

Laboratorio 3

Procesamiento Digital de Señales

1st Juan José Restrepo Rosero
Pontificia Universidad Javeriana
Cali

Cali, Colombia
juanjoestrepo@javerianacali.edu.co

2nd Juan David Vargas Mazuera
Pontificia Universidad Javeriana
Cali

Cali, Colombia
juandavid2001@javerianacali.edu.co

Resumen— En este laboratorio, se analizó la respuesta frecuencial del canto de un ave. Además, se filtraron los sonidos de insectos, los cuales se identificaron estaban en las frecuencias menores a 2 KHz, así como en las frecuencias mayores a 8 KHz. Para este proceso de filtrado se implementaron diferentes métodos de filtrado, tales como filtros FIR y filtros IIR, además del método de aproximación por Chebyshev.

Palabras clave— Respuesta en frecuencia, Ventanas, espectrograma, filtros FIR, filtros IIR, Chebyshev.

I. INTRODUCCIÓN

El desarrollo del laboratorio número 3 se centra en el filtrado de una señal del canto de aves con ruido de insectos. Para este fin se emplearon dos enfoques. Se diseñó un filtro FIR, y se diseñó un filtro IIR. Para ambos casos se empleó el diseño por medio del método de ventanas, las cuales fueron definidas entre la frecuencia de 2 KHz, y de 8 KHz, correspondiente a la frecuencia que se espera no hayan insectos. Adicional a esto, se diseñaron matemáticamente para posteriormente implementarse por medio de código en el laboratorio. Por otra parte, se realizó un diseño adicional de filtros FIR, teniendo en cuenta la aproximación de Chebyshev, esto de manera pedagógica y para lograr tener una mejor visión del tipo de filtros, métodos diferentes, y comparar resultados para determinar cuál podría ser el mejor filtro en este caso particular. La aproximación de Chebyshev se implementó con la rutina remez de Python.

II. OBJETIVOS

1. Diseñar filtros digitales FIR utilizando las técnicas de Ventanas y aproximación de Chebyshev.
2. Diseñar filtros digitales IIR utilizando las técnicas de transformación bilineal.

III. MARCO TEÓRICO

A. Filtrado de Señales

El filtrado de señales es un proceso en el que se altera el contenido espectral de una señal para suprimir o eliminar componentes no deseados en una banda en particular, o para

retener sólo una banda de frecuencias y eliminar otras; con la finalidad de almacenar información de su interés.

B. Filtro FIR

Es un tipo de filtro donde la señal de entrada es una señal de pulso, hay infinitos términos cero en la salida y que se caracteriza por la retroalimentación de la señal de salida, p. B. por la necesidad de filtros FIR de orden inferior.

C. Filtro IIR

Es un tipo de filtro en el que su respuesta a una señal de pulso de entrada tiene infinitos términos distintos de cero. Existen diferentes métodos para su diseño, siendo los más utilizados el método de aproximación, el método de ventana y el método de muestreo de frecuencia. Los filtros FIR se caracterizan por su fase lineal y requieren un orden superior al IIR.

D. Aproximación de Chebyshev

El uso de este enfoque en el diseño de filtros se refiere al uso de polinomios de Chebyshev para la posición de polo en un filtro dado. Su principal característica es que se puede configurar un rizado antes de la banda de paso, lo que permite una respuesta más rápida en la banda de paso.

IV. PROCEDIMIENTO

Inicialmente, tenemos un audio del canto de aves, el cuál tiene ruido de insectos. Este ruido se identifica que está presente tanto en las frecuencias inferiores a 2 KHz, como en las superiores mayores a 8 KHz. Se pretende eliminar este ruido mediante filtros FIR e IIR. La señal del audio se puede visualizar en la figura 1, y su respuesta en frecuencia se puede ver en la figura 2. Igualmente, se procesó el audio a un espectrograma con el fin de visualizar las frecuencias dónde más ruido se concentra.

