Hybrid Digital–Analog Source–Channel Coding for Bandwidth Compression/Expansion

Mikael Skoglund, *Senior Member, IEEE*,
Nam Phamdo, *Senior Member, IEEE*, and
Fady Alajaji, *Senior Member, IEEE*

***Abstract*—An approach to hybrid digital–analog (HDA) source–channel coding for the communication of analog sources over memoryless Gaussian channels is introduced. The HDA system, which exploits the advantages of both digital and analog systems, generalizes a scheme previously presented by the authors, and can operate for any bandwidth ratio (bandwidth compression and expansion). It is based on vector quantization and features turbo coding in its digital component and linear/nonlinear processing in its analog part. Simulations illustrate that, under both bandwidth compression and expansion modes of operation, the HDA system provides a robust and graceful performance with good reproduction fidelity for a wide range of channel conditions.**

***Index Terms*—Additive white Gaussian noise (AWGN) channels, broadcasting, Gaussian sources, hybrid digital–analog coding, robust coding, source–channel coding, turbo codes, vector quantization.**

I. Introduction

Consider the problem of constructing a communication system for the transmission and reproduction of a discrete-time analog-valued (i.e., with continuous alphabet) source over a discrete-time memoryless Gaussian channel. There are two common approaches for building such a system: analog communication, such as amplitude modulation; and digital communication, which typically consists of quantizing the source, followed by error-control coding ,and digital modulation.

One of the main advantages of digital communication over analog communication is the excellent rate–distortion–capacity performance offered by digital coding systems. This excellent performance is achieved by advanced quantization and error-correcting techniques. There are, however, two fundamental disadvantages associated with digital systems. The first is the "threshold effect" [2], which occurs when the channel signal-to-noise ratio (CSNR) falls beneath a certain threshold and the system performance degrades drastically. This threshold effect is due to the total breakdown of the error-correcting code at low CSNRs and the inherent nonlinearity of the quantizer. During the last two decades, various digital joint source–channel coding systems have been introduced to fight the threshold effect, thus improving the system's error resilience at low CSNRs (see, e.g., the references in [3]–[6]). The second disadvantage is the "leveling-off effect," which refers to the fact that the system performance remains constant even when the CSNR is increased above and beyond the threshold.[1] This leveling-off effect is due to the nonrecoverable error introduced by the quantizer.

Note that analog systems do not suffer from these problems to the same extent, in particular concerning the leveling-off effect. On the other hand, in practice, analog systems generally are inferior to digital systems in terms of rate–distortion–capacity performance—particularly at the designed CSNR.

Recently, Mittal and Phamdo [7] proposed a class of hybrid digital–analog (HDA) joint source–channel coding systems. These systems can theoretically achieve the Shannon rate–distortion–capacity limit at the designed CSNR. Furthermore, they do not suffer from the leveling-off effect—the threshold effect is still inherent, though less severe, in the HDA systems [7]. Thus, systems that mix digital and analog techniques can have some of the advantages of digital systems and some of the advantages of analog systems (e.g., [3], [6], [8]–[11]).

In [6], [12], we presented a vector quantization (VQ) based HDA system. This system is valid only for bandwidth ratios larger than one (bandwidth expansion)—i.e., when the channel bandwidth is greater than the source bandwidth. In this correspondence, we introduce a generalized version of the scheme in [6], [12]. The new system, originally proposed in [1], can be used for either bandwidth expansion or bandwidth compression. We begin with describing a highly general version of the system, and then we investigate in some detail the performance, under both bandwidth compression and expansion, of one important typical case for the communication of a Gauss–Markov source over a memoryless Gaussian channel. The new scheme has several important features that were not present in the original work [6], [12]. In particular, the system studied in detail incorporates a turbo error-correcting code [13] to improve the performance at low CSNRs, and uses a Karhunen–Loéve transform (KLT) to decorrelate the source vector (for the bandwidth compression mode). The new scheme also allows for both linear and nonlinear transformations in the analog part of the HDA system (in [6], [12], only linear transformations are used). Other recent methods which employ a direct source–channel analog mapping or combine digital and analog coding include those in [3], [8], [11], [14]–[19].

In the next section, we provide a general description of the new HDA system. In Section III, we study in detail the system under bandwidth compression for a typical scenario and present simulation results. We also compare the performance of the new scheme with i) a purely analog system, ii) a purely digital system, and iii) the Shannon rate–distortion–capacity limit. We examine the new system under bandwidth expansion in Section IV and evaluate its performance vis-a-vis systems i)–iii) and the scheme studied in [6]. Finally, conclusions are given in Section V.

Some notation used in this correspondence is as follows. Bold-faced characters are used for vectors and matrices. Upper case is used for random entities and lower case for their realizations. The notation $(\boldsymbol{x})_m$ denotes the mth component of vector $\boldsymbol{x}$.

II. System Description

In Fig. 1, we depict a general version of the proposed HDA system. The purpose of the system is to convey the p-dimensional random source vector $\boldsymbol{X} \in \mathbb{R}^p$ over a memoryless Gaussian channel, and reproduce it as $\hat{\boldsymbol{X}}$ at the receiver. The upper part of the transmitter is the *digital* part, and the lower part the *analog* part. In the following we describe in detail how the system works.

Manuscript received May 4, 2005; revised April 12, 2006. The work of M. Skoglund was supported in part by the Swedish Research Council (VR) and the Swedish Governmental Agency for Innovation Systems (VINNOVA). The work of F. Alajaji was supported in part by the Premier Research Excellence Award of Ontario and the Natural Sciences and Engineering Research Council of Canada. The material in this work was presented in part at the IEEE International Symposium on Information Theory, Washington, DC, June 2001.

M. Skoglund is with the School of Electrical Engineering, Royal Institute of Technology, SE–100 44 Stockholm, Sweden (e-mail: skoglund@ee.kth.se).

N. Phamdo is with the Applied Physics Laboratory, Johns Hopkins University, Laurel, MD 20723 USA (email: phamdo@jhuapl.edu).

F. Alajaji is with the Department of Mathematics and Statistic, and the Department of Electrical and Computer Engineering, Queen's University, Kingston, ON K7L 3N6, Canada (e-mail: fady@mast.queensu.ca).

Communicated by S. A. Savari, Associate Editor for Source Coding.

