Multistage Trellis Quantization and its Applications *

Stefania Sesia
Motorola Labs
Gif-sur-Yvette, France
stefania.sesia@crm.mot.com

Giuseppe Caire
Eurecom Institute
Sophia-Antipolis, France
giuseppe.caire@eurecom.fr

Abstract

We consider the practical construction of successive refinement source codes for real sources and quadratic distortion. We propose a scheme based on convolutional codes and unitary transformations, efficiently implemented by FFT/IFFT and interleaving. This scheme offers performances comparable to the best known TCQ schemes and very fine granularity of rates. Then, we apply our scheme to adaptive source transmission over noisy channels with fixed spectral efficiency and to the broadcast approach for the BF-AWGN and broadcast Gaussian channel, extending the approach of Steiner and Shamai to the case of lossy transmission.

1 Background

Consider a source $S \in \mathbb{R}$ with rate-distortion function $R(D)$ with respect to a certain distortion measure $d : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$, inducing the distortion measure on $\mathbb{R}^k \times \mathbb{R}^k$ according to

$$d(\mathbf{s}, \widehat{\mathbf{s}}) = \frac{1}{k} \sum_{i=1}^{k} d(s_i, \widehat{s}_i) \tag{1}$$

An L-level successive refinement source code of block length k is defined by the encoding functions $g_\ell : \mathbb{R}^k \to \{1, \dots, M_\ell\}$ and by the reconstruction functions

$$\phi_\ell : \{1, \dots, M_1\} \times \cdots \times \{1, \dots, M_\ell\} \to \mathbb{R}^k$$

The rate L-tuple of the successive refinement code is given by $\{R_\ell = \sum_{j=1}^{\ell} \log_2 M_j : \ell = 1, \dots, L\}$ and the achieved distortion L-tuple is given by

$$\{D_\ell = \mathbb{E}\left[d(\mathbf{s}, \phi_\ell(g_1(\mathbf{s}), \dots, g_\ell(\mathbf{s})))\right] \;:\; \ell = 1, \dots, L\}$$

The successive refinement structure of the code manifests itself in the fact that distortion level D_ℓ is obtained by *refining* the coarser description at level $\ell - 1$ by incorporating additional information at rate increment $R_\ell - R_{\ell-1}$ bits/source symbol.

The source S is said *successively refinable* [1, 2] if, for any desired integer L, distortion L-tuple $D_1 < D_2 < \cdots < D_L$, $\epsilon > 0$ and sufficiently large k, there exists an L-level successive refinement source code of block length k with rate L-tuple $(R_1, \dots, R_L)$ such that

$$R_\ell \leq R(D_\ell) + \epsilon \quad \forall \;\; \ell = 1, \dots, L \tag{2}$$

and

$$\mathbb{E}[d(\mathbf{s}, \phi_\ell(g_1(\mathbf{s}), \dots, g_\ell(\mathbf{s})))] \leq D_\ell + \epsilon \quad \forall \;\; \ell = 1, \dots, L \tag{3}$$

*This work was partly supported by a CIFRE project, Motorola Labs and by Institut Eurecom's industrial partners: Bouygeues Telecom, Fondation d'enterprise Groupe Cegetel, Fondation Hasler, France Telecom, Hitachi, STMicroelectronics, Swisscom, Texas Instruments and Thales.

In other words, the L-tuple of *optimal* rate-distortion pairs $\{(R(D_\ell), D_\ell) : \ell = 1, \ldots, L\}$ is achievable by successive refinement.

In the rest of this work we restrict to the quadratic distortion measure $d(s, \widehat{s}) = |s - \widehat{s}|^2$ and to sources with mean zero and variance 1 (different variances can be handled by normalization).

It is well-known that a Gaussian i.i.d. source $S \sim \mathcal{N}(0, 1)$ is successively refinable [1, 2]. It is also well-known that, in the Gaussian case, optimal successive refinement codes have an additive structure [3], i.e., the ℓ-th level representation vector $\widehat{\mathbf{s}}_\ell$ for the source vector $\mathbf{s}$ is given by

$$\widehat{\mathbf{s}}_\ell = \sum_{j=1}^{\ell} \psi_j(g_j(\mathbf{s})) \tag{4}$$

where $\psi_\ell : \{1, \ldots, 2^{k(R_\ell - R_{\ell-1})}\} \to \mathbb{R}^k$ denotes the reconstruction *increment* function at level ℓ.[1]

Now, consider a spherical codebook

$$\mathcal{C} = \left\{ \mathbf{c}_q \in \mathbb{R}^k \quad : \quad q = 1, \ldots, 2^{k r_s} \right\} \tag{5}$$

where r_s is a design parameter. The codewords of $\mathcal{C}$ lie on a k dimensional sphere of squared radius k. Consider $\Delta \in (0, 1]$ and let $Q_\alpha : \mathbb{R}^k \to \{1, \ldots, 2^{k r_s}\}$ denote the minimum Euclidean distance decoder for the scaled code $\alpha\mathcal{C}$, i.e.,

$$Q_\alpha(\mathbf{s}) = \arg \min_q d(\mathbf{s}, \alpha \mathbf{c}_q) \tag{6}$$

