Lossless Coding for Audio Discs*

PETER CRAVEN, *AES Member*

Algol Applications, Wantage, UK

AND

MICHAEL GERZON, *AES Fellow (deceased)*

Oxford, UK

Strategies are presented that are being used to achieve efficient lossless compression or packing of PCM audio waveform data, allowing storage and transmission at greatly reduced data rates without any alteration of the signal. The disadvantages of simpler difference schemes and the benefits of more advanced schemes using IIR prediction with Huffman coding are explained and described, particularly with regard to the unique requirements of the future high-quality audio disc (HQAD) standard using high-density CD media.

0 INTRODUCTION

Although the proposed new high-density Compact Disc (CD) formats, such as the Digital Versatile Disc (DVD), offer a storage capacity increase of nearly an order of magnitude over conventional CDs, new aspirations in terms of

Number of bits per sample
Sampling frequency
Number of channels

have more than used up the extra capacity available, and lossless data compression of pulse-code-modulated (PCM) audio has been proposed [1] as a means of allowing more of the desired goals to be met within the available constraints.

In fact lossless compression provides the key not only to satisfying the demands for an increased data rate, but also to making it practical to design a system that will handle a wide variety of requirements very simply. We shall also show how lossless compression reduces the need for a complicated system of "flags," and in practice this may prove to be a greater operational advantage than the increased data rate.

Unlike "lossy compression," where the encoder throws away data that it thinks are not psychoacoustically important, the encoder for lossless compression does not throw away any of the data. It merely packs them more efficiently into the available data channel, and the decoder can recover an exact bit-for-bit copy of the original. The ARA document [1] prefers the term "packing" to emphasize this difference, and we shall henceforth use this terminology.

This paper provides a simple introduction to packing (lossless data compression) in an audio context. Section 1 explains the principles, and Section 2 explains how the algorithms can be tailored to a particular medium such as the high-density CD. Section 3 explores the ability to design a very user-transparent system using packing, versatile in the types of data it can convey without a lot of flags. Section 4 deals with subjective aspects, and Section 5 provides a brief description of the various schemes that have been developed by the various research teams around the world.

One important difference between packing and lossy compression is that the data rate after packing is not fixed, but depends on how much redundancy there is in the original signal. Some signals can be packed more tightly into a smaller data rate than others, so lossless coding will by default produce a variable data rate and the greatest advantage in playing time will be obtained on media that can handle a variable rate bit stream.

However, in order to cater for fixed-rate CD standards, and also because the high-density CD has limitations on the available maximum data read rate, it is important to design packing systems that achieve the

* Presented at "Audio for New Media," AES UK Conference, London, UK, 1996 March 25–26.

minimum possible data rate even during hard-to-compress passages of audio.

1 PRINCIPLES OF PACKING

Many readers will already be familiar with packing in a computer context, with utilities such as PKZIP, which compress a file of data down to typically 50% of its original length.

Standard binary-weighted PCM is just one of a number of possible formats for representing digital data, and is not necessarily the most efficient in any particular case. Packing can be regarded as a reformatting of the data for greater economy, based on spotting redundancy in the standard PCM representation and converting to a format where the redundancy is minimized.

The basic principle in packing is to take an input waveform that is represented by a large word length with many bits, and to transform it in a reversible fashion into a waveform represented by a smaller word length, which can be transmitted using fewer bits at a lower data rate. Then at the decoding end, one uses the inverse transformation to restore the original longer word length waveform from the shorter transmitted words.

1.1 Simple Redundancy Elimination

Consider the following portion of a nominally 20-bit digital audio signal:

Sample No.	Binary Value
1	00000000010000110000
2	00000000011000010000
3	00000000011001100000
4	00000000010011110000
5	00000000001000110000
6	11111111111011100000
7	11111111101111010000
8	11111111100111110000
9	11111111100110100000
10	11111111101100010000
11	11111111110111010000
12	00000000000100100000

This is actually a 4-kHz sine wave at 50 dB below peak, sampled at 48 kHz. The 12 samples occupy a total of 240 bits. What opportunities can we see for transmitting this signal in less than 240 bits?

First we notice that the 4 least significant bits (LSBs) are zero in each sample—clearly we have a 16-bit signal, even though the data path is 20 bit wide. An encoder that detects this fact and automatically inserts a header at the start to say that we are only going to transmit the top 16 bit will save 20% on the data rate.

Next we notice that the top 9 bit of each sample are either all zeros or all ones, as one expects for any low-level signal. Thus we can decline to transmit the top 8 bit and tell the decoder that these are to be reconstituted by replicating the ninth bit. We are now transmitting only 8 bit per sample, a saving of 60%.

In this example we have used the limited dynamic range and bit resolution of the original 20-bit words to reduce the transmitted word length from 20 to 8 bit per sample.

In practice we will divide the audio data stream into blocks. A block size of around 500 samples (or about 10 ms) is typical. Each block is processed separately and has its own header to tell the decoder how many most significant bits (MSBs) and LSBs have been stripped off, and thus need reconstituting. Obviously, the number of MSBs that can be stripped off is determined by the loudest transient in the blocks. If the block length is too long, we will fail to take advantage of momentary periods of relative silence. If the block length is too short, the overheads of transmitting the header information more often will outweigh the other advantages.

This is about as far as we can go in a simple system. It will provide a useful extension of playing time for classical music with quiet passages, but it will not help much with rock or pop music heavily compressed up to 0 dB, and it is unlikely to reduce the peak data rate significantly with any type of music.

1.2 Prediction Methods

1.2.1 Simple nth-Order Predictors

The 16-bit numbers in the previous example have the decimal values +67, +97, +102, +79, +35, −18, −67, −97, −102, −79, −35, +18, and the differences between successive pairs of numbers are +30, +5, −23, −44, −53, −49, −30, −5, +23, +44, +53.

If we were to transmit the first number (+67) in the header and then just the differences, clearly the original numbers could be reconstituted by the decoder. As the differences are smaller than the original numbers, they can be transmitted in fewer bits—7 bit instead of 8 bit in this instance. This is not a spectacular improvement, but had the sine wave been at 400 Hz rather than at 4 kHz, we could have transmitted the differences in 4 bit, saving a factor of 2 in data rate.

This is the very simplest "predictive encode–decode" process, and its block diagram is shown in Fig. 1. The symbol z^{-1} is used to denote a delay by one sample. Supposing that the system is already primed and that we are at the second sample instant, the input value is +97. We use the previous sample value (+67) as the "predicted" value of the current sample, and we transmit the "prediction error" (+30). At the decode end, the transmitted value (+30) is added back to the previous

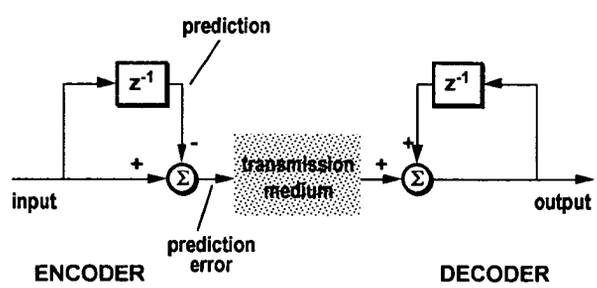


Fig. 1. Simple first-order predictive encode–decode process, using previous sample as predicted value of current sample. z^{-1}—one-sample delay.

output sample (+67) so as to reproduce the current sample value, +97.

More advanced predictive schemes can now be constructed by replacing the one-sample delay element, denoted by z^{-1}, by a more general "prediction filter," as in Fig. 2(a). The encoder of Fig. 1 is a digital differentiator with a transfer function of $(1 - z^{-1})$. One well-known generalization of this is the nth order predictor, with encode transfer function $(1 - z^{-1})^n$. This gives us a whole family of predictive encoders, with $n = 0$ being the trivial case of transmitting the input signal verbatim, $n = 1$ transmitting the differences of successive samples, and $n = 2$ transmitting the second differences, that is, the differences of the differences. It is instructive to look at the difference signal that gets transmitted for different values of n (see Table 1). For this signal the optimum value of n is 4, giving a maximum difference of only 10, which can be transmitted in 5 bit. For centuries mathematicians have been fascinated by difference tables such as this, but from an audio point of view we can say that it is the *high-frequency* content of the signal that limits the order of the predictor that can be used. This can be illustrated in Table 2 by the difference table for a sine wave at the Nyquist frequency of amplitude 1 LSB.

