AUDIO MASTERING EMULATOR

Anthony Sigler, Langchen Fan, Steven Crawford

University of Rochester
Department of Electrical and Computer Engineering / Biomedical Engineering

ABSTRACT

A convenient mastering application has been created in Matlab for the purpose of assisting individuals in achieving reliable and consistent results with the greatest possible simplicity. The application addresses the issue of quality mastering from two different standpoints. The first being user controlled and the second being adaptive. The adaptive portion of the application is based on the widely accepted psychoacoustic model for human hearing proposed by the International Organization for Standardization (ISO 226)[1]. Several audio files (previously mastered, and un-mastered) were processed using the application. Predictable and consistent results have verified the application's efficacy. The application is capable of processing .mp3, .wav, and .wma audio formats.

Index Terms **—** signal processing, mastering, equalization, psychoacoustics, adaptive effects, Matlab

1. INTRODUCTION

Mastering is one of the most important links in the music production chain and can be considered the final check for quality control in any sonic presentation. Homogenizing equalization curves and loudness levels is paramount in achieving aural balance and coherence. In today's world of downloaded and "self-recorded" music, there is little to no consistency in frequency balance or loudness level from song to song. The purpose of this mastering application is to provide the independent musician and/or the avid music connoisseur a method for achieving a consistent, predictable, and subjectively pleasing sonic balance in their audio files.

The two most prevalent tools utilized by the mastering community are equalizers and limiters[2]. The first can be categorized as a frequency modification tool, while the latter a dynamics range modification tool. Our application combines both processes. By combining several digital Butterworth band-pass filters with some spectral processing functions and gain modifications, we have packaged a 'prosumer'- mastering suite into one application.

The equalization curve of an audio file should be implemented such that all frequencies are perceived equally loudly at moderate to loud listening levels[2]. As human beings, we perceive loudness as varying with frequency, even if the electrical reference level for amplitude remains unchanged[2]. Certain frequencies within the audible spectrum require less (more) energy to appear equally as loud (quiet). The Fletcher Munson study done at Bell labs in the early 1930's has solidified this concept into what we know call the 'Equal Loudness Contours', shown below.

This study started by using a 1KHz tone at a fixed electrical reference level as a benchmark. Then a 2Khz tone at the same reference level was played and the level was either increased or decreased to match the perceived level of the 1KHz tone. This process was repeated for all frequencies within the audible range[2]. The conclusions of these experiments demonstrate the indispensable necessity of an amplitude based frequency modification tool to ensure all frequencies are perceived as equally loud.

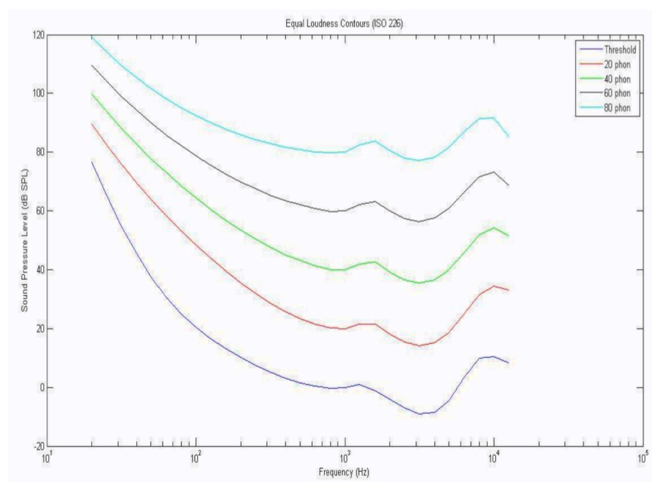


Figure 1: Equal Loudness Contours (ISO 226) [1]

The need for a stereo limiter arises after the spectral power of certain frequency bands has been modified. The possibility of overloading the channel capacity needs to be eliminated to avoid signal clipping and distortion. It is the last processing stage across the entire mix and ensures our audio signal never 'goes into the red.' The compression ratio of a brick wall mastering limiter is set to (infinity:1), which means that as soon as the audio signal hits the ceiling, it will never go beyond that level. A visual portrayal of this concept can be seen in figure 2.

