Investigation of MP3 Compression Quality

Executive Summary

This paper shows a practical application of a general factorial experiment to analyze the interactions between important controllable factors in the creation of high quality compressed (MP3) music files. Traditional sound quality experiments depend on subjective listeners' opinions and this experiment instead utilizes an objective measure of fidelity based on digital signal analysis of an encoded and decoded music file compared with the original clip. Several factors were varied including encoder, bitrate, decoder, and sampling rate on three different music clips. Results indicate that lower sampling frequencies combined with high bitrates using the Blade encoder achieved the highest fidelity. Decoder type was insignificant in determining quality. Future research possibilities include a better measure of fidelity and a larger scale experiment to address recommendations for specific music types.

1.0 Introduction

Portable music players have become widely popular within the past decade. The MP3 format (formally known as MPEG-1 Layer III) has been one of the primary catalysts of this development. It allows for the storage of far more songs in a smaller amount of disk space than the traditional "Red Book" standard used for uncompressed music files stored on compact disc, which defined the baseline of digital audio from its inception in 1980. MP3 offers varying levels of compression to reduce uncompressed music files to a much smaller size, often of a ratio of 20:1 or even higher. Such compression is necessary to fit a large library of songs into a portable device. MP3 is, however, a so-called lossy compression format in that some of the original audio information is irrevocably lost during compression, and cannot be recovered by any means. Thus, the proper implementation of the compression (encoding/decoding) process is essential to maintaining an acceptable level of playback quality.

The format is standardized only based on its file format and to a lesser extent the way in which MP3-encoded content is decoded. Many different encoders have been produced over the years that support the MP3 standard as it is written, although sometimes with very different operational approaches. Basic options in almost all encoders include bitrate (expressed in kbps or kilobits per second, which fixes the size of the resulting compressed file), input sampling frequency (most commonly 44.1kHz, consistent with the original Red Book standard that defined "CD-quality" sound), as well as options for monophonic or stereophonic sound.

2.0 Improvement Opportunity: Define Phase

Of course, with the proliferation of MP3 encoders and various options have come attempts to guage the quality of various encoder, bitrate, sampling frequency, and decoder options. These tests have been almost entirely based on subjective listening tests (such as that presented by Bouvigne 1998 and Amorim 2004), where the same audio track is encoded with several different options and encoders and presented to a trained listener. The audio track is then rated on several factors, mostly with respect to fidelity to the original recording.

There are several problems with such a rating system, as it ultimately relies on subjective perceptual judgments that may differ vastly from person to person. Also, there are limits on how many different option combinations that a human listener can tolerate at any one time. Listening preferences could easily change between testing sessions or even within the session. This could make the usability of a large factorial experiment, for example, quite limited. Also troubling is the potential for highly non-normal results, which could skew or make normal factorial analyses statistically unusable.

3.0 Performance: The Measure Phase

As a result of the limits of traditional listening-based tests, it was decided to form a testing situation that was compatible with designed experiment goals as well as rely on objective rather than subjective measures. The measure chosen for this experiment is based on the idea that a musical sample that has been encoded and then decoded should possess a similar frequency spectrum as the original. Two sounds that possess very similar frequency spectra over their durations sound very similar. Therefore, diversions from this ideal represent noise introduced by the encode/decode process. An example of such a relative spectrum is shown graphically in Figure 1.

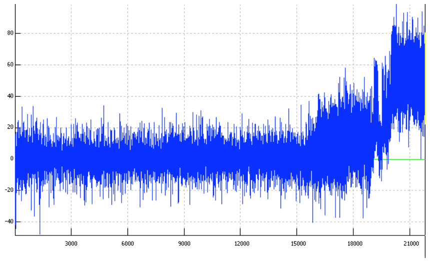


Figure 1: Relative spectrum between two audio files

Figure 1 shows a relative spectrum result between an encoded/decoded file and its original source. The relative spectrum is calculated as:

$$y = 20\log\frac{spectrum1}{spectrum2} \quad \text{(Equation 1)}$$

Spectrum 1 is from the original file and spectrum 2 is from a file that has been encoded and then decoded. The resulting scale is called a dBr scale by the Sigview software. In the example in Figure 1, spectrum 1 (from the original file) has a much higher representation of the higher frequencies above about 19 kHz. It appears the encoder or decoder dropped a lot of information at these high frequencies. This is not unexpected since most humans are quite insensitive to sound at such high frequency ranges. (The limitations of this sort of analysis are discussed later.) To form an overall quality metric based upon this graph, the RMS (root-mean-square) was taken, which gives an overall view of the deviations of the two files from each other. Each y value represents a very narrow part of the frequency spectrum.

$$RMSMetric = \sqrt{y_1^2 + y_2^2 + ... + y_n^2} \quad \text{(Equation 2)}$$

These relative spectra analyses were carred out with the Sigview 1.95 signal analysis software. As a result of this analysis, a treatment combination can be compared with its original source material using a simple quantitative metric. This will be the quality metric used for the rest of this report.

Several controllable factors can be considered for the analysis for this experiment. They include the encoder, the decoder, the sampling rate, and the bitrate. They are shown in tabular form below in Table 1.

Table 1: Controlled Experimental Factors

Encoder	Decoder	Bitrate	Sampling Rate
LAME v3.98a3	MPG123	32	32kHz
(FHG) MP3ENC v3.1	MAD v0.15.2	80	44.1kHz
BladeENC v0.94.2		128	

Uncontrollable factors should only occur if any of the software used does not perform the same operations on different files consistently or even if the same operation is performed differently on the same file at different times. This is assumed not to occur and no other uncontrollable factors should be present.

These factor choices were made based upon commonly available software and achievable settings within software. LAME and BladeENC are enthusiast-written MP3 encoders. MP3ENC is an earlier freely available demo version of the commercial encoder produced by Fraunhofer IIS (FHG), who holds the original MP3 patents. MAD and MPG123 are widely available free command-line MP3 decoders (i.e. they are not music players, they only decode the MP3 compression and output it to a file) All software used claimed compliance with the applicable ISO (International Organization for Standardization) directives for MP3. ISO publishes some reference software programming code for encoding and decoding, although some programs claim not to use any of the ISO code.

A general full factorial analysis was chosen because the encoder treatment levels are qualitative and contain more than two levels. Also, it is possible there might be non-linear responses for bitrate versus quality so checking three levels of this factor will be desirable and center points cannot be utilized to analyze this because there are qualitative factors. As a result, a streamlined 2^k analysis is not possible. Relatively few experimental resources are needed to create and analyze these files, so there is little benefit in running a fractional factorial anyway. In total, to fully explore all treatments at all levels, 36 runs are needed per replicate.

To analyze different types of music (the inputs for this process), three different musical selections were chosen for each of three replicates, and the runs were then blocked on these replicates to account for the differences in conclusions that might result from each different music selection. In total, 108 individual runs were made. Minitab R14 was used to set up and analyze the results. Randomization of experiment execution order was deemed unnecessary since the order in which each treatment was created and analyzed would have no effect on the results of the signal analysis in Sigview. No part of the analysis could be in any way affected by the run order chosen since the runs are in effect totally independent of the order in which they are executed: there is no process “memory” at any point.

To begin, three 10-second audio samples from three different audio tracks were taken to represent general musical selections. The selected tracks were:

1. DJ Shadow – “The Number Song” (0:00 to 0:10) from *Endtroducing…* (1998)
2. Dan Siegel – “Where Are You Now” (0:25 to 0:35) from *Another Time, Another Place* (1984)
3. Mannheim Steamroller – “Going to Another Place” (2:15 to 2:25) from *Fresh Aire II* (1977)

Each song selection was extracted from an original retail CD using Exact Audio Copy V0.95 beta 4 to copy the original 16-bit stereo 44.1kHz files from CD to a PC. These files then had the applicable 10-second portions mentioned above trimmed out of them with the audio editing package Audacity 1.2.3. Audacity was also used to create 32kHz sampling rate equivalents of these 10-second snippets. Generally, sampling rates are chosen to be roughly double or more the top end of human hearing (which is about 20kHz) to avoid signal aliasing but can be set lower to have a smaller file size. Monaural files were created from the original stereo in this experiment to avoid technical limitations with analyzing stereo files. This will only affect the bitrate settings recommended. A 32kbps mono bitrate file would be doubled to about 64kbps to achieve comparable quality results if it were in stereo for most encoders.