Clearly, there is no point in continuing as the difference that we need to transmit gets larger by a factor of 2 each time we increase n by 1. Returning to Table 1, if we had started in with a perfect 4-kHz sine wave, the differences would have gone down to 1 by the time we reached $n = 7$, but because we started with a sine wave quantized to 16 bit, the high-frequency components of

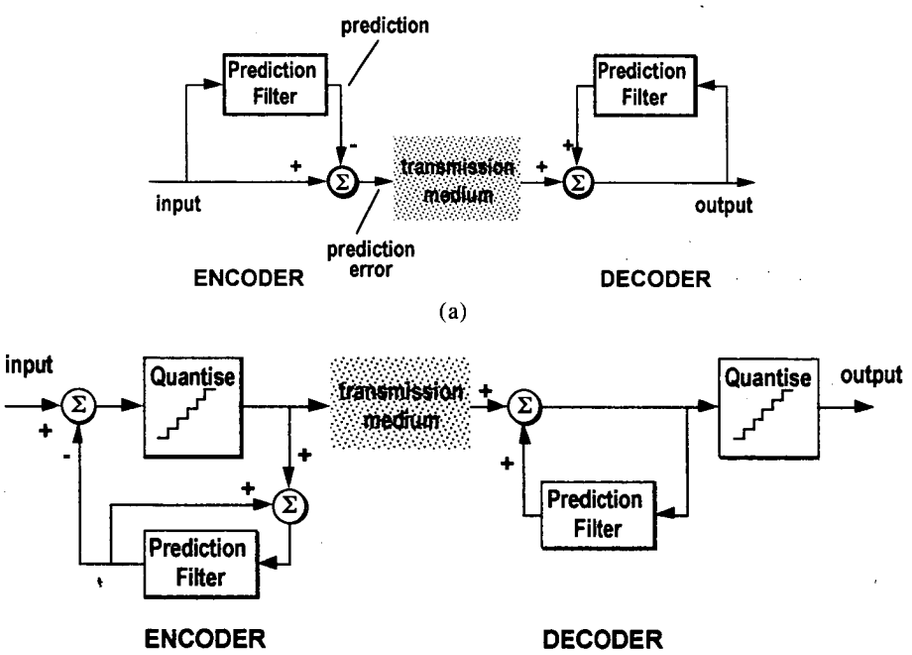


Fig. 2. Lossless coding using difference of input and its value predicted via a general prediction filter. (a) Simple architecture suitable for integer predictors. (b) Possible strategy for quantizing the output of a noninteger predictor. Assuming that the transmission medium is inherently quantized, the quantizer in the encoder ensures that the encode predictor is driven by the same signal as the decode predictor.

Table 1. Differences for 4-kHz signal.

n																		
0	+67,	+97,	+102,	+79,	+35,	−18,	−67,	−97,	−102,	−79,	−35,	+18						
1		+30	+5	−23	−44	−53	−49	−30	−5	+23	+44	+53						
2			−25	−28	−21	−9	+4	+19	+25	+28	+21	+9						
3				−3	+7	+12	+13	+15	+6	+3	−7	−12						
4					+10	+5	+1	+2	−9	−3	−10	−5						
5						−5	−4	+1	−11	+6	−7	+5						
6							+1	+5	−12	+17	−13	+12						
7								+4	−17	+29	−30	+25						

Table 2. Differences for high-frequency signal at Nyquist frequency.

n																		
0	+1	−1	+1	−1	+1	−1	+1	−1	+1	−1	+1	−1						
1		−2	+2	−2	+2	−2	+2	−2	+2	−2	+2	−2						
2			+4	−4	+4	−4	+4	−4	+4	−4	+4	−4						

the quantization noise are amplified by higher order predictors, to the point where they are larger than the signal. For such reasons it is rarely worth going much beyond $n = 3$ on real audio signals.

If we had added dither to the 4-kHz sine wave, the situation would have been even worse. The present authors would never deny the need, from a perceived audio quality point of view, to add dither when a signal is quantized or requantized, but a lossless compression system will insist on reproducing the dithered signal with bit-for-bit accuracy, and if an unnecessarily large amount of dither is applied, this will use up information-carrying capacity, and so will be expensive as well as adding to the background noise heard by the listener.

Table 2 warns us against using a fixed predictor. Even with $n = 1$, the data rate will be increased if the signal consists predominantly of high frequencies. As with PKZIP, for example, the algorithm needs to monitor its own performance and be prepared to switch itself off if it is doing harm rather than good. In practice, one would make the algorithm adaptive, such as with the encoder trying out $n = 0, 1, 2, 3$ and selecting whichever is best on a block-by-block basis.

1.2.2 More Complicated Predictors

Our goal, as in the last section, is to arrange for the prediction filter to predict the next sample as accurately as possible, so as to minimize the number of bits required to transmit the difference signal. In the theory of linear prediction it is well known that, in order to achieve this, the frequency response of the encoder must be the inverse of the spectrum of the input signal, so that the transmitted difference signal will have a flat or white spectrum [2].

The simple integer-coefficient predictors of the last section can manage only constant (upward) slopes of 6, 12, and 18 dB per octave over most of the audio band for $n = 1$, 2, and 3. This does not allow them to cope well with, say, loud musical fundamentals extending up to 2 kHz, plus a lower level of detail extending with more or less constant energy right to the top of the audio band. See the example of Fig. 3(a)–(c), which shows a hypothetical audio signal spectrum with high level below 2 kHz, the frequency responses of nth-order integer prediction systems, and the resulting spectra of the encoded transmitted signals, which remain highly nonflat, contrary to the requirements of the optimal prediction system, which are for a flat or white transmitted spectrum.

This sort of signal can be handled well by using an IIR (recursive) prediction filter. The question is, how complicated is it worth making it?

Even the simple case of 2nd-order IIR prediction filters, with numerator and denominator coefficients quantized as crudely as in steps of 0.25, gives a wide variety of spectral equalization characteristics for the prediction error signal, as illustrated in Fig. 3(d). These examples already match a much wider variety of signal spectral statistics than simple integer predictors. Every 6-dB reduction in level of the encoded transmitted signal reduces the data rate required to transmit it by 1 bit per sample. For example, characteristic 1 in Fig. 3(d) reduces the data rate by 3–4 bit for signals concentrated below 5 kHz with high energy around 3 kHz, whereas characteristic 6 in Fig. 3(d) reduces the data rate by 2.5–3 bit for signals concentrated around 8 kHz whose data rate is not reduced at all by the integer predictors of Fig. 3(b). Characteristic 7 in Fig. 3(d) reduces the data rate by as much as 2.5 bit for low-level signals with heavy high-frequency noise shaping, whose data rate is made much worse by conventional integer predictors.

The present authors have focused on third-order IIR filters with finer coefficient quantization as a reasonable compromise, which permits most plausible musical spectral envelopes to be followed to an adequate degree of accuracy. (However, see Section 2.5 for a discussion of the exceptional case of sharply band-limited signals.) There is also the possibility of using different orders for numerator and denominator, such as a third-order numerator and fourth-order denominator.

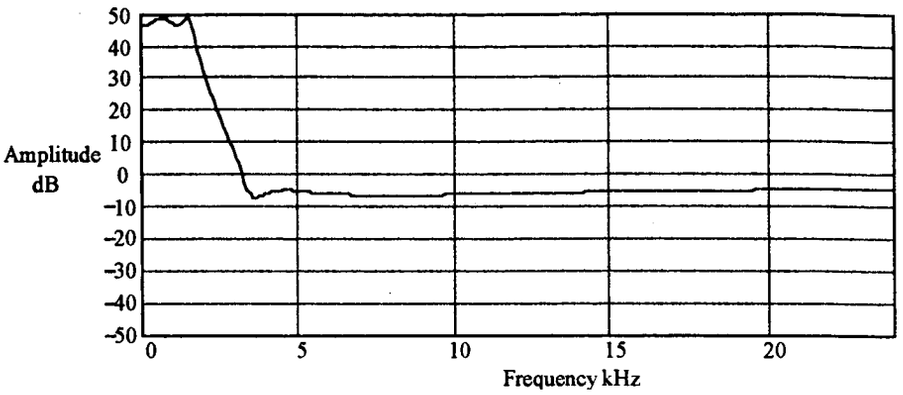
In principle one can achieve better prediction by resolving the input spectrum finely and tuning the prediction filter to follow individual spectral lines. However, this has to be balanced against the complexity involved. Also, the filters will in general need to be retuned at the start of every new block in order to follow the changing musical patterns, and the tuning information needs to be transmitted to the decoder so that its prediction filter can be kept in step with the encoder's. The extra data overheads of conveying this retuning information can very easily outweigh any savings in data rate for the difference signal.

Various theoretical studies based on detailed statistical modeling of the fine structure of audio signal spectra suggest that in the case of most non-band-limited signals, the additional data-rate reduction obtainable from high-order IIR predictors may be quite limited, rarely exceeding an extra 1 bit per sample per channel, and often much less. So it is hard to justify a very complex high-order prediction system that will increase the complexity and cost of decoders in multichannel audio applications.

