Distributed Source Coding Using Syndromes (DISCUS): Design and Construction

S. Sandeep Pradhan, *Member, IEEE,* and Kannan Ramchandran, *Member, IEEE*

***Abstract*—We address the problem of compressing correlated distributed sources, i.e., correlated sources which are not co-located or which cannot cooperate to directly exploit their correlation. In this paper, we consider the related problem of compressing a source which is correlated with another source that is available only at the decoder. This problem has been studied in the information theory literature under the name of the Slepian–Wolf source coding problem for the lossless coding case, and as "rate-distortion with side information" for the lossy coding case. In this work, we provide a *constructive* practical framework based on algebraic trellis codes dubbed as DIstributed Source Coding Using Syndromes (DISCUS), that can be applicable in a variety of settings. Simulation results are presented for source coding of independent and identically distributed (i.i.d.) Gaussian sources with side information available at the decoder in the form of a noisy version of the source to be coded. Our results reveal the promise of this approach: using trellis-based quantization and coset construction, the performance of the proposed approach is 2–5 dB from the Wyner–Ziv bound.**

***Index Terms*—Channel codes, distributed source coding, rate-distortion with side information, Slepian–Wolf coding, trellis cosets, Wyner–Ziv coding.**

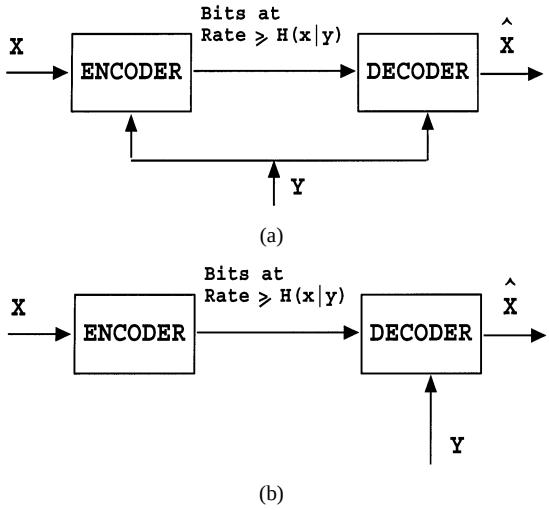


Fig. 1. Communication system: (a) Both encoder and decoder have access to the side information Y (which is correlated to X). (b) Only decoder has access to the side information Y.

I. Introduction

CONSIDER a distributed sensor array communication system [1] consisting of sensors that image a common scene independently. These sensors transmit their highly correlated information to a central processing unit that forms the best picture of the scene based on a fusion of the information collected by all of them. If the sensors could communicate with one another, they could avoid the transmission of any "redundant" information. However, such cooperation not only requires an elaborate intersensor distributed network, but also comes at the expense of substantial communication bandwidth to facilitate this. This raises the following question: what is the best one can do if there is *no communication* among the sensors? If the joint distribution characterizing the correlation structure is known, the answer is that there is *no loss in performance* in certain cases. This result is based on random coding arguments from information theory, under the name of the Slepian–Wolf coding theorem [2], [3] and its extensions. If a practical framework based on these concepts were feasible, it could have a profound impact on the evolution of some distributed applications. In fact, a key article in the 50th year Commemorative Special Issue of the Transactions on Information Theory [4] notes that *"despite the existence of potential applications, the conceptual importance of (Slepian–Wolf) distributed source coding has not been mirrored in practical data compression."*

In this paper, we address the related problem of encoding of sources in the presence of side information available only at the decoder. Consider first the problem where X and Y are correlated discrete-alphabet memoryless sources, and we have to compress X losslessly, with Y (referred to as side information) being known at the decoder but *not* at the encoder. If Y were known at both ends (see Fig. 1(a)), then the problem of compressing X is well-understood: one can compress X at the theoretical rate of its conditional entropy [2] given Y, $H(X|Y)$. But what if Y were known only at the decoder for X and not at the encoder (see Fig. 1(b))? The answer is that one can still compress X using only $H(X|Y)$ bits, the same as the case where the encoder *does* know Y. That is, by just knowing the joint distribution of X and Y, without explicitly knowing Y, the encoder of X can perform as well as an encoder which explicitly knows Y. This is known as the Slepian–Wolf coding theorem [3]. This has been extended to the lossy encoding of continuous-valued sources by Wyner and Ziv [5]–[8], who

Manuscript received November 22, 2000; revised November 5, 2002. This work was supported in part by NSF ITR under Grant CCR-0219735, and by DARPA Sensor Information Technology Office under the Sensorwebs Project F30602-00-2-0538. The material in this paper was presented in part at the IEEE Data Compression Conference (DCC), Snow Bird, UT, March 1999.

S. S. Pradhan is with the Electrical Engineering and Computer Science Department, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: pradhanv@eecs.umich.edu).

K. Ramchandran is with the Electrical Engineering and Computer Science Department, University of California, Berkeley, CA 94720 USA (e-mail: kannanr@eecs.berkeley.edu).

Communicated by P. A. Chou, Associate Editor for Source Coding.

Digital Object Identifier 10.1109/TIT.2002.808103

showed that a similar result holds in the case where X and Y are correlated jointly Gaussian random variables. If the decoder knows Y, then whether or not the encoder knows Y, the information-theoretic rate-distortion performance for coding X is identical. Extensions of the Slepian–Wolf theorem have been studied in [9]–[12]. References [13], [14] provide unified approaches to characterize the rate regions of multiterminal lossless source coding. Use of high-dimensional lattice codes to approach the Wyner–Ziv bound was explored in [15]. In that work, it was shown that in large enough dimensions, nested lattice codes approach the Wyner–Ziv rate-distortion bound asymptotically. But algorithmic constructions of actual algebraic codes for the encoder and the decoder, or their performance analysis, were not addressed.

Prior work on this topic [16] has been restricted to a purely source quantization approach inspired essentially by extensions of the generalized Lloyd algorithm [17] for quantizer design. In this work, inspired by [5], we propose instead a framework resting on algebraic channel coding principles. Of course, this being a source coding problem, there is no avoiding source quantization issues.[1] The main contribution of this work is the construction of efficient algebraic trellis codes for this problem in an attempt to approach the Wyner–Ziv bound, with computationally efficient encoding and decoding algorithms, as well as an analysis of their performance on Gaussian signals. The paper is organized as follows. In Section II, we give preliminaries, and highlight intuition behind the concept of source coding with side information. In Section III, an extension of these concepts to the encoding of sources with a fidelity criterion is described. We briefly mention the information-theoretic rate region in Section IV. Design of trellis codes and numerical results are presented in Section V, and Section VI concludes the paper.

II. Preliminaries

In this section, we give examples to illustrate the basic concepts which will be used in Sections III–V. The arguments (inspired by the tutorial paper [5]) of this section are presented here for completeness. Consider a simple but interesting illustrative example that is related to the key intuition behind the result and our approach. Suppose X and Y are equiprobable 3-bit binary words that are correlated in the following sense: the Hamming distance between X and Y is no more than one. If Y (side information) is available to both the encoder and the decoder, clearly we can describe X using 2 bits (there are only four possibilities for the modulo-two binary sum of X and Y: $\{000,001,010,100\}$). Now what if Y were revealed *only* to the decoder but not the encoder: could X still be described using only 2 bits of information?

A moment's thought reveals that the answer is indeed yes. The solution consists in realizing that since the decoder knows Y, it is wasteful for X to spend any bits in differentiating between $\{X = 000$ and $X = 111\}$, since the Hamming distance between these two words is 3. Thus, if the decoder knows that either $X = 000$ or $X = 111$, it can resolve this uncertainty by checking *which of them is closer in Hamming distance to* Y, and declaring that as the value of X. Note that the set $\{000, 111\}$ is a 3-bit repetition code. Likewise, in addition to the set $\{000, 111\}$, each of the following three sets for X: $\{100, 011\}$, $\{010, 101\}$, and $\{001, 110\}$ is composed of pairs of words whose Hamming distance is 3. These are just simple variants *or cosets* of the 3-bit repetition code, and they cover the space of all binary 3-tuples that X can assume. Thus, instead of describing X by its 3-bit value, we encode *which coset* X *belongs to*, incurring a cost of 2 bits, just as in the case where Y is known to both encoder and decoder.

We can recast the solution of the above example in the language of algebraic channel codes.[2] We want to emphasize that Wyner in [5] first proposed the use of capacity-achieving linear codes to achieve the Slepian–Wolf bound. Recall that a linear block channel code [19] is specified by its 3-tuple (n, k, d), where n is the code length, k is the message length, and d is the minimum distance of the code. In the above example, we considered the cosets of the linear $(3, 1, 3)$ repetition code. Every coset of a linear code is associated with a unique *syndrome* [20]. A word about the notation. Bold-faced lower and upper case letters represent vectors and matrices, respectively. Recall that the syndrome $\boldsymbol{s}$ associated with a linear channel code is defined as $\boldsymbol{s}^T = \boldsymbol{H}\boldsymbol{c}^T$, where $\boldsymbol{H}$ is the parity-check matrix of the code, $\boldsymbol{c}$ is any valid codeword, ^{T} denotes transposition, and $\boldsymbol{s}$, $\boldsymbol{c}$ are row vectors.

In general, if $X, Y \in \{0,1\}^n$ and $d_H(X,Y) \leq t$, the encoding of X is done as follows. The encoder observes X and sends the index of the coset (say l) of an appropriately chosen binary linear code with parameters $(n, k, 2t+1)$, in which it resides. The decoder recovers the value of X to be that vector in the coset l which is closest in Hamming distance to Y. The rate of transmission is $(n-k)$ bits per sample. This meets the Slepian–Wolf bound which is $(n-m)$ bits per sample where

$$m = n - \log \sum_{i=0}^{t} \binom{n}{i}$$

only if $k = m$ which is defined as the sphere-packing[3] bound [20] (note that, in general, $k \leq m$). This can be systematically implemented as follows. Let $\boldsymbol{H}$ be the parity-check matrix of a binary linear $(n, k, 2t+1)$ code in its systematic form, i.e., $\boldsymbol{H} = [\boldsymbol{A}|\boldsymbol{I}]$. Let $\boldsymbol{s}^T = \boldsymbol{H}\boldsymbol{x}^T$ be the syndrome when the outcome of X is $\boldsymbol{x}$. The encoder transmits $\boldsymbol{s}$ to the decoder. Let $\boldsymbol{y}' = \boldsymbol{y} \oplus \boldsymbol{a}$, where $\boldsymbol{a}$ is any vector in the coset whose syndrome is $\boldsymbol{s}$ ($\oplus$ denotes addition modulo-2). Find a vector $\boldsymbol{x}'$ closest to $\boldsymbol{y}'$ in the coset whose syndrome is the all-zero vector. It can be seen that $\boldsymbol{x} = \boldsymbol{x}' \oplus \boldsymbol{a}$. Since any vector in the coset with syndrome $\boldsymbol{s}$ will serve our purpose, we take the vector $[\boldsymbol{0}|\boldsymbol{s}]$ to be $\boldsymbol{a}$ in all future discussions where $\boldsymbol{0}$ is the all-zero vector of length k. Simple

[1]Due to the influence of channel coding, in this framework there is the concept of *source coding with an "outage" (or error) probability*. For transmission scenarios, the system may well be willing to accept poor quality a very tiny percentage of the time (the outage period) for the benefit of having a superior quality for normal operation. Finally, one can absorb the error probability into excess average source distortion.

[2]Connections between the Slepian–Wolf coding result and channel coding principles have been studied in the information theory literature [3], [5], [18], and they have been used in the context of random binning arguments to obtain achievable bounds.

[3]The repetition code is "perfect," i.e., it meets the sphere-packing bound.

extensions of this approach can be applied to nonsystematic parity-check matrices which have specific decoding algorithms.

If X and Y are correlated binary random variables, where $Y = X \oplus e$, $P[X = 1] = 1/2$, $e \in \{0,1\}$ is independent of X and $P[e = 1] = p$. The above method of encoding X by employing an $(n, k, 2t + 1)$ linear code would result in a probability of correct decoding given by:

$$P = \sum_{i=0}^{t} \binom{n}{i} (1-p)^{n-i} p^i. \tag{1}$$

The above algorithm can be extended to convolutional codes as follows. Similar to block codes, we can also define a syndrome sequence in convolutional codes. Let $\boldsymbol{H}(\boldsymbol{D}) = [\boldsymbol{A}(\boldsymbol{D})|\boldsymbol{I}]$ be the systematic parity-check polynomial matrix of a convolutional code, and let $\boldsymbol{x}(\boldsymbol{D})$ be the source sequence in polynomial representation: $\boldsymbol{x}(\boldsymbol{D}) = \sum_{i=1}^{L} x_i D^{i-1}$, where x_i denotes the ith element[4] of $\boldsymbol{x}$. The encoder sends the syndrome sequence $\boldsymbol{s}(\boldsymbol{D})^T = \boldsymbol{H}(\boldsymbol{D})\boldsymbol{x}(\boldsymbol{D})^T$ to the decoder. Let $\boldsymbol{a}(\boldsymbol{D}) = [\boldsymbol{0}|\boldsymbol{s}(\boldsymbol{D})]$ and $\boldsymbol{y}'(\boldsymbol{D}) = \boldsymbol{y}(\boldsymbol{D}) \oplus \boldsymbol{a}(\boldsymbol{D})$, where $\boldsymbol{y}(\boldsymbol{D})$ is the side-information sequence. In the coset with the all-zero syndrome, let $\boldsymbol{x}'(\boldsymbol{D})$ be the sequence closest to $\boldsymbol{y}'(\boldsymbol{D})$, which can be found using the Viterbi algorithm [21] for $\boldsymbol{H}(\boldsymbol{D})$. Now the reconstruction $\hat{\boldsymbol{x}}(\boldsymbol{D}) = \boldsymbol{x}'(\boldsymbol{D}) \oplus \boldsymbol{a}(\boldsymbol{D})$.

III. Encoding With a Fidelity Criterion

A. Problem Formulation

Here we consider the continuous-valued source X and side information Y. Specifically, X and Y are correlated memoryless processes characterized by independent and identically distributed (i.i.d.) sequences $\{X_i\}_{i=1}^{\infty}$ and $\{Y_i\}_{i=1}^{\infty}$, respectively. In this paper, we consider a simple correlation structure between the source and the side information to illustrate the essential concepts, although the proposed approach can be extended to capture more elaborate correlation structures. We consider the specific case where Y is a noisy version of X: i.e., $Y_i = X_i + N_i$, where $\{N_i\}_{i=1}^{\infty}$ is also continuous valued (defined on the real line $\mathbb{R}$), i.i.d, and independent of the X_i's. As before, the setup is that *the decoder alone has access to the Y process (side information),* and the task is to optimally compress the X process. For the rest of this paper, we will confine ourselves to the case where the X_i's and N_i's are zero-mean Gaussian random variables with known variances, so as to benchmark our performance against the theoretical performance bounds.

The goal is to form the best approximation $\hat{X}$ to X given an encoding rate of R bits per sample. We assume encoding in blocks of length L. Let $\rho(\cdot)$ be a function $\rho: \mathbb{R} \times \mathbb{R} \longrightarrow \mathbb{R}^+$. Let the distortion measure be $\rho(\cdot)$ over the L-sequence, and we assume an additive distortion measure

$$\rho(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{1}{L} \sum_{i=1}^{L} \rho(x_i, \hat{x}_i). \tag{2}$$

This problem can be posed as minimizing the rate of transmission R such that the reconstruction fidelity $E[\rho(X, \hat{X})]$ is less than a given value D, where $E(\cdot)$ is the expectation operator.

