

Investigating the effect of pickup position on the harmonic response of an electric guitar

Marko Vejnović

September 2, 2018

(Word count: 2960)

Contents

1	Introduction	1
1.1	Scope of work	1
1.2	Background information	2
1.2.1	Musical background	2
1.2.2	Classical guitar operation	2
1.2.3	Electric guitar pickup operation	3
1.2.4	String harmonic series	3
1.3	The Fourier transformation	5
1.3.1	Fourier series	6
1.3.2	The Fourier transformation	9
2	Method	10
3	Data	10
3.1	Raw data	10
3.2	Data processing	12
4	Analysis	13
4.1	Poor harmonic identification	13
4.2	Sustain	14
4.3	Other minor effects	14
5	Conclusion	14
5.1	Rectifying errors	14
5.1.1	Sustain	14
5.1.2	Timbre from picking	15

1 Introduction

1.1 Scope of work

Given the rise of popularity of electric instruments in the mid-20th century, music has become an art in engineering and physics as well as music itself. To achieve a pleasant tone, music instruments are designed in many different manners, as shown in figure 1.



(a) A Fender Stratocaster (Chicago Music Exchange) (b) A Fender Telecaster (Guitar Center)

Figure 1: Various types of electrical guitars

The guitar pickup is unmistakably the most important piece of an electric guitar. It is precisely what creates the electrical signal from the vibrations of the strings. This study aims at finding the precise physical effect of the position of the pickup, as this plays a major role in the tone of the electric guitar. Research in this field could lead to further improvements in the field, with movable pickups for creating various different tones.

1.2 Background information

1.2.1 Musical background

Musical instruments in general operate on musical scales. Musical scales are sets of ordered tones, ordered in ascending fundamental frequencies. The guitar's strings are *E*, *A*, *D*, *G*, *B*, *E* whose frequencies are 82.41Hz, 110.00Hz, 146.83Hz, 196.00Hz, 246.94Hz and 329.63Hz, respectively. It is worth noting that the relationship between the linear increase of tones and the respective frequency of the tone is exponential. This is due to the fact that human ears hear pitch in a logarithmical fashion, so it is necessary to compensate for this biological factor.

1.2.2 Classical guitar operation

Although the classical guitar was not explored in this experiment, the basic theory of its operation is necessary to understand the operation of the electric guitar. A model guitar is given in 2. When the string

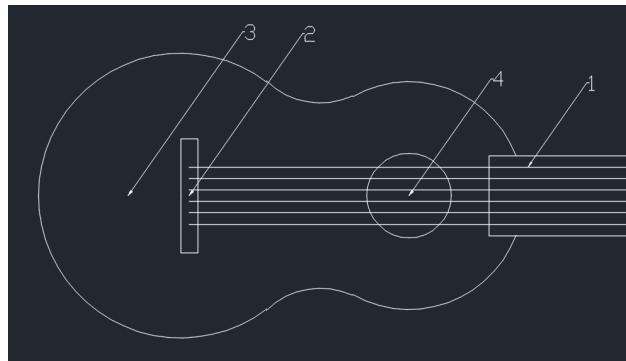


Figure 2: A model of an acoustic guitar

(1) is plucked, it resonates two main components of the guitar through the bridge (2) - the top (3)¹ and the

¹The “top” is the name given to the top wooden plate on the guitar.

air inside the cavity (4), effectively creating *Heimholtz resonance*². The top resonates better with higher frequencies and the *Heimholtz resonance* at play is better with lower frequencies.

1.2.3 Electric guitar pickup operation

The electric guitar operates similarly to the acoustic guitar, except that instead of resonance playing the role of amplifying audio, a *pickup* is used to pick the motion of the string up. The power outputted by the electric pickup is low, and is then amplified by an amplifier to be played on an electric speaker. An electric guitar pickup is given in figure 3. When a string (1) is strung on top of the permanently magnetized bobbin

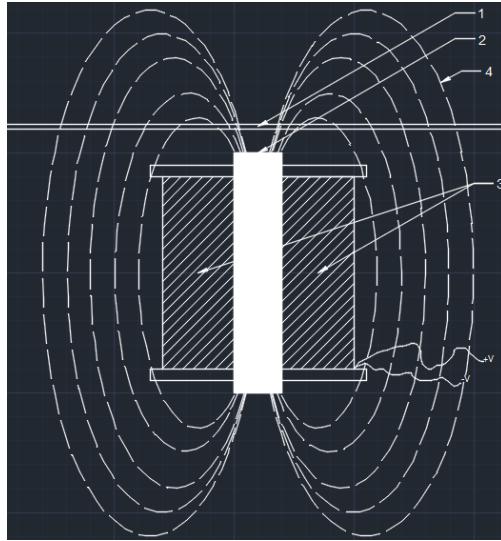


Figure 3: A model of an electric guitar pickup

(2), which is surrounded by very thin electrically insulated conducting wire (3), it becomes magnetized. When the string is plucked, the surrounding magnetic field causes the magnetic field of the bobbin (4) to be disturbed, causing the individual point charges in the wire (3) to experience a *Lorentz force*³. Because of this, an electromotive force appears in the wire (3) which is linearly dependent on the motion of the string (1).

1.2.4 String harmonic series

The string, as any oscillating body with fixed ends, creates harmonic series - sequences of sinusoidal tones whose frequencies are integer multiples of the lowest frequency, ie. the fundamental frequency. How high the amplitude of the individual harmonics is depends on a multitude of factors - the thickness of the string, the material of the string, how it is attached to the bridge, et cetera.

Because of these harmonic series an interesting effect comes to pass when an electric pickup is used. The string oscillating over a pickup at multiple harmonics is given in figure 4. As is visible from the figure, at the position of the pickup, the string oscillates at the fundamental frequency and the second harmonic frequency, however, the third harmonic waveform is at the node at the position of the pickup, and therefore the harmonic is not picked up. Note also, that the pickup will not pick up the first or the second harmonic maximally, as at the line parallel to the magnet intersecting the string, the string is not at its anti-node.

²Heimholtz resonance is a phenomenon of air resonating inside a body with a cavity (Wolfe).

³Lorentz force is the force that is exerted on a particle moving through an electric and magnetic field (The Editors of Encyclopaedia Britannica).

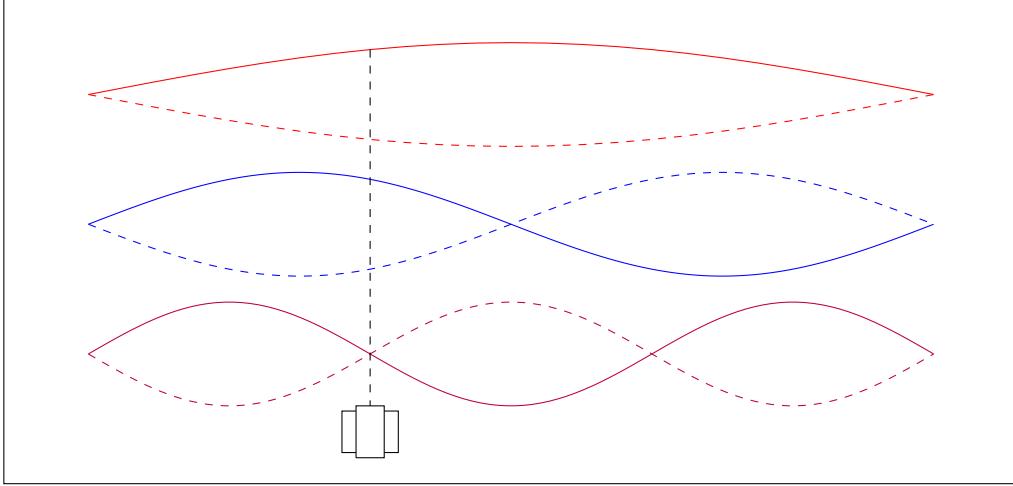


Figure 4: The string oscillating over multiple harmonics over a pickup

A single harmonic can be modeled as a sine wave:

$$w(t) = A \sin(2\pi n\nu_0 t), \quad (1)$$

where A_m is the amplitude of the waveform, n is the number of the harmonic from $n = 1$ to $n = \infty$ and ν_0 is the frequency of the fundamental harmonic.