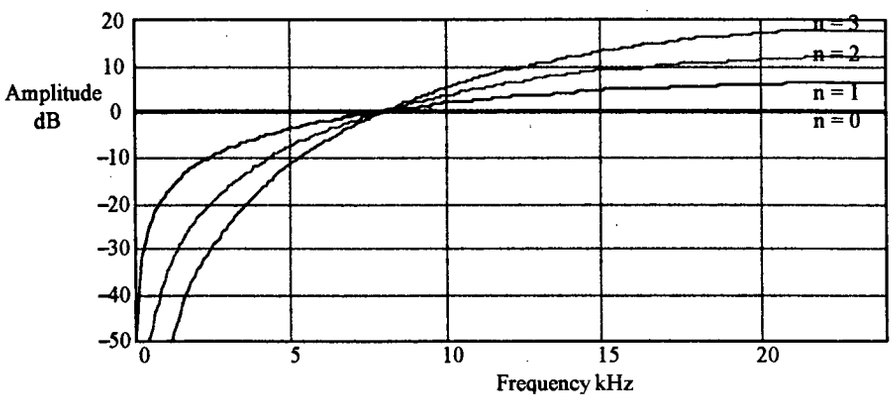
Standard predictors used in the literature [2] often use FIR filters, but these perform less well than IIR filters for the prediction of high-quality audio, giving a generally larger transmitted word length. This is because, as was illustrated in Fig. 3(d), IIR filters of modest order can more accurately match the extremely wide dynamic range often found in the spectrum of high-quality music signals, where some parts of the spectrum can be as much as 60 or 80 dB lower than other parts.

1.2.3 Predictor Quantization

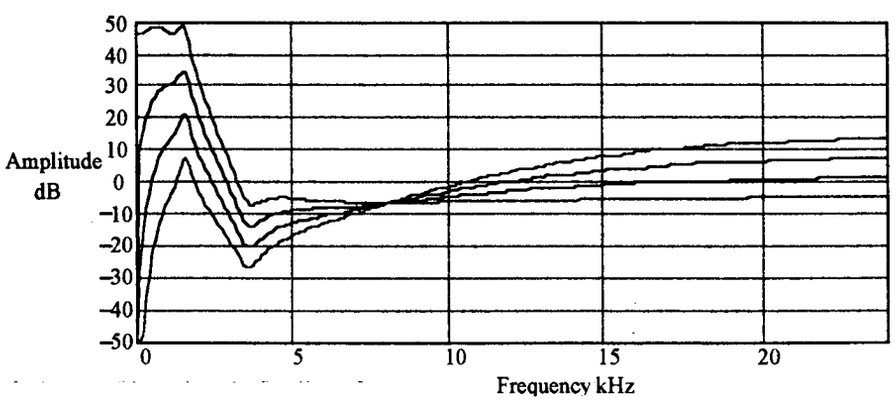
A detail omitted in Fig. 2(a) is that in order to transmit the prediction error signal with a finite data rate, it is necessary that the transmitted signal be quantized to an integer number of LSB levels, as is the input signal. For general predictors with noninteger coefficients, the output is not an integer number of LSBs, but has a fractional value. The standard method in prediction work of quantizing a prediction system (see [2]) is illustrated



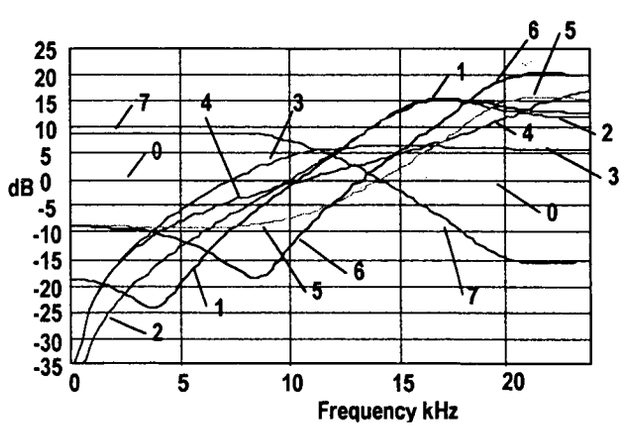
(a)



(b)



(c)



(d)

Fig. 3. Spectra relating to prediction of a hypothetical audio signal. (a) Signal spectrum with main energy concentration at frequencies up to 2 kHz. (b) Transfer function of simple nth-order integer predictor for $n = 0, 1, 2, 3$. (c) Spectrum of prediction error when using predictors of part (b) on hypothetical audio signal spectrum of part (a). No value of n gives a white spectrum, and deviation from flatness represents inefficiency in coding data. (d) Examples of second-order IIR prediction filter characteristics, using coefficients quantized to 0.25, at 48-kHz sampling rate for matching different signal spectral statistics.

in Fig. 2(b). The decoding restores the original signal values for this encoder by the simple expedient of quantizing the output, as described, for example, in [3], [4].

1.2.4 Pitch-Period Extractors

In the case of a single musical instrument, such as a krummhorn emitting a raspy note at 250 Hz, we might have an extremely complicated but repetitive waveform, and in this case a very good prediction of the current sample could be obtained from the previous cycle, that is, 4 ms ago [5]. We estimate that this technique, combined with the spectral techniques described, will allow an improvement of about 1 bit in the prediction accuracy on many signals.

However, it is not likely to be of great use on cymbal crashes and other complex material, and so cannot be considered as a plausible way of reducing the peak data rate. Again, the added encoder and decoder complexity probably does not justify the relatively limited typical extra data-rate reduction obtained.

1.3 Correction Methods

An alternative packing scheme is shown in Fig. 4. Here we use a lossy compression algorithm, and the lossy compressed signal is used by the decoder to reconstruct an approximation to the signal. The encoder then transmits a correction signal, which the decoder adds to the approximation so as to recover a bit-exact copy of the original.

Clearly there is a decision to be made about how accurate the signal furnished by the lossy compression should be. The more accurate it is, the fewer bits are needed to transmit the correction signal. An error of a few LSBs is probably about optimum.

The lossy algorithm could be based on one of the transform-coding schemes used for low-bit-rate consumer applications. The advantage of these schemes is that they are able to allocate a variable number of bits to each frequency component, and thus can deal efficiently with signals having several sharp spectral peaks.

The "psychoacoustic masking" criteria which are used in the commercial embodiments of these algorithms are not, however, appropriate for use within a packing scheme, and in fact a simplified version of the lossy algorithm which just gives a constant and white error spectrum should be used.

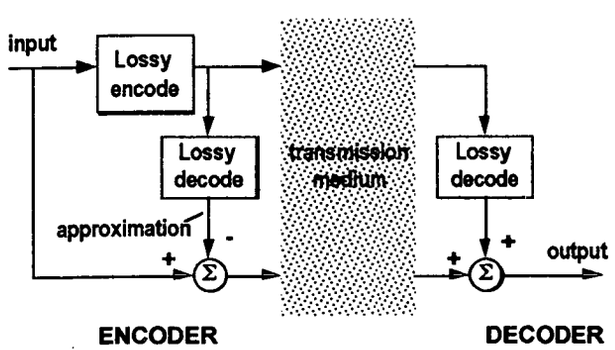


Fig. 4. Lossless coding based on lossy coding plus lossless transmission of error signal.

It will be seen that there is conceptually some similarity between Figs. 4 and 2. In both schemes the encoder and the decoder have to agree on some approximation to the input signal, which is then corrected. The difference is that in Fig. 4 the approximation is transmitted as separate data, whereas in Fig. 2 it is derived entirely from previous samples.

Such schemes based on transform-type coding, however, have more complex decoders than do prediction-type packing systems, and they may not be significantly more efficient in data-rate reduction on real-world audio signals. We are unaware of any commercial scheme of packing using the transform approach.

1.4 Multichannel Issues

Packing is about spotting redundancy and making use of it. If we are presented with an n-channel signal, the first question to ask is whether the channels are genuinely independent.

To take an obvious example, if a mono signal is presented to a stereo encoder as L (left) and R (right) signals, the encoder could detect the fact that the two stereo channels are identical, transmit the left channel, and instruct the decoder to replicate the transmitted signal on both the left and the right outputs. Or if the two channels had different dither, the decoder could transmit L and $(R - L)$. As $(R - L)$ should be a very small signal, it should be possible to transmit it with very little data rate, maybe 1 or 2 bit per sample.

Of course, if the original mono signal was replicated left and right in the analog domain, the $(L - R)$ difference signal could well be much bigger than just dither noise. Amplitude differences, phase shifts, or time delays between the two channels would all put up the data rate, and there could well be interesting surprises in the future when mastering engineers try to encode historical nominally mono material and find a higher data rate than expected.

