

Digital Broadcast Media

MU60012E
Level 6

Assignment 1: Compression



The FLAC Codec
“Free Lossless Audio Codec”

Student: Josh Fairhead (21057665)
Lecturer: John Gummery

Table of Contents

About FLAC.....	1
How FLAC works.....	1
Blocking.....	1
Inter-channel decorrelation.....	1
Prediction.....	2
Verbatim Prediction.....	2
Constant Prediction	2
Fixed Linear prediction.....	2
Orders.....	3
FIR Linear prediction	3
Rice Codes.....	3
Unary.....	3
Binary.....	3
Rice.....	4
Format.....	4
Conclusion.....	4
References.....	5
Glossary.....	6
Data Flow Chart.....	7

About FLAC

The FLAC codec is open source and stands for Free Lossless Audio Codec. (Coalson 2008 a.) It was developed by Josh Coalson and since January 2003 has been developed with the Xiph.Org foundation (Plant 2003).

How FLAC works

According to Salomon (2007) “*FLAC compresses the audio input block by block and is based on prediction and Rice codes... A block of audio samples is compressed by predicting the samples, subtracting each audio sample from its prediction, and encoding the difference with a Rice code*”.

Blocking

“A block is one or more audio samples that span several channels” (Coalson 2008 b)
In FLAC terminology a block is the unencoded audio data and frames are FLAC encoded data.

Blocking is the process of breaking up the audio data into a set range of audio samples that span several channels. Blocks can vary in size dependent on the source material, but currently each FLAC file uses only one block size chosen by the reference encoder, which is essentially the algorithm for choosing any parameter throughout the process.

Optimal block size during prediction is dependent on the input and is determined by many variables such as sample rate and frequency etc. (Ibid)

This block is then broken up into sub-blocks, these contain the same amount of samples as the block but are a single channel of the audio information. (Ibid)

Subblocks are then passed to either the inter-channel decorrelation stage or to the prediction stage dependent on how many channels the block contains, after which they are coded into subframes. Subframes allow a variable size within this stream and are then regrouped into a frame. Because each subblock (channel) is coded separately the encoding method used can be varied and the encoder chooses the most efficient. (Ibid)

Inter-channel decorrelation

Once the audio is blocked the FLAC codec uses common information between stereo channels to reduce data size. It can do this in several ways of which are interchangeable on a frame by frame basis by using differing channel assignments. Note that currently FLAC can only use inter-channel decorrelation on stereo streams, though the format supports multiple channel streams also. (Ibid)

FLAC codes each blocks channel (subblock) independently. In stereo streams it can skip any inter-channel decorrelation known as Independent encoding or it can encode with mid-side, left-side and right-side channel assignments. (Ibid)

Mid channel encoding is the average of the two channels. (Ibid)

Side channel encoding is the difference between the left and right channels and is worked out by subtracting the left channels value from the right channels value. (Ibid)

$$(L+R) / 2 = \text{Mid}$$
$$L - R = \text{Side}$$

When unfolding mid-side, the side channels value is halved and then added to the mid channels value to create the left channel. The halved side channel value is subtracted from the mid channel to create the right channel. (Surround Devices 2007)

$$\begin{aligned} Mid + (Side/2) &= Left \\ Mid - (Side/2) &= Right \end{aligned}$$

When unfolding left-side encoding, the side value is subtracted from the left channel giving the right channel.

When unfolding right-side encoding, the side value is added to the right channel giving the left channel.

The reason there are several methods of encoding the channel assignments is that a specific channel may compress better than others and is thus preferable to other channels. For instance if the left channel compresses better than both the mid channel and the right channel then left-side encoding will be used providing that the side channel is shorter to code than the right channel, which it usually is. The fact that FLAC can vary its channel assignment on a frame by frame basis means that this can greatly reduce the size of a file by choosing the assignment that requires least space for each subframe as the audio changes. To figure out which method of encoding is optimum FLAC tries each method and chooses the best. (Willoughby 2009)

Prediction

A predictive codec works on the principal that the encoder and the decoder will predict the same value. They base their predictions off the data that has come beforehand and try to find the most likely value (Watkinson 2002).

This value is compared to the original signal and examined by the codec to find any discrepancies between them, these discrepancies are known as a residual or a prediction error (Ibid).

This system of rechecking the data allows the decoder to safely predict a correct value if no residual is found in the file, resulting in no residual needing to be coded and stored. Any residuals found by the codec on input are encoded with a rice code to the file. The decoder then adds this residual value to its prediction which once again gives an accurate value on output (Ibid).

