Analysis Tools for Iterative Source-Channel Decoding

Andrew Schaefer, Norbert Görtz, Joachim Hagenauer
Institute for Communications Engineering (LNT)
Munich University of Technology (TUM)
E-mail: {andrew,norbert,hag}@lnt.ei.tum.de

Abstract: *We consider the problem of transmitting a correlated Gaussian source over an AWGN channel. This paper contains two main contributions which aid in the design of such a system. We determine the characteristic curves of the source decoder for use in the extrinsic information transfer charts (EXIT charts) and show how the characteristic curves vary with the amount of correlation in the source. Further, we determine contour lines in the EXIT charts showing the signal to noise ratio of the de-quantized signal given that a particular point in the EXIT chart is reached.*

Keywords: Joint source-channel decoding, EXIT charts

1. Introduction

It is often stated that residual source redundancy can be used for improved decoding, but, due to complexity, optimal joint source-channel decoding is often infeasible. Thus, suboptimal algorithms with lower complexity are usually required. One promising is approach is iterative source-channel decoding for which some foundations were laid in [1].

We provide tools to produce characteristic curves which show how the source decoder interacts with other components in an iterative source-channel decoding scheme and we introduce a method based on EXIT charts to evaluate the performance in terms of source signal-to-noise ratio.

2. System Model

We consider the system model as shown in Figure 1. The correlated source produces a sequence of L continuous-valued, Gaussian distributed, correlated values $X_1^L \doteq \{X_1, X_2, ..., X_L\}$. Each value X_l of the sequence X_1^L is quantized by a Q bit scalar quantizer thus producing a sequence of indices $I_1^L = \{I_1, I_2, ..., I_L\}$. According to a time-invariant bit mapping, each index I_l is assigned a unique binary sequence $\mathbf{B}_l$, where $\mathbf{B}_l = \{B_{l,1}, B_{l,2}, ..., B_{l,Q}\}$, thus producing a bit sequence $\mathbf{B}^K \doteq \{\mathbf{B}_1, \mathbf{B}_2, ..., \mathbf{B}_L\}$ of length $K \doteq L \cdot Q$. This bit sequence is bitwise interleaved, producing $U_1^K = \Pi(\mathbf{B}_1^K)$ where Π defines the interleaving, before being inserted into a channel encoder to produce the bit sequence V_1^N, again where each $v_n \in \{0,1\}$ and $n = 1, ..., N$. Thus the channel code rate is $R_c = K/N$. The sequence V_1^N is transmitted over an AWGN channel using BPSK modulation. The received sequence is called $\tilde{V}_1^N \in I\!R^N$.

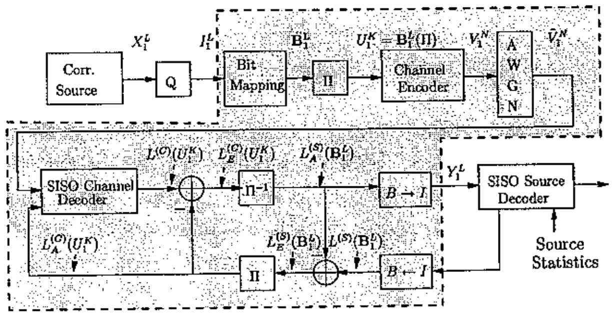

Figure 1: System Model.

Since correlation exists within the source, it contains redundancy and thus may be thought of as a kind of (implicit) channel code. Hence, the entire system may be viewed as a serially concatenated code, the decoder for which is also shown in Figure 1.

The channel encoder may be either a simple convolutional code for example, in which case the decoder would be an APP (A Posteriori Probability, BCJR [2]) decoder for the channel code or the channel encoder may itself be a concatenated code, in which case the output is no longer necessarily APP. Thus we label the channel decoder as being Soft-In/Soft-Out (SISO).

The source and the source decoder will now be described in more detail.

3. The Source and Source Decoder

3.1. Source Generation

We artificially generate a correlated Gaussian sequence by passing a white Gaussian sequence through a recursive filter. As a matter of interest we also generate a correlated sequence by oversampling the white sequence and using interpolation. These possibilities are depicted in Figure 2. We use an oversampling factor of $f_o = 2$ and $H(z) = \frac{z}{z-a}$ with $a = 0.5$ or $a = 0.9$.

3.2. Source Model

Due to correlation in the source, correlations will also exist between the quantized indices. Assuming a Q bit

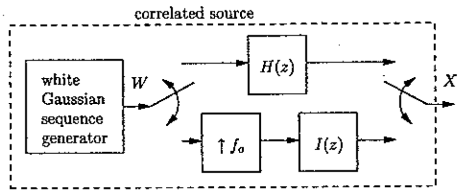

Figure 2: Source Generation.

quantizer, we approximate the correlations using a first order Markov chain where states correspond to quantizer indices. It is possible to determine, either by simulation or by calculation, the transition probabilities of the Markov chain.

