AES Number: 93753
Student Design – Undergraduate
Use of Multiband Dynamic Quantization as Audio Effect

Use of Multiband Dynamic Quantization as Audio Effect

***Abstract*—Most current uses of bit crushing in audio apply the effect statically and independent of amplitude and frequency. This paper outlines an effect that changes bit depth dynamically with frequency, as well as driving four bands of the effect separately, each with different parameters. The project was prototyped in MATLAB and includes a graphical user interface designed in GUIDE to allow easily viewable results and tweaking of parameters to facilitate the prototyping process.**

Keywords—bit crushing; quantization; multi-band; non-linear

I. Introduction

Quantization is a necessary step in the conversion from analog to digital, but also can have uses in digital audio effects. While standard CD quality audio is 16 bits, quantization to 8 bits can audibly change the characteristics of the sound, adding overtones and adding either warmth or harshness. However, most bit crushers quantize the signal uniformly, with the only parameters being the number of bits, sampling rate, and possibly drive and filtering at the output. This paper will examine the frequency and time domain results of using multiple bit depth quantization depending on both frequency content and amplitude. The amplitude processing will be performed on the instantaneous amplitude, like that of a distortion effect as opposed to a dynamics processor. The model was designed and tested in MATLAB, using GUIDE to tweak each of the parameters and easily view results.

II. Dynamic Bit Crusher

The first block that was designed and tested in MATLAB was the Dynamic Bit Crusher, which is a variation on the traditional bit crushing function displayed below.

$$y(t) = \frac{\lfloor 2^{B-1} x(t) \rceil}{2^{B-1}} \qquad (1)$$

B: Bit Depth

The effect takes three input amplitude thresholds and four bit depths as its parameters, allowing the bit depth to vary depending on the amplitude of the input, as seen in the following equation.

$$y(t) = \begin{cases} \frac{\lfloor 2^{B_1-1}|x(t)| \rceil}{2^{B_1-1}}, if x(t) \leq A_1 \\ \frac{\lfloor 2^{B_2-1}|x(t)| \rceil}{2^{B_2-1}}, if A_1 \leq x(t) < A_2 \\ \frac{\lfloor 2^{B_3-1}|x(t)| \rceil}{2^{B_3-1}}, if A_2 \leq x(t) < A_3 \\ \frac{\lfloor 2^{B_4-1}|x(t)| \rceil}{2^{B_1 4-1}}, if x(t) \leq A_3 \end{cases} \qquad (2)$$

$T_{1\text{-}3}$: thresholds
$B_{1\text{-}4}$: Bit Depths

III. Implementation of Knee

Although harsh discontinuities in the output signal are the foundation for the harmonics created by the bit crushing effect, these discontinuities are relatively small when only one bit depth is used. However, as can be seen from the response plot (fig1), instantaneously changing the bit depth can cause extremely harsh discontinuities in the output signal, which may be undesirable. Because of this, a knee parameter was added, similar to that of a compressor, which crossfades between outputs of different bit depths as the amplitude changes.

In order to implement a knee to crossfade between each of the bit depths, the Dynamic Bit Crusher function must be significantly altered. Instead of a set of piecewise functions for each bit depth, the four bit crushing functions have support of the entire interval 0 to 1 and are multiplied by a gain function. The four gain functions are based on the Hanning Window.

$$w(x(t)) = \frac{1 - cos(\frac{2\pi x(t)}{k})}{2} \qquad (3)$$

Each of the gain functions consists of an upward half of a Hanning window centered around the lower threshold, downward Hanning window centered around the upper threshold, 1 within the two thresholds, and 0 elsewhere, seen in the following equations:

$$g_1(|x(t))| = \begin{cases} 1, if\ |x(t)| \leq T_1 - .5k) \\ w(|x(t)| + .5k), if T_1 - .5k < |x(t)| \leq T_1 + .5k \\ 0, if T_1 + .5k < |x(t)| \end{cases} \qquad (4)$$

The gain function for an amplitude band with only one rising or falling edge is made of three piecewise functions, as seen above. A gain function for a band with two rising and falling edges has five piecewise functions and can be seen in the equation below.