Digital Object Identifier 10.1109/TIT.2006.878212

[1] In some multimedia applications (such as high-definition television (HDTV) broadcasting), the leveling-off effect may be desirable because large variations in signal quality over short periods of time may be annoying to the end users.

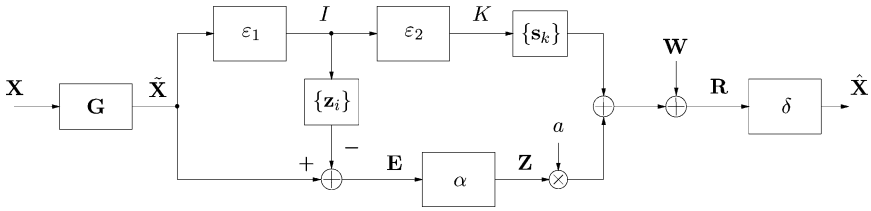

Fig. 1. Proposed hybrid digital–analog system.

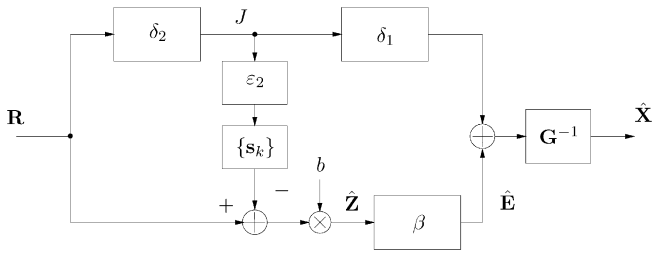

Fig. 2. Proposed decoder structure.

A. Encoding

The source vector is first fed to a linear invertible preprocessing mapping, defined by the matrix $\boldsymbol{G}$. The resulting output, $\tilde{\boldsymbol{X}}$, is then used as input to the first encoder mapping ε_1. In the most general version of the HDA system, ε_1 is a low-delay source or source–channel encoder. Examples include a VQ encoder trained for a noiseless channel, i.e., a source-optimized VQ (SOVQ), a VQ encoder trained for a noisy channel, i.e., a channel-optimized VQ (COVQ), or an SOVQ encoder in tandem with a (short) channel block code. The discrete output $I = \varepsilon_1(\boldsymbol{X}) \in \mathcal{I}_N$, where $\mathcal{I}_N \triangleq \{0, \ldots, N-1\}$ and where we assume $N = 2^L$, is then fed (in its L-bit binary form) to the *high-delay* (n, k) *channel encoder* ε_2, of rate $r_c = k/n < 1$. An example of a specific such mapping, which we use in this correspondence, is the encoder of a rate $r_c = 1/2$ turbo code [13].

A number $M = k/L > 1$ of consecutive outputs from ε_1 are blocked and encoded by ε_2. The output index K of ε_2 is then assigned a q-dimensional channel symbol, $\boldsymbol{s}_K \in \mathbb{R}^q$, from a finite set $\{\boldsymbol{s}_k\}$ of possible symbols. The index I also chooses a vector $\boldsymbol{z}_I$ from the *encoder codebook* $\{\boldsymbol{z}_i\}$. The vector $\boldsymbol{z}_I$ is subtracted from $\tilde{\boldsymbol{X}}$ to form the "error vector" $\boldsymbol{E}$, and this vector is then used as input to the mapping α, with output $\boldsymbol{Z} = \alpha(\boldsymbol{E}) \in \mathbb{R}^r$, where $r = q/M$. Ideally, α is an *analog* mapping, in the sense that $\boldsymbol{Z}$ is a continuous function of $\boldsymbol{E}$. The output $\boldsymbol{s}_K$ of the upper digital part, and a scaled version, $a \cdot \boldsymbol{Z}$, of the output from the analog encoder are added and then fed to a discrete-time channel, with zero-mean additive white Gaussian noise (AWGN) $\boldsymbol{W}$, of variance σ^2 per component. The resulting channel output is denoted $\boldsymbol{R}$. The scaling constant a regulates the contribution of the analog part to the input power of the channel.

Note that for ease of presentation, Fig. 1 does not explicitly illustrate the delay operation associated with the grouping of M consecutive blocks. In the digital part, this operation occurs before the high-delay encoder ε_2; in the analog part, it occurs after[2] the scaling of $\boldsymbol{Z}$ by a. Thus, the transmission of one q-dimensional channel symbol $\boldsymbol{s}_k$ in the digital part corresponds to M different $a\boldsymbol{Z}$'s in the analog part. This point also applies to Fig. 2.

[2]The delay operation occurs after the analog mapping α instead of before. This keeps the complexity of α manageable.

B. Decoding

At the receiver, the *decoder mapping* δ takes the channel output $\boldsymbol{R}$ and outputs a source vector estimate $\hat{\boldsymbol{X}} = \delta(\boldsymbol{R})$. Ideally, this mapping should be chosen to minimize the mean squared-error (MSE) distortion $E\|\boldsymbol{X} - \hat{\boldsymbol{X}}\|^2$, resulting in

$$\delta(\boldsymbol{r}) = E[\boldsymbol{X}|\boldsymbol{R} = \boldsymbol{r}].$$

In general, however, the complexity of implementing this decoder prohibits its use in practice. Therefore, we propose a suboptimal, but more practical, decoder structure, as illustrated in Fig. 2. As shown in the figure, the received vector $\boldsymbol{R}$ is fed to a decoder, δ_2, for the high-delay code. The resulting discrete output $J \in \mathcal{I}_N$ is encoded by ε_2 and assigned a channel symbol. The result is then subtracted from the received vector $\boldsymbol{R}$ and scaled by the constant b, forming an estimate, $\hat{\boldsymbol{Z}}$, of the transmitted analog vector $\boldsymbol{Z}$. This estimate is then fed to the mapping β, with output $\hat{\boldsymbol{E}}$. The purpose of β is to act as decoder for the analog encoder α, and $\hat{\boldsymbol{E}}$ is hence an estimate of the error vector $\boldsymbol{E}$. The output index J of the high-delay channel decoder δ_2 is also fed to a decoder δ_1 for the low-delay source (or source–channel) code. The output $\delta_1(J)$ is then added to $\hat{\boldsymbol{E}}$ and the result is fed to the inverse of the preprocessing map, resulting in the source vector estimate $\hat{\boldsymbol{X}}$.

While Figs. 1 and 2 describe the most general version of the proposed HDA system, we will in the remaining parts investigate some specific instances of the system. In particular, we will first employ the system for bandwidth compression, and then study its performance when used for bandwidth expansion.

