Transactions Papers

Source-Optimized Channel Coding for Digital Transmission Channels

Stefan Heinen and Peter Vary, *Senior Member, IEEE*

***Abstract*—We present a new class of nonlinear block codes called *source-optimized channel codes* (SOCCs), which are particularly designed for parametric source encoding of speech, audio, and video. In contrast to conventional channel codes, the new codes are not optimized for minimizing residual bit-error rate, but maximizing the signal-to-noise ratio of transmitted source codec parameters. The decoding of SOCCs is not based on bit-error correction, but on parameter estimation. We compare SOCCs with other approaches to joint source/channel coding such as channel-optimized vector quantization, channel-constrained vector quantization, unequal error protection, and source-controlled channel decoding. In terms of performance, SOCCs show better robustness if under channel mismatch conditions. For real-world applications, SOCCs are attractive, since the separation of source and channel codec is preserved.**

***Index Terms*—Joint source-channel coding, nonlinear block codes, parameter estimation.**

I. Introduction

In the past, many promising approaches to joint optimization of source and channel coding were proposed. One of the most popular methods is *unequal error protection* (UEP) [3], which assigns high protection to sensitive bits (i.e., bits causing high distortion in the reconstructed signal if inverted by an error), while less important bits are more weakly protected.

Joint optimization is not limited to the coding scheme, but can be extended to the detection/estimation taking place in the receiver. A jointly optimized receiver exploits, e.g., residual redundancy, which results from the imperfections of practical source encoding. So can a channel decoder improve its decisions by exploiting a nonuniform distribution of source bits as *a priori* information (e.g., *source-controlled channel decoding* (SCCD) [4]–[8]). Alternatively, residual redundancy can be used as *a priori* information in an estimation-based source decoder. The advantage of the second approach is that a perception-related quality measurement can be taken into account, e.g., the square of the parameter error in the context of speech transmission [9]–[13].

Farvardin *et al.* [1], [14] propose a joint optimization method, which does not require any explicit channel coding. Instead, the source encoder is directly optimized with respect to the conditions on the disturbed transmission channel. The error-protecting capability of this so-called *channel-optimized vector quantization* (COVQ) is a result of leaving some of the possible quantizer output symbols unused, which implicitly increases the redundancy. Goldsmith *et al.* [15] combine COVQ and UEP in terms of rate-compatible punctured convolutional (RCPC) codes.

Skoglund proposes channel-constrained vector quantization (CCVQ) [2], which adjusts the quantizer regions such that the *a priori* distribution of the codevectors results in an increased redundancy. To achieve higher channel robustness, the code is trained on a subspace of the available code space. He also considers the optimization of redundancy-increasing index assignments for discrete channels. However, no means are given how to determine the *size* of the subspace optimally with respect to the channel.

Bozantzis *et al.* combine vector quantization with a redundancy-increasing index assignment and trains both components alternately [16]. A drawback of their approach is that the number of both codevectors and codewords is constrained to powers of two, which disables a fine-granular adjustment of error-protecting redundancy.

Goodman *et al.* apply simulated annealing to train quantization and channel coding jointly [17]. In the training process, several contiguous cells of an initial quantizer are comprised and assigned the same codeword. This, however, makes it difficult to extend the method to vector quantization.

A joint source/channel coding method, which does not require any training is proposed by Kim [18]. He uses a fixed class of distance-preserving codes, so-called snake-in-the-box codes, to perform a robust, redundancy-increasing index assignment. However, it is difficult to extend this concept to vector quantization. Another drawback is that an adaptation of the code to known source and channel statistics is not possible, which leaves precious information unused.

Paper approved by A. K. Khandani, the Editor for Coding and Information Theory of the IEEE Communications Society. Manuscript received August 13, 2002; revised February 8, 2003; December 11, 2003; November 9, 2004. This work was supported by the Institute of Communication Systems and Data Processing at RWTH Aachen University, Aachen, Germany.

S. Heinen is with the Infineon Technologies AG Development Center Düsseldorf, D-40489 Düsseldorf, Germany (e-mail: stefan.heinen@infineon.com).

P. Vary is with the RWTH Aachen University, Institute of Communication Systems and Data Processing, D-52056 Aachen, Germany (e-mail: vary@ind.rwth-aachen.de).

Digital Object Identifier 10.1109/TCOMM.2005.844936

A. Contribution of This Paper

In this work, we present a new concept of jointly optimizing source and channel coding. We achieve this by a new class of *nonlinear* block codes, which we call *source-optimized channel codes* (SOCCs) [19], [20]. An SOCC is optimized with respect to a quality measure defined in the real-valued domain of source codec parameters [typically the minimum mean-square error (MMSE)] and takes into account both source and channel statistics. The nonlinearity of the code is important to achieve a high degree of freedom for adapting it to both source and channel, which is not offered by a linear code. Unlike a classical channel-optimized index assignment, the SOCC encoder adds redundancy to the transmitted symbols by restricting to a subset of the available alphabet. The SOCC training algorithm (based on the binary switching algorithm (BSA) [21]) is straightforward, and despite its simplicity (e.g., no eigenvalue decomposition needed, as in [2]), very fast due to the new smart update strategy we developed to evaluate the target function.

In contrast to COVQ [1], [14], the quantizer to be used in combination with an SOCC can be trained independently from the channel statistics, i.e., the quantization regions are only determined by the source statistics. Hence, in contrast to COVQ, the separation of source and channel codec is preserved, which is highly desirable from the engineering point of view, as it allows reusing the source codec in other transmission scenarios.

In practice, transmission takes place over continuous output (analog) transmission channels. Therefore, we do not restrict to the simpler case of a discrete channel (as in [2]), but consider the optimization of SOCCs for the continuous additive white Gaussian noise (AWGN) channel. Moreover, we address the important question how to choose the size of the coding subset optimally for a given continuous channel, which was left unanswered by [2]. As a by-product of this investigation, we find an optimized bit assignment to source and channel coding.

The decoding of SOCCs is a further focus of this paper. The optimality criterion of an SOCC decoder basically differs from that of a conventional channel decoder; it is not residual bit- or sequence-error rate minimized, but the MMSE. The required framework for soft decoding of block codes can be found, e.g., in [10], [11], and [22]–[24]. We combine MMSE soft decoding [12] with a recursive, probabilistic approach [25] to exploit natural or residual correlation of source codec parameters at the receiver.

In order to evaluate the performance of our new SOCC approach, we compare it with COVQ, as well as a coding scheme featuring convolutional coding, UEP, and SCCD, and present rich numerical results.

The paper is structured as follows. Section II describes the applied communication model. Section III gives the definition of the new SOCCs, derives the optimum decoder for SOCCs, and describes a sample algorithm for SOCC design. In Section IV, we present the comparison of SOCC with the two benchmark approaches, COVQ and SCCD.

II. Communication Model

We assume a communication system that makes use of model-based source coding, as illustrated in Fig. 1. The basic

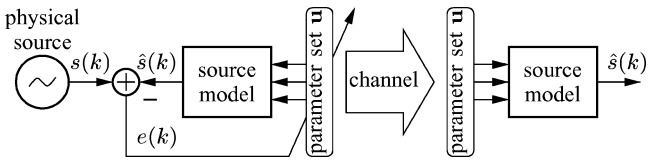

Fig. 1. Generic concept of model-based source coding using the analysis-by-synthesis approach.

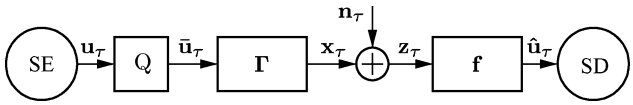

Fig. 2. Communication model. SE: source encoder (parameter source); Q: quantizer with M reproduction levels/vectors; $\mathbf{\Gamma}$: channel encoder, codeword length K bits; $\mathbf{f}$: channel decoder and parameter estimator; SD: source decoder (parameter sink).

idea of this generic coding approach is to approximate the actual source signal $s(k)$ by a model of its underlying physical-generation process. In some cases, the model parameters can be derived directly from the source signal by open-loop processing. Usually more effective is the analysis-by-synthesis (closed-loop) approach shown in Fig. 1, by which the set of model parameters $\mathbf{u}$ is adjusted such that the mean square of the error signal $e(k) = s(k) - \hat{s}(k)$ is minimized. As real-world source signals usually are nonstationary, the parameter set has to be updated and retransmitted periodically. In case of a block-wise update, the source signal is partitioned into a stream of equally sized frames. From the frame at time instant τ, the source encoder extracts a set of real-valued parameters $\mathbf{u}_\tau = [u_{1,\tau}, \ldots, u_{L,\tau}]$, $u_{i,\tau} \in \mathbb{R}$, such as linear predictive coding (LPC) coefficients or gain factors.

