

THE MLP LOSSLESS COMPRESSION SYSTEM

MA GERZON, PG CRAVEN², JR STUART¹, MJ LAW², RJ WILSON¹

¹*Meridian Audio Ltd, Huntingdon, England*
jrs@meridian-audio.com

²*Algol Applications Ltd, Hove, England*
peter@algol.co.uk

Lossless compression provides bit-exact delivery of the original signal and is ideal where the highest possible confidence in the final sound quality is required. MLP (Meridian Lossless Packing) has recently been adopted as the lossless coding method used on DVD-Audio. MLP uses four principal strategies to reduce both the total quantity and the peak rate of encoded data. MLP can invert a matrix transformation losslessly: this allows a 2-channel representation to be transmitted alongside a multichannel signal, with minimal increase in data rate. The paper illustrates how the characteristics of the incoming audio affect the coding performance, and indicates MLP's versatility, achieved by use of substreams and an open-ended metadata specification.

INTRODUCTION

Meridian Lossless Packing (MLP) is a lossless coding system for use on high-quality digital audio data originally represented as linear PCM. High quality audio these days implies high sample rates, large word sizes and multichannel. This paper describes the MLP system while presenting interesting insights into lossless coding in general.

1 OVERVIEW

MLP performs lossless compression of up to 63 audio channels including 24-bit material sampled at rates as high as 192kHz. Lossless compression has many applications in the recording and distribution of audio. In designing MLP we have paid a lot of attention to the application of lossless compression to data-rate-limited transmission (e.g. storage on DVD), to the option of constant data rate in the compressed domain and to aspects that impact on mastering and authoring. MLP was targeted to provide:

- Good compression of both peak and average data rates.
- Use of both fixed and variable-rate data streams.
- Automatic savings on bass-effects channels.
- Automatic savings on signals that do not use all of the available bandwidth (e.g. sampled at 96kHz).
- Automatic savings when channels are correlated.
- Comprehensive metadata.
- Hierarchical access to multichannel information.
- Modest decoding requirements.

The reduction of peak data rate is equivalent to reducing the wordwidth of 48kHz-sampled signals by 4 bits or more. At least 8 bits are removed from signals sampled at 96kHz and so 24-bit audio can be compressed into a 16-bit channel.

MLP provides for up to 63 channels, but applications tend to be limited by the available data rate. To aid compatibility, MLP uses a hierarchical stream structure containing multiple substreams and hierarchical additional data. With this stream structure decoders need only access part of the stream to play back subsets of the audio. Suitable use of the substreams also allows 2-channel compatibility; a low-complexity decoder can recover a stereo mix from a multichannel stream. Figure 1 gives an overview of the process of losslessly compressing a stream containing multiple audio channels and auxiliary data onto a disc.

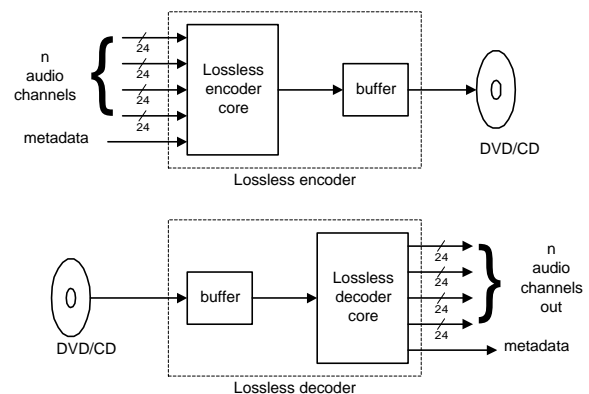


Figure 1: An overview of MLP used on disc.

2 LOSSLESS COMPRESSION

Unlike perceptual or lossy data reduction, lossless coding does not alter the final decoded transmitted signal in any way, but merely ‘packs’ the audio data more efficiently into a smaller data rate.

Audio information that is of interest to the human listener contains some redundancy. On music signals, the information content varies with time and the input channel information capacity is rarely fully exercised. The aim of lossless compression is to reduce incoming audio to a data rate that closely reflects the inherent information content plus a minimum overhead.

An important insight then is that the coded output of a lossless compressor will have a *variable* data rate on normal audio content. Figure 2 illustrates such variation through 30 seconds of a 6-channel recording of baroque chamber music at 96kHz 24-bit precision (original data rate = 13.824Mbps).

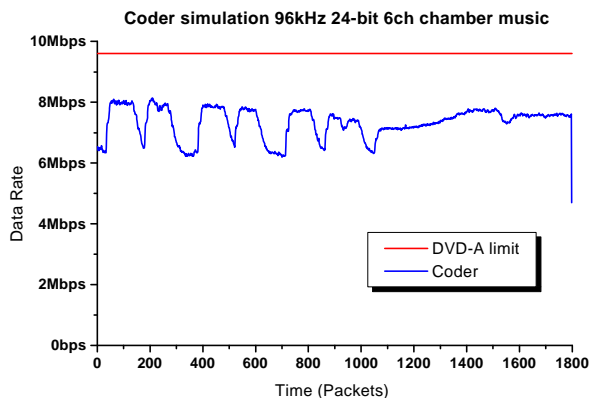


Figure 2: Data rate over 30 seconds of excerpt.

Whilst a music example can show this kind of compression, we reasonably expect (and see) wider variations in compressed rate. There are also pathological signals: for example, silence or near-silence will compress greatly and signals which are nearly random will not. Indeed, should a section of channel data appear to be truly random, then no compression is possible. Fortunately it turns out that real acoustic signals tend not to provide full-scale white noise in all channels for any significant duration!

Previously lossless audio data compression systems have been optimised for reducing *average data rate* (i.e. minimising compressed file size).

The ARA Proposal [1] described the important requirement of reducing the *instantaneous peak data rate* for optimum results at high sampling rates such as 96kHz or 192kHz and for data-limited disc-based applications like DVD-Audio.

MLP was developed by the authors as a simple-to-decode method optimised for these special requirements of high-rate high-quality audio combined with an

unbreakable requirement to reduce peak data rate at all times.

MLP tackles this by attempting to maximise the compression at all times using this set of techniques:

- Looking for ‘dead air’ – channels that do not exercise all the available word size.
- Looking for ‘dead air’ – channels that do not use all the available bandwidth.
- Removing inter-channel correlations.
- Efficiently coding the residual information.
- Smoothing coded information by buffering.

2.1 Application factors

A lossless compression system must guarantee lossless (i.e. bit-for-bit) recovery over an encoding – decoding pass. If this is achieved, then the system will remain lossless over multiple cascades of encoding – decoding; there will be no generation loss.

A significant requirement of a versatile coding system is that the process remains lossless regardless of the encoder or decoder computing platform.

The *average data rate* after compression (coding ratio) most effects playing time and hard-disk storage applications. MLP allows the compressed data to be packed to a *variable data rate* on the disc, which maximises playing time. However, as explained earlier, the *peak data rate* can be very important in two cases:

- When there is a need to fit the compressed data into a channel that has lower rate capacity than the incoming audio.
- When, for a particular application, the compressed data is packed to a *constant data rate*, then this rate cannot be less than the peak rate of the item. Examples include packetising MLP in SPDIF or in a constant-rate stream to accompany motion video.

2.2 Integrity

A lossless encoding – decoding system displays an inherent integrity. Once audio has been ‘wrapped up’ in the MLP stream it will remain intact through any intermediate storage or transmission process.

An MLP decoder can continuously test against checks inserted by the encoder that the overall transmission has been lossless. This makes the audio more secure than transmitting LPCM, since in that case the receiver cannot tell if intermediate processes have occurred on the data.

However, any coded stream is subject to random media transmission errors. To minimise the impact of these, MLP has several error-detection crosschecks in the stream.