[4]Note that, in general, $x_i \in \{0,1\}^m$ for some $m \geq 1$.

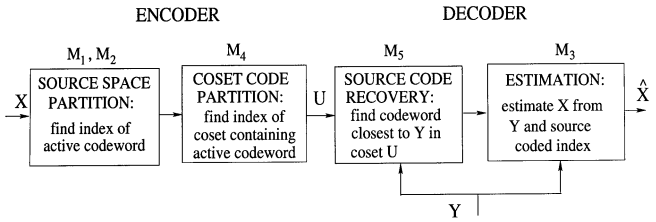


Fig. 2. Encoder and decoder: X is quantized, and the index of the coset containing the quantized codeword is sent to the decoder. The decoder finds this codeword by decoding the side information in the given coset.

The encoder is a mapping from the input space to the index set: $\mathbb{R}^L \longrightarrow \{1, 2, \ldots, 2^{LR}\}$, and the decoder is a mapping from the product space of the encoded index set and the correlated L-sequence Y to the L-sequence reconstruction: $\{1, 2, \ldots, 2^{LR}\} \times \mathbb{R}^L \longrightarrow \mathbb{R}^L$. For the remainder of the paper, we focus on the mean-squared error (MSE) distortion: $\rho(x, \hat{x}) = (x - \hat{x})^2$.

B. Design Algorithm

This problem involves an intricate interplay of source coding, channel coding, and estimation. We consider the following system comprising these three components. This system involving the encoder and the decoder, as shown in Fig. 2, consists of five mappings: $\{M_i\}_{i=1}^{5}$. First consider the source coding component of the system.

Source Coding (M_1, M_2): Due to the finite rate constraint on the information transmitted, the source X has to be quantized. A source codebook is constructed for a given reconstruction fidelity. The source space $\mathbb{R}^L$ is partitioned into 2^{LR_s} disjoint regions, where R_s is defined as the source rate. It is a mapping

$$M_1 : \mathbb{R}^L \longrightarrow \{1, 2, \ldots, 2^{LR_s}\}.$$

Let $\mathcal{T} = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_{2^{LR_s}}\}$ denote the set of 2^{LR_s} disjoint regions. Each region in the above partition is associated with a representation codeword. The set of representation codewords is referred to as the source codebook $(\mathbb{S})$. This is a mapping

$$M_2 : \{1, 2, \ldots, 2^{LR_s}\} \longrightarrow \mathbb{R}^L.$$

The objective is to design these mappings M_1 and M_2. The source is quantized to one of the codewords in $\mathbb{S}$ and the index of the quantized codeword is made available to the decoder error free, by transmitting at a rate of R_s bits per sample. We refer to the representation codeword to which X is quantized as the *active codeword*. Let the random variable characterizing the active codeword be denoted by W. As we will see in the sequel, unlike traditional source coding, the active codeword is not used as the reconstruction for the source. Rather, the decoding further involves estimation of the source based on all the available information about the source, the result of which is used as the final reconstruction. We consider a design of the partition $\mathcal{T}$ of the source space based only on the marginal distribution of X. This can be done, for example, by generalized Lloyd algorithm [17]. Then we exploit the presence of side information in the channel coding component.

Estimation (M_3): The decoder gets the best estimate of X (minimizing the distortion) conditioned on the outcome of the

side information and the element in $\mathcal{T}$ containing X. This is given by

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{a}\in\mathbb{R}^L}{\operatorname{argmin}} E\left[\rho(\boldsymbol{X},\boldsymbol{a}) \,\middle|\, \begin{matrix}\boldsymbol{X}\in\Gamma_i \\ \boldsymbol{Y}=\boldsymbol{y}\end{matrix}\right] \tag{3}$$

for the received message i and the side-information outcome $\boldsymbol{y}$. It can be interpreted as a mapping

$$M_3 : \mathbb{R}^L \times \{1,2,\ldots,2^{LR_s}\} \longrightarrow \mathbb{R}^L.$$

The estimation error is a function of R_s, which is chosen to keep this error within the given fidelity criterion.

Channel Coding (M_4, M_5): At this stage, the system requires a transmission of R_s bits per sample to guarantee the given fidelity criterion. By exploiting the correlation between X and Y, we make the decoder recover (with an arbitrarily small probability of error) the index of the active codeword with a transmission over an error-free channel at a rate lower than R_s. This is done by noting that the random variable W characterizing the quantized source is correlated to X, and this in turn induces a correlation between W and the side information Y. This can be characterized by a conditional distribution $P(Y|W)$ of the side information given W. With this conditional distribution we can associate a fictitious channel with W as input which is observed at the encoder, and Y as output which is observed at the decoder, whose information channel capacity is greater than 0 (due to this correlation). To actually communicate W to the decoder in the absence of side information requires a transmission of R_s bits per sample. With the presence of Y at the decoder, we have this fictitious "helper" channel carrying an amount of information $I(W;Y)$ about W. The remaining uncertainty in W after observing the side information Y is

$$H(W|Y) = H(W) - I(W;Y)$$

and this is the desired final rate of transmission. The rebate in the rate of transmission is $I(W;Y)$.

Using this intuition, our goal is to get a rebate as close to $I(W;Y)$ as possible by building a practical structured "channel code" $(\mathbb{C})$ for this fictitious channel on the space of W. Let 2^{LR_c} denote the number of codewords in the designed channel code where R_c is defined as the channel rate.[5] Suppose, for a given realization, the active codeword (say w) belongs to this channel code and this is known at the decoder (say communicated by a genie), then we do not need to send any information to the receiver, as it can recover the intended codeword with a small probability of error by decoding Y in the channel code[6] $\mathbb{C}$. Since, in general, any codeword in the source codebook can be a quantization outcome with a nonzero probability, we partition the source codebook space into cosets of this channel code. The channel code is designed in such a way that each of its cosets is also an equally good channel code for the channel $P(Y|W)$. Thus, each quantization outcome belongs to a coset of this channel code, and this alone has to be conveyed to the decoder, which can then proceed to use this coset of the channel code for finding the intended active codeword.

The encoder computes the index of the coset of the channel code containing the active codeword using a mapping

$$M_4 : \{1,2,\ldots,2^{LR_s}\} \longrightarrow \{1,2,\ldots,2^{LR}\}$$

and transmits this information with rate $R = R_s - R_c$ bits per sample to the decoder. The decoder recovers the active codeword in the signaled coset by finding (channel decoding) the most likely codeword given the observed side information. This is characterized by a mapping

$$M_5 : \mathbb{R}^L \times \{1,2,\ldots,2^{LR}\} \longrightarrow \{1,2,\ldots,2^{LR_s}\}.$$

In this approach, there is always a nonzero probability of decoding error, where the side information is decoded to a wrong codeword, and this can be made arbitrarily small by designing efficient channel codes. For a given region in $\mathcal{T}$, the choice of the representation codeword determines $I(W;Y)$, and hence R_c. We refer to this constructive framework as distributed source coding using syndromes (DISCUS).

Summary of Design Algorithm: The rate of transmission and the distortion performance depend on the source and the channel codebooks. Thus, for a given fidelity criterion, our objective is to minimize R_s and maximize R_c. This is summarized in the following.

1) M_1 and M_3: Minimize R_s such that the reconstruction distortion is within the given criterion.
2) M_2: The ideal goal is to maximize the mutual information[7] between W and Y, $I(W;Y)$. But as an approximation to it, we take the representation codeword of Γ_i as that vector for which Γ_i is its Voronoi region (which also amounts to a constraint on M_1).
3) M_4: Maximize the rate R_c of the channel code $\mathbb{C}$ (and each of its cosets) such that probability of decoding error meets a desired tolerance level.
4) M_5: Minimize computational complexity of the decoding rule.

The final reconstruction $\hat{\boldsymbol{X}}$ is given by

$$\hat{\boldsymbol{X}} = M_3\left[\boldsymbol{Y}, M_5\left\{\boldsymbol{Y}, M_4(M_1(\boldsymbol{X}))\right\}\right]$$

for the source and the side-information vectors $\boldsymbol{X}$, $\boldsymbol{Y}$, respectively. The system has two separate entities in the design of the encoder and the decoder. The quantizer and the estimator are designed to minimize R_s, and the channel code is designed to maximize R_c. The fidelity during correct decoding is guaranteed by the source coding design, and the probability of decoding error is controlled by the channel coding design. The distortion during decoding error can be bounded.

[5]This should not be confused with the actual channels used for the transmission of information.

[6]This can be interpreted as transmitting w over this fictitious channel and observing the output of this channel Y as side information at the decoder.

[7]In practical quantization, it may be computationally difficult to characterize explicitly the channel $P(Y|W)$. So we use a suboptimal approximation to it for the rest of the paper.

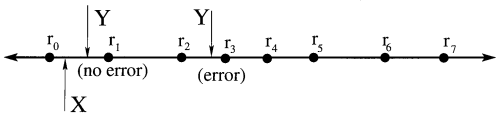


Fig. 3. Reconstruction levels of scalar quantizer with eight levels.

C. Constructions Based on Trellis Codes

In this subsection, we give general design methods for trellis codes in the DISCUS framework.

1) Scalar Quantization and Memoryless Coset Construction: First consider an example of a fixed-rate (rate-$\log_2 V$) Lloyd–Max scalar quantizer [17] with V levels, designed for the distribution of X. Let $V = 8$ for the present discussion. Let $\nabla = \{r_0, r_1, \ldots, r_{V-1}\}$ be the set of reconstruction levels as shown in the Fig. 3. Note that ∇ partitions the real line into V intervals each associated with one of the reconstruction levels. Let $\mathcal{T} = \{\Gamma_i\}_{i=0}^{V-1}$ be the partition of $\mathbb{R}$, where Γ_i is the open interval $\left(\frac{r_{i-1}+r_i}{2}, \frac{r_i+r_{i+1}}{2}\right)$ and we take $r_{-1} = -\infty$ and $r_V = \infty$. We first quantize the source sample by sample using ∇. Thus, the source codebook $\mathbb{S}$ is given by ∇, and $R_s = 3$ bits/sample. Next we construct $\mathbb{C}$ by partitioning the set ∇ into $M(\leq V)$ cosets. For illustration, let $M = 2$. We group r_0, r_2, r_4, and r_6 into one coset. Similarly, r_1, r_3, r_5, and r_7 are grouped into another coset. This is done to keep the minimum distance between any two words in every coset as large as possible. Thus, the channel code is defined as $\mathbb{C} = \{r_0, r_2, r_4, r_6\}$, making $R_c = 2$ and $R = 1$ bit/sample. The representation codeword r_i is the centroid of the disjoint region Γ_i for $0 \leq i \leq V-1$.

The decoder deciphers (with a small probability of error) the active codeword by finding the codeword which is closest[8] to Y in the coset whose index is sent by the encoder. Then the optimal estimate $\hat{x}$ can be computed as follows:

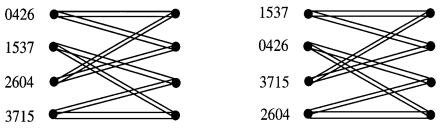
$$\hat{x} = \underset{a \in \mathbb{R}}{\operatorname{argmin}} E\left[\rho(X,a) \,\middle|\, \begin{matrix} X \in \Gamma_i, \\ Y = y \end{matrix}\right] \tag{4}$$

where y is the outcome of Y, i is the index of the active codeword. This can be computed numerically using error functions [23].

Mappings : M_1 is the scalar quantizer, M_2 is rule of selecting the centroids as the source codewords, M_3 is the estimator, M_4 is the memoryless partition of the quantizer into cosets, and M_5 is the minimum-distance decoding rule. Note for this construction $L = 1$. For a general R, $\mathbb{S}$ is a scalar quantizer with 2^{R_s} levels, and is partitioned into 2^R cosets, each containing 2^{R_c} words.

2) Scalar Quantization and Trellis-Based Coset Construction: We now describe scalar (memoryless) quantization which uses a coset construction *having memory*. Here, instead of coset-encoding W sample by sample, we do it as an L-sequence.

[8]This is in the sense of $\arg\max P(r_l|Y)$ for $0 \leq l \leq (V-1)$, which is the same as $\arg\min \|r_l - Y\|^2$ if we use the following approximation: 1) the quantization noise is memoryless i.i.d. Gaussian, and 2) W is uniformly distributed. The basis for this approximation is the result from [22] which states that the distribution of large-dimensional optimal lattice [20] quantization noise converges to a white Gaussian distribution in the sense of information divergence [2].



(a) (b)

Fig. 4. Trellis section for the trellis code built on an alphabet of size 8. The number of paths emanating from any state is four. (a) Principal trellis coset. (b) Complementary trellis coset.

Note that we still use fixed-length scalar quantizers for quantizing $\{X_i\}_{i=1}^L$, but the cosets are built on the space ∇^L (we use the same example for illustration). Thus, the source codebook is given by $\mathbb{S} = \nabla^L$ and $R_s = 3$ bits/sample for the present example.

Source Coding (M_1, M_2): For a general R_s

$$M_1 : \mathbb{R}^L \longrightarrow \{1, 2, \ldots, 2^{LR_s}\}$$

is the L-product scalar quantizer and

$$M_2 : \{1, 2, \ldots 2^{LR_s}\} \longrightarrow \nabla^L$$

is the L-sequence extension of the memoryless mapping

$$\{1, 2, \ldots, 2^{R_s}\} \longrightarrow \nabla$$

of Section III-C1.

The space ∇^L has 2^{3L} distinct sequences. The task is to partition this sequence space $\mathbb{S}$ into cosets of a set of sequences in such a way that the minimum distance between any two sequences in every coset is made as large as possible, while maintaining symmetry among the cosets. Let $R = 1$ bit/sample. In the following, a trellis-based partitioning with an algebraic structure is considered based on convolutional codes and set-partitioning rules of trellis-coded scalar-quantization (TCSQ) [24].

Consider a trellis code built on ∇ with block length L, having 2^{2L} sequences (code rate is $R_c = 2$ bits/sample), using a rate-$2/3$ convolutional code and a mapping $Q : \{0,1\}^3 \to \nabla$. For this example, we use Ungerboeck's four-state trellis as given in [25]. In this paper, we restrict ourselves to systematic trellis codes. ∇ is partitioned into two subsets as in Section III-C1. A coded bitstream selects points from one of these subsets depending on the state of the finite-state machine. The trellis for this code is shown in Fig. 4(a), which we call the *principal trellis coset*. For this example, let $Q(\zeta) = r_\eta$ where $\zeta \in \{0,1\}^3$ is the binary representation of η. The following notation will be used in representing the function Q of sequences or vectors.[9]

$$\boldsymbol{Q}(\boldsymbol{x}) = \boldsymbol{Q}(x_1, x_2, \ldots, x_L) = [Q(x_1), Q(x_2), \ldots, Q(x_L)]$$

where $x_k \in \{0, 1\}^3$, and is a column vector for every $1 \leq k \leq L$.