As indicated by Fig. 2, we assume a digital transmission, which requires quantizing the source codec parameters to $\bar{\mathbf{u}}_\tau = [\bar{u}_{1,\tau}, \ldots, \bar{u}_{L,\tau}] = \mathrm{Q}(\mathbf{u}_\tau)$, $\bar{\mathbf{u}}_\tau \in \bar{\mathbb{U}}$, where $\bar{\mathbb{U}}$ is the quantizer's reproduction set $\{\bar{\mathbf{u}}^{(1)}, \ldots, \bar{\mathbf{u}}^{(M)}\}$ containing M real-valued vectors of dimension L. The channel encoder $\mathbf{\Gamma}$ maps the quantized parameter sets to K-dimensional codewords

$$\mathbf{x}_\tau = \Gamma[\bar{\mathbf{u}}_\tau], \quad \mathbf{x}_\tau \in \mathbb{C}$$

with $\mathbb{C} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(M)}\} \subseteq \mathbb{X}$ being the channel code. If the system is designed such that $K > \log_2(M)$, we obtain a redundancy-increasing code. For convenience, we assign $\mathbf{x}^{(i)} = \mathbf{\Gamma}[\bar{\mathbf{u}}^{(i)}]$.

Further, we assume a binary modulation scheme, such as binary phase-shift keying (BPSK). Consequently, the elements of the code vectors $\mathbf{x}_\tau$ are binary $\mathbf{x}_\tau = [x_{1,\tau}, \ldots, x_{K,\tau}]$, $x_{i,\tau} \in \{-1, +1\}$, and the energy per transmitted symbol is $E_s = 1$. An extension to m-ary modulation schemes is straightforward and will not be considered here. The additive channel noise is represented by the vector $\mathbf{n}_\tau$, whose scalar components are realizations of independent, zero-mean Gaussian distributed random processes with variance $\sigma_n^2 = N_0/2$. At the receiver, the decoder processes the soft channel-output values $\mathbf{z}_\tau = [z_{1,\tau}, \ldots, z_{K,\tau}]$, $z_{i,\tau} \in \mathbb{R}$ and yields estimates $\hat{\mathbf{u}}_\tau$ of the transmitted codec parameters.

III. Source-Optimized Channel Codes

SOCC is based on the idea to design a redundancy-increasing block code such that the transmission quality of the source codec parameters is maximized. Although this is not completely equivalent to maximizing the quality of the source signal itself at the receiver, there is usually a close relationship between error-caused deviation of the parameters and degradation of the received signal, which justifies this approach.

A. Definition

The new channel encoder is an independent functional unit, which is optimized with respect to the channel-dependent parameter distortion $\mathcal{D}_C = \mathcal{D}[\bar{\mathbf{u}}, \hat{\mathbf{u}}]$, while the distortion due to parameter quantization $\mathcal{D}[\mathbf{u}, \bar{\mathbf{u}}]$ is considered to be fixed. Thus the optimization criterion can be stated as

$$\min \mathrm{E}\left\{\mathcal{D}[\bar{\mathbf{u}}, \hat{\mathbf{u}}]\right\} = \min \mathrm{E}\{\mathcal{D}_C\} \tag{1}$$

where $\mathrm{E}\{\cdot\}$ denotes the expectation of the expression in braces.

Now, given the quantizer reproduction values $\bar{\mathbf{u}}$, their statistics $P(\bar{\mathbf{u}})$, a statistical transfer function of the channel $\mathbf{z} = \Xi(\mathbf{x})$ (described by $p_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|\mathbf{x})$) and a decoder (estimator) $\hat{\mathbf{u}} = \mathbf{f}(\mathbf{z})$, we define an SOCC $\mathbb{C}^*$ as the set of codewords

$$\mathbb{C}^* = \left\{\mathbf{x} | \mathbf{x} = \Gamma[\bar{\mathbf{u}}], \bar{\mathbf{u}} \in \bar{\mathbb{U}}\right\} \tag{2}$$

which result from solving the optimization problem

$$\min_{\Gamma} \mathrm{E}\left\{\mathcal{D}\left[\bar{\mathbf{u}}, \mathbf{f} \circ \Xi \circ \Gamma[\bar{\mathbf{u}}]\right]\right\}. \tag{3}$$

The symbol "$\circ$" denotes concatenation.

Concerning the structure of the code, we do not impose any constraints, so, in general, the code will not be linear, i.e., it will not fulfill the condition

$$\mathbf{x}^{(i)} \oplus \mathbf{x}^{(j)} \in \mathbb{C}^* \quad \forall i, j \in \{1, \ldots, M\} \tag{4}$$

where "$\oplus$" denotes bit-wise modulo-2 addition. As a consequence, the number M of quantizer reproduction values, and thus the number of codewords, can be chosen freely and need not to be a power of two, as it is the case for binary linear codes. This additional degree of freedom allows for a high flexibility from the design perspective, because the number of reproduction values can br exactly selected to satisfy a given quality constraint, and it is possible to add error-protecting redundancy at a fine level of granularity.

SOCCs are ideal to protect single parameters or small parameter groups. Large codewords $\mathbf{x}_\tau$ will usually consist of several independent subcodewords $\mathbf{x}_\tau = [\mathbf{x}_{1,\tau}, \ldots, \mathbf{x}_{\tilde{L},\tau}]$, where $\tilde{L}$ is the number of parameter groups. Accordingly, the received vector is $\mathbf{z}_\tau = [\mathbf{z}_{1,\tau}, \ldots, \mathbf{z}_{\tilde{L},\tau}]$.

Although the general definition of SOCCs given above is not restricted to a particular quality measure, in the rest of the paper, we will focus on the square error

$$\mathcal{D}[\mathbf{u}, \hat{\mathbf{u}}] = \|\mathbf{u} - \hat{\mathbf{u}}\|^2 \quad \text{and} \quad \mathcal{D}_C = \|\bar{\mathbf{u}} - \hat{\mathbf{u}}\|^2 \tag{5}$$

due to its high practical relevance, e.g., as quality indicator for the decoded signal.

B. Decoding of SOCCs

The basic framework for optimum soft decoding in combination with block coding was already derived in [26]. Soft decoding methods in the case of correlated sources were proposed in [12], [13], [27], and [28]. We will briefly review the algorithms and adopt them for the problem of SOCC decoding. In order to simplify notation, we assume parameter-individual quantization and encoding, i.e., $\tilde{L} = L$.

1) Optimum Decoder for a Memoryless Parameter Source: Our objective is to derive the estimator $\hat{\mathbf{u}} = \mathbf{f}(\mathbf{z})$, which recovers the parameter set $\mathbf{u}_\tau$ from the received vector $\mathbf{z}$ with MMSE, i.e.,

$$\min \mathrm{E}\left\{\|\mathbf{u}_\tau - \hat{\mathbf{u}}_\tau\|^2\right\}. \tag{6}$$

We drop the time index τ in the following derivation for the sake of notational simplicity. From estimation theory, it is known that the optimal estimator in this case is given by [29]

$$\hat{\mathbf{u}} = \mathrm{E}\{\mathbf{u}|\mathbf{z}\} = \frac{1}{p(\mathbf{z})} \int_{\mathbb{R}^L} \mathbf{u} p(\mathbf{u}, \mathbf{z}) d\mathbf{u}. \tag{7}$$

