

# The Viterbi Algorithm and Markov Noise Memory

Aleksandar Kavčić<sup>1</sup> and José M. F. Moura<sup>2</sup>

<sup>1</sup>Div. of Engineering and Applied Sciences  
Harvard University  
29 Oxford Street  
Cambridge, MA 02138  
kavcic@hrl.harvard.edu

<sup>2</sup>Dept. of Electrical and Computer Engineering  
Carnegie Mellon University  
5000 Forbes Avenue  
Pittsburgh, PA 15213  
moura@ece.cmu.edu

Published in *IEEE Transactions on Information Theory*, vol. 46, pp. 291-301, January 2000.

Submitted for publication: November 1997. Revised: February 1999.

## Abstract

The paper designs sequence detectors for channels with intersymbol interference (ISI) and correlated (and/or signal-dependent) noise. We describe three major contributions: (i) First, by modeling the noise as a finite order Markov process, we derive the optimal Maximum Likelihood sequence detector (MLSD) and the optimal Maximum A Posteriori (MAP) sequence detector, extending to the correlated noise case the Viterbi algorithm. We show that, when the signal-dependent noise is conditionally Gauss-Markov, the branch metrics in the MLSD are computed from the conditional second order noise statistics. We evaluate the branch metrics using a bank of finite impulse response (FIR) filters. (ii) Second, we characterize the error performance of the MLSD and MAP sequence detector. The error analysis of these detectors is complicated by the correlation asymmetry of the channel noise. We derive upper and lower bounds and computationally efficient approximations to these bounds based on the banded structure of the inverses of Gauss-Markov covariance matrices. An experimental study shows the tightness of these bounds. (iii) Finally, we derive several classes of suboptimal sequence detectors, and demonstrate how these and others available in the literature relate to the MLSD. We compare their error rate performance and their relative computational complexity, and show how the structure of the MLSD and the performance evaluation guide us in choosing a best compromise between several types of suboptimal sequence detectors.

**Index terms:** Viterbi algorithm, maximum likelihood sequence detection, intersymbol interference, Markov channel noise, Markov memory, correlated noise, Gauss-Markov processes.

This work was supported in part by the National Science Foundation under Grant No. ECD-8907068. The United States government has certain rights to this material.

## I Introduction

In digital communications, the Viterbi algorithm, [1], is the maximum likelihood sequence detector (MLSD), [2, 3], for channels with intersymbol interference (ISI) and memoryless noise. Under these conditions, if  $\{x_k\}$  is the state-channel input sequence, and  $\{z_k\}$  is the observable sequence, we have that

$$f(z_k | z_{k-1}, \dots, z_{-\infty}, x_{-\infty}, \dots, x_{\infty}) = f(z_k | x_{k-1}, x_k), \quad (1)$$

where  $f$  denotes the conditional probability density function (pdf) of the observable channel output, conditioned on the infinite past observable sequence and on the whole channel input sequence. Equation (1) states that under an appropriate definition of the channel state-sequence, the ISI white-Gauss noise channel is memoryless, [4].

In many applications, Equation (1) is not an appropriate model, because, besides ISI, the channel noise is correlated, with correlation statistics that are possibly signal-dependent. Our research is motivated by detection in high density magnetic recording, [5], where, for example, the statistics of percolation effects between transitions depend strongly on the particular data sequence recorded: percolation is more likely in closely separated transitions (i.e., successive 1's in the NRZI recorded data sequence) than in widely separated recorded transitions (i.e., successive data symbols are 0's). Reference [6] illustrates with experimental evidence this type of signal-dependent noise in high density magnetic recording. Due to percolation (but also due to nonlinear transition shifts), the noise is correlated across transitions, with the correlation statistics depending on the signal recorded. This signal-dependent correlated noise may overwhelm the white noise component, becoming the dominant noise feature, and may severely degrade the performance of the sequence detector designed for a white-Gauss noise ISI channel. Generalizing the Viterbi algorithm to these more complex channels raises the difficulty that, due to the noise correlation, it is no longer valid to truncate the conditioning in Equation (1) to a finite sequence. In other words, with general correlated noise there is no appropriate definition for the state sequence under which the channel becomes memoryless, and so the model (1) becomes invalid. Early work on this problem includes [7, 8, 9]. More recently, the noise prediction method [10, 11, 12] and the so called K-step Viterbi-detector [13] are attempts to combat the noise correlation in magnetic recording.

In this paper, we extend the Viterbi algorithm to channels with ISI and *signal-dependent correlated* noise: we develop the optimal maximum likelihood sequence detector (MLSD) and study its error performance. The key to our approach is our modeling of the noise: we consider ISI channels with correlated

noise, which goes well beyond white noise ISI channels, but we impose structure on the statistical properties of the correlated noise. We describe the noise, not as a general Gaussian correlated noise, but as a **Gauss-Markov** correlated noise whose second order statistics are signal dependent. We use the Markov assumption to reduce the memory of the noise to finite length. We show in section II that under the noise Markovianity property and with an appropriate definition for the input state-sequence, the following holds

$$f(z_k | z_{k-1}, \dots, z_{-\infty}, x_{-\infty}, \dots, x_{\infty}) = f(z_k | z_{k-1}, \dots, z_{k-L}, x_{k-1}, x_k). \quad (2)$$

where  $L$  is the so called Markov memory length of the noise. In section II, we describe the basic Markovian-memory channel model, and then develop examples of channels with such characteristics. For these Markov channels, the derivation of the Viterbi algorithm becomes a simple exercise in application of Bayes' law. We present the Viterbi algorithm in section III as the solution to maximum a posteriori (MAP) and maximum likelihood (ML) sequence detection in Markovian memory channels. In this section, we consider the general case of Markov noise, not necessarily Gauss. The recent paper [14] also notices that the Viterbi algorithm solves the MLSD in channels with additive Markov noise. In Section IV, we derive explicit expressions for the branch metric when the noise in the channel is signal-dependent additive **Gauss-Markov**. To compute efficiently the branch metrics, we use the properties of Gauss-Markov processes, namely the structure of the inverses of the covariance matrices of such processes, [15]; we compute the branch metrics by using finite impulse response (FIR) filters. We also discuss in section IV adaptive implementations of these FIR filters that track the signal-dependence characteristics of the noise statistics.

A second contribution of this paper is the error analysis we develop for the MLSD and MAP sequence detector for ISI channels with finite memory correlated noise. The error performance of the MLSD is considered in section V. We derive upper and lower bounds for the error probability and by exploiting the structure of the inverse of the covariance matrix of a Gauss-Markov process we develop computationally efficient approximations to the bounds. We demonstrate the tightness of these bounds with a simulation study in realistic scenarios.

The third major issue we address is the derivation of suboptimal sequence detectors. Guided by the structure of the optimal MLSD and its error performance, we introduce in section VI new suboptimal sequence detectors and show how these and other suboptimal detectors, including the noise prediction and K-step detectors, relate to the MLSD. We illustrate their error performance and determine within a certain class of sub-optimal sequence detectors the best compromise from an error performance and computational complexity perspective. Finally, section VII concludes the paper.

**Notation** Before leaving this section, we introduce the basic notation used in the paper. Throughout, we assume that the reader is familiar with the Viterbi algorithm as presented in [3]. We attempt to use the notation in [3] to the extent possible. In this paper, column vectors are denoted by underlined characters, matrices are denoted by boldface characters, and the superscript  $^T$  is used to denote matrix and vector transposition. If  $z_k$  is a discrete-time indexed sequence where  $k$  denotes the time, then the column vector of sequence samples at time  $k_1$  through  $k_2 \geq k_1$  is denoted by  $\underline{z}_{k_2}^{k_1} = [z_{k_1}, z_{k_1+1}, \dots, z_{k_2}]^T$ . The notation  $(\underline{a}|B) \sim \mathcal{N}(\underline{m}, \mathbf{C})$  denotes that the random vector  $\underline{a}$ , conditioned on an event  $B$ , has a normal (joint Gaussian) distribution with mean  $\underline{m}$  and covariance matrix  $\mathbf{C}$ . The probability of an event  $A$  is denoted by  $P(A)$ . Further notation is introduced in section V.

## II Process Model

In this section, we establish the appropriate model for ISI channels with signal dependent correlated noise. We structure the noise so that the resulting channel has finite memory. In subsection II-A, we discuss the general Markov chain model with finite memory, and in subsection II-B we illustrate this model with examples that are relevant to the ISI channel with correlated noise.

### A The Model

Let  $x_k$  represent the state of a finite-state machine at time instant  $k$ . The number of possible states  $M$  is finite. We shall assume that the sequence of states  $x_k$  is a Markov chain in the sense that the probability of the machine being in state  $x_k$  conditioned on all states up to the state at time  $k-1$ , depends only on the state  $x_{k-1}$ , i.e.,

$$P(x_k | \underline{x}_{k-1}^{-\infty}) = P(x_k | x_{k-1}). \quad (3)$$

The transition probabilities  $P(x_k | x_{k-1})$  may be time-dependent, as originally presented by Forney [3]. This is assumed throughout, although not explicitly indicated.

Let the transition between states  $x_{k-1}$  and  $x_k$  produce an observable output random variable  $z_k$ , where the statistics of  $z_k$  may depend on the transition  $(x_{k-1}, x_k)$ , and time  $k$ . We assume that the sequence of output variables  $z_k$  is a Markov sequence with a finite memory length  $L$ , whose memory is dependent only

on the state transition  $(x_{k-1}, x_k)$ , i.e., the conditional probability density function (pdf) obeys

$$f(z_k | \underline{z}_{k-1}^{-\infty}, \underline{x}_{\infty}^{-\infty}) = f(z_k | \underline{z}_{k-1}^{k-L}, x_{k-1}, x_k). \quad (4)$$

Again, these pdfs may also depend on time, which we will not show explicitly.

