

JOHN WATKINSON

MPEG-2



Hayes



Focal Press

MPEG-2


John Watkinson



Focal Press

OXFORD AUCKLAND BOSTON JOHANNESBURG MELBOURNE NEW DELHI

Focal Press
An imprint of Butterworth-Heinemann
Linacre House, Jordan Hill, Oxford OX2 8DP
225 Wildwood Avenue, Woburn, MA 01801-2041
A division of Reed Educational and Professional Publishing Ltd

 A member of the Reed Elsevier plc group

First published 1999

© John Watkinson 1999

All rights reserved. No part of this publication may be reproduced in any material form (including photocopying or storing in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication) without the written permission of the copyright holder except in accordance with the provisions of the Copyright, Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd, 90 Tottenham Court Road, London, England W1P 9HE. Applications for the copyright holder's written permission to reproduce any part of this publication should be addressed to the publishers

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

Library of Congress Cataloguing in Publication Data

A catalogue record for this book is available from the Library of Congress

ISBN 0 240 51510 2

Composition by Genesis Typesetting, Rochester, Kent
Printed and bound in Great Britain

PLANT A
TREE



FOR EVERY TITLE THAT WE PUBLISH, BUTTERWORTH-HEINEMANN
WILL PAY FOR BTCV TO PLANT AND CARE FOR A TREE.

Contents

<i>Preface</i>	ix
<i>Acknowledgements</i>	xi
Chapter 1 Introduction to compression	1
1.1 What is MPEG-2?	1
1.2 Why compression is necessary	3
1.3 Some applications of MPEG-2	4
1.4 Lossless and perceptive coding	6
1.5 Compression principles	8
1.6 Audio compression	12
1.6.1 Sub-band coding	12
1.6.2 Transform coding	13
1.6.3 Predictive coding	13
1.7 Video compression	13
1.7.1 Intra-coded compression	15
1.7.2 Inter-coded compression	15
1.7.3 Introduction to motion compensation	17
1.7.4 Film-originated video compression	18
1.8 MPEG-2 profiles and levels	20
1.9 MPEG-2 bitstreams	22
1.10 Drawbacks of compression	23
1.11 Compression preprocessing	24
1.12 Some guidelines	25
References	26

Chapter 2	Fundamentals	27
2.1	What is an audio signal?	27
2.2	What is a video signal?	27
2.3	Types of video	28
2.4	What is a digital signal?	30
2.5	Introduction to conversion	33
2.6	Sampling and aliasing	34
2.7	Reconstruction	37
2.8	Filter design	39
2.9	Sampling clock jitter	42
2.10	Choice of audio sampling rate	44
2.11	Video sampling structures	46
2.12	The phase-locked loop	49
2.13	Quantizing	50
2.14	Quantizing error	53
2.15	Dither	56
2.16	Binary codes for audio	58
2.17	Binary codes for component video	64
2.18	Introduction to digital processes	65
2.19	Logic elements	66
2.20	Storage elements	68
2.21	Binary adding	70
2.22	Gain control by multiplication	73
2.23	Multiplexing principles	74
2.24	Packets	75
2.25	Statistical multiplexing	75
	References	76
Chapter 3	Processing for compression	77
3.1	Filters	77
3.2	Downsampling filters	83
3.3	The quadrature mirror filter	87
3.4	Filtering for video noise reduction	91
3.5	Transforms	92
3.6	The Fourier transform	96
3.7	The discrete cosine transform (DCT)	104
3.8	The wavelet transform	108
3.9	Motion compensation	110
3.10	Motion-estimation techniques	111
3.10.1	Block matching	111
3.10.2	Gradient matching	112
3.10.3	Phase correlation	113

3.11	Compression and requantizing	118
	References	123
Chapter 4 Audio compression		124
4.1	Introduction	124
4.2	The ear	125
4.3	The cochlea	126
4.4	Level and loudness	128
4.5	Frequency discrimination	130
4.6	Critical bands	131
4.7	Beats	133
4.8	Codec level calibration	136
4.9	Quality measurement	137
4.10	The limits	138
4.11	Compression applications	139
4.12	History of MPEG audio coding	139
4.13	MPEG audio compression tools	141
4.14	Transform coding	144
4.15	MPEG Layer I audio coding	146
4.16	MPEG Layer II audio coding	150
4.17	MPEG Layer III audio coding	151
4.18	Dolby AC-3	151
4.19	Compression in stereo	153
	References	159
Chapter 5 MPEG-2 video compression		160
5.1	The eye	160
5.2	Dynamic resolution	164
5.3	Contrast	168
5.4	Colour vision	169
5.5	Colour difference signals	171
5.6	Progressive or interlaced scan?	174
5.7	Spatial and temporal redundancy in MPEG	179
5.8	<i>I</i> and <i>P</i> coding	183
5.9	Bidirectional coding	184
5.10	Coding applications	187
5.11	Spatial compression	188
5.12	Scanning and run-length/variable-length coding	192
5.13	A bidirectional coder	197
5.14	Slices	200

5.15	Handling interlaced pictures	201
5.16	An MPEG-2 coder	206
5.17	The Elementary Stream	208
5.18	An MPEG-2 decoder	209
5.19	Coding artifacts	212
5.20	Processing MPEG-2 and concatenation	214
	References	221
 Chapter 6 Program and transport streams		222
6.1	Introduction	222
6.2	Packets and time stamps	222
6.3	Transport streams	225
6.4	Clock references	226
6.5	Program Specific Information (PSI)	228
6.6	Multiplexing	229
6.7	Remultiplexing	231
	Reference	232
 <i>Glossary</i>		233
 <i>Index</i>		239

Preface

Digital compression technology has been employed for a long time, but until recently the technology was too complex for everyday applications. The onward march of LSI technology means that increasingly complex processes become available at moderate cost. Digital compression has now reached the stage where it can economically be applied to video and audio systems on a wide scale. MPEG-2 has assisted in the process by providing interoperability standards for audio and video compression and communication.

Economic compression does more than make existing processes cheaper. It allows new developments which were impossible without it. This book recognizes the wide application of MPEG by treating the subject from first principles without assuming any particular background for the reader. MPEG is not the only compression technique and other approaches will be mentioned here for completeness. While compression is traditionally described mathematically, it is the author's view that mathematics is no more than a technical form of shorthand *describing* a process. As such it cannot *explain* anything and has no place in a book of this kind where instead the explanations are in plain English.

An introductory chapter is included which suggests some applications of MPEG and how it works in a simplified form. Additionally a fundamentals chapter contains all the background necessary to follow the rest of the book.

Compression theory is balanced with a great deal of practice. The creation of MPEG Elementary Streams and their multiplexing into transport streams is dealt with in detail along with the problems involved in synchronizing all the signals in a multiplex. The practical testing of MPEG transport streams is also considered.

Throughout this book the reader will find notes of caution and outlines of various pitfalls for the unwary. Compression is a useful tool in certain applications but it is a dangerous master and if used indiscriminately the results are bound to be disappointing. Unfortunately there are signs that this is happening. Various descriptions of kinds of artifacts and impairments which result from the misuse of compression are included here. Compression is rather like a box of fireworks; used wisely a pleasing display results – used carelessly it can blow up in your face.

To suggest that compression should never be used to ensure that disasters are avoided is as naive as trying to ban fireworks. The solution is the same: to explain the dangers and propose safe practices. Retire to a safe place and read this book.

John Watkinson
Burghfield Common 1999

Acknowledgements

Information for this book has come from a range of sources to whom I am indebted. I would specifically mention the publications of the ISO, the AES and SMPTE which provided essential groundwork. I have had numerous discussions with Peter de With of Philips, Steve Lyman of CBC, Bruce Devlin and Mike Knee of Snell and Wilcox and Peter Kraniauskas which have all been extremely useful. The assistance of MicroSoft Corp. and Tektronix Inc. is also appreciated.

1

Introduction to compression

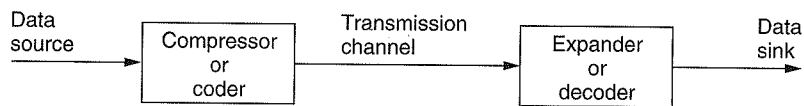
1.1 What is MPEG-2?

MPEG is actually an acronym for the moving pictures experts group which was formed by the ISO (International Standards Organization) to set standards for audio and video compression and transmission. The first compression standard for audio and video was MPEG-1^{1,2} but this was of limited application and the subsequent MPEG-2 standard was considerably broader in scope and of wider appeal.³ For example, MPEG-2 supports interlace whereas MPEG-1 did not.

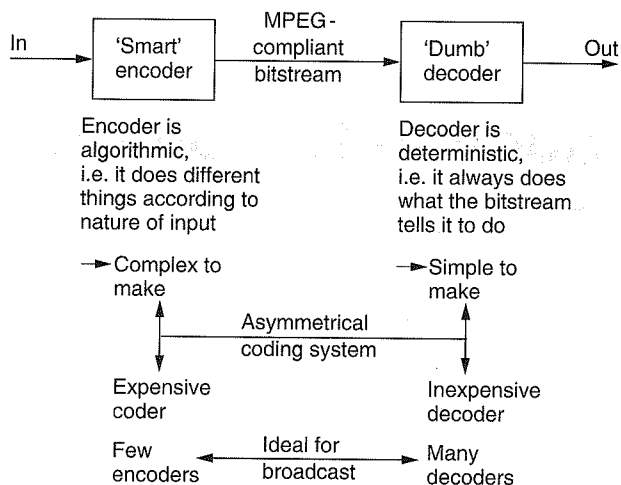
Compression is summarized in Figure 1.1. It will be seen in (a) that the data rate is reduced at source by the *compressor*. The compressed data are then passed through a communication channel and returned to the original rate by the *expander*. The ratio between the source data rate and the channel data rate is called the *compression factor*. The term *coding gain* is also used. Sometimes a compressor and expander in series are referred to as a *combander*. The compressor may equally well be referred to as a *coder* and the expander a *decoder* in which case the tandem pair may be called a *codec*.

In audio and video compression, where the encoder is more complex than the decoder the system is said to be asymmetrical. Figure 1.1(b) shows that MPEG works in this way. The encoder needs to be algorithmic or adaptive whereas the decoder is 'dumb' and carries out fixed actions. This is advantageous in applications such as broadcasting where the number of expensive complex encoders is small but the number of simple inexpensive decoders is large. In point-to-point applications the advantage of asymmetrical coding is not so great.

The approach of the ISO to standardization in MPEG is novel because it is not the encoder which is standardized. Figure 1.2(a) shows that instead the way in which a decoder shall interpret the bitstream is



(a)



(b)

Figure 1.1 In (a) a compression system consists of compressor or coder, a transmission channel and a matching expander or decoder. The combination of coder and decoder is known as a codec. (b) MPEG is asymmetrical since the encoder is much more complex than the decoder.

defined. A decoder which can successfully interpret the bitstream is said to be *compliant*. Figure 1.2(b) shows that the advantage of standardizing the decoder is that over time encoding algorithms can improve yet compliant decoders will continue to function with them.

Manufacturers can supply encoders using algorithms which are proprietary and their details do not need to be published. A useful result is that there can be competition between different encoder designs which means that better designs will evolve. The user will have greater choice because different levels of cost and complexity can exist in a range of coders yet a compliant decoder will operate with them all.

MPEG is, however, much more than a compression scheme as it also standardizes the protocol and syntax under which it is possible to combine or multiplex audio data with video data to produce a digital equivalent of a television program. Many such programs can be combined in a single multiplex and MPEG defines the way in which such multiplexes can be created and transported. The definitions include the metadata which decoders require to demultiplex correctly and which users will need to locate programs of interest.

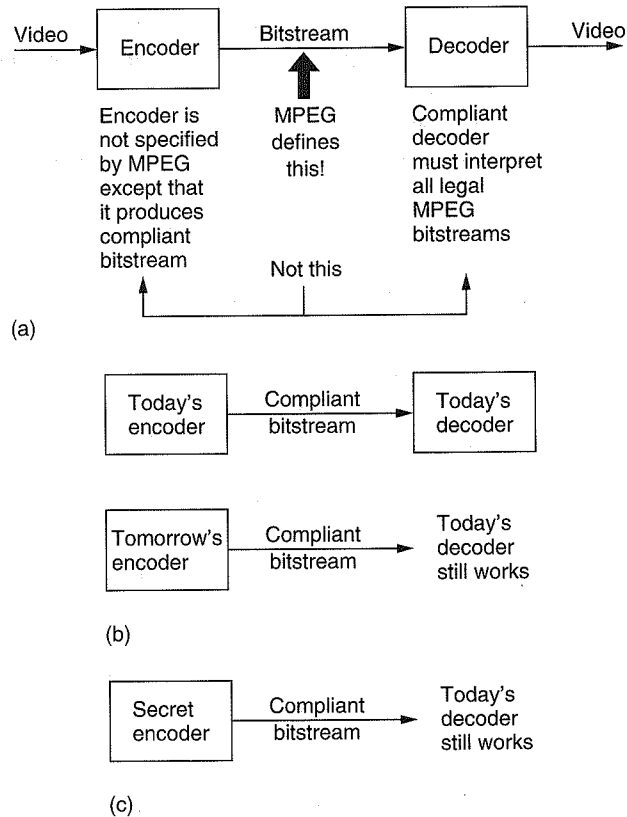


Figure 1.2 (a) MPEG defines the protocol of the bitstream between encoder and decoder. The decoder is defined by implication, the encoder is left very much to the designer. (b) This approach allows future encoders of better performance to remain compatible with existing decoders. (c) This approach also allows an encoder to produce a standard bitstream while its technical operation remains a commercial secret.

As with all video systems there is a requirement for synchronizing or genlocking and this is particularly complex when a multiplex is assembled from many signals which are not necessarily synchronized to one another.

1.2 Why compression is necessary

Compression, bit rate reduction and data reduction are all terms which mean basically the same thing in this context. In essence the same (or nearly the same) information is carried using a smaller quantity or rate of data. It should be pointed out that, in audio, *compression* traditionally means a process in which the dynamic range of the sound is reduced. In

the context of MPEG the same word means that the bit rate is reduced, ideally leaving the dynamics of the signal unchanged. Provided the context is clear, the two meanings can co-exist without a great deal of confusion.

There are several reasons why compression techniques are popular:

1. Compression extends the playing time of a given storage device.
2. Compression allows miniaturization. With less data to store, the same playing time is obtained with smaller hardware. This is useful in ENG (electronic news gathering) and consumer devices.
3. Tolerances can be relaxed. With less data to record, storage density can be reduced making equipment which is more resistant to adverse environments and which requires less maintenance.
4. In transmission systems, compression allows a reduction in bandwidth which will generally result in a reduction in cost. It may make possible some process which would be impracticable without it.
5. If a given bandwidth is available to an uncompressed signal, compression allows faster than real-time transmission in the same bandwidth.
6. If a given bandwidth is available, compression allows a better-quality signal in the same bandwidth.

