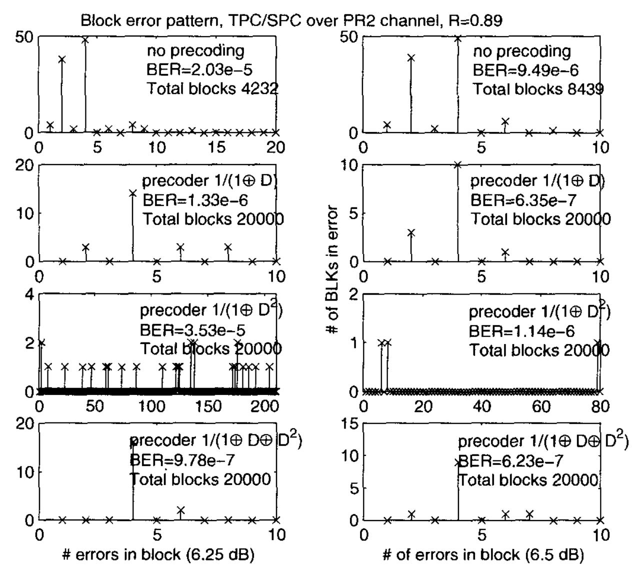

Figure 5: Error distributions of different precoders for TPC/SPC on PR2 channels

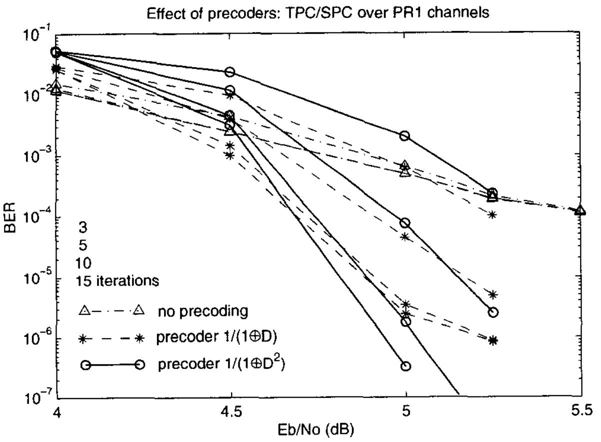

Figure 6: BER and convergence of different precoders for TPC/SPC on PR1 channels

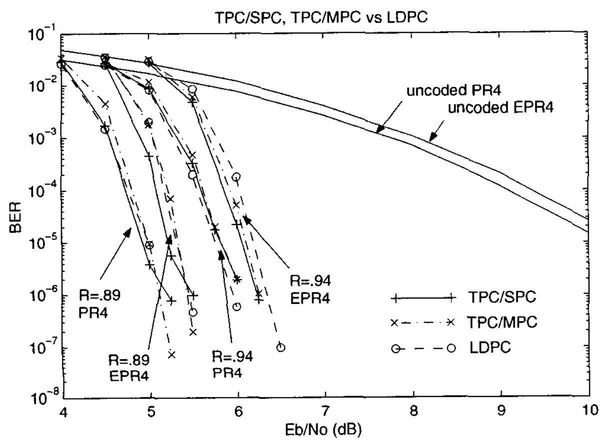

Figure 7: Performance of TPC/SPC, TPC/MPC and LDPC on magnetic recording channels

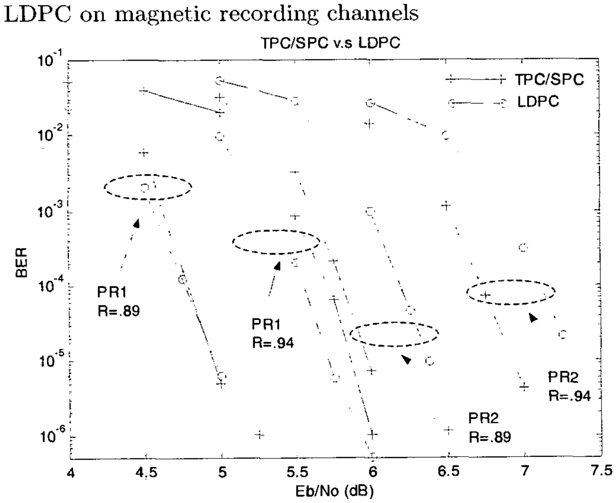

Figure 8: Performance of TPS/SPC and LDPC on MO recording channels

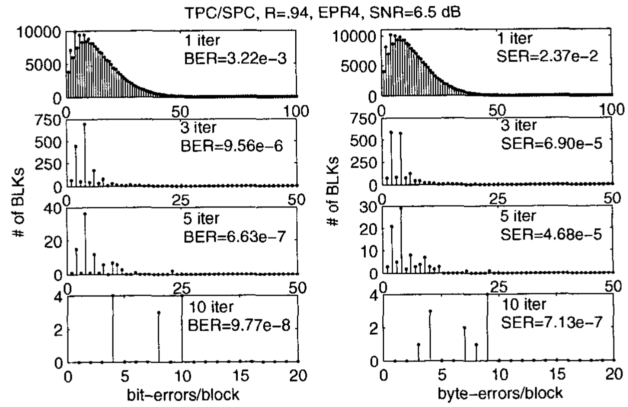

Figure 9: Error statistics of TPC/SPC on EPR4 channels

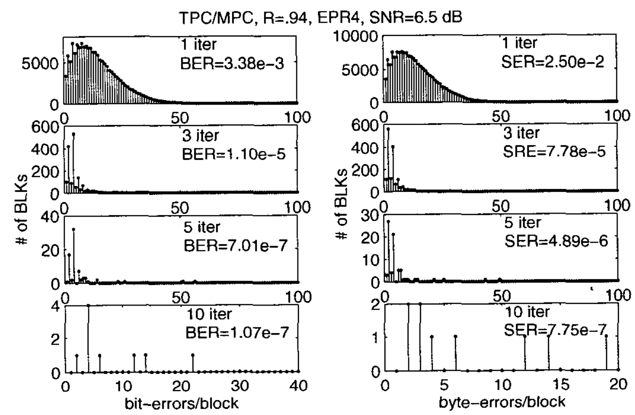

Figure 10: Error statistics of TPC/MPC on EPR4 channels

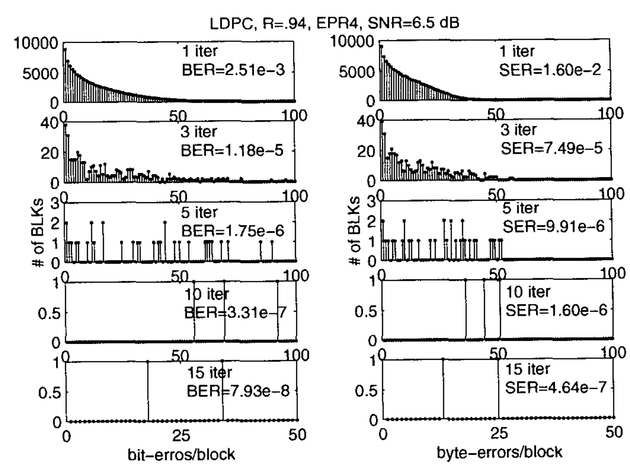

Figure 11: Error statistics of LDPC on EPR4 channels