

Desarrollo de un ecualizador paramétrico de fase lineal y comparación de su aplicación en procesos de calibración de estudios frente a ecualizadores con respuesta en fase no lineal.

Santiago Agudelo Elejalde, ✉ santiagoagudeloelejalde@gmail.com

Trabajo de Grado presentado para optar al título de Ingeniero de Sonido

Asesor: Juan Pablo Hormiga Leal, Magíster (MSc) EN Producción de Audio



Universidad de San Buenaventura Colombia

Facultad de Ingeniería

Ingeniería de Sonido

Medellín, Colombia

2017

Citar/How to cite	[1]	
Referencia/Reference	[1]	S. Agudelo Elejalde. “Desarrollo de un ecualizador paramétrico de fase lineal y comparación de su aplicación en procesos de calibración de estudios frente a ecualizadores con respuesta en fase no lineal.”, Trabajo de grado Ingeniería de Sonido, Universidad de San Buenaventura Medellín, Facultad de Ingenierías, 2017.
Estilo/Style:		
IEEE (2014)		



Bibliotecas Universidad de San Buenaventura



Biblioteca Digital (Repositorio)
<http://bibliotecadigital.usb.edu.co>

- Biblioteca Fray Alberto Montealegre OFM - Bogotá.
- Biblioteca Fray Arturo Calle Restrepo OFM - Medellín, Bello, Armenia, Ibagué.
- Departamento de Biblioteca - Cali.
- Biblioteca Central Fray Antonio de Marchena – Cartagena.

Universidad de San Buenaventura Colombia

Universidad de San Buenaventura Colombia - <http://www.usb.edu.co/>

Bogotá - <http://www.usbbog.edu.co>

Medellín - <http://www.usbmed.edu.co>

Cali - <http://www.usbcali.edu.co>

Cartagena - <http://www.usbctg.edu.co>

Editorial Bonaventuriana - <http://www.editorialbonaventuriana.usb.edu.co/>

Revistas - <http://revistas.usb.edu.co/>

TABLA DE CONTENIDO

RESUMEN	8
ABSTRACT	9
I. INTRODUCCIÓN	10
II. PLANTEAMIENTO DEL PROBLEMA	11
A. Antecedentes.....	11
III. JUSTIFICACIÓN.....	15
IV. OBJETIVOS	16
A. Objetivo General	16
1). Objetivos Específicos.....	16
V. HIPÓTESIS	17
A. Hipótesis de Trabajo.....	17
B. Hipótesis Nula	17
C. Hipótesis Alterna	17
VI. MARCO TEÓRICO.....	18
A. Sistema de reproducción.....	18
B. El recinto como espacio de escucha	19
C. El ecualizador	20
1) Control de tono.....	21
2) Ecualizador Gráfico	21
3) Ecualizador paramétrico	22
D. Filtros FIR	23
1) Diseño de filtros FIR por el método de ventanas.....	23
2) Diseño de filtros FIR por el método de muestreo en frecuencia.....	24
E. Filtros IIR.....	24
F. Fase lineal.....	25
G. Respuesta al impulso	25
H. La era digital.....	26
VII. METODOLOGÍA	27
A. Desarrollo del ecualizador	27
1) Ecualizador de fase lineal	27
2) Ecualizador de fase no lineal	28
B. Preparación del estudio.....	29
C. Proceso de medición	30
VIII. RESULTADOS	33
A. Ecualizadores.....	34

B. Mediciones.....	35
C. consumo de memoria RAM.....	45
IX. DISCUSIÓN	47
X. CONCLUSIONES.....	52
REFERENCIAS	54
ANEXOS.....	57

LISTA DE TABLAS

Tabla I. Parámetros para configuración de un ecualizador que corrigen la respuesta en frecuencia medida.....	36
--	----

LISTA DE FIGURAS

Fig. 1. Ecuador gráfico por octavas y su respuesta en frecuencia con todos los controles en la posición central.....	22
Fig. 2. Principio de funcionamiento de un filtro FIR (a) y un filtro IIR (b).....	23
Fig. 3. Ubicación del punto de medición y los altavoces.	29
Fig. 4. Curvas de igual sonoridad desarrolladas por Fletcher y Munson.	30
Fig. 5. Tolerancia en la respuesta en frecuencia para el punto de escucha del recinto.	31
Fig. 6. Distribución de los puntos de medición de acuerdo al manual de usuario del sistema ARC2 de Ik Multimedia.	32
Fig. 7. Interfaz gráfica del ecualizador de fase lineal desarrollado en MatLab.....	33
Fig. 8. Diagrama de polos y ceros para la función de transferencia del ecualizador con todos los controles de ganancia ubicados en cero dB obtenida a partir de la función <i>zplane</i> de Matlab.....	34
Fig. 9. Promedio de respuesta en frecuencia para el altavoz izquierdo del estudio C de la universidad de San Buenaventura con un suavizado de 1/6 de octava.	35
Fig. 10. Promedio de respuesta en frecuencia para el altavoz derecho del estudio C de la universidad de San Buenaventura con un suavizado de 1/6 de octava.	35
Fig. 11. Promedio de respuesta en frecuencia para los altavoces izquierdo y derecho del estudio C de la universidad de San Buenaventura con un suavizado de 1/6 de octava.	36
Fig. 12. Respuesta deseada al aplicar la ecualización al altavoz izquierdo.....	37
Fig. 13. Respuesta deseada al aplicar la ecualización al altavoz derecho.	37
Fig. 14. Retraso de fase del ecualizador de fase lineal desarrollado para los canales izquierdo y derecho.	38
Fig. 15. Retraso de fase del ecualizador de fase no lineal desarrollado para el canal izquierdo.	38
Fig. 16. Retraso de fase del ecualizador de fase no lineal desarrollado para el canal derecho.....	38
Fig. 17. Retraso de fase del ecualizador analógico para el canal izquierdo.	39
Fig. 18. Retraso de fase del ecualizador analógico para el canal derecho.	39
Fig. 19. Respuesta en frecuencia de los ecualizadores digitales para el canal izquierdo.....	40
Fig. 20. Respuesta en frecuencia de los ecualizadores digitales para el canal derecho	40
Fig. 21. Respuesta en frecuencia del ecualizador analógico dbx 2231 aplicado al canal izquierdo.....	40
Fig. 22. Respuesta en frecuencia del ecualizador analógico dbx 2231 aplicado al canal derecho.	41

Fig. 23. Respuesta en frecuencia del altavoz izquierdo antes y después de aplicar el ecualizador de fase lineal.....	41
Fig. 24. Respuesta en frecuencia del altavoz derecho antes y después de aplicar el ecualizador de fase lineal.....	41
Fig. 25. Respuesta en frecuencia de ambos altavoces antes y después de aplicar el ecualizador de fase lineal.....	42
Fig. 26. Respuesta en frecuencia del altavoz izquierdo antes y después de aplicar el ecualizador de fase no lineal.....	42
Fig. 27. Respuesta en frecuencia del altavoz derecho antes y después de aplicar el ecualizador de fase no lineal.....	42
Fig. 28. Respuesta en frecuencia de ambos altavoces antes y después de aplicar el ecualizador de fase no lineal.....	43
Fig. 29. Respuesta en frecuencia del altavoz izquierdo antes y después de aplicar el ecualizador analógico.....	43
Fig. 30. Respuesta en frecuencia del altavoz derecho antes y después de aplicar el ecualizador analógico.....	43
Fig. 31. Respuesta en frecuencia de ambos altavoces antes y después de aplicar el ecualizador analógico.....	44
Fig. 32. Superposición de la respuesta en frecuencia después de aplicar los ecualizadores al canal izquierdo.....	44
Fig. 33. Superposición de la respuesta en frecuencia después de aplicar los ecualizadores al canal derecho.....	44
Fig. 34. Respuesta en frecuencia de la sala con ambos altavoces simultáneamente después de aplicar cada uno de los ecualizadores.....	45
Fig. 35 Información de consumo de memoria en kilobytes suministrado por el administrador de tareas de Windows. LinearPhase_EQ.exe corresponde al ecualizador de fase lineal y FNL.exe al ecualizador de fase no lineal.....	46
Fig. 36 Información del porcentaje de consumo de memoria RAM suministrado por el administrador de tareas de Windows. LinearPhase_EQ.exe corresponde al ecualizador de fase lineal y FNL.exe al ecualizador de fase no lineal.....	46
Fig. 37 Resultado al sumar dos señales de acuerdo a la diferencia de fase entre ambas.....	50

RESUMEN

Los ecualizadores analógicos y la mayoría de ecualizadores digitales generan cambios en la fase de la señal, lo que puede llevar a una interferencia destructiva y modificar la respuesta en frecuencia de la señal; es por ello que en procesos como masterización en donde se necesitan resultados precisos, se hace uso de ecualizadores de fase lineal. El objetivo de esta investigación es comparar los resultados en la respuesta en frecuencia obtenidos al calibrar un sistema de altavoces estéreo dispuestos en el estudio C de la universidad de San Buenaventura Medellín con ecualizadores de fase lineal y no lineal. Para ello se desarrolló un algoritmo en Matlab con el que se obtuvieron dos ecualizadores con dicho comportamiento de fase, para posteriormente realizar mediciones en el punto de escucha y obtener la respuesta en frecuencia a partir del software Room EQ Wizard y Smaart Live 7, observando diferencias entre los ecualizadores desarrollados y el ecualizador analógico dbx 2231.

Palabras clave: Filtro FIR, Fase lineal, Calibración, Respuesta en frecuencia.

ABSTRACT

Analog equalizers and most of the digital equalizers used in audio systems lead to a phase change in the signal, because of that, both, destructive interference and frequency response changes appear; for this reason, a mastering engineers, who needs an accurate result, use linear phase equalizers in order to obtain the most precise signal processing. This investigation was made in order to compare the frequency response obtained as a result of the calibration of a stereo sound system placed in the Studio C of the San Buenaventura University with a linear and non-linear phase equalizers. A Matlab algorithm was developed and both, a linear phase EQ and non-linear phase EQ were obtained. After that, a frequency response measurement was performed using two different software applications (Room EQ Wizard and Smaart Live 7) in order to analyze the differences between the equalizers created in Matlab and the dbx 2231 analog equalizer.

Keywords: FIR filter, Linear phase, Calibration, Frequency response.

I. INTRODUCCIÓN

Recintos tales como estudios de grabación, mezcla y/o masterización, requieren de una respuesta en frecuencia lo más plana posible, es por ello que sus condiciones acústicas y equipos de audio deben garantizarlo. Sin embargo, la interacción entre sonido directo y reflexiones, así como la no linealidad absoluta en la respuesta en frecuencia de los equipos de reproducción, impiden que se tenga una respuesta completamente plana, haciendo necesaria la calibración de un sistema por medio del análisis acústico y el uso de ecualizadores. Esta investigación busca desarrollar un algoritmo en MatLab con el que se obtenga un ecualizador paramétrico de fase lineal y comparar su implementación en procesos de calibración de estudios (entendido como el proceso de ecualización en la etapa de reproducción de un sistema de sonido, que permita obtener una respuesta en frecuencia lo más plana posible) con respecto a los ecualizadores analógicos o digitales con una respuesta en fase no lineal, esto con el fin de determinar si este tipo de ecualizadores realmente constituyen una alternativa mucho más precisa y confiable sin que esto implique un alto consumo computacional que conlleve a problemas de latencia en la señal de salida.

II. PLANTEAMIENTO DEL PROBLEMA

Los ecualizadores analógicos y gran parte de los digitales generan cambios en la fase de la señal, llevando a que puedan presentarse problemas como distorsión de fase. Es por ello que la ecualización de fase lineal se convierte en una alternativa importante a la hora de buscar resultados mucho más precisos en etapas de postproducción, como la masterización. La presente investigación pretende dar respuesta a la pregunta: ¿Qué ventajas ofrecería aplicar un ecualizador de fase lineal en los procesos de calibración de un sistema de reproducción de audio comparado con ecualizadores digitales no lineales y analógicos? De manera que se pueda obtener una nueva alternativa para la corrección de los errores en la respuesta en frecuencia de un estudio. Sin embargo, debe considerarse además el gasto computacional que implica el procesamiento de audio a través de este tipo de ecualizadores por lo cual se determinará si en realidad es una alternativa viable que no comprometa el rendimiento de los sistemas de audio digital.

A. Antecedentes

Los procesos de calibración de sistemas de reproducción en un estudio son el resultado de décadas de estudios y avances que comenzaron por percepciones subjetivas y necesidades basadas en la percepción del oído humano. Todos los espacios donde existen sistemas de reproducción necesitan calibrarse, pues no existen altavoces ideales, con respuesta en frecuencia completamente plana, y mucho menos ajenos al impacto que genera el recinto donde son colocados. La geometría de la sala, el área, el volumen, el material de las superficies y elementos que allí están, influyen en el resultado final y en la respuesta en frecuencia que se obtiene allí; de hecho, cada lugar del recinto “suena” diferente y es por eso que las calibraciones se hacen en un punto específico de escucha para espacios como estudios de grabación, mezcla o masterización; o se hace un promedio entre varios puntos para calibrar salas donde existen múltiples puntos de escucha, tales como cinemas o teatros. El objetivo de la calibración de los sistemas de audio en un recinto no es simular condiciones anecóicas, y depende exclusivamente del fin para el cual este está diseñado; en estudios de producción y postproducción de audio el objetivo es obtener una respuesta en frecuencia lo más plana posible, objetivo que se alcanza con un sistema de altavoces cuyas características permitan obtener dicha respuesta o por lo menos acercarse a ella, un adecuado acondicionamiento acústico y un proceso de ecualización que corrija los detalles que los dos anteriores no logran satisfacer [1].

La ecualización digital apareció en los años 90 con el desarrollo de los sistemas digitales basados en la teoría de procesamiento digital de señales (DSP). Sin embargo no eran procesos muy efectivos debido a las limitaciones tecnológicas de la época, cuyo problema más significativo fue su largo tiempo de latencia, producto de un alto consumo de recursos informáticos. El desarrollo de la teoría de los filtros de respuesta al impulso finita (FIR) representó un gran impacto para la era digital de los ecualizadores, cuyo valor reposaba más que en tener una respuesta de fase lineal, en la posibilidad de manejar independientemente la fase y la magnitud de la señal, lo que permitió hacer ecualizaciones mucho más precisas [2].

Aunque solo hasta finales del siglo XX aparecieron los ecualizadores digitales, son el resultado de desarrollos que se remontan a la década de 1960, cuando aparece la espectrometría en el dominio del tiempo -desarrollada por el estadounidense Dick Heyser-, e incluso teorías mucho más antiguas como la transformada rápida de Fourier (FFT por sus siglas en inglés), desarrolladas por el francés Jean-Baptiste Joseph Fourier en el siglo XVIII. Estos elementos sentaron las bases para entender más a fondo el comportamiento de los sistemas de sonido y dar inicio a procesos de calibración que solucionaron problemas como la inteligibilidad [3]. Tal es el caso de la empresa Dolby Laboratories, Inc. la cual en 1977 comenzó a calibrar las salas de cine aplicando un análisis espectral y ecualización analógica por tercios de octava [4].

En 1997 Soterios Salamouris y Vassilis Tsakiris aprovecharon los avances del audio digital y el poder de la teoría de filtros desarrollada hasta la fecha para crear un sistema capaz de identificar la respuesta en frecuencia de un recinto y aplicar una curva de ecualización que compensara las falencias de dicha respuesta. Su implementación se basó en un prototipo ya existente llamado DEAS y separó el sistema en hardware y software, diseñado para Windows 3.1 y Windows 95, el cual permitía trabajar a frecuencias de muestreo de 44.100 Hz y 48.000 Hz. Para obtener la respuesta en frecuencia de la sala, el sistema se basaba en la respuesta al impulso y la teoría de convolución; el funcionamiento del ecualizador se basaba en el uso de filtros FIR lo que permitía al usuario escoger entre una respuesta en fase lineal, o mínima. Sin embargo, el módulo de respuesta en fase lineal implicaba un consumo mayor de recursos y por consiguiente problemas de latencia en la señal, uno de los principales problemas del diseño de Salamouris y Tsakiris, seguido por los altos costos económicos que representaba [5].

El 2003 Andrew Goldberg y Aki Mäkivirta publicaron un análisis de un algoritmo de calibración automática de la respuesta en frecuencia de parlantes activos de una, dos y tres vías. Dicho análisis mostró un proceso mucho más eficiente en cuanto a consumo de recursos con respecto al anterior, pero limitado en cuanto a calidad, pues la ecualización no tenía una respuesta en fase lineal. El algoritmo se dividía en cinco módulos, y su funcionamiento se basaba en una conexión en serie, es decir, cada uno de las etapas, dependía completamente de la anterior. El primer módulo realzaba brevemente la energía presente en frecuencias bajas para evitar una sobre excitación de dicha región al momento de aplicar el filtro correctivo, el segundo módulo, ajustaba los niveles de frecuencias medias y altas, dando paso al tercer módulo encargado de ajustar las frecuencias bajas, para luego eliminar el realce aplicado inicialmente; por último, se ajustaba un filtro pasa bajos cuya frecuencia de corte estaba limitada por la respuesta en frecuencia del altavoz. Se trata entonces de un algoritmo eficiente, pero limitado en cuanto al número de bandas de frecuencias que permite ajustar. Sin embargo, se obtuvieron resultados satisfactorios, con un consumo menor de recursos computacionales y un menor costo económico [6].

El problema del alto consumo informático y la necesidad de obtener ecualizadores de fase lineal fue mejorado por Riitta Väänänen y Jarmo Hiipakka en 2007, quienes desarrollaron un ecualizador de alta fidelidad (fase lineal) aplicando filtros FIR. La eficiencia del algoritmo radica en la división de la señal en dos o más bandas, cada una de ellas ecualizada por separado para luego volver a unir las y reconstruir la señal, no obstante, esta etapa de división implicaba decidir entre el uso de filtros FIR o IIR. Los primeros logran evitar la distorsión de fase, pero no eliminan completamente la distorsión de amplitud, mientras que los filtros IIR eliminan la distorsión de amplitud más no la de fase, por lo cual en el algoritmo desarrollado por Väänänen y Hiipakka se aplicaron filtros IIR para la división de la señal. La etapa de ecualización se realizó con filtros FIR, que como ya se ha dicho, a pesar de ser muy estables consumen demasiados recursos. Este algoritmo aplicaba una reducción del muestreo a la señal antes de ser procesada y en la etapa final volvía nuevamente a la frecuencia de muestreo original [7].

En el año 2008, sale al mercado el plugin Advance Room Correction system (ARC system), un sistema desarrollado por la compañía IK Multimedia, el cual obtiene la respuesta en frecuencia de los altavoces en un determinado espacio, a partir de mediciones en diferentes puntos de la sala, para luego aplicar una ecualización de acuerdo a los resultados obtenidos, y corregir los

puntos en el espectro en frecuencia que presentan problemas. No obstante, el proceso de ecualización es de fase no lineal [8].

Todo lo anterior necesita de un software que permita desarrollar el algoritmo y obtener un plugin que funcione a través de un sistema DAW. En 1996 Steinberg desarrolló el formato VST, el cual en sus inicios solo era aceptado por Cubase (un DAW desarrollado por esta misma compañía) pero a través de los años ha tomado fuerza y ha llegado a ser admitido en la mayoría de estaciones de trabajo digital. Hoy en día existen varios software que permiten desarrollar plugins VST y muchos de ellos se basan en el lenguaje de programación C o C++ [9]. CSound o su abreviación CS, es un compilador que permite programar específicamente para uso musical y de sonido, el cual basa su funcionamiento en el lenguaje de programación C. Es un software libre que funciona en todos los sistemas operativos y que permite crear interfaces de usuario y procesar en tiempo real. El sistema fue desarrollado por Barry Vercoe en 1985 [10].

III. JUSTIFICACIÓN

Algunos estudios de audio pueden hacer uso de sistemas hardware para procesar la señal de salida, hecho que implica un alto costo económico, considerando además que debe existir un ecualizador por cada uno de los altavoces de salida, es decir, en un sistema 5.1 por ejemplo, que dispone de seis altavoces, se necesita ese mismo número de ecualizadores para su calibración. Tener un software conlleva a una reducción significativa de los costos e incluso de espacio en el recinto, dando la posibilidad a estudios profesionales pequeños y/o “caseros” de tener una calibración de alta calidad a un bajo costo, es por ello que la mayoría de estos estudios hacen uso de plugins, no obstante, los ecualizadores empleados para ello generan cambios en la fase de la señal, que pueden resultar en problemas de distorsión de fase, o cambios indeseados en bandas de frecuencia que originalmente no presentaban problemas, por lo que a diferencia de los hardware y/o software de calibración y ecualizadores usados hoy en día para ajustar la respuesta en frecuencia de los sistemas de reproducción, el ecualizador a desarrollar permite hacer correcciones en el espectro de frecuencia sin alterar la fase de la señal, y por consiguiente eliminar dichos problemas; es por ello que se va a comparar el impacto al usar este tipo de ecualizadores y definir si realmente constituyen una alternativa mucho más precisa.

IV. OBJETIVOS

A. Objetivo General

- Desarrollar un ecualizador digital paramétrico de fase lineal y comparar la respuesta en frecuencia obtenida en un proceso de calibración de estudios empleando este ecualizador y un ecualizador digital y analógico de fase no lineal.

1). Objetivos Específicos

- Desarrollar un algoritmo en MatLab que permita obtener dos ecualizadores paramétricos, uno de fase lineal y otro no lineal.
- Emplear el ecualizador de fase lineal desarrollado en el proceso de calibración del estudio C de la universidad de San Buenaventura Medellín, y repetir dicho proceso empleando el ecualizador de fase no lineal y un ecualizador analógico dbx 2231.
- Comparar los resultados de las calibraciones con cada uno de los ecualizadores de manera que pueda analizarse la efectividad en la respuesta en frecuencia, y el consumo de memoria de cada uno.