We take the set of all sequences that can be generated from the above machine (i.e., all the sequences of this trellis code) as a channel code $\mathbb{C}$ for the channel $P(Y|W)$. Clearly, $\mathbb{C} \subset \mathbb{S}$. Let $\boldsymbol{H}(\boldsymbol{D})$ be the parity-check matrix polynomial of the underlying convolutional code, and let $\boldsymbol{\Theta}$ be any sequence in ∇^L. We have $\boldsymbol{Q}^{-1}(\boldsymbol{\Theta}) \in \{0, 1\}^{3L}$. We define cosets of this channel code $\mathbb{C}$ as follows: we associate for any $\boldsymbol{s} \in \{0,1\}^L$ a coset of $\mathbb{C}$, given

[9]In the sequel, a sequence $[z_1, z_2, \ldots, z_L]$ will be denoted interchangeably as either $\boldsymbol{z}$ or $\boldsymbol{z}(\boldsymbol{D})$.

by $\mathbb{C}(\boldsymbol{s})$, which consists of all the sequences $\boldsymbol{z} \in \nabla^L$ such that $\boldsymbol{H}(\boldsymbol{D})\boldsymbol{Q}^{-1}(\boldsymbol{z}) = \boldsymbol{s}^T$. Each of 2^L cosets of $\mathbb{C}$ consists of 2^{2L} sequences. This has resulted in a partition of the space ∇^L into 2^L cosets of $\mathbb{C}$. The encoder sends the index $\boldsymbol{s}$ of the coset $\mathbb{C}(\boldsymbol{s})$ containing the quantized source sequence.

Coset Indexing (M_4): For the case of general R_s, and for any $\theta \in \{1, 2, \ldots, 2^{LR_s}\}$

$$M_4(\theta) = D_R \left[\boldsymbol{H}(\boldsymbol{D}) \{ \boldsymbol{Q}^{-1}[M_2(\theta)] \} \right] + 1 \tag{5}$$

where D_R is the R-bit representation to decimal conversion mapping: $\{0, 1\}^{RL} \rightarrow \{0, 1, 2, \ldots, 2^{RL} - 1\}$.

Decoder (M_5): The decoder receives L bits of syndrome $\boldsymbol{s}$, and L samples of the process Y. We need a computationally efficient algorithm for searching through the list of codeword sequences in a given coset $\mathbb{C}(\boldsymbol{s})$, for the most likely codeword sequence given the side-information sequence $\boldsymbol{y}$. For this, we use a modified version of the Viterbi algorithm that is suitable for *any* syndrome sequence. Consider the kth stage of the four-state trellis as shown in Fig. 4(a). This is the trellis for the coset of $\mathbb{C}$ with the all-zero syndrome. If the kth bit of the syndrome sequence is 1 rather than 0, we need to modify the labels on each edge of the principal trellis coset at the kth stage. Let $\boldsymbol{a}(\boldsymbol{D}) = [\boldsymbol{0}|\boldsymbol{0}|\boldsymbol{s}(\boldsymbol{D})]^T$, where $a_k = [0|0|s_k]^T$ for all $1 \leq k \leq L$, and $\boldsymbol{0}$ is the all-zero sequence. As discussed earlier, for the trellis code under consideration, the sequence $\boldsymbol{Q}\left[[\boldsymbol{0}|\boldsymbol{0}|\boldsymbol{s}(\boldsymbol{D})]^T\right]$ belongs to the coset whose syndrome is $\boldsymbol{s}(\boldsymbol{D})$. Thus, at the kth stage of decoding, if the kth bit of $\boldsymbol{s}(\boldsymbol{D})$ is 1, we need to shift from the principal trellis coset to the other trellis coset (there are only two trellis cosets in the given example: see Fig. 4(b)).

This is done by defining the trellis coset at the kth stage as having the edge labels $\xi = Q[Q^{-1}(r_l) \oplus a_k]$, where $r_l \in \nabla$ is the corresponding label in the principal trellis coset $a_k = [0|0|s_k]^T$ (this is possible due to the systematic trellis codes). The set of labels on the edges starting from any given state in the two trellis cosets forms a partition of ∇. For example, in the first state, the edge labels of the principal trellis coset $\{r_0, r_2, r_4, r_6\}$, and that of the complementary trellis coset $\{r_1, r_3, r_5, r_7\}$ form a partition of ∇. Thus, starting from $k = 1$ to $k = L$, at every stage we need to keep relabeling (shifting between the principal trellis coset and the other trellis coset) the edges in the trellis used in the Viterbi decoder. Let $\boldsymbol{x}'$ be the sequence which is closest to $\boldsymbol{y}$ obtained using this algorithm (characterizing the mapping M_5) with relabeled edges. This is illustrated in Fig. 5 with an example. For a general R, there will be 2^R trellis cosets. It can be noted that the minimum distance between any two words in every coset of $\mathbb{C}$ has been increased from that in the memoryless coset construction.

Estimation (M_3): After recovering the active codeword, the optimal estimate is given as follows:

$$\hat{\boldsymbol{x}} = M_3(\boldsymbol{y}, \theta) = \underset{\boldsymbol{a} \in \mathbb{R}^L}{\operatorname{argmin}} \left[E \left\{ \rho(\boldsymbol{X}, \boldsymbol{a}) \,\middle|\, \begin{matrix} \boldsymbol{Y} = \boldsymbol{y} \\ \boldsymbol{X} \in \boldsymbol{\Gamma} \end{matrix} \right\} \right] \tag{6}$$

where θ is the index of the codeword $\boldsymbol{x}'$ in $\mathbb{S}$, and $\boldsymbol{\Gamma}$ is the sequence of intervals associated with the elements of $\boldsymbol{x}'$. Since $\mathbb{S}$ is memoryless, we can simplify the preceding expression as

$$\hat{x}_k = \underset{a \in \mathbb{R}}{\operatorname{argmin}} E \left\{ \rho(X_k, a) \,\middle|\, \begin{matrix} Y_k = y_k \\ X_k \in \Gamma_i \end{matrix} \right\} \tag{7}$$

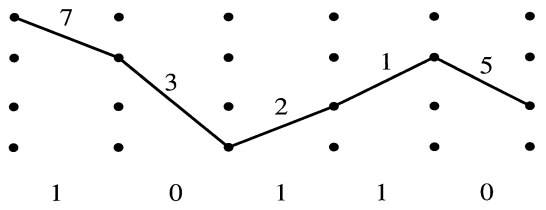


Fig. 5. An example of computation of the syndrome ($(R_s, R_c) = (3, 2)$ bits per sample): let the outcome of quantization be $7, 3, 2, 1, 5$. The syndrome is given by 10110 for five samples. Sample numbers 0, 2, and 3 use complementary trellis coset and the rest use principal trellis coset.

where $\hat{x}_k$ is the kth sample of $\hat{\boldsymbol{x}}$, for $1 \leq k \leq L$, and $x'_k = r_i \in \nabla$ (the estimation assumes correct decoding). This estimate can be computed numerically using error functions [23].

Remark 1: For a general R, $\mathbb{S}$ is the L-product scalar quantizer with 2^{LR_s} levels, and a trellis code having a rate-R_c/R_s convolutional code is built on this quantizer, which constitutes $\mathbb{C}$. The proposed decoding algorithm can be easily implemented for any (R_c, R_s) pair. The source distortion during *correct decoding* is the same as that associated with the memoryless coset construction of Section III-C1. The trellis-based coset construction reduces the probability of decoding error for a given correlation between X and Y, or alternatively, increases the range of correlation between X and Y that can be supported at the same probability of decoding error.

3) Trellis-Based Quantization and Memoryless Coset Construction: We now graduate to the use of more efficient quantizers such as TCSQ as our source codebook, and construct cosets on this space. Consider a TCSQ with rate R_s bits per sample built on a scalar quantizer alphabet ∇ of size 2^{R_s+1}. This has a finite-state machine with a rate-$R_s/(R_s+1)$ convolutional code and a mapping $Q : \{0,1\}^{R_s+1} \longrightarrow \nabla$. The set of 2^{LR_s} "valid" sequences of this TCSQ constitutes the source codebook $\mathbb{S}$. The source is quantized using $\mathbb{S}$ by applying the Viterbi algorithm.

Source Coding (M_1, M_2): M_1 corresponds to this trellis-coded quantization, and M_2 corresponds to the rule of assigning the sequences in $\mathbb{S}$ to their corresponding Voronoi regions.

Consider an example with $(R_s, R_c) = (2, 1)$ bits per sample and the trellis as shown in Fig. 4(a). We obtain a subset $\mathbb{C}$ of $\mathbb{S}$ such that the sequences in $\mathbb{C}$ are memoryless. To construct this subset, we consider a TCSQ with parallel transitions, and assign all the label points on a parallel transition to the same coset of a sequence set $\mathbb{C}$, resulting in a reduction in rate by 1 bit/sample. In other words, we partition the scalar quantizer ∇ into four sets: $\{r_0, r_4\}$, $\{r_1, r_5\}$, $\{r_2, r_6\}$, and $\{r_3, r_7\}$, and the encoder does not expend bits to differentiate between the elements of a set (i.e., labels among parallel transitions between any two connected states in the trellis). Thus, the channel code is given by $\mathbb{C} = \{r_0, r_4\}^L \subset \mathbb{S}$. This amounts to discarding the uncoded bit in the 2-bit representation of the TCSQ codewords and sending only the coded bit to the decoder.

The set of codewords in the TCSQ can be viewed as a coset code [26]. ∇ is partitioned into four cosets of $\{r_0, r_4\}$, and a bit sequence set is used to index the valid set of L-dimensional coset sequences. This is shown in Fig. 6. One of the input bits is fed into the rate-$1/2$ convolutional code and its output speci-

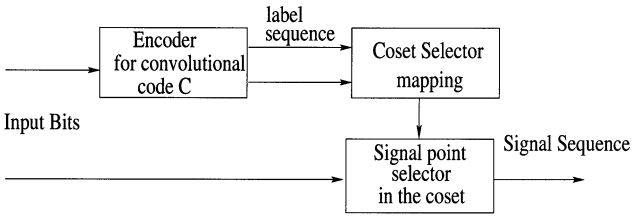


Fig. 6. Coset code: One of the input bit sequences is encoded to index the coset in the signal set. The other input bit sequence is used to select the signal point in the chosen coset.

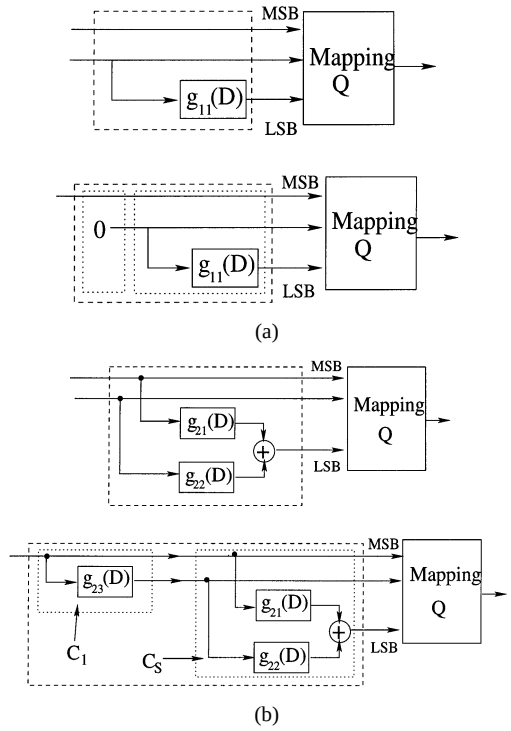


Fig. 7. Finite-state machines of $\mathbb{S}$ and $\mathbb{C}$, respectively. (a) Construction of Section III-C3. (b) Construction of Section III-C4. $\boldsymbol{C}_s$, $\boldsymbol{C}_1$ refer to the rate-$2/3$ and rate-$1/2$ convolutional codes, respectively. $g_{11}(D)$, $g_{21}(D)$, $g_{22}(D)$, and $g_{23}(D)$ are polynomials associated with the convolutional codes.

fies the cosets of $\{r_0, r_4\}$ in ∇, and the other input bit specifies the element in the given coset in ∇. Thus, every element of any trellis coded quantized sequence belongs to any of two permissible cosets among the set of 4, and the coded bit denoting this is transmitted to the decoder, while the other uncoded bit (which specifies the element in the signaled coset of $\{r_0, r_4\}$ in ∇) is discarded. The finite state machines of $\mathbb{S}$ and $\mathbb{C}$ are shown in Fig. 7(a). The two least significant bits of the 3-bit representation of all the sequences of $\mathbb{C}$ are zero.

Coset Indexing (M_4): For a general R_s, and for any $\theta \in \{1, 2, \ldots, 2^{LR_s}\}$, let $\boldsymbol{b}(\boldsymbol{D})$ denote the R_s-bit representation of the codeword with index θ in the rate-R_s TCSQ, where $b_k \in \{0, 1\}^{R_s}$ is the column vector for all $1 \le k \le L$. Let $\boldsymbol{E} = [\boldsymbol{0} | \boldsymbol{I}_R]$ be an $R \times R_s$ matrix, $\boldsymbol{0}$ denote the $R \times R_c$ all-zero matrix, and $\boldsymbol{I}_R$ denote $R \times R$ identity matrix. We have $M_4(\theta) = D_R(\boldsymbol{E}\boldsymbol{b}(\boldsymbol{D})) + 1$.

Decoder (M_5): The decoder recovers the discarded uncoded bits using the side information. The received bit sequence is passed through the finite-state machine of the rate-$2/3$ convolutional code used by the encoder, with the other bit sequence as a variable. This specifies the coset sequence (each element belonging to one of four cosets of $\{r_0, r_4\}$ in ∇) which contains the trellis-coded quantized source sequence. The actual quantized source value in the given coset is determined by minimum-distance decoding the side information sample by sample.

Estimation (M_3): The minimum MSE (MMSE) estimate (characterized by the mapping M_3) is the conditional mean of the source sequence given the index of the Voronoi region in $\mathbb{S}$ containing it, and the side information outcome. In this case, the L-dimensional Voronoi region of every quantized sequence is nonseparable. Finding this optimal estimate is computationally intractable. In Section V, we provide a suboptimal approximation to this estimate by getting insights from the information-theoretic analysis of [7].

Remark 2: In general, a rate-R_s TCSQ using a 2^{R_s+1}-level scalar quantizer, a rate-$(R_s - R_c)/(R_s - R_c + 1)$ convolutional code, and 2^{R_c} parallel transitions[10] between any two connected states in the trellis, is taken as the source code $\mathbb{S}$. The alphabet ∇ of the TCSQ is partitioned into $2^{R_s - R_c + 1}$ cosets of a set containing 2^{R_c} elements. At any state, only $2^{R_s - R_c}$ cosets are allowed by the finite-state machine, and this is transmitted to the decoder. Note the difference between the constructions of Sections III-C2 and III-C3. In the former, the source is scalar quantized and the decoder finds the active codeword using the Viterbi algorithm. In the latter, the source is trellis coded quantized using the Viterbi algorithm at the encoder, and the decoder finds the active codeword by scalar decoding the side information.