1.3 Some applications of MPEG-2

The applications of audio and video compression are limitless and the ISO has done well to provide standards which are appropriate to the wide range of possible compression products. MPEG-2 embraces video pictures from the tiny screen of a videophone to the high-definition images needed for electronic cinema. Audio coding stretches from speech grade mono to multichannel surround sound.

Figure 1.3 shows the use of a codec with a recorder. The playing time of the medium is extended in proportion to the compression factor. In the case of tapes, the access time is improved because the length of tape needed for a given recording is reduced and so it can be rewound more quickly.

In the case of DVD (Digital Video Disk aka Digital Versatile Disk) the challenge was to store an entire movie on one 12 cm disk. The storage density available with today's optical disk technology is such that recording of conventional uncompressed video would be out of the question.

In communications, the cost of data links is often roughly proportional to the data rate and so there is simple economic pressure to use a high compression factor. However, it should be borne in mind that implement-

ing the codec also has a cost which rises with compression factor and so a degree of compromise will be inevitable.

In the case of Video-on-Demand, technology exists to convey full bandwidth video to the home, but to do so for a single individual at the moment would be prohibitively expensive. Without compression, HDTV (high-definition television) requires too much bandwidth. With compression, HDTV can be transmitted to the home in a similar bandwidth to an existing analog SDTV channel. Compression does not make Video-on-Demand or HDTV possible, it makes them economically viable.

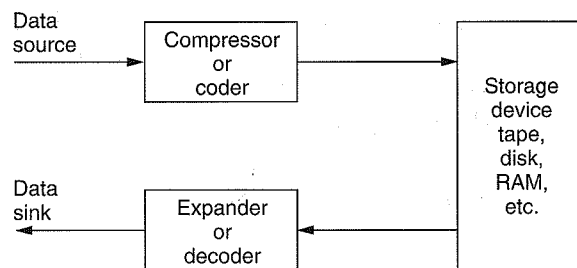


Figure 1.3 Compression can be used around a recording medium. The storage capacity may be increased or the access time reduced according to the application.

In workstations designed for the editing of audio and/or video, the source material is stored on hard disks for rapid access. While top-grade systems may function without compression, many systems use compression to offset the high cost of disk storage. When a workstation is used for *off-line* editing, a high compression factor can be used and artifacts will be visible in the picture.

This is of no consequence as the picture is only seen by the editor who uses it to make an EDL (Edit Decision List) which is no more than a list of actions and the timecodes at which they occur. The original uncompressed material is then *conformed* to the EDL to obtain a high-quality edited work. When *on-line* editing is being performed, the output of the workstation is the finished product and clearly a lower compression factor will have to be used.

Perhaps it is in broadcasting where the use of compression will have its greatest impact. There is only one electromagnetic spectrum and pressure from other services such as cellular telephones makes efficient use of bandwidth mandatory. Analog television broadcasting is an old technology and makes very inefficient use of bandwidth. Its replacement by a compressed digital transmission will be inevitable for the practical reason that the bandwidth is needed elsewhere.

Fortunately, in broadcasting there is a mass market for decoders and these can be implemented as low-cost integrated circuits. Fewer encoders are needed and so it is less important if these are expensive. While the cost of digital storage goes down year on year, the cost of electromagnetic spectrum goes up. Consequently in the future the pressure to use compression in recording will ease whereas the pressure to use it in radio communications will increase.

1.4 Lossless and perceptive coding

Although there are many different coding techniques, all of them fall into one or other of these categories. In *lossless* coding, the data from the expander are identical bit-for-bit with the original source data. The so-called 'stacker' programs which increase the apparent capacity of disk drives in personal computers use lossless codecs. Clearly with computer programs the corruption of a single bit can be catastrophic. Lossless coding is generally restricted to compression factors of around 2:1.

It is important to appreciate that a lossless coder cannot guarantee a particular compression factor and the communications link or recorder used with it must be able to function with the variable output data rate. Source data which result in poor compression factors on a given codec are described as *difficult*. It should be pointed out that the difficulty is often a function of the codec. In other words, data which one codec finds difficult may not be found difficult by another. Lossless codecs can be included in bit-error-rate testing schemes. It is also possible to cascade or *concatenate* lossless codecs without any special precautions.

In *lossy* coding data from the expander are not identical bit-for-bit with the source data and as a result comparing the input with the output is bound to reveal differences. Lossy codecs are not suitable for computer data, but are used in MPEG as they allow greater compression factors than lossless codecs. Successful lossy codecs are those in which the errors are arranged so that a human viewer or listener finds them subjectively difficult to detect. Thus lossy codecs must be based on an understanding of psychoacoustic and psychovisual perception and are often called *perceptive* codes.

In perceptive coding, the greater the compression factor required, the more accurately must the human senses be modelled. Perceptive coders can be forced to operate at a fixed compression factor. This is convenient for practical transmission applications where a fixed data rate is easier to handle than a variable rate. The result of a fixed compression factor is that the subjective quality can vary with the 'difficulty' of the input material. Perceptive codecs should not be concatenated indiscriminately especially if they use different algorithms. As the reconstructed signal from a

perceptive codec is not bit-for-bit accurate, clearly such a codec cannot be included in any bit-error-rate testing system as the coding differences would be indistinguishable from real errors.

Although the adoption of digital techniques is recent, compression itself is as old as television. Figure 1.4 shows some of the compression techniques used in traditional television systems.

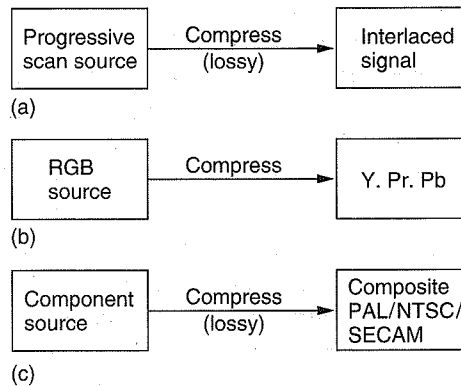


Figure 1.4 Compression is as old as television. (a) Interlace is a primitive way of halving the bandwidth. (b) Colour difference working invisibly reduces colour resolution. (c) Composite video transmits colour in the same bandwidth as monochrome.

One of the oldest techniques is interlace which has been used in analog television from the very beginning as a primitive way of reducing bandwidth. As will be seen in Chapter 5, interlace is not without its problems, particularly in motion rendering. MPEG-2 supports interlace simply because legacy interlaced signals exist and there is a requirement to compress them. This should not be taken to mean that it is a good idea.

The generation of colour difference signals from RGB in video represents an application of perceptive coding. The human visual system (HVS) sees no change in quality although the bandwidth of the colour difference signals is reduced. This is because human perception of detail in colour changes is much less than in brightness changes. This approach is sensibly retained in MPEG.

Composite video systems such as PAL, NTSC and SECAM are all analog compression schemes which embed a subcarrier in the luminance signal so that colour pictures are available in the same bandwidth as monochrome. In comparison with a progressive scan RGB picture, interlaced composite video has a compression factor of 6:1.

In a sense MPEG-2 can be considered to be a modern digital equivalent of analog composite video as it has most of the same attributes. For example, the eight-field sequence of PAL subcarrier which makes editing difficult has its equivalent in the GOP (group of pictures) of MPEG.

1.5 Compression principles

In a PCM digital system the bit rate is the product of the sampling rate and the number of bits in each sample and this is generally constant. Nevertheless the *information* rate of a real signal varies. In all real signals, part of the signal is obvious from what has gone before or what may come later and a suitable receiver can predict that part so that only the true information actually has to be sent. If the characteristics of a predicting receiver are known, the transmitter can omit parts of the message in the knowledge that the receiver has the ability to recreate it. Thus all encoders must contain a model of the decoder.

One definition of information is that it is the unpredictable or surprising element of data. Newspapers are a good example of information because they only mention items which are surprising. Newspapers never carry items about individuals who have *not* been involved in an accident as this is the normal case. Consequently the phrase 'no news is good news' is remarkably true because if an information channel exists but nothing has been sent then it is most likely that nothing remarkable has happened.

The unpredictability of the punch line is a useful measure of how funny a joke is. Often the build-up paints a certain picture in the listener's imagination, which the punch line destroys utterly. One of the author's favourites is the one about the newly married couple who didn't know the difference between putty and petroleum jelly – their windows fell out.

The difference between the information rate and the overall bit rate is known as the redundancy. Compression systems are designed to eliminate as much of that redundancy as practicable or perhaps affordable. One way in which this can be done is to exploit statistical predictability in signals. The information content or *entropy* of a sample is a function of how different it is from the predicted value. Most signals have some degree of predictability. A sine wave is highly predictable, because all cycles look the same. According to Shannon's theory, any signal which is totally predictable carries no information. In the case of the sine wave this is clear because it represents a single frequency and so has no bandwidth.

At the opposite extreme a signal such as noise is completely unpredictable and as a result all codecs find noise *difficult*. There are two consequences of this characteristic. First, a codec which is designed using the statistics of real material should not be tested with random noise

because it is not a representative test. Second, a codec which performs well with clean source material may perform badly with source material containing superimposed noise. Most practical compression units require some form of pre-processing before the compression stage proper and appropriate noise reduction should be incorporated into the pre-processing if noisy signals are anticipated. It will also be necessary to restrict the degree of compression applied to noisy signals.

All real signals fall part way between the extremes of total predictability and total unpredictability or noisiness. If the bandwidth (set by the sampling rate) and the dynamic range (set by the wordlength) of the transmission system are used to delineate an area, this sets a limit on the information capacity of the system. Figure 1.5(a) shows that most real signals only occupy part of that area. The signal may not contain all frequencies, or it may not have full dynamics at certain frequencies.

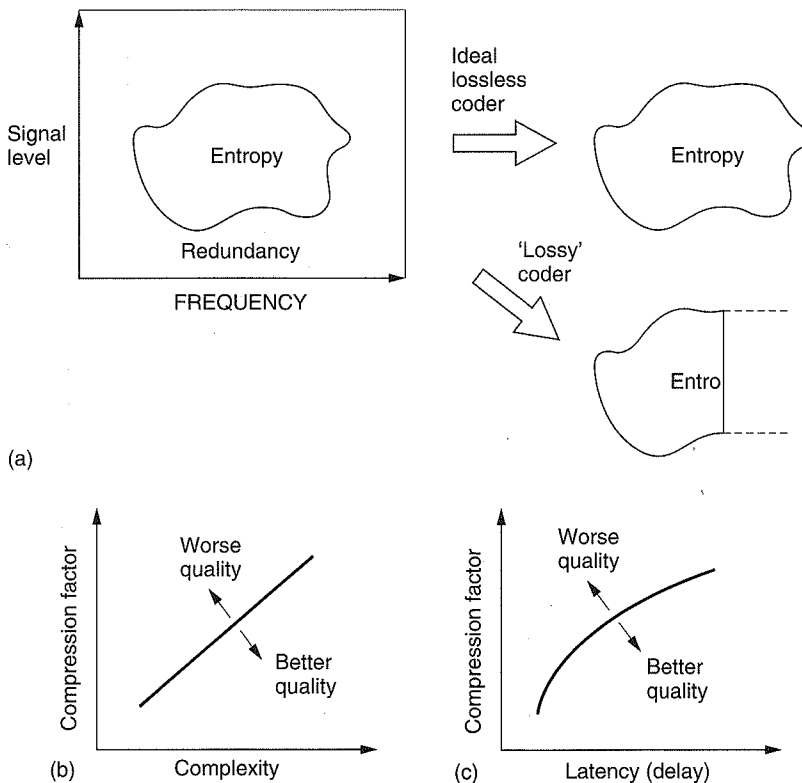


Figure 1.5 (a) A perfect coder removes only the redundancy from the input signal and results in subjectively lossless coding. If the remaining entropy is beyond the capacity of the channel some of it must be lost and the codec will then be lossy. An imperfect coder will also be lossy as it fails to keep all entropy. (b) As the compression factor rises, the complexity must also rise to maintain quality. (c) High compression factors also tend to increase latency or delay through the system.

Entropy can be thought of as a measure of the actual area occupied by the signal. This is the area that *must* be transmitted if there are to be no subjective differences or *artifacts* in the received signal. The remaining area is called the *redundancy* because it adds nothing to the information conveyed. Thus an ideal coder could be imagined which miraculously sorts out the entropy from the redundancy and only sends the former. An ideal decoder would then recreate the original impression of the information quite perfectly.

As the ideal is approached, the coder complexity and the latency or delay both rise. Figure 1.5(b) shows how complexity increases with compression factor and (c) shows how increasing the codec latency can improve the compression factor. Obviously we would have to provide a channel which could accept whatever entropy the coder extracts in order to have transparent quality. As a result moderate coding gains which only remove redundancy need not cause artifacts and result in systems which are described as *subjectively lossless*.

If the channel capacity is not sufficient for that, then the coder will have to discard some of the entropy and with it useful information. Larger coding gains which remove some of the entropy must result in artifacts. It will also be seen from Figure 1.5 that an imperfect coder will fail to separate the redundancy and may discard entropy instead, resulting in artifacts at a sub-optimal compression factor.

A single variable-rate transmission or recording channel is inconvenient and unpopular with channel providers because it is difficult to police. The requirement can be overcome by combining several compressed channels into one constant rate transmission in a way which flexibly allocates data rate between the channels. Provided the material is unrelated, the probability of all channels reaching peak entropy at once is very small and so those channels which are at one instant passing easy material will free up transmission capacity for those channels which are handling difficult material. This is the principle of statistical multiplexing.

Where the same type of source material is used consistently, e.g. English text, then it is possible to perform a statistical analysis on the frequency with which particular letters are used. Variable-length coding is used in which frequently used letters are allocated short codes and letters which occur infrequently are allocated long codes. This results in a lossless code. The well-known Morse code used for telegraphy is an example of this approach. The letter e is the most frequent in English and is sent with a single dot.

An infrequent letter such as z is allocated a long complex pattern. It should be clear that codes of this kind which rely on a prior knowledge of the statistics of the signal are only effective with signals actually having those statistics. If Morse code is used with another language, the

system is used which can learn the statistics as it goes along. The Lempel–Ziv–Welch (LZW) lossless codes are in this category. These codes build up a conversion table between frequent long source data strings and short transmitted data codes at both coder and decoder and initially their compression factor is below unity as the contents of the conversion tables are transmitted along with the data. However, once the tables are established, the coding gain more than compensates for the initial loss. In some applications, a continuous analysis of the frequency of code selection is made and if a data string in the table is no longer being used with sufficient frequency it can be deselected and a more common string substituted.

Lossless codes are less common for audio and video coding where perceptive codes are permissible. The perceptive codes often obtain a coding gain by shortening the wordlength of the data representing the signal waveform. This must increase the noise level and the trick is to ensure that the resultant noise is placed at frequencies where human senses are least able to perceive it. As a result although the received signal is measurably different from the source data, it can *appear* the same to the human listener or viewer at moderate compression factors. As these codes rely on the characteristics of human sight and hearing, they can only be fully tested subjectively.

The compression factor of such codes can be set at will by choosing the wordlength of the compressed data. While mild compression will be undetectable, with greater compression factors, artefacts become noticeable. Figure 1.5 shows that this is inevitable from entropy considerations.

