Multiple Description Coding: Compression Meets the Network

Vivek K Goyal

Your paper on the influence of medieval ornaments on contemporary art is due tomorrow. Luckily you have the latest wireless modem for your laptop, and hundreds of pieces from the Metropolitan Museum of Art collection are displayed on its web site. But as you examine the pictures, your web browser repeatedly gets stuck with partially loaded web pages. You see a reliquary, but an empty box sits where a scepter should appear; you find a Modigliani, but compression artifacts cause a Magritte to appear wrongly cubist.

A lot of things could be going wrong; thus, many technological improvements could save you. Having more antennas on your laptop would make multipath a virtue instead of an impediment and could result in higher throughput and better reliability. More cellular base stations imply smaller cells and could lead to fewer conflicts with other users in your cell. Your wireless service provider could have higher capacities in the wired connections to its base stations. The whole wired infrastructure could be better, with fewer packets lost due to buffer overflows. The museum web site could handle more simultaneous connections or could be cached closer to you. Each of these changes could improve your browsing experience.

This article focuses on the compressed representations of the pictures. The representation does not affect how many bits get from the web server to your laptop, but it determines the usefulness of the bits that arrive. Many different representations are possible, and there is more involved in the choice than merely selecting a compression ratio.

The techniques presented here represent a single information source with several chunks of data ("descriptions") so that the source can be approximated from any subset of the chunks. By allowing image reconstruction to continue even after a packet is lost, this type of representation can prevent a web browser from becoming dormant.

Separate Layers, Separate Responsibilities

Network communication has many separations of functions and levels of abstraction. This is both the cause and product of assigning various design and implementation tasks to different groups of people. In networking, there is the canonical seven-layer open systems interconnection (OSI) reference model. The layers range from the physical layer, characterized by voltage levels and physical connectors, to the application layer, which interacts with the user's software application. All these layers are involved in the example of accessing the Met web site from an untethered laptop.

Beyond the OSI layering, there is a further separation that most people take for granted. This separation is between generating data to be transmitted (creating content) and the delivery of content. The artistic aspect of content generation—writing, drawing, photographing, and composing—is not an engineering function. However, the engineer has great flexibility in creating representations of audio, images, and video to deliver an artistic vision. This article addresses the generation of content and how it is affected by unreliable content delivery.

Where Does a Conventional System Go Wrong?

Current systems typically generate content with a progressive coder and deliver it with TCP, the standard protocol that controls retransmission of lost packets. Though these techniques are well suited to many applications, putting

1053-5888/01/$10.00©2001IEEE

Box 1
Why Wait When You Don't Have to?

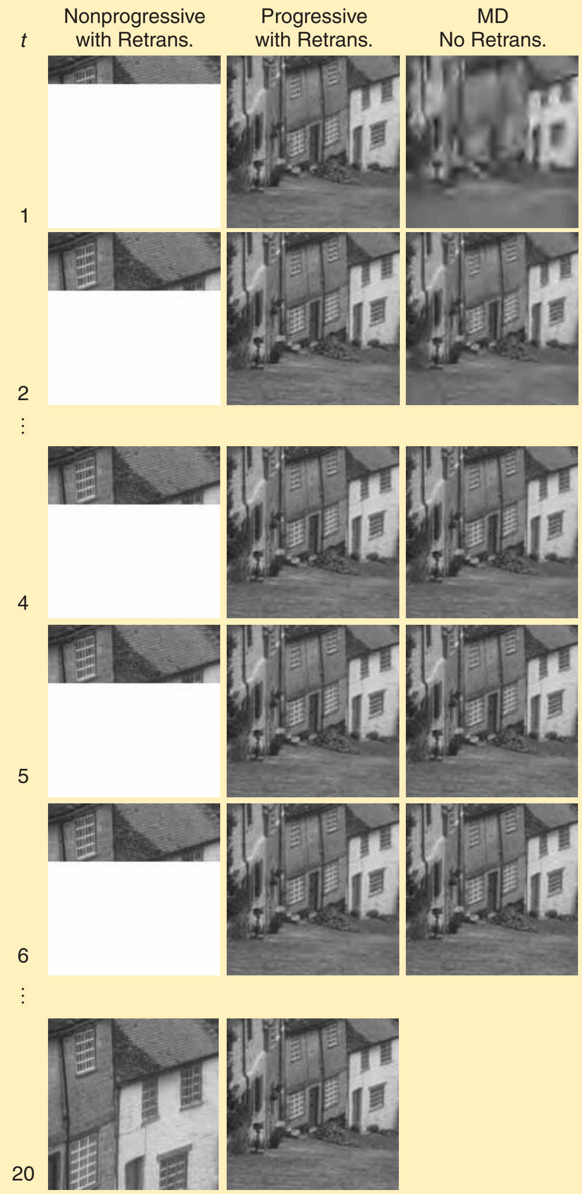
This box illustrates how the representation used for an image can affect a user's experience during a download. Three types of source codes are used: a nonprogressive code designed purely for compression efficiency, a progressive code, and an MD code. The first two are combined with a retransmission protocol but the third is not.

Assuming that the display software reconstructs an image as packets arrive, the displayed image could change in time as shown to the right.

In all three cases the image is represented with six packets, and the third packet is lost in transmission. For the first two cases, the loss is temporary in that the lost packet is eventually retransmitted and received at time 20.

Up until the first packet loss, the progressive scheme is clearly the best. However, the image quality stalls until the retransmission of the third packet is successfully completed. At the end, the nonprogressive code is the best, but only slightly better than the progressive code. Using MD, the image quality continues to improve as packets arrive, despite the loss of the third packet. From soon after the first loss until the retransmission is complete, the MD code gives the best image. For many applications, it is not worth waiting for the retransmission to get a slight improvement in image quality.

Make no mistake: if there were no packet losses, the progressive solution would be the best, and this is what is most commonly used now. However, when packets are lost, using MD coding can get a useful image to the user more quickly.

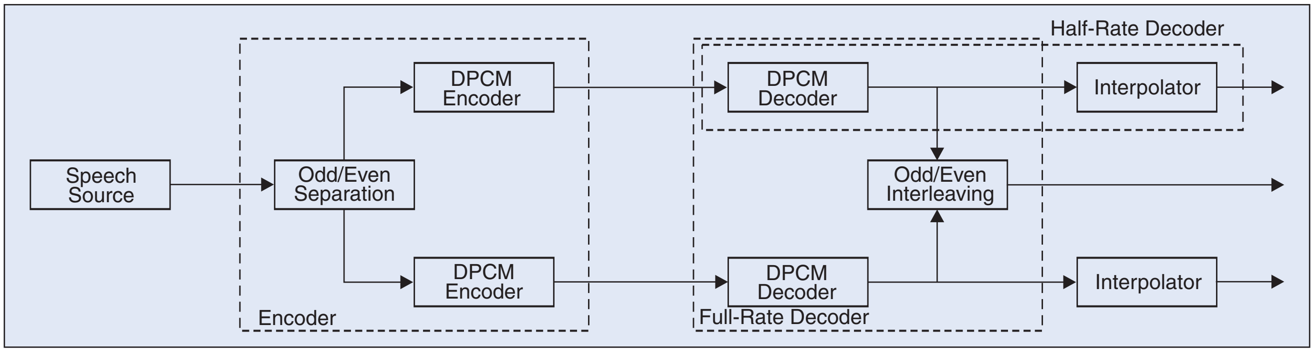


them together can produce avoidable delays when packets are lost.

Suppose L packets, numbered from 1 to L, are used to send a compressed image and that the receiver reconstructs the image as the packets arrive. In a progressive transmission, the quality improves steadily as the number of consecutive packets received, starting from the first, increases.(Two articles in this *Magazine* give details on image representations that are good for progressive transmission. Embedded coding allows the compressed data to be cut off at any point with commensurate image quality [95]. The more flexible concept of scalability is used in the JPEG2000 standard to facilitate many types of progressive transmission [92].) The order of the packets is critical; for example, if packets $\{1,2,4,5, ..., L\}$ are received, the quality is proportionate to the reception of only two packets.

Progressive transmission works well when the packets are sent and received in order without loss. But when a packet is lost, the reconstruction stalls until that particular packet is received. Unfortunately, TCP-based content delivery suffers from these stalls because the delay in receiving a retransmitted packet may be much longer than the interarrival times between received packets.

In creating packets that are only useful if all earlier packets have been received, the source coding (compression) in the conventional system puts too much faith in the delivery mechanism. On the other hand, the delivery



▲ *1. Speech coding for channel splitting as proposed by Jayant [50]. Speech sampled at 12 kHz is split into odd and even streams. These are separately encoded by DPCM.*

mechanism assumes all the transmitted packets are needed at the receiver; this might not be true with different source coding.

What Are the Alternatives?

If losses are inevitable, representations that make all of the received packets useful, not just those consecutive from the first, can be of great benefit. Multiple description (MD) coding creates such representations, and the possible improvement is demonstrated in Box 1. To gain robustness to the loss of descriptions, MD coding must sacrifice some compression efficiency. Thus, MD coding should be applied only if this disadvantage in compression is offset by the advantage of mitigating transport failures.

This article is structured in two parts. The abstract model for MD coding is described in the first part with history and initial motivation. Although packet-based communication is a key application today, the abstract model can be connected to several other communication scenarios. In fact, MD coding was originally intended for the circuit-switched telephone network. The second part of the article describes methods and applications of MD coding. The introduction focused on images to match the theme of this special issue, but MD coding is equally applicable to audio and video. Throughout the article, detailed descriptions of both the theory and the practical techniques are set off in boxes. Readers may wish to refer to the background material on source coding and quantization in the first article of this *Magazine* [34]. The topics of the article are covered more thoroughly in [33].

The MD Model

From the information theory literature, one could get the impression that MD coding arose as a curious analytical puzzle and then found application years later. More accurately, MD coding has come full circle from explicit practical motivation to theoretical novelty and back to engineering application.

Like so much of communication technology, MD coding was invented at Bell Laboratories in connection with communicating speech over the telephone network. Unfortunately, some of this work, though documented through citations in technical reports, was not archived. Little of it is known inside Bell Laboratories and almost none is known outside.

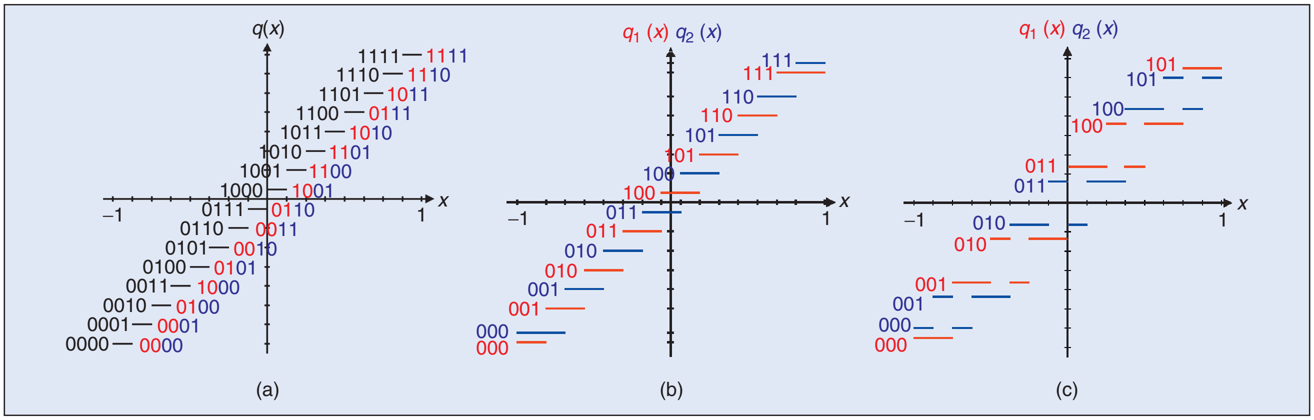
The hallmark of the telephone system is reliability. But outages of transmission links are inevitable. They arise from device failures and also from routine maintenance and upgrades. Thus, achieving high reliability requires mechanisms to handle outages.

In the 1970s, as today, the primary mechanism for providing uninterrupted service despite link outages was to divert calls to standby transmission links. The need for standby links increases costs and implies that not all the installed links are used in normal operation.

To improve reliability without standby links, the information from a single call could be split and sent on two separate links or paths. In normal operation, both halves of the transmitted data would be combined for the usual voice quality; an outage of one link or the other would allow for communication at reduced quality. In [28], this idea of channel splitting is attributed to W.S. Boyle. A citation of [67] and the later archived document [68] indicate that this idea may have been originated by Miller. Another early document on this topic is [9].

Miller sketched a few simple methods for sending digital and analog information over split, discrete-time, analog links. Miller's methods for digital information are all more or less equivalent to Gray coding [85]. For analog information, he was clearly interested in speech. Because of its decaying frequency spectrum, speech that is initially sampled at the Nyquist rate can be subsampled by two without too much aliasing. Thus, sending odd-numbered samples on one channel and even-numbered samples on the other works reasonably well. More details on a related technique are given below.