Lapidoth [4] showed that for $r_s > \frac{1}{2} \log_2 \frac{1}{\Delta}$, $\alpha = \sqrt{1 - \Delta}$, $\epsilon > 0$ and for sufficiently large k there exist spherical codes $\mathcal{C}$ such that

$$\mathbb{E}[d(\mathbf{s}, \alpha \mathbf{c}_{Q_\alpha(\mathbf{s})})] \leq \Delta + \epsilon \tag{7}$$

This result holds for any source S, not necessarily Gaussian, i.i.d., or even ergodic, under the condition that $\frac{1}{k}|\mathbf{s}|^2 \to 1$ in probability [4]. In some sense, scaled spherical codes with minimum distance encoding are *robust* in the sense that they achieve the Gaussian rate distortion bound under very mild conditions on the source. On the other hand, for these codes all sources appear as hard to compress as the Gaussian i.i.d. source.

We shall construct L-levels successive refinement codes from a single spherical code $\mathcal{C}$, denoted as the "basic code". Fig. 1 provides a pictorial representation of the geometry of the proposed construction.

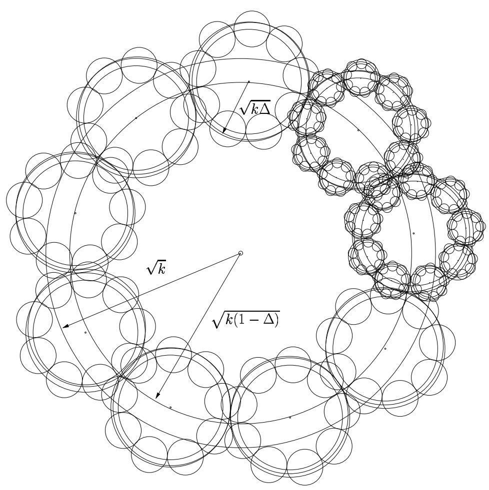


Figure 1: Geometry of a successive refinement source code based on spherical code.

[1] We define $R_0 = 0$ and $D_0 = 1$.

The encoding function at level ℓ is based on the minimum distance decoder of the basic code. It computes the ℓ-level index

$$g_\ell(\mathbf{s}) = Q_{\alpha\sqrt{\Delta^{\ell-1}}}(\mathbf{s} - \widehat{\mathbf{s}}_{\ell-1}) \tag{8}$$

where

$$\widehat{\mathbf{s}}_\ell = \sum_{j=1}^{\ell} \alpha\sqrt{\Delta^{j-1}}\mathbf{c}_{g_j(\mathbf{s})} \tag{9}$$

is the representation vector at level ℓ. Such multistage structure can achieve the rate L-tuple $\{R_\ell = \ell r_s : \ell = 1, \ldots, L\}$ with distortion L-tuple $\{D_\ell = \Delta^\ell : \ell = 1, \ldots, L\}$.

Lastras and Berger [5] showed that any well-behaved source can be encoded by successive refinement incurring a bounded rate penalty at each level. In particular, let S be an arbitrary i.i.d. source with mean zero, variance 1, finite differential entropy $h(S)$ and rate-distortion function $R(D)$. The distortion L-tuple $(D_1, \ldots, D_L)$ can be achieved by successive refinement at rates $(R_1, \ldots, R_L)$ such that

$$R_\ell \leq R(D_\ell) + \frac{1}{2}\log_2\frac{1}{\mathcal{P}_S} \tag{10}$$

where $\mathcal{P}_S = \frac{2^{2h(S)}}{2\pi e}$ is the *entropy power* of S, i.e., it is the variance of a Gaussian source with the same differential entropy of S.

The multistage spherical code can achieve $D_\ell = \Delta^\ell$ at rate $R_\ell = \frac{\ell}{2}\log_2\frac{1}{\Delta}$. By using the Shannon lower bound on the rate distortion function [6], we find that the rate penalty is bounded by

$$R_\ell - R(D_\ell) = \frac{\ell}{2}\log_2\frac{1}{\Delta} - R(\Delta^\ell) \leq \frac{\ell}{2}\log_2\frac{1}{\Delta} - \frac{1}{2}\log\frac{\mathcal{P}_S}{\Delta^\ell} = \frac{1}{2}\log_2\frac{1}{\mathcal{P}_S} \tag{11}$$

which coincides with the bound in (10). In other words, the behavior of the proposed scheme meets Lastras and Berger bound for any source for which Lapidoth result [4] holds. In practice, a successive refinement code that *approaches* the Gaussian rate-distortion bound for any target distortion L-tuple and any well-behaved source is highly desirable. This is pretty much all what we can hope for in practical applications, when the statistics of the source is not known a priori and might not be ergodic. A typical example is provided by image coding, where the statistics of the output of the "analog" part of the encoder, essentially given by a linear transformation followed by segmentation and decimation, gives origin to blocks of signal $\mathbf{s}$ to be quantized, that are nearly uncorrelated and whose statistics may change from image to image and it is usually estimated adaptively, [7].