The probability density function (pdf) in (7) can be expanded into

$$p(\mathbf{u}, \mathbf{z}) = \sum_{\ell=1}^{M} p\left(\mathbf{z}|\cancel{\mathbf{u}}, \bar{\mathbf{u}}^{(\ell)}\right) P\left(\bar{\mathbf{u}}^{(\ell)}|\mathbf{u}\right) p(\mathbf{u}) \tag{8}$$

where $\mathbf{u}$ can be crossed out in the first conditional pdf, as $\mathbf{z}$ depends only on the quantized parameter set $\bar{\mathbf{u}}$, i.e., $p(\mathbf{z}|\mathbf{u}, \bar{\mathbf{u}}^{(\ell)}) = p(\mathbf{z}|\bar{\mathbf{u}}^{(\ell)})$. Since the channel-encoder mapping $\mathbf{x} = \Gamma[\bar{\mathbf{u}}]$ is deterministic, the equivalence $p(\mathbf{z}|\bar{\mathbf{u}}^{(\ell)}) = p(\mathbf{z}|\mathbf{x}^{(\ell)})$ holds. A given quantizer Q establishes the partitioning $\mathbb{R}^L = \mathbb{Q}_1 \cup \mathbb{Q}_2 \cup \cdots \cup \mathbb{Q}_M$, $\mathbf{u} \in \mathbb{R}^L$. This partitioning implies

$$P\left(\bar{\mathbf{u}}^{(\ell)}|\mathbf{u}\right) = \begin{cases} 1, & \mathbf{u} \in \mathbb{Q}_\ell \\ 0, & \text{else.} \end{cases} \tag{9}$$

Substituting (8) back into (7) and interchanging sum and integration, we obtain

$$\hat{\mathbf{u}} = \frac{1}{p(\mathbf{z})} \sum_{\ell=1}^{M} p\left(\mathbf{z}|\mathbf{x}^{(\ell)}\right) \int_{\mathbf{u} \in \mathbb{Q}_\ell} \mathbf{u} p(\mathbf{u}) d\mathbf{u}. \tag{10}$$

Further, we assume the reproduction values $\bar{\mathbf{u}}^{(i)}$ being centroids of their respective quantization cells, i.e.,

$$\begin{aligned} \bar{\mathbf{u}}^{(\ell)} &= \mathrm{E}\{\mathbf{u}|\mathbf{u} \in \mathbb{Q}_\ell\} \\ &= \frac{\int_{\mathbf{u} \in \mathbb{Q}_\ell} \mathbf{u} p(\mathbf{u}) d\mathbf{u}}{\int_{\mathbf{u} \in \mathbb{Q}_\ell} p(\mathbf{u}) d\mathbf{u}} = \frac{\int_{\mathbf{u} \in \mathbb{Q}_\ell} \mathbf{u} p(\mathbf{u}) d\mathbf{u}}{P\left(\bar{\mathbf{u}}^{(\ell)}\right)}. \end{aligned} \tag{11}$$

Recognizing that the integral in the numerator is identical to that in (10), we finally get the optimal estimator

$$\hat{\mathbf{u}} = \mathbf{f}(\mathbf{z}) = \frac{1}{p(\mathbf{z})} \sum_{\ell=1}^{M} \bar{\mathbf{u}}^{(\ell)} p\left(\mathbf{z}|\mathbf{x}^{(\ell)}\right) P\left(\bar{\mathbf{u}}^{(\ell)}\right). \tag{12}$$

The assumption of individual coding allows applying (12) to each parameter $\hat{u}_i$ independently

$$\hat{u}_i = \frac{1}{p(\mathbf{z}_i)} \sum_{\ell=1}^{m_i} \bar{\mathbf{u}}^{(\ell)} p_{\mathbf{z}_i|\mathbf{x}_i}\left(\mathbf{z}_i|\mathbf{x}_i^{(\ell)}\right) P\left(\bar{u}_i^{(\ell)}\right). \tag{13}$$

This solves the complexity problem inherent in (12), as the number of summations in (12) is $M = \prod_{i=1}^{L} m_i$, while applying (13) to each component requires only $\sum_{i=1}^{L} m_i$ summations.

2) Optimum Decoder for Sources With Interframe Memory: Now we take into account that source codec parameters usually have residual correlation in time

$$\mathrm{E}\{u_{i,\tau} u_{i,\tau'}\} = \varphi_{u_i,u_i}(\tau - \tau') \tag{14}$$

where $\varphi_{u_i,u_i}(\cdot)$ represents the autocorrelation series of the parameter sequence $u_{i,1}, \ldots, u_{i,\tau}$. We describe the correlation property of the parameters by a first-order Markov source with

$$p(u_{i,\tau}|u_{i,1}, \ldots, u_{i,\tau-1}) = p(u_{i,\tau}|u_{i,\tau-1}). \tag{15}$$

An extension of the decoding algorithm to a higher Markov order is straightforward [30] and will not be considered here. A source model with parameters a_i, b_i, and c_i exhibiting property (15) is shown in Fig. 3. To be precise, (15) does not hold exactly for the *quantized* parameters $\bar{u}_{i,\tau}$, but for quantization with more than 2 b per parameter, it is valid with high accuracy. Setting $c_i^2 = (1 - a_i^2)/\sigma_{\tilde{u}_i}^2$, we obtain

$$\varphi_{\bar{u}_i,\bar{u}_i}(0) = 1 \quad \text{and} \quad \varphi_{\bar{u}_i,\bar{u}_i}(1) = a_i. \tag{16}$$

For our simulations, we adjust the coefficients of the model in Fig. 3 according to typical correlation values we measured with standardized speech and audio codecs (ETSI, ITU, MPEG).

To improve the estimate, we take now into account the history of received vectors $\mathbf{z}_1, \ldots, \mathbf{z}_{\tau-1}$. To enhance readability of the equations, in the following, we denote a sequence by applying start and end index to the respective variables, e.g., $\mathbf{z}_1^{\tau-1} = \mathbf{z}_1, \ldots, \mathbf{z}_{\tau-1}$. The estimator can then be stated as $\hat{\mathbf{u}}_\tau = \mathrm{E}\{\mathbf{u}_\tau|\mathbf{z}_1^\tau\}$. Note that this estimator is only optimum under the constraint that no algorithmic decoding delay is allowed. Without this restriction, the estimation can be further improved by exploiting vectors $\mathbf{z}_{\tau+1}, \ldots$ received in future [28], [30], yet we do not consider this case here. With the result of Section III-B.1, we obtain

$$\hat{\mathbf{u}}_\tau = \frac{1}{p(\mathbf{z}_1^\tau)} \sum_{\ell=1}^{M} \bar{\mathbf{u}}^{(\ell)} p_{\mathbf{z}_1^\tau|\mathbf{x}_\tau}\left(\mathbf{z}_1^\tau|\mathbf{x}^{(\ell)}\right) P\left(\bar{\mathbf{u}}^{(\ell)}\right). \tag{17}$$

The conditional pdf in (17) is closely related to the joint pdf $p(\mathbf{z}_1^\tau, \mathbf{x}_\tau)$, which can be interpreted as marginal distribution of $p(\mathbf{z}_1^\tau, \mathbf{x}_{\tau-1}, \mathbf{x}_\tau)$. Applying the chain rule, we obtain the recursion formula

$$p(\mathbf{z}_1^\tau, \mathbf{x}_\tau) = \sum_{\mathbf{x}_{\tau-1}} p\left(\mathbf{z}_\tau, \mathbf{x}_\tau|\mathbf{z}_1^{\tau-1}, \mathbf{x}_{\tau-1}\right) \cdot p\left(\mathbf{z}_1^{\tau-1}, \mathbf{x}_{\tau-1}\right) \tag{18}$$

where the sequence $\mathbf{z}_1^{\tau-1}$ can be omitted because we assumed a Markov source and a memoryless channel. Furthermore, the conditional pdf in (18) can be expressed in known terms

$$\begin{aligned} p(\mathbf{z}_\tau, \mathbf{x}_\tau|\mathbf{x}_{\tau-1}) &= p(\mathbf{z}_\tau|\mathbf{x}_\tau) P(\mathbf{x}_\tau|\mathbf{x}_{\tau-1}) \\ &= p(\mathbf{z}_\tau|\mathbf{x}_\tau) P(\bar{\mathbf{u}}_\tau|\bar{\mathbf{u}}_{\tau-1}). \end{aligned} \tag{19}$$

As the components u_i of the parameter set $\mathbf{u}$ are assumed to be mutually independent, complexity can again be reduced by parameter-individual estimation. It should be mentioned that (18) corresponds to the forward recursion of the famous maximum *a posteriori* (MAP) algorithm first presented by Bahl *et al.* [25]. In our case, the backward recursion is dropped because of the zero-delay constraint.