V. HIPÓTESIS

A. Hipótesis de Trabajo

Un ecualizador de fase lineal constituye una solución mucho más precisa en la búsqueda de una respuesta en frecuencia ideal (plana) en un punto específico de escucha de un recinto, comparado con ecualizadores analógicos y o digitales de fase no lineal.

B. Hipótesis Nula

Un ecualizador de fase lineal no presenta diferencias significativas en la búsqueda de una respuesta en frecuencia ideal (plana) en un punto específico de escucha de un recinto comparado con ecualizadores analógicos y o digitales de fase no lineal.

C. Hipótesis Alternativa

El ecualizador de respuesta en fase lineal, aunque genera una corrección mucho más precisa de la respuesta en frecuencia, no es una herramienta eficiente a la hora de emplearse en procesos de calibración de estudios debido al alto consumo de recursos computacionales que implica el procesamiento de la señal de audio

VI. MARCO TEÓRICO

A. Sistema de reproducción

Para llegar a los altavoces, la señal debe pasar por una serie de procesos tanto analógicos como digitales, cada uno responsable de entregar la señal lo más fiel posible. Estos procesos tienen una influencia directa en el resultado. La cadena comienza en el DAW (digital audio workstation), de donde continúa a un conversor digital-análogo que permite la comunicación entre el ordenador y los altavoces. Tanto el conversor como los altavoces deben garantizar que no exista distorsión, entendida esta como una variación entre la señal de entrada y la de salida. Sin embargo, todos los equipos modifican la señal y su calidad puede calificarse de acuerdo al grado de distorsión que presentan. Los criterios principales que determinan la fidelidad de un equipo y/o sistema son: respuesta en fase, “qué tanto se ve afectada la fase de una señal al pasar por el equipo”; distorsión armónica, la cantidad de contenido en frecuencias que se agrega a la señal acústica de salida y que no se encontraba en la señal de entrada; relación señal a ruido, la relación existente entre la señal que se está pasando por el equipo y el ruido inherente a este (ruido medido en el equipo cuando no hay ninguna señal conectada a este); respuesta en frecuencia, en este caso no se mide la cantidad de frecuencias agregadas a la señal, sino los cambios de amplitud entre la señal de entrada y la de salida en todo el rango de frecuencias contenido en esta [11].

El proceso de conversión digital-análogo es entonces tan importante como los altavoces mismos, y tiene la importante tarea de entregar a estos una réplica exacta de la señal que está saliendo del sistema DAW. Idealmente, esta etapa de la cadena de reproducción debería tener una impedancia de salida de cero Ohms en todo el rango de frecuencias, de manera que se garantice una transferencia completa de la información, que no existan filtros que puedan modificar la señal que sale del conversor y donde toda la corriente se convierta en un voltaje equivalente (que no agregue distorsión armónica a la señal) [12]. Se debe entender que el audio digital se basa en el muestreo de la señal analógica a una frecuencia no inferior a 44100 Hz, fundamentada en el teorema de Nyquist, el cual establece en términos generales que para la reconstrucción de una señal analógica se debe contar con al menos dos puntos con respecto a la frecuencia más alta que se va a reconstruir (20000 Hz para el caso de las señales audibles para el ser humano); y se basa también en una profundidad en bits, entendida como la representación digital de la amplitud de la señal original. A mayor frecuencia de muestreo y mayor profundidad

en bits, más similar será la señal digital con respecto a la analógica. Hoy en día los sistemas digitales alcanzan frecuencias de muestreo de 192 KHz y profundidades de 32 Bits.

B. El recinto como espacio de escucha

En los procesos de pre y post producción, sea para la industria musical o medios audiovisuales, el recinto y altavoces juegan un papel fundamental a la hora de tomar decisiones y llegar al producto final; incluso se ha comprobado que diferentes recintos, con el mismo sistema de altavoces y bajo la misma señal de excitación, no suenan de la misma manera, demostrando que la arquitectura y el diseño acústico son tan importantes como la calidad del sistema de reproducción [13]. Para cualquiera que sea el fin, el recinto debe cumplir con características que permitan al músico, ingeniero y/o persona que trabaja allí estar seguro de que aquello que está escuchando es realmente lo que se está reproduciendo, es decir, que no se omita o cree información por sistemas de altavoces inadecuados o problemas en la arquitectura del recinto. Es por ello que los estudios de audio -independientemente de la función que desempeñen- deben cumplir con ciertas características básicas como: evitar las superficies reflejantes que influyan en el punto de escucha, ya que es posible que la interacción entre el sonido directo y las reflexiones que generan dichas superficies cambien la respuesta en frecuencia (sea por interferencia constructiva o destructiva); mantener un tiempo de reverberación bajo, de manera que no se pierda la directividad de las fuentes y mucho menos se vea afectada la inteligibilidad; evitar los paralelismos entre las paredes, el techo y el suelo pues esto puede generar resonancia para ciertas frecuencias; Mantener una simetría entre el lado izquierdo y derecho respecto al punto de escucha para garantizar que ambos oídos tengan la misma información (por lo menos, producto de la interacción del sonido en el recinto); lograr un aislamiento adecuado del recinto garantizando un nivel de presión sonora (NPS) mínimo, producto del ruido externo [14], [15].

Como ya se ha dicho, la arquitectura juega un papel fundamental en la funcionalidad de un recinto como espacio de escucha, y una de las características que tiene mayor relevancia a la hora de tomar decisiones sobre lo que debe hacerse para lograr un espacio ideal ya sea para procesos de grabación, mezcla y/o masterización es la respuesta en frecuencia, en donde alternativas como la difusión y la absorción (especialmente para frecuencias bajas) juegan un papel fundamental [16]. Sin embargo, lograr una respuesta en frecuencia completamente plana dependiendo exclusivamente del diseño acústico del espacio, es prácticamente imposible, y es ahí donde la ecualización se convierte en una alternativa fundamental. En una investigación

realizada por Tervo, Laukkanen, PÄtynen y Lokki [17] en la que se compararon nueve estudios de mezcla y masterización bajo el criterio de quince ingenieros de sonido, se demostró que las características a las que más se presta atención son localización, impresión de espacialidad, distancia, claridad y coloración, siendo este último completamente dependiente de la respuesta en frecuencia.

C. El ecualizador

Un ecualizador permite modificar la amplitud de una señal para un rango de frecuencia específico de acuerdo a la configuración y tipo de ecualizador que se esté usando. Sus usos van desde procesos correctivos hasta usos netamente artísticos que buscan dar un carácter específico a la señal. Son ampliamente usados en los procesos de posproducción tanto en la industria musical como en el audio para fines audiovisuales y es posible encontrarlos tanto analógicos como digitales. Su funcionamiento se basa en tres parámetros que determinan el grado de variación que se va a aplicar a la señal que pasa por este proceso:

- Frecuencia: las señales de audio generalmente contienen información desde los 20 Hz hasta los 20 kHz. los ecualizadores no afectan una frecuencia específica, este parámetro selecciona la frecuencia central de un ancho de banda específico que se va a ver modificado por el ecualizador.
- Ganancia: Como se dijo anteriormente, los ecualizadores modifican la amplitud en función de la frecuencia de la señal, este parámetro determina qué grado de amplificación o atenuación se va a aplicar a un rango de frecuencia específico.
- Factor Q: Este parámetro determina qué tan selectivo será el ancho de banda que se va a modificar a partir de la frecuencia central. El factor Q se calcula con la *ecuación 1* donde se observa una relación inversamente proporcional entre Q y el ancho de banda. Es decir, cuando mayor es el Q, menor será el ancho de banda y viceversa.

$$Q = \frac{Fc}{BW} \quad (1)$$

Donde Fc es la frecuencia central y BW el ancho de banda.

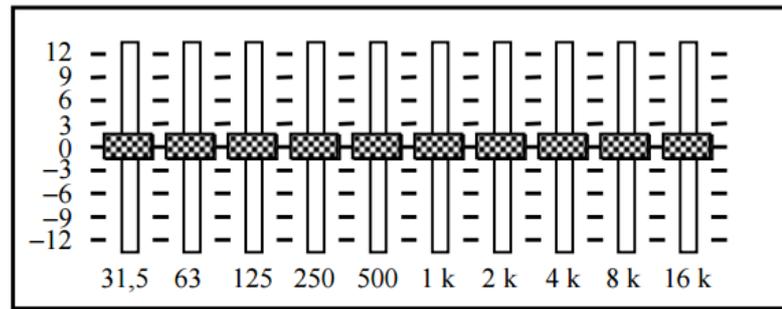
No todos los ecualizadores permiten variar estos tres parámetros, de hecho, existen básicamente tres tipos de ecualizadores y su clasificación se basa en la cantidad de características que permite modificar.

1) Control de tono

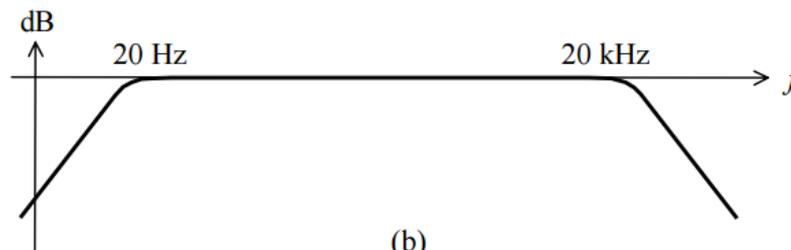
Este tipo de ecualizador permite solo modificar la ganancia para dos o tres rangos de frecuencia divididos en bajos, medios y altos, generalmente con frecuencias centrales de 100 Hz, 500 Hz y 1250 Hz respectivamente. En este caso se tiene un ancho de banda fijo que depende del fabricante del ecualizador [18].

2) Ecualizador Gráfico

Este tipo de ecualizador, aunque tiene frecuencias centrales fijas, cuenta con un número mayor de bandas; 10 bandas para aquellos cuya separación entre cada banda es por octava, y 31 bandas cuya separación entre banda es de 1/3 de octava. Generalmente su rango de amplitud está entre -12 dB y 12 dB, pero es posible encontrar ecualizadores con rangos de -6 dB a 6 dB o -18 dB a 18 dB [18]. En este tipo de ecualizador el factor Q, al igual que en el anterior, es fijo, sin embargo, es el tipo de ecualizador elegido para realizar la calibración de recintos.



(a)



(b)

Fig. 1. Ecualizador gráfico por octavas y su respuesta en frecuencia con todos los controles en la posición central (sin cambios en la amplitud para ninguna banda de frecuencia). Tomado de: *Ecualizadores. Facultad de ciencias exactas, ingeniería y agrimensura. Argentina.*

3) Ecualizador paramétrico

Como su nombre lo indica, este tipo de ecualizador permite modificar los tres parámetros (frecuencia, ganancia y factor Q) en un número de bandas determinadas por el fabricante. Es mucho más selectivo con respecto a los otros dos ecualizadores y es por ello que es el más utilizado en los procesos de posproducción de audio.

Al igual que los sistemas de reproducción, los ecualizadores también presentan características inherentes que ponen a prueba su fidelidad, tales como respuesta en fase, distorsión armónica, o relación señal a ruido; estas características están ligadas al tipo de ecualizador, es decir, si es analógico o digital [19], [18]. Los ecualizadores basan su funcionamiento a partir de filtros, la calidad y selectividad de cada una de las bandas depende del orden de estos filtros, es decir la pendiente de atenuación a partir de la frecuencia de corte (cada orden representa una disminución de 6 dB por octava). Los sistemas digitales permiten desarrollar ecualizadores a partir de filtros FIR e IRR, lo que lleva a la posibilidad de crearlos de manera que no afecten la fase de la señal, y por consiguiente se obtenga un resultado mucho más fiel.

D. Filtros FIR

Los filtros de respuesta al impulso finita (FIR por sus siglas en inglés), utilizan retrasos mínimos de la señal de entrada y la combinan con la señal de entrada original (Ver figura 2) para crear una nueva señal a la salida. Estos filtros pueden describirse a partir de la ecuación 2.

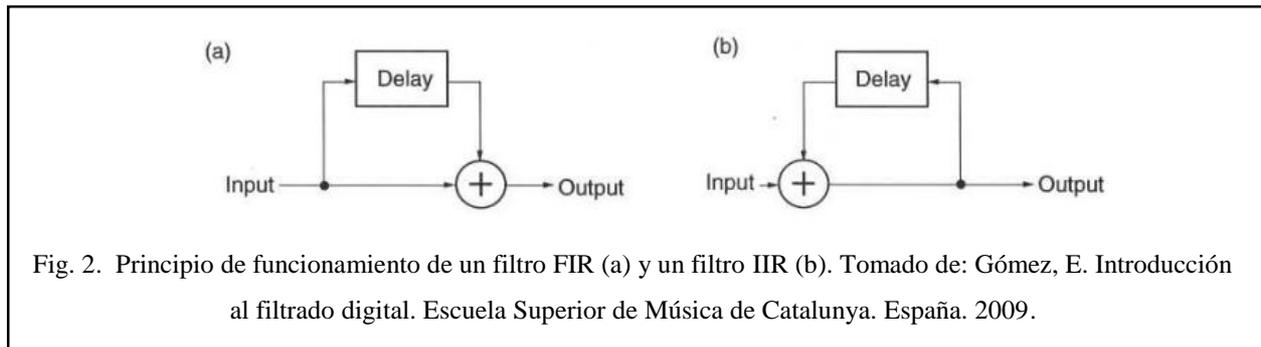


Fig. 2. Principio de funcionamiento de un filtro FIR (a) y un filtro IIR (b). Tomado de: Gómez, E. Introducción al filtrado digital. Escuela Superior de Música de Catalunya. España. 2009.

$$y[n] = a_0 \cdot x[n] + a_1 \cdot x[n - 1] + a_2 \cdot x[n - 2] + \dots + a_N \cdot x[n - N] \quad (2)$$

Donde y representa la señal de salida y x la de entrada.

Se trata de una combinación lineal entre la señal de entrada actual y muestras anteriores de la misma; como se observa en la ecuación 2, cada muestra anterior está multiplicada por un coeficiente a_i que determina las características del filtro.

1) Diseño de filtros FIR por el método de ventanas.

La respuesta al impulso de un filtro FIR en el dominio del tiempo, está representada por una función no causal, es decir, toma valores negativos en el eje x (tiempo). El método de ventanas propone casualizar la señal, es decir, retrasarla un número M de muestras de manera que la función no tome valores negativos en el eje horizontal, aunque este procedimiento genera un rizado en la banda de paso conocido como efecto Gibbs, es por ello que se hace uso de una ventana que corrige lo anterior y determina el nivel de atenuación del filtro. La aplicación de la ventana en el dominio del tiempo corresponde a la multiplicación de ambas funciones.

2) Diseño de filtros FIR por el método de muestreo en frecuencia.

El método se basa en la discretización o muestreo de la respuesta en frecuencia deseada $H_d(e^{jw_k})$, donde w_k corresponde a las frecuencias $k=0,1,\dots,N-1$, tomadas a lo largo del espectro, siendo N el número de muestras a tomar. La respuesta en frecuencia está dada por:

$$H(e^{jw_k}) = \sum_{n=0}^{N-1} h[n]e^{-jw_k n} \quad (3)$$

Considerando que el espectro de frecuencia está dado por el intervalo $[0, 2\pi]$, se puede decir que el espacio entre cada una de las muestras equivale a $\frac{2\pi}{N}$, llevando a la conclusión de que la magnitud de la respuesta en frecuencia se puede expresar como:

$$H(e^{jw_k}) = \sum_{n=0}^{N-1} h[n]e^{-j\frac{2\pi}{N}kn} \quad (4)$$

La ecuación 2 corresponde a la transformada discreta de Fourier (TDF), siendo así, es posible calcular los coeficientes de la respuesta al impulso $h[n]$, a través de la transformada inversa discreta de Fourier (TIDF).

E. Filtros IIR

Estos se conocen como filtros de respuesta al impulso infinita (IIR por sus siglas en inglés). El funcionamiento de este tipo de filtros se basa en la retroalimentación de la señal de salida (Ver figura 2), lo que hace que se puedan obtener respuestas más complejas con menos consumo de recursos digitales. Esta retroalimentación hace que circule constantemente energía en el circuito y la respuesta al impulso para este tipo de filtros tienda a ser infinita. La ecuación que describe el funcionamiento de estos filtros es:

$$y[n] = a_0 \cdot x[n] + a_1 \cdot x[n-1] + a_2 \cdot x[n-2] + \dots + a_N \cdot x[n-N] + -b_1 \cdot y[n-1] - b_2 \cdot y[n-2] - b_3 \cdot y[n-3] - \dots - b_M \cdot y[n-M] \quad (5)$$

Donde y representa la señal de salida y x la de entrada.

A diferencia de la ecuación de los filtros FIR, en esta, la salida está expresada en muestras anteriores tanto de la entrada como de la salida.

Comparando ambos filtros se puede evidenciar que, aunque los IIR proporcionan ordenes de filtros mucho mayor con menos consumo de recursos digitales, tienden a ser inestables debido a la retroalimentación de la señal de salida, mientras que los filtros FIR brindan la posibilidad de obtener respuesta de fase lineal [20].

F. Fase lineal

Los equipos de audio crean cambios en la fase de la señal, sin embargo, los sistemas digitales permiten crear procesos conocidos como ‘de fase lineal’, basados en la posibilidad de procesar todas las componentes del espectro en frecuencia de la señal de audio a la misma velocidad, permitiendo que todas las componentes lleguen al mismo tiempo a la salida del procesador y obteniendo así una réplica exacta de la señal de entrada (Con cambios en su magnitud de acuerdo al proceso realizado), lo que no ocurre en los dispositivos de fase no lineal, ya que la velocidad de los procesos varía a lo largo del espectro [21].

G. Respuesta al impulso

Por medio de la respuesta al impulso es posible predecir el comportamiento de un sistema frente a una señal de entrada, esto se calcula con la transformada inversa de Laplace de la función de transferencia (cociente entre la señal de salida y la de entrada de un sistema), definida como la respuesta de un sistema frente a una señal impulso (idealmente una señal infinitamente corta en duración con una amplitud infinitamente alta) [22].

Para medir la respuesta al impulso de un altavoz y/o recinto existen diferentes técnicas, de las cuales se destacan: la técnica MLS, y la técnica de barrido logarítmico de señal seno. La técnica MLS corresponde a las siglas en inglés para ‘secuencia de máxima longitud’. Esta técnica se basa en la excitación del espacio con una señal pseudoaleatoria periódica tal como ruido blanco. Para extraer la respuesta al impulso se hace una correlación cruzada circular entre la señal obtenida a la salida y la señal pseudoaleatoria empleada a la entrada. Por su parte, la técnica de

barrido logarítmico de señales seno calcula el comportamiento de cada frecuencia por separado, y extrae la respuesta al impulso a partir de la teoría de convolución. La ventaja de esta última con respecto a la técnica MLS es que puede desarrollarse bajo condiciones no anecóicas [23].

H. La era digital

En un principio, el audio solo disponía de medios analógicos para su procesamiento, sistemas de gran tamaño y costosos a los que no todas las personas podían acceder. La era digital y el surgimiento de los DAW (Digital Audio Workstation) abrió las posibilidades de adaptar esos sistemas analógicos a aplicaciones dentro de un computador; a estos archivos se les llama plugins. Hoy en día existen decenas de DAW's y para cada uno existen formatos específicos de plugins. Entre los sistemas más populares en la actualidad, se encuentran, Pro Tools, Cubase, Logic, y Reaper, y aunque sus funcionalidades son muy similares, no todos los plugins pueden utilizarse en todos los sistemas. Como cualquier otro software, su desarrollo se basa en un lenguaje de programación específico, siendo los más famosos: AU (Audio Unit), AAX (Avid Audio eXtension), RTAS (Real Time Audio Suite) y VST (Virtual Studio Technology).

VST fue lanzado al mercado en el año 1996, desarrollado por la compañía Steinberg (creadora de Cubase, uno de los DAW más famosos del mercado). La mayoría de VST se basan en el lenguaje de programación C++; además de ser admitido en la mayoría de estaciones de trabajo digital, funcionan tanto en Windows como en Mac, lo que no ocurre por ejemplo en protocolos como AU, disponibles solo para Mac. Cabe aclarar que un plugin desarrollado para Windows no funcionará para Mac y viceversa [9].

VII. METODOLOGÍA

A. Desarrollo del ecualizador

1) Ecualizador de fase lineal

Se hizo uso de MatLab para generar un ecualizador de fase lineal; para ello se implementaron filtros FIR en siete bandas del espectro en frecuencia (20Hz a 20.000 Hz) a partir de la función *fir2*, la cual se basa en el método de muestreo en frecuencia para diseño de filtros. Esta función retorna un filtro de orden n , con las características de frecuencia y amplitud descritas en las variables de entrada f y m , estas últimas son vectores de igual magnitud en donde f define las frecuencias a modificar normalizadas (valores expresados entre cero y uno, siendo uno la mitad de la frecuencia de muestreo o frecuencia de Nyquist) y m describe los valores de amplitud con un coeficiente al cual se multiplica la amplitud de la señal; la estructura de escritura en el lenguaje de MatLab es: *fir2(n,f,m)* [24].

Cada una de las siete bandas tenía la posibilidad de modificar su frecuencia de corte, ancho de banda y amplitud. Los valores de frecuencia de corte fueron normalizados (cálculo del cociente entre la frecuencia de corte y la frecuencia de Nyquist), almacenados y ordenados de forma ascendente, asociando cada uno de estos a su valor de amplitud, con el objetivo de ingresar valores válidos a la función *fir2*.