4) Trellis-Based Quantization and Trellis-Based Coset Construction: The most general trellis-based system would use a trellis-based quantizer *and* a trellis-based coset construction. The source is encoded using a TCSQ with rate R_s bits per sample (as shown in Fig. 7(b) for $R_s = 2$ and $R_c = 1$), built on an alphabet ∇ with 2^{R_s+1} levels. The source code $\mathbb{S}$ is the set of 2^{LR_s} valid sequences permitted by the structure of the TCSQ. The TCSQ has a rate-$R_s/(R_s + 1)$ convolutional code $\boldsymbol{C}_s$ with a mapping $Q : \{0,1\}^{R_s+1} \longrightarrow \nabla$. $\mathbb{S}$ has to be partitioned into 2^{LR} cosets each having 2^{LR_c} sequences. The channel code $\mathbb{C}$ which enables this partition constitutes all 2^{LR_c} sequences that can be generated by the finite-state machine which uses Q and a convolutional code $\boldsymbol{C}_c$, obtained as a concatenation (as shown in Fig. 7(b)) of $\boldsymbol{C}_s$ with $\boldsymbol{C}_1$, where $\boldsymbol{C}_1$ is a rate-R_c/R_s convolutional code. The index of the coset of $\mathbb{C}$ in $\mathbb{S}$ containing the active codeword is computed (mapping M_4) by using the parity-check matrix $\boldsymbol{H}_1$ (by multiplying it with the R_s-bit representation of the TCSQ sequence) of $\boldsymbol{C}_1$, and this is transmitted to the decoder. The decoder finds the active codeword by decoding the side-information sequence in the given coset of $\mathbb{C}$ using the Viterbi algorithm (for the trellis based on Q and $\boldsymbol{C}_c$), which involves a relabeling of the trellis associated with $\mathbb{C}$. We assume that the generator matrices $\boldsymbol{G}_1(\boldsymbol{D}) = [\boldsymbol{I} | \boldsymbol{A}_1(\boldsymbol{D})]$ and $\boldsymbol{G}_s(\boldsymbol{D}) = [\boldsymbol{I} | \boldsymbol{A}_s(\boldsymbol{D})]$, respectively, of $\boldsymbol{C}_1$ and $\boldsymbol{C}_s$ are in the

[10]In general, the TCSQ can have N parallel transitions, with $N \ge 2^{R_c}$. In such cases, one needs to partition these N parallel transitions into cosets, with each coset having 2^{R_c} parallel transitions. The encoder expends $(\log_2 N - R_c)$ bits per sample out of R bits per sample to specify this coset.

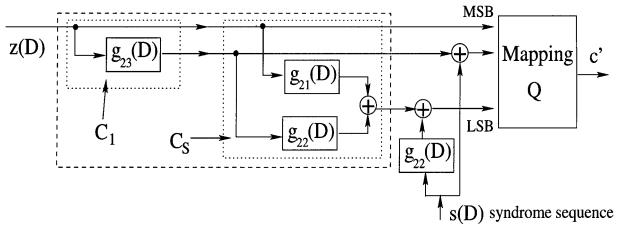


Fig. 8. Relabeling of the trellis of $\mathbb{C}$ for different syndrome sequences.

systematic form. Consider any sequence $\boldsymbol{c}$ in $\mathbb{C}$ (coset with the all-zero syndrome)

$$\boldsymbol{c} = \boldsymbol{Q}\left(\boldsymbol{z}(\boldsymbol{D})\boldsymbol{G}_1(\boldsymbol{D})\boldsymbol{G}_s(\boldsymbol{D})\right)$$

for some sequence $\boldsymbol{z}(\boldsymbol{D})$, where $z_k \in \{0,1\}^{R_c}$ for all $1 \le k \le L$. Let $\boldsymbol{s}(\boldsymbol{D})$ be the syndrome sequence received by the decoder, where $s_k \in \{0,1\}^{R}$ for all $1 \le k \le L$. The sequence $\boldsymbol{c}$ is relabeled to

$$\boldsymbol{c}' = \boldsymbol{Q}\left(\boldsymbol{z}(\boldsymbol{D})\boldsymbol{G}_1(\boldsymbol{D})\boldsymbol{G}_s(\boldsymbol{D}) \oplus [\boldsymbol{0}|\boldsymbol{s}(\boldsymbol{D})]\boldsymbol{G}_s(\boldsymbol{D})\right)$$

where $\boldsymbol{0}$ is the all-zero vector of length R_c. This is shown in Fig. 8 for the case of $(R_s, R_c) = (2, 1)$ bits per sample. Note that the encoder uses the Viterbi algorithm on the trellis structure of the source code $\mathbb{S}$, and the decoder uses it on that of the channel code $\mathbb{C}$. The decoder gets the best estimate of the source based on all the received information. Note that, in general, if $\boldsymbol{C}_s$ and $\boldsymbol{C}_1$ have η_1 and η_2 memory elements, respectively, the trellis of $\mathbb{S}$ and $\mathbb{C}$ will have 2^{η_1} and $2^{\eta_1+\eta_2}$ states.

Mappings : M_1 is the rate-R_s TCSQ, M_2 is the rule of assigning the codewords of TCSQ to its corresponding Voronoi regions, M_3 is the nonseparable MMSE estimator, M_4 is computed using the parity-check matrix of $\boldsymbol{C}_1$, and M_5 is the decoding rule of the Viterbi algorithm on the trellis of $\mathbb{C}$. When the source code has memory, the L-dimensional Voronoi region is nonseparable, and we provide a suboptimal fast algorithm for computing an estimate of the source in Section V.

5) Distance Properties: In this subsection, we analyze the distance properties of these constructions by studying the group structure [27] of these codes. We need the partitions of the source code considered in the preceding sections to have good distance properties. Since the probability of decoding error depends on the minimum distance of each coset in the partition, the minimum distance of each coset should be as large as possible. For the analysis of the distance properties we assume the TCSQ to be built on a uniform scalar quantizer. The symmetry properties of the trellis codes have been studied in [28] in a group-theoretic [27] setting. Let us analyze the proposed approach in this setting; we assume familiarity with the definition of isometry and symmetry groups as given in [28]. A geometric figure (a collection of points) S is said to be geometrically uniform if for any two points $x, y \in S$, there exists an isometry (distance-preserving function) u_{xy} such that $u_{xy}(x) = y$ and $u_{xy}(S) = S$. It was shown in [28] that if a signal set is geometrically uniform, then all the Voronoi regions of its elements have the same shape. A generating group $U(S)$ of S is a subgroup of the symmetry group[11] that is minimally sufficient to generate S from any arbitrary point $s_0 \in S$. Thus, we have $S = \bigcup_{u \in U(S)} u(s_0)$. Let U' be a normal subgroup of $U(S)$. Let the set generated by U' be S'. Let A be a label group that is isomorphic to the quotient group $U(S)/U(S')$ (we refer to U' as $U(S')$). Then the elements of A can be used to label the cosets of $U(S')$ in $U(S)$ which is determined by the mapping $m : A \longrightarrow S/S'$. For simplicity, let us also assume that A is abelian [28]. For any element $c \in A$, we have

$$S'(c) = \bigcup_{u \in U(S')} u_c[u(s_0)] \tag{8}$$

and $u_c \in U(S)$ is the coset representative of $S'(c)$. A generalized coset code $\mathbb{C}(S/S'; \mathcal{C})$ is the set of all sequences $\{\boldsymbol{s} \in \boldsymbol{m}(\boldsymbol{c}), \boldsymbol{c} \in \mathcal{C}\}$, where $\mathcal{C}$ is a subgroup of A^I, I is some index set, and $\boldsymbol{m} : A^I \longrightarrow (S/S')^I$ is the sequence extension[12] of the mapping $m : A \longrightarrow S/S'$. The generalized coset code can be represented as

$$\mathbb{C}\left(S/S'; \mathcal{C}\right) = \bigcup_{\boldsymbol{c} \in \mathcal{C}} \boldsymbol{m}(\boldsymbol{c}) = \bigcup_{\boldsymbol{c} \in \mathcal{C}} u_{\boldsymbol{c}}\left((S')^I\right). \tag{9}$$

In the following, we consider a trellis code as an example of generalized coset codes.

Example: Consider the uniform infinite lattice $S = \mathbb{Z} - 1/2$. Let the generating group $U(S) = V \cdot T(2\mathbb{Z})$ where V denotes the group with two elements $\{I_a, R_-\}$, and I_a, R_- denote identity and reflexion about the origin, respectively, with composition as the group operation, $T(\cdot)$ denotes translation. Let $U(S') = T(4\mathbb{Z})$. Clearly, $U(S)/U(S')$ is isomorphic to $A = \mathbb{Z}_2^2$. Let $I = \{1, 2, \ldots, L\}$. The set of all L-sequence of codewords denoted by $\mathcal{C}$, that can be generated by a rate-$1/2$ convolutional code is a subgroup of $\mathbb{Z}_2^{2L} = A^I$. $\mathbb{C}(S/S'; \mathcal{C})$ has the structure shown in Fig. 6. One of the input bit sequence is used to generate the label sequence in A^L. The coset selector mapping is given by $m : \mathbb{Z}_2^2 \longrightarrow U(S)/U(S')$. The remaining bit sequences are used to index the signal point in the selected coset sequence.[13] The set of all signal sequences thus generated form the generalized coset code.

A coset [28] of $\mathbb{C}(S/S'; \mathcal{C})$ in S^I, can be represented as

$$\mathbb{C}\left(S/S'; \mathcal{C} + \boldsymbol{a}\right) = \bigcup_{\boldsymbol{c} \in \mathcal{C}} u_{\boldsymbol{a}}\left[u_{\boldsymbol{c}}\left((S')^I\right)\right] \tag{10}$$

for any $\boldsymbol{a} \in A^I$. There are $|A^I|/|\mathcal{C}|$ number of cosets of $\mathbb{C}(S/S'; \mathcal{C})$ in S^I. For the constructions of Sections III-C1 and III-C2, the source is first quantized[14] using S^I. For any given source quantized sequence $\boldsymbol{c} \in S^I$, we compute the index (characterized by some $\boldsymbol{a} \in A^I$) of the coset of $\mathbb{C}(S/S'; \mathcal{C})$ in S^I, containing $\boldsymbol{c}$ and send this to the decoder. The decoder finds this sequence $\boldsymbol{c}$ by decoding the side information in $\mathbb{C}(S/S'; \mathcal{C} + \boldsymbol{a})$. Using the result from [28] that $S^I/\mathbb{C}(S/S'; \mathcal{C})/(S')^I$ form a chain of geometrically uniform partition, and the cosets $\mathbb{C}(S/S'; \mathcal{C} + \boldsymbol{a})$ are geometrically uniform, *we note that the*

[11] A group whose elements are the isometries that transform one element of S to another.

[12] A^I refers to $|I|$-fold Cartesian product of A with itself [26].

[13] If we restrict ourselves to eight points in $\mathbb{Z} - 1/2$, only one of the remaining bit sequence is used. All other bit sequences are set to zero. This corresponds to $\mathbb{C}$ of Section III-C2, and $\mathbb{S}$ of Sections III-C3 and III-C4.

[14] In all our previous illustrations, we have considered finite points in the quantizer. By restricting the quantizer to finite points we incur the overload distortion [17], and the distance properties given by the analysis remains unaffected.

TABLE I
$\mathbb{S}/\mathbb{C}$ PARTITION FOR DIFFERENT CASES

Case	$\mathbb{S}/\mathbb{C}$ partition
Section III-C1	$S^I/(S')^I$
Section III-C2	$S^I/\mathbb{C}(S/S';\mathcal{C})$
Section III-C3	$\mathbb{C}(S/S';\mathcal{C})/(S')^I$
Section III-C4	$\mathbb{C}(S/S';\mathcal{C})/\mathbb{C}(S/S';\mathcal{C}')$

minimum distance of every coset of the trellis code built on uniform lattices is the same.

In Section III-C1, the trivial subgroup $\mathcal{C} = (e)^I$ is used, where e is the identity element of A, $I = \{1\}$, and $U(S)/U(S')$ is isomorphic to Z_2^R for a rate of transmission of R bits per sample. The channel code $\mathbb{C}$ corresponds to S' restricted to 2^{R_c} points, and the source code $\mathbb{S}$ is S restricted to 2^{R+R_c} points.

In Section III-C2, a nontrivial $\mathcal{C}$ with $|A^I|/|\mathcal{C}| = 2^{LR}$ for $I = \{1, 2, \ldots, L\}$ is used. If we restrict to 2^l points in every coset of S' in S, the channel code $\mathbb{C}$, given by $\mathbb{C}(S/S';\mathcal{C})$ will have $|\mathcal{C}|2^{Ll}$ sequences and the source code, given by $(S)^L$, will have $2^{L(l+\log|A|)}$ sequences.

In Section III-C3, the source is quantized using $\mathbb{C}(S/S';\mathcal{C})$ and $R = \frac{1}{L}\log|\mathcal{C}|$. If we restrict to 2^{R_c} points in every coset of S' in S, the source codebook $\mathbb{S}$, given by $\mathbb{C}(S/S';\mathcal{C})$ will have $2^{LR_c}|\mathcal{C}|$ sequences. The channel code $\mathbb{C}$, given by $(S')^L$, will have $2^{R_c L}$ sequences. For every sample of the source quantized sequence, R bits specify the index of the coset of S' (among a valid set permitted by $\mathcal{C}$) containing it.

Now consider a normal subgroup $\mathcal{C}'$ of $\mathcal{C}$. The generalized coset code $\mathbb{C}(S/S';\mathcal{C}')$ is a subset of $\mathbb{C}(S/S';\mathcal{C})$. A coset of $\mathbb{C}(S/S';\mathcal{C}')$ in $\mathbb{C}(S/S';\mathcal{C})$ can be defined as $\mathbb{C}(S/S';\mathcal{C}'+\boldsymbol{a})$ for any $\boldsymbol{a} \in \mathcal{C}$. Using the techniques of [28], it can be shown that

$$S^I/\mathbb{C}(S/S';\mathcal{C})/\mathbb{C}(S/S';\mathcal{C}')/(S')^I$$

forms a geometrically uniform partition, and the cosets of $\mathbb{C}(S/S';\mathcal{C}')$ in $\mathbb{C}(S/S';\mathcal{C})$ are geometrically uniform. Thus, the constructions of Section III-C4 can be interpreted as follows: the source is quantized (to say $\boldsymbol{c}'$) using $\mathbb{C}(S/S';\mathcal{C})$. The index (characterized by some $\boldsymbol{a} \in \mathcal{C}$) of the coset of $\mathbb{C}(S/S';\mathcal{C}')$ in $\mathbb{C}(S/S';\mathcal{C})$ containing $\boldsymbol{c}'$ is sent to the decoder. $\mathbb{S} = \mathbb{C}(S/S';\mathcal{C})$ and $\mathbb{C} = \mathbb{C}(S/S';\mathcal{C}')$, with S' being restricted to a finite number of points. The decoder recovers $\boldsymbol{c}'$ by decoding the side information in $\mathbb{C}(S/S';\mathcal{C}'+\boldsymbol{a})$. For example, with $(R_s, R_c) = (2,1)$ bits per sample, $A = \mathbb{Z}_2^3$, $\mathcal{C}$ is the set of codewords generated by the rate-$2/3$ convolutional code $(\boldsymbol{C}_s)$ as shown in Fig. 7(b). $\mathcal{C}'$ is the codeword set generated by the rate-$1/3$ convolutional code $\boldsymbol{C}_c$ (a concatenation $\boldsymbol{C}_1$ with $\boldsymbol{C}_s$). It can be verified that $\mathcal{C}'$ is indeed a normal subgroup of $\mathcal{C}$. The partitions for different cases are summarized in Table I.

