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-Elliott 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-Elliott burst channel and the instantaneous amplitudes for the realistic fading channel. These estimates will then be used in the turbo decoder.

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

2.1 Gilbert-Elliott 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 \( d_k \) 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-Elliott 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](image)

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 transmis-
sion over the bad state, the noise is white Gaussian with power spectral density $N_i/2$ where $N_i > N_0$. 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,

$$
Z_{i,k} = 0 \iff y_{i,k} = \sqrt{E} c_{i,k} + \eta_{i,k}^0
$$
$$
Z_{i,k} = 1 \iff y_{i,k} = \sqrt{E} c_{i,k} + \eta_{i,k}^1
$$

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

### 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 hopping. 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 $R_c = 3 \times R_b \times L$. The transmission bandwidth of the system is $W = Q \times R_c$.

Two measured channels are considered. Pine Street (PS) is taken from an urban area and has 12 independent 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 $\{y_{i,k}\}_{i=1}^3$ are the $L$ chips corresponding to each coded bit $c_{i,k}$.

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

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

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

(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-Elliott Channel

We will first consider the modifications to the turbo decoder necessary for the Gilbert-Elliott 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

$$
p(y_{i,k} | d_k = i, S_k = m, S_{k+1} = m', z_{i,k} = z) = \frac{1}{\sqrt{\pi N_x}} e^{-\frac{1}{N_x} (y_{i,k} - \tilde{z}_{i,k})^2}.
$$

(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.

$$
p(y_{i,k} | d_k = i, S_k = m, S_{k+1} = m') = \frac{1}{\sqrt{\pi N_1}} e^{-\frac{1}{N_1} (y_{i,k} - \tilde{z}_{i,k})^2} \cdot p(z_{i,k} = 1) + \frac{1}{\sqrt{\pi N_0}} e^{-\frac{1}{N_0} (y_{i,k} - \tilde{z}_{i,k})^2} \cdot p(z_{i,k} = 0)
$$

(4)
Note that $p(z_{i,k} = z)$ is not known. One possibility is to use the steady state probability $\lim_{n \to \infty} p(z_{i,k} = z)$. This can be solved using $p = \nu P$, $\sum_{j} v_j = 1$, 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 $k = 1$,

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

(5)

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

$$p(z_{i,k} = 0|y_{i,1}, \ldots, y_{i,k-1}) = \frac{p(y_{i,k}|z_{i,k} = 0, y_{i,1}, \ldots, y_{i,k-1})}{p(y_{i,1}, \ldots, y_{i,k})}$$

$$\approx \frac{p(y_{i,k}|z_{i,k} = 0) p(z_{i,k} = 0, y_{i,1}, \ldots, y_{i,k-1})}{p(y_{i,1}, \ldots, y_{i,k})}$$

(7)

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

$$p(z_{i,k} = 0, y_{i,1}, \ldots, y_{i,k-1})$$

$$= \sum_{z=0}^{1} p(z_{i,k} = 0, z, y_{i,1}, \ldots, y_{i,k-1})$$

(8)

$$= \sum_{z=0}^{1} p(y_{i,k-1}|z_{i,k-1} = z, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = 0, z_{i,k-1} = z, y_{i,1}, \ldots, y_{i,k-2})$$

(9)

$$= \sum_{z=0}^{1} p(y_{i,k-1}|z_{i,k-1} = z) \cdot p(z_{i,k} = 0, z_{i,k-1} = z, y_{i,1}, \ldots, y_{i,k-2})$$

(10)

and (10) is computed using

$$p(z_{i,k} = a, z_{i,k-1} = b, y_{i,1}, \ldots, y_{i,k-2})$$

$$= \begin{cases} p(y_{i,1}, \ldots, y_{i,k-2}|z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(y_{i,1}, \ldots, y_{i,k-2}|z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(y_{i,1}, \ldots, y_{i,k-2}|z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \cdot p(z_{i,k} = a, z_{i,k-1} = b) \end{cases}$$

(11)

$$p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2})$$

$$= \begin{cases} p(y_{i,1}, \ldots, y_{i,k-2}|z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \cdot p(z_{i,k} = b, y_{i,1}, \ldots, y_{i,k-2}) \end{cases}$$

(12)

$$p(z_{i,k} = 0, y_{i,1}, \ldots, y_{i,k-2})$$

$$= \sum_{z=0}^{1} p(y_{i,k-1}|z_{i,k-1} = z) \cdot p(z_{i,k} = 0, z_{i,k-1} = z, y_{i,1}, \ldots, y_{i,k-2})$$