Since the maximal amplitude at the location of the pickup will not be equal to the amplitude of the waveform itself, it is necessary to derive the formula for the amplitude of the waveform at the location of the pickup. This is achieved as follows.

Consider the waveform given in figure 5. Because the shape of the string is sinusoidal, it follows that the

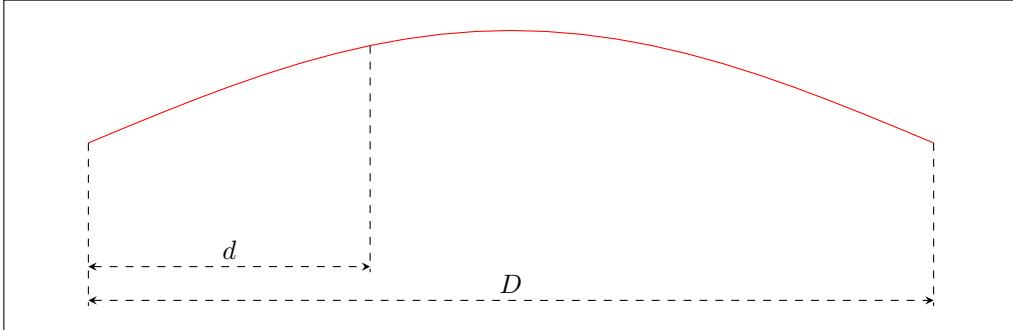


Figure 5: The relationship between the distance of the pickup from the leftmost position and the amplitude at the pickup.

relationship between the distance d of the pickup from the leftmost position and the amplitude at the pickup is sinusoidal as well.

If one were to think of figure 5 as a function, it is the general function of:

$$f_A(x) = k_h \sin\left(\frac{2\pi}{2D}x\right)$$

Note the use of the constant k_h . The constant k_h is in place due to the fact that the amplitude of a harmonic depends heavily on the way the string is plucked, the sustain of the musical instrument, the string tension, and other effects. Usually the most powerful harmonic is the first one, although this is not a rule.

It can be inferred that the amplitude A , when d is substituted for x is equal to:

$$A = k_{h_1} \sin\left(\pi \frac{d}{D}\right) \quad (2)$$

For the second harmonic, the graph is plotted in figure 6.

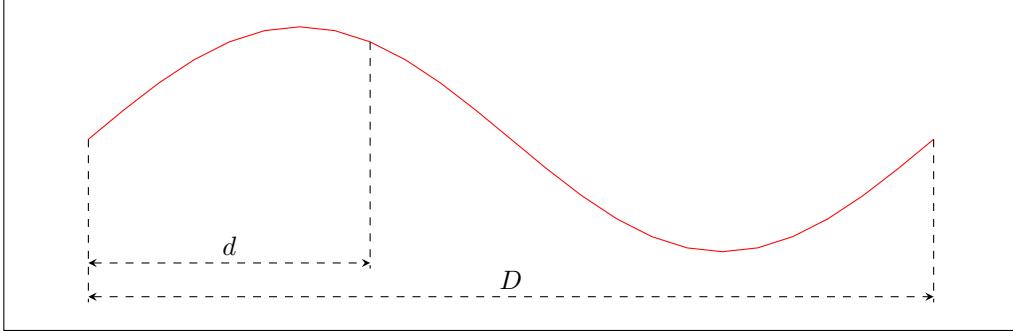


Figure 6: The relationship between the distance of the pickup from the leftmost position and the amplitude at the pickup.

One can infer from figure 6 that the value of amplitude A is:

$$A = k_{h_2} \sin\left(2\pi \frac{d}{D}\right) \quad (3)$$

It follows from equations 2 and 3 that the general formula is equal to:

$$A_n = k_{h_n} \sin\left(n\pi \frac{d}{D}\right), \quad (4)$$

where n is the number of the harmonic, $n = 1, 2, 3, 4 \dots$

When this derived amplitude is substituted into equation 1, the final formula is derived for a single harmonic:

$$w(t) = k_{h_n} \sin\left(n\pi \frac{d}{D}\right) \sin(2\pi n\nu_0 t) \quad (5)$$

To exemplify how the amplitude changes with increasing distance from the left-most position of the pickup, a graph is given in figure 7.

Given that the individual harmonic waves are superimposing, the final waveform will be a sum of all individual harmonics:

$$f(t) = \sum_{n=0}^{\infty} k_{h_n} \sin\left(n\pi \frac{d}{D}\right) \sin(2\pi n\nu_0 t)$$

To differentiate these individual harmonics, it is necessary to find their amplitudes at different frequencies. The Fourier transformation is perfect for this goal.

1.3 The Fourier transformation

To prove the Fourier transformation, a definition of what the Fourier series is necessary.

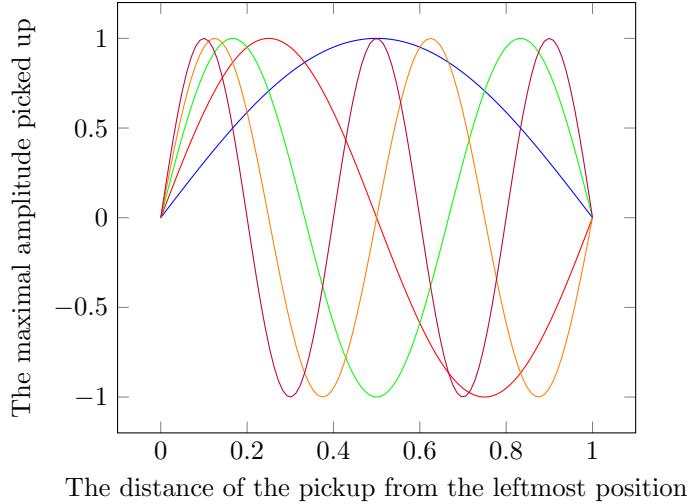


Figure 7: The relationship of the amplitude and the distance of the pickup from the leftmost position for the first five harmonics in relative units

1.3.1 Fourier series

The Fourier series is a method of representing arbitrary periodic functions as a sum of simple sine and cosine functions. A method for deriving it follows.

Suppose an even function such as the one graphed in figure 8 exists.

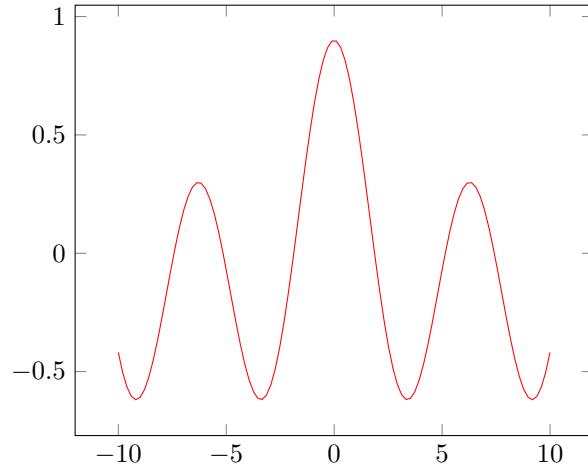


Figure 8: An arbitrary even function

The premise is that it is possible to express this function as a sum of cosine waves:

$$f(t) = \sum_{n=0}^{\infty} a_n \cos(2\pi nt)$$

It is necessary to calculate the values of a_n . This problem is done in the following manner:

Both sides are multiplied by $\cos(2\pi mt)$:

$$f(t) \cos(2\pi mt) = \sum_{n=0}^{\infty} b_n \cos(2\pi nt) \cos(2\pi mt)$$

Next, both sides are integrated over one period of the function, and the trigonometric identity of the product of cosines is employed:

$$\begin{aligned} \int_T f(t) \cos(2\pi mt) dt &= \int_T \sum_{n=0}^{\infty} a_n \cos(2\pi nt) \cos(2\pi mt) dt \\ &= \sum_{n=0}^{\infty} a_n \int_T \cos(2\pi nt) \cos(2\pi mt) dt \\ &= \frac{1}{2} \sum_{n=0}^{\infty} a_n \int_T (\cos((m+n)2\pi t) + \cos((m-n)2\pi t)) dt \end{aligned} \quad (6)$$

Suppose $m \geq 0$ (this is possible to suppose because m is arbitrary, and more importantly, it is physically impossible for frequency to be negative). From this it follows that $\cos((m+n)2\pi t)$ has $m+n$ oscillations in a single period T , and consequently, $m+n$ oscillations in the integration interval. Since this is true, when this function is integrated, the result is 0⁴. Equation 6 is then simplified to:

$$\begin{aligned} \int_T f(t) \cos(2\pi mt) dt &= \frac{1}{2} \sum_{n=0}^{\infty} a_n \left(\int_T \cos((m+n)2\pi t) dt + \int_T \cos((m-n)2\pi t) dt \right) \\ &= \frac{1}{2} \sum_{n=0}^{\infty} a_n \int_T \cos((m-n)2\pi t) dt \end{aligned} \quad (7)$$

Since the cosine function is even, in a single period T (the integration interval - from $-\frac{T}{2}$ to $\frac{T}{2}$) the following is true:

$$\cos((m-n)2\pi t) = \cos((n-m)2\pi t)$$

When $m = n$:

$$\cos((m-n)2\pi t) = \cos(0) = 1$$

From these statements, the following follows:

$$\int_T \cos((m-n)2\pi t) dt = \begin{cases} \int_T \cos((m-n)2\pi t) dt = 0 & \text{when } m \neq n \\ \int_T 1 \cdot dt = T & \text{when } m = n \end{cases}$$

It is key to note that:

$$\forall (0 < m < \infty, 0 < n < \infty \wedge m \neq n) \rightarrow \int_T \cos((m-n)2\pi t) dt = 0$$

From this it follows that the whole summation in equation 7 can be reduced to the case when $m = n$:

$$\int_T f(t) \cos(m2\pi t) dt = \frac{1}{2} a_m T$$

To get a_m is trivial:

$$a_m = \frac{2}{T} \int_T f(t) \cos(m2\pi t) dt$$

Finally, replacing m with n we get:

$$a_n = \frac{2}{T} \int_T f(t) \cos(m2\pi t) dt \quad (8)$$

⁴The proof for this is trivial, however, it is published by a third party online and is available to the reader at the following link: <http://planetmath.org/integraloveraperiodinterval>.

However, the case of $m = 0$ is not considered. This case gives the y axis offset a_0 . Substituting with $m = 0$ in equation 6, we get:

$$\begin{aligned}\int_T f(t) \cos(2\pi mt) dt &= \sum_{n=0}^{\infty} a_n \int_T \cos(2\pi nt) \cos(2\pi mt) dt \\ \int_T f(t) \cos(0) dt &= \sum_{n=0}^{\infty} a_n \int_T \cos(2\pi nt) \cos(0) dt \\ \int_T f(t) dt &= \sum_{n=0}^{\infty} a_n \int_T \cos(2\pi nt) dt \\ \int_T f(t) dt &= Ta_0\end{aligned}$$

This gives the final result for the y axis offset:

$$a_0 = \frac{1}{T} \int_T f(t) dt \quad (9)$$

This result is coincidentally intuitively the average of all of the y values of the points of the function in a single period.

Odd functions are represented in a similar manner:

$$f_o(t) = \sum_{n=1}^{\infty} b_n \sin(2\pi nt)$$

The coefficients b_n are, then:

$$b_n = \frac{2}{T} \int_T f_o(t) \sin(2\pi nt) dt$$

As the derivation is quite similar to the one for the even function, it has been omitted.

Since any arbitrary function $f(t)$ can be represented as a sum of odd and even functions:

$$\begin{aligned}f_o(t) &= \frac{1}{2}(f(t) - f(-t)) \\ f_e(t) &= \frac{1}{2}(f(t) + f(-t)) \\ f(t) &= f_o(t) + f_e(t)\end{aligned}$$

The Fourier series' components can be applied:

$$f(t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos(2\pi nt) + b_n \sin(2\pi nt))$$

with the coefficients being:

$$\begin{aligned}a_0 &= \frac{1}{T} \int_T f(t) dt \\ a_n &= \frac{2}{T} \int_T f(t) \cos(2\pi nt) dt, n \neq 0 \\ b_n &= \frac{2}{T} \int_T f(t) \sin(2\pi nt) dt\end{aligned}$$

It is possible to express the Fourier series in its complex form, using Euler's formulas:

$$\cos\phi = \frac{e^{i\phi} + e^{-i\phi}}{2}$$

$$\begin{aligned}
\sin\phi &= \frac{e^{i\phi} - e^{-i\phi}}{2i} \\
f(t) &= a_0 + \sum_{n=1}^{\infty} (a_n \cos(2\pi nt) + b_n \sin(2\pi nt)) \\
&= a_0 + \sum_{n=1}^{\infty} \left(a_n \frac{e^{i2\pi nt} + e^{-i2\pi nt}}{2} + b_n \frac{e^{i2\pi nt} - e^{-i2\pi nt}}{2i} \right) \\
&= a_0 + \sum_{n=1}^{\infty} \frac{a_n - ib_n}{2} e^{i2\pi nt} + \sum_{n=1}^{\infty} \frac{a_n + ib_n}{2} e^{-i2\pi nt}
\end{aligned}$$

Let us know define c_n as

$$c_n = \frac{a_n - ib_n}{2}$$

This gives the complex Fourier series:

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi nt}$$

This is referred to as the *synthesis* equation.

From this it follows that c_n , the complex Fourier coefficient, is:

$$\begin{aligned}
c_n &= \frac{1}{2} \cdot \left(\frac{2}{T} \int_T f(t) \cos(2\pi nt) dt - i \frac{2}{T} \int_T f(t) \sin(2\pi nt) dt \right) \\
&= \frac{1}{T} \int_T f(t) \cos(2\pi nt) - i f(t) \sin(2\pi nt) dt \\
&= \frac{1}{T} \int_T f(t) \left(\frac{e^{i2\pi nt} + e^{-i2\pi nt}}{2} - i \frac{e^{i2\pi nt} - e^{-i2\pi nt}}{2i} \right) dt \\
&= \frac{1}{T} \int_T f(t) \frac{2e^{i2\pi nt}}{2} dt \\
c_n &= \frac{1}{T} \int_T f(t) e^{-i2\pi nt} dt
\end{aligned}$$

This is referred to as the *analysis* equation. This equation gives spectral lines for a frequency n . Note that it is not continuous.

1.3.2 The Fourier transformation

The Fourier transformation can be thought of as a continuous version of the Fourier series:

$$\hat{f}(t) = \int_{-\infty}^{\infty} f(t) e^{-i2\pi\nu t} dt$$

Let the Fourier transformation be defined as the limit of the analysis formula of the Fourier series as T approaches infinity:

$$\hat{f}(t) = \lim_{T \rightarrow \infty} c_n = \lim_{T \rightarrow \infty} \int_{-T/2}^{T/2} f(t) e^{-i2\pi nt} dt$$

Note the factor $\frac{1}{T}$ is completely ignored, due to the fact that $\lim_{n \rightarrow \infty} \frac{1}{n} = 0$, which would ruin the function. This is far more relevant when doing the inverse Fourier transformation, however the inverse Fourier transformation is outside of the scope of this paper and therefore is not explored. The limit makes the spectral lines come closer, creating the final equation:

$$\hat{f}(t) = \int_{-\infty}^{\infty} f(t) e^{-i2\pi\nu t} dt$$

The power of this equation lies in the fact that it gives a continuous function showing the frequencies of a *non-periodic* signal. In turn, it has practical applications in converting a signal from the time domain to the frequency domain.