1.5 Huffman Coding

Whether we use a prediction scheme (Fig. 2) or a correction scheme (Fig. 4), we end up with a difference, or correction, signal, and as stated earlier, it is usual to transmit this signal on a block-by-block basis, using the minimum possible number of bits for each block.

However, audio signals can be peaky, and if most of the samples have an amplitude of, say, 10 LSB, it would be a pity to have to use 8 bit for the whole block just because two or three samples had a magnitude of around 100.

Huffman coding is a popular solution to this problem [6], [7]. The idea is to use a look-up table for the input words to convert them to transmitted code words of varying length, so that commonly occurring input data words should be represented by short output code words, whereas rarely occurring input words may be represented by fairly long code words, so that the average coded word length is small.

Huffman coding is an approximation to ideal entropy coding and embodies the principle that, in an efficient

coding system, the decoder should use each bit of the transmitted stream to make a decision between two possibilities that have roughly equal probability. To take a highly artificial example (see Fig. 5), suppose that half the samples in the block had an amplitude in the range of $-8 \leq x \leq 7$, whereas the other half were more or less uniformly distributed in the rest of the range of $-128 \leq x \leq 127$. In a Huffman coding scheme we would transmit 1 bit to distinguish between the two possibilities, and then code the actual value in 4 bit in the first case, 8 bit in the second.

Numbers in the range of -8 to 7 have thus been encoded in a total of 5 bit, and numbers in the rest of the range, -128 to 127, in 9 bit—an average data rate of 7 bit per sample, compared to 8 bit with straight PCM. However, if a signal were to come along with *all* values outside the range of -8 to 7, we would be using 9 bit with the scheme, compared with 8 bit with ordinary PCM.

This is a situation akin to that of Section 1.2.1 if an inappropriate predictor is used. Huffman decoders tend to be table driven, so a sensible strategy is to have a selection of decoding tables available, and the Huffman encoder can select which one is most appropriate, depending on the distribution of sample values in the current block. (If the sample values were uniformly distributed, ordinary PCM would in fact be best, but this is a special case of a Huffman code, and it can be accommodated in a Huffman decoder by selecting the right table.)

Table 3 gives an example of a code that could be used for a correction signal that was 0 for almost half the time, ± 1 for almost 25% of the time, and otherwise, infrequently, was -3, -2, 2, or 3. The scheme might code this signal with an average of 1.5 bit per sample, a factor of 2 more efficient than PCM binary.

In practice audio waveform data words tend to follow amplitude statistics known as Laplacian, and it is found that Huffman coding of Laplacian statistics can typically reduce the data rate by around 1.5 bit per sample per channel as compared to the simple word-length reduction scheme described earlier. This is a significant saving in data rate in many applications. The price paid is the necessity to use a table-driven Huffman coder and decoder.

1.6 Complete System

Fig. 6 shows the key features of a practical single-channel predictive encoder and decoder incorporating these ideas. First the signal is split into blocks. Each block is examined for trailing zeros as in Section 1.1 and, if necessary, right justified. The shift count is transmitted in the block header so that the decoder can undo this process.

The encoder now needs to choose the best prediction filter for the block, as mentioned at the end of Section 1.2.1. The coefficients for the filter are also transmitted as part of the block header. The difference between the actual signal and the predicted signal is now formed and passed to the encoder. This Huffman-encoded difference signal is transmitted as the "main" data in the block. The block length may vary, depending on the nature of the original signal.

1.7 Filter Initialization

In general the prediction filter used will vary to match the signal statistics block by block to minimize the transmitted signal amplitude, and hence data rate, for each block. But prediction filters contain memory elements, and the "state variables" stored in these memory elements preferably should be transmitted at the start of each block from the encoding filter to the decoding filter such that the output of the latter is identical to the input of the former. These internal filter data at the start of each block are termed "initialization data."

In general, if an IIR filter with nth-order numerator and mth-order denominator is used, this preferably requires the transmission, at the start of each block, of $m + n$ state variables in the delay memories of the filter from the encoder to the decoder as part of the block header information. The higher the order of the filter used, the more of these initialization data are required, which means that using a very high-order IIR filter adds considerably to the extra header data required for each block. This is one reason that orders of numerator and denominator very much higher than 3 or 4 are not practical.

Table 3. Example of Huffman coding.

Sample Value	PCM Code	Huffman Code
−3	101	1100
−2	110	1101
−1	111	100
0	000	0
+1	001	101
+2	010	1110
+3	011	1111

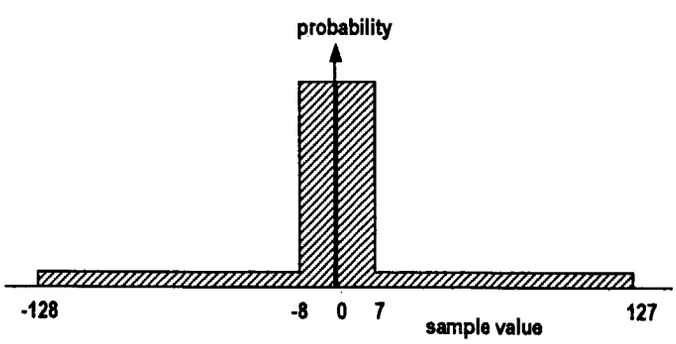


Fig. 5. Hypothetical probability distribution used to illustrate Huffman coding.

2 PACKING FOR THE HIGH-DENSITY AUDIO DISC

2.1 Cost

Looking at the decode side of Fig. 6 first, there is nothing complicated about a Huffman decoder, and the "left shift" box is trivial. The predictor can be as simple or as complicated as we like, but if we restrict ourselves to, say, a third-order IIR filter, the resulting decoder will be extremely simple compared to a decoder for any of the lossy compression systems currently in use. It is entirely feasible to incorporate several channels of such processing into a cheap commerically available digital signal processor (DSP).

The encode side is more complicated, because of the need to choose the filter coefficients and also because of the decision logic (not shown in Fig. 6) required to choose the Huffman coding table.

Early encoders, and encoders for a recordable high-density CD, will probably take an easy option here, for example, trying a small number of preselected prediction filters, which may be selected differently for different kinds of musical material, and choosing whichever is best for a particular block. Algorithms are known for optimizing FIR prediction filters. Although corresponding algorithms are less well understood for IIR prediction filters, it should be possible to incorporate a degree of optimization in the procedure for selecting the encoding filter.

The IIR filter optimization problem is inherently much more difficult than the FIR case, partly because the very virtue of IIR filters—that they have a very wide spectral dynamic range—also means that their optimal coefficients can vary drastically for very small changes of energy in the low-level parts of the spectrum of the audio signal.

As the technology develops, encoders for commercial mastering may well become quite sophisticated using computation-intensive DSP. As always, it is the decoder that needs to be standardized, but it is important to keep options for future encoder ingenuity open as far as possible, so that the data rate is minimized in the future, even on material with unusual spectral statistics. For example, we would suggest that even if early or simplified low-cost encoders merely choose from, say, five standard prediction filters, the five sets of coefficients should *not* be built into the decoder, but should be transmitted explicitly so that future encoders can be more adventurous.

The multichannel case opens up a lot more possibilities for complexity. The stereo–mono case discussed in Section 1.4 requires, at its simplest, a flag to say that the right channel is transmitted either explicitly or as a difference from the left channel. The question is, do we also cater to the case where there is a gain mismatch between the two channels?

In any event, the main complexity will again be in the encoder, and the decoder will merely have to decode the transmitted channels as before and then perform simple matrixing.

With up to eight independent channels proposed for the high-quality audio disc (HQAD) [1], the number of possibilities for using matrixing to minimize the multichannel data rate, making use of partial repetition of data in different channels, grows much larger. In practice there is a tradeoff between system complexity and maximum data-rate reduction for multichannel signals.

2.2 Portability

Lossy compression systems are specified as numerical algorithms, typically involving transforms or quadrature mirror filter band splitting, and it is understood that the arithmetic involved will be executed to finite precision. The rounding errors must of couse be kept within acceptable limits, but no one worries if two decoders, implemented using different hardware, give answers that occasionally differ in the LSB.

Referring to Fig. 4, however, it is essential that the two "lossy decode" elements, in the encoder and the decoder, give bit-exact identical outputs, otherwise the original signal will not be reconstituted.

