

LOW COMPLEXITY TURBO DECODING FOR BINARY HIDDEN MARKOV CHANNELS

Javier Garcia-Frias

Department of Electrical and Computer Engineering
University of Delaware

John D. Villasenor

Electrical Engineering Department
University of California, Los Angeles

Abstract – We describe parallel concatenated codes for communications over binary hidden Markov channels. We present a simplified decoding system that utilizes the *a priori* statistics of the channel and clearly outperforms systems based on the traditional approach of using a channel interleaver to create a channel which is assumed to be memoryless. Although the performance of this method is slightly worse than that of the best known methods, its main advantage (besides the reduced complexity) is that there is no need to change the turbo encoder structure depending on the channel parameters.

I. Introduction

Many practical digital communications channels exhibit statistical dependencies among errors. The error pattern of the discrete channel (modulator-real channel-demodulator) can be modeled using binary hidden Markov channels [1], [2]. Such channels are characterized by a set of states $S_j, 0 \leq j \leq S - 1$, the matrix of transition probabilities among states ($A=(a_{ij})$), with a_{ij} the probability of transition from state S_i to state S_j , i.e., $a_{ij} = P_t(S_j|S_i), 0 \leq i, j \leq S - 1$, and the list giving the bit error probability to associate with each state ($B=(b_j(v))$), with $b_j(v)$ the probability of getting the output v in state S_j , i.e., $b_j(v) = P_o(v|S_j), 0 \leq j \leq S - 1, v \in \{0, 1\}$.

It is intuitive that the presence of memory in these channels leads to increased capacity relative to memoryless channels with the same stationary bit error probability [3]. In practice, many communication systems make use of a channel interleaver to distribute the errors so that codes designed for a memoryless channel can be used. While the application of interleaving does not change the capacity of the channel, assuming that the channel is memoryless limits the achievable performance of the decoder. Exploiting the higher capacity of hidden Markov channels in practice has proven to be challenging. In [4] decision feedback decoders which perform recursive state estimations are used in the decoding process. However, the recursions are vulnerable to error propagation, and the decision feedback decoder can not be reliably used when the quality of the channel degrades. Turbo coding [5] for binary hidden Markov channels has been previously described in [6], [7]. However, these previous methods involve a considerable increase in complexity, since supertrellises jointly describing the constituent encoders and the hidden Markov channel have to be built. We propose a simplified decoding method, which performs slightly

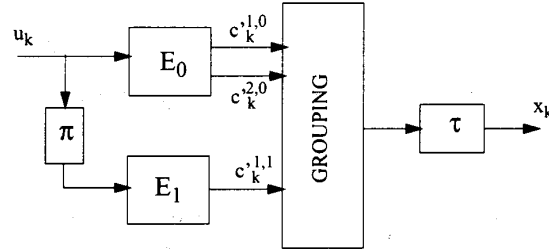


Fig. 1. Encoder structure used for the simplified method proposed in this paper. π represents the turbo encoder interleaver (length M). τ represents the channel interleaver, with length, J , depending on M and the code rate.

worse than the method in [6], [7] but with the main advantage (besides the reduced complexity) that there is no need to change the turbo encoder structure depending on the hidden Markov channel parameters.

II. Simplified turbo decoding for binary hidden Markov channels

We consider the case of a parallel concatenated code with a single interleaver and two constituent convolutional encoders. The k th input bit is denoted by u_k (with $k = 1 \dots M$, where M is the block length) and can take on values $i, i \in \{0, 1\}$. As shown in figure 1, after the bits have been turbo coded, they are grouped and interleaved (forming the sequence $\{x_k\}$) and sent through the channel. The received bits are denoted by $\{v_k\}$. Notice that there are two different interleavers: the one corresponding to the turbo encoder (of length M) and the channel interleaver (of length J). The relation between M and J is fixed for a given turbo code.