Another important consideration for a practical system is to be able to start and stop decoding quickly and to avoid unnecessary latency.

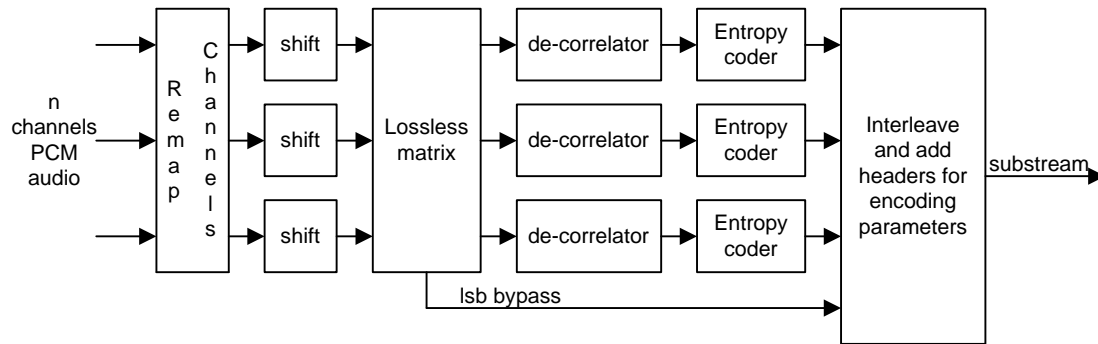


Figure 3: Block diagram of the lossless encoder core.

3 HOW DOES IT WORK?

MLP coding is based on established concepts, however there are some important novel techniques used in this system, including:

- Lossless processing.
- Lossless matrixing.
- Lossless use of IIR filters.
- Managed FIFO buffering across transmission.
- Decoder lossless self-check.
- Operation on heterogeneous channel sample rates.

These methods are described next, in the context of the encoder.

4 MLP ENCODER

The MLP encoder core is illustrated in Figure 3. The steps for encoding blocks of data are:

1. Incoming channels may be remapped to optimise the use of substreams (described later).
2. Each channel is shifted to recover unused capacity (e.g. less than 24-bit precision or less than full-scale).
3. A lossless matrix technique optimises the channel use by reducing inter-channel correlations.
4. The signal in each channel is de-correlated using a separate predictor for each channel.
5. The de-correlated audio is further optimised using entropy coding.
6. Each substream is buffered using a FIFO memory system to smooth the encoded data rate.
7. Multiple data substreams are interleaved.
8. The stream is packetised for fixed or variable data rate and for the target carrier.

4.1 Lossless Matrix

A multichannel audio mix will usually share some common information between channels.

On occasions, e.g. when widely spaced microphones are used, the correlations will be weak. However, there are other cases where the correlations can be high. Examples include multi-track recordings where a mix-down to the delivered channels may pan signals between channels and thus place common information in some channels.

There are also specific examples where high inter-channel correlations occur, including:

- Mono presented as dual-mono with identical left and right (common in ‘talking book’ or archive recordings).
- Derived surround signals based on left minus right.
- Multichannel speaker feeds resulting from a hierarchical upscale.
- Multichannel speaker feeds resulting from an ambisonic decode from B-format WXYZ.

The MLP encoder uses a matrix that allows the encoder to reduce correlations, thereby concentrating larger amplitude signals in fewer channels. A trivial (though important) example would be the tendency of the matrix process to rotate a stereo mix from left/right to sum/difference. In general the encoded data rate is minimised by reducing commonality between channels. However, conventional matrixing is not lossless: a conventional inverse matrix reconstructs the original signals but with rounding errors.

The MLP encoder decomposes the general matrix into a cascade of affine transformations. Each affine transformation modifies just one channel by adding a quantised linear combination of the other channels, see Figure 4. For example, if the encoder subtracts a particular linear combination, then the decoder must add it back. The quantisers Q in Figure 4 ensure constant input – output wordwidth and lossless operation on different computing platforms.

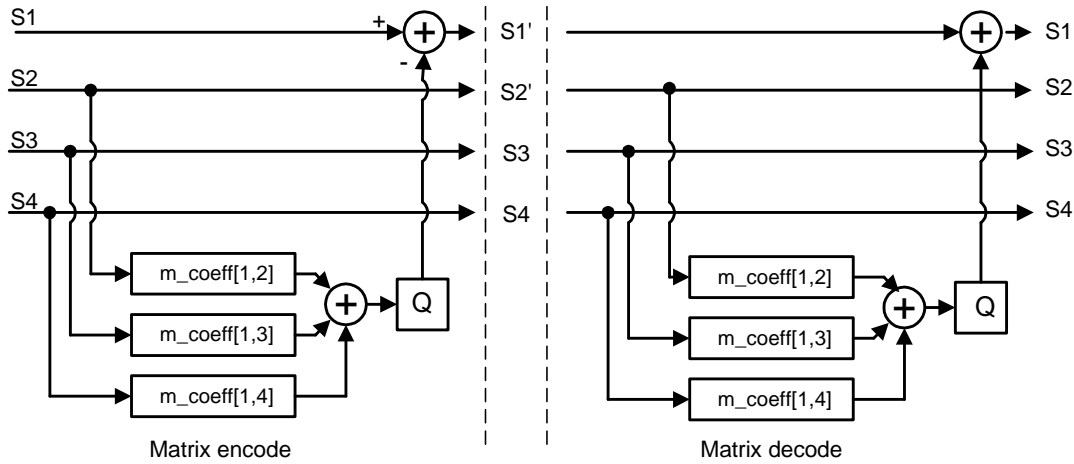


Figure 4: A single lossless matrix encode and decode.

4.2 Prediction

If the values of future audio samples can be estimated, then it is only necessary to transmit the rules of prediction along with the difference between the estimated and actual signals.

This is the function of the de-correlator (optimal coding shows no correlation between the currently transmitted difference signal and its previous values).

It is useful to consider how prediction operates in the frequency (Shannon) domain.

Figure 5 shows the short-term spectrum of a music excerpt. If this spectrum were flat a linear prediction filter could make no gains. However it is far from flat, so a de-correlator can make significant gains by flattening it, ideally leaving a transmitted difference signal with a flat spectrum – essentially being white noise. The Gerzon/Craven theorems [10] show that the level of the optimally de-correlated signal is given by the average of the original signal spectrum when plotted as dB versus linear frequency. As illustrated in Figure 5, this dB average can have significantly less power than the original signal, hence the reduction in data rate. In fact this power reduction represents the information content of the signal as defined by Shannon [2].

In practice, the degree to which any section of music data can be ‘whitened’ depends on the content and the complexity allowed in the prediction filter.

Infinite complexity could theoretically achieve a prediction at the entropy level shown in Figure 5, however all the coefficients which define this de-correlator would then need to be transmitted to the decoder (as well as the residual signal) to reconstruct (re-correlate) the signal. There is therefore a need to obtain a good balance between predictor complexity and performance.

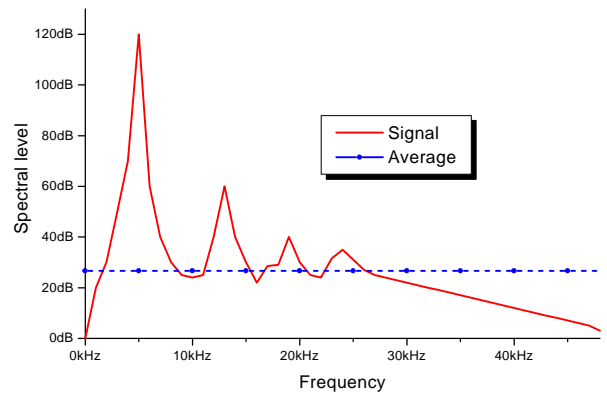


Figure 5: Spectra of a signal and its average level.