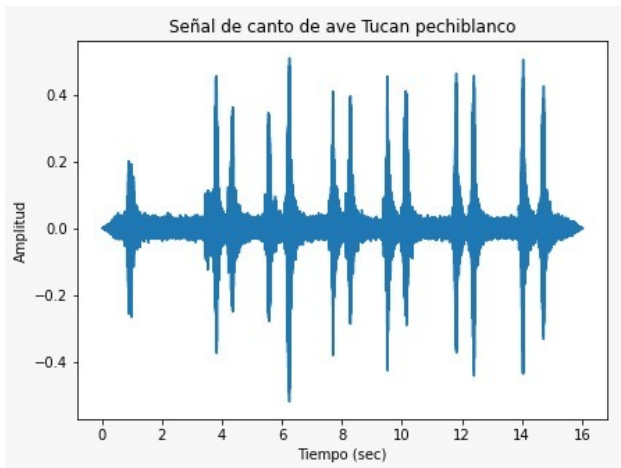


Figura 1. Visualización en el tiempo del audio.

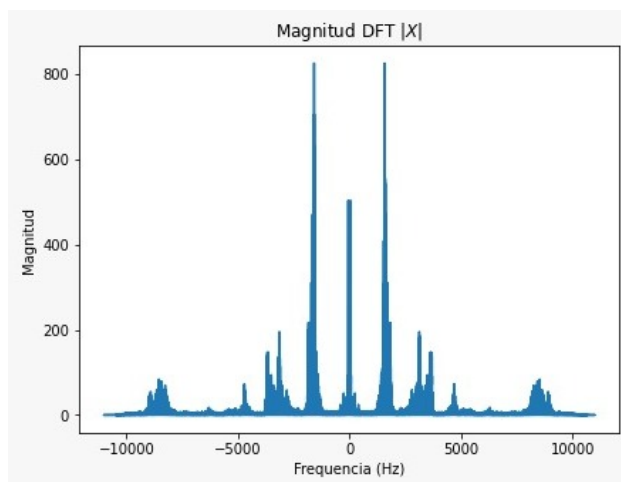


Figura 2. Visualización en frecuencia del audio.

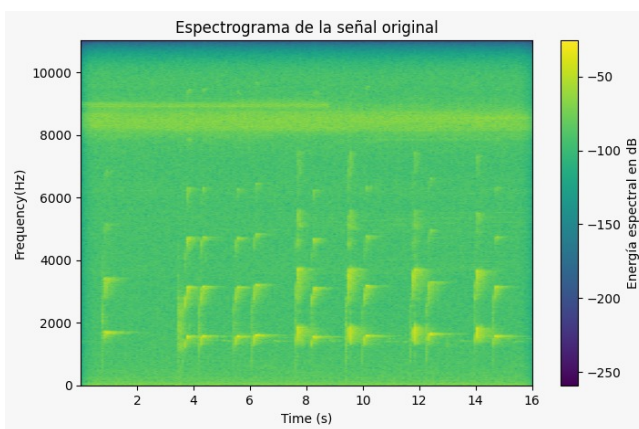


Figura 3. Espectrograma del audio.

A partir del espectrograma, pudimos observar como se ve un nivel de ruido blanco entre aproximadamente 7,6-7,8 KHz y 9KHz. Por esta razón, se decidió diseñar un filtro pasa banda entre 2 KHz y 7,6 KHz.

Para el desarrollo de este laboratorio se utilizó como lenguaje de programación Python, el cual, a lo largo de los años ha tenido un constante crecimiento logrando ser uno de los lenguajes más utilizados en aplicaciones como análisis de datos o Procesamiento Digital de señales. Además, Python cuenta con gran cantidad de librerías que hacen que las tareas complejas sean cada vez más fáciles; un ejemplo de estas es Matplotlib, que contiene funciones para aplicaciones gráficas. Se diseñaron soluciones tanto con filtros FIR, como con un filtro IIR.

A. Filtros FIR

Para la implementación del filtro FIR por ventanas, se tuvo en cuenta que para obtener un filtro de longitud M se tiene que aplicar que:

$$n = M - 1$$

A partir de esto, definimos las frecuencias de corte en 2 KHz por especificación del problema, y de 7,6 KHz después de observar el espectrograma.

Para realizar el truncamiento de la señal, debemos multiplicarla por una ventana, en este hamming. Decidimos usar esta ventana debido a que los lóbulos laterales se iban a haber reducidos y por ende podría pasar una señal más precisa y parecida a la señal original en frecuencia. Además de esto, se aplicará un filtro pasa banda, dado que requerimos cortes en ambos extremos debido al ruido gaussiano y al sonido de los insectos por debajo de 2 KHz.