2 Method

When a Fourier transformation is ran on a waveform it produces a continuous function of frequency, showing precisely amplitudes of frequencies. To explore the relationship between the pickup position and the harmonic effects, it is necessary to look at a three-dimensional graph where one axis is the independent variable - the position of the pickup, and two dependent variables - the frequency and the amplitude.

To measure the effect of pickup position on harmonics of the electric guitar, an electric guitar with a movable pickup was constructed, as shown in 9. It has several major elements to it, similar to the normal

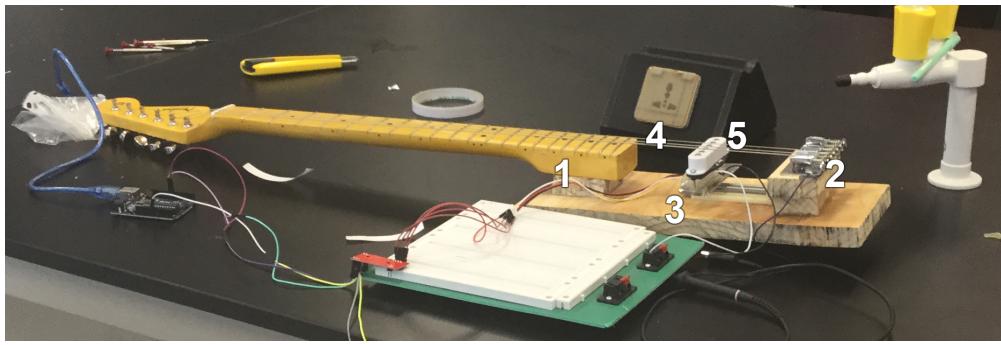


Figure 9: The constructed experimental electric guitar

electric guitar: (1) the neck of the electric guitar, (2) the bridge of the guitar, (3) the body of the guitar, (4) the strings, and, most notably, (5) a pickup whose position can be changed.

The body of the guitar (3) was marked with hatch marks so the pickup could be set to a desired value. These hatch marks were marked using a ruler. They are shown in 10. Since it was mechanically impossible to position the pickup immediately next to the bridge, the distance of the starting (left-most) position of the pickup was measured with a caliper.

The pickup wires were connected to a *GW Insteek GDS-1000A-U* oscilloscope. The ground wire was also connected to the bridge of the electric guitar and the ground of the oscilloscope, to remove any potential noise that might occur, as given in figure 11.

The string (4) was plucked for every position of the pickup. At the same time, the oscilloscope was set to record the waveform. The waveform was recorded using the oscilloscope and stored on a computer for further data processing. Measurements were done for all full-tones, until the first octave, of the guitar for the *E* and *A* strings.

3 Data

3.1 Raw data

The recorded data was a collection of wave-forms in the *CSV*⁵ format, as the oscilloscope generated these files. As an example, the waveform for the *E* string on the 5th fret is given in figure 12 in the left-most pickup position.

⁵ Comma separated values



Figure 10: The hatch marks of the pickup position

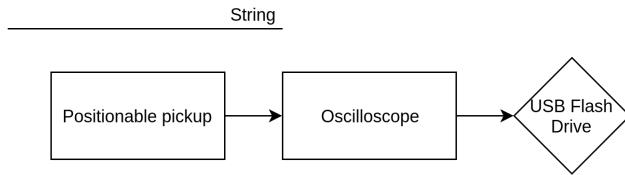


Figure 11: A schematic of how the individual devices were connected

Due to the fact that there has been a huge amount of data, it has been published online and is available at the following link: <https://github.com/markovejnovic/relationship-pickup-distance/tree/master/src/Data>

The uncertainty for the *GW Insteek GDS-1000A-U* is $\pm 2\text{mV}$ (Good Will Instrument Co., LTD).

The left-most position of the pickup is not at a true 0. The minimum offset of the pickup from the bridge could not be mechanically made to be 0, and has been achieved to be equal to

$$d_0 = (31.6 \pm 0.01)\text{mm}$$

Note the very low value of the uncertainty. This has been achieved by using a caliper to measure d_0 from the bridge of the guitar.

The uncertainty for the distance of the pickup from the left-most position d can be calculated as follows. Since the hatch marks were copied from a ruler, the uncertainty during the copying is equal to 0.5mm. Since the pickup was positioned at the copied hatch marks, the uncertainty increases by another 0.5mm. Finally, the uncertainty increased when the distance from the bridge to the first hatch mark was measured with a caliper by the uncertainty of the caliper - 0.01mm. The final uncertainty of d is then:

$$\Delta d = \pm(0.5 + 0.5 + 0.01)\text{mm} = \pm 1.01\text{mm}$$

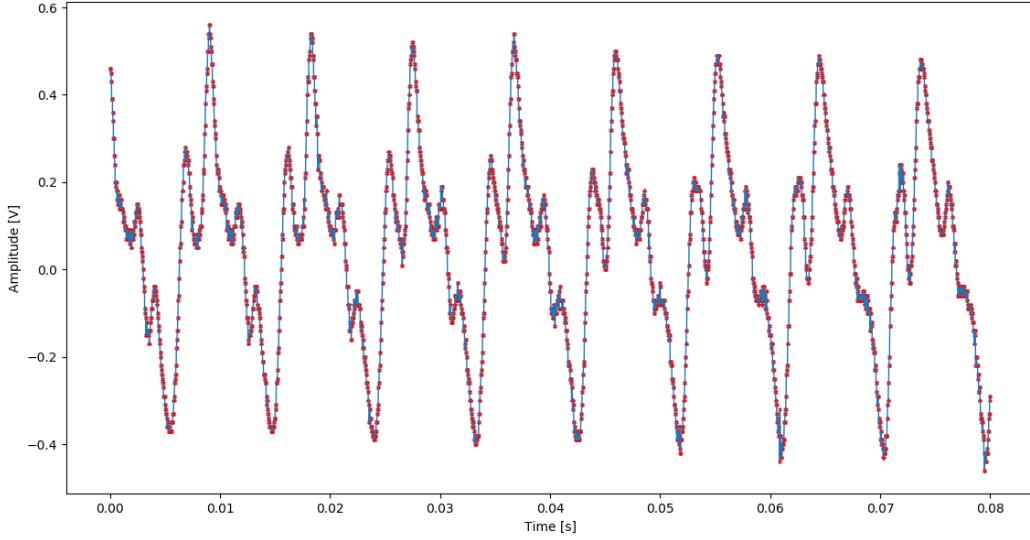


Figure 12: The waveform of the *E* string played on the 5rd fret in the left-most pickup position

3.2 Data processing

The data processing was done in the *python* programming language (Van Rossum, and L. Drake), *numpy* (E. Oliphant) and *matplotlib* (Hunter) as follows.

First the oscilloscope data was loaded into memory, denoted here as a vector w , with the number of elements being 4000. Next, the Fourier transformation was calculated of the waveform, showing the harmonics of the guitar:

$$f = \text{fft}(w)$$

Only the positive side of the Fourier transformation was used, due to the fact that the Fourier transformation in this case is symmetrical, as exemplified in figure 13.

Next, the Fourier transformation was normalized, ie. mapped from 0 to 1 by multiplying the reciprocal of the highest scalar in the vector by the vector itself:

$$f' = \frac{1}{\max_{1 \leq i \leq N} w_i} \cdot f'$$

The previous example in figure 13 becomes normalized as given in figure 14.

