Home

Combined Turbo Coding and Precoding for Wireless Communication Channels†

Ali Ghrayeb
ECE Dept., Concordia University
Montreal, QC, Canada
aghrayeb@ece.concordia.ca

Mohamed El-Tarhuni
EE Dept., American University of Sharjah
Sharjah, United Arab Emirates
mtarhuni@aus.ac.ae

***Abstract*—In this paper, we study a serial concatenation scheme comprising a high-rate convolutional code and a precoded intersymbol interference (ISI) channel. We devise techniques that lead to the joint optimization of precoder and interleaver for a fixed convolutional code and precoded ISI channel, where optimality is in the sense of achieving the lowest error floor. We extend these design techniques to time varying channels, such as the case in mobile radio communications, using statistical models for these channels. We show that the level of the error floor can be significantly lowered by selecting the proper precoder and interleaver, while maintaining the same level of system complexity. We present analytical bounds and simulation results that support our approach.**

I. Introduction

THE DISCOVERY of turbo codes by Berrou *et al.* in 1993 [1] marked one of the most prominent breakthroughs in coding theory since 1948 when Shannon first introduced his theories of channel coding. A standard turbo encoder consists of two parallel concatenated convolutional codes separated by a random interleaver and is decoded using two iterative decoders that implement the *a posteriori* probability (APP) algorithm. Since their discovery, turbo codes have enjoyed a tremendous attention from the communication community where a considerable work has been done on their applications to a wide range of communication systems. In all of these applications, turbo codes have been shown to achieve astonishing performance improvement relative to previously existing coding schemes.

Benedetto *et al.* [2] proposed a new concatenation scheme that involves employing two convolutional codes concatenated in a serial fashion through a random interleaver, and is decoded using two iterative APP decoders. The performance of this scheme was shown to be comparable to that of parallel concatenation. In some cases, it was shown that serial concatenation was superior as its performance does not exhibit an error floor at low bit error rates, unlike that of the parallel concatenation schemes. The channel model assumed in most of these studies was the additive white Gaussian noise (AWGN) channel. This assumption is rather simplistic because in most applications, the channels are bandwidth limited, which gives rise to intersymbol interference (ISI).

† The work of A. Ghrayeb was in part supported by the Concordia University Start-up Grant UEN325.

To this end, considerable work has been done recently on the application of turbo codes to partial response (PR) channels [3]-[7]. PR channels can be regarded as a sub-class of intersymbol interference (ISI) channels and, hence, results for the PR case may be extended to ISI channels. Both parallel concatenated codes (PCC) and serial concatenated codes (SCC) have been studied. In the PCC system, the outer code comprises two parallel concatenated convolutional codes, and the inner code is the precoded ISI channel. The precoded ISI channel can be thought of as a rate-1 recursive convolutional code followed by the discrete-time equivalent model of the ISI channel. The SCC system is similar to the PCC system except that its outer code is just a single convolutional code. In both cases, the outer and inner codes are separated by a random interleaver.

The bit error rate performance of both systems have been investigated in recent works, where significant coding gains have been demonstrated. Lately, more attention has been given to the SCC system as it is less complex, with performance comparable to that of the PCC system. However, one of the weaknesses of the SCC system is that, in most of the cases, the bit error rate curves tend to hit an error floor somewhere near a bit error rate of 10^{-6} [4]. For a given outer convolutional code, the level of error floors have been found to depend greatly on the size of the interleaver and the choice of precoder [6], [7]. In many applications, increasing the size of the interleaver is not acceptable since it increases the decoding latency. This steers us toward the design of precoders in conjunction with interleavers.

In this paper, we discuss analytical techniques for designing optimal precoders for fixed ISI channels in SCC schemes, where optimality is in the sense of achieving the lowest error rate floor. We extend these design techniques to time varying channels using statistical models for such channels. We also demonstrate that the so-called $S-$random interleaver [8] may not be the best choice since it usually fails to break the "worst" error patterns that dominate the performance in the floor region, unlike what is usually assumed in the literature [6]-[8].

The remainder of the paper is organized as follows. In the next section, we discuss the system model under consideration. In Section III, we discuss precoder design techniques for fixed ISI channels, and demonstrate the effect of the interleaver on performance. We extend these precoder design techniques to time varying channel in Section IV. In Section V, we present semi-analytical and simulation results. Finally, we conclude the paper in Section VI.