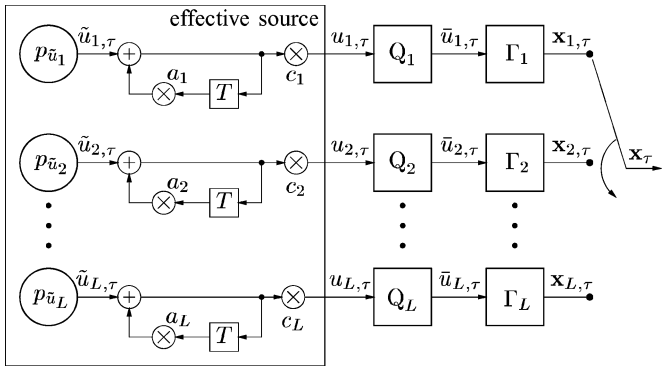

Fig. 3. Model of a parameter source with interframe memory of order 1; inner sources $\tilde{u}_i$ memoryless; individual quantization and SOCC of each parameter.

C. SOCC Design

Solving the optimization problem (3) by exhaustive search would require probing $2^K!/(2^K - M)!$ possible encoder mappings $\boldsymbol{\Gamma}$, which becomes infeasible for codeword lengths K longer than four bits. Therefore, we will apply a suboptimal search algorithm, which finds at least a good local optimum in reasonable time [21], [31]. A further speed-up is achieved by leaving the correlation of subsequent data sets out of account for the *code design*, i.e., we assume $p(\mathbf{u}_\tau|\mathbf{u}_{\tau'}) = p(\mathbf{u}_\tau)$ for all τ'. But note that we do exploit this correlation for the *decoding*, as shown in Section III-B.

We start with a modified expression for the channel-related mean square error (MSE) (Appendix I)

$$\mathrm{E}\left\{\|\bar{\mathbf{u}} - \hat{\mathbf{u}}\|^2\right\} = \sum_{\ell=1}^{M} \left\|\bar{\mathbf{u}}^{(\ell)}\right\|^2 P\left(\bar{\mathbf{u}}^{(\ell)}\right) - \int_{\mathbf{z}\in\mathbb{R}^K} \|\mathbf{f}(\mathbf{z})\|^2 p(\mathbf{z}) d\mathbf{z}. \tag{20}$$

The sum depends only on the fixed quantizer and can be ignored for the optimization. Therefore, a sufficient condition for the optimality of $\boldsymbol{\Gamma}$ is the maximization of the integral term.

As the integration cannot be solved analytically for the Gaussian channel, we apply a numerical approximation by quantizing the channel output values $\mathbf{z}$. This converts the integral into a sum over $\bar{\mathbf{z}} = \mathrm{Q_c}(\mathbf{z})$, where $\bar{\mathbf{z}} \in \{\bar{\mathbf{z}}^{(1)}, \ldots, \bar{\mathbf{z}}^{(W)}\}$. Then, the optimization criterion (3) can be stated as

$$\max_{\boldsymbol{\Gamma}} \sum_{j=1}^{W} \left\|\mathbf{f}\left(\bar{\mathbf{z}}^{(j)}\right)\right\|^2 P\left(\bar{\mathbf{z}}^{(j)}\right). \tag{21}$$

By choosing an appropriate quantizer $\mathrm{Q_c}$ it can easily be ensured that the sum in (21) converges to the integral term of (20), i.e., by increasing the number of reproduction values W, the error due to the applied approximation can be made arbitrarily small.

A powerful means to optimize the encoder mapping is the BSA proposed by Zeger and Gersho [21]. This algorithm iteratively approaches a local minimum of (21) by repeatedly swapping the assignment of two codewords. If $\boldsymbol{\Gamma}$ and $\boldsymbol{\Gamma}'$ represent

the mapping before and after a swap, then one swap iteration is characterized by

$$\mathbf{\Gamma}'\left[\bar{\mathbf{u}}^{(\ell)}\right] = \begin{cases} \mathbf{\Gamma}\left[\bar{\mathbf{u}}^{(\ell)}\right], & \ell \neq i,\ \ell \neq k \\ \mathbf{\Gamma}\left[\bar{\mathbf{u}}^{(i)}\right], & \ell = k \\ \mathbf{\Gamma}\left[\bar{\mathbf{u}}^{(k)}\right], & \ell = i. \end{cases} \tag{22}$$

Note that swapping a code word $\mathbf{x}^{(k)} \in \mathbb{C}$ with some element from the pool of unused bit combinations $\mathbf{x} \in \mathbb{X} \setminus \mathbb{C}$ is allowed as well, which enables modifications not only of the mapping $\mathbf{\Gamma}$ but of the code $\mathbb{C}$ itself. In each iteration step of the BSA, the candidates for a swap are selected such that a maximum growth of the target function (21) is achieved. Appendix II shows how the computational complexity for evaluating a tentative swap can be kept low. The BSA terminates if there is no swap yielding a further increase of the target function left. We will denote the optimized mapping obtained in this way by $\mathbf{\Gamma}^*$.

For the numerical efficiency of the code design, it is important to keep the number W of discretized channel-output values in (21) as low as possible, since W has a big impact on computational complexity and memory consumption. On the other hand, suboptimality due to coarse quantization should be avoided. To find a tradeoff, we carried out two experiments.

With the first experiment, we determined the minimum W to achieve saturation of the estimator performance. Using the communication model according to Fig. 2, with a single, Gaussian-distributed codec parameter $\mathbf{u} = [u]$ with $\mathrm{E}\{u^2\} = \sigma_u^2 = 1$ and $\mathrm{E}\{u\} = 0$, we measured the overall distortion $\mathcal{D}_\mathrm{t} = \|\mathbf{u} - \hat{\mathbf{u}}\|^2$ as a function of W and E_s/N_0 on the AWGN channel. We observed performance saturation if each received symbol $z_{i,\tau}$ is uniformly quantized with at least five bits. We concluded that this quantization accuracy is sufficient to avoid a degradation of the code design.

Next, we investigated the impact of W on the SOCC design algorithm. We expected a similar saturation of the resulting code performance if W is increased to the saturation value found by the first experiment. Instead, the code performance did not change significantly. This is due to the BSA's property to select a mapping $\mathbf{\Gamma}$ by *relative* comparisons. Although the absolute value of the target function (21) is changed by the quantization, the relative relations remain almost the same. Hence, in the SOCC design phase, hard decisions may be applied to the received symbols $z_{i,\tau}$.

D. Bit Allocation Between Source and Channel Coding

For a particular transmission scenario given in terms of transmission rate $r = K/L$ (number of channel bits/number of parameters per set) and E_s/N_0, an optimum number M of reproduction levels can be found, which yields the best performance achievable by means of SOCCs, i.e.,

$$M^* = \arg\min_M \mathrm{E}\left\{\|\mathbf{u} - \hat{\mathbf{u}}\|\right\}. \tag{23}$$

The simulation results of this optimization for a scalar Gaussian distributed parameter u and codeword lengths $K \in \{4, 6, 8\}$ are depicted in Fig. 4. Implicitly, also the optimum bit allocation between source and channel coding is determined, i.e., $\log_2 M^*$ bits are assigned to the source coding, and $K - \log_2 M^*$ bits to the channel coding. Fig. 5 displays the resulting coding rates $\rho = \log_2 M^*/K$, and for reference, the capacity $\mathcal{C}$ of the AWGN channel as functions of E_s/N_0.

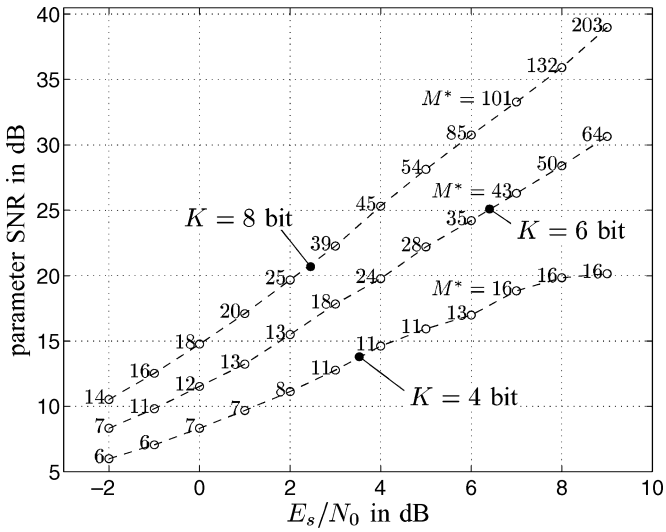

Fig. 4. Best performance achievable by means of SOCCs; individual code design and optimization of M for each simulation point; white Gaussian parameter source, $\sigma_u^2 = 1$; Lloyd–Max quantizer; gross data rate: $r = K/1$ bit per parameter value; AWGN channel; figures: optimum number M^* of reproduction values/codewords; evaluation: Monte Carlo simulation, no quantization of channel output values.