1.6 Audio compression

Perceptive coding in audio relies on the principle of auditory masking, which is treated in detail in section 4.1. Masking causes the ear/brain combination to be less sensitive to sound at one frequency in the presence of another at a nearby frequency. If a first tone is present in the input, then it will mask signals of lower level at nearby frequencies. The quantizing of the first tone and of further tones at those frequencies can be made coarser. Fewer bits are needed and a coding gain results. The increased quantizing error is allowable if it is masked by the presence of the first tone.

1.6.1 Sub-band coding

Sub-band coding mimics the frequency analysis mechanism of the ear and splits the audio spectrum into a large number of different bands. Signals in these bands can then be quantized independently. The

quantizing error which results is confined to the frequency limits of the band and so it can be arranged to be masked by the program material. The techniques used in Layers 1 and 2 of MPEG audio are based on sub-band coding as are those used in DCC (Digital Compact Cassette).

1.6.2 Transform coding

In transform coding the time-domain audio waveform is converted into a frequency domain representation such as a Fourier, discrete cosine or wavelet transform (see Chapter 3). Transform coding takes advantage of the fact that the amplitude or envelope of an audio signal changes relatively slowly and so the coefficients of the transform can be transmitted relatively infrequently. Clearly such an approach breaks down in the presence of transients and adaptive systems are required in practice. Transients cause the coefficients to be updated frequently whereas in stationary parts of the signal such as sustained notes the update rate can be reduced. Discrete cosine transform (DCT) coding is used in Layer III of MPEG audio and in the compression system of the Sony MiniDisc.

1.6.3 Predictive coding

In a predictive coder there are two identical predictors, one in the coder and one in the decoder. Their job is to examine a run of previous sample code values and to extrapolate forward to estimate or predict what the next code value will be. This is subtracted from the *actual* next code value at the encoder to produce a prediction error which is transmitted. The decoder then adds the prediction error to its own prediction to obtain the output code value again. Predictive coders work with a short encode and decode delay and are useful in telephony where a long loop delay causes problems.

1.7 Video compression

Video signals exist in four dimensions: these are the attributes of the sample, the horizontal and vertical spatial axes and the time axis. Compression can be applied in any or all of those four dimensions. MPEG-2 assumes 8-bit colour difference signals as the input, requiring rounding if the source is 10-bit. The sampling rate of the colour signals is less than that of the luminance. This is done by downsampling the colour samples horizontally and generally vertically as well. Essentially an

MPEG-2 system has three parallel simultaneous channels, one for luminance and two colour difference, which after coding are multiplexed into a single bitstream.

Figure 1.7(a) shows that when individual pictures are compressed without reference to any other pictures, the time axis does not enter the process which is therefore described as *intra-coded* (intra = within) compression. The term *spatial coding* will also be found. It is an advantage of intra-coded video that there is no restriction to the editing which can be carried out on the picture sequence. As a result compressed VTRs such as Digital Betacam, DVC and D-9 use spatial coding. Cut editing may take place on the compressed data directly if necessary. As spatial coding treats each picture independently, it can employ certain techniques developed for the compression of still pictures. The ISO JPEG (Joint Photographic Experts Group) compression standards^{5,6} are in this category. Where a succession of JPEG coded images are used for television, the term 'Motion JPEG' will be found.

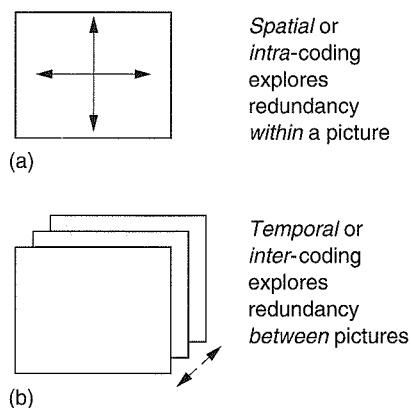


Figure 1.7 (a) Spatial or intra-coding works on individual images. (b) Temporal or inter-coding works on successive images.

Greater compression factors can be obtained by taking account of the redundancy from one picture to the next. This involves the time axis, as Figure 1.7(b) shows, and the process is known as *inter-coded* (inter = between) or *temporal* compression.

Temporal coding allows a higher compression factor, but has the disadvantage that an individual picture may exist only in terms of the differences from a previous picture. Clearly editing must be undertaken with caution and arbitrary cuts simply cannot be performed on the MPEG bitstream. If a previous picture is removed by an edit, the difference data will then be insufficient to recreate the current picture.

1.7.1 Intra-coded compression

Intra-coding works in three dimensions on the horizontal and vertical spatial axes and on the sample values. Analysis of typical television pictures reveals that while there is a high spatial frequency content due to detailed areas of the picture, there is a relatively small amount of energy at such frequencies. Often pictures contain sizeable areas in which the same or similar pixel values exist. This gives rise to low spatial frequencies. The average brightness of the picture results in a substantial zero-frequency component. Simply omitting the high-frequency components is unacceptable as this causes an obvious softening of the picture.

A coding gain can be obtained by taking advantage of the fact that the amplitude of the spatial components falls with frequency. It is also possible to take advantage of the eye's reduced sensitivity to noise in high spatial frequencies. If the spatial frequency spectrum is divided into frequency bands the high-frequency bands can be described by fewer bits not only because their amplitudes are smaller but also because more noise can be tolerated. The wavelet transform and the discrete cosine transform used in MPEG allows two-dimensional pictures to be described in the frequency domain and these are discussed in Chapter 3.

1.7.2 Inter-coded compression

Inter-coding takes further advantage of the similarities between successive pictures in real material. Instead of sending information for each picture separately, inter-coders will send the difference between the previous picture and the current picture in a form of differential coding. Figure 1.8 shows the principle. A picture store is required at the coder to allow comparison to be made between successive pictures and a similar store is required at the decoder to make the previous picture available. The difference data may be treated as a picture itself and subjected to some form of transform-based spatial compression.

The simple system of Figure 1.8(a) is of limited use as in the case of a transmission error, every subsequent picture would be affected. Channel switching in a television set would also be impossible. In practical systems a modification is required. One approach is the so-called 'leaky predictor' in which the next picture is predicted from a limited number of previous pictures rather than from an indefinite number. As a result errors cannot propagate indefinitely. The approach used in MPEG is that periodically some absolute picture data are transmitted in place of difference data.

Figure 1.8(b) shows that absolute picture data, known as *I* or *intra pictures* are interleaved with pictures which are created using difference

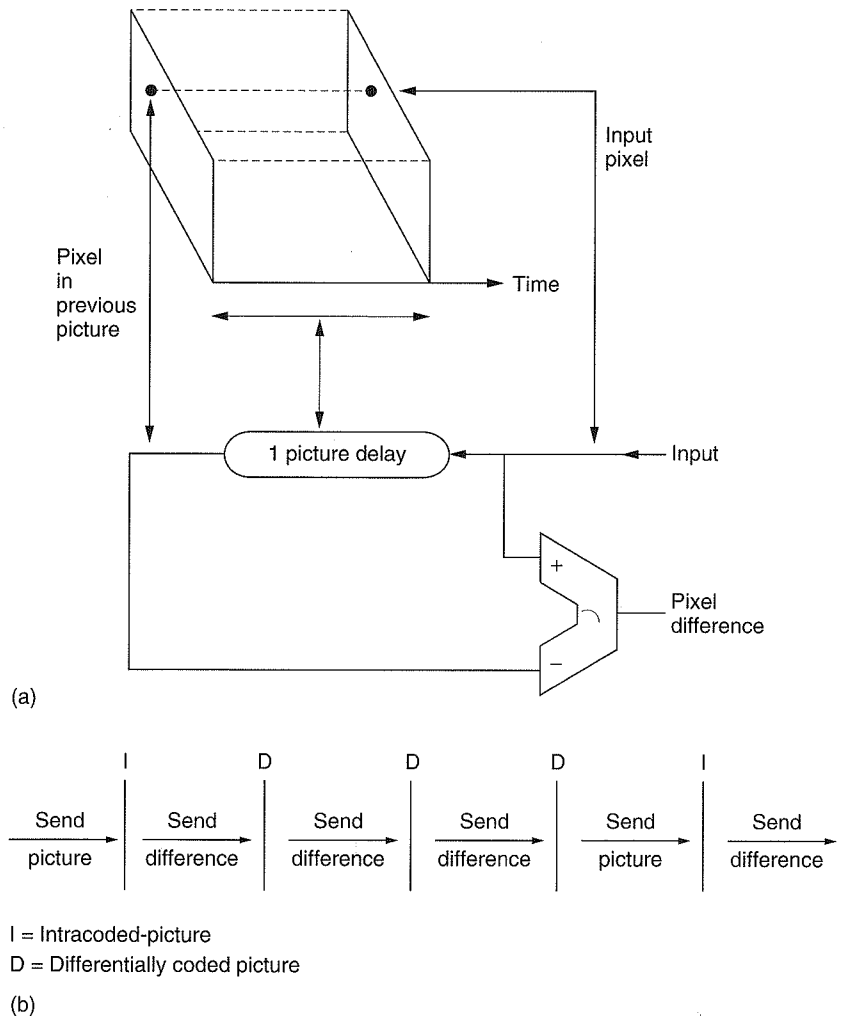


Figure 1.8 An inter-coded system (a) uses a delay to calculate the pixel differences between successive pictures. To prevent error propagation, intra-coded pictures (b) may be used periodically.

data, known as *P* or *predicted* pictures. The *I* pictures require a large amount of data, whereas the *P* pictures require less data. As a result the instantaneous data rate varies dramatically and buffering has to be used to allow a constant transmission rate. The leaky predictor needs less buffering as the compression factor does not change so much from picture to picture.

The *I* picture and all the *P* pictures prior to the next *I* picture are called a group of pictures (GOP). For a high compression factor, a large number of *P* pictures should be present between *I* pictures, making a long GOP.

However, a long GOP delays recovery from a transmission error. The compressed bitstream can only be edited at *I* pictures as shown.

In the case of moving objects, although their appearance may not change greatly from picture to picture, the data representing them on a fixed sampling grid will change and so large differences will be generated between successive pictures. It is a great advantage if the effect of motion can be removed from difference data so that they only reflect the changes in appearance of a moving object since a much greater coding gain can then be obtained. This is the objective of motion compensation.

1.7.3 Introduction to motion compensation

In real television program material objects move around before a fixed camera or the camera itself moves. Motion compensation is a process which effectively measures motion of objects from one picture to the next so that it can allow for that motion when looking for redundancy between pictures. Figure 1.9 shows that moving pictures can be expressed in a three-dimensional space which results from the screen area moving along the time axis. In the case of still objects, the only motion is along the time axis. However, when an object moves, it does so along the *optic flow axis* which is not parallel to the time axis. The optic flow axis joins the same point on a moving object as it takes on various screen positions.

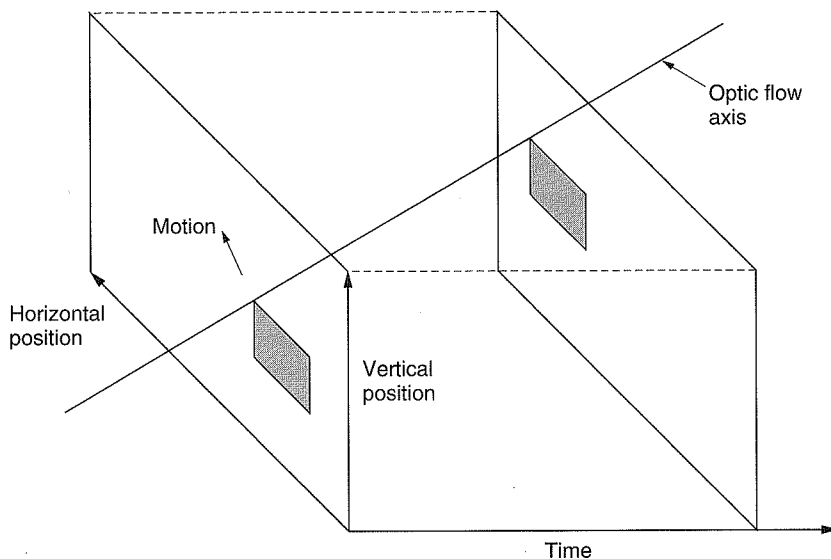


Figure 1.9 Objects travel in a three-dimensional space along the optic flow axis which is only parallel to the time axis if there is no movement.

It will be clear that the data values representing a moving object change with respect to the time axis. However, looking along the optic flow axis the appearance of an object only changes if it deforms, moves into shadow or rotates. For simple translational motions the data representing an object are highly redundant with respect to the optic flow axis. Thus if the optic flow axis can be located, coding gain can be obtained in the presence of motion.

A motion-compensated coder works as follows. An *I* picture is sent, but is also locally stored so that it can be compared with the next input picture to find motion vectors for various areas of the picture. The *I* picture is then shifted according to these vectors to cancel inter-picture motion. The resultant *predicted* picture is compared with the actual picture to produce a *prediction error* also called a *residual*. The prediction error is transmitted with the motion vectors. At the receiver the original *I* picture is also held in a memory. It is shifted according to the transmitted motion vectors to create the predicted picture and then the prediction error is added to it to recreate the original. When a picture is encoded in this way MPEG calls it a *P* picture.

1.7.4 Film-originated video compression

Film can be used as the source of video signals if a telecine machine is used. The most common frame rate for film is 24 Hz, whereas the field rates of television are 50 Hz and 60 Hz. This incompatibility is patched over in two different ways. In 50 Hz telecine, the film is simply played slightly too fast so that the frame rate becomes 25 Hz. Then each frame is converted into two television fields giving the correct 50 Hz field rate. In 60 Hz telecine, the film travels at the correct speed, but alternate frames are used to produce two fields then three fields. The technique is known as 3:2 pulldown. In this way two frames produce five fields and so the correct 60 Hz field rate results. The motion portrayal of telecine is not very good as moving objects judder, especially in 60 Hz systems. Figure 1.10 shows how the optic flow is portrayed in film-originated video.

When film-originated video is input to a compression system, the disturbed optic flow will play havoc with the motion compensation system. In a 50 Hz system there appears to be no motion between the two fields which have originated from the same film frame, whereas between the next two fields large motions will exist. In 60 Hz systems, the motion will be zero for three fields out of five.

