Instituto Tecnológico de Buenos Aires

22.05 Análisis de Señales y Sistemas Digitales

Trabajo práctico 2: Análisis y Síntesis de Audio Digital

Grupo 1

Máspero, Martina	57120
Mestanza, Joaquín Matías	58288
Nowik, Ariel Santiago	58309
Parra, Rocío	57669
Regueira, Marcelo Daniel	58300

Profesores

Jacoby, Daniel Andrés
Iribarren, Rodrigo Iñaki
Belaustegui Goitia, Carlos

Presentado: 03/05/2019

Índice

1. FFT

1.1. Enfoque

Se decidió desarrollar un algoritmo FFT, ya que se contó con conocimiento del campo de ciencias de la computación para resolver problemas de programación dinámica. Se mostrará la derivación de un algoritmo menos eficiente pero más entendible, y luego se describirán sus posteriores optimizaciones. En el resultado final se llegó a un algoritmo con $\frac{n}{2}log_2 n$ multiplicaciones, $6n$ memoria, en otras palabras, $\Theta(nlog(n))$ tiempo, $\Theta(n)$ memoria.

1.2. El problema

El cálculo de la DFT consiste en producir una la salida $X[k]$ de coeficientes mediante la entrada $x[n]$, ambos arreglos de igual longitud N. La fórmula que describe la DFT que utilizamos es:

$$DFT[X[n]] = X[k] = \sum_{n=0}^{N-1} x(n)e^{-i2\pi kn/N} \tag{1}$$

Se necesita resolver la DFT en tiempo subcuadrático, es decir en menos de $\Theta(n^2)$, que es la solución trivial.

1.3. Primera observación de interés

Si la longitud de la sucesión N es par se puede escribir $X[k]$ como:

$$DFT[X[n]] = X[k] = \underbrace{\sum_{n=0}^{\frac{N}{2}-1} x(2n)e^{-i2\pi kn/(N/2)}}_{DFT[X[2n]=Y[n]]} + e^{i2\pi k/N} \overbrace{\sum_{n=0}^{\frac{N}{2}-1} x(2n+1)e^{-i2\pi kn/(N/2)}}^{DFT[X[2n+1]=Z[n]]} \tag{2}$$

$$DFT[\underbrace{X[n]}_{0\leq n<N}] = DFT[\underbrace{Y[n]}_{0\leq n<N/2}] + e^{i2\pi k/N} DFT[\overbrace{Z[n]}^{0\leq n<N/2}] \tag{3}$$

Donde $X[2n] = Y[n]$, $X[2n+1], Y[2n+1]$ son sub-sucesiones de $X[n]$ tomando los índices pares e impares respectivamente. Podemos decir que descompusimos el problema de la DFT como dos problemas de DFT con sucesiones de la mitad de tamaño, es decir, conseguimos una solución recursiva. Es importante observar que la única forma de que la solución sea viable es que N sea una potencia de dos, para poder garantizar que al dividir los subproblemas siempre N_i sea par.

1.4. Algoritmo conceptual

Si bien la formulación anterior da lugar a la implementación de un algoritmo recursivo "top-down" se procedió a desarrollar una fórmula "bottom-up" para resolver el problema iterativamente, y por lo tanto provocar que la ejecución tenga una constante menor.

Se definió el subproblema $DFT[i][j][k]$ como el k-ésimo coeficiente, del subarreglo de longitud 2^i que comienza en el casillero j. Se puede mostrar mediante la ecuación 3 en esta formulación es equivalente a

$$DFT[i][j][k] = \begin{cases} DFT[i-1][j][k\,\%2^{i-1}] + W[2^i][k]DFT[i-1][j+N/2^i][k\,\%2^{i-1}] & \begin{array}{l} 1 \leq i \leq log_2 N \\ 0 \leq j < N/2^i \\ 0 \leq k < 2^i \end{array} \\ A[j] & \begin{array}{l} i = 0 \\ k = 0 \end{array} \end{cases} \tag{4}$$

Donde $W[N][k] = e^{-i2\pi k/N}$. Con esta precisa formulación se puede implementar con 3 sentencias *for* un algoritmo de $O(nlogn)$ tiempo y $O(n^2logn)$ memoria.

1.5. Optimizaciones de memoria

Se presentan dos modificaciones superficiales y técnicas, no conceptuales, que no alteran la lógica escencial del algoritmo, pero que lo vuelven práctico.

1.5.1. Optimización 1

Se necesita reducir el orden de memoria de $\Theta(n^2)$, el cual es muy limitante. La primera optimización es mediante la observación de que para calcular $DFT[i]$ solo necesitamos tener previamente calculado $DFT[i-1]$. Por lo tanto podemos calcular en un arreglo el DFT siguiente y en la posterior iteración escribir de dicho arreglo al anterior, "yendo y volviendo". De esta forma se reduce el uso de memoria a $\Theta(n^2)$. Este valor todavia no es conveniente, ya que la expectativa es que tanto el tiempo como la memoria del algoritmo sean subcuadráticos.

1.5.2. Optimización 2

La observación necesaria para reducir el uso de memoria es que, en cada iteración $jk < N$, por lo que en realidad es posible almacenar todos los valores en un arreglo de longitud N (es decir, "comprimiendo" la matriz). Como el número de filas y columnas de la matriz es variable para cada valor de i es necesario implementar una lógica que se adapte a este hecho ($mat[i][j] = arr[i * colsize + j]$). De esta forma se redujo la memoria del algoritmo a $\Theta(n)$.

1.6. Optimizaciones de tiempo

Se mostrarán dos optimizaciones de tiempo que si bien no mejoran la complejidad del algoritmo en tiempo (de hecho, no son conocidos algoritmos mejores que $\Theta(nlogn)$), reducen, en una constante, su demora.

1.6.1. Cálculo $W[n][k] = e^{-i2\pi k/n}$

Calcular cada vez que se necesitan dichos factores si bien no modificaría la complejidad del algoritmo, si impacta en el tiempo total. Para ello se precalcula en el inicio del algoritmo $W[k] = e^{-2i\pi k/N}$ con el N más grande. Como vale que:

$$W[n][k] = e^{-i2\pi K/n} = e^{-i2\pi K/N*N/n} = W[K*N/n] = W[K*N/2^i] = W[K*N/2^i \,\%N] \tag{5}$$

Por lo tanto calculando inicialmente $W[k]$ con $0 \leq k < N$ ya se tiene información para obtener en una sola operación la exponencial correspondiente cuando sea necesario.

1.6.2. Optimización Butterfly

Otra optimización importante que permitirá reducir a la mitad la cantidad de multiplicaciones complejas que realiza el algoritmo se denomina optimización "Butterfly". Consiste en observar dos propiedades:

$$W[n][k] = -W[n][k+n/2] \tag{6}$$

$$DFT[..][..][k\,\%2^{i-1}] = DFT[..][..][(k+n/2)\,\%2^{i-1}] \tag{7}$$

La forma de utilizarla consiste en, computar en paralelo las siguientes expresiones:

$$DFT[i][j][k] = DFT[i-1][j][k\,\%2^{i-1}] + W[2^i][k]DFT[i-1][j+N/2^i][k\,\%2^{i-1}] \tag{8}$$

$$DFT[i][j][k+n/2] = DFT[i-1][j][k\,\%2^{i-1}] - W[2^i][k]DFT[i-1][j+N/2^i][k\,\%2^{i-1}] \tag{9}$$

Realizando el producto $W[2^i][k]DFT[i-1][j+N/2^i][k\,\%2^{i-1}]$ una sola vez.

2. Conclusión

Se llegó a un algoritmo muy optimizado ($6n$ memoria, $\frac{n}{2}log_2 n$ tiempo) con una complejidad equivalente a las de los más utilizados. El código combinado con todas las optimizaciones es díficil de leer porque consiste en la combinación de todos los detalles técnicos descriptos juntos. Si bien es posible optimizar aun más la memoria utilizando un método in-place que no utilice arreglos adicionales para almacenar datos parciales, como de todos modos es necesario utilizar un arreglo de N elementos para almacenar $W[k]$, dicha técnica no reduciría asintóticamente el uso de memoria adicional. El algoritmo final demoró, en una computadora moderna del 2015 con Windows 10 unos 127ms en calcular 100 FFT's de 4096 puntos. Fue verificado su correcto funcionamiento en comparación con el resultado de librerias de Python que calculaban FFT.

3. Programa principal

4. Síntesis aditiva de sonidos

5. Síntesis mediante modulación en frecuencia

Introducción teórica

La modulación en frecuencia esta dada por:

$$x(t) = A(t)cos(2\pi f_c t + I(t)cos(2\pi f_m t + \phi_m) + \phi_c)$$

Parámetros de la modulación en frecuencia:

- f_c = frecuencia de la señal portadora
- f_m = frecuencia de la señal modulante
- f_d = desviación de la frecuencia (respecto de la portadora)
- I = $\frac{f_d}{f_m}$ = índice de modulación

Por simple inspección se puede ver que si I = 0 la desviación de la frecuencia (f_d) debe ser 0 y por ende no hay modulación en frecuencia. Un caso más interesante es cuando I es mayor que 0, se puede observar que aparecen nuevas componentes espectrales lo que provoca que la energía de la señal sea redistribuida.

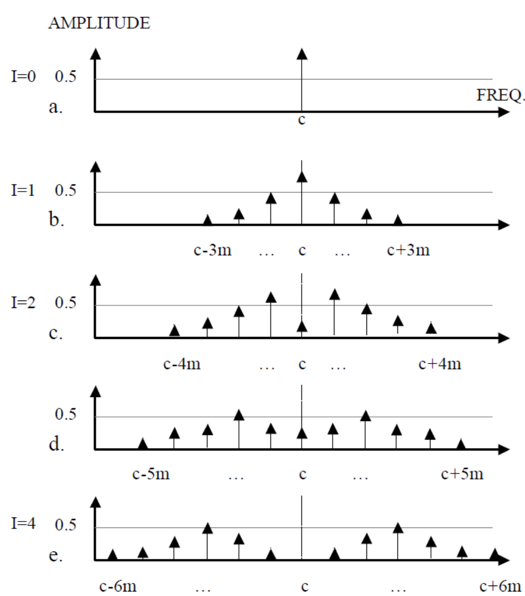

Figura 1: Nuevo contenido espectral con I $\neq$ 0 (c = f_c y m = f_m)

Las amplitudes de la portadora y las componentes laterales están dadas por las funciones de Bessel de primera especie y orden n-ésimo, donde el argumento de la función es el índice de modulacion I (J_n(I)).
En general, $J_n(1)$ representa un coeficiente de escalamiento de amplitud:

$J_0(1) \longleftarrow$ para la portadora

$J_1(1) \longleftarrow$ para las primeras bandas laterales

$J_2(1) \longleftarrow$ para las segundas bandas laterales

Y así sucesivamente. Mientras más grande sea el orden de la banda lateral se necesita más índice de modulación para que la misma tenga una amplitud significativa.

$$BW \approx 2(f_d + f_m)$$

Expresamos a x(t) con funciones de Bessel:

$$\begin{aligned} x(t) = A(t) \cdot \{ \ J_0(I)sen(2\pi f_c t) \\ +J_1(I)[sen(2\pi(f_c+f_m)t) - sin(2\pi(f_c-f_m)t)] \\ +J_2(I)[sen(2\pi(f_c+2f_m)t) + sin(2\pi(f_c-2f_m)t)] \\ ... \\ +J_n(I)[sen(2\pi(f_c+nf_m)t) \pm sin(2\pi(f_c-nf_m)t)] \ \} \end{aligned}$$

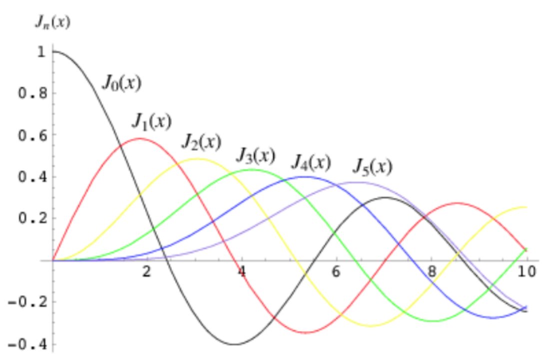

Figura 2: Funciones de Bessel de primer especie desde J_0 hasta J_5

Se puede ver que las bandas laterales bajas de orden impar poseen un signo negativo. Además se sabe que para para un I>2.5 las funciones de Bessel producen coeficientes de escalamiento negativos para algunas componentes. En general se suelen ignorar los signos negativos en el espectro ya que solo indican una inversión de la fase de la frecuencia correspondiente. En la proxima sección se verá por qué en realidad si son importantes esas inversiones de fase.