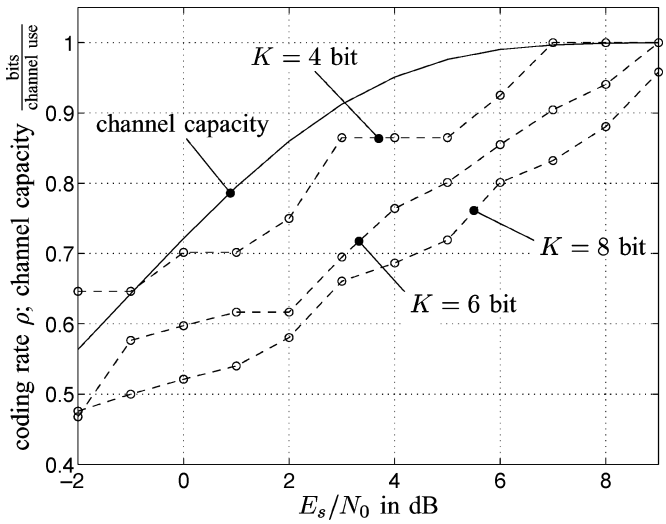

Fig. 5. Optimum coding rates in case of a limited code length; individual code design for each simulation point; white Gaussian parameter source, $\sigma_u^2 = 1$; AWGN channel; gross data rate: $r = K/1$ bit per parameter value.

IV. Performance Comparison

A. Channel-Optimized Vector Quantization

Before comparing SOCC and COVQ performance experimentally, we will discuss the differences between both approaches in terms of their optimization criteria. The COVQ criterion is given by

$$\min_{\mathrm{Q},M,\mathbf{\Gamma}} \mathrm{E}\left\{\|\mathbf{u} - \hat{\mathbf{u}}\|^2\right\} \tag{24}$$

i.e., it is a true joint optimization of the parameter quantizer Q, the number of reproduction values M, and the mapping $\mathbf{\Gamma}$.

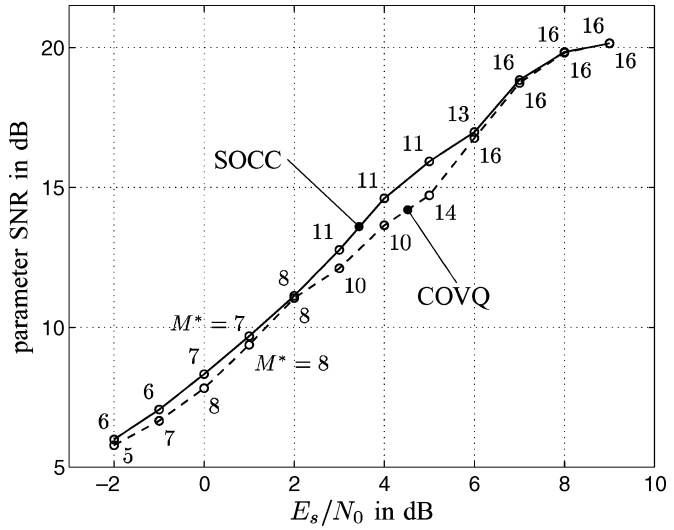

Fig. 6. Comparison of SOCC and COVQ for gross data rate $r = 4$ bit per parameter value; code length $K = 4$; individual code design for each simulation point; figures: numbers of reproduction values M^*; white Gaussian parameter source, $\sigma_u^2 = 1$; AWGN channel.

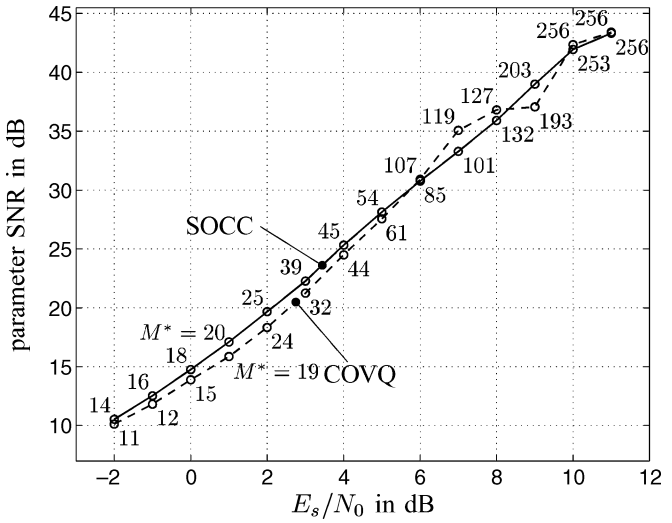

Fig. 7. Comparison of SOCC and COVQ for gross data rate $r = 8$ bit per parameter value; code length $K = 8$; individual code design for each simulation point; figures: numbers of reproduction values M^*; white Gaussian parameter source, $\sigma_u^2 = 1$; AWGN channel.

Denoting the COVQ centroids with $\bar{\mathbf{u}}$ we can decompose (24) to

$$\min_{\mathrm{Q},M,\boldsymbol{\Gamma}} \left[\mathrm{E}\left\{ \|\mathbf{u} - \bar{\mathbf{u}}\|^2 \right\} + \mathrm{E}\left\{ \|\bar{\mathbf{u}} - \hat{\mathbf{u}}\|^2 \right\} \right]. \tag{25}$$

With SOCC, the total MSE for a given number of reproduction values M is

$$\min_{\mathrm{Q}} \mathrm{E}\left\{ \|\mathbf{u} - \bar{\mathbf{u}}\|^2 \right\} + \min_{\boldsymbol{\Gamma}} \mathrm{E}\left\{ \|\bar{\mathbf{u}} - \hat{\mathbf{u}}\|^2 \right\}. \tag{26}$$

In contrast to COVQ, the bit allocation between source and channel coding is explicitly determined by a second optimization step with respect to M

$$\min_{M} \left[\min_{\mathrm{Q}} \mathrm{E}\left\{ \|\mathbf{u} - \bar{\mathbf{u}}\|^2 \right\} + \min_{\boldsymbol{\Gamma}} \mathrm{E}\left\{ \|\bar{\mathbf{u}} - \hat{\mathbf{u}}\|^2 \right\} \right]. \tag{27}$$

Applying the inequality $\min[A + B] \leq \min A + \min B$ to the above criteria, we conclude that only the COVQ criterion ensures a minimization of the total MSE, while the independent design of Q and $\boldsymbol{\Gamma}$ in the case of SOCC does not guarantee this.

In order to verify this consideration, we trained COVQ codebooks for $r = 4$ and $r = 8$ and different E_s/N_0 by applying the original design algorithm proposed by Farvardin [1]. However, as our communication model employs the continuous-output AWGN channel, the update equations for COVQ training contain integrals over the conditional channel pdf, which cannot be solved analytically. We coped with this by approximating the integrals by sums over a precomputed set of channel realizations, as proposed in [10]. In total, we used 10^6 realizations for the source and 10^6 realizations for the channel training set. For the highest E_s/N_0 under consideration, the COVQ training algorithm was initialized by a source-optimized codebook trained with the split-LBG (Linde Buzo Gray) algorithm. Then, proceeding toward lower E_s/N_0, the COVQ codebook of the previous optimization was each time applied as initialization.