III. Bandwidth Compression

In this part, we focus on using the system in Fig. 1, with the decoder in Fig. 2, for *bandwidth compression*, that is, under the assumption that the "total" source vector dimension $M \cdot p$ is larger than the channel signal space dimension q, and consequently the bandwidth ratio $\rho = q/(Mp) = r/p < 1$ (channel dimensions/uses per source dimension). For the sake of clarity and concreteness, we describe our system explicitly in terms of a typical example with $\rho = 1/2$ (the generalization for systems with arbitrary $\rho < 1$ is straightforward). More precisely, we have implemented a system with the following parameters.

The source vector $\boldsymbol{X}$ is $p = 32$ dimensional, drawn from a zero-mean Gauss–Markov source, with normalized correlation 0.9 between samples. Letting $\boldsymbol{R}_{\boldsymbol{xx}} = E[\boldsymbol{X}\boldsymbol{X}^T]$, where ^{T} denotes transposition, have eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_{32} > 0$, the mapping $\boldsymbol{G}$ is defined by the KLT specified by $\boldsymbol{R}_{\boldsymbol{xx}}$. That is, $\boldsymbol{G} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{32}]$, where $\boldsymbol{v}_i$ is the eigenvector of $\boldsymbol{R}_{\boldsymbol{xx}}$ corresponding to eigenvalue λ_i, resulting in $\tilde{\boldsymbol{X}} = \boldsymbol{G}\boldsymbol{X}$ having independent components with variances λ_1 through λ_{32}. The low-delay encoder ε_1 is an SOVQ encoder, of dimension $p = 32$ and size $L = 8$ bits, trained using the Linde, Buzo, and Gray (LBG) algorithm [20], and the encoder codebook $\{\boldsymbol{z}_i\}$ is chosen to be identical to the codebook defining ε_1. The high-delay encoder ε_2 is an $(n = 2048, k = 1024)$, rate $r_c = 1/2$, turbo encoder, with generators $(37, 21)$ (punctured to rate $1/2$) and with a random interleaver [13]. The 8-bit output blocks from ε_1 are blocked into one $k =$

1024-bit "superblock" which is fed to ε_2, resulting in a codeword of length $n = 2048$ bits. The output bits from ε_2, corresponding to the index K, are mapped directly into binary phase-shift keying (BPSK) symbols,[3] with alphabet $\{\pm 1\}$. Consequently, the channel signal space dimension is $q = 2048$ and $\boldsymbol{s}_k \in \{\pm 1\}^{2048}$. Since $M = 1024/8 = 128$, one ε_2-codeword represents 128 source vectors, and hence, $\rho = q/(Mp) = 2048/(128 \cdot 32) = 1/2$ (channel uses per source dimension).

The scaling constant a, in the analog part, is chosen so that a fraction $0 < \Delta < 1$ of the total input power to the channel is assigned to the analog part. That is, since the power in the digital (BPSK) part is 1, the constant a is solved to satisfy

$$\Delta = \frac{Ma^2 E\|\boldsymbol{Z}\|^2}{q + Ma^2 E\|\boldsymbol{Z}\|^2}$$

for a given Δ. The above equation is based on the assumption that the analog and digital signals at the channel input are uncorrelated. Experimental observations indicate that this is a reasonable assumption. The output $\boldsymbol{Z}$ of the analog encoder has dimension $r = q/M = 16$ and $M = 128$ such vectors are transmitted simultaneously with the turbo encoder codewords in one superblock.

The high-delay decoder δ_2 is a turbo decoder for the encoder ε_2, implemented using 10 iterations and given access to the noise variance σ^2. The low-delay decoder δ_1 is defined by a table lookup in a codebook identical to the encoder codebook $\{\boldsymbol{z}_i\}$. The constant b is chosen to minimize the MSE $E\|\boldsymbol{Z} - \hat{\boldsymbol{Z}}\|^2$, under the assumption that δ_2 is powerful enough to correct *all* errors in the digital part and, again, assuming that σ^2 is known at the receiver. That is,

$$b = \frac{1}{a} \cdot \frac{\Delta/(1-\Delta)}{\sigma^2 + \Delta/(1-\Delta)}.$$

What remains to be specified is the analog encoder–decoder pair (α, β). We have investigated two systems, which are described in the following two subsections. Simulations results for bandwidth compression are then provided in Section III-C.

A. Linear Analog Part

The first system employs linear mappings to define α and β. More precisely, α is the linear mapping that projects $\boldsymbol{E}$ onto the subspace spanned by the eigenvectors corresponding to the 16 strongest eigenvalues of $\boldsymbol{R}_{\boldsymbol{xx}}$. Hence, since a KLT is performed on $\boldsymbol{X}$, the mapping α is simply the operation of dropping the 16 low-energy components of $\boldsymbol{E}$, resulting in the $r = 16$ dimensional output $\boldsymbol{Z}$.

The decoder β is defined by the linear mapping of extending $\hat{\boldsymbol{Z}}$ from 16 to 32 dimensions, by filling in zeros in the 16 low-energy dimensions.

B. Nonlinear Analog Part

The second system employs a "discrete approximation" of the optimal, analog, generally nonlinear mappings (α, β) that minimize the MSE, $E\|\bar{\boldsymbol{E}} - \hat{\bar{\boldsymbol{E}}}\|^2$ (so in this case the analog part is not really analog, but "close-to-analog"). The mappings are described as follows.

The components $(\boldsymbol{Z})_m, m = 1, \ldots, 16$, of $\boldsymbol{Z}$ are constrained to belong to a discrete set of equally spaced signal points, i.e., multilevel pulse amplitude modulation (PAM). The resolution is 256 PAM levels per component. Hence, there are $256^{16} = 2^{128}$ different possible values for the transmitted $\boldsymbol{Z}$. The encoder α maps a realization of $\boldsymbol{E}$ into one of these values. Due to the prohibitive encoding complexity of mapping a vector $\boldsymbol{E}$ into one out of 2^{128} possible values, the encoder is split into 16 different encoders $\alpha_m, m = 1, \ldots, 16$, working independently on two-dimensional parts of $\boldsymbol{E}$. More precisely

$$(\boldsymbol{Z})_m = \alpha_m\Big((\boldsymbol{E})_{2m-1}, (\boldsymbol{E})_{2m}\Big), \qquad m = 1, \ldots, 16.$$

Similarly, the decoder is split into 16 independent decoders $\beta_m, m = 1, \ldots, 16$, that each implements hard-decision maximum-likelihood (ML) detection of the transmitted PAM symbol $a \cdot (\boldsymbol{Z})_m$, based on the received value for $(\hat{\boldsymbol{Z}})_m$, and then performs a table lookup in a codebook to assign values for $(\hat{\boldsymbol{E}})_{2m-1}$ and $(\hat{\boldsymbol{E}})_{2m}$. The implementation of the ML detector is based on the assumption that the turbo decoder δ_2 works without errors.