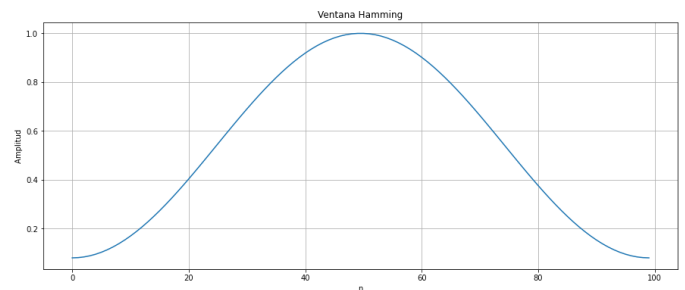


Figura 4. Ventana Hamming en el tiempo y en la frecuencia

Dada esta elección, nuestra ecuación de ventana hamming es:

$$w(n) = 0.54 - 0.46\cos(2\pi\frac{n}{M-1})$$

Para finalizar, otro factor a considerar para generar el código es la indeterminación que se presenta, y por ende, se debe definir un valor para dicha indeterminación. Esta se muestra a continuación:

B. Método de Chebyshev

La aproximación por el método de Chebyshev requiere ciertos parámetros de diseño. Estos son la banda de paso, la banda supresora, el tipo filtro a implementar, en este caso una pasa banda, la amplitud de la pasa banda y la amplitud de la banda supresora.

Con estos parámetros, calculamos el orden del filtro (longitud del filtro), esto lo realizamos mediante la fórmula de Kaiser:

$$M = \lceil (-20 \log_{10} \sqrt{\delta_{1\delta 2}} - 13) / (14,6 \Delta f) \rceil + 1$$

Ya contando con la longitud del filtro y con los parámetros de diseño definidos procedimos a usar la función *remez* para calcular los coeficientes del filtro. Finalmente, usamos la transformada de Fourier para obtener el dominio de la frecuencia e implementar el filtro con la señal de prueba del canto de aves.

C. Filtros IIR

Para el desarrollo del filtro IIR usamos el método de transformación bilineal. Para esto, primero obtenemos un vector de frecuencias que va desde 0 hasta la frecuencia de muestreo. Además, se obtiene el índice de la frecuencia de corte. Los parámetros de diseño son: el orden del filtro, la frecuencia de muestreo, y la frecuencia de corte. Con estos parámetros se obtiene la frecuencia angular y el tiempo de muestreo. Se calcula el *prewarp* haciendo uso de la fórmula:

$$\Omega = (2/T) * \tan(\omega/2)$$

Posteriormente, se usa la función *butter*, y se calculan los coeficientes del filtro. Para obtener los coeficientes del filtro bilineal se usa la función *bilinear*. Además, para obtener el filtro se hace uso de la función *freqz*.

Posteriormente, esta señal recibida debe ser procesada sumando unos y ceros, para ello se ha creado una función en la que se calcula la frecuencia de muestreo, en la que se encuentra la muestra. Se forma un vector de unos con esta muestra, que va desde cero hasta este valor calculado de la muestra. El filtro creado previamente se adjunta a este vector. Teniendo en cuenta que el filtro debe crearse con un determinado número de muestras, se concatena un vector de ceros con el previamente creado con el vector de unos, y de esta forma el filtro ya tendría el número de muestras especificado. Finalmente, es el filtro el que se multiplica por la transformada para poder filtrar

V. RESULTADOS

A. Filtros FIR

Se generó el código del filtro FIR por ventanas. Se usó la ventana *hamming*, la cual puede verse en la figura 5. Además, en la figura 6 se puede observar la implementación de la ventana *hamming* y la ganancia y magnitud del filtro FIR por ventanas.

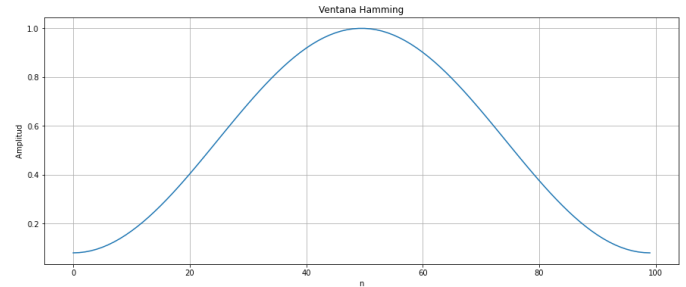


Figura 5. Ventana hamming.

