Robust Decoding of Variable-Length Encoded Markov Sources Using a Three-Dimensional Trellis

Ragnar Thobaben and Jörg Kliewer, *Member, IEEE*

***Abstract*—In this letter, we present an improved index-based *a-posteriori* probability (APP) decoding approach for the error-resilient transmission of packetized variable-length encoded Markov sources. The proposed algorithm is based on a novel two-dimensional (2-D) state representation which leads to a three-dimensional trellis with unique state transitions. APP decoding on this trellis is realized by employing a 2-D version of the BCJR algorithm where all available source statistics can be fully exploited in the source decoder. For an additional use of channel codes the proposed approach leads to an increased error-correction performance compared to a one-dimensional state representation.**

***Index Terms*—Iterative decoding, joint source-channel coding, residual source redundancy, variable-length codes (VLCs).**

I. INTRODUCTION

MANY authors have considered trellis- or graph-based joint source-channel decoding techniques for the reliable transmission of variable-length encoded data (see e.g., [1], [2]). Especially, a bit-level variable-length code (VLC) trellis representation is proposed in [3], which allows to use the BCJR algorithm [4] as *a-posteriori* probability (APP) decoder. In [5] the bit-level trellises for a VLC and a convolutional channel code are merged for joint source-channel decoding of turbo-encoded VLC source indices. On the other hand, a symbol-level trellis representation for uncorrelated variable-length encoded source data is presented in [6]. This approach is extended in [7] to first-order Markov sources, where the residual source index correlation is additionally exploited in the decoding process.

In the following we present a novel source decoding algorithm for packetized variable-length encoded Markov sources based on a two-dimensional (2-D) symbol-level state representation, which can be regarded as a generalization of the approach in [7]. APP decoding on the corresponding three-dimensional (3-D) VLC trellis can then be carried out by a 2-D modification of the classical BCJR algorithm. In combination with a mean-squares or maximum *a-posteriori* (MAP) estimation the proposed VLC decoder is optimal.

II. TRANSMISSION MODEL

The transmission model is shown in Fig. 1, where $\mathbf{U} = [U_1, U_2, \ldots, U_K]$ denotes one packet consisting of K correlated source symbols. After subsequent (vector-) quantization the resulting indices $I_k \in \mathcal{I}$ from the finite

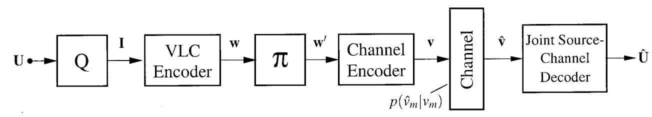

Fig. 1. Model of the transmission system.

alphabet $\mathcal{I} = \{0, 1, \ldots, 2^M - 1\}$ are represented with M bits. In the following, we model the residual index correlation in the index vector $\mathbf{I} = [I_1, I_2, \ldots, I_K]$, which is due to delay and complexity constraints for the quantization stage, by a first-order (stationary) Markov process with index transition probabilities $P(I_k = \lambda \,|\, I_{k-1} = \mu)$ for $\lambda, \mu \in \mathcal{I}$. VLC encoding is then carried out by mapping a fixed-length index I_k to a variable-length bit vector $\mathbf{c}(\lambda) = \mathcal{C}(I_k = \lambda)$ with the prefix code $\mathcal{C}$, resulting in a binary sequence $\mathbf{w} = [w_1, w_2, \ldots, w_N]$ of variable length N where $w_n \in \{0, 1\}$ represents a single bit at bit index n. After interleaving, the bit vector $\mathbf{w}'$ is finally channel-encoded, thus leading to the binary sequence $\mathbf{v}$. The subsequent transmission of one single bit $v_m \in \{0, 1\}$ of $\mathbf{v}$ over the channel characterized by the p.d.f. $p(\hat{v}_m \,|\, v_m)$ results in a soft-bit $\hat{v}_m \in \mathbb{R}$ at the channel output.

III. IMPROVED APP DECODING BY USING A 3-D TRELLIS

In this section we just consider robust VLC source decoding without using a channel code at the encoder side, such that $\mathbf{v} = \mathbf{w}$ in Fig. 1. A new generalized decoding algorithm is presented based on a 3-D VLC trellis representation, where both the bit position n and the source hypotheses $I_k = \lambda, \lambda \in \mathcal{I}$, are treated as separate state variables. This approach allows to exploit all available source *a-priori* information for error protection.