For each $m \in \{1, \ldots, 16\}$, the encoder α_m and decoder β_m are trained to minimize

$$E\Big[\Big((\boldsymbol{E})_{2m-1} - (\hat{\boldsymbol{E}})_{2m-1}\Big)^2\Big] + E\Big[\Big((\boldsymbol{E})_{2m} - (\hat{\boldsymbol{E}})_{2m}\Big)^2\Big]$$

for a fixed channel noise power σ^2, a given Δ, a fixed a and under a constraint on the total transmit power. Note that a power constraint is needed in the design, since even if the PAM constellation for $(\boldsymbol{Z})_m$ and the value of the constant a are fixed, the encoder can still assign different probabilities to different transmitted symbols (note that the PAM symbols have different energy).

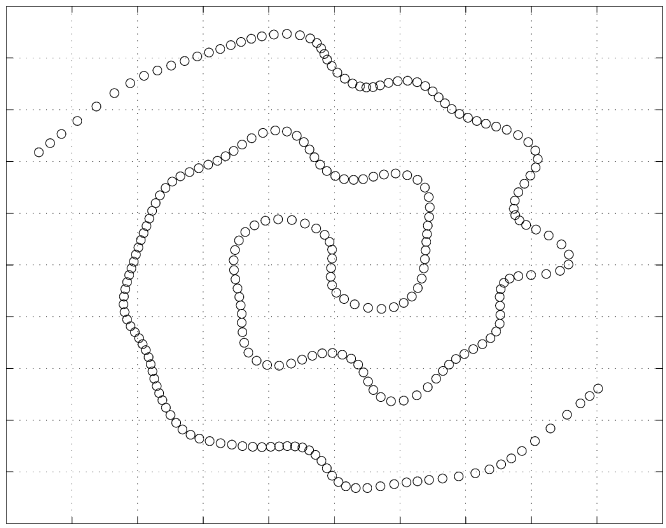

Fig. 3. Illustration of a two-dimensional nonlinear analog part implemented for compression.

In Fig. 3, we illustrate, schematically, the typical structure of the nonlinear compression. The circles mark code vectors in the two-dimensional input space, and α_m is defined by a nearest neighbor search among these code vectors to produce the corresponding 256-PAM symbol. How code vectors are mapped to the PAM alphabet can clearly be seen in the figure. The two endpoints of the PAM constellation are mapped to the two endpoints of the "spiral" in Fig. 3, and any two neighboring code vectors correspond to two neighboring PAM points. At the receiver, a nearest neighbor search over the PAM constellation produces the corresponding code vector (circle in the figure) to give a value for $((\hat{\boldsymbol{E}}_{2m-1}), (\hat{\boldsymbol{E}}_{2m}))$.

The approach we use for the nonlinear analog part, as described, is essentially equivalent to the "BDCE system" studied by Vaishampayan in his Ph.D. dissertation [17] (see, in particular, [17, Secs. 5.4–5.5]). A similar system (α, β) has also been investigated in [14]. We refer the reader to [17] for results on optimal encoder and decoder mappings, how to handle the power constraint, and a design algorithm.

[3]Although we only treat the case of BPSK signaling in the digital component of the system, we can clearly accommodate multilevel signaling schemes in general.

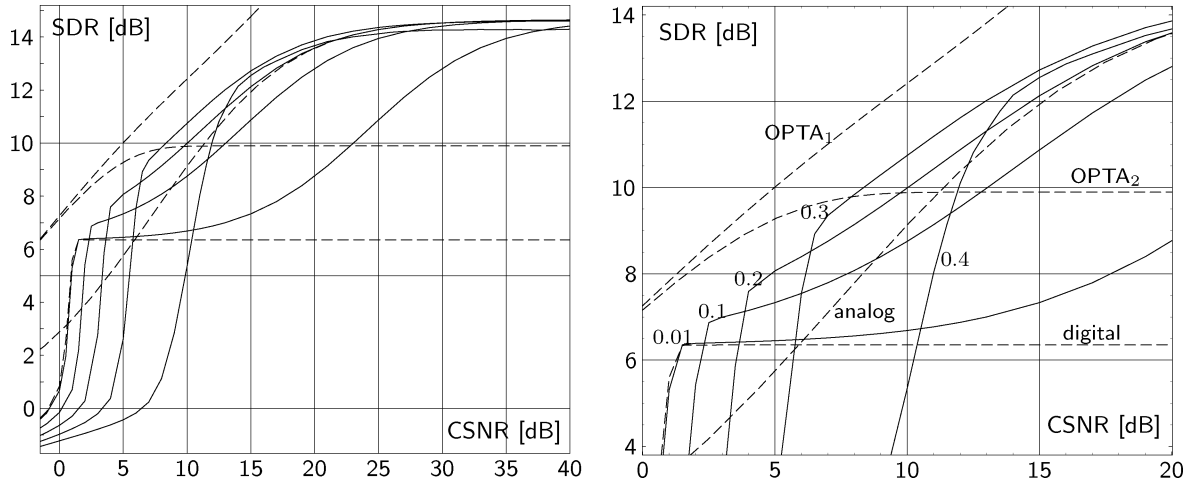

Fig. 4. Compression, linear analog part. *Solid lines* from the left at SDR $= 5$ dB: $\Delta = 0.01, 0.1, 0.2, 0.3, 0.4$. *Dashed lines* from above at CSNR $= 15$ dB: OPTA for analog input (OPTA$_1$), purely analog, OPTA for binary input (OPTA$_2$), and purely digital. (The right plot is an enlargement of parts of the left plot.).