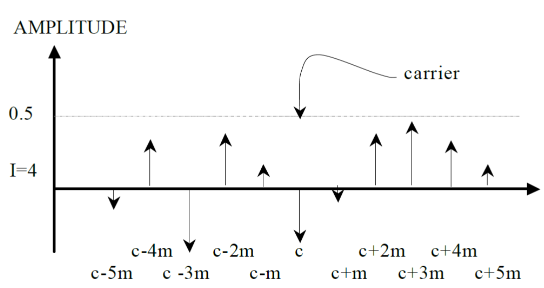

Figura 3: Espectro de ejemplo con I=4 e inversiones de fase

Frecuencias laterales reflejadas

La riqueza de la técnica de modulación en frecuencia yace en que el espectro ubicado en la parte negativa del dominio interfiere con la parte positiva dando lugar a una mezcla de componentes en ambas partes del espectro. La relación entre frecuencias f_c y f_m es la que da origen a un espectro armónico o uno inarmónico.

Espectro armónico e inarmónico

La relación entre frecuencias esta dada por

$$\frac{f_c}{f_m} = \frac{N_1}{N_2}$$

$$f_0 = gcd(f_c, f_m)$$

La posición de las frecuencias laterales en serie armónica puede ser determinada por las siguientes relaciones:

$$k = N_1 \pm nN_2 \; para \; n = 0, 1, 2, 3, 4...$$

Donde k es el número de armónico y n es el orden de la frecuencia lateral. Excepto para $n = 0$, k toma dos valores por cada orden.

A continuación dejamos unas generalizaciones útiles:

- La portadora es siempre el N_1-ésimo armónico en la serie
- Si $N_2 = 1$ el espectro contiene todos los armonicos y el fundamental es la frecuencia de la modulante
- Si N_2 es un numero par el espectro tiene solo armónicos impares
- Si $N_2 = 3$ cada tercer armónico no se encuentra más en la serie

El número de armónicos que tienen amplitud significativa dependen del índice de modulación.

Para relaciones entre frecuencias bajas e indices chicos donde $N_1 \neq 1$ la fundamental puede no estar presente en el espectro.

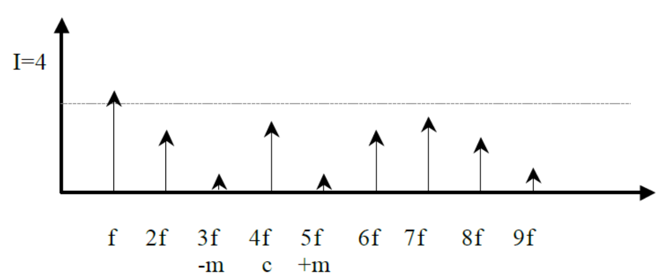

Figura 4: Espectro donde la relacion $\frac{f_c}{f_m} = \frac{4}{1}$ (armónico) con I=4

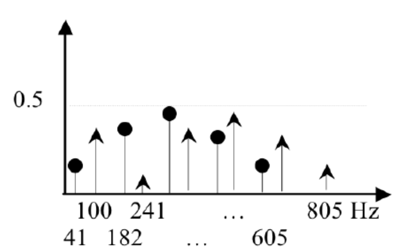

Figura 5: Espectro donde la relacion $\frac{f_c}{f_m} = \frac{1}{\sqrt{2}}$ (inarmónico) con I=4

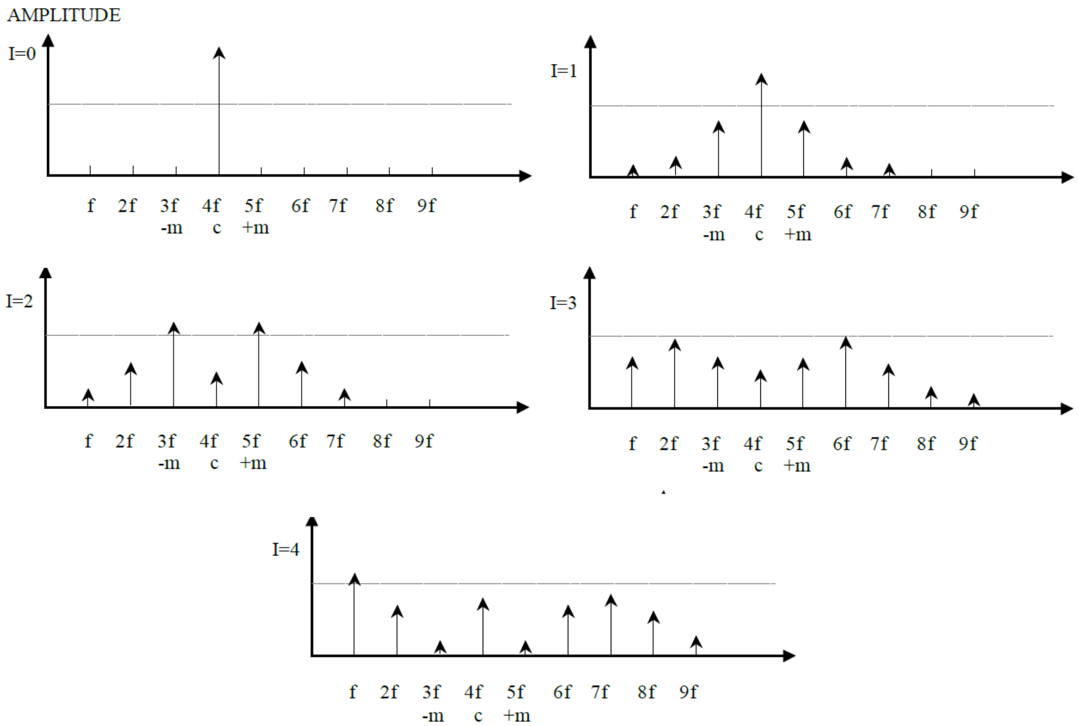

Figura 6: Espectros con distintos índices de modulación (cociente racional)

En resumen:
El espectro inarmónico se origina cuando el cociente $\frac{f_c}{f_m}$ da como resultado un número irracional.
El índice de modulación $\frac{f_d}{f_m}$ determina el número de componentes que van a tener amplitud significativa.

5.1. Espectro dinámico

La complejidad del espectro esta relacionada con el índice de modulación de forma tal que si el índice crece el bandwidth también crece. Entonces si pensamos en un índice de modulación que varía con el tiempo la evolución del bandwidth del espectro puede estar generalmente descripta por la forma de la función. No obstante, la evolución de cada componente es determinado por la forma de la funcion de Bessel correspondiente. Entonces, si el indice de modulación aumenta con el tiempo el bandwidth también lo hace, pero un componente del espectro va a crecer o decrecer en amplitud dependiendo de la pendiente de la función de bessel en ese rango de indices.

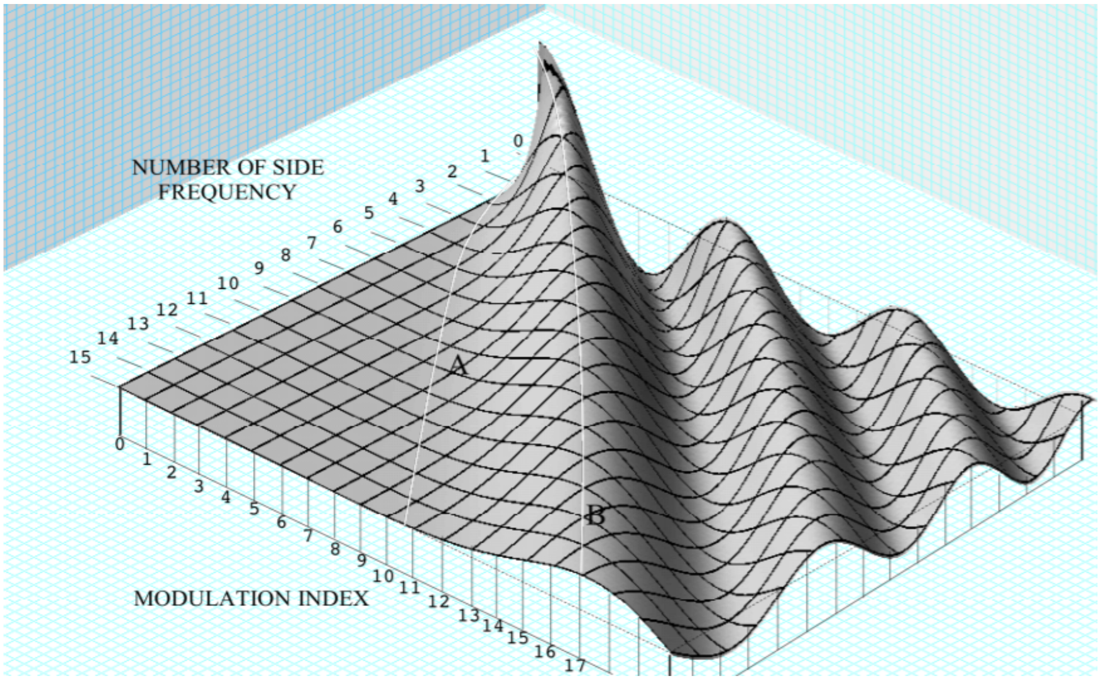

Figura 7: Funciones de Bessel desde J_0 hasta J_{15} e I desde 0 hasta 20. En este gráfico se puede ver rápidamente el bandwidth dado un índice I

5.2. Síntesis para instrumentos de viento

Para los intrumentos de viento:

$$\phi_m = \phi_c = -\frac{\pi}{2}$$

Resultando

$$x(t) = A(t) \cdot sen(2\pi f_c t + I(t) \cdot sen(2\pi f_m t))$$

5.2.1. Síntesis de clarinete

Para poder sintetizar un clarinete se establecieron los parámetros recomendados por la literatura:

$$\frac{f_c}{f_m} = \frac{2}{3}$$

Las envolventes sugeridas por la bibliografía poseen la siguiente forma:

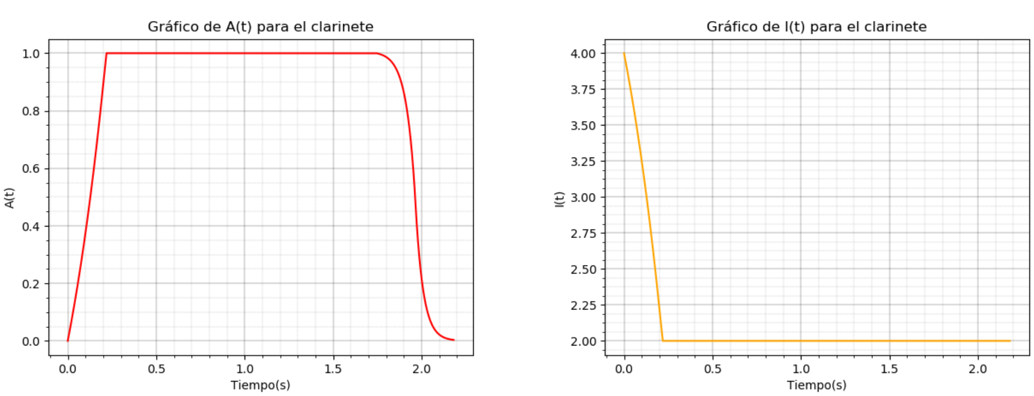

Figura 8: A(t) e I(t) respectivamente

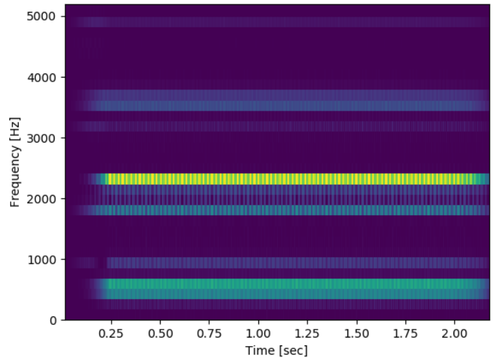

Figura 9: Espectrograma de la síntesis del clarinete a 440Hz (nota LA)

5.2.2. Síntesis de campana

Parámetros recomendados por la literatura para la síntesis:

$$\frac{f_c}{f_m} = \frac{2}{1}$$

Envolventes sugeridos:

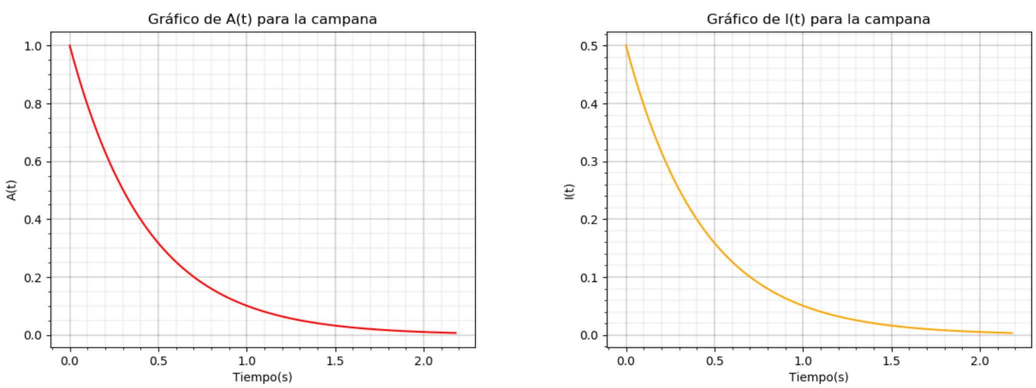

Figura 10: A(t) e I(t) respectivamente

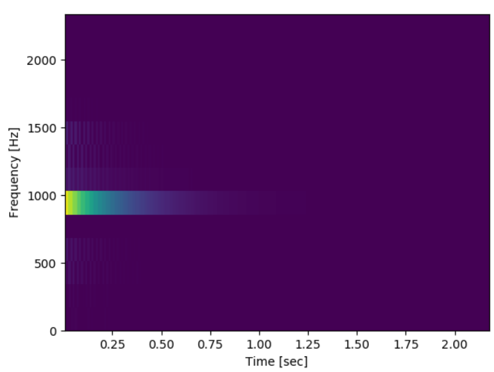

Figura 11: Espectrograma de la síntesis de la campana a 440Hz (nota LA)

5.2.3. Síntesis de trombón

Para los Brass Instruments se recomienda utilizar una relación de:

$$\frac{f_c}{f_m} = \frac{1}{1}$$

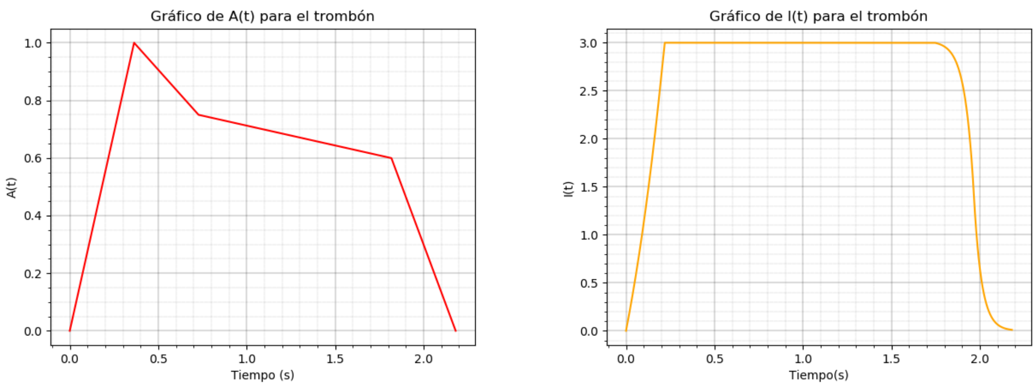

Figura 12: A(t) e I(t) respectivamente

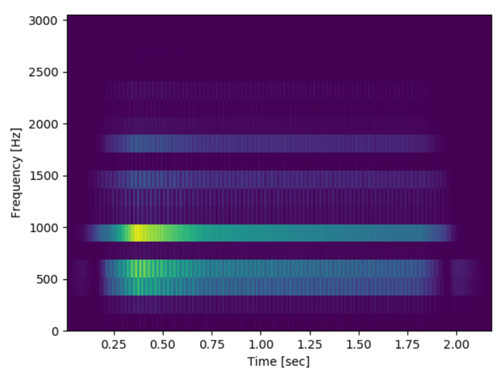

Figura 13: Espectrograma de la síntesis del trombón a 440Hz (nota LA)

5.3. Conclusiones

Para poder realizar una síntesis de un instrumento suele ser útil tener espectrogramas del mismo ya que nos dan información de cómo evoluciona el sonido con el tiempo y de esa forma se puede llegar a tener una idea de cómo puede llegar a ser A(t) e I(t). Si bien en la literatura hay envolventes sugeridas, las mismas son sólo un molde inicial por el cual uno debería empezar a probar la síntesis pero para poder ejecutar una síntesis apropiada se debe cambiar el molde. De la misma forma la relación entre f_m y f_c no siempre tiene que ser la sugerida.

Referencias

[1] The Synthesis of Complex Audio Spectra by Means of Frequency Modulation,
`https://ccrma.stanford.edu/sites/default/files/user/jc/fm_synthesispaper-2.pdf`

[2] FM Synthesis for Musical Instruments: Bells and Clarinets,
`http://home.eng.iastate.edu/~julied/classes/ee224/Labs/FMSynthesis_lab.pdf`

6. Síntesis de sonidos mediante modelos físicos

Se analizarán dos variantes del modelo Karplus-Strong en paralelo, tanto de manera teórica como práctica.

6.1. Bloque A Elemental

Se resolverá, como cálculo auxiliar un bloque sencillo definido como:

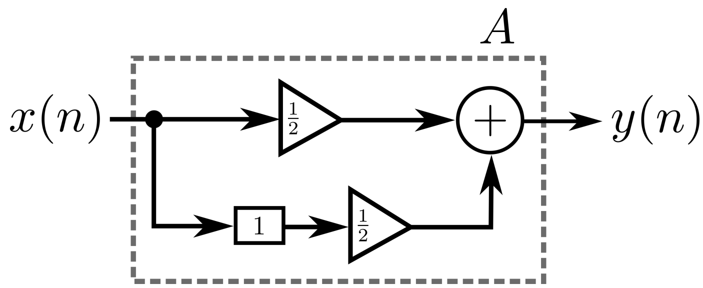

Figura 14: Bloque elemental

Este bloque solo promedia los dos ultimos valores de entrada. Su transferencia esta dada por

$$x(n) = \frac{1}{2}x(n) + \frac{1}{2}x(n-1) \tag{10}$$

$$Y(z) = \frac{1}{2}X(z) + \frac{1}{2}X(z)z^{-1} \implies A(z) = \frac{z+1}{z} \tag{11}$$

Se puede observar que el bloque A es pasa-bajos (cero en $Z = -1$); lo cual es, en principio, razonable, el bloque A suaviza la entrada.

6.2. Karplus Strong 1

6.2.1. Análisis teórico

Se resolverá un nuevo sistema, denominado S_1, el cuál consiste en una adicion de realimentación al sistema anterior.

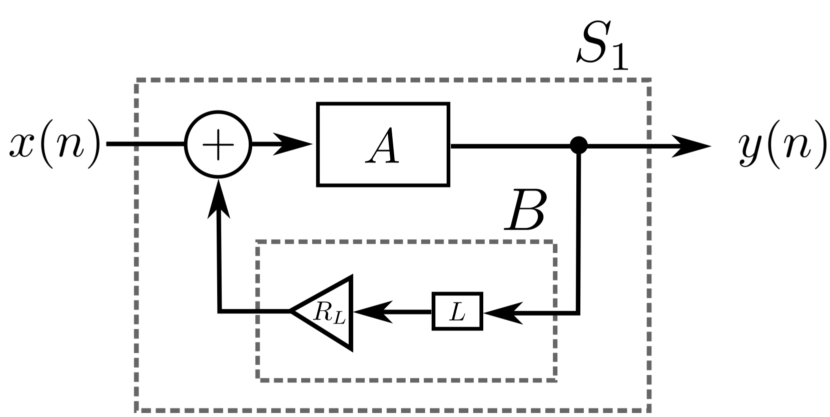

Figura 15: Bloque elemental

Mediante teoria de feedback a considerando que $B(z) = z^{-L}R_L$ se llega a que

$$S_1(z) = \frac{\frac{1}{2} + \frac{1}{2}z^{-1}}{1 - \frac{1}{2}R_L z^{-L} - \frac{1}{2}R_L z^{-L-1}} = \frac{\frac{1}{2}z^{L+1} + \frac{1}{2}z^{L}}{z^{L+1} - \frac{1}{2}R_L z - \frac{1}{2}R_L} \tag{12}$$

$$y(n) = \frac{1}{2}x(n) + \frac{1}{2}x(n-1) + \frac{1}{2}R_L y(n-L) + \frac{1}{2}R_L y(n-L-1) \tag{13}$$

Es decir, una expresión con $L+1$ polos y $L+1$ ceros, en otras palabras, una ecuación diferencial de orden $L+1$ con vector de longitud L como condición inicial. De la ecuación de diferencias se puede observar que la unica forma de garantizar estrictamente la estabilidad del sistema será exigiendo $R_L < 1$. No obstante, en la práctica como la frecuencia de resonancia del sistema no será exacta; colocar $R_L = 1$ no provocará inestabilidad. Más aun; resulta provechoso aproximar R_L a 1 lo más posible para lograr estirar la duración de las oscilaciones.