2 Code design

Suppose that we are given a "capacity achieving" spherical code $\mathcal{C}$, for the real AWGN channel with SNR τ [2]. Then, we choose $r_s = \frac{1}{2}\log_2(1+\tau)$, $\Delta = 1/(1+\tau)$ and $\alpha = \sqrt{\tau/(1+\tau)}$. We can write the source vector as

$$\mathbf{s} = \sum_{\ell=1}^{L} \alpha\sqrt{\Delta^{\ell-1}}\mathbf{c}_{g_\ell(\mathbf{s})} + \mathbf{e}_L \tag{12}$$

where $\mathbf{e}_L$ is the representation error vector at level L. By interpreting (12) as the output of a multiple-access channel with background noise $\mathbf{e}_L$, we notice that the levels are successively decodable by stripping in the order $1, \ldots, L$. In fact, the interference plus noise ratio (SINR) seen by stage ℓ of the stripping decoder is given by

$$\frac{\alpha^2\Delta^{\ell-1}}{\Delta^L + \alpha^2\sum_{j=\ell+1}^{L}\Delta^{j-1}} = \tau \tag{13}$$

[2]This notion is meaningless since a single code cannot *achieve capacity*. However, what we mean here is that $\mathcal{C}$ is a member of a sequence of codes that work arbitrarily close to the capacity limit for increasing block length.

Unfortunately, spherical codes that work very close to the AWGN capacity and admit minimum distance decoders with practical complexity (say, polynomial in the block length) have not been found so far. If they were available, both the problems of channel coding and of source coding would have been already solved. Hence, driven by complexity considerations, we propose to use as basic code a trellis-terminated binary convolutional code with binary antipodal modulation (i.e., mapping the alphabet $\{0, 1\}$ onto $\{+1, -1\}$). In this case, the minimum distance decoder $Q_\alpha(\cdot)$ is efficiently implemented by the Viterbi algorithm.

Since trellis-terminated convolutional codes with fixed (not increasing with the block length) trellis complexity do not approach the AWGN capacity, the choice of α and Δ according to a threshold SNR τ outlined at the beginning of this section is not optimal any longer. On the contrary, for a given basic code we find the optimal scaling factor α and the resulting optimal distortion Δ numerically. Let $\mathbf{s}$ be Gaussian i.i.d. $\sim \mathcal{N}(0, 1)$. By Monte Carlo simulation, we find

$$\alpha = \arg \min_{\beta \geq 0} \mathbb{E}\left[d(\mathbf{s}, \beta \mathbf{c}_{Q_\beta(\mathbf{s})})\right] \tag{14}$$

and the resulting distortion is given by $\Delta = \mathbb{E}\left[d(\mathbf{s}, \alpha \mathbf{c}_{Q_\alpha(\mathbf{s})})\right]$. Fig.2 shows $\mathbb{E}\left[d(\mathbf{s}, \beta \mathbf{c}_{Q_\beta(\mathbf{s})})\right]$ versus β, where the optimal pair (α, Δ) is clearly evidenced.

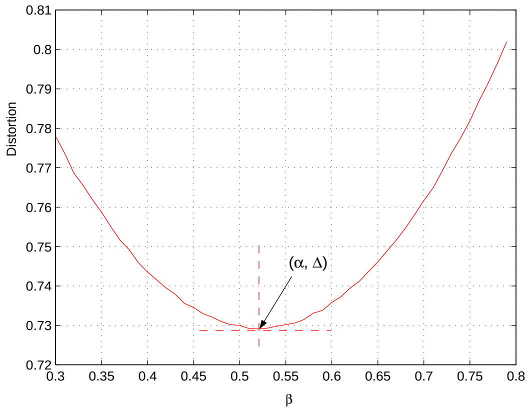


Figure 2: Distortion vs β for $r_s = 1/4$ and 128 states for Gaussian sources.

It is interesting to observe that for the optimal value of α and Δ, the convolutional code works "above capacity". More precisely, suppose that we can write the source $\mathbf{s}$ as

$$\mathbf{s} = \alpha \mathbf{c}_{Q_\alpha(\mathbf{s})} + \mathbf{e}_1 \tag{15}$$

where $\mathbf{e}_1$ is the representation error signal, such that $\mathbb{E}[\frac{1}{k}|\mathbf{e}_1|^2] = \Delta$ (by definition). If we interpret (15) as a binary-input AWGN channel, its SNR is given by α^2/Δ. The corresponding capacity, $C_{\text{biawgn}}(\alpha^2/\Delta)$, is found to be less than the rate r_s of the basic code $\mathcal{C}$. For example, for the code of rate $r_s = 1/4$ and 128 states of Fig.2 we find $\alpha = 0.52$ and $\Delta = 0.729$, yielding capacity $C_{\text{biawgn}}(\alpha^2/\Delta) = 0.2268$, which is less than $1/4$.

