TURBO CODES FOR FADING AND BURST CHANNELS *

Joseph H. Kang, Wayne E. Stark, and Alfred O. Hero
Dept. of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor

Abstract

In this paper, turbo codes are investigated in fading and burst channels. In particular, we consider the design and performance of turbo codes for a Gilbert-Elliot burst channel and in a realistic fading channel. For both cases, our approach is to compute estimates of unknown channel state parameters and use these in the turbo decoder. For the burst channel model, calculation of state estimates requires knowledge of the hidden Markov model (HMM) transition probabilities. When these probabilities are unknown, the Baum-Welch reestimation procedure is used.

1 Introduction

Recently, turbo codes have received a lot of notoriety as they have been shown to achieve data communication at signal-to-noise ratios close to the Shannon limit. The excellent performance of turbo codes explains why much of the current research is focused on applying turbo codes to different systems.

Channel coding schemes have generally been designed to increase the reliability of information transmission when the errors are statistically independent. However, many channels such as multipath and fading which exhibit bursts of errors. A common method for dealing with these bursts is to interleave the information in such a manner that the channel appears memoryless. Thus if interleaving is applied to a burst channel, a code devised for independent errors can be applied. However, such a scheme does not make use of the information inherent in the memory. Because multiple bits have been transmitted over similar channel conditions, it might be useful to estimate the channel state and use this information in the decoder. This was the approach taken in [2] where turbo codes in a frequency-hopped spread spectrum (FH-SS) system were considered. In [2], the channel consisted of an on-off jammer in a FH-SS system. The approach was to exploit the channel memory by estimating whether the jammer was on or off.

In this paper, we will use the information inherent in the memory by estimating the state of the Gilbert-Elliot burst channel and the instantaneous amplitudes for the realistic fading channel. These estimates will then be used in the turbo decoder.

*This research is supported in part by the Army Research Office under grants DAAH04-96-1-0110 and DAAH04-96-1-0377, and by DARPA under grant SRA-588510732

2 System Model

In this section, we describe the system models of the Gilbert-Elliot communications system and the FH-SS system with realistic fading. The Gilbert-Elliot system will be discussed first.

2.1 Gilbert-Elliot Burst Model

The encoder is formed using two constituent codes. The constituent codes considered in this paper are recursive systematic convolutional codes [4]. The encoder is formed by concatenating the constituent codes in parallel and then separating the codes by an interleaver. The encoder takes as input the data sequence $d_k \epsilon \{0, 1\}$ of length N and then produces three streams: the information bits d_k, the parity bits $p_{1,k}$ of the first component encoder with input d_k, and the parity bits $p_{2,k}$ of the second component encoder with interleaved d_k as input. BPSK modulation is considered with coherent demodulation.

The Gilbert-Elliot burst noise channel is a two state hidden Markov model where one state represents a bad state which generally has high error probabilities and the other state is a good state which generally has low error probabilities. This model is shown below in Figure 1, where at time k, $z_k = 1$ represents the bad state and $z_k = 0$ represents the good state. The probability of moving from state $z_k = i$ to $z_{k+1} = j$ is denoted by p_{ij}.

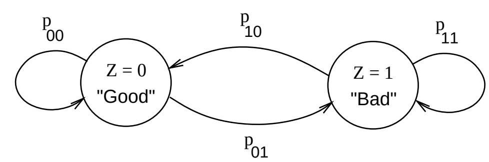

Figure 1: Hidden Markov Model of Channel

If transmission occurs over the good state at time k, the noise is additive white Gaussian noise (AWGN) with power spectral density $N_0/2$ and typically has low magnitude. Similarly, for transmission over the bad state, the noise is white Gaussian with power spectral density $N_1/2$ where $N_1 > N_0$. Let $(y_{1,k}, y_{2,k}, y_{3,k})$ be the channel outputs and let $(c_{1,k}, c_{2,k}, c_{3,k}) = ((-1)^{d_k}, (-1)^{p_{1,k}}, (-1)^{p_{2,k}})$. Thus,

$$z_{i,k} = 0 \implies y_{i,k} = \sqrt{E}\, c_{i,k} + \eta^0_{i,k}$$
$$z_{i,k} = 1 \implies y_{i,k} = \sqrt{E}\, c_{i,k} + \eta^1_{i,k}$$

where $\eta^0_{i,k} \sim N(0, N_0/2)$ and $\eta^1_{i,k} \sim N(0, N_1/2)$.