Once the audio has been given a channel assignment and the process of decorrelating each channel has been completed, the audio is passed on to the prediction stage.

FLAC currently has four methods of prediction but has reserved space for future development. These are Verbatim prediction, Constant prediction, Fixed linear prediction and FIR Linear prediction (Coalson 2008b).

Verbatim Prediction

Verbatim prediction is when the encoder predicts the signal to be zero (which it won't be), this leaves the residual to be the whole signal, resulting in no compression. The residual of the verbatim predictor, which is the raw signal, is not coded in the residual coding stage and is added to the the FLAC stream as "verbatim" (Ibid).

Constant Prediction

Constant prediction is used when the signal is a constant value for the length of the subblock. This signal is then run length encoded and added to the stream (Ibid).

Run length encoding is when a string of consecutive numbers are stored as their count and representative value (Campos 1999).

Fixed Linear prediction

Fixed linear prediction is the computation of a given sample in a linear fashion from a preceding value(s), this is then quantized to a set value. Since the quantization values are static there is only need to store the order class in the compressed stream.

Orders

With the FLAC fixed linear predictor there are four orders of prediction after which the error is passed on for residual coding (Coalson 2008b).

In first order prediction the sample that will be predicted $s(t)$ is based off the sample preceding it $s(t-1)$ and will be the same value. (Salomon 2007)

Second order prediction is when the predictor computes a straight line or linear function from the two samples preceding it $s(t-1)$ and $s(t-2)$ this is also known as a degree 1 polynomial (Ibid).

A polynomial is an expression involving x raised to a whole number power (exponent). A degree of a polynomial is the highest exponent of x . (Ransom 2009)

A third order prediction is obtained through computing a degree 2 polynomial that passes through three preceding samples $s(t-1)$, $s(t-2)$ and $s(t-3)$. A degree 2 polynomial is a parabolic curve (Ibid).

A fourth order prediction is based of four previous values and a degree 3 polynomial running through them. This should be continuos nonlinear curve (Ibid).

Though the fixed linear encoder only goes up to fourth order prediction the FIR linear predictor can go as high as 32nd order (Coalson 2008b). Due to this I will explain how to calculate and graph further polynomials.

You can calculate an n th order prediction through calculating the degree ($n-1$) polynomial that passes through n points $s(t-n)$ through to $s(t-1)$. (Salomon 2007)

To graph polynomials, which is essentially what a predictor does, you can use the formula:

$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \dots + a_nx^n$, where a_n is the value of $s(t-n)$ in a series, this should result in a continuos non linear curve. (Seward 2008)

FIR Linear prediction

A FIR Linear predictor is a forward prediction error filter (Constantinides, date unknown) and can also be called a non-recursive filter. This predictor differs from the fixed linear predictor in that the quantization value is not static throughout the whole subblock.

As the quantization value is allowed to change to be more efficient it can add an overhead since it has to store the quantization value as well as the order and residual. The output residual is then rice coded. (Coalson 2008b)

Rice Codes

To understand a rice code it is necessary to understand unary and binary coding.

Unary

A number in unary code is written with $n-1$ zeros followed by a one. (Moffat & Turpin 2002) Here n represents the number to be encoded. So in order to write 4 in unary, it is 4-1 zeros followed by a one, resulting in 0001.

It is worth mentioning that unary numbers can be coded either as ones followed by a zero or zeros followed by a one, it makes little difference as the last digit represents the end of the code (Anderson 2008). This last digit allows a decoder to realise when a new number begins.

Binary

Binary uses place to give a value. Each place is worth twice as much as the place after it (Bradley 2009). Taking the number 23 as an example the binary would be 10111

The binary value 10111 translates to one 16, zero 8s, one 4, one 2 and a 1 which equals 23.

Rice

The residuals from prediction are split into two numbers by dividing another number into them. This dividing number is dependent on an encoding parameter which is a power of two and is determined by the codec. The codec selects the most efficient parameter for coding the data and attaches an identifier to the block for decoding (Sayood 2000).

The residual is divided by the encoding parameter and the resulting numbers, the quotient and remainder are then encoded in unary and binary respectively. (Fietz 2009)

For reference in this circumstance, a quotient is a whole number resulting from a division with anything left over called a remainder.

An example of rice coding:

Encoding the number 50 as a rice code, taking the encoding parameter to be 3.