We will denote by $O_k^p = [c_k^{1,p} \dots c_k^{n_p,p}]$ the subset of elements of the observation sequence associated with the $1/n_p$ rate "present" decoder (non-interleaved or interleaved); i.e., the one in which processing is occurring. $O_k^f = [c_k^{1,f} \dots c_k^{n_f,f}]$ is used to denote the elements of the observation sequence associated with the other, or "former" decoder (interleaved or non-interleaved) of rate $1/n_f$. For example, if the present constituent encoder has rate $1/2$, and the other encoder has rate 1, $O_k^p = [c_k^{1,p}, c_k^{2,p}]$ and $O_k^f = [c_k^{1,f}]$. $c_k^{i,p}$ (analogously $c_k^{i,f}$) represents the corresponding coded bit associated with the present encoder (i.e., before being corrupted by the error pattern introduced by the hidden Markov channel). Obviously, there is a one to one relation (given by the channel in-

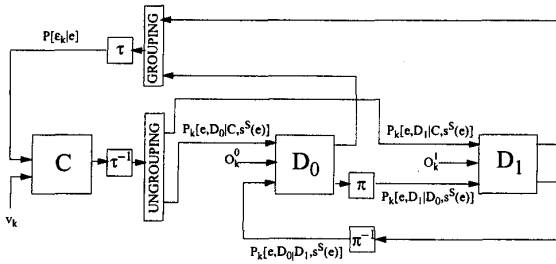


Fig. 2. Decoder structure for the simplified method proposed in this paper. The figure outlines the information flow as defined in section II. $\{v_k\}$ is the received observation sequence.

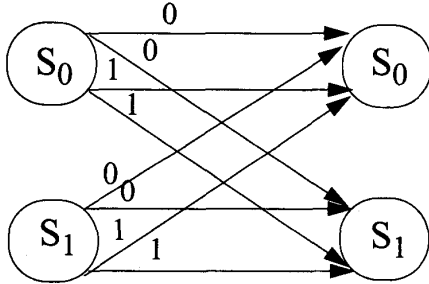


Fig. 3. Trellis representing the binary hidden Markov channel. The figures represent the error pattern associated with each branch.

terleaver) between $\{c_k^{1,p}, c_k^{2,p}, c_k^{1,f}\}$, $k \in [1, K]$ and $\{x_k\}$, $k \in [1, J]$. The input and coded bits associated with the branch e in the present constituent encoder are denoted as $u(e)$ and $O(e) = [c(e)^{1,p} \dots c(e)^{n_p,p}]$, respectively.

The basic idea of the method proposed in this paper is to treat the trellis describing the binary hidden Markov channel as another constituent decoder which exchanges extrinsic information with the other constituent decoders in each one of the turbo decoding iterations. The channel block uses as extrinsic information the estimation of the probability of the error pattern that is provided by the constituent decoder blocks. On the other hand, it produces a new estimation of such a probability which will be used as extrinsic information by the constituent turbo decoders. We will give the equations separately for the block corresponding to the hidden Markov channel (denoted by C) and for the two constituent convolutional decoders (denoted by D_0 and D_1). The decoder structure is shown in figure 2.

A. Binary hidden Markov channel block

As in the case of the trellises representing the constituent encoders, we will use e to symbolize the trellis edges, or branches, with the starting and ending state associated with a particular edge e given by $s^S(e)$ and $s^E(e)$ respectively. The error pattern bit corresponding to branch e will be denoted by $\epsilon(e)$. ϵ_k denotes the error pattern introduced at time k ($\epsilon_k \in \{0, 1\}$). As shown in figure 3, the trellis rep-

resenting the binary hidden Markov channel has two parallel branches between states (one associated with the error pattern $\epsilon(e) = 0$ and the other with the error pattern $\epsilon(e) = 1$). Each one of the branches in the trellis will have an associated *a priori* probability, which is obtained from the hidden Markov model as $a_e = P(e|s^S(e)) = P_o(\epsilon(e)|s^S(e)) \times P_t(s^E(e)|s^S(e))$. If the observation sequence (error pattern $\{\epsilon_k\}$) associated with the binary hidden Markov channel were available at the decoder, it would be possible to apply the forward/backward equations for the channel block:

$$\alpha_k(s) =$$

$$\sum_{e: s^E(e)=s} \alpha_{k-1} [s^S(e)] a_e P[\epsilon_k|e], \quad 1 \leq k \leq J \quad (1)$$

$$\beta_k(s) =$$

$$\sum_{e: s^S(e)=s} \beta_{k+1} [s^E(e)] a_e P[\epsilon_{k+1}|e], \quad J-1 \geq k \geq 1 \quad (2)$$

$$P(\epsilon_k = i|C) \propto$$

$$\sum_{e: \epsilon(e)=i} \alpha_{k-1} [s^S(e)] a_e P[\epsilon_k|e] \beta_k [s^E(e)], \quad (3)$$

with $1 \leq k \leq J$.