2.2 Realistic Fading

The second system we consider is a hybrid direct-sequence frequency-hopped spread spectrum (DS-FH SS) system with a realistic frequency selective fading channel. Unlike the previous channel which used hidden Markov models to model the channel state, we will use a more conventional FH-SS model described below. Often times, idealistic channel assumptions are made for analytical convenience. While these results are important, they do not necessarily mimic realistic situations closely. In this section, we discuss two systems which do not make ideal channel assumptions.

The encoder is the same as the one described in Section 2.1 where a data sequence of length N is put into the encoder and for each information bit, three coded bits are produced. These coded bits are then each spread using an L chip sequence. BPSK modulation is considered with coherent detection. The resultant signal is frequency hopped. It is assumed that the frequency hopper will choose each of the Q frequencies or subchannels with uniform probability. If R_c is the chip rate and R_b is the data rate, then $R_c = 3 * R_b * L$. The transmission bandwidth of the system is $W = Q * R_c$.

Two measured channels are considered. Pine Street (PS) is taken from an urban area and has 12 indepedent paths [5]. American Legion Drive (ALD) is taken from a suburban area and has 5 independent paths [6]. The delay spreads of ALD and PS are $1.87\,\mu s$ and $2.53\,\mu s$, respectively.

The channel model shown below takes a standard form, but the fade amplitudes, are taken from the measured channel. Similar to before, $(c_{1,k}, c_{2,k}, c_{3,k}) = ((-1)^{d_k}, (-1)^{p_{1,k}}, (-1)^{p_{2,k}})$ and $\{y_{i,k,l}\}_{l=1}^{L}$ are the L chips corresponding to each coded bit $c_{i,k}$.

$$y_{i,k,l} = \sqrt{E}\, a_{i,k,l}\, c_{i,k} + \eta_{i,k,l} \qquad (1)$$

where $a_{i,k,l}$ is the fading amplitude and $\eta_{i,k,l}$ is i.i.d. with density $N(0, N_0/2)$.

2.3 Original Turbo Decoder

Because the optimal decoder is too complex, the turbo decoder provides a suboptimal alternative which iteratively passes log-likelihood information between a pair of MAP decoders matched to each of the component encoders. The turbo decoding algorithm has been well documented in previous papers [2][3][4], thus it will not be repeated here. Of particular interest, however, are the branch transition probabilities which are needed for turbo decoder calculations. The computation of branch transition probabilities depend on the channel, so they play a key role in the design of the turbo decoder for fading and burst channel models. Let S_k be the state of the first encoder at time k. The branch transition probabilities used by the MAP algorithm are calculated as

$$\begin{aligned}\gamma_i(y_{1,k}, y_{2,k}, m', m) = &\\ &p(y_{1,k}|d_k = i, S_k = m, S_{k+1} = m') \cdot \\ &p(y_{2,k}|d_k = i, S_k = m, S_{k+1} = m') \cdot \\ &P(S_{k+1} = m'|d_k = i, S_k = m) \cdot \\ &P(d_k = i|S_k = m) \end{aligned} \qquad (2)$$

where $P(S_{k+1} = m'|d_k = i, S_k = m) = 1$ if bit i is associated with the given state transition and equals 0 if it is not. $P(d_k = i|S_k = m) = p(d_k = i)$ depends on the *a priori* probabilities of the information bits.

3 Turbo Decoder for Gilbert-Elliot Channel

We will first consider the modifications to the turbo decoder necessary for the Gilbert-Elliot burst channel model. The turbo decoding algorithm is dependent on what information is available to the turbo decoder. This paper considers three cases: known channel state; unknown channel state but known HMM transition probabilities p_{ij}; and finally, unknown channel state and unknown p_{ij}.

