THE USE OF EXIT CHARTS IN ITERATIVE SOURCE-CHANNEL DECODING

Norbert Görtz and Andrew Schaefer

Institute for Communications Engineering (LNT), Munich University of Technology (TUM),
80290 München, Germany, phone: +49 89 289 23494, fax: +49 89 289 23490,
email: {norbert.goertz, andrew.schaefer}@ei.tum.de, web: http://www.lnt.ei.tum.de

ABSTRACT

We consider the problem of quantizing, channel coding, and transmitting a continuous-valued correlated source over a noisy channel. This serial concatenation of the channel code and the source-redundancies calls for the use of the turbo principle for decoding; in our context this concept is known as iterative source-channel decoding. We determine the characteristic curves of the source decoder for use in the extrinsic information transfer charts (EXIT charts) and show how they vary with the quantizer bit mappings and with the amount of correlation in the source. Moreover, we discuss a method for the optimization of the quantizer bit mappings for use in iterative source-channel decoding.

1. INTRODUCTION

It was already pointed out by Shannon [1] that source redundancy will help to combat noise at the receiving end if no attempt is made to eliminate it by source coding. Although in today's communications systems sophisticated source coding schemes are widely used, it is usually impossible to remove all redundancy from the source signals (e.g., speech, audio, or image signals), due delay and complexity constraints that enforce limited block lengths for source and channel coding. Hence, the *parameters*,[1] into which the actual source signals are usually decomposed prior to quantization, are regarded as our correlated source signals below. Although for certain system setups optimal joint source-channel decoding algorithms are known (e.g., [2]), they are often infeasible due to complexity. Hence, suboptimal algorithms with lower complexity are usually required. One promising approach is *iterative* source-channel decoding (ISCD) for which some foundations were laid in [2], extensions were presented in [3], and analysis tools were discussed in [4]. In this paper we discuss the characteristic curves which show how the source decoder interacts with other components in an iterative source-channel decoding scheme and we present a method based on EXIT charts [5] to evaluate the performance in terms of source signal-to-noise ratio. Further, we show that the performance of such an iterative decoding system is strongly influenced by the quantizer bit mappings and we sketch a method of how to optimize them.

2. SYSTEM MODEL

Our system model is given in Figure 1. The source produces a sequence $X_1^L \doteq \{X_1, X_2, ..., X_L\}$ of L continuous-valued, Gaussian distributed, correlated values X_l, which are generated by passing uncorrelated Gaussian samples through a discrete-time linear filter with the transfer function $H(z) = \frac{z}{z-a}$, $a = \{0.5, 0.9\}$. Each value X_l is quantized by a Q-bit scalar quantizer thus producing an index sequence $I_1^L = \{I_1, I_2, ..., I_L\}$. According to a time-invariant bit mapping μ, each index I_l is assigned a unique Q-bit binary sequence $\mathbf{B}_l = \mu(I_l)$, where $\mathbf{B}_l = \{B_{l,q},\ q = 1, 2, ..., Q\}$, $B_{l,q} \in \{0, 1\}$, thus producing a bit sequence $\mathbf{B}_1^K \doteq \{\mathbf{B}_1, \mathbf{B}_2, ..., \mathbf{B}_L\}$ 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 $V_1^N \doteq \{V_1, ..., V_N\}$, where each $V_n \in \{0, 1\}$, $n = 1, ..., N$, and $N \geq K$. Thus the channel code rate is $R_c = K/N$. The sequence V_1^N is transmitted over an AWGN channel using coherently detected BPSK modulation;[2] $N_0/2$ is the power spectral density of the channel noise and E_s is the energy used to transmit each code-bit. The received sequence is called $\tilde{V}_1^N \in I\!R^N$. Due to the modulation/transmission model, we obtain channel L-values [6] for individual bits V_n from the received values $\tilde{v}_n$ by

$$\mathrm{L}(\tilde{v}_n|V_n) \doteq \log_e \frac{p(\tilde{v}_n|V_n = 0)}{p(\tilde{v}_n|V_n = 1)} = 4\frac{E_s}{N_0}\tilde{v}_n \ , \qquad (1)$$

i.e., we obtain reliability information serving as the channel input to the channel decoder simply by multiplying the channel outputs $\tilde{v}_n$ with the channel constant $4\frac{E_s}{N_0}$. 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 iterative decoder for which is also shown in Figure 1. In the decoder, information between the components is passed by L-values for *individual* bits; for brevity, we use the vector notation $\mathrm{L}(\mathbf{B}_1^L) \doteq \{\mathrm{L}(B_{l,q}), l = 1, ..., L; q = 1, ..., Q\}$.