A. Trellis Representation

The proposed symbol-level trellis with the 2-D states $S_k = (\lambda, n) \in (\mathcal{I} \times \mathcal{N}_k)$ represents all possible bit positions n and source hypotheses λ for a certain time instant k, where $\mathcal{N}_k$ denotes the set of all feasible bit positions n for constant k. Furthermore, the resulting trellis allows to change the codetable $\mathcal{C}$ and thus also the bit allocation within a source packet, which may be useful in practical source coding situations with short-time stationary source signals. An example for such a 3-D trellis representation is depicted in Fig. 2 for $K = 5$, $N = 10$, and the Huffman codetables

$$\begin{aligned} \mathcal{C}_1 = \{&\mathbf{c}_1(0) = [1], \mathbf{c}_1(1) = [0,1], \mathbf{c}_1(2) = [0,0,0], \\ &\mathbf{c}_1(3) = [0,0,1]\}, \qquad \text{for } k \in \{0,1,2,3\} \\ \mathcal{C}_2 = \{&\mathbf{c}_2(0) = [1], \mathbf{c}_2(1) = [0,1], \mathbf{c}_2(2) = [0,0]\}, \\ &\text{for } k \in \{4,5\}. \end{aligned} \tag{1}$$

Manuscript received January 13, 2003. The associate editor coordinating the review of this letter and approving it for publication was Dr. A. Haimovich.

The authors are with the Institute for Circuits and Systems Theory, Faculty of Engineering, University of Kiel, 24143 Kiel, Germany (e-mail: rat@tf.uni-kiel.de; jkl@tf.uni-kiel.de).

Digital Object Identifier 10.1109/LCOMM.2003.814718

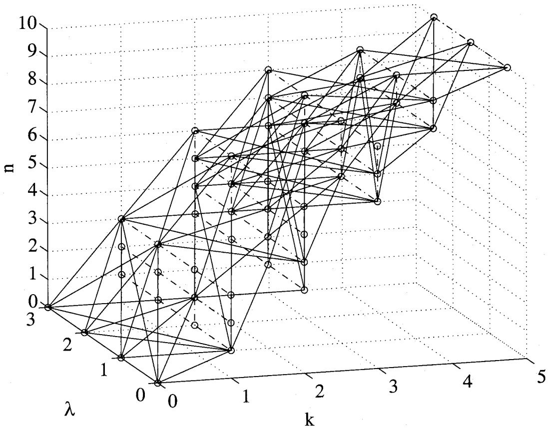

Fig. 2. 3-D VLC trellis for $K = 5, N = 10, \mathcal{C}_1$ and $\mathcal{C}_2$ from (1).

Assuming a certain source hypothesis $I_{k-1} = \mu$, $\mu \in \mathcal{I}$, for the previous time instant we can observe from Fig. 2 that the transition from state $S_{k-1} = (\mu, n_1)$ to $S_k = (\lambda, n_2)$ is caused by the corresponding Huffman codeword $\mathbf{c}_\ell(\lambda)$, $\ell \in \{1, 2\}$, of length $l(\mathbf{c}_\ell(\lambda)) = n_2 - n_1$. Note that the state transitions are unique due to the separation of the state variables n and λ.

B. A-Posteriori Probabilities (APPs)

The 3-D VLC trellis representation can be used in order to calculate soft-outputs in form of APPs $P(I_k = \lambda \mid \hat{\mathbf{v}})$ via a modification of the classical BCJR algorithm [4]. This approach is denoted as 2-D APP decoding in the following. For the sake of clarity we restrict ourselves in the derivation to just a single codetable for the whole packet. In a first step the APPs $P(S_k = (\lambda, n_2) \mid \hat{\mathbf{v}})$ are decomposed as

$$P(S_k = (\lambda, n_2) \mid \hat{\mathbf{v}}) = \frac{1}{p(\hat{\mathbf{v}})} \underbrace{p(\hat{\mathbf{v}}_{n_2+1}^N \mid S_k = (\lambda, n_2))}_{=:\beta_k^{(2D)}(\lambda, n_2)} \cdot \underbrace{p(S_k = (\lambda, n_2), \hat{\mathbf{v}}_1^{n_2})}_{=:\alpha_k^{(2D)}(\lambda, n_2)}. \quad (2)$$

In this case the 2-D forward recursion for the calculation of $\alpha_k^{(2D)}(\lambda, n_2)$ can be stated according to

$$\alpha_k^{(2D)}(\lambda, n_2) = \sum_{\substack{(\mu, n_1) \\ \in (\mathcal{I} \times \mathcal{N}_{k-1})}} \gamma_k^{(2D)}(\lambda, n_2, \mu, n_1)\, \alpha_{k-1}^{(2D)}(\mu, n_1) \quad (3)$$

and for obtaining $\beta_k^{(2D)}(\lambda, n_1)$ we have the backward recursion

$$\beta_k^{(2D)}(\lambda, n_1) = \sum_{\substack{(\mu, n_2) \\ \in (\mathcal{I} \times \mathcal{N}_{k+1})}} \beta_{k+1}^{(2D)}(\mu, n_2)\, \gamma_{k+1}^{(2D)}(\mu, n_2, \lambda, n_1). \quad (4)$$