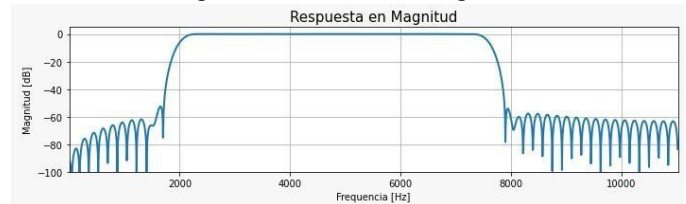


Figura 6. Magnitud del filtro FIR por ventanas.

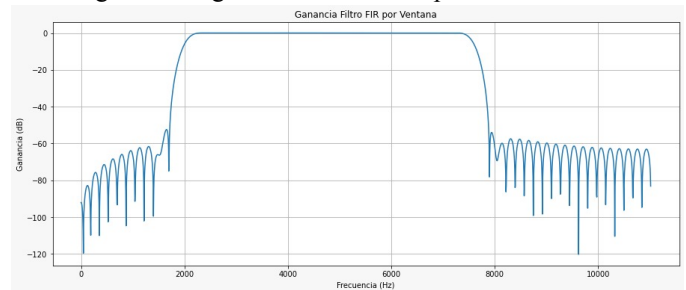


Figura 7. Ganancia filtro FIR.

Para este proceso usamos un filtro pasa banda entre 2 KHz y 7,6 KHz. Igualmente, en la figura 8 podemos ver igualmente la fase del filtro FIR por ventanas.

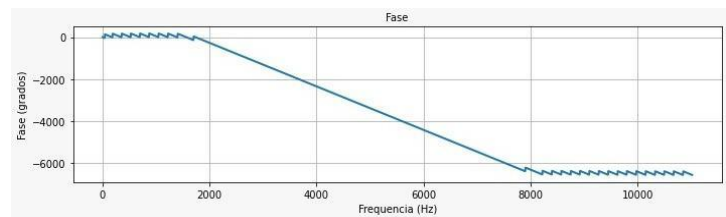


Figura 8. Fase del filtro FIR por ventanas.

Posterior a esto, aplicamos el filtro. En la figura 9 y 10 podemos observar la respuesta en tiempo y en frecuencia de la señal posterior a ser sometida al filtro.

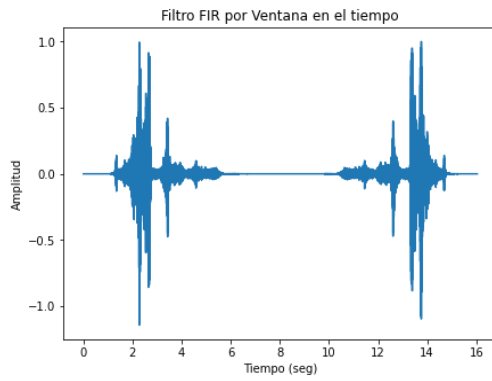


Figura 9. Respuesta en el tiempo implementando el filtro FIR

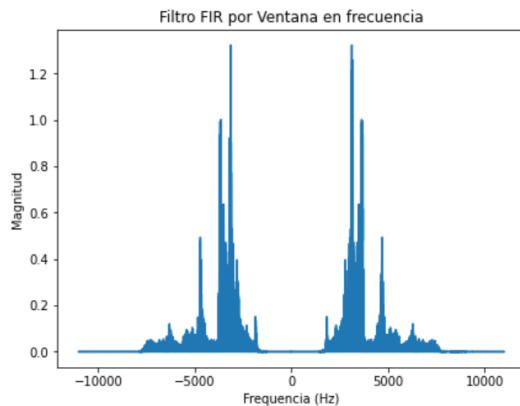


Figura 10. Respuesta en la frecuencia implementando el filtro FIR.

Finalmente, se procesó esta señal resultante por un espectrograma para poder analizar si efectivamente el filtro produjo los efectos deseados. Igualmente, la señal resultante fue guardada como un tipo de señal de audio.