Another countermeasure we take to partially compensate for the gap of binary convolutional codes from the AWGN capacity consists of introducing unitary transformations at each level such that the signals input to the Viterbi decoders look like Gaussian. In particular, let $\mathbf{U}_\ell$ denote a unitary transformation of $\mathbb{R}^k$. Each Viterbi decoder at level ℓ computes

$$g_\ell(\mathbf{s}) = Q_{\alpha\sqrt{\Delta^{\ell-1}}}(\mathbf{U}_\ell(\mathbf{s} - \widehat{\mathbf{s}}_{\ell-1})) \tag{16}$$

Then, the representation vector at level ℓ is given by

$$\widehat{\mathbf{s}}_\ell = \widehat{\mathbf{s}}_{\ell-1} + \alpha\sqrt{\Delta^{\ell-1}}\mathbf{U}_\ell^{-1}\mathbf{c}_{g_\ell(\mathbf{s})} \tag{17}$$

Ideally, we should select the unitary transformations independently at each level, according to the Haar measure, i.e., uniformly distributed on the manifold of unitary $k \times k$ matrices. This approach requires common randomness between encoder and decoder, and might be seen as a spherical version of the *dithering* approach commonly used in lattice quantizers [8]. In fact, since lattices are additive groups, randomization with lattice quantizers is obtained by *translating* the source vector by a dither vector $\mathbf{u}$ uniformly distributed over the lattice Voronoi cell. In our case, since spherical codes obtained from binary convolutional codes are *multiplicative* groups, we obtain randomization by *rotating* the source vector by a unitary matrix $\mathbf{U}$ uniformly distributed over the unit sphere (Haar measure), and hence also over the code Voronoi cell because of the geometric uniformity property.

Notice that both translations and rotations are *isometries* of $\mathbb{R}^k$, therefore, they preserve Euclidean distance (distortion). This means that the only effect of randomization via the unitary transformation is to present to each level Viterbi decoder a signal whose statistics is more *adapted* to the basic code.

We notice also that this approach might be extended to other families of spherical geometrically uniform codes, such as linear trellis codes over $\mathbb{Z}_M$ mapped to the M-PSK constellation [9].

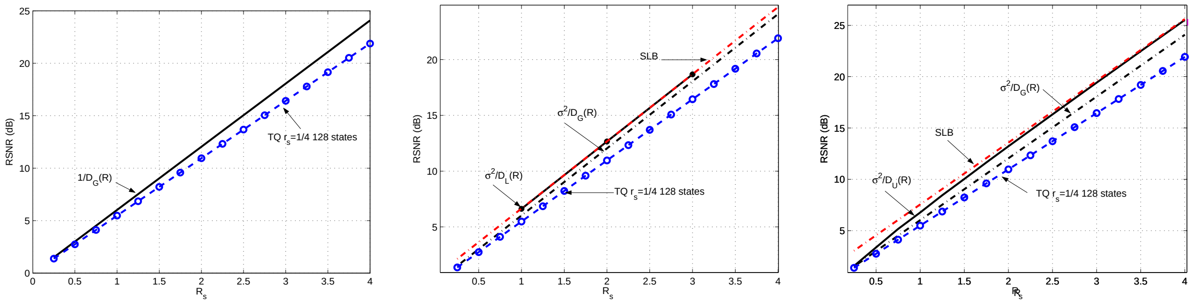
In practice, sampling elements from the Haar measure is quite computationally intensive for large dimension k. Moreover, matrix-vector multiplications have complexity $O(k^2)$ and matrix inverse $O(k^3)$. Also, precomputing and storing $k \times k$ real matrices with no special structure is highly impractical for large k. Hence, for the sake of complexity and practical implementation, we propose the use of structured unitary transformations given by

$$\mathbf{U}_\ell = \mathbf{\Pi}_\ell \begin{bmatrix} \mathbf{C} & -\mathbf{S} \\ \mathbf{S} & \mathbf{C} \end{bmatrix} \tag{18}$$

where $\mathbf{\Pi}_\ell$ is a random permutation of size k (interleaving), $\mathbf{C} + j\mathbf{S} = \sqrt{\frac{2}{k}}\mathbf{F}$ and where $\mathbf{F}$ is the Fourier matrix of dimension $k/2$, with (n,m) elements $e^{-j\frac{4\pi}{k}mn}$, for $m,n \in \{0,\ldots,k/2-1\}$. In this way, the product $\mathbf{U}_\ell\mathbf{x}$ can be efficiently computed by FFT and interleaving.

In standard trellis coded quantization (TCQ) [10], a trellis code defined over a multilevel alphabet is used. The resulting code is similar to Ungerboeck TCM [11]. It turns out that the probability with which the points in the code alphabet are selected is not uniform. Hence, rate improvement can be obtained by binary labeling the points with variable-length labels. A modified Max-Lloyd algorithm that exploits Viterbi decoding and training vectors is used in order to optimize the code alphabet and the binary representation of the points. This approach is generally known as Entropy-Constrained Trellis Coded Quantization (ECTCQ). The best known trellis quantizers for standard i.i.d. sources such as Gaussian, uniform and Laplacian, are found in the family of ECTCQ [12]. It is natural to ask if some rate improvement can be achieved in our scheme by applying entropy coding on the quantization indices $g_\ell(\mathbf{s})$. Notice that $g_\ell(\mathbf{s})$ is the sequence of information bits (input to the convolutional encoder) that corresponds to the codeword found by the Viterbi algorithm in (16). We run some experiments by applying the BWT-MDL source modeler of [13]. This modeler identifies the tree source model that best explains the binary sequence $g_\ell(\mathbf{s})$ by using the Burrows-Wheeler transform and the Minimum Description Length principle, i.e., the tree source model for which the overall description length (including coding and model redundancy) of $g_\ell(\mathbf{s})$ is minimized. We simulated 2000 independent source sequence of length $k = 1000$ and we computed the empirical entropy of $g_\ell(\mathbf{s})$ according to the BWT-MDL model. For all simulated frames this was always equal to 1 bit per symbol. This shows that the output of our multistage quantizer is close to an i.i.d. sequence of fair bits and that, in practice, post-processing entropy coding cannot improve performance.