If we include all the elements not relevant to the source in a composite channel described by the probability density function $p(Y_1^L|I_1^L)$, then we have an equivalent model from the source perspective which is shown in Figure 3. The composite channel includes the shaded blocks in Figure 1. We make the assumption of the channel being memoryless (which is approximately true due to the bit interleaver) which means that we may write,

$$p(Y_1^L|I_1^L) = \prod_{l=1}^{L} R(Y_l|I_l). \tag{1}$$

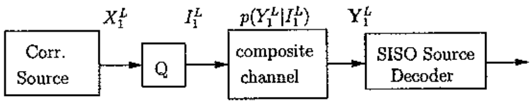

Figure 3: System model from source perspective.

Our composite channel is relatively complicated. However $R(Y_l|I_l)$ simply indicates the probability density of receiving Y_l given that a particular index was sent.

3.3. Source Decoder

After transmission through the noisy composite channel, the problem is to determine probabilities of the states in the Markov source, observed through this noisy channel. This problem is solved by the BCJR algorithm [2] and we now repeat the most important points in our context.

We have a discrete time, finite state Markov source where the states (quantizer indices in our case) are labeled as i, with

$$i \in \mathcal{I} \doteq \{1, ..., 2^Q\}\,. \tag{2}$$

The state at time l is labeled I_l and in our case, the output of the Markov source $Z_l = I_l$. A sequence is denoted by $I_l^{l'} = \{I_l, I_{l+1}, ..., I_{l'}\}$. Further, we have the state transition probabilities,

$$p_l(i|i') = \Pr\{I_l = i|I_{l-1} = i'\}, \tag{3}$$

(see Figure 4) and the output probabilities,

$$q_l(z|i', i) = \Pr\{Z_l = z|I_{l-1} = i', I_l = i\}. \tag{4}$$

Since in our case Z_l=I_l,

$$q_l(z|i', i) = \begin{cases} 1, & z = i \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

We input the sequence $Z_1^L = I_1^L$ into a discrete, memoryless channel which produces the sequence Y_1^L. Thus as stated above, we have (1). Our aim is to determine the conditional probabilities

$$\Pr\{I_l = i|Y_1^L\} = \frac{\Pr\{I_l = i, Y_1^L\}}{\sum_{i\in\mathcal{I}} \Pr\{I_l = i, Y_1^L\}}. \tag{6}$$

According to [2], they can be computed by

$$\Pr\{I_l = i, Y_1^L\} = \alpha_l(i) \cdot \beta_l(i) \tag{7}$$

with the standard forward recursion

$$\alpha_l(i) = \sum_{i'\in\mathcal{I}} \gamma_l(i', i) \cdot \alpha_{l-1}(i') \tag{8}$$

and the backward recursion

$$\beta_l(i) = \sum_{i'\in\mathcal{I}} \gamma_{l+1}(i, i') \cdot \beta_{l+1}(i')\,, \tag{9}$$

where the γ values are defined as

$$\gamma_l(i', i) = \sum_{z\in\mathcal{I}} p_l(i|i') \cdot q_l(z|i', i) \cdot R(Y_l|Z_l = z) \tag{10}$$

$$= p_l(i|i') \cdot R(Y_l|Z_l = i)\,; \tag{11}$$

(11) follows from (5). Note that in a convolutional code of rate $R_c = K/N$, the probabilities $p_l(i|i')$ may only take on values of 2^{-K}. Ours is a more general case, and as an example, we plot the values in Figure 4 for a correlated source with $a = 0.9$.

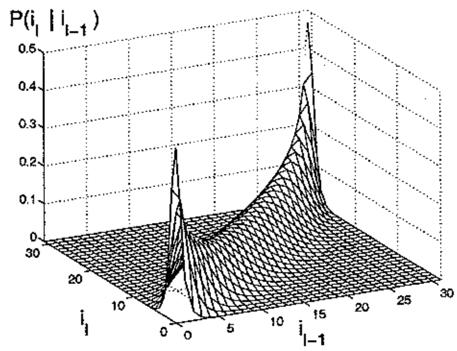

Figure 4: State transition probabilities of Markov source with correlation coefficient $a = 0.9$.

3.4. Bit Mapping

We considered three types of mapping: natural binary coding, Gray mapping and an optimized mapping according to [3] (the mapping is optimized in the sense of obtaining the most extrinsic output given perfect a priori input). The mapping has an important influence on the characteristic curves of the source decoder.

The time-invariant mapping associates a unique binary label $\mathbf{B} = \{B_1, ..., B_Q\}$ where $B_q \in \{0,1\}$, $q \in \{1, ..., Q\}$, to each index i. We write $\mathbf{B} = \text{map}(i)$ and for bit q of the bit label we write $B_q = \text{map}_q(i)$.