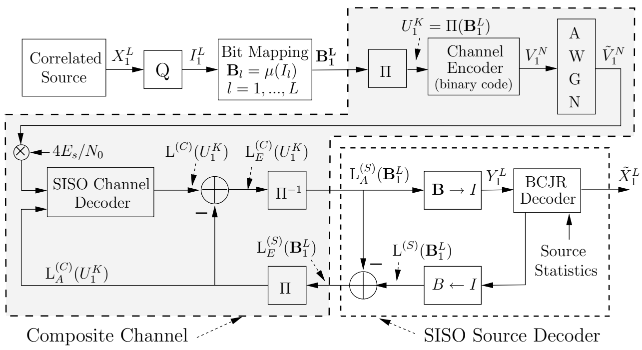

Figure 1: System Model.

The channel encoder may be either a simple binary convolutional code for example, in which case the corresponding component decoder would be a symbol-by-symbol APP (A Posteriori Probability, BCJR [7]) decoder or the channel encoder may itself be a concatenated code, in which case the outputs are no longer necessarily APPs. Thus we label the channel decoder as being Soft-In/Soft-Out (SISO).

[1] Two examples for such parameters are the mean power of the signal and the linear-predictor coefficients that describe the spectral shape of the signal.

[2] The mapping from bits $v_n \in \{0, 1\}$ to antipodal BPSK signal values is included in the AWGN channel. The channel pdf is given by $p(\tilde{v}_n|v_n) = \frac{1}{\sqrt{2\pi}\sigma} \cdot \exp(-\frac{1}{2\sigma^2}(\tilde{v}_n - (1-2v_n))^2)$ with $\sigma^2 = \frac{N_0}{2E_s}$.

3. THE SISO SOURCE DECODER

Due to correlation in the source, correlations will also exist between the quantizer indexes $I_l, I_{l-1}, \ldots$. We model their dependencies by a first order Markov chain where states correspond to quantizer indexes. It is possible to determine, e.g., by simulation, the transition probabilities of the Markov chain. In Figure 2 we give the example of a Gauss-Markov source with filter coefficient $a = 0.9$, and scalar optimal quantization by $Q = 5$ bits. In what follows, we will show how to determine the probabilities of the states of the Markov source, when it is observed through the noisy composite channel that includes the shaded blocks in Figure 1. As the correlations exist between the quantizer *indexes*, i.e., *groups* of bits, we first have to convert our composite channel for *bits* into a "vector channel." For this, we think of the L-values $\mathrm{L}_A^{(S)}(B_{l,q})$ at the channel decoder output as being generated by a memoryless, additive white Gaussian noise channel with some noise variance σ_A^2 (we will not need to know). The corresponding channel model for one of the bits $B_{l,q}$ from the sequence $\mathbf{B}_1^L$ is depicted in Figure 3. The scaling factor $\sigma_A^2/2$ is caused by the L-value consistency condition [5] that links the variance and the mean of a conditional L-value variable at the output of a Gaussian channel. The conditional channel pdf corresponding to Figure 3 equals[3]

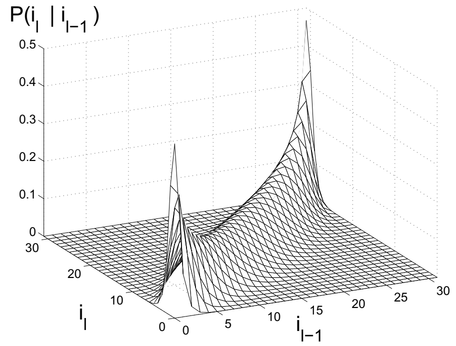

Figure 2: State transition probabilities of Markov source with correlation coefficient $a = 0.9$. Optimal scalar (Lloyd-Max) quantization by $Q = 5$ bits.