Each uncompressed file (either in its 32kHz or 44.1kHz sampling rate form) was then encoded using all combinations of encoder, sampling rate, and decoder. Specific command line options invoked for each encoder and decoder are shown in the Appendix. Sigview 1.95 then was used to create the frequency spectra and calculate the relative spectra discussed above, along with their RMS values.

4.0 Analysis and Interpretation: The Analyze Phase

Before proceeding with the analysis, it was decided to check to see if a data transformation might be appropriate for this data, as it was highly possible that the data was non-normal since sound intensity and frequency scales are based on logarithms. The results of the Box-Cox transformation performed on all 108 RMS readings are shown below in Figure 2.

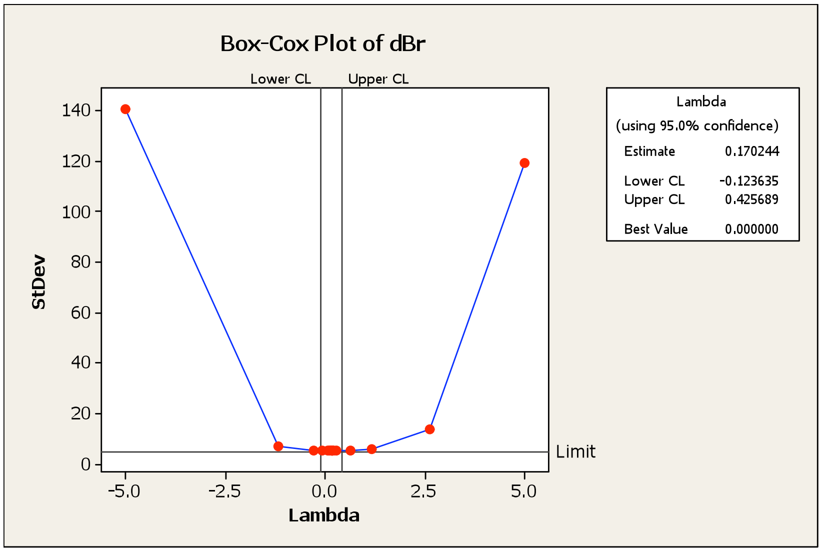


Figure 2: Box-Cox Transformation

This analysis recommends a log transformation of this data as the best fit, which was somewhat anticipated as mentioned above. The data was transformed to a natural log scale and then it was analyzed in its entirety. The resulting ANOVA table is shown below.

Table 2: ANOVA for General Full Factorial (General Linear Model) (α=0.05)

```
Analysis of Variance for dBr-norm, using Adjusted SS for Tests

Source                            DF   Seq SS   Adj SS   Adj MS       F      P
Blocks                             2   1.2285   1.2285   0.6143    9.13  0.000
Sampling Rate                      1   6.9949   6.9949   6.9949  103.93  0.000
Bitrate                            2  29.9415  29.9415  14.9707  222.43  0.000
Encoder                            2  11.3195  11.3195   5.6597   84.09  0.000
Decoder                            1   0.1908   0.1908   0.1908    2.83  0.097
Sampling Rate*Bitrate              2   1.7034   1.7034   0.8517   12.65  0.000
Sampling Rate*Encoder              2   0.4427   0.4427   0.2213    3.29  0.043
Sampling Rate*Decoder              1   0.0007   0.0007   0.0007    0.01  0.917
Bitrate*Encoder                    4   2.8176   2.8176   0.7044   10.47  0.000
Bitrate*Decoder                    2   0.0110   0.0110   0.0055    0.08  0.922
Encoder*Decoder                    2   0.0001   0.0001   0.0000    0.00  0.999
Sampling Rate*Bitrate*Encoder      4   1.8846   1.8846   0.4711    7.00  0.000
Sampling Rate*Bitrate*Decoder      2   0.0018   0.0018   0.0009    0.01  0.987
Sampling Rate*Encoder*Decoder      2   0.0021   0.0021   0.0010    0.02  0.985
Bitrate*Encoder*Decoder            4   0.0056   0.0056   0.0014    0.02  0.999
Sampling Rate*Bitrate*Encoder*     4   0.0073   0.0073   0.0018    0.03  0.999
  Decoder
Error                             70   4.7114   4.7114   0.0673
Total                            107  61.2634

S = 0.259434   R-Sq = 92.31%   R-Sq(adj) = 88.24%
```