A la hora de implementar el filtro fue necesario determinar el orden para el cual la respuesta del ecualizador fuera coherente a los valores deseados, y en dicho proceso aparecieron dos problemas:

- Obtener anchos de banda muy estrechos necesitaba un orden de filtro demasiado alto.
- El orden también debía ser muy alto cuando se buscaba ecualizar frecuencias bajas.

Debido a esto, y en un proceso de tanteo, se determinó que un orden 100 funcionaba muy bien para frecuencias por encima de 1000 Hz, por lo que se definió este como el orden a utilizar. Para las frecuencias bajas fue necesario hacer un proceso adicional: la señal de entrada fue dividida en dos a partir de un filtro pasa bajas y pasa altas, con frecuencia de corte en 1/20 (2250 Hz) de la frecuencia de muestreo, la señal por encima de dicha frecuencia fue procesada

directamente con el filtro descrito anteriormente; para la señal por debajo de 2050 Hz primero se redujo la frecuencia de muestreo de la señal a 1/10 de la frecuencia de muestreo original (4410 Hz), una vez aplicado el filtro, se retornó a la frecuencia de muestreo original para luego sumar las dos señales y reconstruir el espectro completo. Cabe la pena aclarar que los filtros *pasa bajas* y *pasa altas* aplicados para dividir la señal corresponden a filtros FIR, pues en todas las etapas del ecualizador se debe mantener una respuesta en fase lineal.

La señal de entrada en cada uno de los procesos de calibración tuvo una frecuencia de muestreo de 44100 Hz y una profundidad de 24 bits, considerando que es la frecuencia que soporta la tarjeta de sonido que se empleó durante la calibración. (M-Audio MobilePre)

Para la elaboración de la interfaz gráfica del ecualizador, se empleó la opción *guide* de Matlab, la cual permite crear una interfaz gráfica y programar bajo el mismo lenguaje empleado en este software. Se utilizaron sliders para la asignación de los valores de frecuencia, amplitud y ancho de banda, dados en frecuencia, decibeles y octavas respectivamente. El algoritmo permite cargar un archivo de audio sea estéreo o monofónico e identificar la frecuencia de muestreo y profundidad en bits. La aplicación de los filtros creados para las siete bandas del espectro en frecuencia, se hizo a partir de la convolución entre la señal de entrada y la ecuación en diferencias obtenida de la función *fir2*, proceso realizado con la función *conv*.

2) *Ecualizador de fase no lineal*

El proceso para el diseño de este filtro fue el mismo, pero se empleó la función *Yulewalk*, la cual permite crear un filtro IIR, con los mismos parámetros de entrada de la función *fir2* (n, f, m ; orden, frecuencia y amplitud respectivamente) [25].

En este caso la aplicación del filtro a la señal de entrada se realizó con la función *filter*. Tanto el funcionamiento como la interfaz gráfica de este ecualizador son iguales a la del filtro de fase lineal.

Nota: La división de la señal para el filtro de baja y alta frecuencia se hizo con filtros FIR, de manera que la fase de la señal solo se viera afectada a partir de la ecualización.

B. Preparación del estudio

Existen variables que deben evaluarse antes de realizar el proceso de calibración, tales como los modos y las características en el acondicionamiento acústico de la sala, la ubicación de los altavoces y el punto de escucha; sin embargo, las únicas variables que podían ser modificadas para el desarrollo de esta investigación eran la ubicación de los altavoces y del oyente. Para la ubicación de los monitores con respecto al oyente se tomó como referencia la posición de mezcla y se ubicaron los monitores (referencia Yamaha HS-80) de manera que se obtuviera un triángulo equilátero entre oyente y altavoces y colocando estos a una altura aproximada de 1,2 metros con respecto al suelo. La separación entre altavoces y oyente fue de 0.9 metros (ver figura 3). Lo anterior basado en la norma ITU-R BS.708 [26].

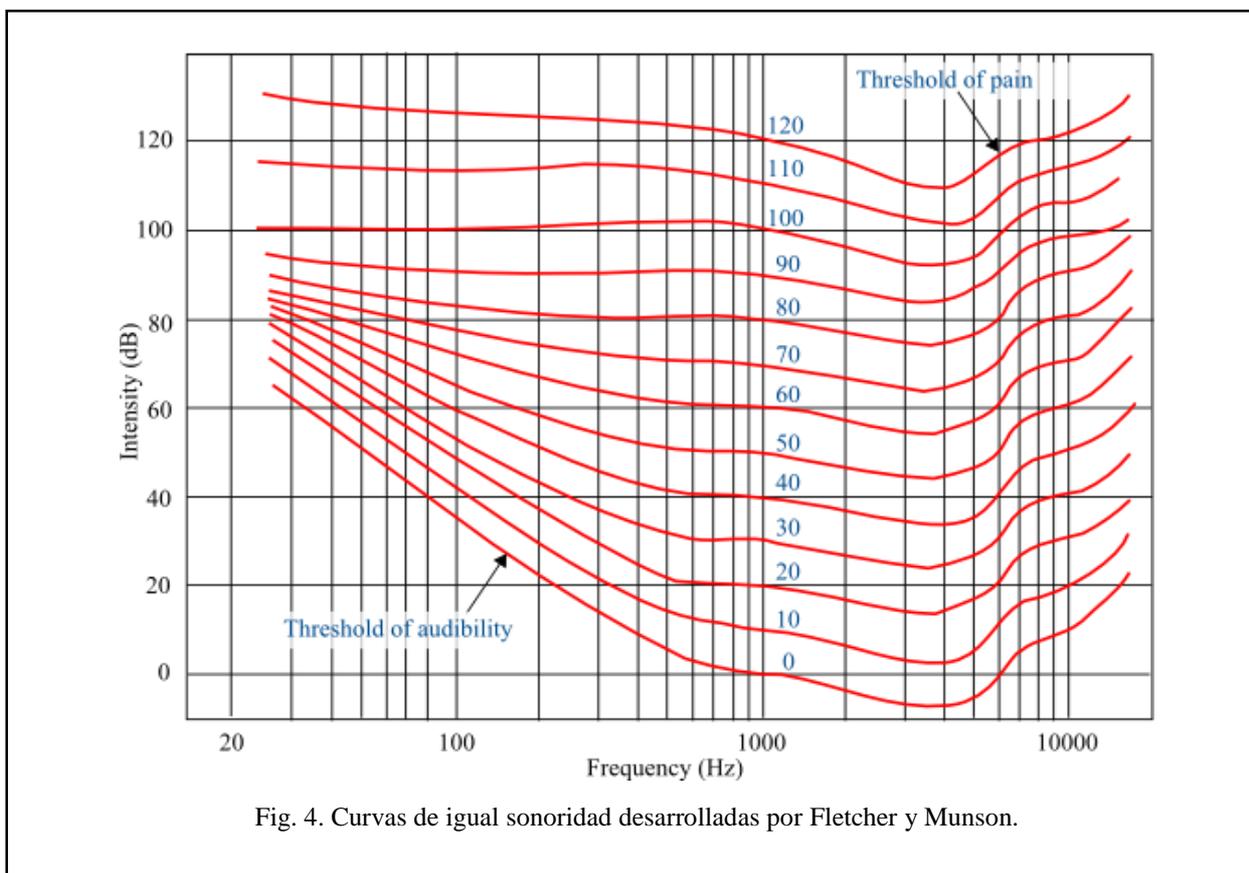


Fig. 3. Ubicación del punto de medición y los altavoces.

Por otra parte, fue necesario determinar el nivel de presión sonora (SPL por sus siglas en inglés) al cual debían hacerse las mediciones y la calibración, y para ello debe conocerse el nivel recomendado para trabajar en los procesos de mezcla y masterización.

En 1930 Fletcher y Munson crearon las curvas de igual sonoridad en las cuales se muestra la respuesta en frecuencia del oído humano con respecto al nivel de presión sonora (ver figura 4),

donde se evidencia claramente que a mayor nivel, la percepción a lo largo del espectro en frecuencia audible es mucho más plana, es por ello que el nivel recomendado para los procesos de postproducción de audio oscila entre los 80 dB y los 86 dB, además que se trata de un nivel en el cual se puede estar expuesto por periodos largos de tiempo sin que la estructura y funcionalidad del oído se vean afectados. Por otra parte el ingeniero de masterización Bob Katz, recomienda trabajar a un nivel de 83dB SPL como nivel equivalente con ambos altavoces encendidos [27]. Basados en esto, este será el nivel elegido para hacer la calibración de respuesta en frecuencia. Dicho ajuste de nivel se hizo con un ruido rosa a un nivel pico de -12 dBFs como señal de entrada (ya que es el nivel de amplitud de la señal de entrada para la calibración) y ubicando el sonómetro Cesva SC310, en ponderación tipo C, y con un tiempo de integración lento, ubicado en el punto de escucha a una altura de 1.2 m sobre el nivel del suelo ya que es la altura promedio del oído de una persona sentada.



C. Proceso de medición

Una vez ajustados los monitores y el nivel al cual reproducirá el sistema se procederá a hacer el ajuste de la respuesta en frecuencia con cada uno de los ecualizadores elegidos, con los cuales

se debe obtener una respuesta en frecuencia que satisfaga las especificaciones de la figura 5, tomada de la norma EBU Tech. 3276.

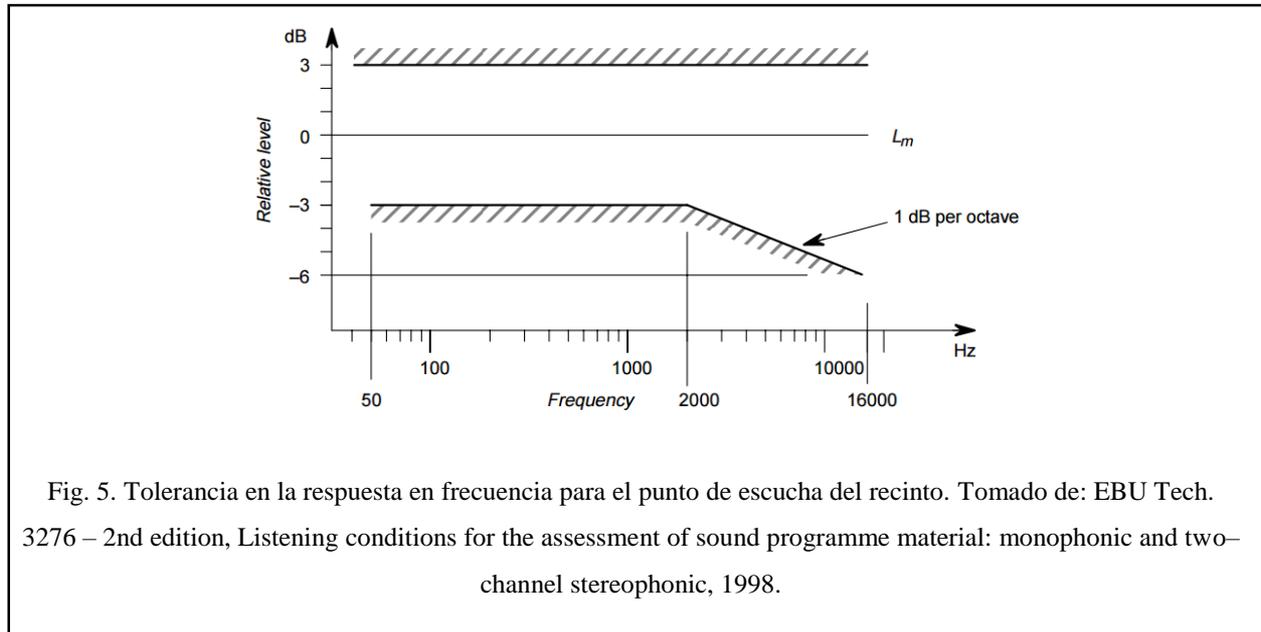


Fig. 5. Tolerancia en la respuesta en frecuencia para el punto de escucha del recinto. Tomado de: EBU Tech. 3276 – 2nd edition, Listening conditions for the assessment of sound programme material: monophonic and two-channel stereophonic, 1998.

La norma también determina que el nivel emitido por cada uno de los altavoces no debe diferir en más de 1 dB.

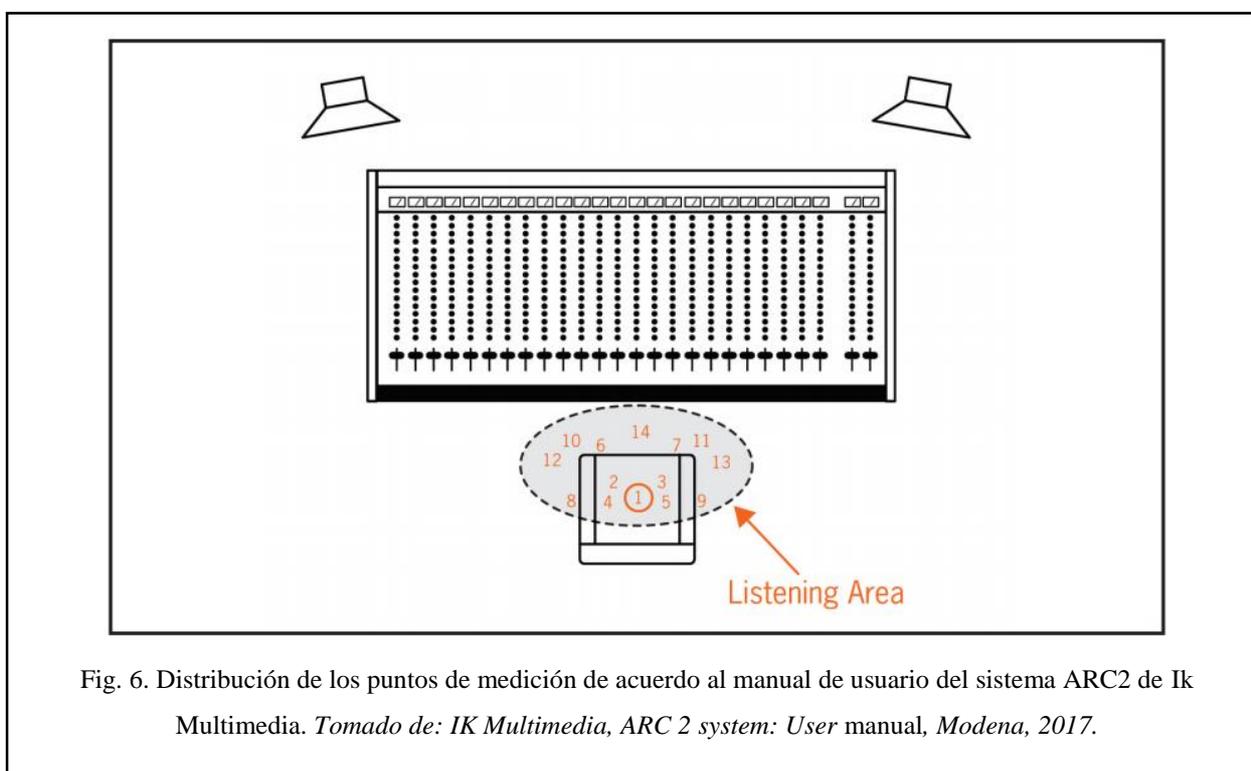
La medición se hizo siguiendo el protocolo utilizado por ARC2 de IK Multimedia, un plugin utilizado para la calibración de estudios el cual realiza de 7 a 16 mediciones distribuidas uniformemente en la zona de escucha (ver figura 6) y de las cuales se obtiene un promedio con el cual se aplica la ecualización correctiva que lleva a obtener una respuesta en frecuencia mucho más plana [28], para cada una de las calibraciones desarrolladas en esta investigación se emplearon siete mediciones con cada altavoz.

El primer paso fue conocer la respuesta de la sala antes de aplicar cualquier proceso correctivo, para ello se empleó el software Room EQ wizard (REW), un software libre que permite obtener graficas de SPL contra frecuencia y promediar un numero N de mediciones para luego obtener las características de un ecualizador que logren mejorar la respuesta en frecuencia medida. Se realizaron siete mediciones para el altavoz izquierdo, siete para el derecho y siete con ambos altavoces simultáneamente ubicando el micrófono de medición en las posiciones 1 a la 7 descritas en la figura 6. La señal de entrada estaba compuesta por cuatro barridos logarítmicos de ondas seno (entre 20 Hz y 20KHz) con una duración total de 28 segundos. La medición se

realizó con un micrófono de medición dbx RTA-M conectado a una tarjeta de sonido Mibile-Pre de M-Audio.

El software (REW), permite obtener las características que debe tener un ecualizador para corregir la respuesta en frecuencia medida y promediada anteriormente, dichos parámetros fueron aplicados a los ecualizadores desarrollados, y al ecualizador dbx 2231. Dado que el ecualizador analógico no permite ajustar exactamente los parámetros entregados por REW, se hizo una medición de doble canal con el software Smart live 7, comparando la señal de entrada y salida de manera que se pudiera apreciar los cambios que se estaban aplicando a la señal y tratar de obtener una respuesta cercana a la de los ecualizadores digitales.

Una vez configurado cada uno de los ecualizadores, se aplicó el mismo procedimiento de medición con cada uno de ellos.



La respuesta en fase de cada uno de los ecualizadores fue medida con el software Smart live 7. Para ello se hizo una medición de dos canales teniendo como referencia la señal antes del proceso y comparándola con la señal de salida de cada uno de los sistemas.

VIII. RESULTADOS

En la figura 7 se observa la interfaz gráfica desarrollada en MatLab. El código permite procesar cada canal de una señal estéreo de manera independiente o enviar los datos de una señal monofónica a dos canales (izquierdo y derecho); es posible elegir el número de bandas a emplear en el ecualizador teniendo un máximo de siete, y se observan las gráficas de la señal en el dominio del tiempo, respuesta en frecuencia del ecualizador de los canales uno y dos, y retraso de fase para el canal uno, el cual es exactamente igual al canal dos.

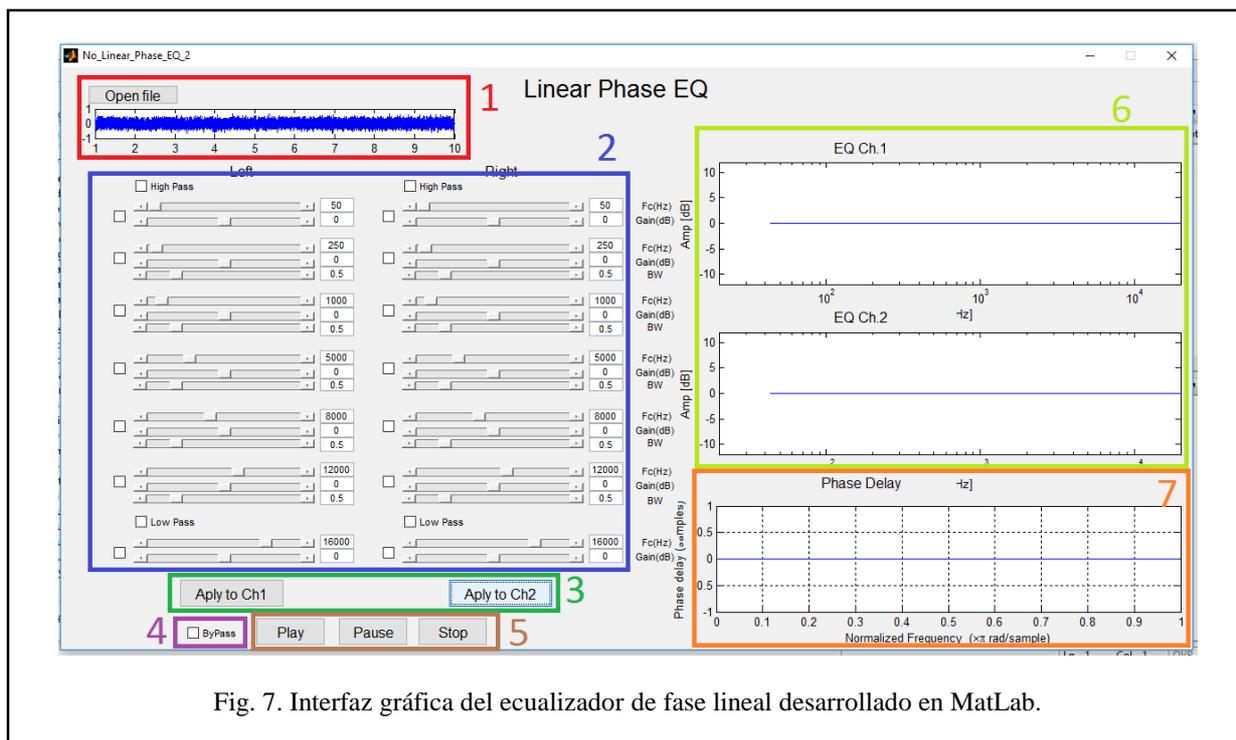


Fig. 7. Interfaz gráfica del ecualizador de fase lineal desarrollado en MatLab.

En la figura 7 se muestran las siguientes partes:

1. Botón para cargar el archivo de audio en formato wav, mono o estéreo y gráfica de la señal con respecto al tiempo.
2. Controles de frecuencia, ganancia y ancho de banda para 7 bandas del espectro en frecuencia. Cada banda tiene la opción de activarse o desactivarse.
3. Botones para aplicar el ecualizador a los canales izquierdo y derecho de acuerdo con lo descrito en cada una de las bandas activadas.
4. Control Bypass, permite activar o desactivar el ecualizador para la reproducción.
5. Controles de reproducción.
6. Gráficas de respuesta en frecuencia del ecualizador (frecuencia contra ganancia).

