Analysis and Performance of Iterative Source-Channel Decoding

Norbert Görtz

Faculty of Engineering, University of Kiel, Kaiserstr. 2, 24143 Kiel / Germany
Tel: +49 431 77572 406, Fax: +49 431 77572 403
E-mail: norbert.goertz@ieee.org

Abstract: *A new iterative source-channel decoding technique is proposed, which approximates optimal joint source-channel decoding for the channel-encoded quantizer indices of various source signals, that are transmitted in parallel. The algorithm is realized by two types of soft-in/soft-out decoders, which alternately utilize the channel-code and the source-code redundancies to compute and interchange extrinsic information on the data-bits within several iterations. After the last iteration the output signals of the receiver are computed by mean-square estimators, which exploit the approximately determined a-posteriori probabilities of the quantizer indices.*

Keywords: Joint Source-Channel Decoding, Iterative Decoding, APP-algorithm, Optimal Estimation

1. INTRODUCTION

Figure 1 shows the model of a transmission system that is used throughout the paper. The signal-vectors $X_k^1, ..., X_k^M$ are transmitted at each time k. The vectors, which can be thought of as M independent source-signals, but also as the parameters of a speech, audio, or image-codec, are quantized (source-encoded) by the indices I_k^j by N^j bits, $j = 1, ..., M$. The index-bits are interleaved, commonly channel-encoded, and the codewords $V_k = \{v_{l,k} \in \{0,1\},\ l = 1, ..., N_V\}$, $N_V > N \doteq \sum_{j=1}^{M} N^j$, are transmitted over an AWGN-channel. Coherently detected binary phase-shift keying is assumed

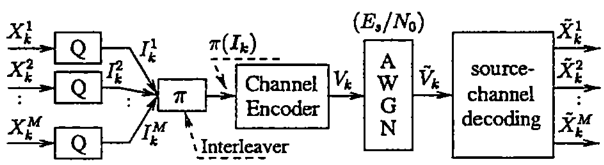

Figure 1: Model of the transmission system

for the modulation. The conditional p.d.f. of the received value $\tilde v_{l,k} \in \mathbb{R}$ at the channel output, given that the code bit $v_{l,k} \in \{0,1\}$ has been transmitted, is given by

$$p_c(\tilde v_{l,k}|v_{l,k}) = \frac{e^{-\frac{1}{2\sigma_n^2}(\tilde v_{l,k}-v'_{l,k})^2}}{\sqrt{2\pi}\sigma_n}\Bigg|_{v'_{l,k}=1-2\cdot v_{l,k}}, \quad (1)$$

with the variance $\sigma_n^2 = \frac{N_0}{2E_s}$. E_s is the energy that is used to transmit each channel-code bit and N_0 is the one-sided power spectral density of the channel noise. The joint conditional p.d.f. $p_c(\tilde V_k|V_k)$ of the vector $\tilde V_k \in \mathbb{R}^{N_V}$ to be received, given that the codeword $V_k \in \{0,1\}^{N_V}$ has been transmitted equals the product of (1) over all N_V code-bits since the channel noise is "white" and normally distributed.

If the signals X_k^j are autocorrelated, adjacent indices I_{k-1}^j, I_k^j show dependencies. They are modeled by first-order stationary Markov-processes, which are described by the index-transition probabilities $P_{k|k-1}(\lambda^j|\mu^j) \doteq P(I_k^j = \lambda^j | I_{k-1}^j = \mu^j)$, with $\lambda^j, \mu^j = 0, ..., 2^{N^j}-1$ and $j = 1, ..., M$. It is assumed that these probabilities and the distributions of the indices are known. The indices are assumed to be *mutually* independent. This simplifies the analysis and the realization of source-channel decoding, and it is at least approximately fulfilled in many practically relevant situations.

In the terminology of turbo-codes the coding scheme is serially concatenated since the redundancies of the indices, i.e. the time-based dependencies and the non-uniform probability distributions, form the "outer" codes, which are entirely encoded by the "inner" channel code.