With such inputs, it is more efficient to adopt a different processing mode which is based upon the characteristics of the original film. Instead of attempting to manipulate fields of video, the system de-interlaces pairs of fields in order to reconstruct the original film frames. This can be done by a fairly simple motion detector. When substantial motion is measured

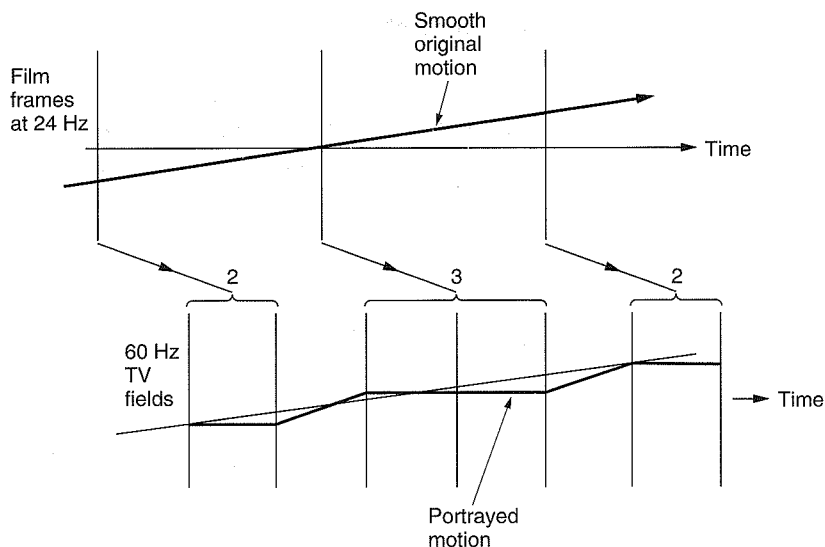


Figure 1.10 Telecine machines must use 3:2 pulldown to produce 60Hz field rate video.

between successive fields in the output of a telecine, this is taken to mean that the fields have come from different film frames. When negligible motion is detected between fields, this is taken to indicate that the fields have come from the same film frame.

In 50 Hz video it is quite simple to find the sequence and produce de-interlaced frames at 25 Hz. In 60 Hz 3:2 pulldown video the problem is slightly more complex because it is necessary to locate the frames in which three fields are output so that the third field can be discarded, leaving, once more, de-interlaced frames at 25 Hz. While it is relatively straightforward to lock-on to the 3:2 sequence with direct telecine output signals, if the telecine material has been edited on videotape the 3:2 sequence may contain discontinuities. In this case it is necessary to provide a number of field stores in the de-interlace unit so that a series of fields can be examined to locate the edits. Once telecine video has been de-interlaced back to frames, intra- and inter-coded compression can be employed using frame-based motion compensation.

MPEG transmissions include flags which tell the decoder the origin of the material. Material originating at 24 Hz but converted to interlaced video does not have the motion attributes of interlace because the lines in two fields have come from the same point on the time axis. Two fields can be combined to create a progressively scanned frame. In the case of 3:2 pulldown material, the third field need not be sent at all as the decoder can easily repeat a field from memory. As a result the same compressed film material can be output at 50 or 60 Hz as required.

1.8 MPEG-2 profiles and levels

MPEG-2 has too many applications to solve with a single standard and so it is subdivided into Profiles and Levels. Put simply, a Profile describes a degree of complexity whereas a Level describes the picture size or resolution which goes with that Profile. Not all Levels are supported at all Profiles. Figure 1.11 shows the available combinations. In principle there are twenty-four of these, but not all have been defined. An MPEG decoder having a given Profile and Level must also be able to decode lower profiles and levels.

		Profiles					
		Simple	Main	4:2:2	SNR	Spatial	High
Levels	High		4:2:0 1920 × 1152 90 Mb/S				4:2:0 or 4:2:2 1920 × 1152 100 Mb/S
	High 1440		4:2:0 1440 × 1152 60 Mb/S			4:2:0 1440 × 1152 60 Mb/S	4:2:0 or 4:2:2 1440 × 1152 80 Mb/S
	Main	4:2:0 720 × 576 15 Mb/S NO B	4:2:0 720 × 576 15 Mb/S	4:2:2 720 × 608 50 Mb/S	4:2:0 720 × 576 15 Mb/S		4:2:0 or 4:2:2 720 × 576 20 Mb/S
	Low		4:2:0 352 × 288 4 Mb/S		4:2:0 352 × 288 4 Mb/S		

Figure 1.11 Profiles and levels in MPEG-2. See text for details.

The simple profile does not support bidirectional coding and so only *I* and *P* pictures will be output. This reduces the coding and decoding delay and allows simpler hardware. The simple profile has only been defined at Main level (SP@ML).

The Main Profile is designed for a large proportion of uses. The low level uses a low-resolution input having only 352 pixels per line. The majority of broadcast applications will require the MP@ML (Main Profile at Main Level) subset of MPEG which supports SDTV (standard definition television).

The High-1440 level is a high-definition scheme which doubles the definition compared to main level. The high level not only doubles the resolution but maintains that resolution with 16:9 format by increasing the number of horizontal samples from 1440 to 1920.

In compression systems using spatial transforms and requantizing it is possible to produce scaleable signals. A scaleable process is one in which

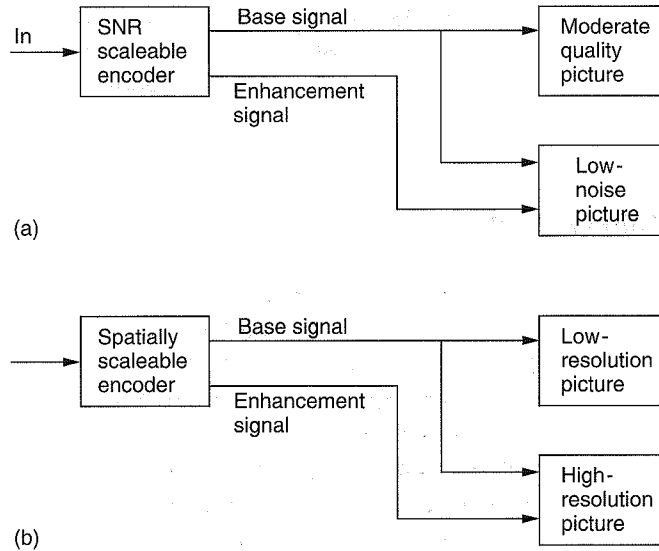


Figure 1.12 (a) An SNR scalable encoder produces a 'noisy' signal and a noise cancelling signal. (b) A spatially scalable encoder produces a low-resolution picture and a resolution enhancing picture.

the input results in a main signal and a 'helper' signal. The main signal can be decoded alone to give a picture of a certain quality, but if the information from the helper signal is added some aspect of the quality can be improved.

Figure 1.12(a) shows that in a conventional MPEG coder, by heavily requantizing coefficients a picture with moderate signal-to-noise ratio results. If, however, that picture is locally decoded and subtracted pixel by pixel from the original, a 'quantizing noise' picture would result. This can be compressed and transmitted as the helper signal. A simple decoder only decodes the main 'noisy' bitstream, but a more complex decoder can decode both bitstreams and combine them to produce a low-noise picture. This is the principle of SNR scaleability.

As an alternative, Figure 1.12(b) shows that by coding only the lower spatial frequencies in a HDTV picture a main bitstream can be made which an SDTV receiver can decode. If the lower-definition picture is locally decoded and subtracted from the original picture, a 'definition-enhancing' picture would result. This can be coded into a helper signal. A suitable decoder could combine the main and helper signals to recreate the HDTV picture. This is the principle of spatial scaleability.

The High profile supports both SNR and spatial scaleability as well as allowing the option of 4:2:2 sampling (see section 2.11).

The 4:2:2 profile has been developed for improved compatibility with existing digital television production equipment. This allows 4:2:2

working without requiring the additional complexity of using the high profile. For example, a HP@ML decoder must support SNR scalability which is not a requirement for production.

1.9 MPEG-2 bitstreams

MPEG-2 supports a variety of bitstream types for various purposes and these are shown in Figure 1.13. The output of a single compressor (video or audio) is known as an Elementary Stream. In transmission, many Elementary Streams will be combined to make a transport stream. A transport stream has a complex structure because it needs to incorporate metadata indicating which audio Elementary Streams and ancillary data are associated with which video Elementary Stream. It is possible to have a single program transport stream (SPTS) which carries only the Elementary Streams of one TV program.

For certain purposes, such as video disc recording, the complexity of a transport stream is not warranted. In this case a program stream can be used. A program stream is a simplified bitstream which multiplexes audio and video for a single program together, provided they have been

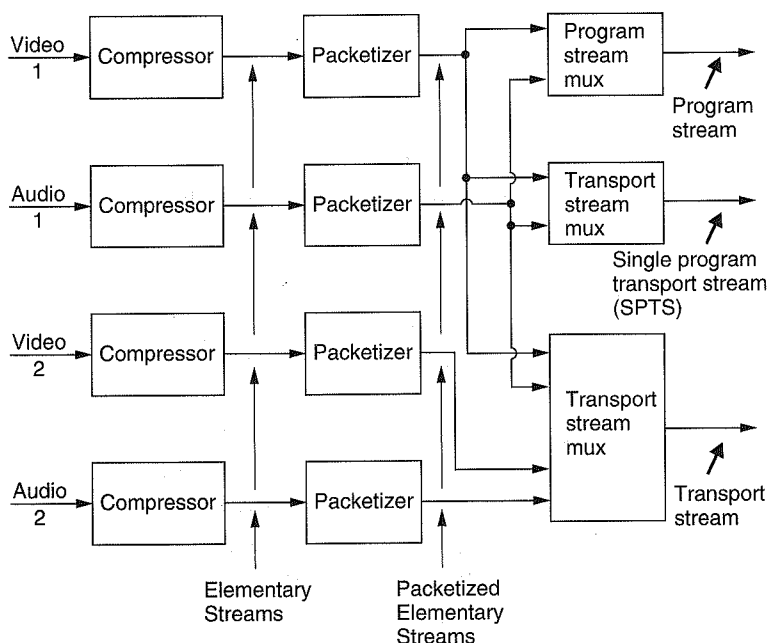


Figure 1.13 The bitstream types of MPEG-2. See text for details.

encoded from a common locked clock. When a recording is played back, the clock can be created by a crystal oscillator and the replay locked to it.

1.10 Drawbacks of compression

By definition, compression removes redundancy from signals. Redundancy is, however, essential to making data resistant to errors. As a result, it is generally true that compressed data are more sensitive to errors than uncompressed data. Thus transmission systems using compressed data must incorporate more powerful error-correction strategies and avoid compression techniques which are notoriously sensitive. As an example, the Digital Betacam format uses relatively mild compression and yet requires 20% redundancy whereas the D-5 format does not use compression and only requires 17% redundancy even though it has a recording density 30% higher.

Techniques using tables such as the Lempel–Ziv–Welch codes are very sensitive to bit errors as an error in the transmission of a table value results in bit errors every time that table location is accessed. This is known as error propagation. Variable-length techniques such as the Huffman code are also sensitive to bit errors. As there is no fixed symbol size, the only way the decoder can parse a serial bitstream into symbols is to increase the assumed wordlength a bit at a time until a code value is recognized. The next bit must then be the first bit in the next symbol. A single bit in error could cause the length of a code to be wrongly assessed and then all subsequent codes would also be wrongly decoded until synchronization could be re-established. Later variable-length codes sacrifice some compression efficiency in order to offer better resynchronization properties.

In non-real-time systems such as computers an uncorrectable error results in reference to the back-up media. In real-time systems such as audio and video this is impossible and concealment must be used. However, concealment relies on redundancy and compression reduces the degree of redundancy. Media such as hard disks can be verified so that uncorrectable errors are virtually eliminated, but tape is prone to dropouts which will exceed the burst correcting power of the replay system from time to time. For this reason the compression factors used on audio or video tape should be moderate.

As perceptive coders introduce noise, it will be clear that in a concatenated system the second codec could be confused by the noise due to the first. If the codecs are identical then each may well make, or better still be designed to make, the same decisions when they are in tandem. If the codecs are not identical the results could be disappointing.

Signal manipulation between codecs can also result in artefacts which were previously undetectable being revealed because the signal which was masking them is no longer present.

In general, compression should not be used for its own sake, but only where a genuine bandwidth or cost bottleneck exists. Even then the mildest compression possible should be used. While high compression factors are permissible for final delivery of material to the consumer, they are not advisable prior to any post-production stages. For contribution material, lower compression factors are essential and this is sometimes referred to as *mezzanine* level compression.

One practical drawback of compression systems is that they are largely generic in structure and the same hardware can be operated at a variety of compression factors. Clearly the higher the compression factor, the cheaper the system will be to operate so there will be economic pressure to use high compression factors. Naturally the risk of artefacts is increased and so there is counterpressure from those with engineering skills to moderate the compression. The way of the world at the time of writing is that the accountants have the upper hand. This was not a problem when there were fixed standards such as PAL and NTSC, as there was no alternative but to adhere to them. Today there is a danger that the variable compression factor control will be turned too far in the direction of economy.

It has been seen above that concatenation of compression systems should be avoided as this causes generation loss. Generation loss is worse if the codecs are different. Interlace is a legacy compression technique and if concatenated with MPEG, generation loss will be exaggerated. In theory and in practice better results are obtained in MPEG for the same bit rate if the input is progressive. Consequently the use of interlace with MPEG coders cannot be recommended for new systems. Chapter 5 explores this theme in greater detail.

1.11 Compression pre-processing

Compression relies completely on identifying redundancy in the source material. Consequently anything which reduces that redundancy will have a damaging effect. Noise is particularly undesirable as it creates additional spatial frequencies in individual pictures as well as spurious differences between pictures. Where noisy source material is anticipated some form of noise reduction will be essential.

When high compression factors must be used to achieve a low bit rate, it is inevitable that the level of artifacts will rise. In order to contain the artifact level, it is necessary to restrict the source entropy prior to the coder. This may be done by spatial low-pass filtering to reduce the picture resolution, and may be combined with downsampling to reduce the

number of pixels per picture. In some cases, such as videophones and teleconferencing it will also be necessary to reduce the picture rate.

A compression pre-processor will combine various types of noise reduction (see Chapter 3) with spatial and temporal downsampling.

1.12 Some guidelines

Although compression techniques themselves are complex, there are some simple rules which can be used to avoid disappointment. Used wisely, MPEG-2 compression has a number of advantages. Used in an inappropriate manner, disappointment is almost inevitable and the technology could get a bad name. The next few points are worth remembering.

- Compression technology may be exciting, but if it is not necessary it should not be used.
- If compression is to be used, the degree of compression should be as small as possible; i.e. use the highest practical bit rate.
- Cascaded compression systems cause loss of quality and the lower the bit rates, the worse this gets. Quality loss increases if any post-production steps are performed between compressions.
- Avoid using interlaced video with MPEG.
- Compression systems cause delay.
- Compression systems work best with clean source material. Noisy signals or poorly decoded composite video give poor results.
- Compressed data are generally more prone to transmission errors than non-compressed data. The choice of a compression scheme must consider the error characteristics of the channel.
- Audio codecs need to be level calibrated so that when sound pressure level-dependent decisions are made in the coder those levels actually exist at the microphone.
- Low bit rate coders should only be used for the final delivery of post-produced signals to the end user.
- Compression quality can only be assessed subjectively.
- Don't believe statements comparing codec performance to 'VHS quality' or similar. Compression artefacts are quite different to the artefacts of consumer VCRs.
- Quality varies wildly with source material. Beware of 'convincing' demonstrations which may use selected material to achieve low bit rates. Use your own test material, selected for a balance of difficulty.
- Don't be browbeaten by the technology. You don't have to understand it to assess the results. Your eyes and ears are as good as

anyone's so don't be afraid to criticize artifacts. In the case of video, use still frames to distinguish spatial artifacts from temporal artifacts.

