University of North Carolina Asheville
Journal of Undergraduate Research
Asheville, North Carolina
December 2021

# Harmonic Modulator Envelope filters: Expanding Acoustic Instruments Through Digital synthesis

Thomas Benton Goodall
Computer Science Department
The University of North Carolina Asheville
One University Heights
Asheville, North Carolina 28804

Faculty Advisor: Brian Drawert

## Abstract

To many music listeners, electronic instruments such as synthesizers or virtual instruments can be easily distin- guished from physical instruments due a synthetic or inorganic quality of their timbre. A seasoned musician that has developed a technique of sound and articulation with a physical instrument would find it challenging to obtain the same kind of real-time physical control over the timbre of a virtual instrument or synthesizer. To address this issue, we have developed the Harmonic Modulator Envelope filters which combines the dynamics of a physical instrument's timbre with audio synthesized from a virtual instrument. We demonstrate the power of this tool by extracting features of the timbre of a melody that was played on a guitar, and re-combining those features with the same melody rendered on a virtual instrument.

## 1. Introduction

Since the mid 20th century, synthesizers have served as a ubiquitous tool in music production and sound design for creating a wide range of timbres or characters of sound. In this time digital signal processing (DSP) engineers have developed many musical synthesis methods such as FM, additive, subtractive, and wavetable synthesis in order to expand musical timbre. While many of these synthesis techniques can achieve very dynamic timbres, to many listeners and musicians these sounds can seem contrived or lacking in the variable qualities present in physical sources of sound. Guitar players, for example, can physically control the natural dynamics of their instrument's timbre in the various subtle ways they can pluck a guitar string. It is currently impossible for a guitar player to have this same level of real time, physical control over the timbre of a synthesizer. This paper will be examining the physics that contribute to this natural variety of timbre in the sound of a guitar as well as computational methods for isolating these dynamic qualities from recordings and utilizing them in musical synthesis.

Historically, the Vocoder has served as the general solution to this problem for music producers. The Vocoder, originally developed by Homer Dudley of Bell Laboratories, utilizes a modulator and a carrier signal. The frequency spectrum of the modulator signal is used to modulate, or filter the magnitudes of the frequency spectrum of the carrier signal. Typically the modulator signal is an acoustic source such as a human voice, and the carrier signal is a synthesized sound. In our system the modulator signal would correspond to a recording of a melody played on a guitar and the carrier signal would be a precise replica of the same melody played on a synthesizer.

While the Vocoder modulates discrete frequency bands of the carrier signal's frequency spectrum it is now possible to obtain and modulate the precise harmonics present in a signal using spectral modeling synthesis (SMS) techniques. This paper will present a system that extracts features from measured sinusoidal harmonics in a recording of a plucked guitar string to produce filters that modulate the corresponding overtones that were measured in audio produced by a virtual instrument. We call these Harmonic Modulator Envelope filters (HMEs). We produced three types of HMEs to model three distinct features of the guitar's timbre. Finally, we produced combinations of these signals by fine tuning the presence of each of these timbral artifacts of the guitar in the carrier signal. As seen in fig. 1 we achieve this by converting a recording of a monophonic guitar melody to midi, resynthesizing the melody on a virtual instrument, summing the HMEs that were generated using the features extracted from the guitar recording with the sinusoidal harmonic values obtained from the synthesizer, and replacing the harmonics in the guitar signal with these modulated synthesizer harmonics.

Other contemporary attempts to model and combine the dynamic qualities of natural sounds have been explored by Xavier Serra, who has previously investigated this kind of application of SMS. In Serra's paper "Voice Morphing

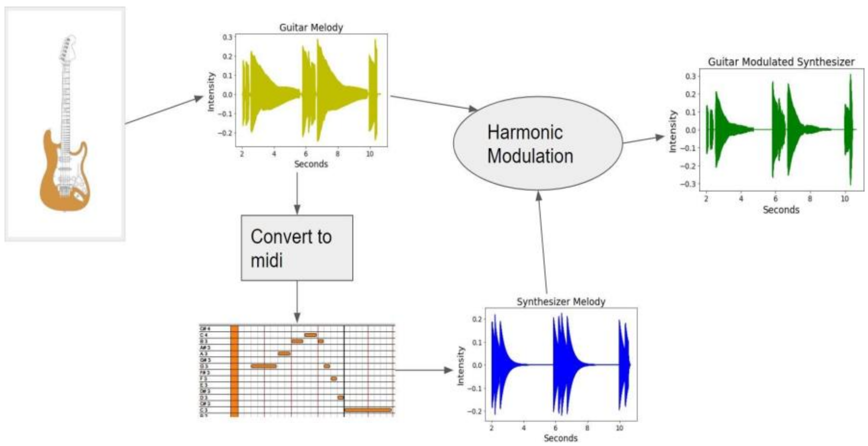


Figure 1: An overview of the HME system.

The audio obtained from the guitar (yellow waveform) is converted to midi data and resynthesized using a virtual instrument. Both the synthesized signal (blue waveform) and guitar recording are passed through the harmonic modulation system wherein the HME filters are rendered using features extracted from the guitar signal and summed with the synthesizer harmonics. These modulated harmonics replace the original guitar harmonics. The resultant signal (green waveform) is a nuanced combination of both the guitar and synthesizer signals.

System for Impersonating in Karaoke Applications"[3] he presents a system that morphs between a live input of a karaoke singer and a spectral model of the original singer's voice in real-time. The Google research group, Magenta, has also developed techniques for combining musical timbres using neural networks. One of their seminal products the Nsynth[5] used recurrent neural networks that were trained on raw sequential audio data of various musical sounds. The subtle characteristics of these musical timbres could then be combined to create new sounds. One of Magenta's new models for musical synthesis is the GANSynth[4], which uses a Progressive General Adversarial Network architecture to train the generator and discriminator starting from a low resolution to higher resolution on raw audio. While these techniques are exciting they don't allow the user to isolate and combine specific elements of a sound's timbre. Concerning the specific task of modeling the dynamics of a guitar's timbre and reproducing it in a carrier signal, we require an understanding of the attributes that compose a guitar's timbre that are subject to change.

In the following sections we present the theoretical background of both the existence and relevance of the three dynamic attributes of a guitar's timbre that we are concerned with modeling and resynthesizing in the carrier signal as well as the spectral modeling techniques we used to measure these attributes. "Vocoders and Cross-Synthesis" presents the system and algorithms that enable Vocoders and Cross-Synthesis techniques to blend sounds and provides a theo- retical foundation for the HME system. "Overview of the Guitar's Timbre" contains a brief background of the physics behind the three attributes of the guitar's timbre we are concerned with modeling as well as the relevance of these attributes in the design of a well known physical modeling algorithm of plucked strings known as the Karplus Strong algorithm[13]. "Spectral Modeling Synthesis (SMS)" describes the process of obtaining models of an instrument's timbre from sinusoidal data obtained from a series of signal processing techniques known as "Spectral Modeling Syn- thesis"(SMS). "Harmonic Modulator Envelope Filters (HME)" details the process of creating Harmonic Modulation Envelope filters from a spectral model of a melody played on a guitar in order to resynthesize the dynamic qualities of it's timbre in a corresponding synthesized version of the melody. "Results" discusses the challenges encountered when implementing the HME system, the aesthetic effects of each filter and how to optimize system parameters.

## 2. Background

In order to understand the architecture of the HME system, it is useful to start with an understanding of the Vocoder[6] and Cross-Synthesis[17]. As previously mentioned, both the Vocoder and Cross-Synthesis have been utilized by music producers and musicians to reproduce the timbral dynamics of a recorded sound in a synthesized signal. Understanding this relationship between the modulator and carrier signals is a useful analog for understanding how the HME filters will replicate the natural dynamics of the guitar's sound in a synthesized signal.

### 2.1 Vocoders and Cross-Synthesis

Originally developed by Homer Dudley of Bell Laboratories[6], the Vocoder utilized the frequency spectrum of a modulator and carrier signal to encode and resynthesize a human voice. The modulator signal, typically the frequency bands of a recorded voice, would modulate or control the magnitude of the frequency bands of the carrier signal, a harmonic synthesized sound (typically pulse train or sawtooth waveforms). Cross-Synthesis is a similar technique wherein an envelope of the modulator's spectrum is used to modulate the spectrum of the carrier signal. We utilized the framework of the Vocoder when designing the HME system, thus understanding its components is key to understanding our harmonic modulator system. In our case, a virtual instrument acts as a carrier signal and a recording of a plucked string is used as the modulator.

In order for Cross-Synthesis or a Vocoder to work, it is necessary to have a robust and reliable algorithm for transforming both the modulator and carrier signals from the time to the frequency domain. The Discrete Fourier Transform (DFT)[2] is a useful algorithm for this process. In its common incarnation as seen in eq. 1 and eq. 2 $X(k)$ represents the amplitude of frequency bin $k$ of angular frequency $w$ for $N$ time domain samples.