Miller and Boyle worked primarily on the physical layer, designing optical equipment. In 1978 and 1979, the idea of channel splitting became popular with two groups in Bell Laboratories: speech coders and information theorists. Gersho, spanning these two camps, was instrumental in this spread. He learned of the problem from Goodman, proposed an encoding technique [28], and likely was the first to share the idea with Jayant, Ozarow, Witsenhausen, Wolf, and Wyner.



▲ 2. *(a) A four-bit uniform scalar quantizer. With the natural labelling of cells (black), there is no way to split the output into two pairs of bits such that each pair gives a reasonable estimate of the input. Another labelling (red and blue) is somewhat better, but still does not consistently give good estimates (see Box 7). (b) Two three-bit quantizers that complement each other so that both outputs together give about four-bit resolution. (c) More complicated quantizers together attain four-bit resolution while each having fewer output levels.*

Speech Coding for Channel Splitting

Jayant, working independently of Miller, also proposed a separation of odd and even samples in a speech coding method for channel splitting [50]. Jayant's simple, yet quite effective system is depicted in Fig. 1. A similar system had been motivated by random losses in packet-switched telephony [52].

The usual practice in telephony is for a speech signal to be bandlimited to 3.2 kHz and sampled at 8 kHz. In Jayant's system, the initial sampling is at 12 kHz so that subsampling by a factor of two results in only slight aliasing. The odd- and even-numbered samples are compressed separately with differential pulse code modulation (DPCM) and sent on two separate channels. Quantization step sizes are adapted once for each block of several millisecond duration.

Decoding from both channels requires a DPCM decoder for each channel and the interleaving of the samples, resulting in a signal with 12 kHz sampling and some amount of DPCM quantization noise. To decode from a single channel, adaptive linear interpolation is used. It is almost the same thing to consider this to be speech sampled at 6 kHz with aliasing and quantization noise.

Judging by both SNR and informal perceptual testing, this technique works very well for the range of 2 to 5 bits/sample (24 to 60 kbits/s): The quality of either half-rate reception in a system designed for total rate R is similar to that of the full-rate reception in a system optimized for total rate $R/2$. In particular, at 60 kbits/s even the half-rate receiver approaches "toll quality." The good performance is not surprising because, at the bit rates used for telephony, halving the sampling rate while keeping the quantization unchanged is a reasonable way to halve the bit rate.

A mathematical idealization of this system is analyzed in Box 2. It is critical that the odd- and even-numbered samples have redundancy. If the redundancy was removed before odd/even separation—for example, with linear prediction—the performance would be unacceptable.

Scalar Quantization for Channel Splitting

To see the difficulty of channel splitting for a source without redundancy, consider trying to communicate a single real number $x \in [-1,1]$ with 4 bits. The natural choice, which is optimal if the source is uniformly distributed, is the uniform quantizer shown in Fig. 2(a) with black labels. No manner of splitting the four bits into two pairs for transmission over two channels is satisfactory; any estimate computed only from the channel that does not receive the most significant bit will, on average, be poor.

One way to use scalar quantizers for channel splitting was recognized almost immediately. Different uniform quantizers, shown in red and blue in Fig. 2(b), are used for the two channels, with the quantizers offset so that combining the information from both gives one additional bit of resolution. To avoid clutter, the reconstructions computed when both channels are available are not shown; if $q_1(x) = 110$ and $q_2(x) = 101$, for example, x must be in the interval [7/16, 9/16] and thus is reconstructed to 1/2.

Although the reconstructions from either channel alone are good, the quantizers in Fig. 2(b) have a glaring weakness: the total rate over both channels is 6 bits per sample even though the reconstruction quality when both channels are received is only about as good as with the 4-bit quantizer in Fig. 2(a). Reudink was the first to propose channel splitting techniques that do not increase the total rate so much and do not rely entirely on preexisting redundancy in the source sequence. Fig. 2(c) is inspired by one of Reudink's families of quantizer pairs.

Individually, the red and blue quantizers are strange in that their cells are not connected. But together, they complement each other by having cells with small intersections. For example, knowing $q_1(x) = 100$ only limits x to $[1/4, 3/8] \cup [1/2, 3/4]$; also knowing $q_2(x) = 100$ then localizes x to the interval [1/2,5/8]. Each quantizer in Fig. 2(c) has only six outputs. If we are willing to call these

Box 2
Channel Splitting by Odd/Even Separation

In signals with high correlation between consecutive samples, the odd- and even-numbered samples carry almost the same information. This is exploited in the channel splitting scheme for speech shown in Fig. 1. With mathematical idealizations of the speech signal and of the encoding of the separate odd and even sequences, we can assess how well this simple technique works.

Consider a discrete-time source sequence generated as

$$x[k] = a_1 x[k-1] + z[k], \quad k \in \mathbb{Z},$$

where $z[k]$ is a sequence of independent, zero-mean Gaussian random variables. This is a first-order autoregressive (AR) signal model [74]. The scalar constant a_1, $|a_1| < 1$, is the correlation between consecutive samples. Setting the variance of $z[k]$ to $1-a_1^2$ is a normalization that makes $x[k]$ have unit power.

The performance of the best possible compression of $x[k]$ is described by the distortion rate function of the source (see Box 3). The distortion rate functions of Gaussian AR sources can be computed easily for any rate [6]; simple distortion rate expressions hold except at very low rates. For the given source

$$D(R) = \left(1-a_1^2\right)2^{-2R} \quad \text{for} \quad R \ge \log_2(1+a_1).$$

The separation of odd and even samples gives sequences $x_1[k] = x[2k+1]$ and $x_2[k] = x[2k]$. Each of these is an AR sequence with correlation a_1^2 between consecutive samples. The lower correlations make the split sequences harder to compress than the original source. Using the new correlation a_1^2 in place of a_1 in the previous formula gives the best possible performance of the full-rate decoder:

$$D_{\text{full}}(R) = \left(1-a_1^4\right)2^{-2R} \quad \text{for} \quad R \ge \log_2\left(1+a_1^2\right).$$

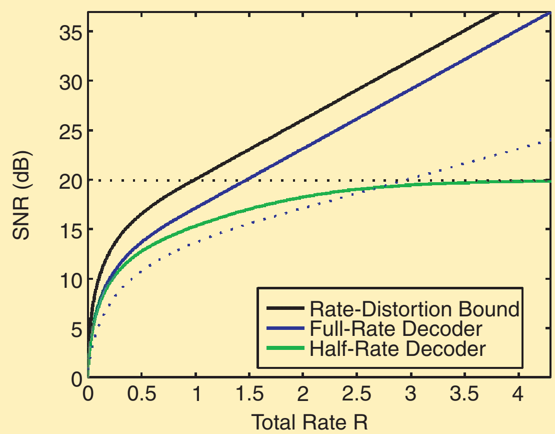
The loss in compression efficiency created by the odd/even separation is a constant multiplicative increase in distortion (for large enough rates)

$$D_{\text{full}}(D)/D(R) = 1 + a_1^2.$$

The half-rate decoder that receives the even samples has the same distortion on the even samples as the full-rate decoder. In addition, it must estimate the odd samples from the even samples. In Jayant's system, missing samples are estimated by linear interpolation between the two neighboring received samples. The resulting mean-squared interpolation error is $(1-a_1)^2 + (1/2)(1-a_1^2) + \omega D_q$, where D_q is the mean-squared quantization error and $\omega \in [0,1]$ depends on the correlation of the quantization error. Averaging the distortion on odd and even samples, the performance of each half-rate decoder is thus

$$D_{\text{half}}(R) = \frac{1}{2}\left[(1-a_1)^2 + \frac{1}{2}(1-a_1^2)\right] + \frac{1+\omega}{2} D_{\text{full}}(R).$$

Using the conservative assumption of $\omega = 1$ gives the performances shown below for $a_1 = 0.98$. For clarity, SNR in decibels, $-10\log_{10} D$, is plotted instead of distortion.



The performance of the half-rate decoder approaches an asymptote for high rates because of the interpolation error. However, at low rates the performance is competitive with, and at times better than, the full-rate decoder at half the rate (dashed curve). This is consistent with the results obtained with speech signals [50].

Correlated samples can be exploited to produce MDs of other sources. For images, separating odd- and even-numbered samples both horizontally and vertically leads to simple techniques for generating four descriptions [94]. Any method can be used to compress the four subsampled images of quarter size.

$\log_2 6 \approx 2.6$-bit quantizers, the quantizer pair in Fig. 2(c) attains 4-bit resolution with lower total rate than the quantizers in Fig. 2(b).

Though Reudink's work was archived as a technical report [81], it was not published. His techniques for memoryless sources were reinvented, analyzed in detail, and popularized many years later by Vaishampayan [96]. Reudink also developed techniques for correlated sources that built upon work by Gersho.

Gersho proposed the use of modulo-PCM encoding for channel splitting [28]. Returning to the quantizer in Fig. 2(a), consider the problem of estimating x after receiving only the two least significant bits. (This is called a modulo-PCM representation of x because the PCM representation is reduced modulo 4 [25].) Because there are two missing (most significant) bits, we know only that x lies in one of four intervals of length 1/8. Now suppose that the previous sample is also available and is highly correlated with the present sample. This correlation can be used to choose the most likely amongst the four intervals or to produce a linear least-squares estimate conditioned on the previous sample. For channel splitting, one channel can carry the most significant bits of the even-numbered samples and the least significant bits of the odd-numbered samples. This assignment of bits to channels was suggested by Goodman and studied by Quirk

Box 3
A Rate Distortion Primer

The articles of this special issue discuss the approximate representation of signals, mostly images. There is always a trade-off between the length of the representation (rate) and the quality of the approximation (distortion). Rate distortion theory is the branch of information theory devoted to bounds on achievable rates and distortions. Tight bounds are difficult to compute in all but a few simple situations. Nevertheless, rate distortion bounds give valuable intuition on how quality should vary with description length.

This box defines a few technical terms and gives an example of a rate distortion function. For readers new to information theory, this should make it easier to understand Boxes 2 and 4 and some of the main text. The definitions given here are simplified from [13] by assuming a real, memoryless source and a single-letter distortion measure; to learn more, read [6], [13], [43].

Assume that a source produces a sequence of independent, identically distributed, real random variables $X_1, X_2, \ldots, X_n$. A *distortion measure* d gives a nonnegative numerical rating $d(x,\hat{x})$ to how well a source letter x is approximated by a reproduction $\hat{x}$. The distortion between sequences $x^{(n)} = (x_1, x_2, \ldots, x_n)$ and $\hat{x}^{(n)} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)$ is defined by

$$d(x^{(n)}, \hat{x}^{(n)}) = \frac{1}{n}\sum_{i=1}^{n} d(x_i, \hat{x}_i).$$

The most common distortion measure is the squared error $d(x,\hat{x}) = (x-\hat{x})^2$. The squared error distortion between sequences of length n is the squared Euclidean norm between the sequences divided by n.

A *source code* is a mechanism for approximately representing a source sequence. For a sequence $x^{(n)}$ of length n, a source code with rate R consists of an encoder mapping α from all possible source sequences to an index in $\{1, 2, \ldots, 2^{nR}\}$ followed by a decoder mapping β from $\{1, 2, \ldots, 2^{nR}\}$ to a reproduction sequence $\hat{x}^{(n)}$. The distortion associated with the code is defined as the expected value of the distortion measure applied to the source and reproduction:

$$D = E\left[d\left(X^{(n)}, \beta\left(\alpha\left(X^{(n)}\right)\right)\right)\right].$$

A rate distortion pair (R, D) is called *achievable* if, for some positive integer n, there exists a source code with length n, rate R, and distortion D. The closure of the set of achievable rate distortion pairs is called the rate distortion (RD) region. The rate distortion function $R(D)$ is the minimum rate such that (R, D) is in the RD region. Conversely, the distortion rate function $D(R)$ is the minimum distortion such that (R, D) is in the RD region. Note that these all depend on the source and distortion measure.

The boundary of an RD region is nearly impossible to determine by working directly from the definitions. First, it is hard to design a length-n source code to minimize D for a given R, or vice versa. (Explicit source codes of the type described here are usually called fixed-rate vector quantizers. The difficulty of optimizing vector quantizers is discussed in [20], [29], [44].) Moreover, the boundary points are generally not attained with finite n; thus, finding even a single boundary point seems to require a sequence of optimal source codes.

The main theorem of rate distortion theory turns a hopeless situation into a merely difficult one. This theorem equates the rate distortion function to the result of a constrained minimization problem for a scalar conditional density [6], [90]. Though difficult, this minimization problem has been solved for a few sources and distortion measures. For example, a Gaussian source with variance σ^2 has distortion rate function

$$D(R) = \sigma^2 2^{-2R}$$

for squared error distortion. Although this is just a single example, it bounds the distortion rate functions of all continuous-valued sources: For a source with probability density $f(x)$ and variance σ^2, the distortion rate function with respect to squared error satisfies

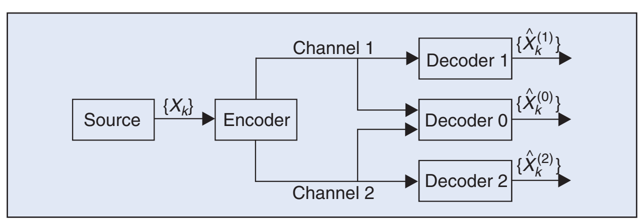
$$\frac{1}{2\pi e} 2^{2h} 2^{-2R} \le D(R) \le \sigma^2 2^{-2R}$$

where $h = -\int f(x) \log_2 f(x)\, dx$ is called the differential entropy. The upper bound above shows that, for a given variance, Gaussian sources are the most difficult to compress.

