GENERATING SERIES AND PERFORMANCE BOUNDS FOR CONVOLUTIONAL CODES - PART 2: TRANSMISSION ON BURST-ERROR CHANNELS

Cecilio Pimentel

Communications Research Group - CODEC
Department of Electronics and Systems, P.O. Box 7800
Federal University of Pernambuco
P.O. Box 7800 - 50711-970 - Recife - PE
E.mail: cecilio@npd.ufpe.br

ABSTRACT

We present an analytical method for evaluating the performance of a communication system that employs a convolutional code, a block interleaving with finite interleaving depth, a binary channel that exhibits statistical dependence in the occurrence of errors, and a decoder that implements the Viterbi algorithm with Hamming metric. The main idea is to apply combinatorial methods to derive a formula for bounds to first-event and bit error probabilities in terms of coefficients of a generating series. The method is used to investigate the tradeoff between coding parameters and interleaving depth to achieve a required performance.

1. INTRODUCTION

The error process encountered in many digital communication systems introduces distortion in the transmission process in such a way that errors occur in bursts. The channel model considered in this work is an additive binary burst channel modeled according to a finite state channel (FSC) [1] model. The model is described statistically by a probabilistic function of a Markov chain whose parameters are conveniently chosen to capture the bursty nature of errors symbols produced by several real channels. Examples of FSC models proposed in the literature include the Gilbert-Elliott channel [2] and the Fritchman channel [3].

A detailed calculation of performance bounds for maximum likelihood decoding of convolutional codes over memoryless channels is found in reference [4]. The bounds developed in [4] are commonly measured using the distance weight enumerator, also referred to as the *transfer function*, of the code. The information derived by the transfer function (Hamming weights of the input and output sequences of the convolutional code) cannot be used to measure the performance of convolutional codes over burst channels, because, in this case, the probability of an error event depends not only on the number of errors in the binary stream generated by the channel, but also on the error positions. In this paper, we propose a generalization for the conventional transfer function, and apply these results to obtain performance bounds for a specific convolutional code over *interleaved FSC models* (the cascade of block interleaver, FSC model, and block deinterleaver) with finite interleaved depth. Previous theoretical analyses of error correcting codes over FSC models have been mainly concentrated on block codes [6, 7, 8, 9, 10]. Results for convolutional codes over FSC models are obtained from computer simulation in [6, 11]. The technique presented here is an interesting alternative approach with respect to computer simulations.

We use a combinatorial approach to find the generating series that enumerates all error patters that produce error events in the decoding process. By defining appropriate indeterminates we are able to extract useful information from the resulting generating series for the evaluation of coded system performance. A linear mapping incorporates the model parameters into the generating series. The approach is not channel specific and is valid for general FSC models irrespective of the number of states and structure of the Markov chain. One motivation for this research is to develop analytical methods for analyzing interleaved communication systems where delay constraints limit the maximum value of the interleaving depth.

The remainder of this paper is organized as follows. Section 2 describes the communication system. A brief review of FSC models is contained in this section. In Section 3, we introduce the method which will enable the enumeration of error events. The rest of this section is dedicated to obtaining performance bounds for convolutional codes over FSC models from the generating series. Numerical results are presented for the special case of Gilbert-Elliott channels with known model parameters. Conclusions are summarized in Section 4.

We adopt throughout this paper the following notation. Given a matrix $\mathbf{A}$, the superscripts $\mathbf{A}^T$ and $\mathbf{A}^{-1}$ represent respectively the transpose, and the inverse of a matrix. The matrices $\mathbf{I}$ and $\mathbf{1}$ stand for the identity matrix and a column vector of ones, whose dimensionality is clear from the context. If s and z are commutative indeterminates, $[s^k z^n] T(s,z)$ denotes the coefficient of $s^k z^n$ in the formal power series $T(s,z)$. Let $R[[x]]$ be the ring of all formal power series in commuting indeterminate x with coefficients taken from the field of real numbers R, and let $R[x]$ be the set of all polynomials in x. $R \ll x_0, x_1 \gg$ is the ring of all power series in the non-commuting indeterminates x_0, x_1.