References

1. ISO/IEC JTC1/SC29/WG11 MPEG, International standard ISO 11172, Coding of moving pictures and associated audio for digital storage media up to 1.5 Mbits/s (1992)
2. Le Gall, D., MPEG: a video compression standard for multimedia applications. *Communications of the ACM*, **34**, No.4, 46–58 (1991)
3. MPEG Video Standard: ISO/IEC 13818–2: Information technology – generic coding of moving pictures and associated audio information: Video (1996) (aka ITU-T Rec. H-262 (1996))
4. Huffman, D.A., A method for the construction of minimum redundancy codes. *Proc. IRE.* **40** 1098–1101 (1952)
5. ISO Joint Photographic Experts Group standard JPEG-8-R8
6. Wallace, G.K., Overview of the JPEG (ISO/CCITT) still image compression standard. ISO/JTC1/SC2/WG8 N932 (1989)

4.9 Quality measurement

As was seen in Chapter 3, one way in which coding gain is obtained is to requantize sample values to reduce the wordlength. Since the resultant requantizing error results in energy moving from one frequency to another, the masking model is essential to estimate how audible the effect will be. The greater the degree of reduction required, the more precise the model must be. If the masking model is inaccurate, then equipment based upon it may produce audible artifacts under some circumstances. Artifacts may also result if the model is not properly implemented. As a result, development of data reduction units requires careful listening tests with a wide range of source material.^{9,10} The presence of artifacts at a given compression factor indicates only that performance is below expectations; it does not distinguish between the implementation and the model. If the implementation is verified, then a more detailed model must be sought. Naturally comparative listening tests are only valid if all the codecs have been level calibrated.

Properly conducted listening tests are expensive and time consuming, and alternative methods have been developed which can be used objectively to evaluate the performance of different techniques. The noise-to-masking ratio (NMR) is one such measurement.¹¹ Figure 4.14

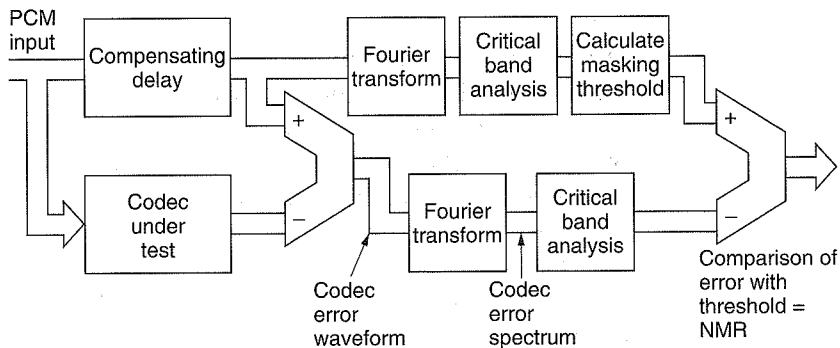


Figure 4.14 The noise-to-masking ratio is derived as shown here.

shows how NMR is measured. Input audio signals are fed simultaneously to a data reduction coder and decoder in tandem and to a compensating delay whose length must be adjusted to match the codec delay. At the output of the delay, the coding error is obtained by subtracting the codec output from the original. The original signal is spectrum analysed into critical bands in order to derive the masking threshold of the input audio, and this is compared with the critical band spectrum of the error. The NMR in each critical band is the ratio between

the masking threshold and the quantizing error due to the codec. An average NMR for all bands can be computed. A positive NMR in any band indicates that artifacts are potentially audible. Plotting the average NMR against time is a powerful technique, as with an ideal codec the NMR should be stable with different types of program material. If this is not the case the codec could perform quite differently as a function of the source material. NMR excursions can be correlated with the waveform of the audio input to analyse how the extra noise was caused and to redesign the codec to eliminate it.

Practical systems should have a finite NMR in order to give a degree of protection against difficult signals which have not been anticipated and against the use of post-codec equalization or several tandem codecs which could change the masking threshold. There is a strong argument that professional devices should have a greater NMR than consumer or program delivery devices.

4.10 The limits

There are, of course, limits to all technologies. Eventually artifacts will be heard as the amount of compression is increased which no amount of detailed modelling will remove. The ear is only able to perceive a certain proportion of the information in a given sound. This could be called the perceptual entropy,¹² and all additional sound is redundant or irrelevant. Data reduction works by removing the redundancy, and clearly an ideal system would remove all of it, leaving only the entropy. Once this has been done, the masking capacity of the ear has been reached and the NMR has reached zero over the whole band. Reducing the data rate further must reduce the entropy, because raising noise further at any frequency will render it audible. In practice the audio bandwidth will have to be reduced in order to keep the noise level acceptable. In MPEG-1 pre-filtering allows data from higher sub-bands to be neglected. MPEG-2 has introduced some low sampling rate options for this purpose. Thus there is a limit to the degree of data reduction which can be achieved even with an ideal coder. Systems which go beyond that limit are not appropriate for high-quality music, but are relevant in news gathering and communications where intelligibility of speech is the criterion.

Interestingly, the data rate out of a coder is virtually independent of the input sampling rate unless the sampling rate is very low. This is because the entropy of the sound is in the waveform, not in the number of samples carrying it.

The compression factor of a coder is only part of the story. All codecs cause delay, and in general the greater the compression, the longer the delay. In some applications, such as telephony, a short delay is required.¹³

In many applications, the compressed channel will have a constant bit rate, and so a constant compression factor is required. In real program material, the entropy varies and so the NMR will fluctuate. If greater delay can be accepted, as in a recording application, memory buffering can be used to allow the coder to operate at constant NMR and instantaneously variable data rate. The memory absorbs the instantaneous data rate differences of the coder and allows a constant rate in the channel. A higher effective compression factor will then be obtained. Near-constant quality can also be achieved using statistical multiplexing.

4.11 Compression applications

One of the fundamental concepts of PCM audio is that the signal-to-noise ratio of the channel can be determined by selecting a suitable wordlength. In conjunction with the sampling rate, the resulting data rate is then determined. In many cases the full data rate can be transmitted or recorded. A high-quality digital audio channel requires around one megabit per second, whereas a standard definition component digital video channel needs two hundred times as much. With mild video compression the video bit rate is still so much in excess of the audio rate that audio compression is not worth while. Only when high video compression factors are used does it become necessary to apply compression to the audio.

Professional equipment traditionally has operated with higher sound quality than consumer equipment in order to allow for some inevitable quality loss in production, and so there will be less pressure to use data reduction. In fact there is even pressure against the use of audio compression in critical applications because of the loss of quality, particularly in stereo and surround sound.

With the rising density and falling cost of digital recording media, the use of compression in professional audio storage applications is hard to justify. However, where there is a practical or economic restriction on channel bandwidth compression becomes essential. Now that the technology exists to broadcast television pictures using compressed digital coding, it is obvious that the associated audio should be conveyed in the same way.

4.12 History of MPEG audio coding

The ISO (International Standards Organization) and the IEC (International Electrotechnical Commission) recognized that compression would have an important part to play in future digital video products

and in 1988 established the ISO/IEC/MPEG (Moving Picture Experts Group) to compare and assess various coding schemes in order to arrive at an international standard. The terms of reference were extended the same year to include audio and the MPEG/Audio group was formed.

As part of the Eureka 147 project, a system known as MUSICAM¹⁴ (Masking pattern adapted Universal Sub-band Integrated Coding and Multiplexing) was developed jointly by CCETT in France, IRT in Germany and Philips in the Netherlands. MUSICAM was designed to be suitable for DAB (digital audio broadcasting).

As a parallel development, the ASPEC¹⁵ (Adaptive Spectral Perceptual Entropy Coding) system was developed from a number of earlier systems as a joint proposal by AT&T Bell Labs, Thomson, the Fraunhofer Society and CNET. ASPEC was designed for high degrees of compression to allow audio transmission on ISDN.

These two systems were both fully implemented by July 1990 when comprehensive subjective testing took place at the Swedish Broadcasting Corporation.^{9,16,17} As a result of these tests, the MPEG/Audio group combined the attributes of both ASPEC and MUSICAM into a draft standard¹⁸ having three levels of complexity and performance.

These three different levels are needed because of the number of possible applications. Audio coders can be operated at various compression factors with different quality expectations. Stereophonic classical music requires different quality criteria to monophonic speech. As was seen in Chapter 1, the complexity of the coder will be reduced with a smaller compression factor. For moderate compression, a simple codec will be more cost effective. On the other hand, as the compression factor is increased, it will be necessary to employ a more complex coder to maintain quality.

At each level, MPEG coding allows input sampling rates of 32, 44.1 and 48 KHz and supports output bit rates of 32, 48, 56, 64, 96, 112, 128, 192, 256 and 384 Kbits/s. The transmission can be mono, dual channel (e.g. bilingual), stereo and joint stereo which is where advantage is taken of redundancy between the two audio channels.

MPEG Layer I is a simplified version of MUSICAM which is appropriate for the mild compression applications at low cost. It is very similar to PASC. Layer II is identical to MUSICAM and is very likely to be used for DAB. Layer III is a combination of the best features of ASPEC and MUSICAM and is mainly applicable to telecommunications where high compression factors are required.

The earlier MPEG-1 standard compresses audio and video into about 1.5 Mbits/s. The audio content of MPEG-1 may be used on its own to encode one or two channels at bit rates up to 448 Kbits/s. MPEG-2 allows the number of channels to increase to five: Left, Right, Centre, Left surround, Right surround and Subwoofer. In order to retain reverse

compatibility with MPEG-1, the MPEG-2 coding converts the five-channel input to a compatible two-channel signal, L_o, R_o , by matrixing.¹⁹ The data from these two channels is encoded in a standard MPEG-1 audio frame, and this is followed in MPEG-2 by an ancillary data frame which an MPEG-1 decoder will ignore. The ancillary frame contains data for any three of the audio channels. An MPEG-2 decoder will extract those three channels in addition to the MPEG-1 frame and then recover all five original channels by an inverse matrix.

In the USA, it has been proposed to use an alternative compression technique for the audio content of ATSC (Advanced Television Systems Committee) digital television broadcasts. This is the AC-3 system developed by Dolby Laboratories. The MPEG transport stream structure has also been standardized to allow it to carry AC-3 coded audio. The Digital Video Disc can also carry AC-3 or MPEG audio coding.

4.13 MPEG audio compression tools

There are many different approaches to audio compression, each having advantages and disadvantages. MPEG audio coding combines these tools in various ways in the three different coding levels. The approach of this chapter will be to examine the tools separately before seeing how they are used in MPEG and AC-3.

The simplest coding tool is companding which is a digital parallel of the analog noise reducers used in tape recording. Figure 4.15(a) shows that in companding the input signal level is monitored. Whenever the input level falls below maximum, it is amplified at the coder. The gain which was applied at the coder is added to the data stream so that the decoder can apply an equal attenuation. The advantage of companding is that the signal is kept as far away from the noise floor as possible. In analog noise reduction this is used to maximize the SNR of a tape recorder, whereas in digital compression it is used to keep the signal level as far as possible above the noises and artifacts introduced by various coding steps.

One common way of obtaining coding gain is to shorten the wordlength of samples so that fewer bits need to be transmitted. Figure 4.15(b) shows that when this is done, the noise floor will rise by 6 dB for every bit removed. This is because removing a bit halves the number of quantizing intervals which then must be twice as large, doubling the noise level. Clearly if this step follows the compander of (a), the audibility of the noise will be minimized. As an alternative to shortening the wordlength, the uniform quantized PCM signal can be converted to a non-uniform format. In non-uniform coding, shown in (c), the size of the quantizing step rises with the magnitude of the sample so that the noise level is greater when higher levels exist.

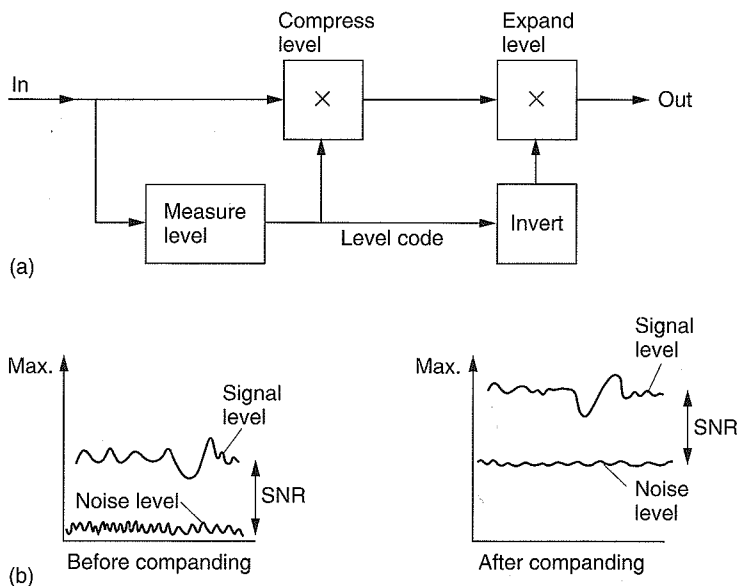


Figure 4.15 Digital companding. In (a) the encoder amplifies the input to maximum level and the decoder attenuates by the same amount. (b) In a companded system, the signal is kept as far as possible above the noise caused by shortening the sample wordlength.

Companding is a relative of floating-point coding shown in Figure 4.16 where the sample value is expressed as a mantissa and a binary exponent which determines how the mantissa needs to be shifted to have its correct absolute value on a PCM scale. The exponent is the equivalent of the gain setting or scale factor of a compandor.

Clearly in floating point the signal-to-noise ratio is defined by the number of bits in the mantissa, and, as shown in Figure 4.17, this will vary as a sawtooth function of signal level, as the best value, obtained when the mantissa is near overflow, is replaced by the worst value when the mantissa overflows and the exponent is incremented. Floating-point

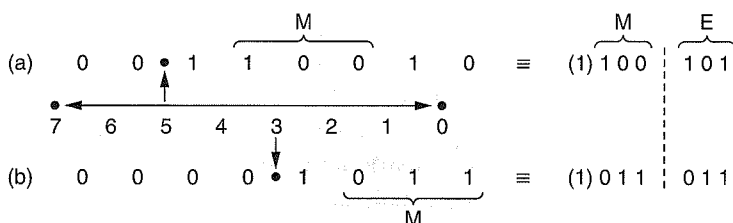


Figure 4.16 In this example of floating-point notation, the radix point can have eight positions determined by the exponent E. The point is placed to the left of the first '1', and the next 4 bits to the right form the mantissa M. As the MSB of the mantissa is always 1, it need not always be stored.