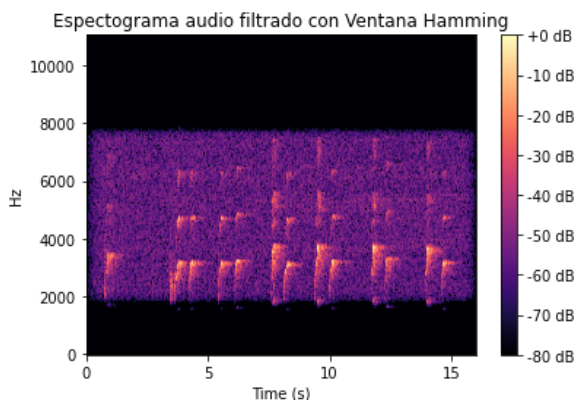


Figura 11. Espectrograma filtro FIR con ventanas.

En este espectrograma podemos ver dos efectos. Por una parte, se puede observar que el ruido gaussiano fue significativamente eliminado. Además de esto, las frecuencias bajas fueron minimizadas.

B. Método de Chebyshev

En la figura 12 se muestra el filtro pasa banda en el dominio de la frecuencia. Se puede observar que el rango es desde 2000 Hz hasta 7600 Hz. Las otras frecuencias tienen un valor de 0, las cuales se quieren eliminar. Además de esto, en la figura 13 se puede ver la fase del filtro.

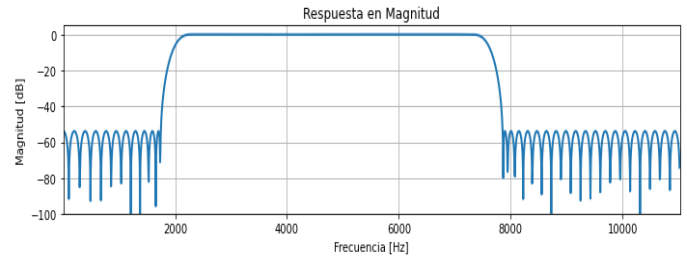


Figura 12. Filtro FIR de Chebyshev.

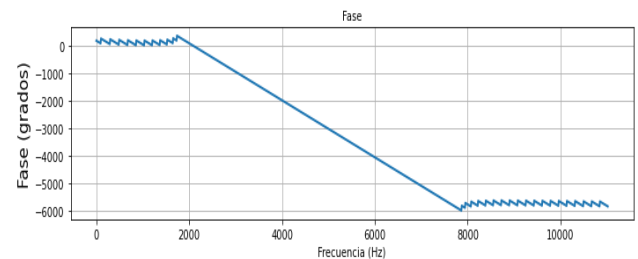


Figura 13. Fase del filtro FIR por el método de la aproximación de Chebyshev.

Posterior a esto, podemos observar en la figura 14 y 15 los resultados en el tiempo y en la frecuencia del filtrado por el método de Chebyshev.

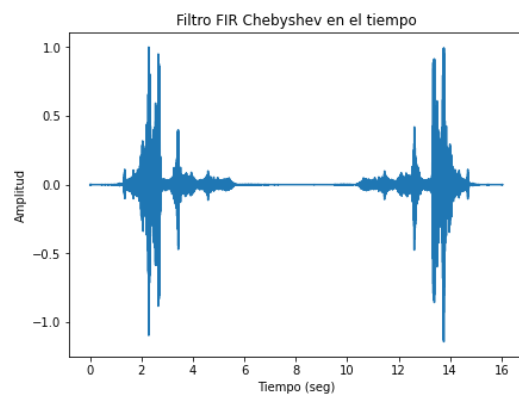


Figura 14. Respuesta en el tiempo por el método de la aproximación de Chebyshev.

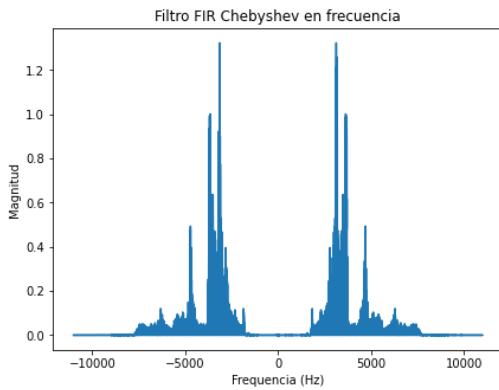


Figura 15. Respuesta en la frecuencia por el método de la aproximación de Chebyshev.