2. COMMUNICATION SYSTEM DESCRIPTION

Consider a convolutional encoder of rate $R_c = 1/n_0$ and constraint length K, with memory cells arranged as a serial shift register. The encoder state diagram is a labeled directed graph with $2^{(K-1)}$ vertices (states), and 2^K branches, each labeled with 1-bit input and n_0-bit output strings. Figure 1 shows a representation of the state diagram for an eight-state encoder of rate $R_c = 1/2$, $K = 4$, and generating polynomial (in octal) 15, 17.

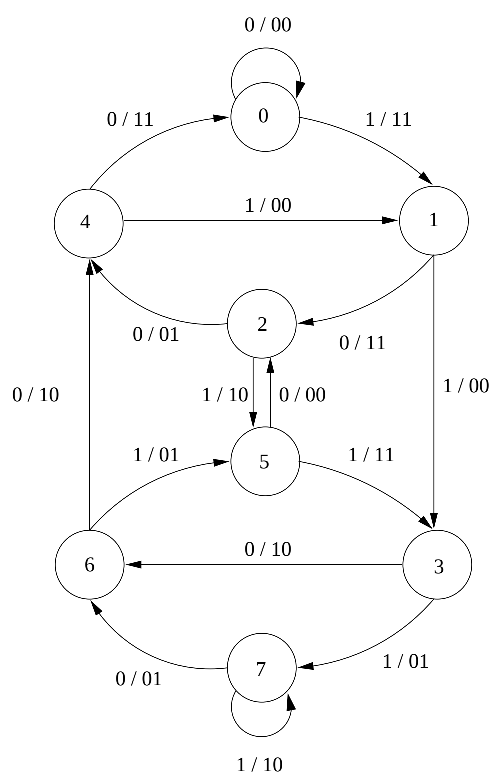

Figure 1: State diagram for a rate $R_c = 1/2$, constraint length $K = 4$, convolutional code.

The information sequence is encoded into a single codeword of unbounded length and transmitted across the interleaved FSC model. A block interleaver consists of an array of $n_0\, L_D$ columns and I_d (interleaving depth) rows. The output sequence of the interleaver is corrupted by an additive error sequence $\{e_k\}_{k=1}^{\infty}$ statistically distributed according to an FSC model. The error bit e_k is characterized by a probabilistic function of an N state Markov chain with transition probability matrix $\mathbf{P}$. Define two $N \times N$ matrices, $\mathbf{P}(0)$ and $\mathbf{P}(1)$, where the $(i,j)th$ entry of the matrix $\mathbf{P}(e_k)$, $e_k \in \{0,1\}$, is the probability that the output symbol is e_k when the chain makes a transition from state i to j. We are assuming that the distribution of the initial state is the stationary distribution $\mathbf{\Pi} = [\pi_0, \pi_1, \ldots, \pi_{N-1}]^T$. The probability of an error sequence of length n, $\mathbf{e}_n = e_1 \ldots e_n$, is expressed in a matrix form as:

$$P(\mathbf{e}_n) = \mathbf{\Pi}^T \left(\prod_{k=1}^{n} \mathbf{P}(e_k) \right) \mathbf{1}.$$

For example, the Gilbert-Elliott channel is a two-state Markov chain composed of a good state, state 0, where errors occur with small probability, and a bad state, state 1, where errors occur with higher probability. The transition probabilities of the Markov chain are Q and q, as shown in Fig. 2. When the chain is in the good state the error bit e_k is zero (correct) with probability $1-g$, or one (error) with probability g. Otherwise, when it is in the bad state, the error bit is zero with probability $1-b$, or one with probability b. The parameter $\mu = 1 - q - Q$ is defined in [12] as the memory of the Gilbert-Elliott channel. The matrices $\mathbf{P}(0)$, $\mathbf{P}(1)$, where $\mathbf{P}(0) + \mathbf{P}(1) = \mathbf{P}$, and $\mathbf{\Pi}$ for the Gilbert-Elliott channel are given by:

$$\mathbf{P}(0) = \left[\begin{array}{cc} (1-Q)\,(1-g) & Q\,(1-b) \\ q\,(1-g) & (1-q)\,(1-b) \end{array} \right]; \qquad (1)$$

$$\mathbf{P}(1) = \left[\begin{array}{cc} (1-Q)\,g & Q\,b \\ q\,g & (1-q)\,b \end{array} \right]; \qquad (2)$$

$$\mathbf{\Pi} = \left[\begin{array}{c} \pi_0 \\ \pi_1 \end{array} \right] = \left[\begin{array}{c} \frac{q}{q+Q} \\ \frac{Q}{q+Q} \end{array} \right]. \qquad (3)$$

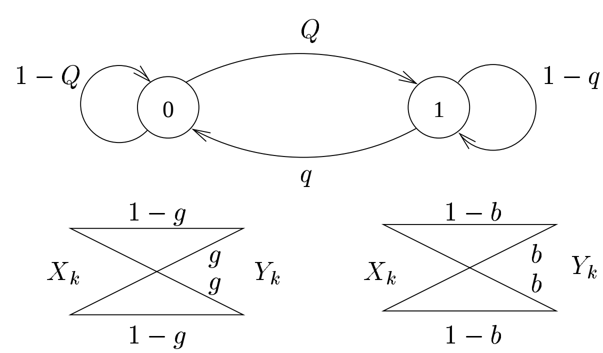

Figure 2: Gilbert-Elliott model for burst channels.

The deinterleaver restores the original order of the transmitted symbols. Notice that two consecutive received symbols in each row of the deinterleaver are corrupted by two error symbols separated exactly by I_d. The Viterbi decoding algorithm generates an estimate of the transmitted symbols using the Hamming metric. The parameter L_D is the decision depth of the Viterbi algorithm.

3. PERFORMANCE ANALYSIS

We start this section introducing the main concepts regarding the performance calculation of convolutional codes on memoryless channels. Later, we generalize these results to treat the case of burst channels.

Let $\mathcal{S}$ be the set of all incorrect paths, where the all-zero codeword was transmitted. Each path in $\mathcal{S}$ constitute an error event. The *first-event error probability*, P_{ev}, and the *bit error probability*, P_b, for memoryless channels are bounded above by:

$$\begin{aligned} P_{ev} &\leq \sum_{d=d_{free}}^{\infty} a_d\, P_d; \\ P_b &\leq \sum_{d=d_{free}}^{\infty} b_d\, P_d, \end{aligned} \tag{4}$$

where d_{free} is the minimum free distance of the code, a_d is the number of paths in $\mathcal{S}$ of Hamming weight d, b_d is the total number of nonzero information bits in all paths of Hamming weight d in $\mathcal{S}$, and P_d is the pairwise probability that a path in $\mathcal{S}$ (a wrong path) of Hamming weight d is chosen instead of the correct path. This probability depends on the metric used by the Viterbi algorithm, and on the channel parameters. For example, for a BSC channel with crossover probability p, and for a Viterbi decoder using Hamming distance as metric, a path with weight d is chosen over the all-zero path if $(d+1)/2$ or more transmission errors are made in these particular d positions. Therefore

$$P_d = \begin{cases} \displaystyle\sum_{k=(d+1)/2}^{d} \binom{d}{k} p^k (1-p)^{d-k}, & d \text{ odd}; \\ \frac{1}{2}\binom{d}{d/2} p^{d/2}(1-p)^{d-d/2} + \displaystyle\sum_{k=d/2+1}^{d} \binom{d}{k} p^k (1-p)^{d-k}, & d \text{ even}. \end{cases} \tag{5}$$

It is assumed in the second row of Equation (5) that $P_d = 1/2$ if exactly $d/2$ errors occur. The parameters a_d and b_d are commonly calculated using the transfer function $T(x,y)$ of the code. Let w_1 and w_2 be weight functions such that $w_1(\boldsymbol{\sigma})$ and $w_2(\boldsymbol{\sigma})$ is the number of 1's in the input and output (codeword) sequence, respectively, corresponding to an incorrect state sequence $\boldsymbol{\sigma}$. $T(x,y)$ is the generating series for the set $\mathcal{S}$ with respect to the weight function w_1 and w_2, that is:

$$T(x,y) = \sum_{\boldsymbol{\sigma}\in\mathcal{S}} x^{w_1(\boldsymbol{\sigma})}\, y^{w_2(\boldsymbol{\sigma})} \qquad \in R[x][[y]]. \tag{6}$$