Figs. 6 and 7 show the comparison of COVQ and SOCC performance. In many cases, SOCC slightly outperforms COVQ,

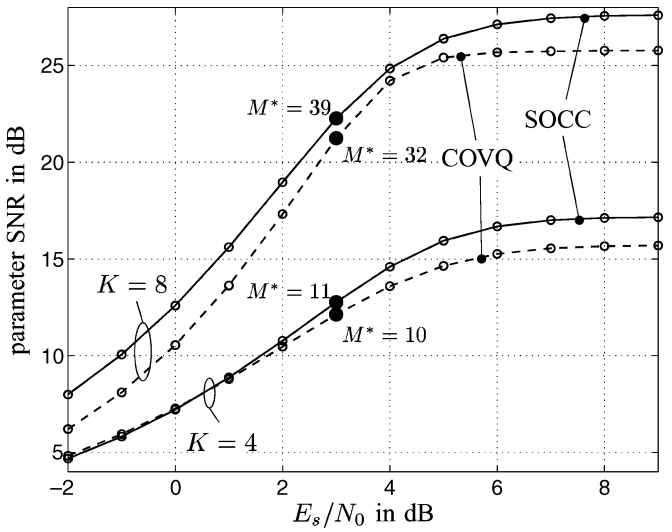

Fig. 8. Comparison of SOCC and COVQ at channel mismatch; design for highlighted simulation points; white Gaussian parameter source, $\sigma_u^2 = 1$; gross data rate $r = K/1$ bit per parameter value, where $K \in \{4, 8\}$; AWGN channel.

which indicates that the applied COVQ training algorithm did not converge close to the optimum. Certainly, the COVQ results could be somewhat improved by more sophisticated training methods, such as simulated annealing. However, what can be derived from the curves, is that SOCC, despite the independent optimization of quantizer and channel encoder, provides a performance close to that of COVQ. This means that SOCC allows optimizing the source encoder independently from the channel without a big performance loss, which is a crucial design advantage, compared with COVQ.

So far we assumed that the encoder (Q, $\boldsymbol{\Gamma}$ in Fig. 2/COVQ) is matched to the conditions on the transmission channel. However, in many practical systems, we encounter time-variant transmission conditions, e.g., due to a fluctuating interference situation. As run-time adaptation is often not possible, the coding scheme should ensure a high transmission quality even under such mismatch conditions. In fact, SOCC is better fitted

TABLE I
MODEL ASSUMPTIONS

transmission rate ($r = K/L$)	4	6
transmitted bits per set (K)	120	120
parameter values per set (L)	30	20
required SNR @ $E_s/N_0 \to \infty$	9 dB	13 dB

to this, as can be verified by the results shown in Fig. 8. Unlike in the previous experiments, the training is performed only for one *single* channel condition, which is highlighted by black dots in the plot. In all other simulation points, SOCC/COVQ are not matched to the channel. We observe that SOCC performs better, especially under good channel conditions, where the usage of the source-optimized quantizer becomes most advantageous.

In conclusion, we can state that due to the beneficial design aspects and its better robustness under channel mismatch, SOCC is an attractive alternative to COVQ for real-world applications.

B. Convolutional Codes

As a second benchmark, we consider a sophisticated coding system based on convolutional codes, UEP, and SCCD [32], [33].

The convolutional coding approach employs a rate-1/2, memory-6 recursive systematic convolutional (RSC) encoder. UEP is achieved by placing sensitive bits at the beginning of a frame, where error protection is high due to the known initial state of the encoder. No tail bits are appended, so that less-sensitive bits can be placed at the end of a frame, where in this case, error protection is weak. This UEP approach despite its simplicity is very effective, and has found application in mobile communications, such as GSM. The convolutional decoder uses an advanced version of SCCD [4], which exploits *a priori* information of the source on the *parameter* level in a two-step fashion, where the first step produces extrinsic information for the second one. However, each decoding step requires a costly soft-output trellis decoding of a received block $\mathbf{z}$ [5], [7], [33].

For the comparison, we assume a block-wise transmission of parameter sets. A fixed number of $K = 120$ bits per parameter set is available for transmission. Two transmission rates are here investigated, $r = 4$ and $r = 6$ bits per scalar parameter value, which means that we can transmit either $L = K/r = 30$ or 20 parameter values per block. An additional constraint is put on the transmission quality to be achieved under noise-free conditions $E_s/N_0 \to \infty$, as shown in Table I. The channel is assumed to be AWGN. Each component u_i of the parameter set $\mathbf{u}$ is individually quantized, yielding $\bar{u}_i$. While a convolutional encoder can easily map $\bar{\mathbf{u}}$ to one single codeword $\mathbf{x}$, with SOCC, we have to restrict the codeword length due to complexity aspects. A first choice was to protect each parameter u_i individually by an SOCC, but the performance of such short codewords was not satisfying. We found a reasonable compromise by encoding pairs of parameters, i.e., u_{2i-1}, u_{2i}, $i \in \{1, \ldots, L/2\}$ are jointly encoded by one codeword. The $L/2$ codewords $\mathbf{x}_i$ obtained in this way are then compiled to a transmission block $\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_{L/2}]$. The SOCC is designed for $E_s/N_0 = 0$ dB, otherwise it is mismatched. The comparison covers both the case of a memoryless parameter source and that of a Markov source (Fig. 3), which models a residual parameter correlation in time (interframe correlation). For transmission rate $r = 4$, we use an SOCC consisting of $M^* = 6^2 = 36$ code vectors out of $2^{4 \cdot 2} = 256$ possible bit combinations, which represents a coding rate of $\log_2(6^2)/8 \approx 0.65$. For $r = 6$, we found $M^* = 12^2$, resulting in a coding rate of $\log_2(12^2)/12 \approx 0.6$.

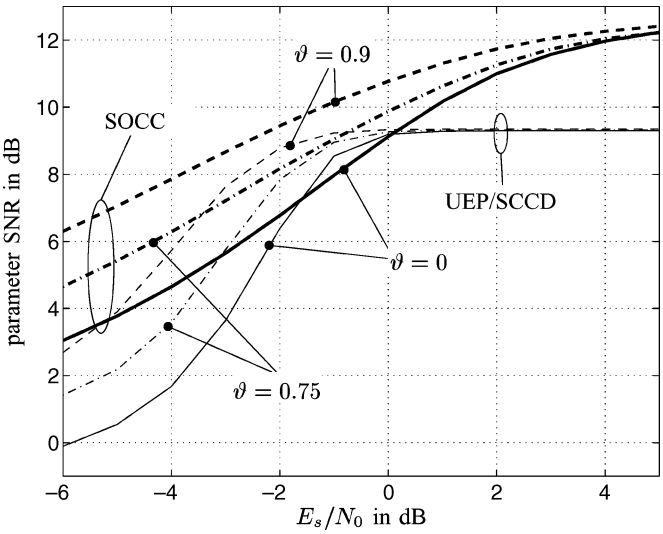

Fig. 9. SOCC versus UEP/SCCD; transmission rate $r = 8/2$ bits per parameter value (two parameter values encoded by one single codeword); Gauss–Markov source, $\sigma^2_{u_i} = 1$, $\mathrm{E}\{u_{i,\tau} u_{i,\tau-1}\} = \vartheta$; setup according to Table I; AWGN channel.

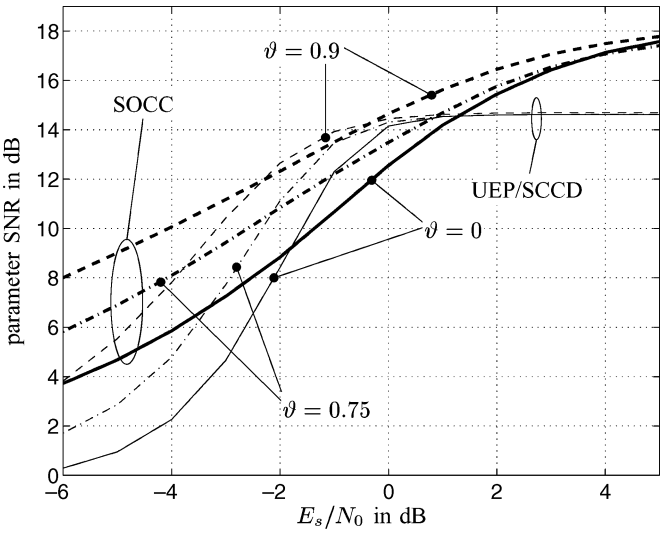

Fig. 10. SOCC versus UEP/SCCD; transmission rate $r = 12/2$ bits per parameter value (two parameter values encoded by one single codeword); Gauss–Markov source, $\sigma^2_{u_i} = 1$, $\mathrm{E}\{u_{i,\tau} u_{i,tau-1}\} = \vartheta$; setup according to Table I; AWGN channel.