It can be seen that for all the four cases of coset constructions, every coset in the given partition has the same distance properties resulting in the same probability of decoding error. For the systems based on nonuniform scalar quantizer, we study their performance using simulations. The constructions given in this section for the trellis codes can also be extended to other source and channel coding techniques such as lattice codes [20], multilevel codes [29], [30], and codes on graph [31].

IV. INFORMATION-THEORETIC RATE-DISTORTION BOUNDS

It appears that for a given rate of R bits per sample, there are a number solutions for R_s and R_c such that $R = R_s - R_c$. The issue of good selection of R_s and R_c is intimately connected to the correlation between X and Y. To understand this, we briefly give the information-theoretic rate region available in the literature [6], [7]. Consider the communication system as shown in Fig. 1(b), with the source, side information, the encoder, and the decoder defined as in Section III-A. In general, let the alphabets of the source, side information, and the reconstruction be given by $\mathcal{X}$, $\mathcal{Y}$, and $\hat{\mathcal{X}}$, respectively. The rate-distortion function for this setup $R^*(d)$ is given by

$$R^*(d) = \min_{p(w|x)} \min_f [I(X;W) - I(Y;W)] \tag{11}$$

where the minimization is carried out over all functions $f : \mathcal{Y} \times \mathcal{W} \to \hat{\mathcal{X}}$, and conditional distributions $p(w|x)$, such that $Y \to X \to W$ form a Markov chain, and $E(\rho(X, \hat{X})) \le d$. So if $Y = X + N$, where X and N are i.i.d. Gaussian with variances σ_x^2 and σ_n^2, and $\rho(x, \hat{x}) = (x - \hat{x})^2$, then

$$R^*(d) = R_{x|y}(d) = \begin{cases} \frac{1}{2}\log\frac{\sigma_x^2\sigma_n^2}{(\sigma_x^2+\sigma_n^2)d}, & 0 < d \le \frac{\sigma_x^2\sigma_n^2}{\sigma_x^2+\sigma_n^2} \\ 0, & d \ge \frac{\sigma_x^2\sigma_n^2}{\sigma_x^2+\sigma_n^2} \end{cases} \tag{12}$$

where $R_{x|y}(d)$ is the conditional rate-distortion function [2] when both encoder and decoder have access to Y.

Using the test channel of [8], it can be noted that the auxiliary random variable W is the quantized representation of X and

$$I(X;W) = \frac{1}{2}\log\left\{\frac{\sigma_x^2(\sigma_n^2 - d)}{\sigma_n^2 d}\right\}$$

and

$$I(W;Y) = \frac{1}{2}\log\left\{\frac{(\sigma_n^2 - d)(\sigma_x^2 + \sigma_n^2)}{\sigma_n^4}\right\}.$$

In [8], it was shown that a source codebook with $2^{L(I(X;W)+\epsilon)}$ codewords is partitioned into $2^{L(R^*(d)+2\epsilon)}$ cosets each containing $2^{L(I(W;Y)-\epsilon)}$ codewords where L is the block length, and $\epsilon > 0$ and close to zero. Let $R_{so} = I(X;W)$ and $R_{co} = I(W;Y)$ be the optimum achievable source and channel rates for the i.i.d Gaussian case. Let us denote the ratio σ_x^2/σ_n^2 as *correlation-SNR*. The mutual information $I(X;W)$ and $I(Y;W)$ are plotted in Fig. 9 for a fixed $R^*(d)$. When the correlation-SNR is small, R_{so} is close to $R^*(d)$ and R_{co} is close to 0, and for a large correlation-SNR both the rates R_{so} and R_{co} are significantly larger than $R^*(d)$.

Remark 3: Thus, even for the constructions studied in Section III-C, the optimal strategy in a large correlation-SNR regime would be to use a source code with a large source rate, which is partitioned into cosets of the channel code such that the effective rate of transmission is R bits per sample. When the correlation is weak, the number of codewords in the source codebook will be small. This intricate relationship between the source rate, the channel rate, and the correlation-SNR is also incorporated in the proposed constructive approach. This is given in Section V.

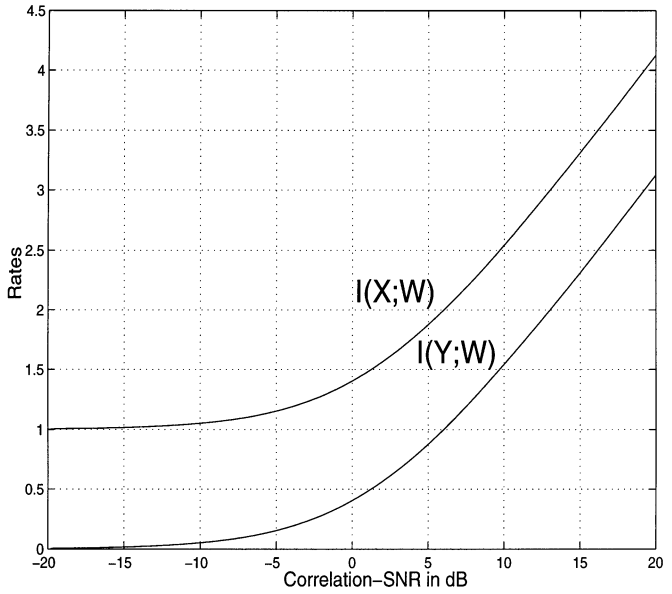


Fig. 9. Source and channel rates $I(X;W)$ and $I(W;Y)$ for $R^*(d) = 1$ bit/sample: at large correlation SNR, source and channel rates are significantly larger than 1.

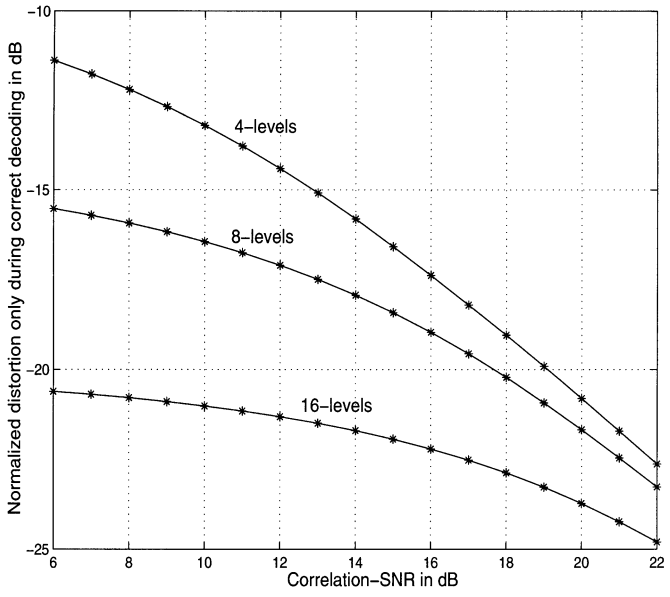
V. Design, Construction and Simulation Results

In this section, we study the design and construction of trellis codes in the DISCUS framework, and analyze the performance of our approach using simulations. We take the following model for X and Y: $Y = X + N$, where X is i.i.d. Gaussian with zero mean and unit variance and N is i.i.d. Gaussian with zero mean and independent of X. In all of our simulations, we use the squared Euclidean distance as the metric in decoding as we approximate the fictitious channel between the quantized codeword and the side information as a Gaussian channel.[15] Performance of various schemes are evaluated as a function of the variance of N. For every construction, we give 1) probability of decoding error with distortion during correct decoding, and 2) average excess distortion by absorbing the decoding error. We first consider the constructions for $R = 1$ bit/sample.

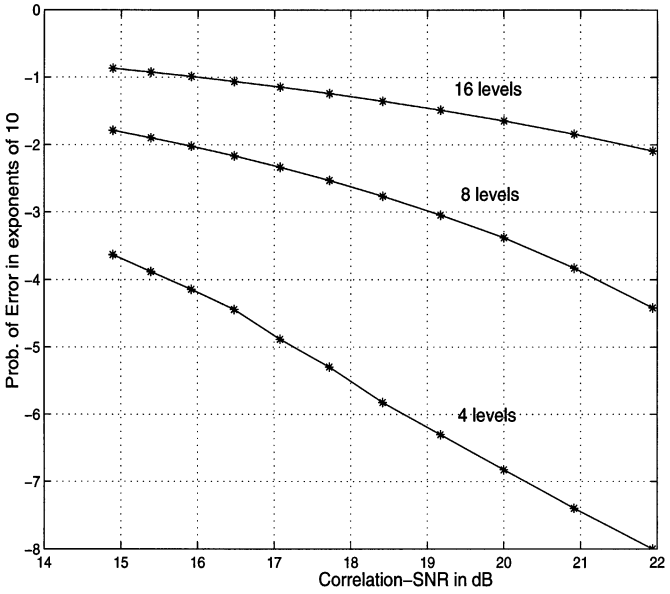
A. Unit Rate of Transmission

1) Memoryless Source Codes: Consider memoryless source and channel codes as given in Section III-C1. We use 4-, 8-, and 16-level scalar quantizers, each partitioned into two cosets, with each coset containing two, four, and eight codewords, respectively. Distortion during correct decoding only is plotted versus correlation-SNR (which is the ratio of the variance of X and N) for these three schemes in Fig. 10(a). Fig. 10(b) shows the probability of decoding error (P_e) for the same system. These are the results of Monte Carlo simulations. As can be noted from Fig. 10, there is a tradeoff between the distortion and probability of decoding error. For a given correlation-SNR, as the number of levels in the quantizer is increased, the distortion decreases and the probability of decoding error increases. Thus, the 4-level quantizer can tolerate more noise, while its distortion performance is poor. But the

[15]Note that in the test channel given in [8], the ideal channel between the quantized representation W and the side information Y is indeed Gaussian.



(a)



(b)

Fig. 10. (a) Distortion performance of different quantizers during correct decoding for a Gaussian source. (b) Probability of error for $R = 1$ bit/sample for memoryless quantization and memoryless coset construction.

16-level quantizer has very good distortion performance, at the same time operating with an unacceptable probability of error. Typically, any $P_e \leq 10^{-4}$ can be termed as acceptable. Ideally, for a given rate of transmission, we would like to use a source code with a large number of levels (to get lower distortion), but the probability of error puts a limitation on this. Under this constraint, as the correlation-SNR is increased, the source rate R_s that can be supported also increases.

Next consider the trellis-based coset construction of Section III-C2. We use 4- and 8-state trellis built on a 4-level scalar quantizer as shown in Fig. 11(a). These trellises are designed based on the set partitioning rules of [25], [26]. Fig. 12(a) gives the probability of decoding error versus correlation-SNR. Note that by using trellis-based cosets, we get gains of around 3–4 dB

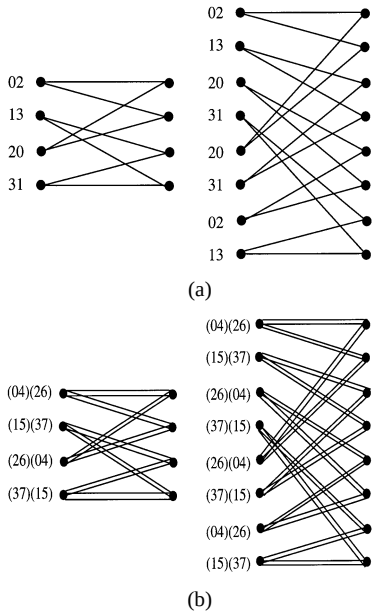


Fig. 11. Trellises for the channel code $\mathbb{C}$ for $R = 1$ bit/sample using memoryless source codes (a) for 4-level quantizer; (b) for 8-level quantizer. Both (a) and (b) have the same structure for a given number of states.

in correlation-SNR over memoryless coset construction. Thus, without increasing the rate, at $P_e \leq 10^{-4}$, we can operate at correlation-SNRs no less than 12 dB (see Fig. 12(a)) compared to 15.5 dB (see Fig. 10(b)). Note that for a given source code, the distortion during correct decoding is independent of the coset construction (in terms of memory versus memoryless). Similar coset constructions are done on an 8-level quantizer using 4-, 8-state trellises for $R = 1$ bit/sample as shown Fig. 11(b). Fig. 12(b) gives the performance in terms of P_e. Here again we get 3-dB gain over memoryless coset construction with four and eight state trellis at $P_e \leq 10^{-4}$.

Remark 4: The trellis codes given in [25], [26], which are built on scalar alphabets use the group structure of the partition $\mathbb{Z}/4\mathbb{Z}$. Such rate-n bits per sample trellis codes built on 2^{n+1}-level alphabet for all $n = 1, 2, 3, \ldots$, for a given fixed number of states, use an underlying rate-$1/2$ convolutional code only on the least significant bit of the input, and the rest of the bits use the identity mapping to get the rate-$n/(n+1)$ convolutional code. The algebraic structure of the trellis for different n remains the same except with differing number of parallel transitions (see Fig. 11(a) and (b) for $n = 1, 2$). All convolutional codes used in this paper have this structure.

For a fixed probability of decoding error (say $\leq 10^{-4}$), when the correlation-SNR is between 12 and 18 dB, the only choice is the 4-level scalar quantizer which gives a distortion performance of up to -19 dB. For correlation-SNR from 18 to 24 dB, the 8-level scalar quantizer gives the smallest distortion, ranging from -19.9 to -25 dB. For a correlation-SNR larger than 24 dB, the 16-level quantizer is used (not shown here due to lack of space). Note that as the correlation-SNR is increased, the source rate R_s that can be supported also increases, as prescribed by information theory (see Section IV).

Performance with Average Excess Distortion: Now consider the case when the probability of error is absorbed into the average excess distortion. As we have designed the codes to minimize the probability of decoding error, its performance during error in decoding cannot be guaranteed. It is similar to the concept of conventional channel coding, where one can guarantee a given performance during correct decoding, and no guarantees are made when decoding error occurs.[16] The average distortions as a function of correlation-SNR are given in Fig. 13 for $R = 1$ bits/sample for memoryless quantization, along with the Wyner–Ziv bound. The average distortion has two components: 1) MSE in estimation during correct decoding, 2) a significantly larger MSE in estimation in the wrongly decoded Voronoi region during decoding error. It can be observed from Fig. 12 that in the very low correlation-SNR regime, the probability of decoding error of the trellis-based coset construction exceeds that of the memoryless coset construction.[17] In this region, due to a large probability of decoding error of the trellis-based coset construction (larger than that of the memoryless coset-construction), the MSE with incorrect decoding is a dominant factor in the average MSE. This makes the average distortion of trellis-based coset construction to exceed that of memoryless coset construction.

