

# Simulación de Una Guitarra a partir de archivos de Audio

Alfredo Ricci  
Juan Andrés Urrea

21 de julio de 2015



## 1. Introducción

Este proyecto pretende presentar la posibilidad de, a partir de una canción interpretada puramente en guitarra, simular las vibraciones correspondientes en cada cuerda de una guitarra que crearían el sonido escuchado en cada instante de la canción. Este objetivo final supone entonces el uso de herramientas vistas durante el curso como lo son, entre otras:

- La transformada de Fourier y sus asociadas en Python
- El módulo Matplotlib y sus funciones de graficación
- La realización de animaciones

Para lograr completar este proyecto se postulan objetivos parciales a desarrollar, de manera que se vaya avanzando progresivamente a construir el programa funcional. Cada uno será un paso cuyo funcionamiento individual representa una adición al proyecto global. Estos se muestran a continuación:

1. Importar y leer el contenido de un archivo **.wav**, el cual contiene la canción en si. Esta debe permitir tomar cualquier archivo de canción genérico y dejar su contenido disponible para edición dentro de Python.
2. Tomar dicho contenido del archivo **.wav** estudiado y obtener su transformada de Fourier discreta, de manera que sea posible determinar la frecuencia fundamental que lo caracteriza.

Lo anterior se realiza ya en el notebook que contiene el desarrollo del proyecto, sin embargo a continuación se muestran los distintos casos de análisis realizados utilizando la herramienta creada para así mostrar la aplicabilidad de los principios de los que se parte.

### 1. Una Nota

El primer caso de estudio se basa en un archivo de audio que contiene una sola nota, tocada durante un intervalo de tiempo de 2 segundos. Esta es la prueba más básica para determinar la efectividad de las herramientas creadas en el notebook para lograr identificar la frecuencia fundamental que define dicha nota y así poder "interpretar" dicha frecuencia durante ese intervalo de tiempo en la guitarra simulada.

### 2. Dos Notas distanciadas por un Tiempo

Este caso de estudio consiste en un archivo de audio que contiene 2 notas distanciadas por un intervalo de tiempo de 2 segundos, teniendo una duración de 2 segundos cada una. Para esto se empieza a tomar en cuenta el concepto del avance del tiempo y la aparición y desaparición de frecuencias a tener en cuenta. Para esta situación, se divide el tiempo entonces en arreglos, de manera que en cada arreglo exista únicamente una nota, para poder identificar la frecuencia fundamental y conocer la nota que debe ser tocada durante cada intervalo de tiempo.

### 3. Escala de Notas

Este caso representa una generalización de una sucesión de notas independientes tocadas una tras otra a lo largo del tiempo. El procedimiento de análisis es igual al descrito en el caso anterior. Este caso se toma en cuenta para poder mostrar la efectividad de las herramientas creadas para analizar notas independientes tocadas en un intervalo de tiempo.

### 4. Pulso de 2 Notas

Este caso viene a representar la situación de dos notas distintas tocadas al mismo tiempo. Dado esto, se deben identificar 2 frecuencias fundamentales que están siendo tocadas. Para analizarlo, se tuvo entonces que lograr obtener dichas frecuencias determinando no una sino 2 frecuencias notables en la transformada de Fourier del pulso. Dado que no es posible descomponer dicho pulso en las notas que lo componen, se analiza unido. Esto es posible ya que ambas notas se tocan durante el mismo intervalo de tiempo, por lo que el uso de la función fourier, descrita y documentada en el desarrollo del código, permitió identificar las 2 frecuencias fundamentales del pulso.

### 5. Dos pulsos de 2 notas

Este caso representa un análisis conjunto de las condiciones de avance temporal y la ejecución simultánea de dos notas. De esta forma se busca mostrar la efectividad de las herramientas construidas para determinar las 4 frecuencias fundamentales que deben simularse y el intervalo de tiempo durante el cual se deben ejecutar. En este procedimiento se combinan los pasos realizados para el análisis de notas individuales y el pulso compuesto de 2 notas, de manera que se divida el arreglo en los intervalos de sonido y las frecuencias que deben tocarse simultáneamente, una por cada cuerda.

### 6. Acorde

En este caso se presenta una situación similar al pulso, tan solo que se tienen en cuenta 3 notas. A partir de esto, se utilizan las herramientas creadas para encontrar las frecuencias fundamentales que definen las notas que componen el acorde, para poder simular tocarlas simultáneamente en cada cuerda de la guitarra.

Todos estos casos anteriores generan como resultado, después de su análisis, una matriz de  $N \times 3$ , siendo  $N$  el número de notas tocadas en cada intervalo de tiempo. Esto luego se utiliza dentro del código que define la simulación de la guitarra para visualizar las vibraciones de cada cuerda durante cada instante de tiempo.