Prior to using the analysis above to come to any conclusions, the residuals were examined for possible departures from normality that might invalidate an ANOVA analysis. The following figure (Figure 3) shows the residuals versus the fitted values from the model. Some clustering of residuals is apparent on the right side of the figure, as well as a few rather large residuals at the top of graph, but nothing highly non-random appears. A normality plot of residuals confirmed that there is no reason to reject the normality assumption at a significance level of 0.05, with a P-value of 0.163 on the Anderson-Darling test. This analysis presents no reasons to reject the analysis on the basis of lack of normality.

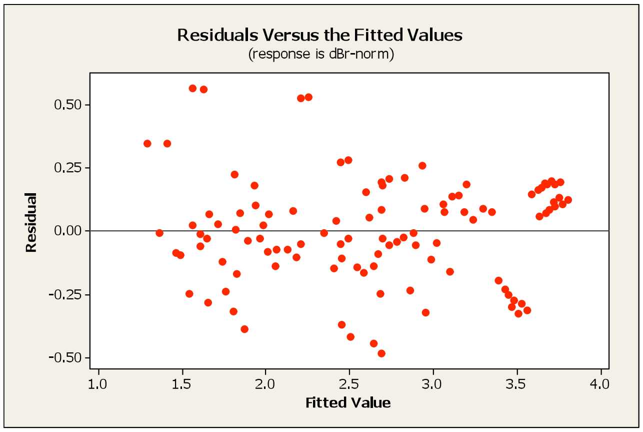


Figure 3: Residuals Versus Fitted Values (Transformed Data)

Upon first inspection of the ANOVA table for this model, it is evident there are many significant terms. The main effects are plotted in Figure 4. It is important to note that lower values of the quality metric are better, as it indicates better fidelity to the original file. All of the main effects except for decoder have a statistically significant effect on the spectrum quality of the encoded/decoded music files with respect to the original files. Decoder type is the only main effect not to be significant in this analysis. Perhaps predictably, bitrate is the single most important factor (it is responsible for the largest portion of the sum of squares) in the measured quality. This makes sense as it fixes the size of the encoded files and thus also most directly controls the compression level, which will ultimately have large effects on the amount of fidelity of the encoded file as compared to the original. It also exhibits a non-linear response, with diminishing returns occurring as bitrates increase. The quality improvement going from 32kbps to 80kbps is several times larger than the similarly spaced 80kbps to 128kbps improvement.