Both forward and backward recursion may be initialized with the source index probability distribution, where $\alpha_0^{(2D)}(\mu, 0) = \beta_K^{(2D)}(\mu, N) = P(I_k = \mu)$, $\mu \in \mathcal{I}$, for a stationary source process. The term $\gamma_k^{(2D)}(\cdot)$ in (3) and (4), resp., corresponds to the reliability information of a single trellis branch which is given as $\gamma_k^{(2D)}(\lambda, n_2, \mu, n_1) = p(\hat{\mathbf{v}}_{n_1+1}^{n_2} \mid S_k = (\lambda, n_2)) \cdot P_c(\lambda, n_2, \mu, n_1)$ with

$$P_c(\lambda, n_2, \mu, n_1) = \frac{1}{C(n_1, \mu)} \begin{cases} P(I_k = \lambda \mid I_{k-1} = \mu), & \text{for } n_2 - n_1 = l(\mathbf{c}(\lambda)) \\ 0, & \text{otherwise} \end{cases}. \quad (5)$$

The normalization factor $C(n_1, \mu)$ considers that due to the non-stationary VLC trellis not all possible transitions from $S_{k-1} = (\mu, n_1)$ to $S_k = (\lambda, n_2)$ are also valid state transitions. This can for instance be observed in Fig. 2 for the transitions from $k = 4$ to $k = 5$.

Besides the source *a-priori* information $P_c(\cdot)$ in (5), $\gamma_k^{(2D)}(\cdot)$ also includes bit reliability information from the soft-output channel specified by the p.d.f. $p(\hat{\mathbf{v}}_{n_1+1}^{n_2} \mid S_k = (\lambda, n_2))$. Note that due to the unique state transitions the first-order Markov property of the source is utilized as *a-priori* information in *both* forward and backward recursion. This is in contrast to the one-dimensional (1-D) APP decoding approach from [7], where the residual source index correlation can only be exploited in the forward recursion.

Finally, we obtain the APPs $P(I_k = \lambda \mid \hat{\mathbf{v}})$ by calculating the marginal distribution of the conditional probabilities $P(S_k = (\lambda, n_2) \mid \hat{\mathbf{v}})$ in (2) according to

$$P(I_k = \lambda \mid \hat{\mathbf{v}}) = \frac{1}{p(\hat{\mathbf{v}})} \sum_{n_2 \in \mathcal{N}_k} \beta_k^{(2D)}(\lambda, n_2)\, \alpha_k^{(2D)}(\lambda, n_2). \quad (6)$$

These APPs can now be utilized for reconstructing the source symbol packet $\hat{\mathbf{U}}$ at the decoder output via subsequent mean-squares or MAP estimation, leading to an optimal decoder in terms of maximization of the signal-to-noise ratio or minimization of the symbol error rate.

C. APP Decoding for Constant Codeword Length

We now consider the special case of a constant codeword length and show that in this case 2-D APP decoding degenerates to the classical BCJR algorithm working on a 1-D state space. Since in the constant-length case the bit position n at time instant k is uniquely given by $n = k \cdot M$, the states $S_k = (\lambda, n)$ do not depend on n anymore and can thus be directly expressed as source hypotheses $S_k = I_k = \lambda$. Inserting this into (2) leads to

$$P(I_k = \lambda \mid \hat{\mathbf{v}}) = \frac{1}{p(\hat{\mathbf{v}})} \beta_k(\lambda)\, \alpha_k(\lambda) \quad (7)$$

with $\alpha_k(\lambda) = \alpha_k^{(2D)}(\lambda, k \cdot M)$ and $\beta_k(\lambda) = \beta_k^{(2D)}(\lambda, k \cdot M)$, respectively. Thus, the α- and β-recursions are identical to those for the BCJR algorithm formulated with the 2^M-ary source indices I_k. This property is especially useful in the case of multiple codetables where now constant-length codes can be optimally decoded within a single trellis as well.