## B Examples

**Example 1. Channel with ISI and data-dependent Gauss-Markov noise** Let  $a_k$  be a sequence of transmitted symbols (bits). We model a channel with intersymbol interference (ISI) of length  $I \geq 0$  as

$$z_k = y(\underline{a}_k^{k-I}) + n_k. \quad (5)$$

In Equation 5,  $y(\underline{a}_k^{k-I})$  is the noiseless channel output dependent only on the  $I + 1$  latest transmitted bits. For linear channels,  $y(\underline{a}_k^{k-I})$  may be represented as the convolution of the channel response with the transmitted sequence. We consider the channel to be nonlinear. The additive noise term  $n_k$  is considered to be a signal-dependent Gauss-Markov noise process with Markov memory length  $L$ , i.e.,

$$n_k = \underline{b}(\underline{a}_k^{k-I})^T \underline{n}_k^{k-L} + \sigma(\underline{a}_k^{k-I}) w_k. \quad (6)$$

The vector  $\underline{b}(\underline{a}_k^{k-I}) = [b_L(\underline{a}_k^{k-I}), \dots, b_1(\underline{a}_k^{k-I})]^T$  collects the  $L$  coefficients of an autoregressive filter whose values are dependent on the transmitted symbols  $\underline{a}_k^{k-I}$ ,  $w_k$  is a zero-mean unit-variance white Gaussian noise process, and  $\sigma(\underline{a}_k^{k-I})$  is a signal dependent standard deviation. Figure 1 depicts the channel described by equations (5) and (6). As we will derive in Section IV, condition (4) holds for this channel if the state is chosen as a collection of  $I + L$  consecutive transmitted symbols  $x_k = \underline{a}_k^{k-I-L+1}$ . If we further have  $P(\underline{a}_k | \underline{a}_{k-1}^{-\infty}) = P(\underline{a}_k | \underline{a}_{k-1}^{k-I-L})$ , then condition (3) holds too and the channel may be represented by a finite-state machine. If  $a_k$  are binary symbols, then there are  $2^{I+L}$  states and  $2^{I+L+1}$  transition branches in this finite-state machine.

**Example 2. Noncausal ISI and noncausal data-dependent Gauss-Markov noise** The following is a generalization of Example 1 to noncausal ISI and noncausal data-dependent Gauss-Markov noise.

$$z_k = y(\underline{a}_{k+I_2}^{k+I_1}) + n_k \quad (7)$$

$$n_k = \underline{\alpha}(\underline{a}_{k+A_2}^{k+A_1})^T \underline{n}_{k+L_2}^{k-L_1} + \sigma(\underline{a}_{k+B_2}^{k+B_1}) w_k. \quad (8)$$

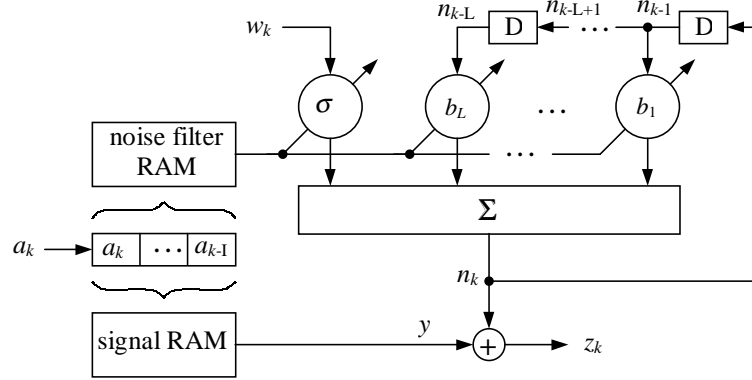


Figure 1: Block diagram of a channel with intersymbol interference (ISI) of length  $I$  and signal-dependent Gauss-Markov noise with Markov memory length  $L$ . The random access memory (RAMs) blocks indicate that both the autoregressive filter and the intersymbol interference are signal-dependent.

Here  $I_2 \geq I_1$ ,  $A_2 \geq A_1$ ,  $B_2 \geq B_1$ ,  $L_1 \geq 0$ , and  $L_2 \geq 0$ . For this channel the state is chosen as  $x_k = \underline{a}_{k+C_2}^{k+C_1+1}$ , where  $C_1 = \min(I_1 - L_2 - L_1, A_1 - L_2, B_1 - L_2)$ , and  $C_2 = \max(I_2, A_2 - L_2, B_2 - L_2)$ . Example 1 is a special case of this Example when  $I_2 = A_2 = B_2 = L_2 = 0$ ,  $I_1 = A_1 = B_1 = -I$ , and  $L_1 = L$ . Note that in this example the noise model is *noncausal* Markov, i.e., the noise at the current time may be dependent on past and future noise values. This is the appropriate model, for example, in magnetic recording, where time stands for the spatial variable, and the ‘past’ and ‘future’ times stand for the immediately adjacent bit-slots on a physical medium.

**Example 3. Non-Markov observables** Often the observed process is a deterministic part corrupted by additive noise. The additive noise may not be Markov, but often it may be approximated by a Markov process if the memory length  $L$  is taken long enough [16]. In this case, the Viterbi algorithm which we present in the next section is the asymptotically optimal detector of the finite-state machine state sequence.

### III Viterbi Sequence Detection

#### A MAP Sequence Detection

We study the estimation of the state sequence  $x_1, \dots, x_K$  ( $K > 0$ ) given a sequence of observables  $z_1, \dots, z_K$  when conditions (3) and (4) hold. We assume that the initial state  $x_0$  is known and that the initial  $L$  realizations  $z_{-L+1}, \dots, z_0$  of the observable random variable are also given, where  $L$  is the Markovian memory length of the observable sequence  $z_k$ .

The maximum a posteriori (MAP) sequence estimate of the state sequence  $\underline{x}_K^1 = [x_1, \dots, x_K]^T$  is the sequence  $\hat{\underline{x}}_K^1 = [\hat{x}_1, \dots, \hat{x}_K]^T$  that maximizes the joint conditional pdf, i.e.,

$$\hat{\underline{x}}_K^1 = \arg \max_{\text{all } \underline{x}_K^1} f(\underline{x}_K^1, \underline{z}_K^1 | x_0, \underline{z}_0^{-L+1}). \quad (9)$$

As shorthand notation, denote by  $f(x, z | \text{i.c.})$  the conditional pdf on the right-hand side of (9), where i.c. stands for *initial conditions*. Using conditions (3) and (4), we can factor  $f(x, z | \text{i.c.})$  as

$$f(x, z | \text{i.c.}) = \prod_{k=1}^K P(x_k | \underline{x}_{k-1}^0, \underline{z}_0^{-L+1}) \prod_{k=1}^K f(z_k | \underline{z}_{k-1}^{-L+1}, \underline{x}_K^0) = \prod_{k=1}^K P(x_k | x_{k-1}) \prod_{k=1}^K f(z_k | \underline{z}_{k-1}^{k-L}, x_{k-1}, x_k). \quad (10)$$

In Equation (10), we assumed that  $P(x_k | \underline{x}_{k-1}^0, \underline{z}_0^{-L+1}) = P(x_k | \underline{x}_{k-1}^0)$ , i.e., that the state transition probabilities are independent of the observables. Using the definition of conditional pdfs, we may further substitute

$$f(z_k | \underline{z}_{k-1}^{k-L}, x_{k-1}, x_k) = \frac{f(\underline{z}_k^{k-L} | x_{k-1}, x_k)}{f(\underline{z}_{k-1}^{k-L} | x_{k-1}, x_k)} = \frac{f(\underline{z}_k^{k-L} | x_{k-1}, x_k)}{\int_{z_k=-\infty}^{\infty} f(\underline{z}_k^{k-L} | x_{k-1}, x_k) dz_k} \quad (11)$$

for the last term in (10). Maximizing (10) is the same as minimizing its negative logarithm. Taking the negative logarithm of  $f(x, z | \text{i.c.})$ , we get

$$-\ln f(x, z | \text{i.c.}) = \sum_{k=0}^{K-1} \Lambda_{\text{MAP}}(\underline{z}_k^{L+1}, x_{k-1}, x_k), \quad (12)$$

where the MAP branch metrics are

$$\Lambda_{\text{MAP}}(\underline{z}_k^{L+1}, x_{k-1}, x_k) = -\ln P(x_k | x_{k-1}) - \ln \frac{f(\underline{z}_k^{k-L} | x_{k-1}, x_k)}{\int_{z_k=-\infty}^{\infty} f(\underline{z}_k^{k-L} | x_{k-1}, x_k) dz_k}. \quad (13)$$

The MAP sequence estimate is thus that sequence  $\hat{\underline{x}}_K^1$  for which the sum of the branch metrics in (12) is minimized.

## B ML Sequence Detection

The maximum likelihood (ML) sequence estimate is that sequence  $\hat{\underline{x}}_K^1$ , for which the conditional pdf

$$f(z | x, \text{i.c.}) = f(\underline{z}_K^1 | \underline{x}_K^0, \underline{z}_0^{-L+1}) \quad (14)$$

is maximized. With arguments similar to those for MAP sequence detection, it follows that the ML estimate is that sequence  $\hat{\underline{x}}_K^1$  for which the following sum of branch metrics is minimized

$$-\ln f(z|x, \text{i.c.}) = \sum_{k=0}^{K-1} \Lambda_{ML}(\underline{z}_k^{k-L}, x_{k-1}, x_k). \quad (15)$$

The ML branch metric is

$$\Lambda_{ML}(\underline{z}_k^{k-L}, x_{k-1}, x_k) = -\ln \frac{f(\underline{z}_k^{k-L}|x_{k-1}, x_k)}{\int_{z_k=-\infty}^{\infty} f(\underline{z}_k^{k-L}|x_{k-1}, x_k) dz_k}. \quad (16)$$

## C Viterbi Trellis Implementation

The implementation of the minimization in (12) and (15) is very similar to the standard implementation described in [2, 3]. The major difference is that the computation of the branch metrics relies now on a window of observed samples  $\underline{z}_k^{k-L} = [z_{K-L}, \dots, z_k]^T$ , instead of on just one sample  $z_k$ . Depending on whether we are interested in MAP or ML sequence detection, the branch metrics are given by (13) and (16) respectively.

## IV Gaussian Statistics and Branch Metrics

We now consider the important case in applications, [16], of channels with intersymbol interference (ISI) and signal-dependent Gauss-Markov noise as described in Example 1 in Section II and depicted in Figure 1. The extension to the more general context of Example 2 is relatively easy. In the model of Example 1,  $I \geq 0$  is the ISI length, and  $L \geq 0$  is the Markov memory length. We also assume that  $a_k$  is a *binary* symbol sequence.