[77]. Reudink's generalization combined the grouping of nearby cells seen in Fig 2(c) with the grouping of highly separated cells in modulo-PCM.

The Information Theory of Channel Splitting

At the same time that Gersho, Jayant, Quirk, Reudink, and possibly others worked on channel splitting for speech, Witsenhausen recognized that channel splitting poses intriguing information theoretic problems. Since the speech coding work was barely known outside Bell Laboratories, and not much known inside, subsequent practical work seems to have been inspired by the information theory literature.



▲ *3. Scenario for MD source coding with two channels and three receivers. The general case has M channels and 2^{M-1} receivers.*

A Formal Problem

Channel splitting inspired the following question (stated first without mathematics): If an information source is

Box 4
The MD Rate Distortion Region

The MD rate distortion region, or simply MD region, for a particular source and distortion measure is the closure of the set of simultaneously achievable rates and distortions in MD coding. (The meaning of "achievable" is an extension from the one-description rate distortion problem. See Box 3.) For the two-description case, the MD region is the closure of the set of achievable quintuples $(R_1, R_2, D_0, D_1, D_2)$.

As described in Box 3 for the basic rate distortion problem, the MD region can, in principle, be determined by optimizing sequences of MD source codes. Again, as with the RD region, this computation is not feasible. However, as the theory stands, MD regions are inherently more difficult to determine; unlike RD regions, MD regions have not been completely characterized in terms of single-letter information theoretic quantities (entropies, conditional entropies, mutual informations, ...).

A theorem of El Gamal and Cover [22] shows how to determine achievable quintuples from joint distributions of source and reproduction random variables. The points that can be obtained in this manner are not *all* achievable quintuples, so these points are called *inner bounds* to the MD region. Other characterizations of achievable points in terms of information theoretic quantities have been given in [103], [119], but these also generally do not give the entire MD region. The achievable region of [103] generalizes the regions of [22] and [119] to more than two descriptions.

Points that are certainly not in the MD region are called *outer bounds*. The simplest outer bounds come from rate distortion functions. For Decoder 1 to have distortion D_1, it must receive at least $R(D_1)$ bits per symbol. Making similar arguments for the other two decoders gives the bounds

$$R_1 + R_2 \geq R(D_0) \tag{1}$$

$$R_i \geq R(D_i), \quad \text{for } i = 1,\ 2. \tag{2}$$

Because of the conflict in making the individual and joint descriptions good, the bounds (1) and (2) are usually loose.

The MD region is completely known only for memoryless Gaussian sources and the squared error distortion measure. For this source and distortion measure, Ozarow [72] showed that the MD region is precisely the largest set that can be obtained with the achievable region of El Gamal and Cover. This case gives insight into the limitations in MD coding. Furthermore, the MD region for any continuous-valued memoryless source with squared error distortion can be bounded using the MD region for Gaussian sources [117].

For a memoryless Gaussian source with variance σ_2^2, the MD region consists of $(R_1, R_2, D_0, D_1, D_2)$ that satisfy

$$D_i \geq \sigma^2 2^{-2R_i}, \quad \text{for } i = 1,\ 2, \tag{3}$$

$$D_0 \geq \sigma^2 2^{-2(R_1+R_2)} \cdot \gamma_D(R_1,\ R_2,\ D_1,\ D_2) \tag{4}$$

where $\gamma_D = 1$ if $D_1 + D_2 > \sigma^2 + D_0$ and

$$\gamma_D = \frac{1}{1-\left(\sqrt{(1-D_1)(1-D_2)} - \sqrt{D_1 D_2 - 2^{-2(R_1+R_2)}}\right)^2} \tag{5}$$

otherwise. The key relation is (4), which indicates that the central distortion must exceed the distortion rate minimum by the multiplicative factor γ_D. When one or both side distortions is large, $\gamma_D = 1$ so the central reconstruction can be very good. Otherwise, there is a penalty in the central distortion.

In the *balanced* case, where $R_1 = R_2$ and $D_1 = D_2$, Ozarow's result can be used to prove the following side distortion bound for a source with unit variance [33]:

$$D_1 \geq \min\left\{\frac{1}{2}\left[1 + D_0 - (1-D_0)\sqrt{1-2^{-2(R_1+R_2)}/D_0}\right],\ 1-\sqrt{1-2^{-2(R_1+R_2)}/D_0}\right\} \tag{6}$$

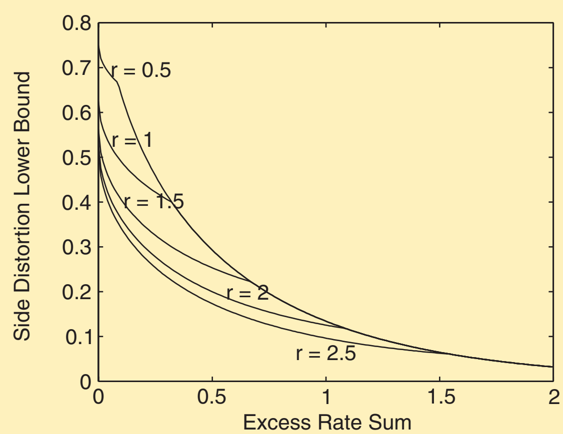
(In addition, the bound $D_1 > \sigma^2 2^{-2R_1}$ must still hold.) Written in terms of base rate $r = R(D_0)$ and redundancy $\rho = R_1 + R_2 - R(D_0)$,

$$D_1 \geq \begin{cases} \frac{1}{2}\left[1 + 2^{-2r} - (1-2^{-2r})\sqrt{1-2^{-2\rho}}\right], & \text{for } \rho \leq r - 1 + \log_2(1+2^{-2r}) \\ 1-\sqrt{1-2^{-2\rho}}, & \text{for } \rho > r - 1 + \log_2(1+2^{-2r}). \end{cases} \tag{7}$$

The bound (7) is plotted below for several values of the base rate r. An interesting thing to glean from this bound is the slope of the low-redundancy D_1 versus ρ characteristic:

$$\frac{\partial D_1}{\partial \rho} = -\frac{1-2^{-2r}}{2}\frac{2^{-2\rho}\ln 2}{\sqrt{1-2^{-2\rho}}}. \tag{8}$$

At $\rho = 0^+$, the slope is infinite. To interpret this, consider starting with a system that achieves the rate distortion bound for the central decoder and then increasing the rate by a small amount. The infinite slope means that the small additional rate will have much more impact if dedicated to reducing the side distortion than if dedicated to reducing the central distortion.



In many of the situations where MD codes are used—or should be used—some linear combination of central and side distortions is a good performance measure. (The weights could correspond to probabilities of receiving certain combinations of descriptions.) The infinite slope of (8) indicates

Box 4 (Continued)

that such a system should ideally have nonzero redundancy.

When actual performance is considered, instead of bounds, the large effect of a small amount of redundancy is again observed. The MD coding technique with correlating transforms (Box 9), for example, easily allows continuous variation of the redundancy and exhibits infinite side distortion slope at zero redundancy [36], [37].

Under the assumptions $R_1 = R_2 >> 1$ and $D_1 = D_2 \approx 2^{-2(1-\nu)R_1}$ with $0 < \nu \leq 1$, γ_D can be estimated as $(4D_1)^{-1}$. Thus bound (4) gives $D_0 \cdot D_1 \geq (1/4)\sigma^2 2^{-4R_1}$. For this reason, Ozarow's result is often interpreted as an exponential lower bound on the product of central and side distortions. MD quantization techniques can attain the optimal rate of exponential decay of this product [97], [98].

Box 5
Minimal Breakdown Degradation

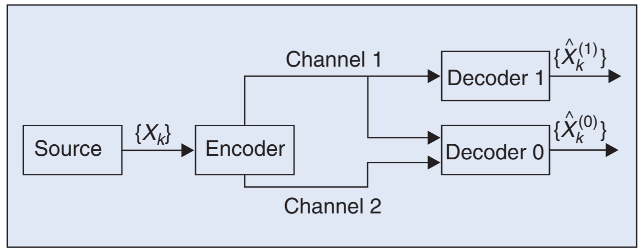
Much of the MD literature, including [2], [7], [111], [113], [114], [119], is focused on the memoryless binary symmetric source (BSS) with Hamming distortion. This means the source produces a sequence of bits with independent flips of a fair coin and that the distortion is measured by the expected fraction of bits that are incorrect in the reconstructed sequence. The source is incompressible: achieving zero Hamming distortion requires a rate of 1 bit per source symbol.

Suppose a BSS is communicated in an MD system with $R_1 = R_2 = 1/2$ and $D_0 = 0$. Since the total rate of 1 bit per symbol is precisely the minimum rate needed to achieve zero central distortion, there seems to be little room to achieve good side reconstructions. How small can $d = D_1 = D_2$ be? This is termed the minimal breakdown degradation [7], [111], [113], [114].

Since the source bits are independent and incompressible, it might seem that the best one can do is send half of the source bits on each channel. In this case, either side decoder receives half of the bits and can correctly guess half of the remaining bits, so $d = 1/4$ is achieved.

It is remarkable that the minimum common side distortion is lower than 1/4—specifically, $d = (\sqrt{2}-1)/2 \approx 0.207$. Obtaining this side distortion with the specified rates depends on choosing a suitable conditional distribution for the pair of side decoder reconstructions $(\hat{X}_1, \hat{X}_2)$ given the source bit X.

A conditional distribution that can be used to show the achievability of $d = (\sqrt{2}-1)/2$ is as follows. When $X = 1$, have the source bit reproduced correctly at both side decoders; i.e., $(\hat{X}_1, \hat{X}_2) = (1, 1)$ with probability 1. When $X = 0$, place probabilities $3 - 2\sqrt{2}$, $\sqrt{2} - 1$, and $\sqrt{2} - 1$ on $(0, 0)$, $(1, 0)$, and $(0, 1)$, respectively. One can easily check that the side distortion $d = (\sqrt{2}-1)/2$ is achieved and that X can always be recovered from the pair $(\hat{X}_1, \hat{X}_2)$. Calculations detailing why this conditional distribution can be brought about with $R_1 = R_2 = 1/2$ are given in [7] and [33]. Generalizations to more channels are considered in [103] and [112].



▲ *4. The successive refinement problem. It is obtained from the MD problem when one of the side decoders is removed.*

described with two separate descriptions, what are the concurrent limitations on qualities of these descriptions taken separately and jointly? Wyner presented this question, along with preliminary results obtained with Witsenhausen, Wolf, and Ziv at an information theory workshop in September 1979. This eventually came to be known as the MD problem.

The MD situation, with two descriptions, is shown schematically in Fig. 3. An encoder is given a sequence of source symbols $\{X_k\}_{k=1}^N$ to communicate to three receivers over two noiseless (or error corrected) channels. One decoder (the *central decoder*) receives information sent over both channels while the remaining two decoders (the *side decoders*) receive information only over their respective channels. The transmission rate over Channel i is denoted by R_i, $i = 1,2$, in bits per source sample. The reconstruction sequence produced by Decoder i is denoted by $\{\hat{X}_k^{(i)}\}_{k=1}^N$, and distortions attained by these reconstructions are denoted by D_i, $i = 0,1,2$. The technical meaning of attaining a particular distortion at a particular rate is described in Box 3.

As drawn, Fig. 3 suggests a situation in which there are three separate users or three classes of users. This could arise in broadcasting on two channels. The same abstraction holds if there is a single user that can be in one of three states depending on which descriptions are received. The introductory material is consistent with the latter interpretation.

MD coding is difficult because of conflicting requirements. If you design a good description at rate R_1 to send over Channel 1 and another good description at rate R_2 to send over Channel 2, there is no reason for the two descriptions together to be a good way to spend $R_1 + R_2$ total bits. Similarly, a good compressed representation at rate $R_1 + R_2$ cannot easily be split into two useful descriptions.

Heuristically, good descriptions at rates R_1 and R_2 are similar to each other; therefore, combining them at the central decoder does not give much advantage over just using the better one. Making descriptions individually good, yet not too similar, is the fundamental tradeoff of MD coding.

The MD model leads to several problems in rate distortion theory and practical compression. The central theoretical problem is to determine, for a given source and distortion measure, the set of achievable values for the quintuple $(R_1, R_2, D_0, D_1, D_2)$. In compression, one may attempt to explicitly design and implement good encoders and decoders. The theoretical bounds are ad-

MD coding has come full circle from explicit practical motivation to theoretical novelty and back to engineering application.

dressed in Box 4, and we will turn to practical techniques and applications after a look at related problems.

