[10] G. D. Forney, Jr., "Generalized minimum distance decoding," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 125–131, Apr. 1966.
[11] D. Chase, "A class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 170–181, Jan. 1972.
[12] J. Lacan and E. Delpeyroux, "Permutation group of the q-ary image of some q^m-ary cyclic codes," in *Finite Field: Theory, Applications and Algorithms*. Providence, RI: Amer. Math. Soc., 1999, Contemporary Mathematics, 225.
[13] O. Papini and J. Wolfmann, *Algèbre Discrète et Codes Correcteurs (Collections Mathématiques et Applications)*. Berlin, Germany: Springer-Verlag, 1995, vol. 20, .
[14] D. J. Taipale and M. J. Seo, "An efficient soft-decision Reed–Solomon decoding algorithm," *IEEE Trans. Inform. Theory*, vol. 43, pp. 1130–1139, July 1994.
[15] R. Kötter, "Fast generalized minimum-distance decoding of algebraic-geometry and Reed–Solomon codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 721–737, May 1996.
[16] H. Tang, Y. Liu, M. Fossorier, and S. Lin, "On combining chase-2 and GMD decoding algorithms for nonbinary block codes," *IEEE Commun. Lett.*, vol. 5, pp. 209–211, May 2001.
[17] B. Vucetic, V. Ponnampalam, and J. Vuckovic, "Low complexity soft decision algorithms for Reed–Solomon codes," *IECE Trans. Commun.*, vol. E84-B, no. 3, pp. 392–399, Mar. 2001.

Variance of the Turbo Code Performance Bound Over the Interleavers

Atousa H. S. Mohammadi, *Member, IEEE,* and
Weihua Zhuang, *Senior Member, IEEE*

***Abstract*—In this correspondence, we evaluate the variance of the union performance bound for a rate-$1/3$ turbo code over all possible interleavers of length N, under the assumption of a maximum-likelihood (ML) decoder. Theoretical and simulation results for turbo codes with two-memory component codes indicate that the coefficient of variation of the bound increases with the signal-to-noise ratio and decreases with the interleaver length. Theoretical analysis for large interleaver lengths shows that the coefficient of variation asymptotically approaches a constant value. The results also demonstrate that the majority of the interleavers have performance bounds very close to the average value of the bound. This phenomenon is more palpable for larger interleaver lengths.**

***Index Terms*—Channel coding, concatenated codes, turbo codes, union performance bound.**

I. Introduction

Turbo codes, introduced in 1993 [1], are composed of the parallel concatenation of two (or more) recursive systematic convolutional (RSC) component codes, connected through an interleaver(s). The interleaver, which reorders the input block of data given to the second encoder, plays a key role in the pseudorandom nature and, consequently, the high performance of turbo codes. Thus, the study and design of the interleaver has been an attractive subject for many researchers in this area.

In [2], an interleaver design technique is proposed which searches for a random interleaver resulting in the fewest output sequences with low weights corresponding to input weights of 2 or 3. The authors then use simulation results to show that for short frame transmission systems and bit error rates (BERs) of around 10^{-3}, a block interleaver outperforms the best such found pseudorandom interleaver, and the overall effect of the interleaver is not significant in this range [3]. In [4], however, it is shown that, for turbo codes of large interleaver lengths, pseudorandom interleavers outperform block interleavers significantly, e.g., 2.7 dB at BER of 10^{-5}. Recently, a systematic approach for the design of the interleaver has been proposed in [5]. The method is based on recursively minimizing a cost function to find an interleaver which best breaks a set of *a priori* chosen error patterns. The weight distribution of a turbo code employing the best such found interleaver of length 100 shows 0.5- to 0.9-dB improvement over a randomly selected interleaver of the same length. In [6], a deterministic interleaver design algorithm is proposed based on linear recursion to produce an initial interleaver which is subsequently optimized by pairwise exchange of its elements. These optimized interleavers show more than 0.5-dB improvement over a randomly selected interleaver and about 0.2-dB improvement over an S-random interleaver for BERs of less than 10^{-5} and block length 576. In [7], a mathematical structure is developed for turbo-code interleaver design at low BERs, which achieves more than 0.5-dB improvement over random interleavers for interleaver length 1176. In [8], high-spread interleavers have been designed for specific short interleaver lengths. These interleavers are shown to significantly lower the error floor occurring at high signal-to-noise ratios. Other works related to the design of the interleaver include [9]–[13].

Although the above works implicitly suggest some conclusions regarding the effect of different choices of interleavers on the performance of turbo codes, they are mainly focused on either search algorithms for the best (or at least *good*) interleaver(s) or explaining the behavior of these codes in general. So far, the only statistical study of the turbo code behavior with respect to interleavers considers the upper bound on the maximum-likelihood (ML) performance of the turbo code, averaged over all possible interleavers (e.g., [14]).

If higher order statistical averages of the turbo code performance with respect to the interleaver are known, it will be possible to have a more accurate estimate of the distribution of the performance bound with respect to the interleaver. As a first step, in this correspondence, we study the effect of the interleaver by looking at the variance of the turbo-code performance with respect to all possible interleavers of the same length, under the assumption of an ML decoder. Note that, in practice, turbo codes are decoded iteratively using a non-ML decoder, however, it is a widely accepted conjecture that the performance of the suboptimum iterative decoding converges toward the ML performance. This study tackles the question brought up in [14, Question 3] to give more insight regarding what performance to expect from a turbo code with fixed component codes and interleaver length. It also provides an estimate of how well a particular interleaver performs among the range of all possible interleavers and helps to evaluate the performance of an interleaver search algorithm.

The correspondence is organized as follows. In Section II, a brief review of the turbo-code average performance bound [14] is given and following that, the mathematical formulations for the second moment

Manuscript received June 17, 1999; revised January 21, 2002. This work was supported by Communications and Information Technology Ontario (CITO) and by the Natural Sciences and Engineering Research Council (NSERC) of Canada. The material in this correspondence was presented in part at the IEEE International Symposium on Information Theory (ISIT'98), Cambridge, MA, August 1998 and in part at the IEEE Vehicular Technology Conference (VTC'99), Houston, TX, May 1999.

A. H. S. Mohammadi is with LSI Logic Inc., Mailstop AH340, Milpitas, CA 95035 USA (e-mail: atousa@lsil.com).
W. Zhuang is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: wzhuang@bbcr.uwaterloo.ca).

Communicated by F. R. Kschischang, Associate Editor for Coding Theory.
Publisher Item Identifier S 0018-9448(02)05159-3.

of the weight enumerating function (WEF) and of the union performance bound for a turbo code are derived. Asymptotic analysis of the derived formulas for large interleaver lengths is presented in Section III. Section IV shows some numerical and simulation results for the nonasymptotic and asymptotic cases, and explains the approach in deriving those results. Finally, Section V concludes this correspondence.

II. Variance of the Performance Bound

The presence of the interleaver in the turbo-code structure makes it very difficult to enumerate the exact weight distribution of the code. The idea of averaging the performance of the code over all possible interleavers was presented in [14] by introducing the concept of the "uniform interleaver" (UI). The UI is a hypothetical interleaver that selects a permutation for each block of data uniformly at random from the set of all possible permutations. Coding with this hypothetical turbo code is equivalent to coding with a turbo code which randomly chooses an interleaver for each block of input data independently from all other blocks. The WEF of a turbo code employing a UI of length N is in fact the average weight enumerating function (AWEF) of the turbo code over all possible interleavers of this length. The truncated AWEF is then used to find the average performance bound of a turbo code of interleaver length N [14].