4.3 FIR and IIR Prediction

Most previous lossless compression schemes use FIR prediction filters and can achieve creditable reduction of data-rate on conventional CD-type material [3] [4] [5]. However, we pointed out in [6], [7] and [8] that IIR filters have advantages in some situations, particularly:

- Cases where control of peak data-rate is important.
- Cases where the input spectrum exhibits an extremely wide dynamic range.

The ARA proposal [1] pointed out the particularly increased likelihood of wide dynamic range in the spectrum of audio sampled at higher rates such as 96 or 192kHz. Spectral energy at high frequencies is normally quite low and may be further attenuated by microphone response or air-absorption.

The ARA also indicated the desirability that a music provider should have the freedom to control lossless data rate by adjusting supersonic filtering during mastering.

A powerful lossless compression system will require the use of FIR and IIR prediction.

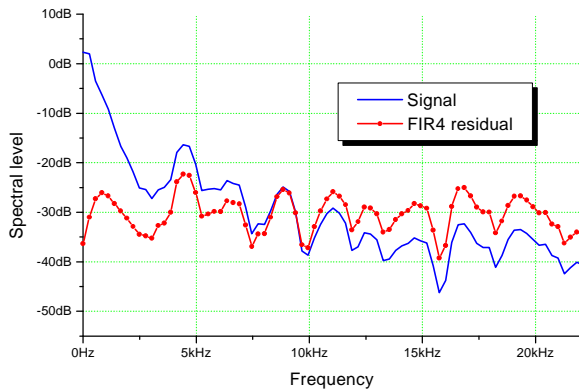


Figure 6: Example of spectra for a signal excerpt and the residual using a 4th order FIR predictor.

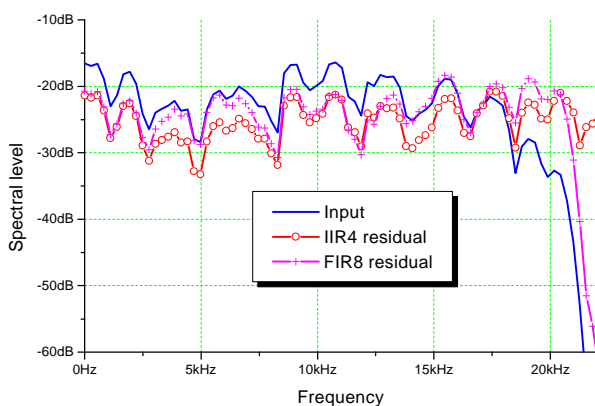


Figure 7: Spectra for a signal excerpt and the residuals using 8th order FIR and 4th order IIR predictors.

Figure 6 shows the spectrum of a 3-6ms frame taken from the ending of the *William Tell Overture*. This section is high level, contains a cymbal crash and has a spectrum that is easily flattened by a low order filter. Figure 6 also shows the residual spectrum after decorrelation by a 4th order FIR filter.

Track 6 of the CD *Hello, I must be going!* by Phil Collins shows an example that is quite difficult to compress. The original signal spectrum in Figure 7 includes a percussion instrument with an unusually extended treble response. An 8th order FIR filter is able to flatten the major portion of the spectrum. However, it is completely unable to deal with the drop above 20kHz caused by the anti-alias filter. A 4th order denominator IIR filter is able to do this very effectively as shown. In this case the improvement in compression is small, as there is only 2kHz of under-used spectrum between the 20kHz cut-off and the Nyquist frequency of 22.05kHz. IIR filtering gives a bigger improvement if filtering leaves a larger region of the spectrum unoccupied, for example if audio is sampled at 96kHz but a filter is placed at say 30 or 35kHz (see [8]).

4.4 Lossless IIR filtering

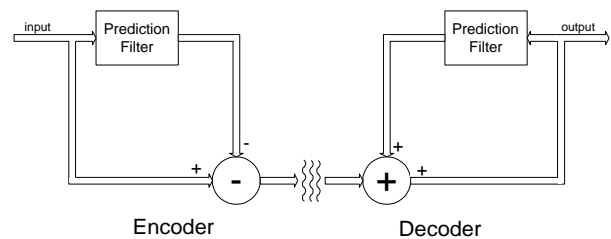


Figure 8: General schematic of encode and decode using prediction filters.

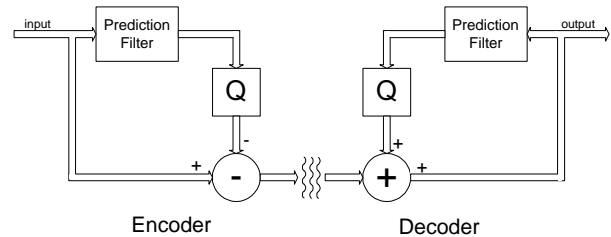


Figure 9: Schematic of encode and decode using prediction filters with quantisers.

Until recently there has been a difficulty in using IIR predictors for lossless coding. To see this, consider that the output of the prediction filter in Figure 8 generally has a longer wordlength than the input signal because of the multiplication by fractional coefficients. As the transmitted data rate depends on total wordlength at this point, extending wordsize would be counter-productive. Figure 9 shows a conventional way of dealing with this, in which the output of the prediction filter is quantised so that the transmitted prediction error has the same wordlength as the input signal. By symmetry, the decoder (also in Figure 9) can recover the original signal despite the fact that the side chain is now non-linear. However, this assumes that the decoder's predictor and the encoder's predictor produce outputs which, when requantised, agree exactly. This is not difficult to achieve with an FIR filter, as its output can be computed exactly using finite wordlength arithmetic.

In contrast, IIR filters with fractional coefficients cannot be exactly implemented since representation of the re-circulating signal requires an ever-increasing wordlength. The IIR output is thus dependent on the rounding behaviour of the underlying arithmetic, and it is difficult to ensure that this will not sometimes affect the quantised output also.

Thus, we have the possibility that a decoder implemented on different hardware (e.g. computer or DSP chip) from the encoder will not reproduce exactly the same bits and the compression will not be lossless.

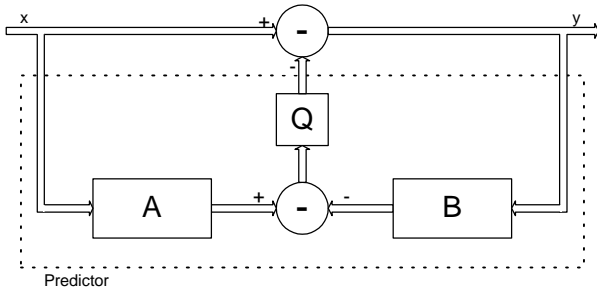


Figure 10: Lossless IIR prediction structure (encoder).

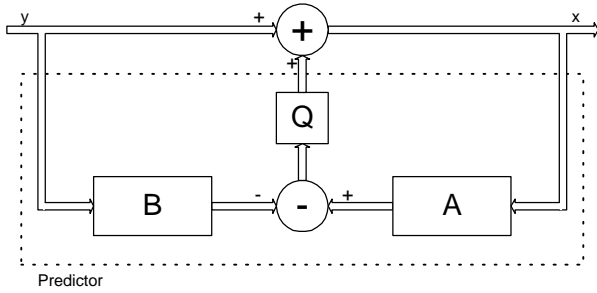


Figure 11: Lossless IIR prediction structure (decoder).

We proposed in [7] the encoding architecture of Figure 10 with the corresponding decoder of Figure 11. As the input and output signals are both quantised and filters A and B are both FIR, the input to the quantiser Q is a finite-precision signal and the quantisation can therefore be precisely specified. On the other hand, because of the re-circulation through filter B , the total response is IIR. We have achieved the aim of constructing an IIR predictor that is portable across hardware platforms.