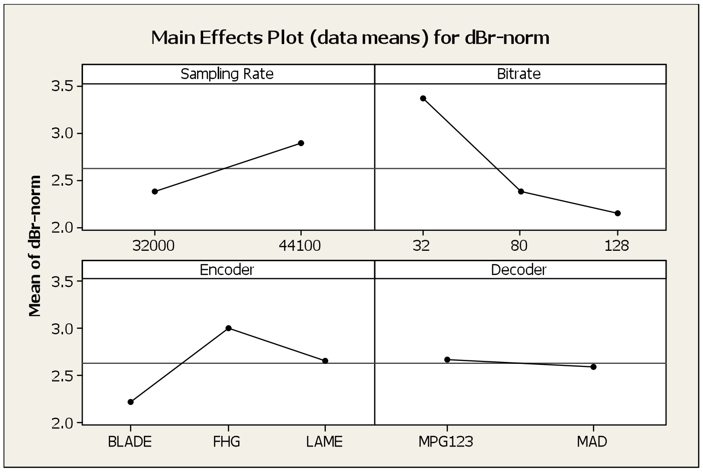


Figure 4: Main Effects Plots

Oddly enough, the lower sampling rate of 32kHz actually results in higher measured quality. This is understandable because the compression in 32kHz files will actually be lower as they contain less data to begin with, and thus need less compression to arrive at the predetermined size that the bitrate selection fixes. Note that fidelity measurements were taken with respect to original 32kHz or 44.1kHz files. If all fidelity measurements could have been compared to the original 44.1kHz files that would hold the 32kHz files to a higher standard, it is likely 44.1kHz encoded/decoded files would outperform 32kHz files. This was not possible, though, due to technical limitations in Sigview, where comparing two files with different sampling rates is not possible.

Finally, the best encoder by this analysis is actually BladeEnc. This is surprising because it is much older than LAME is now and also was not the "official" MP3 encoder from Fraunhofer (FHG), which actually performed worst of all! As expected, decoder type has no discernible effect on quality.

The two-way interaction plots are depicted in Figure 5 with significant interactions circled. All interactions not associated with the decoder are significant. For sampling rate vs. bitrate, it appears 44.1kHz sampling rate files exhibit far more linear total improvements in quality as bitrate increases compared to 32kHz files. 32kHz files experience more dramatic improvements in quality as bitrate increases.

For encoder vs. bitrate, it appears FHG's encoder actually performs more poorly at 128kbps than at 80kbps, causing a significant interaction to occur because the other encoders do not experience this anomalous behavior. The sampling rate vs. encoder interaction is barely significant (P=0.043) and no clear interaction pattern can be seen.

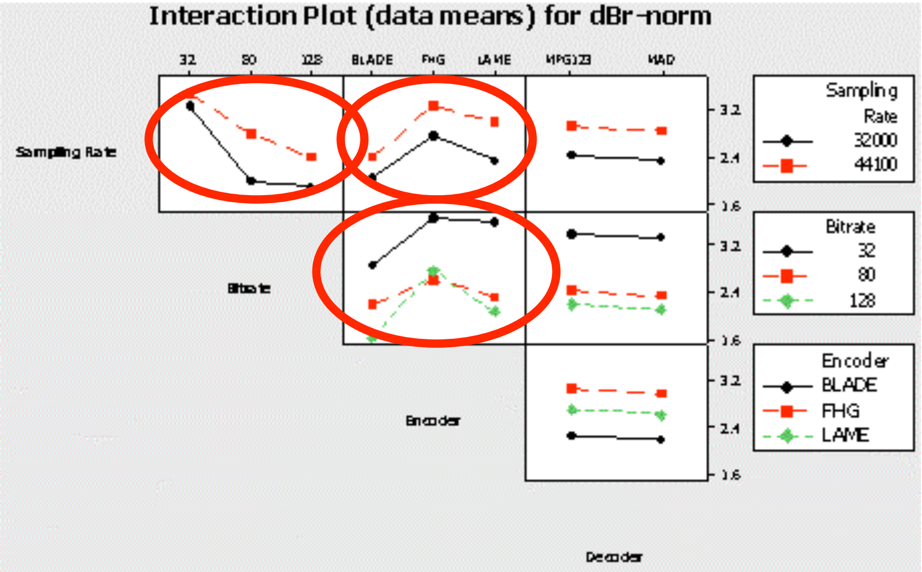
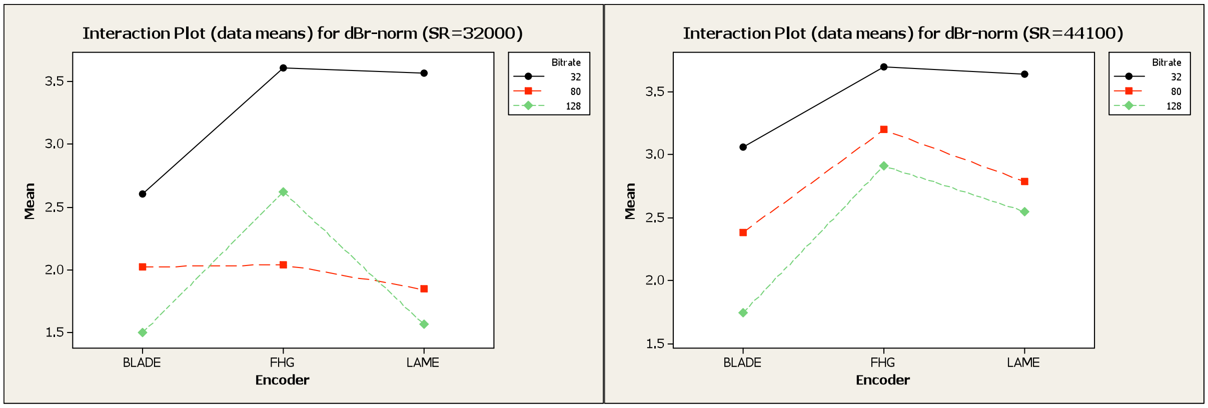