6.2.2. Estudio de la distribución polos y ceros, respuesta en frecuencia

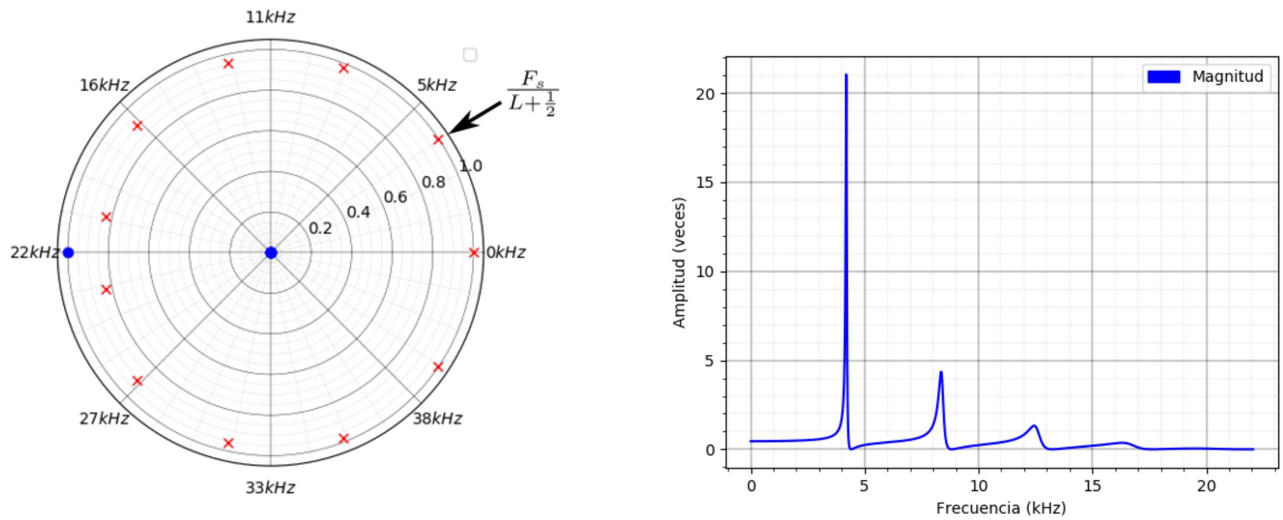

Figura 16: Polos y ceros (derecha), Rta en frecuencia (izquierda), S_1 con $R_L = 1$, $L = 10$, $f_s = 44.1kHz$

Del diagrama de polos y ceros y la respuesta en frecuencia podemos observar que hay una frecuencia de resonancia $F_R = F_s/(L + \frac{1}{2})$ que tiende a cumplir las hipotesis del criterio de Barkhausen, y por lo tanto provocar oscilaciones, lo cuál es el objetivo del bloque; conseguir una salida que perdure en el tiempo a partir de una entrada de longitud L muy corta.

6.2.3. Cálculo de F_R

Se mostrará porque el sistema resuena en $F_R = F_s/(L + \frac{1}{2})$. En la sección anterior se observó que la frecuencia de resonancia debía ser aquella que tendiera a cumlir las hipotesis del criterio de barkhausen; en otras palabras; que la ganancia del lazo se aproxime a 1 con 0 grados. El sistema esta compuesto por la superposición de un sistema con un retraso de lazo L y otro sistema con retraso de lazo $L+1$. Por lo tanto; para que una señal no tienda a interferirse al recorrer el lazo debe tener periodo $\frac{L+L+1}{2}$; es decir una frecuencia $\frac{F_s}{L+1/2}$

6.2.4. Análisis mediante señales

Se procederá a estudiar como el sistema responde a diversas entradas, entre ellas, un impulso unitario, ruido gaussiano de longitud L, y ruido lineal de longitud L. Se decidió aumentar L en estos casos a 50 para conseguir frecuencias más cercanas a las audibles.

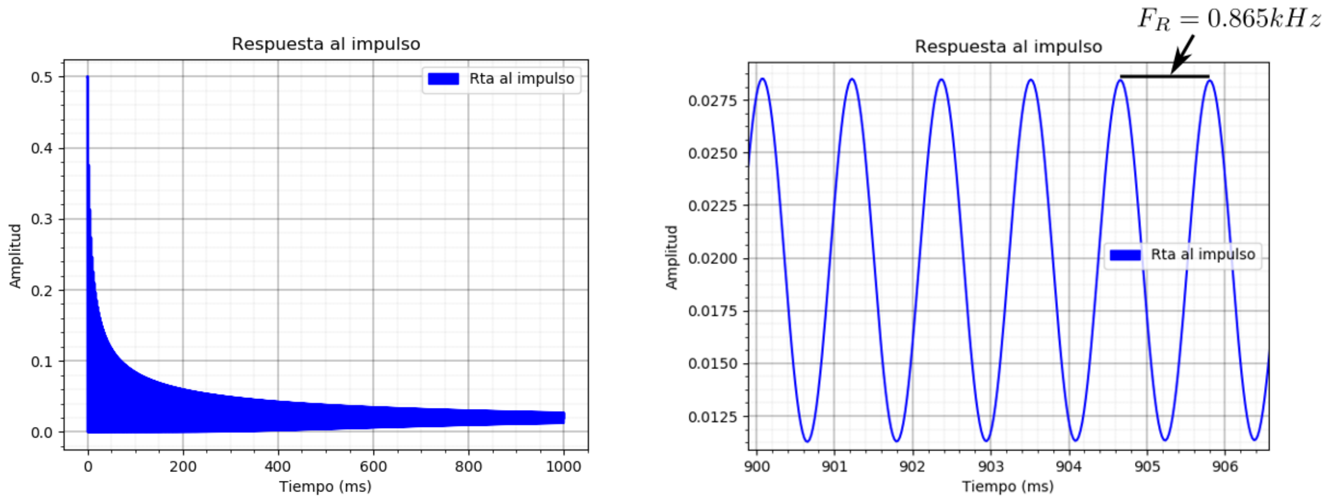

Figura 17: Respuesta al impulso con y sin zoom, $R_L = 1$, $L = 50$, $f_s = 44.1kHz$

Se puede observar que, de todas las frecuencias pertenecientes al impulso, la más amplificada vale $865Hz \approx \frac{F_s}{1/2+L}$. Por otro lado es importante observar que la salida se encuentra montada sobre una tensión continua, más precisamente, nunca es menor que 0. Esto se debe a que cuando la excitación es exclusivamente positiva, como la realimentación es positiva y no hay inversiones la salida siempre será positiva. Por ello es importante que la excitación inicial sea tanto positiva como negativa para lograr que la salida este centrada en 0.

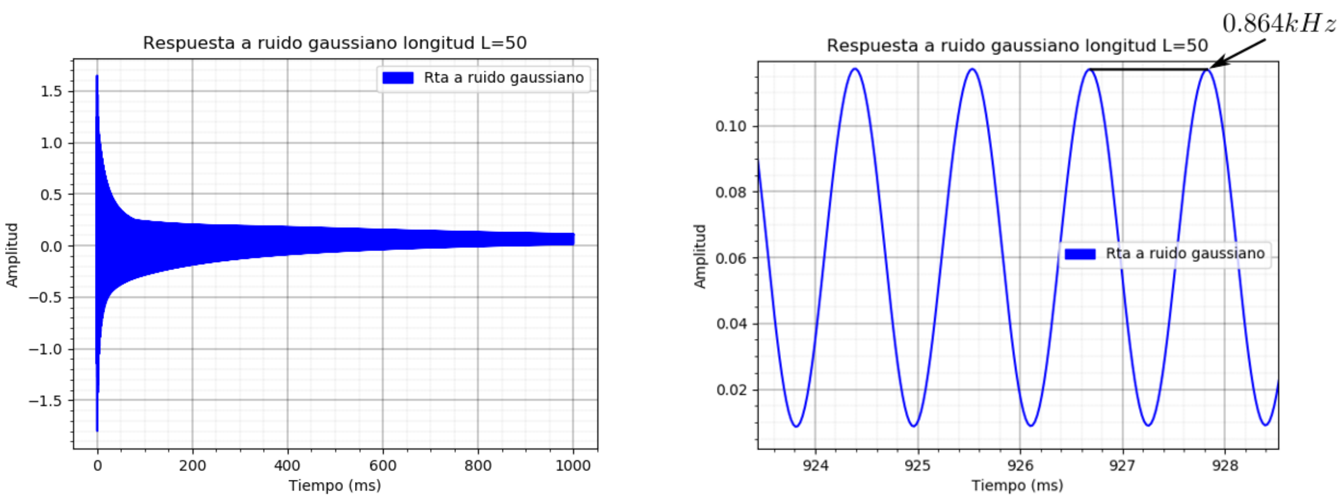

Figura 18: Respuesta a entrada Gaussiana con y sin zoom, $R_L = 1$, $L = 50$, $f_s = 44.1kHz$

Se puede ver que la salida fue a la misma frecuencia que en el caso anterior, al mismo tiempo que nuevamente, estuvo montada sobre una continua. No obstante la salida tuvo la posiblidad de tomar valores negativos.

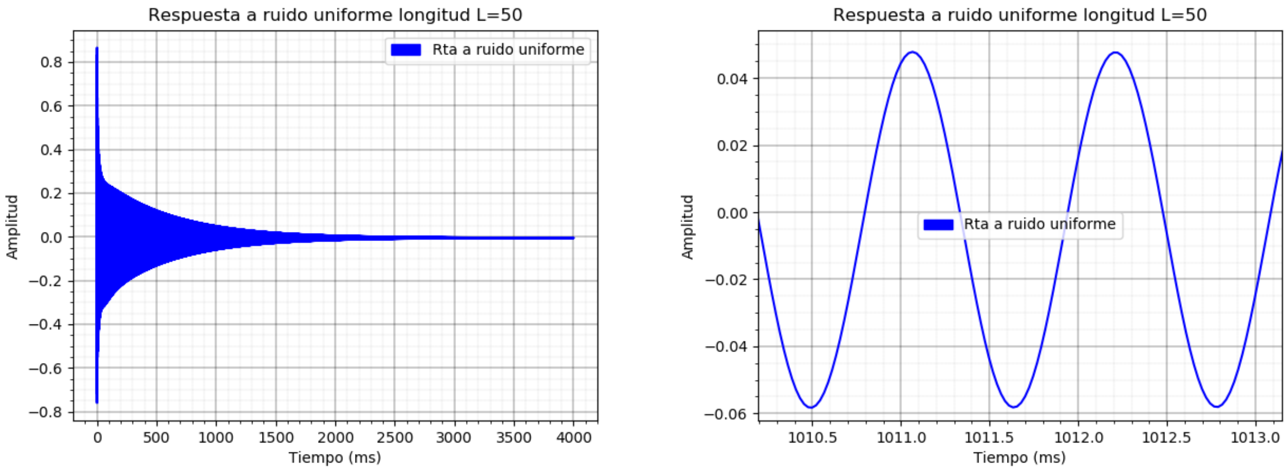

Figura 19: Respuesta a entrada de ruido uniforme con y sin zoom, $R_L = 1$, $L = 50$, $f_s = 44.1kHz$

Podemos observar que la salida fue similar a la de ruido gaussiano

6.3. Karplus Strong 2

6.3.1. Analisis teórico elemental

Se estudiará un nuevo sistema con una pequeña modificación la cual consiste en agregar un multiplicador en el bloque realimentador, cuyo valor es aleatorio, puede ser 1 o -1. La probabilidad de que sea 1 se llamará b

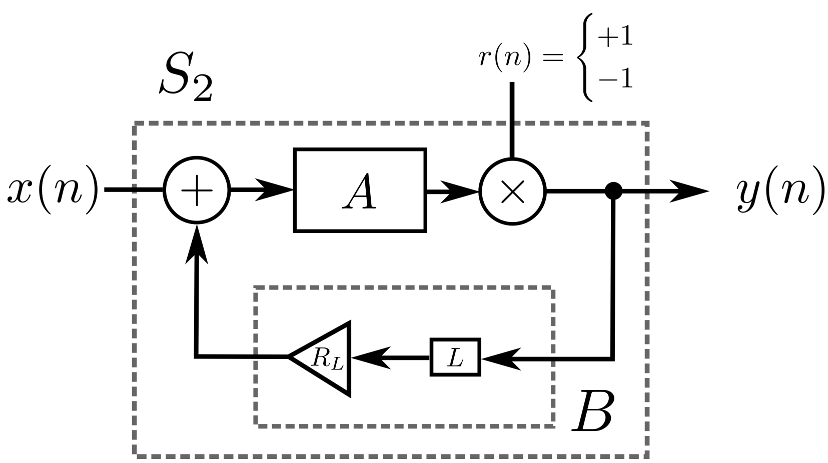

Figura 20: Diagrama del sistema S_2

Escrito en lenguaje matemático

$$A(z) = b(\frac{1}{2} + \frac{1}{2}z^{-1}) \tag{14}$$

$$B(z) = \frac{1}{2}z^{-L} \tag{15}$$

Usando teória de feedback concluimos que

$$S_2(z) = \frac{\frac{1}{2}r + \frac{1}{2}rz^{-1}}{1 - \frac{1}{2}rR_Lz^{-L} - r\frac{1}{2}R_Lz^{-L-1}} \tag{16}$$

$$y(n) = \frac{1}{2}rx(n) + \frac{1}{2}rx(n-1) + \frac{1}{2}rR_L y(n-L) + \frac{1}{2}rR_L y(n-L-1) \quad (17)$$

Observamos de la ecuación de diferencias que el sistema es prácticamente equivalente al anterior con una diferencia; ahora la salida se invierte de manera aleatoria ciclo a ciclo. Esta aleatoriedad ayudará a simular la aleatoriedad de un sonido percutido.