II. System Model

The system under consideration is depicted in Fig.1. Observe in the figure the presence of the convolutional encoder puncturer, used to achieve code rates higher than $1/2$. Observe also in the figure the presence of a binary precoder. The precoder has been inserted in accordance with the design rules of [2]. (The channel is modeled as non-recursive in this work: while its discrete-time impulse response may be infinite in length, the response values beyond some memory size m are assumed to be negligible.) Most precoders of the form given in the figure are effective, but our goal is to select the optimal precoder, where optimality is in the sense of achieving the lowest error rate floor.

The ISI channel in the figure can be fixed or time varying, depending on the application. The receiver front end APP detector is matched to the combined trellis of the precoder and ISI channel. The outer decoder is matched to the outer recursive systematic convolutional (RSC) encoder. Both the detector and decoder work cooperatively and iteratively for a number of iterations before a final decision is made on the transmitted information sequence.

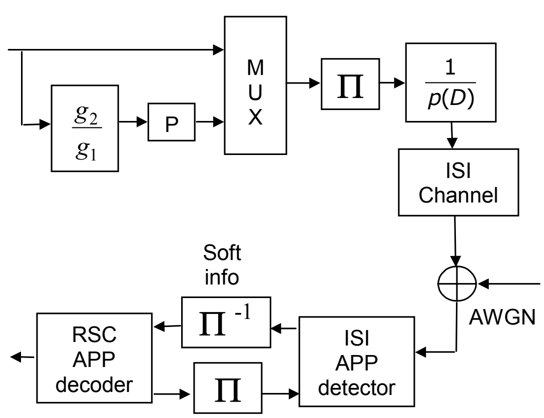

Fig. 1. Model for the SCC encoding and iterative deccoding on the precoded ISI channel. [Π indicates a permuter and P indicates a puncturer.]

III. Precoding for Fixed ISI Channels

A. Precoded ISI Channel Viewed as a Trellis Code

The ISI channel can be viewed as a trellis encoder, that is, as a binary convolutional code followed by a memoryless mapper. The same is true when the channel is combined with a precoder. The reason being is that the precoder is modeled by a finite state machine (FSM) and so is the ISI channel. Therefore, the combination of the precoder and ISI channel can also be modeled by a FSM.

A number of techniques exist in the literature for analytically evaluating the performance of trellis codes [9]. Most of these techniques usually start with finding the error state diagram of the code, and then using this to compute the distance enumerator function (DEF). From the DEF, one can obtain all of the necessary information to evaluate the code performance, including the minimum Euclidean distance and nearest-neighbor multiplicity. Obviously, the computational complexity of such techniques is directly related to the number of states in the error state diagram, which also greatly depends on the type of code employed. For example, the error state diagram for non-linear codes such as trellis codes in the most general case has M^2 states and its branch labels are matrices of size $M^2 \times M^2$, where M is the number of encoder trellis states.

Biglieri, *et al.* [9] show that if the trellis code is based on linear convolutional codes of rate $k/(k+r)$ followed by a memoryless mapper, the error state diagram involves only M states, but the branch labels are matrices of size $M \times M$. Thus all precoded ISI channels under consideration are included in this class of trellis codes.

B. Distance Enumerators for Precoded ISI Channels

Let us assume that the ISI channel has m taps. The polynomial representation of this channel will be of the form $h(D) = h_0 + h_1 D + h_2 D^2 + \cdots + h_m D^m$ where $h_i \in \mathbb{R}$. To keep complexity down, we shall limit our analysis to those precoders whose memory size is less than or equal to the memory size of the ISI channel. Therefore, the polynomial representation of the precoder will be of the form $p(D) = p_0 + p_1 D + p_2 D^2 + \cdots + p_m D^m$ where $p_i \in \{0, 1\}$. Consequently, the number of precoders to be considered is $2^m - 1$. This excludes the case when $p_0 = p_1 = \cdots = p_m = 0$, i.e., of no precoder, because we require the inner code to be recursive per the design rules for serial concatenated codes [2]. Note that the number of states of the combined trellis is $M = 2^m$ (assuming binary modulation.) The error state diagram in this case will consist of M states with matrix branch labels of size $M \times M$.

