OPTIMUM PREDICTIVE ITERATIVE DECODING OF TURBO CODES IN COLOURED GAUSSIAN NOISE

K. Vasudevan
Department of Electrical Engineering
Indian Institute of Technology Kanpur-208 016, INDIA
Email: vasu@iitk.ac.in

ABSTRACT
This paper addresses the problem of iterative decoding in the presence of coloured noise. Although most of the current literature deals with iterative decoding in additive white Gaussian noise (AWGN), in many practical situations, the noise is correlated. This correlation is usually due to the use of a linear equalizer to remove ISI. In this paper, we first derive the optimum detector for coloured noise and then extend the concept to iterative decoding. Simulation results show that depending on the noise correlation, performance improvement of the order of 5–11 dB can be obtained by using the predictive iterative decoder.

KEY WORDS
Turbo codes, prediction filter, coloured noise.

1 Introduction

Turbo decoding has been well studied under various kinds of channel impairment like AWGN [1] and Rayleigh fading [2]. Non-coherent turbo decoding in AWGN has also been studied [3]. In this paper we propose the Predictive Iterative Decoding (PID) algorithm for additive coloured Gaussian noise (ACGN) channels. Maximum likelihood detection of signals in coloured noise has been studied earlier in [4] and *independently* in [5, 6].

The paper is organized as follows. In Section 2 we derive the optimum detector for ACGN channels, which can be efficiently implemented using the Viterbi algorithm (VA). This detector is referred to as the predictive VA [5, 6]. In Section 3 we derive the predictive iterative decoder (PID) for ACGN channels. In Section 4 we present the simulation results. Finally, in Section 5 we present our conclusions and scope for future work.

2 System Model

Assume that L symbols have been transmitted. The symbols are taken from an M-ary alphabet. The received signal can be written as [5]:

$$\tilde{\mathbf{r}} = \tilde{\mathbf{S}}^{(i)} + \tilde{\mathbf{w}} \tag{1}$$

where $\tilde{\mathbf{r}}$ is an $L \times 1$ column vector of the received samples, $\tilde{\mathbf{S}}^{(i)}$ is an $L \times 1$ vector of the i^{th} possible symbol sequence ($0 \le i \le M^L - 1$) and $\tilde{\mathbf{w}}$ is an $L \times 1$ column vector of correlated Gaussian noise samples. Throughout this paper, we denote complex quantities by a tilde, e.g. $\tilde{x}$. Real quantities are denoted without a tilde, e.g. x. Boldface letters denote vectors or matrices.

The maximum likelihood (ML) detector maximizes the joint conditional pdf:

$$\max_j p\left(\tilde{\mathbf{r}}|\tilde{\mathbf{S}}^{(j)}\right) \qquad \text{for } 0 \le j \le M^L - 1 \tag{2}$$

which is equivalent to:

$$\max_j \frac{1}{(2\pi)^L \det\left(\tilde{\mathbf{R}}\right)} \exp\left[-\frac{1}{2}\left(\tilde{\mathbf{r}} - \tilde{\mathbf{S}}^{(j)}\right)^H \left(\tilde{\mathbf{R}}\right)^{-1} \left(\tilde{\mathbf{r}} - \tilde{\mathbf{S}}^{(j)}\right)\right] \tag{3}$$

where the covariance matrix, conditioned on the j^{th} possible symbol sequence, is given by

$$\tilde{\mathbf{R}} = \frac{1}{2}E\left[\tilde{\mathbf{w}}\tilde{\mathbf{w}}^H\right]. \tag{4}$$

The diagonal elements of $\tilde{\mathbf{R}}$ is a constant equal to $\sigma_{\tilde{w}}^2$, which is equal to the variance of coloured noise. Since $\tilde{\mathbf{R}}$ is not a diagonal matrix, the maximization in (3) cannot be implemented recursively using the Viterbi algorithm. However, if we perform a Cholesky factorization on $\tilde{\mathbf{R}}$, we get [7, 5, 6]:

$$\left(\tilde{\mathbf{R}}\right)^{-1} = \tilde{\mathbf{A}}^H \mathbf{D}^{-1} \tilde{\mathbf{A}} \tag{5}$$

where $\tilde{\mathbf{A}}$ is an $L \times L$ lower triangular matrix given by:

$$\tilde{\mathbf{A}} \stackrel{\Delta}{=} \begin{bmatrix} 1 & 0 & \ldots & 0 \\ \tilde{a}_{1,\,1} & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{a}_{L-1,\,L-1} & \tilde{a}_{L-1,\,L-2} & \ldots & 1 \end{bmatrix} \tag{6}$$