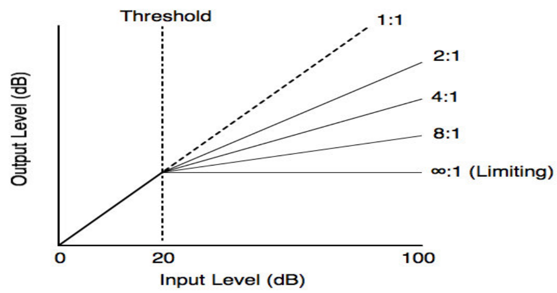


Figure 2: Brick Wall Limiting

2. MATLAB IMPLEMENTATION

2.1. Equalization

Implementation of the equalizer is based on thirty-one instantiations of a linear recursive second order biquadratic filter. Butterworth filters were chosen due to their "maximally flat" response (i.e. the pass band is designed to have a frequency response which is as close to flat as mathematically possible from DC up to Fc with no ripples), and their linear phase response[3]. A comparative depiction of typical frequency responses for common filters can be seen below in figure 3. Note the pass-band ripple found in the Chebyshev filter and the dramatic ripple in the Elliptic filter. The center frequencies of each individual filter are matched to the standard ISO for 31 band equalizers[3].

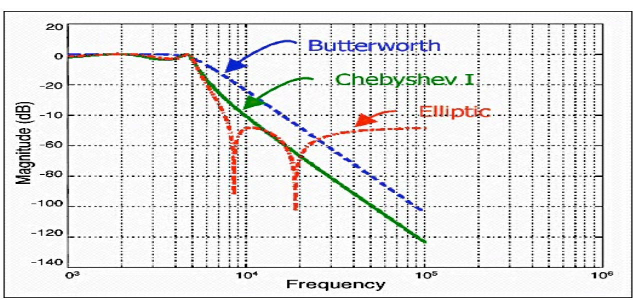
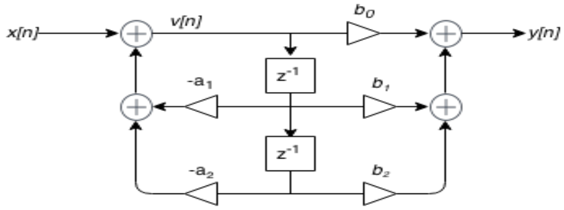


Figure 3: Frequency Responses of Common Filters

Second order filters proved to be the best tradeoff between roll-off attenuation, phase response, and pole/zero coefficient quantization error sensitivity[4]. The transfer function derivations for our Butterworth filters are based on Direct Form II. A flow chart and difference equation for DF-II are shown in figure 4 below.



$$v[n] = x[n] - a_1 v[n-1] - a_2 v[n-2]$$

$$y[n] = b_0 v[n] + b_1 v[n-1] + b_2 v[n-2]$$

Figure 4: DF-II Flow Chart / Difference Equations

2.2. Dynamic Range Limiting

The specific purpose for our limiter is to grant the user control over the highest signal peaks that might lead to distortion, but otherwise effect the dynamic range as little as possible.

This is achieved by applying an infinite ratio curve above the limiting threshold LT (i.e. a gain factor of 0dB is applied if the input signal remains below LT and a gain factor of LT minus the input signal is applied if the signal exceeds the threshold)[5]. In our instantiation of the limiter, LT is set to 1. A flow diagram of the limiter is seen in figure 5, below.