### A Branch Metrics

We first define the state  $x_k$ . We consider the ML sequence detection and work with Equation (16). To find the metric  $\Lambda_{ML}(\underline{z}_k^{k-L}, x_{k-1}, x_k)$ , we must find the logarithm of the pdf  $f(\underline{z}_k^{k-L}|x_{k-1}, x_k)$ . The vector  $\underline{z}_k^{k-L}$  in (16) involves observables  $z_{k-L}$  through  $z_k$ , whose conditional means (ideal channel outputs) are  $y(\underline{a}_{k-L}^{k-L-I})$  through  $y(\underline{a}_k^{k-I})$ , see (5). Therefore, the trellis branch  $(x_{k-1}, x_k)$  should be chosen so that



it collects all inputs  $a_{k-L-I}$  through  $a_k$  involved in determining the channel outputs  $y(\underline{a}_{k-L}^{k-L-I})$  through  $y(\underline{a}_k^{k-I})$  corresponding to the observables  $z_{k-L}$  through  $z_k$ . A simple choice is to label the state by grouping the incoming bits as

$$x_k = \underline{a}_k^{k-I-L+1}. \quad (17)$$

We define the vector of  $L+1$  ideal channel outputs  $\underline{Y}(x_{k-1}, x_k) = [y(\underline{a}_{k-L}^{k-L-I}), \dots, y(\underline{a}_k^{k-I})]^T$  and rewrite (5) as

$$\underline{z}_k^{k-L} = \underline{Y}(x_{k-1}, x_k) + \underline{n}_k^{k-L}. \quad (18)$$

Since  $n_k$  is a conditionally Gaussian process, we have  $(\underline{z}_k^{k-L} | x_{k-1}, x_k) \sim \mathcal{N}(\underline{Y}(x_{k-1}, x_k), \mathbf{C}(x_{k-1}, x_k))$ , where  $\mathbf{C}(x_{k-1}, x_k)$  is the  $(L+1) \times (L+1)$  covariance matrix of  $\underline{n}_k^{k-L}$  conditioned on the pair of states  $(x_{k-1}, x_k)$ . Substituting this pdf into the expression for the ML branch metric (16), and cancelling constant terms common to all branches, we obtain the Gaussian ML branch metric

$$\begin{aligned} \mathcal{M}_{ML}(\underline{z}_k^{k-L}, x_{k-1}, x_k) &= \ln \frac{\det \mathbf{C}(x_{k-1}, x_k)}{\det \mathbf{c}(x_{k-1}, x_k)} \\ &+ [\underline{z}_k^{k-L} - \underline{Y}(x_{k-1}, x_k)]^T \mathbf{C}(x_{k-1}, x_k)^{-1} [\underline{z}_k^{k-L} - \underline{Y}(x_{k-1}, x_k)] \\ &- [\underline{z}_{k-1}^{k-L} - \underline{y}(x_{k-1}, x_k)]^T \mathbf{c}(x_{k-1}, x_k)^{-1} [\underline{z}_{k-1}^{k-L} - \underline{y}(x_{k-1}, x_k)]. \end{aligned} \quad (19)$$

Here  $\mathbf{c}(x_{k-1}, x_k)$  is the upper  $L \times L$  principal minor of  $\mathbf{C}(x_{k-1}, x_k)$ , i.e.,

$$\mathbf{C}(x_{k-1}, x_k) = \begin{bmatrix} \mathbf{c}(x_{k-1}, x_k) & \underline{\mathbf{c}}(x_{k-1}, x_k) \\ \underline{\mathbf{c}}(x_{k-1}, x_k)^T & c(x_{k-1}, x_k) \end{bmatrix}, \quad (20)$$

and  $\underline{y}(x_{k-1}, x_k)$  collects the first  $L$  elements of  $\underline{Y}(x_{k-1}, x_k)$ .

In a similar fashion, the Gaussian MAP branch metric can be shown to equal

$$\mathcal{M}_{MAP}(\underline{z}_k^{k-L}, x_{k-1}, x_k) = \mathcal{M}_{ML}(\underline{z}_k^{L+1}, x_{k-1}, x_k) - 2 \ln P(x_k | x_{k-1}). \quad (21)$$

## B FIR Filter Branch Metric Implementation

The metric given in (19) involves two vector-matrix multiplications, making the complexity of the metric computation order  $\mathcal{O}((L+1)^2)$ . We next exploit the structure of the covariance matrices to bring the

computational complexity down to  $\mathcal{O}(L+1)$  by implementing the branch metric computation using an FIR (finite impulse response) filter.

Let the covariance matrix be partitioned as in (20). Using the matrix inversion lemma [17], the inverse can be written as

$$\mathbf{C}(x_{k-1}, x_k)^{-1} = \begin{bmatrix} \mathbf{c}(x_{k-1}, x_k)^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{\underline{w}_c(x_{k-1}, x_k) \underline{w}_c(x_{k-1}, x_k)^T}{\gamma(x_{k-1}, x_k)}. \quad (22)$$

In Equation (22),  $\gamma(x_{k-1}, x_k)$  and  $\underline{w}_c(x_{k-1}, x_k)$  are given by the solutions of the ‘signal-dependent’ Yule-Walker equations [18] and can be shown to equal [19]

$$\gamma(x_{k-1}, x_k) = \frac{\det \mathbf{C}(x_{k-1}, x_k)}{\det \mathbf{c}(x_{k-1}, x_k)} = \sigma^2 \left( \underline{a}_k^{k-I} \right) \quad (23)$$

$$\underline{w}_c(x_{k-1}, x_k) = \begin{bmatrix} -\mathbf{c}(x_{k-1}, x_k)^{-1} \underline{c}(x_{k-1}, x_k) \\ 1 \end{bmatrix} = \begin{bmatrix} -\underline{b} \left( \underline{a}_k^{k-I} \right) \\ 1 \end{bmatrix}, \quad (24)$$

where  $\sigma \left( \underline{a}_k^{k-I} \right)$  and  $\underline{b} \left( \underline{a}_k^{k-I} \right)$  are the autoregressive coefficients from the model Equation (6).

Substituting (22) into (19), we rewrite the Gaussian ML branch metric as

$$\mathcal{M}_{ML}(\underline{z}_k^{k-L}, x_{k-1}, x_k) = \ln \sigma^2 \left( \underline{a}_k^{k-I} \right) + \frac{\left( \begin{bmatrix} -\underline{b} \left( \underline{a}_k^{k-I} \right)^T & 1 \end{bmatrix} \left[ \underline{z}_k^{k-L} - \underline{Y}(x_{k-1}, x_k) \right] \right)^2}{\sigma^2 \left( \underline{a}_k^{k-I} \right)} \quad (25)$$

The metric in (25) is obtained by filtering the observed vector  $\underline{z}_k^{k-L}$  through an FIR filter with coefficients  $\underline{w}_c(x_{k-1}, x_k)$  given in (24). This filter is the inverse of the autoregressive filter in Figure 1 and it uncorrelates the autoregressive (AR) process  $n_k$ . Thus we may interpret the metric (25) as first uncorrelating the noise with an FIR filter, and then applying the Euclidean (square) metric to the output of the filter. This is illustrated in Figure 2 where the branch metric is shown to be the processed output of a branch-dependent FIR filter. Since the FIR filter coefficients  $-\underline{b} \left( \underline{a}_k^{k-I} \right)$  depend on only  $I+1$  symbols, and since there is a total of  $2^{I+L+1}$  branches in each stage of the trellis, it turns out that there are  $2^{I+1}$  distinct FIR filters in the detector, where  $2^L$  branches may share a single filter.

A special case of the above FIR filter occurs when the covariance matrix is not signal-dependent, i.e.,  $\mathbf{C}(x_{k-1}, x_k) = \mathbf{C}$ . In this case, the filter  $\underline{w}_c(x_{k-1}, x_k) = \underline{w}_c$  is a state-invariant tap-delay line that whitens the stationary Gauss-Markov noise process. This is an intuitively pleasing result, since it is well known that a stationary (causal) Gauss-Markov AR sequence can be whitened by a time-invariant FIR filter. Notice

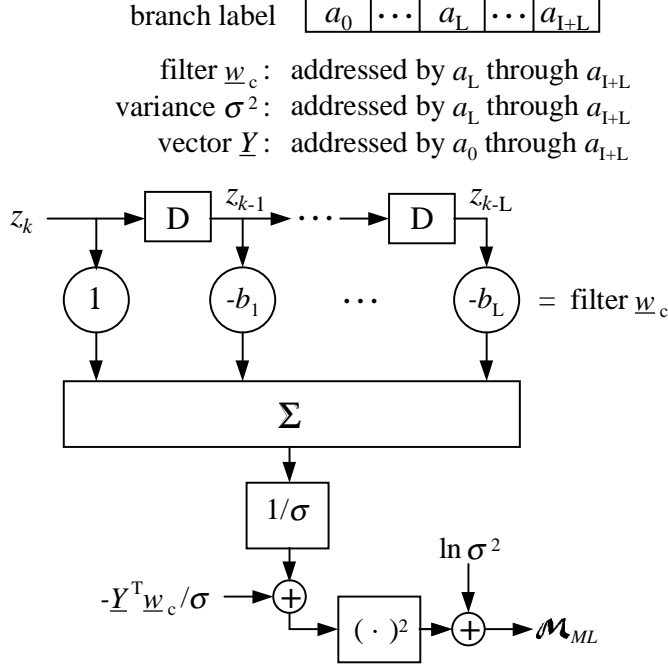


Figure 2: Block diagram of an FIR filter computing the ML branch metric for a channel with ISI of length  $I$  and Gauss-Markov noise with memory length  $L$ . Since there are  $2^{I+L+1}$  branches in each stage of the trellis, the branches may be labeled by  $I + L + 1$  bits. Of these bits, only  $I + 1$  are used to address the FIR filters. A bank of  $2^{I+1}$  different FIR filters is needed for optimal detection, while  $2^L$  branches share the same filter.

also that, if  $\underline{w}_c$  is state-invariant, *all* branches share the same filter. We can then move the whitening filter to the front-end of the Viterbi detector. This provides another explanation of why the Viterbi detector is optimal only if the channel noise is Markov. Suppose that the noise is not Markov, i.e., it is not the output of an autoregressive filter as in (6). Then to whiten this noise requires passing the observed channel outputs through an infinite impulse response (IIR) filter, thus introducing infinite-length ISI. However, with infinite length ISI, the Viterbi detector requires an infinite number of states in order to be the optimal detector, which is clearly unrealizable.