6.3.2. Respuesta en frecuencia

Debido a la complejidad trigonometricá del cálculo exacto se opto por aproximar númericamente el gráfico de $\phi(w) = \angle H(e^{jw})$ (respuesta en frecuencia del sistema) Por otro lado se decidió tampoco realizar el cálculo de la frecuencia de resonancia con exactitud, debido a su elevada complejidad. En realidad no es estrictamente necesario, en realidad en la práctica cuando b sea distinto de 1 o 0 ocurrira notable interferencia en el lazo que destruira la resonancia; lo cual provocará que el sonido se extinga luego de breves instantes.

Ahora se mostrará la respuesta en frecuencia con $b = 1, b = \frac{1}{2}, b = 0$. Es importante observar que dado que la salida es estrictamente una variable aleatoria, la respuesta en frecuencia entonces estrictamente también es una variable aleatoria. Se decidió mostrar el valor medio de dicha variable aleatoria.

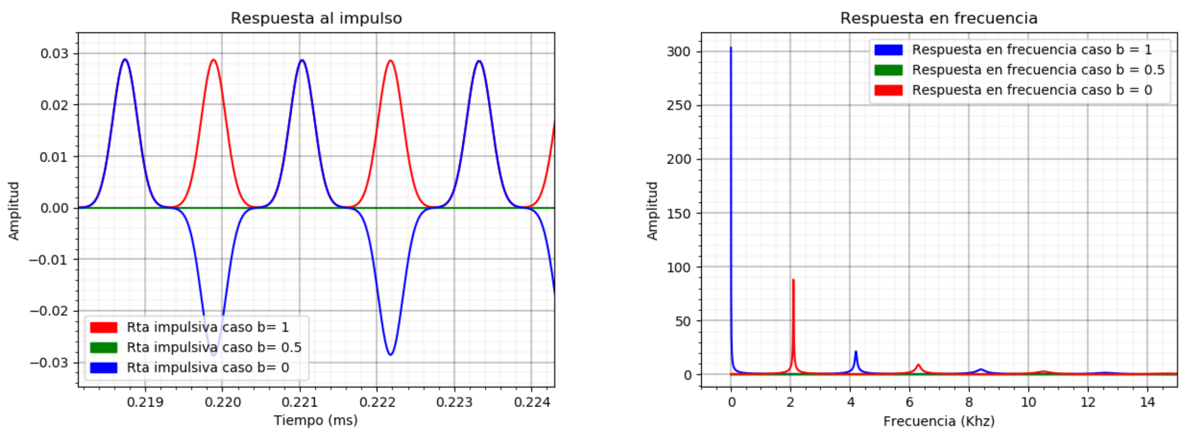

Figura 21: Respuesta al impulso y en frecuencia con $b = 1, 0.5, 0$ (izquierda a derecha)

Se observa que, como fue predicho, en el caso b=0.5 todas las frecuencias son destruidas luego de un pequeño transitorio. Además, se observó que el caso de b=0 mostró una inversión espectral que provocó que la frecuencia de resonancia se redujerá en dicho casó lo cual implicó un sonido más grave, como el de un arpa. También se observó una gran amplificación en el caso $b = 1$ en la continua. Esto no es un gran problema ya que si la entrada tiene valor medio cercano a 0 y se filtra dicha componente a la salida es un problema evitable.

7. Sintetización de instrumentos

En base a las distintas pruebas realizadas anteriormente se procederá a realizar las funciones que permitan sintetizar el sonido de varios instrumentos que son:

- Arpa
- Guitarra
- Tambor (percusión)

7.1. ¿Como tener libertad con la frecuencia fundamental?

La primera necesidad fue tener la libertad de poder sintetizar un sonido con cualquier frecuencia que se necesite. La solución fue sencilla; consistió en modificar el bloque A para que nos permita una mayor libertad al elegir la frecuencia

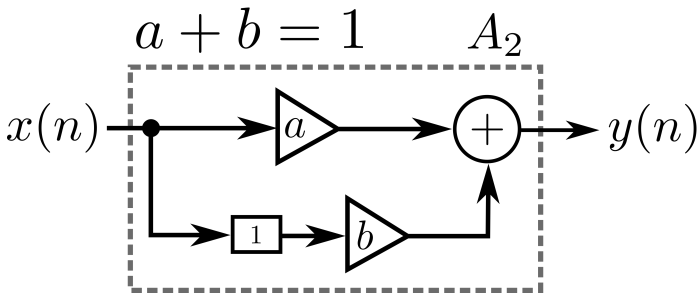

Figura 22: Modificación bloque A para poder personalizar la frecuencia

Con la nueva modificación la frecuencia de resonancia paso a estar dada por

$$F_R = F_s(a\frac{L}{2} + b\frac{L+1}{2}) \tag{18}$$

Eligiendo L adecuadamente se puede garantizar que existan a y b entre 0 y 1 que verifiquen la ecuación anterior

7.2. ¿Comó filtrar la componente continua?

Se colocó un filtro pasabajo al final del filtro para elimiar la "molesta" componente continua que el sistema poseeía. Dicha característica de la señal era aleatoria e impredecible por lo tanto fue muy importante colocar dicho filtro.

7.3. ¿Como atenuar el sonido de manera suave?

Cuando termina el sonido la amplitud no puede ser disminuida de manera inmediata. Se necesito entonces multiplicar la señal por una ventana para regular la duración y provocar una disminución gradual de la intensidad del sonido

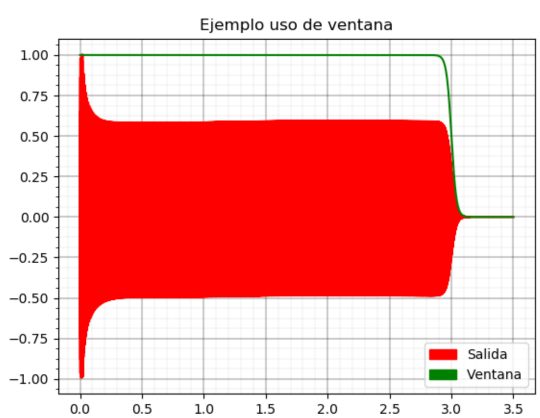

Figura 23: Ejemplo uso de ventana

Los resultados del uso de ventanas fueron satisfactorios, la atenuación del sonido se observó de forma gradual

7.4. Sintetización de sonido de guitarra

Se implemento el sonido de una guitarra utilizando el modelo Karplus Strong utilizando la modificación que permitió una elección continua de frecuencias. A continuación se mostrará el espectograma del sonido de la guitarra producido.

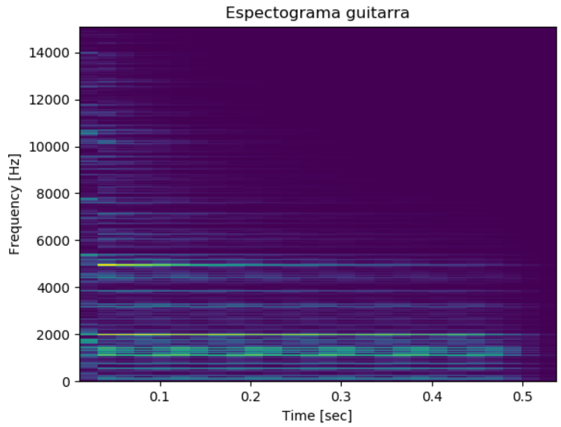

Figura 24: Espectograma sonido de guitarra generado, nota: $A_0 = 440Hz$

El sonido escuchado fue como el de una guitarra, por lo tanto se consideró que los resultados fueron satisfactorios; los armónicos se aproximaron a los de una guitarra acústica.

7.5. Distorsión del sonido de guitarra

Se distorsionó el sonido de una guitarra utilizando una modificación al modelo. Se colocó un valor de $R_L > 1$ pero, utilizando una funcion normalizadora (entre -1 y 1) se conformó un control automatico de ganancia que evitó la inestabilidad del sistema. Colocando diversos valores de R_L se armó distorsión de distintos tipos. Se mostrará un espectograma de uno de los sonidos generados

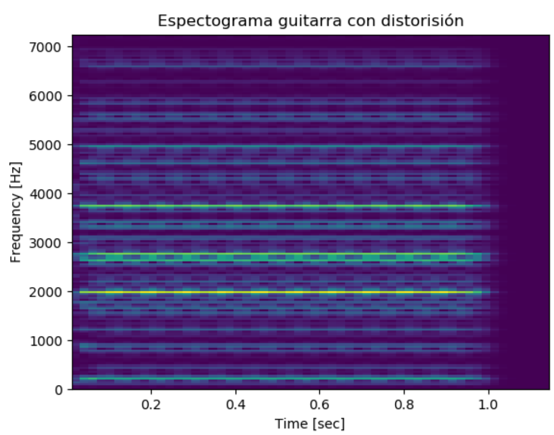

Figura 25: Espectograma sonido de guitarra con distorsión, $R_L = 2$, nota: $A_0 = 440Hz$

Se observa que en general los armónicos generados producto de la distorsión son muchos más numerosos y además tienden a tener una frecuencia mayor que los de la guitarra acústica. El sonido generado fue percibido como el de un sonido eléctrico, que sonaba bien pero que no se asocia a ningún instrumento acústico.

7.6. Sintetización de instrumento de percusión

Utilizando la modificación del modelo de Karplus Strong la cual establecía una variable aleatoria se sintetizó el sonido de instrumentos de percusión. Se necesitó aplicar algunas modificaciones, en primer lugar se decidió colocar un valor de L elevado para elevar el tiempo de decaimiento. Por otro lado, se provocó que el estímulo al sistema cayera de manera abrupta, de otra manera no se podía observar el sonido percutido, sino simplemente ruido. A continuación se mostrará la salida en el tiempo y en frecuencia.