where $\tilde{a}_{k,\,p}$ denotes the p^{th} coefficient of the optimal k^{th}-order prediction filter. The $L \times L$ matrix $\mathbf{D}$ is a diagonal matrix of the prediction error variance denoted by:

$$\mathbf{D} \stackrel{\Delta}{=} \begin{bmatrix} \sigma^2_{e,\,0} & \ldots & 0 \\ \vdots & \vdots & 0 \\ 0 & \ldots & \sigma^2_{e,\,L-1} \end{bmatrix}. \tag{7}$$

Observe that $\sigma^2_{e,\,j}$ denotes the prediction error variance for the optimal j^{th}-order predictor ($0 \le j \le L-1$). Substituting (5) into (3) and noting that det $\tilde{\mathbf{R}}^{(j)}$ is *independent* of the symbol sequence j, we get:

$$\max_j \exp\left[-\frac{1}{2}\left(\tilde{\mathbf{r}}-\tilde{\mathbf{S}}^{(j)}\right)^H \tilde{\mathbf{A}}^H\mathbf{D}^{-1}\tilde{\mathbf{A}}\left(\tilde{\mathbf{r}}-\tilde{\mathbf{S}}^{(j)}\right)\right]. \tag{8}$$

which is equivalent to:

$$\max_j \prod_{k=0}^{L-1} \exp\left[-\frac{1}{2}\frac{\left|\tilde{z}_k^{(j)}\right|^2}{\sigma^2_{e,\,k}}\right] \tag{9}$$

where the prediction error at time k for the j^{th} symbol sequence, $\tilde{z}_k^{(j)}$, is an element of $\tilde{\mathbf{z}}^{(j)}$ and is given by:

$$\left[\begin{array}{c} \tilde{z}_0^{(j)} \\ \tilde{z}_1^{(j)} \\ \vdots \\ \tilde{z}_{L-1}^{(j)} \end{array}\right] \stackrel{\Delta}{=} \tilde{\mathbf{z}}^{(j)} = \tilde{\mathbf{A}}\left(\tilde{\mathbf{r}}-\tilde{\mathbf{S}}^{(j)}\right). \tag{10}$$

Note that the prediction error variance is given by:

$$\sigma^2_{e,\,k} \stackrel{\Delta}{=} \frac{1}{2}E\left[\tilde{z}_k^{(j)}\left(\tilde{z}_k^{(j)}\right)^*\right]. \tag{11}$$

When $\tilde{\mathbf{w}}$ consists of samples from a P^{th}-order AR process, then it is easy to see that a P^{th}-order prediction filter is sufficient to completely decorrelate the elements of $\tilde{\mathbf{w}}$. In this situation we have

$$\sigma^2_{e,\,k} = \sigma^2_{e,\,P} \qquad \text{for } k \ge P. \tag{12}$$

In practical situations, we would in any case use a finite order prediction filter.

3 Predictive Iterative Decoding in Coloured Noise

In this section we make use of the prediction filter approach derived in Section 2, to arrive at the predictive iterative decoding of turbo codes in coloured noise. In this section all variables are real-valued (note the absence of tilde), since we are considering binary turbo codes. The concept is later extended to QPSK turbo codes. We begin by stating some assumptions and modifications over the conventional turbo encoder, that are necessary, to handle coloured noise.

Firstly, we note that the *interleaved* version of the uncoded (systematic) symbol sequence *must* also be transmitted. If the interleaved version is not transmitted, the second decoder would have to use the interleaved version of the systematic bits, which would decorrelate the noise, and hence the prediction filter in the second decoder would be ineffective. The implication of the above discussion is that the code-rate would now be $1/4$ instead of $1/3$. The block diagram of the transmitter is shown in Figure 1.

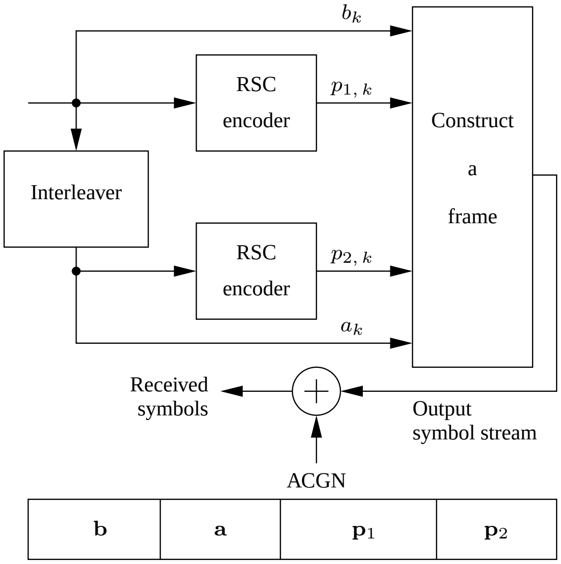


Figure 1. Block diagram of the transmitter.