For example, the generating series corresponding to a particular the state sequence $\boldsymbol{\sigma} = 01240$ is xy^7. We now turn to the computation of P_{ev} and P_b for convolutional codes over FSC models. In this case, the parameters a_d and b_d have no relevance to the performance calculation, since error sequences of Hamming weight d have different probability. First, we treat the non-interleaved case. To find a union bound expression for P_{ev} and P_b, a complete enumeration of all codewords in a non-commuting ring is needed, since the probability of each sequence produced by the channel depends on the non-commutative product of matrices $\mathbf{P}(0)$ and $\mathbf{P}(1)$. The new generating series, which is denoted by $T(y_0, y_1, x, y)$, records all the information about the sequence of 0's and 1's that constitute each codeword, and is defined next.

Let the non-commuting indeterminates y_0 and y_1 mark a bit equal to zero or one in a codeword, and let $w(i \to j)$, which is in $R < y_0, y_1 >$ (the set of polynomials in non-commuting indeterminates y_0 and y_1), mark the output string corresponding to the state transition from i to j. For example, from the state diagram of Figure 1, $w(0 \to 1) = y_1\, y_1$, $w(1 \to 3) = y_0\, y_0$. The generating series we are interested in finding is defined as:

$$T(y_0, y_1, x, y) = \sum_{\boldsymbol{\sigma}\in\mathcal{S}} \left(\prod_{k\geq 1} w(\sigma_k \to \sigma_{k+1}) \right) x^{w_1(\boldsymbol{\sigma})}\, y^{w_2(\boldsymbol{\sigma})}. \tag{7}$$

For example, the generating series corresponding to a state sequence $\boldsymbol{\sigma} = 01240$ is $y_1 y_1 y_1 y_1 y_0 y_1 y_1 y_1\, xy^7$. The information needed to calculate $T(y_0, y_1, x, y)$ is encoded into an adjacent matrix, denoted by $\mathbf{A}$, whose $(i,j)th$ entry is defined as follows:

$$a_{i,j} = w(i \to j)\, x^{w_1(i \to j)}\, y^{w_2(i \to j)},$$

where $w_1(i \to j)$ and $w_2(i \to j)$ are the Hamming weights of the input and output strings on the branch that connects the states i and j, respectively, for $0 \leq i, j \leq 2^{K-1} - 1$ (rows and columns of $\mathbf{A}$ are indexed by states). If states i and j are not joined on the state diagram, then $a_{i,j}$ is set to zero. For example, the adjacent matrix for the state diagram of Figure 1 becomes:

$$\mathbf{A} = \begin{bmatrix} b & dxy^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & dy^2 & ax & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & by & cxy & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & cy & bxy \\ dy^2 & ax & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a & dxy^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & cy & bxy & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & by & cxy \end{bmatrix},$$

where $a = y_0y_0$, $b = y_0y_1$, $c = y_1y_0$, $d = y_1y_1$. It is interesting to notice that any state sequence in $\mathcal{S}$ has the following structure: The first symbol is 0, the second is 1, and so on, until it reaches the state 2^{K-2}. The next state may be either 0 (the final state), or 1, when the process restarts all over again. Let $\mathcal{S}_2$ be the set of all state sequences that start in state 1 and reach state 2^{K-2} for the first time some time later. The main step to find an expression to $T(y_0, y_1, x, y)$ is to enumerate the set $\mathcal{S}_2$, yielding the following generating series $T_2(y_0, y_1, x, y)$:

$$T_2(y_0, y_1, x, y) = \sum_{\boldsymbol{\sigma} \in \mathcal{S}_2} \left(\prod_{k \geq 1} w(\sigma_k \to \sigma_{k+1}) \right) x^{w_1(\boldsymbol{\sigma})} \, y^{w_2(\boldsymbol{\sigma})}. \tag{8}$$