C. Simulation Results: Bandwidth Compression

Here we evaluate the performance of the described HDA system when used for bandwidth compression. We investigate the system both with linear and nonlinear analog parts. The systems were trained for a fixed relative power level Δ in the analog part, and a fixed CSNR,[4] where CSNR $= 10\log_{10}(P_{\text{in}}/\sigma^2)$ (in decibels), with P_{in} denoting the total channel input power per component. In our simulations, motivated by a broadcast scenario, we allow the receiver to have knowledge of the true CSNR and thus to adapt to it as it varies, while the transmitter is kept fixed. We employed 500 000 vectors in the training of the SOVQ $(\varepsilon_1, \delta_1)$ and 100 000 vectors in the training of the nonlinear (α, β) maps. The simulations were run with $M = 128$ and using 1000 "superblocks" (128 000 source vectors). All considered systems have an overall bandwidth ratio of $\rho = 1/2$ channel uses per source symbol.

In Figs. 4 and 5, we show performance results in terms of the source signal-to-distortion ratio (SDR)

$$\text{SDR} = \frac{E\|\boldsymbol{X}\|^2}{E\|\boldsymbol{X} - \hat{\boldsymbol{X}}\|^2}$$

for the following systems.

- Five linear analog HDA schemes (Fig. 4), evaluated at an analog power level Δ of 1%, 10%, 20%, 30%, and 40%, respectively.
- Four nonlinear analog HDA schemes (Fig. 5) trained at the following values of the pair (CSNR, Δ): CSNR $= 20$ dB and $\Delta = 20\%$, CSNR $= 30$ dB and $\Delta = 20\%$, CSNR $= 40$ m dB and $\Delta = 30\%$, and CSNR $= 50$ dB and $\Delta = 30\%$. The performance was evaluated over a range of different CSNRs, and with $\Delta = 30\%$ in all cases.
- A purely analog system (Figs. 4 and 5) employing solely the analog part of the linear analog HDA system (with the digital part turned off). The SDR of this system is obtained as

$$\text{SDR} = \frac{s_1 + s_2}{d(\sigma^2) + s_2}, \qquad \text{with } d(\sigma^2) = \frac{p\sigma^2 s_1}{2s_1 + p\sigma^2}$$

  where $s_1 = \sum_{i=1}^{16} \lambda_i$ and $s_2 = \sum_{i=17}^{32} \lambda_i$, that is, the power in the strong half and the weak half of the dimensions, respectively.
- A purely digital tandem system (Fig. 4) employing solely the digital part of the HDA systems (with the analog part turned off).

[4]Note that, unlike the encoder of the nonlinear analog system, the encoder of the linear analog system does not need any knowledge about the CSNR value.

- The optimal performance theoretically attainable (OPTA) shown in Figs. 4 and 5, which is obtained by setting $R(D) = \rho C$, where $R(D)$ is the rate-distortion function in bits per source sample of the Gauss–Markov source (under the squared-error distortion measure), $\rho = 1/2$, and C is the channel capacity. The figure shows two OPTA curves, OPTA$_1$ corresponds to the analog-input AWGN channel, that is with

$$C = \frac{1}{2}\log_2(1 + \text{CSNR}) \quad \text{[bits/channel use]}.$$

  Also, as a reference when judging the purely digital system, the curve labeled OPTA$_2$ (Fig. 4 only) shows the OPTA for the binary-input Gaussian channel, that is with

$$C = -\frac{1}{2}\log_2 2\pi e\sigma^2 - \int_{-\infty}^{\infty} g(x)\log_2 g(x)dx \ [\text{ bits/channel use}]$$

  where

$$g(x) = \frac{1}{2\sqrt{2\pi\sigma^2}}\left(e^{-\frac{(x+1)^2}{2\sigma^2}} + e^{-\frac{(x-1)^2}{2\sigma^2}}\right).$$

We observe from the figures that the HDA systems offer a robust and graceful performance over the entire range of the CSNRs. We also remark that the performance of the HDA systems at low to medium CSNRs is strongly affected by the power allocation provided to the analog part, with the value of Δ playing a role similar to that of "rate allocator" between the digital and analog parts. The linear analog HDA systems outperform the purely analog systems for a wide range of CSNRs, depending on Δ. The systems with $\Delta = 30\%$ or 20% can be said to provide the best overall performance. The HDA systems also provide substantial improvements over the purely digital system at medium to high CSNRs. A drawback of using a linear analog part, however, is that the performance saturates at SDR ≈ 14 dB. This can be counteracted by using the nonlinear maps (α, β) in the analog part, as can be seen in Fig. 5. They perform very well (with a strictly positive SDR curve slope) in the vicinity of the CSNR at which their encoder was designed; they also provide a smooth degradation/improvement as the true CSNR varies away from the designed CSNR. Indeed, their SDR is within 5 dB of OPTA for a wide range of CSNRs (e.g., for 6 dB $\leq$ CSNR $\leq$ 45 dB in Fig. 5). Note also that the HDA system with nonlinear analog part can be made to saturate at an arbitrarily high SDR, by increasing the resolution of the maps (α, β).

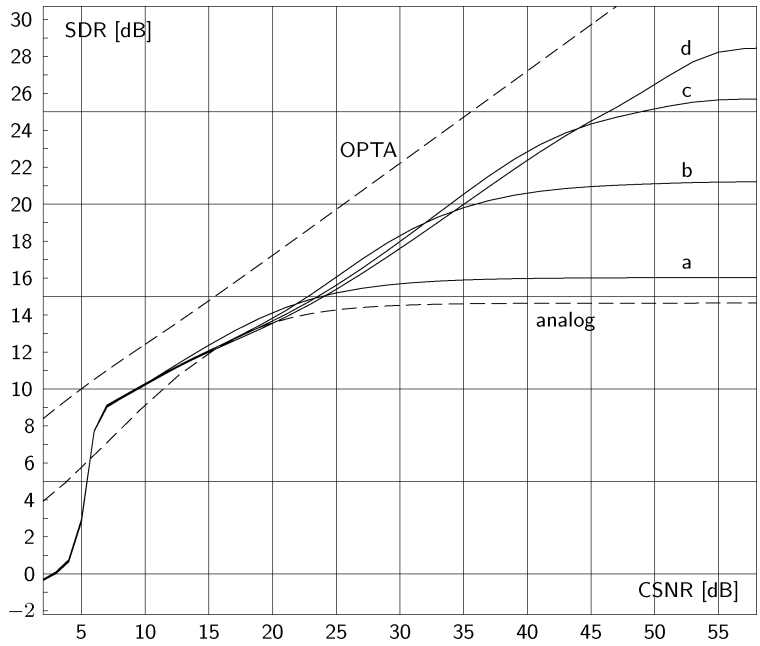

Fig. 5. Compression, nonlinear analog part. *Solid lines*: (α, β) trained at *a*: CSNR $= 20$ dB and $\Delta = 0.2$; *b*: CSNR $= 30$ dB and $\Delta = 0.2$; *c*: CSNR $= 40$ dB and $\Delta = 0.3$; *d*: CSNR $= 50$ dB and $\Delta = 0.3$. In *a–d*, the performance is evaluated over different CSNRs and with the power allocation set at $\Delta = 0.3$ in all cases. *Dashed lines* from above: OPTA (analog input) and purely analog.