Fig. 3 shows the performances of the multistage trellis quantizer for Gaussian, Laplacian and uniform sources, in terms of RSNR defined as $-10\log_{10}\Delta^\ell$ vs $R_s = \ell r_s$, with $r_s = 1/4$ and 128 states. The performances are compared with the optimal RSNR corresponding to the distortion-rate function. For Laplacian and uniform sources we plot also the Shannon lower bound (SLB) [6] and the RSNR obtained with the Gaussian distortion-rate function. Out scheme achieves the same performance for all three sources, in agreement with Lapidoth result [4]. Unfortunately, due to the suboptimality of convolutional codes with respect to ideal spherical codes, it suffers some gap from the Gaussian distortion-rate function. Nevertheless, these performance are comparable with the best known TCQ (not successive refinement) and are only slightly outperformed by the best known ECTCQ (also not successive refinement).



(a) (b) (c)

Figure 3: RSNR vs R of Multistage TQ scheme for rate $1/4$ and 128 states for (a) Gaussian, (b) Laplacian, and (c) uniform source.

3 Lossy adaptive transmission over noisy channels

We consider the transmission of a source S over a channel $P_{Y|X}$. The decoder must provide a reproduction of the source such that end-to-end distortion is minimized. Roughly speaking, practical source encoders involve some linear transformation (e.g., wavelet transform, subband decomposition, predictive filtering, etc ...) followed by some segmentation and decimation (e.g., deleting certain high-frequency components of the signal). Eventually, the analog data is quantized and the sequence of quantization indices is lossless compressed by the so-called "entropy coding". We identify the weak point of the conventional approach in the fact that the source encoding inverse function may have a catastrophic behavior: a little Hamming distortion in its input may cause large distortion in its output. This imply an overly strict constraint in the BER at the output of the channel decoder.

The non-catastrophic behavior of convolutional *encoders* has been widely studied. Convolutional codes admit non-catastrophic encoders such that small Hamming distance between encoder input sequences cause small distance in encoded sequences. In particular, this is the case of feedback-free non-catastrophic convolutional encoders. Our multistage source encoder inherits this property from its basic code $\mathcal{C}$. We consider the concatenation of the multistage source code with a standard channel code and compare the achieved end-to-end distortion with the separation limit. Let η denote spectral efficiency measured by the number of source symbols per channel use (equivalently, by the ratio of the (discrete-time) source bandwidth over the (discrete-time) channel bandwidth). Let $R(D)$ denote the source rate-distortion function and $C(\Gamma)$ denote the channel capacity-cost function. Hence, spectral efficiency η can be achieved with distortion D and input cost Γ if and only if $\eta \leq \frac{C(\Gamma)}{R(D)}$. For fixed spectral efficiency, the best achievable distortion as a function of the channel input cost is given by $D_{\rm opt} = R^{-1}(C(\Gamma)/\eta)$.

In our example, for simplicity, we fix the channel to be a binary-input AWGN (BI-AWGN) channel, defined by

$$y = \sqrt{E_s}x + z \tag{19}$$

where $x \in \{-1, +1\}$ and $z \sim \mathcal{N}(0, N_0/2)$, and the source to be Gaussian i.i.d. with quadratic distortion. Notice that in this case the conditions of [14] do not hold, hence we *have* to code the source and the channel in smart ways. The multistage source encoder produces the indices $(g_1(\mathbf{s}), \ldots, g_L(\mathbf{s}))$ in the form of binary sequences. Namely, $g_\ell(\mathbf{s})$ is the sequence of information bits corresponding to the codeword $\mathbf{c}_{g_\ell(\mathbf{s})}$ selected by the Viterbi decoder at level ℓ. As channel codes we may consider any family of good binary codes for the BI-AWGN channel. In particular, in our example we considered convolutional codes with 64 states and rates $1/4$, $1/3$, $1/2$, $2/3$, $3/4$, $5/6$, and the turbo code with component generators (37,21) (octal notation) taken from [15] with interleaving size 65536 and puncturing in order to have rates $1/3$, $1/2$, $2/3$, $3/4$, $5/6$, $11/12$. The source code is based on the convolutional code of rate 1/4 and 128 states already used in Fig. 2.