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-i2Nkn} \tag{1}$$

$$= \sum_{n=0}^{N-1} x(n) \cdot [\cos(\frac{2\pi}{N}kn) - i \cdot \sin(\frac{2\pi}{N})] \tag{2}$$

$X(k)$ returns a complex number in the form $(a, ib)$. The magnitude is given by $\sqrt{a^2 + b^2}$ and the phase by $\sqrt{\frac{b}{a}}$. As seen in eq. 3 the DFT can be inverted to obtain a time domain waveform $X(n)$ from a given $X(k)$ spectrum. The Fast Fourier Transform (FFT)[2] is a method of computing the DFT in an efficient manner by exploiting the symmetry of sizes of $N$ that are powers of 2. This efficient implementation of the algorithm has made the FFT very popular in signal processing

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \cdot e^{\frac{\pm j2\pi nk}{N}} \tag{3}$$

#### *2.1.1 short time fourier transform (stft)*

In order to model the spectrum of a sound that changes with time it is not effective to simply compute a series of FFTs.

In order to improve the time resolution of the frequency domain data, Cross-Synthesis techniques will employ the Short Time Fourier Transform (STFT)[17] as a method of computing the FFT over the propagation of a sound. Serra's[17] depiction of the STFT is seen in eq. 4 where $l$ represents the frame number and $H$ represents the hop size or number samples to shift the frame by. $w(n)$ represents the windowing function that is applied to the time domain signal to reduce spectral leakage. The correct hop size and windowing function can significantly improve the corresponding time and frequency resolution of our model.

$$X_l(k) = \sum_{n=0}^{N-1} w(n) \cdot x(n + lH) \cdot e^{-\frac{i2\pi}{N}kn} \tag{4}$$

### 2.1.2 windowing functions

In order to avoid spectral leakage between frames of the STFT a windowing function is multiplied with the time domain samples $x(n + lH)$. These windows are symmetrical functions whose frequency spectrum resembles a sinc function[17]. They are multiplied with the time domain signal before the FFT is computed. Choosing a window will determine a degree of trade-off between time and frequency resolution. Two aspects of a windowing function that we are concerned with are its bandwidth and the amplitude of it's highest sidelobe. In selecting a windowing function there is typically a tradeoff between bandwidth and sidelobe suppression. Both a wide bandwidth and lack of sidelobe suppression can lead to spectral leaking or non-zero values in the frequency spectrum that can be mistaken for harmonics. For our model we utilized a Blackman-Harris window[10] as its highest level sidelobes are below the noise floor.

### 2.1.3 window size

When selecting a size of samples $M$ to pass through a windowed FFT, it is important to establish the resolution of frequency that is desired. This resolution is measured by the delta of two frequencies your spectrum is expected to distinguish. Since our system tracks harmonics of a musical note, the frequency resolution of our STFT should be able to distinguish multiples of the lowest fundamental frequency on a guitar, which corresponds to an open e string at 82 hz. Serra[17] demonstrates how the window size of an STFT should correspond to the product of the bandwidth of the window and the period of the delta of two frequencies your spectrum is expected to distinguish. The period of 82 hz in samples is 538 with a sampling frequency of 44100 hz. The product of the 8 bin Blackman-Harris window we used and 538 samples comes out to a window size of 4304. Serra[17] also recommends that the window size be odd so as to center the data in the middle of the windowing function. This practice maintains the accuracy and continuity of phase values. For this reason we used a window size of 4305 samples

### 2.1.4 zero padding

Since the FFT requires a sample size of a power of two we cannot simply use our calculated window size of 4305. To mitigate this, Serra[17] recommends taking a larger FFT size and filling the remaining sample bins with zeros. With a window size of 4305 we took an FFT size of 8192. This process is known as zero padding and it improves the resolution of our spectrum as it results in interpolation of values in the frequency domain.

### 2.1.5 hop size

The hop size of our STFT will determine how many samples our window will step through before computing each FFT. This value is also referred to as the overlap factor which describes how many samples of the previous FFT are reused or overlapped in the next FFT. For the Blackman-Harris window, Serra[17] recommends a hop size of (M/2)/j where M is the window size and j is a positive integer. For our window size of 4305 we used a j value of 21 which rounds to a hop size of 103. This produced a very smooth, time-frequency resolution.

### *2.16 cross-synthesis*

In Serra's example of Cross-Synthesis[17], the spectral envelope or approximated spectral shape of each frame of the modulator's STFT is impressed on each frame of the carrier's STFT. The spectral envelope of the $l^{th}$ frame of the modulator's STFT is approximated by an interpolated line segment approximation of the spectrum. This envelope is then normalized to have an average amplitude of 0dB. This envelope is then summed with the carrier signal's magnitude spectrum as seen in eq. 5 where $\bar{E}_l(k)$ represents the approximated modulator envelope, $A_l(k)$ is the carrier's magnitude spectrum and $B_l(k)$ is the resultant modulated signal.

$$B_l(k) = A_l(k) + \bar{E}_l(k) \tag{5}$$

## 2.2 Overview Of The Guitar's Timbre

While the Vocoder and previous Cross-Synthesis methods have been used to great effect in blending musical timbres, our goal with the HME filters is to replicate specific aspects of the guitar's timbre in the carrier signal as opposed to simply replicating the magnitude spectrum. In order to understand these aspects of the guitar's timbre we must understand the previous attempts to model it. This section will examine three dynamic features of the amplitudes of the guitar's harmonics: spectral onset conditions, harmonic decay and beat patterns. These are the three key features that our HME system models and replicates in the carrier signal.

### *2.2.1 onset and initial harmonic conditions*

Much of the variety in the timbre of a plucked string can be attributed to the initial spectral conditions of a note. The magnitude of each overtone at the onset of a note can produce different textures that many musicians might describe as "bright" or "dark". We can trace back models that explain this dynamic of the timbre of a plucked string to the 18th century with D'Alembert's discovery of the wave equation[19]:

$$\frac{\delta^2 u}{2} = c^2 \frac{\delta^2 u}{\delta x^2} \tag{6}$$

$$c = \sqrt{\frac{T}{\mu}} \tag{7}$$

This second order partial differential equation describes the 1 dimensional motion of a transverse wave on a string of length $l$ that is fixed at both ends $(x = 0$ and $x = l)$.. The common form of this equation is shown in eq. 6 wherein $\mu$ , the displacement of the string from equilibrium is related to the position $x$ at time $t$. As shown in eq. 7 this speed $c$ of the transverse wave is equal to the square root of the string's tension $T$ over it's linear mass density. Given the constraints, we can solve this equation as a sum of complex sinusoids[19] as seen in eq. 8

$$y(x,t) = \sum_{n=1}^{\infty} (A_n \cos(w_n t) + B_n \sin \frac{n\pi x}{l}) \tag{8}$$

where $n$ is the order, $A_n$ and $B_n$ are constant coefficients that can be determined from the shape and velocity of the string for any given time $t$ and $w_n = nw_0 = n(2\pi f_0)$[19]. These sinusoids correspond to the standing waves of each harmonic that a plucked string produces as a result of the interference of the two transverse waves that propagate to the left and right of the plucking point or initial point of maximum displacement of the string. Any physical phenomenon that explains the harmonic motion of these normal modes will be essential to our understanding of the guitar's timbre. The initial conditions of a plucked string have tremendous consequences on the propagation of its harmonics. For example, the plucking point can result in exciting different harmonics. According to Traube[19] different harmonic standing waves are excited based on the proximity of the plucking point to the harmonic's antinode. Consequently, harmonics cannot be initiated at the nodal point of the corresponding standing wave[12]. As seen in fig. 2, more of the higher harmonics are excited with a plucking position close to the guitar's bridge as opposed to $L/2$ wherein $L$ corresponds to the length of the string.

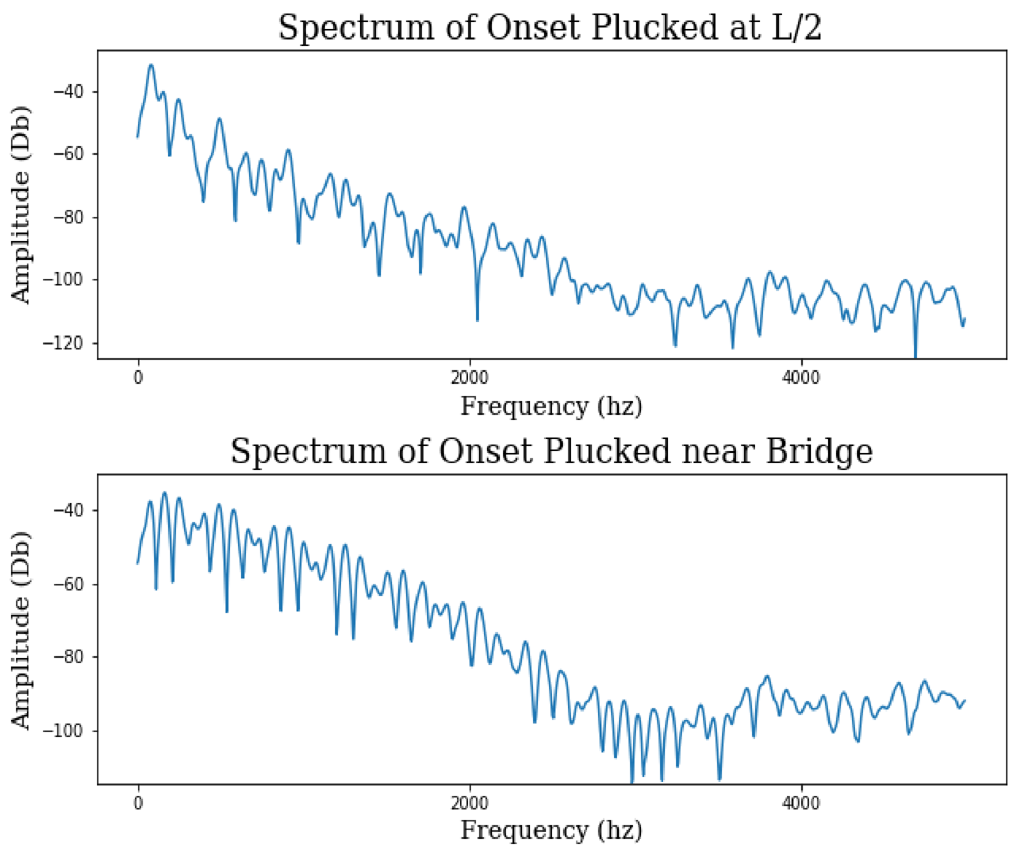


Figure 2: Magnitude spectrum at the onset of two notes played on the guitar.