We will divide 50 by eight because the encoding parameter 3 is used as a power of 2 i.e. 2^3 or $2 \times 2 \times 2$ which results in 8.

So $50 \div 8 = 6$ with remainder of 2.

The 6 is encoded in unary which is 000001

The 1 is encoded in binary which is 10.

The resultant rice code is a combination of the two 000001 10.

Format

The resulting codes from the prediction stage are then set into a subframe with header information about the data. These subframes are then framed and each frame is added to the stream along with meta data relating to the file such as album art etc.

Conclusion

As the project leader admits the project could still “explore better context-modeling of the Rice parameter, or Huffman coding.”
(Coalsen 2008b).

The format has been built with the idea of improving and has reserved space for new methods (Ibid). In my opinion this will prove a good choice for the future though the lack of these coding methods are currently a weakness.

With drive space and bandwidth increasing people will likely move away from lossy formats and may start requiring full quality audio. The way FLAC has been designed allows for safe streaming which will be of much importance in the choice of a broadcast format (Coalsen 2008b). It has in fact been adopted in systems of the EBU such as the Musipop file transfer system though not yet as a streamer. (Cook 2007)

FLAC is backwards compatible with older versions of the format and since it has also been designed to allow older versions of the decoder to skip over metadata that is not understood thus been made relatively future-proof. This along with being a free format, make it a good codec to implement into hardware which may prove important for the FLAC future. It also makes it a good choice as an archival format.

As a format I think FLAC will be a good choice of codec to supersede most older formats of music codecs. Its lossless, its free, it can be implemented into hardware, it can be streamed effectively and is relatively efficient compared to other lossless codecs.

References

Anderson, T. (2008) Golomb-Rice Coding, Tom who is contrary.

[http://urchin.earth.li/~twic/Golomb-Rice_Coding.html]

Date accessed 31/10/09

Bradley, A. (2009) Binary Code, Welcome to About.com.

[<http://php.about.com/od/programingglossary/qt/binary.htm>]

Date accessed 31/10/09

Campos, A. (1999) Run Length Encoding by Arturo Campos, Arturo Campos home page.

[http://www.arturocampos.com/ac_rle.html]

Coalson, J. (2008a). FLAC Features, FLAC Free Audio Lossless Codec. [<http://flac.sourceforge.net/features.html>]

Date accessed 15/10/09

Coalson, J. (2008b) FLAC Format, FLAC Free Audio Lossless Codec. [<http://flac.sourceforge.net/format.html>]

Date accessed 29/10/09

Constantinides. A (date unknown) A prediction problem, Power Point Slides.

[http://www.commsp.ee.ic.ac.uk/~agc/course4_files/Lecture_Slides_0304/6-Spectrum%20Estimation/6-A%20Prediction%20Problem.ppt]

Date accessed 03/11/09

Cook, T. (2007) Announcements, Welcome to LWN.net. [<http://lwn.net/Articles/262095/>]

Moffat & Turpin. (2002) Compression and coding algorithms first edition, The Netherlands: Kluwer Academic Publishers Group. (p29)

Fietz, H. (2009) I don't get Golomb/Rice coding, Stack Overflow.

[[http://stackoverflow.com/questions/728966/i-don't-get-golomb-rice-coding-it-does-make-more-bits-of-the-input-or-does-it](http://stackoverflow.com/questions/728966/i-don-t-get-golomb-rice-coding-it-does-make-more-bits-of-the-input-or-does-it)]

Date accessed 31/10/09

Plant, E. (2003) Flac Joins Xiph!, Xiph.org.

[<http://xiph.org/ogg/flac.html>]

Date accessed 15/10/09

Ransom, M. (2009) Graphing Polynomials, AlgebraLAB: Making math and science connections.

[http://www.algebralab.org/lessons/lesson.aspx?file=algebra_poly_graphs.xml]

Date accessed 15/10/09

Salomon, D. (2007) Data compression: the complete reference fourth edition. London: Springer (p758 and 763)

Seward, K. (2008) College algebra tutorial on graphs of polynomial functions, Virtual Math Lab.

[http://www.wtamu.edu/academic/anns/mps/math/mathlab/col_alg_tut35_polyfun.htm]

Sayood, K. (2000) Introduction to data compression second edition. San Diego CA: Academic Press (p64)

Surround Devices. (2007)

MS stereo basics, Sound Devices LLC || Professional, Portable Audio Products.