2. OPTIMAL DECODING

For the system model stated above, the goal is to minimize the distortion of the decoder output signals $\tilde X_k^j$ due to the channel noise, i.e. we would like to perform optimal joint source-channel *de*coding (JSCD) for a *fixed* transmitter. This allows to improve existing systems by the algorithms stated below with decoder-only modifications. The mathematical formulation of the optimization criterion is given by the conditional expectation of the mean-squared error (accumulated over all signals):

$$D \doteq E\Big\{\sum_{j=1}^{M}\|\tilde X_k^j - X_q^j(I_k^j)\|_2^2 \,\Big|\, \tilde V_0^k\Big\} \underset{\tilde X_k^j}{\longrightarrow} \min. \quad (2)$$

This distortion measure has to be minimized over the output vectors $\tilde X_k^j$ of the joint source-channel decoder. In (2) $X_q^j(I_k^j)$ is the entry with the index I_k^j of the quantization table, which is used to quantize the vector X_k^j, and $\tilde V_0^k \doteq \{\tilde V_k, \tilde V_{k-1}, ..., \tilde V_0\}$ is the set of channel words which were received up to the current time k.

Let the realizations $\boldsymbol{\mu} = \{\mu^1, ..., \mu^M\}$ of the set of indices $\boldsymbol{I_k} = \{I_k^1, ..., I_k^M\}$ take on "values" from the set $\boldsymbol{S}$, i.e. $\boldsymbol{\mu} \in \boldsymbol{S}$, with

$$S \doteq S^1 \times .... \times S^M \,, \quad S^j \doteq \{0, ..., 2^{N^j}-1\} \,. \quad (3)$$

Then, the minimization (2) results in mean-square estimators for the signals $\tilde{X}_k^j$, $j = 1, 2, ..., M$, i.e.

$$\tilde{X}_k^j = \sum_{\forall \lambda^j \in S^j} X_q^j(\lambda^j) \cdot P_k(\lambda^j \mid \tilde{V}_0^k) \,, \quad (4)$$

with the *index a-posteriori probabilities* (APPs)

$$P_k(\lambda^j \mid \tilde{V}_0^k) \doteq P(I_k^j = \lambda^j \mid \tilde{V}_0^k) \,. \quad (5)$$

Introducing the mutual independence of the indices into the general solution in [1] and assuming that a systematic channel code with the codewords

$$V_k = \{\, \pi(I_k), \overbrace{C_k}^{\text{redundancy-bits}} \,\}, \quad I_k \doteq \{I_k^1, I_k^2, ..., I_k^M\} \quad (6)$$

is used, the algorithm to *optimally* compute the APPs (5) can be shown to be given by

$$P_k(\lambda^j | \tilde{V}_0^k) = M_k^{(1)} \cdot P_e(I_k^j = \lambda^j) \cdot p_c(\tilde{I}_k^j \mid I_k^j = \lambda^j) \cdot P_a(I_k^j = \lambda^j) \quad (7)$$

for $\lambda^j \in S^j$, with the *extrinsic index-probabilities*

$$P_e(I_k^j = \lambda^j) \doteq M_k^{(2)} \cdot \sum_{\forall \mu \in S:\, \mu^j = \lambda^j} \cdots \sum \Bigg[p_c(\tilde{C}_k \mid C_k^{(\mu)}) \cdot \prod_{m=1, m \neq j}^{M} p_c(\tilde{I}_k^m \mid I_k^m = \mu^m) \cdot P_a(I_k^m = \mu^m) \Bigg] \quad (8)$$

and the *index a-priori probabilities*

$$P_a(I_k^j = \lambda^j) \doteq \sum_{\forall \mu^j \in S^j} P_{k|k-1}(\lambda^j | \mu^j) \cdot P_{k-1}(\mu^j | \tilde{V}_0^{k-1}). \quad (9)$$

The factors $M_k^{(1)}$ and $M_k^{(2)}$ normalize (7) and (8) in such a way that both equations sum up to one over all $\lambda^j \in S^j$. The notation $C_k^{(\mu)}$ corresponds to the redundancy-bits of the channel code that result from some input index-combination $\mu \in S$.