Se eligió esta temática como proyecto principalmente por dos motivos. Por un lado, permitía la utilización de varias de las herramientas aprendidas a lo largo del curso que consideramos más interesantes, como lo son la realización de animaciones o el análisis de frecuencias por medio del uso de la transformada de Fourier. Esto debido a nuestro doble programa con ingeniería eléctrica y electrónica, donde el tratamiento de señales puede llegar a ser de especial importancia, por lo que la realización de este proyecto podría significar llegar a cursos posteriores con un entendimiento más profundo de este tema. Por otro lado, la orientación hacia una temática musical vino dada por el interés que nos generaba poder aplicar lo aprendido en el curso a un ámbito menos científico y más creativo, de manera que se pudiera abordar Python de manera más recreacional". A pesar de que la realización de todo el código sí requería conocimiento medianamente amplio de Python, su finalidad no está orientada hacia un fin científico, y permite ver el lado amigable y entretenido de lo que se puede realizar con un poco de empeño en Python.

## 2. Marco Teórico

### 2.1. Ecuación de onda en una dimensión y ondas estacionarias

La guitarra es un instrumento que basa su funcionamiento acústico en el principio de ondas estacionarias. Las ondas estacionarias, son una solución a la ecuación de onda, con condiciones de frontera fijas. Es decir:

$$\frac{\partial^2 \psi}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 \psi}{\partial t^2} \quad (1)$$

$$\psi(0) = \psi(L) = 0 \quad (2)$$

$$\left. \frac{\partial \psi}{\partial t} \right|_{x=0} = \left. \frac{\partial \psi}{\partial t} \right|_{x=L} = 0 \quad (3)$$

Al solucionar esta ecuación se obtiene que:

$$\psi(x, t) = \cos(\omega t) \sin(kx) \quad (4)$$

En donde  $k = \frac{2\pi}{\lambda}$  y  $\omega = 2\pi f$ .

De este modo si se conoce la frecuencia es posible y la longitud de onda que para el primer armónico (sobre el cual se estará trabajando) es posible generar una animación de la cuerda moviéndose en la guitarra. Es importante saber también que para ondas estacionarias se cumple que:

$$f = \frac{nc}{2L} \quad (5)$$

En donde  $n$  es el número de nodos (se trabajara con  $n=1$  en todo momento),  $c$  es la velocidad de la onda, y está dada por  $\sqrt{\frac{T}{\mu}}$ , es decir que la velocidad de la onda, solo depende de la configuración de la cuerda. Finalmente  $f$  y  $L$  son, la frecuencia de la onda, y la longitud de la cuerda respectivamente, que son precisamente las variables a relacionar. [1]

Las velocidades encontradas en las cuerdas de una guitarra son:



$$v1 = 424,59 \frac{m}{s} \quad (6)$$

$$v2 = 318,63 \frac{m}{s} \quad (7)$$

$$v3 = 252,59 \frac{m}{s} \quad (8)$$

$$v4 = 189,29 \frac{m}{s} \quad (9)$$

$$v5 = 141,90 \frac{m}{s} \quad (10)$$

$$v6 = 106,31 \frac{m}{s} \quad (11)$$

## 2.2. Transformada de Fourier

La transformada de Fourier es una herramienta matemática altamente utilizada en la física e ingeniería Electrónica, ya que permite separar cualquier señal en las señales que la componen en el dominio de la frecuencia. De este modo, si se tiene un sonido, que es creado por varios sonidos más pequeños como es el caso de la guitarra, se puede aplicar la transformada de Fourier para encontrar las frecuencias de estos sonidos y así poder animarlas usando la teoría anterior. Cabe resaltar que su implementación se pudo realizar en Python gracias a la función `fft()`, la cual proporciona la transformada de Fourier discreta de un arreglo numérico cualquiera. A pesar de que existe la transformada discreta y continua, se emplea la discreta pues así se define un arreglo.

De manera analítica, la transformada de Fourier  $f(\omega)$  de una función en el dominio del tiempo  $f(t)$  viene definida como:

$$f(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \quad (12)$$

Con esta operación se pasa del dominio del tiempo al dominio de la frecuencia una función continua. La forma genérica de esta transformada toma todos los reales como límites de integración, pues se asume la función definida en un valor para todo  $t$ . [4] Para el caso del estudio de arreglos compuestos de elementos

no continuos, fue necesario utilizar la transformada de Fourier discreta, que viene definida para un arreglo discreto  $f(t)$  de tamaño  $N$  como [5]:

$$f(k) = \sum_0^{N-1} f(t) e^{-\frac{2\pi k i t}{N}} \quad (13)$$

Es esta definición la que la función `fft()` utiliza para procesar los arreglos analizados.