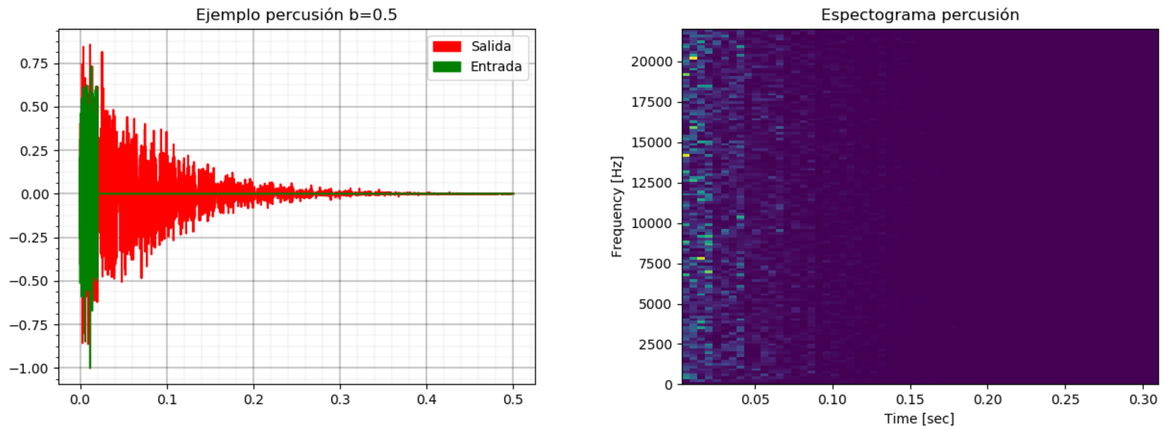

Figura 26: Respuesta en tiempo y en frecuencia de sonido percutido

7.7. Explicación caja de resonancia

Se explicará el funcionamiento de la caja de resonancia de una guitarra. No se implementará ya que la necesidad que existe de utilizar dicho elemento en una guitarra es física, en realidad, desde la perspectiva de señales, es posible sintetizar el sonido de una guitarra sin necesidad de simular la caja. Lo que se busca cuando se fabrica una guitarra es que el sonido generado por el estimo de las cuerdas se propage en una dirección con una intensidad tal, que permita una transmisión efectiva por el aire. Sin embargo, el aire es un medio de impedancia baja mientras que las cuerdas son una fuente de impedancia alta. Por lo tanto de solo existir las cuerdas el sonido si bien en el sentido más estricto sería producido, no lograría propagarse con intensidad suficiente. Por lo tanto surge la necesidad de agregar un elemento físico adicional, la caja de resonancia, que toma el sonido propagado por las cuerdas hacia "adentro" y lo vuelve a propagar hacia "afuera" con una intensidad mayor, amplificando todas las frecuencias por igual (de lo contrario ocurriría distorisión).

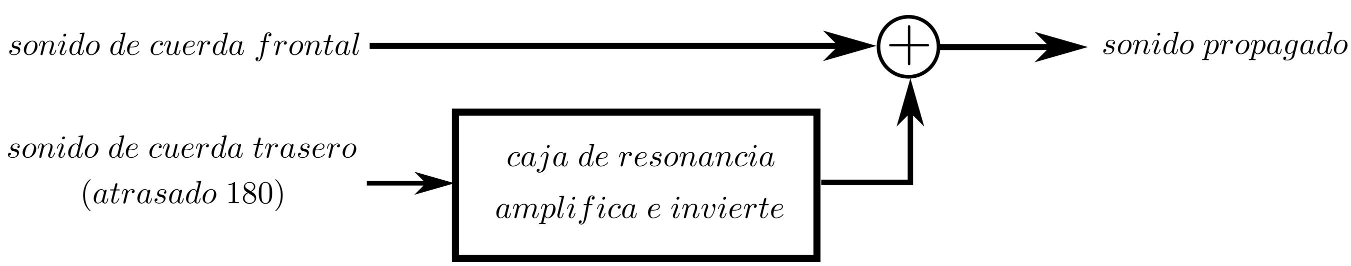

Figura 27: Caja de resonancia

Además impone una inversión de fase, idealmente en todas las frecuencias, ya que el sonido propagado por las cuerdas hacia "adentro" esta invertido en fase comparando con el emitido por las cuerdas hacia "afuera".

Los diseños modernos de cajas de resonancia intentan maximizar tanto la inversión de fase como la amplificación de la mayor cantidad de frecuencias, en particular las más bajas que tienen más relevancia en el sonido.

8. Síntesis basada en muestras

9. Espectrograma

9.1. Función para realización de espectrograma

Para la realización de los espectrogramas, se utilizó la función "spectrogram", de la librería «scipy» de Python. La misma se basa en la implementación de transformadas de Fourier consecutivas sobre la señal a lo largo del tiempo. La sintaxis para su utilización es la siguiente:

scipy.signal.spectrogram(x, fs=1.0, window=('tukey', 0.25), nperseg=None, noverlap=None, nfft=None, detrend='constant', return_onesided=True, scaling='density', axis=-1, mode='psd')

Donde los parámetros y sus efectos en detalle se describen a continuación.

- **x [array]**: arreglo con los valores que toma la señal en el tiempo (sobre los que se aplica la transformada de Fourier).

- **fs [float]**: frecuencia de sampleo de la señal $x(t)$. Por defecto normalizada: 1.0.

- **window [string ó tuple ó array]**: ventana temporal a utilizar. Influye tanto el tipo de ventana como como el ancho de la misma.

  Respecto al tipo de ventana, si se utiliza por ejemplo una ventana cuadrada, esta posee un corte abrupto. Si en dicha ventana no entra un número entero de períodos de la señal, se produce un corte abrupto.

  Esto remite en la aparición de otras componentes de alta frecuencia que antes no había, lo que se conoce como «fuga espectral» (dado que la energía de los armónicos principales se "fuga" a los otros armónicos nuevos). Otras ventanas (como la Blackman-Harris) tienen un corte suave en los extremos (tienden a cero gradualmente), lo que minimiza la fuga espectral considerablemente.

  El ancho de la ventana interviene en la resolución en tiempo y en frecuencia. Si la ventana es más ancha, se obtiene mayor resolución en frecuencia, dado que si la frecuencia de la señal sufre algún cambio en el tiempo (como en una señal FM), es posible captarlo con la ventana con mayor definición. Pero en el tiempo se pierde resolución dado que se sabe con menor precisión dónde ocurre exactamente el cambio de frecuencia. Con una ventana angosta, se gana resolución en el tiempo, pero en frecuencia se podría perder el cambio que antes se lograba captar en el tiempo con una ventana ancha, por lo que se vería una sola frecuencia en lugar de dos.

- **npersec [int]**: es el largo de cada segmento. Por defecto es «None», pero si la ventana se da en formato de «string» se considera 256, y si se da como «array» es el largo del mismo.

- **noverlap [int]**: es el número de puntos a solapar entre segmentos. Por defecto es «None», que es npersec // 8. Es decir, define la separación resultante entre ventanas.

- **nfft [int]**: es el largo de la FFT utilizada. Por defecto es «None», que determina el largo igual a "npersec".

- **detrend [string ó function ó False]**: elimina la tendencia lineal a lo largo del eje temporal. Por defecto es "constant". Si es de tipo string, se pasa directamente a la función «detrend» (también de scipy), y si es una función, se le pasa un segmento y lo devuelve ya con el dsvío correspondiente. Si es «False», no se aplica ningún desvío. Para el caso de audio, no nos es de interés.

- **return_ onesided [bool]**: si se asigna «True», se devuelve un espectro unilateral. Si es «False», el espectro será bilateral.

- **scaling ['density', 'spectrum']**: considerando a $x(t)$ en volts [V], se elije si procesar la densidad espectral de potencia [V^2/Hz] ó el espectro de potencia [V^2]. Por defecto se procesa la densidad espectral. Se simboliza como S_{xx}.

- **axes [int]**: es el eje a lo largo del cual se procesa el espectrograma. Por defecto es «-1» (es decir, el último utilizado).

- **mode** [**str**]: define que es lo que se espera que devuelva la función, entre [“psd”, “complex”, “magnitude”, “angle”, “phase”]. “psd” es la densidad espectral de potencia; con “complex” devuelve la STFT (Short-Time Fourier Transform) compleja; “magnitude” devuelve el valor absoluto de la STFT, y “angle” y “phase” el ángulo correspondiente complejo.

Los parámetros que devuelve son los siguientes:

- **f** [**ndarray**]: arreglo de dimensión “n” con las frecuencias de sampleo.
- **t** [**ndaray**]: arreglo de dimensión “n” con los segmentos de tiempo.
- $S_x x$ [**ndarray**]: arreglo de dimensión “n” con el espectrograma de $x(t)$.

En referencia a las diferentes ventanas utilizables, además del ancho se debe considerar realizar el análisis con o sin overlap (solapamiento entre segmentos). En el caso más sencillo, no utilizar overlap podría conllevar pérdidad de información: si se tiene un cambio de frecuencia de corta duración en comparación a la duración total del segmento, y se encuentra cerca del final de éste, al no utilizar segmentos solapados para la STFT podría perderse dicho evento. Para evitar dicha pérdida de información, se aplica la STFT a segmentos con un cierto grado de solapamiento. Se gana en no perder información, pero se requiere mayor tiempo de procesamiento, dado que al ser segmentos solapados, para cubrir la totalidad de la señal se necesita realizar más veces la transformación. El porcentaje de overlap se encuentra en general entre el 50 % y el 75 %, de acuerdo al paper «*On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform*» (1978, Fredric J. Harris, member IEEE).

9.2. Aplicación a escala de Sol Mayor (G3)

La tonalidad de *sol* mayor (en el sistema inglés abreviada como **G**) consiste en la escala mayor del sol y contiene las notas *sol, la, si, do, re, mi, fa* sostenido y *sol*. La notación **G3** corresponde a la pequeña octava, cuya frecuencia es aproximadamente 195.998Hz.

El espetrograma realizado se muestra a continuación.

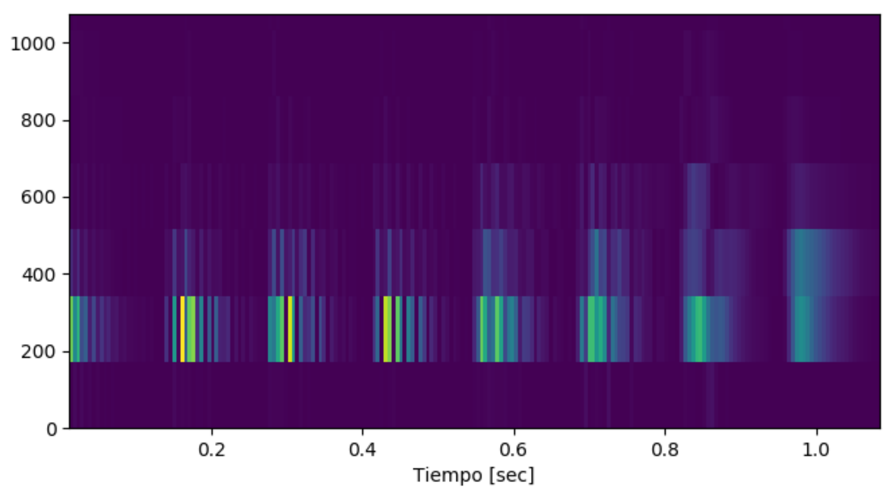

Figura 28: Espectrograma de Sol mayor

En el espectrograma obtenido sintetizando con un piano, resulta visible la frecuencia de la primera octava cercana a los 200Hz. Se utilzan 120ms por nota según lo sugerido, de manera tal que puede diferenciarse cada una por la escala creciente en frecuencia.