Consider a rate–$1/3$ turbo code of interleaver length N. Let S_N denote the set of all possible permutations of the length N input block, and let π be the interleaver selected from S_N with uniform probability. For a turbo code employing interleaver π, let $X_{i,j_1,j_2}(\pi)$ denote the number of codewords having total weight $i+j_1+j_2$ with i, j_1, j_2 ones in the systematic, first encoder parity check, and second encoder parity check bits, respectively. Define the WEF $A(W, Z_1, Z_2, \pi)$ as

$$A(W, Z_1, Z_2, \pi) = \sum_{i\geq 1, j_1, j_2 \geq 0} X_{i,j_1,j_2}(\pi) W^i Z_1^{j_1} Z_2^{j_2}.$$

Then the AWEF is

$$\boldsymbol{E}\left[A(W, Z_1, Z_2, \pi)\right] = \sum_{i\geq 1, j_1, j_2 \geq 0} \boldsymbol{E}\left[X_{i,j_1,j_2}(\pi)\right] W^i Z_1^{j_1} Z_2^{j_2} \tag{1}$$

where

$$\boldsymbol{E}\left[X_{i,j_1,j_2}(\pi)\right] = \frac{1}{N!}\sum_{\pi\in S_N} X_{i,j_1,j_2}(\pi).$$

Let $B(\pi, E_b/N_0)$ denote the union bound on the decoded BER given the interleaver π and the received E_b/N_0 (the ratio of the signal energy per information bit to the one-sided power spectral density of the channel additive white Gaussian noise), under the assumption that the decoder selects the ML codeword. The value of the bound can be calculated by

$$B(\pi, E_b/N_0) = \sum_{i\geq 1, j_1, j_2 \geq 0} \frac{i}{N} X_{i,j_1,j_2}(\pi) Q(i, j_1, j_2, E_b/N_0)$$

where

$$Q(i, j_1, j_2, E_b/N_0) = \frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{E_b}{3N_0}(i+j_1+j_2)}\right).$$

The average union bound (with respect to the randomly chosen interleaver) can be evaluated as

$$\boldsymbol{E}\left[B(\pi, E_b/N_0)\right] = \sum_{i\geq 1, j_1, j_2 \geq 0} \frac{i}{N} \boldsymbol{E}\left[X_{i,j_1,j_2}(\pi)\right] Q(i, j_1, j_2, E_b/N_0). \tag{2}$$

Note that, in (2), adding the term for $i=0$ to the right-hand side does not change the value of the summation.

Here, we attempt to compute the variance of the bound $B(\pi, E_b/N_0)$ over the ensemble of the interleavers S_N, i.e.,

$$\mathbf{Var}\left[B(\pi, E_b/N_0)\right] = \boldsymbol{E}\left[B^2(\pi, E_b/N_0)\right] - \boldsymbol{E}^2\left[B(\pi, E_b/N_0)\right]. \tag{3}$$

To evaluate the second moment of the performance bound

$$\boldsymbol{E}\left[B^2(\pi, E_b/N_0)\right] = \sum_{i\geq 1, j_1, j_2 \geq 0} \sum_{i'\geq 1, j_1', j_2' \geq 0} \frac{ii'}{N^2} \boldsymbol{E}\left[X_{i,j_1,j_2}(\pi) X_{i',j_1',j_2'}(\pi)\right] \cdot Q(i, j_1, j_2, E_b/N_0) \cdot Q(i', j_1', j_2', E_b/N_0) \tag{4}$$

we start with the mean square of the WEF (MSWEF), which is equal to

$$\boldsymbol{E}\left[A^2(W, Z_1, Z_2, \pi)\right] = \sum_{i\geq 1, j_1, j_2 \geq 0} \sum_{i'\geq 1, j_1', j_2' \geq 0} \boldsymbol{E}\left[X_{i,j_1,j_2}(\pi) X_{i',j_1',j_2'}(\pi)\right] \cdot W^i Z_1^{j_1} Z_2^{j_2} W'^{i'} Z_1'^{j_1'} Z_2'^{j_2'}. \tag{5}$$

It should be noted that the random variables $X_{i,j_1,j_2}(\pi)$ and $X_{i',j_1',j_2'}(\pi)$ are not independent as they both depend on the same permutation π. The value of one random variable (e.g., $X_{i,j_1,j_2}(\pi)$) imposes restriction on the structure of the interleaver and consequently on the value of other random variables (e.g., $X_{i',j_1',j_2'}(\pi)$). Thus, in evaluating the MSWEF, we assume that every probabilistic experiment consists of choosing any one of the $N!$ possible interleavers with equal probability, and fixing it for the rest of that experiment.

For simplicity in notations, in the following, $X_{i,j_1,j_2}(\pi)$ and $X_{i',j_1',j_2'}(\pi)$ are, respectively, replaced by $X(\pi)$ and $X'(\pi)$, for fixed values of $i, j_1, j_2, i', j_1', j_2'$. In order to find $\boldsymbol{E}[X(\pi)X'(\pi)]$, the following definitions are made. Let $S_{x,y}$ be the set of all input words of weight x which result in codewords of weight y from the first component code, and $S^I_{x,y}$ be the set of all input words of weight x which result in codewords of weight y from the second component code. Also let

i) $S_{i,j_1} = \{s_1, \ldots, s_G\}$, where $G = |S_{i,j_1}|$,
ii) $S_{i',j_1'} = \{s_1', \ldots, s_{G'}'\}$, where $G' = |S_{i',j_1'}|$,
iii) $S^I_{i,j_2} = \{s_1^I, \ldots, s_K^I\}$, where $K = |S^I_{i,j_2}|$,
iv) $S^I_{i',j_2'} = \{s_1'^I, \ldots, s_{K'}'^I\}$, where $K' = |S^I_{i',j_2'}|$.

Letting

$$X_g(\pi) = \begin{cases} 1, & \text{if } \pi(s_g) \in S^I_{i,j_2} \\ 0, & \text{otherwise} \end{cases}$$

and

$$X'_{g'}(\pi) = \begin{cases} 1, & \text{if } \pi(s'_{g'}) \in S^I_{i',j_2'} \\ 0, & \text{otherwise} \end{cases}$$

we have

$$\begin{aligned} X(\pi) &= X_1(\pi) + X_2(\pi) + \cdots + X_G(\pi) \\ X'(\pi) &= X_1'(\pi) + X_2'(\pi) + \cdots + X_{G'}'(\pi) \end{aligned} \tag{6}$$

and consequently

$$\boldsymbol{E}\left[X(\pi)X'(\pi)\right] = \boldsymbol{E}\left[\sum_{g=1}^{G} X_g(\pi) \sum_{g'=1}^{G'} X'_{g'}(\pi)\right] = \sum_{g=1}^{G}\sum_{g'=1}^{G'} \boldsymbol{E}\left[X_g(\pi) X'_{g'}(\pi)\right]. \tag{7}$$