Figure 5: Two-way Interaction Plots

Perhaps most interesting, though, is the presence of a 3-way interaction between sampling rate, bitrate, and encoder. To analyze this, the two-way interaction plots for bitrate and encoder were stratified by sampling rate. These are illustrated below in Figures 6 and 7. At a bitrate of 32kbps, little difference is noticed in stratifying by sampling rate. Larger discrepancies are noted at the higher bitrates. Most notable is the fact that the FHG encoder at a sampling rate of 32kHz and a bitrate of 80kbps appears to outperform itself compared to 128kbps. This is unexpected as generally lower bitrates induce more compression, which lowers quality. This doesn't hold true for the FHG encoder at 32kHz. It is unclear what meaning this has, other than there might be a potential programming or algorithm error in the FHG encoder at 32kHz that produces higher than expected compression artifacts at 128kbps. The best explanation may well be that this particular encoder may not have anticipated being used in a rather unusual condition (32kHz source material is rarely, if ever, used for MP3s) and thus performed in an unexpected manner.



Figures 6 and 7: Examining the 3-way interaction of sampling rate, encoder, and bitrate.

The blocking factor (on replicates: music clips) was significant with the P-value very close to zero in this analysis in the ANOVA table. Therefore, blocking on replicates was an appropriate experimental procedure to follow.

Finally, the unadjusted R-Squared value for this model is about 92%. This means that 92% of the differences in quality between these files could be explained by the four factors or the blocked input variable (music clip). 8% of the variation must stem from other sources not explained by this model. Overall, though, an R-Squared value of 92% means that this model is excellent and explains almost all of the variation in quality measured.

5.0 Recommendations: The Improve Phase

A few recommendations can be made on the analysis above. First, the highest bitrate possible should always be selected that is practical for the storage space available, although returns do diminish with increasing bitrates. Based on this analysis, a 160kbps stereo MP3 would be much higher in quality than a 64kbps stereo MP3 given any encoder, but as large of an improvement would probably not be noticed by increasing it to a 256kbps bitrate. Decoders appear to play little role in ultimate quality as long as they are compliant with the applicable standards that define their operation. 32kHz outperforms 44.1kHz on the surface, but this is tempered with the knowledge that the 32kHz encoded/decoded files were only compared with 32 kHz originals, not with the inherently higher quality 44.1kHz originals. The lower sampling rate may have simply allowed for less compression and thus less distortion as compared to the originals they were compared to. 32kHz is not widely used as a sample rate for audio and 44.1kHz should be used so as to avoid unnecessary file conversions. BladeEnc appears to be the top performer with respect to the quality metrics discussed here, but the author subjectively would not have chosen it, especially at the lowest bitrate, where several aural artifacts were more obvious than those from the other encoders. Objective measures are good, but they should be sure to be tied with a user's subjective experience because that will ultimately determine the success of a music compression format, not an objective abstract measure.