If the state, $z_{i,k}$, is known, then the modification to the turbo decoder is straightforward. The decoder can simply use the relevant noise variance to calculate the branch transition probabilities. Thus, (2) can be calculated using

$$\begin{aligned}&p(y_{i,k}|d_k = i, S_k = m, S_{k+1} = m', z_{i,k} = z)\\ &\quad = \frac{1}{\sqrt{\pi N_z}} e^{-\frac{1}{N_z}(y_{i,k} - c_{i,k})^2}.\end{aligned} \qquad (3)$$

If the channel state is unknown, but the transition probabilities are known, then (2) can be calculated by invoking total probability with respect to the channel state.

$$\begin{aligned}&p(y_{i,k}|d_k = i, S_k = m, S_{k+1} = m')\\ &\quad = \frac{1}{\sqrt{\pi N_1}} e^{-\frac{1}{N_1}(y_{i,k} - c_{i,k})^2} \cdot p(z_{i,k} = 1) + \\ &\quad\quad \frac{1}{\sqrt{\pi N_0}} e^{-\frac{1}{N_0}(y_{i,k} - c_{i,k})^2} \cdot p(z_{i,k} = 0)\end{aligned} \qquad (4)$$

Note that $p(z_{i,k}=z)$ is not known. One possibility is to use the steady state probability $\lim_{k\to\infty} p(z_{i,k}=z)$. This can be solved using $\underline{v}=\underline{v}P$, $\sum_j v_j = 1$, and setting $p(z_{i,k}=j)=v_j$ for j = 0,1. Another possibility is to estimate the probability of being in each state given the received sequence and the HMM transition probabilities. For $k=1$,

$$p(z_{i,1}=0|y_{i,1}) = \frac{p(y_{i,1}|z_{i,1}=0)\,p(z_{i,1}=0)}{p(y_{i,1})} \tag{5}$$

where $p(z_{i,1}=0)$ is set to the steady state probability v_0. For $k\geq 2$,

$$\begin{aligned} p(z_{i,k} &= 0|y_{i,1},...,y_{i,k}) \\ &= \frac{p(y_{i,k}|z_{i,k}=0,y_{i,1},..,y_{i,k-1})}{p(y_{i,1},..,y_{i,k})}\cdot \\ &\quad p(z_{i,k}=0,y_{i,1},..,y_{i,k-1}) \qquad (6)\\ &\approx \frac{p(y_{i,k}|z_{i,k}=0)\,p(z_{i,k}=0,y_{i,1},..,y_{i,k-1})}{p(y_{i,1},..,y_{i,k})} \qquad (7) \end{aligned}$$

where (7) is approximate since the sequence $\{y_{i,j}\}_{j=1}^{k}$ is lightly correlated. Furthermore, (7) is computed using

$$\begin{aligned} &p(z_{i,k}=0,y_{i,1},...,y_{i,k-1}) \\ &= \sum_{z=0}^{1} p(z_{i,k}=0,z_{i,k-1}=z,y_{i,1},...,y_{i,k-1}) \qquad (8)\\ &= \sum_{z=0}^{1} p(y_{i,k-1}|z_{i,k}=0,z_{i,k-1}=z,y_{i,1},...,y_{i,k-2})\cdot \\ &\quad p(z_{i,k}=0,z_{i,k-1}=z,y_{i,1},...,y_{i,k-2}) \qquad (9)\\ &= \sum_{z=0}^{1} p(y_{i,k-1}|z_{i,k-1}=z)\cdot \\ &\quad p(z_{i,k}=0,z_{i,k-1}=z,y_{i,1},...,y_{i,k-2}) \qquad (10) \end{aligned}$$

and (10) is computed using

$$\begin{aligned} p(z_{i,k} &= a,z_{i,k-1}=b,y_{i,1},...,y_{i,k-2}) \\ &= p(y_{i,1},...,y_{i,k-2}|z_{i,k}=a,z_{i,k-1}=b)\cdot \\ &\quad p(z_{i,k}=a,z_{i,k-1}=b) \qquad (11)\\ &= p(y_{i,1},...,y_{i,k-2}|z_{i,k-1}=b)\cdot \\ &\quad p(z_{i,k}=a|z_{i,k-1}=b)\,p(z_{i,k-1}=b) \qquad (12)\\ &= \frac{p(z_{i,k-1}=b,y_{i,1},...,y_{i,k-2})}{p(z_{i,k-1}=b)}\cdot \\ &\quad p(z_{i,k}=a|z_{i,k-1}=b)\,p(z_{i,k-1}=b) \qquad (13)\\ &= p(z_{i,k-1}=b,y_{i,1},...,y_{i,k-2})\,p_{ba}. \qquad (14) \end{aligned}$$

Combining (10) and (14),

$$\begin{aligned} p(z_{i,k} &= 0,y_{i,1},...,y_{i,k-1}) \\ &= \sum_{z=0}^{1} p(y_{i,k-1}|z_{i,k-1}=z)\cdot \\ &\quad p(z_{i,k-1}=z,y_{i,1},...,y_{i,k-2})\,p_{z0} \qquad (15) \end{aligned}$$

Note that due to the recursive nature of (15), (7) can be computed efficiently.