The equations (7), (8), and (9) form a recursion that allows the computation of the index APPs (5) using only the known quantities which were defined in section 1. At $k = 0$ the "old" APPs are initialized by the "unconditioned" probability distributions of the indices in order to compute (9) for the first time.

3. ITERATIVE DECODING

Unfortunately, the optimal algorithm for JSCD is not feasible if realistic numbers of index-bits are commonly encoded by the channel code. This is due to the summation in (8) that is taken over all possible combinations of the indices excluding only I_k^j.

On principle the probabilities (8) can be computed by the symbol-by-symbol APP-algorithm in [2] (BCJR-algorithm). An efficient implementation is possible, if a *binary* channel code with a regular trellis with a small number of states is used. This will be assumed in the sequel. However, the quantizer-indices are still non-binary: In general they consist of arbitrary numbers of bits. In order to be able to efficiently compute extrinsic *bit*-probabilities from the channel-code but to exploit the correlations of *non-binary* indices, the *iterative source-channel decoding* (iterative SCD) is introduced. The idea is to simplify only the computation of the extrinsic probabilities (8) but to keep the rest of the optimal algorithm.

A possible realization is given by two types of soft-in/soft-out decoders (SISO decoders) which are applied alternately within several iterations at each time k: The SISO *channel* decoder exploits only the channel-code redundancies while the SISO *source* decoder processes only the index correlations. Both SISO decoders accept the received channel-values and the a-priori information for the index-*bits*, and they compute the new (extrinsic) information for the index-*bits* by exploiting only "their" type of redundancy. The extrinsic information is passed to the other SISO decoder as the new a-priori information for the next step of the iteration. An iterative source-channel decoder that is based on this principle, and which directly fits into Fig. 1, is depicted in Fig. 2.

Information is passed between the blocks by log-likelihood-ratios (L-values, [3]). They can be directly derived from the corresponding probabilities, but the use of L-values is advantageous, because normalizing factors cancel out and the APP-algorithm for channel decoding can be implemented with less numerical problems.

The notation $L_{a/e}^{(C/S)}(I_k)$ is used for the a-priori/extrinsic information of the channel/source decoder. It is a *vector* that contains the L-values for all index-bits in the index-set I_k which was defined in (6). The bits $i_{l,k}^j \in \{0,1\}$ are associated with the quantizer indices by $I_k^j = \{i_{l,k}^j : l = 1, 2, ..., N^j\}$, $j = 1, ..., M$.

In the following the L-values will be stated for the index-*bits*. In contrast to that, the vectors of the L-values corresponding to the index-set I_k are noted in Fig. 2 in order to simplify the drawing.

3.1. Soft-In/Soft-Out Channel Decoder

The APP-algorithm in [2] (BCJR-algorithm) is used for SISO channel decoding. It has been discussed in detail e.g. in [3], thus here only the interface to the SISO source decoder is stated. The interleaver is treated as a part of the channel (de)coder.

Besides the channel-outputs $\tilde{v}_{l',k}, l' = 1, .., N_V$, the channel decoder accepts the a-priori L-values

$$L_a^{(C)}(i_{l,k}^j) \doteq \log\left(P_a^{(C)}(i_{l,k}^j = 0) / P_a^{(C)}(i_{l,k}^j = 1)\right), \quad (10)$$

with $j = 1, .., M$, $l = 1, .., N^j$. The APP-algorithm efficiently computes (8) for *single bits*. Accordingly, the *bit*