10. Efectos de audio

10.1. Reverbebrador

10.1.1. Implementación de eco simple

Se implementó un eco simple utilizando el sistema

$$y(n) = x(n) + gx(n - M) \tag{19}$$

Se muestran a continuación los resultados con una señal de prueba

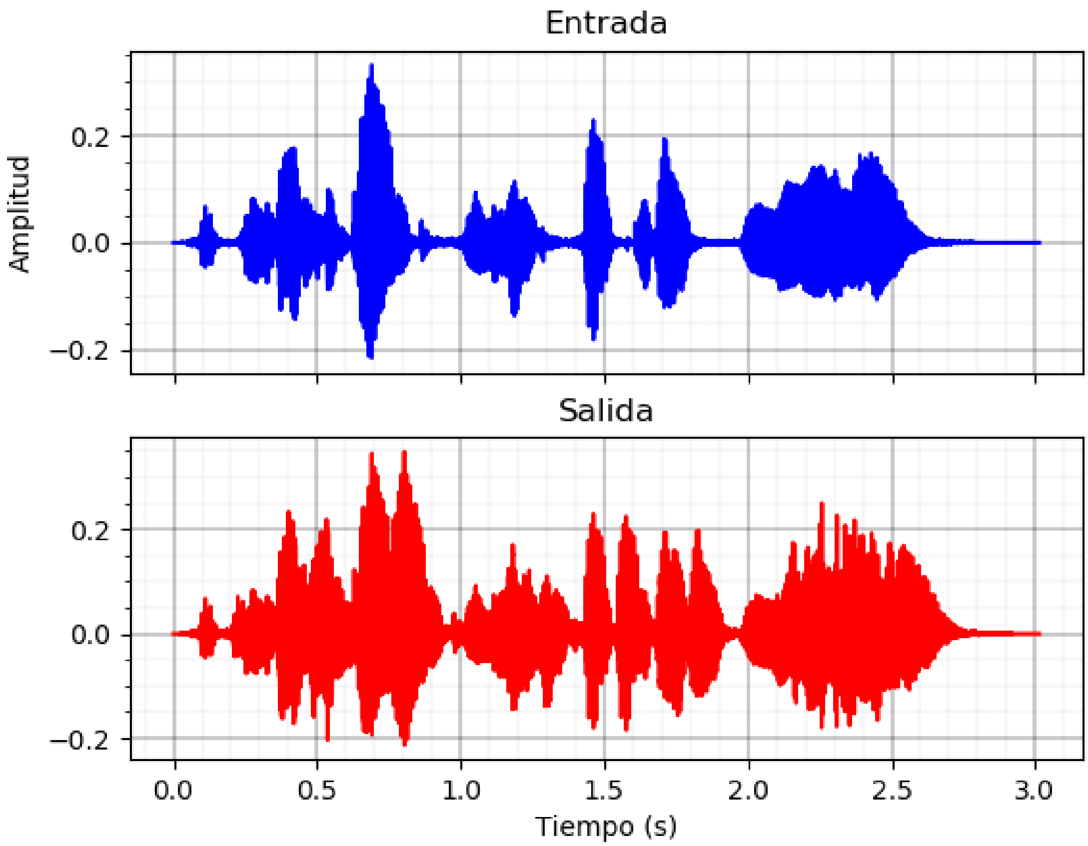

Figura 29: Resultados con $M = 5000$, $g = 0.999$

Se puede observar de los resultados intuitivamente como la señal de salida contiene repeticiones de la señal de entrada, y al escuchar el audio se pudo notar dicho efecto de eco. Fue necesario colocar un retraso muy grande ($M = 5000$) y una ganancia muy alta ($g = 0.999$) para que el efecto fuera notorio

10.1.2. Implementación de reverberación plana

Se implementó una reverberación plana utilizando una ecuación de diferencias con feedback.

$$y(n) = x(n) + gy(n - M) \tag{20}$$

Se muestran a continuación los resultados con una señal de prueba

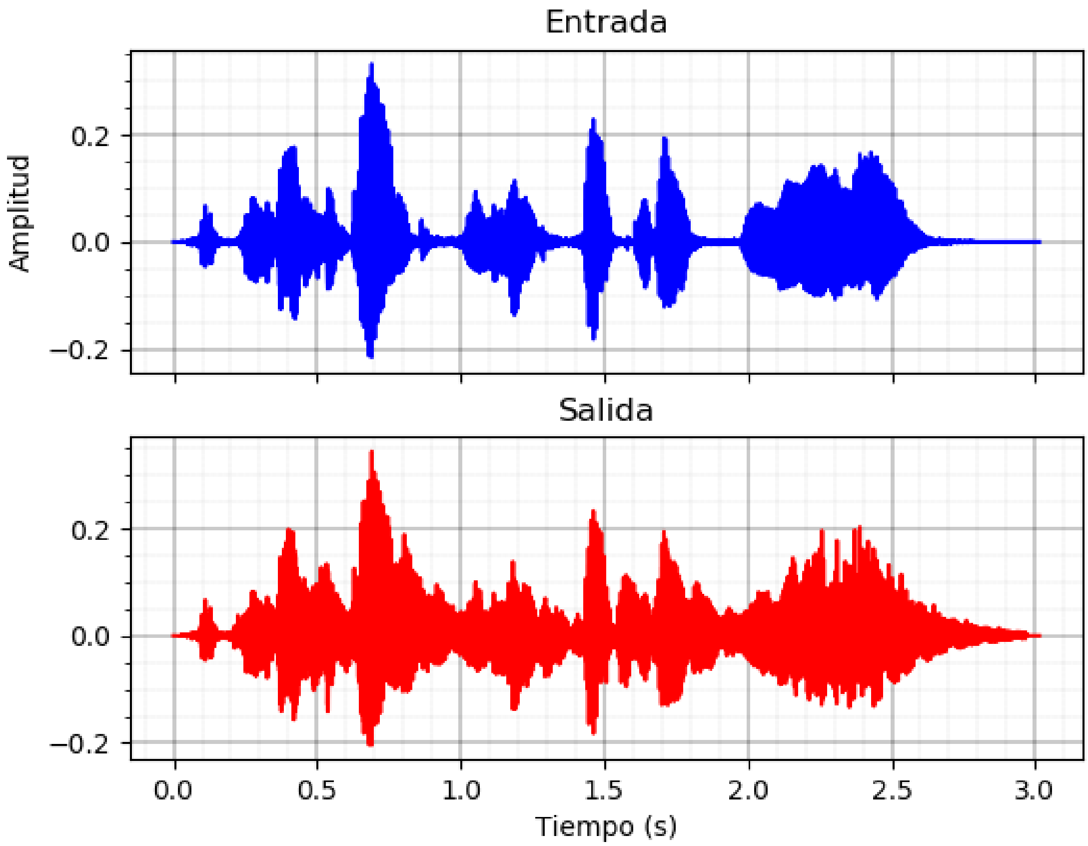

Figura 30: Resultados reverberación plana con $M = 500$, $g = 0.5$

Se necesitó disminuir fuertemente el valor de g para evitar que la salida saturara. Al tener realimentación (es decir, ser IIR) el sistema puede perder la estabilidad con facilidad

10.1.3. Implementación de reverberación pasa bajos

Se le agrego un filtro pasabajo a la realimentación del sistema anterior. Se optó por un sencillo pasabajos similar al utilizado en el modelo Karplus Strong, de la forma $y(n) = \frac{x(n)+x(n-1)}{2}$ El sistema por lo tanto quedo descrito como

$$y(n) = x(n) + \frac{1}{2}g(y(n-M) + y(n-M-1)) \tag{21}$$

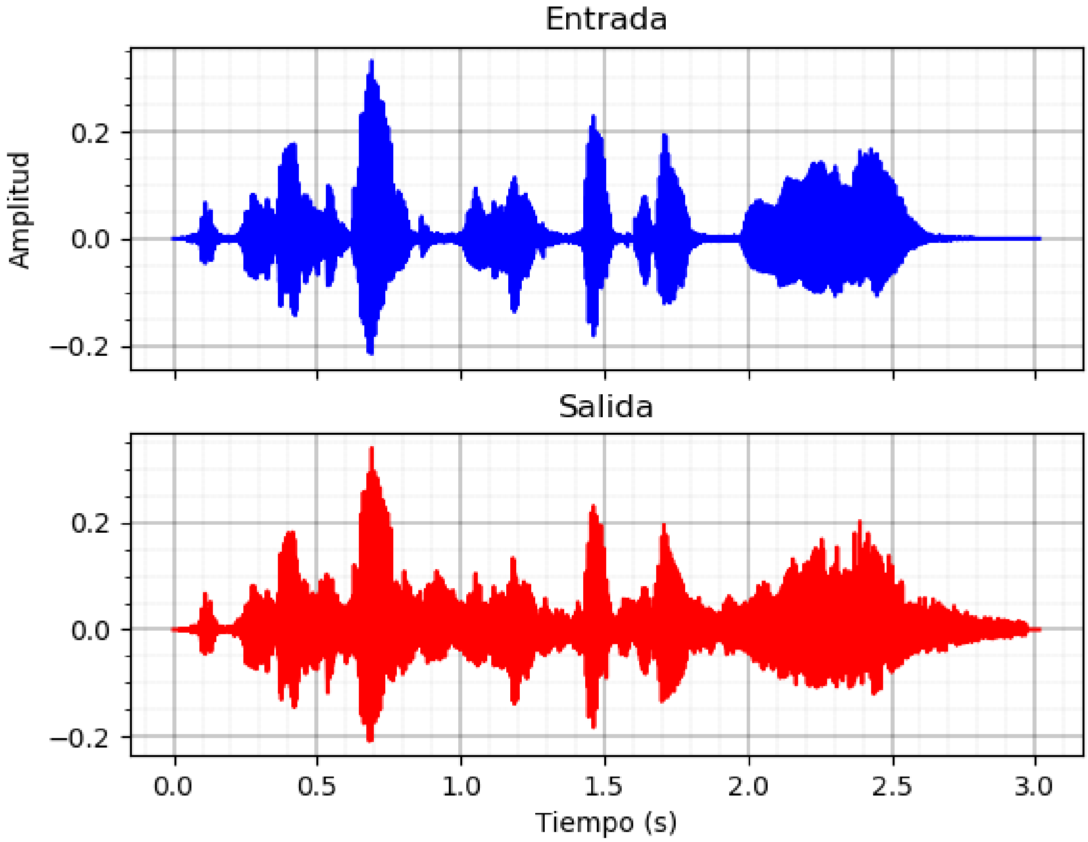

Figura 31: Resultados reververación con pasabajos $M = 500$, $g = 0.5$

La señal fue similar al resultado del caso anterior con una sutil diferencia, el sonido se escuchó un con poco menos ruido, esto se debe muy probablamente a que el filtro pasa bajos evito la propagación de una frecuencia no deseada la cual no estaba presente en la señal original.