If the HMM transition probabilities are unknown, it is necessary to estimate the transition probabilities of the chain. Once the transition probabilities have been adaptively estimated, the states can be estimated and then used by the turbo decoder, as seen above. Thus the problem is to find the HMM model which maximizes the probabilities of the observation sequence. The Baum-Welch reestimation procedure yields an ML estimate of the HMM which is a locally optimal solution. However, because there is no globally optimal solution to this problem, this is the best we can do.

4 Turbo Decoder for Realistic Fading

Because the turbo decoding algorithm is dependent on what information is available to the decoder, we will again consider multiple cases. In the first case, we will assume that the fading amplitudes are perfectly known to the decoder. In the second case, such side information is unavailable and thus needs to be estimated.

The first case is the one where fading side information (SI) is available to the decoder. For the case of diversity, maximum ratio combining is optimal. If we let

$$x_{i,k} = \sum_{l=1}^{L} a_{i,k,l}\,y_{i,k,l} \tag{16}$$

then $p(x_{i,k}|\{a_{i,k,l}\}_{l=1}^{L})$ has density $N(\mu_{x_{i,k}},\sigma^2_{x_{i,k}})$ where

$$\mu_{x_{i,k}} = c_{i,k}\sum_{l=1}^{L} a^2_{i,k,l} \tag{17}$$

$$\sigma^2_{x_{i,k}} = \frac{N_0}{2}\sum_{l=1}^{L} a^2_{i,k,l}. \tag{18}$$

Thus, (2) can be computed using

$$\begin{aligned} &p(x_{i,k}|\{a_{i,k,l}\}_{l=1}^{L},d_k=i,S_k=m,S_{k+1}=m') \\ &= \frac{1}{\sqrt{2\pi\sigma^2_{x_{i,k}}}}e^{-\frac{(x_{i,k}-\mu_{x_{i,k}})^2}{2\sigma^2_{x_{i,k}}}}. \qquad (19) \end{aligned}$$

For the second case, there is no fading side information (NSI) available to the decoder. As before in the Gilbert-Elliot channel, our approach is to use the information inherent in the memory to compute channel state estimates. In this case, the instantaneous fade amplitudes will be estimated. For the realistic slow fading channel, instantaneous fade amplitudes will change slowly over a given hop. If we assume that the rate of change is slow enough to be considered constant, we can calculate fading estimates in a manner analogous to the way jamming state estimates were computed in [2].

Before, we denoted the fade amplitudes as $a_{i,k,l}$. For notational simplicity, let us denote the fade amplitudes as a_k for $k = 1, ..., 3*L*N/h$ where $3*L*N$ denotes the total number of chips per packet and h is the number of chips per hop. It is assumed that the fade level remains constant over a hop, so that the total number of fade levels is equivalent to the total number of hops. Let $\underline{R}$ be the vector of received channel outputs that is available to the MAP decoder, $\underline{R}_k$ be the subset of $\underline{R}$ that has been received with the same fading amplitude a_k, and $\tilde{\underline{R}}_k$ be the subset of $\underline{R}$ that has not been received with the same fading amplitude a_k. Consider the quantization of the fading amplitudes into M regions, $B_1, ..., B_M$, and the corresponding M output levels or centroids, $l_1, ..., l_M$ where $l_i \epsilon B_i$. Then, the calculation of *a posteriori* fading probabilities is as follows for $i = 1, ..., M$.

$$p(a_k \epsilon B_i | \underline{R}) = \frac{p(\underline{R} | a_k \epsilon B_i) \cdot p(a_k \epsilon B_i)}{p(\underline{R})} \tag{20}$$

$$= p(\underline{R}_k | a_k \epsilon B_i) \cdot p(a_k \epsilon B_i) \cdot \frac{p(\tilde{\underline{R}}_k)}{p(\underline{R})} \tag{21}$$

$$= p(\underline{R}_k | a_k \epsilon B_i) \cdot p(a_k \epsilon B_i) \cdot K \tag{22}$$

where K is a normalizing factor chosen to make the probability density function sum to 1.