Graph A depicts the spectrum produced with a plucking position at $L/2$ wherein $L$ is the length of the string and Graph B depicts the spectrum produced with a plucking position near the bridge. This demonstrates the effect of plucking position on the initial harmonic conditions of a note since more harmonic energy is excited in the 500 to 2000 hz range in the note plucked near the bridge.

The initial velocity of a pluck will also change the initial magnitudes of the string's harmonics. Typically a higher velocity pluck will correspond to higher amplitudes produced in the higher harmonic frequencies. Smith[11] says that this is due to non-linearities becoming more important at higher velocities.

One of the earliest and more commercially successful attempts at computationally modeling the sound of a plucked string was the Karplus Strong algorithm[13]. This field of study, related to virtual simulations of physical instruments is known as Physical Modeling. This simple algorithm provides insight into the defining characteristics of plucked strings. Their system utilized an 8 bit wavetable to generate random numbers which was effectively a burst of noise that would be routed through a delay line with a delay time short enough to produce pitches. The last effect used a two point averaging of the last delayed sample and the previous delayed sample to create a low pass filtering effect. This algorithm produces a note that initially contains a lot of harmonic content but decays to a nearly pure tone with higher harmonics decaying more quickly.

In 1983 Julius Smith and David Jaffe engineered an extension of the Karplus Strong algorithm to create a more realistic sound of a plucked string[11]. Among their many modifications there were two filters that simulated the nuances of the initial plucking conditions of a string. They simulated the effect of the plucking position with a comb filter and the difference between up and down picks with a low pass filter applied for "up" picks. They also detail a "Dynamics Filter", a low pass filter that is intended to replicate the effect of higher harmonics becoming more pronounced with higher initial velocities. This is achieved by modulating the filter according to the initial velocity of a note. This speaks to the relevance and motivation of modeling the spectral onset conditions of a string in our harmonic modulation system.

### *2.2.2 harmonic decay*

The rate at which the amplitude of the harmonics decay also contributes to many of the artifacts of a plucked string's timbre. Namely, the increased harmonic decay rate at higher frequencies. This relationship of frequency to the decay rate of a string's overtones is a key feature of the guitar's timbre and is thus of great interest and relevance to our harmonic modulation system. It is noted in the Karplus Strong paper that without the decay algorithm the resulting sound is similar to a reed organ, however, with decay, the sound resembles a plucked string or guitar[13].

Harmonic decay can be explained by the motion of a plucked string being impeded by damping forces that eventually bring it back to rest. Fletcher[8] describes three sources of damping in plucked strings: internal damping, air damping and the transfer of energy to the body of the instrument through the nut and bridge. For metal strings like the ones we are modeling, the primary damping force is air damping[7][9]. Fletcher[7] references G.G. Stokes' proof of the viscous drag on a vibrating metal string producing a frequency-dependent exponential decay. Stoke's equation as seen in eq. 9 models the retardant force $R$ which is out of phase with the velocity $v$ that acts on a string of length $L$ and radius $r$ moving with frequency $\nu$[7]. The density of air is represented here by $\varrho_a$.

$$R \approx 2\pi^2 \varrho_a \nu r^2 L(\frac{\sqrt{2}}{M} + \frac{l}{2M^2})v \qquad (9)$$

Fletcher[7] also presents the equation for the value $M$, see eq. 10 where $\mu_a$ represents the viscosity of air.

$$M = \frac{r}{2}\sqrt{\frac{2\pi\nu}{\mu_a}} \qquad (10)$$

Fletcher[7] states that the validity for equation 9 holds only if $M > 0.3$. Our calculations for the values of $M$ for our 11 gauge guitar strings met this condition. Fletcher[7] also mentions that $R$ is proportional to $\nu$ so that the rate of energy loss varies as $v^2$, which is proportional to the kinetic energy. This is in part why the higher harmonics of a plucked string decay at a faster rate and why we expect an exponential decay for a simple oscillation at a single frequency[7].

### *2.2.3 beat patterns*

While our 2 dimensional model of a plucked string may suggest a clean, exponential decay of harmonics, this is not consistent with what we observe in the harmonic data of a real string. Instead there are periodic fluctuations in amplitude that accompany an exponential decay. Some of these fluctuations are due to beat patterns. According to Politzer[16], every mode of a plucked string is typically composed of two degenerate modes that are slightly coupled and similar enough in frequency to be heard as a single harmonic. The consequence of this slight difference in frequency and phase can result in beat patterns. These degenerate modes correspond to the possible vertical and horizontal range of motion for each standing wave harmonic[16].

## 2.3 Spectral Modeling Synthesis (SMS)

While we discussed the utility of Vocoders in their ability to blend timbres by modulating the discrete frequency bands of a carrier signal with the spectrum of a modulator signal, our HME system requires a method for obtaining and modulating the sinusoidal values of a signal's harmonics in order to more precisely reproduce the dynamic features of the guitar's timbre. Utilizing spectral modeling synthesis techniques developed by Julius Smith and Xavier Serra we are able to reliably detect precise amplitude, phase and frequency values of sinusoids in the frequency spectrum of a signal. We will use these methods to track the propagation of harmonics through the recording of a plucked guitar string. This data will then be used to generate the HMEs to modulate the overtones of our additive synthesizer.

### *2.3.1 peak detection and interpolation*

In a single FFT frame, sinusoidal harmonics will appear as peaks or discrete frequency bins with higher magnitude values than neighboring bins as seen in fig. 3. The SMS method of extracting precise sinusoidal values from these discrete frequency bins is outlined by Serra[17]. Quadratic interpolation is employed to gain more accurate frequency, amplitude and phase data for the sinusoids that are denoted by discrete peak values in the frequency spectrum. As seen in fig. 4 this method approximates a parabola with the two adjacent bins of a peak frequency bin. The peak of this parabola corresponds to the true peak of the sinusoid. Finding these precise sinusoidal values is what will enable us to process the harmonics of a signal separately from the residual or non-harmonic component of the sound.

The peak detection algorithm locates discrete peak bins in the magnitude spectrum by selecting bins with a higher magnitude than their left and right neighboring bins. In order to perform quadratic interpolation on these harmonic peaks, we must obtain the log decibel values of the peak bin referred to in eq. 12 as $\beta$, the left neighboring bin $\alpha$ see eq. 11, and the right neighboring bin $\gamma$ see eq. 13.

$$\alpha \stackrel{\text{def}}{=} 20log_{10}|X(k_\beta) - 1| \tag{11}$$

$$\beta \stackrel{\text{def}}{=} 20log_{10}|X(k_\beta)| \tag{12}$$

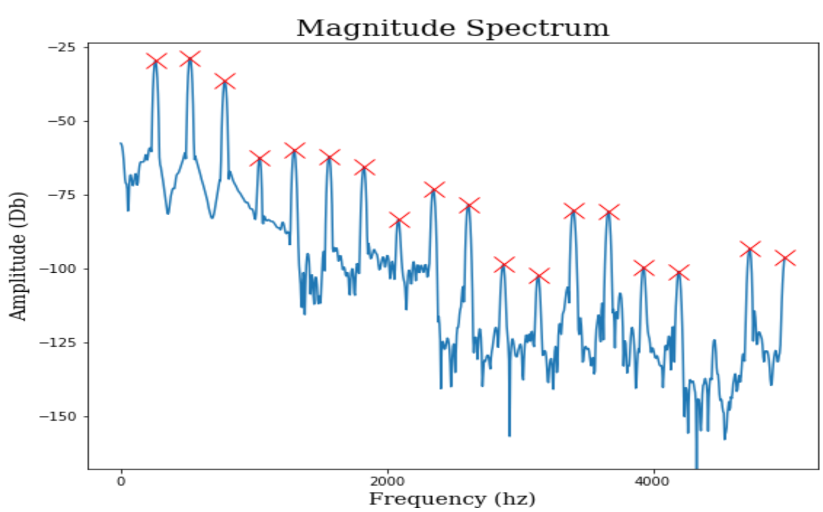


Figure 3: Depicts the magnitude spectrum of a frame of an STFT obtained from a plucked guitar string. The detected harmonic peaks are marked with an x.