10.1.4. Implementación de reverberación completa

Se buscó estudiar un caso de reverberación completa. Se eligió el sistema denominado Schroeder. El mismo consistió de la conexión en paralelo de N reverberadores planos, continuados de M filtros tipo comb. Se utilizó algunos criterios de los expresados en los apuntes de clase como también se jugó empiricamente para conseguir sonidos de reverberación interesantes. A continuación se muestran los resultados

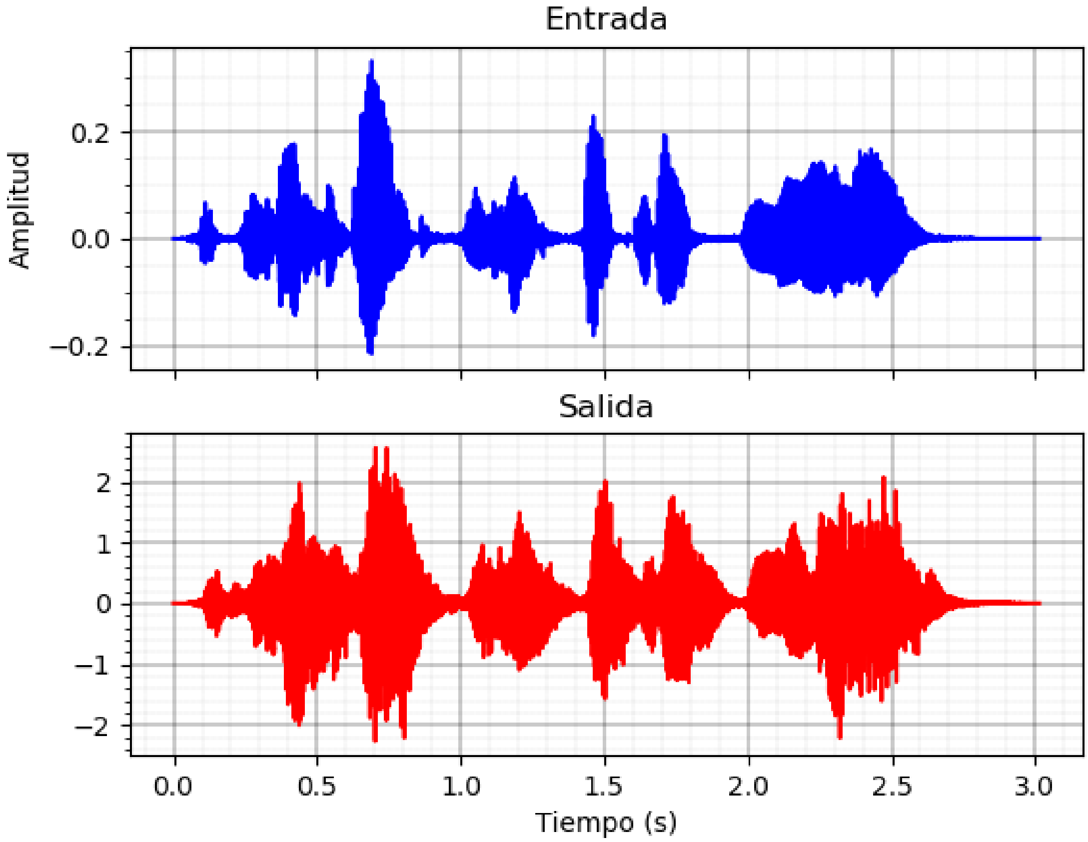

Figura 32: Resultados de un caso de reverberación completa, $N = 12$, $M = 2$, $a_{etapa1} = 0.999$ $a_{comb} = 0.5$

El sonido resultante fue muy interesante, fue algo robotizado y conseguidó tan solo ajustando parametros del filtro. Esto da la pauta de que con el reverberador completo se puede facilmente ajustando sus parametros conseguir tipos muy distintos de reverberación, sin necesidad de tener que convolucionar con una respuesta al impulso

10.1.5. Implementación de reverberación por convolución

Se implemento una reverberación utilizando convolución con la respuesta al impulso característica de una fabrica. Se utilizo la ecuación de diferencias génerica siguiente

$$y(n) = \sum_{i=k}^{N} h(k)x(n-k) \tag{22}$$

Debido a la complejidad algoritmica de la aplicación de la fórmula se debió limitar la longitud de la respuesta al impulso a solo 20000 muestras. Se muestran a continucación los resultados con una señal de prueba

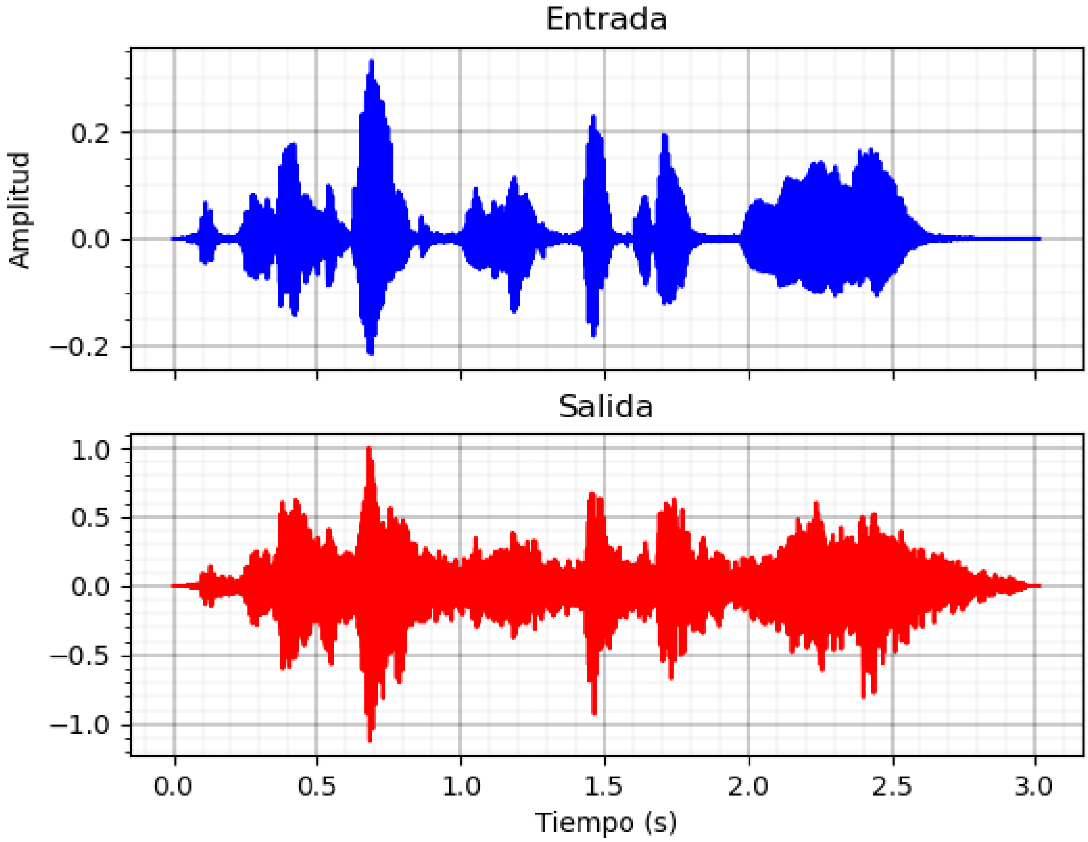

Figura 33: Resultados reverberación característica de una fábrica

Se observa como a diferencia de los casos anteriores la señal tiende mucho más a sostener sonido, esto se debe a que la respuesta al impulso es mucho más completa, y por lo tanto el sonido resultante persiste más tiempo. El sonido que se escuchó se correspondió con un eco muy realista, que podría ser el de una fábrica

10.2. Efectos basados en delays variables

Para algunos efectos, es necesario que los delays presentes en la ecuación de diferencias que describe al sistema sean variables con el tiempo. Este es el caso tanto del flanger como del vibrato, entre otros.

Los efectos basados en delays variables pueden analizarse de manera general a partir de un filtro comb universal, cuyo diagrama de bloques se observa en la figura 34.

La variación del delay está dada por $M(n)$ (que debe ser $\geq 0 \forall n$ para que el sistema sea causal), mientras que los factores FF (feedforward), FB (feedback) y BL (blend) determinan qué tipo de efecto se producirá.

Las ecuaciones de este sistema son:

$$\begin{cases} x_h(n) = x(n) & +\mathrm{FB} \cdot x_h(n - M(n)) \\ y(n) = \mathrm{BL} \cdot x_h(n) & +\mathrm{FF} \cdot x_h(n - M(n)) \end{cases} \tag{23}$$

Reemplazando la primera ecuación en la segunda, obtenemos que:

$$x_h(n - M(n)) = \left(\frac{1}{\mathrm{BL} \cdot \mathrm{FB} + \mathrm{FF}}\right) \cdot y(n) - \left(\frac{\mathrm{BL}}{\mathrm{BL} \cdot \mathrm{FB} + \mathrm{FF}}\right) \cdot x(n)$$

Aplicando este resultado en la primera ecuación de 23, se llega a:

$$x_h(n) = \left(\frac{\mathrm{FF}}{\mathrm{BL} \cdot \mathrm{FB} + \mathrm{FF}}\right) \cdot x(n) + \left(\frac{\mathrm{FB}}{\mathrm{BL} \cdot \mathrm{FB} + \mathrm{FF}}\right) \cdot y(n) \tag{24}$$

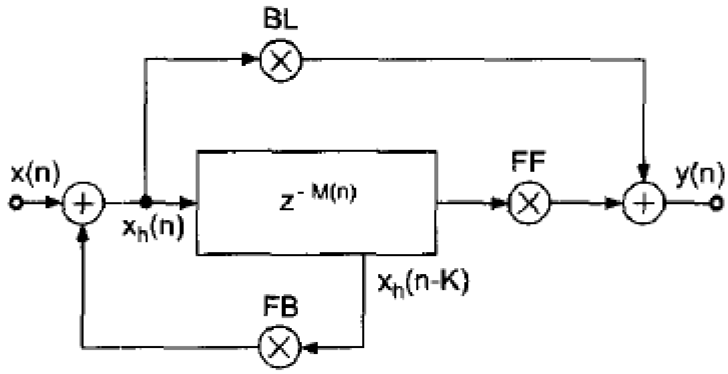

Figura 34: Diagrama de bloques del filtro comb universal

Finalmente, reemplazando con esta expresión de $x_h(n)$ en la segunda ecuación del sistema 23, se obtiene que:

$$y(n) = \mathrm{BL} \cdot x(n) + \mathrm{FF} \cdot x(n - M(n)) + \mathrm{FB} \cdot y(n - M(n)) \tag{25}$$

10.2.1. Vibrato

El efecto de vibrato consiste en introducir pequeñas variaciones periódicas en la frecuencia, resultando en una variación periódica de los tonos que se escuchan. Tomando como principio el efecto Doppler, podemos aplicar estos cambios en la frecuencia como cambios en el delay: de la misma manera que a medida que una ambulancia se aleja, se escucha más grave la sirena, si simulamos que el delay sube y baja periódicamente, la nota que se escucha subirá y bajará de la misma manera.

La ecuación que define este efecto es:

$$y(n) = x\,(n - k(n)) \tag{26}$$

En la ecuación 26, $k(n)$ representa el delay variable con el tiempo, y se implementa con una senoidal que toma valores naturales entre 0 y un máximo K:

$$k(n) = \left\lfloor \frac{K}{2} \cdot (1 + \sin(2\pi f_0 \cdot nT)) \right\rfloor \tag{27}$$

Los parámetros que caracterizan al vibrato son pues:

$$\begin{cases} f_0 = \text{frecuencia de modulación} \\ K = \text{profundidad de modulación} \end{cases} \tag{28}$$

10.2.2. Flanger

El flanger es similar al vibrato, en cuanto a que también estará presente en la salida la entrada con un delay variable. Sin embargo, la misma es sumada (con un cierto factor de ponderación) a la entrada actual, obteniéndose:

$$y(n) = x(n) + g \cdot x(n - k(n)) \tag{29}$$

La definición de $k(n)$ que usaremos es idéntica a la del vibrato (ecuación 27), es decir, una senoidal que varía entre 0 y un máximo.

Para obtener la respuesta en frecuencia de este filtro, se puede calcular la respuesta al impulso:

$$h(n) = \delta(n) + g \cdot \delta(n - k(n))$$

Y por lo tanto se obtiene:

$$H(z, n) = 1 + g \cdot z^{-k(n)} = \frac{z^{k(n)} + g}{z^{k(n)}}$$

Por lo tanto, el flanger puede interpretarse como un filtro FIR comb, con $k(n)$ polos en el origen, y $k(n)$ ceros con radio $g^{-k(n)}$, equidistantes entre sí. Esto quiere decir que tanto el orden como las frecuencias que pasan varían con el tiempo.

10.3. Robotización