As a simple example, consider an ISI channel with discrete-time impulse response $h(D) = 0.6652 + 0.2447D + 0.09D^2$. Following the discussion above, the error state diagram for this precoded ISI channel with precoder polynomial $p(D) = 1 + D + D^2$ can be obtained and is shown in Fig. 2. (Note that there are three precoders to be considered for this ISI channel.) The structure of the error state diagram is determined by the linear convolutional code, and differs from it only in the branch labels. The second branch label indicates the bit error corresponding to the error state diagram transition. In the first branch label, L

Top

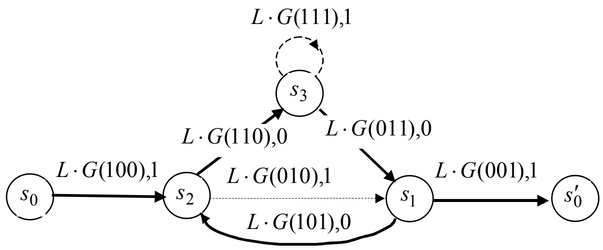

Fig. 2. Error state diagram for the precoded ISI channel (with $p(D) = 1 + D + D^2$).

accounts for transitions between states, and $G(\mathbf{e}_i)$ is a matrix of size 4×4 whose $(p,q)^{th}$ entry is defined as

$$\begin{aligned}[G(\mathbf{e}_i)]_{p,q} &= \frac{1}{V}\sum_{p\to q} Z^{\|f(\mathbf{x}_{p\to q})-f(\mathbf{x}_{p\to q}\oplus\mathbf{e}_i)\|^2} \\ &= \frac{1}{2}Z^{\|f(\mathbf{x}_{p\to q})-f(\mathbf{x}_{p\to q}\oplus\mathbf{e}_i)\|^2} \qquad (1)\end{aligned}$$

where $\mathbf{e}_i$ is a three-bit error vector whose decimal value is i, V is the source alphabet size, Z is an indeterminate, $f(\cdot)$ is the mapping function, and $\mathbf{x}_{p\to q}$ is the binary vector of length 3 generated by the transition from state p to state q. The sum in the first line of (1) accounts for the possible parallel transitions between state p and state q. (In this case, there is only one transition between any given two states.) The $(p,q)^{th}$ entry of $G(\mathbf{e}_i)$ is zero if there is no transition between state p and state q.

The error events of interest in this study are *weight-two error events* as they are the events that dominate the performance in the floor region of the error rate curves [6]. A weight-two error event is simply an error event in the precoded ISI channel trellis that corresponds to paths whose input sequences differ in exactly two positions, which results in the following situation. When the input to the outer convolutional encoder is of the form $D^{j'}(1+D^{n'})$, its corresponding parity output will be of the form $D^{j'}(1+D^{n'})g_2(D)/g_1(D)$. In the case when $1+D^n$ is divisible by $g_1(D)$, then the parity sequence $D^{j'}(1+D^{n'})g_2(D)/g_1(D)$ will have a relatively low Hamming weight. After puncturing this sequence to achieve a high code rate (e.g., 16/17), the punctured parity sequence is occasionally the all-zeros sequence. Thus, the codeword that corresponds to the input $D^{j'}(1+D^{n'})$ will have weight two. The interleaver then permutes this codeword and produces another word of the form $D^j(1+D^n)$, also of weight two. The words of this form that dominate the performance in the floor region are the ones for which $1+D^n$ is divisible by $p(D)$; this represents the minimum Hamming distance between all pairs of words in the linear system comprised of the convolutional encoder, interleaver, and precoder. This often leads to channel sequences (output of $h(D)$) that are at the minimum Euclidean distance.

In accordance with the foregoing discussion, we are primarily interested in a *conditional distance enumerator*, where conditioning is on weight-two inputs. The portion of the error state diagram of Fig. 2 which corresponds to weight-two inputs is indicated by solid lines in the figure. The conditional distance enumerator for this particular portion is computed from Fig. 2 to be

$$\mathbf{G} = L^4 G_2 G_6 G_4 G_1 \left[I_4 - L^3 G_2 G_4 G_5\right]^{-1} \qquad (2)$$

where I_4 is an 4×4 identity matrix and $G(\mathbf{e}_i)$ is defined in (1). The $(p,q)^{th}$ entry of the matrix $\mathbf{G}$ enumerates the squared Euclidean distances involved in the transition from state p to state q in l steps, where l is the exponent of L in that term. The conditional distance enumerator of the error state diagram, which we denote by $T_2(D,L)$, can be found by adding all entries of $\mathbf{G}$ as follows