Once this has been done, the data was plotted on a three-dimensional graph where one axis was the independent pickup position, and the other two axes were dependent frequencies and amplitudes, as shown in figure 15.

Again, the sheer amount of data obstructs the ease of processing, so in figure 16, uncertainties have been omitted. Also, using the open-source algorithm *peakutils* (Negri), Fourier peaks have been identified. Note that the first harmonics were completely omitted from identification. This is due to the fact that the Fourier transformation was somewhat incomplete, due to the waveform recorded by the oscilloscope being temporally very short, making the fundamental frequencies cut off, and therefore imprecise.

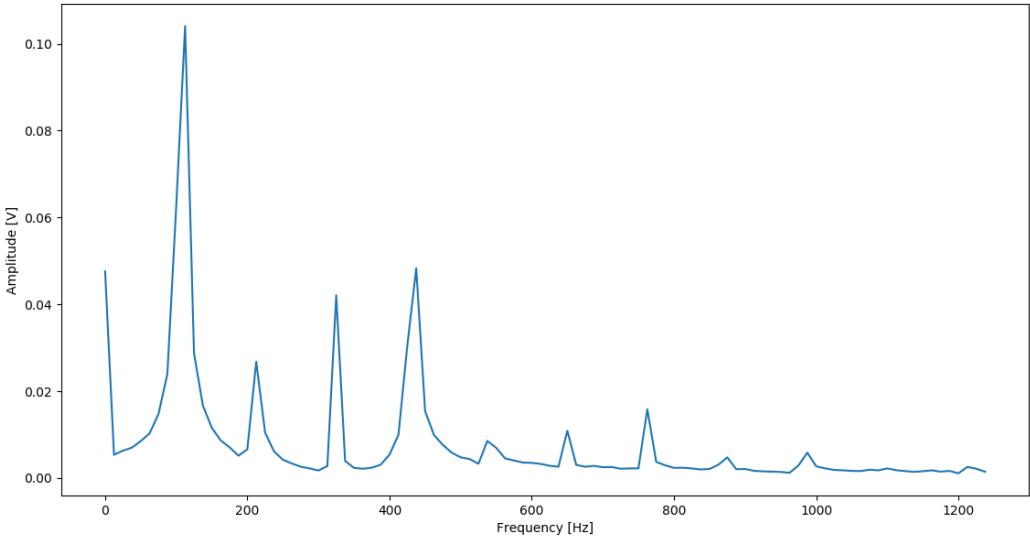


Figure 13: The Fourier transformation of the played 5th fret of the *E* string at the left-most pickup position

Figure 17 shows the same graph as figure 16 with the peaks of the harmonics of different pickup positions being connected. This allows to follow how exactly, the pickup position moves.

The code used to process the data is available both online at <https://github.com/markovejnovic/relationship-pickup-distance/tree/master/src> and in the appendix.

4 Analysis

Considering that the scale length⁶ of the *Fender Stratocaster* is 648mm (“Scale Length Explained”), and the model guitar used in this experiment was constructed according to the *Stratocaster* specification the distance from the nut to the bridge is equal to:

$$D = (648 \pm 0.5)\text{mm}$$

From this it follows that the general formula of the relationship between the harmonic amplitude and pickup position is:

$$\sin\left(n\pi \frac{d}{(648 \pm 0.5)\text{mm}}\right)$$

In figure 18, the plot for the relationship between the first 3 harmonics and the position of the pickup is given, as well as the expected values calculated using the aforementioned formula.

The very poor fit can be attributed to two major factors, and other minor factors.

4.1 Poor harmonic identification

When the same graph as given in figure 18 is observed from a top-down perspective, as given in figure 19, it becomes obvious that the peaks of the Fourier transformation are not colinear. It is notable that the harmonics were poorly aligned. This poor alignment may stem from the following factor.

⁶The distance from the nut to the bridge of the guitar

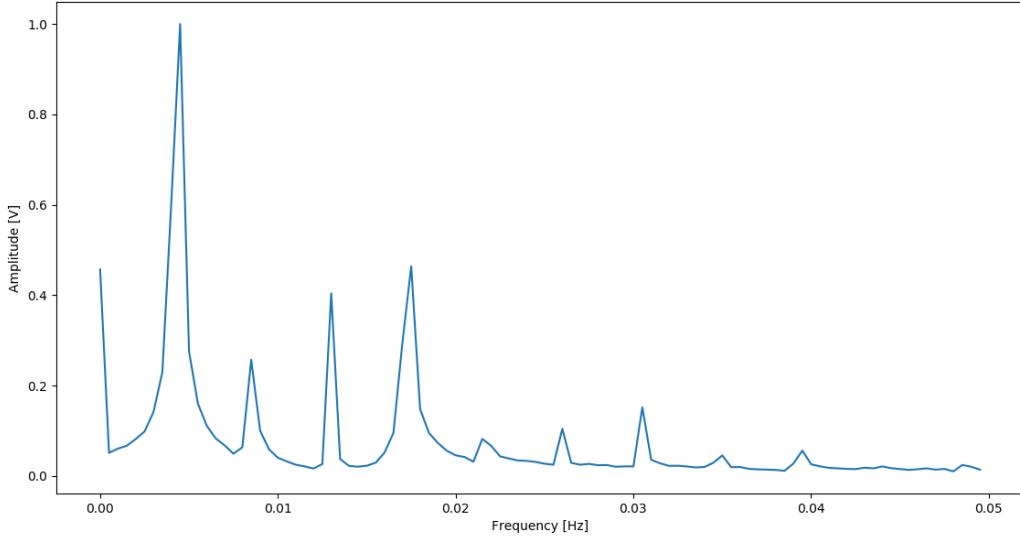


Figure 14: The normalized Fourier transformation of the played 5th fret of the *E* string at the left-most-pickup position.

4.2 Sustain

A guitar string, when first plucked, oscillates at multiple harmonics of varying amplitudes. However, the sustain⁷ of each individual harmonics is not equal. Because of this, certain harmonics will decay faster than others. This data heavily indicates that, during the experiment, the minor difference in time of recording waveforms for each pickup position affected the final Fourier transformation in a major manner.

4.3 Other minor effects

Other effects which could have affected the final Fourier transformation of the guitar waveform are the way the string was plucked, as plucking the string in a different manner creates a different tone; the gradual detuning of the guitar in an environment of varying temperature and mechanical impacts near the guitar which could have caused the strings to vibrate.

5 Conclusion

The experiment results are inconclusive. Non-experimental observations point to there being a significant change in the tone of the guitar with the change of the position of the pickup. Donald Tillman's experimental results reaffirm this observation (Donald Tillman). However, the research conducted in this paper was unsuccessful to the point of showing no notable correlation.

5.1 Rectifying errors

In the following, I have described possible solutions to the issues faced in this research paper.

5.1.1 Sustain

To rectify the issue that was created during sustain, it would be best to record a complete waveform - from the plucking of the string to the complete decay of any sound. This would also rectify the issue of poor

⁷Sustain is the time it takes for a musical note to become inaudible, ie. to decay.