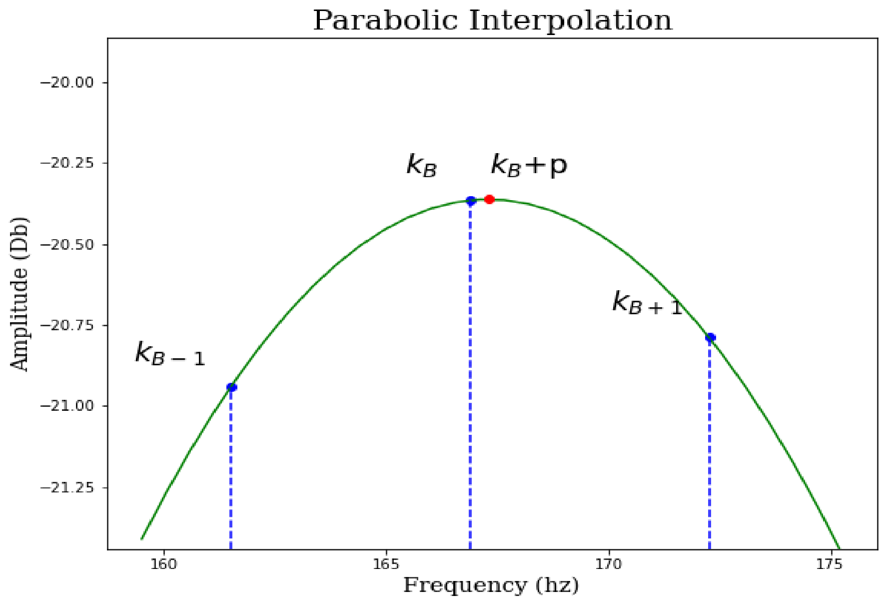


Figure 4: Depicts the parabolic interpolation of harmonic magnitude peaks with the peak bin at index $K_\beta$, the right neighboring bin at index $K_{\beta+1}$ with magnitude and the left neighboring bin at index $K_{\beta+1}$.

Eq. 14 depicts the equation to solve for the parabolic peak location and eq. 15 depicts the expression for the peak location in terms of bins where k is the peak bin index.

$$\gamma \stackrel{\text{def}}{=} 20log_{10}|X(k_\beta) - 1| \tag{13}$$

$$p = \frac{1}{2}\frac{\alpha - \gamma}{\alpha - 2\beta + \gamma} \tag{14}$$

$$k^* \stackrel{\text{def}}{=} k_\beta + p \tag{15}$$

### *2.3.2 sine tracking*

In order to track sinusoids through an STFT, it is not sufficient to simply determine the locations of harmonics in each frame. SMS utilizes a sinusoidal tracking technique to determine the continuity of sinusoids through the propagation of a sound. As Serra [17] describes it, for a given frequencies at frame $N - 1$ the ith frequency is matched to a frequency $j$ in frame $N$ for which $|fi - fj|$ is minimum. Serra [17] refers to this value as the maximum-peak deviation. A tolerance of this peak deviation is established so that if no harmonic peaks are detected within a track's deviation tolerance, the sinusoid is considered dead and its magnitude is set to zero.

## 3 Harmonic Modulation Envelope Filters (HME)

Depending on the amount of harmonics that are excited in the onset of a note, the HME is able to obtain the precise frequency, phase and amplitude values for the first 30-100 overtones present in the spectral model of a recording of a plucked string. Using this data we can begin to generate our HMEs. The motivation behind the HME is to modulate the amplitudes of the corresponding harmonics of a spectral model of a carrier signal. For the purpose of this paper we used a 30 second recording of a plucked E string on a guitar with a fundamental at 82hz as our modulator signal and a 30 second audio clip of a sawtooth waveform synthesized at 82 hz using Ableton Live's[1] "Operator" plugin to serve as a carrier signal. The HMEs we generated from the sinusoidal model of a guitar string will control the presence of three attributes of the guitar's timbre: the approximated initial power of the harmonics at the onset of the note, the presence of beats and non-linearities and the exponential decay rates for each overtone. Once the parameters for these HMEs are determined they will be summed with the amplitudes of the carrier signal's overtones. Once our modulated carrier wave harmonics are obtained we can synthesize the signal into the time domain using the inverse STFT. We can then sum this harmonic signal with the residual component of the original recording of the guitar. This residual component is obtained by subtracting the measured sinusoids from the original signal. This residual sound in the case of a plucked string contains the transient of a pick hitting the string followed by a twangy sound of vibrating metal. The sound that results from this process retains familiar attributes of a plucked string, but possesses a timbre that contains qualities of both the carrier and modulator signal creating an ambiguity between the two sounds.

## 3.1 Acquiring Harmonic Data

All of the harmonic data presented in this paper was rendered from recordings of a single plucked open E string and a 82hz sawtooth waveform synthesized using Ableton Live's [1] "Operator" plugin. All of these recordings were accomplished using an American 1990 Fender Stratocaster routed through a Radial Pro Passive Direct Box into a Focusrite Scarlett Solo audio interface. We used the open source audio editor, Audacity as our digital audio interface and set all recordings at a sample rate of 44.1 kHz and a 24 bit depth. For most of our spectral modeling operations, we utilized the sms-tools repository that was developed by the Music Technology Group, a research team founded by Xavier Serra at the Universitat Pompeu Fabra.[18] The "SineModelAnal" script from this package enabled us to precisely detect the first 30 harmonics in our recordings. The three features of the guitar's timbre that the HME emulates in the carrier wave are extracted from the amplitude data of each individual harmonic in a single guitar string signal. For this reason we visualise our harmonic data as the amplitude envelope of an overtone over time as seen in fig. 5. This also provides us with a visual insight into the artifacts that emerged in our harmonic amplitude data. For the purposes of this paper, we will focus on the data acquired from the 2nd, 8th, 16th and 21st overtones of one particular recording to provide a sense of the range of artifacts we encountered.

In order to align our HMEs at the start of the note being played in our recordings, we had to enable our system to detect the onset of a note or the time instant in which a note is played[15] before initiating the spectral modeling

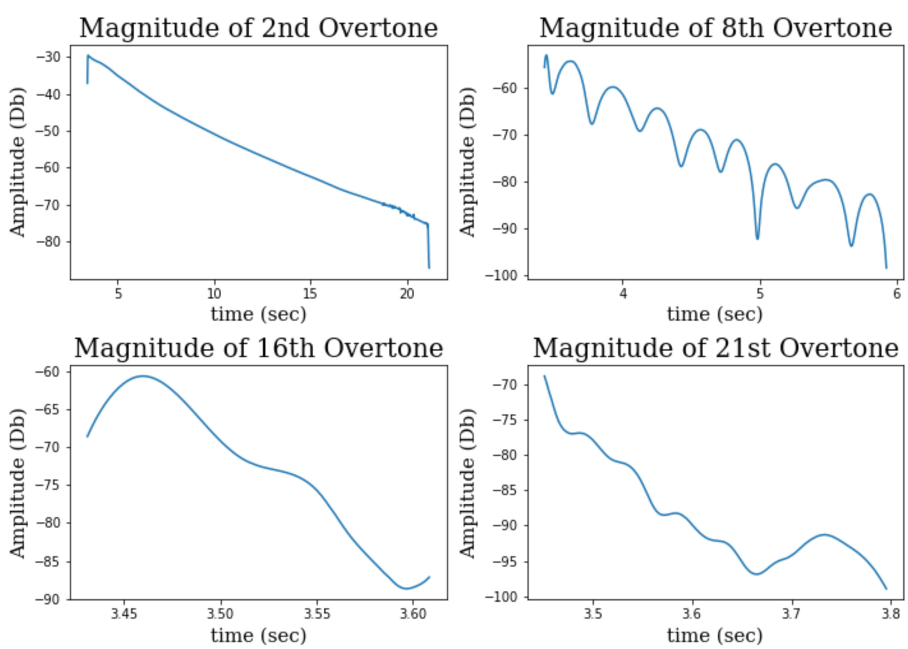


Figure 5: Displayed are the amplitude values of four overtones obtained from a recording of a plucked E string using an STFT size of 8192, a Blackman-Harris window of size 4305 and a hopsize of 103.

Values for the 2nd overtone are displayed in Graph A in which a clear exponential decay is visible with minimal beat patterns and non-linearities. This overtone also exhibits a pronounced transient growth or initial sharp increase in amplitude before settling into an exponential decay. The values for the 8th overtone are shown in graph B which contains clear, consistent beat patterns. The values for the 16th overtone are shown in graph C which presents challenges in analysis due to the short length, making distinctions between transient growth, harmonic decay and beat patterns less clear. The values for the 21st overtone are shown in graph D. This represents the most erratic behavior of an overtone obtained by this particular model. This harmonic appears to follow an exponential decay until around 3.7 seconds after the onset when there is a sharp increase in its amplitude. This behavior could be due to non-linearities becoming more present in the behavior of higher harmonics.