The desired generating series $T(y_0, y_1, x, y)$ is expressed in terms of $T_2(y_0, y_1, x, y)$ as shown below:

$$\begin{aligned} &T(y_0, y_1, x, y) = \\ &a_{0,1} T_2(y_0, y_1, x, y)[1 - a_{2^{K-2},1}\, T_2(y_0, y_1, x, y)]^{-1} a_{2^{K-2},0}. \end{aligned} \tag{9}$$

We have used the graph reduction technique proposed in [14] to calculate $T_2(y_0, y_1, x, y)$. The next example illustrates the calculation of $T_2(y_0, y_1, x, y)$ for the encoder described by the state diagram of Figure 1.

Example 1 *Figure 3 shows a reduced state diagram. The branch labels* h, c, d, e *and* f *are defined below:*

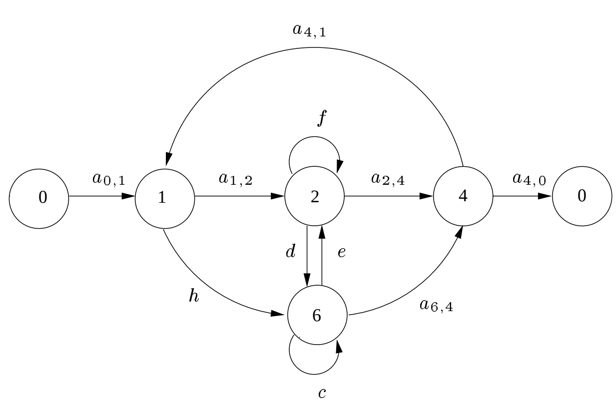

Figure 3: Reduced state diagram.

$$\begin{aligned} h &= a_{1,3}[a_{3,6} + a_{3,7}(1 - a_{7,7})^{-1} a_{7,6}]; \\ c &= a_{3,6} + a_{3,7}(1 - a_{7,7})^{-1} a_{7,6}; \\ d &= a_{2,5} a_{5,3}[a_{3,6} + a_{3,7}(1 - a_{7,7})^{-1} a_{7,6}]; \\ e &= a_{6,5} a_{5,2}; \\ f &= a_{2,5} a_{5,2}. \end{aligned}$$

The generating series for all paths that start from state 1 and reach state 4 for the first time is shown below:

$$\begin{aligned} &T_2(y_0, y_1, x, y) = a_{1,2}(1 - f)^{-1} \Psi (a_{2,4} + d(1 - c)^{-1}\, a_{6,4}) \\ &+ h\,(1 - c)^{-1}\, \Phi (a_{6,4} + e(1 - f)^{-1}\, a_{2,4}). \end{aligned} \tag{10}$$

where $\Psi = [1 - d(1-c)^{-1} e(1-f)^{-1}]^{-1}$ *and* $\Phi = [1 - e(1-f)^{-1} d(1-c)^{-1}]^{-1}$. *It is worth noting that all multiplications in* $T_2(y_0, y_1, x, y)$ *are non-commutative. The desired generating series* $T(y_0, y_1, x, y)$ *is obtained from (9):*

$$\begin{aligned} &T(y_0, y_1, x, y) = \\ &a_{0,1} T_2(y_0, y_1, x, y)[1 - a_{4,1}\, T_2(y_0, y_1, x, y)]^{-1} a_{4,0}. \end{aligned} \tag{11}$$