4.5 Lossless Prediction in MLP

The MLP encoder uses a separate predictor for each encoded channel. The encoder is free to select IIR or FIR filters up to 8th order from a wide palette. These extensive options ensure that good data reduction can be provided on as many types of audio as possible. The effectiveness of the encoder tactics described so far can be seen in Figure 12, which graphs the data rate through a 30-second 96kHz 24-bit 6-channel orchestral excerpt.

The lowest curve in Figure 12 is the data rate for the normal MLP encoder; the flat-topped sections will be explained later.

The middle curve shows the impact of switching off the lossless matrix and illustrates that in this case a significant improvement in coding ratio was obtained by removing inter-channel correlations. The upper curve shows the further reduced effectiveness by constraining the predictor choices to a simple FIR. The top line shows the 9.6Mbps data-rate limit for DVD-Audio.

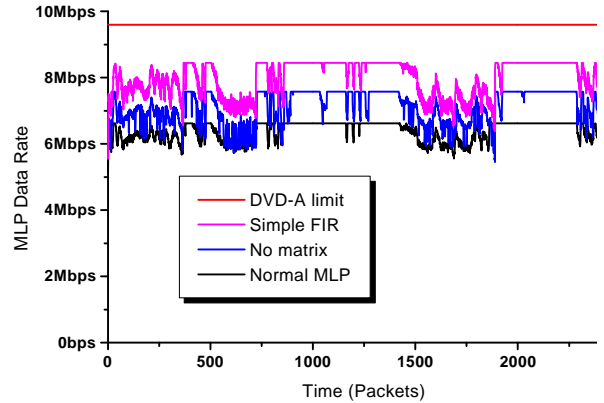


Figure 12: Resulting data rate in MLP encodes showing the benefit of the encoder stages.

The input data rate is 13.824Mbps, so in this example the options of IIR and lossless matrixing improved the coding ratio from 1.64:1 to 2.08:1.

4.6 Entropy coding

Once the cross-channel and inter-sample correlations have been removed, it remains to encode the individual samples of the de-correlated signal as efficiently as possible. ‘Entropy coding’ is the general name given to this process, its aim being to match the coding of each value to the probability that it occurs. Infrequent values are coded to a large number of bits, but this is more than compensated by coding frequent values to a small number of bits.

Audio signals tend to be peaky and so linear coding is inefficient. For example, in PCM one has to allocate enough bits to describe the highest peak, and the most significant bits (MSBs) will be used infrequently. Audio signals often have a Laplacian distribution (c.f. [3] [4] [5]), that is, the histogram is a two-sided decaying exponential; this appears to be true even after de-correlation. The Rice Code (see [3] and [4]) provides a simple and near-optimal way of encoding such a signal to a binary stream and has the advantage that encoding and decoding need not use tables.

The Rice code is not used unconditionally, the MLP encoder may choose from a number of entropy coding methods.

Although MLP is designed principally for music or speech signals, it is always possible that it may be asked to encode peak-level RPDF (all values equally probable) white noise. In fact ordinary PCM (which would be optimal for this rogue case) is one of the coding options available to the MLP encoder.

4.7 Buffering

We have explained that while normal audio signals can be well predicted, there will be occasional fragments like sibilants, synthesised noise or percussive events that have high entropy.

MLP uses a particular form of stream buffering that can reduce the variations in transmitted data rate, absorbing transients that are hard to compress.

FIFO memory buffers are used in the encoder and decoder as shown in Figure 13. These buffers are configured to give a constant notional delay across encode and decode. This overall delay is small – typically of the order of 75ms. To allow rapid start-up or cueing, the FIFO management minimises the part of the delay due to the decoder buffer. So, this buffer is normally empty and fills only ahead of sections with high instantaneous data rate.

During these sections, the decoder's buffer empties and is thus able to deliver data to the decoder core at a higher rate than the transmission channel is able to provide. In the context of a disc, this strategy has the effect of moving excess data away from the stress peaks, to a preceding quieter passage.

The encoder can use the buffering for a number of purposes, e.g.:

- Keeping the data-rate below a preset (format) limit.
- Minimising the peak data rate over an encoded section.

Figure 14 shows an example of the latter. The entropy-coded data rate from the encoder core is shown along with the buffered result. The buffered data has a characteristic flat-topped curve. This is not due to clipping or overload, but to rate absorption in the encoder/decoder FIFOs.

Another illustration of data-rate minimisation is shown in Figures 15 and 16. Again the encoded data rate is plotted through a 30-second 96kHz 24-bit 6-channel excerpt featuring a close recording of a jazz saxophone. Figure 15 indicates the underlying compression with the encoder set to limit above 9.5Mbps. The minimum-rate encode shown in Figure 16 makes long-term (but low occupancy) use the decoder buffer.

It should be obvious that the situation in Figure 16 is preferable if the transmission channel (maybe DVD disc) has other calls on the bandwidth – for example bandwidth to transmit associated picture or text.

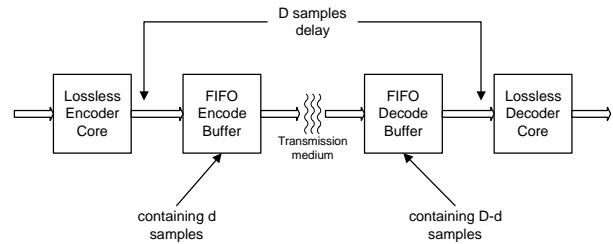


Figure 13: Outlining the buffering used in MLP.

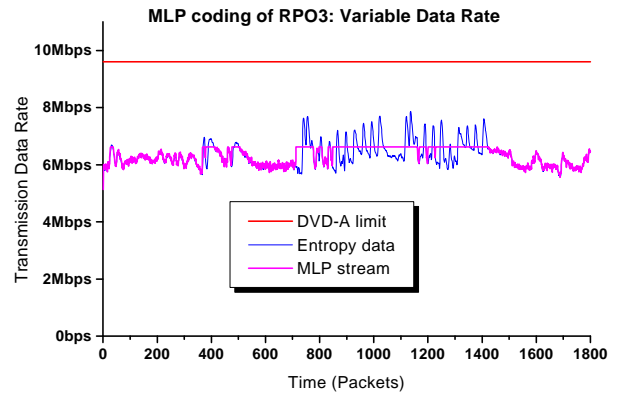


Figure 14: Showing how buffering can minimise data rate in the transmission channel.

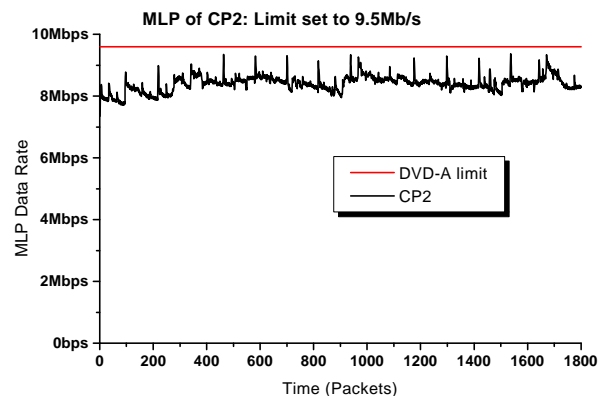


Figure 15: Showing an unlimited data rate encoding.

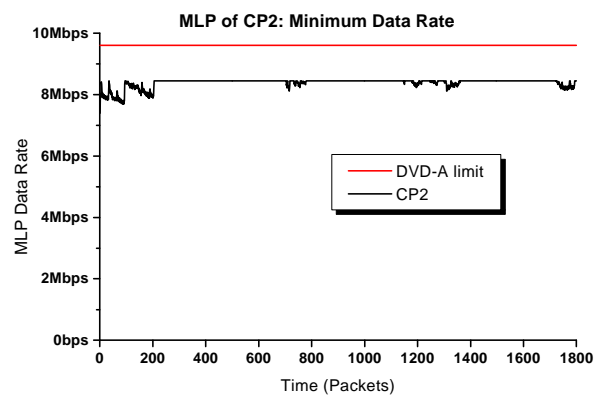


Figure 16: Showing a minimum data rate encoding.