We make a distinction here with what is usually done in Kalman-Bucy filtering (KBF) [20] with Markov observation noise, see ch. 11 of [21]. In this case, the (vector) observations  $\underline{z}_k$  are modeled as

$$\underline{z}_{k+1} = \mathbf{H}_k \underline{x}_k + \underline{v}_k, \quad (26)$$

where  $\underline{x}_k$  is the state vector and  $\underline{v}_k$  is a first order Markov vector following the model

$$\underline{v}_{k+1} = \mathbf{A}_k \underline{v}_k + \underline{w}_k, \quad (27)$$

where  $\underline{w}_k$  is a white Gauss vector sequence. Preprocessing the observations  $\underline{z}_k$  into the new observation

$$\underline{z}'_k = \underline{z}_{k+1} - \mathbf{A}_k \underline{z}_k, \quad (28)$$

one can then apply standard KBF theory. Although similar, there is a fine distinction with Viterbi decoding. Due to the requirement of finite ISI, the Markov noise model (27) needs to be restricted to autoregressive processes (see Equation (6)), while in KBF no such restriction is required.

Finally, we comment on the practical implementation of the branch metric computation. As seen from Equations (19) or (25), implementing the ML branch metric requires knowledge of the second order statistics  $\mathbf{C}(x_{k-1}, x_k)$ . In practice, the second order statistics are typically not known and need to be estimated from the data. Tracking the statistics adaptively is preferable in many applications because the covariance statistics may time-drift or be nonstationary for some other reason. Adaptive covariance tracking has been widely covered in the literature, see e.g. [18, 22]. For a method applied in magnetic recording signal detection see [5]. Notice also that, if the involved covariance matrices are signal-dependent, and so non-stationary and non-Toeplitz, many efficient AR estimation algorithms [18, 23] based on the Levinson-Durbin recursion will then not work.

## V Error Analysis

We develop in this section the error analysis for the Viterbi sequence decoder when the noise in the ISI channel is signal-dependent and correlated. Our results extend the error analysis in [2, 24] to correlated finite memory ISI channels. However, due to the channel nonlinearity and the signal-dependent noise correlation, the analysis must carefully treat the lack of symmetry. For example, in contrast with the linear channel with stationary white noise, if  $a'_1, \dots, a'_m$  and  $a''_1, \dots, a''_m$  are two valid binary sequences in the trellis, the probability of detecting  $a'$  when  $a''$  is sent is not the same as detecting  $a''$  when  $a'$  is sent. Also, unlike the linear channel with stationary white noise, we cannot formulate the upper bound using the flow-graph transfer function [2] because relative distances between trellis paths are not symmetric due to the asymmetric signal-dependent noise correlation.

To formulate the bounds, we introduce first the necessary notation. Then, in subsection V-A, we derive the *binary hypothesis* error probability for asymmetric Gauss-Markov noise channels, followed in subsection V-B by a computationally efficient approximation to this error probability. In subsections V-C and V-D, we formulate upper and lower bounds using the binary hypothesis error probability expressions. Finally, in subsection V-E, we illustrate the error performance results by analyzing the detector performance of a simple ISI channel with finite memory Markov noise.

**Notation** Let  $\varepsilon_M = (\Psi'_M, \Psi''_M)$  denote a length- $M$  error event. By  $\Psi'_M$ , we denote the correct path through the trellis spanning  $M + 1$  consecutive states  $x'_k, \dots, x'_{k+M}$ . The erroneous path is denoted by  $\Psi''_M$  and spans states  $x''_k, \dots, x''_{k+M}$ . It is assumed that  $x'_k = x''_k$ ,  $x'_{k+M} = x''_{k+M}$ , and  $x'_{k+m} \neq x''_{k+m}$  for  $1 \leq m \leq M - 1$ . Since a state is defined as  $x_k = \underline{a}_k^{k-I-L+1}$ , an equivalent notation is  $\Psi'_M = \underline{a}'_{k+M}^{k-I-L+1}$  and  $\Psi''_M = \underline{a}''_{k+M}^{k-I-L+1}$ , where  $\underline{a}'_{k+m}^{k-I-L+m+1} \neq \underline{a}''_{k+m}^{k-I-L+m+1}$  for  $1 \leq m \leq M - 1$  and  $\underline{a}'_{k+m}^{k-I-L+m+1} = \underline{a}''_{k+m}^{k-I-L+m+1}$  for all other  $m$ .

## A Binary Hypothesis Error Probability

The binary hypothesis error probability  $P_2(\varepsilon_M)$  is the probability of detecting  $\Psi''_M$  when  $\Psi'_M$  is the true path. This error occurs when the accumulated metric (25) along the path  $\Psi'_M$  is greater than the accumulated metric along  $\Psi''_M$ , i.e.<sup>1</sup>,

$$P_2(\varepsilon_M) = P \left\{ \sum_{m=1}^M \mathcal{M}_{ML} \left( \underline{z}_{k+m}^{k+m-L}, \underline{a}'_{k+m}^{k+m-I-L} \right) > \sum_{m=1}^M \mathcal{M}_{ML} \left( \underline{z}_{k+m}^{k+m-L}, \underline{a}''_{k+m}^{k+m-I-L} \right) \right\}. \quad (29)$$

Notice that we can use an arbitrary  $k$  in (29). For convenience, we shall use  $k = L$  throughout this subsection.

To evaluate the expression on the right hand side of (29), we use the special structural properties of Gauss-Markov covariance matrices [19] and rewrite (29) as

$$P_2(\varepsilon_M) = P \left\{ \ln \det \mathbf{C}_{\Psi'_M} + \left( \underline{N}'_{M+L} \right)^T \mathbf{C}_{\Psi'_M}^{-1} \underline{N}'_{M+L} > \ln \det \mathbf{C}_{\Psi''_M} + \left( \underline{N}''_{M+L} \right)^T \mathbf{C}_{\Psi''_M}^{-1} \underline{N}''_{M+L} \right\}. \quad (30)$$

The matrices  $\mathbf{C}_{\Psi'_M}$  and  $\mathbf{C}_{\Psi''_M}$  are the covariance matrices of  $\underline{z}_{L+M}^1$  when  $\underline{a}'_{L+M}^{1-I}$  and  $\underline{a}''_{L+M}^{1-I}$  are the transmitted (or written on a memory medium) binary sequences, respectively. The innovation vector  $\underline{N}'_{L+M}$  is

<sup>1</sup>If the detector is an MAP detector rather than the ML detector, we substitute the MAP metric into Equation (29) to get the MAP binary hypothesis error probability.

defined as

$$\underline{N}_{L+M}'^1 = \underline{z}_{L+M}^1 - \underline{y}' = \underline{z}_{L+M}^1 - \left[ y \left( \underline{a}_1'^{1-I} \right), \dots, y \left( \underline{a}_{L+M}'^{L+M-I} \right) \right]^T, \quad (31)$$

where the values  $y \left( \underline{a}_k'^{k-I} \right)$  are the same as those used in the model Equation (5). The vectors  $\underline{N}_{M+L}''^1$  and  $\underline{y}''$  are defined similarly as in (31), where the superscript  $'$  is replaced by  $''$ .

Next, we briefly describe how to obtain the inverses  $\mathbf{C}_{\Psi_M'}^{-1}$  and  $\mathbf{C}_{\Psi_M''}^{-1}$  of the covariance matrices from the model parameters in (6). Since the process  $n_k$  is conditionally Gauss-Markov with Markov memory length  $L$ , it follows from [15] that  $\mathbf{C}_{\Psi_M'}^{-1}$  is an  $L$ -banded matrix. Its upper Cholesky decomposition is given by

$$\mathbf{C}_{\Psi_M'}^{-1} = \mathbf{U}_{\Psi_M'} \mathbf{D}_{\Psi_M'} \mathbf{U}_{\Psi_M'}^T. \quad (32)$$

In Equation (32), according to [19, 25], the upper triangular  $L$ -banded matrix  $\mathbf{U}_{\Psi_M'}$  is

$$\mathbf{U}_{\Psi_M'} = \begin{bmatrix} \mathbf{U}_{1'} & -\underline{b} \left( \underline{a}_{L+1}'^{L+1-I} \right) & & \mathbf{0} \\ & 1 & \ddots & \\ & & \ddots & -\underline{b} \left( \underline{a}_{L+M}'^{L+M-I} \right) \\ \mathbf{0} & & & 1 \end{bmatrix}, \quad (33)$$

and the diagonal matrix  $\mathbf{D}_{\Psi_M'}$  is

$$\mathbf{D}_{\Psi_M'} = \text{diag} \left[ \mathbf{D}_{1'} \quad 1/\sigma^2 \left( \underline{a}_{L+1}'^{L+1-I} \right) \quad \dots \quad 1/\sigma^2 \left( \underline{a}_{L+M}'^{L+M-I} \right) \right]. \quad (34)$$

The values  $\underline{b} \left( \underline{a}_i'^{i-I} \right)$  and  $\sigma \left( \underline{a}_i'^{i-I} \right)$  in (33) and (34) are the same as those in the model Equation (6). The upper triangular matrix  $\mathbf{U}_{1'}$  and the diagonal matrix  $\mathbf{D}_{1'}$  are the upper Cholesky factors of  $\mathbf{C}_{1'}^{-1} = \mathbf{U}_{1'} \mathbf{D}_{1'} \mathbf{U}_{1'}^T$ , where  $\mathbf{C}_{1'}$  is the conditional covariance matrix of  $\underline{z}_L^1$  when the sequence of transmitted (written) symbols is  $\underline{a}_L'^{1-I}$ . The elements of  $\mathbf{C}_{1'}$  can be obtained by rearranging a signal-dependent set of Yule-Walker equations and solving a linear system of equations. Since computing  $\mathbf{C}_{1'}$  is tedious but straight-forward, we refer the reader to an example provided in [26]. The Cholesky decomposition of  $\mathbf{C}_{\Psi_M''}^{-1}$  is obtained similarly.