Our objective for the rest of this paper is to express P_b and P_{ev} in terms of the generating series $T(y_0, y_1, x, y)$ defined in (9). This goal is achieved by enumerating all error patterns that produce an error event, for each codeword in the set $\mathcal{S}$. This is the same union bound argument that led to Equation (4). The Viterbi decoder chooses the incorrect path instead of the all zero path if the channel produces more ones (errors) than zeros (correct) in those positions marked by y_1. Let the indeterminates x_0 and x_1 mark an error bit (produced by the channel) equal to 0 or 1, respectively. Let Δ_0 be the mapping defined as:

$$\begin{aligned} \Delta_0 : \quad & R \ll y_0, y_1 \gg \longrightarrow R[w] \ll x_0, x_1 \gg : \\ & y_0 \mapsto x_0 + x_1; \\ & y_1 \mapsto x_0 + w\, x_1, \end{aligned} \tag{12}$$

extended as a homomorphism to the whole of the ring, where w marks the number of errors produced by the channel in the codeword positions marked by y_1. Define

$$U(x_0, x_1, w, x, y) \;=\; \Delta_0[T(y_0, y_1, x, y)]. \tag{13}$$

Notice that acting on $T(y_0, y_1, x, y)$ with the mapping Δ_0 enumerates all possible error patters that corrupts each codeword in $\mathcal{S}$.

After having enumerated the sets of all error patterns of interest, the probability of these sets follows directly from (13) and (2). Consider a general FSC model defined by the matrices $\mathbf{P}(0)$, $\mathbf{P}(1)$ and $\mathbf{\Pi}$. Therefore, by defining a mapping Δ that replaces x_i by the corresponding matrix $\mathbf{P}(i)$ and replaces 1 by the identity matrix $\mathbf{I}$ [13], the generating series for the probability of error patterns is expressed as:

$$\mathcal{T}(w, x, y) = \mathbf{\Pi}^T\, \Delta[\, U(x_0, x_1, w, x, y)\,]\, \mathbf{1}. \tag{14}$$

$\mathcal{T}(w, x, y)$ is a ratio of two polynomials in w, x, and y, and its series expansion is easily performed using a symbolic manipulation program. The probability of all possible error sequences that produces an error event is enumerated by:

$$\begin{aligned} \mathcal{T}_1(x) &= \sum_{d \text{ even}} [y^d w^{d/2}]\, \mathcal{T}(w,x,y); \\ \mathcal{T}_2(x) &= \sum_{d \geq d_{free}} [y^d] \sum_{j \geq \lceil (d+1)/2 \rceil} [w^j]\, \mathcal{T}(w,x,y). \end{aligned} \tag{15}$$

Recall that y records the number of 1's in a codeword and w records the number of errors in the positions marked by y_1. Therefore, we can conclude that $\mathcal{T}_2(x)$ enumerates the probability of all error patterns whose number of 1's is greater than the number of zeros in the positions marked by y_1 for each codeword in $\mathcal{S}$. On the other hand, $\mathcal{T}_1(x)$ enumerates the probability of all error patterns with equal number of 0's and 1's in the positions marked by y_1. Finally, we are able to give new expressions to upper bounds on P_{ev} and P_b for convolutional codes over FSC models. The expressions are given by:

$$P_{ev} \leq \frac{1}{2}\mathcal{T}_1(1) \,+\, \mathcal{T}_2(1); \tag{16}$$

$$P_b \leq \frac{1}{2}\left\{\frac{d\mathcal{T}_1(x)}{dx}\right\}_{x=1} + \left\{\frac{d\mathcal{T}_2(x)}{dx}\right\}_{x=1}. \tag{17}$$

Example 2 *This numerical example illustrates the successive action on $T(y_0, y_1, x, y)$ with the mappings Δ_0 and Δ. To simplify the calculations, we consider a four-state convolutional code and a simplified Gilbert-Elliott channel with the following parameters: $Q = 0.001$, $q = 0.1$, $b = 1$, $g = 0$. $\mathcal{T}(w, 1, y)$ is given by:*