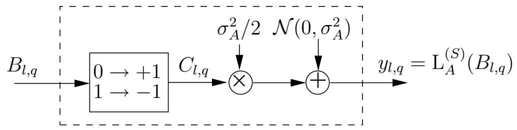

Figure 3: L-values generated by a Gaussian channel.

$$p_c(y_{l,q}|b_{l,q}) = \frac{1}{\sqrt{2\pi}\sigma_A} \cdot \exp\Big(-\frac{1}{2\sigma_A^2}\big(y_{l,q} - \frac{\sigma_A^2}{2}(1 - 2 \cdot b_{l,q})\big)^2\Big) \, . \tag{2}$$

The consequence of the consistency condition is that $\log_e \frac{p_c(Y_{l,q}=\mathrm{L}_A^{(S)}(B_{l,q})|B_{l,q}=0)}{p_c(Y_{l,q}=\mathrm{L}_A^{(S)}(B_{l,q})|B_{l,q}=1)} = \mathrm{L}_A^{(S)}(B_{l,q})$. Note that the L-value $\mathrm{L}_A^{(S)}(B_{l,q})$ we get from the channel decoder can be interpreted as a realization of the random variable $Y_{l,q}$.

With the pdf (2) and the assumption that our composite "L-value channel" is memoryless[4] we can now state a pdf for a *vector* $y_l = \{y_{l,1}, \ldots, y_{l,Q}\}$ of received L-values (from the channel decoder), given some *index* i was transmitted:

$$\begin{aligned} p_c(y_l|i_l) &= \prod_{q=1}^{Q} p_c(y_{l,q}|b_{l,q} = \mu_q(i_l)) \\ &= g(y_l, \sigma_{A,l}) \cdot \exp\big(-\sum_{q=1}^{Q} y_{l,q} \cdot \mu_q(i_l)\big) \, . \end{aligned} \tag{3}$$

This computation is performed by the "$B \to I$"-block in Figure 1. The notation $\mu_q(i_l)$ is used in (3) to address the bit number q that the bit mapping μ generates from the index i_l. The second line in (3) follows, if we insert (2). Note that the function $g(y_l, \sigma_{A,l})$ depends only on the channel observations and the noise variances, but *not* on the hypothesis for the quantizer index i_l.

The problem of how to determine probabilities of the states in the Markov source, observed through a memoryless noisy channel as described by (3), is solved by the BCJR algorithm [7] and we now repeat the most important points in our context.

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

$$i \in \mathcal{I} \doteq \{0, \ldots, 2^Q - 1\} \, . \tag{4}$$

The state at time l is labeled I_l and in our case, the output of the Markov source $Z_l = I_l$. Further, we have the state transition probabilities,

$$p_l(i|i') \doteq \Pr\{I_l = i | I_{l-1} = i'\}, \tag{5}$$

(see Figure 2) and the output probabilities,

$$q_l(z|i', i) \doteq \Pr\{Z_l = z | I_{l-1} = i', I_l = i\}. \tag{6}$$

Since in our case $Z_l = I_l$,

$$q_t(z|i', i) = \begin{cases} 1, & z = i \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

We input the sequence I_1^L into a memoryless channel and observe y_1^L (a length-L sequence of L-value vectors y_l, each of dimension Q) at the output. Our aim is to determine from the whole sequence y_1^L the conditional probabilities

$$\Pr\{I_l = i | y_1^L\} = \frac{p(I_l = i, y_1^L)}{\sum_{i \in \mathcal{I}} p(I_l = i, y_1^L)} \tag{8}$$

of the individual states I_l, $l = 1, \ldots, L$. According to [7], $p(I_l = i, y_1^L)$ can be computed by

$$p(I_l = i, y_1^L) = \alpha_l(i) \cdot \beta_l(i) \tag{9}$$

with the forward recursion

$$\alpha_l(i) = \sum_{i' \in \mathcal{I}} \gamma_l(i', i) \cdot \alpha_{l-1}(i') \tag{10}$$

and the backward recursion

$$\beta_l(i) = \sum_{i' \in \mathcal{I}} \gamma_{l+1}(i, i') \cdot \beta_{l+1}(i') \, , \tag{11}$$

where the γ values are defined as

$$\gamma_l(i', i) = \sum_{z \in \mathcal{I}} p_l(i|i') \cdot q_l(z|i', i) \cdot p(y_l | Z_l = z) \tag{12}$$

$$= p_l(i|i') \cdot p_c(y_l | I_l = i) \, ; \tag{13}$$

[3] We use small letters to refer to realizations and capitals to refer to the corresponding random variables. The latter are omitted as long as there is no risk of confusion.