Similarly in Figs. 2 and 6 the two predictors must give identical outputs. This is not a problem for the simple predictors discussed in Section 1.1, which in-

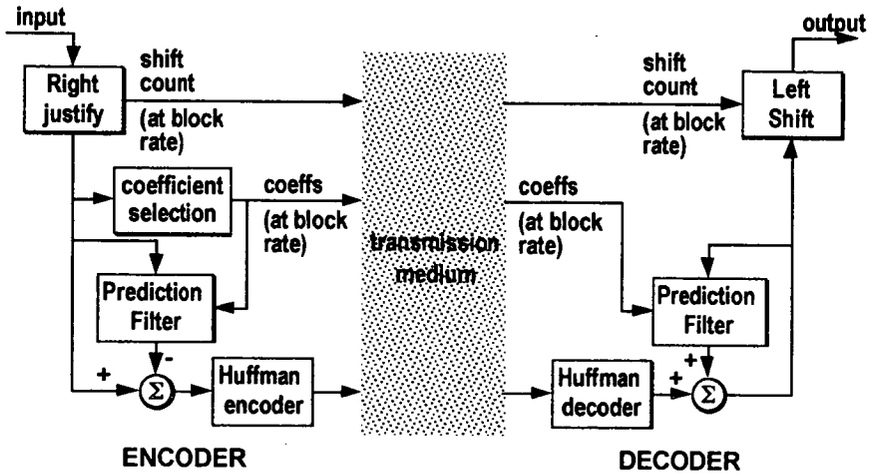


Fig. 6. Schematic of packing system using prediction with Huffman coding of blocks. Some additional housekeeping information at block rate (such as Huffman table selection) is omitted for clarity.

volve only integer arithmetic. Careful design is, however, needed if the more complicated IIR predictors are to give identical outputs when implemented on different hardware platforms.

Clearly, it would not be acceptable to standardize an algorithm that would work "correctly" on only one manufacturer's hardware. This difficulty has in the past somewhat inhibited the adoption of IIR predictors, since traditional IIR filter topologies are highly sensitive to rounding errors and their precise nature. It is necessary to incorporate LSB quantization and rounding into filter topologies in the encoder and decoder in a manner that is easy to implement consistently and without ambiguity on a wide variety of platforms.

2.3 Error Robustness

Referring to the simple decoder in Fig. 1, note that the final output is recirculated and used in the calculation of the next output, and so on. It follows that an error in the transmitted sample will propagate indefinitely, or at least until the end of the current block. With some predictors, including second- and third-order integer predictors, the error may blow up in an alarming manner on successive samples. The use of Huffman coding further complicates matters as it is a variable-length encoding scheme, and an error may cause the decoder to get out of step with the encoder.

Various techniques can be devised to minimize these problems, but in the case of high-density CDs this may not be necessary, as the CD–ROM applications will demand an effective extra layer of error protection. If this is used for audio as well, uncorrected errors will hardly ever occur. This would surely be the preferred situation for an audiophile disc.

In such a case one needs only a detector for catastrophic failure from very gross disc read problems in order to implement temporary muting to prevent loudspeakers and ears from being damaged by noise bursts. Given that the HQAD may well have a dynamic range of order 120 dB or more, peak unwanted noise passages could be very loud indeed, and so should be attenuated or muted.

2.4 Peak Data Rate

Packing has the potential to reduce both peak and average data rates. Both are important, but in the context of the high-density CD it is the peak rate that is often the limiting factor. This is in contrast to packing in the context of hard-disc editors and audio data backup systems, where the total amount of data to be stored is of paramount importance.

In the case of audio sampled at the usual 44.1 or 48 kHz, the authors of the ARA proposal [1] did not claim that packing could guarantee any reduction in the peak data rate, though a reduction of 6 bit per sample (say, 30%) was conservatively claimed as the average for typical music. This is because, in the worst case, say, a loud cymbal crash, the signal has continuous high-energy components at high frequencies, and little advantage can be gained either from prediction or from coding.

However, these figures represent a very worst-case scenario, and typically using Huffman coding, one expects a reduction of the peak data rate on the order of 1 or 2 bit per sample per channel for most natural material that has not been subjected to violent limiting.

At a 96-kHz sampling rate, the ARA proposal quotes [1, table 1] a peak data-rate reduction of 5 bit per sample. This assumes that there is relatively little energy in the upper part (20–48 kHz) of the available audio bandwidth, so that prediction can work effectively, but it also includes data-rate savings from Huffman coding. Here there is an assumption that the combination of very wide dynamic range and wide audio bandwidth makes it less likely that extreme limiter processing of peaks will be employed than would be the case at lower sampling rates.

Of course, one could install a RAM buffer to smooth out the peak demand, but at today's prices one could probably not justify more than 1 Mb in a consumer product, that is, about 1 s of data. If buffer RAM is used to control the peak data rate on a disc, the minimum amount needed in any player should be standardized. One has to bear in mind that if the HQAD becomes widely adopted, it will become a mass consumer medium with a need to minimize the costs of decoding chips.

Given the immense variety of average and peak signal statistics encountered in different styles of music, a simple survey of a few of the more popular styles may fail to reveal extreme problem material, which might occur in particular ethnic music or avant-garde styles or in unusual circumstances with particular instruments played in a particular way.[1] In any case at present the available database of recordings making full use of a 96-kHz sampling rate is extremely limited by the limited current availability of both recorders and microphones, taking full advantage of the available frequency and dynamic range.

Therefore to some extent one has to accept that there may be a low probability, in very exceptional cases, that even a packing system that reduces peak data rates with high efficiency may very occasionally encounter material that exceeds the peak data-rate limitation of an HQAD. Similar problems have occurred in the past with traditional analog media with peak overload on exceptional material.

Provided such cases are very rare and unusual, they will usually not be a problem, and when encountered may be dealt with by some additional signal processing akin to limiting or a small reduction in word length.

[1] As an example, noted by Fielder [10], trombone transient peak levels can reach 129 dB SPL at normal live listening positions. While Fielder's figure has been met with disbelief by much of the audio community, it is entirely in accord with one of the author's live recording experiences where peak trombone transient levels can exceed that of loud drums if the trombone is played ferociously and happens to be pointing in the precise direction of a recording microphone. There are probably other extreme examples that rarely show up in music. One imagines examples such as krummhorns, jangling keys close to a microphone, exotic percussions, and so forth.

But this peak data-rate limitation problem does emphasize the importance of using a packing strategy that minimizes peak data rates as far as possible. Packing systems previously proposed for other applications have been very poor with regard to the reduction of peak data rates.

2.5 Trading Bandwidth against Playing Time

While there is a fairly wide body of opinions which hold that the sharp filters just above the audio band, necessitated by the current 44.1- and 48-kHz sampling rates, are audibly detrimental, the proposal to adopt 96 kHz as the standard may seem to be unnecessarily wasteful of data rate. Why not go for a 64- or a 66.15-kHz sampling rate, allowing a generous audio bandwidth of 25 kHz followed by a gentler tapering off before the Nyquist frequency of 32 or 33.075 kHz?

The answer is that, using lossless compression, there would be little economy to be gained by using the lower sampling rate. Shannon's work on information theory tells us that the information content of the signal is proportional to the resolution (expressed as a logarithm, such as in decibels or bits) times the bandwidth or, where the resolution is not constant with frequency, the integral of resolution with respect to frequency.

Referring to Fig. 7, we plot the resolution of a hypothetical audio signal as a function of frequency. We are assuming a sampling rate of 96 kHz, but the signal is filtered as if for a sampling rate of 66.15 kHz. Thus the only information above 33.075 kHz is dither and quantization noise.

According to Shannon's theorem, the information content of this signal is given by the shaded area under the curve. Had we used a sampling rate of 66.15 kHz, the information would have been the part of the shaded area that lies to the left of the line at 33.075 kHz. Adding in the part to the right of the line will make a difference of less than 5% of the information content.

Using predictive lossless compression, we can transmit the signal using the number of bits predicted by Shannon's measure provided the encoding filter can follow the (inverse of) the spectrum of the signal. In this case the ideal encoding filter should be almost flat to 25 kHz, and then rise sharply to achieve a boost of about 18 bits (108 dB) up to 33 kHz. Although this cannot be fully achieved by low-order predictors, they can still give a high proportion of the attainable reduction in data rate. IIR predictors are much better at approximating this requirement than FIR filters.

Also, such an extreme boost is required only for peak-level signals with near white spectra, and for most of the time on most music much more modest boosts get us close to the ideal Shannon data rate.

Even a third-order IIR filter allows useful savings in data rate on signals band limited to less than 48 kHz, as the third-order IIR predictor characteristics shown in Fig. 8(a) indicate. An IIR filter with a third-order numerator and fourth-order denominator gives an almost 1-bit greater reduction in data rate for band-limited signals, as shown in Fig. 8(b).

Although the limitation to, say, a third-order IIR filter reduces the saving in data rate for band-limited signals to below the ideal Shannon rate, the saving is still significant, so that the data-rate penalties from using a 96-kHz sampling rate rather than, say, 66.15 kHz for band-limited material are actually quite modest.

Following this reasoning, the ARA proposal [1] suggests 96 kHz as the only sampling rate apart from 48 kHz. There is little reason to depart from this suggestion on grounds of economy, and the simplification of consumer equipment that will result from not having to deal with a multitude of sampling rates is very much to be welcomed.