Figure 17 shows how hard-to-compress signals can be squeezed below a preset format limit. This 30-second 96kHz 24-bit recording features closely recorded cymbals in 6 channels. At the crescendo this signal is virtually random and the underlying compressed data rate is 12.03Mbps. Buffering allows the MLP encoder to hold the transmitted data rate below 9.2Mbps by filling the decoder buffer to a short-term maximum of 86kbyte (bottom curve).

Figure 18 shows the potential for peak data-rate reduction on this item with different amounts of available FIFO memory.

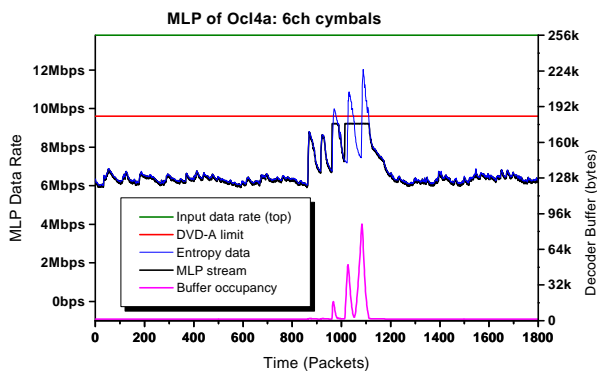


Figure 17: Buffering allows a difficult passage to remain below a hard format limit.

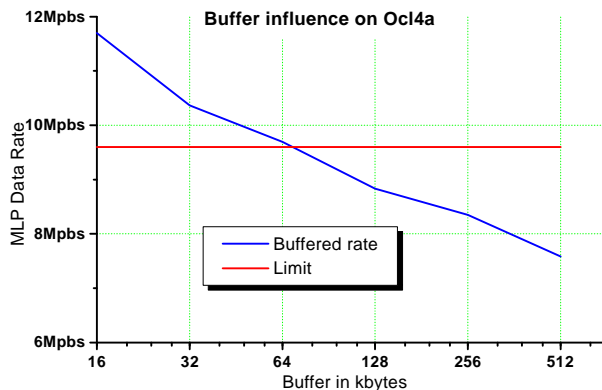


Figure 18: Showing the effect of buffering on data rate.

5 USE OF SUBSTREAMS

The MLP stream contains a hierarchical structure of substreams. Incoming channels can be matrixed into two (or more) substreams. This method allows simpler decoders to access a subset of the overall signal.

This substream principle is illustrated for the encoder in Figure 19 and the decoder in Figure 20: note that each substream is separately buffered.

We see in Figure 20 that the output of decoder 0 is (losslessly) matrixed into the output of decoder 1 to build up the overall signal.

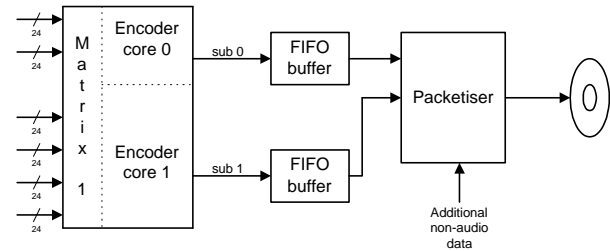


Figure 19: Illustrating two substreams in encoding.

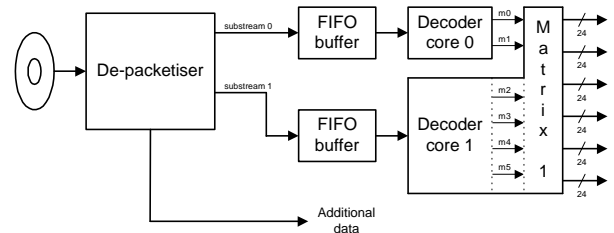


Figure 20: Decoding two substreams.

6 MLP DECODER

The MLP decoder core is shown in Figure 21. The decoder unwinds each encoder process in reverse order. The decoder is relatively low complexity. A decoder capable of extracting a 2-channel stream at 192kHz requires approximately 27MIPs, while 40MIPs will be required to decode 6 channels at 96kHz.

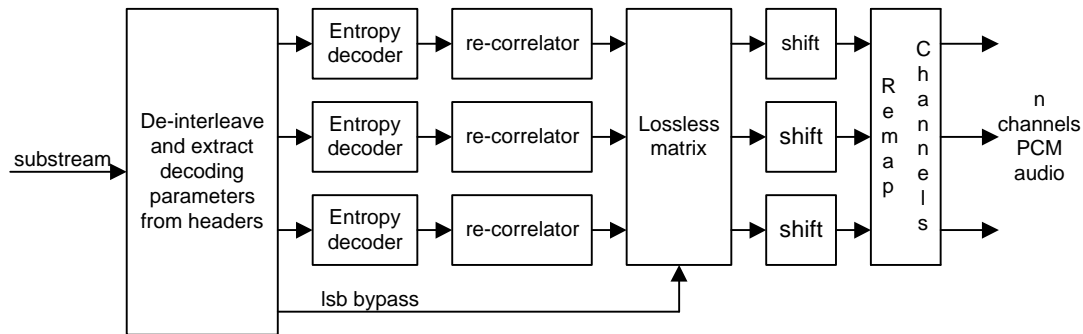


Figure 21: Block diagram of the lossless decoder core.

7 TWO CHANNEL DOWNMIX

It is often useful to provide a means for accessing high-resolution multichannel audio streams on 2-channel playback devices. In an application such as DVD-Audio, the content provider can place separate multi- and 2-channel streams on the disc. However to do this requires separate mix, mastering and authoring processes and uses disc capacity.

In cases where only one multichannel stream is available, then there are very few options at replay – one is to use either a fixed or guided downmix. However, to create such a downmix it is first necessary to decode the full multichannel signal; this contravenes the desirable principle that decoder complexity should decrease with functionality.

7.1 Performing mix-down in the lossless encoder

MLP provides an elegant and unique solution. The encoder combines lossless matrixing with the use of two substreams in such a way as to optimally encode *both* the Lt/Rt downmix *and* the multichannel version. This method is shown in Figure 22.

Downmix instructions are used to determine some coefficients for the lossless matrices. The matrices then perform a rotation such that the two channels on substream 0 decode to the desired stereo mix and combine with substream 1 to provide full multichannel.

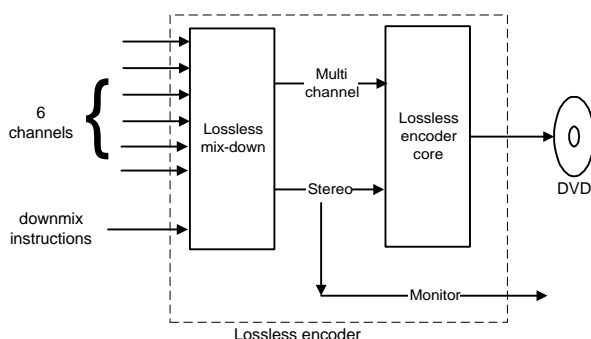


Figure 22: Illustrating encoder downmix.

Because the 2-channel downmix is a linear combination of the multichannel mix, then strictly, no new information has been added. In the example shown in Figure 22 there are still only six independent channels in the encoded stream. So, theoretically, the addition of the 2-channel version should require only a modest increase in overall data rate (typically 1 bit per sample, e.g. 96kbit/s at 96kHz). Figure 23 shows an example where a downmix is added to the 6-channel segment from Figure 16.