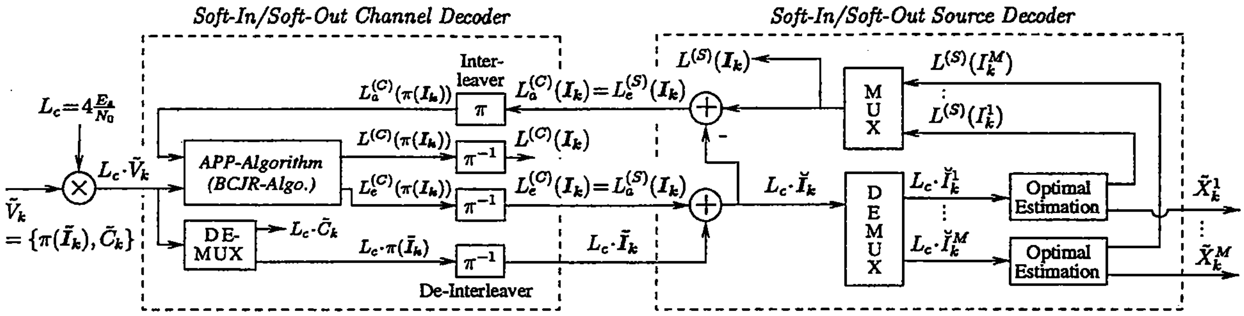

Figure 2: Iterative source-channel decoding

a-priori probabilities, which are required in (10), can be computed from the *index* a-priori probabilities (9) by

$$P_a^{(C)}(i_{l,k}^j = \lambda_l^j) = \sum_{\forall \mu^j \in S^j : \mu_l^j = \lambda_l^j} P_a(I_k^j = \mu^j), \; \lambda_l^j \in \{0,1\}. \quad (11)$$

The channel decoder issues the extrinsic L-values

$$L_e^{(C)}(i_{l,k}^j) \doteq \log\left(P_e^{(C)}(i_{l,k}^j = 0) / P_e^{(C)}(i_{l,k}^j = 1)\right) \quad (12)$$

that are associated with the extrinsic bit-probabilities $P_e^{(C)}(i_{l,k}^j)$. The extrinsic L-values $L_e^{(C)}(i_{l,k}^j)$ are passed to the SISO source decoder as its a-priori information.

3.2. Soft-In/Soft-Out Source Decoder

The idea for the simplification of the optimal JSCD was to replace the extrinsic term in (7). This is achieved by approximating the extrinsic probability (8) for the *index* I_k^j by the product of the corresponding extrinsic *bit*-probabilities. The L-values of the latter have been computed by the SISO channel decoder. It should be noticed that the mutual correlations of the index-bits are neglected by the factorization of the extrinsic information. We obtain

$$P_k(\lambda^j | \tilde{V}_0^k) = M_k^{(S)} \cdot P_a(I_k^j = \lambda^j) \cdot \prod_{l=1}^{N^j} \underbrace{\left(P_e^{(C)}(i_{l,k}^j = \lambda_l^j) \cdot p_c(\tilde{i}_{l,k}^j \mid i_{l,k}^j = \lambda_l^j)\right)}_{\doteq\, p_c(\check{i}_{l,k}^j \mid i_{l,k}^j = \lambda_l^j)} \quad (13)$$

as the approximation of (7). Using (1), the product within the brackets can be expressed by the L-value

$$\log \frac{p_c(\check{i}_{l,k}^j | i_{l,k}^j = 0)}{p_c(\check{i}_{l,k}^j | i_{l,k}^j = 1)} = L_c \cdot \check{i}_{l,k}^j = L_e^{(C)}(i_{l,k}^j) + L_c \cdot \tilde{i}_{l,k}^j$$

with $L_c = 4 \cdot E_s/N_0$. This means that the extrinsic information $L_e^{(C)}(i_{l,k}^j)$ from the channel decoder produces another channel for the index-bits with the modified output values $\check{i}_{l,k}^j$. The redundancy-bits of the channel code do not appear explicitly in (13). Therefore, the SISO source decoder formally computes the index APPs (13) as the Optimal Estimation algorithm (OE) in [4], [1]. Thus, the blocks in Fig. 2 are labeled accordingly.