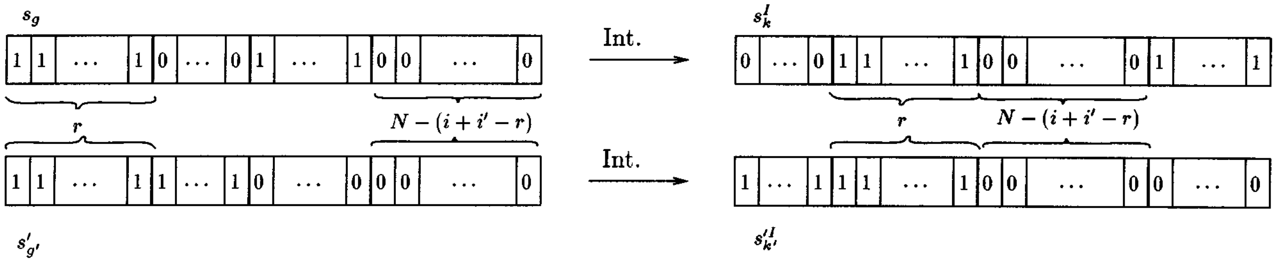

Fig. 1. An example of two input words and their interleaved versions with r commonly positioned 1's.

Since the random variables $X_g(\pi)$ and $X'_{g'}(\pi)$ only take on the value of 0 or 1, we have

$$\begin{aligned}\boldsymbol{E}\left[X_g(\pi)X'_{g'}(\pi)\right] &= \Pr\left[X_g(\pi)=1,\, X'_{g'}(\pi)=1\right]\\ &= \Pr\left\{\bigcup_{k=1}^{K}\bigcup_{k'=1}^{K'}\left[\pi(s_g)=s_k^I,\, \pi(s'_{g'})=s'^I_{k'}\right]\right\}\\ &= \sum_{k=1}^{K}\sum_{k'=1}^{K'}\Pr\left[\pi(s_g)=s_k^I,\, \pi(s'_{g'})=s'^I_{k'}\right] \qquad (8)\end{aligned}$$

where $\Pr[\pi(s_g)=s_k^I,\, \pi(s'_{g'}=s'^I_{k'}])$ is the probability that the randomly chosen interleaver π re-orders the input words s_g and $s'_{g'}$ into s_k^I and $s'^I_{k'}$, respectively, and the last equality results from the fact that the events $\pi(s_g)=s_k^I$ and $\pi(s'_{g'})=s'^I_{k'}$ are disjoint for different values of k or k'. Substituting (8) in (7), we get

$$\boldsymbol{E}\left[X(\pi)X'(\pi)\right] = \sum_{s_g\in S_{i,j_1}}\sum_{s_{g'}\in S_{i',j'_1}}\sum_{s_k^I\in S^I_{i,j_2}}\sum_{s'^I_{k'}\in S^I_{i',j'_2}} \cdot \Pr\left[\pi(s_g)=s_k^I,\, \pi(s'_g)=s'^I_{k'}\right]. \qquad (9)$$

The following theorem evaluates the probability involved in (9).

Theorem 1: If s_g and $s'_{g'}$ have r 1's in common positions, and s_k^I and $s'^I_{k'}$ have r^I 1's in common positions, where $0 \le r,\, r^I \le \min(i,\, i')$, then

$$\Pr\left[\pi(s_g)=s_k^I,\, \pi(s'_g)=s'^I_{k'}\right] = \begin{cases}\dfrac{r!(i-r)!(i'-r)![N-(i+i'-r)]!}{N!}, & \text{if } r=r^I\\ 0, & \text{if } r\neq r^I.\end{cases} \qquad (10)$$

Proof: Since s_g and $s'_{g'}$ have r commonly positioned 1's, their interleaved versions should have the same number of commonly positioned 1's, as well. Thus, there exists no interleaver that realizes this event when $r\neq r^I$, i.e.,

$$\Pr\left[\pi(s_g)=s_k^I,\, \pi(s'_{g'})=s'^I_{k'}\right]=0, \qquad \text{if } r\neq r^I.$$

Fig. 1 shows a schematic example of s_g and $s'_{g'}$, interleaved to s_k^I and $s'^I_{k'}$, respectively, where $r=r^I$. In this case, the interleaver has the following properties:

- mapping the r positions of the common 1's in s_g and $s'_{g'}$ to the r positions of the common 1's in s_k^I and $s'^I_{k'}$;
- mapping the $(i-r)$ remaining 1's in s_g to the positions of the remaining $(i-r)$ 1's in s_k^I;
- mapping the $(i'-r)$ remaining 1's in $s'_{g'}$ to the positions of the remaining $(i'-r)$ 1's in $s'^I_{k'}$;
- finally, mapping the $[N-(i+i'-r)]$ positions of the common 0's in s_g and $s'_{g'}$ to the positions corresponding to the common 0's in s_k^I and $s'^I_{k'}$.

Within each of the above bit groups, the permutation of the bits does not matter. Thus, the number of interleavers satisfying the above conditions is

$$r!(i-r)!(i'-r)!\left[N-(i+i'-r)\right]!$$

and, consequently, the probability that one of these interleavers is selected when randomly picked from the ensemble of all $N!$ interleavers is

$$\frac{r!(i-r)!(i'-r)!\,[N-(i+i'-r)]!}{N!}.$$

In fact, the number of the interleavers having the above mentioned properties is given by the multinomial coefficient

$$\binom{N}{r,\, i-r,\, i'-r,\, N-(i+i'-r)}.$$

With the UI, all the interleavers are equally likely. Hence, the probability of choosing one of such interleavers is the inverse of the multinomial coefficient. □

We denote the nonzero part of the probability in (10) with $\rho_r(i,\, i')$. Note that in (9), for each pair of input words $(s_g,\, s'_{g'})$ with r commonly positioned 1's, only those pairs of input words $(s_k^I,\, s'^I_{k'})$ which also have r commonly positioned 1's result in the nonzero terms $\rho_r(i,\, i')$. Let $q_r(i,\, j_1,\, i',\, j'_1)$ and $q_r(i,\, j_2,\, i',\, j'_2)$ denote the number of input word pairs $(s_g,\, s'_{g'})$ and $(s_k^I,\, s'^I_{k'})$, respectively, where each pair has r commonly positioned 1's. Equation (9) can now be written as

$$\boldsymbol{E}\left[X(\pi)X'(\pi)\right] = \sum_{r=0}^{\min(i,\,i')} q_r(i,\, j_1,\, i',\, j'_1)q_r(i,\, j_2,\, i',\, j'_2)\rho_r(i,\, i'). \qquad (11)$$

And, finally, substituting (11) in (4) results in

$$\begin{aligned}\boldsymbol{E}\left[B^2(\pi,\, E_b/N_0)\right] &= \sum_{i,\,j_1,\,j_2\ge 0}\sum_{i',\,j'_1,\,j'_2\ge 0}\sum_{r=0}^{\min(i,\,i')}\frac{ii'}{N^2}\,q_r(i,\, j_1,\, i',\, j'_1)\\ &\quad\cdot q_r(i,\, j_2,\, i',\, j'_2)\cdot\rho_r(i,\, i')Q(i,\, j_1,\, j_2,\, E_b/N_0)\\ &\quad\cdot Q(i',\, j'_1,\, j'_2,\, E_b/N_0). \qquad (12)\end{aligned}$$

The variance of the bound can then be evaluated.