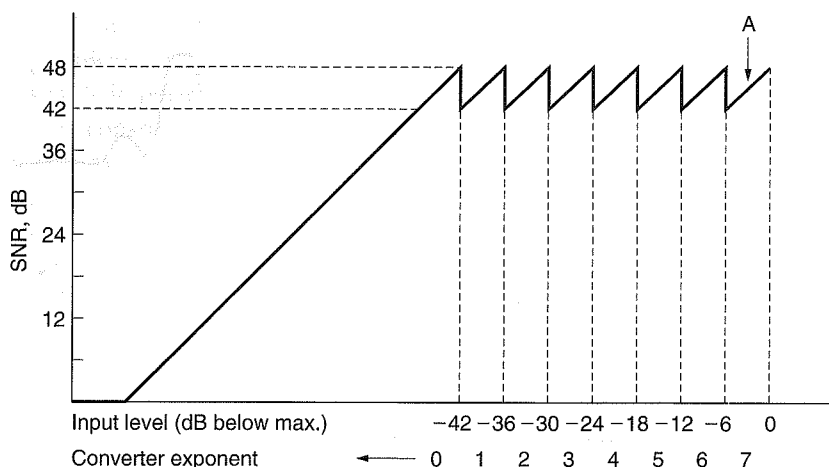


Figure 4.17 In this example of an 8-bit mantissa, 3-bit exponent system, the maximum SNR is $6 \text{ dB} \times 8 = 48 \text{ dB}$ with maximum input of 0 dB. As input level falls by 6 dB, the converter noise remains the same, so SNR falls to 42 dB. Further reduction in signal level causes the converter to shift range (point A in the diagram) by increasing the input analog gain by 6 dB. The SNR is restored, and the exponent changes from 7 to 6 in order to cause the same gain change at the receiver. The noise modulation would be audible in this simple system. A longer mantissa word is needed in practice.

notation is used within DSP chips as it eases the computational problems involved in handling long wordlengths. For example, when multiplying floating-point numbers, only the mantissae need to be multiplied. The exponents are simply added.

A floating-point system requires one exponent to be carried with each mantissa and this is wasteful because in real audio material the level does not change so rapidly and there is redundancy in the exponents. A better alternative is floating-point block coding, also known as near-instantaneous companding, where the magnitude of the largest sample in a block is used to determine the value of an exponent which is valid for the whole block. Sending one exponent per block requires a lower data rate than in true floating point.²⁰

In block coding the requantizing in the coder raises the quantizing noise, but it does so over the entire duration of the block. Figure 4.18 shows that if a transient occurs towards the end of a block, the decoder will reproduce the waveform correctly, but the quantizing noise will start at the beginning of the block and may result in a pre-noise (also called pre-echo) where the noise is audible before the transient. Temporal masking may be used to make this inaudible. With a 1 millisecond block, the artifacts are too brief to be heard.

Another solution is to use a variable time window according to the transient content of the audio waveform. When musical transients occur,

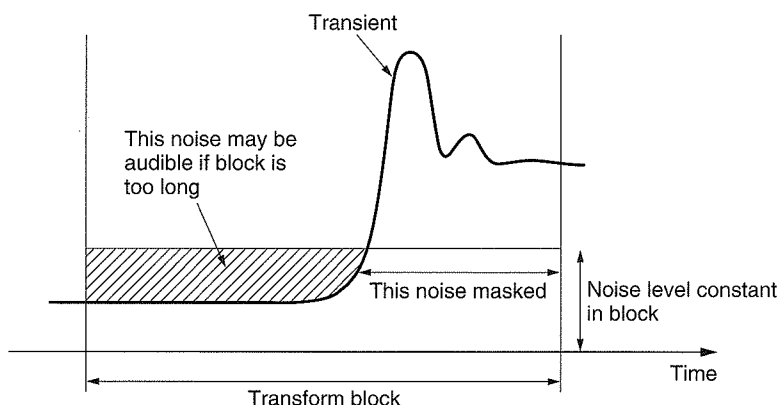


Figure 4.18 If a transient occurs towards the end of a transform block, the quantizing noise will still be present at the beginning of the block and may result in a pre-echo where the noise is audible before the transient.

short blocks are necessary and the coding gain will be low. At other times the blocks become longer allowing a greater coding gain.

While the above systems used alone do allow coding gain, the compression factor has to be limited because little benefit is obtained from masking. This is because the techniques above produce noise which spreads equally over the entire audio band. If the audio input spectrum is narrow, the noise will not be masked.

Sub-band coding splits the audio spectrum up into many different frequency bands. Once this has been done, each band can be individually processed. In real audio signals most bands will contain lower-level signals than the loudest one. Individual companding of each band will be more effective than broadband companding. Sub-band coding also allows the noise floor to be raised selectively so that noise is only added at frequencies where spectral masking will be effective.

There is little conceptual difference between a sub-band coder with a large number of bands and a transform coder. In transform coding, a Fourier or DCT transform of the waveform is computed periodically. Since the transform of an audio signal changes slowly, it needs be sent much less often than audio samples. The receiver performs an inverse transform. Transform coding is considered in section 4.14. Finally the data may be subject to a lossless binary compression using, for example, a Huffman code.

4.14 Transform coding

Audio is usually considered to be a time-domain waveform as this is what emerges from a microphone. As has been seen in Chapter 3, spectral

analysis allows any periodic waveform to be represented by a set of harmonically related components of suitable amplitude and phase. In theory it is perfectly possible to decompose a periodic input waveform into its constituent frequencies and phases, and to record or transmit the transform. The transform can then be inverted and the original waveform will be precisely re-created.

Although one can think of exceptions, the transform of a typical audio waveform changes relatively slowly. The slow speech of an organ pipe or a violin string, or the slow decay of most musical sounds allow the rate at which the transform is sampled to be reduced, and a coding gain results. At some frequencies the level will be below maximum and a shorter wordlength can be used to describe the coefficient. Further coding gain will be achieved if the coefficients describing frequencies which will experience masking are quantized more coarsely. The transform of an audio signal is computed in the main signal path in a transform coder, and has sufficient frequency resolution to drive the masking model directly.

In practice there are some difficulties: real sounds are not periodic, but contain transients which transformation cannot accurately locate in time. The solution to this difficulty is to cut the waveform into short segments and then to transform each individually. The delay is reduced, as is the computational task, but there is a possibility of artifacts arising because of the truncation of the waveform into rectangular time windows. A solution is to use window functions (see Chapter 2) and to overlap the segments as shown in Figure 4.19. Thus every input sample appears in just two transforms, but with variable weighting depending upon its position along the time axis.

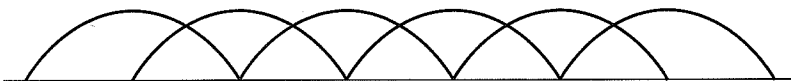


Figure 4.19 Transform coding can only be practically performed on short blocks. These are overlapped using window functions in order to handle continuous waveforms.

The DFT (discrete frequency transform) does not produce a continuous spectrum, but instead produces coefficients at discrete frequencies. The frequency resolution (i.e. the number of different frequency coefficients) is equal to the number of samples in the window. If overlapped windows are used, twice as many coefficients are produced as are theoretically necessary. In addition the DFT requires intensive computation, owing to the requirement to use complex arithmetic to render the phase of the components as well as the amplitude. An alternative is to use discrete cosine transforms (DCT).

4.15 MPEG Layer I audio coding

Figure 4.20 shows a block diagram of a Layer I coder which is a simplified version of that used in the MUSICAM system. A polyphase quadrature mirror filter network divides the audio spectrum into 32 equal sub-bands. The output data rate of the filter bank is no higher than the input rate because each band has been heterodyned to a frequency range from DC upwards.

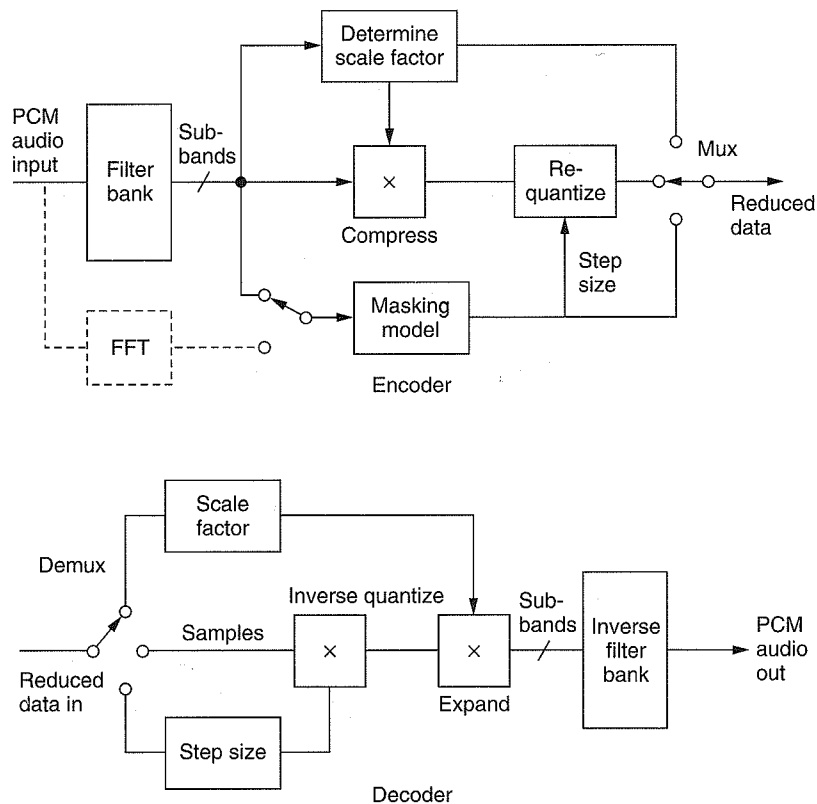


Figure 4.20 A simple sub-band coder. The bit allocation may come from analysis of the sub-band energy, or, for greater reduction, from a spectral analysis in a side chain.

Sub-band compression takes advantage of the fact that real sounds do not have uniform spectral energy. The wordlength of PCM audio is based on the dynamic range required and this is generally constant with frequency although any pre-emphasis will affect the situation. When a signal with an uneven spectrum is conveyed by PCM, the whole dynamic range is occupied only by the loudest spectral component, and all the

other components are coded with excessive headroom. In its simplest form, sub-band coding²¹ works by splitting the audio signal into a number of frequency bands and companding each band according to its own level. Bands in which there is little energy result in small amplitudes which can be transmitted with short wordlength. Thus each band results in variable-length samples, but the sum of all the sample wordlengths is less than that of PCM and so a coding gain can be obtained.

As MPEG audio coding relies on auditory masking, the sub-bands should preferably be narrower than the critical bands of the ear, hence the large number required. Figure 4.21 shows the critical condition where the masking tone is at the top edge of the sub-band. It will be seen that the narrower the sub-band, the higher the requantizing noise that can be masked. The use of an excessive number of sub-bands will, however, raise complexity and the coding delay.

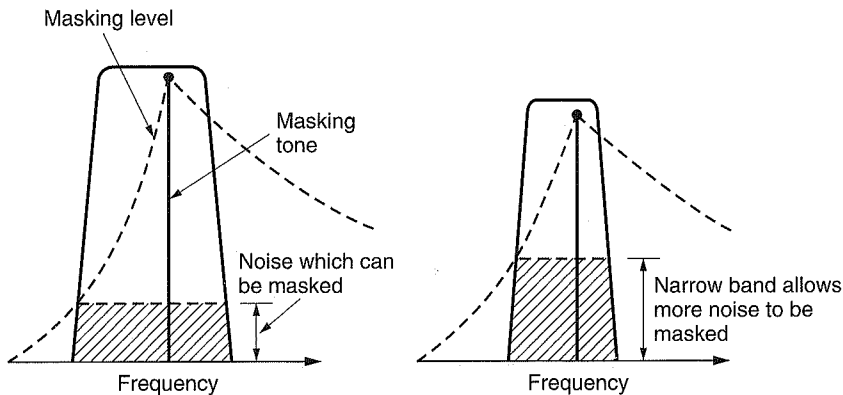


Figure 4.21 In sub-band coding the worst case occurs when the masking tone is at the top edge of the sub-band. The narrower the band, the higher the noise level which can be masked.

Constant-size input blocks are used, containing 384 samples. At 48 KHz, 384 samples corresponds to a period of 8 milliseconds. After the sub-band filter each band contains 12 samples per block. The block size was based on the pre-masking phenomenon of Figure 4.18. The samples in each sub-band block or *bin* are companded according to the peak value in the bin. A 6-bit scale factor is used for each sub-band which applies to all 12 samples.

If a fixed compression factor is employed, the size of the coded output block will be fixed. The wordlengths in each bin will have to be such that the sum of the bits from all the sub-band equals the size of the coded block. Thus some sub-bands can have long wordlength coding if others have short wordlength coding. The process of determining the

requantization step size, and hence the wordlength, in each sub-band is known as bit allocation.

For simplicity, in Layer I the levels in the 32 sub-bands themselves are used as a crude spectral analysis of the input in order to drive the masking model. The masking model uses the input spectrum to determine a new threshold of hearing which in turn determines how much the noise floor can be raised in each sub-band. Where masking takes place, the signal is quantized more coarsely until the quantizing noise is raised to just below the masking level. The coarse quantization requires shorter wordlengths and allows a coding gain. The bit allocation may be iterative as adjustments are made to obtain the best NMR within the allowable data rate.

The samples of differing wordlength in each bin are then assembled into the output coded block. Unlike a PCM block, which contains samples of fixed wordlength, a coded block contains many different wordlengths and these can vary from one block to the next. In order to deserialize the block into samples of various wordlength and demultiplex the samples into the appropriate frequency bins, the decoder has to be told what bit allocations were used when it was packed, and some synchronizing means is needed to allow the beginning of the block to be identified.

The compression factor is determined by the bit-allocation system. It is not difficult to change the output block size parameter to obtain a different compression factor. If a larger block is specified, the bit allocator simply iterates until the new block size is filled. Similarly the decoder need only deserialize the larger block correctly into coded samples and then the expansion process is identical except for the fact that expanded words contain less noise. Thus codecs with varying degrees of compression are available which can perform different bandwidth/performance tasks with the same hardware.

Figure 4.22 shows the format of the Layer I data stream. The frame begins with a sync pattern to reset the phase of deserialization, and a header which describes the sampling rate and any use of pre-emphasis. Following this is a block of 32 4-bit allocation codes. These specify the wordlength used in each sub-band and allow the decoder to deserialize the sub-band sample block. This is followed by a block of 32 6-bit scale factor indices, which specify the gain given to each band during companding. The last block contains 32 sets of 12 samples. These samples vary in wordlength from one block to the next, and can be from 0 to 15 bits long. The deserializer has to use the 32 allocation information codes to work out how to deserialize the sample block into individual samples of variable length.