The second important observation is that puncturing *may not be* the best way to increase the code-rate or decrease the transmission bandwidth. We may have to resort to M-ary turbo coding (instead of binary) [8, 9, 10, 11] to reduce the transmission bandwidth. We later show that the code-rate can be increased to $1/2$ by using QPSK.

Thirdly, the decoder trellis must be modified to account for the additional memory due to the prediction filter. In fact, the decoder trellis for binary turbo codes, becomes a *super-trellis* containing $N \times 2^P$ states, where P is the order of the prediction filter and N is the number of states in the encoder trellis.

Finally, we assume that the noise is wide sense stationary. Hence, the noise correlation is the same for the systematic symbols and the parity symbols. In most practical situations, this is a reasonable assumption. Based on the above discussion and on the derivation in Section 2, we are now ready to derive the iterative decoding equations for coloured noise. We also assume for simplicity that a P^{th}-order prediction filter completely decorrelates the noise.

Assume that a received block of $4L$ symbols needs to be decoded. The block has L uncoded symbols, $2L$ parity symbols and the interleaved version of the L uncoded symbols. The first decoder computes the *a posteriori* probabilities:

$$P\left(b_k = +1|\mathbf{r}_1\right) \text{ and } P\left(b_k = -1|\mathbf{r}_1\right) \tag{13}$$

for $0 \le k \le L-1$, where $b_k \in \{\pm 1\}$ is the uncoded symbol at the k^{th} time instant and $\mathbf{r}_1$ is a $2 \times L$ matrix denoted by:

$$\mathbf{r}_1 = \left[\begin{array}{ccc} r_{b,\,0} & \ldots & r_{b,\,L-1} \\ r_{p1,\,0} & \ldots & r_{p1,\,L-1} \end{array}\right]. \tag{14}$$

In the above equation, $r_{b,\,k}$ and $r_{p1,\,k}$ respectively denote the received samples corresponding to the uncoded symbol and the parity symbol emanating from the first encoder, at time k. We assume that the noise autocorrelation has a finite span, hence due to the frame structure shown in Figure 1, the noise affecting the data and the parity are uncorrelated. Now (for $0 \leq k \leq L-1$)

$$\begin{aligned} P\left(b_k = +1 | \mathbf{r}_1\right) &= \frac{p\left(\mathbf{r}_1 | b_k = +1\right) P\left(b_k = +1\right)}{p(\mathbf{r}_1)} \\ P\left(b_k = -1 | \mathbf{r}_1\right) &= \frac{p\left(\mathbf{r}_1 | b_k = -1\right) P\left(b_k = -1\right)}{p(\mathbf{r}_1)} \end{aligned} \tag{15}$$

where $p(\cdot)$ denotes the probability density function and $P(b_k = +1)$ denotes the *a priori* probability that $b_k = +1$. Noting that $p(\mathbf{r})$ is a constant and can be ignored, the above equation can be written as:

$$\begin{aligned} P\left(b_k = +1 | \mathbf{r}_1\right) &= S_{k+} P\left(b_k = +1\right) \\ P\left(b_k = -1 | \mathbf{r}_1\right) &= S_{k-} P\left(b_k = -1\right) \end{aligned} \tag{16}$$

where

$$\begin{aligned} S_{k+} &= \sum_{j=1}^{2^{L-1}} p\left(\mathbf{r}_1 | \mathbf{b}_{k+}^{(j)}\right) P_{\bar{k}}^{(j)} \\ S_{k-} &= \sum_{j=1}^{2^{L-1}} p\left(\mathbf{r}_1 | \mathbf{b}_{k-}^{(j)}\right) P_{\bar{k}}^{(j)} \end{aligned} \tag{17}$$