En la figura 15, además, podemos observar cómo se atenúa significativamente y casi a 0 después de las frecuencias de corte. Esto significa que al escuchar el audio tendríamos que escuchar mínimamente o de forma casi nula el ruido de los insectos.

Finalmente, se procesó la señal filtrada y se obtuvo un espectrograma en la figura 16.

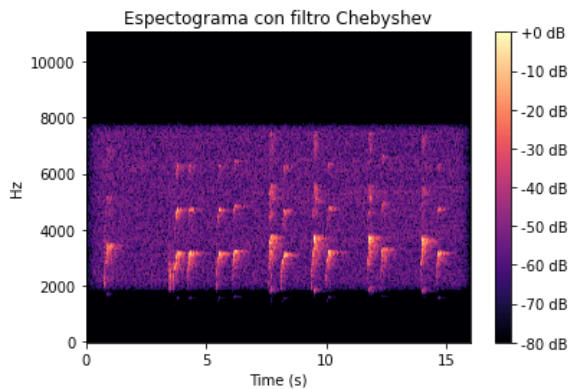


Figura 16. Espectrograma filtro FIR por el método de la aproximación de Chebyshev.

En la figura 16 podemos observar que no se elimina el ruido gaussiano completamente, aunque se elimina significativamente.

Usando los resultados de la implementación del filtro FIR por ventanas y por Chebyshev, se llegó a la conclusión de que, aunque se presente un mayor ruido en el método de Chebyshev, este es idealmente mejor para este tipo de implementación debido a que por la naturaleza del método podemos controlar parámetros como las frecuencias de corte y atenuaciones, lo que nos permite una mayor precisión. Además de esto, nos permite minimizar el error máximo mediante el criterio de aproximación en el que se basa. Sin embargo, en estas pruebas en especial, obtuvimos un mejor resultado tanto en el audio como en el espectrograma del

método de ventanas, por lo cual este método sería a corto plazo implementado sin menores complicaciones y posibles ajustes que sí tendrían que hacerse en el método de aproximación de Chebyshev.

C. Filtros IIR

Se implementó el filtro por transformación bilineal. Se puede ver la ganancia, magnitud, y fase del filtro IIR en las figuras 17, 18, y 19. De hecho, se nota que la fase tiende a ser no lineal.

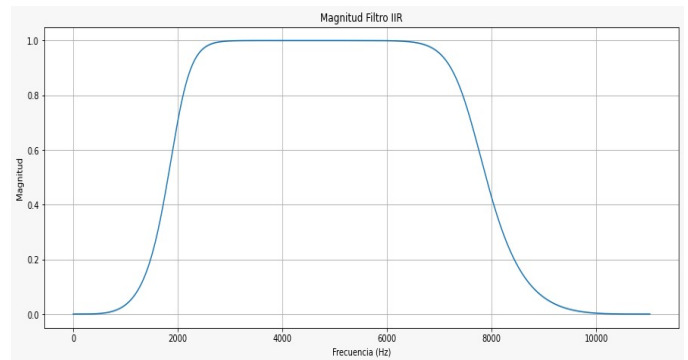


Figura 17. Magnitud del filtro IIR.

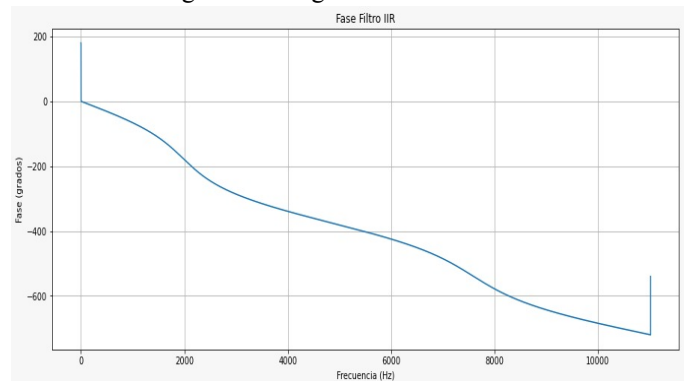


Figura 18. Fase del filtro IIR.