Related Problems

Much of the early theoretical MD work was for coding memoryless binary sources with no more *total* bits than would be necessary to communicate the source in one description. This is consistent with the original motivation of channel splitting: providing a fail-safe quality level without increasing the transmitted bit rate. A key result is described in Box 5.

Several other problems can be recognized retrospectively as special cases of MD coding. Removing Decoder 2, as in Fig. 4, gives what is known as successive refinement (SR) coding. Unlike in MD coding, the channels in SR coding play asymmetric roles. One channel (Channel 1) is received by both decoders, while the other (Channel 2) is received by only one decoder. Thus the information sent on Channel 2 need not be useful in isolation, i.e., without Channel 1. The description on Channel 2 is said to refine the information on Channel 1. The theoretical bounds for SR coding are established in [23], [57]-[59], and [82].

The SR coding abstraction applies to layered broadcasting when the decoders represent different users. The two classes of users are labeled 0 and 1. Both receive Channel 1, but only Class 0 receives Channel 2. This situation may occur in wireless transmission with multiresolution constellations [78] or in packet communication when multicasting to users with different available bandwidths [64].

When the decoders in Fig. 4 represent two different states of the same user, SR coding can be used for progressive transmission. The information on Channel 1 is sent first and the receiver uses Decoder 1. If communication is not terminated at this point, the information on Channel 2 is then sent, and the receiver uses Decoder 0.

Good SR source codes share the following characteristic: The most important data is sent on Channel 1 (thus to all users in a broadcast or first in sequence) and additional data to improve the reconstruction quality is sent on Channel 2. For example, most significant bits can be sent on Channel 1 and least significant on Channel 2. For images, a coarse or low pass version can be sent on Channel 1 with additional details sent on Channel 2; this is easy and common with wavelet representations [92], [95], [107].

Another antecedent network communication problem was introduced by Gray and Wyner in 1974 [45]. Instead of having a single source sequence to communicate over two channels to three receivers, they have a sequence of pairs of random variables $\{(X_k, \Upsilon_k)\}_{k=1}^{N}$ to communicate to two receivers over three channels. Receiver 1 is interested only in $\{X_k\}$ and forms its estimate from Channel 1 and a common channel. Receiver 2 has its own private channel and is interested in the other sequence $\{\Upsilon_k\}$. This is a special case of MD coding with three channels and seven receivers [103].

Applications

The application of MD coding requires a correspondence between "descriptions" and units of data that are transported between a sender and one or more receivers. As in our opening scenario, the packets in a data network are a good example. Further examples are detailed in this section.

Having established this analogy to the abstract model, MD coding is unlikely to be useful in the absence of the following conditions:

▲ *One or more users sometimes fail to receive one or more descriptions.* In point-to-point communication this means descriptions are sometimes lost. In broadcasting or multicasting, it may mean there are some users that always get a proper subset of the descriptions. If all the users always receive all of the descriptions, there is no need to worry about the robustness of the source code.

▲ *Various quality levels (distortions) are acceptable and distinguishable.* Essentially, the distortion measure must reflect utility. The reconstructions produced at side decoders should be more valuable than nothing, and central distortions lower than side distortions should create some value.

These two conditions determine, respectively, the *where* (communication media) and *what* (types of information sources) for applications of MD techniques. *How* to do MD coding well intimately depends on both of these. The remainder of this article addresses the how, where, and what of MD coding.

How to Generate MDs

The design of a source code always depends intimately on the source. However, there are a few basic components that appear in many practical techniques, like prediction, quantization, decorrelating transforms, and entropy coding [34]. MD codes can also be built from these basic components, along with channel codes and MD versions of quantizers and transforms.

This section focuses on MD quantizers and transforms. But first, let us consider the simplest ways to produce MDs. One is to partition the source data into several sets and then compress independently to produce descriptions. Interpolation is used to decode from any proper subset of the descriptions. The separation can be into odd- and even-numbered samples, as shown in Fig. 1, or a similar separation for more than two descriptions or multidimensional data. This technique can be very effective (see Box

2), but it relies entirely on the redundancy, in the form of correlation or memory, already present in the source.

Modern compression systems tend to begin by reducing redundancy with prediction and decorrelating transforms (see the earlier articles of this special issue [34], [92], [95], [106]). An odd/even separation of the resulting transform coefficients is not effective. To be complementary to these systems, a technique must work well on memoryless sources. All of the techniques described below work independently of the memory of the source.

Channel splitting inspired the following: If an information source is described with two separate descriptions, what are the concurrent limitations on qualities of these descriptions taken separately and jointly?

Progressive Coding and Unequal Error Protection

A trivial way to send two descriptions is to send the same description twice. The description can be produced with the best available compression technique. When only one description is received, the performance is as good as possible; however, no advantage is gained from receiving both descriptions.

A more flexible approach is to repeat only some fraction of data. It would be nice if the repeated data was the most important, and thus this type of fractional repetition is naturally matched to progressive source coding. To produce two R-bit descriptions, first encode to a rate $(2-\zeta)R$ with a progressive source code, with $\zeta \in [0,1]$. The first ζR bits are the most important and thus are repeated in both descriptions. The remaining $2(1-\zeta)R$ bits can be split between the descriptions. This is depicted as follows:

Description 1	ζR	$(1-\zeta)R$
Description 2	$\boxed{\zeta R}$	$(1-\zeta)R$

The box indicates redundancy added by repetition. Because some bits are protected with a rate-1/2 channel code (repetition) and the other bits are unprotected, this is called an unequal error protection (UEP) strategy for MD coding. (The rate of a channel code is defined in Box 6.)

UEP easily generalizes to more than two descriptions. To produce L descriptions, use channel codes with rates $1/L, 2/L, \ldots, 1$. The general case is clear from the following depiction of a system with four descriptions:

Description 1	$\zeta_1 R$	$\zeta_2 R$	$\zeta_3 R$	$\zeta_4 R$
Description 2	$\boxed{\zeta_1 R}$	$\zeta_2 R$	$\zeta_3 R$	$\zeta_4 R$
Description 3	$\boxed{\zeta_1 R}$	$\boxed{\zeta_2 R}$	$\zeta_3 R$	$\zeta_4 R$
Description 4	$\boxed{\zeta_1 R}$	$\boxed{\zeta_2 R}$	$\boxed{\zeta_3 R}$	$\zeta_4 R$

Vertically aligned portions are outputs of the same channel code.

The main design difficulty in using UEP is deciding how much of the data to code at each channel code rate. Techniques for this assignment given in [70] and [76] optimize a scalar objective function.

To assess the value of UEP as an MD coding technique, one can compare the rate distortion region (see Box 4) to the corresponding region attained with UEP. These regions can be determined completely for two descriptions, memoryless Gaussian sources, and squared error distortion. The difference in the regions can be characterized by comparing the minimum attainable central distortions for fixed values of the side distortions. The maximum over all rates and side distortions of this difference is about 4.18 dB [33]. The boundedness of the gap can be taken as a positive result, but 4.18 dB is significant. Analogous comparisons show that the maximum gap increases for more descriptions [103].

MD Quantization

In Fig. 2 and the related discussion, we saw how quantizers can be used to produce two complementary descriptions of the same scalar quantity. The example in Fig. 2(b) has two similar, ordinary-looking quantizers offset from each other. With B-bit quantizers producing each description, the central decoder has approximately $(B+1)$-bit resolution. This is a high redundancy (using $2B$ total bits for $(B+1)$-bit resolution), so it is not a good solution unless the side reconstructions are very important.

Reudink [81] invented several techniques with lower redundancy, one of which is exemplified by Fig. 2(c). Later Vaishampayan [97] independently developed a theory for designing MD scalar quantizers. (Details are given in Box 7.) This theory was extended from fixed-rate quantization to entropy-constrained quantization in [99].

MD scalar quantization is flexible in that it allows a designer to choose the relative importance of the central distortion and each side distortion. For the moment, consider the balanced case where $R_1 = R_2$ and $D_1 = D_2$. At high rates, the central and side distortion can be traded off while keeping their product constant [97], [99]. The exponential decay of this product as a function of the rate matches the optimal decay implied by [72]. The leading constant terms are consistent with what would be expected from high-resolution analysis of ordinary (single description) scalar quantizers [98].

Box 6
Channel Coding

Channel codes are used to protect discrete-valued data against errors and erasures. They can be used in conjunction with ordinary or progessive source codes to produce MD codes. This box gives a few definitions and properties of channel codes and then contrasts the philosophies behind channel coding and MD coding.

A block channel code that maps k input symbols to n output symbols, with $n > k$, is called an (n, k) code and is said to have rate k/n. Block channel codes are often *systematic*, meaning that the first k of the n output symbols are the input symbols themselves.

Suppose $n = k+1$ and the symbols to be encoded are b-bit binary numbers. Add to the inputs $z_1, z_2, \ldots, z_k$ a parity symbol z_n obtained from the bitwise exclusive-or of the inputs: $z_n = z_1 \oplus \cdots \oplus z_k$. Then any one lost symbol can be recovered by computing the exclusive-or of the k received symbols. However, there is no way to fully recover the inputs if more than one symbol is lost.

For $n > k+1$, the $n-k$ redundant symbols added by a systematic (n, k) code generalize the concept of parity. The codes can be designed so that the k input symbols can be recovered from any k of the n output symbols. Decoding fails if more than $n-k$ outputs are lost. Because the code is systematic, there may be some input symbols among the received symbols; the other input symbols are difficult or impossible to recover. This causes a sharp drop-off in reconstruction quality which is called the "cliff effect."

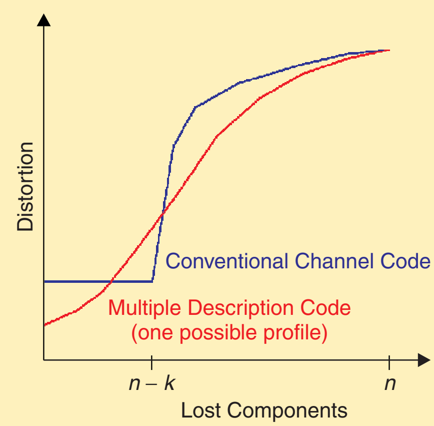
Lower rate codes withstand more losses, but send less information (fewer input symbols) per channel symbol. So how should the code rate be chosen? The code rate is typically chosen as high as possible subject to a maximum probability of decoding failure. This computation of code rate depends on the block length; short blocks necessitate low rate codes.

Assume the output symbols are lost independently and with a common probability p. When n is large, the number of received descriptions is predictable in that its standard deviation is small in comparison to its mean. Because of this predictability, one can choose k so that the probability of decoding error is small and the rate is as high as possible. In more technical terms, for any $\varepsilon > 0$ and $k = \lfloor n(1-p)(1-\varepsilon) \rfloor$, the probability of failed decoding can be made arbitrarily small by choosing n large enough. When ε is very small, the number of input symbols communicated $\lfloor n(1-p)(1-\varepsilon) \rfloor$ is only slightly smaller than the expected number of symbols received $n(1-p)$.

Channel codes do not work as well with small values of n; to have a low probability of failure requires k significantly less than $n(1-p)$. For example, suppose $p = 0.2$, $k = 4$, and $n = 5$. The probability of losing more than one description, and hence being unable to decode the channel code, exceeds 0.26. With $k = 3$, the probability of failure drops below 0.06, but a (5,3) code seems to have too much redundancy because the probability of receiving four or more symbols exceeds 0.73.

In the communication of continuous-valued information—the primary concern in this article—the discrete-valued inputs to a channel code are the outputs of a source code. Perfect transmission of the source coder output is an artificial aim; the real goal is to have a low-distortion reconstruction of the original source.

MD coding directly attacks the problem of communicating the continuous-valued source. MD codes can be designed with concern for every combination of received descriptions with appreciable probability. If so desired, this can give performance that varies gracefully with the number of received descriptions where a technique based on conventional channel coding would exhibit the cliff effect.



Extending the formalism of MD scalar quantization to vectors is easy. However, the index assignment problem becomes more difficult because the code vectors cannot be naturally ordered. In addition, the encoding complexity increases with dimension. An elegant technique that avoids these difficulties is the MD lattice vector quantization (MDLVQ) of Servetto et al. [89], [101]. The index assignment problem is simplified by lattice symmetries, and the lattice structure also reduces encoding complexity [12]. More details are given in Box 8. Other MD quantization techniques are described in [26], [27], and [49].

MD scalar quantization can be applied to transform coefficients. This is analyzed for high rates in [5], where it is shown that the transform optimization problem is essentially unchanged from conventional, single description transform coding. The next two techniques also use transforms and scalar quantizers, but in contrast they get their essential MD character from the transforms.