$$g_2(|x(t)|) = \begin{cases} 0, if\ |x(t)| \leq T_1 - .5k \\ w(|x(t)|), if T_1 - .5k < |x(t)| \leq T_1 + .5k \\ 1, if T_1 + .5k < |x(t)| \leq T_2 - .5k) \\ w(|x(t)| + .5k), if T_2 - .5k < |x(t)| \leq T_2 + .5k \\ 0, if T_2 + .5k < |x(t)| \end{cases} \quad (5)$$

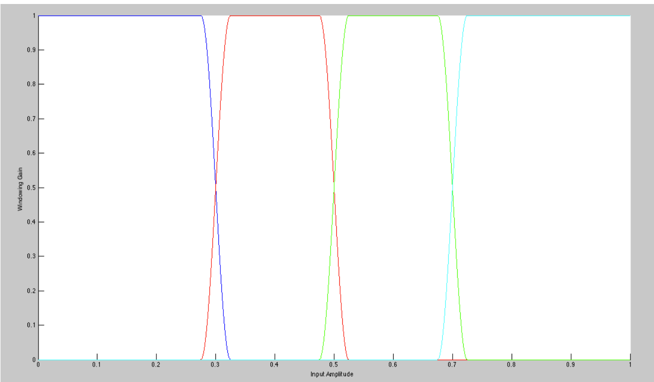
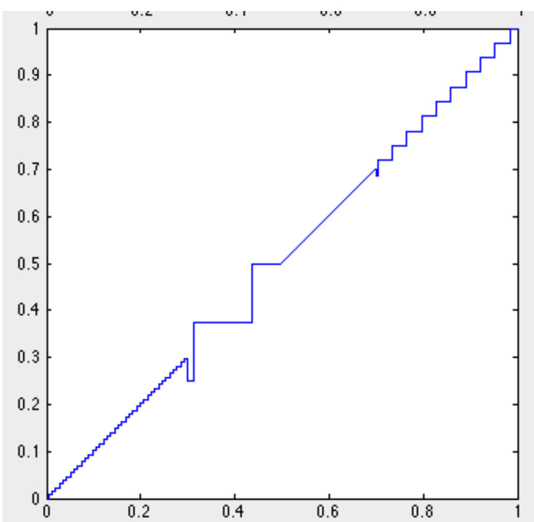


Fig.1. The four gain functions superimposed over each other can be seen in Fig. 1.

By multiplying each of the gains which each of the bit crushed signals from (1), the following equation is obtained.

$$y(t) = g_1(x(t)) * y_1(t) + g_2(x(t)) * y_2(t) + g_3(x(t)) * y_3(t) + g_4(x(t)) * y_4(t) \quad (6)$$



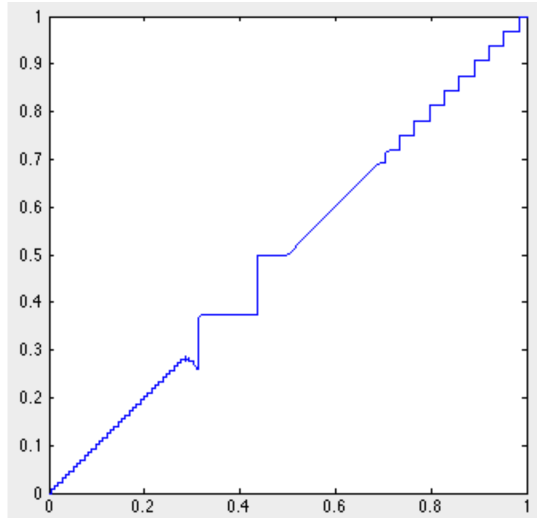


Fig. 2. A. Amplitude respose without knee B. Amplitude Response with knee

In Figure 2, a response plot for thresholds of .3,.5, and .7 and bit depths of 12, 4,16, and 8 can be seen. A comparison of the response plot without knee (Fig. 2A) and with knee (Fig. 3B), shows the redution of harsh discontinuities as the bit depth changes. Although there is still a brief section of the plot where output amplitude decreases as input amplitude increases, this can be eliminated by tweaking the thresholds. This process is faciliated by the response plot in GUIDE being updated in real time as thresholds are changed.