[4] This is approximately true due to the bit interleaver in Fig. 1.

(13) follows from (7). As we insert (3) into (13), we get product-terms such as $\prod_{l'=1}^{l} g(y'_l, \sigma_{A,l'})$ from the recursions (10), (11). Note that they cancel out in (8), i.e., we don't need to know the variances σ_A, which comes down to the known fact [6] that we may scale α, β in the BCJR recursions to our convenience without changing the decoding result.

The actual source decoder output is given by the minimum mean-square estimator [2]

$$\tilde{X}_l = \sum_{i \in \mathcal{I}} \hat{x}_i \cdot \Pr\{I_l = i \mid y_1^L\}, \tag{14}$$

where the results from (10)–(12), inserted into (9) and (8), are used and $\hat{x}_i$ denotes the quantizer reproduction value with the index i.

4. ITERATIVE DECODING

After SISO channel decoding and BCJR source decoding as described above we could, in principle, compute the mean-square estimates (14). Channel decoding up to now was, however, carried out without a priori information from the source; hence, there is room for quality improvements by using the new information about the indexes, generated by the source decoding, for another round of channel decoding. As again new information can be generated by the source decoder, after it gets improved inputs from the channel decoder, we can carry out several iterations of SISO source and channel decoding. At this point we face the problem that the channel code is binary, which means its decoder requires bit-based a-priori information, but the a-posteriori probabilities we computed by (8) are for quantizer *indexes*, i.e., groups of bits. Hence, we compute

$$\Pr\{B_{l,q} = c | y_1^L\} = \sum_{i \in \mathcal{I}: \mu_q(i) = c} \Pr\{I_l = i | y_1^L\} \tag{15}$$

for individual bits, with $c \in \{0, 1\}$, i.e., we sum all the index probabilities from (8) with a certain bit value c at the bit-position q. Within the iterations the source decoder sends its messages to the channel decoder (see Figure 1) in terms of L-values. Hence, we convert the probabilities (15) to L-values according to

$$\mathrm{L}^{(S)}(B_{l,q}) = \log_e \frac{\Pr\{B_{l,q} = 0 | y_1^L\}}{\Pr\{B_{l,q} = 1 | y_1^L\}} \,. \tag{16}$$

The operations described by (15) and (16) are carried out by the "$I \to B$"-block in Figure 1. As the L-values in (16) contain the new information generated by the source decoder but also its a-priori information, we subtract the latter (to avoid looping back information) to generate the extrinsic L-values $L_E^{(S)}(B_{l,q})$ from the source redundancies, which form the a-priori information for the channel decoder in the next iteration (see Figure 1, where "E" and "A" indicate "extrinsic" and "a priori" respectively whilst "(C)" and "(S)" indicate "channel decoder" and "source decoder" respectively).

5. OPTIMIZED QUANTIZER BIT MAPPINGS

If we assume a low-pass correlation for the source, the value of the sample X_l will be close to X_{l-1}. Thus, if the input at "time" l–1 is scalar quantized, e.g., by reproduction value $\hat{x}_1$ (see Figure 4), the next quantized value at "time" l will be $\hat{x}_0, \hat{x}_1$, or $\hat{x}_2$ with high probability, while the probability for, say, $\hat{x}_7$ is small. For the transmission, the indexes i of the quantizer reproduction values $\hat{x}_i$ are mapped to bit-vectors $b = \mu(i)$ as illustrated by Figure 4.