Figs. 9 and 10 depict the simulation results. For $r = 4$, the new SOCC approach is clearly ahead; there is only one simulation point ($\vartheta = 0$ and $E_s/N_0 = -1$ dB) where the convolutional code performs slightly better. At transmission rate $r = 6$, the convolutional code gains at medium channel conditions ($E_s/N_0 \in [-2 \text{ dB}, \ldots, 1 \text{ dB}]$), but the lead of SOCC outside this interval is still dramatic. This tendency can be found also at higher transmission rates. Highly desirable is the graceful degradation of transmission quality achievable with SOCCs. The convolutional coding system suffers from the known threshold effect of conventional channel coding.

We concluded from these results that there is not an overall winner, as it depends on the operating point and on the variance of E_s/N_0 which one of the two approaches is preferable. A promising way out of this dilemma is to benefit from both coding strategies by combining them [19].

V. Conclusion

We have proposed a novel joint source-channel coding concept called SOCC based on a new class of nonlinear block codes. The new scheme is intended to protect source codec parameters against transmission errors such that the maximum parameter SNR is achieved at the receiver. This is a crucial innovation, compared with conventional channel coding, which aims at a minimum residual bit-error rate. For the overall system design, SOCC offers the additional degree of freedom that the number of codewords is not bound to a power of two. For the decoding of SOCCs, we employ the optimum estimator in the minimum mean-square sense. We derived this estimator for memoryless sources and for Markov sources under zero-delay decoding constraint.

Compared with COVQ, we found that SOCC has a comparable performance for channel-match conditions and is more robust if channel mismatch occurs. Regarding the overall design of a transmission system, SOCC is easier to implement than COVQ, because the source encoder needs not to be optimized with respect to the channel and can completely be reused in other transmission scenarios. This makes it more attractive for real-world application. We compared SOCC with a sophisticated convolutional coding system featuring UEP and SCCD [32]. Our simulations show that SOCC outperforms this benchmark system, especially for low transmission rates ($r < 6$), and is beaten only in a small medium range of E_s/N_0 for higher rates.

Appendix I
Optimality Criterion

In case of the square error-distance measure, the generic optimality criterion (3) becomes

$$\mathrm{E}\left\{\left\|\bar{\mathbf{u}}-\mathbf{f}(\mathbf{z})\right\|^2\right\}=\sum_{\ell=1}^{M}\int_{\mathbf{z}\in\mathbb{R}^K}\left\|\bar{\mathbf{u}}^{(\ell)}-\mathbf{f}(\mathbf{z})\right\|^2 p\left(\bar{\mathbf{u}}^{(\ell)},\mathbf{z}\right)d\mathbf{z}. \tag{28}$$

Using the abbreviation $\hat{\mathbf{u}}=\mathbf{f}(\mathbf{z})$, the quadratic term expands to

$$\begin{aligned}\left\|\bar{\mathbf{u}}^{(\ell)}-\hat{\mathbf{u}}\right\|^2&=\left(\bar{\mathbf{u}}^{(\ell)}-\hat{\mathbf{u}}\right)\left(\bar{\mathbf{u}}^{(\ell)}-\hat{\mathbf{u}}\right)^T\\&=\left\|\bar{\mathbf{u}}^{(\ell)}\right\|^2+\|\hat{\mathbf{u}}\|^2-2\bar{\mathbf{u}}^{(\ell)}\hat{\mathbf{u}}^T.\end{aligned} \tag{29}$$

The expectation of the mixed term in (29) can be expressed as

$$\begin{aligned}\mathrm{E}\{\bar{\mathbf{u}}\hat{\mathbf{u}}^T\}&=\sum_{\ell=1}^{M}\int_{\mathbf{z}\in\mathbb{R}^K}\bar{\mathbf{u}}^{(\ell)}\hat{\mathbf{u}}^T p\left(\bar{\mathbf{u}}^{(\ell)},\mathbf{z}\right)d\mathbf{z} &(30)\\&=\int_{\mathbf{z}\in\mathbb{R}^K}\left(\sum_{\ell=1}^{M}\bar{\mathbf{u}}^{(\ell)}p\left(\bar{\mathbf{u}}^{(\ell)}|\mathbf{z}\right)\right)\hat{\mathbf{u}}^T p(\mathbf{z})d\mathbf{z}\\&=\int_{\mathbf{z}\in\mathbb{R}^K}\|\mathbf{f}(\mathbf{z})\|^2 p(\mathbf{z})d\mathbf{z} &(31)\end{aligned}$$

where the last line results from (7). Back substitution of (31) into (28) yields (20).

Appendix II
Efficient Swap Evaluation

We define the following abbreviation for the nominator term of the target function (21):

$$\mathcal{Z}_{\mathbf{\Gamma}}\left(\bar{\mathbf{z}}^{(j)}\right):=\sum_{\ell=1}^{M}\bar{\mathbf{u}}^{(\ell)}P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(\ell)}\right)P\left(\bar{\mathbf{u}}^{(\ell)}\right). \tag{32}$$

After a code-word swap, according to (22), it is not necessary to recompute $\mathcal{Z}_{\mathbf{\Gamma}'}(\bar{\mathbf{z}}^{(j)})$ from scratch. It can be computed by updating each $\bar{\mathbf{z}}^{(j)}$, $j\in\{1,\ldots,W\}$ according to

$$\begin{aligned}\mathcal{Z}_{\mathbf{\Gamma}'}\left(\bar{\mathbf{z}}^{(j)}\right)=\mathcal{Z}_{\mathbf{\Gamma}}\left(\bar{\mathbf{z}}^{(j)}\right)&-\bar{\mathbf{u}}^{(k)}P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(k)}\right)P\left(\bar{\mathbf{u}}^{(k)}\right)\\&-\bar{\mathbf{u}}^{(\ell)}P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(\ell)}\right)P\left(\bar{\mathbf{u}}^{(\ell)}\right)\\&+\bar{\mathbf{u}}^{(k)}P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(\ell)}\right)P\left(\bar{\mathbf{u}}^{(k)}\right)\\&+\bar{\mathbf{u}}^{(\ell)}P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(k)}\right)P\left(\bar{\mathbf{u}}^{(\ell)}\right).\end{aligned}$$

Collecting equal terms yields

$$\begin{aligned}\mathcal{Z}_{\mathbf{\Gamma}'}\left(\bar{\mathbf{z}}^{(j)}\right)=\mathcal{Z}_{\mathbf{\Gamma}}\left(\bar{\mathbf{z}}^{(j)}\right)&+\left[\bar{\mathbf{u}}^{(k)}P\left(\bar{\mathbf{u}}^{(k)}\right)-\bar{\mathbf{u}}^{(\ell)}P\left(\bar{\mathbf{u}}^{(\ell)}\right)\right]\\&\cdot\left[P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(\ell)}\right)-P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(k)}\right)\right].\end{aligned} \tag{33}$$

By analogy, the denominator expression

$$\mathcal{N}_{\mathbf{\Gamma}}\left(\bar{\mathbf{z}}^{(j)}\right):=P\left(\bar{\mathbf{z}}^{(j)}\right)=\sum_{\ell=1}^{M}P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(\ell)}\right)P\left(\bar{\mathbf{u}}^{(\ell)}\right) \tag{34}$$

results in

$$\begin{aligned}\mathcal{N}_{\mathbf{\Gamma}'}\left(\bar{\mathbf{z}}^{(j)}\right)=\mathcal{N}_{\mathbf{\Gamma}}\left(\bar{\mathbf{z}}^{(j)}\right)&+\left[P\left(\bar{\mathbf{u}}^{(k)}\right)-P\left(\bar{\mathbf{u}}^{(\ell)}\right)\right]\\&\cdot\left[P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(\ell)}\right)-P\left(\bar{\mathbf{z}}^{(j)}|\mathbf{x}^{(k)}\right)\right].\end{aligned} \tag{35}$$

In case $\mathbf{x}^{(i)}$ is swapped with an element from the pool of unused bit combinations $\mathbf{x}\in\mathbb{X}\setminus\mathbb{C}$, the respective update rules are readily obtained from (33) and (35) by simply substituting $\mathbf{x}^{(k)}=\mathbf{x}$ and $P(\bar{\mathbf{u}}^{(k)})=0$. The value of the target function for the modified mapping $\mathbf{\Gamma}'$ is

$$\sum_{j=1}^{W}\frac{\left\|\mathcal{Z}_{\mathbf{\Gamma}'}\left(\bar{\mathbf{z}}^{(j)}\right)\right\|^2}{\mathcal{N}_{\mathbf{\Gamma}'}\left(\bar{\mathbf{z}}^{(j)}\right)}. \tag{36}$$

Acknowledgment

The authors gratefully acknowledge the anonymous reviewers for their detailed critiques. Their comments and suggestions really helped to improve the clarity of the paper.