IV. Bandwidth Expansion

Here we study the system in Figs. 1 and 2 when used for *bandwidth expansion*, that is, $\rho = q/(Mp) = r/p > 1$. The precise system we have implemented is specified as follows.

The source vector $\boldsymbol{X}$ is $p = 8$ dimensional. As in Section III, the vector $\boldsymbol{X}$ is drawn from a zero-mean Gauss–Markov source with normalized correlation 0.8. In this section, we do not use linear preprocessing, so $\boldsymbol{G}$ is the identity matrix. The low-delay encoder ε_1 is again an SOVQ encoder, this time of dimension $p = 8$ and size $L = 8$ bits, and the codebook $\{\boldsymbol{z}_i\}$ is identical to the codebook defining ε_1. The high-delay encoder ε_2 is the same $(k = 1024, n = 2048)$, rate $r_c = 1/2$, turbo encoder as used in Section III, and $M = 128$ blocking is again used. The output bits from ε_2 are mapped to ± 1 BPSK symbols, and 2048 bits are transmitted to represent $M = 128$ source vectors. This gives a bandwidth ratio $\rho = 2$ channel uses per source sample.

A. Linear Analog Part

In the case of bandwidth expansion with a linear analog part, α is the linear mapping corresponding to transmitting each component of $\boldsymbol{E}$ twice. More precisely

$$(\boldsymbol{Z})_m = (\boldsymbol{Z})_{p+m} = (\boldsymbol{E})_m, \ m = 1, \ldots, p.$$

The constant scaling in the analog part of the receiver is chosen as $b = 1$ for simplicity (since b anyhow can be absorbed into β), and the decoder β is defined as the linear mapping that computes the component-wise linear minimum MSE estimate of the vector $\boldsymbol{E}$ based on $\boldsymbol{R}$, again assuming the digital decoder δ_2 works without errors. That is (with $b = 1$)

$$(\hat{\boldsymbol{E}})_m = \frac{a s_m}{2a^2 s_m + \sigma^2} \left[(\hat{\boldsymbol{Z}})_m + (\hat{\boldsymbol{Z}})_{p+m} \right], \qquad m = 1, \ldots, p$$

where $s_m = E\left[\{(\boldsymbol{E})_m\}^2\right]$.

B. Nonlinear Analog Part

We again employ the discrete approximation described in Section III-B, the only essential difference being that α is split into eight parts α_m that each maps one input dimension into two channel dimensions. That is, the 8-dimensional vector $\boldsymbol{E}$ is transmitted via one use each of $\alpha_m, m = 1, \ldots, 8$, producing a 16-dimensional vector $\boldsymbol{Z}$. The components of $\boldsymbol{Z}$ are restricted to the same PAM alphabet as used in Section III. The decoder β maps $(\hat{\boldsymbol{Z}})_{2m-1}$ and $(\hat{\boldsymbol{Z}})_{2m}$ into $(\hat{\boldsymbol{E}})_m$ for $m = 1, \ldots, 8$, as before based on ML decisions and table lookup decoding. As in Section III, the pair (α, β) is trained subject to a power constraint on the channel input symbols.

C. Simulation Results: Bandwidth Expansion

Here we evaluate the performance of HDA bandwidth expansion, with linear and nonlinear analog parts. As in Section III-C, the systems were trained for a fixed relative power level Δ in the analog part. The receiver knows the true CSNR and can thus adapt to it, while the transmitter is kept fixed. As before, we employed 500 000 vectors in the training of the SOVQ $(\varepsilon_1, \delta_1)$ and 100 000 vectors in the training of the nonlinear (α, β) maps. The simulations were run with $M = 128$ and using 1000 "superblocks" (128 000 source vectors). All systems in the comparison have an overall bandwidth ratio of $\rho = 2$ channel uses per source dimension.

Figs. 6 and 7 illustrate the performance for the following systems.

- Four linear analog HDA schemes (Fig. 6), evaluated at an analog power level Δ of 1%, 10%, 20%, and 30%, respectively.
- Four nonlinear analog HDA schemes (Fig. 7) trained at $\Delta = 0.3$ (in all cases) and for the following CSNRs: 10, 15, 20, 25 dB. The performance was evaluated over a range of different CSNRs, and with $\Delta = 30\%$ in all cases.
- A purely analog system (Figs. 6 and 7) employing solely the analog part of the linear analog HDA system (with the digital part turned off).
- A purely digital tandem system (Fig. 6) employing solely the digital part of the HDA systems (with the analog part turned off).
- The "HDA-VQ" system presented in [6] (cf. [6, Fig. 4]).

The figures also show the OPTA curves for $\rho = 2$.

We remark from Figs. 6 and 7 that our bandwidth expansion systems perform analogously to the bandwidth compression systems studied in the previous section (cf. Figs. 4 and 5). Indeed, the gains vis-a-vis the purely analog and digital systems are maintained at medium to high