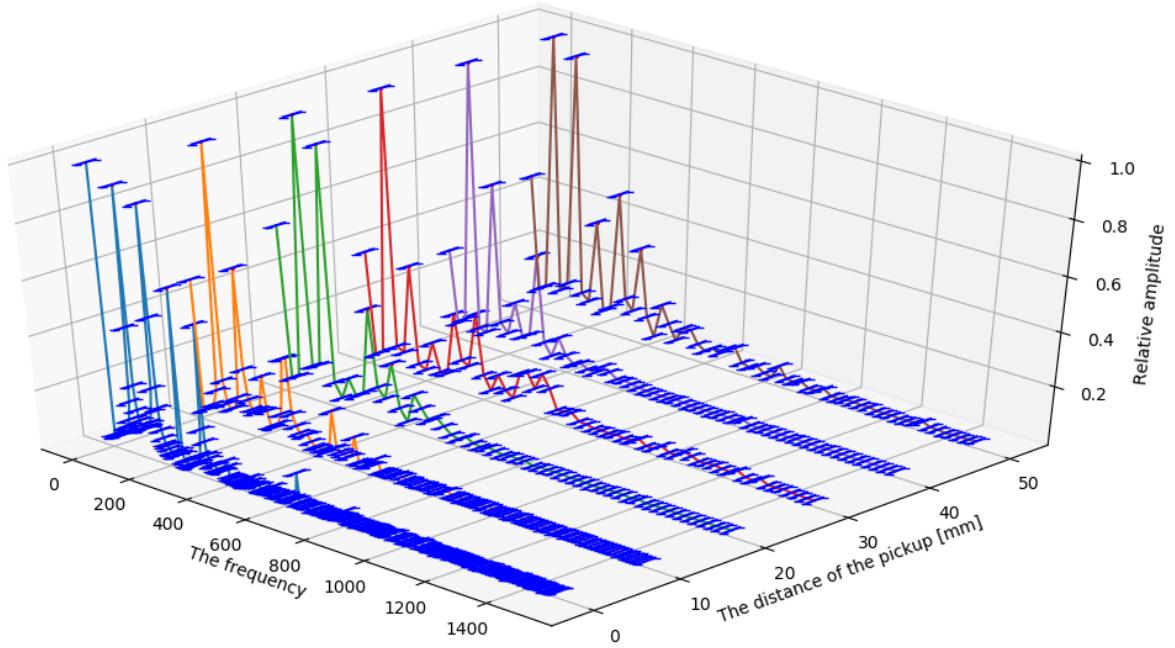


Figure 15: The graph of the dependency of the frequency and amplitudes with respect to the pickup position

harmonic identification.

5.1.2 Timbre from picking

To rectify the various different timbres that were encountered with different picking styles, it is necessary to construct a mechanism which can pick a string with a constant force and angle. A solenoid presents itself as an ideal solution, however, it is important for the solenoid to be far away from the pickup, so as not to interfere with the magnetic field generated by the pickup.

Although the data is highly inconclusive, the experiment remains a valid one. As Donald Tillman's data shows, there are indeed effects on harmonics with the position of pickups. As the pickups move from the left-most position, Tillman's data shows that the frequency response moves translatory to the lower pitches, as the hypothesis suggests.

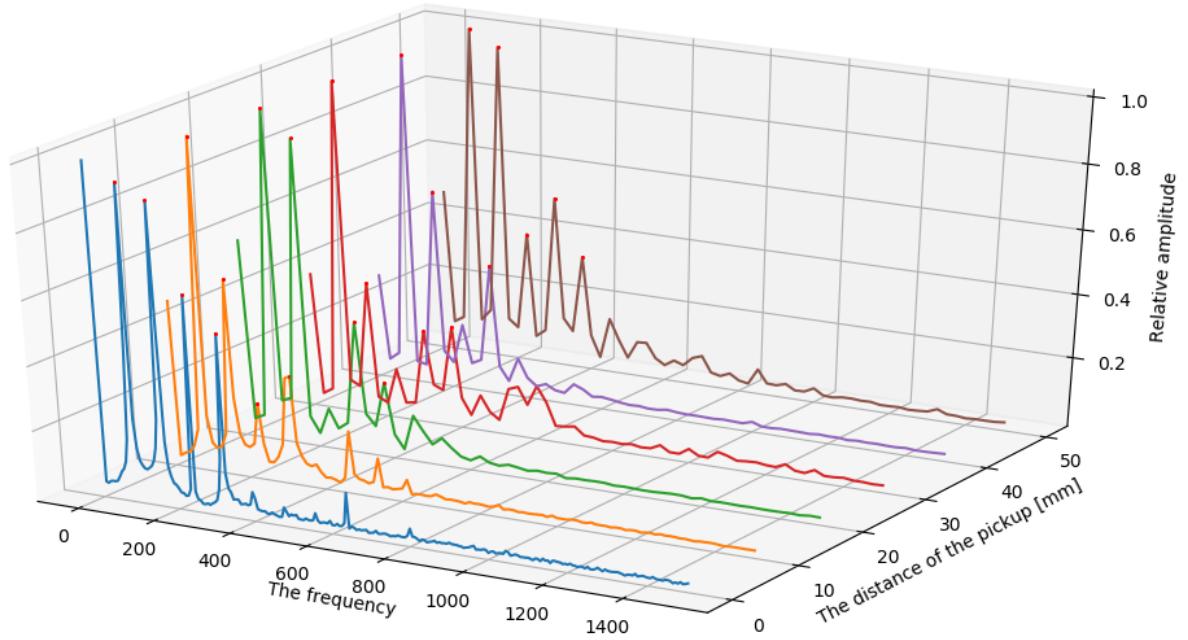


Figure 16: The graph of the dependency of the frequency and amplitudes with respect to the pickup position with omitted uncertainties and with identified Fourier peaks.

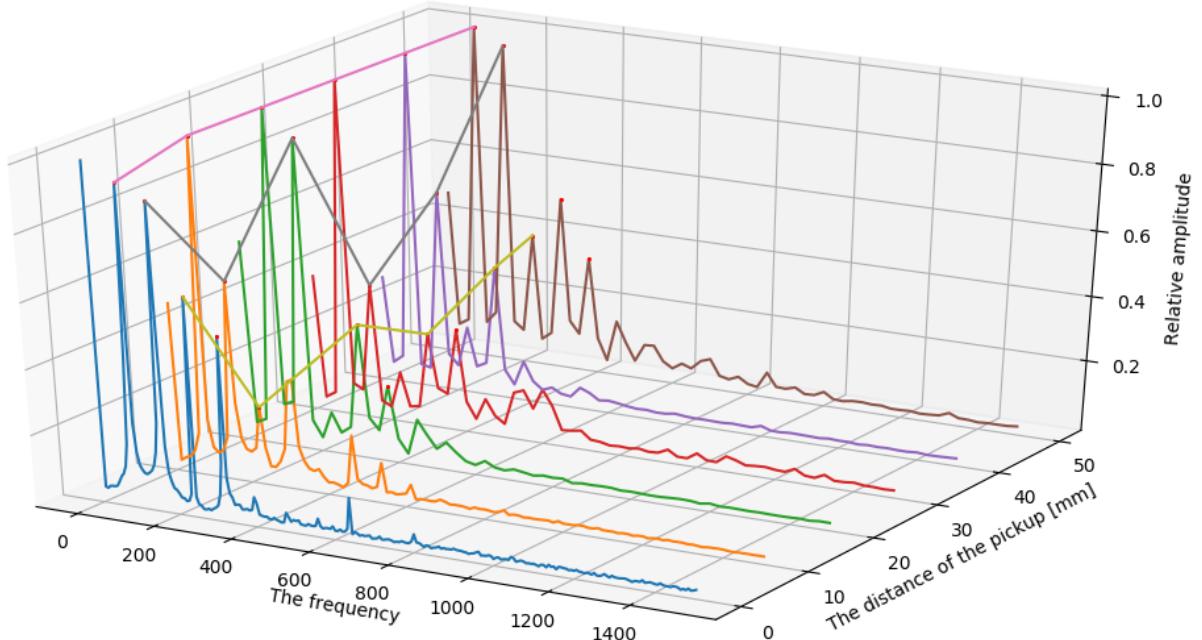


Figure 17: The graph of the relationship of the harmonic amplitudes and the pickup position with connected peaks.