$$\begin{aligned}T_2(D,L) &= \frac{1}{4}\mathbf{1}^T\mathbf{G}\mathbf{1} \\ &= \frac{L^4\left(D^{2.61}+D^{5.53}\right)}{2-L^3D^{3.33}-L^3D^{4.29}} \\ &= \frac{1}{2}L^4D^{2.61}+\frac{1}{2}L^4D^{5.53}+\cdots \qquad (3)\end{aligned}$$

where $\mathbf{1}$ is an 4×1 vector of ones. The terms of $T_2(D,L)$ are of the form $b_1L^{b_2}D^{b_3}$, where b_1 is the average number of weight-2 error events of length b_2 that result in a squared Euclidean distance b_3, where the average is over all possible trellis paths. From the last line of (3), one may observe the following. On average, half of the error events of length 4 result is a squared Euclidean distance $d_E^2 = 2.61$ (the exponent of D in the first term), and the other half result is a squared Euclidean distance of $d_E^2 = 5.53$. Therefore, the minimum squared Euclidean distance for this channel is $d_{E,\min}^2 = 2.61$ and it results from error events of the form $D^t\left(1+D^3\right)$ for integer t. Other weight-2 error events that contribute to the transfer function are of the form $D^t\left(1+D^{3k}\right)$ for $k=1,2,3,\ldots$.

When the precoder $p(D) = 1+D^2$ is used, following the same steps mentioned above, one can easily found its conditional distance enumerator to be

$$T_2(D,L) = L^3D^{2.04}+\frac{1}{2}L^5D^{3.60}+\frac{1}{2}L^5D^{4.56}+\cdots \qquad (4)$$

We observe from (4) that the minimum squared Euclidean distance is $d_{E,\min}^2 = 2.04$ and is caused by error events of the form $D^t\left(1+D^2\right)$. Moreover, the weight-2 error events that contribute to the transfer function are of the form $D^t\left(1+D^{2k}\right)$ for $k=1,2,3,\ldots$. As for the precoder $p(D) = 1+D$, its conditional distance enumerator is found to be

$$T_2(D,L) = L^3D^{2.04}+\frac{1}{2}L^4D^{2.61}+\frac{1}{2}L^4D^{5.56}+\cdots \qquad (5)$$

Top

Similarly, the minimum squared Euclidean distance for this case is $d^2_{E,\min} = 2.04$ and is caused by error events of the form $D^t(1+D)$. Also, it can be shown that the second smallest Euclidean distance is $d^2_E = 2.61$ and is caused by error events of the form $D^t\left(1+D^2\right)$. Other weight-2 error events that contribute to the transfer function are of the form $D^t\left(1+D^{k+1}\right)$ for $k = 1, 2, 3, \ldots$.[1]

By comparing the above three cases, one may observe the following. The minimum squared Euclidean distance for each case can be obtained from the first term of the conditional transfer function (after expansion); the squared distance increases as the length of the weight-2 error event increases; only particular weight-2 error events contribute to the transfer function; and finally, the precoder that has the largest minimum squared Euclidean distance is $p(D) = 1+D+D^2$, and, thus, is expected to yield the lowest error floor for a given outer convolutional code and for this particular ISI channel.

C. Joint Design of Precoder and Interleaver

It is well understood that the parameter that dominates the performance at relatively high SNR is the minimum squared Euclidean distance since it dictates the slope of the performance curves. Therefore, to improve the performance further, every effort should be made to increase the minimum squared distance. By taking a closer look at the above transfer functions, as mentioned before, the squared Euclidean distance increases as the length of the weight-2 error events increases. For example, for precoder $p(D) = 1+D+D^2$, the squared distance that corresponds to the error event $D^t\left(1+D^3\right)$ is $d^2_{E,\min} = 2.61$, which is the absolute minimum squared distance. Note that the length of this error event is 4. When the length of the error event increases to 7 (from the second term of the transfer function), the squared distance increases to 4.73. It increases further to 6.86 when the length of the error event is 10 (from the third term of the transfer function.) Therefore, to increase the overall minimum squared distance, the interleaver has to be designed such that weight-2 error events of short lengths should be eliminated. The consequence of this is that the first few terms of the transfer function will be eliminated.