MD Correlating Transforms

The basic idea in transform coding is to produce uncorrelated transform coefficients because otherwise there is statistical dependency that could be exploited to improve performance (see [34], which details and qualifies this). For MD coding, statistical dependencies between transform coefficients can be useful because the estimation of transform coefficients that are in a lost de-

Box 7
MD Scalar Quantization

This box gives a formal notation for MD quantization and a systematic way to construct scalar quantizer pairs like the ones in Fig. 2. A fixed-rate MD scalar quantizer is comprised of an encoder α_0 and three decoders β_0, β_1, and β_2. (Compare to the decomposition of an ordinary source code in [34, Fig. 1].) The encoder α_0 produces from each scalar sample x a pair of quantization indices (i_1, i_2), and the three decoders produce estimates from (i_1, i_2), i_1, and i_2, respectively. In Fig. 2(b) and (c), for example, the i_1 and i_2 indices are shown in blue and red, respectively. The β_1 and β_2 decoder mappings are indicated by vertical positions; the action of β_0 is implicit.

Vaishampayan [97] introduced a convenient way to visualize the encoding operation and its range. First, the encoding is decomposed as two steps: $\alpha_0 = \ell \circ \alpha$. The initial encoder α is a regular quantizer, i.e., it partitions the real line into cells that are each intervals. The index assignment ℓ takes the index produced by the ordinary quantizer α and produces the pair of indices (i_1, i_2). The index assignment must be invertible so that the central decoder can recover the output of α. The visualization technique is to write out ℓ^{-1}, forming the index assignment matrix.

For the quantizer of Fig. 2(b), the index assignment matrix is

	000	001	010	011	100	101	110	111
000	0							
001	1	2						
010		3	4					
011			5	6				
100				7	8			
101					9	10		
110						11	12	
111							13	14

(The cells of the encoder α, taken in increasing values of x, are numbered from 0 to 14.) This matrix shows the redundancy in the representation through having only 15 of 64 cells occupied. It shows the quality of side reconstructions by the small ranges of values in any row or column.

An index assignment matrix with a higher fraction of occupied cells leads to a quantizer pair with lower redundancy. The matrix below corresponds to the quantizers in Fig. 2(c):

	000	001	010	0111	100	101
000	0	1				
001	2	3	5			
010		4	6	7		
011			8	9	11	
100				10	12	13
101					14	15

The extended range of values in any row or column (a maximum difference of 3, as compared to 1 in the previous index assignment) indicates higher side distortions.

If the index assignment matrix is full, there is no redundancy, and, necessarily, the side distortions are quite high. With a four-bit quantizer α, the following is the best that can be done with no redundancy. These index assignments are also marked in Fig. 2(a).

	00	01	10	11
00	0	1	5	6
01	2	4	7	12
10	3	8	11	13
11	9	10	14	15

In designing an MD scalar quantizer, one can optimize α, β_0, β_1, and β_2 quite easily. The optimization of the index assignment ℓ is very difficult. Thus, instead of addressing the exact optimal index assignment problem, Vaishampayan [97] gave several heuristic techniques that likely give close to the best possible performance. The basic ideas are to number from upper-left to lower-right and to fill from the main diagonal outward as in the examples above.

scription is improved. This idea for MD coding was originated by Wang et al. [109]. The transform in this technique explicitly adds redundancy where odd/even separation uses similar inherent redundancy.

Suppose X_1 and X_2 are independent, zero-mean Gaussian random variables with variances σ_1^2 and σ_2^2, $\sigma_1^2 \neq \sigma_2^2$. The possibility proposed in [109] to use $\Upsilon_1 = 2^{-1/2}(X_1 + X_2)$ and $\Upsilon_2 = 2^{-1/2}(X_1 - X_2)$ as descriptions of the vector (X_1, X_2). (Ignore for the moment that these would have to be quantized.) Obviously, (X_1, X_2) can be recovered from both descriptions. The descriptions Υ_1 and Υ_2 are correlated with correlation coefficient $(\sigma_1^2 + \sigma_2^2)^{-1}(\sigma_1^2 - \sigma_2^2)$. Thus, when one description is lost it can be more effectively estimated from the received description than if X_1 and X_2 were used as descriptions. Though [109] uses only this transformation, one can extend the technique to more general mappings from (X_1, X_2) to (Υ_1, Υ_2) and to longer vectors [36], [37].

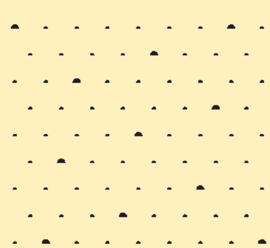
Instead of producing correlated transform coefficients and then quantizing them, quantizing first and then applying a transform turns out to work better [71], [110]. Although the transform maps from a discrete set to a dis-

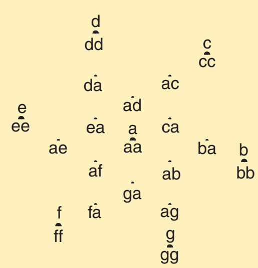
Box 8
MD Lattice Vector Quantization

The formalism of MD scalar quantization (given in Box 7) applies to vector quantization (VQ) without modification. For vectors of length N, the domain for the encoder α and codomain of the decoders β_0, β_1, and β_2 is $\mathbb{R}^N$.

Unless constraints are placed on encoding and decoding mappings, MDVQ may be impractical. First, the complexity of the initial quantization mapping α increases exponentially with the dimension N. Second, the heuristic design of the index assignment ℓ in MDSQ does not extend to MDVQ because there is no natural order on $\mathbb{R}^N$.

MD lattice vector quantization (MDLVQ) [89], [101] avoids these difficulties by using the symmetries of lattices. A lattice $\Lambda \subset \mathbb{R}^N$ and a sublattice $\Lambda' \subset \Lambda$ are chosen. The first lattice Λ fixes the resolution at the central decoder. The second lattice Λ', which is required to be a scaled and rotated version of Λ, determines the reconstruction points for the side decoders. Quantization is simplified by making α a nearest-neighbor encoder for Λ. The optimal index assignment $\ell: \Lambda \to \Lambda' \times \Lambda'$ is relatively simple because, under high-resolution conditions, it can be defined on an elementary cell and then extended to the whole space with appropriate symmetries.





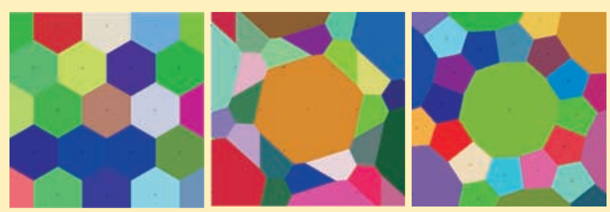
Below on the left, all the points form a lattice Λ and the larger points are a geometrically similar sublattice Λ'. On the right is an optimal index assignment.

As an example, for a vector quantized to the fine-lattice point denoted by `ab`, the reconstructions at the side decoders are `a` and `b`. The index assignment has been designed so that the side reconstructions are nearby elements of the coarse lattice Λ'. The side reconstructions cannot always be the closest coarse lattice points because the index assignment would then not be invertible.

MDLVQ was originally limited to the balanced case with $R_1 = R_2$ and $D_1 = D_2$. Diggavi et al. [16] extended this method to unbalanced descriptions by using two different sublattices and a similar index assignment technique.

In MDLVQ the choice of Λ fixes the central distortion and the choice of Λ' determines some minimum side distortion. A generalization by Kelner et al. [35], [55] alters α but not ℓ to give a continuum of central and side distortion operating points for any lattice and sublattice. This is achieved by encoding to minimize a weighted sum of central and side distortions; in contrast, α in the original encoding minimizes the central distortion. In addition, the points in Λ can be optimized while keeping Λ' fixed to further improve performance. These changes have only a small impact on the encoding complexity.

The diagrams below show the partitioning with the original encoding (left), with a weighted minimization of central and side distortions (center), and with optimization of fine lattice points (right).



The weighted minimization makes the points in the coarse lattice Λ' preferable to the other points because they have lower side distortions. The optimization of Λ with Λ' fixed makes the cells more spherical.

crete set, it can be designed to approximate a linear transform [36], [37]. See Box 9 for more details and an example.

MD Coding with Frames

The final technique has great similarity to a block channel code and attempts to alleviate the "cliff effect" (see Box 6). Suppose the source produces a vector $x \in \mathbb{R}^N$. The idea introduced by Goyal at el. [40], [41] is to left-multiply by a rectangular matrix $F \in \mathbb{R}^{M \times N}$, $M > N$, to produce M transform coefficients. These are scalar quantized and partitioned into L sets, $L \le M$, to form L descriptions. Under some mild conditions, the multiplication by F is a frame expansion [14], [19], so this representation is called a quantized frame expansion (QFE).

The source vector x can be simply estimated from the quantized expansion coefficients $y = Q(Fx)$ as a least-squares problem: $\hat{x} = \operatorname{argmin}_x \|y - Fx\|^2$. In this case, the distortion is proportional to N/M. More complicated reconstruction methods can improve the accuracy of the estimate [10], [42], [79].

The loss of some coefficients is equivalent to deleting the corresponding rows of F to get F'. As long as F' has rank N, this does not change the reconstruction techniques. If the rank of F' is less than N, a simple estimate is $\hat{x} = \operatorname{argmin}_{x: F'x = y} \|x\|^2$. With statistical information, better estimates can be computed.

The role of the frame operator F is similar to that of a block channel code. From N symbols (in this case real numbers, while a block channel code would have a discrete domain) it produces M symbols with linear dependencies. In QFE, the dependent quantities are quantized, breaking the strict linear dependence. Thus, each transform coefficient gives some independent informa-

Box 9
MD Coding with Correlating Transforms

A *correlating transform* adds redundancy between transform coefficients that makes these coefficients easier to estimate if they are lost. For example, let

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} \theta & (2\theta)^{-1} \\ -\theta & (2\theta)^{-1} \end{bmatrix}}_{T} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}$$

where θ is a positive real number and x_1 and x_2 are independent Gaussian random variables with variances σ_1^2 and σ_2^2, $\sigma_1^2 > \sigma_2^2$. The originally uncorrelated vector x yields y with correlation $E[y_1 y_2] = -\theta^2\sigma_1^2 + (2\theta)^{-2}\sigma_2^2$. Unless $\theta^4 = (4\sigma_1^2)^{-1}\sigma_2^2$, the transform coefficients are correlated. T is called a correlating transform.

Now suppose y_1 and y_2 are descriptions of x. (Quantization is neglected for now.) Since y_1 and y_2 are jointly Gaussian, the conditional expectation of x given either description alone is a linear function:

$$\hat{x}^{(1)} = \frac{2\theta}{4\theta^4\sigma_1^2 + \sigma_2^2} \begin{bmatrix} 2\theta^2\sigma_1^2 \\ \sigma_2^2 \end{bmatrix} y_1$$

$$\hat{x}^{(2)} = \frac{2\theta}{4\theta^4\sigma_1^2 + \sigma_2^2} \begin{bmatrix} 2\theta^2\sigma_1^2 \\ \sigma_2^2 \end{bmatrix} y_2$$

These estimates attain the per component MSE

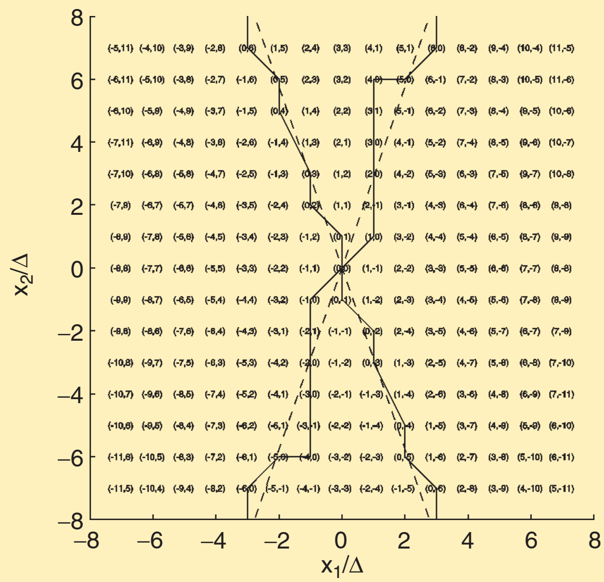
$$D_i = \frac{1}{2} E\left\| x - \hat{x}^{(i)} \right\|^2 = \frac{(\theta^2 + (4\theta^2)^{-1})\sigma_1^2\sigma_2^2}{2(\theta^2\sigma_1^2 + (4\theta^2)^{-1}\sigma_2^2)}.$$

This side distortion ranges from $\sigma_1^2/2$ to $\sigma_2^2/2$ as θ varies between zero and infinity. Thus, if θ is chosen appropriately, the side distortion compares favorably with the $(1/4)(\sigma_1^2 + \sigma_2^2)$ average side distortion that results from using x_1 and x_2 as descriptions.

Two issues have been neglected: quantization and how θ affects the rate for a given central distortion. Suppose y_1 and y_2 are uniformly quantized with a small step size Δ. As θ is increased above $\sqrt{\sigma_2/(2\sigma_1^2)}$, the magnitude of the correlation increases and thus the rate increases. Also, as θ is increased above $2^{-1/2}$, the transform strays further from orthogonality so the central distortion increases (see [34, Box 3]). The increase in rate is a necessary concession to lower the side distortions. However, the increase in central distortion can be eliminated by quantizing in the original coordinates and then applying a transform to the quantized values.