[http://www.sounddevices.com/tech/ms_stereo.htm]

Watkinson, J. (2002) An introduction to digital audio second edition. Oxford: Focal Press

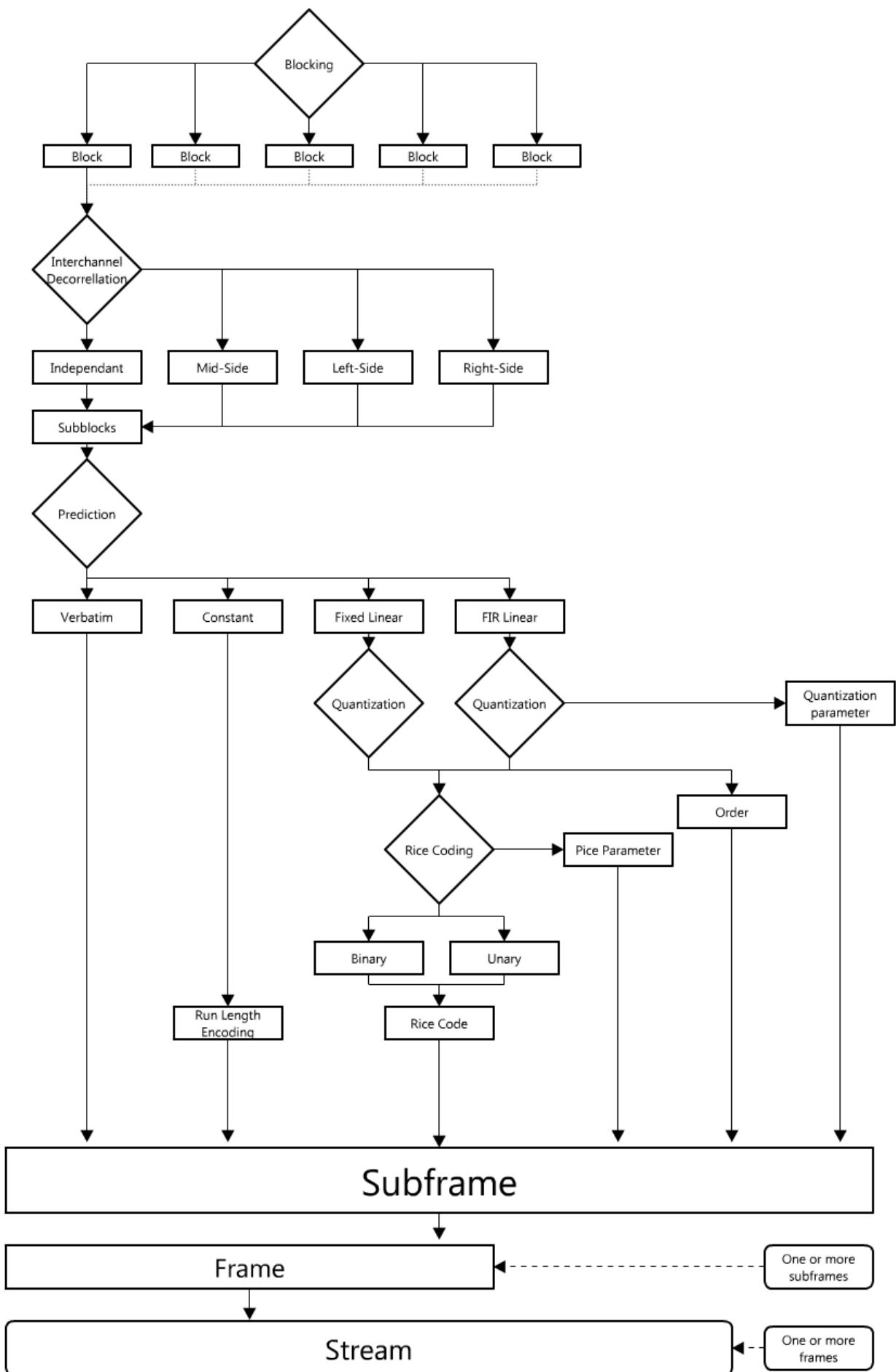
(p160)

Willoughby, B. (2009) Technical questions about FLAC, The Mail Archive.

[<http://www.mail-archive.com/flac@xiph.org/msg00985.html>] Date accessed 01/11/09

Glossary

Block:	One or more audio samples that span one or more channels
Subblock:	One ore more audio sample within a block
Frame:	A frame header and one or more subframes
Subframe:	A subframe header and one or more encoded samples from a given channel
(Coalson 2008b)	



Data Flow Chart