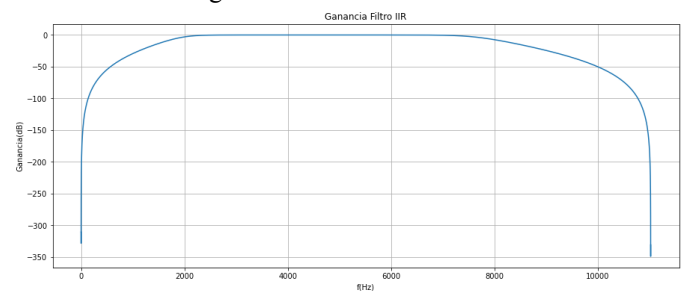


Figura 19. Ganancia filtro IIR.

Posterior a esto, se procesó el audio por el filtro IIR. La respuesta en el tiempo como en la frecuencia de la señal de audio pasada por el filtro IIR se puede ver en las figuras 20 y 21.

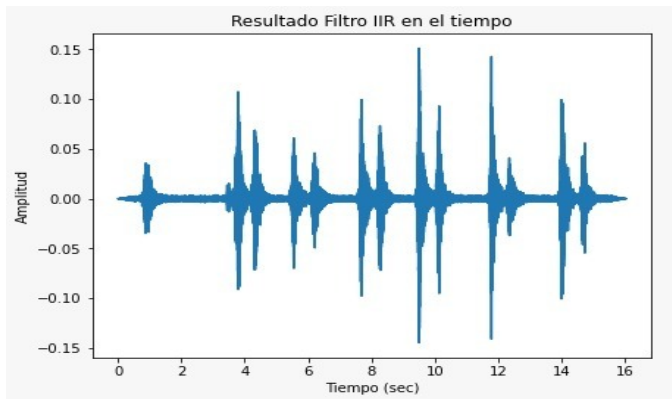


Figura 20. Respuesta en el tiempo del filtro IIR.

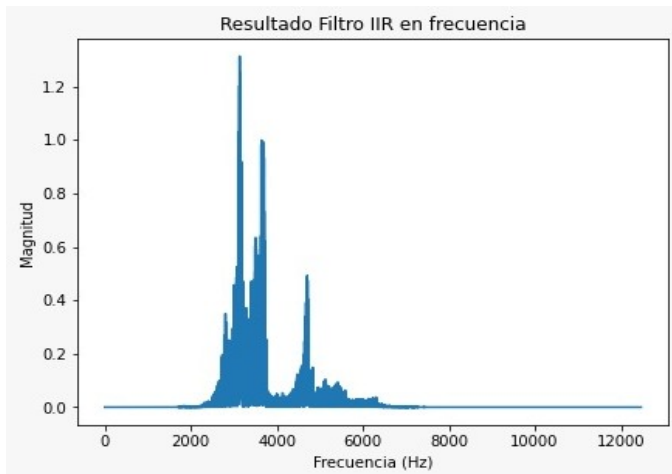


Figura 21. Respuesta en la frecuencia del filtro IIR.

En este caso, se puede observar que las frecuencias por fuera de la banda fueron atenuadas considerablemente, por lo que esta implementación fue exitosa.

Finalmente, procesamos la señal resultante del filtro y la convertimos a un espectrograma, el cual puede verse en la figura 22.

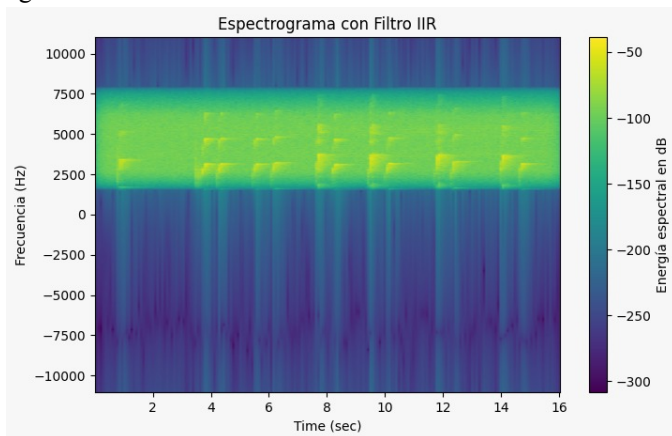


Figura 22. Espectrograma del filtro IIR.