If the channel code is strong, we can idealize this situation by assuming, that its extrinsic output information

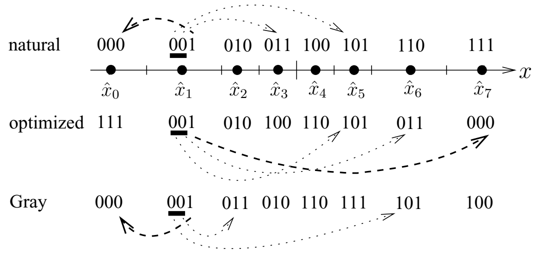

Figure 4: Bit mappings μ for a Q-bit quantizer ($Q = 3$)

is perfect; within the iterative decoding scheme this means that the source decoder tries to generate extrinsic information for a particular data bit, while it knows all *other* bits exactly. The situation is illustrated by Figure 4: as an example, we consider the case that the bit-vector of the quantizer reproduction value $\hat{x}_1$ has been transmitted and that the two leftmost bits are known (both are zero in all mappings), due to the a-priori information from the channel decoder. We now try to generate extrinsic information for the rightmost bit from the source redundancies.

If we use the natural or the Gray mapping and flip the rightmost bit, we end up with quantizer value $\hat{x}_0$ instead of $\hat{x}_1$. Since $\hat{x}_0$ and $\hat{x}_1$ are neighbors in the source signal space we cannot decide with low error probability whether the rightmost bit is "one" or "zero," because both $\hat{x}_0$ *and* $\hat{x}_1$ are highly probable due to the low-pass correlation of the source samples. The situation is different if we use the optimized mapping: since we jump to quantizer value $\hat{x}_7$ (which is highly improbable) if we flip the rightmost bit, we can take a safe decision in favor of "one." Thus, the extrinsic information generated by the source decoder is strong and it will aid the channel decoder in the next iteration.

The example above suggests the concept of how to optimize the bit mapping: we have to allocate bit-vectors to the quantizer reproduction values such that, if we flip one of the bits, each pair of reproduction values has a *large* distance in the source signal space.[5] A detailed description of a feasible optimization algorithm is given in [3].

6. CHARACTERISTIC CURVES OF SOURCE DECODER AND SNR CONTOUR LINES

Following [5], we simulate the source decoder alone, for various amounts of a priori information[6] $I(B_{l,q}; \mathrm{L}_A^{(S)}(B_{l,q}))$ and measure thereby the mutual information $I(B_{l,q}; \mathrm{L}_E^{(S)}(B_{l,q}))$ between the information bits and their extrinsic L-values at the source decoder output. The quantities I_A and I_E in plots below are the averages of $I(B_{l,q}; \mathrm{L}_A^{(S)}(B_{l,q}))$ and $I(B_{l,q}; \mathrm{L}_E^{(S)}(B_{l,q}))$, respectively, over the whole block of information bits and many simulated blocks.

We used the BCJR algorithm as described above for source decoding and we tested correlated sources with $a = 0.5$ and $a = 0.9$ scalar (Lloyd-Max) quantized by $Q = 5$ bits. Figure 5 shows the results. 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 values of the Lloyd-Max quantizer that is used.

Over all, we note that the optimized mapping enables the highest extrinsic output for an a priori mutual information of $I_A = 1$ as it is designed to do.

In [5], it was shown how to determine the BER from

[5]This still holds, if the source signal is high-pass correlated.

[6]$I(A; B)$ indicates the mutual information between random variables A and B.

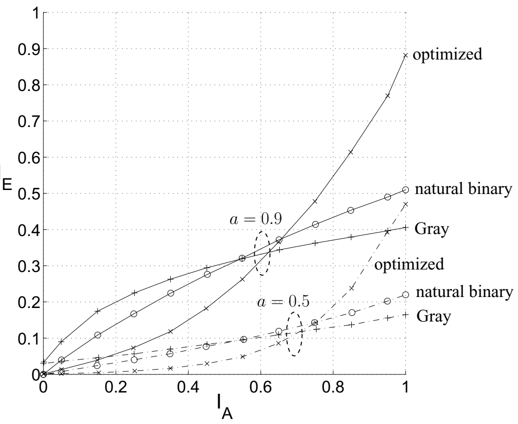

Figure 5: Characteristic curves for filtered source ($a = 0.9$ and $a = 0.5$), $Q = 5$, several mappings.

the EXIT chart thus producing BER contour lines. In our case it is possible to go a step further and produce source SNR_{dB} contour lines, which are 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 conditionally Gaussian distributed). We then transmitted bit-mapped symbols over this channel and determined the corresponding $SNR_{\text{dB}} \doteq 10 \log_{10} \frac{\sum_l x_l^2}{\sum_l (x_l - \tilde{x}_l)^2}$ after using the minimum mean-square estimator (14).