Once we have computed the Cholesky decompositions (32) of both  $\mathbf{C}_{\Psi_M'}^{-1}$  and  $\mathbf{C}_{\Psi_M''}^{-1}$ , we can rewrite the binary hypothesis error probability for the error event  $\varepsilon_M = (\Psi_M', \Psi_M'')$  as

$$P_2(\varepsilon_M) = P \left\{ \underline{w}^T \underline{w} > \ln \det \mathbf{\Sigma}_{\varepsilon_M} + (\underline{w} - \underline{m}_{\varepsilon_M})^T \mathbf{\Sigma}_{\varepsilon_M}^{-1} (\underline{w} - \underline{m}_{\varepsilon_M}) \right\}, \quad (35)$$

where  $\underline{w}$  is an  $(L+M) \times 1$  Gaussian random vector with  $E[\underline{w}] = \underline{0}$  and  $E[\underline{w} \underline{w}^T] = \mathbf{I}$ . The vector  $\underline{m}_{\varepsilon_M}$  is

given by

$$\underline{m}_{\varepsilon_M} = \mathbf{Q} \mathbf{D}_{\Psi'_M}^{1/2} \mathbf{U}_{\Psi'_M}^T (\underline{y}'' - \underline{y}') = \begin{bmatrix} d & 0 & \cdots & 0 \end{bmatrix}^T, \quad (36)$$

where  $d = \|\mathbf{D}_{\Psi'_M}^{1/2} \mathbf{U}_{\Psi'_M}^T (\underline{y}'' - \underline{y}')\|$ ,  $\underline{y}'$  and  $\underline{y}''$  are defined in (31),  $\|\cdot\|$  is the  $L_2$  vector norm,  $\mathbf{Q}$  is a unitary matrix (for example, the Householder reflector  $\mathbf{Q} = 2\underline{v} \underline{v}^T / \|\underline{v}\|^2 - \mathbf{I}$ , where  $\underline{v} = \underline{q}/\|\underline{q}\| + \underline{e}_1$ ,  $\underline{q} = \mathbf{D}_{\Psi'_M}^{1/2} \mathbf{U}_{\Psi'_M}^T (\underline{y}'' - \underline{y}')$ , and  $\underline{e}_1 = [1, 0, \dots, 0]^T$ , see [27]), and  $\Sigma_{\varepsilon_M}$  satisfies

$$\Sigma_{\varepsilon_M}^{-1} = \mathbf{Q} \mathbf{D}_{\Psi'_M}^{-1/2} \mathbf{U}_{\Psi'_M}^{-1} \mathbf{C}_{\Psi''_M}^{-1} \mathbf{U}_{\Psi'_M}^{-T} \mathbf{D}_{\Psi'_M}^{-1/2} \mathbf{Q}^T \quad \text{and} \quad \det \Sigma_{\varepsilon_M} = \frac{\det \mathbf{D}_{\Psi'_M}}{\det \mathbf{D}_{\Psi''_M}} \quad (37)$$

In the general case, the expression on the right-hand side of (35) cannot be simplified by integrating a  $\chi^2$ -like distribution. These simplifications are possible only in special cases, for example when  $\Sigma_{\varepsilon_M}$  is diagonal (for a list of all special cases where a simplification is possible, see [28]). In the general case, however, evaluation of (35) involves integrating a zero-mean  $\mathbf{I}$ -covariance multivariate Gaussian probability density function (pdf) in the region where the condition on the right-hand side of (35) holds. Since this is a region outlined by a quadratic form, its shape must be determined on a case-by-case basis, which ultimately may be computationally impractical. A brute force alternative is to evaluate (35) numerically by Monte-Carlo, but this is also computationally intensive for low error rate (high SNR) scenarios. In the next subsection, we develop a computationally efficient Q-function approximation that leads to an accurate evaluation of (35) for high SNR scenarios where Monte Carlo numeric evaluations are not viable.

## B Approximate Binary Hypothesis Error Probability

Computing (35) requires integrating a zero-mean  $\mathbf{I}$ -covariance multivariate Gaussian pdf over a region where

$$f_{\varepsilon_M}(\underline{w}) = \underline{w}^T \underline{w} - \ln \det \Sigma_{\varepsilon_M} - (\underline{w} - \underline{m}_{\varepsilon_M})^T \Sigma_{\varepsilon_M}^{-1} (\underline{w} - \underline{m}_{\varepsilon_M}) > 0. \quad (38)$$

Since a Gaussian function has a very fast decay, we approximate (35) by

$$P_2(\varepsilon_M) \approx Q(d_2), \quad (39)$$

where  $d_2$  is the point on  $f_{\varepsilon_M}(\underline{w}) = 0$  closest to the origin and  $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt$ . To determine  $d_2$ , one may use any minimization technique (Lagrange multipliers, steepest descent, etc.). A particularly fast

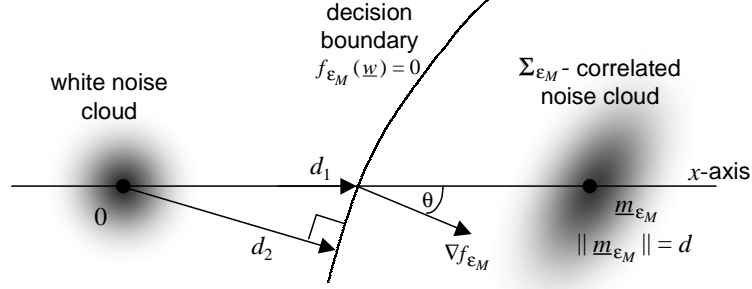


Figure 3: Binary hypothesis signal and noise constellation encountered when computing the binary hypothesis error probability. Note the asymmetric noise clouds.

method is to find the point  $d_1$  on the boundary  $f_{\epsilon_M}(\underline{w}) = 0$  between 0 and  $\underline{m}_{\epsilon_M}$  by solving a quadratic equation, see Figure 3. For a given matrix  $\Sigma_{\epsilon_M}$ , the point  $d_1$  will be between 0 and  $\underline{m}_{\epsilon_M}$  if  $d$  is large enough, i.e., if the signal-to-noise ratio (SNR) is large enough. Since  $d_2$  is typically close to  $d_1$ , we can obtain  $d_2$  in a few iterations, or we may simply set  $d_2 \approx d_1 \cos \theta$ , see Figure 3. We prefer the iterative method since it is not too computationally expensive.

## C Upper Bound

The bit error probability  $P_b$  can be upper bounded by the union bound [2]. Since our channel is asymmetric (nonlinear with signal-dependent noise correlation) we need a union bound expression that generalizes the one in [2]. One can verify that the two expressions are equivalent for linear symmetric channels. Let  $M$  denote the length (in branches) of an error event  $\epsilon_M = (\Psi'_M, \Psi''_M)$ . For our binary channel, the minimum allowable error event length can be verified to be  $M_{min} = I + L + 1$ , where  $I$  is the ISI length, and  $L$  is the Markov memory length of noise. The set of all allowable error events of length  $M$  is denoted by  $E_M$ . Denote by  $b_{\#}(\epsilon_M)$  the number of erroneous bits corresponding to the error event  $\epsilon_M$ . Let  $P(x)$  be the a priori probability of the trellis state  $x$ . Similarly, let  $P(x_k, x_{k-1})$  denote the conditional probability of transition from state  $x_{k-1}$  to state  $x_k$ . The binary hypothesis error probability associated with the error event  $\epsilon_M$  is  $P_2(\epsilon_M)$  and can be computed by either Monte Carlo simulation or approximated by (39). The upper bound on bit error probability is given by

$$P_b \leq \sum_{M=M_{min}}^{\infty} \sum_{\Psi'_M} \left[ P(x'_k) \prod_{i=1}^M P(x'_{k+i} | x'_{k+i-1}) \right] \sum_{\substack{\Psi''_M \text{ such that} \\ \epsilon_M = (\Psi'_M, \Psi''_M) \in E_M}} b_{\#}(\epsilon_M) P_2(\epsilon_M). \quad (40)$$



This bound accounts for both the ML and the MAP sequence detector, where the corresponding binary hypothesis error probability  $P_2(\varepsilon_M)$  needs to be substituted (see footnote to Equation (29)). If all symbol sequences are equally likely, then  $P(x'_k) = 2^{-(L+I)}$  and  $P(x'_{k+i}|x'_{k+i-1}) = 1/2$ . A practical method for evaluating the bound in (40) is to truncate the first sum to values of  $M < M_{max}$ , where  $M_{max}$  is a predetermined large enough constant. Notice that in (40), the binary hypothesis error probability  $P_2(\varepsilon_M)$  needs to be determined for every qualifying error event  $\varepsilon_M$ , which makes the bound computationally expensive if  $P_2(\varepsilon_M)$  is computed by Monte Carlo simulations rather than by the approximation (39).

## D Lower Bound

Obviously, a lower bound is obtained by picking any single term of the union bound (40). This, however, may not be a tight bound. To get a tighter bound, we modify the genie-assisted bound [24]

**Modified genie-assisted bound** Denote by  $P(\varepsilon)$  the probability of an error event. Let  $\pi_c(P_c)$  be the probability that the input sequence  $a'_k$  is such that there is a probability  $P \geq P_c$  of confusion with some allowable sequence  $a''_k$ . Even with a genie-assisted detector that has to choose only between  $a'_k$  and  $a''_k$ , the probability of an error event  $P(\varepsilon)$  will be greater than  $\pi_c(P_c) \cdot P_c$ . Clearly, then

$$P(\varepsilon) \geq \max_{0 \leq P_c \leq 1} [\pi_c(P_c) \cdot P_c]. \quad (41)$$

The difference between this formulation and the one in [24] is that we assume  $P \geq P_c$  instead of  $P = P_c$ . In fact, the bound in (41) will never be looser (and in many cases tighter) than the one in [24], even in linear symmetric channels.