The estimation of the decoder outputs $\tilde{X}_k^j$ could now be carried out using the index APPs (13). However, the bit a-priori probabilities (11), that were used to perform SISO channel decoding, did not contain the mutual correlations of the index-bits. The first step to insert at least a part of this information into the decoding scheme is to compute new *bit* a-posteriori probabilities from the temporary *index* APPs by

$$P^{(S)}(i_{l,k}^j = \lambda_l^j \mid \tilde{V}_0^k) = \sum_{\forall \mu^j \in S^j : \mu_l^j = \lambda_l^j} P_k(\mu^j | \tilde{V}_0^k), \quad (14)$$

for $j = 1,..,M,\; l = 1,..,N^j$. Using (13), this equals

$$P^{(S)}(i_{l,k}^j = \lambda_l^j \mid \tilde{V}_0^k) = M_k^{(S,1)} \cdot p_c(\tilde{i}_{l,k}^j \mid i_{l,k}^j = \lambda_l^j) \cdot P_e^{(C)}(i_{l,k}^j = \lambda_l^j) \cdot P_e^{(S)}(i_{l,k}^j = \lambda_l^j) \quad (15)$$

with the extrinsic *bit*-probabilities

$$P_e^{(S)}(i_{l,k}^j = \lambda_l^j) = M_k^{(S,2)} \cdot \sum_{\forall \mu^j \in S^j : \mu_l^j = \lambda_l^j} P_a(I_k^j = \mu^j) \cdot \prod_{\nu=1, \nu \neq l}^{N^j} p_c(\tilde{i}_{\nu,k}^j \mid i_{\nu,k}^j = \mu_\nu^j) \cdot P_e^{(C)}(i_{\nu,k}^j = \mu_\nu^j). \quad (16)$$

As stated before, the factors $M_k^{(S,1)}$ and $M_k^{(S,2)}$ are constants that normalize (15) and (16). The extrinsic bit-probability (16) contains the "new" information on the bit $i_{l,k}^j$ due to the index correlation. Since (16) is an extension of (11) because of the additional weighting in the lower line that corresponds to improved information on the *other* bits of the index, we can close the loop and use (16) as the new bit a-priori probability in (10), i.e. $P_a^{(C)}(i_{l,k}^j = \lambda_l^j) \doteq P_e^{(S)}(i_{l,k}^j = \lambda_l^j)$, for a second and further runs of the APP-algorithm for channel decoding. This can be easily implemented by computing (13) and (14), and using (15) in L-value notation according to:

$$L_a^{(C)}(i_{l,k}^j) = L^{(S)}(i_{l,k}^j) - L_c \cdot \tilde{i}_{l,k}^j - L_e^{(C)}(i_{l,k}^j). \quad (17)$$

After several iterations of the SISO decoders, the final index APPs are computed by (13) and the receiver outputs $\tilde{X}_k^j$ are estimated by (4).

4. SIMULATION RESULTS

Independent Gaussian random signals were correlated by a first-order recursive low-pass filter (coefficient $a = 0.9$) in order to generate the signals $X_k^j, j = 1, ..., M$. As source encoders, 5-bit optimal scalar quantizers were used. Their output-bits were spread by a new random-interleaver at each time k and afterwards they were commonly channel-encoded by a terminated rate-1/2 recursive systematic convolutional code (RSC-code, [5]). The codewords were transmitted over the AWGN-channel.