3.5. From Bit to Index Probabilities and Back Again

Put in words, index probabilities are approximated by the product of the corresponding bit probabilities. Similarly in the other direction, bit probabilities are obtained by summing over index probabilities with that bit as a label.

Formally, if we write the probabilities for each bit input to the "$B \to P$" block as $p_A^{(S)}(B_{l,q} = c)$, where $c \in \{0,1\}$ then we can write

$$p(I_l = i) = \prod_{q=1}^{Q} p_A^{(S)}(B_{l,q} = \text{map}_q(i)). \qquad (12)$$

Note that these probabilities define $R_l(Y_l|I_l)$, i.e., (12) enables us to calculate the probabilities of the received values given that each of the indices was sent.

Similarly, if the output of the source decoder is $p(I_l = i)$ (equivalent to $\Pr\{I_l = i|Y_1^L\}$ in the BCJR notation), for each time l and each state i, then we can write,

$$p(B_{l,q} = c) = \sum_{i \in \mathcal{I}:\text{map}_q(i)=c} p(I_l = i). \qquad (13)$$

The source decoder receives and sends its messages in terms of log-likelihood ratios or L-values [4] such that

$$L_A^{(S)}(B_{l,q}) = \ln \frac{p_A^{(S)}(B_{l,q} = 0)}{p_A^{(S)}(B_{l,q} = 1)} \qquad (14)$$

and

$$L^{(S)}(B_{l,q}) = \ln \frac{p^{(S)}(B_{l,q} = 0)}{p^{(S)}(B_{l,q} = 1)} \qquad (15)$$

are the input and output messages of the source decoder respectively.

3.6. The Iterative Source-Channel Decoder

L-values are passed around the iterative decoder according to Figure 1 where L-values for each of the quantities are defined similarly to above. In Figure 1, "E" and "A" indicate extrinsic and a priori respectively whilst "(C)" and "(S)" indicate channel decoder and source decoder respectively.

4. Characteristic Curves of Source Decoder

Following [5], we simulate the source decoder alone, for various amounts of a priori information $I_A = I(B_1^L; L_A^{(S)}(B_1^L))$ and measure thereby the mutual information $I_E = I(L_E^{(S)}(B_1^L); B_1^L)$ between the extrinsic outputs and the transmitted bits (note that $I(A; B)$ indicates the mutual information between random variables A and B). In all simulations, $Q = 5$ was used.

We used the BCJR algorithm as described above for source decoding, and we tested correlated sources with $a = 0.5$ and $a = 0.9$; Figure 5 shows the results. Figure 6 shows results for a source with an oversampling factor of $f_o = 2$.

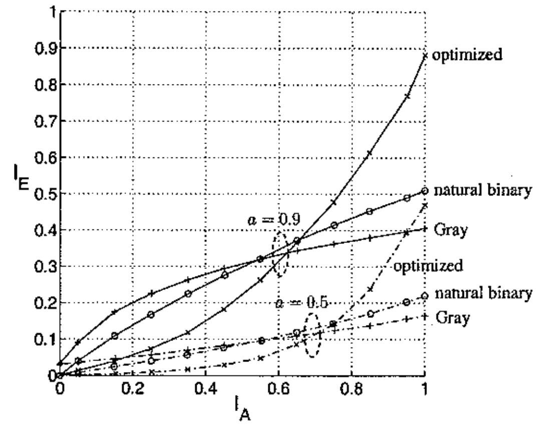

Figure 5: Characteristic curves for filtered source ($a = 0.9$ and $a = 0.5$), $Q = 5$, several mappings.

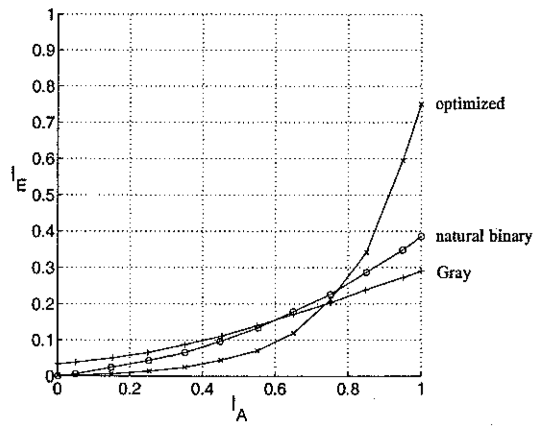

Figure 6: Characteristic curves for oversampled source ($f_o = 2$), $Q = 5$, several mappings.