We use the result in (41) to find a lower bound for the bit error probability. Consider only error events of length  $M_{min} = I + L + 1$ . The number of these events for binary signaling is  $N = 2^{I+L+M_{min}} = 2 \cdot 4^{I+L}$ . Label these error events as  $\varepsilon_{M_{min}}(1), \varepsilon_{M_{min}}(2), \dots, \varepsilon_{M_{min}}(N)$ , where their order is such that  $P_2(\varepsilon_{M_{min}}(1)) \geq P_2(\varepsilon_{M_{min}}(2)) \geq \dots \geq P_2(\varepsilon_{M_{min}}(N))$ , and  $P_2(\varepsilon_{M_{min}}(i))$  is the binary error probability of the error event  $\varepsilon_{M_{min}}(i)$ . The probability  $P_2(\varepsilon_{M_{min}}(i))$  can be computed either through Monte Carlo simulation or with the approximation in (39). In the binary signaling case, for every sequence  $\Psi'_{M_{min}}(i)$  of  $M_{min} + 1$  states, there exists only one erroneous sequence  $\Psi''_{M_{min}}(i)$ . We therefore have  $\sum_{i=1}^N P(\Psi'_{M_{min}}(i)) = 1$ , i.e.,  $\Psi'_{M_{min}}(i)$  are disjoint sequences. If we choose  $P_c = P_2(\varepsilon_{M_{min}}(i))$  for some  $i$  where  $1 \leq i \leq M_{min}$ , then  $\pi_c(P_c) \geq \sum_{j=1}^i P(\Psi'_{M_{min}}(j))$  because there may be error events of length  $M > M_{min}$  whose binary hypothesis error probability is greater than  $P_c$ . In the interest of keeping the bound computationally simple, we ignore

these error events of length  $M > M_{min}$ . Since for all error events of length  $M_{min}$ , the number of erroneous bits is  $b_{\#}(\varepsilon_{M_{min}}(i)) = 1$ , we have

$$P_b \geq \pi_c(P_c) \cdot P_c \geq \left[ \sum_{j=1}^i P(\Psi'_{M_{min}}(j)) \right] P_2(\varepsilon_{M_{min}}(i)) \quad (42)$$

for any  $1 \leq i \leq N$ . Maximizing the right hand side of (42), we obtain a tighter lower bound

$$P_b \geq \max_{1 \leq i \leq N} \left\{ \left[ \sum_{j=1}^i P(\Psi'_{M_{min}}(j)) \right] P_2(\varepsilon_{M_{min}}(i)) \right\}. \quad (43)$$

This expression provides for a lower bound for both the ML and the MAP sequence detector, where the appropriate binary hypothesis error probability would be used, see footnote to Equation (29). When all sequences  $\Psi'_{M_{min}}(i)$  are equiprobable, we have  $P(\Psi'_{M_{min}}(j)) = \frac{1}{N} = \frac{1}{2}4^{-(I+L)}$ . Thus, the modified genie-assisted bound provides the lower bit error probability bound

$$P_b \geq \frac{1}{2}4^{-(I+L)} \max_{1 \leq i \leq N} [i \cdot P_2(\varepsilon_{M_{min}}(i))]. \quad (44)$$

Since the terms  $P_2(\varepsilon_{M_{min}}(i))$  have already been computed as part of (40), no further computation is needed for the lower bound. Notice also that while (44) is always a lower bound, it may not be the tightest of the modified genie-assisted bounds because we considered only the minimum-length error events. The following example shows, however, that this bound is relatively tight.

## E Performance Evaluation Example

We study the performance of the maximum likelihood sequence detector for a simple channel with ISI length  $I = 1$  and Markov memory length  $L = 1$ . The channel is given by the model equations (5) and (6), also depicted in Figure 1, where the values of the parameters are given in Table 1. The channel represented in Table 1 can be viewed as a nonlinearly deviated  $1 - D$  partial response channel with signal-dependent correlated noise.

Figure 4 shows the performance of the ML sequence detector. The different values of the signal-to-noise ratio (SNR) shown in Figure 4 are obtained by scaling the noiseless channel response (column denoted by  $y(\underline{a}_k^{k-1})$  in Table 1) while keeping the relative distance between the signal points the same. This corresponds to scaling the signal power while the noise constellation is kept the same. In Figure 4, the solid line represents the simulated performance of the ML sequence detector (MLSD). We do not show

$\left(\underline{a}_k^{k-1}\right)^T$	$y\left(\underline{a}_k^{k-1}\right)$	$\sigma\left(\underline{a}_k^{k-1}\right)$	$b_1\left(\underline{a}_k^{k-1}\right)$
00	-0.1	1.0	-0.2
01	1.2	1.8	-0.6
10	-0.8	1.2	-0.6
11	0.1	1.0	-0.2

Table 1: Nonlinearly distorted 1 –  $D$  partial response channel with ISI length  $I = 1$  and Markov memory length  $L = 1$ .

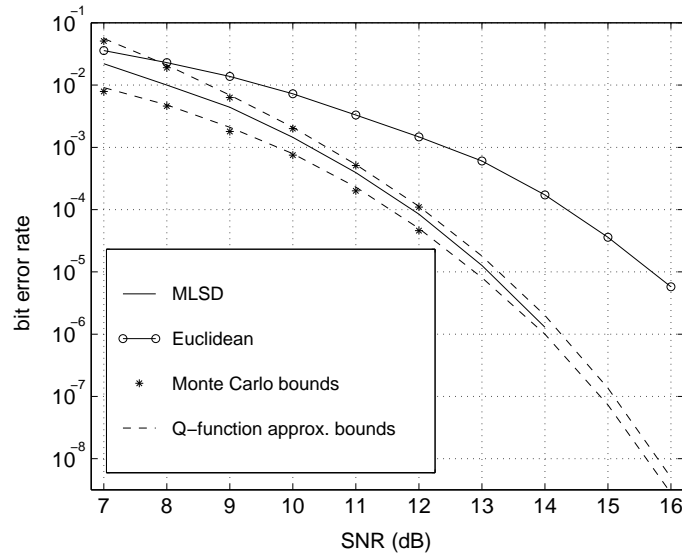


Figure 4: Performance evaluations of detectors applied to the channel in Table 1.

the MLSD performance curve beyond SNR=14dB since the Monte Carlo simulations become expensive to compute. Figure 4 also shows the simulated performance of the Euclidean detector applied to the channel in Table 1. The Euclidean detector is a Viterbi detector that assumes the noise to be white. Figure 4 shows that at the error rate of  $10^{-5}$  the MLSD has a gain of 2.5 dB over the Viterbi Euclidean detector.

Figure 4 also plots the bit error probability bounds. The stars (\*) in Figure 4 show the bounds computed by evaluating the binary hypothesis error probability with Monte Carlo simulations. As the Monte Carlo simulations become too computationally intensive beyond SNR=12dB, we only show the stars (\*) for SNR ≤ 12dB. On the other hand, the approximate bounds (dashed lines in Figure 4) computed by approximating the binary hypothesis error probability with a Q-function, see equation (39), can be computed at a fractional computational cost for any value of the SNR. Figure 4 also shows that the Q-function approximation is extremely close to the bounds computed by Monte Carlo simulations. As

expected, the bounds are tightening as the SNR increases which helps us evaluate the MLSD performance at high SNRs where Monte Carlo simulations of the MLSD are impractical. We point out that at extremely high error rates ( $10^{-1}$  and above) the upper bound may fail to converge, but that is rarely a concern since at those error rates the MLSD performance can readily be evaluated through simulations.

## VI Suboptimal Detectors

Being the optimal detector, the maximum likelihood sequence detector (MLSD) bounds from below the error rate of any suboptimal detector of lower complexity. Furthermore, the structure of the signal-dependent MLSD gives new insight into developing suboptimal detectors. Here we cover several suboptimal schemes and discuss their relationship to the MLSD.

From Figure 4 we see that there is a performance region between the MLSD and the Euclidean detector that can be filled with suboptimal detectors with complexities possibly lower than that of the MLSD. There are many ways to construct such suboptimal receivers. We limit ourselves to suboptimal receivers with a certain structure. We look for suboptimal receivers that are Viterbi-like, i.e., that operate on a trellis and use branch/path metric minimization as their detection strategy. We define the complexity of a Viterbi-like detector as the number of multiplications it has to perform per clock interval (symbol interval) in order to compute all its branch metrics. We ignore additions. In many realizations of the Viterbi detector this may not be an accurate representation of the detector complexity since there exist several practical finite-precision methods to simplify the multiplication operation. Nevertheless, for simplicity, we trust that the multiplications count provides at least a reasonable representation for the complexity of a detector.

When  $I$  is the ISI length and  $L$  the Markov memory length, the MLSD for the binary channel model in Equations (5) and (6) has  $2^{I+L}$  states and  $2^{I+L+1}$  branches. Thereby,  $2^{I+1}$  branches share a single FIR filter used to compute the branch metric, see Figure 2. In Figure 2, if we count each tap-multiplication by  $b_i \left( \underline{a}_k^{k-I} \right)$ , each division by  $\sigma \left( \underline{a}_k^{k-I} \right)$ , and each squaring operation as one multiplication, the count of multiplications needed to compute all the branch metrics of the MLSD in one clock interval is  $(2^L + L + 1) 2^{I+1}$ . Since the computational complexity of the Euclidean Viterbi detector is  $2^{I+1}$  per clock interval, we have that the computational complexity increase factor of the MLSD over the Euclidean detector is

$$F_{MLSD}(L) = 2^L + L + 1. \quad (45)$$

We next consider suboptimal detectors with lower computational increase factors.

## A Low Order Autoregressive (AR) Approximation (Q-AR Detection)

We introduce in this subsection a new sub-optimal detector that we refer to as the low order  $Q$ -AR detector.