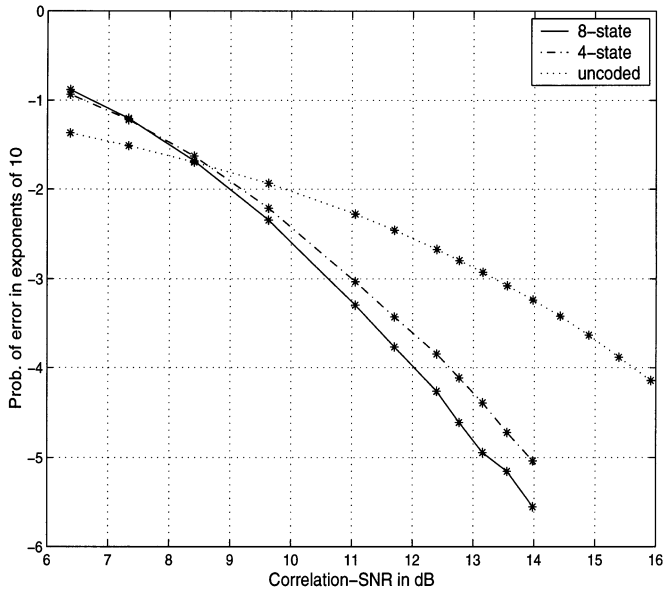
The difference in average distortion between these two systems reduces to zero as the correlation-SNR is increased. After this point, this difference changes its sign and reaches a maximum. In this interesting region, there a right mix of the MSE with incorrect decoding and the MSE with correct decoding in the average distortion, where the coding gain of the trellis-based coset construction is most effective. This difference asymptotically goes to zero for very large correlation-SNR. This is because, in this range, 1) the probability of error of both types of coset constructions are very small and comparable, 2) the contribution of the MSE with incorrect decoding in the average distortion is very small, and 3) distortion during correct decoding is the same in both types of coset constructions, which together result in almost equal average distortions.

Thus, as we reduce the correlation-SNR from a sufficiently large value, the rate of increase of the average distortion of the coded system (while remaining lower than the uncoded system) comes down (plateau effect) and reaches the minimum and then starts increasing further. This effect becomes more pronounced as we put more complexity into the trellis-based coset constructions. For $R = 1$, we are 3.3 and 3.4 dB from the Wyner–Ziv bound at correlation-SNR of 9 and 15 dB, respectively, with gains of 1.2 dB over the uncoded coset construction.

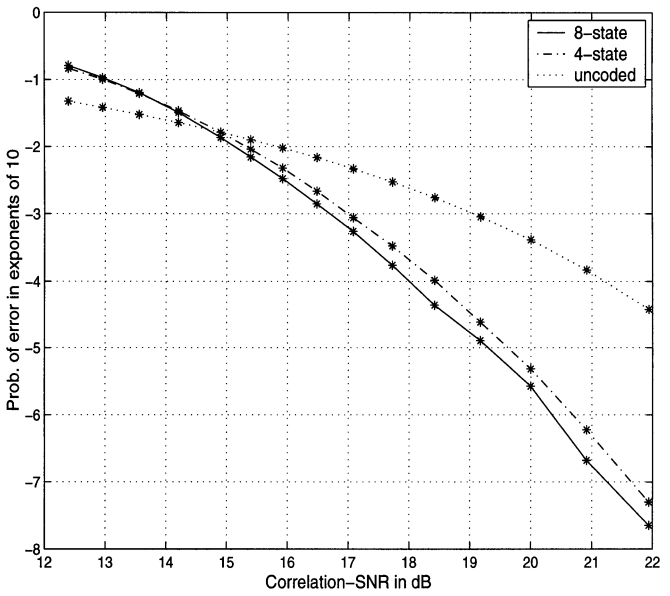
2) TCSQ as a Source Code: We now construct trellis-based quantizers and coset construction studied in Section III-C4. Consider a TCSQ of rate R_s bits per sample built on a 2^{R_s+1}-levelquantizer, with a trellis having 2^{R_s-1} parallel transitions (see Remark 4). The generator matrix of its convolutional code (denoted by $\boldsymbol{C}_s$) can be written as $\boldsymbol{G}_s(\boldsymbol{D}) = [\boldsymbol{I}_{R_s}|\boldsymbol{A}]$, where $\boldsymbol{I}_{R_s}$ is the $R_s \times R_s$ identity matrix, $\boldsymbol{A}^T = [0, 0, \ldots, 0, g(D)]$, and $g(D)$ is a polynomial. For $R = 1$ bit/sample, the objective is to construct a coset partition of the

[16]It is possible to construct these cosets based on some algebraic structure, where error detection can be accomplished. This is beyond the scope of this paper.

[17]Note that in this case, both the systems are operating at a very high ($\geq 10^{-2}$) probability of decoding error.

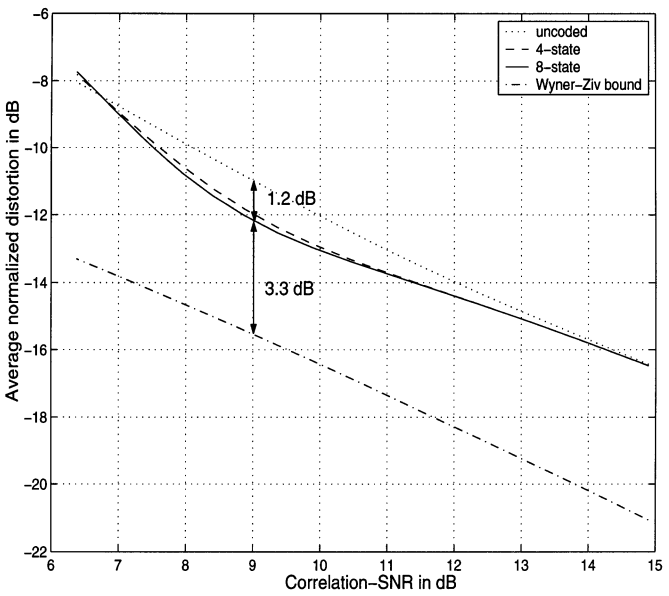


(a)

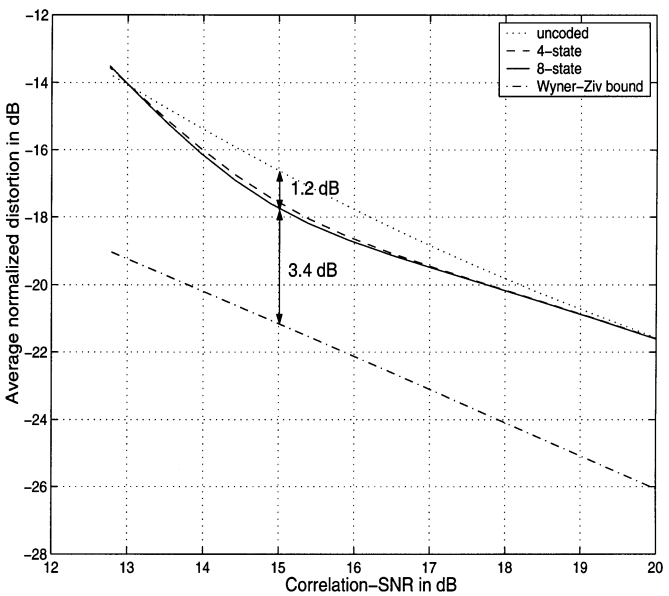


(b)

Fig. 12. Probability of error for $R = 1$ bit/sample for memoryless quantization and trellis-based coset construction: (a) $R_s = 2$ bits/sample, $\mathbb{S}$ is 4-level quantizer; (b) $R_s = 3$ bits/sample, $\mathbb{S}$ is 8-level quantizer.



(a)



(b)

Fig. 13. Average normalized distortion for $R = 1$ bit/sample, memoryless quantization and trellis-based coset construction: (a) $\mathbb{S}$ is 4-level quantizer, (b) $\mathbb{S}$ is 8-level quantizer.

sequences of this TCSQ, with each coset having the structure of a trellis code with rate $(R_s - 1)$ bits per sample, built on the same alphabet. As given in Section III-C4, we use a concatenation of a convolutional code $\boldsymbol{C}_1$ with $\boldsymbol{C}_s$. We construct such a trellis code (for the channel code $\mathbb{C}$) using a generator matrix having the following structure: $\boldsymbol{G}_c(\boldsymbol{D}) = [\boldsymbol{I}_{R_s-1} | \boldsymbol{A}_1 | \boldsymbol{A}_2]$, where $\boldsymbol{A}_1{}^T = [0, 0, \ldots, 0, g(D)]$ and $\boldsymbol{A}_2{}^T = [0, 0, \ldots, 0, g^2(D)]$. The intuition behind this choice is that the code $\boldsymbol{C}_1$ (with $\boldsymbol{G}_1(\boldsymbol{D}) = [\boldsymbol{I}_{R_s-1} | \boldsymbol{A}_1]$) obtains an efficient partition of the space of $(R_s L)$-bit sequences, which the TCSQ takes as the input (for block length L). If the trellis of $\mathbb{S}$ has η states, then the trellis of $\mathbb{C}$ has η^2 states. We construct such TCSQs and coset partitions on 8- and 16-level quantizers. The trellis and the finite state machines for the source and the channel codes for the 8-level quantizer are shown in Fig. 14. Recall that the encoder uses the trellis of Fig. 14(a) in quantization, while the receiver uses the trellis of Fig. 14(b) in decoding the side information. The probability of decoding error as a function of correlation-SNR is plotted in Fig. 15(a) and (b) with the number states in $g(D)$ ranging from 2 to 32. The performance of these systems is better than that of the corresponding constructions with memoryless cosets. Using these methods we can now operate at correlation-SNRs as low as 8.5 dB with 32 states in $g(D)$ with $P_e \leq 10^{-4}$.

Performance With Average Excess Distortion: After the recovery of the active codeword sequence, the next objective is to get the optimal estimate of the source X. Finding this estimate

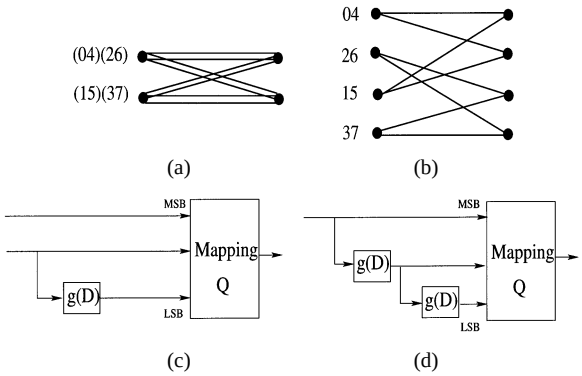


Fig. 14. Code structure for $R = 1$ bit/sample for the constructions of Section III-C4 on an 8-level quantizer. (a) Source code $\mathbb{S}$: rate-2 trellis code. (b) Channel code $\mathbb{C}$: rate-1 trellis code. (c) Finite-state machine of $\mathbb{S}$. (d) Finite-state machine of $\mathbb{C}$.

is computationally intractable. So we provide the following suboptimal approximation inspired from the information-theoretic analysis (see Section IV). Let $\boldsymbol{W}$ denote the quantized version of the source sequence $\boldsymbol{X}$.

1) We model the quantization noise given by $\boldsymbol{Q} = \boldsymbol{X} - \boldsymbol{W}$ to be memoryless zero-mean Gaussian noise with variance equal to the MSE in quantization and independent of $\boldsymbol{W}$.
2) We assume that for every $1 \leq k \leq L$, $Y_k \rightarrow X_k \rightarrow W_k$ form a Markov chain. This is based on the test channel interpretation of the Wyner–Ziv's characterization of the rate-distortion function $R^*(d)$.

It was shown in [7], [8] that in the limit of large block length, a test channel that achieves the rate-distortion bound would satisfy the above assumptions. With these approximations, and using the fact that $Y = X + N$, the suboptimal estimate of the source is given by a convex combination [7], [8] of $\boldsymbol{W}$ and $\boldsymbol{Y}$

$$\hat{\boldsymbol{X}} = \frac{\sigma_q^2}{\sigma_q^2 + \sigma_n^2}\boldsymbol{Y} + \frac{\sigma_n^2}{\sigma_q^2 + \sigma_n^2}\boldsymbol{W} \tag{13}$$

where

$$\sigma_q^2 = \frac{1}{L}\sum_{k=1}^{L}(X_k - W_k)^2.$$

The average normalized distortion after this reconstruction is plotted in Fig. 16 when $\mathbb{S}$ is TCSQ built on 8- and 16-level quantizers. We observe a behavior similar to that of the constructions with memoryless source codebook. At correlation-SNR of 7.1 and 13.56 dB, this system is 2.04 and 1.97 dB, respectively, from the Wyner–Ziv bound. The constructions with source codebooks $\mathbb{S}$ having memory perform better than those without memory. Note for $R = 1$, at correlation-SNR of 12.22 dB, systems with $R_s = 2$ and $R_s = 3$ give the same average distortion of -14.48 dB. The former gets lower distortion to the left of this value, and the latter does better to the right of it.

B. Higher Rates of Transmission

1) Memoryless Source Codes: Let us first consider the constructions of Section III-C2, for $R = 2$ bits/sample. For a trellis

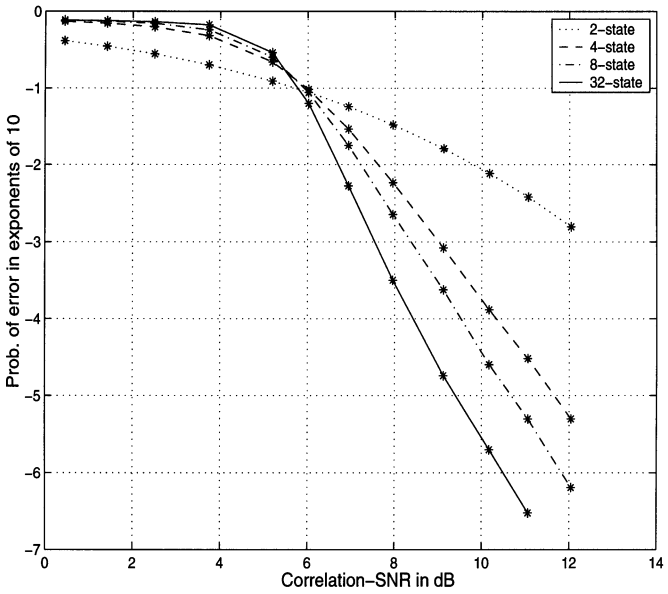
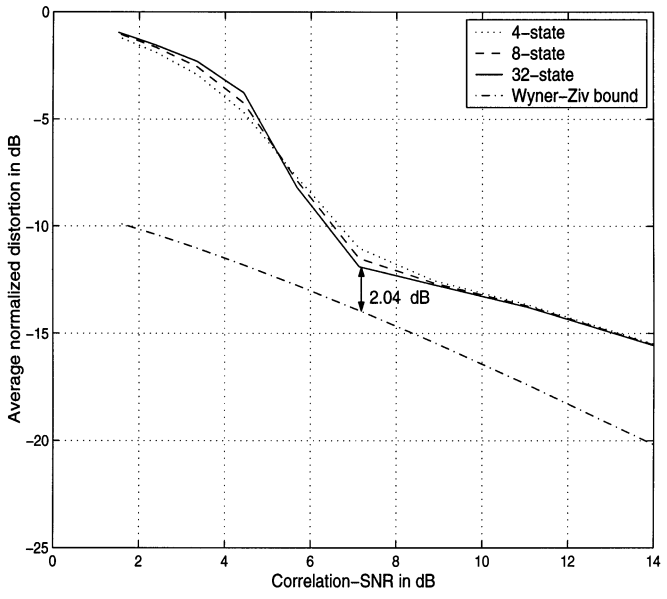
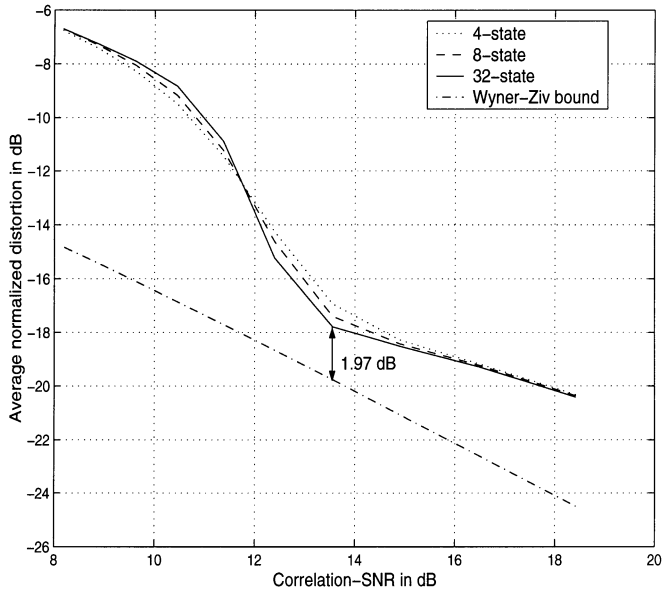