where the $1 \times L$ vectors

$$\begin{aligned} \mathbf{b}_{k+}^{(j)} &= \left[\begin{array}{ccccc} b_1^{(j)} & \ldots & +1 & \ldots & b_L^{(j)} \end{array}\right] \\ \mathbf{b}_{k-}^{(j)} &= \left[\begin{array}{ccccc} b_1^{(j)} & \ldots & -1 & \ldots & b_L^{(j)} \end{array}\right] \end{aligned} \tag{18}$$

are constrained such that the k^{th} symbol is $+1$ or -1 respectively, for all j. We wish to emphasize that all the elements of $\mathbf{b}_{k+}^{(j)}$ and $\mathbf{b}_{k-}^{(j)}$ are identical excepting for the k^{th} element. Assuming that the uncoded symbols occur independently

$$P_{\bar{k}}^{(j)} = \prod_{\substack{i=0 \\ i \neq k}}^{L-1} P\left(b_i^{(j)}\right). \tag{19}$$

Expanding the conditional pdf in (17) we get

$$\begin{aligned} S_{k+} &= \sum_{j=1}^{2^{L-1}} \prod_{i=0}^{L-1} \gamma_{\mathrm{in},\, i}^{(j)} \gamma_{\mathrm{ex},\, i,\, k+}^{(j)} P_{\bar{k}}^{(j)} \\ S_{k-} &= \sum_{j=1}^{2^{L-1}} \prod_{i=0}^{L-1} \gamma_{\mathrm{in},\, i}^{(j)} \gamma_{\mathrm{ex},\, i,\, k-}^{(j)} P_{\bar{k}}^{(j)} \end{aligned} \tag{20}$$

where $\gamma_{\mathrm{in},\, i}^{(j)}$ denotes *intrinsic* information, $\gamma_{\mathrm{ex},\, i,\, k+}^{(j)}$ and $\gamma_{\mathrm{ex},\, i,\, k-}^{(j)}$ denote *extrinsic* information and are calculated as (from Section 2):

$$\begin{aligned} \gamma_{\mathrm{in},\, i}^{(j)} &= \exp\left[-\frac{\left(z_{b,\, i}^{(j)}\right)^2}{2\sigma_Q^2}\right] \\ \gamma_{\mathrm{ex},\, i,\, k+}^{(j)} &= \exp\left[-\frac{\left(z_{p1,\, i,\, k+}^{(j)}\right)^2}{2\sigma_Q^2}\right] \\ \gamma_{\mathrm{ex},\, i,\, k-}^{(j)} &= \exp\left[-\frac{\left(z_{p1,\, i,\, k-}^{(j)}\right)^2}{2\sigma_Q^2}\right] \end{aligned} \tag{21}$$

where σ_Q^2 is the noise variance at the output of the optimal Q^{th}-order predictor, $a_{Q,\, l}$ is the l^{th} coefficient of the optimal Q^{th}-order predictor and

$$\begin{aligned} Q &= \min(i,\, P) \\ z_{b,\, i}^{(j)} &= \sum_{l=0}^{Q} a_{Q,\, l}\left(r_{b,\, i-l} - b_{i-l}^{(j)}\right) \\ z_{p1,\, i,\, k+}^{(j)} &= \sum_{l=0}^{Q} a_{Q,\, l}\left(r_{p1,\, i-l} - p_{1,\, i-l,\, k+}^{(j)}\right) \\ z_{p1,\, i,\, k-}^{(j)} &= \sum_{l=0}^{Q} a_{Q,\, l}\left(r_{p1,\, i-l} - p_{1,\, i-l,\, k-}^{(j)}\right). \end{aligned} \tag{22}$$

It is understood that when S_{k+} is being computed, $b_k^{(j)}$ is set to $+1$ for all j. Likewise, when S_{k-} is being computed, $b_k^{(j)}$ is set to -1 for all j. The terms $p_{1,\, i,\, k+}^{(j)}$ and $p_{1,\, i,\, k-}^{(j)}$ are used to denote the parity sequence that is generated by the first encoder when $b_k^{(j)}$ is a $+1$ or a -1 respectively.