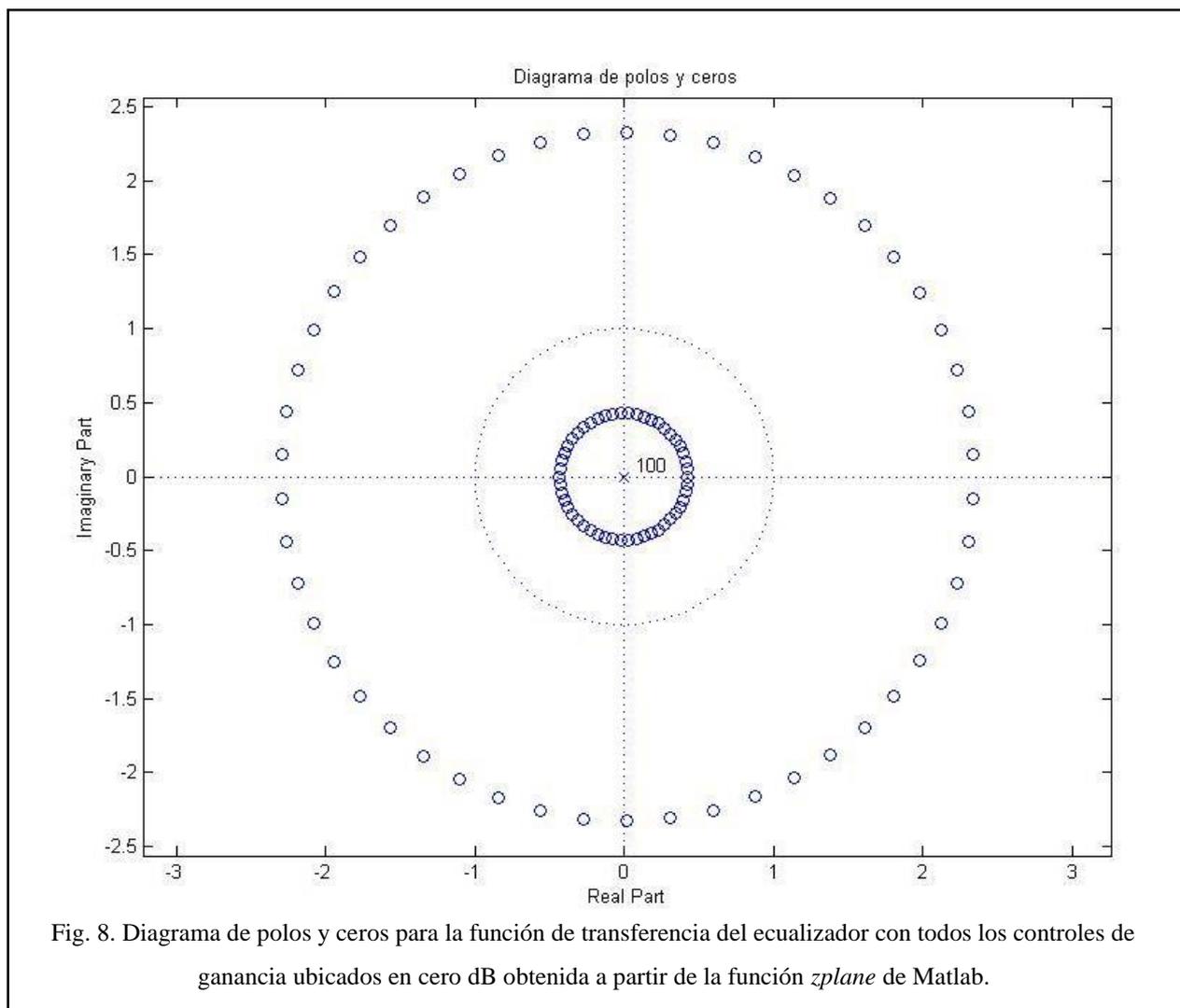
7. Gráfica de retraso de fase del ecualizador.

Nota: La interfaz para el ecualizador de fase no lineal es la misma a la descrita en la figura 7.

En el anexo 1 se expone el código desarrollado para esta aplicación.

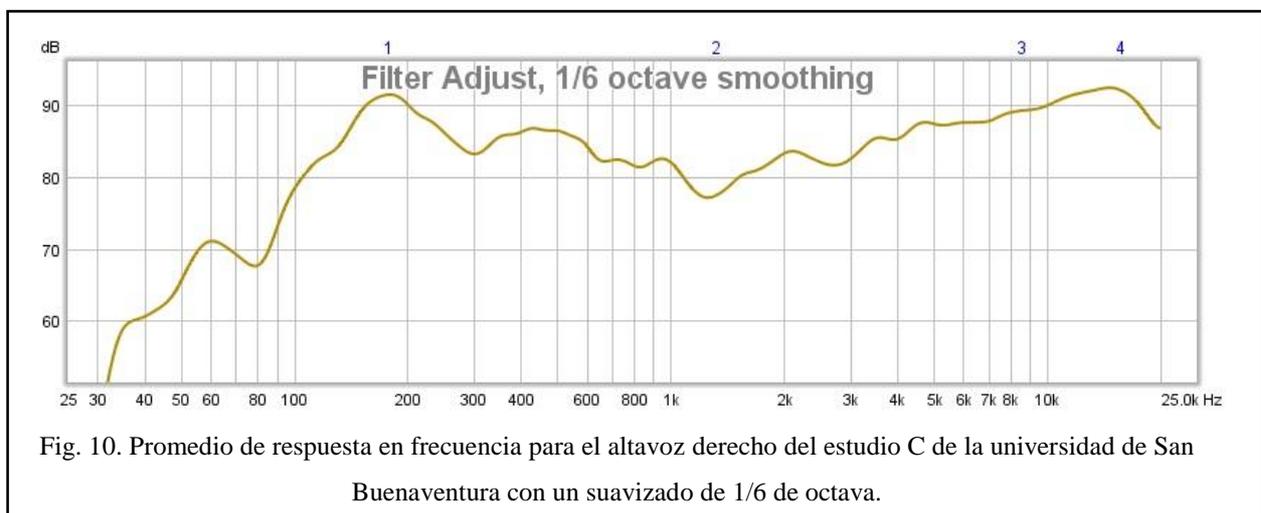
A. Ecualizadores

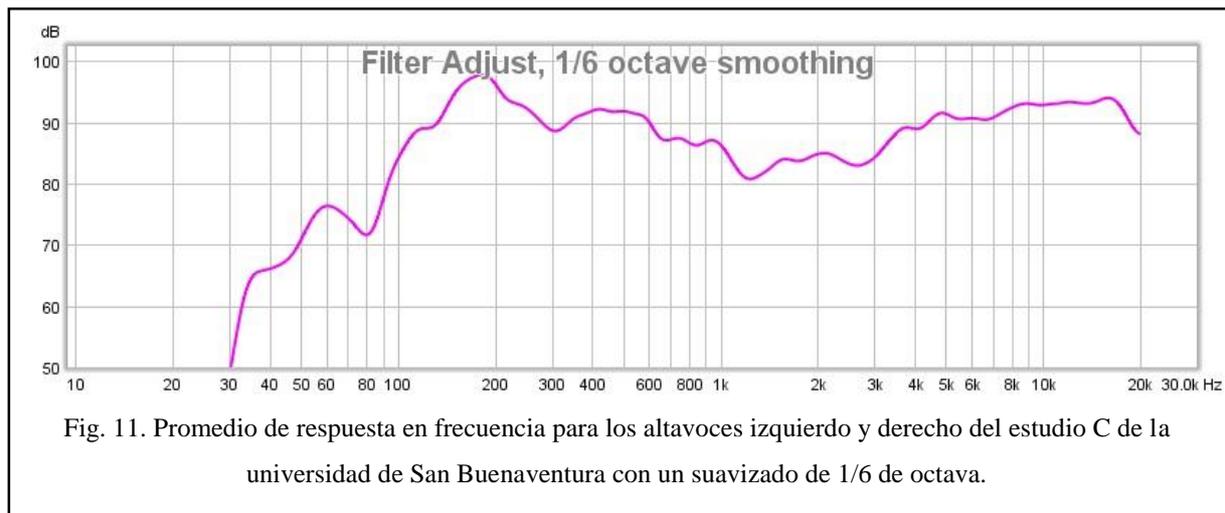
La figura 8 muestra el diagrama de polos y ceros para la función de transferencia del filtro de fase lineal; se marcan los ceros con “o” y los polos con “x”, la gráfica corresponde a el filtro con todos los controles de ganancia en cero, sin embargo, al cambiar las características del filtro, el diagrama de polos y ceros continúa reflejando una distribución de ceros simétrica con un único polo en el centro de la gráfica.



B. Mediciones

Al realizar las siete mediciones para conocer la respuesta en frecuencia de cada uno de los altavoces y de ambos simultáneamente con el software Room EQ Wizard, se obtuvieron las siguientes gráficas, las cuales expresan los valores de amplitud en decibeles (eje vertical) para cada una de las frecuencias entre 20 Hz y 20kHz (eje horizontal).



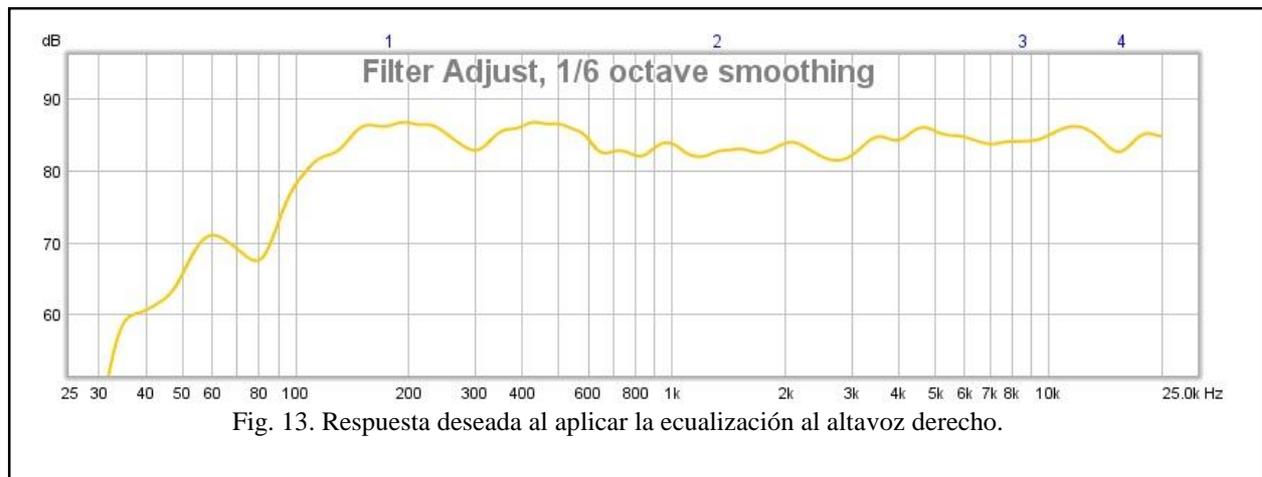
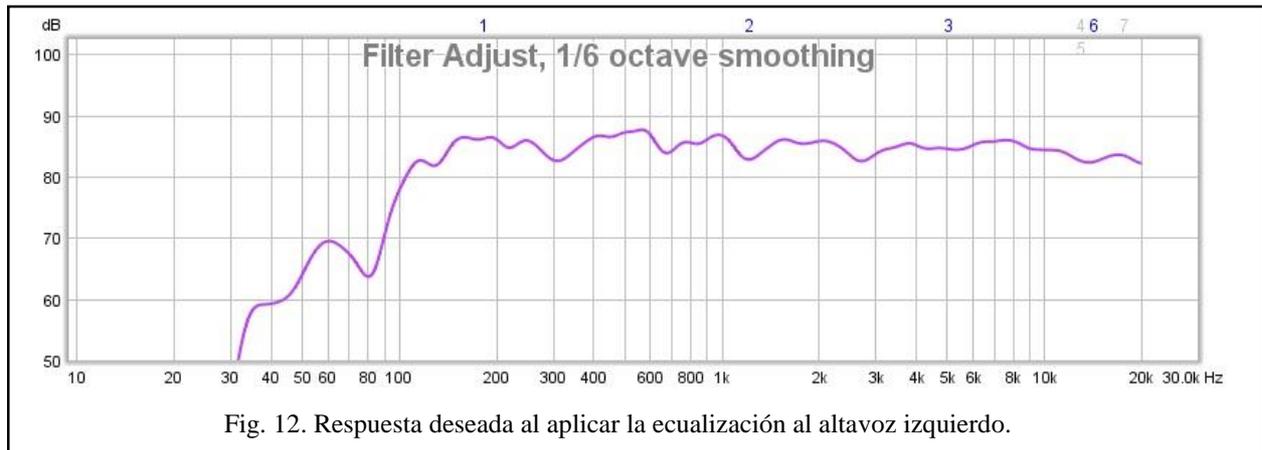


Al procesar los datos del software de medición se obtuvieron los parámetros de configuración para un ecualizador que logran corregir (hacer más plana) la respuesta en frecuencia en los puntos de medición; dichos parámetros se exponen en la tabla 1.

TABLA I. PARÁMETROS PARA CONFIGURACIÓN DE UN ECUALIZADOR QUE CORRIGEN LA RESPUESTA EN FRECUENCIA MEDIDA.

	Frecuencia [Hz]	Ganancia [dB]	Factor Q
Altavoz izquierdo	181.5	-7.1	3.71
	1200	6.6	1
	5000	-3.4	2.81
	14000	-9	0.93
Altavoz derecho	175.5	-6	3.33
	1300	6	3.12
	8458	-4.4	1
	15456	-10.5	1.69

El software REW permite predecir la respuesta en frecuencia que se podría obtener al aplicar los filtros descritos en la tabla 1, dicha respuesta se muestra en las figura 12 y 13.



Los parámetros descritos en la tabla 1 fueron aplicados a cada uno de los ecualizadores desarrollados.

Se hizo una medición de doble canal con el software Smaart live 7 para cada uno de los ecualizadores, comparando la señal antes y después de ser procesada. En cada una de las gráficas se observa la diferencia en grados de la señal de salida con respecto a la de entrada (eje vertical) a lo largo del espectro en frecuencia (eje horizontal). La figura 14 muestra el retraso de fase del canal izquierdo del ecualizador de fase lineal (para el canal derecho se obtuvo la misma respuesta). Las figuras 15 y 16 corresponden al retraso de fase del ecualizador de fase no lineal desarrollado, y las figuras 17 y 18 corresponden al ecualizador analógico.



Fig. 14. Retraso de fase del ecualizador de fase lineal desarrollado para los canales izquierdo y derecho.

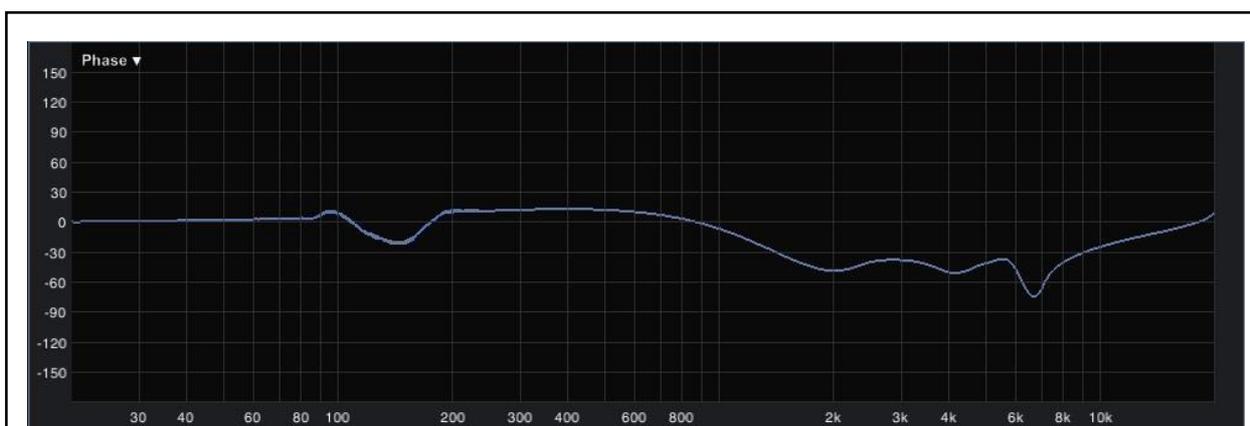


Fig. 15. Retraso de fase del ecualizador de fase no lineal desarrollado para el canal izquierdo.

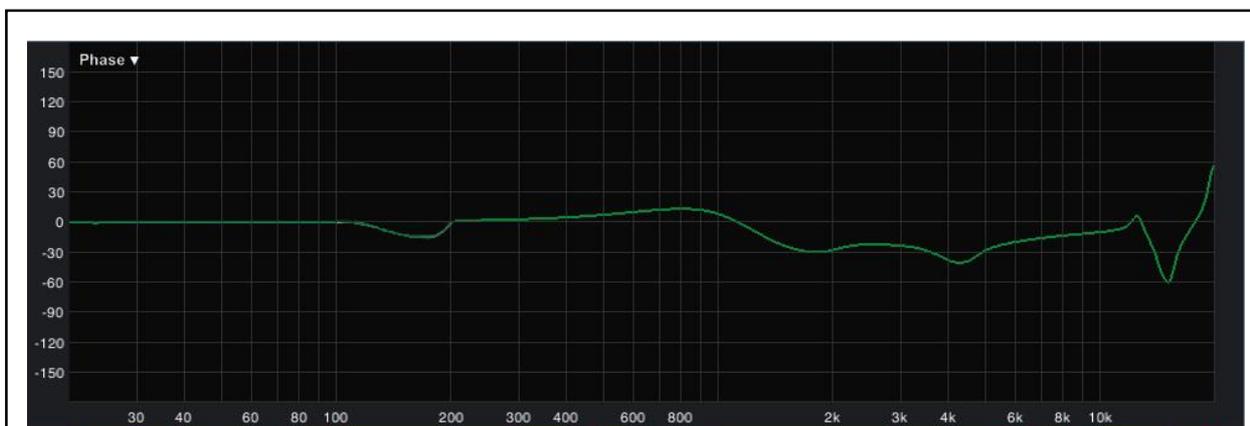


Fig. 16. Retraso de fase del ecualizador de fase no lineal desarrollado para el canal derecho.

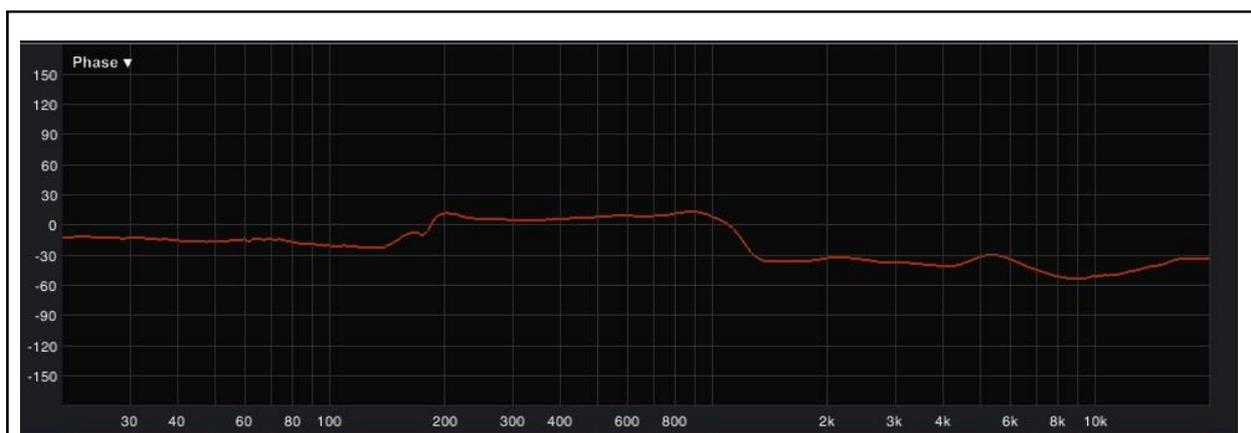


Fig. 17. Retraso de fase del ecualizador analógico para el canal izquierdo.

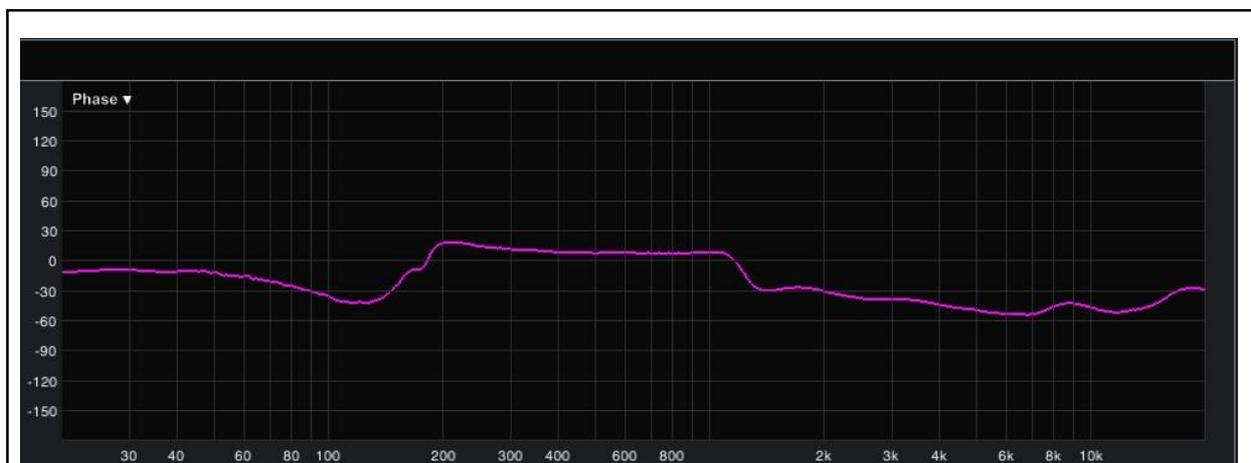


Fig. 18. Retraso de fase del ecualizador analógico para el canal derecho.