Fig. 4 shows the resulting distortion for $\eta = 1/3$ versus the channel SNR, defined as E_s/N_0. The separation limit is shown for comparison. Remarkably, the performance of the turbo-coded system is quite close to the theoretical optimum. Degradation comes from two effects: a horizontal displacement due to the SNR gap of the punctured turbo codes with respect to their capacity limit, and a vertical displacement due to the gap of the multilevel source code with respect to its rate-distortion limit. In

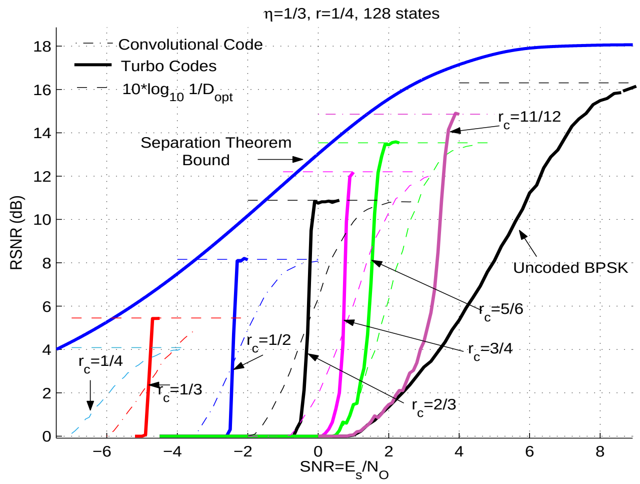


Figure 4: RSNR vs SNR for $r_s = 1/4$ with 128 states. The channel codes are 64 states convolutional codes and $(37, 21)$ turbo codes [15], punctured to obtain different rates. The bound based on separation theorem and the performances of uncoded BPSK transmission are also plotted.

practice, coupling our multilevel source code with channel coding of different rates can easily implement a variable-quality scheme that operates at fixed target spectral efficiency and adapts itself to the user SNR condition.

As far as the reconstruction is concerned, several recent works focused on soft reconstruction, where the channel decoder provides soft-output symbol-by-symbol information and this is used by the source decoder to mitigate the effect of residual post-decoding channel errors. We have checked that soft reconstruction, accomplished using the BCJR algorithm [16] to re-encode the convolutional codewords, processing the soft reliability values produced by the channel soft-in soft-out decoder, yields practically no improvement with respect to simple hard reconstruction.

4 Broadcast approach to the BF-AWGN channel

In this section we consider a (complex baseband equivalent) block-fading AWGN channel (BF-AWGN) described by

$$y_t = hx_t + z_t, \quad t = 1, \ldots, n \tag{20}$$

where h is a fading coefficient, random but constant for the whole duration of transmission (n channel uses), and $z_t \sim \mathcal{CN}(0, N_0)$ is complex circularly-symmetric AWGN. We let $a = |h|^2$ denote the fading power gain, with *continuous* pdf $f_A(z)$ and cdf $F_A(z)$. The input constraint is given by $\mathbb{E}[|x|^2] \leq E_s$, and we define $\Gamma = E_s/N_0$ as before.

This channel may also model a Gaussian broadcast channel [6] with $K \to \infty$ users, such that the empirical distribution of the users SNRs converges almost everywhere to the cdf $F_A(z)$. The BF-AWGN channel becomes completely equivalent to the Gaussian broadcast channel under the assumption, made here, that the transmitter is not informed about the value of the fading gain a (it does now, though, its statistics). A *broadcast approach* to the BF-AWGN channel was proposed and analyzed in [17] (and references therein). It consists of splitting the information message into L parallel streams and mapping each stream onto a layer of a superposition coding scheme. Each layer is modulated with a power level $\gamma(a)$ such that if the fading gain is $A = a$ then the layers up to a can be reliably decoded. The power profile $\gamma(a)$ is optimized under the overall power constraint $\mathbb{E}[\gamma(A)] \leq \Gamma$ such that the average successfully received rate is maximized. This problem can be solved elegantly and in closed form in [17] since the associated Euler equation for the optimization problem takes on a particularly simple form.

In our case, we wish to optimize the layer power profile in order to minimize the end-to-end average distortion. Due to the fact that the source distortion-rate function, $R^{-1}(\cdot)$, is generally non-linear, the elegant solution of [17] does not apply here and in general the Euler equation associated to the distortion minimization problem cannot be solved in closed form. Hence, we take a different route and consider a more practical discretized system with L layers, where the number of source code layers coincides with the number of superposition channel coding layers and each layer has source coding rate r_s bit/source symbol and channel coding rate r_c bit/channel uses, so that $\eta = r_c/r_s$. Clearly, by letting L arbitrarily large with r_s arbitrarily small our numerical computable solution will approach arbitrarily closely the solution of the non-computable problem defined in the continuous domain. Moreover, our formulation encompasses either ideal Gaussian channel coding with ideal successive refinement source coding, either practical given channel codes and multistage source codes.