III. Asymptotic Behavior

In this section, we study the MSWEF and $\boldsymbol{E}[B^2(\pi,\, E_b/N_0)]$ for a large interleaver length. The function $q_r(i,\, j,\, i',\, j')$, defined in

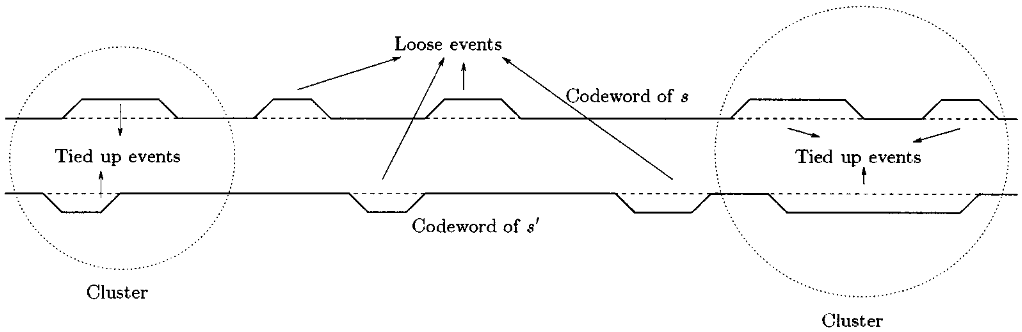

Fig. 2. Schematic illustration of the relative position of two codewords resulting from input words s and s' with $n = 5,\ n' = 4,\ m = 3,\ m' = 2$, and $c = 2$.

Section II, depends on N as well as the component codes. In the following, this function will be further studied in order to represent the MSWEF and $\boldsymbol{E}[B^2(\pi, E_b/N_0)]$ in the form of polynomials in N. The behavior of these functions for asymptotically large N is dominated by the terms with the highest power of N.

A codeword of the form W^iZ^j in each component code is constructed by the concatenation of n $(1 \le n \le \lfloor i/2 \rfloor)$ error events with possible zeros in between consecutive error events. Suppose two input words, s and s', consisting of n and n' error events, respectively, have r commonly positioned 1's. Furthermore, assume that these r common bits are contained in m and m' error events of s and s', respectively, where $\min(1, r) \le m \le \min(r, n)$ and $\min(1, r) \le m' \le \min(r, n')$. We call these error events the "tied up" events. The reason for choosing this term is that each of these error events is bound to be placed in a certain position(s) relative to one or more tied up events of the other codeword, such that the two codewords have r commonly positioned 1's in their corresponding input words. The remaining $(n - m)$ and $(n' - m')$ error events (called "loose" events) can be placed anywhere along the block of length N as long as they do not cause extra overlapping 1's between the two input words. Fig. 2 illustrates the above definitions, where the horizontal lines represent the all-zero path and the diverged paths represent the error events. The tied up events form c "cluster" of events, $\min(1, r) \le c \le \min(m, m')$. These c clusters plus the $(n - m) + (n' - m')$ loose error events can be placed in different positions along the block of length N by adding zeros between them or placing them adjacent to each other. This can be done in $\binom{N-L_t+1}{n+n'-m-m'+c}$ ways, where L_t is the summation of the lengths of the clusters and loose events in both codewords. The length of the cluster is defined as the distance between the point where the first error event in the cluster starts, up to the point where the last error event in that cluster ends. In the following formulas for asymptotic behavior, $(L_t - 1)$ is eliminated with respect to N in order to reduce the complexity of computations.

Now $q_r(i, j, i', j')$ can be written as

$$q_r(i, j, i', j') = \sum_n \sum_{n'} \sum_m \sum_{m'} \sum_c \cdot \hat{q}_r(i, j, i', j', n, n', m, m', c) \binom{N}{n + n' - m - m' + c} \tag{13}$$

where $\hat{q}_r(i, j, i', j', n, n', m, m', c)$ is the number of codeword pairs of the form $(W^iZ^j, W'^{i'}Z'^{j'})$ with n and n' error events; these error events have r commonly positioned 1's, resulting in m and m' tied up events which form c clusters. Note that $\hat{q}_r$ depends only on the component codes and not on N. As a result, (11) can be rewritten as

$$\boldsymbol{E}\left[X(\pi)X'(\pi)\right] = \sum_r \sum_{e_1 \in \mathcal{E}_1} \sum_{e_2 \in \mathcal{E}_2} C_{e_1} f_{e_1}(N) C_{e_2} f_{e_2}(N) \rho_r(i, i') \tag{14}$$

where $\mathcal{E}_1$ and $\mathcal{E}_2$ are the sets of the possible 5-tuples

$$(n_1, n_1', m_1, m_1', c_1) \quad \text{and} \quad (n_2, n_2', m_2, m_2', c_2)$$

respectively, and

$$C_{e_1} = \hat{q}_r(i, j_1, i', j_1', n_1, n_1', m_1, m_1', c_1)$$
$$f_{e_1}(N) = \binom{N}{n_1 + n_1' - m_1 - m_1' + c_1}$$
$$C_{e_2} = \hat{q}_r(i, j_2, i', j_2', n_2, n_2', m_2, m_2', c_2)$$
$$f_{e_2}(N) = \binom{N}{n_2 + n_2' - m_2 - m_2' + c_2}.$$

Substituting (14) in (5) and using the approximation $\binom{a}{b} \approx a^b/b!$ for $a \gg b$, we have

$$\boldsymbol{E}\left[A^2(W, Z_1, Z_2, \pi)\right] \approx \sum_{t \in \mathcal{T}} F_t N^{G_t} W^i Z_1^{j_1} Z_2^{j_2} W'^{i'} Z_1'^{j_1'} Z_2'^{j_2'} \tag{15}$$

where $\mathcal{T}$ represents the set of possible 17-tuples of all the variables involved in the summation,

$$G_t = n_1 + n_1' + n_2 + n_2' - m_1 - m_1' - m_2 - m_2' - i - i' + c_1 + c_2 + r$$

and F_t is not a function of N. The following theorem finds the conditions under which G_t takes its maximum value.

Theorem 2: The maximum value of G_t is equal to 0 and is achieved if and only if the following conditions hold:

$$\begin{aligned} i &= 2k \\ n_1 = n_2 &= i/2 = k \\ i' &= 2k' \\ n_1' = n_2' &= i'/2 = k' \\ r &= 2l \\ m_1 = m_2 = m_1' = m_2' = c_1 = c_2 &= r/2 = l \end{aligned} \tag{16}$$

where $k, k' = 1, 2, 3, \ldots$, and $l = 0, 1, 2, \ldots, \min(k, k')$.

Proof: It can be easily seen that, in G_t, maximizing the terms corresponding to subscript 1 and those corresponding to 2 can be performed independently, i.e.,

$$(G_t)_{\max} = (n_1 + n_1' - m_1 - m_1' + c_1)_{\max} + (n_2 + n_2' - m_2 - m_2' + c_2)_{\max} - i - i' + r.$$

Thus, in this proof, we only find $M_1 = (n_1 + n_1' - m_1 - m_1' + c_1)_{\max}$ as the maximization corresponding to the second RSC code follows with exactly the same analogy.