IV. Multiband Effects

As the edge of a square wave becomes steeper, the harmonics of the signal decay slower. In a bit crushed signal, the edges of the signal are 90 degrees and as a result, harmonics decay slowly as frequency increases. Because of this, processing the signal in multiple bands will have little effect on the Fourier Transform of the output, once all the bands are summed back together. The motivation for processing the signal in different bands is because the bit crushing is done dynamically, different frequency bands of the input can drive the signal at different times and at different bit depths. The crossover filters are created using digital 2nd order Butterworth high pass, low pass, and band pass filters [1] with cutoff frequencies set equal so that the overall gain is unity and the overall frequency response is flat.

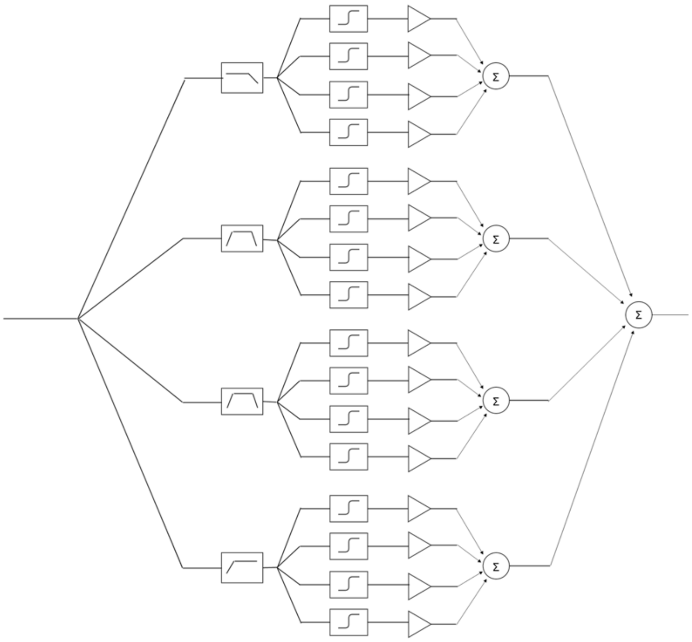


Fig. 3. Block diagram for entire effect.

Each filter of the crossover feeds the input of a dynamic bit crusher block whose parameters can be set independently of each other. The overall block diagram can be seen in Fig. 3.

V. Test Results

Because of the effect's non-linearity and dependence on amplitude, frequency response plots will give little information on the actual performance of the plugin. Like the amplitude dependent distortion of tube amplifiers, the quality of the effect is subjective in nature and cannot be inferred based on transient and frequency responses alone. What the effect and its interface allow is a wide range of configuration that can potentially lead to many sonically interesting results.

The example output from Fig. 4 occurs when the thresholds are evenly spaced and for low amplitudes, the bit depth is high and at high amplitudes, the bit depth is low. This leads to increased distortion at higher amplitudes, simulating that the system is being driven to distortion, but retaining the harsh, digital sounding distortion of the bit crusher effect.

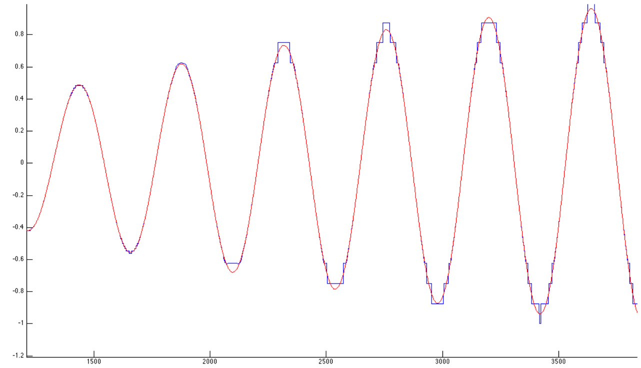


Fig. 4. Input (red) and output (blue) with low bit depth at high amplitude levels.