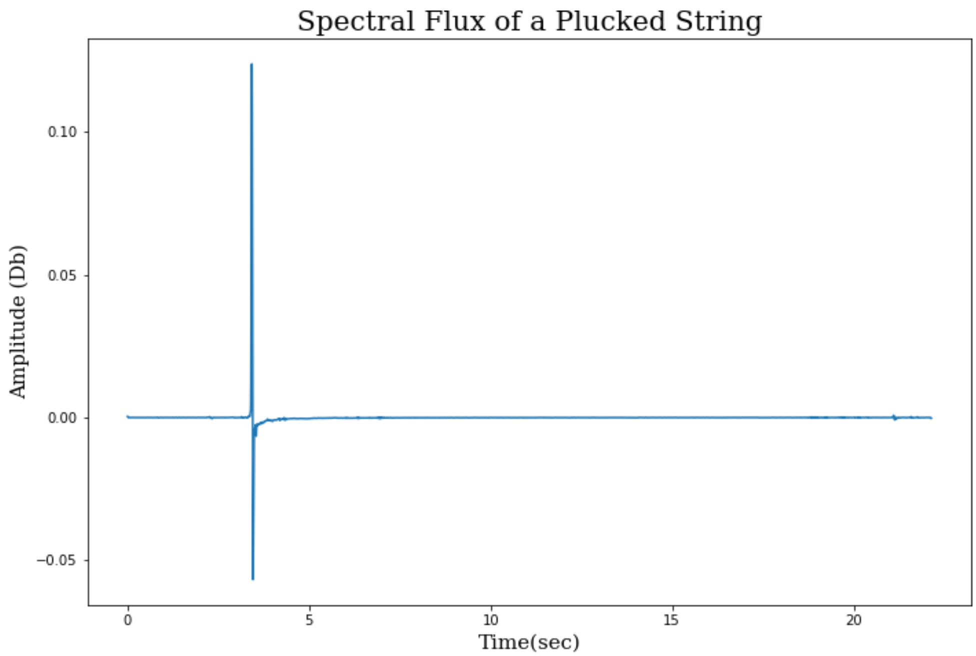


Figure 6: Plot of the spectral flux measured in a recording of a plucked guitar string. The peak of the spike in spectral flux between 0 and 5 seconds is used to determine the onset STFT frame of the note that was played.

process. We utilized a method that measures the spectral flux of the signal to determine the particular onset STFT frame. Spectral flux is a measure of the difference of spectral content between frame $l$ and frame $l - 1$ of the STFT. This method is particularly useful when analyzing a guitar string since the most spectral content is present at the onset of a note. When we measure the spectral flux of our guitar string, there is a clear peak at the onset set of the note when the energy in the spectrum increases sharply see fig. 6. This location of this spectral flux peak will determine the onset frame.

## 3.2 Decay HME

The first HME we generate reproduces the exponential decay of each overtone in the carrier signal. Since we observe the amplitudes of these overtones on a log decibel scale, an exponential decay in our linear amplitude data should correspond to a linear decay in decibels. By approximating the slope of this exponential decay in each harmonic, we can generate our decay HME, which replicates the exponential decay rates of all 30 overtones of the guitar in the carrier signal. see fig. 7 for a display of the approximated harmonic decay against the original sinusoidal amplitude data. The range of the success in detecting a precise exponential decay is shown in the figures below. The approximated decay for the 2nd overtone resembles the original harmonic data much more closely than the 21st overtone which exhibits less linear behavior in its decay and more pronounced beat patterns. These higher harmonics also decay much more quickly which limits the amount of data we have to determine a precise decay rate. The amplitude of 21st overtone appears to decay initially before spiking up around 3.7 seconds after the onset of the harmonic. This erratic behavior of the 21st overtone could be due to non-linearities in the higher harmonics of the string becoming more present.

Due to the presence of beats and nonlinearities, we cannot simply perform linear regression on our sinusoidal data. In order to properly measure the decay rates, we needed a reliable method for measuring the decay of average power of each harmonic. Julius Smith documented the relevance of a measurement known as Energy Decay Relief (EDR)[14] for this particular problem. For frame $m$ in a total of $M$ frames in the STFT, the EDR corresponds to the summation of squared amplitudes in a frequency bin $k$ for frames $m$ to $M$ see eq. 16.

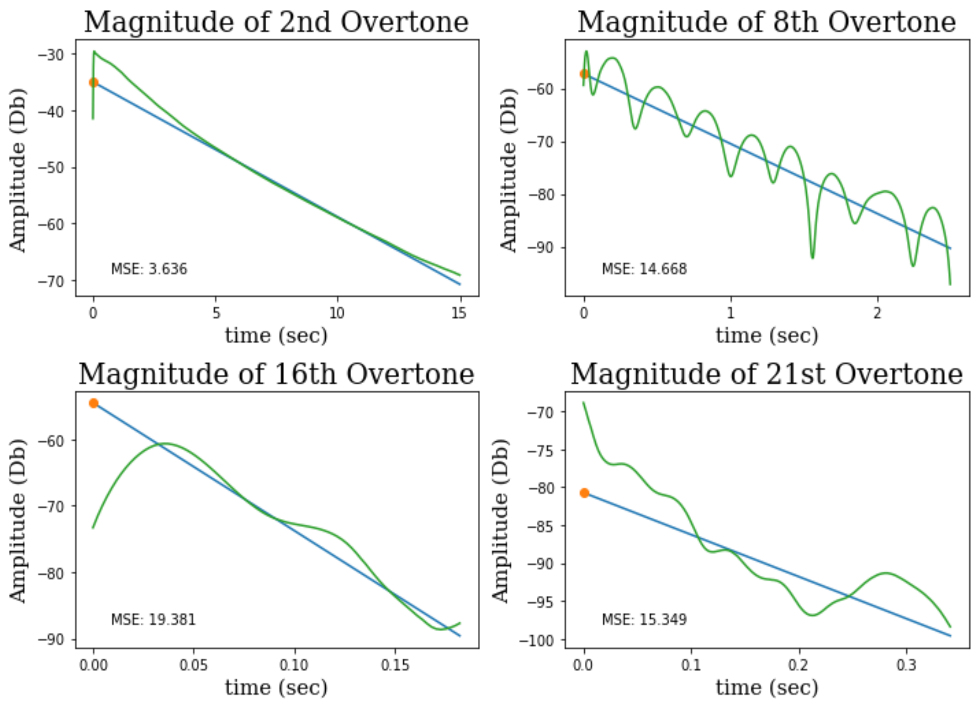


Figure 7: Displayed are the amplitudes of four harmonics (green lines) obtained from a recording of a plucked E string accompanied by their approximated exponential decay (blue lines) and approximated initial power (orange dots).

Notice the close alignment of the 2nd overtone's amplitude to the approximated decay in Graph A and the mean squared error (MSE) value of 3.578, this suggests a relatively small presence of beat patterns and non-linearities. This overtone also demonstrates transient growth, which is marked by an initial increase in amplitude before settling into an exponential decay[16]. Graph B displays the capability of the EDR to approximate the decay of the 8th overtone despite the presence of steep beat patterns. Notice the close proximity of the approximated initial power to the initial magnitude value which demonstrates the lack of transient growth in different harmonics. The values for the 16th overtone are displayed in graph C in which the initial increase in amplitude is interpreted as transient growth. Graph D displays the amplitude of the 21st overtone which contains the largest MSE value at 20.947 due to the erratic behavior of this harmonic.

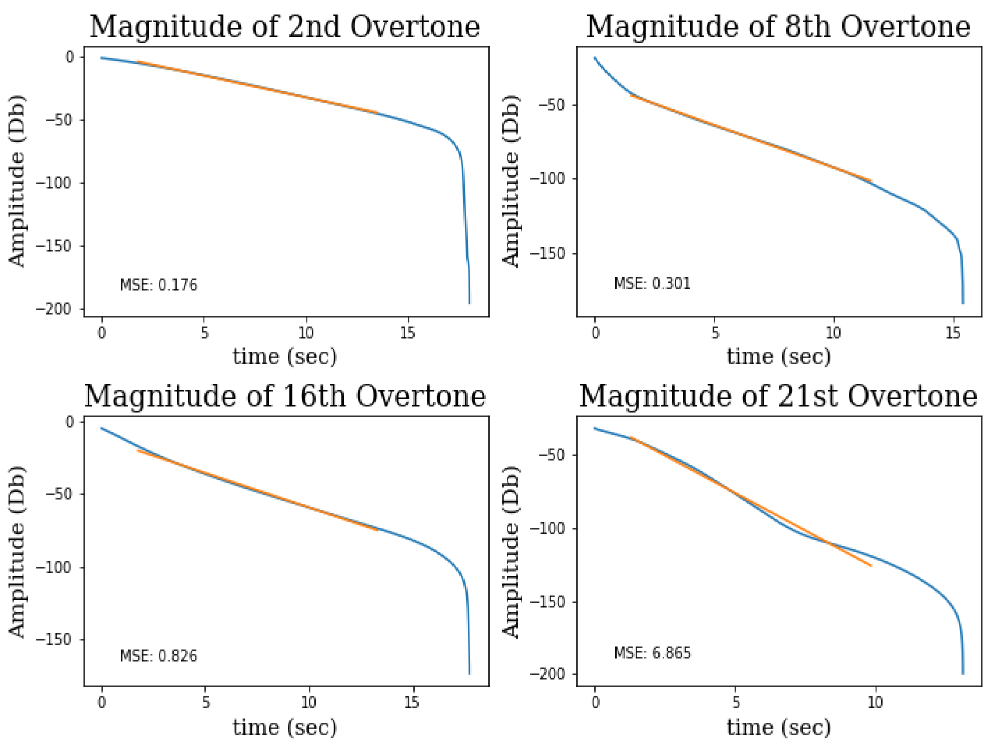


Figure 8: Displayed are the EDR amplitudes of four harmonics (blue line) obtained from a recording of a plucked E string.