The advantages of this method are considerable:

- The quality of the mix-down is guaranteed. The producer can listen to it at the encoding stage and the lossless method delivers it bit-accurate to the end user.
- A 2-channel-only playback device does not need to decode the multichannel stream and then perform mix-down. Instead, the lossless decoder need only decode substream 0.
- A more complex decoder may access both the 2-channel and multichannel versions losslessly.
- The downmix coefficients do not have to be constant for a whole track, but can be varied under artistic control.

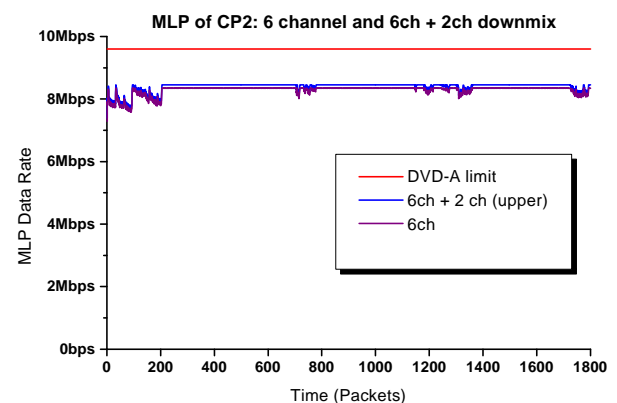


Figure 23: Showing the impact on data rate of adding a 2-channel downmix to 6-channel content.

8 MLP BITSTREAM FORMATS

The encoded stream carries all the information necessary to decode the stream. This information includes:

- Instructions to the decoder.
- Compressed data.
- Auxiliary data (content provider's information).
- CRC check information.
- Lossless testing information.

Incoming audio is encoded in segments and the bitstream uses a packet structure as follows:

- Data is encoded in blocks that typically contain between 40 and 160 samples.
- Blocks are assembled into packets. The user and/or the encoder can adjust the length of packets. A typical range is between 640 and 2560 samples.
- Each packet contains full initialisation and restart information. Therefore the decoder can recover from severe transmission errors, or start up losslessly in mid-stream typically within 7ms.

8.1 Error handling

MLP has powerful built-in detection that allows rapid recovery from bitstream errors, in addition:

- Errors cannot propagate beyond a packet boundary.
- Recovery from 1-bit errors generally occurs within 1-6ms.
- Multiple checks in the stream prevent erroneous noises, 'clicks' or 'bangs'.

8.2 Variable rate bitstream

A variable-rate MLP stream is packetised to minimise file size. The packetising method can ensure that the short-term peak data rate is kept as low as possible. Several examples of variable-rate streams have been given in this paper.

8.3 Fixed rate bitstream

The fixed-rate stream is packetised to provide losslessly compressed audio at a constant data rate. Encoding for fixed rate can be a single-pass process if the target data rate is always attainable.

At times when the compressed data rate is less than the target, the encoder will fold in padding data or transmit a pending payload of additional data (see section 12).

8.4 MLP stream transcoding

An MLP bitstream contains sufficient data to allow transcoding between fixed and variable-rate streams. Figure 24 shows circumstances in disc production and playback where transcoding may be useful. Transcoding is a lightweight operation, not requiring a full decode and re-encode.

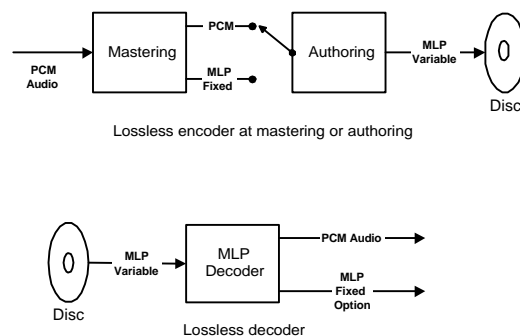


Figure 24: In a mastering or replay environment, it may be desirable to transcode MLP streams.

9 HOW MUCH COMPRESSION?

In specifying a lossy system, the critical compression measure is the final bit rate for a given perceptual quality, and this is independent of the input wordwidth. With lossless compression, increases in incoming precision (i.e. additional LSBs on the input) must be losslessly reproduced. However, these LSBs typically contain little redundancy that can be removed by an encoder and thus they contribute directly to the transmitted data rate. Therefore we tend to quote the *saving* of data rate, as this measure is relatively independent of incoming precision. The saving of data rate (in bits per original sample) is indicated in Table 1 below.¹

Table 1 Sampling kHz	Data-rate reduction: bits/sample/channel	
	Peak	Average
48	4	5–11
96	8	9–13
192	9	9–14

At 44.1 or 48 kHz, the peak data rate can almost always be reduced by at least 4 bits/sample, i.e. 16-bit audio can be losslessly compressed to fit into a 12-bit channel.

At 96kHz, the peak data rate can similarly be reduced by 8 bits/sample, i.e. 24-bit audio can be compressed to 16 bits and 16-bit 96kHz audio can be losslessly compressed to fit into an 8-bit channel.

¹ In Table 1 peak savings are for 'difficult' signals while average savings reflect the uncertainty introduced by quiet passages and other variables. Table 1 gives the compression for a 2-channel signal. Compression increases as more channels are added or if any channels are correlated, or have low bandwidth (like a subwoofer channel) or low occupancy (like a surround channel).

The important parameter for transmission applications is the reduction of the peak rate. In the case of DVD-Audio peak rate is a key parameter because the encoded stream must always operate below the audio buffer data-rate limit of 9.6Mbps.

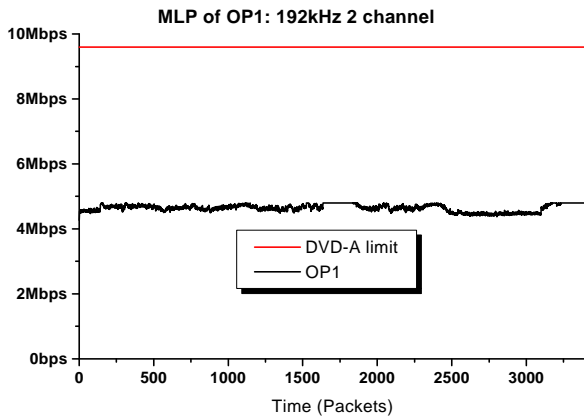


Figure 25: Compressed data rate for a 24-bit 2-channel item sampled at 192kHz.

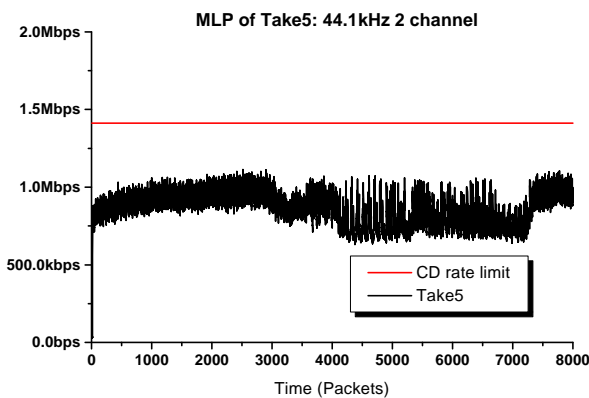


Figure 26: Compressed data rate for 'Take Five' – a 16-bit 2-channel 44.1kHz item from CD.

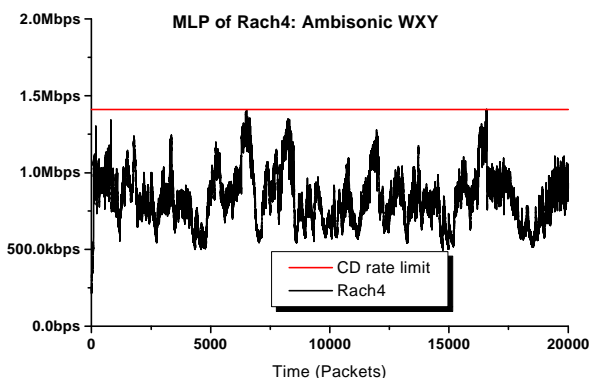


Figure 27: Compressed data rate for a horizontal ambisonic WXY 16-bit 3-channel 44.1kHz fragment compressed for delivery on CD.