Suppose m_1 of the n_1 error events of the input word s are tied up with m_1' of the n_1' error events of the input word s', and the tied up events result in c_1 clusters. Since s and s' have r commonly positioned 1's, together they contain $i + i' - r$ positions containing a 1. Define a "bunch" of events to be either a cluster or a loose event. The number of bunches is $b = c_1 + (n_1 - m_1) + (n_1' - m_1')$. Since for recursive convolutional codes each error event has a Hamming weight of at least two in the systematic part, and each bunch contains at least one error event

$$c_1 + n_1 - m_1 + n_1' - m_1' \le \frac{i + i' - r}{2}. \tag{17}$$

The equality holds if and only if each loose event and each cluster contains exactly two positions containing a 1. This condition is met by codewords satisfying the following conditions.

- Each codeword is constructed of error events with input weight 2.
- Each error event is either a loose event or exactly matches and is tied up with an error event of the other codeword.

Similarly

$$c_2 + n_2 - m_2 + n_2' - m_2' \le \frac{i + i' - r}{2} \tag{18}$$

with equality holding under the same conditions. As a result

$$(G_t)_{\max} = 0. \tag{19}$$

□

Keeping only the terms with the largest power of N (i.e., corresponding to the terms with the largest value of G_t) and defining a new function

$$\tilde{q}(k, j, k', j', l) \triangleq \hat{q}_{2l}(2k, j, 2k', j', k, k', l, l, l)$$

result in the following formula for the asymptotic MSWEF:

$$\begin{aligned} &\boldsymbol{E}\left[A^2(W, Z_1, Z_2, \pi)\right] \\ &= \sum_{(k, k', j_1, j_2, j_1', j_2', l)} \frac{(2l)!(2k-2l)!(2k'-2l)!}{[(k+k'-l)!]^2} \\ &\quad \cdot \tilde{q}(k, j_1, k', j_1', l)\tilde{q}(k, j_2, k', j_2', l) \\ &\quad \cdot W^i Z_1^{j_1} Z_2^{j_2} W'^{i'} Z_1'^{j_1'} Z_2'^{j_2'}. \end{aligned} \tag{20}$$

The asymptotic formula for the mean square of the bound is then

$$\begin{aligned} &\boldsymbol{E}\left[B^2(\pi, E_b/N_0)\right] \\ &= \sum_{(k, k', j_1, j_2, j_1', j_2', l)} \frac{(2k)(2k')}{N^2} \frac{(2l)!(2k-2l)!(2k'-2l)!}{[(k+k'-l)!]^2} \\ &\quad \cdot \tilde{q}(k, j_1, k', j_1', l)\tilde{q}(k, j_2, k', j_2', l) Q(i, j_1, j_2, E_b/N_0) \\ &\quad \cdot Q(i', j_1', j_2', E_b/N_0). \end{aligned} \tag{21}$$

As can be seen from (21), $\boldsymbol{E}[B^2(\pi, E_b/N_0)]$ is proportional to N^{-2}. On the other hand, $\boldsymbol{E}[B(\pi, E_b/N_0)]$ is proportional to N^{-1} for asymptotically large N [15]. Thus, the variance of the performance bound is also proportional to N^{-2}. As a result, we see that the coefficient of variation of $B(\pi, E_b/N_0)$ does not change with N as N approaches infinity.

IV. Numerical Results and Discussion

To reduce the operational complexity in calculating the numerical results, the looser upper bound of [14], using $Q(x) \le \exp(-x^2)$, has been adopted.

A. Nonasymptotic Case

For this case, $\boldsymbol{E}[B^2(\pi, E_b/N_0)]$ is evaluated according to (12). We consider a super error state diagram constructed by the combination of the component code error digram with itself.[1] The super state diagram is constructed as follows.

- The state $[S, S']$ corresponds to states S and S' of the RSC code.
- The transition labels from state $[S_1, S_1']$ to $[S_2, S_2']$ are in the form of $W^i Z^j W'^{i'} Z'^{j'} R^r L$, where i and j correspond to the systematic and parity-check bits of the transition from state S_1 to S_2, and i' and j' correspond to the transition from S_1' to S_2', in one component code. For rate–$1/2$ RSC component codes, these variables evaluate to 0 or 1, accordingly. r is equal to 1 if both i and i' are equal to 1, and is equal to 0 otherwise, and L represents the length of the codeword and has power 1 on all transition labels.
- The state diagram starts from the state $[0, 0]_s$ and ends at the state $[0, 0]_f$, where the subscripts are added to distinguish between the starting and the finishing states. There is no transition to the state $[0, 0]_s$ and no transition from the state $[0, 0]_f$.

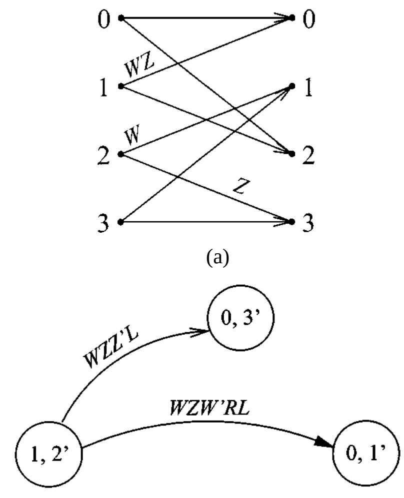

Fig. 3. (a) A trellis section for the $(5, 7)$ code. (b) Part of the super error state diagram.

TABLE I
NONASYMPTOTIC RESULTS CORRESPONDING TO TURBO CODES WITH $(5, 7)$ RSC CODES

E_b/N_0	$N = 100$		$N = 1000$	
(dB)	$\frac{\text{stdv}[B]}{\mathbf{E}[B]}$	$\log_{10} \mathbf{E}[B]$	$\frac{\text{stdv}[B]}{\mathbf{E}[B]}$	$\log_{10} \mathbf{E}[B]$
2	0.45	-2.60	0.44	-4.23
3	0.61	-3.65	0.55	-4.96
4	1.02	-4.50	0.78	-5.17
5	1.36	-5.43	1.10	-6.28

Fig. 3 shows the trellis diagram and part of the super state diagram corresponding to a $(5, 7)$ RSC code, where the brackets in denoting the states are omitted.

The transfer function of this state diagram enumerates the error events of the super trellis corresponding to the state diagram. Each error event is characterized by the number of 1's in the systematic and parity-check bits of each codeword, the number of their overlapping

[1] If different component codes are employed, we need to construct two super state diagrams, as each state diagram enumerates codeword pairs corresponding to one component code.