IV. Results

In order to increase the error robustness of the variable-length encoded bitstream we additionally consider an explicit error

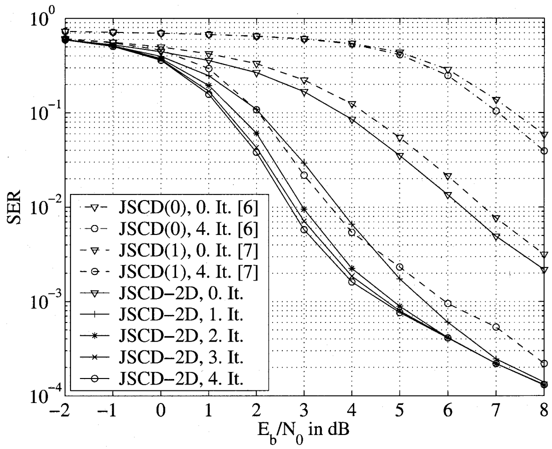

Fig. 3. MAP estimation: symbol error rate (SER) vs. E_b/N_0 for a flat Rayleigh channel ($K = 150$, three Huffman tables with a resolution of $M_1 = 4$, $M_2 = 3$, $M_3 = 4$ bits, each generated for 50 symbols from a strongly correlated AR(1) source with $a = 0.9$).

protection by a channel code (see Fig. 1), where a terminated memory-4 rate-3/4 recursive systematic convolutional code is employed. As joint source-channel decoder an iterative decoding scheme analog to the decoding of serially concatenated codes [8] is applied where the outer constituent decoder corresponds to the 2-D APP VLC decoder from above. As inner decoder a BCJR APP channel decoder is used, where both decoders exchange extrinsic information in form of conditional log-likelihood ratios. It is assumed that all side information is transmitted without errors, including N, K, and the parameters for the individual VLC tables used in different sections of the source packet.

In the simulations we use an AR(1) source process with correlation coefficient $a = 0.9$, and different Huffman tables for the first, the middle, and the last 50 symbols ($K_1 = K_2 = K_3 = 50$) in the source packet corresponding to a resolution of $M_1 = 4$, $M_2 = 3$, and $M_3 = 4$ bits, respectively. For a MAP estimation the resulting symbol error rate (SER) versus the E_b/N_0 is depicted in Fig. 3 for a fully-interleaved BPSK-modulated Rayleigh channel. Both the JSCD(0) and JSCD(1) techniques utilize a 1-D APP decoding approach. The JSCD(0) approach, which is essentially the same method as in [6], only exploits the probability distribution of the source indices, whereas the JSCD(1) method [7] uses the residual source index correlation due to the first-order Markov model. The latter also applies to the JSCD-2D method, however, here the 2-D APP decoding technique from Section III is used. We can see from Fig. 3 that compared to the JSCD(1) technique the JSCD-2D approach leads to an E_b/N_0 gain of approximately 1 dB in the waterfall region. A similar gain can be observed for the case where channel codes are not considered. Generally, this strong increase in decoding performance is due to the fact that the 2-D APP decoding approach is capable of exploiting *all* available source *a-priori* information.

V. Conclusion

As a new result we have presented an improved VLC APP decoding algorithm based on a general 2-D state representation which exploits the residual source correlation of variable-length encoded Markov sources for error correction. This leads to an 3-D symbol-level VLC trellis with unique transitions between adjacent trellis states, where the corresponding APP decoder employs a 2-D version of the BCJR algorithm. When utilizing an additional error protection by channel codes and an iterative source-channel decoding scheme, where the proposed VLC APP source decoder is used as outer constituent decoder, a subsequent MAP estimation leads to a gain of 1 dB in channel signal-to-noise ratio compared to the APP decoding approach previously published in [7].

References

[1] K. Sayood, H. H. Otu, and N. Demir, “Joint source/channel coding for variable length codes,” *IEEE Trans. Commun.*, vol. 48, pp. 787–794, May 2000.

[2] M. Park and D. J. Miller, “Joint source-channel decoding for variable-length encoded data by exact and approximate MAP sequence estimation,” *IEEE Trans. Commun.*, vol. 48, pp. 1–6, Jan. 2000.

[3] V. B. Balakirsky, “Joint source-channel coding with variable length codes,” *Proc. IEEE Int. Symp. Information Theory*, p. 419, June 1997.

[4] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inform. Theory*, pp. 284–287, Mar. 1974.

[5] K. Laković and J. Villasenor, “Combining variable length codes and turbo codes,” *Proc. IEEE 55th Vehicular Technology Conf.*, pp. 1719–1723, May 2002.

[6] R. Bauer and J. Hagenauer, “Iterative source/channel decoding using reversible variable length codes,” *Proc. IEEE Data Compression Conf.*, pp. 93–102, Mar. 2000.

[7] J. Kliewer and R. Thobaben, “Combining FEC and optimal soft-input source decoding for the reliable transmission of correlated variable-length encoded signals,” *Proc. IEEE Data Compression Conf.*, pp. 83–91, Apr. 2002.

[8] S. Benedetto, G. Montorsi, D. Divsalar, and F. Pollara, “Iterative decoding of serially concatenated codes with interleavers and comparison with turbo codes,” *Proc. IEEE Global Telecommunications Conf.*, pp. 654–658, Nov. 1997.