Finally, the *a posteriori* probabilities that are to be fed as *a priori* probabilities to the second decoder, are computed as F_{k+} and F_{k-} as follows:

$$\begin{aligned} F_{k+} &= S_{k+}/(S_{k+} + S_{k-}) \\ F_{k-} &= S_{k-}/(S_{k+} + S_{k-}). \end{aligned} \tag{23}$$

Note that F_{k+} and F_{k-} are the *normalized* values of S_{k+} and S_{k-} respectively, such that they satisfy the fundamental law of probability:

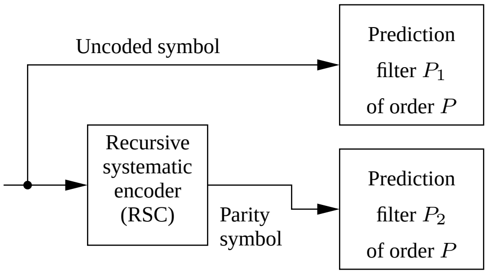
$$F_{k+} + F_{k-} = 1. \tag{24}$$

The equations for the second decoder are identical, excepting that $\mathbf{r}_2$ is used instead of $\mathbf{r}_1$, where

$$\mathbf{r}_2 = \left[\begin{array}{ccc} r_{a,\, 0} & \ldots & r_{a,\, L-1} \\ r_{p2,\, 0} & \ldots & r_{p2,\, L-1} \end{array}\right]. \tag{25}$$

It is clear that a straightforward computation of S_{k+} and S_{k-} as given in (20) would be prohibitively expensive. Let us now turn our attention to the efficient computation of S_{k+} and S_{k-}.

Note that S_{k+} and S_{k-} are nothing but a sum-of-products (SOP) and can be efficiently computed by using the trellis. However, we need to keep in mind that due to the additional memory of the prediction filter, we can no longer use the encoder trellis for optimal decoding (the encoder trellis could still be used, but we expect this approach to be suboptimal). In fact, we need to construct a


Uncoded symbol
Prediction filter P_1 of order P
Recursive systematic encoder (RSC)
Parity symbol
Prediction filter P_2 of order P

Encoder state: E_s $(0 \leq E_s \leq N-1)$
State of P_1: P_{1s} $(0 \leq P_{1s} \leq 2^P - 1)$
Super trellis state: $E_s \times 2^P + P_{1s}$

Figure 2. Computation of the super-trellis state from the encoder state and the contents of the prediction filter P_1.

supertrellis. Let N denote the number of states in the encoder trellis. Then the number of supertrellis states would be $M = N \times 2^P$ (and not $N \times 4^P$, since the parity sequence is dependent on the information sequence). The computation of a supertrellis state is shown in Figure 2. Given the encoder state and the state of P_1, the state of P_2 (which is nothing but the sequence of parity symbols corresponding to the sequence of information symbols in P_1) can be uniquely found out. Thus, given a supertrellis state, we can find out the encoder state, the state of P_1 and the state of P_2.

Let $\mathscr{D}_n$ denote the set of states that diverge from state n. For example

$$\mathscr{D}_0 = \{0,\, 3\} \tag{26}$$

implies that states 0 and 3 can be reached from state 0. Similarly, let $\mathscr{C}_n$ denote the set of states that converge to state n. Let $\alpha_{i,\,n}$ denote the forward SOP at time i $(0 \leq i \leq L-2)$ at state n $(0 \leq n \leq M-1)$. Then the forward SOP can be recursively computed as follows:

$$\begin{aligned}
\alpha'_{i+1,\,n} &= \sum_{m \in \mathscr{C}_n} \alpha_{i,\,m}\gamma_{\text{in},\,i,\,m,\,n}\gamma_{\text{ex},\,i,\,m,\,n}P\left(b_{i,\,m,\,n}\right) \\
\alpha_{0,\,n} &= 1/M \qquad \text{for } 0 \leq n \leq M-1 \\
\alpha_{i+1,\,n} &= \alpha'_{i+1,\,n}\Big/\left(\sum_{n=0}^{M-1} \alpha'_{i+1,\,n}\right)
\end{aligned} \tag{27}$$

where $P(b_{i,\,m,\,n})$ denotes the *a priori* probability of the uncoded symbol corresponding to the transition from superstate m to superstate n, at time i and

$$\begin{aligned}
\gamma_{\text{in},\,i,\,m,\,n} &= \exp\left[-\frac{\left(z_{b,\,i,\,m,\,n}\right)^2}{2\sigma_Q^2}\right] \\
\gamma_{\text{ex},\,i,\,m,\,n} &= \exp\left[-\frac{\left(z_{p,\,i,\,m,\,n}\right)^2}{2\sigma_Q^2}\right]
\end{aligned} \tag{28}$$

where

$$\begin{aligned}
z_{b,\,i,\,m,\,n} &= \sum_{l=0}^{Q} a_{Q,\,l}\left(r_{b,\,i-l} - b_{l,\,m,\,n}\right) \\
z_{p,\,i,\,m,\,n} &= \sum_{l=0}^{Q} a_{Q,\,l}\left(r_{p1,\,i-l} - p_{l,\,m,\,n}\right).
\end{aligned} \tag{29}$$