The quantization of x yields an element of the set $\Delta\mathbb{Z}^2$. We wish to have an invertible transform $\hat{T}:\Delta\mathbb{Z}^2 \to \Delta\mathbb{Z}^2$ that approximates T. Transforms of this type can be obtained in several ways [47], [118], [8] and can also be used for single description transform coding [32].

An example for $\theta = 6/5$ is depicted below. Each $([x_1]_\Delta/\Delta, [x_2]_\Delta/\Delta)$ position is labeled with $(y_1/\Delta, y_2/\Delta)$, where $y = \hat{T}([x]_\Delta)$ and $[\cdot]_\Delta$ denotes rounding to the nearest multiple of Δ. The solid curves connect the (x_1, x_2) pairs on the grid that map to $(y_1, 0)$ or $(0, y_2)$; they are like inverse images of the axes. These curves are somewhat erratic, but they approximate the dashed straight lines which are the points with at least one component of Tx equal to zero. This shows that $\hat{T}$ is a relabeling of the points $\Delta\mathbb{Z}^2$, but at the same time approximates the linear transform T.



Analyzed under the assumptions of high-resolution quantization theory, the use of a discrete transform lowers the central distortion by a factor of 2.6 in this example. Details on all of the calculations above can be found in [33], [37].

The discrete transform $\hat{T}$ performs a relabeling similar to an index assignment in MD quantization. However, unlike MD quantization, all index vectors are possible; thus, the redundancy is "softer" than in MD quantization. When one or more correlated transform coefficients are lost, the range of possible source values is infinite, but the posterior distribution of source values may be peaked. This is suggestive of the fact that correlating transforms are a good way to add a little redundancy, but a bad way to add a lot [33], [37].

tion—even those in excess of N coefficients. An example is given in Box 10.

Optimizations of F for MD coding are presented in [39] and [65]. Methods that use infinite-dimensional operators are given in [4], [17], [18], and [116].

Where to Use MDs

Packet Networks

It is almost automatic to connect packets in a packet network to descriptions in MD. This possibility is mentioned in the majority of recent works presenting MD techniques and seems to be the driving force behind the rapid development of these techniques.

Packets are lost in data networks for a variety of reasons. On the Internet, this creates a packet loss probability that varies widely with time of day, day of the week, and connection routing. Adding to problems created by congestion, the Internet is becoming more heterogeneous as backbone capacities increase and more

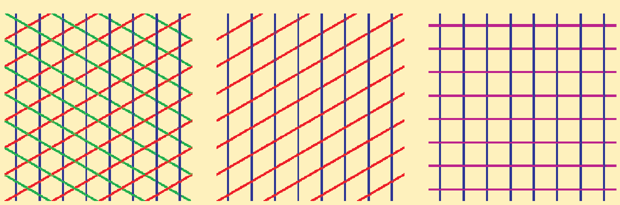
Box 10
Quantized Frame Expansions with Erasures

A quantized frame expansion gives measurements of an N-dimensional vector in M directions, $M > N$. The simplest interesting case is with $N = 2$ and $M = 3$. For example, let

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} 1 & 0 \\ -1/2 & -\sqrt{3}/2 \\ -1/2 & \sqrt{3}/2 \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}$$

where x_1 and x_2 are independent Gaussian random variables with unit variance. Then the transform coefficients y_1, y_2 and y_3 are also Gaussian random variables with unit variance, but they are not independent. We will consider the quantized transform coefficients to be three descriptions of x.

The information in any one quantized transform coefficient is the position of the source vector relative to a set of $(N-1)$-dimensional hyperplanes perpendicular to the corresponding row of F. Thus, the quantized version of y_1 gives the position of x relative to lines perpendicular to (1,0), as shown in blue below. The green and red boundaries correspond to the quantized values of y_2 and y_3, respectively. Quantizing x_1 and x_2 directly, instead of using any transform, represents x using the cells formed by the blue and magenta lines.



The partition on the left shows the cells to within which a decoder that receives all the descriptions can isolate x. Note from this partition that even though y_1, y_2 and y_3 are linearly dependent, their quantized values each give some information unavailable in the other two. This makes the partition on the left the finest.

The center partition shows the information available from two descriptions. It would be better if the cells were squares (see [34, Box 3]), but it is not possible to design F so that the rows are pairwise orthogonal (excluding a row of zeros).

For a numerical comparison with a block channel code, suppose each transform coefficient y_i is quantized with an optimal eight-bit quantizer to obtain $\hat{y}_i$. The quantization noise power is $E[(y_i - \hat{y}_i)^2] \approx 8.8 \cdot 10^{-5}$. The MSE per component $(1/2)E[\|x - \hat{x}\|^2]$ will be less than this value when three descriptions are received and more otherwise.

The classical approach with a block channel code is to apply an eight-bit quantizer to each component of x, yielding two eight-bit strings w_1 and w_2. These strings are each used as descriptions, and a third parity description is formed from the bitwise exclusive-or or $\mathbb{Z}_2$ addition of these strings: $w_3 = w_1 \oplus w_2$. The first two descriptions can be be recovered from any two of the three descriptions. However, receiving all the descriptions is no better than receiving two, and the third description is essentially useless alone.

The per component MSE distortions for all of the possible combinations of received descriptions are given in the following table:

Descriptions	{1,2,3}	Any two	{1} or {2}	{3}	∅
Parity	$8.8 \cdot 10^{-5}$	$8.8 \cdot 10^{-5}$	0.5	1	1
QFE	$5.8 \cdot 10^{-5}$	$1.2 \cdot 10^{-4}$	0.5	0.5	1

The quantized frame system is worse in one situation and better in two others; the best choice depends on the likelihoods of these events and other design criteria. The advantages of QFE are the symmetry of the three descriptions and the fact that each description contains information that cannot be inferred completely from the other two.

low-bandwidth, wireless devices are connected. If the packet injection rate is not matched to the capacity of the bottleneck link, dropping packets becomes necessary. For example, packets injected at a rate sustainable on a fiber link might be routed to a wireless link, necessitating packet drops. If the network is able to provide preferential treatment to some packets, a progressive source code is suitable. Networks, however, usually will not look inside packets and discriminate; packets are dropped at random, making a perfect setting for MD coding.

The conventional way to handle a packet loss in a data network is to retransmit the lost packet. Retransmission protocols, whereby the receiver tells the sender either what arrived or what did not arrive, facilitate end-to-end reliability despite unreliable links. When packet losses are sporadic, not the result of consistently insufficient bandwidth, retransmission makes efficient use of network resources; packets are each successfully sent once, and the additional load on the network is just the small amount of feedback from the receiver. On the other hand, when packet losses are frequent, retransmission can create an even more congested environment. This vicious circle can result in intolerable delay, especially for real-time services. TCP, the retransmission protocol usually used on the Internet, avoids this by dropping the packet injection rate.

A few things can preclude retransmission. The most obvious is a lack of feedback. In some situations with packetized communication, the receiver either has no way to get a message back to the sender, or it is too expensive. Too much feedback creates a problem, applicable to broadcast, called feedback implosion. In broadcasting, acknowledgment or negative acknowledgment messages from every receiver to the broadcaster creates too much traffic. Moreover, the broadcaster cannot afford to honor independent retransmission requests from each receiver.

Where retransmission is possible its primary drawback is delay. Whether positive or negative acknowledgments are used, retransmission implies an added delay of at least one round-trip transport time. Furthermore, each repeated transmission may be lost, so the delay may be arbitrarily large. Small delays are of critical importance in interactive communications. Also, for streaming audio or video, the transport delay variation determines the size of buffer required and the buffering delay.

The alternative to retransmission is to make do with whatever arrives upon first transmission. This is exactly MD coding. The established way to improve robustness without retransmission is to use a conventional channel code (see Box 6). In the networking literature, channel coding is usually referred to as forward error correction (FEC). Note that reliable use of channel codes requires long block sizes, which in turn creates the difficulties associated with delay.

The length of an FEC code is limited to the number of packets used in the communication since robustness comes from placing channel code output symbols in different packets. As an example, consider a network using Internet Protocol, Version 6 (IPv6) [15]. An IPv6 node is required to handle 576-byte packets without fragmentation, and it is recommended that larger packets be accommodated. Accounting for packet headers, a 576-byte packet may have a payload as large as 536 bytes. With packets of this size, a typical Internet image may be communicated in about ten packets. (This number of packets is based on the mode of JPEG image sizes of the Hot 100 web sites reported in [46].) Ten is low for the number of output symbols of a channel code, so good performance is hard to attain with FEC.

In summary, MD techniques seem appropriate for packet networks when retransmission is not possible, long delays are not acceptable, and/or the number of packets needed is more than one but not too large. Routing diversity—where packets are purposely sent on different routes to insure against the failure of a single route—increases the applicability of MD coding, as does the high packet loss probability on wireless links [30], [31]. The importance of latency requirements is described in [91].

Distributed Storage

Perhaps even more than packet communication, distributed storage matches the MD framework well. Consider a database of images stored at several locations with MD encoding. A typical user would have fast access to the local image copies; for higher quality, one or more remote copies could be retrieved and combined with the local copy.

Distributed storage is common in the use of edge servers for popular content. In current implementations, identical data is stored at the servers, so there is no advantage in receiving multiple copies. Storage can also be distributed to make the reliability of each device less important; lowering reliability requirements can decrease costs [75].

Box 11
How Was Box 1 Created?

The images shown in two right columns of Box 1 represent an honest experiment. The SPIHT coder [84] publicly-available at http://www.cipr.rpi.edu/research/SPIHT/ was used to encode the standard 512-by-512 pixel goldhill image at 1.2 bits/pixel. (A 128-by-128 pixel segment is shown.) The resulting bits are split into six equal-sized packets for the middle column. To produce the right column, a UEP scheme is applied to the SPIHT output to generate six descriptions at 0.2 bits/pixel/description.

The SPIHT coder is a state-of-the-art progressive coder, so the performance in the middle column is essentially as good as possible. Guided by theoretical results for memoryless Gaussian sources, we should not expect the UEP scheme to be nearly optimal among MD coders. Thus the right column could be improved upon.

The images in the left column are a bit contrived. For illustration, a SPIHT-compressed image is used at a slightly higher rate (1.3 bits/pixel), and the transmission in packets is imagined to correspond to a decomposition in six horizontal strips. In principle, with a nonprogressive representation it could be impossible to recover anything meaningful from one or two of six packets.

Frequency-Hopping Wireless Systems

In wireless channels, bit errors are usually more important than losses of chunks of information; hence, error correcting codes are critical. However, there are situations in which a wireless channel is naturally decomposed into more than one virtual channel. MD coding may be appropriate on these virtual channels. Two such scenarios are frequency-hopping systems and radio broadcast with separate upper- and lower-neighboring interferers.

Frequency-hopping systems provide robustness to the sensitivity of bit error probability as a function of carrier frequency. Having the transmitter hop through a set of carrier frequencies, in a manner known to the receiver, gives insurance against picking a bad carrier frequency: some frequencies will be good, and others will be bad. This is especially important when the channel varies quickly or feedback is not available, and thus the transmitter cannot know these variations.

Now suppose that the variation of the propagation environment is such that the error characteristics are approximately constant for a unit time interval and that the transmitter hops among L carrier frequencies in this interval. The L carrier frequencies can be considered separate channels for an MD source code. Channel codes can be applied separately for each carrier frequency. For some channels all errors will be corrected; the other channels are considered lost.

Choosing the number of frequencies L is critical. Larger L gives more diversity within a fixed time interval and thus allows the use of longer, more effective channel codes. However, hopping more frequently is technologically more difficult because, for example, of synchronization and signal acquisition times. This can cause the time available for the transmission of data to be shortened and hence for the data rate to be reduced. This type of tradeoff between diversity and data rate is present in the Bluetooth standard [1, Sect. 4.4.2]. MD coding requires some diversity, but would tend to work with lower diversity—hence higher data rates—than methods based purely on channel coding. See [115] for related results.

Hybrid Digital Broadcast

In radio broadcasting, stations are separated in the spectrum: FM stations are separated by 200 MHz and AM stations are separated by 10 kHz. The frequencies are assigned geographically so that equal or neighboring frequencies are not used in close proximity. Still, at the edge of a radio station's range the interference from a station with neighboring frequency may be significant. For a given station, because of geography, some users will have interference at the upper edge of the band and other will have it at the lower edge.

Recently, ways to add digital information to radio broadcasts without significantly degrading the audio quality for legacy receivers have been devised [51], [73]. These systems put the digital information at the edges of a station's band because there it has the least impact on audio quality. Because of the different interference at upper and lower edges of the band, these edges make for good MD channels: some users will get only one or the other channel, and other users will get both.

General Principles, Again

The MD model is quite explicit, but it can be simplified and generalized by saying: Do not require the transport mechanism (modulation, channel coding, transmission protocol) to be flawless, and design the source coding appropriately.