A general successive refinement source code is defined by a distortion function $D_\ell = D(\ell r_s)$ (where $D_0 = 1$), where ℓ is the number of layers successively decoded. The superposition coding scheme is obtained by summing independently selected codewords of a "a basic channel code" $\mathcal{C}'$ modulated at different power levels. In general $\mathcal{C}'$ is identified by the rate SNR-threshold pair (r_c, τ) such that for SNR larger than τ the code yields *acceptable* performance (roughly speaking, low-enough bit-error rate). The transmitted superposition codeword is given by

$$\mathbf{x} = \sum_{\ell=1}^{L} \sqrt{\gamma_\ell}\mathbf{c}'_\ell$$

where γ_ℓ and $\mathbf{c}'_\ell$ are the power level and the codeword of $\mathcal{C}'$ associated to level ℓ, respectively.

We define fading gain thresholds $0 < a_1 < \cdots < a_L$ (where $a_{L+1} = \infty$) such that layers up to ℓ can be decoded if $A \in [a_\ell, a_{\ell+1})$. The condition for successive decodability of the superposition code up to layer ℓ is given by

$$\frac{a_\ell \gamma_\ell}{1 + a_\ell \sum_{j=\ell+1}^{L} \gamma_j} \geq \tau \tag{21}$$

The resulting average distortion is given by

$$D_{\text{av}}(r_s, \gamma_1, \ldots, \gamma_L) = F_A(a_1) + \sum_{\ell=1}^{L} D_\ell \left(F_A(a_{\ell+1}) - F_A(a_\ell)\right) \tag{22}$$

The levels a_ℓ are uniquely defined by the power levels γ_ℓ by imposing the constraint (21) with equality. Conversely, the γ_ℓ's can be expressed in terms of the a_ℓ's by solving the triangular linear system $a_\ell\gamma_\ell - \tau a_\ell \sum_{j=\ell+1}^{L} \gamma_j = \tau, \quad \ell = 1, \ldots, L$, which yields $\gamma_\ell = \frac{\tau}{a_\ell} + \frac{\tau^2}{a_{\ell+1}} + \sum_{j=\ell+2}^{L} \frac{\tau^2}{a_j} (1+\tau)^{j-\ell-1}$. In order to minimize D_{av} with respect to $\{\gamma_1, \ldots, \gamma_L\}$ subject to $\sum_\ell \gamma_\ell \leq \Gamma$ we consider the Lagrangian functional

$$\Phi = D_{\text{av}}(r_s, \gamma_1, \ldots, \gamma_L) + \lambda \sum_{\ell=1}^{L} \gamma_\ell \tag{23}$$

The ℓ-th partial derivative is given by

$$\frac{\partial \Phi}{\partial \gamma_\ell} = \sum_{j=1}^{\ell-1} \left[D_{j-1} - D_j\right] a_j^2 f_A(a_j) - \left[D_{\ell-1} - D_\ell\right] \frac{1}{\tau} a_\ell^2 f_A(a_\ell) + \lambda \tag{24}$$

¿From the Kuhn-Tucker conditions, we see that these derivatives must be zero for $\gamma_\ell > 0$, and non-negative for $\gamma_\ell = 0$. With the substitution $x_\ell = a_\ell^2 f_A(a_\ell)$, the system given by $\frac{\partial \Phi}{\partial \gamma_\ell} = 0$ is linear and lower triangular, and the solution is given by $x_\ell = \frac{\lambda\tau(1+\tau)^{\ell-1}}{D_{\ell-1}-D_\ell} = \lambda \mathcal{G}_\ell$ where $\mathcal{G}_\ell \triangleq \frac{\tau(1+\tau)^{\ell-1}}{D_{\ell-1}-D_\ell}$.

Fig. 5 shows the average distortion achievable with the successive refinement and superposition coding scheme over a BF-AWGN channel with Rayleigh fading for spectral efficiency $\eta = 8/3$ and an i.i.d. Gaussian source. Using ideal Gaussian codes, characterized by the rate-threshold pair $r_c = \log(1+\tau)$ and an ideal successive refinement source code characterized by $D_\ell = 2^{-2r_s\ell}$, we see that $r_s = 1/4$ achieves already almost optimal average distortion performance (the optimal is for $r_s \to 0$). We have designed and simulated a *practical* system obtained by using our multistage source code (with basic code of rate $1/4$ and 128 states) and as $\mathcal{C}'$ the turbo code of rate $1/3$ of [15] with interleaving size 65536 and generators $(37, 21)$ mapped onto QPSK modulation. For comparison, an *ideal* system based on separation theorem, transmitting a single layer with optmized rate r_s is shown. This single-level optimized system is representative of a conventional approach, where a non-successive refinement code is coupled with a channel code, as in the examples of the previous section.

Fig. 6 shows the number of layers that a user can reliable decode and the associated reconstruction quality of the practical system with multilevel coding versus the channel istantaneous SNR for given average SNR $\Gamma = 29.75$dB. It is interesting to notice that the RSNR smoothly degrades as a function of the SNR yielding acceptable reconstruction quality for low SNR conditions.