The term $b_{0,\,m,\,n} \in \pm 1$ denotes the uncoded input symbol corresponding to the transition from superstate m to superstate n. Similarly, the term $p_{0,\,m,\,n} \in \pm 1$ denotes the parity symbol corresponding to the transition from superstate m to superstate n. The other terms $b_{l,\,m,\,n}$ and $p_{l,\,m,\,n}$, for $1 \leq l \leq Q$, are "extracted" from superstate m. Thus, given a superstate m and an input symbol $b_{0,\,m,\,n}$, we obtain a parity symbol $p_{0,\,m,\,n}$ and reach the next superstate n. The normalization step in the last equation of (27) is done to prevent numerical instabilities.

Similarly, let $\beta_{i,\,n}$ denote the backward SOP at time i $(1 \leq i \leq L-1)$ at state n $(0 \leq n \leq M-1)$. Then the recursion for the backward SOP can be written as:

$$\begin{aligned}
\beta'_{i,\,n} &= \sum_{m \in \mathscr{D}_n} \beta_{i+1,\,m}\gamma_{\text{in},\,i,\,n,\,m}\gamma_{\text{ex},\,i,\,n,\,m}P\left(b_{i,\,n,\,m}\right) \\
\beta_{L,\,n} &= 1/M \qquad \text{for } 1 \leq n \leq M \\
\beta_{i,\,n} &= \beta'_{i,\,n}\Big/\left(\sum_{n=0}^{M-1} \beta'_{i,\,n}\right).
\end{aligned} \tag{30}$$

Once again, the normalization step in the last equation of (30) is done to prevent numerical instabilities.

Let $\rho^+(n)$ denote the state that is reached from state n when the input symbol is $+1$. Similarly let $\rho^-(n)$ denote the state that can be reached from state n when the input symbol is -1. Then for $0 \leq k \leq L-1$ we have

$$\begin{aligned}
S_{k+} &= \sum_{n=0}^{M-1} \alpha_{k,\,n}\gamma_{\text{ex},\,k,\,n,\,\rho^+(n)}\beta_{k+1,\,\rho^+(n)} \\
S_{k-} &= \sum_{n=0}^{M-1} \alpha_{k,\,n}\gamma_{\text{ex},\,k,\,n,\,\rho^-(n)}\beta_{k+1,\,\rho^-(n)}.
\end{aligned} \tag{31}$$

Equations (27), (30) and (31) constitute the MAP recursions for the first decoder. The MAP recursions for the second decoder are similar.

4 Simulation Results

Figure 3 shows the simulation model used for comparing the conventional iterative decoder with the predictive iterative decoder (PID) in ACGN (additive coloured Gaussian noise). All simulation results for the PID use the optimum prediction filter for the given order. Note that in general a finite-order predictor is suboptimum compared to the infinite-order predictor. However, most of the performance improvement is obtained by using a 1st or 2nd-order predictor. We first present the simulation results for BPSK and

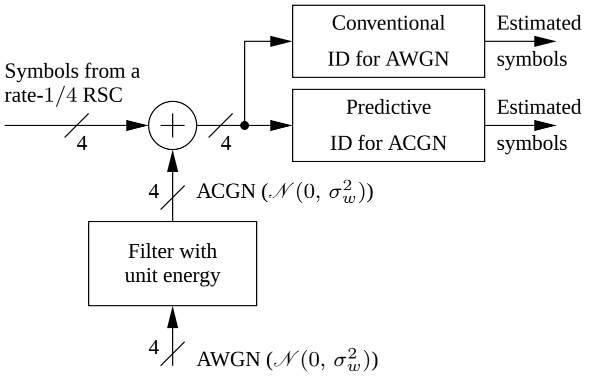


Figure 3. Simulation model for comparing the conventional iterative decoder (CID) and the predictive iterative decoder (PID).

then for QPSK. The framesize was taken to be 10^3 bits and the simulations were carried out over 10^4 frames. The generator matrix for the encoder is given by:

$$G(D) = \left[\ 1 \quad \frac{1+D^2+D^3+D^4}{1+D+D^4}\ \right] \tag{32}$$

Random interleaving was used.