Fig. 15. Probability of error for $R = 1$ bit/sample, trellis-based quantization and coset construction. (a) $R_s = 2$ bits/sample, $\mathbb{S}$ is TCSQ based on the 8-level quantizer. (b) $R_s = 3$ bits/sample, $\mathbb{S}$ is TCSQ based on the 16-level quantizer.

code (characterizing $\mathbb{C}$with $R_c = 1$) built on an 8-level quantizer (characterizing $\mathbb{S}$ with $R_s = 3$), we need a rate-$1/3$ convolutional code, and a mapping rule Q to maximize the coding gain. In principle, for this trellis code, the set partition rule works by increasing the alphabet size four times, as compared to two times in most of the trellis designs available in the literature. We need either to develop a new code design or to use a computer-based search for getting "good" codes. Instead, with the goal of reducing the trellis complexity, we take a suboptimal approach by making the following observation. It was pointed out in the context of trellis-coded modulation [25], using the concepts of alphabet-constrained channel capacity theorem, that by doubling the alphabet size most of the attainable coding gains can be realized. Increasing the alphabet size



(a)



(b)

Fig. 16. Average normalized distortion for $R = 1$ bit/sample, trellis-based quantization and coset construction. (a) $R_s = 2$ bits/sample, $\mathbb{S}$ is TCSQ based on 8-level quantizer. (b) $R_s = 3$ bits/sample, $\mathbb{S}$ is TCSQ based on 16-level quantizer.

(and keeping the average energy of the constellation the same) beyond this typically yields only marginal gains [25]. Similar observations were also made in the context of trellis-coded quantization in [24]. We infer from these two related problems that the minimum distance of the trellis codes will increase only marginally after doubling the alphabet size.

We, therefore, choose a subset (containing four elements) of the available alphabet (eight levels), and use only them in one coset. We partition the 8-level quantizer into two cosets (as in memoryless coset construction for $R = 1$), and construct trellis codes on each of these memoryless cosets. Let Coset-1 and Coset-2 specify the codeword sets $\{0, 2, 4, 6\}$ and $\{1, 3, 5, 7\}$, respectively. A rate-$1/2$ convolutional code with set partitioning

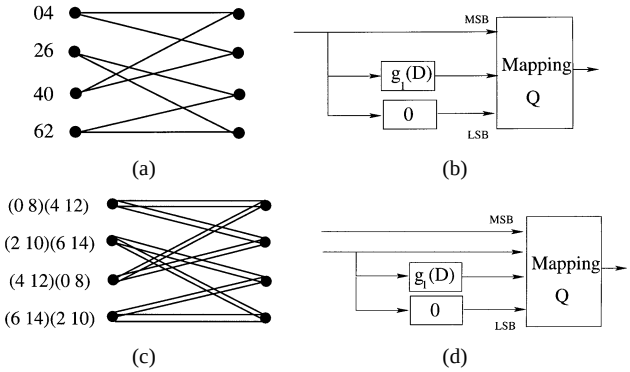


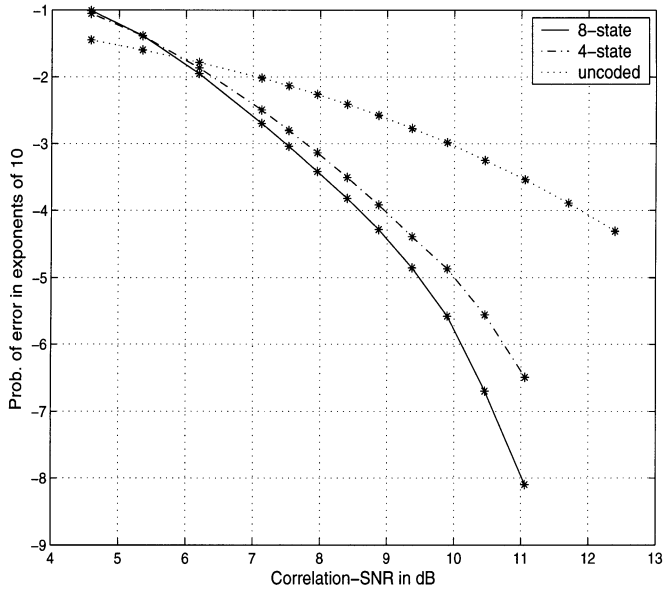
Fig. 17. Codes obtained using the same convolutional code for memoryless quantization and trellis-based coset construction for $R = 2$ bits/sample. (a) 4-state trellis (of $\mathbb{C}$) on the 8-level quantizer. (b) Finite-state machine of $\mathbb{C}$ built on the 8-level quantizer. (c) 4-state trellis (of $\mathbb{C}$) on the 16-level quantizer. (d) Finite-state machine of $\mathbb{C}$ built on the 16-level quantizer.

on Coset-1 can be used to get trellis codes with a 4-state trellis as shown in Fig. 17(a). The corresponding finite-state machine is shown in Fig. 17(b). This method essentially "grounds" the least significant bit to zero, and for a general R_c, the convolutional codes for such a construction of $\mathbb{C}$ will have a generator matrix of the form $\boldsymbol{G}(\boldsymbol{D}) = [\boldsymbol{I}_{R_c}|\boldsymbol{A}_1|\boldsymbol{A}_2]$, where $\boldsymbol{A}_1{}^T = [0, 0, \ldots, 0, g(D)]$, $\boldsymbol{A}_2{}^T$ is an all-zero vector, and the trellis uses an alphabet of size 2^{R_c+2}. There are four trellis cosets, with two of them built on Coset-1 and two built on Coset-2. The index of one of the four trellis cosets containing the quantized source sample is sent to the decoder. Fig. 18(a) shows the decoding error performance of this construction. Similar to the case of $R = 1$, we get 3–4-dB gains over memoryless coset construction when $P_e \leq 10^{-4}$. As R is increased from 1 to 2 bits/sample, for the same trellis complexity and scalar quantizer, the probability of error is decreased, and the correlation-SNR that can be supported is decreased from 18 to 8.5 dB.

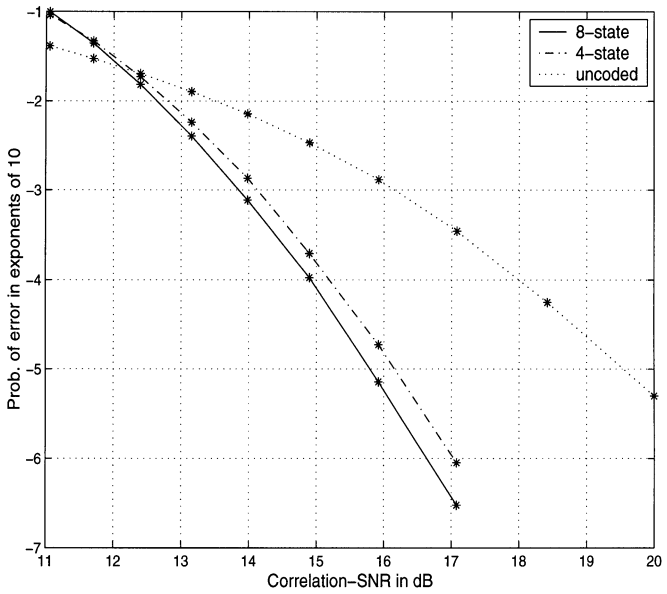
Similar codes are constructed on the 16-level quantizer using a rate-$2/3$ (instead of rate-$2/4$) convolutional code with memoryless partitioning of the alphabet into two cosets as shown in Fig. 17(c) and (d). The performance in terms of probability of decoding error is shown in Fig. 18(b). The operating range of the 16-level quantizer has been increased (from a correlation-SNR of $\geq$24 dB when $R = 1$ bit/sample) to $\geq$15 dB when $R = 2$ bits/sample corresponding to $P_e \leq 10^{-4}$. Note that for a given number of states, all the trellises in Figs. 17 and 11 have the same underlying structure.

From these performance plots, the operational rate-distortion function (only during correct decoding) for the case of memoryless $\mathbb{S}$ can be computed as follows: for a given correlation-SNR, rate, and probability of error, choose the largest quantizer which satisfy these criteria and read out the distortion from these plots. For example, with $P_e \leq 10^{-4}$ for a correlation-SNR of 15 dB, and $R = 1$ bit/sample, the 4-level quantizer is the solution giving distortion of -16.5 dB, and when $R = 2$ bits/sample, the solution is 16-level quantizer giving a distortion of -21.8 dB.

Remark 5: Similar constructions for a general R would involve a quantizer with 2^{R_s} levels $(R_s > R)$, which is partitioned into 2^{R-1} memoryless cosets. A trellis code is built on one of the memoryless cosets with a convolutional code of rate

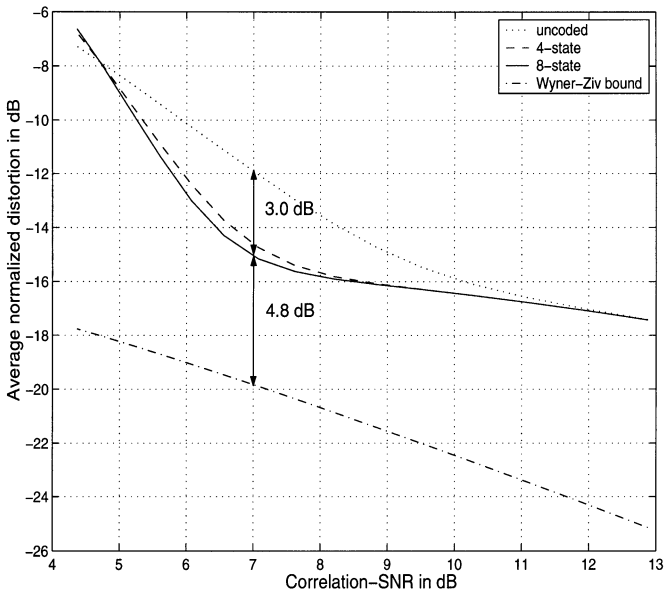


(a)

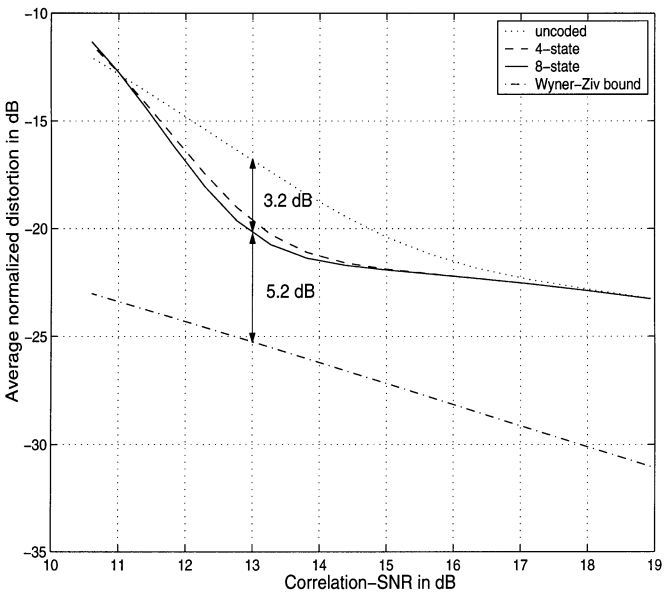


(b)

Fig. 18. Probability of error for $R = 2$ bits/sample, memoryless quantization, and trellis-based coset construction. (a) $R_s = 3$ bits/sample, $\mathbb{S}$ is 8-level quantizer. (b) $R_s = 4$ bits/sample, $\mathbb{S}$ is 16-level quantizer.



(a)



(b)

Fig. 19. Average normalized distortion for $R = 2$ bits/sample, memoryless quantization and trellis-based coset construction: (a) $\mathbb{S}$ is 8-level quantizer, (b) $\mathbb{S}$ is 16-level quantizer.

$(R_s - R)/(R_s - R + 1)$. Essentially, in encoding, the $(R-1)$ least significant bits of the quantized source are sent as is, the two next significant bits on which the rate-$1/2$ convolutional code is built (see Remark 4) are syndrome-encoded to a single bit and sent, and the rest of the $(R_s - R - 1)$ most significant bits are discarded.

Performance with Average Excess Distortion: The performance of the above system in terms of average distortion is shown in Fig. 19 along with the Wyner–Ziv bound. Recall that in this case we can numerically compute the optimal estimate of the source as $\mathbb{S}$ is memoryless. The gains over the uncoded coset construction are increased to 3.0 and 3.2 dB for $R = 2$ at correlation-SNR of 7 and 13 dB, respectively, while the deviations from the Wyner–Ziv bound are also increased to 4.8 and 5.2 dB. Note that as R increases, the deviations from the Wyner–Ziv bound also increase. This can be due to suboptimal trellis-based coset constructions for $R > 1$.

2) TCSQ as Source Code: Now let us graduate to the case where the source code is a TCSQ with rate $(R_c + 2)$ bits per sample (for $R = 2$), built on an alphabet of size 2^{R_c+3} for a given channel rate of R_c bits per sample. Our objective is to partition the space of $2^{(R_c+2)L}$ sequences of this TCSQ into 2^{2L} cosets. The trellis structure of each of these cosets uses a rate-$R_c/(R_c+3)$ convolutional code. Note that the set partition rule works by increasing the alphabet size eight times, as compared to two times in the TCSQ. We construct such $\mathbb{C}$ based on a generator matrix having the form: $\boldsymbol{G}_c(\boldsymbol{D}) = [\boldsymbol{I}_{R_c}|\boldsymbol{A}_1|\boldsymbol{A}_2]$ where $\boldsymbol{A}_1{}^T = [0, 0, \ldots, 0, g(D)]$, $\boldsymbol{A}_2$ is the all-zero matrix of

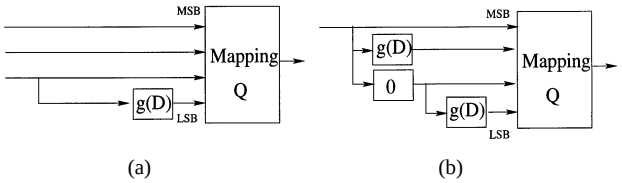
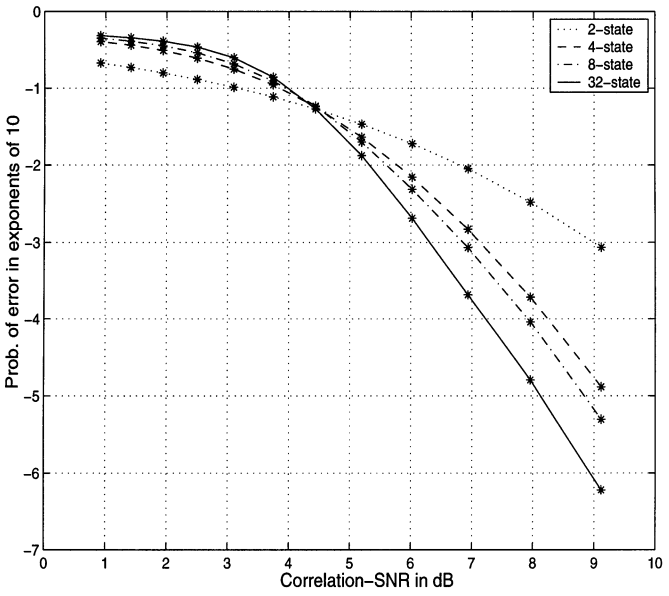


Fig. 20. Finite-state machines for $R = 2$ bits/sample, based on a 16-level quantizer for trellis-based quantization and coset construction. (a) Source code with $R_s = 3$ bits/sample. (b) Channel code with $R_c = 1$ bit/sample.