The example in Fig. 5 retains a high bit depth at high amplitudes and bit crushes the signal to 1 bit at a threshold of .1. This retains the original signal, but creates a flat section between the halves of the sine waves, similar to that of crossover distortion in a Class B Tube Amplifier. This type of distortion adds to the 5^{th} harmonic of the signal, creating a crunchy sound [2]. However, the knee smooths the transition from the 16-bit region to the 1 bit region, so that the harmonics are only generated from the simulated crossover distortion and not from the discontinuity between bit depths.

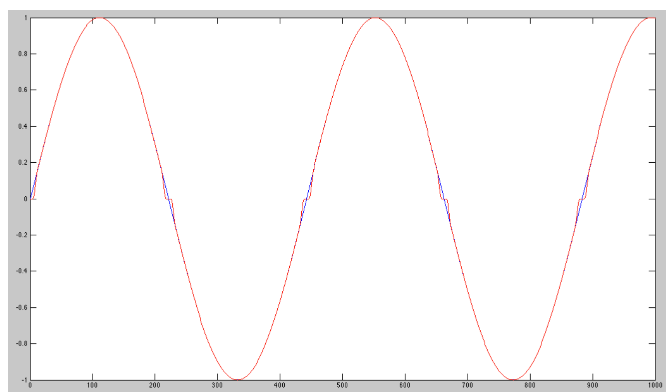


Fig. 5. Input (blue) and output (red) with high bit depth except at thresholds below .1, where bit depth is 1.

Along with these two examples, there are many more possible configurations of the effect, allowing harsh sounding, artificial digital effects and emulation of analog distortion. There is the potential from interesting effects involving driving of the different bands of the effect by using multi-band amplitude modulated signals as the input.

VI. Graphical User Interface

Because the effect is in the prototyping phase and not the final product phase, the user interface is designed to allow optimal experimentation and immediate data to analyze. The amplitude response plot is updated as parameters are changed, and when the "play" button is pressed, the output audio is played and the transient and frequency response graphs are both updated.

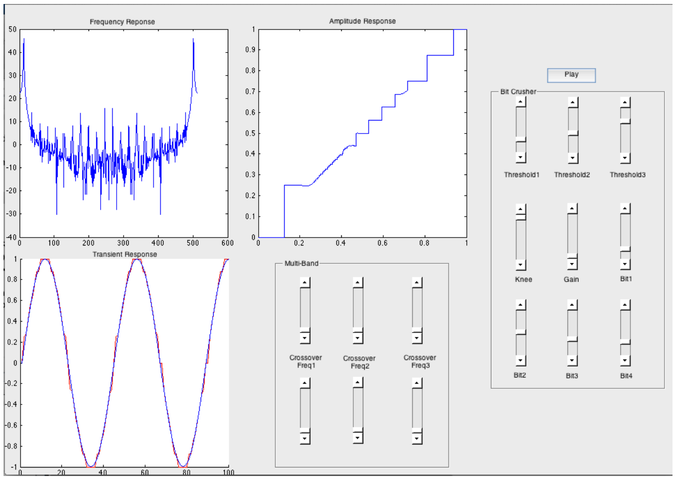


Fig. 6. Graphical User Interface for Effect including Frequency Response, Amplitude Response, and Transient Response.

CONCLUSION

The effect discussed in this paper allows for a wide range of possible configurations through a combination of dynamically setting bit depth based on amplitude of the input signal, as well as processing the signal in multiple bands. The range of effects possible range from artificial sounding digital distortion to modeling of analog drive and distortion without the harsh discontinuities and high levels of high frequency harmonics found in traditional bit crusher effects. Experimentation with parameters is made easy with a user interface designed in GUIDE, and a wide array of final audio effect plugins is possible using this prototype.

REFERENCES

[1] Pirkle, Will "Designing Audio Effect Plug-Ins in C++" Burlington, MA: Focal Press, 2013.

[2] Pirkle, Will "Analog Audio Electronics" 1997-2010