In the first simulation $M = 2$ signals and a memory-2 RSC-code were considered (code-rate $R = 10/24$). The results are depicted in Fig. 3. The mean SNR (averaged

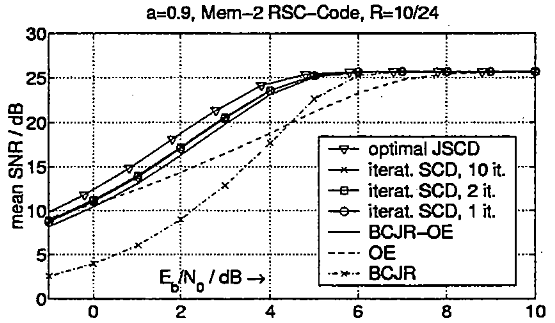

Figure 3: Performance of iterative and optimal joint source-channel decoding. Transmission of $M = 2$ autocorrelated signals ($a = 0.9$) in parallel.

over all signals) is plotted over E_b/N_0, the ratio of the energy E_b per transmitted *data*-bit and the one-sided power spectral density N_0 of the channel-noise.

Clearly, the "optimal JSCD" decoder works best, followed by the iterative decoding (curves "iterat. SCD"). An additional decoder was realized (curve "BCJR-OE") which used the BCJR-algorithm for channel decoding with "zero" a-priori information followed by Optimal Estimation (OE) for source decoding, i.e. the scheme had the same constituent algorithms as the iterative SCD. Due to its better initialization the iterative SCD outperforms the "BCJR-OE"-decoder, even if only one iteration is carried out. In contrast to "BCJR-OE", the iterative SCD utilizes the time-based correlations of the index-bits for channel decoding, but obviously no advantage can be taken of the mutual dependencies of the index-bits since the performance hardly increases with the number of iterations. This can be explained by the short length of the channel code which does not allow sufficient interleaving to make the extrinsic bit-information from the source decoder "virtually" independent for the channel decoder.

It is known that the performance of turbo codes and their iterative decoding is better, if a code with "long" codewords is used. Therefore, a similar simulation as above was carried out, but with $M = 50$ commonly channel-encoded indices and a memory-4 RSC-code (code-rate $R = 250/508$). The results are depicted in Fig. 4: Now, the second iteration leads to a strong im-

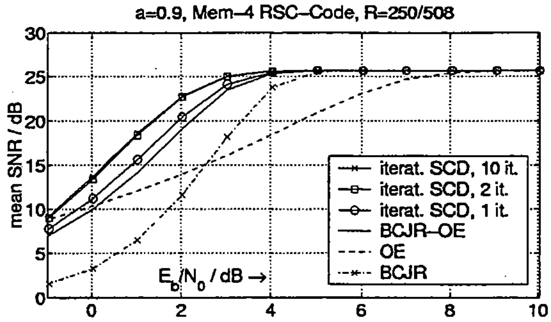

Figure 4: Performance of iterative source-channel decoding and its constituent algorithms. Transmission of $M = 50$ autocorrelated signals ($a = 0.9$) in parallel.

provement, but more than two iterations do not significantly increase the SNR. This is due to the property of the concatenated code, that the "short" source codewords, which consist of just five bits, influence only a small number of RSC-code-bits.

5. CONCLUSIONS

Optimal joint source-channel decoding was approximated by the new iterative algorithm. If binary convolutional channel codes with short memory are used, the iterative SCD can be implemented with moderate complexity. It is suitable for practical source-encoding schemes because the quantizer-indices are allowed to have different, arbitrary numbers of bits. If convolutional codes with "long" codewords are applied, the iterative SCD achieves a strongly better quality of transmission than the other realizable algorithms. The maximum SNR is almost reached after two iterations.

REFERENCES

[1] N. Görtz, "Joint source-channel decoding by channel-coded optimal estimation," in *Proceedings of the 3rd ITG Conference Source and Channel Coding*, Jan. 2000, pp. 267–272.

[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, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[4] T. Fingscheidt and P. Vary, "Robust speech decoding: A universal approach to bit error concealment," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Apr. 1997, vol. 3, pp. 1667–1670.

[5] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Transactions on Communications*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.