For BPSK we considered two different noise power spectral densities (psd). The first kind of psd is obtained by passing AWGN through a 1st-order IIR filter given by:

$$w_n = u_n\sqrt{1-a^2} + a w_{n-1}. \tag{33}$$

In this case, it is easy to see that a 1st-order prediction filter of the form $z_n = w_n - a w_{n-1}$ completely decorrelates the noise and is the *optimum* predictor. The simulation results corresponding to the filter in (33) is shown in Figure 4. We have defined the SNR as

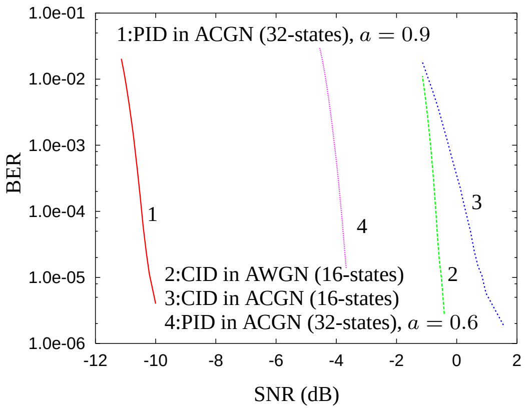


Figure 4. Simulation results for coloured noise obtained from a 1st-order IIR filter with $a = 0.9$ and $a = 0.6$.

$$\text{SNR} = 10\log_{10}\left(\frac{1}{\sigma_w^2}\right) \quad \text{dB}. \tag{34}$$

We see that there is 11 dB improvement in the PID over the CID for ACGN channels for $a = 0.9$. For $a = 0.6$, the improvement over the CID for ACGN is about 5 dB. Thus we conclude that the improvement depends on the noise correlation. However, the PID (using the 1st-order prediction filter) requires a 32-state super-trellis compared to the 16-state encoder trellis required by the CID.

The 2nd kind of psd is obtained by passing noise through an FIR filter having an impulse response

$$h_n = \frac{n+1}{\sqrt{91}} \qquad \text{for } 0 \le n \le 5. \tag{35}$$

In this case the order of the prediction filter required to completely decorrelate the noise is greater than unity. However, we see from Figure 5 that the PID using a 1st-order prediction filter (32-state super-trellis) obtains most of the performance improvement. The PID using a 2nd-order prediction filter (64-state super-trellis) is only marginally better than the one using a 1st-order prediction filter. We also notice that the PID performs better than the CID by about 6 dB.

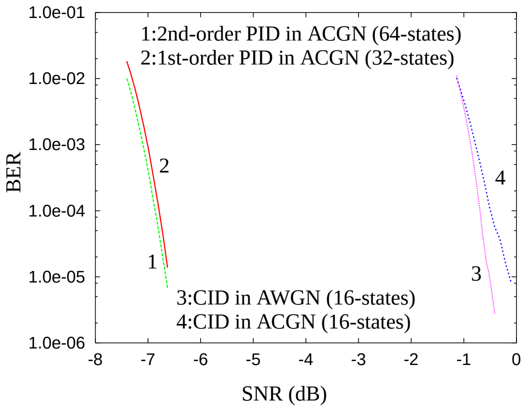


Figure 5. Simulation results for coloured noise obtained from the FIR filter in (35).

Next, we address the issue of improving the code-rate to $1/2$. This can be done by transmitting QPSK instead of BPSK, that is, symbols b_k and p_{1k} in Figure 1 constitute a QPSK symbol, a_k and p_{2k} constitute the other QPSK symbol. Thus two QPSK symbols are transmitted in one uncoded bit duration, hence the code-rate is $1/2$. Though we have not derived the PID for complex symbols in this paper, we nevertheless present the simulation results. The impulse response of the FIR filter used to generate coloured noise was taken as:

$$\tilde{h}_n = \frac{1}{\sqrt{1300}}\left[n+1+\mathrm{j}\,(12-n)\right] \qquad \text{for } 0 \le n \le 11. \tag{36}$$

The performance of the PID using a 1st-order predictor and the CID is shown in Figure 6. We see that there is 10 dB improvement in performance. Increasing the predictor order to two did not yield any noticeable improvement.