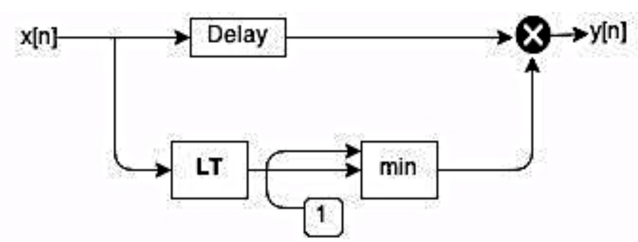


Figure 5: Flow Diagram of the Limiter

2.3. Pre-Set Adjustments (ISO 226)

To modify the signal appropriately, the power spectral density of the signal is measured and compared against the ISO 226 curve. If a preset equalization curve is employed by the user, the input signal is modified to reflect the PSD of the pertaining curve and then output as an equalized / limited version of the input.

2.4. Graphical User Interface

The GUI was designed to increase ease of use and functionality. From the GUI, the user can customize their desired equalization curve (each of the 31 bands has a corresponding gain slider), display spectrogram i/o plots, generate, playback, and save processed files, or apply preset curves. Several 'help' features (e.g. tips displayed by hovering the mouse over certain buttons) enable the user to grasp the overall functionality and utilize the synergy of the application and GUI without a priori knowledge of equalization or compression.

3. AUDIO MASTERING EMULATOR

A simplification of the application's signal flow is visualized in figure 6 below. The user interacts with the application via the GUI to customize their desired equalization curve. Optionally, the user can visualize the spectrogram of the input signal as well as the modified output signal. The modified file is then output from the application and saved to disk.

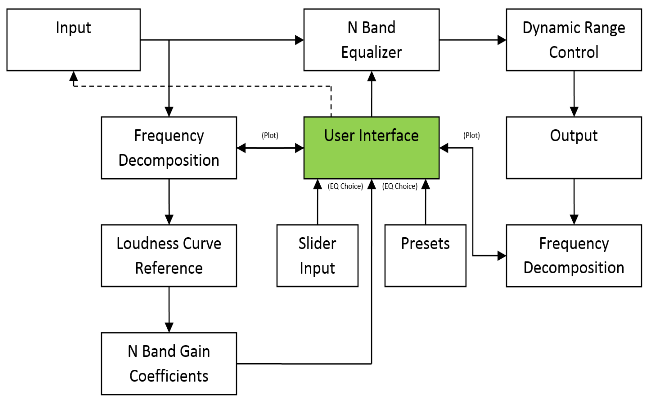


Figure 6: Flow Diagram of Application

White noise contains every audible frequency, each with equal power (i.e. a constant power spectral density). On the other hand, pink noise contains every audible frequency, but the power contained in each octave, rather than each frequency, is equal (i.e. the power spectral density is inversely proportional to the frequency)[2]. The efficacy of the audio mastering emulator can be demonstrated by processing both pink and white noise.

Following are i/o plots of white and pink noise, both processed by the application. As can be seen from the plots, the application has successfully modified the power spectral densities of the input signals to reflect that of the ISO 226 curve.