A Gauss-Markov noise process of order  $L$  is the output of an autoregressive (AR) model of order  $L$ . The inverse covariance matrix of an  $L$ -th order Gauss-Markov process is  $L$ -banded, meaning that it has  $2L + 1$  non-zero diagonals, while the remaining are all-zero diagonals [15]. We obtain a suboptimal detector by using an AR approximation [16] of order  $Q \leq L$ . Let  $\mathbf{C}_Q$  be the covariance matrix of the  $Q$ -th order AR process approximation. The matrix  $\mathbf{C}_Q^{-1}$  is then  $Q$ -banded. We can show that the  $Q$ -banded matrix  $\mathbf{C}_Q^{-1}$  that minimizes the Kullback-Leibler mean information loss due to approximating the original  $L$ -th order AR process with a  $Q$ -th order AR process is given by the following decomposition [19]

$$\mathbf{C} = \mathbf{C}_Q + \begin{bmatrix} & & & \Delta \\ & & \text{0}_{2Q+1} & \\ \Delta^T & & & \end{bmatrix} \quad \text{and} \quad \mathbf{C}_Q^{-1} = \begin{bmatrix} & & & \mathbf{0} \\ & & \mathbf{D}_{2Q+1} & \\ \mathbf{0} & & & \end{bmatrix}, \quad (46)$$

where  $\mathbf{0}_{2Q+1}$  denotes a band of  $2Q + 1$  all-zero diagonals, and  $\mathbf{D}_{2Q+1}$  denotes a band of  $2Q + 1$  non-zero diagonals. Note that the decomposition in (46) is unique since there exists only one matrix  $\mathbf{C}_Q$  such that the  $Q$ -band of  $\mathbf{C}_Q$  equals the  $Q$ -band of  $\mathbf{C}$ , and that  $\mathbf{C}_Q^{-1}$  is  $Q$ -banded. The suboptimal Viterbi detector based on the low order AR approximation computes the branch metric using a  $Q$ -tap FIR filter instead of an  $L$ -tap FIR filter<sup>2</sup>. Consequently, the detector has  $2^{I+Q}$  states and  $2^{I+Q+1}$  branches where  $2^{I+1}$  branches share a single  $Q$ -tap FIR filter. The computational complexity increase factor of the low-order  $Q$ -AR detector over the Euclidean detector is

$$F_{Q-AR}(Q) = 2^Q + Q + 1. \quad (47)$$

Obviously, if  $Q = L$ , we have the MLSD. If  $Q = 0$ , we obtain the suboptimal detector presented in [7, 8] for which the complexity increase factor is  $F_{Q-AR}(0) = 2$ .

---

<sup>2</sup>Note that the  $Q$ -th order filter is *not* simply a truncation of the optimal  $L$ -th order filter.

## B Block Partitioning (K-Step Detection)

In [13], Altekari and Wolf present a suboptimal detector by replacing the covariance matrix  $\mathbf{C}$  by a matrix  $\mathbf{C}'$  obtained by keeping only the  $K \times K$  block diagonal elements of  $\mathbf{C}$ , i.e.,

$$\mathbf{C} \approx \begin{bmatrix} \mathbf{R} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{R} \end{bmatrix} = \mathbf{C}', \quad (48)$$

where  $\mathbf{R}$  is a  $K \times K$  principal minor of  $\mathbf{C}$ . This contrasts with the  $Q$ -AR approximation of Equation (46) where it is the *inverse* covariance that is *banded* rather than the covariance matrix itself, which is *block-partitioned*. In [13], the method is called the  $K$ -step Viterbi detector. In [13], the authors provide a metric computation method involving vectors of observed samples and covariance matrices  $\mathbf{R}$ . Using techniques presented in Subsection IV-B, we can show that the  $K$ -step detector may also be implemented with low-order FIR filters (which is also a computationally less expensive implementation than in [13]). However, the order  $Q$  of the FIR filters is cyclicly time-varying. For example, the 3-step detector would use banks of  $2^{I+1}$  ( $Q=0$ )-FIR filters for the first stage of the trellis, ( $Q=1$ )-FIR filters for the second stage and ( $Q=2$ )-FIR filters for the third stage (for low order signal-dependent AR processes, see [16, 19]). In the fourth stage of the trellis, the detector goes back to the bank of ( $Q=0$ )-FIR filters and repeats the cycle. Thus, generally, a  $K$ -step detector would cyclicly employ ( $Q=0$ )-AR through ( $Q=K-1$ )-AR approximations at different stages of the trellis. Thereby, the FIR filters used to calculate the branch metrics are the exact inverses of the  $Q$ -AR filter approximations, see Section IV-B. At trellis stages where  $Q \geq L$ , the bank of FIR filters are the inverses of the bank of  $L$ -AR filters used in the channel model Equations (5) and (6). Since the  $K$ -step detector uses non-uniform AR-approximations at different trellis stages, the number of multiplications has to be averaged over a cycle of  $K$  trellis stages to get the average computational complexity per clock interval. The computational complexity increase factor of the  $K$ -step Viterbi detector over a Euclidean Viterbi detector is

$$F_{K\text{-step}}(K) = \begin{cases} \frac{2^K-1}{K} + \frac{K+1}{2} & \text{for } K \leq L \\ \frac{2^L-1+(K-L)(2^L+L-1)}{K} + \frac{L(L+1)}{2K} & \text{for } K > L \end{cases}. \quad (49)$$

Clearly, the ( $K=1$ )-step detector is the same as the ( $Q=0$ )-AR detector, which is the detector presented in [7, 8].

## C Block Skipping (S-Skip Detection)

A drawback of the K-step detector is that it reaches the MLSD performance only in the limit as  $K \rightarrow \infty$ . We propose here a new class of suboptimal detectors that reaches the MLSD performance for a finite order of suboptimality. The following equation shows the  $L$ -band of a covariance matrix  $\mathbf{C}$  of an  $L$ -th order Gauss-Markov process partitioned in contiguous principal blocks of size  $(L+1) \times (L+1)$

$$\mathbf{C} = \begin{bmatrix} \boxed{\begin{smallmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{smallmatrix}} & & \Delta \\ & \boxed{\begin{smallmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{smallmatrix}} & \\ \Delta^T & & \boxed{\begin{smallmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{smallmatrix}} \end{bmatrix}. \quad (50)$$

It can be shown, see for example [19], that the elements of the  $L$ -band of  $\mathbf{C}$  determine in a unique way the covariance matrix  $\mathbf{C}$  of the  $L$ -th order Gauss Markov process, in particular, they determine the elements of the blocks  $\Delta$ . We can conclude that the MLSD derives the branch metrics based on all covariance coefficients in this  $L$ -band of  $\mathbf{C}$ . As an alternative, out of the  $(L+1) \times (L+1)$  blocks outlined on the right hand side of (50), a suboptimal algorithm may ignore the blocks outlined by the dashed lines in (50). We refer to such a procedure as the  $(S=1)$ -skip detector. In general, if we skip  $S$  blocks at a time, we have the S-skip detector. Much like detailed in subsection VI-B, the S-skip detector may be designed through cyclic implementation of non-uniform Q-AR approximations. Each cycle spans  $S+1$  trellis stages and the detector cyclicly employs banks of  $(Q=L-S)$ -FIR,  $(Q=L-S+1)$ -FIR,  $\dots$ ,  $(Q=L)$ -FIR filters. The bank of  $(Q=L)$ -FIR filters is given by the inverses of the actual  $L$ -th order AR filters in (5) and (6). The computational complexity increase factor of the S-skip Viterbi detector over the Euclidean Viterbi detector is

$$F_{S\text{-skip}}(S) = \frac{2^{L+1} - 2^{L-S}}{S+1} + \frac{2L - S + 2}{2} \quad \text{for } 1 \leq S \leq L. \quad (51)$$

The  $(S=0)$ -skip detector is clearly the MLSD. It is also apparent that the  $(S=L)$ -skip detector is equivalent to the  $K$ -step block partitioning detector when  $K=L+1$ , where  $L$  is the Markov memory length of the noise.

## D Non-Uniform AR-Approximations

The K-step and S-skip detectors are subclasses of Viterbi detectors that apply different  $Q$ -th order AR approximations at different stages of the trellis. One can easily think of numerous other Viterbi-like

detectors based on this concept, where the different orders of AR approximations are not necessarily cyclicly employed. A typical scenario would be in nonstationary detection. Suppose that at some point during data transmission the data rate increases, i.e., the clock interval (symbol interval) shrinks. If the new clock interval is too short to perform Viterbi detection based on a high order AR approximation, the detector may resort to a less computationally expensive lower order AR approximation, still retaining the communication link at the price of a less reliable communication. Combinations of different orders of AR approximations will lead to a large number of possible detector designs.

## E Decision Feedback

Decision feedback mechanisms can be used to lower the computational complexity of Viterbi detectors. Reduced-state sequence estimators (RSE) [29] use premature decisions to reduce the ISI length and thus lower the number of states. Noise predictors [30, 11, 12], on the other hand, use premature decisions to guess the future noise sample. In the context of Markov noise models, this corresponds to lowering the Markov memory length. Hybrids between reduced-state sequence estimators and noise predictors are also possible. Furthermore, we can envision combining the non-uniform AR-approximations with decision feedback strategies to form a larger subclass of Viterbi-like suboptimal detectors. A common feature to all decision feedback suboptimal detectors is error propagation, see results presented within [13]. Error propagation precludes the use of decision feedback detectors in many applications where larger blocks of errors cannot be tolerated, e.g., in block error-correction coding/decoding.

## F Comparative Results

In this subsection we provide the error performance for a number of detectors and show that the  $Q$ -AR detectors provide usually the best compromise between error rate and computational complexity. Table 2 shows the parameters (for explanations, see Equations (5) and (6)) of a very noisy real channel encountered in magnetic recording applications [16]. The channel in Table 2 is a nonlinearly distorted class-4 partial response (PR4) channel with ISI length  $I = 2$  and signal-dependent correlated noise with Markov memory length  $L = 3$ . Figure 5 shows for this channel the error rate versus the computational complexity increase factor  $F$  for a number of detectors: the MLSD, the Euclidean Viterbi decoder, and several suboptimal  $Q$ -AR, K-step and S-skip detectors. The computational complexity increase factors are normalized with respect to that of the Euclidean detector, whose operating point in Figure 5 corresponds then to the value



$\left(\underline{a}_k^{k-2}\right)^T$	$y\left(\underline{a}_k^{k-2}\right)$	$\sigma\left(\underline{a}_k^{k-2}\right)$	$b_1\left(\underline{a}_k^{k-2}\right)$	$b_2\left(\underline{a}_k^{k-2}\right)$	$b_3\left(\underline{a}_k^{k-2}\right)$
000	-0.12	0.10	-0.3	-0.1	0.2
001	0.8	0.20	-0.4	-0.4	0.1
010	0.0	0.12	-0.2	-0.3	0.0
011	0.8	0.17	-0.1	-0.3	-0.1
100	-0.62	0.11	-0.4	-0.2	-0.2
101	-0.12	0.14	-0.6	-0.5	-0.2
110	-0.62	0.10	-0.6	-0.4	-0.2
111	-0.12	0.12	-0.3	-0.1	0.0

Table 2: Nonlinearly distorted  $1 - D^2$  partial response channel (PR4 channel) with ISI length  $I = 2$  and Markov memory length  $L = 2$ .