Even without band limiting, in practice for a given kind of audio material, the packed data rate needed for a given audio quality generally increases much more slowly than the sampling rate. This is for several reasons, including the following.

1) For a given number of bits, the quantization noise energy per unit bandwidth is inversely proportional to the sampling rate, so that for a given quantization noise level in the audio band, a higher sampling rate requires fewer bits.

2) For a given *n*th-order integer predictor, the ampli-

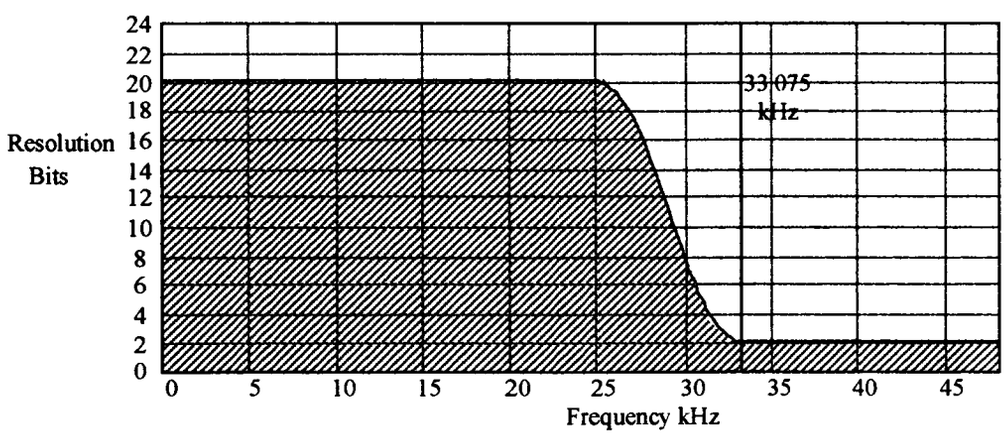


Fig. 7. Bit resolution as a function of frequency of audio signal filtered as if for transmission at a sampling rate of 66.15 kHz, but sampled at 96 kHz. Shaded area represents total information content, according to Shannon, of 96-kHz sampled signal. Area left of line at 33.075 kHz represents information content that the signal would have if transmitted at 66.15-kHz sampling rate.

tude gain reduction of lower audio frequencies increases proportional to the *n*th power of the sampling rate, giving an extra bit-rate reduction.

In particular, this means that for a doubling of the sampling rate from 48 to 96 kHz with comparable audio-band quality, we will expect typically at least a 2–2.5-bit further reduction of the packed data rate in bits per sample per channel. Any effect of IIR predictors taking advantage of band limiting in addition to this is an additional bonus.

So packing in practice allows an extension of audio to higher sampling rates without providing anything like a proportional increase in the packed data rate. For example, an increase in the sampling rate from, say, 64 to 96 kHz, which one naively may expect to increase the data rate by 50%, will in practice probably increase the packed data rate only on the order of 15% for comparable audio-band quality.

The 96-kHz sampling rate should be seen as giving the recording producer a choice of audio bandwidth—anything up to 48 kHz. If he or she chooses less than 48 kHz, there will be a corresponding increase in the available playing time and the ability to use a larger number of bits or channels before encountering data-rate limitations.

It goes without saying that bass effect signals (such as the 0.1 in a 5.1-channel system) do not need to be catered for separately. The signal will be presented to an ordinary channel, and if it has very little information above, say, 100 Hz, the transmitted information rate will be correspondingly very low.

2.6 Noise-Shaped Signals

Following the work of Lipshitz et al. [8], Stuart and Wilson [9], and others, we have become used to the idea of psychoacoustic noise shaping as a way of increasing the subjective resolution of a given digital channel "for free."

In the context of packing, this continues to apply provided that the increased ultrasonic noise from the noise shaping does not exceed the level of the original signal (including analog hiss). If the noise shaping increases the total digital signal level over any part of the spectrum, there will be an increase in the total information rate (see Fig. 7) and the psychoacoustic advantage is no longer obtained "for free."

Thus if one has the choice of encoding to 17 bit with moderate noise shaping or to 16 bit with heavy noise shaping, it may turn out that the 17-bit option is no more expensive with regard to playing time, though if the peak data rate is the limiting factor, the 16-bit option may still be advantageous.

As with straight PCM, noise shaping is an option that is applied when the signal is first digitized, or when it is

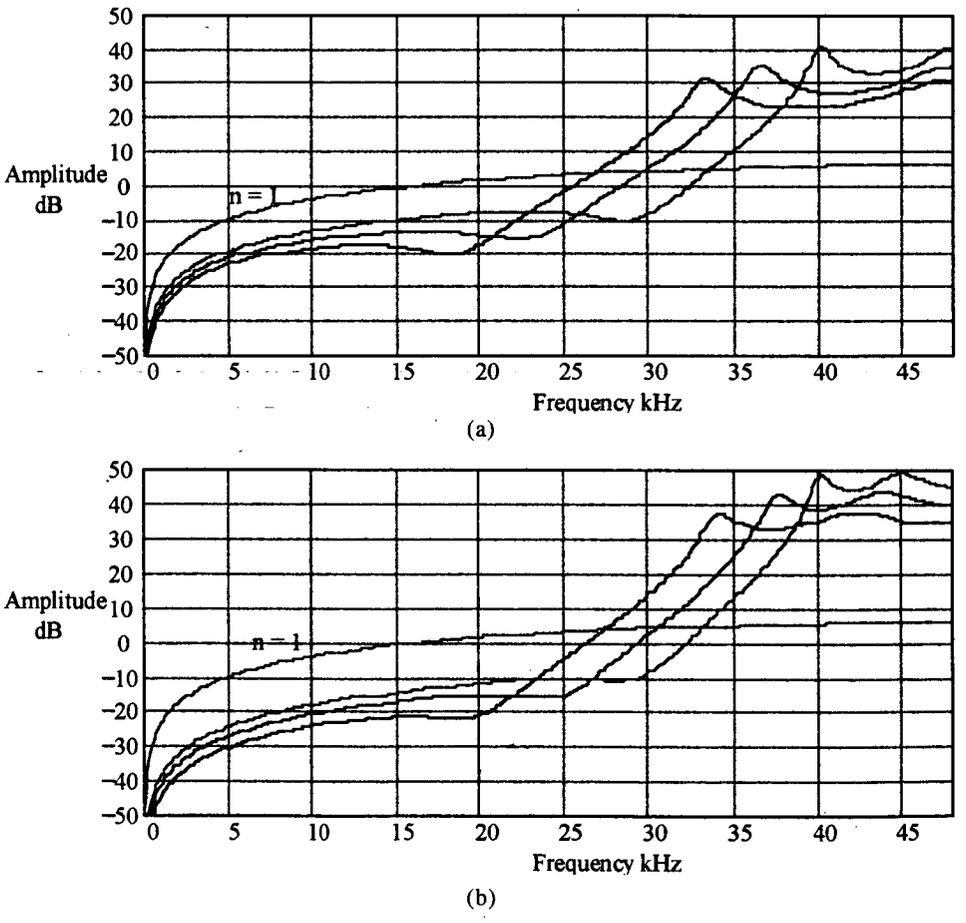


Fig. 8. Transfer functions of IIR predictors for use at 96-kHz sampling rate matched to different input bandwidths between 20 kHz and 30 kHz and compared to a first-order integer predictor. (a) IIR filters with third-order numerator and denominator. (b) IIR filters with third-order numerator and fourth-order denominator. Note the approximately 6-dB (1-bit) reduction in level in audio band from use of an extra order in denominator compared to part (a).

requantized to a lower resolution. The packing algorithm will adapt itself automatically and does not need to be so told, nor does it need to be standardized as it requires no action on the part of the decoder.

2.7 Preemphasis and Packing

Preemphasis is another technique for allegedly enhancing the subjective dynamic range of a PCM digital channel "for free." Once again, with regard to Fig. 7 and the discussion in Section 2.5, it will be seen that the advantage is not obtained "for free" if the signal is to be packed.

A full discussion is outside the scope of this paper, but there are some very interesting mathematical connections between prediction, preemphasis, and noise shaping. It turns out that the effect of applying pre- and deemphasis to a digital channel can be simulated exactly by using a PCM channel with more bits and encoding with a noise-shaping filter. However, with the appropriate prediction filters matched to the two respective cases, the resulting packed data rate is exactly the same in the two cases.

In other words, preemphasis gives no advantage (in channel data rate) if packing is to be used. As it requires a corresponding deemphasis in the decoder and hence standardization, it can be regarded as an unnecessary extra complication.