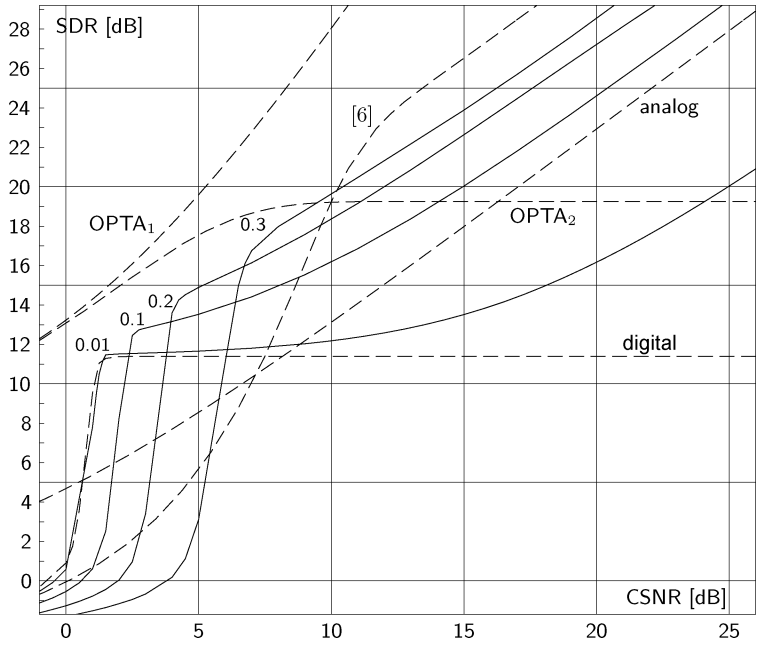

Fig. 6. Expansion, linear analog part. *Solid lines* from the left at SDR $= 5$ dB: $\Delta = 0.01, 0.1, 0.2, 0.3$. *Dashed lines* from above at CSNR $= 9$ dB: OPTA for analog input (OPTA_1), OPTA for binary input (OPTA_2), the "HDA-VQ" system in [6], purely analog and purely digital.

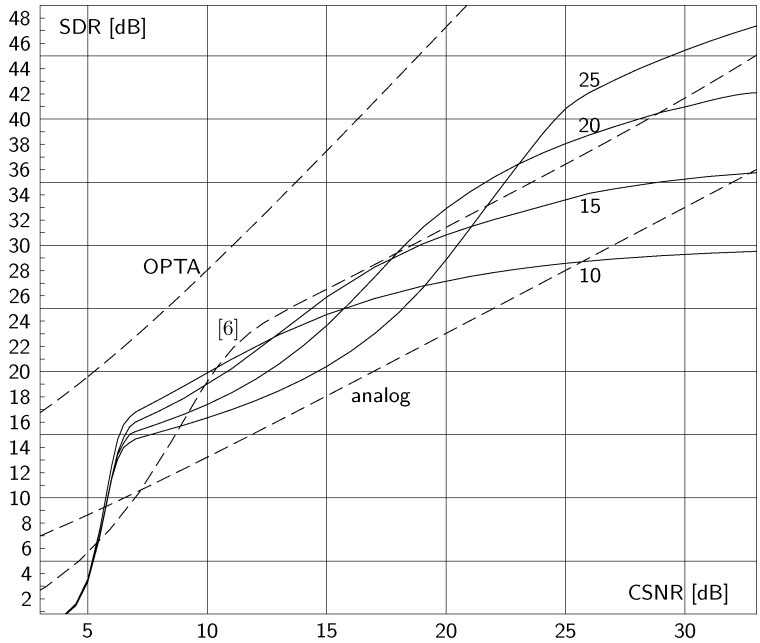

Fig. 7. Expansion, nonlinear analog part. *Solid lines*: (α, β) trained at $\Delta = 30\%$ and CSNR $= 10$, 15, 20, 25 dB, as marked. The performance is evaluated over different CSNRs and with the power allocation set at $\Delta = 0.3$. *Dashed lines* from above at CSNR $= 15$ dB: OPTA (analog input), the "HDA-VQ" system in [6], and purely analog.

CSNRs. Furthermore, the HDA system is improved at high CSNRs when the linear maps in its analog component are replaced by the nonlinear maps. For example, the HDA system with a linear analog part with $\Delta = 30\%$ has an SDR of 28 dB for CSNR $= 20$ dB (see Fig. 6), while the HDA system with a nonlinear analog part trained for CSNR $= 20$ dB provides an SDR ≈ 33 dB at the same CSNR (see Fig. 7), resulting in a substantial gain. This gain is however reduced if there is a mismatch between the true CSNR and the CSNR for which the nonlinear encoder of the analog part is designed; for example, when the true CSNR is 20 dB and the nonlinear encoder's design CSNR is 15 dB, the gain is 3 dB (it is 1 dB for a design CSNR of 25 dB). This indicates that one advantage of the linear analog part is that it does not need to know the CSNR at the encoder and thus it is not affected by a CSNR mismatch. The main difference between the bandwidth expansion and compression systems is that the SDR in our bandwidth expansion schemes, with a linear or infinite-resolution[5] nonlinear analog part, have no leveling-off effect—the slope of their SDR curve is positive for any CSNR. The slope, however, is noticeably less than that of the OPTA curve (slope $= 2$).

[5]For finite resolution in the proposed implementation of the nonlinear analog part, the SDR will level off asymptotically, however, in principle, the CSNR at which this happens can be pushed arbitrarily high by increasing the resolution.

With respect to the "HDA-VQ" system of [6], it is first worthy to point out that our system employs superposable coding (as the digital and analog signals are added to each other at the encoder output before transmission over the channel), while the system of [6] does not. In Fig. 6, we observe that our system with the linear analog part provides a better performance at low to medium CSNRs. This can be explained in virtue of the turbo channel coding employed in the digital part of our system, which helps combat channel error in the "waterfall" error region of the turbo code at low to medium CSNRs. On the other hand, the system of [6] does not employ strong channel coding and is hence prone to the significant channel impairment in that CSNR range. However, in the high-CSNR regime, the system of [6] is less susceptible to channel noise and its analog component becomes "cleaner" than our system's since it does not use superposable coding; i.e., unlike our system, it does not need to "filter" out the digital and analog signals from each other at the decoder. Still, as illustrated in Fig. 7, our system with the nonlinear analog part can match or outperform the system of [6] at high CSNRs that lie in the vicinity of the CSNR for which the nonlinear encoder map is designed; e.g., the nonlinear system designed for a CSNR of 25 dB outperforms the scheme of [6] for CSNRs in an interval starting at 23 dB (for finite resolution in the nonlinear analog part, the curve from [6] will cross the new curve at a CSNR $\approx$ 35 dB, however, by increasing the resolution, the range over which the new system outperforms the one in [6] can be improved). Finally, it is important to note that the new system is more general than that of [6] as it allows for both expansion and compression modes. In fact, it subsumes the scheme in [6]; e.g., for $\rho = 2$, the new system can be converted to the one in [6] if we replace the high-delay channel encoding map ε_2 by a simple $r_c = 1/2$ map resulting in 16 BPSK symbols where the 8 bits of index I appear in the first eight positions and zeros are stacked in the last eight positions (the decoder δ_2 performs the reverse operation), and if we choose the analog map α to produce a vector $\boldsymbol{Z} \in \mathbb{R}^{16}$ such that the first eight components of $\boldsymbol{Z}$ are zeros and $\boldsymbol{E}$ appears within the last eight components.