This paradigm calls into question the meaning of the capacity of a channel. Intuitively, it is less demanding to ask for every bit that gets across the channel to be correct than to ask for every bit that is transmitted to correctly get across the channel. This leads to new capacity definitions and, potentially, higher capacities [21]. Ideally, an MD source code would make all the received data useful and the loss of some of the transmitted data not catastrophic. Furthermore, the dependencies in descriptions could be used to improve demodulation and decoding performance [93].

What Can MDs Be Used On?

Convinced, hopefully, that MD coding applies to various communication media, let us turn to the data itself. What types of data can MD coding be applied to? As was emphasized earlier, the data must be useful at various quality levels. In very few cases is this true for text (e-mails, newspaper articles, etc.) or numerical data (stock quotes, part specifications, etc.). However, it is true for most data that are lossily compressed including speech, audio, image, video, and volumetric data.

Each of the generic techniques described in this article can be applied to various types of data, but each application has its own idiosyncrasies. There are too many combinations to describe in detail. Thus, this final part of the article serves mostly as a guide to the literature.

Audio

Speech coding for the telephone network was the original motivation for MD coding. The first technique used simple odd/even separation [50]. More recent techniques use prediction [48], [100], perceptual models [60], and repetition with optimized bit allocations [54]. The inclusion of correlating transforms in a perceptual audio coder is described in [3].

Images

The recent surge in MD coding was sparked by a pair of image coding papers. In a paper that has not received much attention, a two-dimensional analogue of odd/even separation was presented in [94]. The correlating transform method was introduced in the context of image coding in [109]. Better results obtained with more general correlating transforms appear in [110]. Other transform-based techniques are given in [11] and [53].

Progressive image codes are common and very effective. Thus, several research teams have attempted to apply UEP to progressively compressed images or to integrate UEP into such coders. Examples include [66], [69], [76], and [102]. The application of UEP is combined with additional channel coding for wireless channels in [83].

MD scalar quantizers are introduced in a wavelet image coder in [88] and quantized frames are applied to images in [10] and [38].

Video

The use of MD coding instead of retransmission or long channel codes is often motivated by delay. This is critical for the data that is projected to dominate network traffic: streaming video.

Several MD coding techniques have been proposed for video. In [62], a codec relatively similar to H.263 is proposed with the addition of MD protection of the most significant DCT coefficients in the macroblocks whose errors are most difficult to conceal. Also in an H.263 framework, a technique for MD coding of motion vectors is presented in [56]. Other techniques that alter prediction loops include [80], [99].

Because of latency and jitter requirements and high data rates, sending streaming video on packet networks is challenging. Joint design of the compression technique and transport protocols can yield significant improvements. This is considered, for example, in [61], [86], and [87].

Summary

Dividing critical tasks between yourself and a partner requires a certain faith. If that faith is rightly absent, each participant should plan accordingly. So it is with the division between compression and transport. If the transport mechanism is imperfect, the source coding should be adjusted to accommodate transport failures.

The building blocks of today's systems are superb under certain idealizations. Progressive coding is good with perfect transport. Retransmission is good when there is feedback from the receiver to the transmitter and the delays are not prohibitive. Conventional channel codes work best when the fraction of received symbols is predictable. In contrast, MD codes are designed for the imperfect and unpredictable transport in many real systems.

Acknowledgments

J. Kovačević was instrumental in the conception and organization of this article. Her comments and those of A.K. Fletcher, G. Kramer, R. Singh, N. Srinivasamurthy, and L. Vasudevan are gratefully acknowledged. Thanks also to A. Gersho, L. Ozarow, M. Quirk, and H. S. Witsenhausen for providing unpublished documents.

Vivek K Goyal received the B.S. in mathematics and the B.S.E. in electrical engineering (both with highest distinction), in 1993 from the University of Iowa, Iowa City. He received the M.S. and Ph.D. degrees in electrical engineering from the University of California, Berkeley, in 1995 and 1998, respectively. He was a Research Assistant in the Laboratoire de Communications Audiovisuelles at Ècole Polytechnique Fédérale de Lausanne, Switzerland, in 1996. He was with Lucent Technologies' Bell Laboratories as an intern in 1997 and again as a Member of Technical Staff from 1998 to 2001. He is currently a Senior Research Engineer for Digital Fountain, Inc., Fremont, CA. His research interests include source coding theory, quantization theory, and practical robust compression. He is a member of the IEEE, Phi Beta Kappa, Tau Beta Pi, Sigma Xi, Eta Kappa Nu and SIAM. In 1998 he received the Eliahu Jury Award of the University of California, Berkeley.

References

[1] Specification of the Bluetooth system, v. 1.1. Available: http://www.bluetooth.com (until superseded), Feb. 2001.

[2] R. Ahlswede, "On multiple descriptions and team guessing," *IEEE Trans. Inform. Theory*, vol. 32, no. 4, pp. 543-549, July 1986.

[3] R. Arean, J. Kovacevic, and V.K. Goyal, "Multiple description perceptual audio coding with correlating transforms," *IEEE Trans. Speech Audio Processing*, vol. 8, no. 2, pp. 140-145, Mar. 2000.

[4] R. Balan, I. Daubechies, and V. Vaishampayan, "The analysis and design of windowed Fourier frame based multiple description source coding schemes," *IEEE Trans. Inform. Theory*, vol. 46, no. 7, pp. 2491-2536, Nov. 2000.

[5] J.-C. Batllo and V.A. Vaishampayan, "Asymptotic performance of multiple description transform codes," *IEEE Trans. Inform. Theory*, vol. 43, no. 2, pp. 703-707, Mar. 1997.

[6] T. Berger, *Rate Distortion Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[7] T. Berger and Z. Zhang, "Minimum breakdown degradation in binary source encoding," *IEEE Trans. Inform. Theory*, vol. 29, no. 6, pp. 807-814, Nov. 1983.

[8] R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo, "Wavelet transforms that map integers to integers," *Appl. Comput. Harmon. Anal.*, vol. 5, no. 3, pp. 332-369, July 1998.

[9] J.C. Candy, D. Gloge, D.J. Goodman, W.M. Hubbard, and K. Ogawa, "Protection by diversity at reduced quality," Bell Labs, Memo for Record (not archived), June 1978.

[10] P.A. Chou, S. Mehrotra, and A. Wang, "Multiple description decoding of overcomplete expansions using projections onto convex sets," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1999, pp. 72-81.

[11] D.-M. Chung and Y. Wang, "Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 6, pp. 895-908, Sept. 1999.

[12] J.H. Conway and N.J.A. Sloane, "Fast quantizing and decoding algorithms for lattice quantizers and codes," *IEEE Trans. Inform. Theory*, vol. 28, no. 2, pp. 227-232, Mar. 1982.

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

[14] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA: SIAM, 1992.

[15] S. Deering and R. Hinden. Internet Protocol, version 6 (IPv6) specification. Network Working Group Request for Comments 1883, Dec. 1995. Available FTP: ftp.isi.edu/in-notes/rfc1883.txt.

[16] S.N. Diggavi, N.J.A. Sloane, and V.A. Vaishampayan, "Design of asymmetric multiple description lattice vector quantizers," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2000, pp. 490-499.

[17] P.L. Dragotti, J. Kovačević, and V.K. Goyal, "Quantized oversampled filter banks with erasures," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2001, pp. 173-182.

[18] P.L. Dragotti, S.D. Servetto, and M. Vetterli, "Analysis of optimal filter banks for multiple description coding," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2000, pp. 323-332.

[19] R.J. Duffin and A.C. Schaeffer, "A class of nonharmonic Fourier series," *Trans. Amer. Math. Soc.*, vol. 72, pp. 341-366, 1952.

[20] M. Effros, "Optimal modeling for complex system design," *IEEE Signal Processing Mag.*, vol. 15, pp. 51-73, Nov. 1998.

[21] M. Effros and A. Goldsmith, "Capacity definitions and coding strategies for general channels with receiver side information," in *Proc. IEEE Int. Symp. Information Theory*, Cambridge, MA, Aug. 1998, p. 39.

[22] A.A. El Gamal and T.M. Cover, "Achievable rates for multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 28, pp. 851-857, Nov. 1982.

[23] W.H.R. Equitz and T.M. Cover, "Successive refinement of information," *IEEE Trans. Inform. Theory*, vol. 37, pp. 269-275, Mar. 1991. See also [24].

[24] W.H.R. Equitz and T.M. Cover, "Addendum to 'Successive refinement of information'," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1465-1466, July 1993.

[25] T. Ericson and V. Ramamoorthy, "Modulo-PCM: A new source coding scheme," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Washington, DC, Apr. 1979, pp. 419-422.

[26] M. Fleming and M. Effros, "Generalized multiple description vector quantization," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1999, pp. 3-12.

[27] Y. Frank-Dayan and R. Zamir, "Universal lattice-based quantizers for multiple descriptions," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2000, pp. 500-509.

[28] A. Gersho, "The channel splitting problem and modulo-PCM coding," Bell Labs Memo for Record (not archived), Oct. 1979.

[29] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*. Boston, MA: Kluwer, 1992.

[30] N. Gogate, D. Chung, S.S. Panwar, and Y. Wang, "Supporting image/video applications in a multihop radio environment using route diversity and multiple description coding," *IEEE Trans. Circuits Syst. Video Technol.*, submitted for publication.

[31] N. Gogate and S.S. Panwar, "Supporting video/image applications in a mobile multihop radio environment using route diversity," in *Proc. IEEE Int. Conf. Communications*, Vancouver, BC, June 1999, vol. 3, pp. 1701-1706.

[32] V.K. Goyal, "Transform coding with integer-to-integer transforms," *IEEE Trans. Inform. Theory*, vol. 46, pp. 465-473, Mar. 2000.

[33] V.K. Goyal, *Single and Multiple Description Transform Coding with Bases and Frames*. Philadelphia, PA: SIAM, 2001.

[34] V.K. Goyal, "Theoretical foundations of transform coding," *IEEE Signal Processing Mag.*, vol. 18, pp. 9-21, Sept. 2001.

[35] V.K. Goyal, J.A. Kelner, and J. Kovačević, "Multiple description vector quantization with a coarse lattice," *IEEE Trans. Inform. Theory*, to be published.

[36] V.K. Goyal and J. Kovačević, "Optimal multiple description transform coding of Gaussian vectors," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar.-Apr. 1998, pp. 388-397.

[37] V.K. Goyal and J. Kovačević, "Generalized multiple description coding with correlating transforms," *IEEE Trans. Inform. Theory*, vol. 47, Sept. 2001.

[38] V.K. Goyal, J. Kovačević, R. Arean, and M. Vetterli, "Multiple description transform coding of images," in *Proc. IEEE Int. Conf. Image Processing*, Chicago, IL, Oct. 1998, vol. 1, pp. 674-678.

[39] V.K. Goyal, J. Kovačević, and J.A. Kelner, "Quantized frame expansions with erasures," *Appl. Comput. Harmon. Anal.*, vol. 10, no. 3, pp. 203-233, May 2001.

[40] V.K. Goyal, J. Kovačević, and M. Vetterli, "Multiple description transform coding: Robustness to erasures using tight frame expansions," in *Proc. IEEE Int. Symp. Information Theory*, Cambridge, MA, Aug. 1998, p. 408.

[41] V.K. Goyal, J. Kovačević, and M. Vetterli, "Quantized frame expansions as source-channel codes for erasure channels," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1999, pp. 326-335.

[42] V.K. Goyal, M. Vetterli, and N.T. Thao, "Quantized overcomplete expansions in $\mathbb{R}^N$: Analysis, synthesis, and algorithms," *IEEE Trans. Inform. Theory*, vol. 44, pp. 16-31, Jan. 1998.

[43] R.M. Gray, *Source Coding Theory*. Boston, MA: Kluwer, 1990.

[44] R.M. Gray and D.L. Neuhoff, "Quantization," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2325-2383, Oct. 1998.

[45] R.M. Gray and A.D. Wyner, "Source coding for a simple network," *Bell Syst. Tech. J.*, vol. 53, no. 9, pp. 1681-1721, Nov. 1974.

[46] R. Han, P. Bhagwat, R. LaMaire, T. Mummert, V. Perret, and J. Rubas, "Dynamic adaptation in an image transcoding proxy for mobile web browsing," *IEEE Pers. Commun.*, vol. 5, pp. 8-17, Dec. 1998.

[47] J. Hong, "Discrete Fourier, Hartley, and cosine transforms in signal processing," Ph.D. dissertation, Columbia Univ., 1993.

[48] A. Ingle and V.A. Vaishampayan, "DPCM system design for diversity systems with applications to packetized speech," *IEEE Trans. Speech Audio Processing*, vol. 3, , pp. 48-57, Jan. 1995.

[49] H. Jafarkhani and V. Tarokh, "Multiple description trellis-coded quantization," *IEEE Trans. Commun.*, vol. 47, pp. 799-803, June 1999.

[50] N.S. Jayant, "Subsampling of a DPCM speech channel to provide two 'self-contained' half-rate channels," *Bell Syst. Tech. J.*, vol. 60, no. 4, pp. 501-509, Apr. 1981.