However, there is another use of preemphasis and deemphasis, which is applying them in the analog domain and using them to reduce the audibility of artifacts produced by analog-to-digital and digital-to-analog converters. This remains a valid technique, but once again it is unnecessary to complicate the transmission standard, as deemphasis can be applied digitally immediately after the analog-to-digital conversion, and preemphasis can be applied again digitally immediately before the digital-to-analog conversion.

2.8 Percentage Data Reduction

The most common question asked about packing systems is what percentage of data reduction they achieve. This is in fact highly dependent on the sampling rate, the type of music or audio signal, and also on the number of bits used. At a given sampling rate and for a given type of music signal, the saving obtained from a packing algorithm is roughly a constant number of bits per sample per channel, irrespective of whether the incoming signal is a 12-bit or a 24-bit one. So the proportional data-rate saving is greater when coding low-precision audio than for very high-precision audio.

For example, if the average data-rate saving is about 9 bit per sample per channel, then one will save 56.25% of the data rate for 16-bit PCM signals, 50% of the data rate for 18-bit signals, 45% of the data rate for 20-bit signals, and only 37.5% of the data rate for 24-bit signals. The 16-bit input case would require only an average of 7 bit for packed transmission, whereas the 24-bit input case would require 15 bit—a rate that is more than double the 16-bit input case.

Using packing, every bit of precision added to the input signal also increases the packed data rate by 1 bit per sample per channel, all other things being equal.

3 DESIGN OF VERSATILE FOUR-SQUARE SYSTEMS

3.1 Minimum Need for Flags

The ARA proposal [1, table 3] lists 17 different examples of how information might be recorded on the HQAD in terms of the number of channels, number of bits, and sampling rate. The list is certainly not exhaustive.

At a recent presentation a question was raised about whether this would need complicated hardware and a complicated system of flags to indicate all the various options. This is a concern since experience with practical production environments shows that flags get lost very easily in a signal-handling chain involving equipment from a variety of manufacturers, some of whom may not conform to strict flag-handling protocols, and so systems relying on flags create many production problems.

Packing provides the key to the design of a system where the interface can be very straightforward and four-square, not reliant on flagging, but "big" enough to allow producers a generous capability to extend themselves in any particular direction, subject to the constraint that they will come up against a data-rate limitation if they choose to stretch themselves in all directions simultaneously.

In Section 2.5 we discussed flexibility of bandwidth. Different bandwidths do not require different sampling rates, nor does the lossless encoder need to be told what the bandwidth actually is—it will determine its best packing parameters from direct analysis of the signal.

A similar situation occurs with regard to the number of bits and the number of channels. For example, if the maximum number of channels is eight, we can manufacture equipment that has eight channels in and out, and leave it to the packing system to determine whether all the eight channels are carrying a signal. If not, the packing will make arrangements such that the data rate is not wasted unnecessarily on the channels that are carrying little or no information.

One particular feature of the proposal is that there is no need to standardize the digital word length. Thanks to the operation of packing, any word length shorter than 24 bit can still be transmitted and decoded as a 24-bit word with zeros in the unused LSBs, without penalty in data rate. That way the player only has to deal with one word length, 24 bit, even when the transmitted signal actually has fewer bits.

If a producer decides that, say, 18 bit is good enough, then he or she can get the increased playing time given by this lower word length without having to send any flags to the encoder or player. The encoder can automatically detect on a block-by-block basis the fact that the input only uses, for example, 18 of the bits, and it can automatically encode this correctly without any need for flagging. The producer only has to round the signal to

18 bit using a commercial noise-shape dither system designed for this purpose. Then the encoder will automatically adapt itself to this without any flagging.

3.2 Flags for Multichannel Formats

Of course, flags are still useful to indicate the type of recording we have. If there are four channels, are these discrete loudspeaker feeds or the W, X, Y, and Z of an Ambisonic recording [11], or are they simply two different two-channel mixes of the same material? The same questions could be asked if a four-track analog tape, and we would like a label to be on the box to save a lot of experimentation.

As far as possible, the need for flags should even here be minimized by ensuring that default modes of playback, such as reproducing a basic two-channel mix, always give good results, and the ARA proposal [1] is designed to ensure this. The use of mutually compatible multichannel formats, as described in [12], [13], again minimizes the need for flags.

The essential point, however, is that the user-interface format does not need to keep track of separate flags relating to the technicalities of the encoding needed for packing. The packing system uses header information internally, of course, but once the decoder has used this information in order to restore the original PCM format, it is of no further value and can be thrown away. This is in contrast to the preemphasis flag in current practice, which must be preserved correctly as separate channel-status information, with the usual well-known consequences if it gets lost.

4 SUBJECTIVE ASPECTS

Designing a packing system is in one respect much easier than designing a lossy compression system. As it will recover the input signal exactly, it should be a purely technical exercise, without the need to balance psychoacoustic compromises and to appeal to masking theory.

However, some critical users have reported degradation of the subjective sound quality when packing is used. We would suggest that this cannot be intrinsic to the use of packing and must be due to some incidental factor. For example, there may be effects on power supplies and the timing and jitter of servos when data are pulled off a disc at a nonconstant rate, as will be the case when packing is used. Clearly, it is up to hardware manufacturers to ensure that any such effects do not produce audible consequences.

5 COMMERCIAL PACKING SCHEMES

The various systems that have been proposed and implemented are often proprietary, and it is hard to find published details. However, most commercial schemes appear to fall within several simple categories.

All known schemes appear to divide signals into blocks of samples, with block lengths typically in the range 384 to around 1500 samples. Apart from the choice of block length, commercial schemes differ from one another in two other major ways.

1) The nature of the predictors used, varying from a single integer predictor to adaptive schemes using integer, FIR, or IIR predictors. More elaborate prediction options can lead to a better data-rate reduction for both average and peak data rates, but at the expense of greater decoding complexity and block header data overheads.

2) Simpler schemes simply do not transmit those MSBs or LSBs that do not change within a block. More complex schemes reduce the data rate further by using Huffman coding.

5.1 Nature of Predictors

The simplest schemes use only simple integer predictors such as $(1 - z^{-1})^n$, with $n = 1$, 2, or 3, as described at the start of this paper. These systems appear to include those of Cellier et al. [14]. Decca, and the Canadian company DHJ Research. Such integer predictors do not give maximum data-rate reduction due to their poor matching of signal statistics, and they actually increase the data rate of some signals. They can be reasonably satisfactory for reducing the average data rate in, for example, hard disc editing and tape backup systems, especially on classical music, but they are less effective in reducing data rates during peak passages on demanding treble-heavy material, including much pop music.

More sophisticated schemes using FIR prediction filters have been proposed, and an example is the "Shorten" waveform compression program of A. J. Robinson of the Cambridge University Engineering Department. FIR predictors (with fractional or real coefficients) are more versatile than integer predictors, and even a second-order FIR predictor can "notch out" a single narrow band of information at any frequency. However, FIR predictors are not good at dealing with wider bands of information at a level of, say, 60 dB above the rest of the spectrum. As noted in Section 2.5, an IIR filter can be used to code efficiently a spectrum such as Fig. 7, whereas an FIR filter of low order would give little advantage in this case.

Another way to visualize the advantages of IIR predictors is to note that, according to the theory, we cannot reduce the high-level parts of the spectrum without increasing the low-level parts. With low-order FIR predictors we are unable to boost the low-level parts of the spectrum, whereas IIR filters allow us to place "poles" to provide the necessary boost. These poles can be seen as peaks in the curves of Fig. 8(a) and (b).

We are unaware of any currently commercialized schemes using IIR predictors, but these allow a better matching to the case where the levels in a signal spectrum have an extremely wide dynamic range, often in excess of 60 dB. This is especially important with higher sampling rates, such as 96 kHz.

5.2 Huffman Coding

As noted earlier, with typical audio statistics, Huffman coding can reduce data rates by around 1.5 bit per

sample per channel. Of the commercial systems, those of Decca and DHJ are believed not to use Huffman coding, but that of Cellier et al. [14] does.

5.3 Proposals for HQAD

The present authors believe that for the HQAD it is important to use a system that minimizes data rates in both average and peak data-rate passages. This means using predictors capable of handling spectra with very wide dynamic ranges efficiently, which means using IIR predictors. Our provisional proposal is to use third-order predictors with fairly fine coefficient quantization. In addition we propose block lengths that are integer multiples of a minimum block length such as 384 samples.

The quantization structure of the IIR filters is designed such that there is little difficulty in porting the encoders and decoders to almost any fixed-point DSP processing platform, avoiding the problem of different implementations giving different rounding errors.

In addition the IIR filters incorporate features making the data-rate reduction maximally effective even for low-level input signals, a situation where standard prediction schemes often have poor data-rate reduction performance. A saving of 1 or 2 bit per sample per channel is just as valuable in quiet passages as it is in loud ones, especially for music with a lot of quiet passages.