The Layer I MPEG decoder is shown in Figure 4.23. The elementary stream is deserialized using the sync pattern and the variable-length samples are assembled using the allocation codes. The variable length

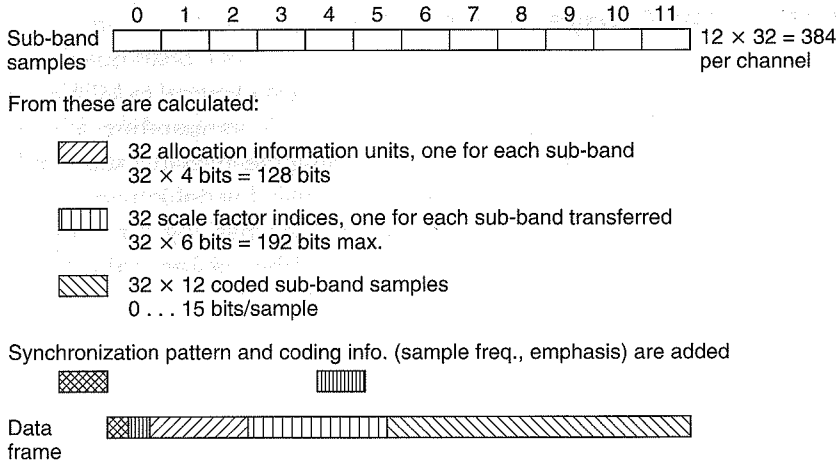


Figure 4.22 The Layer I data frame showing the allocation codes, the scale factors and the sub-band samples.

samples are returned to 15-bit wordlength by adding zeros. The scale factor indices are then used to determine multiplication factors used to return the waveform in each sub-band to its original level. The 32 sub-band signals are then merged into one spectrum by the synthesis filter. This is a set of bandpass filters which returns every sub-band to the correct place in the audio spectrum and then adds them to produce the audio output.

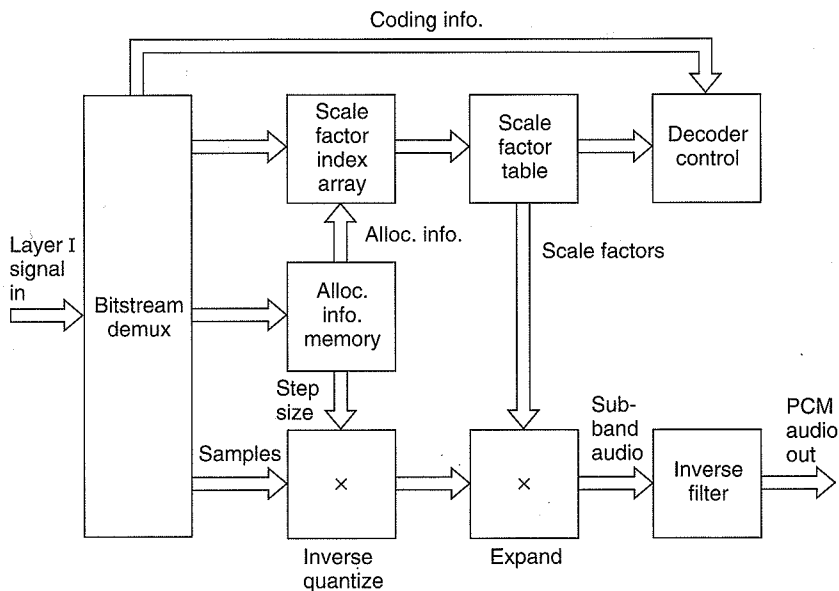


Figure 4.23 The Layer I decoder. See text for details.

4.16 MPEG Layer II audio coding

MPEG Layer II audio coding is identical to MUSICAM. The same 32-band filterbank and the same block companding scheme as Layer I is used. Figure 4.24 shows that using the level in a sub-band to drive the masking model is sub-optimal because it is not known where in the sub-band the energy lies. As the skirts of the masking curve are asymmetrical, the noise floor can be raised higher if the masker is at the low end of the sub-band than if it is at the high end.

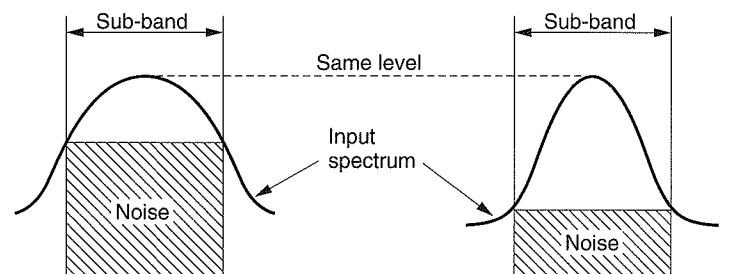


Figure 4.24 Accurate knowledge of the spectrum allows the noise floor to be raised higher while remaining masked.

In order to give better spectral resolution than the filterbank, a side chain FFT is computed having 1024 points, resulting in an analysis of the audio spectrum eight times better than the sub-band width. The FFT drives the masking model which controls the bit allocation. In order to give the FFT sufficient resolution, the block length is increased to 1152 samples. This is three times the block length of Layer I.

The QMF bandsplitting technique is restricted to bands of equal width. It might be thought that this is a drawback because the critical bands of the ear are non-uniform. In fact this is only a problem when very low bit rates are required. In all cases it is the masking model of hearing which must have correct critical bands. This model can then be superimposed on bands of any width to determine how much masking and therefore coding gain is possible. Uniform-width sub-bands will not be able to obtain as much masking as bands which are matched to critical bands, but for many applications the additional coding gain is not worth the added filter complexity.

The block companding scheme of Layer II is the same as in Layer I because the 1152-sample block is divided into three 384-sample blocks. However, not all the scale factors are transmitted, because they contain a degree of redundancy on real program material. The difference between scale factors in successive blocks in the same band exceeds 2 dB less than

10% of the time. Layer II analyses the set of three successive scale factors in each sub-band. On stationary program, these will be the same and only one scale factor out of three is sent. As the transient content increases in a given sub-band, two or three scale factors will be sent. A scale factor select code must be sent to allow the decoder to determine what has been sent in each sub-band. This technique effectively halves the scale factor bit rate.

The requantized samples in each sub-band, bit-allocation data, scale factors and scale factor select codes are multiplexed into the output bit stream.

The Layer II decoder is not much more complex than the Layer I decoder as the only additional processing is to decode the compressed scale factors to produce one scale factor per 384-sample block.

4.17 MPEG Layer III audio coding

This is the most complex layer of the ISO standard, and is only really necessary when the most severe data rate constraints must be met with high quality. It is a transform code based on the ASPEC system with certain modifications to give a degree of commonality with Layer II. The original ASPEC coder used a direct MDCT on the input samples. In Layer III this was modified to use a hybrid transform incorporating the existing polyphase 32-band QMF of Layers I and II. In Layer III, the 32 sub-bands from the QMF are each processed by a 12-band MDCT to obtain 384 output coefficients. Two window sizes are used to avoid pre-echo on transients. The window switching is performed by the psychoacoustic model. It has been found that pre-echo is associated with the entropy in the audio rising above the average value.

A highly accurate perceptive model is used to take advantage of the high-frequency resolution available. Non-uniform quantizing is used, along with Huffman coding. This is a technique where the most common code values are allocated the shortest wordlength.

4.18 Dolby AC-3

Dolby AC-3 is in fact a family of transform coders based on time-domain aliasing cancellation (TDAC) which allow various compromises between coding delay and bit rate to be used. In the modified discrete cosine transform (MDCT),²² windows with 50% overlap are used. Thus twice as many coefficients as necessary are produced. These are sub-sampled by a factor of two to give a critically sampled transform, which results in potential aliasing in the frequency domain. However, by making a slight

6

Program and transport streams

After compression, audio and video signals are simply data and only differ from generic data in that they relate to real-time signals. MPEG bitstreams are basically means of transporting data while allowing the time axis of the original signal to be recreated.

6.1 Introduction

There are two basic applications of MPEG bitstreams: recording and transmission and these have quite different requirements. In the multi-channel recording application the encoders and decoders can all share the same clock. The transport speed can be altered so that the incoming bit rate can be slaved to a reference generated by the decoder itself.

In the case of multichannel transmission the program sources are not necessarily synchronous. The source of timing is the encoder and the decoder must synchronize or genlock to that. Thus the difference between a program stream and a transport stream is that the latter must contain additional synchronizing information to lock the encoder and decoder clocks together independently in each program.

6.2 Packets and time stamps

The video Elementary Stream is an endless bitstream representing pictures which are not necessarily in the correct order and which take a variable length of time to transmit. Storage and transmission systems prefer discrete blocks of data and so Elementary Streams are packetized to form a PES (packetized Elementary Stream). Audio Elementary

Streams are also packetized. A packet is shown in Figure 6.1. It begins with a header containing an unique packet start code and a code which identifies the type of data stream. Optionally the packet header also may contain one or more *time stamps* which are used for synchronizing the video decoder to real time and for obtaining lip-sync.

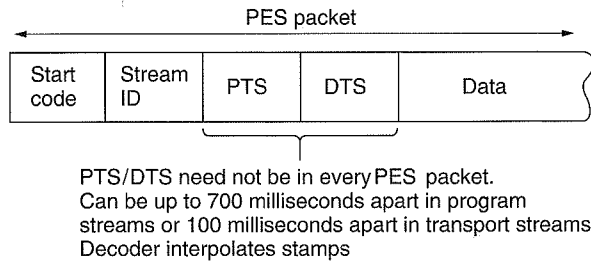


Figure 6.1 A PES packet structure is used to break up the continuous Elementary Stream.

Figure 6.2 shows that a time stamp is a sample of the state of a counter which is driven by a 90 KHz clock. This is obtained by dividing down the master 27 MHz clock of MPEG-2. There are two types of time stamp: PTS and DTS. These are abbreviations for presentation time stamp and decode time stamp. A presentation time stamp determines when the associated picture should be displayed on the screen, whereas a decode time stamp determines when it should be decoded. In bidirectional coding these times can be quite different.

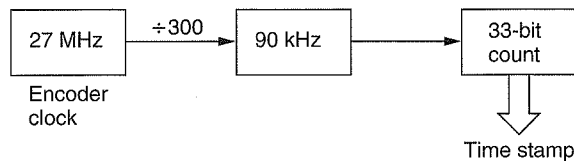


Figure 6.2 Time stamps are the result of sampling a counter driven by the encoder clock.

Audio packets only have presentation time stamps. Clearly if lip-sync is to be obtained, the audio and the video streams of a given program must have been locked to the same master 27 MHz clock and the time stamps must have come from the same counter driven by that clock.

With reference to Figure 6.3, the GOP begins with an *I* picture, and then *P1* is sent out of sequence prior to the *B* pictures. *P1* has to be decoded before *B1* and *B2* can be decoded. As only one picture can be decoded at a time, the *I* picture is decoded at time *N*, but not displayed until time *N*

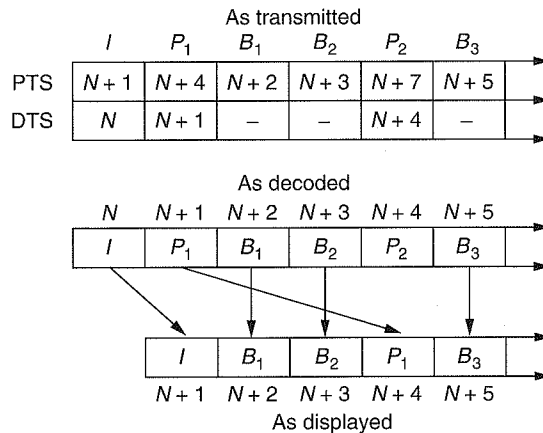


Figure 6.3 An example of using PTS/DTS to synchronize bidirectional decoding.

+ 1. As the I picture is being displayed, P_1 is being decoded at $N+1$. P_1 will be stored in RAM. At time $N+2$, B_1 is decoded and displayed immediately. For this reason B pictures need only PTS. At $N+3$, B_2 is decoded and displayed. At $N+4$, P_1 is displayed, hence the large difference between PTS and DTS in P_1 . Simultaneously P_2 is decoded and stored ready for the decoding of B_3 and so on.

In practice the time between input pictures is constant and so there is a certain amount of redundancy in the time stamps. Consequently PTS/DTS need not appear in every PES packet. Time stamps can be up to 700 milliseconds apart in program streams and up to 100 milliseconds apart in transport streams. As each picture type (I , P or B) is flagged in the bitstream, the decoder can infer the PTS/DTS for every picture from the ones actually transmitted.

Figure 6.4 shows that one or more PES packets can be assembled in a *pack* whose header contains a sync pattern and a *system clock reference* code which allows the decoder to recreate the encoder clock. Clock reference operation is described in section 6.4.

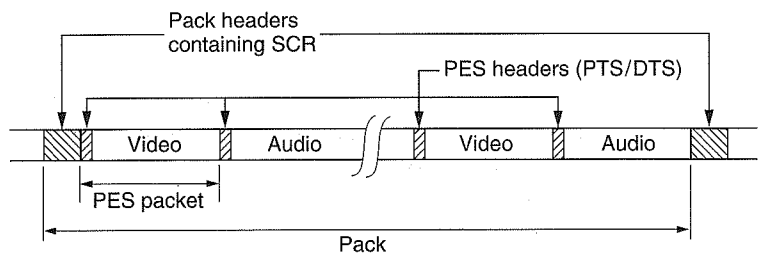


Figure 6.4 A pack is a set of PES packets. The pack header contains a clock reference code.

6.3 Transport streams

The MPEG-2 transport stream is intended to be a multiplex of many TV programs with their associated sound and data channels, although a single program transport stream (SPTS) is possible. The transport stream is based upon packets of constant size so that adding error correction codes and interleaving¹ in a higher layer is eased. Figure 6.5 shows that these are always 188 bytes long. Transport stream packets should not be confused with PES packets which are larger and which vary in size.

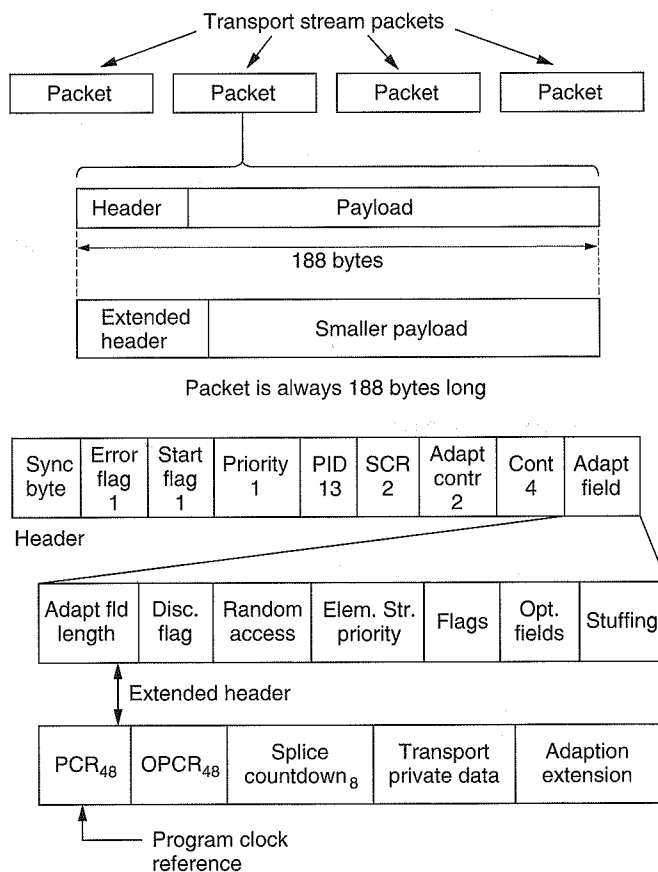


Figure 6.5 Transport stream packets are always 188 bytes long to facilitate multiplexing and error correction.