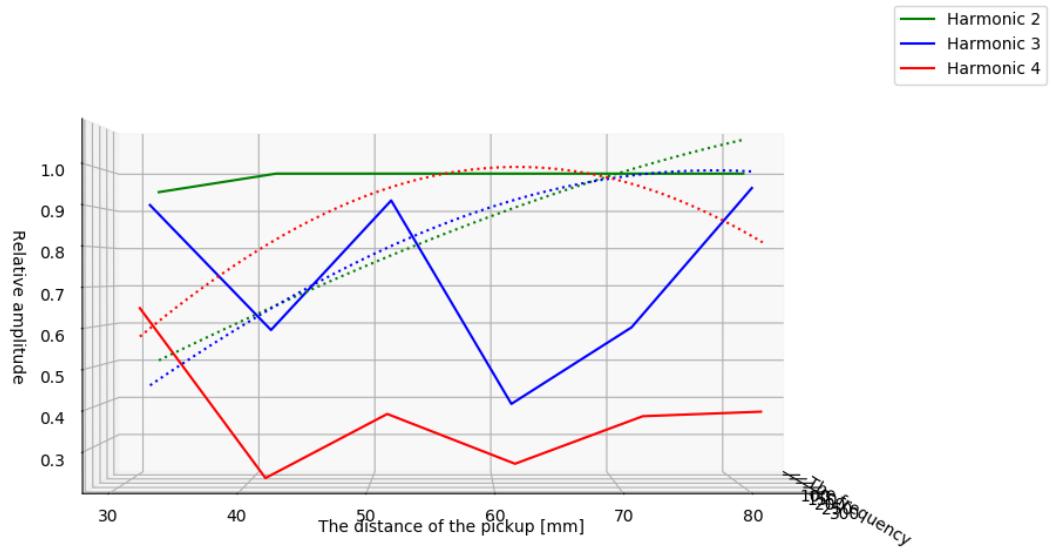


Figure 18: The relationship between the amplitudes of the second, third and fourth harmonics and the position of the pickup with the expected values

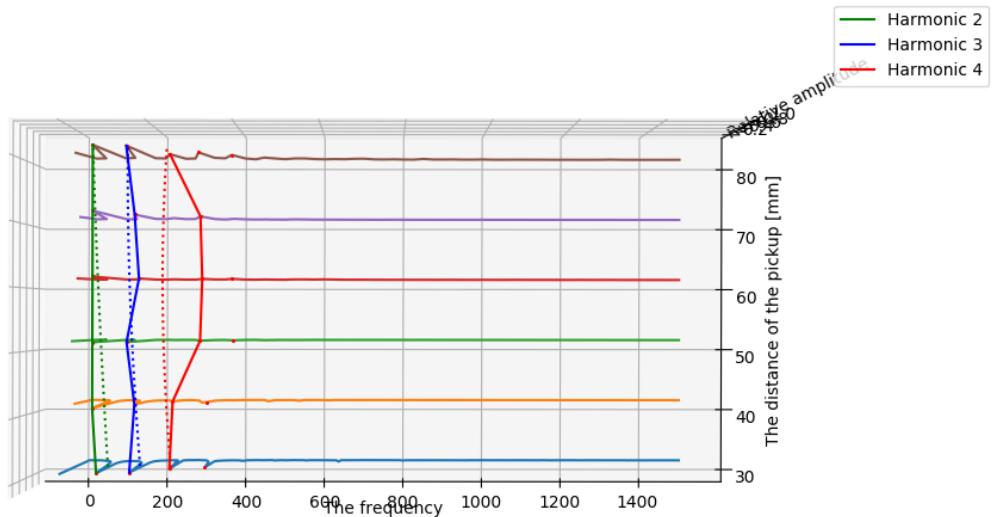


Figure 19: The relationship between the amplitudes of the second, third and fourth harmonics and the position of the pickup with the expected values as seen from a top-down perspective

Works cited

- Autodesk. *Autocad 2014*. Version 19.1, Autodesk, 2014.
- Chicago Music Exchange. *Fender Stratocaster Sunburst 1956*. https://images.reverb.com/image/upload/s--iiQ3rVKI--/a_exif,c_limit,e_unsharp_mask:80,f_auto,f_l_progressive,g_south,h_1600,q_80,w_1600/v1497644319/ax58jibpcrmys6id16hl.jpg. Accessed 29 Aug 2018.
- Donald Tillman, J. "Response Effects Of Guitar Pickup Position And Width". *Till.Com*, 2002, <http://www.till.com/articles/PickupResponse/>. Accessed 31 Aug 2018.
- E. Oliphant, Travis. *A Guide To Numpy*. Trelgol Publishing, 2006.
- Good Will Instrument Co., LTD. *GDS-1000-U Specifications*. Good Will Instrument Co., LTD, <https://www.gwinstek.com/en-global/products/downloadSeriesSpec/192>. Accessed 28 Aug 2018.
- Guitar Center. *Fender Special Edition HH Maple Fingerboard Standard Telecaster Sea Foam Pearl*. <https://www.guitarcenter.com/Fender/Special-Edition-HH-Maple-Fingerboard-Standard-Telecaster.gc>. Accessed 29 Aug 2018.
- Hunter, John D. "Matplotlib: A 2D Graphics Environment". *Computing In Science & Engineering*, vol 9, no. 3, 2007, pp. 90-95. *Institute Of Electrical And Electronics Engineers (IEEE)*, doi:10.1109/mcse.2007.55. Accessed 31 Aug 2018.
- Negri, Lucas. *PeakUtils*. Version 1.2.0, 2018.
- "Scale Length Explained". *Stewmac.Com*, 2018, https://www.stewmac.com/How-To/Online_Resources/Learn_About_Guitar_and_Instrument_Fretting_and_Fretw/Scale_Length_Explained.html. Accessed 31 Aug 2018.
- The Editors of Encyclopaedia Britannica. "Lorentz Force". *Encyclopædia Britannica*, 2017, <https://www.britannica.com/science/Lorentz-force>. Accessed 31 Aug 2018.
- Van Rossum, Guido, and Fred L. Drake. *Python 3 Reference Manual*. Paramount, CA, 2009, <https://dl.acm.org/citation.cfm?id=1593511>. Accessed 28 Aug 2018.
- Wolfe, Joe. "Helmholtz Resonance". *Newt.Phys.Unsw.Edu.Au*, <http://newt.phys.unsw.edu.au/jw/Helmholtz.html>. Accessed 31 Aug 2018.