(13)

Combining (10) and (14),

$$p(z_{i,k} = 0, y_{i,1}, \ldots, y_{i,k-2})$$

$$= \sum_{z=0}^{1} p(y_{i,k-1}|z_{i,k-1} = z) \cdot p(z_{i,k-1} = z, y_{i,1}, \ldots, y_{i,k-2}) p_{z_0}$$

(15)

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 combination is optimal. If we let

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

(16)

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

$$\mu_{x_{i,k}} = c_{i,k} \sum_{l=1}^{L} a_{i,k,l}^2$$

(17)

$$\sigma_{x_{i,k}}^2 = \frac{N_0}{2} \sum_{l=1}^{L} a_{i,k,l}^2$$

(18)

Thus, (2) can be computed using

$$p(x_{i,k}|\{a_{i,k,l}\}_{l=1}^{L}, d_k = i, S_k = m, S_{k+1} = m')$$

$$= \frac{1}{\sqrt{2\pi \sigma_{x_{i,k}}^2}} e^{-\frac{(x_{i,k} - \mu_{x_{i,k}})^2}{2\sigma_{x_{i,k}}^2}}.$$ (19)

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 \( a_{k,j} \). For notational simplicity, let us denote the fade amplitudes as \( a_k \) for \( k = 1, \ldots, 3 \times L \times N / h \) where \( 3 \times L \times N \) 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 \( \mathbf{R} \) be the vector of received channel outputs that is available to the MAP decoder, \( \mathbf{R}_k \) be the subset of \( \mathbf{R} \) that has been received with the same fading amplitude \( a_k \) and \( \mathbf{R}_k^c \) be the subset of \( \mathbf{R} \) that has not been received with the same fading amplitude \( a_k \). Consider the quantization of the fading amplitudes into \( M \) regions, \( B_1, \ldots, B_M \), and the corresponding \( M \) output levels or centroids, \( l_1, \ldots, l_M \) where \( l_i \in B_i \). Then, the calculation of a posteriori fading probabilities is as follows for \( i = 1, \ldots, M \).

\[
p(a_k \in B_i | \mathbf{R}) = \frac{p(\mathbf{R}_k | a_k \in B_i) \cdot p(a_k \in B_i)}{p(\mathbf{R})} \quad (20)
\]

\[
= p(\mathbf{R}_k | a_k \in B_i) \cdot \frac{p(\mathbf{R}_k)}{p(\mathbf{R})} \quad (21)
\]

\[
= p(\mathbf{R}_k | a_k \in B_i) \cdot p(a_k \in B_i) \cdot K \quad (22)
\]

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 \( p(\mathbf{R}_k | a_k \in B_i) \) rises exponentially. To overcome this problem, the following recursion was developed.

\[
p(a_k \in B_i | R_{k,1}, R_{k,2}, \ldots, R_{k,i-1}, R_{k,i})
= \frac{p(\mathbf{R}_k, \mathbf{R}_{k,1} | a_k \in B_i, \mathbf{R}_{k,1}, \ldots, R_{k,i-1})}{p(\mathbf{R}_k, \mathbf{R}_{k,1} | \mathbf{R}_{k,1}, \ldots, R_{k,i-1})}
= \frac{p(\mathbf{R}_k, \mathbf{R}_{k,1} | a_k \in B_i) \cdot p(a_k \in B_i | \mathbf{R}_{k,1}, \ldots, R_{k,i-1})}{p(\mathbf{R}_k, \mathbf{R}_{k,1} | \mathbf{R}_{k,1}, \ldots, R_{k,i-1})} \tag{23}
\]

\[
= \frac{p(\mathbf{R}_k, \mathbf{R}_{k,1} | a_k \in B_i) \cdot p(a_k \in B_i | \mathbf{R}_{k,1}, \ldots, R_{k,i-1})}{p(\mathbf{R}_k, \mathbf{R}_{k,1} | \mathbf{R}_{k,1}, \ldots, R_{k,i-1})} \tag{24}
\]

Once the fading estimates have been computed, maximum ratio combining can be performed using the estimated fading value, \( \hat{a}_{i,k,j} \).

\[
\hat{a}_{i,k,m} = \sum_{i=1}^{M} l_i p(a_{i,k,m} \in B_i | \mathbf{R}) \tag{25}
\]

5 Simulation Results

In this section, we present the simulation results for the Gilbert-Elliott 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-Elliott 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.
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

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

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

Figure 5: Plots for Unknown State, $p_{ij}$
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 $10^{-3}$ or $10^{-5}$, packet error 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 \textit{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