The EDR measures the decay of power thus reducing the presence of beats and non-linearities enabling us to fit a line to approximate the exponential decay (orange line). Half of the slope of the orange line determines the slope of the corresponding harmonic's exponential decay. The line fit to the EDR data for the 2nd overtone is closer than any other example. In contrast to the 16th and 21st overtones which display a clear linear slope accompanied by subdued beat patterns. This demonstrates the utility of the EDR in reducing the presence of beat patterns and non-linearities when determining a harmonic's slope.

$$EDR(t_n, f_k) = \sum_{m=n}^{M} |H(m,k)|^2 \tag{16}$$

We obtain decay rates from the EDR of an overtone's amplitude by performing linear regression on the resultant slope it produces as seen in fig. 8. Since the square of an exponential decay value results in doubling the slope in a log scale, we must use half the slope of our EDR results for the harmonic's decay rate. Note the clean linear decay exhibited by the EDR of the 2nd harmonic in contrast to the more distorted results from the 21st overtone which contained more pronounced beats and non-linearities.

The decay HME is produced with a simple linear function for each overtone wherein the y-intercept is optimized by minimizing the root mean error of the function with the original amplitude envelope of each overtone. As seen in eq. 17 the value of the decay HME at a sinusoidal track $r$ and frame $l$ is produced by sum of the approximated y-intercept or initial power for sinusoidal track $r$ and frame $l$ is produced by sum of the approximated y-intercept or initial power for sinusoidal track $r$, $y_r$, with the product of the approximated slope for sinusoidal track $r$, $S_r$ and the frame $l$.

$$d(l)_r = S_r l + y_r \tag{17}$$

When finished, the decay HME is simply 30 amplitude envelopes that exponentially attenuate each harmonic over time. This HME is weighed and summed with the amplitude envelopes of the carrier signal's overtones. We multiply the slopes of this HME with a weighting value to allow the user to control the presence of the original decay times in every overtone in the see fig. 9. This retains the effect of higher overtones decaying faster while proportionally increasing or decreasing every decay rate. This effect can only produce decay rates as slow as that of the carrier signal's harmonics, since a weighting of 0 will produce the same decay rates as the carrier signal. After we generate the onset HME and beat HME, we can combine them with these new decay rates in our decay HME.

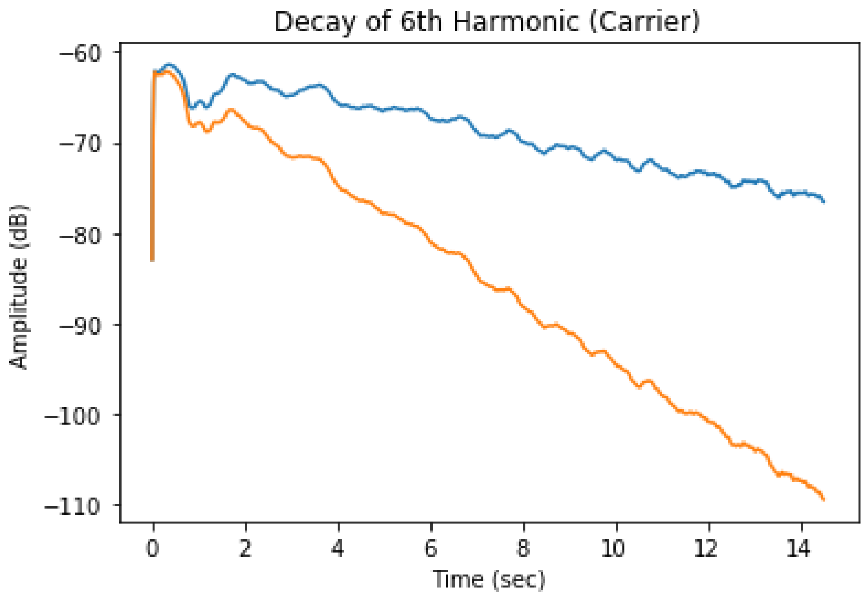


Figure 9: Demonstration of weighting original decay rates of the guitar in the carrier signal. Displays amplitude data for the 6th harmonic of the carrier with an unweighted decay rate (orange line) and a low weighted decay rate (blue line).

## 3.3 Onset HME

Once we have obtained the decay HME, we can then generate the Onset HME. These envelopes are intended to replicate the initial magnitude of the guitar's overtones in the carrier signal at the onset of the note. Intuitively these envelopes would simply use the initial values of each harmonics' amplitude, however, this creates problems in the carrier wave's dynamics due to a phenomenon known as "transient growth"[16]. When observing some of the trajectories of guitar's harmonic amplitudes, we can see an initial growth in amplitude before settling into an exponential decay see fig. 10. This phenomenon is known to occur in coupled oscillator systems like the one Politzer[16] uses to describe the motion of a vibrating string. Politzer[16] states that "transient growth" is a generic term applied whenever all eigenvectors and eigenvalues decay exponentially in time and yet some combinations initially grow before ultimately dying off. Notice the particularly sharp transient growth that occurred in the 2nd Overtone in contrast to the 16th and 21st overtones in fig. 10. Due to the presence of transient growth in certain harmonics, we do not define the initial amplitude of each harmonic by their initial values, but rather by an approximated y-intercept value using our harmonic slope data. While we were able to approximate the slope of the harmonics' decay using the EDR we cannot use it to determine the y-intercept of this slope. As previously mentioned we instead approximate the y-intercept by generating values that minimize the root mean error between the harmonic decay and the original harmonic amplitude data as represented by an orange dot in fig. 10. We call this value, the initial power of the harmonic. Notice the difference in amplitude between the initial values of the harmonic data and the approximated y-intercept.

Once the initial power of each harmonic's decay is determined, we use these values to generate the onset HME. We achieve this by obtaining the difference of the initial power of the guitar's harmonics with the initial harmonic amplitudes of the carrier signals. As seen in eq. 18 the onset HME value obtained for a sinusoidal track $r$ by subtracting the initial harmonic amplitude of the $r$th overtone of the carrier signal, $y_r^c$, from the approximated y-intercept or initial power of the $r$th overtone of the modulator signal, $y_r^m$. When this HME is summed with the carrier wave's overtones, it has the effect of introducing the initial spectral conditions of the recorded pluck. If the guitar was plucked at the bridge or palm muted the initial brightness or darkness of those effects would be audible in the carrier wave. It may also be useful to think of this HME filter as introducing the initial spectral conditions of a note played on the guitar.

$$O(r) = y_r^m - y_r^c \tag{18}$$

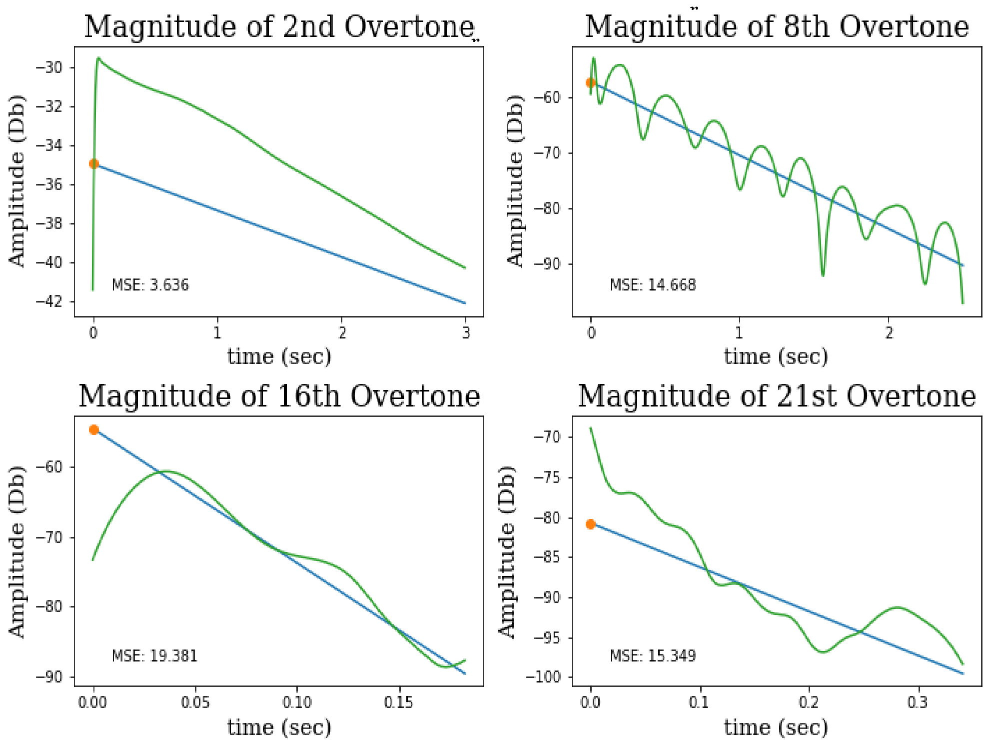


Figure 10: Displayed are the first 1-3 seconds of the amplitudes of four overtones (green line) obtained from a recording of a plucked E string accompanied by their approximated exponential decay (blue lines) and the approximated y- intercept of their decay (orange dots).