[51] N.S. Jayant, E.Y. Chen, J.D. Johnston, S.R. Quackenbush, S.M. Dorward, K. Thompson, R.L. Cupo, J.-D. Wang, C.-E.W. Sundberg, and N. Seshadri, "The AT&T in-band adjacent channel system for digital audio broadcasting," in *Proc. Int. Symp. Digital Audio Broadcasting*, Toronto, ON, Mar. 1994, pp. 254-267.

[52] N.S. Jayant and S.W. Christensen, "Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation procedure," *IEEE Trans. Commun.*, vol. 29, pp. 101-109, Feb. 1981.

[53] W. Jiang and A. Ortega, "Multiple description coding via polyphase transform and selective quantization," in *Proc. SPIE Conf. Visual Commun. and Image Processing*, San Jose, CA, 1999, vol. 3653, pp. 998-1008.

[54] W. Jiang and A. Ortega, "Multiple description speech coding for robust communication over lossy packet networks," in *Proc. IEEE Int. Conf. Multimedia & Expo*, New York, NY, July-Aug. 2000, vol. 1, pp. 444-447.

[55] J.A. Kelner, V.K. Goyal, and J. Kovacevic, "Multiple description lattice vector quantization: Variations and extensions," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2000, pp. 480-489.

[56] C.-S. Kim and S.-U. Lee, "Multiple description motion coding algorithm for robust video transmission," in *Proc. IEEE Int. Symp. Circuits and Systems*, Lausanne, Switzerland, May 2000, vol. 4, pp. 717-720.

[57] V. Koshelev, "Multilevel source coding and data-transmission theorem," in *Proc. VII All-Union Conf. Theory of Coding and Data Transmission*, Vilnius, USSR, 1978, pt. 1, pp. 85-92.

[58] V. Koshelev, "Hierarchical coding of discrete sources," *Probl. Peredachi. Inf.*, vol. 16, no. 3, pp. 31-49, 1980.

[59] V. Koshelev, "An evaluation of the average distortion for discrete scheme of sequential approximation," *Probl. Peredachi. Inf.*, vol. 17, no. 3, pp. 20-33, 1981.

[60] G. Kubin and W.B. Kleijn, "Multiple-description coding (MDC) of speech with an invertible auditory model," in *Proc. IEEE Workshop Speech Coding*, Porvoo, Finland, June 1999, pp. 81-83.

[61] K.-W. Lee, R. Puri, T. Kim, K. Ramchandran, and V. Bharghavan, "An integrated source coding and congestion control framework for video streaming in the Internet," in *Proc. IEEE INFOCOM*, Tel Aviv, Israel, Mar. 2000, vol. 2, pp. 747-756.

[62] W.S. Lee, M.R. Pickering, M.R. Frater, and J.F. Arnold, "A robust codec for transmission of very low bit-rate video over channels with bursty errors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 1403-1412, Dec. 2000.

[63] T. Linder, R. Zamir, and K. Zeger, "The multiple description rate region for high resolution source coding," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar.-Apr. 1998, pp. 149-158.

[64] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in *Proc. ACM SIGCOMM*, Stanford, CA, Aug. 1996, pp. 117-130.

[65] S. Mehrotra and P.A. Chou, "On optimal frame expansions for multiple description quantization," in *Proc. IEEE Int. Symp. Information Theory*, Sorrento, Italy, June 2000, p. 176.

[66] A.C. Miguel, A.E. Mohr, and E.A. Riskin, "SPIHT for generalized multiple description coding," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, Oct. 1999, vol. 3, pp. 842-846.

[67] S.E. Miller, "New transmission configuration," Bell Labs, lab notebook #55637, May 1978.

[68] S.E. Miller, "Fail-safe transmission without standby facilities," Bell Labs, Tech. Rep. TM80-136-2, Aug. 1980.

[69] A.E. Mohr, E.A. Riskin, and R.E. Ladner, "Generalized multiple description coding through unequal loss protection," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, Oct. 1999, vol. 1, pp. 411-415.

[70] A.E. Mohr, E.A. Riskin, and R.E. Ladner, "Unequal loss protection: Graceful degradation of image quality over packet erasure channels through forward error correction," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 819-828, June 2000.

[71] M.T. Orchard, Y. Wang, V. Vaishampayan, and A.R. Reibman, "Redundancy rate-distortion analysis of multiple description coding using pairwise correlating transforms," in *Proc. IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, vol. I, pp. 608-611.

[72] L. Ozarow, "On a source-coding problem with two channels and three receivers," *Bell Syst. Tech. J.*, vol. 59, no. 10, pp. 1909-1921, Dec. 1980.

[73] H.C. Papadopoulous and C.-E.W. Sundberg, "Simultaneous broadcasting of analog FM and digital audio signals by means of adaptive precanceling techniques," *IEEE Trans. Commun.*, vol. 46, pp. 1233-1242, Sept. 1998.

[74] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.

[75] D.A. Patterson, G. Gibson, and R.H. Katz, "A case for redundant arrays of inexpensive disks," in *Proc. ACM SIGMOD*, Chicago, IL, June 1988, pp. 109-116.

[76] R. Puri and K. Ramchandran, "Multiple description source coding using forward error correction," in *Conf. Rec. 33rd Asilomar Conf. Sig., Sys., & Computers*, Pacific Grove, CA, Oct. 1999, vol. 1, pp. 342-346.

[77] M. Quirk, "Diversity coding for communication systems," Bell Labs, Engineer's Notes (not archived), Dec. 1979.

[78] K. Ramchandran, A. Ortega, M. Uz, and M. Vetterli, "Multiresolution broadcast for digital HDTV using joint source/channel coding," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 6-23, Jan. 1993.

[79] S. Rangan and V.K. Goyal, "Recursive consistent estimation with bounded noise," *IEEE Trans. Inform. Theory*, vol. 47, pp. 457-464, Jan. 2001.

[80] A.R. Reibman, H. Jafarkhani, Y. Wang, M.T. Orchard, and R. Puri, "Multiple description coding for video using motion compensated prediction," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, Oct. 1999, vol. 3, pp. 837-841.

[81] D.O. Reudink, "The channel splitting problem with interpolative coders," Bell Labs, Tech. Rep. TM80-134-1, Oct. 1980.

[82] B. Rimoldi, "Successive refinement of information: Characterization of the achievable rates," *IEEE Trans. Inform. Theory*, vol. 40, pp. 253-259, Jan. 1994.

[83] D.G. Sachs, A. Raghavan, and K. Ramchandran, "Wireless image transmission using multiple-description based concatenated codes," in *Proc. SPIE Image Video Processing*, San Jose, CA, Jan. 2000, vol. 3974, pp. 300-311.

[84] A. Said and W.A. Pearlman, "A new fast and efficient codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243-250, June 1996.

[85] M. Schwartz, *Information Transmission, Modulation, and Noise*, 4th ed. New York: McGraw-Hill, 1990.

[86] S.D. Servetto and K. Nahrstedt, "Video streaming over the public internet: Multiple description codes and adaptive transport protocols," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, Oct. 1999, vol. 3, pp. 85-89.

[87] S.D. Servetto and K. Nahrstedt, "Broadcast-quality video over IP," *IEEE Trans. Multimedia*, vol. 3, pp. 162-173, Mar. 2001.

[88] S.D. Servetto, K. Ramchandran, V. Vaishampayan, and K. Nahrstedt, "Multiple description wavelet based image coding," *IEEE Trans. Image Processing*, vol. 9, pp. 813-826, May 2000.

[89] S.D. Servetto, V.A. Vaishampayan, and N.J.A. Sloane, "Multiple description lattice vector quantization," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1999, pp. 13-22.

[90] C.E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE Int. Conv. Rec., part 4*, vol. 7, pp. 142-163, 1959. Reprinted with changes in *Information and Decision Processes*, R. E. Machol, Ed. New York: McGraw-Hill, 1960, pp. 93-126.

[91] R. Singh, A. Ortega, L. Perret, and W. Jiang, "Comparison of multiple description coding and layered coding based on network simulations," in *Proc. SPIE Image Video Proc.*, San Jose, CA, Jan. 2000, pp. 929-939.

[92] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG2000 still image compression standard," *IEEE Signal Processing Mag.*, vol. 18, pp. 36-58, Sept. 2001.

[93] M. Srinivasan, "Iterative decoding of multiple descriptions," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1999, pp. 463-472.

[94] P. Subrahmanya and T. Berger, "Multiple descriptions encoding of images," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1997, p. 470.

[95] B.E. Usevitch, "A tutorial on modern lossy wavelet image compression: Foundations of JPEG 2000," *IEEE Signal Processing Mag.*, vol. 18, pp. 22-35, Sept. 2001.

[96] V. Vaishampayan and A.A. Siddiqui, "Speech predictor design for diversity communication systems," in *Proc. IEE Workshop Speech Coding for Telecommunications,* Annapolis, MD, Sept. 1995, pp. 69-70.

[97] V.A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Trans. Inform. Theory*, vol. 39, pp. 821-834, May 1993.

[98] V.A. Vaishampayan and J.-C. Batllo, "Asymptotic analysis of multiple description quantizers," *IEEE Trans. Inform. Theory*, vol. 44, pp. 278-284, Jan. 1998.

[99] V.A. Vaishampayan and J. Domaszewicz, "Design of entropy-constrained multiple-description scalar quantizers," *IEEE Trans. Inform. Theory*, vol. 40, pp. 245-250, Jan. 1994.

[100] V.A. Vaishampayan and S. John, "Balanced interframe multiple description video compression," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, Oct. 1999, vol. 3, pp. 812-816.

[101] V.A. Vaishampayan, N.J.A. Sloane, and S.D. Servetto, "Multiple description vector quantization with lattice codebooks: Design and analysis," *IEEE Trans. Inform. Theory*, to be published.

[102] N. Varnica, M. Fleming, and M. Effros, "Multi-resolution adaptation of the SPIHT algorithm for multiple description," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2000, pp. 303-312.

[103] R. Venkataramani, G. Kramer, and V.K. Goyal, "Multiple description coding with many channels," *IEEE Trans. Inform. Theory*, in preparation.

[104] R. Venkataramani, G. Kramer, and V.K. Goyal, "Bounds on the achievable region for certain multiple description coding problems," in *Proc. IEEE Int. Symp. Information Theory*, Washington, DC, June 2001, p. 148.

[105] R. Venkataramani, G. Kramer, and V.K. Goyal, "Successive refinement on trees: A special case of a new MD coding region," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2001, pp. 293-301.

[106] M. Vetterli, "Wavelets, approximation and compression," *IEEE Signal Processing Mag.*, vol. 18, pp. 59-73, Sept. 2001.

[107] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[108] J. Walrand, *Communication Networks: A First Course*, 2nd ed. Boston, MA: McGraw-Hill, 1998.

[109] Y. Wang, M.T. Orchard, and A.R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. IEEE Workshop on Multimedia Signal Processing*, Princeton, NJ, June 1997, pp. 419-424.

[110] Y. Wang, M.T. Orchard, V. Vaishampayan, and A.R. Reibman, "Multiple description coding using pairwise correlating transforms," *IEEE Trans. Image Processing*, vol. 10, pp. 351-366, Mar. 2001.

[111] H.S. Witsenhausen, "On source networks with minimal breakdown degradation," *Bell Syst. Tech. J.*, vol. 59, no. 6, pp. 1083-1087, July-Aug. 1980.

[112] H.S. Witsenhausen, "An achievable region for the breakdown degradation problem with multiple channels," Bell Labs, Tech. Rep. TM81-11217-3, Jan. 1981.

[113] H.S. Witsenhausen and A.D. Wyner, "Source coding for multiple descriptions II: A binary source," *Bell Syst. Tech. J.*, vol. 60, no. 10, pp. 2281-2292, Dec. 1981.

[114] J.K. Wolf, A.D. Wyner, and J. Ziv, "Source coding for multiple descriptions," *Bell Syst. Tech. J.*, vol. 59, no. 8, pp. 1417-1426, Oct. 1980.

[115] S.-M. Yang and V.A. Vaishampayan, "Low-delay communication for Rayleigh fading channels: An application of the multiple description quantizer," *IEEE Trans. Commun.*, vol. 43, pp. 2771-2783, Nov. 1995.

[116] X. Yang and K. Ramchandran, "Optimal subband filter banks for multiple description coding," *IEEE Trans. Inform. Theory*, vol. 46, pp. 2477-2490, Nov. 2000.

[117] R. Zamir, "Gaussian codes and Shannon bounds for multiple descriptions. *IEEE Trans. Inform. Theory*, vol. 45, pp. 2629-2635, Nov. 1999. See [63] for generalizations to locally quadratic distortion measures.

[118] A. Zandi, J.D. Allen, E.L. Schwartz, and M. Boliek, "CREW: Compression with reversible embedded wavelets," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1995, pp. 212-221.

[119] Z. Zhang and T. Berger, "New results in binary multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 33, no. 4, pp. 502-521, July 1987.