Note that as h, the number of chips per hop, increases, the complexity of directly computing $p(\underline{R}_k | a_k \epsilon B_i)$ rises exponentially. To overcome this problem, the following recursion was developed.

$$p(a_k \epsilon B_i | R_{k,1}, R_{k,2}, ..., R_{k,i-1}, R_{k,i})$$
$$= \frac{p(R_{k,i} | a_k \epsilon B_i, R_{k,1}, ..., R_{k,i-1})}{p(R_{k,i} | R_{k,1}, ..., R_{k,i-1})} \cdot p(a_k \epsilon B_i | R_{k,1}, ..., R_{k,i-1}) \tag{23}$$
$$= \frac{p(R_{k,i} | a_k \epsilon B_i)\, p(a_k \epsilon B_i | R_{k,1}, ..., R_{k,i-1})}{p(R_{k,i} | R_{k,1}, ..., R_{k,i-1})} \tag{24}$$

Once the fading estimates have been computed, maximum ratio combining can be performed using the estimated fading value, $\tilde{a}_{i,k,l}$.

$$\tilde{a}_{i,k,m} = \sum_{i=1}^{M} l_i \, p(a_{i,k,m} \epsilon B_i | \underline{R}) \tag{25}$$

5 Simulation Results

In this section, we present the simulation results for the Gilbert-Elliot channel and the realistic fading channel. For all simulations, the two component encoders were rate $\frac{1}{2}$ convolutional encoders with memory 4 and octal generators $(37, 21)$. Each block had 1920 information bits and a total of 8 turbo decoding iterations were used.

For simulations of the Gilbert-Elliot burst channel, the SNR of the good state was set to 4 dB for all realizations.

Figure 2 shows the BER performance of the system when the channel state is known for different values of p_{ij}. This set of performance curves serve as a reference from which the following cases can be based.

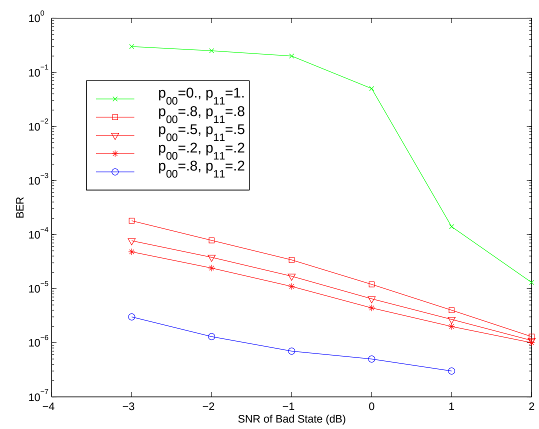

Figure 2: Plot for Known Channel State

Note that the average received SNR is not the same for each of the systems in Figure 2. The average received SNR is $10 \log \frac{E_b}{v_0 N_0 + v_1 N_1}$ where v_i represents the steady state probability of being in state i. For the first system ($p_{00} = 0$, $p_{11} = 1.$), $v_0 = 0$ and $v_1 = 1$. For the last system ($p_{00} = .8$, $p_{11} = .2$), $v_0 = .8$ and $v_1 = .2$. For the middle three systems, $v_0 = v_1 = .5$. If we assume that $N_1 > N_0$, then for a given N_0 and N_1, the first system has the lowest average SNR and the second system has the highest average SNR. The average SNR of the middle three systems is the same and their average SNR is between those of the first and last systems. This explains the difference in performance between the three clusters of curves (grouped by v_0 and v_1). Among the curves with $v_0 = v_1 = .5$, as expected, performance degrades if the sequence of states is more bursty (i.e. higher values of p_{00} and p_{11}).

The simulation results for the case of known HMM transition probabilities but unknown state is shown in Figures 3 and 4. At low SNRs, the *a posteriori* state estimation method performs better than the steady state method as the decoder successfully uses the information inherent in the memory of the channel. There is, however, still a large gap between these two cases and the known channel state case. With just one bit transmitted over each state, channel state