El propósito de crear un ecualizador de fase no lineal en lugar de compararlo con un plugin existente en el mercado fue garantizar que la curva de ecualización fuera lo más similar posible entre ambos ecualizadores digitales, las figuras 19 y 20 muestran las magnitudes superpuestas de ambos ecualizadores para los canales izquierdo y derecho respectivamente. Por su parte las figuras 21 y 22 exponen la magnitud de la respuesta en frecuencia del ecualizador analógico obtenida a partir de una medición de doble canal en el software Smart live 7, y la cual trató de acercarse lo más posible a la respuesta de los ecualizadores digitales.

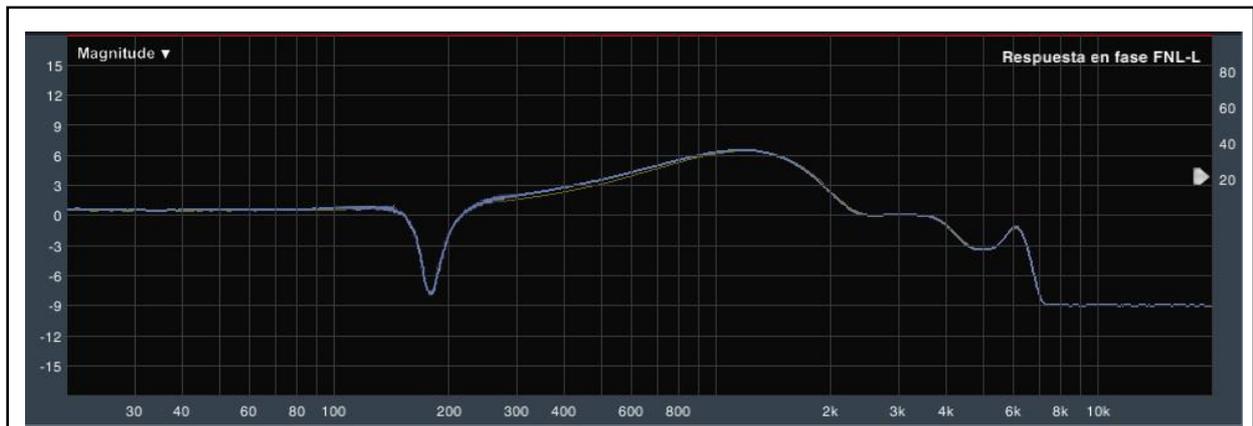


Fig. 19. Respuesta en frecuencia de los ecualizadores digitales para el canal izquierdo (Fase lineal en azul y no lineal en verde)



Fig. 20. Respuesta en frecuencia de los ecualizadores digitales para el canal derecho (Fase lineal en rosado y no lineal en verde)

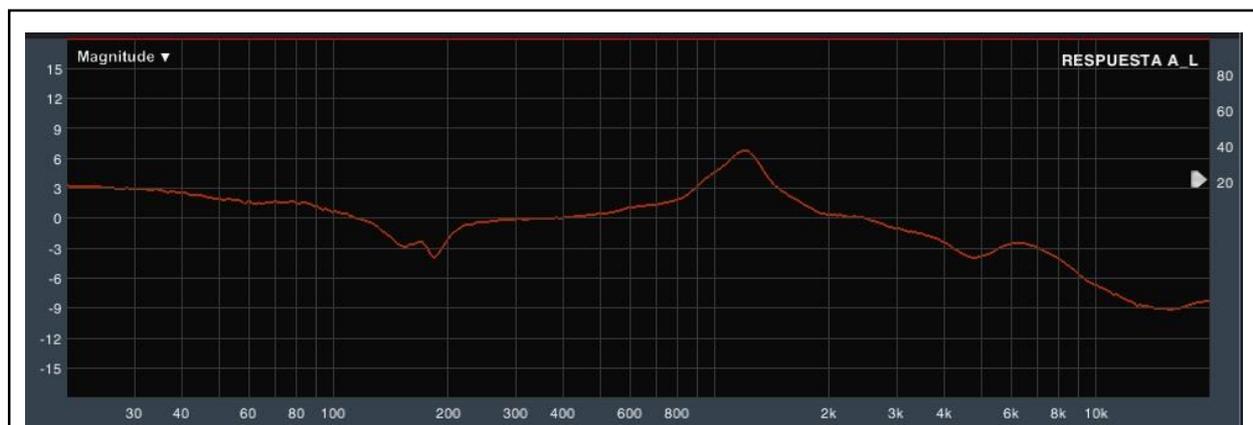


Fig. 21. Respuesta en frecuencia del ecualizador analógico dbx 2231 aplicado al canal izquierdo.

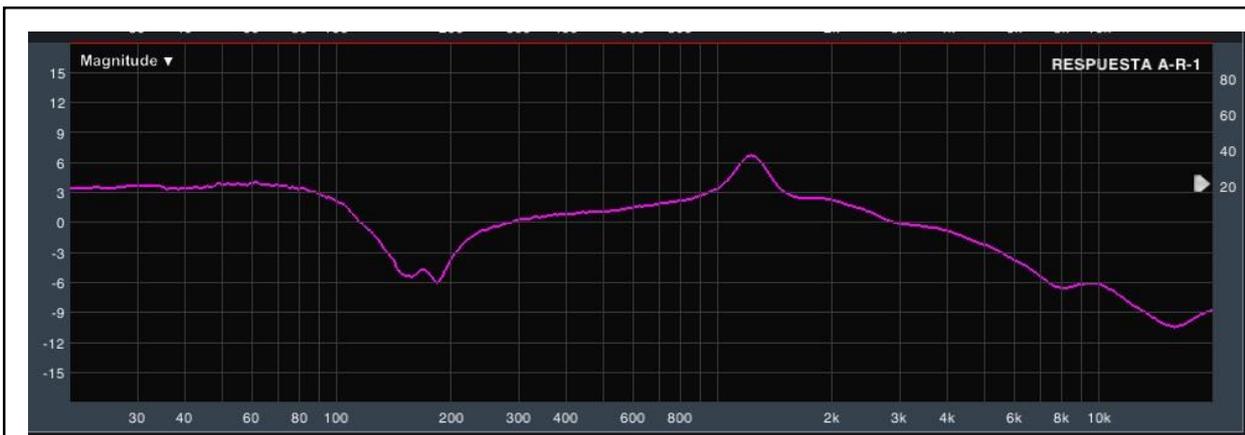


Fig. 22. Respuesta en frecuencia del ecualizador analógico dbx 2231 aplicado al canal derecho.

Al aplicar cada uno de los ecualizadores con los parámetros descritos en la tabla 1, se obtuvieron los siguientes resultados.

Para el ecualizador digital de fase lineal:

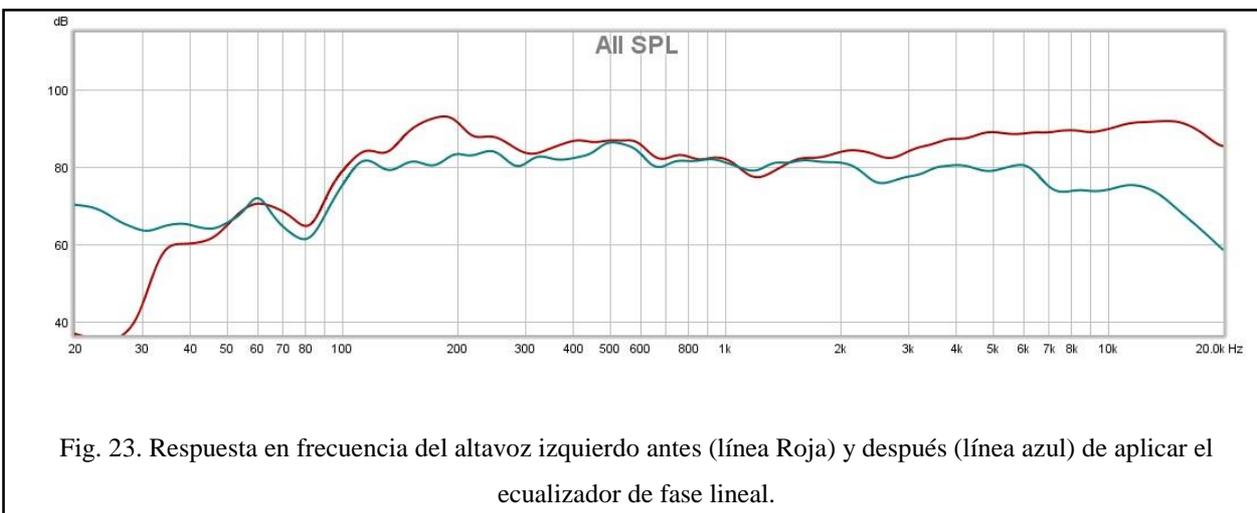


Fig. 23. Respuesta en frecuencia del altavoz izquierdo antes (línea Roja) y después (línea azul) de aplicar el ecualizador de fase lineal.

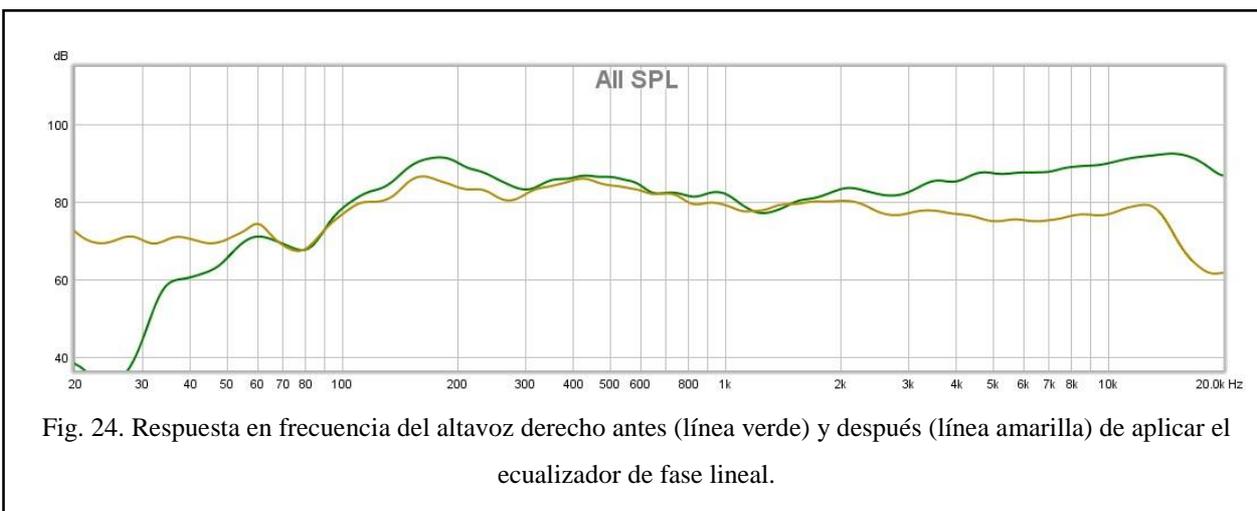


Fig. 24. Respuesta en frecuencia del altavoz derecho antes (línea verde) y después (línea amarilla) de aplicar el ecualizador de fase lineal.

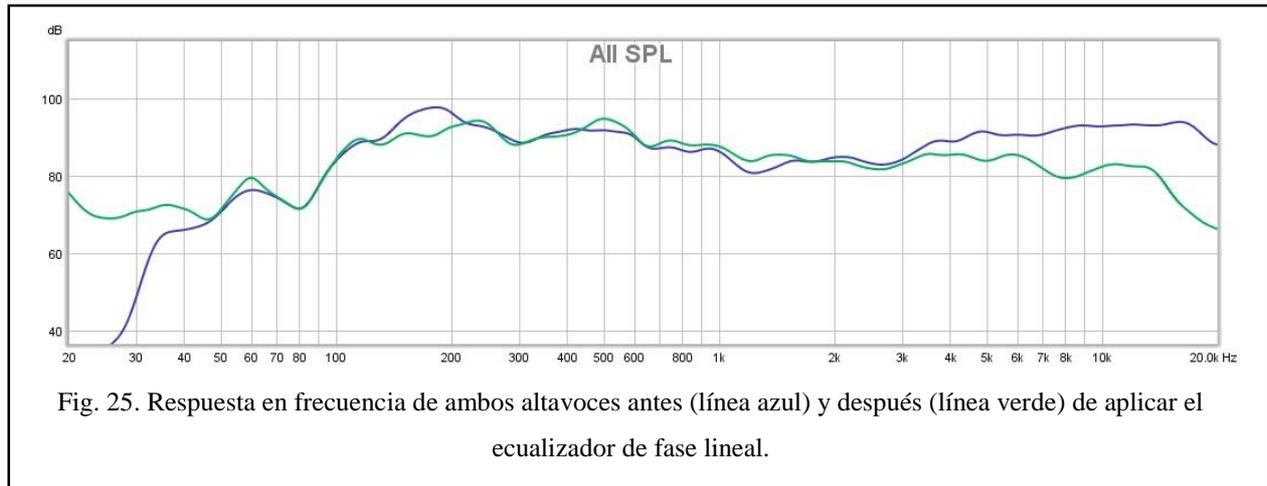


Fig. 25. Respuesta en frecuencia de ambos altavoces antes (línea azul) y después (línea verde) de aplicar el ecualizador de fase lineal.

Para el ecualizador digital de fase no lineal:

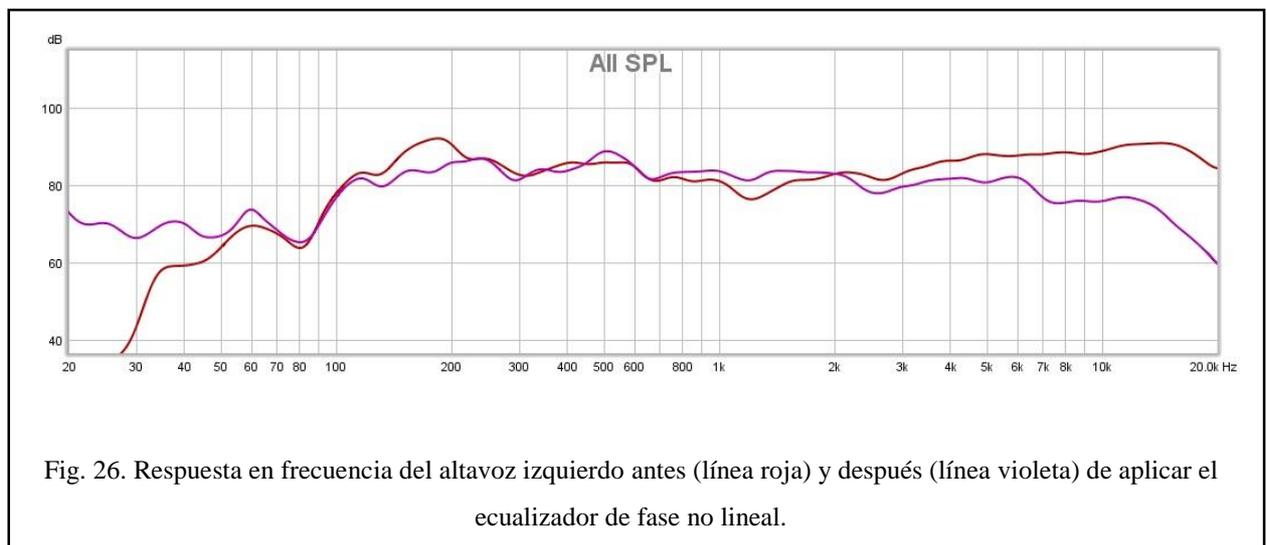


Fig. 26. Respuesta en frecuencia del altavoz izquierdo antes (línea roja) y después (línea violeta) de aplicar el ecualizador de fase no lineal.

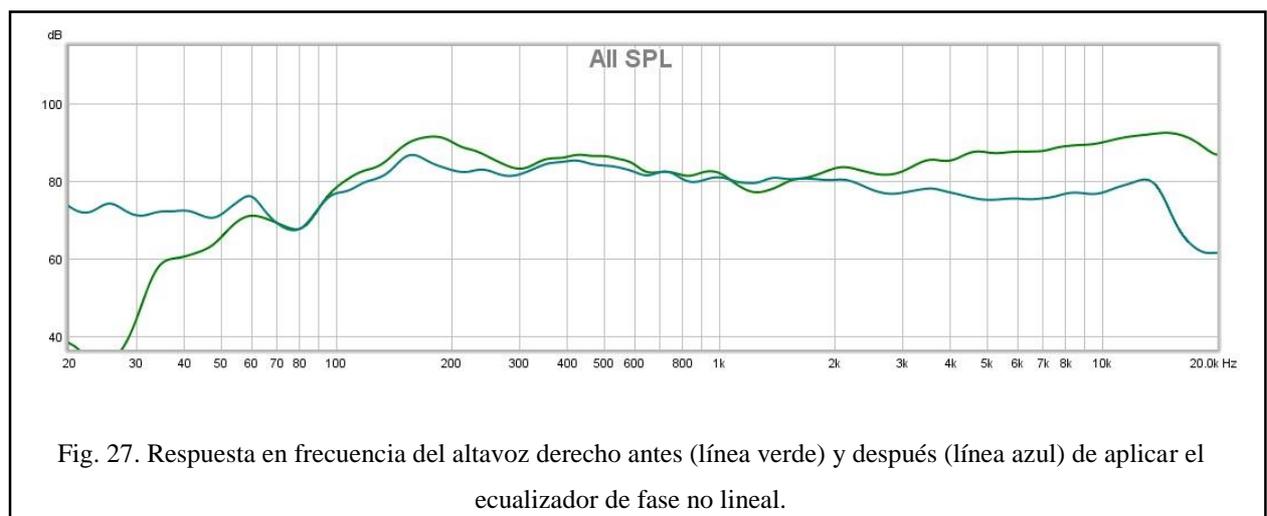


Fig. 27. Respuesta en frecuencia del altavoz derecho antes (línea verde) y después (línea azul) de aplicar el ecualizador de fase no lineal.

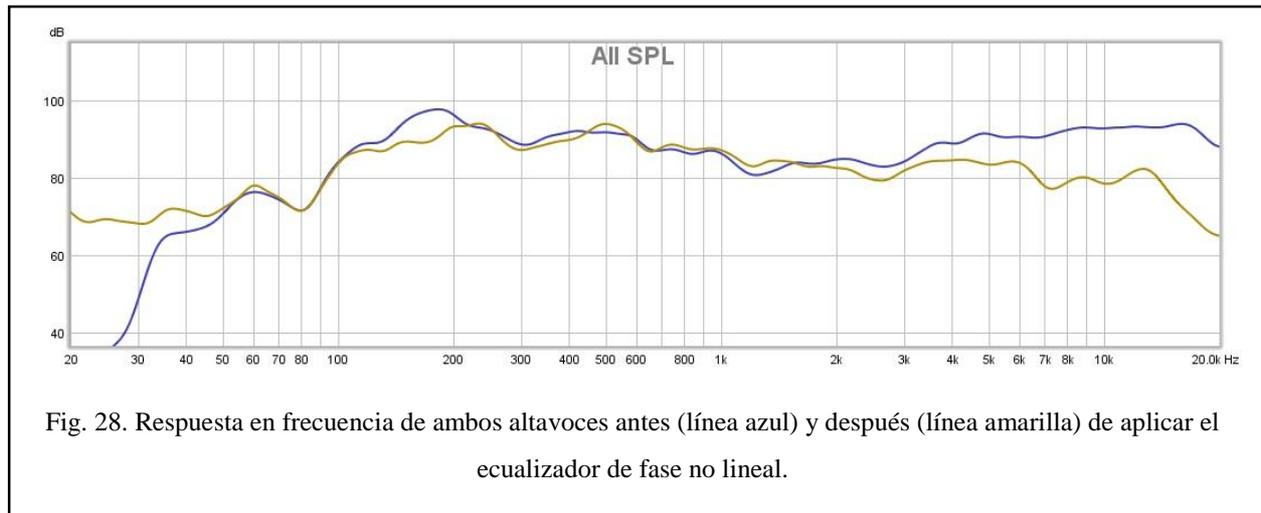


Fig. 28. Respuesta en frecuencia de ambos altavoces antes (línea azul) y después (línea amarilla) de aplicar el ecualizador de fase no lineal.