There are improvments that could be made to this line of research to remedy the limitations of this quality metric. The relative spectrum calculated in these analyses relied on an unweighted frequency curve. However, human hearing is not unweighted. In fact, it is much more sensitive in some areas than others as shown in Figure 8.

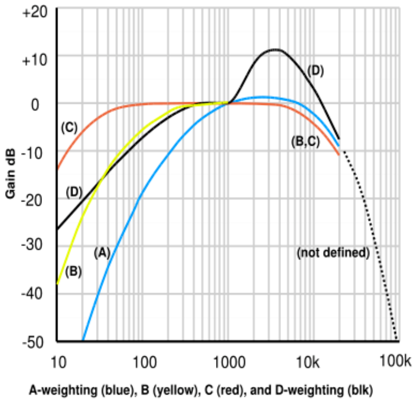


Figure 8: Frequency Weighting Curve (wikipedia.org)

Human hearing most closely follows the (A) curve indicated above. It is clearly not weighted evenly along the 0 to approximately 20kHz scale used in this analysis. As shown in Figure 1, it can be reasonably concluded that sacrifices are made at frequency levels where the human ear is not as sensitive. Also, the quality level is limited because, even if it is not intentional, departures from the baseline in encoded/decoded files will be perceived more readily by listeners at certain frequency bands. Unfortunately, no analysis software found was able to apply weighting curves to the RMS calculations. It would be a valuable extension to the current analysis and one that may become available as signal analysis software continues to advance.

If such analysis could be totally automated, it might be desirable also to analyze many different blocked replicates to come to widely applicable conclusions. Alternatively, different musical styles or types could be compared in different experimental analyses to compare recommendations. More encoders could be examined, as well as implementing stereophonic processing in the analysis, which would allow for greater resemblance between this model and reality.

References

Amorim, Roberto (2004). Public Listening Tests. (http://www.rjamorim.com/test/), accessed March 5, 2006.

Bouvigne, Gabriel (1998). MP3 Tech Listening Tests. (http://www.mp3-tech.org/tests/gb/index.html), accessed March 5, 2006.

Montgomery, Douglas C. Design and Analysis of Experiments, 6th ed. Hoboken, NJ: John Wiley & Sons, Inc., 2005.

Wikipedia.org (2006). A-weighting. (http://en.wikipedia.org/wiki/A-weighting), accessed March 5, 2006.

Software Used In This Report

Audacity. Audio Editor. (http://audacity.sourceforge.net/). Version 1.2.3.

BladeEnc. MP3 Encoding Software. (http://www2.arnes.si/~mmilut/BladeEnc.html). Version 0.94.2.

Exact Audio Copy. (http://www.exactaudiocopy.de/). Version 0.95 beta 4.

LAME. MP3 Encoding/Decoding Software. (http://lame.sourceforge.net/). Binaries available at (http://www.rarewares.org/mp3.html). Version 3.98a3.

MAD. MP3 Decoder. (http://www.underbit.com/products/mad/). Version 0.15.2.

Sigview: Digital Signal Analysis Software: (http://www.sigview.com/). Version 1.95.

Appendix

Table A-1: Examples of command-line options used for encoding and decoding software for a 32kbps, 32kHz MP3.

BLADE	*bladeenc -br 32 input.wav output.mp3* "-br 32" indicates 32kbps bitrate. Sampling rate is automatically calculated from the source.
MP3ENC	*mp3encdemo31 -if input.wav -of output.mp3 -br 32000 -esr 32000* "-br 32000" indicates 32kbps bitrate and "-esr 32000" indicates 32kHz sampling rate
LAME	*lame -q 0 -s 32 -b 32 --resample 32 input.wav output.mp3* "-s 32" indicates 32kHz sampling rate and "-b 32" indicates 32kbps bitrate "-q 0" is the highest quality setting
MPG123	Used MPG123 decoder embedded in LAME 3.98a3: *lame --decode input.mp3 output.wav*
MAD	*madplay input.mp3 output.wav*