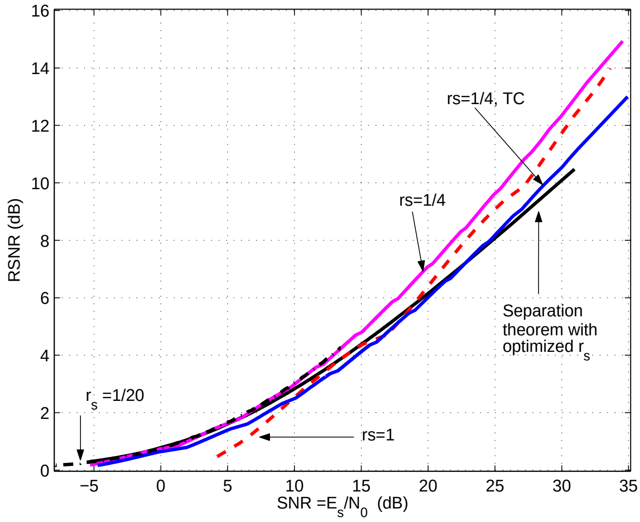


Figure 5: RSNR vs SNR for multilevel coding for $\eta = 8/3$ with ideal (Gaussian) channel and source code, $r_s = 1/4$ and 128 states convolutional source code and $(37, 21)$-TC with rate $1/3$. The ideal ssystem based on separation theorem optimized for each SNR is given as a comparison.

References

[1] W. H. R. Equitz and T. M. Cover, "Successive refinement of information," *IEEE Trans. on Information Theory*, vol. 37, no. 2, pp. 269–275, Mar. 1991.

[2] B. Rimoldi, "Successive refinement of information: characterization of the achievable rates," *IEEE Trans. on Information Theory*, vol. 40, no. 1, pp. 253–259, Jan. 1994.

[3] E. Tuncel and K. Rose, "Additive successive refinement," *IEEE Trans. on Information Theory*, vol. 49, no. 8, pp. 1983–1991, Aug. 2003.

[4] A. Lapidoth, "On the role of mismatch in rate distortion theory," *IEEE Trans. on Information Theory*, vol. 43, no. 1, pp. 38–47, Jan. 1997.

[5] L. Lastras and T. Berger, "All sources are nearly successively refinable," *IEEE Trans. on Information Theory*, vol. 47, no. 3, pp. 918–926, Mar. 2001.

[6] T. Cover and J. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

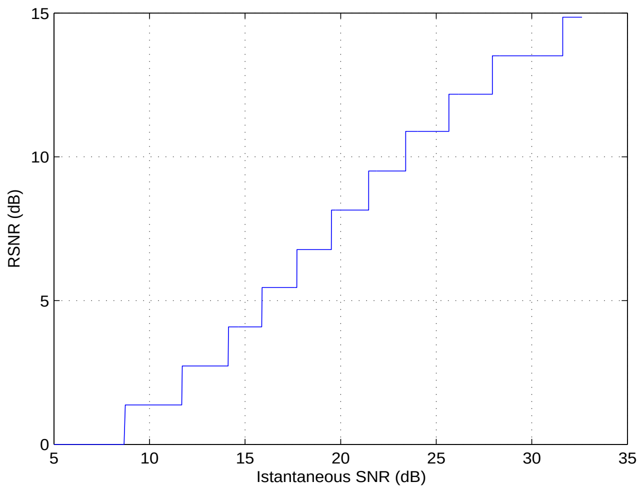


Figure 6: RSNR vs SNR for multilevel coding for $\eta = 8/3$ with $r_s = 1/4$ and 128 states convolutional source code and $(37, 21)$-TC with rate $1/3$. $\Gamma = 29.75$dB.

[7] D.S. Taubman and M.W. Marcellin, "JPEG2000: standard for interactive imaging," in *Proceedings of the IEEE*, Aug. 2002, pp. 1336–1357.

[8] R. Zamir and M. Feder, "On universal quantization by randomized uniform/lattice quantizer," *IEEE Trans. on Information Theory*, vol. 38, no. 2, pp. 428–436, Mar. 1992.

[9] G. D. Jr. Forney, "Geometrically uniform codes," *IEEE Trans. on Information Theory*, vol. 37, no. 5, pp. 1241–1260, Sept. 1991.

[10] M.W. Marcellin, *Trellis coded quantization: an efficient technique for data compression*, Ph.D. thesis, Texas A&M Univ, Dec. 1987.

[11] G. Ungerboeck, "Channel coding with multilevel/phase signal," *IEEE Trans. on Information Theory*, vol. 28, pp. 56–67, Jan. 1982.

[12] T.R. Fischer and M. Wang, "Entropy-constrained trellis-coded quantization," *IEEE Trans. on Information Theory*, vol. 38, no. 2, pp. 415–426, Mar. 1992.

[13] D. Baron and Y. Bresler, "An $o(n)$ semipredictive universal encoder via the bwt," *IEEE Trans. on Information Theory*, vol. 50, no. 5, pp. 928–937, May 2004.

[14] M. Gastpar, Rimoldi B., and Vetterli M., ""To code or not to code," in *IEEE Information Theory Symposium on Information Theory*, June 2000, p. 236.

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

[16] L.R. Bahl, J. Cocke, F. Jelinek, and J Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Information Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[17] S. Shamai and A. Steiner, "A broadcast approach for a single-user slowly fading MIMO channel," *IEEE Trans. on Information Theory*, vol. 49, no. 10, pp. 2617–2635, Oct. 2003.