Para el ecualizador analógico:

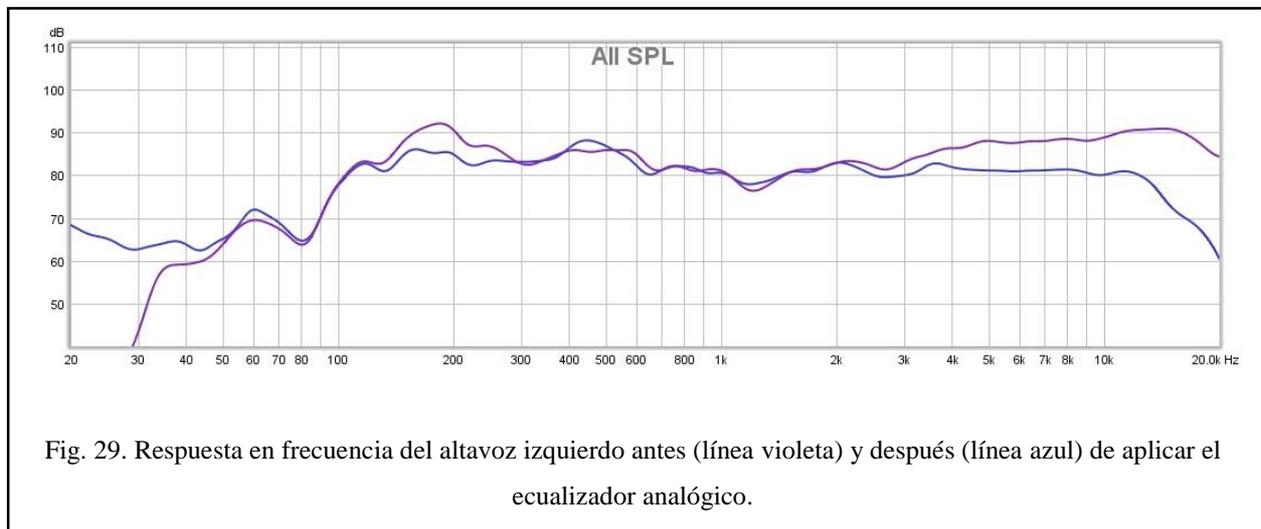


Fig. 29. Respuesta en frecuencia del altavoz izquierdo antes (línea violeta) y después (línea azul) de aplicar el ecualizador analógico.

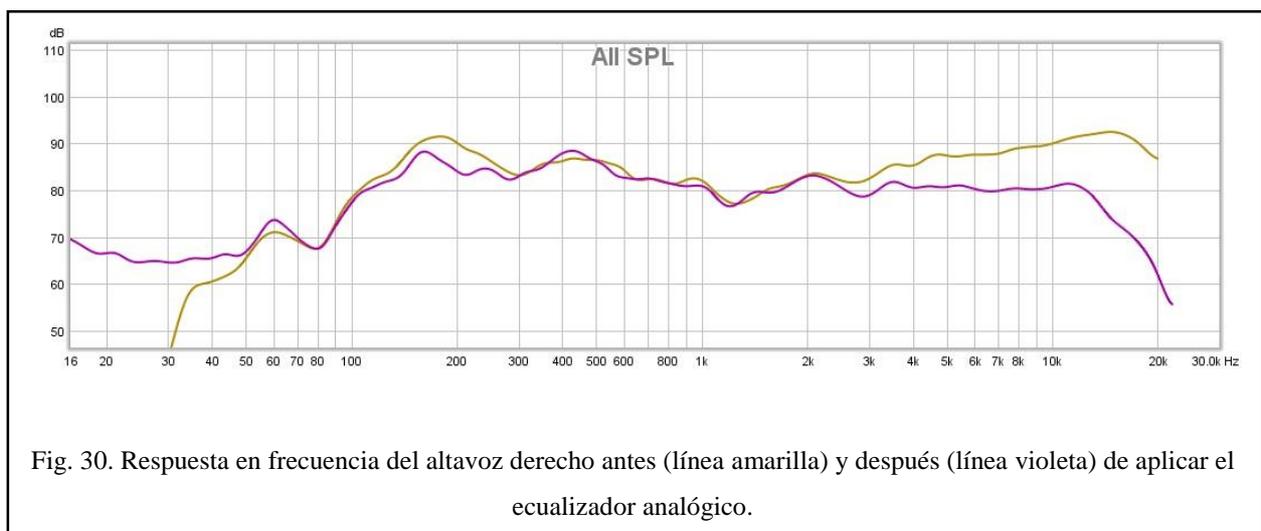
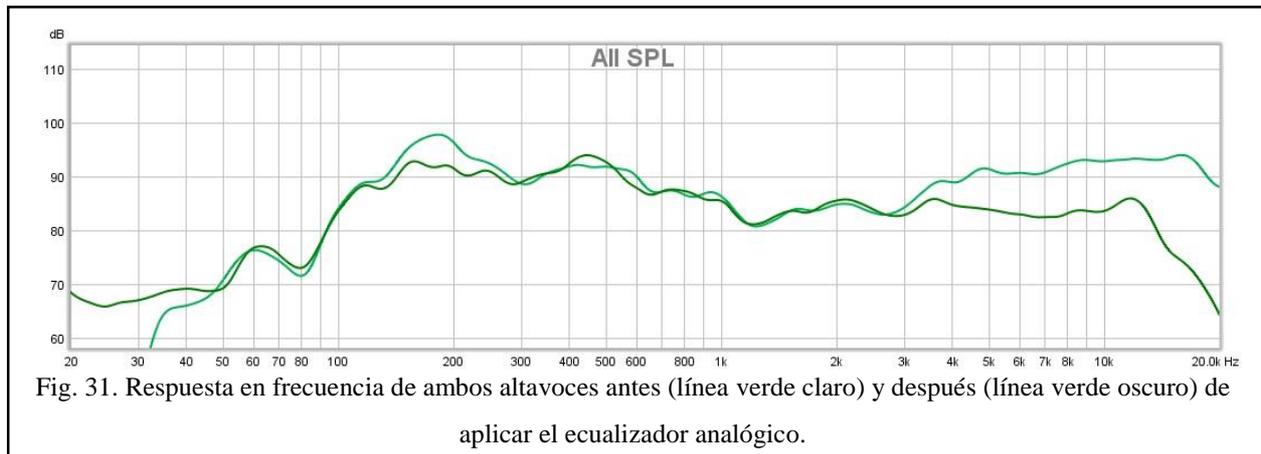
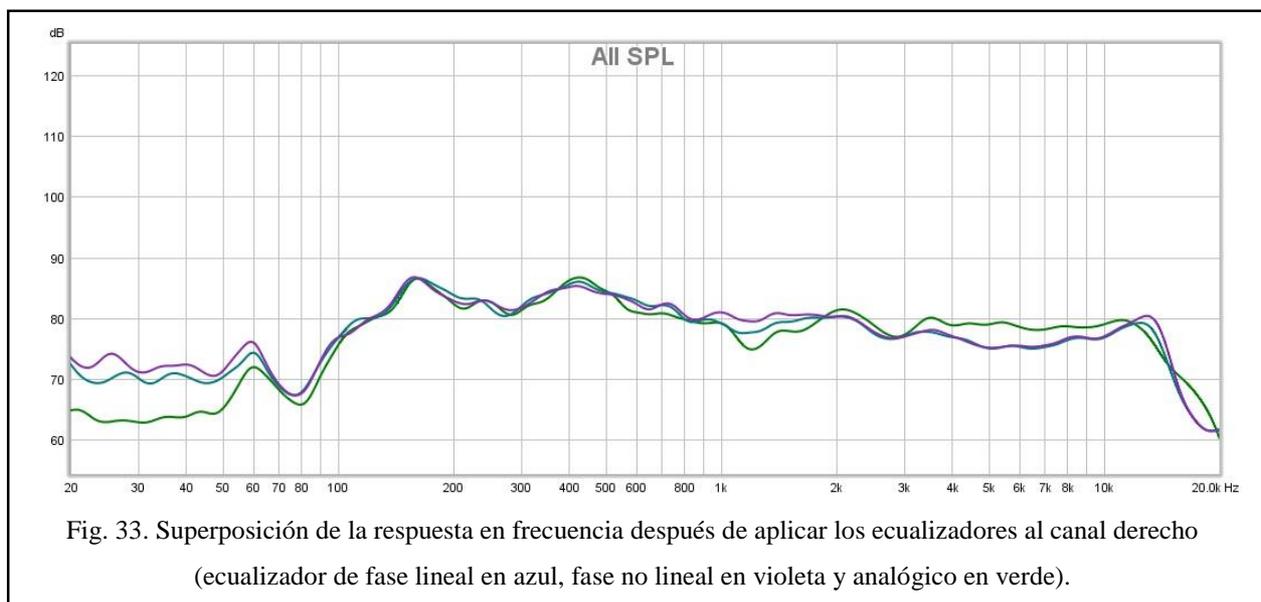
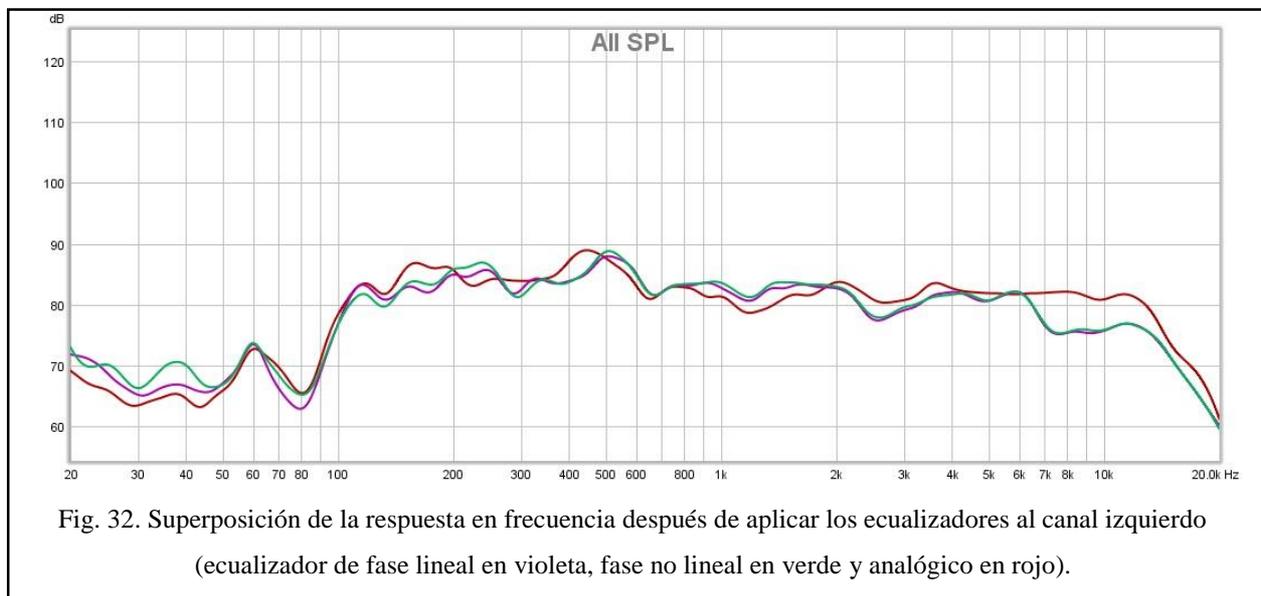
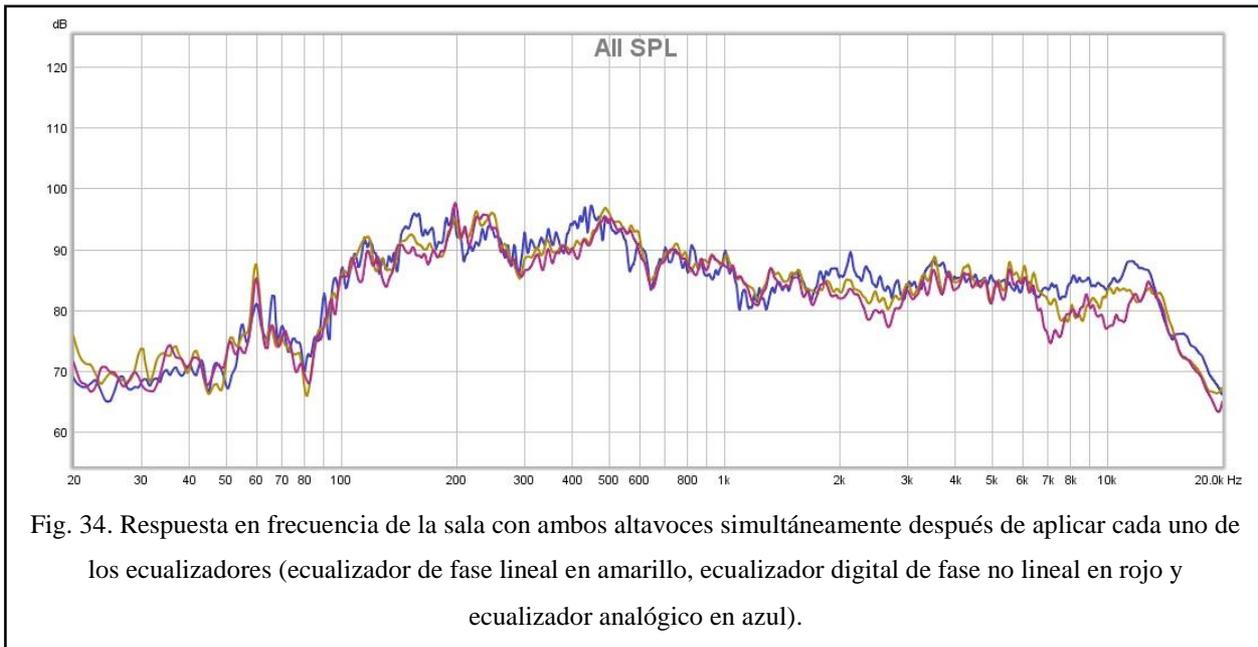


Fig. 30. Respuesta en frecuencia del altavoz derecho antes (línea amarilla) y después (línea violeta) de aplicar el ecualizador analógico.



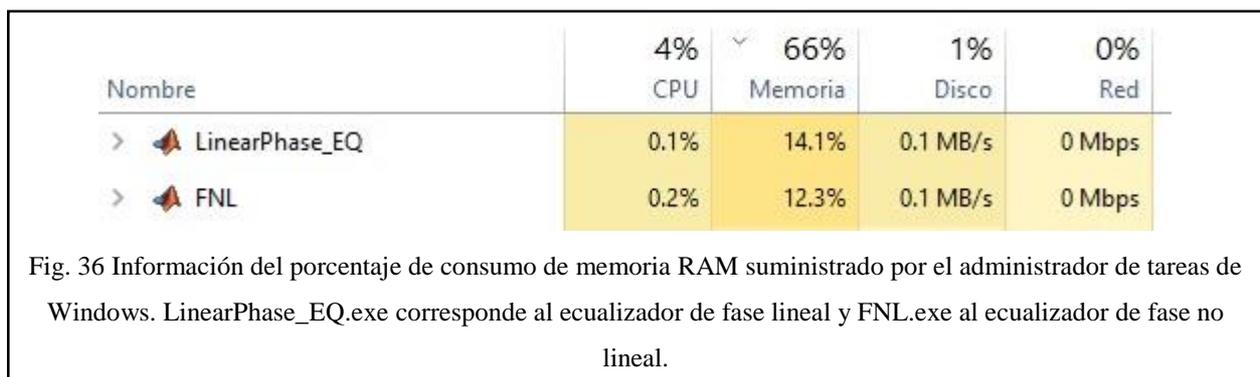
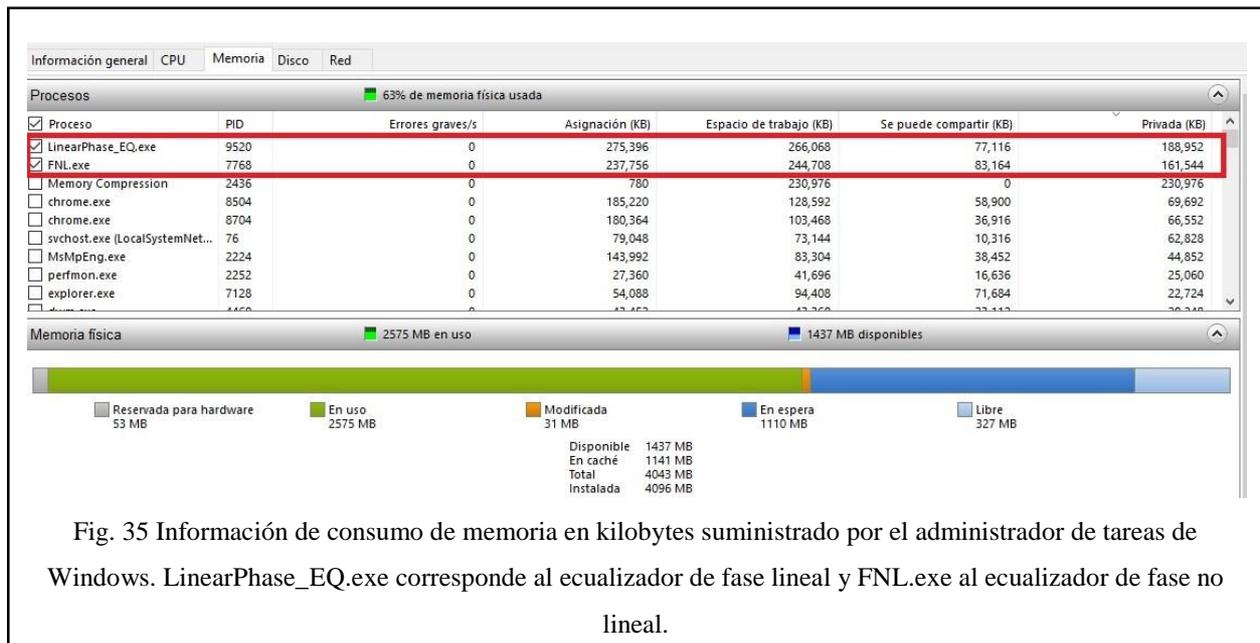
A continuación se exponen las gráficas superpuestas de las tres calibraciones:





C. consumo de memoria RAM

Todo sistema digital tiene asociado un porcentaje de consumo de memoria RAM que depende del número de procesos que deba realizar para cumplir su función. Las figuras 8 y 9 muestran el consumo de los ecualizadores digitales desarrollados funcionando en un computador con sistema operativo Windows 10, procesador Intel core i3-2310M con una velocidad de procesamiento de 2.10 GHz y memoria RAM de 4 Gb. Ambos ecualizadores tienen la misma señal de entrada (barrido logarítmico de onda seno de un segundo de duración) y las mismas modificaciones en los canales izquierdo y derecho.



En la figura 35 se observa que el consumo del ecualizador de fase lineal es de 266.068 KB, mientras que el ecualizador de fase no lineal tiene un consumo de 244.708 KB.

IX. DISCUSIÓN

Con respecto al ecualizador de fase lineal desarrollado, el diagrama de polos y ceros (figura 8), muestra una distribución simétrica de los ceros, es decir, los coeficientes del numerador de la función de transferencia se comportan como pares complejos conjugados y la simetría se mantiene sin importar la configuración del ecualizador, tal como se muestra en el anexo 2, lo que demuestra que su respuesta al impulso es simétrica y por consiguiente su respuesta en fase es lineal, hecho que se reafirma en la figura 14. Por otra parte, no existen coeficientes en el denominador, lo que lleva a que no existan polos y sea posible asegurar que se trata de un sistema estable, es decir, a medida que la señal de entrada tienda a infinito, la respuesta del sistema tenderá a cero. La respuesta en frecuencia del ecualizador digital de fase no lineal y del analógico (figuras 15, 16, 17 y 18) refleja claramente cambios en la fase a lo largo del espectro en frecuencia, con cambios mucho más marcados en las frecuencias de corte de las bandas modificadas.

En las gráficas de respuesta en frecuencia de la sala antes de aplicar cualquier proceso correctivo (figuras 9, 10 y 11) se observa una respuesta efectiva de los altavoces a partir de 100 Hz, por consiguiente, todos los filtros aplicados a la salida del sistema estuvieron por encima de dicha frecuencia y el rango de 20 a 100 Hz no será tenido en cuenta en este análisis.

Con cada uno de los ecualizadores se observa una corrección efectiva y acorde con los límites establecidos por la norma EBU 3276 (ver figura 5) en casi todo el espectro, sin embargo existieron algunos puntos en los que la respuesta en frecuencia estuvo por fuera de los límites:

- Para el ecualizador de fase lineal:
 - Entre 235 y 255 Hz del canal izquierdo la respuesta se encuentra 1 dB por encima del límite superior permitido.
 - Entre 455 y 600 Hz del canal izquierdo la respuesta se encuentra 2.5 dB por encima del límite superior permitido.
 - Entre 2.34 y 3.15 KHz del canal izquierdo la respuesta se encuentra 2 dB por debajo del límite inferior permitido.
 - Entre 6.8 y 10.5 KHz del canal izquierdo la respuesta se encuentra 2 dB por debajo del límite inferior permitido.

- Entre 144 y 200 Hz del canal derecho la respuesta se encuentra 2.5 dB por encima del límite superior permitido.
- Entre 350 y 500 Hz del canal derecho la respuesta se encuentra 2 dB por encima del límite superior permitido.
- Entre 4.5 y 8 KHz del canal derecho la respuesta se encuentra 2 dB por debajo del límite inferior permitido.

- Para el ecualizador digital de fase no lineal.
 - Entre 200 y 255 Hz del canal izquierdo la respuesta se encuentra 1 dB por encima del límite superior permitido.
 - Entre 455 y 600 Hz del canal izquierdo la respuesta se encuentra 3 dB por encima del límite superior permitido.
 - Entre 2.34 y 3.15 KHz del canal izquierdo la respuesta se encuentra 1.8 dB por debajo del límite inferior permitido.
 - Entre 6.8 y 10.5 KHz del canal izquierdo la respuesta se encuentra 2 dB por debajo del límite inferior permitido.
 - Entre 144 y 185 Hz del canal derecho la respuesta se encuentra 2.5 dB por encima del límite superior permitido.
 - Entre 350 y 500 Hz del canal derecho la respuesta se encuentra 1.3 dB por encima del límite superior permitido.
 - Entre 4.5 y 8 KHz del canal derecho la respuesta se encuentra 2 dB por debajo del límite inferior permitido.