It has been recently a common understanding that the $S-$random interleaver is capable of eliminating all terms that correspond to error events of length less than S [6]-[8]. As we will show later, there is no guarantee that such an interleaver will eliminate these error events. It is true that this interleaver will map each input location pair l_1 and l_2 with $|l_2 - l_1| < S$ to an output location pair $P(l_1)$ and $P(l_2)$ with $|P(l_2) - P(l_1)| \geq S$. However, it sometimes maps input location pairs that are separated by more than S to output location pairs separated by less than S. Such error events are the ones that dominate the performance in the floor region, as we will demonstrate shortly. Therefore, a code-matched interleaver should be used instead.

To show the impact of the interleaver on the choice of the precoder, let us assume that there exists an interleaver that eliminates error events of length, say, 14 or less. With this, for precoder $p(D) = 1+D+D^2$, the next (uneliminated) immediate error event would be of length 16 with a squared distance of 11.11 and multiplicity of $\frac{1}{512}$; of length 15 with a squared distance of 14.07 and multiplicity of $\frac{1}{4096}$ for precoder $p(D) = 1+D$; and of length 15 with a squared distance of 11.42 and multiplicity of $\frac{1}{64}$ for precoder $p(D) = 1+D^2$. Accordingly, the optimal precoder in this case is $p(D) = 1+D$. However, when an S-random interleaver (or any other randomly generated interleaver), the optimal precoder is $p(D) = 1+D+D^2$.

IV. Precoding for Time Varying ISI Channels

The above design techniques can be extended in a straightforward manner to arbitrary *fixed* ISI channels. However, when the channel taps vary with time such as the case in mobile radio communications, it would be difficult to design the optimal interleaver for every state of the channel. The difficulty stems from the fact that for every realization of the channel taps, a search over all possible precoders is necessary to determine the optimal precoder. This is obviously prohibitively complex to implement.

As an alternative, we use statistical models of the time varying channels, namely, the power delay profile (PDP). The PDP of a channel reflects the distribution of the power among different paths. Most channels have an exponentially PDP where the power of the path is exponentially related to the path delay. Another important channel model is the uniform PDP, where the power is equally distributed among the paths. In this paper we focus on the exponential PDP channel model. We expect similar results to hold for other statistical models.

Representing the time varying channel as such, we follow the design techniques outlined above for fixed ISI channels to select the optimal precoder based on the PDP of the channel. Once the optimal precoder is found, we use it for the time varying channel. Obviously, the selected precoder is considered suboptimal since it was optimized for the PDP of the channel and not for every realization of the channel coefficients. However, as we will demonstrate later through simulations, the selected precoder seems to be optimal or "near" optimal for the time varying channel.

V. Simulation Results

The system simulated for the AWGN case is shown in Fig. 1. It uses a rate $16/17$ outer convolutional code punctured from a rate $1/2$ code with generator polynomials $(g_1, g_2) = (23, 31)_{oct}$, where g_1 is the feedback polynomial and g_2 is the feedforward polynomial. The interleaver employed is an S-random interleaver of size $N = 544$, with $S = 15$. As for

[1] We remark that one can easily obtain these results from the error state diagram corresponding to that particular precoder.

Top

the inner code, we consider a precoded 3-path fading channel whose PDP is modeled by the polynomial $f(D) = 0.6652 + 0.2447D + 0.09D^2$, which is the same ISI channel we did the analysis for in Section III.

Fig. 3 presents the P_b performance for the above channel for three precoders (for 5 decoder iterations). The precoders employed are $p_1(D) = 1 + D + D^2$, $p_2(D) = 1 + D$, and $p_3(D) = 1 + D^2$. If it was true that the S-random interleaver would eliminate all weight-2 error of length less than S, then precoder $p_3(D)$ would be optimal. But since the interleaver fails to do so, the optimal interleaver is $p_1(D)$, suggesting that there are still error events of length 3 and 4 at the output of the interleaver. We also observe from the figure that the performance curves corresponding to precoders $p_2(D) = 1 + D$ and $p_3(D) = 1+D^2$ have the same slope, suggesting that they have the same minimum squared Euclidean distance, which agrees with the theoretical results.