$$\begin{aligned} &\mathcal{T}(w,1,y) = y^5\,[0.099 + 2.4\,10^{-4}w + 2.9\,10^{-4}w^2 \\ &+2.3\,10^{-4}w^3 + 1.3\,10^{-4}w^4 + 6.0\,10^{-4}w^5 + \\ &(-0.15w - 3.0\,10^{-4}w^2 - 3.0\,10^{-5}w^3 - 2.6\,10^{-4}w^4 \\ &-0.001w^5)y + (1.7\,10^{-5} - 2.7\,10^{-5}w + 4.4\,10^{-6}w^2 \\ &+5.9\,10^{-6}w^3 + 6.2\,10^{-8}w^4 - 4.7\,10^{-8}w^5 - 1.2\,10^{-7}w^6 \\ &+1.0\,10^{-7}w^7)y^2 + (6.5\,10^{-10} - 2.3\,10^{-5}w + 4.7\,10^{-5}w^2 \\ &-2.3\,10^{-5}w^3 - 1.7\,10^{-7}w^4 - 7.3\,10^{-8}w^5 + 3.8\,10^{-7}w^6 \\ &-2.1\,10^{-7}w^7 + 5.9\,10^{-12}w^8)y^3]\,/\,[0.1 - (0.2 + 0.15w)y \\ &+0.3wy^2 + (-1.6\,10^{-5} + 2.0\,10^{-5}w + 7.4\,10^{-6}w^2 \\ &-1.2\,10^{-5}w^3)y^3 + (-6.6\,10^{-10} + 2.4\,10^{-5}w \\ &-4.7\,10^{-4}w^2 + 2.4\,10^{-5}w^3 - 6.5\,10^{-10}w^4)y^4]. \end{aligned}$$

For example, the coefficient y^5 in $\mathcal{T}(w, 1, y)$ is shown below:

$$\begin{aligned} &[y^5]\,\mathcal{T}(w,1,y) \,=\, 0.985 \,+\, 0.0025w \,+\, 0.0028w^2 + \\ &0.0023w^3 + 0.0013w^4 + 0.0059w^4. \end{aligned}$$

To compute the performance of convolutional codes over the interleaved channel with finite depth I_d, we just need to redefine the mapping Δ_0 as following:

$$\begin{aligned} \Delta_0: \quad & R \ll y_0, y_1 \gg \longrightarrow R[w] \ll x_0, x_1 \gg: \\ & y_0 \mapsto (x_0 + x_1)\,(x_0 + x_1)^{I_d - 1}; \\ & y_1 \mapsto (x_0 + w\,x_1)\,(x_0 + x_1)^{I_d - 1}. \end{aligned} \tag{18}$$

It is important to notice that the $I_d - 1$ symbols produced by the channel between two received symbols are irrelevant to the decision process. Figure 4 shows an upper bound to P_{ev} versus the memory μ, for the encoder of Figure 1, $R_c = 1/2, K = 4$, over the interleaved Gilbert-Elliott channel, for several values of I_d. The bound was computed using expressions (10), (11), (18), (13)-(16), and the matrices used in (14) are defined in (1)-(3), where the parameters of the Gilbert-Elliott channel are $b = 0.4$, $g = 1 \times 10^{-3}$, and $\rho = q/Q = 20$. The model parameters Q and q are uniquely determined from $\mu = 1 - q - Q$ and ρ. The average error rate of the channel is fixed along the curves and is equal to 2×10^{-2}. The solid lines are analytical results and the dotted lines are obtained by simulations. The values of I_d considered are $I_d = 1$ (no interleaving), $8, 32$. It is seen from the curves that the probability P_{ev} converges quickly to 1 when the channel memory increases. The performance improves substantially when interleaving is incorporated into the system, and the optimum choice of I_d for a given memory value can be determined from this figure. Because the Gilbert Elliott model has a parameter that can be interpreted as the memory of the channel, the effectiveness of coding schemes under several memory conditions can be evaluated. The analytical bounds matches the simulation results very closely when $P_{ev} < 10^{-3}$.