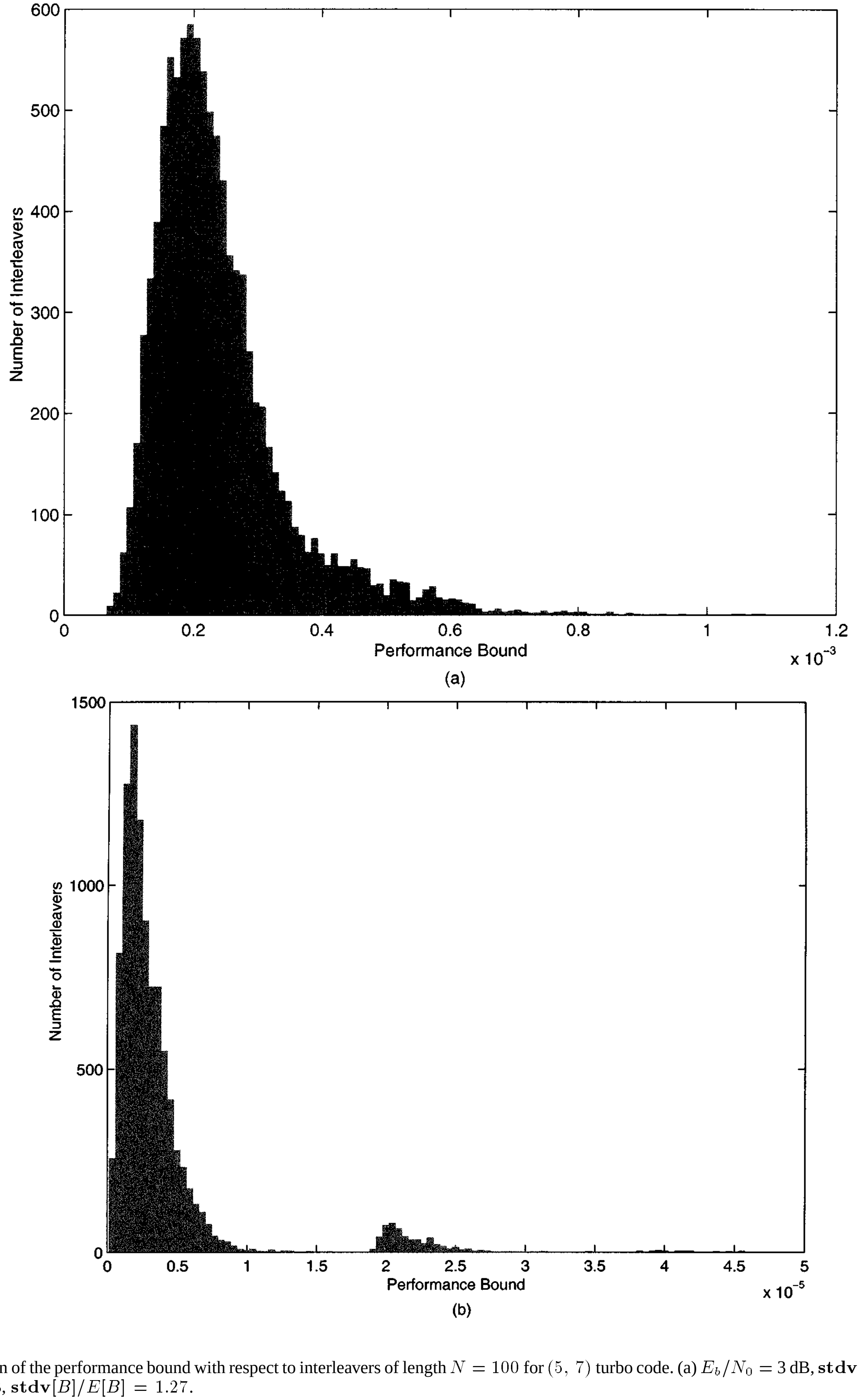

Fig. 4. Distribution of the performance bound with respect to interleavers of length $N = 100$ for $(5,\ 7)$ turbo code. (a) $E_b/N_0 = 3$ dB, $\mathbf{stdv}[B]/\boldsymbol{E}[B] = 0.45$. (b) $E_b/N_0 = 5$ dB, $\mathbf{stdv}[B]/E[B] = 1.27$.

1's in the systematic part, and the length of the event from the point where at least one of the codes diverges from the all-zero path to the point where both codes remerge to the all zero path. The transfer function is then used to find $q_r(i,\ j,\ i',\ j')$ for different $i,\ j,\ i',\ j'$, and r, with an approach analogous to that of finding the conditional weight enumerating function explained in [14].

Table I shows the coefficient of variation of the bound $B(\pi,\ E_b/N_0)$, $\mathbf{stdv}[B]/\boldsymbol{E}[B]$, for rate-1/3 turbo codes with