The average number in Table 1 indicates the degree of compression that could be obtained when using MLP in an archive, mastering or editing environment. For example, a peak data rate reduction of 8 bits/sample means that a 96kHz 24-bit channel can be carried on the disc with a rate equal to that of a $24-8 = 16$ -bit LPCM channel. However the space used on the disc is estimated by the average saving, in this case the residual will be $24-11 = 13$ bits/channel.

Consider that an 11-bit saving represents a compression ratio of 1.85:1 with 24-bit material, whereas the same saving compresses 16-bit audio by 3.2:1!

Figure 25 shows a typical progression through 2-channel 192kHz 24-bit material (original data rate 9.216Mbps). Figures 26 and 27 show compression examples at CD quality. The 2-channel example in Figure 26 shows an average 2:1 compression.

Note that the 3-channel horizontal ambisonic B-format (WXY) stream in Figure 27 (opening of Rachmaninov's 2nd piano concerto) shows sufficient peak rate compression to allow the stream to fit on a CD.

9.1 Compression adjustment

A producer may wish to save space used by a recording, or to reduce data rate. Lossless compression extends the number of options.

With MLP, data is automatically saved if the incoming precision is reduced. So, reducing (for example) a few or all channels in a mix from 24 to 22-bit will provide an automatic data saving. The concept is illustrated in Figure 28. The authors have previously described appropriate quantising strategies. [10][11][12][13]

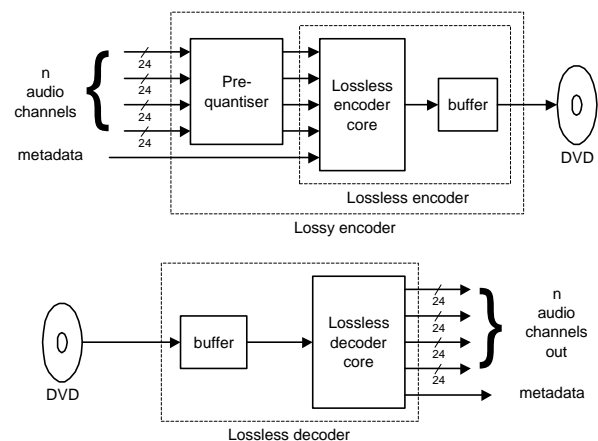


Figure 28: Generalised schematic of pre-quantisation showing a lossy-lossless encode and lossless decode.

In an overall sense the process of Figure 28 could be viewed as lossy – however this is *not* the case if the producer makes the adjustment. A conventional lossy system provides no choice about how the signal is modified in order to fit the desired data rate, whereas in Figure 28 the producer can use his artistic judgement to select both the method and amount of wordwidth reduction. The output of the quantiser can be monitored and that signal will be delivered losslessly by the MLP decoder.

This does not exclude the possibility that a quantiser for use with an MLP encoder could adapt incoming precision automatically – a circumstance envisaged by the authors in [9]. An intriguing property of a lossy encode made in this way is that it can be losslessly cascaded – that is it would be lossless over subsequent encode–decode passes.

Another option for reducing encoded data is to low-pass filter some of the incoming channels. Low-pass filtering reduces the entropy in the signal and the lossless coder generally provides a lower data rate.

A typical 96kHz 24-bit 6-channel program would encode to an average of 7.2Mbps. Reducing the audio bandwidth with simple filtering from 48kHz to 24kHz will generally reduce the rate to below 5Mbps.

10 FEATURES FOR CONTENT PROVIDERS

MLP allows the record producer to make a personal trade-off between playing time, frequency range, number of active channels and their precision. The packed channel conveys this choice implicitly in its control data, and the system operation is transparent to the user.

This method has the following example benefits:

- A producer mastering at 48kHz can control the incoming precision of each channel – and trade playing time or channels for noise-floor.
- A producer mastering at 96kHz or 192kHz can additionally trade bandwidth for playing time, active channels and precision.

For example:

- a) Playing time or precision may be extended by pre-filtering information above some arbitrary frequency (e.g. 30kHz), thereby allowing more compression.
- b) Playing time or precision may be extended by only supplying a 2, 3 or 4-channel mix.
- c) Feeding smaller word sizes to the encoder will extend playing time (e.g. reducing from 24 bits to 23 or 22 bits: each bit removed will increase playing time by around 8%).

MLP always returns the streams bit-for-bit intact once any mastering adjustments have been made.

10.1 DVD-Audio Content

MLP has some features that assist content providers in providing material for issue on DVD Audio, including:

- Longer playing time than allowed by LPCM.
- Higher quality by delivering more channels or bits for the same playing time.
- Guaranteed quality. The lossless decoder delivers bit-accurate data.
- High-quality mix-down options; longer playing time with multichannel material.
- Fine control over delivered quality and playing time.
- The large reduction in the audio data rate means that many more options for audio with pictures are possible.
- Additional data channel in the stream to carry copyright information.
- Additional data channel in the stream to carry signature information.
- Bitstream definition allows more than 6 channels for recording and archive.

10.2 Playing time on DVD-Audio

DVD Audio holds approximately 4.7Gbytes of data and has a maximum data transfer rate of 9.6Mbps for an audio stream.

Six channels of 96kHz 24-bit LPCM audio has a data rate of 13.824Mbps which is well in excess of 9.6Mbps. Also, at 13.824Mbps, the data capacity of the disc would be used up in approximately 45 minutes.

So, lossless compression is needed to reduce the data on the disc to extend playing time to the industry norm of 74 minutes *and* to guarantee a *minimum* reduction of 31% in instantaneous data rate.

MLP meets this requirement with a sophisticated encoder, a simple decoder and a specific subset of features limited to two substreams and 6 channels. [14] Here are some examples of playing times that can be obtained:

- 5.1 channels 96kHz 24-bit: 100 minutes.
- 6 channels 96kHz 24-bit: 86 minutes.
- 2 channels 96kHz 24-bit: 4 hours.
- 2 channels 192kHz 24-bit: 2 hours.
- 2 channels 44.1kHz 16-bit: 12 hours.
- 1 channel 44.1kHz 16-bit: 25 hours (talking book).

11 SYSTEM DEFINITION AND FLEXIBILITY

MLP was conceived as a general-purpose lossless compression system. However a high-density replacement for the consumer CD was foreseen as an early application, and this has driven the system design in two directions:

- Any complexity must be in the encoder rather than the decoder.
- The system is defined in terms of the bitstream and the required decoder behaviour.

As a result of the second point, encoder developments may continue (for example, for increased compression) without outdating the installed base of decoders. Current decoders are required to decode any legal bitstream, so there will be no question of 'old' decoders being unable to decode 'new' software.

The bitstream has been designed to keep open as many options as possible for future encoder developments, while not impacting decoder complexity and MIPS more than necessary.

While the highest compression requires sophisticated encoders, near optimal encoding of most music signals can be obtained with much simpler encoders that have modest MIPS requirements and can run in real time on cheaply available DSP devices. Thus future use in consumer record/playback systems is entirely feasible. Neither encoding nor decoding mandate the use of FFTs or other block processing, so it is also possible to construct encoders and decoders with very low latency, for use in radio microphones or other real-time applications.

12 SIGNAL AND METADATA

A design aim of MLP was to provide a simple external connectivity. An encoder has (conceptually) n identical input sockets and the corresponding decoder has n output sockets. Externally the system is just like an n -channel 24-bit PCM link.