Notice, however that the observation sequence is not available at the decoder, since the only available observation is the sequence of coded bits. In other words, when we receive a coded bit (say bit 1), we do not know if the transmitter has sent bit 1 and no error has been introduced by the channel or if the transmitter has sent bit 0 and an error has been generated. However, the constituent decoder blocks produce an estimation, that we will denote by \tilde{x}_k , for such a coded bit (specifically, from equation (15) we obtain $P[x_k = j|D_i]$, $j \in \{0, 1\}$). Therefore we have an estimation $\tilde{\epsilon}_k = v_k \oplus \tilde{x}_k$ of the error pattern. Looking at the trellis of the binary hidden Markov channel, it is clear that $P(\tilde{\epsilon}_k|e) = 0$ if $\tilde{\epsilon}_k \neq \epsilon(e)$ and $P(\tilde{\epsilon}_k|e) = 1$ if $\tilde{\epsilon}_k = \epsilon(e)$. We can then estimate the value of $P[\epsilon_k|e]$ by using:

$$P[\epsilon_k|e] = E[P(\tilde{\epsilon}_k|e)] = P(\tilde{\epsilon}_k = \epsilon(e)) =$$

$$P(x_k = v_k \oplus \epsilon(e)|D_i) = P(c_{k'}^{j,i} = v_k \oplus \epsilon(e)|D_i), \quad (4)$$

where $P(c_{k'}^{j,i} = v_k \oplus \epsilon(e)|D_i)$ is calculated by using equation (15). Notice that due to the channel interleaver and grouping process, bit x_k will correspond to one of the coded bits (for example the one in position j) associated with trellis transition k' of one of the constituent encoders (for example, encoder i) and therefore it will be denoted by $c_{k'}^{j,i}$.

With this estimation, the resulting equations for the hidden Markov channel block are given by:

$$\alpha_k(s) =$$

$$\sum_{e: s^E(e)=s} \alpha_{k-1} [s^S(e)] a_e P[\epsilon_k|e], \quad 1 \leq k \leq J \quad (5)$$

$$\beta_k(s) = \sum_{e: s^E(e)=s} \beta_{k+1} [s^E(e)] a_e P[\epsilon_{k+1}|e], \quad J-1 \geq k \geq 1 \quad (6)$$

$$P(\epsilon_k = i|C) \propto \sum_{e: \epsilon(e)=i} \alpha_{k-1} [s^S(e)] a_e \beta_k [s^E(e)], \quad 1 \leq k \leq J, \quad (7)$$

where as indicated before $P[\epsilon_k|e]$ is obtained from equation (4) and represents the extrinsic information passed from the constituent decoders to the channel block. Notice that in order to avoid positive feedback with the constituent decoders blocks, the value of $P[\epsilon_k|e]$ is not used in equation (7).

B. Constituent decoder blocks

Since we are using a channel interleaver between the turbo encoder and the channel, the error pattern seen by the turbo decoder is i.i.d (i.e., similar to a BSC channel). If we do not consider the block corresponding to the hidden Markov channel, the corresponding equations for a single convolutional code would be:

$$\alpha_k(s) = \sum_{e: s^E(e)=s} \alpha_{k-1} [s^S(e)] P[e|s^S(e)] P[O_k^p|e], \quad (8)$$

with $1 \leq k \leq K$.

$$\beta_k(s) = \sum_{e: s^S(e)=s} \beta_{k+1} [s^E(e)] P[e|s^S(e)] P[O_{k+1}^p|e], \quad (9)$$

with $K-1 \geq k \geq 1$.

$$P(u_k = i|D_p) \propto \sum_{e: u(e)=i} \alpha_{k-1} [s^S(e)] P[e|s^S(e)] P[O_k^p|e] \beta_k [s^E(e)], \quad (10)$$

with $1 \leq k \leq M$

$$P(c_k^{i,p} = i|D_p) \propto \sum_{e: c(e)^{r,p}=i} \alpha_{k-1} [s^S(e)] P[e|s^S(e)] P[O_k^p|e] \beta_k [s^E(e)], \quad (11)$$

with $1 \leq k \leq K$ and $P[O_k^p|e] = \prod_{i=1}^{n_p} P(\epsilon = c_k^{i,p} \oplus c(e)^{i,p})$, where $P(\epsilon = 1)$ is the stationary probability of error in the channel.