Data processing code

waveform.py

```
import matplotlib.pyplot as plt
import numpy as np
import peakutils

class GuitarWaveform:
    """Represents a guitar waveform

    Arguments:
        array - An array containing the values of the oscilloscope read"""

    def __init__(self, array, fs, v_scale, h_scale, name):
        array = np.array(array)
        self._fs = fs
        self._samples_n = array.size
        self._times = np.linspace(0, self._samples_n / self._fs,
                                 self._samples_n)
        self._amplitudes = array * v_scale / 5
        self._name = name

    @property
    def name(self):
        return self._name

    @name.setter
    def name(self, name):
        self._name = name

    _dft = np.array([])
    def dft(self, force=False):
        if force or self._dft.size == 0:
            self._dft = np.fft.fft(self._amplitudes) / self._samples_n
            self._dft = self._dft[range(int(self._samples_n / 2))]
        return self._dft

    _abs_dft_freq = np.array([])
    _abs_dft = np.array([])
    def abs_dft(self, force=False):
        if force or not self._abs_dft.size or not self._abs_dft_freq.size:
            frq = np.arange(self._samples_n) * self._fs / self._samples_n
            self._abs_dft_freq = frq[range(int(self._samples_n / 2))]
            self._abs_dft = abs(self.dft())
        return self._abs_dft_freq, self._abs_dft

    _normalized_abs_dft_freq = np.array([])
    _normalized_abs_dft = np.array([])
    def normalized_abs_dft(self, force=False, cutoff_threshold=1500):
        if force or not self._normalized_abs_dft.any() or not \
           self._normalized_abs_dft.size:
            n_abs_dft = self.abs_dft()[1] / np.amax(self.abs_dft()[1])
            n_abs_dft_freq = self.abs_dft()[0]
```

```

        if cutoff_threshold:
            i = 0
            for val in n_abs_dft_freq:
                if val > cutoff_threshold:
                    break
                else:
                    i += 1
            self._normalized_abs_dft = n_abs_dft[0:i]
            self._normalized_abs_dft_freq = n_abs_dft_freq[0:i]

        else:
            self._normalized_abs_dft = n_abs_dft
            self._normalized_abs_dft_freq = n_abs_dft_freq

    return self._normalized_abs_dft_freq, self._normalized_abs_dft

_peaks = np.array([])
def ft_peaks(self, force=False, min_dist=10, thres=0.03):
    if force or not self._peaks.size:
        self._peaks = \
            peakutils.indexes(self.normalized_abs_dft(force=force)[1],
                              min_dist=min_dist, thres=thres)
    return self._peaks

def plot(self, x_unc=0, y_unc=0):
    if x_unc == 0 and y_unc == 0:
        plt.plot(self._times, self._amplitudes)
    else:
        plt.errorbar(self._times, self._amplitudes,
                     xerr=x_unc, yerr=y_unc, ecolor='r',
                     capsize=1.5, linewidth=1)

def plot_ft(self, length=100, ax=None, zs=0):
    x, y = self.abs_dft()
    if not ax:
        plt.plot(x[0:length], y[0:length])
    else:
        ax.plot(x[0:length], y[0:length], zs=zs, zdir='y')

def plot_normalized_ft(self, length=100, ax=None, zs=0):
    x, y = self.normalized_abs_dft()
    if not ax:
        plt.plot(x[0:length], y[0:length])
    else:
        ax.plot(x[0:length], y[0:length], zs=zs, zdir='y')

```

data_reader.py

```

import csv
from util import isfloat
from os import path
from waveform import GuitarWaveform

```

```

def read(f, positions=['0', '1', '2', '3', '4', '5'], strings=['E', 'A'],
         frets=['0', '3', '5', '7', '9', '12']):
    """Tries to read the data in f

If f is a file the data will be read from the file, otherwise if f is a
folder the program will try to read all csv files

Arguments:
f - folder or file
strings - The strings to analyze

Returns:
An array containing the data

TODO:
Implement the non-recursive behavior
"""

data = []

if path.isfile(f):
    with open(f, newline='') as csvfile:
        values = []

        reader = csv.reader(csvfile, delimiter=',')
        for row in reader:
            if not isfloat(row[0]):
                if 'Sampling_Period' in row[0]:
                    fs = 1 / float(row[1])
                elif 'Vertical_Scale' in row[0]:
                    v_scale = float(row[1])
                elif 'Horizontal_Scale' in row[0]:
                    h_scale = float(row[1])
                else:
                    continue
            else:
                values.append(int(row[0]))

    return GuitarWaveform(values, fs, v_scale, h_scale, f)

elif path.isdir(f):
    for i in positions:
        for j in frets:
            for k in strings:
                with open(path.join(f, i, j, k, 'DS0000.CSV'),
                          newline='') as csvfile:
                    values = []

                    reader = csv.reader(csvfile, delimiter=',')
                    for row in reader:
                        if not isfloat(row[0]):
                            if 'Sampling_Period' in row[0]:
                                fs = 1 / float(row[1])
                            elif 'Vertical_Scale' in row[0]:
                                v_scale = float(row[1])

```

```

        elif 'Horizontal_Scale' in row[0]:
            h_scale = float(row[1])
        else:
            continue
    else:
        values.append(int(row[0]))

    data.append(GuitarWaveform(values, fs, v_scale, h_scale,
                               f + i + '/' + j + '/' + k))

return data

```

util.py

```

def isfloat(value):
    try:
        float(value)
        return True
    except ValueError:
        return False

```

boiler.py

```

#!/usr/bin/env python
from mpl_toolkits.mplot3d import Axes3D
import sys
import data_reader
import matplotlib.pyplot as plt
import numpy as np

DMIN = 31.6 #nm
DERROR = 1.01 #nm
LENGTH = 100

UNC=False
CONNECT=False
CONNECT_N = 3
EXPECTED=False
PLOT=False

if __name__ == '__main__':
    if '-u' in sys.argv:
        UNC=True

    if '-c' in sys.argv:
        CONNECT=True

    if '-e' in sys.argv:
        EXPECTED=True

    if '-p' in sys.argv:
        PLOT=True

gtr_waves = data_reader.read(sys.argv[-1], strings=['E'], frets=['0'])
peaks_all = []

```

```

fig = plt.figure()
ax = fig.gca(projection='3d')

if PLOT:
    i = 0
    for gtr_wave in gtr_waves:
        x_s, y_s = gtr_wave.normalized_abs_dft()
        peaks = gtr_wave.ft_peaks(min_dist=0.001, thres=0.2)
        peaks_all.append(peaks)

        ax.plot(x_s, np.full(x_s.size, i) + D_MIN, y_s)

        for peak in peaks:
            ax.plot([x_s[peak]], [i+D_MIN], [y_s[peak]], marker='o',
                    color='r',
                    markersize=1.2)

    if UNC:
        for j in range(0, x_s.size):
            ax.plot([x_s[j], x_s[j]],
                    [i - D_ERROR, i + D_ERROR],
                    [y_s[j], y_s[j]],
                    marker='_', color='b')

    i += 10

if CONNECT:
    colors = ['g', 'b', 'r']
    for i in range(0, CONNECT_N):
        ax.plot([gtr_wave.normalized_abs_dft()[0][
                    gtr_wave.ft_peaks(min_dist=0.001, thres=0.2)[i]],
                    for gtr_wave in gtr_waves],
                    np.array([0, 10, 20, 30, 40, 50]) + D_MIN,
                    [gtr_wave.normalized_abs_dft()[1][
                    gtr_wave.ft_peaks(min_dist=0.001, thres=0.2)[i]],
                    for gtr_wave in gtr_waves],
                    color=colors[i], label=('Harmonic' + str(i+2)))

if EXPECTED:
    colors = ['g', 'b', 'r']
    offsets = [0.2, 0, 0]
    for i in range(2, CONNECT_N + 2):
        wav = gtr_waves[0]
        ax.plot(np.full(np.linspace(D_MIN, D_MIN + 50, num=100).size,
                        wav.normalized_abs_dft()[0][
                            wav.ft_peaks(min_dist=0.001, thres=0.2)[i-2]
                        ]),
                np.linspace(D_MIN, D_MIN + 50, num=100),
                [np.sin(i * np.pi * (val+D_MIN) / 648) + offsets[i-2],
                 for val in np.linspace(0, D_MIN + 50, num=100)],
                linestyle=':', color=colors[i-2])

ax.set_xlabel('The_frequency')

```

```
    ax.set_ylabel('The distance of the pickup [mm]')
    ax.set_zlabel('Relative amplitude')
    ax.legend()
    plt.show()
```

plot_ft.py

```
#!/usr/bin/env python

from data_reader import read
import matplotlib.pyplot as plt
import sys

gtr_w = read(sys.argv[-1])
gtr_w.plot_normalized_ft()
plt.xlabel('Frequency [Hz]')
plt.ylabel('Amplitude [V]')
plt.show()
```

plot.py

```
#!/usr/bin/env python

from data_reader import read
import matplotlib.pyplot as plt
import sys

gtr_w = read(sys.argv[-1])
gtr_w.plot(y_unc=0.002)
plt.xlabel('Time [s]')
plt.ylabel('Amplitude [V]')
plt.show()
```