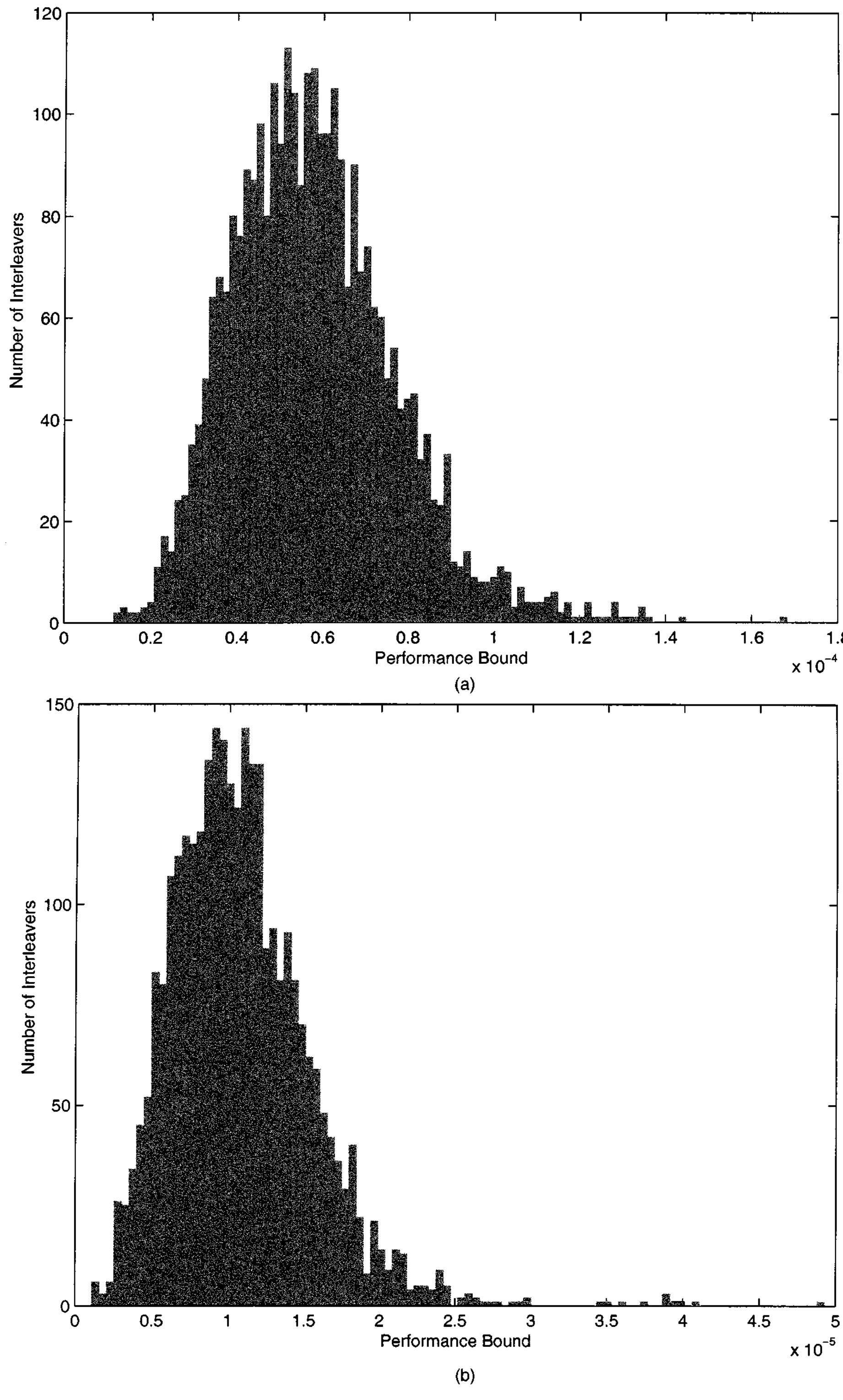

Fig. 5. Distribution of the performance bound with respect to interleavers of length $N = 1000$ for $(5, 7)$ turbo code. (a) $E_b/N_0 = 2$ dB, $\mathbf{stdv}[B]/\boldsymbol{E}[B] = 0.33$. (b) $E_b/N_0 = 3$ dB, $\mathbf{stdv}[B]/\boldsymbol{E}[B] = 0.43$.

identical $(5, 7)$ RSC component codes and interleaver lengths $N = 100$ and $N = 1000$.

In order to obtain an estimation of the distribution of the performance bound with respect to different interleavers, the union upper bound is calculated for the same codes over a number of randomly selected interleavers. Figs. 4 and 5 show the corresponding results. In these histograms, the x-axis represents the performance bound and the y-axis represents the number of interleavers which result in that performance. In calculating these results only input words of weights up to 6 (for $N = 100$) and 4 (for $N = 1000$), resulting in codewords with total weights up to 30, are considered. These limitations cause the simulation results to differ from the theoretical results of Table I.

As can be seen from Figs. 4 and 5 and Table I, the coefficient of variation, $\mathbf{stdv}[B]/\boldsymbol{E}[B]$, increases with E_b/N_0. This is due to the fact that

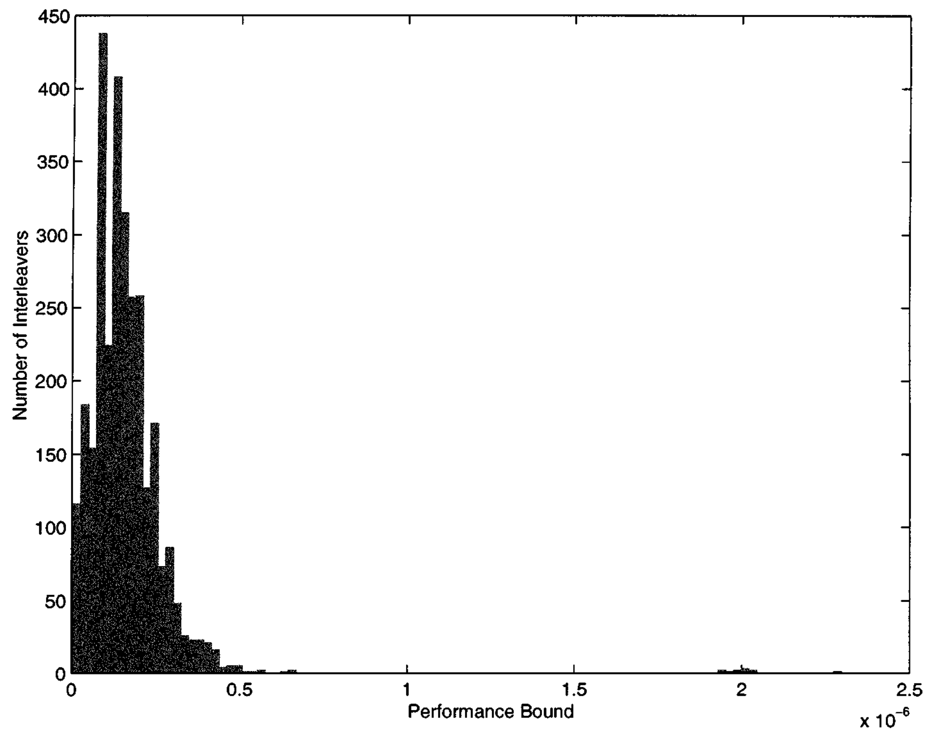

Fig. 5. *(Continued)* Distribution of the performance bound with respect to interleavers of length $N = 1000$ for $(5, 7)$ turbo code. (c) $E_B/N_0 = 5$ dB, $\mathbf{stdv}[B]/\boldsymbol{E}[B] = 0.90$.

TABLE II
ASYMPTOTIC RESULTS CORRESPONDING TO TURBO CODES WITH $(5, 7)$ AND $(7, 5)$ COMPONENT CODES

E_b/N_0 (dB)	$\mathbf{stdv}[B]/\mathbf{E}[B]$	
	(7,5)	(5,7)
2	0.11	0.33
3	0.21	0.41
4	0.27	0.48
5	0.34	0.55

different interleavers lead to turbo codes with different distance spectra. Although the majority of interleavers result in rather similar distance spectra, there is a low percentage of interleavers which leads to a large number of low-weight codewords. Since the E_b/N_0 impacts the BER in an exponential manner, as E_b/N_0 increases, the low-weight codewords become more dominant in the BER performance of the code. This causes the performance of the latter group of interleavers to have a larger deviation from the mean where the majority of interleavers operate. Note that, for $N = 100$ and $E_b/N_0 = 5$ dB, less than 7% of the randomly chosen interleavers result in error performance bounds higher than 10^{-5}; and for $N = 1000$ and $E_b/N_0 = 5$ dB, only 0.37% of the interleavers result in bounds higher than 0.7×10^{-6}.

B. Asymptotic Case

In this case, only codeword pairs which satisfy the conditions stated in Theorem 2 are enumerated. For this reason, in the super state diagram, only those paths corresponding to weight 2 in the systematic part of the codewords are taken into account and the error events are either completely overlapping $(r = 2)$ or have no overlapping bits. Table II shows the results of the asymptotic analysis.

The asymptotic results for the $(5, 7)$ turbo code follow the trend of Table I; the values of asymptotic $\mathbf{stdv}[B]/\boldsymbol{E}[B]$ increase with E_b/N_0 but remain lower than the corresponding values for finite interleaver lengths.

In order to compare turbo codes with primitive and those with nonprimitive feedback polynomials, the asymptotic results corresponding to the $(7, 5)$ turbo code are shown in Table II as well. As can be seen from the table, the nonprimitive feedback polynomial turbo code has a smaller coefficient of variation of the performance bound.

V. CONCLUSION

In this correspondence, the variance of the turbo-code performance bound over all possible interleavers is evaluated. It is shown that the coefficient of variation of the bound is asymptotically constant with the interleaver length. Furthermore, this coefficient is relatively small for lower E_b/N_0 values and increases as the E_b/N_0 value increases. Study of the analytical results and the distribution of the performance bound over a sample of randomly chosen interleavers shows that: a) as the interleaver length increases, the coefficient of variation decreases and b) as the E_b/N_0 increases, the distributions get more concentrated around the average performance bound and only a small percentage of interleavers result in high BERs, which cause the coefficient of variation to increase. These results support the statement made in [4], where for a turbo code of length $65,536$ it is stated that *most* pseudorandom interleavers result in the same multiplicity of the free-distance codewords. In addition, it can be observed that the performance of those interleavers which are not close to the performance of the majority of the pseudorandom interleavers, in fact, deviate quite significantly from the average bound.