The decoding equations are, however, different from the ones corresponding to BSC channels, since the information calculated in the binary hidden Markov channel block has to be incorporated into the equations. The idea is to use that information to calculate an estimation of the transition probability for each branch, modifying the equations of the decoder in which processing is occurring in such a way that the factor $P[e|s^S(e)]$ is substituted by $P_k[e, D_p|D_{f_p}, s^S(e)]$, where D_{f_p} denotes all the other constituent decoders (including the hidden

Markov channel block) and D_p represents the present decoder. In other words, we obtain an estimation of the transition probability of going through branch e by using the information available from the other constituent decoders. However, this substitution has to be done in such a way that positive feedback to the other decoders is avoided (i.e., passing only the so-called extrinsic information). The resulting equations are:

$$\alpha_k(s) = \sum_{e: s^E(e)=s} \alpha_{k-1} [s^S(e)] P_k [e, D_p|C, s^S(e)] \times$$

$$P_k [e, D_p|D_f, s^S(e)] P[O_k^p|e], \quad 1 \leq k \leq K \quad (12)$$

$$\beta_k(s) = \sum_{e: s^S(e)=s} \beta_{k+1} [s^E(e)] P_{k+1} [e, D_p|C, s^S(e)] \times$$

$$P_{k+1} [e, D_p|D_f, s^S(e)] P[O_{k+1}^p|e], \quad K-1 \geq k \geq 1 \quad (13)$$

$$P(u_k = i|D_p) \propto \sum_{e: u(e)=i} \alpha_{k-1} [s^S(e)] \times$$

$$P_k [e, D_p|C, s^S(e)] P[O_k^p|e] \beta_k [s^E(e)], \quad (14)$$

with $1 \leq k \leq M$.

$$P(c_k^{r,p} = i|D_p) \propto \sum_{e: c(e)^{r,p}=i} \alpha_{k-1} [s^S(e)] P'[O_k^p|e] \times$$

$$P_k [e, D_p|D_f, s^S(e)] P_k [e, D_p|C, s^S(e)] \beta_k [s^E(e)], \quad (15)$$

with $1 \leq k \leq K$, $P[O_k^p|e] = \prod_{i=1}^{n_p} P(\epsilon = c_k^{i,p} \oplus c(e)^{i,p})$ and $P'[O_k^p|e] = \prod_{i=1, i \neq r}^{n_p} P(\epsilon = c_k^{i,p} \oplus c(e)^{i,p})$, where $P(\epsilon = 1)$ is the stationary probability of error in the channel. In the second (interleaved) constituent decoder, u_k in equation (14) should be replaced by $u_{\pi^{-1}(k)}$. $P_k [e, D_p|D_f, s^S(e)]$ is the usual extrinsic information in standard turbo codes and its value is calculated by:

$$P_k [e, D_p|D_f, s^S(e)] = P(u_{k'} = u(e)|D_f), \quad (16)$$

where $P(u_{k'} = u(e)|D_f)$ is calculated by using equation (14), with k' accounting for the effect of the two interleavers. $P_k [e, D_p|C, s^S(e)]$ is the extrinsic information that is passed from the channel block to the constituent decoders and its value is calculated using:

$$P_k [e, D_p|C, s^S(e)] = \prod_{i=1}^{n_p} P(\epsilon_{k''} = c_k^{i,p} \oplus c(e)^{i,p}|C), \quad (17)$$

where $P(\epsilon_{k''} = c_k^{i,p} \oplus c(e)^{i,p}|C)$ is calculated using equation (7), with k'' accounting for the effect of the two interleavers and the grouping.

The method proposed in this paper can be also applied for the case in which the parameters of the channel are not known *a priori*, but this would probably require some modifications to the above equations and it has not been considered here. In the case in which the channel parameters are known, the

performance of this method is worse than that of the best method proposed in [6], but it achieves important gains with respect to the case in which no modifications to the standard turbo decoder are used.

III. Simulation results

In order to assess the performance of the proposed method, we consider two binary hidden Markov channels with two states. For the first channel, the transition probability from the good to the bad state is .0486, and .0914 is the value of the transition probability from the bad to the good state. For the second channel these values are .006943 and .013057, respectively. In both cases, the bit error probability in the bad state is fixed to .5. The performance of the system is studied as a function of the value of the bit error probability in the good state (notice that, since all the other parameters are fixed, there is a one to one correspondence between the bit error probability in the good state and the stationary bit error probability, ρ , which is the parameter used in figures 4 and 5).