In both Figures, the Gray-mapping produces an extrinsic output $I_E > 0$, even if the a-priori information is zero ($I_A = 0$). This is due to the non-uniform, symmetric probability distribution of the reproduction levels of the Lloyd-Max quantizer that is used. For the Gray mapping, the average entropy of the individual bits is significantly different from one, i.e., the bits contain redundancies which cause the non-zero starting point of the characteristic curves in Figures 5 and 6. For the natural binary mapping the average entropy of the bits is exactly one and it is very close to one for the optimized mapping. Thus, the characteristic curves both start at $I_E \approx 0$ for $I_A = 0$.

We note that the optimal mapping enables the highest extrinsic output for an a priori mutual information of 1 as it is designed to do.

5. Source-SNR Contour Lines

We first define the signal-to-noise ratio of the whole system as $SNR_{PCM} = \sigma_x^2/\sigma_d^2$ where σ_x^2 is the signal variance and σ_d^2 is the variance of some distorting signal.

In [5], it was shown how to determine the BER from the EXIT chart thus producing BER contour lines. In our case it is possible to go a step further and produce SNR_{PCM} contour lines.

The contour lines were obtained by simulation. At each point in the EXIT chart, we determined an equivalent channel noise variance (using the assumption that our extrinsic L-values are Gaussian distributed). We then transmitted bit mapped symbols over this channel and determined the corresponding SNR_{PCM} after using a minimum mean square estimator as presented in [1].

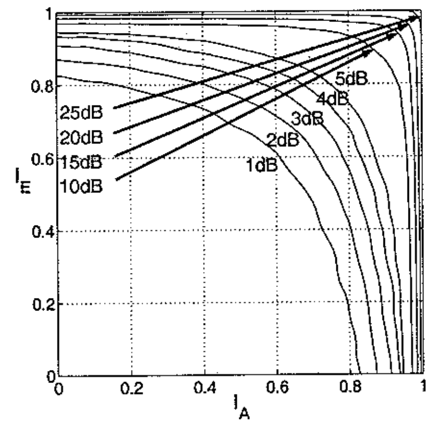

Figure 7: SNR_{PCM} contour lines on the EXIT chart for $Q = 5$.

Figure 7 indicates that is necessary for the EXIT chart trajectory to come close to the outer edge of the EXIT chart. Thus from results in Section 4. we can say that to achieve a reasonable SNR_{PCM}, either a strongly correlated source is required or the channel code of the system should itself be a concatenated code.

6. Results for Iterative Decoder

We simulated a system with $a = 0.9$ and a $m = 2$ recursive convolutional code as the channel encoder. There were 10000 samples transmitted as one 'turbo-block'. The systematic bits of the channel code were punctured such that only a small portion of them (10%) were transmitted (this is similar to 'doping' as presented by the author of [5]). The simulation was performed at $E_s/N_0 = -0.1$dB. Results are shown in Figure 8.

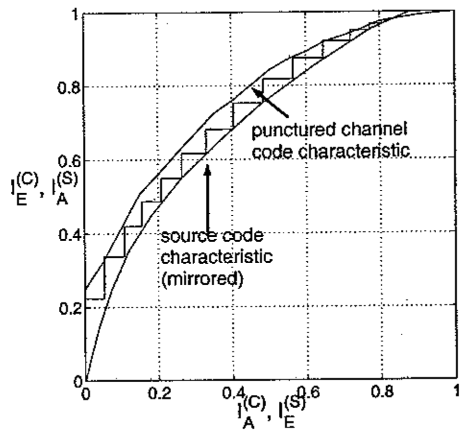

Figure 8: EXIT chart decoding trajectory for transmission of 10000 samples of the correlated Gaussian source with $a = 0.9$ at $E_s/N_0 = -0.1$dB.

7. Conclusions

We have shown that the sequence of source indices from a correlated Gaussian source can be described by a first order Markov model. Thus it can be thought of as a type of soft convolutional code. After transmitting the indices through a noisy channel we can apply the BCJR algorithm to again estimate these states.

Further, we have shown to what extent the correlations still existing in a Gaussian source may be exploited in an iterative source-channel decoder. We have shown contour lines on the EXIT chart which show the resulting signal to noise ratio at the system output given that a certain point on the EXIT chart is reached. Results were presented for a 5 bit quantizer. Finally we have verified the EXIT tool by plotting the decoding trajectory on the EXIT chart for a transmission of 10000 samples.

REFERENCES

[1] N. Görtz, "On the iterative approximation of optimal joint source-channel decoding," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 1662–1670, Sept. 2001.

[2] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. IT-20, pp. 284–287, Mar. 1974.

[3] J. Hagenauer and N. Görtz, "The turbo principle in joint source-channel coding," in *Proceedings of the IEEE Information Theory Workshop*, Apr. 2003.

[4] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, pp. 429–445, Mar. 1996.

[5] S. ten Brink, "Design of serially concatenated codes based on iterative decoding convergence," in *Proceedings of the International Symposium on Turbo Codes & Related Topics*, pp. 319–322, Sept. 2000.