The 2nd overtone exhibits transient growth, rising from an initial value of -44 dB to around -35 dB before settling into an exponential decay. The 8th and 21st overtones do not exhibit transient growth as demonstrated by the close proximity of their initial amplitude values to their approximated initial values. While the initial value of the 16th overtone is more than 10 dB lower than it's approximated y-intercept, it is not clear whether this is due to transient growth or a beat pattern since this oscillation in amplitude repeats, however briefly, through the propagation of the harmonic.

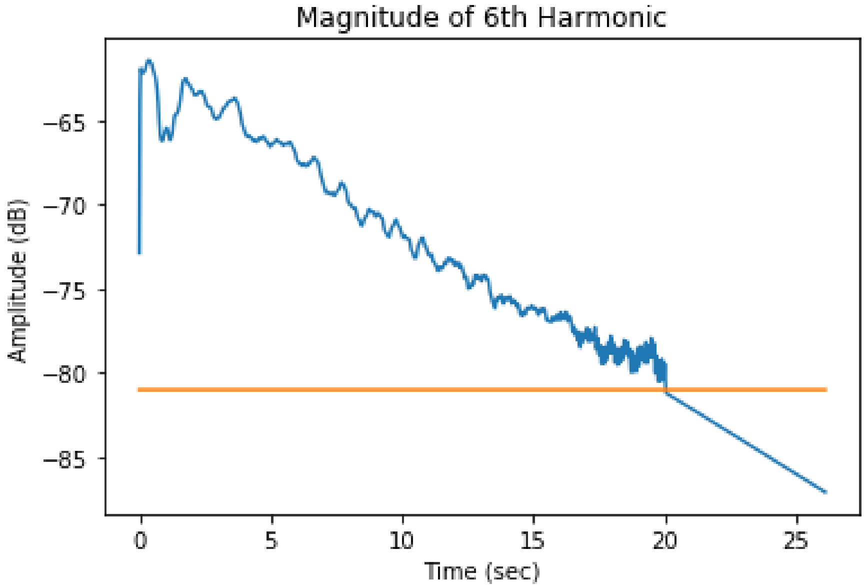


Figure 11: Amplitude of the 6th Harmonic falls below the threshold of -81 dB. Harmonic data is generated after falling below threshold using the harmonic decay slopes and approximated y-intercepts for the 6th harmonic.

## 3.4 Beat HME

Using our simulation of the guitar's harmonic decay we can generate the beat HME to expose the beat patterns and non-linearities present in the amplitudes of the guitar's partials. To do this, we start by generating the decay HME. We then subtract the generated amplitude values of the decay HME with the original amplitude data of each overtone. As seen in eq. 19 the value produced by the beat HME at a sinusoidal track $r$ and frame $l$, $B_r(l)$, is obtained by the difference of the magnitude of the $r$th harmonic of the guitar recording or modulator, $M_l^r$ at frame $l$ and the value of the decay HME for the $r$th harmonic at frame $l$. We are left with an amplitude envelope of each harmonic's beat patterns over time see fig 9. for the beat envelope for the 6th harmonic. Since this particular model doesn't measure harmonics that fall below -100 dB, our beat HME will run out of data after a harmonic falls below that threshold. If an overtone falls below this threshold or if a dropout of data occurs in this decibel range, we will use the measured exponential decay of that harmonic to generate the rest of the required data for synthesis as seen in fig. 11.

$$B_r(l) = M_l^r - d_r(l) \tag{19}$$

As previously mentioned, the transient growth that can occur in the partials of a vibrating string can result in abrupt changes in their amplitudes at the onset of a note. This transient growth will appear in some harmonics in the beat HME, in order to attenuate its presence, we employ a time domain weighting envelope to these envelopes. This envelope can be used to reduce the weighting of the beat HME at the beginning of the note when transient growth can occur, and fade into a higher weighting during the note's decay. An example of an unweighted beat envelope can be seen in fig. 12. This introduces an attack or initial gradient into the beat HME. It may also be necessary to utilize a frequency domain weighting envelope with the beat HME as the amplitudes of higher harmonics are more dynamic and behave less linearly.

We also observed benefits when weighting our decay HME. As Kevin Karplus and Alex Strong discussed in their paper[13], the sounds that were synthesized that utilized the original or similar decay rates of the guitar still sounded distinctly like a plucked string. Stretching these day rates resulted in a sound that retained some ambiguity between the guitar and the carrier signal. The ability to control what dynamics of the guitar's timbre are present in different proportions in synthesis is a very interesting concept for sound design. Being able to expose these dynamics and resynthesize them in other sounds is what makes this process unique.

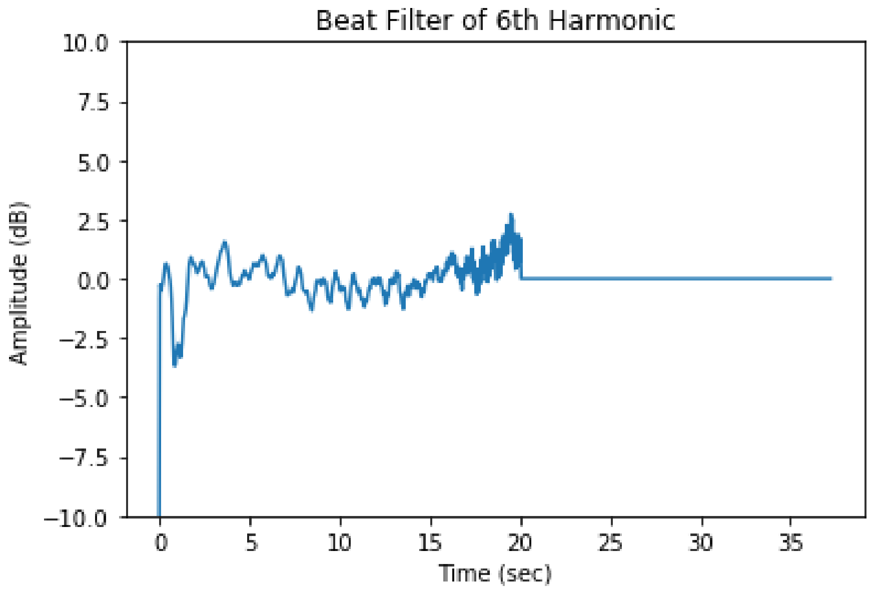


Figure 12: This amplitude envelope for the 6th harmonic of the beat HME isolated the harmonic's beat patterns and non-linearities, as well as the transient growth present in the first few milliseconds which produces a 10 dB increase at the note onset.

## 3.5 Weighting the HME

The recordings we produced using an open e string as a modulator signal on a sawtooth wave revealed to us the importance of calibrating the weightings for our HMEs. Our first attempt at synthesis did not utilize weightings and as a result, a lot of the dynamic beat patterns of higher harmonics resulted in high frequencies being synthesized at dangerously high levels of volume. This was due to the sawtooth wave containing more sustained harmonic content than our string recording, resulting in more audible, dynamic beat patterns in the higher harmonics of the modulated sawtooth wave. Anyone that utilizes a system like this to a creative end could foreseeably prefer an option to decrease the presence of the beat patterns in higher harmonics. As seen in eq. 20 the final amplitude values are obtained by the sum of the magnitude of the $r^{th}$ harmonic of the synthesizer or carrier signal, $C_l^r$ at frame l with the product of the value of the beat HME, $B_r(l)$, and it's correspond weight $w_B$, the product of the value of the onset HME $O(r)$ and it's corresponding weight $w_o$ and the product of the value of the decay HME $d_r(l)$ and its corresponding weight $w_d$.

$$H_r(l) = C_l^r + w_B B_r(l) + w_o O(r) + w_d d_r(l) \tag{20}$$

## 3.6 Implementation Details

In the process of testing our HME system on single notes and four note melodic phrases we encountered some issues in obtaining the desired timbral effect of the HME filters, consistently acquiring clean harmonic data, and controlling the dynamics of the HME filters. We implemented the following solutions to mitigate these issues in our results.

### *3.6.1 HME compressor*

When we first synthesized audio from our HME system, we encountered frequencies that were being modulated to dangerously high volumes, especially in the higher harmonics. Due to this dynamic nature of the HME filters, particularly in the beat and onset filters, we had to develop a compression function to reduce their dynamic range. Just like an audio compressor, this function reduces the dynamic range of the attenuation being produced by the HME. This resulted in more control of the presence and consistency of these filters in the final synthesis.

### *3.6.2 guitar volume envelope*

A second measure that was taken to control the dynamic range of the synthesized audio was to modulate it's volume with the volume envelope of the guitar. This maintained all of the contributions of the HME filter at a listenable volume.