The proposal uses simplified Huffman code tables optimized primarily for Laplacian amplitude statistics in order to reduce the data rate. It is also adaptive to the input quantization step size so that it can cope with a wide range of input word lengths.

6 CONCLUSIONS

Although there are many possibilities for lossless coding or packing of PCM data, all commercial schemes known to the authors are based on prediction methods because of the relative simplicity of decoding. While most packing schemes give significant or substantial reductions of average data rates, they differ very greatly in their ability to reduce peak data rates, with Huffman-coded IIR prediction schemes being superior in applications where minimizing both average and peak data rates is important, such as the multichannel HQAD.

In addition an optimally designed packing system guarantees efficient data coding with a wide variety of word lengths and possible audio bandwidths, avoiding the need for using a multiplicity of different sampling rates and word lengths in a packed HQAD standard. A packed system thus allows producers to choose a wide variety of standards of quality to meet future commercial or artistic needs without need for a multiplicity of different technical standards.

A producer can choose a very wide bandwidth, the dynamic range, and the number of channels to meet future state-of-the-art quality while still getting HQAD playing times on the order of 70 min [1], or the producer can choose to compromise quality in some aspects (but still with results superior to current CD standards) in order to gain longer playing times, all within a single technical standard. Thus a packed standard provides far fewer constraints to future improvements in the audio art and in the commercial range of applications than did previous fixed PCM standards.

7 REFERENCES

[1] "A Proposal for the High-Quality Audio Application of High-Density CD Carriers (Version 1.3)," Acoustic Rennaissance for Audio, Tech. Subcommittee Doc. (1996 Jan.1). Available from ARA Secretariat, Stonehill, Stukeley Meadows, Huntingdon, Cambs., PE18 6ED, UK, or on the World Wide Web at http://www.meridian_audio.com/ara. Version 1.2 was reprinted in Stereophile, 1995 August.

[2] J. I. Makhoul, "Linear Prediction: A Tutorial Review," *Proc. IEEE*, vol. 63, pp. 561–580 (1975 Apr.).

[3] R. C. Gonzales and R. E. Woods, *Digital Image Processing* (Addison Wesley, Reading, MA, 1992), chap. 6, sec. 6.4.3, pp. 358–362.

[4] M. Rabbani and P. W. Jones, *Digital Image Compression Techniques* (SPIE Press, Bellingham, WA, 1991).

[5] B. S. Atal and M. R. Schroeder, "Adaptive Predictive Coding of Speech Signals," *Bell Sys. Tech. J.*, vol. 49, pp. 1973–1986 (1970 Oct.).

[6] J. Weiss and D. Schremp, "Putting Data on a Diet," *IEEE Spectrum*, vol. 30, pp. 36–39 (1993 Aug.).

[7] N. S. Jayant and P. Noll, *Digital Coding of Waveforms* (Prentice-Hall, Englewood Cliffs, NJ, 1984).

[8] S. P. Lipshitz, J. Vanderkooy, and R. A. Wannamaker, "Minimally Audible Noise Shaping," *J. Audio Eng. Soc.*, vol. 39, pp. 836–852 (1991 Nov.).

[9] J. R. Stuart and R. J. Wilson, "Dynamic Range Enhancement Using Noise-Shaped Dither Applied to Signals with and without Preemphasis," presented at the 96th Convention of the Audio Engineering Society, *J. Audio Eng. Soc. (Abstracts)*, vol. 42, p. 400 (1994 May), preprint 3871.

[10] L. Fielder, "Dynamic Range Issues in the Modern Digital Audio Environment," in *Proc. of "Managing the Bit Budget" AES UK Conf.* (1994 May 16–17), pp. 3–19.

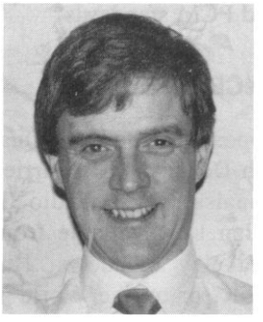
[11] M. A. Gerzon and G. J. Barton, "Ambisonic Decoders for HDTV," presented at the 92nd Convention of the Audio Engineering Society, *J. Audio Eng. Soc. (Abstracts)*, vol. 40, p. 438 (1992 May), preprint 3345.

[12] D. J. Meares, "High Definition Sound for High Definition Television," in *Proc. AES 9th Int. Conf. "Television Sound, Today and Tomorrow"* (Detroit, MI, 1991 Feb.), pp. 187–215.

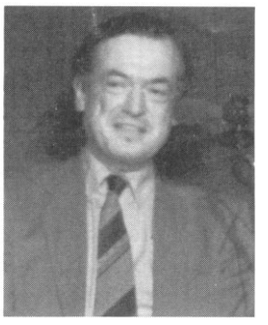
[13] M. A. Gerzon, "Hierarchical System of Surround Sound Transmission for HDTV," presented at the 92nd Convention of the Audio Engineering Society, *J. Audio Eng. Soc. (Abstracts)*, vol. 40, p. 445 (1992 May), preprint 3339.

[14] C. Cellier, P. Chenes, and M. Rossi, "Lossless Audio Bit Rate Reduction," in *Proc. of "Managing the Bit Budget" AES UK Conf.* (1994 May 16–17), pp. 107–122.

THE AUTHORS



P. Craven



M. Gerzon

Peter Craven attended Oxford University from 1966–74, studying mathematics as an undergraduate and astrophysics as a postgraduate. Much of this time was devoted to the design of recording equipment and making "purist" uncompressed recordings of groups such as the Schola Cantorum of Oxford. He met Michael Gerzon in 1967, starting a collaboration that was to last for 29 years.

A career in academic computing followed, including much work on compilers for the programming language Algol68. This work was later extended in a project funded by N.A. Software Ltd. and is now the basis for their commercial Fortran90 compiler.

In 1982 Dr. Craven left university life to become an independent consultant specializing in audio digital signal processing (DSP) software and in high-level methods of generating efficient DSP code. In the late 1980s, an extensive collaboration with B&W Loudspeakers on room equalization resulted in patents relating to high-resolution D/A conversion and to digital PWM power amplifiers. Current consultancy projects include Motorola DSP56000 audio software for use in consumer audio–visual systems, and the audio DSP for the Jubilee Line Extension's public address system (London Underground).

The many activities jointly with Michael Gerzon include the invention of the Ambisonic Soundfield Microphone in 1973, a seminal paper on noise shaping and dither published in 1989, and the inventions of Autodither and Buried Data. The work on lossless data compression was the last major collaboration before Michael Gerzon's death, and is continuing.

Despite involvement with state-of-the-art reproduction technology, for relaxation Peter Craven turns either to live music or to prewar 78s. In his view, 1927 was a particularly good year.

•

Michael A. Gerzon was born in Birmingham, England, in 1945. He died on May 6, 1996.

He received an M.A. degree in mathematices from Oxford University in 1967, after which he did postgraduate work in axiomatic quantum theory. His interests in audio stemmed from interests in music, sensory perception, and information theory. Beginning in 1967 he became active in recording live music, recording artists as diverse as Emma Kirkby, Michael Tippett, Pere Ubu, and Anthony Braxton, and recorded music for more than 15 LP and CD releases.

Arising from this interest, in 1971 he started earning his living from consultancy work in audio and signal processing. He was one of the main inventors of the Ambisonic surround sound technology, working in the 1970s and early 1980s with the British National Research Development Corporation.

With Peter Craven, he co-invented the Soundfield microphone. He also developed mathematical models for human directional psychoacoustics for use in the design of directional sound production systems. He was made a fellow of the Audio Engineering Society in 1978 for this work and was awarded the AES Gold Medal in 1991 for his work on Ambisonics. With Dr. Craven, he also developed the basis of the noise-shaped dither technologies now widely used for resolution enhancement of CDs, and, until the time of his death, continued to work actively in this area.

He published about 100 articles and papers in the field of audio and signal processing, including numerous papers on stereo and surround sound systems. By 1995 he had been granted 9 British patents and corresponding applications internationally. He published papers on practical and theoretical aspects of linear and nonlinear signal processing and systems theory, digital reverberation, room equalization, data compression, spectral analysis, and noise-shaping and dither technologies.

Michael's more recent work included digital signal processing algorithms for professional digital audio for the Israeli company K.S. Waves, multiloudspeaker directional reproduction technologies for Trifield Productions Ltd., and dither and buried data technologies for XtraBits.

He had many professional research interests, including the development of general methods for designing complex signal processing systems based on the methods of *-algebras and category theory, image data compression, and advanced mathematical modeling of auditory perceptual effects relevant to audiophile quality.

His personal interests included writing poetry, the history of contemporary musics, and the mathematical conceptual foundations of theoretical physics.