Thus, there is no concept of a '5.1' channel or a '7.1' channel encoder or decoder. If a '5.1' channel signal is presented to a 6-channel encoder, the '1' channel will be recognised by the encoder as being highly predictable (on account of its low bandwidth) and should be encoded to an extremely low data rate, ideally about two bits per sample.

If someone were to invent a '4.2' multichannel format having two low frequency channels, this too would be automatically and optimally handled.

Likewise, there is no need for the wordwidth to be flagged to the encoder: if a 20-bit signal were presented to some or all channels of an MLP encoder, the four unused bits would be evident to the encoder and the appropriate economies made.

Channel meaning and wordwidth (discussed above) are examples of *metadata*. MLP regards metadata as

important and provides intact delivery of any metadata that is supplied along with the audio, however the feeding of metadata to the encoder is entirely optional (unless mandated by a particular application such as DVD-Audio) and in no way affects the handling of the audio signals.

12.1 MLP Metadata Specification

The MLP metadata specification is deliberately open-ended. Items that have been discussed include:

- Dynamic-range control data (should there be a wish to compress after a lossless decode).
- Ownership and copy protection fields.
- SPL reference.
- SMPTE timecode.
- Content signature.
- Provenance information for decoders.
- A Rosetta stone text field.

In a system in which the number of signal channels may be up to 63 in the future, it is hard to predict exactly what variations of 'channel meaning' data may be needed. Therefore, in designing the MLP metadata format:

- Fixed-length bitfields have been avoided.
- Hierarchical data-structures are supported.

The significance of the second item above may become apparent from the next section.

12.2 Future audio possibilities

In [15] the ARA highlighted the desirability of recording and storing multichannel signals in 'hierarchical' (MSTBF) or 'ambisonic' format (WXYZ), while recognising the need to matrix these signals to feeds for the standard '3+2' speaker layout before issuing on a consumer disc.

If the matrix coefficients are recorded within the MLP metadata, a suitable decoder can apply the inverse matrix and recover the original 'hierarchical' or 'ambisonic' signals. This is advantageous if it is desirable to use a different layout or number of speakers from the standard 3+2.

Moreover, using the same technique of *lossless matrixing* discussed in section 4.1, it is possible to re-create the original MSTBF or WXYZ signals in a *bit-exact* fashion.

Having done this, the enthusiast will then wish to have a metadata description of the original signal, hence the need for a hierarchical capability on the description language.

Further possibilities include *lossless equalisation*: having established in section 4.4 the possibility of a losslessly invertible IIR filter architecture, it is natural to apply this to equalisation. Thus a mastering engineer may choose to apply such equalisation to 'sweeten' a track for consumer release, but if the coefficients are

recorded as MLP metadata, it will be possible for the original signal to be recovered losslessly by an appropriate decoder.

Clearly a sophisticated audiophile decoder could unravel several nested layers of metadata description, and thus undo several cascaded stages of studio processing, to recover an original signal with bit-exact precision.

13 SUMMARY

The authors set out to develop a lossless coding system whose options enabled the highest possible audio quality in a hierarchical architecture that would allow future extensions.

First and foremost MLP is truly lossless and guarantees delivery of the original audio data. The decoder can confirm true end-to-end lossless operation.

Great attention has been paid to the audio compression strategies: a four-level approach incorporating novel lossless use of matrices, processing and IIR filters allows a high degree of compression at all times.

Because MLP will be used on carriers like DVD-Audio which have limited data rate, particular attention was also paid to methods that control the peak rate of the encoded bitstream.

The bitstream itself has been defined to allow robust operation, fast error recovery and rapid cueing (typically recovering in 7ms).

An unusual feature is the ability to use fixed or variable rate streams according to the application.

Following the sensible paradigm that as much system complexity as possible should be embodied in the encoder rather than the decoder, the MLP decoder is relatively simple. The decoder is also hierarchical, has a low computational complexity, is portable and is lossless over different hardware platforms.

Flexible encoding options include automatic adaptation to the bandwidth of incoming audio and to the incoming word-size in 1-bit steps.

In addition to audio, the MLP stream carries additional information of benefit to the decoder, to the content provider and to the end user. A flexible extensible hierarchical metadata option also allows very effective use of MLP in advanced surround applications.

14 REFERENCES

- [1] *Acoustic Renaissance for Audio*, 'A Proposal for High-Quality Application of High-Density CD Carriers', private publication (February 1995) Available for download at: <http://www.meridian-audio.com/ara>.
- [2] Shannon, C.E. 'A Mathematical Theory of Communication', *Bell System Technical Journal* vol 27 pp.379-423,623-656 (July, October 1948)
- [3] Robinson, A. 'Shorten: Simple lossless and near-lossless waveform compression.' Tech Rep CUED/F-INFENG/TR.156, Cambridge University, Cambridge, UK (Dec 1994).
- [4] Bruekers, AAML, Oomen, AWJ and van der Vleuten, RJ. 'Lossless Coding for DVD Audio.' *AES 101st Convention*, Los Angeles (Preprint #4358) (November 1996).
- [5] Cellier, C, Chenes, P and Rossi, M. 'Lossless Audio Bit Rate Reduction.' *AES UK Conference 'Managing the Bit Budget'*, pp107-122 (May 1994).
- [6] Craven, P.G. and Gerzon, M.A., 'Lossless Coding for Audio Discs', *J. Audio Eng. Soc.*, vol. 44 no. 9 pp. 706–720 (September 1996)
- [7] Craven, P.G. and Gerzon, M.A., 'Lossless Coding Method for Waveform Data', International Patent Application no. PCT/GB96/01164 (May 1996)
- [8] Craven, P.G., Law, M.J. and Stuart, J.R., 'Lossless Compression using IIR Prediction Filters', *J. Audio Eng. Soc.* (Abstracts), vol. 45 no. 5 p. 404 (Preprint #4415) (March 1997)
- [9] Craven, P.G. and Stuart, J.R., 'Cascadable Lossy Data Compression Using a Lossless Kernel', *J. Audio Eng. Soc.* (Abstracts), vol. 45 no. 5 p. 404 (Preprint #4416) (March 1997)
- [10] Craven, PG and Gerzon, MA. 'Optimal Noise Shaping and Dither of Digital Signals.' presented at *AES 87th Convention*, New York (Preprint #2822) (October 1989).
- [11] Gerzon, M.A., Craven, P.G., Stuart, J.R. and Wilson, R.J., 'Psychoacoustic Noise Shaped Improvements in CD and Other Linear Digital Media', *AES 94th Convention*, Berlin, (Preprint #3501) (March 1993)

- [12] Stuart, J.R. and Wilson, R.J., 'Dynamic Range Enhancement Using Noise-shaped Dither Applied to Signals with and without Pre-emphasis', *AES 96th Convention*, Amsterdam, (Preprint #3871) (1994)
- [13] Stuart, J.R. and Wilson, R.J., 'Dynamic Range Enhancement using Noise-Shaped Dither at 44.1, 48 and 96 kHz', *AES 100th Convention*, Copenhagen (May 1996)
- [14] Toshiba et al., 'DVD Specifications for Read-Only Disc, Part 4: Audio Specification' Version 1.0 (March 1999)
- [15] *Acoustic Renaissance for Audio*, 'DVD: Application of Hierarchically Encoded Surround Sound – including Ambisonics', private publication (October 1996). Available for download: <http://www.meridian-audio.com/ara>.

15 PATENT NOTICE

Several aspects of the MLP encode, decode, packetising and bitstream are the subject of patent application.

16 ACKNOWLEDGEMENTS

We are grateful to Tony Faulkner (Green Room Productions), Warner Music, JVC and Nimbus Records for providing experimental recordings and material used in the compression examples.

Meridian, *Meridian Lossless Packing* and *MLP* are trademarks of Meridian Audio Ltd.