### *3.6.3 improving harmonic data resolution*

When we were creating our first harmonic models of the guitar, the synthesized audio of the harmonics closely resembled the timbre of the guitar. However, when creating HME filters with these models, we realized there were sinusoidal tracks of harmonics that were being conflated with one another. This resulted in abrupt fluctuations in our HME filters. We were using the "sinetrack" function from the sms-tools "Sine Model" script to produce sinusoidal candidates for each note from the guitar's STFT. We then created and refined two more functions to properly select and order the har- monics of each note from this input of sinusoidal candidates in order to reduce this harmonic distortion. The function "ben clean" was created to eliminate frequencies that are lower than the guitar's range, and eliminates abrupt changes in the frequency and amplitude of each sinusoid. The function "ord harmonics" was created to select the harmonics by prioritizing the sinusoids with the highest amplitudes within the frequency range of each harmonic.

## 3.7 Interpolating HME Parameters

We also observed that the final sinusoidal model that resulted from our HME system contained less harmonic content than the sounds they were derived from. This was due to the harmonics measured in the guitar's STFT not overlapping with the harmonics that were measured in the synthesized signal. In order to mitigate these gaps in our harmonic data, we implemented a function that linearly interpolates the decay slopes and initial power values for missing harmonics in the guitar harmonic model. For example if our system measures $N$ harmonics in a note played on the guitar and the highest harmonic measured is $F_n$ and $F_{n-1}$ is not was not detected, then the slope and initial power values for the missing harmonic $F_{n-1}$ will be linearly interpolated using known values from harmonics $F_{n-2}$ and $F_n$.

# 4 Results

The power of the HME is in its ability to leverage the timbral dynamics of both acoustic and synthesized sources of sound. The expressive physical control of timbre achieved when playing a guitar can be combined with the variety of timbre that can be accessed by a virtual instrument.

To demonstrate this effect we applied this system to an 8 note motif from the melody of Herbie Hancock's composition "Dolphin Dance". We recorded a performance of the melody on a guitar to produce the HME filters and converted the recording to midi data in order to play the melody on some of Ableton Live's[1] virtual instruments. This melody was performed with expressive articulations with certain notes slightly muted or struck with more force. We demonstrate each of our three HME filters with this recording using a sawtooth generated with Ableton Live's operator plugin as the carrier signal in a video which can be found at this link: https://youtu.be/sRibG7nVxG4. This resulted in a very unique expression of the melody that leveraged both acoustic and digital dynamics of the timbre. Finally, we combined the HME filtered harmonics with the residual sound of the guitar. In order to obtain the residual sound of the guitar we subtracted the harmonic content we obtained for each note from the original recording which isolated the percussive transient sounds of the pick hitting the string. This residual sound was then summed with the modulated carrier harmonics. The result contains a very interesting ambiguity between both the guitar and the synthesizer. We demonstrated this process on three other virtual instruments from Ableton Live[1] in a video which can be found at this link: https://youtu.be/TCx3QeKtMTc . As seen in the video, very interesting, musical timbres can be achieved by properly weighting and compressing each HME filter.

## 4.1 Limit of Harmonic Guitar

The first shortcoming we noticed in our implementation of the HME technique is the inability to resynthesize harmonics that are higher than the highest harmonic measured by our spectral model of the guitar. This results in certain notes that only contain three harmonics when our model of the synthesizer contains 30 harmonics. While this does not significantly diminish the majority of our results it still exposes a significant fault in our design.

## 4.2 Onset Filter

The aesthetics of each of our HME filters presents a variety of choices in synthesizing the degree of resemblance to either the guitar or the virtual instrument. The choice of weighting each of these filters is what contributes to the presence of these attributes in the final sound.

The onset filter, which controls the influence of the guitar's initial frequency spectrum on the synthesized sound, can express how muted a note is or the presence of higher frequencies, but significantly attenuates presence of the synthesizer's harmonics to that of the guitar. We utilized very light weightings in order to retain some resemblance of the frequency spectrum of the virtual instruments. We noticed that the onset filter would make the final sound resemble the guitar's timbre too closely.

## 4.3 Decay Filter

The decay filter can express the presence of higher frequencies in the guitar without affecting the initial timbre of the synthesizer. This results in a sound that initially sounds like the synthesizer with higher frequencies decaying faster. These filters are useful for making the final sound more similar to the guitar as the higher harmonic content at the beginning of a note and harmonic decay are hallmark qualities of a guitar's timbre.

## 4.4 Beat Filter

The beat filter is very useful in making the synthesizer sound more "natural" without sounding too much like a guitar. This is because the dynamics of beat patterns are present in many acoustic instruments and are not associated with any one particular instrument. The beat filter is however the most dynamic and unpredictable of the three filters and should be weighted with caution so as not to increase frequencies to undesirable volumes. It is highly recommended to compress this particular filter to contain these dynamics. We achieved very satisfying results with the decay and onset weights set at .5, the beat filter weighted at 2 and the beat HME compressed at a threshold of 10 dB and a ratio of $\frac{1}{4}$.

# 5 Conclusion

This paper outlines a method of musical synthesis utilizing spectral modeling to extract timbral features of a guitar string to modulate the partials of a virtual instrument.

## 5.1 HME Application

What makes the HME unique, is in its ability to replicate the dynamics of natural musical timbres. While we have analyzed the guitar as a modulating source, the human voice as well as any natural harmonic sound could be utilized to this end. This implementation of the HME was designed to represent the component dynamics of a guitar string, which if scaled to a real time application could allow a guitar player to combine their instrument's timbre with midi in a more detailed way than is currently possible. While this software does not currently operate with a real time guitar signal, it does work with pre-recorded audio files. A more feasible utilization of this software in the near future would be in audio post production. If a producer or a sound designer records with a virtual instrument, or any harmonic sound for that matter, they could use the HMEs of a guitar with a virtual instrument to provide more natural variability to the final sound. For example if a producer records a synthesized brass sound in a digital audio workstation, they could record the same melody on a guitar and use the HME to combine the nuances of both sounds to create something entirely different.

## 5.2 Further Research

While guitar players would require a real time application in order to use a system like this, it was not within the scope of this project to prototype the hardware and software necessary to do so. For this reason none of the software that was built is scalable to real time functionality. Further research could be done investigating a real time application of these concepts. Machine learning or statistical modeling could be leveraged to this end by predicting parameters such as the guitar's harmonic decay rates and overtone locations in the frequency domain in real time.

## 6. Acknowledgements

This work was supported by funding from the UNCA Undergraduate Research Summer Grant.

## 7. References

[1] Ableton AG. Ableton live.

[2] S. Bagchi and S. K. Mitra. *The nonuniform discrete Fourier transform and its applications in signal processing*, volume 463. Springer Science & Business Media, 2012.

[3] P. Cano, A. Loscos, J. Bonada, M. Boer, and X. Serra. Voice morphing system for impersonating in karaoke applications. 08 2002.

[4] J. Engel, K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue, and A. Roberts. Gansynth: Adversarial neural audio synthesis. *arXiv preprint arXiv:1902.08710*, 2019.

[5] J. H. Engel, C. Resnick, A. Roberts, S. Dieleman, D. Eck, K. Simonyan, and M. Norouzi. Neural audio synthesis of musical notes with wavenet autoencoders. *CoRR*, abs/1704.01279, 2017.

[6] J. Flanagan and R. Golden. Phase vocoder. *Musica/Tecnologia*, pages 9–25, 2013.

[7] N. H. Fletcher. Analysis of the design and performance of harpsichords. *Acta Acustica united with Acustica*, 37(3):139–147, 1977.

[8] N. H. Fletcher et al. Plucked strings-a review. *Catgut Acoust. Soc. Newsletter*, 26:13–17, 1976.

[9] N. Giordano, H. Gould, and J. Tobochnik. The physics of vibrating strings. *Computers in Physics*, 12(2):138–145, 1998.

[10] F. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1):51–83, 1978.

[11] D. A. Jaffe and J. O. Smith. Extensions of the karplus-strong plucked-string algorithm. *Computer Music Journal*, 7(2):56–69, 1983.

[12] E. V. JANSSON. Acoustics for the guitar player. *Function, Construction, and Quality of the Guitar*, pages 7–26, 1983.

[13] K. Karplus and A. Strong. Digital synthesis of plucked-string and drum timbres. *Computer Music Journal*, 7(2):43–55, 1983.

[14] N. Lee and J. Smith. Virtual stringed instruments, 2007.

[15] M. Mounir, P. Karsmakers, and T. van Waterschoot. Guitar note onset detection based on a spectral sparsity measure. In *2016 24th European Signal Processing Conference (EUSIPCO)*, pages 978–982. IEEE, 2016.

[16] D. Politzer. The plucked string: an example of non-normal dynamics. *American Journal of Physics*, 83(5):395– 402, 2015.

[17] X. Serra. *A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decom- position*. PhD thesis, Stanford University, 1990.

[18] X. Serra and the Universitat Pompeu Fabra Music Technology Group. sms-tools: Sound analysis/synthesis tools for music applications written in python (with a bit of c) plus complementary teaching materials. https://github.com/MTG/sms-tools.

[19] C. Traube. *An interdisciplinary study of the timbre of the classical guitar*. PhD thesis, McGill University, 2004.