### 3. Conclusiones

Luego de la realización completa del notebook de este proyecto, fue posible concluir acerca de los siguientes aspectos:

- Las herramientas de Python, como lo son aquellas de graficación en el entorno de Matplotlib o el análisis de frecuencias con las transformadas de Fourier disponibles en el entorno de Scipy, son efectivamente útiles y suficientes para un análisis exitoso de archivos de audio en formato wav, de manera que se puede visualizar el contenido de estos de, manera gráfica, para luego proceder a realizar modificaciones a partir de este.
- Durante la identificación de las frecuencias fundamentales de un pulso compuesto de varias notas tocadas simultáneamente se halló una dificultad notable pues, a pesar de que se conocían el número de notas que componían el pulso, la condición de grabación de los mismos archivos de audio y posibles imperfecciones de procesamiento dieron lugar a que existieran frecuencias aparte de las fundamentales, y que estas fueran tan notables como las fundamentales, por lo que analizar el arreglo en el dominio de la frecuencia se vio severamente sesgado por este aspecto. esto ocurrió principalmente cuando se trabajó con la nota "La".
- Enfatizando en la conclusión anterior, en el caso de análisis de un acorde, compuesto de 4 notas tocadas simultáneamente, resultó imposible, con las herramientas construidas y utilizadas, determinar las 4 frecuencias fundamentales. Siguiendo el procedimiento que se había establecido para analizar los pulso de notas solo fue posible identificar 3 de las 4 frecuencias fundamentales. Para remediar este problema, llegaría a ser conveniente mejorar las condiciones de grabación de los archivos de audio, o utilizar unos que presenten una mayor gama de frecuencias, de manera que las frecuencias ruido presentes sean menos significativas y no limiten tan drásticamente el análisis realizado.
- Haciendo referencia a la orientación recreativa que se quería dar a este proyecto, se podría afirmar que este objetivo se cumplió satisfactoriamente, pues las animaciones producidas como resultado permiten visualizar de manera simple a la vista la simulación de una guitarra. A pesar de que la complejidad del código programado sigue presente en el trasfondo, el resultado final permite que usuarios no muy familiarizados a este entorno de programación, o de la programación en general, vean usos interesantes que se pueden lograr. Esta simplicidad aparente y el enfoque musical y no científico que tomó este proyecto podría catalogarse como una de sus ventajas en originalidad, pues se aleja del estigma que se da generalmente a la computación científica y permite ver a Python, y la programación en general, de manera más amigable.
- Vale la pena también mencionar que algunas de las animaciones obtenidas como resultado, aquellas que reflejaban el comportamiento de las cuerdas para la creación de pulsos de notas, presentaron fallas que, aunque no fatales, si se notaban. Estas fueron principalmente la no simulación de todos los pulsos en la secuencia de estos, de manera que solo se logró simular exitosamente la reproducción de un solo pulso en una secuencia de 2. Ya que no fue posible encontrar solución a estos problemas antes del máximo plazo de entrega, se sugiere trabajo posterior, específicamente ampliando el trasfondo teórico musical y un análisis más detallado del código y su comportamiento con distintos pulsos como parámetro de entrada, de manera que se pueda solucionar el problema encontrado para una secuencia de pulsos. Aún así, las animaciones resultantes siguen cumpliendo el propósito recreativo y simulan de manera aceptable los datos que se analizan correctamente del archivo de audio que no presentan el error mencionado.

## 4. Bibliografía y Material Utilizado

1. **Ecuaciones de Onda** VaxaSoftware. Documento en PDF obtenido de  
[http://www.vaxasoftware.com/doc\\_edu/fis/ondasx.pdf](http://www.vaxasoftware.com/doc_edu/fis/ondasx.pdf)
2. **Imagen de Joseph Fourier** Obtenida de Wikipedia.  
[http://www.vaxasoftware.com/doc\\_edu/fis/ondasx.pdf](http://www.vaxasoftware.com/doc_edu/fis/ondasx.pdf)
3. **Imagen de Stitch** Obtenida de Deviant Art.  
<http://linnmicphoto.deviantart.com/art/Stitch-as-Elvis-181414142>
4. **Transformada de Fourier** MATLAB. Obtenido de  
[http://www.sc.ehu.es/sbweb/energias-renovables/MATLAB/simbolico/fourier/fourier\\_1.html](http://www.sc.ehu.es/sbweb/energias-renovables/MATLAB/simbolico/fourier/fourier_1.html)
5. **Transformada de Fourier Discreta** Obtenido de  
<http://www.ehu.eus/Procesadodesenales/tema7/ty2.html>
6. **Material de La Clase Métodos Computacionales** Periodo 2015-19. Disponible en el Github de Juan David Lizarazo.