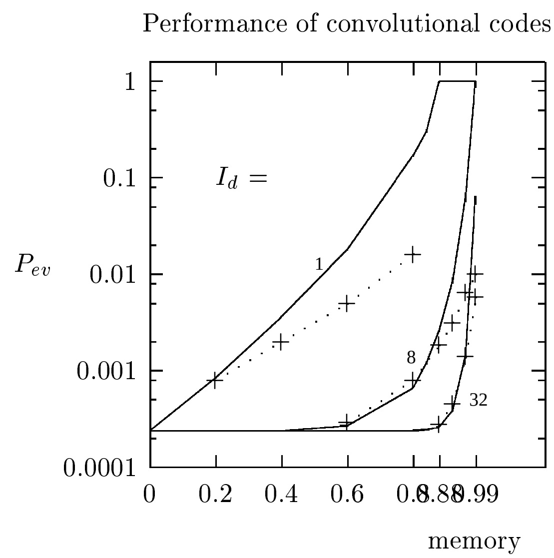

Figure 4: P_{ev} versus memory for a specific convolutional code, $R_c = 1/2$, $K = 4$, on interleaved Gilbert-Elliott channels, having I_d as a parameter. $I_d = 1, 8, 32$. The channel parameters are $b = 0.4$, $g = 1 \times 10^{-3}$, and $\rho = q/Q = 20$. The dotted lines are obtained by simulations.

4. CONCLUSIONS

The generating series for all codewords of infinite dimensionality produced by a convolutional code was derived and new bounds to the first-event error probability and bit error probability for convolutional codes over general FSC models were given. The generating series for burst channels is more refined than the one required for memoryless channels. We take into account the presence of delay constraints, which may limit the maximum value of the interleaving depth. The accuracy of the analytical bounds was demonstrated by comparing them with simulation results.

5. ACKNOWLEDGMENTS

The author acknowledges partial support from the Brazilian National Council for Scientific and Technological Development (CNPq) under Grant No. 300987/96-0.

6. REFERENCES

[1] L. N. Kanal and A.R.K. Sastry, "Models for channels with memory and their applications to error control", Proceedings of the IEEE, Vol. 66, pp. 724-744, July 1978.

[2] E. O. Elliott, "Estimates of error rates for codes on burst-noise channels", *The Bell System Technical Journal*, Vol. 42, pp. 1977-1997, September 1963.

[3] B. D. Fritchman, "A binary channel characterization using partitioned Markov chains", *IEEE Transactions on Information Theory*, Vol. 13, pp. 221-227, April 1967.

[4] Andrew J. Viterbi and Jim K. Omura, *Principles of Digital Communication and Coding*, McGraw-Hill Book Company, New York, 1979.

[5] I. P. Goulden and D. M. Jackson, *Combinatorial Enumeration*, John Wiley & Sons, 1983.

[6] H. Bischl and E. Lutz, "Packet error rate in the non-interleaved Rayleigh channel", IEEE Transactions on Communications, Vol.43, pp.1375-1383, April, 1995.

[7] J. R. Yee and E. J. Weldon Jr., "Evaluation of the performance of error-correcting codes on a Gilbert channel", IEEE Transactions on Communications, Vol.43, pp.2316-2323, August, 1995.

[8] A. I. Drukarev and K. P. Yiu, "Performance of error-correcting codes on channels with memory", IEEE Transactions on Communications, Vol.34, pp.513-521, June,1986.

[9] T. Berman and J. Freedman, "Non-interleaved Reed-Solomon coding over a bursty channel", IEEE Military Communications Conference - MILCOM, San Diego - CA, pp.25.3.1-25.3.4, October 1992.

[10] Cecilio Pimentel and Ian F. Blake, "Concatenated coding performance for FSK modulation on time correlated Rician fading channels", *IEEE Transactions on Communications*, Vol. 46, No.12, pp.1610-1618, December 1998.

[11] M. V. Hedge and M. Naraghi-Pour and X. Chen, "Convolutional coding for finite state channels," *IEEE Transactions on Communications*, Vol. 42, pp. 2231-2238, June 1994.

[12] M. Mushkin and I. Bar-David, "Capacity and coding for the Gilbert-Elliott channels", IEEE Transactions on Information Theory, Vol. 35, pp. 1277-1289, November, 1989.

[13] Cecilio Pimentel and Ian F. Blake, "Enumeration of Markov chains and burst error statistics for finite state models", *IEEE Transactions on Vehicular Technology*, Vol. 48, pp. 415-428, March 1999.

[14] Robert J. McEliece, *The Theory of Information and Coding*, Addison-Wesley, Reading,MA, 1977.