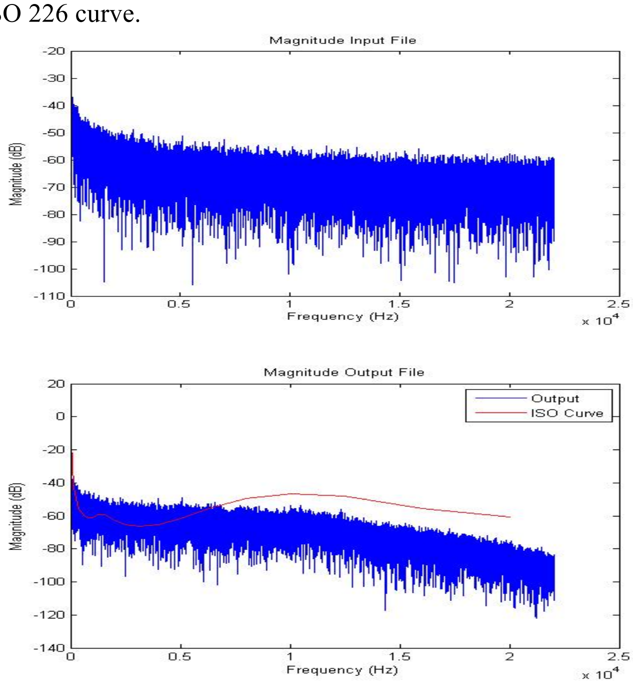


Figure 7: i/o Plots of Pink Noise

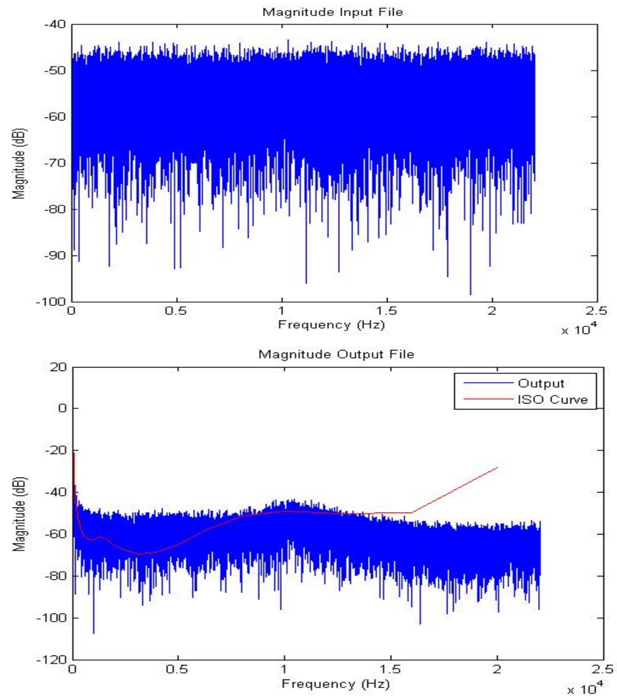


Figure 8: i/o Plots of White Noise

4. CONCLUSIONS / FUTURE CONSIDERATIONS

The Audio Mastering Emulator has been shown to provide consistently predictable and accurate results. The application is capable of processing the most widely used file formats and the GUI has been to shown to be extremely user friendly.

Presently, the ISO 226 curves are only effectively defined up to 12.5Khz[1]. This is primarily due to the fact that dramatic variations in listening experiments exist. These variations have prevented an accepted representative figure from being proposed. To compensate for this, basic polynomial extrapolation has been implemented. Future considerations would take the 12.5Khz to 20Khz bandwidth into account by incorporating and utilizing more precision within the polynomial extrapolation.

In addition, more presets could be included to deal with the ever-increasing styles and genres of music. The variety of acceptable file formats might also be expanded to include more obscure extensions, such as .flac or .ape. One remaining possibility would be to implement a variability feature pertaining to the Q of each equalization band.

5. REFERENCES

[1] Acoustics, Normal Equal Loudness Contours. (n.d.). Retrieved April 15,2015, from http://www.iso.org/iso/catalogue_detail.htm

[2] Katz, R. (2007). *Mastering Audio: The Art and The Science*(2nd ed.). Amsterdam: Elsevier/Focal Press.

[3] Rumsey, F., & McCormick, T. (2006). *Sound and Recording, An Introduction*(5th ed.). Oxford: Focal Press.

[4] Spectral Audio Signal Processing. (n.d.). Retrieved April 17, 15, from http://www.dsprelated.com/dspbooks/sasp.html.

[5] Zolzer, U. (2011). *DAFX: Digital Audio Effects, Second Edition* (2nd ed.). Chichester: John Wiley & Sons.

[6] 41 Complete GUI Examples - File Exchange - MATLAB Central. (n.d.). Retrieved April 23, 2015, from http://www.mathworks.com/matlabcentral/fileexchange/24861-41-complete-gui-examples