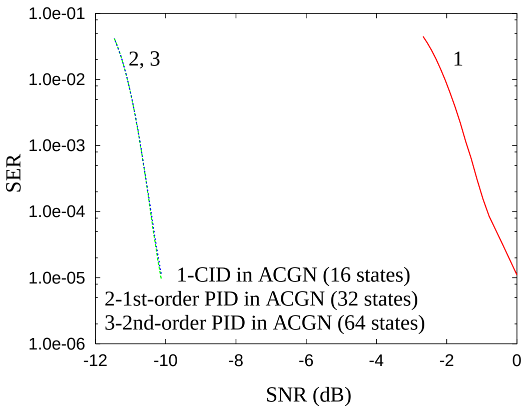


Figure 6. Simulation results for rate-$1/2$ (QPSK) turbo codes in coloured noise obtained from the FIR filter in (36).

5 Conclusions and Future Work

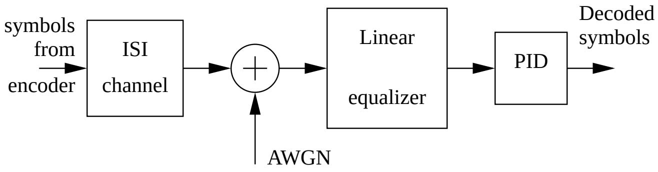


Figure 7. Block diagram of the proposed application of the predictive iterative decoder.

An optimum detector for additive coloured Gaussian noise (ACGN) channels was derived. This result was extended to iterative decoding to obtain the predictive iterative decoder (PID) for ACGN channels. The PID for a rate-$1/4$ turbo code (BPSK signalling) was derived. Finally simulation results were presented to demonstrate the improvement of the PID over the conventional iterative decoder for binary and 4-ary turbo codes. In the future work we would like to investigate whether there is any performance degradation if the encoder trellis is used for decoding instead of the supertrellis. We would also apply the PID to decode symbols in ISI channels. This can be done by using a linear equalizer in front of the PID as shown in Figure 7. Since the noise at the linear equalizer output is correlated, this seems to be an ideal application of the PID. The present work seems to be a good alternative to the turbo equalization techniques presented in [12] where the MAP equalizer is embedded in the iterative decoder.

References

[1] C. Berrou and A. Glavieux, "Near Optimum Error Correcting Coding and Decoding – Turbo Codes," *IEEE Trans. on Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[2] P. Frenger, "Turbo Decoding for Wireless Systems with Imperfect Channel Estimates," *IEEE Trans. on Commun.*, vol. 48, no. 9, pp. 1437–1440, Sept. 2000.

[3] G. Colavolpe, G. Ferrari, and R. Raheli, "Noncoherent Iterative (Turbo) Decoding," *IEEE Trans. on Commun.*, vol. 48, no. 9, pp. 1488–1498, Sept. 2000.

[4] Jonathan D. Coker, Evangelos Eleftheriou, Richard L. Galbraith, and Walter Hirt, "Noise Predictive Maximum Likelihood (NPML) Detection," *IEEE Trans. on Magnetics*, vol. 34, no. 1, pp. 110–117, Jan. 1998.

[5] K. Vasudevan, "Detection of Signals in Correlated Interference Using a Predictive VA," in *Proc. of the IEEE International Conference on Communication Systems*, Singapore, Nov. 2002.

[6] K. Vasudevan, "Detection of Signals in Correlated Interference using a Predictive VA," 2004, Accepted for publication in the Signal Processing Journal, Elsevier Science.

[7] K. Vasudevan, K. Giridhar, and Bhaskar Ramamurthi, "An Efficient Suboptimum Detector Based on Linear Prediction in Rayleigh Flat-Fading Channels," *Signal Processing Journal, Elsevier Science*, vol. 81, no. 4, pp. 819–828, April 2001.

[8] S. Le Goff, A. Glavieux, and C. Berrou, "Turbo Codes and High Spectral Efficiency Modulation," in *Proc. of the Intl. Conf. on Commun.*, 1994, pp. 645–649.

[9] P. Robertson and T. Wörz, "Coded Modulation Scheme Using Turbo Codes," *Electronics Letters*, vol. 31, no. 18, pp. 1546–1547, Aug. 1995.

[10] P. Robertson and T. Wörz, "Bandwidth Efficient Turbo Trellis Coded Modulation Using Punctured Component Codes," *IEEE J. on Select. Areas in Commun.*, vol. 16, no. 2, pp. 206–218, Feb. 1998.

[11] A. Chouly and O. Pothier, "A Non-Pragmatic Approach to Turbo Trellis Coded Modulations," in *Proc. of the Intl. Conf. on Commun.*, 2002, pp. 1327–1331.

[12] M. Tüchler, R. Koetter, and A. C. Singer, "Turbo Equalization: Principles and New Results," *IEEE Trans. on Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.