V. Summary and Conclusion

An HDA source–channel coding system for the reliable communication and reproduction of discrete-time analog-valued sources over AWGN channels is proposed and investigated. The HDA system, which is based on VQ source coding, employs turbo channel coding in its digital component and linear/nonlinear coding in its analog component, before superposing the analog and digital signals for transmission over the channel. As a result, the system accommodates all bandwidth ratios and, unlike the scheme studied in [6], it can operate in both bandwidth compression and expansion modes. Numerical results show that the HDA system provides a robust and graceful performance for a wide range of channel conditions (medium to high CSNRs), substantially outmatching purely digital and analog coding systems. Under bandwidth compression, the system performs within 5 dB (in SDR) of the OPTA limit for a large CSNR range. The advantages of using linear and nonlinear coding in the analog part of the system are also illustrated: linear coding is simple and does not need the knowledge of the CSNR at the encoder, while nonlinear coding can significantly improve the system performance at high CSNRs.

Future work may include improving the system performance at low CSNRs. An interesting direction is to optimize the performance of the digital component of the system using joint source–channel coding techniques without affecting its performance at high CSNRs. This can be accomplished by leaving the VQ encoder unoptimized and designing a joint source–channel decoder for the VQ–turbo decoder pair according to the methods of [4], [5], [21]. A first step in this direction is undertaken in [22], in the context of image communication without the use of turbo coding, and the digital encoder and decoder are optimized under bandwidth compression. Finally, since the HDA system is general, it can be applied for a variety of source and channel models, including fading channels used in conjunction with multilevel modulation.

References

[1] M. Skoglund, N. Phamdo, and F. Alajaji, "Hybrid digital–analog coding for bandwidth compression/expansion using VQ and turbo codes," in *Proc. IEEE Int. Symp. Information Theory*, Washington, DC, Jun. 2001, p. 260.

[2] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, vol. 37, no. 1, pp. 10–21, Jan. 1949.

[3] H. Coward, "Joint source–channel coding: Development of methods and utilization in image communications," Ph.D. dissertation, Norwegian Univ. Sci. Technol., Trondheim, Norway, 2001.

[4] N. Phamdo and F. Alajaji, "Soft-decision demodulation design for COVQ over white, colored and ISI Gaussian channels," *IEEE Trans. Commun.*, vol. 48, no. 9, pp. 1499–1506, Sep. 2000.

[5] M. Skoglund and P. Hedelin, "Hadamard-based soft decoding for vector quantization over noisy channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 515–532, Mar. 1999.

[6] M. Skoglund, N. Phamdo, and F. Alajaji, "Design and performance of VQ-based hybrid digital–analog joint source–channel codes," *IEEE Trans. Inf. Theory*, vol. 48, no. 3, pp. 708–720, Mar. 2002.

[7] U. Mittal and N. Phamdo, "Hybrid digital-analog (HDA) joint source–channel codes for broadcasting and robust communications," *IEEE Trans. Inf. Theory*, vol. 48, no. 3, pp. 1082–1102, May 2002.

[8] J. M. Lervik, A. Grovlen, and T. Ramstad, "Robust digital signal compression and modulation exploiting the advantages of analog communications," in *Proc. IEEE Global Telecommunications Conf.*, Singapore, Nov. 1995, pp. 1044–1048.

[9] H. Coward and T. A. Ramstad, "Quantizer optimization in hybrid digital–analog transmission of analog source signals," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Istanbul, Turkey, Jun. 2000, pp. 2637–2640.

[10] U. Mittal and N. Phamdo, "A joint source–channel speech coder using hybrid digital–analog (HDA) modulation," *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 4, pp. 222–231, May 2002.

[11] S. Shamai (Shitz), S. Verdú, and R. Zamir, "Systematic lossy source/channel coding," *IEEE Trans. Inf. Theory*, vol. 44, no. 2, pp. 564–579, Mar. 1998.

[12] M. Skoglund, N. Phamdo, and F. Alajaji, "VQ-based hybrid digital–analog joint source–channel coding," in *Proc. IEEE Int. Symp. Information Theory*, Sorrento, Italy, Jun. 2000, p. 403.

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

[14] A. Fuldseth and T. A. Ramstad, "Bandwidth compression for continuous amplitude channels based on vector approximation to a continuous subset of the source signal space," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Munich, Germany, Apr. 1997, pp. 3093–3096.

[15] I. Kozintsev and K. Ramchandran, "Hybrid compressed–uncompressed framework for wireless image transmission," in *Proc. IEEE Int. Conf. Communications*, Montréal, QC, Canada, Jun. 1997, pp. 77–80.

[16] T. A. Ramstad, "Combined source coding and modulation for mobile multimedia communication," in *Insights into Mobile Multimedia Communications*, D. Bull, N. Canagarajah, and A. Nix, Eds. San Diego, CA: Academic , 1999, ch. 26, pp. 415–430.

[17] V. Vaishampayan, "Combined source–channel coding for bandlimited waveform channels," Ph.D. dissertation, Univ. Maryland, College Park, MD, 1989.

[18] V. A. Vaishampayan and S. I. R. Costa, "Curves on a sphere, shift-map dynamics, and error control for continuous alphabet sources," *IEEE Trans. Inf. Theory*, vol. 49, no. 7, pp. 1658–1672, Jul. 2003.

[19] S. Sesia, G. Caire, and G. Vivier, "Lossy transmission over slow-fading AWGN channels: A comparison of progressive, superposition and hybrid approaches," in *Proc. IEEE Int. Symp. Information Theory*, Adelaide, Australia, Sep. 2005, pp. 224–228.

[20] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Dordrecht, The Netherlands: Kluwer Academic, 1992.

[21] G.-C. Zhu and F. Alajaji, "Soft-decision COVQ for turbo-coded AWGN and Rayleigh fading channels," *IEEE Commun. Lett.*, vol. 5, no. 6, pp. 257–259, Jun. 2001.

[22] Y. Wang, F. Alajaji, and T. Linder, "Design of VQ-based hybrid digital–analog joint source–channel codes for image communication," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2005, pp. 193–202.