Finally, the asymptotic results corresponding to turbo codes of memory 2 show that turbo codes with nonprimitive feedback polynomials have smaller standard deviations. This may suggest that the choice of the interleaver has a stronger effect on the performance of turbo codes with primitive than for those with nonprimitive feedback component codes.

ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers for their thorough and meticulous reviews which improved the quality and presentation of this correspondence.

REFERENCES

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. Int. Conf. Communications (ICC'93)*, Geneva, Switzerland, May 1993, pp. 1064–1070.
[2] P. Jung and M. Naßhan, "Performance evaluation of turbo-codes for short frame transmission systems," *Electron. Lett.*, vol. 30, no. 2, pp. 111–113, Jan. 1994.
[3] ——, "Dependence of the error performance of turbo-codes on the interleaver structure in short frame transmission systems," *Electron. Lett.*, vol. 30, no. 4, pp. 287–288, Feb. 1994.
[4] L. C. Perez, J. Seghers, and D. J. Costello, Jr., "A distance spectrum interpretation of turbo codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1698–1709, Nov. 1996.
[5] F. Daneshgaran and M. Mondin, "On design of interleaver for turbo codes," in *Proc. CISS'97*. Baltimore, MD, Mar. 1997, pp. 509–514.
[6] A. K. Khandani, "Dynamic generation of good turbo-code interleavers," Dept. Elec. and Comput. Eng., Univ. Waterloo, Waterloo, ON, Canada, Tech. Rep. UW-E&CE99-02, 1999.
[7] K. S. Andrews, C. Heegard, and D. Kozen, "Interleaver design methods for turbo codes," in *Proc. IEEE Int. Symp. Information Theory (ISIT'98}*, Cambridge, MA, Aug. 1998, p. 420.
[8] S. N. Crozier, "New high-spread high-distance interleavers for turbo codes," in *Proc. 20th B. Symp. Communications*, Kingston, ON, Canada, May 28–31, 2000, pp. 3–7.
[9] A. S. Barbulescu and S. S. Pietrobon, "Interleaver design for turbo-codes," *Electron. Lett.*, vol. 30, no. 25, pp. 2107–2108, Dec. 1994.
[10] R. Garello, G. Montorsi, S. Benedetto, and G. Cancellieri, "Interleaver properties and their application to the trellis complexity analysis of turbo codes," *IEEE Trans. Commun.*, vol. 49, pp. 793–807, May 2001.
[11] J. Hokfelt, O. Edfors, and T. Maseng, "A turbo code interleaver design criterion based on the performance of iterative decoding," *IEEE Commun. Lett.*, vol. 5, pp. 52–54, Feb. 2001.
[12] A. K. Khandani, "Group structure of turbo-codes," *Electron. Lett.*, vol. 34, no. 2, pp. 168–169, Jan. 1998.
[13] O. Y. Takeshita and D. J. Costello, Jr., "New classes of algebraic interleavers for turbo-codes," in *Proc. IEEE Int. Symp. Information Theory (ISIT'98)*, Cambridge, MA, Aug. 1998, p. 419.
[14] S. Benedetto and G. Montorsi, "Unveiling turbo codes: Some results on parallel concatenated coding schemes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 409–428, Mar. 1996.
[15] ——, "Design of parallel concatenated convolutional codes," *IEEE Trans. Commun.*, vol. 44, pp. 591–600, May 1996.

On the Asymptotic Eigenvalue Distribution of Concatenated Vector-Valued Fading Channels

Ralf R. Müller, *Associate Member, IEEE*

***Abstract*—The linear vector-valued channel $\boldsymbol{x} \mapsto \prod_n \boldsymbol{M}_n\boldsymbol{x} + \boldsymbol{z}$ with $\boldsymbol{z}$ and $\boldsymbol{M}_n$ denoting additive white Gaussian noise and independent random matrices, respectively, is analyzed in the asymptotic regime as the dimensions of the matrices and vectors involved become large. The asymptotic eigenvalue distribution of the channel's covariance matrix is given in terms of an implicit equation for its Stieltjes transform as well as an explicit expression for its moments. Additionally, almost all eigenvalues are shown to converge toward zero as the number of factors grows over all bounds. This effect cumulates the total energy in a vanishing number of dimensions. The channel model addressed generalizes the model introduced in [1] for communication via large antenna arrays to N-fold scattering per propagation path. As a byproduct, the multiplicative free convolution is shown to extend to a certain class of asymptotically large non-Gaussian random covariance matrices.**

***Index Terms*—Antenna arrays, Catalan numbers, channel models, fading channels, multiplicative free convolution, random matrices, S-transform, Stieltjes transform.**

I. INTRODUCTION

Consider a communication channel with K_0 transmitting and K_N receiving antennas grouped into a transmitter and a receiver array, respectively. Let there be $N - 1$ clusters of scatterers each with K_n, $1 \le n \le N - 1$, scattering objects. Assume that the vector-valued transmitted signal propagates from the transmitter array to the first cluster of scatterers, from the first to the second cluster, and so on, until it is received from the $(N - 1)$st cluster by the receiver array. Such a channel model is discussed and physical motivation is given in [2, Sec. 3]. Indoor propagation between different floors, for instance, may serve as an environment where multifold scattering can be typical, cf. [3, Sec. 13.4.1]. Some data of recent measurements can be found at [4].

The communication link outlined above is a linear vector channel that is canonically described by a channel matrix

$$\boldsymbol{H}_N = \boldsymbol{M}_N\boldsymbol{M}_{N-1}\cdots\boldsymbol{M}_2\boldsymbol{M}_1 \triangleq \prod_{n=1}^{N} \boldsymbol{M}_n \tag{1}$$

where the matrices $\boldsymbol{M}_1$, $\boldsymbol{M}_{1<n<N}$, and $\boldsymbol{M}_N$ denote the subchannels from the transmitter array to the first cluster of scatterers, from the $(n-1)$st cluster of scatterers to the nth cluster, and from the $(N-1)$st cluster to the receiving array, respectively. This means that $\boldsymbol{M}_n$ is of size $K_n \times K_{n-1}$. Assuming distortion by additive white Gaussian noise $\boldsymbol{z}$, the complete channel is given by

$$\boldsymbol{y} = \boldsymbol{H}_N\boldsymbol{x} + \boldsymbol{z} \tag{2}$$

with $\boldsymbol{x}$ and $\boldsymbol{y}$ denoting the vectors of transmitted and received signals, respectively.

Manuscript received November 22, 2000; revised February 1, 2002. The material in this correspondence was presented in part at the IEEE International Symposium on Information Theory, Washington, DC, June 2001.

The author is with Forschungszentrum Telekommunikation Wien (ftw.), A-1220 Vienna, Austria (e-mail: mueller@ftw.at).

Communicated by G. Caire, Associate Editor for Communications.
Publisher Item Identifier S 0018-9448(02)05335-X.