References

[1] N. Farvardin, "A study of vector quantization for noisy channels," *IEEE Trans. Inf. Theory*, vol. 36, no. 4, pp. 799–809, Apr. 1990.

[2] M. Skoglund, "On channel-constrained vector quantization and index assignment for discrete memoryless channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 2615–2622, Jul. 1999.

[3] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. COM-36, no. 4, pp. 389–400, Apr. 1988.

[4] ——, "Source-controlled channel decoding," *IEEE Trans. Commun.*, vol. 43, no. 9, pp. 2449–2457, Sep. 1995.
[5] T. Hindelang, S. Heinen, and J. Hagenauer, "Source controlled channel decoding: Estimation of correlated parameters," in *Proc. 3rd ITG Conf. Source Channel Coding*, Munich, Germany, Jan. 2000, pp. 251–258.
[6] S. Heinen, A. Geiler, and P. Vary, "MAP channel decoding by exploiting multilevel source *a priori* knowledge," in *Proc. Eur. Pers. Mobile Commun. Conf.*, Bonn, Germany, Oct. 1997, pp. 467–473.
[7] T. Hindelang and A. Ruscitto, "Kanaldecodierung mit *A Priori*-Wissen bei nicht binären Quellensymbolen" (in German), in *ITG-Fachbericht Codierung für Quelle, Kanal und Übertragung*. Aachen, Germany: VDE Verlag, Mar. 1998, vol. 146, pp. 163–168.
[8] F. Alajaji, N. Phamdo, and T. Fuja, "Channel codes that exploit the residual redundancy in CELP-encoded speech," *IEEE Trans. Speech Audio Process.*, vol. 4, no. 5, pp. 325–336, Sep. 1996.
[9] M. Skoglund and P. Hedelin, "Vector quantization over a noisy channel using soft decision decoding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 5, Adelaide, Australia, Apr. 1994, pp. 605–608.
[10] M. Skoglund, "Soft decoding for vector quantization over noisy channels with memory," *IEEE Trans. Inf. Theory*, vol. 45, no. 4, pp. 1293–1307, May 1999.
[11] M. Skoglund and P. Hedelin, "Hadamard-based soft decoding for vector quantization over noisy channels," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 515–532, Mar. 1998.
[12] T. Fingscheidt and P. Vary, "Robust speech decoding: A universal approach to bit error concealment," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, Apr. 1997, pp. 1667–1670.
[13] ——, "Softbit speech decoding: A new approach to error concealment," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 3, pp. 240–251, Mar. 2001.
[14] N. Farvardin and V. Vaishampayan, "Optimal quantizer design for noisy channels: An approach to combined source-channel coding," *IEEE Trans. Inf. Theory*, vol. IT-33, no. 11, pp. 827–837, Nov. 1987.
[15] I.-A. J. Goldsmith and M. Effros, "Joint design of fixed-rate source codes and multiresolution channel codes," *IEEE Trans. Commun.*, vol. 46, no. 10, pp. 1301–1312, Oct. 1998.
[16] V. Bozantzis and F. H. Ali, "Combined vector quantization and index assignment with embedded redundancy for noisy channels," *Electron. Lett.*, vol. 36, no. 20, pp. 1711–1713, Sep. 2000.
[17] D. J. Goodman and T. J. Moulsley, "Using simulated annealing to design digital transmission codes for analogue sources," *Electron. Lett.*, vol. 24, no. 10, pp. 617–619, May 1988.
[18] S. Kim and D. L. Neuhoff, "Snake-in-the-box codes as robust quantizer index assignments," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2000, p. 402.
[19] S. Heinen, T. Hindelang, and P. Vary, "Channel codes for softbit source decoding: Estimation of correlated parameters," in *Proc. 3rd ITG Conf. Source Channel Coding*, Munich, Germany, Jan. 2000, pp. 259–266.
[20] S. Heinen, S. Bleck, and P. Vary, "Robust speech transmission over noisy channels employing nonlinear block codes," in *Proc. IEEE Speech Coding Workshop*, Porvoo, Finland, Jun. 1999, pp. 72–74.
[21] K. Zeger and A. Gersho, "Pseudo-Gray coding," *IEEE Trans. Commun.*, vol. 38, no. 12, pp. 2147–2158, Dec. 1990.
[22] N. Görtz, "Joint source channel decoding using bit-reliability information and source statistics," in *Proc. IEEE Int. Symp. Inf. Theory*, Aug. 1998, p. 9.
[23] S. Heinen, M. Adrat, O. Steil, P. Vary, and W. Xu, "A 6.1 to 13.3-kb/s variable rate CELP codec (VR-CELP) for AMR speech coding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, Phoenix, AZ, Mar. 1999, pp. 9–12.
[24] T. Fingscheidt, S. Heinen, and P. Vary, "Joint speech codec parameter and channel decoding of parameter individual block codes (PIBC)," in *Proc. IEEE Speech Coding Workshop*, Porvoo, Finland, Jun. 1999, pp. 75–77.
[25] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.
[26] M. Skoglund and P. Hedelin, "Soft decoding for vector quantization in combination with block channel coding," in *Proc. IEEE Int. Symp. Inf. Theory*, Whistler, BC, Canada, Sep. 1995, p. 436.
[27] C. Gerlach, "A probabilistic framework for optimum speech extrapolation in digital mobile radio," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Minneapolis, MN, Apr. 1993, pp. 419–422.
[28] T. Fingscheidt, "Softbit-Sprachdecodierung in digitalen Mobilfunksystemen," Ph.D. dissertation (in German), RWTH Aachen, Aachen, Germany, 1998.
[29] J. L. Melsa and D. L. Cohn, *Decision and Estimation Theory*. New York: McGraw-Hill, 1978.
[30] S. Heinen, "Quellenoptimierter Fehlerschutz für digitale Übertraungskanäle," Ph.D. dissertation (in German), RWTH Aachen, Aachen, Germany, 2001.
[31] P. Knagenhjelm and E. Agrell, "The Hadamard transform—A tool for index assignment," *IEEE Trans. Inf. Theory*, vol. 42, no. 7, pp. 1139–1151, Jul. 1996.
[32] T. Hindelang, S. Heinen, P. Vary, and J. Hagenauer, "Two approaches to combined source-channel coding: A competition in estimating correlated parameters," *AEÜ Int. J. Electron. Commun.*, pp. 364–378, Dec. 2000.
[33] T. Hindelang, "Source Controlled Channel Encoding and Decoding for Mobile Communications," Ph.D. dissertation, TU München, München, Germany, 2002.

Stefan Heinen received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from RWTH Aachen University, Aachen, Germany, in 1994 and 2001, respectively.

From 1995 to 2000, he worked as research assistant at the Institute of Communication Systems and Data Processing of RWTH Aachen University. Beside his main research topic, joint/source channel coding, he was technically responsible for the development of an adaptive multi rate (AMR) speech codec that took part as a candidate in the ETSI standardization contest in 1998. During the years 2000–2003, he was a member of the wireless communications design services team of Synopsys Inc., Herzogenrath, Germany, and provided technical expertise in digital receiver design to international customers. Since 2004, he has been with Infineon Technologies, Düsseldorf, Germany, now focusing on system-level design methodology in the 3G context. His research interests include speech coding, joint source/channel coding, digital receiver algorithms, and system-level design methodology.

Peter Vary (M'85–SM'04) received the Dipl.-Ing. degree in electrical engineering in 1972 from the Technical University of Darmstadt, Darmstadt, Germany. In 1978, he received the Ph.D. degree from the University of Erlangen-Nuremberg, Nuremberg, Germany.

In 1980, he joined Philips Communication Industries (PKI) in Nuremberg, Germany. He became Head of the Digital Signal Processing Group. Since 1988, he has been a Professor with RWTH Aachen University, Aachen, Germany, and Head of the Institute of Communication Systems and Data Processing. His main research interests are speech coding, channel coding, error concealment, adaptive filtering for acoustic echo cancellation and noise reduction, and concepts of mobile radio transmission.