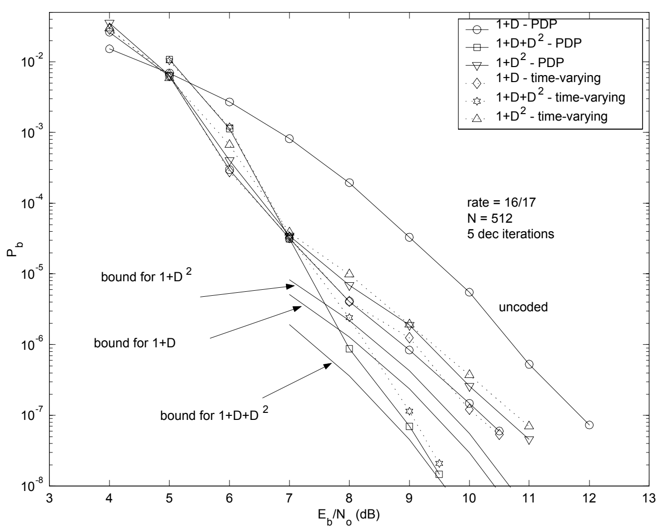

Fig. 3. Performance of rate $16/17$ SCC on a precoded Rayleigh fading channel (with PDP $f(D) = 0.6652 + 0.244D + 0.09D^2$) with various precoders.

We also included in Fig. 3 the performance bounds for all precoders. To obtain these bounds, we first generate a random information word of length $N = 512$. We use this information word to generate another information word that differs only in two locations. This is equivalent to generating two information words that are at a Hamming distance of 2. We feed separately both information words to the outer convolutional encoder, puncture the output of that encoder to achieve rate $16/17$, multiplex the information bits with the remaining parity bits, interleave using the S-random interleaver, send through the precoded ISI channel, and calculate the squared Euclidean distance between these two channel sequences. We do that for all possible weight-2 error patterns and over a large number of random information words. While doing that, we monitor the minimum squared Euclidean distance and the number of times it repeats (multiplicity). Then the performance can be approximated by

$$P_b \approx \frac{2N^*}{N} Q\left(\sqrt{\frac{r d^2_{E,\min}}{2N_0}}\right)$$

where $d^2_{E,\min}$ is the minimum squared Euclidean distance, N^* is the average multiplicity, and r is the code rate. N^* was found to be 1.964 for $p_1(D)$, 0.999 for $p_2(D)$ and 1.998 for $p_3(D)$. $d^2_{E,\min}$ for these cases are given in Section III. We observe that the performance corresponding to $p_2(D)$ is about twice better than that corresponding to $p_3(D)$ while both performances have the same slope. These theoretical results match the simulation results.

VI. Concluding Remarks

We have presented techniques that lead to the joint optimization of precoder and interleaver for a given convolutional code and fixed ISI channel. We have extended these design techniques to time varying channels where the design was based on statistical models for these channels. Simulation results suggested that when the optimal precoder and interleaver are selected based on the PDP of the channel, they are "near" optimal for the time varying channel. However, a rigorous mathematical justification is still required in this regard.

We have also demonstrated that the S-random interleaver may not be the best choice of interleaver since it usually fails to break the worst error patterns that dominate the performance in the floor region. This suggest that a code-matched interleaver is very necessary to achieve further improvement in the performance.

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo-codes," *Proc. 1993 Int. Conf. on Comm.,* pp. 1064-1070.

[2] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding," *IEEE Trans. Info. Theory*, vol. 44, pp. 909-926, May 1998.

[3] W. E. Ryan, "Performance of high rate turbo codes on a PR4-equalized magnetic recording channel," *Proc. 1998 Int. Conf. on Comm.,* vol. 2, pp. 947-951.

[4] W. Ryan, "Optimal code rates for concatenated codes on a PR4-equalized magnetic recording channel," *Proc. Conf. on Inf. Sciences and Systems*, pp. 432-437, Mar. 1999.

[5] T. Souvignier, A. Friedman, M. Öberg, P. Siegel, R. Swanson, and J. Wolf, "Turbo decoding for PR4: Parallel versus serial concatenation," *Proc. 1998 Int. Conf. on Comm,* vol. 3, pp. 1638-1642.

[6] W. E. Ryan, "Concatenated codes for class IV partial response channels," *IEEE Trans. Commun.*, vol. 47, pp. 445-454, Mar. 2001.

[7] A. Ghrayeb and W. Ryan "Concatenated code system design for storage channels," *IEEE J. on Selected Areas in Commun.,* vol. 19, no. 4, pp. 709-718, April 2001.

[8] D. Divsalar and F. Pollara, "Turbo codes for PCS Applications," *Proc. 1995 Int. Conf. on Comm.,* pp. 54-59.

[9] Y. Liu, I. Oka, and E. Biglieri, "Error probability for digital transmission over nonlinear channels with applications to TCM," *IEEE Trans. Info. Theory*, September 1990.

Top