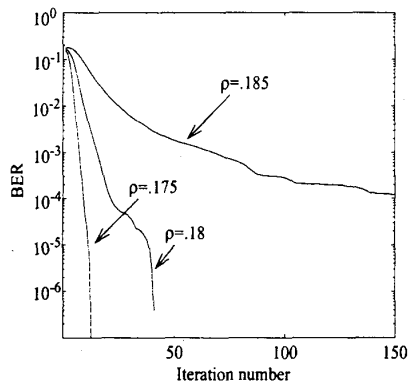


Fig. 4. Convergence behavior for the rate 1/3 turbo code and the first channel described in section III. For a BSC channel, capacity occurs for $\rho = .174$. Notice that we can decode above this limit and close to the real capacity of the channel, which corresponds to a value $\rho = .2083$.

We use a rate 1/3 turbo code that includes a systematic bit and two identical recursive 8-state convolutional encoders with generator matrix $G(D) = \frac{1+D+D^2+D^3}{1+D^2+D^3}$ and an interleaver with length 16384. As indicated before, in order to obtain good performance it is necessary to use a channel interleaver which "separates" the hidden Markov channel and the turbo decoder. Each simulation consisted of at least 40 million bits. For rate 1/3 codes, the bit error probability corresponding to the capacity of a BSC channel is $\rho = .174$. Therefore, by using channel interleaving and ignoring the memory of the channel (the usual approach to cope with bursty channels), it is impossible to send reliable information through each one of these channels when the stationary bit error probability is higher than .174. Figure 4 shows, for the first channel and the proposed decoding method, the decoded bit error rate (BER)

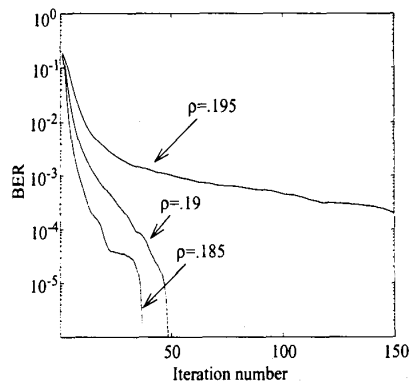


Fig. 5. Convergence behavior for the rate 1/3 turbo code and the second channel described in section III. For a BSC channel, capacity occurs for $\rho = .174$. Notice that we can decode above this limit and close to the real capacity of the channel, which corresponds to a value $\rho = .2307$.

as a function of the stationary bit error probability. Convergence is achieved at $\rho = .18 - .185$, which is higher than the memoryless limit and close to the theoretical limit for this channel (which corresponds to a value $\rho = .2083$). As we can see in figure 5, for the second channel convergence is achieved at $\rho = .19 - .195$. The theoretical limit in this case is $\rho = .2307$.

IV. Conclusion

We have introduced a simplified method for combining turbo decoding and binary hidden Markov channels. The performance that we obtained is slightly worse than that obtained with supertrellis approaches (about .01 or .005 degradation in the stationary bit error probability required for convergence), but it clearly outperforms traditional systems based on channel interleaving. Moreover, the complexity is much lower than in the supertrellis case and the structure of the encoder does not depend on the parameters of the hidden Markov model describing the channel.

References

- [1] L. N. Kanal and A. R. K. Sastry, "Models for Channels with Memory and Their Application to Error Control," *Proc. IEEE*, vol. 66, no. 7, pp. 724-744, 1978.
- [2] W. Turin, "Performance Analysis of Digital Transmission Systems," New York: Computer Science, 1990.
- [3] M. Mushkin and I. Bar-David, "Capacity and Coding for the Gilbert-Elliott Channels," *IEEE Trans. on Inf. Theory*, vol. 35, pp. 1277-1290, Nov. 1989.
- [4] A. J. Goldsmith and P.P. Varaiya, "Capacity, Mutual Information, and Coding for Finite-State Markov Channels," *IEEE Trans. on Inf. Theory*, vol. 42, pp. 868-886, May 1996.
- [5] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," *Proc. IEEE ICC'93*, pp. 1064-1070, 1993.
- [6] J. Garcia-Frias and John D. Villasenor, "Turbo Codes for Binary Markov Channels," *Proc. IEEE ICC'98*, pp. 110-115, June 1998.
- [7] J. Garcia-Frias and John D. Villasenor, "Exploiting Binary Markov Channels with Unknown Parameters in Turbo Coding," *Proc. IEEE Globecom'98*, pp. 3244-3249, November 1998.