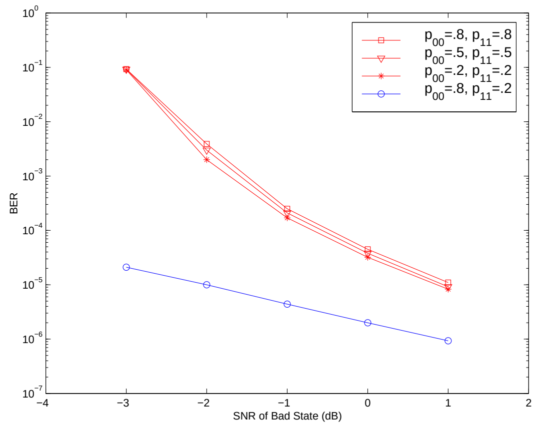

Figure 3: Plot for Known p_{ij}, Steady State

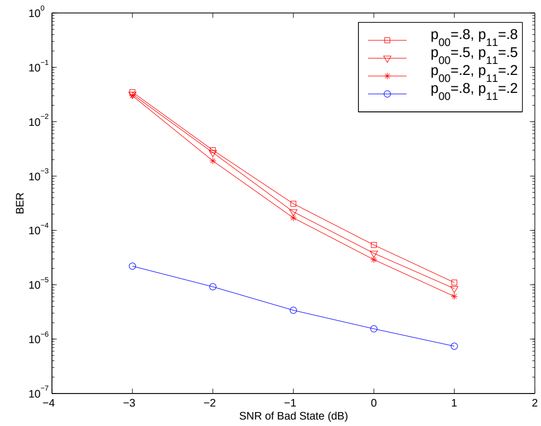

Figure 5: Plots for Unknown State, p_{ij}

estimates are far from reliable and this degrades decoding performance. As the SNR increases, the performance difference between the graphs of Figures 3 and 4 decreases as the channel estimation method has a more difficult time distinguishing between the two states. Note, however, that for high SNRS, the performance of both systems is comparable to that of

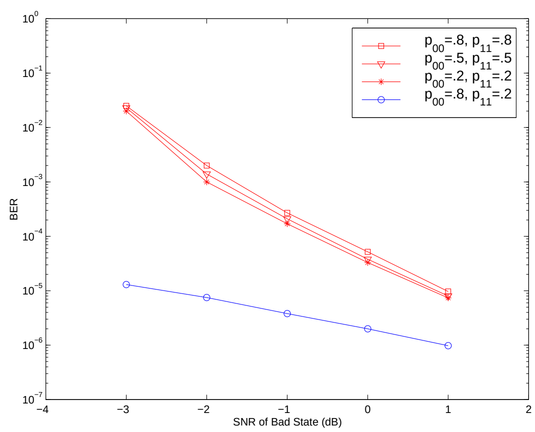

Figure 4: Plot for Known p_{ij}, Channel Estimation

the known channel state case. At high SNRs, the two state system essentially reduces to a one state, high SNR system. Thus, even though channel estimation methods might lose accuracy at high SNRs, the performance is still good since precise state estimation is not necessary in high SNR regions.

Figure 5 contains the simulation results for the case where neither channel state nor transition probability information is available to the decoder. Note that the performance is close to that of the known transition probability, *a posteriori* state estimation case. Comparing the plots of Figures 4 and 5, there is little difference in performance particularly at low SNRs, because the Baum Welch algorithm is doing a good job at estimating channel states. Knowing the state does not give a major advantage at low SNRs since large magnitude noise makes tracking the information bits difficult. At high SNRs, the Baum-Welch algorithm performs with less success, but the more dominating factor is that state estimation at high SNRs is less important.

5.1 Realistic Fading

The simulations for the realistic fading channels of ALD and PS used a transmission bandwidth of approximately 10 MHz with 62 subchannels, data rate of 9600 bits per second, hopping rate of 9600 hops per second, and prespreading factor of 5. Thus, the chip rate was 144 Kchips per second, the chip duration was about 6.9 microseconds, and there were 15 chips per hop. Two velocities are considered: 30 meters/second (m/s) and 0 m/s. The coherent time of the channels at 30 m/s is about 50 ms while at 0 m/s, the coherent time is infinity. The carrier frequency was set to 38 MHz. For cases with no fading side information, the estimation technique used 8 levels of quantization (i.e. M = 8). The simulation results are shown in Figure 6 for various levels of side information and different velocities (shown in meters per second).

First, let us consider the results for AWGN. To achieve a bit error rate (BER) of 10^{-5}, approximately 2 dB is needed. The original results reported by [4] required just 0.7 dB to achieve a BER of 10^{-5} for rate 1/2 codes, but used a block size of 65,536. The difference between the two systems is in rate and block size. Lower rate should result in better performance (for coherent reception), so the dominating factor must be the difference in block size. Thus, the results shown in Figure 8 could probably be considerably improved if larger block sizes are used.