1 on the horizontal axis. We do not consider decision feedback suboptimal detectors since they suffer from error-propagation. The error rates were computed by Monte Carlo simulations with  $10^6$  bits. The results confirm, as mentioned above, that the suboptimal detectors based on low order AR approximations (Q-AR detectors) provide the best complexity-vs.-error-rate trade-off. The K-step detector is a reasonable alternative to the Q-AR detector only for low complexities and high error rates. On the other hand, the S-skip detector is a reasonable alternative to the Q-AR detector only for high complexities and low error rates.

## VII Summary

The paper studies the Maximum Likelihood sequence detector (MLSD) and the MAP sequence detector for intersymbol interference (ISI) channels with correlated noise whose statistics are possibly signal-dependent. We describe three major contributions to the problem. First, by modeling the correlated noise as a Markov process of finite order, we derive the optimal MLSD and MAP sequence detector as extensions of the Viterbi decoder for ISI channels with finite memory noise. Second, we provide an error performance study of the MLSD and MAP sequence detector, deriving tight upper and lower bounds for the probability of error. Finally, we present several classes of suboptimal sequence detectors, including new algorithms, and compare their computational complexity and their error performance.

The key to deriving a finite implementation for the optimal sequence detector is our modeling the correlated noise as a finite order Markov process, see section II. We showed in section III that, as long as the memory of the noise is Markov, we can extend the Viterbi algorithm to implement the maximum

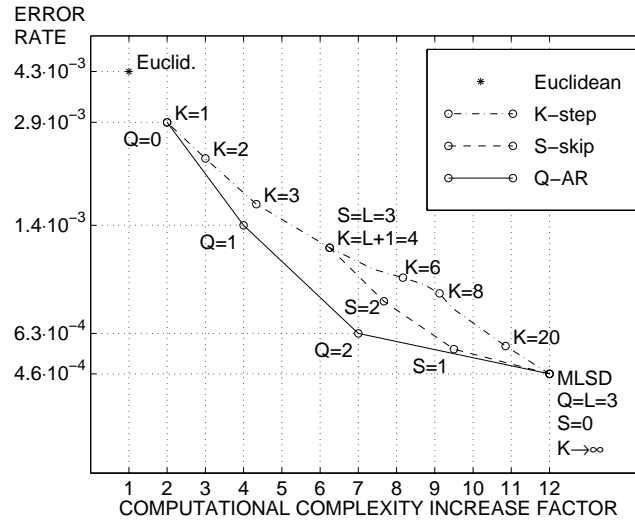


Figure 5: Performance vs. computational complexity. Shown are the operating points of the Euclidean detector, the MLSD, and different orders of the suboptimal Q-AR, S-skip, and K-step detectors. The detectors are applied to the channel given in Table 2.

a posteriori (MAP) detector and the maximum likelihood sequence detector (MLSD). The complexity of the MLSD is exponentially proportional to the sum of the ISI length  $I$  plus the Markovian noise memory length  $L$ . The branch metrics of the detector are functions of  $L + 1$  consecutive samples of the observable process.

When the noise is Gaussian with Markovian memory (signal-dependent Gauss-Markov noise), see section IV, the branch metrics are expressed in terms of the conditional second order statistics of the noise. In the binary signaling case, there are  $2^{I+L}$  states in the trellis and  $2^{I+L+1}$  branches in each stage of the trellis. We presented an efficient FIR filter based computation of the branch metrics using a bank of  $2^{I+1}$   $L$ -th order FIR filters. The structure of the receiver resembles the structure of the detector in the memoryless noise scenario, in which we preprocess the received sequence to uncorrelate the noise and then apply the Viterbi detection. In the case of noise with memory, however, the preprocessing cannot be done up-front – it needs to be implemented separately in every branch of the Viterbi trellis. The constraint of finite ISI length requires that we use only FIR filters for the preprocessing, which in turn limits us to Markov (i.e., autoregressive) channel noise only. When the channel noise memory is not Markov, we find the best fitting Markov process to describe it and then apply the Viterbi algorithm to asymptotically reach optimality. In nonstationary environments, adaptive tracking of the conditional covariance statistics is required since the branch metrics depend on these statistics.

We presented in section V the error analysis in terms of bounds on the probability of bit errors. While formally similar to the analysis for white and signal-independent noise, the bounds are computationally more complex due to the signal-dependent noise correlation asymmetry in the channel. We simplified the computation of the bounds by developing a method that exploits the banded structure of the inverses of Gauss-Markov covariance matrices. We have further simplified the bound computations using Q-function approximations. A simulation study was carried out showing the good quality of the bounds as predictors of the error performance. As expected, the bounds are tighter at higher values of the signal-to-noise ratios (SNR).

We use the structure of the extended Viterbi detector for Markovian memory channels as a guideline to derive new suboptimal detectors that are computationally simpler than the MLSD. We also showed how several suboptimal algorithms proposed in the literature are related to the MLSD. In section VI, low order autoregressive (AR) approximations were utilized to yield a whole class of suboptimal sequence detectors. An experimental study was carried out to show that, out of all these suboptimal methods, the strategy that provides the best complexity-vs.-performance trade-off is the one based on fully utilizing a  $Q$ -band of the noise covariance matrix, i.e., the Q-AR detector. We have shown how to exploit the banded form of the inverse of Markov covariance matrices to construct suboptimal receivers that asymptotically reach optimality with higher order approximations and that do not suffer from error propagation problems.

## References

- [1] A. D. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. 13, pp. 260–269, April 1967.
- [2] G. D. Forney Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 363–378, March 1972.
- [3] G. D. Forney Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, March 1973.
- [4] R. G. Gallager, *Information Theory and Reliable Communication*. New York: John Wiley and Sons, Inc., 1968.
- [5] A. Kavčić and J. M. F. Moura, "Correlation-sensitive adaptive sequence detection," *IEEE Trans. Magn.*, vol. 34, pp. 763–771, May 1998.

- [6] J.-G. Zhu and H. Wang, "Noise characteristics of interacting transitions in longitudinal thin film media," *IEEE Trans. Magn.*, vol. 31, pp. 1065–1070, March 1995.
- [7] I. Lee and J. M. Cioffi, "Performance analysis of the modified maximum likelihood sequence detector in the presence of data-dependent noise," in *Proc. 26th Asilomar Conference*, pp. 961–964, Oct. 1992.
- [8] W. Zeng and J. Moon, "Modified Viterbi algorithm for jitter-dominated  $1 - D^2$  channel," *IEEE Trans. Magn.*, vol. MAG-28, pp. 2895–2897, Sept. 1992.
- [9] L. C. Barbosa, "Towards the detection of signals in presence of signal-dependent noise," in *Proc. SPIE: Coding and Signal Processing for Information Storage*, vol. 2605, (Philadelphia), pp. 48–56, Oct. 1995.
- [10] P. R. Chevillat, E. Eleftheriou, and D. Maiwald, "Noise predictive partial response equalizers and applications," in *IEEE ICC'92 Conference Record*, pp. 942–947, June 1992.
- [11] E. Eleftheriou and W. Hirt, "Noise-predictive maximum-likelihood (NPML) detection for the magnetic recording channel," in *IEEE ICC'96 Conference Record*, (Dallas, TX), pp. 556–560, June 1996.
- [12] J. D. Coker, E. Eleftheriou, R. L. Galbraith, and W. Hirt, "Noise-predictive maximum likelihood NPML detection," *IEEE Trans. Magn.*, vol. 34, pp. 110–117, Jan. 1998.
- [13] S. A. Altekari and J. K. Wolf, "Improvements in detectors based upon colored noise," *IEEE Trans. Magn.*, vol. 34, pp. 94–97, Jan. 1998.
- [14] F. Alajaji, N. Phamdo, F. Nariman, and T. E. Fuja, "Detection of binary Markov sources over channels with additive Markov noise," *IEEE Trans. Inform. Theory*, vol. 42, pp. 230–239, Jan. 1996.
- [15] J. M. F. Moura and N. Balram, "Recursive structure of noncausal Gauss Markov random fields," *IEEE Trans. Inform. Theory*, vol. 38, pp. 334–354, March 1992.
- [16] A. Kavčić and A. Patapoutian, "Signal-dependent autoregressive channel model," in *IEEE INTERMAG Conference*, (Kyongju, Korea), May 1999.
- [17] L. L. Scharf, *Statistical Signal Processing: Detection, Estimation and Time Series Analysis*. New York: Addison-Wesley Publishing Co., 1991.
- [18] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs: Prentice Hall, 2nd ed., 1991.
- [19] A. Kavčić and J. M. F. Moura, "Gauss-Markov approximations and information loss," tech. rep., Dept. of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, March 1998. 30 pages, submitted for publication.

- [20] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *ASME J. Basic Engr.*, March 1961.
- [21] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs: Prentice-Hall, Inc., 1979.
- [22] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs: Prentice-Hall, Inc., 1987.
- [23] J. P. Burg, "A new analysis technique for time series data." NATO Advanced Study Institute on Signal Processing, Enschede, The Netherlands, 1968.
- [24] G. D. Forney Jr., "Lower bounds on error probability in the presence of large intersymbol interference," *IEEE Trans. Commun.*, vol. 20, pp. 76–77, Feb. 1972.
- [25] A. Kavčić and J. M. F. Moura, "Correlation structures for optimizing information criteria," in *Proc. IEEE Inform. Theory Workshop on Detection, Estimation, Classification and Imaging*, (Santa Fe, NM), p. 25, Feb. 1999.
- [26] A. Kavčić, "Error probability bounds for nonlinear channels with intersymbol interference and signal-dependent correlated noise," in submitted to *IEEE GLOBECOM 99*, (Rio de Janeiro), Dec. 1999.
- [27] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore: The Johns Hopkins University Press, 1983.
- [28] K. S. Miller, *Multidimensional Gaussian Distributions*. New York: John Wiley, 1964.
- [29] M. V. Eyuboğlu and S. U. H. Qureshi, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. Commun.*, vol. 36, pp. 13–20, Jan. 1988.
- [30] M. V. Eyuboğlu, "Detection of coded modulation signals on linear, severely distorted channels using decision-feedback noise prediction with interleaving," *IEEE Trans. Commun.*, vol. 36, pp. 401–409, April 1988.