Note that the error-free performance for the scalar, optimal quantization of a Gaussian source with "high rate" equals $SNR_{\text{dB}} = (6.02 \cdot Q - 4.35)$ dB, i.e., with $Q = 5$ the best possible source SNR_{dB} is 25.75 dB. It is reached in the upper right corner of the contour plot in Figure 6, which can be put as an "SNR-template" on the EXIT charts.

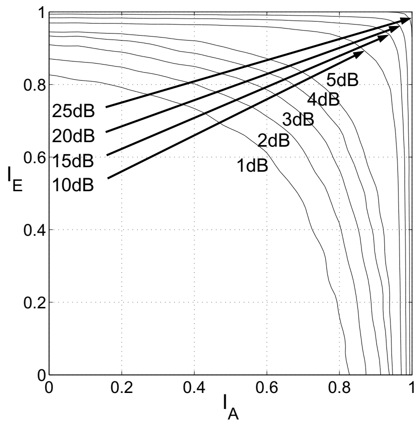

Figure 6: SNR_{dB} contour lines for $Q = 5$ on the EXIT chart.

7. RESULTS

We simulated a system with $a = 0.9$ (filter coefficient, Gauss-Markov source) and a memory-2 recursive binary 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, resulting in a channel code rate of $R_c = 0.9$. The simulation was performed at $E_s/N_0 = -0.1$dB. Results are shown in Figure 7. It is obvious that the optimized mapping works much better than the natural mapping, as the characteristic curves intersect at point "B" which corresponds, by use of Figure 6, to an

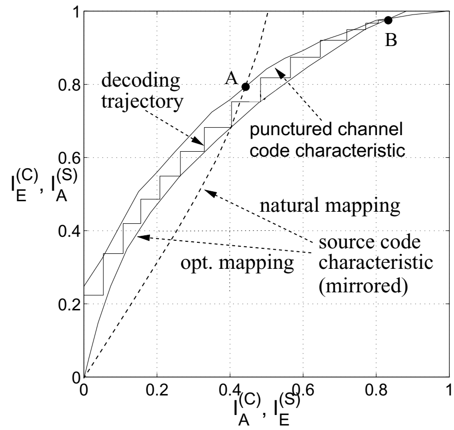

Figure 7: EXIT chart decoding trajectory for transmission of 10000 samples of the correlated Gaussian source with $a = 0.9$ at $E_s/N_0 = -0.1$dB; channel code rate $R_c = 0.9$.

SNR-performance of about 20 dB. The characteristic curve of the natural mapping, on the other hand, intersects with that of the convolutional code already at point "A," which corresponds to a source SNR of only 3 dB. Both predicted SNR-results match the true simulation results well which is also confirmed by the decoding trajectory inserted into Figure 7.

8. CONCLUSIONS

We showed to what extent correlations existing in a Gaussian source may be exploited in an iterative source-channel decoder. As a tool for analysis we used EXIT charts to which we added source SNR contour lines that allow to determine the transmission quality after convergence of iterative decoding. We verified the EXIT tool by plotting the decoding trajectory of a simulated system on the EXIT chart.

REFERENCES

[1] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423/623–656, July/October 1948.

[2] 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.

[3] N. Görtz, "Optimization of bit-mappings for iterative source-channel decoding," in *Proceedings of the International Symposium on Turbo Codes & Related Topics*, pp. 255–258, Sept. 2003.

[4] A. Schaefer, N. Görtz, and J. Hagenauer, "Analysis tools for iterative source-channel decoding," in *Proceedings of the International Symposium on Turbo Codes & Related Topics*, pp. 579–582, Sept. 2003.

[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.

[6] 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.

[7] 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.