Next, consider the "ideal" fading cases. For these cases, the assumption is that the fade amplitudes take on a Rayleigh distribution and are constant over each hop, but independent between hops. With a diversity factor of 5, the case with fading side information

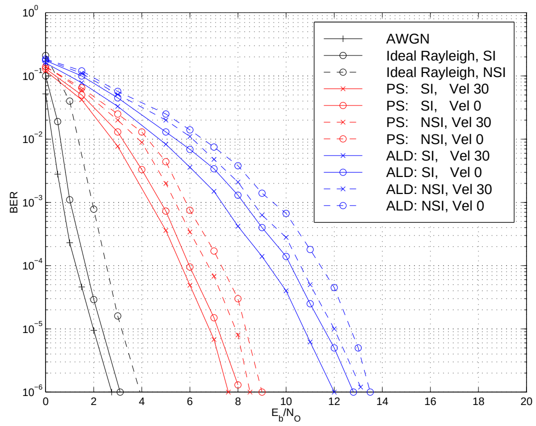

Figure 6: Plot for Realistic Fading

performs comparably with the AWGN case. The difference between cases with and without fading side information is about 1 dB. Presumably, this gap can be reduced if the number of chips per hop increases since the estimation procedure will have more observations.

Finally, let us consider the realistic fading channels. Notice the large gap, first between the ideal fading and cluster of PS curves, then between the cluster of PS and ALD curves. The realistic fading models not only perform considerably worse than "ideal" models, but depending on the parameters of the realistic channel itself, performance can drastically differ. For the realistic fading curves, the greatest impact on performance was whether the channel was ALD or PS; the next greatest impact on performance was whether the velocity was 30 m/s or 0 m/s; the least impact on performance was whether SI was or was not available to the decoder.

It is interesting to note that in particular for the ALD channel, the bit errors were extremely bursty. Generally speaking, each packet of data was either corrected within one iteration or a large proportion of the bits were decoded in error. Thus, at bit error rates on the order of 10^{-4} or 10^{-5}, packet errors rates were very low (i.e. on the order of 10^{-3} or 10^{-4}). One way to take advantage of this information would be to use a CRC to detect at the end of each iteration whether or not any errors remain. Thus, rather than fixing the number of iterations, the decoder could stop when zero errors remained. For the most part, the decoder could end after one iteration and thus would drastically save on decoder complexity. In addition, if at the end of the allocated number of decoder iterations, there still existed bit errors, one could throw out the packet. The result is that throughput would be affected only minutely since the incidence of packet error is low and yet the BER would be drastically improved since the average number of bit errors per packet error is large.

6 Conclusion

The performance of turbo codes in fading and burst channels has been investigated. First, we considered a Gilbert-Elliot model with varying levels of side information. It was shown that if the Baum-Welch procedure is used, performance is not seriously degraded if the transition probabilities of the HMM are unknown. While there exist some improvements of using *a posteriori* state estimates versus steady state probabilities, there still exists a large gap between these cases and the known channel state case. Essentially, the state estimates are not very accurate. The lack of precision arises from the fact that only one bit is transmitted over each state. It would be interesting to generalize these results to the case where H bits are transmitted over each state before a state transition can be made. Next, we considered turbo codes in a hybrid direct-sequence frequency-hop spread spectrum system with realistic fading. It was shown that ideal fading models while analytically convenient, do not necessarily portray a fading channel accurately. In addition, a low complexity method for reducing the bit error rate while only slightly reducing throughput was discussed for the ALD fading channel.

Acknowledgement: The authors would like to thank Do-Sik Yoo for helping to build the realistic fading channel models.

References

[1] J. Kang and W. Stark, "Turbo Codes for Coherent FH-SS With Partial Band Interference," *Proc. MILCOM '97*, November 1997.

[2] J. Kang and W. Stark, "Turbo Codes for Noncoherent FH-SS With Partial Band Interference," *IEEE Trans. Communications*, To Appear November 1998.

[3] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Trans. Information Theory*, vol. 20, March 1974.

[4] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding," *Proc. ICC '93*, May 1993.

[5] D. Cox, "910 MHz Urban Mobile Radio Propagation: Multipath Characteristics in New York City," *IEEE Trans. Comm.*, Nov. 1973.

[6] D. Cox, "Delay Doppler Characteristics of Multipath Propagation at 910 MHz in a Suburban Mobile Radio Environment," *IEEE Trans. Antennas Propagat*, Sept. 1972.