En este espectrograma podemos notar cómo obtenemos tramos de la señal en las frecuencias deseadas, tal como fue visualizado en la respuesta en frecuencia.

D. Tablas de Filtros

Se consignó y comparó las frecuencias de corte y las atenuaciones obtenidas usando cada uno de los filtros. Esto se puede ver en la tabla 1.

| | Frecuencia de corte baja | Frecuencia de corte alta | Atenuación |
|-----------------------------------|--------------------------|--------------------------|------------|
| Filtro FIR por ventanas esperado | 2 KHz | 7,6 KHz | -60 dB |
| Filtro FIR por ventanas real | 1,85 KHz | 7,85 KHz | -70 dB |
| Filtro FIR por Chebyshev esperado | 2 KHz | 7,6 KHz | -55 dB |
| Filtro FIR por Chebyshev real | 1,85 KHz | 7,9 KHz | -70 dB |
| Filtro IIR esperado | 2 KHz | 7,6 KHz | -150 dB |
| Filtro IIR real | 2,05 KHz | 7,45 KHz | -200 dB |

Tabla 1. Frecuencias de corte y atenuaciones ideales y reales para cada uno de los filtros.

VI. CONCLUSIONES

- El diseño de ventana de los filtros FIR no nos permite tener mucho control sobre parámetros como la frecuencia de corte y la atenuación del filtro diseñado.
- El método Chebyshev permite un nivel muy alto de control sobre los parámetros y especificaciones del diseño del filtro.
- El primer filtro FIR tiene una fase lineal independientemente del método utilizado (Ventana o Chebyshev). Por otro lado, el filtro IIR se obtiene al tener una fase no lineal.

- Al comparar los resultados de los filtros por su audio filtrado pudimos notar una disminución en la magnitud de la señal de audio resultante, esto con el filtro IIR. Esto se nota ya que tiene una atenuación de hasta -150 dB, en comparación a los otros filtros cuya atenuación es de cerca de 80 dB. Igualmente, el filtro FIR con ventana Hamming se escuchó con una interferencia de salida un poco mayor al caso de usar el método de aproximación de Chebyshev. De esta manera, y teniendo en cuenta los resultados de frecuencias de corte de la tabla 1, podemos decir que el mejor filtro para remover el sonido de los insectos es el Chebyshev.

VII. DISTRIBUCIÓN DEL TRABAJO

Cada miembro del equipo tiene experiencia en la realización de las distintas partes del laboratorio. Sin embargo, a continuación, se describirán los aportes distintivos de cada miembro:

A. Juan José Restrepo: Desarrolló e implementó el programa que da solución al uso de filtros IIR. Además, ayudó a implementar el método de Chebyshev.

B. Juan David Vargas: Desarrolló e implementó el programa que da solución al uso de filtros FIR. Además, ayudó a implementar el método de Chebyshev.

Adicionalmente, cada miembro colaboró en el informe desde lo que realizó en Python. Los resultados fueron compartidos y en muchos casos se ejecutaron los códigos en medio de las reuniones de equipo.

VIII. REFERENCIAS

- [1] Benítez, H., 2022. Correlación de señales.
- [2] Benítez, H., 2022. Introducción.
- [3] PIL. (2019). *Crop a part of the image with Python, Pillow (trimming)* | [note.nkmk.me](https://note.nkmk.me/en/python-pillow-image-crop-trimming/).
<https://note.nkmk.me/en/python-pillow-image-crop-trimming/>
- [4] Lewis, C. (2021). *How to Create & Understand Mel-Spectrograms* | by Christopher Lewis | Medium.
<https://importchris.medium.com/how-to-create-understand-mel-spectrograms-f17634991056>
- [5] MLK. (2021). *4 Ways to Convert Image to Grayscale in Python using Skimage, Pillow and OpenCV* - MLK - Machine Learning Knowledge.
<https://machinelearningknowledge.ai/ways-to-convert-image-to-grayscale-in-python-using-skimage-pillow-and-opencv/>
- [6] MTU PHISYCS. (2018). Physics of music - notes. Frequencies for equal-tempered scale. [online] Available at: <https://pages.mtu.edu/~suits/notefreqs.html>.
- [7] Benítez, H., 2022. Transformada discreta de Fourier.
- [8] Benítez, H., 2022. Transformada rápida de Fourier (FFT).