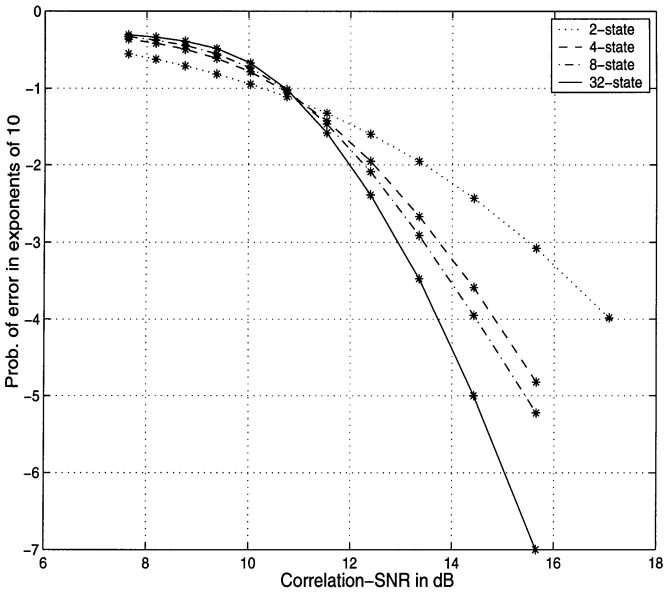
dimension $R_c \times 2$. As pointed out in Section III-C4, this is obtained by a concatenation of the rate-$(R_c+2)/(R_c+3)$ convolutional code $\boldsymbol{C}_s$ (of TCSQ) and a rate-$R_c/(R_c+2)$ convolutional code $\boldsymbol{C}_1$ with $\boldsymbol{G}_1(\boldsymbol{D}) = [\boldsymbol{I}_{R_c}|\boldsymbol{A}_1|\boldsymbol{A}_2]$, where ${\boldsymbol{A}_1}^T = [0, 0, \ldots, 0, g(D)]$ and $\boldsymbol{A}_2$ is the all-zero vector. The intuition behind this choice is that $\boldsymbol{C}_1$ obtains an efficient partition of the space of $(R_c+2)L$-bit sequences, which the TCSQ takes as the input. This is illustrated in Fig. 20 for $R_c = 1$ bit/sample with the TCSQ built on a 16-level quantizer. The performance of such partitions based on the 16- and 32-level scalar quantizers are plotted in Fig. 21(a) and (b), respectively, for trellises with states in $g(D)$ ranging from 2 to 32.

Remark 6: For a general R, a rate-R_s TCSQ (for some $R_s > R$) is constructed using a rate-(R_s/R_s+1) convolutional code $\boldsymbol{C}_s$ to give $\mathbb{S}$, and the channel code $\mathbb{C}$ is given by a trellis code obtained by a concatenation of rate-R_c/R_s convolutional code $\boldsymbol{C}_1$ (with $\boldsymbol{G}_1(\boldsymbol{D}) = [\boldsymbol{I}_{R_c}|\boldsymbol{A}_1|\boldsymbol{A}_2]$, where ${\boldsymbol{A}_1}^T = [0, \ldots, g(D)]$ and $\boldsymbol{A}_2$ is the all-zero matrix of dimension $R_c \times R-1$) with $\boldsymbol{C}_s$. The $(R-1)$ least significant bits of the output of $\boldsymbol{C}_1$ are all zero. Thus, in encoding, out of the R_s-bit representation of TCSQ codewords, $(R-1)$ least significant bits are sent as is, the next two bits are syndrome-encoded into a single bit and sent, and the rest of the $(R_s - R - 1)$ most significant bits are discarded.

Performance with average excess distortion: The average normalized distortions of the above system are plotted in Fig. 22 along with the Wyner–Ziv bound. Similar to the case of $R = 1$ bit/sample, we use a convex combination of the quantized codeword and the side information as an estimate of the source for reconstruction. At correlation-SNR of 6.56 and 13.07 dB, the system operates at 3.12 and 3.17 dB, respectively, from the Wyner–Ziv bound. This also shows that for the same R and correlation-SNR, constructions when $\mathbb{S}$ is TCSQ give gains over that when $\mathbb{S}$ is memoryless. As in the case of memoryless quantization, the deviations from the Wyner–Ziv bound increase as R is increased. Note for $R = 2$, at correlation-SNR of 11.58 dB, the systems with $R_s = 3$ and $R_s = 4$ give the same average distortion of -17.42 dB, with the former giving better distortion to the left of this value, and the latter performing better to the right of it. Comparing Figs. 16 and 22 (constructions where $\mathbb{S}$ is TCSQ) observe that 1) for a fixed correlation-SNR, the average distortion is decreasing in R, and 2) for a fixed R, as correlation-SNR is increased, the system with larger R_s performs better. Similar behavior can also be observed for those constructions with $\mathbb{S}$ memoryless as seen from Figs. 13 and 19. The gap in average distortion between the proposed constructions and the Wyner–Ziv bound can be reduced using more efficient quantization, and coset constructions based on the algebraic structure of capacity-achieving multilevel codes [29].



(a)

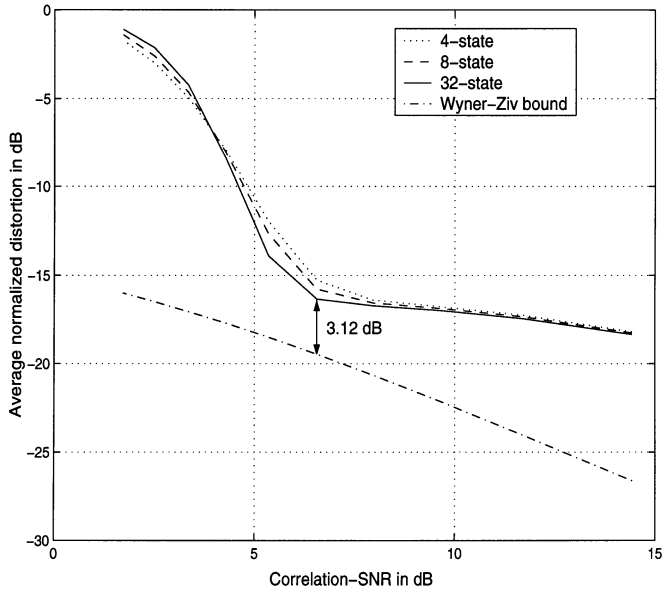


(b)

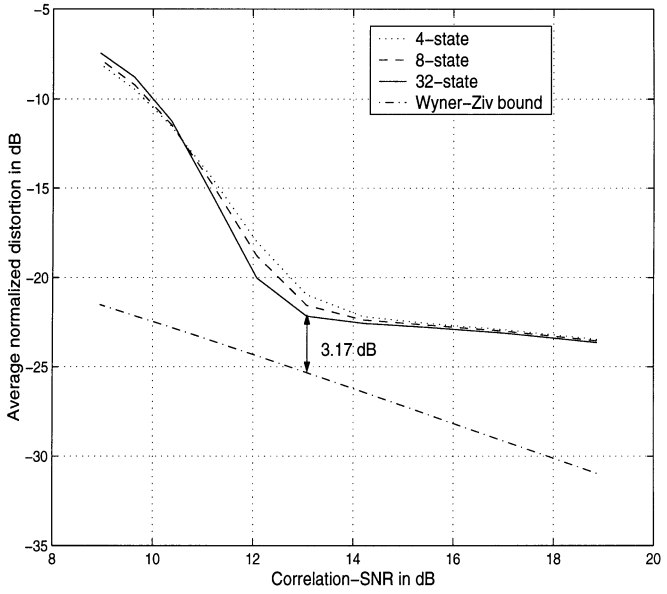
Fig. 21. Probability of error for $R = 2$ bits/sample, trellis-based quantization, and coset construction. (a) $R_s = 3$ bits/sample, $\mathbb{S}$ is TCSQ based on 16-level quantizer. (b) $R_s = 4$ bits/sample, $\mathbb{S}$ is TCSQ based on 32-level quantizer.

VI. Conclusion and Future Work

In this paper, we have presented a constructive approach for source coding in the presence of side information at the decoder. This approach is built on algebraic channel coding ideas and yields promising results. The basic concept on which these systems are built is the following: the source is first quantized using a source codebook designed for the marginal probability density function of the source. The source codebook is partitioned into cosets of a channel code designed for the fictitious channel between the side information and the quantized source.



(a)



(b)

Fig. 22. Average normalized distortion for $R = 2$ bits/sample, trellis-based quantization, and coset construction. (a) $R_s = 3$ bits/sample, $\mathbb{S}$ is TCSQ based on 16-level quantizer. (b) $R_s = 4$ bits/sample, $\mathbb{S}$ is TCSQ based on 32-level quantizer.

The encoder, instead of sending the index of the quantized codeword, sends only the index of the coset containing the quantized codeword. The decoder recovers the quantized codeword by decoding the side information in the given coset of the channel code. The complex interplay between the source coding, the channel coding, and the estimation component is quantified and analyzed. We first considered a construction where both the source code and the channel code are memoryless and graduated to the case when both have memory. Design methods for such codes are also discussed with their performance evaluated using simulations. These systems are very attractive in applications such as sensor array networks.

We have provided computationally efficient algorithms for some cases and have considered integer rates of transmission for simplicity. For noninteger rates of transmission, one can use quantizers and trellis codes in higher dimensional Euclidean spaces, and this requires a more detailed study. We have used quantizers based on the marginal distribution of the source for our approach. The optimal choice of quantizers is an issue that requires more scrutiny. The constructions based on trellis codes presented here can be generalized to any source and channel coding techniques such as lattice codes and multilevel codes [29], [30]. Note that the decoding rule employed here in recovering the active codeword in the given coset is based on the squared Euclidean distance between the side information and the codeword in the coset. This is an approximation which needs to be improved.

In this paper, we have addressed the case where X and Y are jointly Gaussian. When Y is a noisy version of X with the noise being Gaussian, the channel code construction which maximizes the minimum Euclidean distance works well. The extensions of these concepts for the cases where X and Y are arbitrarily correlated signals, and the correlation structure is partially known, need to be explored, and this is part of ongoing and future work. Regarding applications to sensor networks, the generalizations and modifications of the concepts presented here for building real systems are still far from over. Some of the promising approaches inspired by the concepts of distributed source coding have been proposed in [32].

ACKNOWLEDGMENT

The authors wish to thank anonymous reviewers for their critical comments on an earlier version of this manuscript that led to a much improved version.

REFERENCES

[1] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Mobile networking for smart dust," in *ACM/IEEE Int. Conf. Mobile Computing and Networking*, Seattle, WA, Aug. 1999.

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

[3] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 471–480, July 1973.

[4] S. Verdú, "Fifty years of Shannon theory," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2057–2078, Oct. 1998.

[5] A. D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 2–10, Jan. 1974.

[6] ——, "On source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 294–300, May 1975.

[7] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 1–10, Jan. 1976.

[8] A. D. Wyner, "The rate-distortion function for source coding with side information at the decoder-II: General sources," *Inform. Contr.*, vol. 38, pp. 60–80, 1978.

[9] T. M. Cover, "A proof of data compression theorem of Slepian and Wolf for ergodic sources," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 226–228, Mar. 1975.

[10] R. F. Ahlswede and J. Korner, "Source coding with side information and a converse for degraded broadcast channels," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 629–637, Nov. 1975.

[11] A. H. Kaspi and T. Berger, "Rate-distortion for correlated sources with partially separated encoders," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 828–840, Nov. 1982.

[12] T. Berger, "Multiterminal source coding," in *Information Theory Approach to Communications (CISM Courses and Lectures)*, G. Longo, Ed. Vienna/New York: Springer-Verlag, 1977, vol. 229.
[13] T. S. Han and K. Kobayashi, "A unified achievable rate region for a general class of multiterminal source coding systems," *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 277–288, May 1980.
[14] I. Csiszár and J. Körner, "Toward a general theory of source networks," *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 155–165, Mar. 1980.
[15] R. Zamir and S. Shamai , "Nested linear/ lattice codes for Wyner-Ziv encoding," presented at the IEEE Information Theory Workshop, Killarney, Ireland, 1998.
[16] T. J. Flynn and R. M. Gray, "Encoding of correlated observations," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 773–787, Nov. 1987.
[17] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA: Kluwer, 1992.
[18] S. Shamai (Shitz), S. Verdú, and R. Zamir, "Systematic lossy source/channel coding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 564–579, Mar. 1998.
[19] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes* Amsterdam, The Netherlands, 1977.
[20] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*. New York: Springer-Verlag, 1988.
[21] G. D. Forney, "The viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, Mar. 1973.
[22] R. Zamir and M. Feder, "On lattice quantization noise," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1152–1159, July 1996.
[23] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1995.
[24] M. W. Marcellin and T. R. Fischer, "Trellis coded quantization of memoryless and Gauss-Markov sources," *IEEE Trans. Commun.*, vol. 38, pp. 82–93, Jan. 1990.
[25] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 55–67, Jan. 1982.
[26] G. D. Forney, "Coset codes- part 1: Introduction and geometrical classification," *IEEE Trans. Inform. Theory*, vol. 34, pp. 1123–1151, Sept. 1988.
[27] M. A. Armstrong, *Groups and Symmetry*. New York: Springer-Verlag, 1988.
[28] G. D. Forney, "Geometrically uniform codes," *IEEE Trans. Inform. Theory*, vol. 37, pp. 1241–1260, Sept. 1991.
[29] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: Theoretical concepts and practical design rules," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1361–1391, July 1999.
[30] R. Garello and S. Benedetto, "Multilevel construction of block and trellis group codes," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1257–1264, Sept. 1995.
[31] D. J. C. MacKay, "Good error-correcting codes based on very sparce matrices," *IEEE Trans. Inform. Theory*, vol. 45, pp. 399–431, Mar. 1999.
[32] L. Doherty, "Algorithms for Position and Data Recovery in Wireless Sensor Networks," M.S. thesis, Dept. Elec. Eng. Comput. Sci., Univ. Calif., Berkeley, 2000.