- Para el ecualizador analógico
 - Entre 143 y 213 Hz del canal izquierdo la respuesta se encuentra 1 dB por encima del límite superior permitido.
 - Entre 387 y 543 Hz del canal izquierdo la respuesta se encuentra 3 dB por encima del límite superior permitido.
 - Entre 1.07 y 1.36 KHz del canal izquierdo la respuesta se encuentra 1.7 dB por debajo del límite inferior permitido.
 - Entre 3.44 y 3.89 KHz del canal izquierdo la respuesta se encuentra 0.6 dB por debajo del límite inferior permitido.
 - Entre 144 y 185 Hz del canal derecho la respuesta se encuentra 2.4 dB por encima del límite superior permitido.

- Entre 370 y 500 Hz del canal derecho la respuesta se encuentra 2.2 dB por encima del límite superior permitido.
- Entre 1.20 y 1.73 KHz del canal derecho la respuesta se encuentra 4 dB por debajo del límite inferior permitido.
- Entre 2.57 y 3.20 KHz del canal derecho la respuesta se encuentra 4 dB por debajo del límite inferior permitido.

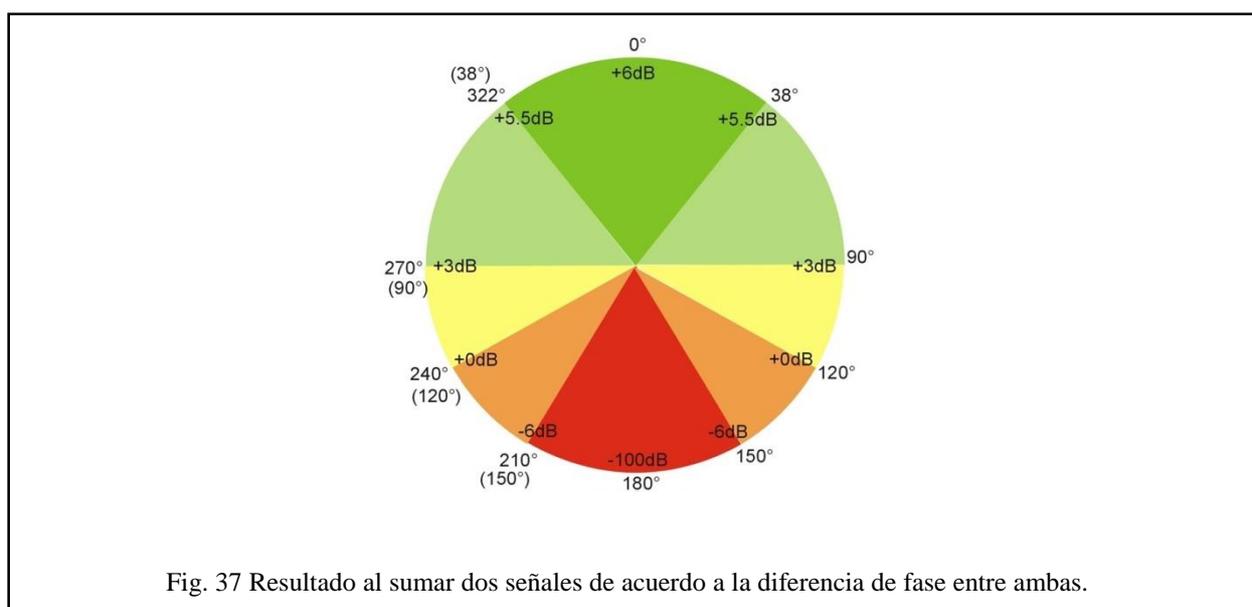
Los datos anteriores muestran una respuesta muy aproximada a lo que establece la norma EBU 3276, con niveles en algunos intervalos del espectro que no se alejan más de 4 dB de los límites inferior y superior recomendados y con una respuesta mucho más plana a la obtenida antes del proceso de ecualización. Los intervalos en donde la respuesta en frecuencia está por fuera de los límites no difieren entre los ecualizadores digitales y los niveles en estos intervalos no se alejan más de 0.7 dB entre un ecualizador y otro, lo que demuestra que el comportamiento entre cada uno de estos es muy similar analizando cada uno de los canales por separado. Por su parte, en el ecualizador analógico los intervalos de frecuencia en que la respuesta sobrepasa los límites no se alejan más de un tercio de octava con respecto a las frecuencias que superan los límites en los ecualizadores digitales. Los picos máximos y mínimos son prácticamente iguales en los tres ecualizadores.

Las respuestas en frecuencia de los altavoces en la sala, después de aplicar los dos ecualizadores digitales desarrollados son muy similares y no difieren en más de 1.6 dB a lo largo de todo el espectro, sin embargo hay una considerable diferencia de estos con respecto al ecualizador dbx 2231, dado que al tratarse de un ecualizador gráfico no es posible modificar una frecuencia específica ni ajustar exactamente el ancho de banda para obtener la misma respuesta de un ecualizador paramétrico. La respuesta obtenida para frecuencias bajas y medias bajas (100 a 500 Hz) es muy similar en los tres ecualizadores, para frecuencias medias (500 a 3000 Hz) se obtuvo una respuesta más plana en los ecualizadores digitales, mientras que en frecuencias altas (por encima de 4 KHz), hubo un mejor resultado con el ecualizador analógico; sin embargo, el comportamiento entre los ecualizadores digitales y el analógico es diferente, por lo que no es correcto afirmar hasta este punto, que existe un ecualizador que funcione mejor para un proceso de calibración.

Las gráficas 25 y 26 de respuesta en frecuencia medidas con el software Room EQ Wizard están dadas con un suavizado de 1/6 de octava, hecho que no permite ver en detalle lo que puede

ocurrir en varios puntos del espectro que pueden ser críticos a la hora de hacer una comparación, como las frecuencias de corte de los filtros aplicados, es por ello que la gráfica 27 expone la respuesta estéreo del recinto después de aplicar cada uno de los ecualizadores con una resolución de 1/48 de octava.

La respuesta estéreo permite analizar el impacto de la fase al realizar la calibración, ya que la interacción de dos señales hace que se presenten interferencias constructivas o destructivas, dependiendo del grado de desfase entre estas, tal como se muestra en la figura 37.



Las respuestas del recinto para los canales izquierdo y derecho individualmente después de aplicar los ecualizadores digitales son muy similares, sin embargo al comparar la respuesta estéreo de la sala, se observan dos intervalos en los cuales dicha respuesta difiere considerablemente: En el intervalo 6.7 a 7.4 KHz, la respuesta en la señal proveniente del ecualizador de fase no lineal se encuentra 6 dB por debajo de la señal procesada con el ecualizador de fase lineal, la misma diferencia que existente en el intervalo 9.8 a 11.2 KHz, una diferencia bastante amplia considerando que para las respuestas individuales en estos intervalos la señal no se aleja más de 0.3 dB entre ambas mediciones. Comparando la respuesta en fase del ecualizador de fase no lineal en ambos canales, se observa un comportamiento muy similar a lo largo de todo el espectro, no obstante, hay una diferencia entre los canales izquierdo y derecho de 57 grados en el intervalo 6.7 a 7.4 KHz; dicho desfase, no corresponde a una diferencia de nivel de 6 dB (tal diferencia se obtiene con un desfase de 120 grados entre dos señales), sin embargo, e las gráficas expuestas en el anexo 3, se observa que las atenuaciones

de nivel en las frecuencias cercanas a los intervalos mencionados anteriormente se presentan de manera más evidente en las mediciones de los puntos 2 a 7, puntos en los que la distancia entre ambos altavoces y el micrófono de medición, no es la misma, lo que lleva a que el desfase de 57 grados generado por el ecualizador pueda aumentar y generar interferencias destructivas (desfases entre 120 y 180 grados) o impedir una suma completa (+6 dB) como en aquellas señales que se encuentran completamente en fase. El impacto de la diferencia en la distancia entre monitores y micrófonos se observa en las mediciones con el ecualizador de fase lineal (gráficas 29 a 34 del anexo 3), en donde a pesar de no existir cambios en la fase de las señales izquierda y derecha, se observan puntos a lo largo del espectro donde se presentan algunas atenuaciones que no son evidentes en la posición 1 (gráfica 28, punto equidistante entre los altavoces y el micrófono de medición).

Se observa una diferencia en el porcentaje de memoria RAM consumido por los ecualizadores, siendo mayor en el ecualizador de fase lineal; la diferencia es de 21360 KB, la cual se asocia a la función que define el filtro, es decir, *fir2* para el ecualizador de fase lineal y *yulewalk* para el de fase no lineal, pues es el único punto en el código desarrollado que difiere entre un ecualizador y otro. Sin embargo, la carga de procesamiento representa solo el 0.1% para el ecualizador de fase lineal y el 0.2% para el de fase no lineal, por lo que ninguno de los dos conlleva a un consumo significativo que afecte la funcionalidad y velocidad de procesamiento del computador (teniendo como base de operación una memoria RAM de 4 GB y un procesador Intel Core i3). Además cabe aclarar que aunque cada uno de los algoritmos fueron compilados para trabajar sin necesidad de iniciar Matlab, estos trabajan anclados a este, lo que hace que gran parte de la memoria y procesamiento requeridos no sean netamente asociados al funcionamiento del ecualizador sino a procesos internos del software. Lo anterior hace viable su implementación como plugin que funcione directamente en un sistema DAW y el cual pueda ser utilizado en procesos de postproducción de audio, para lo cual es necesario generar un código en un lenguaje de programación como C++ y utilizar un compilador que permita obtener extensiones como VST.

X. CONCLUSIONES

El ecualizador analógico dbx 2231 es una buena alternativa y da solución a los errores en la respuesta en frecuencia del sistema de reproducción del estudio C de la universidad de San Buenaventura, sin embargo, al tratarse de un ecualizador gráfico, no es posible hacer una configuración precisa pues no permiten modificar directamente frecuencias intermedias a las bandas contenidas en el ecualizador, ni ajustar exactamente el ancho de banda. Aun así, se logró una óptima corrección, sin cambios significativos en la fase de la señal que pudieran generar interferencias destructivas en algún punto del espectro.

Comparando los dos ecualizadores digitales desarrollados en Matlab, se observa un comportamiento muy similar en el rango contenido entre 100 Hz y 6000 Hz, sin embargo a partir de allí hay diferencias evidentes, obteniendo una respuesta más plana con el ecualizador de fase lineal; dicha diferencia se asocia a los cambios de fase entre los canales izquierdo y derecho generados con el ecualizador de fase no lineal, los cuales son mucho mayores a partir de 6000 Hz. Por su parte el ecualizador analógico a pesar de generar cambios a la fase de la señal, no presentó atenuaciones significativas en ningún punto del espectro, debido a que el retraso de fase entre los canales izquierdo y derecho no se alejó más de 20 grados, aunque no puede asegurarse que el ecualizador dbx 2231 siempre mantendrá una relación baja en el desfase de las señales que procesa, pues la fase depende directamente de las bandas modificadas, de la amplitud de cada una de ellas y de la diferencia en los parámetros del ecualizador que deben aplicarse a cada canal, llevando a la conclusión de que los filtros de fase lineal representan una solución un poco más precisa cuando se necesita ecualizar sistemas multicanal.

El consumo de memoria de los filtros digitales desarrollados es muy similar y en ningún momento comprometieron el rendimiento y velocidad de procesamiento del computador en el cual fueron ejecutados, por lo que se concluye que, una calibración a partir de sistemas digitales, incluso ecualizadores de fase lineal, no representan un gasto computacional que afecte el rendimiento de un computador, siempre y cuando sus especificaciones técnicas permitan trabajar software de audio sin problemas de latencia o estabilidad; aunque, los sistemas analógicos son la solución más segura (en cuanto al rendimiento en cada una de las etapas del sistema de reproducción) pues se elimina por completo la carga computacional.

La investigación parte de la hipótesis de trabajo que afirma que los ecualizadores de fase lineal representan una solución mucho más precisa en un proceso de calibración, no obstante, se obtuvieron resultados satisfactorios con los tres ecualizadores evaluados de acuerdo con las recomendaciones de la norma EBU Tech. 3276, por lo que no puede afirmarse que un proceso de fase lineal, supere de manera muy significativamente a aquellos sistemas que generan cambios de fase a la señal, lo que da validez a la hipótesis nula: “Un ecualizador de fase lineal no presenta diferencias significativas en la búsqueda de una respuesta en frecuencia ideal (plana) en un punto específico de escucha de un recinto, comparado con ecualizadores analógicos y o digitales de fase no lineal”. Pero debe considerarse que la investigación fue desarrollada para un sistema estéreo, dejando la posibilidad a futuras investigaciones, de estudiar el impacto de la fase lineal en sistemas surround.

REFERENCIAS

- [1] F. Toole, «The Measurement and Calibration of Sound Reproducing Systems,» *J. Audio Eng. Soc.*, vol. 63, pp. 512-541, 2015.
- [2] R. Vaucher, «Linear Phase Implementation in Loudspeaker Systems: Measurements, Processing Methods, and Application Benefits,» de *Audio Engineering Society*, New York, 2013.
- [3] P. Mapp, «Some Effects of Equalisation on Sound System Intelligibility and Measurement,» de *Audio Engineering Society*, New York, 2003.
- [4] P. Newell, K. Holland, J. Newell y B. Neskov, «New Proposals for the Calibration of Sound in Cinema Rooms,» de *Audio Engineering Society*, Londres, 2011.
- [5] S. Salamouris y V. Tsakiris, «A Versatile Digital Audio Processing Engine for Loudspeaker and Room Equalization,» de *Audio Engineering Society*, 1997.
- [6] A. Goldberg y A. Mäkivirta, «Statistical Analysis of an Automated In-Situ Frequency Response Optimisation Algorithm for Active Loudspeakers,» de *Audio Engineering Society Conference*, 2003.
- [7] R. Väänänen y J. Hiipakka, «Efficient and high-quality equalization using a multirate filterbank and FIR filters,» de *Audio Engineering Society*, Vienna, 2007.
- [8] IK Multimedia, «ARC System 2,» 2017. [En línea]. Available: <https://goo.gl/KFCe6>.
- [9] L. R. Martínez Núñez, «Procesamiento Digital y Control Gestural en Tiempo Real Utilizando una PC con Drivers ASIO para Efectos de Audio,» Universidad de las Americas Puebla, Cholula, 2007.
- [10] Csound, «Get started,» 2016. [En línea]. Available: <https://goo.gl/5yJkn>.
- [11] J. Mantel, «Definitions and Measurements of Fidelity and Fidelity Index of Electroacoustical Components and Electroacoustical Chains,» de *Audio Engineering Society*, Rotterdam, 1973.
- [12] M. Smedegaard, «Low-Input Impedance Current-to-Voltage Conversion Circuit for Current-Output Digital-to-Analog Converters,» de *Audio Engineering Society*, Los Angeles, 1996.
- [13] F. Rumsey, «Loudspeakers, Reflections, and Rooms,» *J. Audio Eng. Soc.*, vol. 56, pp. 394-400, 2008.

- [14] J. M. Storyk, «Application of Varying Architectural and Programmatic Requirements to Audio Recording Studio Acoustic Design,» de *Audio Engineering Society*, New York, 1989.
- [15] E.-J. Valker, «Defects in Studios-Consequences for Recording and Control,» de *Audio Engineering Society*, Hamburg, 1989.
- [16] Z. Davidovic y B. Petrovic, «Acoustical Design of Control Room for Stereo and Multichannel Production and Reproduction, A Novel Approach,» de *Audio Engineering Society*, San Francisco, 2010.
- [17] S. Tervo, P. Laukkanen, J. Pääntynen y T. Lokki, «Preferences of Critical Listening Environments Among Sound Engineers,» *J. Audio Eng. Soc.*, vol. 62, nº 5, pp. 300-314, 2014.
- [18] F. Miyara, *Ecualizadores*, Argentina: Facultad de ciencias exactas, ingeniería y agrimensura.
- [19] A. Ayala Coca, *Manuel de postproducción de audio*, Baena: Editorial Galisgamdigital, 2011.
- [20] E. Gómez, «Introducción al filtrado digital,» Escola Superior de Musica de Catalunya, Catalunya, 2009.
- [21] G. J. Krauss, «Advantages of FIR Filters in Digital Loudspeaker Controllers,» de *Audio Engineering Society*, Paris, 2006.
- [22] G. James y D. Burley, «Matemáticas avanzadas para ingeniería,» Pearson Educación, Naucalpan de Juarez, 2002.
- [23] G.-B. Stan, J.-J. Embrechts y D. Archambeau, «Comparison of Different Impulse Response Measurement Techniques,» *Audio Eng. Soc.*, vol. 50, nº 5, 2002.
- [24] The MathWorks, Inc., «fir2: Frequency sampling-based FIR filter design,» The MathWorks, Inc., 2017. [En línea]. Available: <https://goo.gl/e6E6mE>.
- [25] The MathWorks, Inc, «yulewalk, Recursive digital filter design,» 2017. [En línea]. Available: <https://goo.gl/AGKA5G>.
- [26] International Telecommunication Union, *Recomendation ITU-R BS.1116: Methods for the subjective assessment of small impairments in audio systems*, Ginebra: ITU, 2015.
- [27] K. Bob, «An Integrated Approach to Metering, Monitoring, and Levelling Practices,» *J.AudioEng.So*, vol. 48, 2000.

- [28] IK Multimedia, *ARC 2 system: User manual*, Modena, 2017.
- [29] O. Moreno, S. Pérez y P. Antonio, «Respuesta en frecuencia de altavoces sin utilizar cámara anecoica,» Universidad Nacional Autónoma de México, Ciudad de México, 2006.
- [30] M. Laparte, «Caracterización acústica de recintos mediante software desarrollado en Matlab,» Escuela Técnica superior de ingenieros industriales y de telecomunicación, Pamplona, 2011.
- [31] B. Boren y A. Roginska, «Multichannel Impulse Response Measurement in Matlab,» de *Audio Engineering Society*, New York, 2011.
- [32] J.-w. Jeong, S.-W. Jeong, J. Lee, Y.-c. Park y D.-h. Youn, «Effective Room Equalization Based on Warped Common Acoustical Poles and Zeros,» de *Audio Engineering Society*, Paris, 2006.
- [33] F. Maya, «Ecuallizadores,» 2014. [En línea]. Available: <https://goo.gl/zPP3XN>.
- [34] J. Sandoval Gío, C. A. Luján Ramírez, J. A. Hernández Benítez y A. Chan Mex, «Diseño e implementación de un ecualizador paramétrico digital en el PDS TMS320C713,» Instituto Tecnológico de Mérida, Mérida, 2007.

ANEXOS

Anexo 1. Código desarrollado en Matlab

```

function varargout = Linear_Phase_EQ(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @Linear_Phase_EQ_OpeningFcn, ...
                  'gui_OutputFcn',  @Linear_Phase_EQ_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Linear_Phase_EQ_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

guidata(hObject, handles);

%inicialización de variables

set(handles.slider1, 'value', 50);
set(handles.slider2, 'value', 0);

set(handles.slider3, 'value', 250);
set(handles.slider4, 'value', 0);
set(handles.slider5, 'value', 0.5);

set(handles.slider6, 'value', 1000);
set(handles.slider7, 'value', 0);
set(handles.slider8, 'value', 0.5);

set(handles.slider9, 'value', 5000);
set(handles.slider10, 'value', 0);
set(handles.slider11, 'value', 0.5);

set(handles.slider12, 'value', 8000);
set(handles.slider13, 'value', 0);
set(handles.slider14, 'value', 0.5);

set(handles.slider15, 'value', 12000);
set(handles.slider16, 'value', 0);
set(handles.slider17, 'value', 0.5);

set(handles.slider18, 'value', 16000);
set(handles.slider19, 'value', 0);

function varargout = Linear_Phase_EQ_OutputFcn(hObject, eventdata, handles)

```

```
varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)

[FileName Path]=uigetfile({'*.wav'});

[s,fs,Nbits]=wavread(strcat(Path,FileName));

L=s(:,1);
handles.L=L;
guidata(hObject,handles);

FyC=size(s);
if FyC(1,2)==2
R=s(:,2);
handles.R=R;
guidata(hObject,handles);
else
R=s(:,1);
handles.R=R;
guidata(hObject,handles);
end

%Creación de funciones de reproducción

stereo=[L,R];
handles.stereo=stereo;
guidata(hObject,handles);

player=audioplayer(s,fs);
handles.player=player;
guidata(hObject,handles);

player2=audioplayer(stereo,fs);
handles.player2=player2;
guidata(hObject,handles);

%lectura de características del audio de entrada

handles.s=s;
guidata(hObject,handles);

handles.fs=fs;
guidata(hObject,handles);

handles.Nbits=Nbits;
guidata(hObject,handles);

f=0;
handles.f=f;
guidata(hObject,handles);

m=1;
handles.m=m;
guidata(hObject,handles);

fn=fs/2;
```

```

handles.fn=fn;
guidata(hObject,handles);

p='no';
handles.p=p;
guidata(hObject,handles);

axes(handles.axes3);

time=linspace(1,length(s)/fs,length(s));

plot(time,s);

function checkbox1_Callback(hObject, eventdata, handles)

%cada Slider corresponde a una variable modificable del ecualizador como
frecuencia, ganancia o ancho de banda

function slider1_Callback(hObject, eventdata, handles)

fc1=get(handles.slider1,'value');

set(handles.edit1,'string',num2str(fc1));

function slider1_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider2_Callback(hObject, eventdata, handles)

g1=get(handles.slider2,'value');

set(handles.edit2,'string',num2str(g1));

k1=10^(g1/20);

function slider2_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function checkbox2_Callback(hObject, eventdata, handles)

function slider3_Callback(hObject, eventdata, handles)

fc2=get(hObject,'Value');

set(handles.edit3,'string',num2str(fc2));

function slider3_CreateFcn(hObject, eventdata, handles)