Transport stream packets always begin with a header. The remainder of the packet carries data known as the payload. For efficiency, the normal header is relatively small, but for special purposes the header may be extended. In this case the payload gets smaller so that the overall size of the packet is unchanged.

The header begins with a sync byte which is an unique pattern detected by a demultiplexer. A transport stream may contain many different elementary streams and these are identified by giving each an unique 13-bit Packet Identification Code or PID which is included in the header. A multiplexer seeking a particular elementary stream simply checks the PID of every packet and accepts those which match, rejecting the rest.

In a multiplex there may be many packets from other programs in between packets of a given PID. To help the demultiplexer, the packet header contains a continuity count. This is a 4-bit value which increments at each new packet having a given PID.

This approach allows statistical multiplexing as it does matter how many or how few packets have a given PID; the demux will still find them. Statistical multiplexing has the problem that it is virtually impossible to make the sum of the input bit rates constant. Instead the multiplexer aims to make the average data bit rate slightly less than the maximum and the overall bit rate is kept constant by adding 'stuffing' or null packets. These packets have no meaning, but simply keep the bit rate constant. Null packets always have a PID of 8191 (all ones) and the demultiplexer discards them.

6.4 Clock references

A transport stream is a multiplex of several TV programs and these may have originated from widely different locations. It is impractical to expect all the programs in a transport stream to be synchronous and so the stream is designed from the outset to allow asynchronous programs. A decoder running from a transport stream has to genlock to the encoder and the transport stream has to have a mechanism to allow this to be done independently for each program. The synchronizing mechanism is called Program Clock Reference (PCR).

In program streams all the programs must be synchronous so that only one clock is required at the decoder. In this case the synchronizing mechanism is called System Clock Reference (SCR).

Figure 6.6 shows how the PCR/SCR system works. The goal is to recreate at the decoder a 27 MHz clock which is synchronous with that at the encoder. The encoder clock drives a 48-bit counter which continuously counts up to the maximum value before overflowing and beginning again.

A transport stream multiplexer will periodically sample the counter and place the state of the count in an extended packet header as a PCR (see Figure 6.5). The demultiplexer selects only the PIDs of the required program, and it will extract the PCRs from the packets in which they were

inserted. In a program stream the count is placed in a pack header as an SCR which the decoder can identify.

The PCR/SCR codes are used to control a numerically locked loop (NLL). This is similar to a phase-locked loop, except that the two phases concerned are represented by the state of a binary number. The NLL contains a 27 MHz VCXO (Voltage Controlled Crystal Oscillator), a variable-frequency oscillator based on a crystal which has a relatively small frequency range.

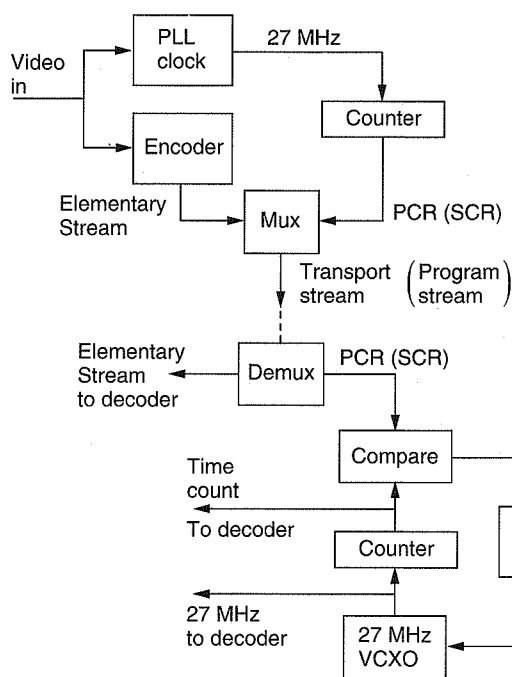


Figure 6.6 Program or System Clock Reference codes regenerate a clock at the decoder. See text for details.

The VCXO drives a 48-bit counter in the same way as in the encoder. The state of the counter is compared with the contents of the PCR/SCR and the difference is used to modify the VCXO frequency. When the loop reaches lock, the decoder counter would arrive at the same value as is contained in the PCR/SCR and no change in the VCXO would then occur. In practice the transport stream packets will suffer from transmission jitter and this will create phase noise in the loop. This is removed by the loop filter so that the VCXO effectively averages a large number of phase errors.

A heavily damped loop will reject jitter well, but will take a long time to lock. Lock-up time can be reduced when switching to a new program if the decoder counter is jammed to the value of the first PCR received in

the new program. The loop filter may also have its time constants shortened during lock-up. Once a synchronous 27 MHz clock is available at the decoder, this can be divided down to provide the 90 KHz clock which drives the time stamp mechanism. The entire timebase stability of the decoder is no better than the stability of the clock derived from PCR/SCR. MPEG-2 sets standards for the maximum amount of jitter which can be present in PCRs in a real transport stream.

6.5 Program Specific Information (PSI)

In a real transport stream, each Elementary Stream has a different PID, but the demultiplexer has to be told what these PIDs are and what audio belongs with what video before it can operate. This is the function of PSI. Figure 6.7 shows the structure of PSI. When a decoder powers up, it knows nothing about the incoming transport stream except that it must search for all packets with a PID of zero. PID zero is reserved for the Program Association Table (PAT). The PAT is transmitted at regular intervals and contains a list of all the programs in this transport stream. Each program is further described by its own Program Map Table (PMT) and the PIDs of the PMTs are contained in the PAT.

Figure 6.7 also shows that the PMTs fully describe each program. The PID of the video Elementary Stream is defined, along with the PID(s) of

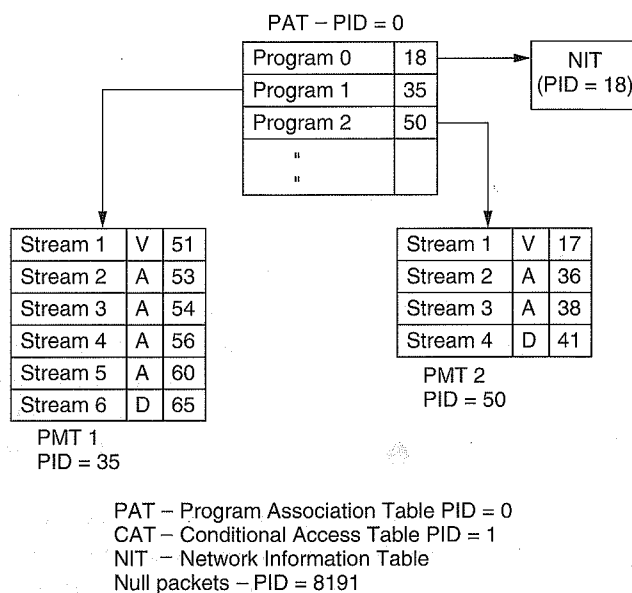


Figure 6.7 MPEG-2 Program Specific Information (PSI) is used to tell a demultiplexer what the transport stream contains.

the associated audio and data streams. Consequently when the viewer selects a particular program, the demultiplexer looks up the program number in the PAT, finds the right PMT and reads the audio, video and data PIDs. It then selects Elementary Streams having these PIDs from the transport stream and routes them to the decoders.

Program 0 of the PAT contains the PID of the Network Information Table (NIT). This contains information about what other transport streams are available. For example, in the case of a satellite broadcast, the NIT would detail the orbital position, the polarization, carrier frequency and modulation scheme. Using the NIT a set top box could automatically switch between transport streams.

Apart from 0 and 8191, a PID of 1 is also reserved for the Conditional Access Table (CAT). This is part of the access control mechanism needed to support pay per view or subscription viewing.

6.6 Multiplexing

A transport stream multiplexer is a complex device because of the number of functions it must perform. A fixed multiplexer will be considered first. In a fixed multiplexer, the bit rate of each of the programs must be specified so that the sum does not exceed the payload bit rate of the transport stream. The payload bit rate is the overall bit rate less the packet headers and PSI rate.

In practice the programs will not be synchronous to one another, but the transport stream must produce a constant packet rate given by the bit rate divided by 188 bytes, the packet length. Figure 6.8 shows how this is

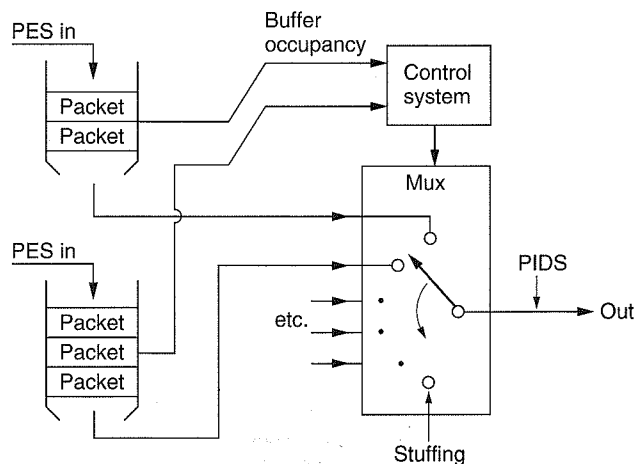


Figure 6.8 A transport stream multiplexer can handle several programs which are asynchronous to one another and to the transport stream clock. See text for details.

handled. Each elementary stream entering the multiplexer passes through a buffer which is divided into payload-sized areas. Note that periodically the payload area is made smaller because of the requirement to insert PCR.

MPEG-2 decoders also have a quantity of buffer memory. The challenge to the multiplexer is to take packets from each program in such a way that neither its own buffers nor the buffers in any decoder either overflow or underflow. This requirement is met by sending packets from all programs as evenly as possible rather than bunching together a lot of packets from one program. When the bit rates of the programs are different, the only way this can be handled is to use the buffer contents indicators. The fuller a buffer is, the more likely it should be that a packet will be read from it. This a buffer content arbitrator can decide which program should have a packet allocated next.

If the sum of the input bit rates is correct, the buffers should all slowly empty because the overall input bit rate has to be less than the payload bit rate. This allows for the insertion of Program Specific Information. While PATs and PMTs are being transmitted, the program buffers will fill up again. The multiplexer can also fill the buffers by sending more PCRs as this reduces the payload of each packet. In the event that the multiplexer has sent enough of everything but still can't fill a packet then it will send a null packet with a PID of 8191. Decoders will discard null packets and as they convey no useful data, the multiplexer buffers will all fill while null packets are being transmitted.

In a statistical multiplexer or statmux, the bit rate allocated to each program can vary dynamically. Figure 6.9 shows that there must be tight connection between the statmux and the associated compressors. Each compressor has a buffer memory which is emptied by a demand clock from the statmux. In a normal, fixed bit rate, coder the buffer contents feeds back and controls the requantizer. In statmuxing this process is less

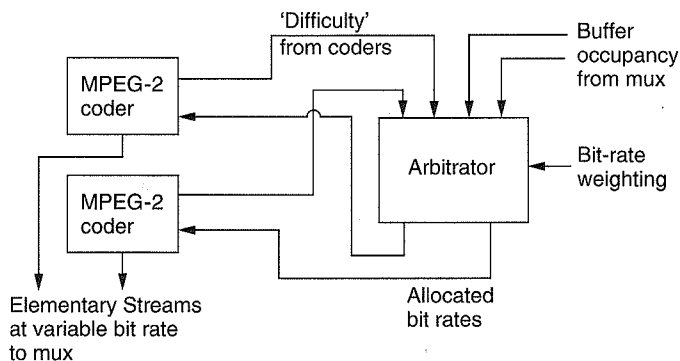


Figure 6.9 A statistical multiplexer contains an arbitrator which allocates bit rate to each program as a function of program difficulty.

severe and only takes place if the buffer is very close to full, because the buffer content is also fed to the statmux.

The statmux contains an arbitrator which allocates packets to the program with the fullest buffer. Thus if a particular program encounters difficult material it will produce large prediction errors and begin to fill its output buffer. This will cause the statmux to allocate more packets to that program. In order to fill more packets, the statmux clocks more data out of that buffer, causing the level to fall again. Of course, this is only possible if the other programs in the transport stream are handling typical video.

In the event that several programs encounter difficult material at once, clearly the buffer contents will rise and the requantizing mechanism will have to operate.

6.7 Remultiplexing

In real life a program creator may produce a transport stream which carries all its programs simultaneously. A service provider may take in several such streams and create its own transport stream by selecting different programs from different sources. In an MPEG-2 environment this requires a remultiplexer, also known as a transmultiplexer. Figure 6.10 shows what a remultiplexer does.

Remultiplexing is easier when all the incoming programs have the same bit rate. If a suitable combination of programs is selected it is obvious that the output transport stream will always have sufficient bit rate. Where statistical multiplexing has been used, there is a possibility that the sum of the bit rates of the selected programs will exceed the bit

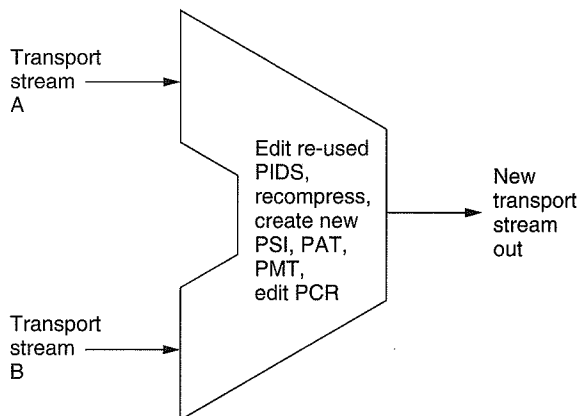


Figure 6.10 A remultiplexer creates a new transport stream from selected programs in other transport streams.

rate of the output transport stream. To avoid this, the remultiplexer will have to employ recompression.

Recompression requires a partial decode of the bitstream to identify the DCT coefficients. These will then be requantized to reduce the bit rate until it is low enough to fit the output transport stream.

Remultiplexers have to edit the Program Specific Information (PSI) such that the Program Association Table (PAT) and the Program Map Tables (PMT) correctly reflect the new transport stream content. It may also be necessary to change the packet identification codes (PIDs) since the incoming transport streams could have inadvertently used the same values.

When Program Clock Reference (PCR) data are included in an extended packet header, they represent a real-time clock count and if the associated packet is moved in time the PCR value will be wrong. Remultiplexers have to recreate a new multiplex from a number of other multiplexes and it is inevitable that this process will result in packets being placed in different locations in the output transport stream than they had in the input. In this case the remultiplexer must edit the PCR values so that they reflect the value the clock counter would have had at the location the packet at which the packet now resides.

Reference

1. Watkinson, J.R., *The Art of Digital Video*, Chapter 6, Oxford: Focal Press (1994)