```

```
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider4_Callback(hObject, eventdata, handles)

g2=get(hObject,'Value');

set(handles.edit4,'string',num2str(g2));

k2=10^(g2/20);

function slider4_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider5_Callback(hObject, eventdata, handles)

BW1=get(hObject,'Value');

set(handles.edit5,'string',num2str(BW1));

function slider5_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function checkbox3_Callback(hObject, eventdata, handles)

function slider6_Callback(hObject, eventdata, handles)

fc3=get(hObject,'Value');

set(handles.edit6,'string',num2str(fc3));

function slider6_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider7_Callback(hObject, eventdata, handles)

g3=get(hObject,'Value');

set(handles.edit7,'string',num2str(g3));

k3=10^(g3/20);

function slider7_CreateFcn(hObject, eventdata, handles)
```

```

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider8_Callback(hObject, eventdata, handles)

BW2=get(hObject,'Value');

set(handles.edit8,'string',num2str(BW2));

function slider8_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function checkbox4_Callback(hObject, eventdata, handles)

function slider9_Callback(hObject, eventdata, handles)

fc4=get(hObject,'Value');

set(handles.edit9,'string',num2str(fc4));

function slider9_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider10_Callback(hObject, eventdata, handles)

g4=get(hObject,'Value');

set(handles.edit10,'string',num2str(g4));

k4=10^(g4/20);

function slider10_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider11_Callback(hObject, eventdata, handles)

BW3=get(hObject,'Value');

set(handles.edit11,'string',num2str(BW3));

function slider11_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))

```

```
        set(hObject, 'BackgroundColor', [.9 .9 .9]);
    end

function checkbox5_Callback(hObject, eventdata, handles)

function slider12_Callback(hObject, eventdata, handles)

fc5=get(hObject, 'Value');

set(handles.edit12, 'string', num2str(fc5));

function slider12_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', [.9 .9 .9]);
end

function slider13_Callback(hObject, eventdata, handles)

g5=get(hObject, 'Value');

set(handles.edit13, 'string', num2str(g5));

k5=10^(g5/20);

function slider13_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', [.9 .9 .9]);
end

function slider14_Callback(hObject, eventdata, handles)

BW4=get(hObject, 'Value');

set(handles.edit14, 'string', num2str(BW4));

function slider14_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', [.9 .9 .9]);
end

function checkbox6_Callback(hObject, eventdata, handles)

function slider15_Callback(hObject, eventdata, handles)

fc6=get(hObject, 'Value');

set(handles.edit15, 'string', num2str(fc6));

function slider15_CreateFcn(hObject, eventdata, handles)
```

```

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider16_Callback(hObject, eventdata, handles)

g6=get(hObject,'Value');

set(handles.edit16,'string',num2str(g6));

k6=10^(g6/20);

function slider16_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider17_Callback(hObject, eventdata, handles)

BW5=get(hObject,'Value');

set(handles.edit17,'string',num2str(BW5));

function slider17_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function checkbox7_Callback(hObject, eventdata, handles)

function checkbox9_Callback(hObject, eventdata, handles)

function slider18_Callback(hObject, eventdata, handles)

fc7=get(hObject,'Value');

set(handles.edit18,'string',num2str(fc7));

function slider18_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function slider19_Callback(hObject, eventdata, handles)

g7=get(hObject,'Value');

set(handles.edit19,'string',num2str(g7));

k7=10^(g7/20);

```

```

function slider19_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function edit1_Callback(hObject, eventdata, handles)

fc=get(hObject,'string');

set(handles.slider1,'value',str2num(fc));

%Cada Edit corresponde al valor numérico descrito en el slider

function edit1_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_Callback(hObject, eventdata, handles)

g1=get(hObject,'string');

set(handles.slider2,'value',str2num(g1));

function edit2_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject, eventdata, handles)

fc2=get(hObject,'string');

set(handles.slider3,'value',str2num(fc2));

function edit3_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)

g2=get(hObject,'string');

set(handles.slider4,'value',str2num(g2));

function edit4_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```
end

function edit5_Callback(hObject, eventdata, handles)
BW1=get(hObject, 'string');

set(handles.slider5, 'value', str2num(BW1));

function edit5_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit6_Callback(hObject, eventdata, handles)

fc3=get(hObject, 'string');

set(handles.slider6, 'value', str2num(fc3));

function edit6_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit7_Callback(hObject, eventdata, handles)

g3=get(hObject, 'string');

set(handles.slider7, 'value', str2num(g3));

function edit7_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit8_Callback(hObject, eventdata, handles)

BW2=get(hObject, 'string');

set(handles.slider8, 'value', str2num(BW2));

function edit8_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function edit9_Callback(hObject, eventdata, handles)
```

```

fc4=get(hObject,'string');

set(handles.slider9,'value',str2num(fc4));

function edit9_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit10_Callback(hObject, eventdata, handles)

g4=get(hObject,'string');

set(handles.slider10,'value',str2num(g4));

function edit10_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit11_Callback(hObject, eventdata, handles)

BW3=get(hObject,'string');

set(handles.slider11,'value',str2num(BW3));

function edit11_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit12_Callback(hObject, eventdata, handles)

fc5=get(hObject,'string');

set(handles.slider12,'value',str2num(fc5));

function edit12_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit13_Callback(hObject, eventdata, handles)

g5=get(hObject,'string');

set(handles.slider13,'value',str2num(g5));

```

```
function edit13_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit14_Callback(hObject, eventdata, handles)

BW4=get(hObject,'string');

set(handles.slider14,'value',str2num(BW4));

function edit14_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit15_Callback(hObject, eventdata, handles)

fc6=get(hObject,'string');

set(handles.slider15,'value',str2num(fc6));

function edit15_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit16_Callback(hObject, eventdata, handles)

g6=get(hObject,'string');

set(handles.slider16,'value',str2num(g6));

function edit16_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit17_Callback(hObject, eventdata, handles)

BW5=get(hObject,'string');

set(handles.slider17,'value',str2num(BW5));

function edit17_CreateFcn(hObject, eventdata, handles)
```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit18_Callback(hObject, eventdata, handles)

fc7=get(hObject,'string');

set(handles.slider18,'value',str2num(fc7));

function edit18_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit19_Callback(hObject, eventdata, handles)

g7=get(hObject,'string');

set(handles.slider19,'value',str2num(g7));

function edit19_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function slider1_KeyPressFcn(hObject, eventdata, handles)

function edit1_KeyPressFcn(hObject, eventdata, handles)

%Boton que aplica el ecualizador al canal izquierdo

function pushbutton2_Callback(hObject, eventdata, handles)

s=handles.s;
fs=handles.fs;
Nbits=handles.Nbits;
fn=handles.fn;
f=handles.f;
m=handles.m;
stereo=handles.stereo;
player2=handles.player2;

%LECTURA DE VARIABLES
fc1=get(handles.slider1,'value');
g1=get(handles.slider2,'value');
k1=10^(g1/20);

fc2=get(handles.slider3,'value');
g2=get(handles.slider4,'value');
k2=10^(g2/20);
BW1=get(handles.slider5,'value');

```

```

fc3=get(handles.slider6,'value');
g3=get(handles.slider7,'value');
k3=10^(g3/20);
BW2=get(handles.slider8,'value');

fc4=get(handles.slider9,'value');
g4=get(handles.slider10,'value');
k4=10^(g4/20);
BW3=get(handles.slider11,'value');

fc5=get(handles.slider12,'value');
g5=get(handles.slider13,'value');
k5=10^(g5/20);
BW4=get(handles.slider14,'value');

fc6=get(handles.slider15,'value');
g6=get(handles.slider16,'value');
k6=10^(g6/20);
BW5=get(handles.slider17,'value');

fc7=get(handles.slider18,'value');
g7=get(handles.slider19,'value');
k7=10^(g7/20);

%identificación de bandas activas
if get(handles.checkbox1,'value')==1

    if get(handles.checkbox10,'value')==0

        fcn1=fc1/fn;
        f(1,length(f)+1)=fcn1;
        f(1,length(f)+1)=fcn1;

        m(1,1:length(m)+1)=k1;
        m(1,length(m)+1)=1;

        else

            fcn1=fc1/fn;
            f(1,length(f)+1)=fcn1;
            f(1,length(f)+1)=fcn1;

            m(1,1:length(m)+1)=0;
            m(1,length(m)+1)=1;
            end

    end

if get(handles.checkbox2,'value')==1

    fcn2=fc2/fn;
    f(1,length(f)+3)=fcn2;

    fc21=fc2-(fc2*(BW1/2));
    fc22=fc2+(fc2*(BW1/2));
    fcn21=fc21/fn;
    fcn22=fc22/fn;

```

```

f(1,length(f)-2)=fcn21;
f(1,length(f)-1)=fcn21;
f(1,length(f)+1)=fcn22;
f(1,length(f)+1)=fcn22;

m(1,length(m)+3)=k2;

m(1,length(m)-1)=k2;
m(1,length(m)+1)=k2;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox3,'value')==1

fc3=fc3/fn;
f(1,length(f)+3)=fcn3;

fc31=fc3-(fc3*(BW2/2));
fc32=fc3+(fc3*(BW2/2));
fcn31=fc31/fn;
fcn32=fc32/fn;

f(1,length(f)-2)=fcn31;
f(1,length(f)-1)=fcn31;
f(1,length(f)+1)=fcn32;
f(1,length(f)+1)=fcn32;

m(1,length(m)+3)=k3;

m(1,length(m)-1)=k3;
m(1,length(m)+1)=k3;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox4,'value')==1

fc4=fc4/fn;
f(1,length(f)+3)=fcn4;

fc41=fc4-(fc4*(BW3/2));
fc42=fc4+(fc4*(BW3/2));
fcn41=fc41/fn;
fcn42=fc42/fn;

f(1,length(f)-2)=fcn41;
f(1,length(f)-1)=fcn41;
f(1,length(f)+1)=fcn42;
f(1,length(f)+1)=fcn42;

m(1,length(m)+3)=k4;

m(1,length(m)-1)=k4;

```

```

m(1,length(m)+1)=k4;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox5,'value')==1

fc5=fc5/fn;
f(1,length(f)+3)=fc5;

fc51=fc5-(fc5*(BW4/2));
fc52=fc5+(fc5*(BW4/2));
fc51=fc51/fn;
fc52=fc52/fn;

f(1,length(f)-2)=fc51;
f(1,length(f)-1)=fc51;
f(1,length(f)+1)=fc52;
f(1,length(f)+1)=fc52;

m(1,length(m)+3)=k5;

m(1,length(m)-1)=k5;
m(1,length(m)+1)=k5;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox6,'value')==1

fc6=fc6/fn;
f(1,length(f)+3)=fc6;

fc61=fc6-(fc6*(BW5/2));
fc62=fc6+(fc6*(BW5/2));
fc61=fc61/fn;
fc62=fc62/fn;

f(1,length(f)-2)=fc61;
f(1,length(f)-1)=fc61;
f(1,length(f)+1)=fc62;
f(1,length(f)+1)=fc62;

m(1,length(m)+3)=k6;

m(1,length(m)-1)=k6;
m(1,length(m)+1)=k6;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox7,'value')==1

```

```

        if get(handles.checkbox11, 'value')==0

fcn7=fc7/fn;
f(1,length(f)+1)=fcn7;
f(1,length(f)+1)=fcn7;

m(1,length(m)+1)=1;
m(1,length(m)+1)=k7;

        else

                fcn7=fc7/fn;
f(1,length(f)+1)=fcn7;
f(1,length(f)+1)=fcn7;

m(1,length(m)+1)=1;
m(1,length(m)+1)=0;

        end

end

for i=2:length(f)

        if f(i) < f(i-1)

                f(i)=f(i-1);
                m(i)=m(i-1);

        end

end

f(1,length(f)+1)=1;

if get(handles.checkbox11, 'value')==0
m(1,length(m)+1)=k7;
else
m(1,length(m)+1)=0;
end

fs0=fs/20;
fc0=(fs0/2)/fn;
f1=0;

for i=1:length(f);
        if f(i)<=fc0;
                f0(i)=(f(i)*fn)/(fs0/2);

        else
                f1(i-(length(f0)-1))=f(i);
        end
end

```

```

end

f0(length(f0)+1)=1;
m0=m(1:length(f0));
m1=m(length(f0)-1:length(f));

L=handles.L;

%Aplicación de los filtros

fpa=[0 (fs/20)/fs (fs/20)/fs 1];
mpa=[0 0 1 1];
pasaaltas=fir2(100,fpa,mpa);
%Aplicación del filtro pasa altas
L1=conv(L,pasaaltas,'same');

fpb=[0 (fs/20)/fs (fs/20)/fs 1];
mpb=[1 1 0 0];
pasabajas=fir2(100,fpb,mpb);
%Aplicación del filtro pasa bajas

L2=conv(L,pasabajas,'same');
L22=resample(L2,1,10);

%Definición de la función que describe el filtro

b01 = fir2(100,f0,m0,fs,51);
[h01,w0] = freqz(b01,1);

b11 = fir2(100,f1,m1,fs,51);
[h11,w1] = freqz(b11,1);

L1=conv(L1,b11,'same');
L22=conv(L22,b01,'same');

%Remuestreo de la señal por debajo de 2050Hz
L2=resample(L22,10,1);
L=L1+L2;
handles.L=L;

%Definición de la señal a reproducir
R=handles.R;
guidata(hObject,handles);
stereo=[L,R];
handles.stereo=stereo;
guidata(hObject,handles);
player2=audioplayer(stereo,fs);
handles.player2=player2;
guidata(hObject,handles);
axes(handles.axes1);

%Grafica del filtro
semilogx(f*(fs/2),20.*log10(m),(w/pi)*(fs/2),20.*log10(abs(h1)))
axis([20 20000 -12 12])
xlabel('\omega / \pi')
lgs = {'Ideal','fir2 default'};

```

```

legend(lgs);

axes(handles.axes2);

%grafica de respuesta en fase
phasedelay(b1);

function pushbutton3_Callback(hObject, eventdata, handles)

s=handles.s;
fs=handles.fs;
Nbits=handles.Nbits;
fn=handles.fn;
player=handles.player;
player2=handles.player2;

p=handles.p;

if get(handles.checkbox9, 'value')==1;
    if p=='si';
        resume(player);
    else
        play(player);
    end
else
    if p=='si';
        resume(player2);
    else
        play(player2);
    end
end

function pushbutton4_Callback(hObject, eventdata, handles)

player=handles.player;
player2=handles.player2;

pause(player);
pause(player2);

p='si';
handles.p=p;
guidata(hObject,handles);

function pushbutton5_Callback(hObject, eventdata, handles)

player=handles.player;
player2=handles.player2;

stop(player);
stop(player2);

p='no';
handles.p=p;

function pushbutton6_Callback(hObject, eventdata, handles)

s=handles.s;

```

```

fs=handles.fs;
Nbits=handles.Nbits;
fn=handles.fn;
f=handles.f;
m=handles.m;
player2=handles.player2;

%LECTURA DE VARIABLES
fc1=get(handles.slider1,'value');
g1=get(handles.slider2,'value');
k1=10^(g1/20);

fc2=get(handles.slider3,'value');
g2=get(handles.slider4,'value');
k2=10^(g2/20);
BW1=get(handles.slider5,'value');

fc3=get(handles.slider6,'value');
g3=get(handles.slider7,'value');
k3=10^(g3/20);
BW2=get(handles.slider8,'value');

fc4=get(handles.slider9,'value');
g4=get(handles.slider10,'value');
k4=10^(g4/20);
BW3=get(handles.slider11,'value');

fc5=get(handles.slider12,'value');
g5=get(handles.slider13,'value');
k5=10^(g5/20);
BW4=get(handles.slider14,'value');

fc6=get(handles.slider15,'value');
g6=get(handles.slider16,'value');
k6=10^(g6/20);
BW5=get(handles.slider17,'value');

fc7=get(handles.slider18,'value');
g7=get(handles.slider19,'value');
k7=10^(g7/20);

%identificación de bandas activas
if get(handles.checkbox1,'value')==1

    if get(handles.checkbox10,'value')==0

        fcn1=fc1/fn;
        f(1,length(f)+1)=fcn1;
        f(1,length(f)+1)=fcn1;

        m(1,1:length(m)+1)=k1;
        m(1,length(m)+1)=1;

    else

        fcn1=fc1/fn;
        f(1,length(f)+1)=fcn1;
        f(1,length(f)+1)=fcn1;

```

```

m(1,1:length(m)+1)=0;
m(1,length(m)+1)=1;
    end

end

if get(handles.checkbox2,'value')==1

fc2=fc2/fn;
f(1,length(f)+3)=fc2;

fc21=fc2-(fc2*(BW1/2));
fc22=fc2+(fc2*(BW1/2));
fcn21=fc21/fn;
fcn22=fc22/fn;

f(1,length(f)-2)=fcn21;
f(1,length(f)-1)=fcn21;
f(1,length(f)+1)=fcn22;
f(1,length(f)+1)=fcn22;

m(1,length(m)+3)=k2;

m(1,length(m)-1)=k2;
m(1,length(m)+1)=k2;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox3,'value')==1

fc3=fc3/fn;
f(1,length(f)+3)=fc3;

fc31=fc3-(fc3*(BW2/2));
fc32=fc3+(fc3*(BW2/2));
fcn31=fc31/fn;
fcn32=fc32/fn;

f(1,length(f)-2)=fcn31;
f(1,length(f)-1)=fcn31;
f(1,length(f)+1)=fcn32;
f(1,length(f)+1)=fcn32;

m(1,length(m)+3)=k3;

m(1,length(m)-1)=k3;
m(1,length(m)+1)=k3;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox4,'value')==1

```

```

fc4=fc4/fn;
f(1,length(f)+3)=fc4;

fc41=fc4-(fc4*(BW3/2));
fc42=fc4+(fc4*(BW3/2));
fcn41=fc41/fn;
fcn42=fc42/fn;

f(1,length(f)-2)=fcn41;
f(1,length(f)-1)=fcn41;
f(1,length(f)+1)=fcn42;
f(1,length(f)+1)=fcn42;

m(1,length(m)+3)=k4;

m(1,length(m)-1)=k4;
m(1,length(m)+1)=k4;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox5,'value')==1

fc5=fc5/fn;
f(1,length(f)+3)=fc5;

fc51=fc5-(fc5*(BW4/2));
fc52=fc5+(fc5*(BW4/2));
fcn51=fc51/fn;
fcn52=fc52/fn;

f(1,length(f)-2)=fcn51;
f(1,length(f)-1)=fcn51;
f(1,length(f)+1)=fcn52;
f(1,length(f)+1)=fcn52;

m(1,length(m)+3)=k5;

m(1,length(m)-1)=k5;
m(1,length(m)+1)=k5;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox6,'value')==1

fc6=fc6/fn;
f(1,length(f)+3)=fc6;

fc61=fc6-(fc6*(BW5/2));
fc62=fc6+(fc6*(BW5/2));
fcn61=fc61/fn;
fcn62=fc62/fn;

```

```

f(1,length(f)-2)=fc61;
f(1,length(f)-1)=fc61;
f(1,length(f)+1)=fc62;
f(1,length(f)+1)=fc62;

m(1,length(m)+3)=k6;

m(1,length(m)-1)=k6;
m(1,length(m)+1)=k6;

m(1,length(m)-3)=1;
m(1,length(m)+1)=1;

end

if get(handles.checkbox7,'value')==1

    if get(handles.checkbox11,'value')==0

        fcn7=fc7/fn;
        f(1,length(f)+1)=fc7;
        f(1,length(f)+1)=fc7;

        m(1,length(m)+1)=1;
        m(1,length(m)+1)=k7;

        else

            fcn7=fc7/fn;
            f(1,length(f)+1)=fc7;
            f(1,length(f)+1)=fc7;

            m(1,length(m)+1)=1;
            m(1,length(m)+1)=0;

        end

    end

end

for i=2:length(f)

    if f(i) < f(i-1)

        f(i)=f(i-1);
        m(i)=m(i-1);

    end

end

end

f(1,length(f)+1)=1;
if get(handles.checkbox11,'value')==0
m(1,length(m)+1)=k7;
else
m(1,length(m)+1)=0;
end

```

```

fs0=fs/20;
fc0=(fs0/2)/fn;
f1=0;

for i=1:length(f);
    if f(i)<=fc0;
        f0(i)=(f(i)*fn)/(fs0/2);

        else
            f1(i-(length(f0)-1))=f(i);
        end
end

f0(length(f0)+1)=1;
m0=m(1:length(f0));
m1=m(length(f0)-1:length(f));

R=handles.R;

%Separación de la señal

fpa=[0 (fs/20)/fs (fs/20)/fs 1];
mpa=[0 0 1 1];
pasaaltas=fir2(100,fpa,mpa);
R1=conv(R,pasaaltas,'same');

fpb=[0 (fs/20)/fs (fs/20)/fs 1];
mpb=[1 1 0 0];
pasabajas=fir2(100,fpb,mpb);
R2=conv(R,pasabajas,'same');
R22=resample(R2,1,10);

b01 = fir2(100,f0,m0,fs,51);
[h01,w0] = freqz(b01,1);

b11 = fir2(100,f1,m1,fs,51);
[h11,w1] = freqz(b11,1);

%Aplicación de los filtros pasa altas y pasa bajas

R1=conv(R1,b11,'same');
R22=conv(R22,b01,'same');
R2=resample(R22,10,1);
R=R1+R2;
handles.R=R;

b2 = fir2(100,f,m,8192,25);
[h2,w] = freqz(b2,1);

f2=phase(h2);

L=handles.L;

```

```
FyC=size(s);

if FyC(1,2)==2
R=conv(s(:,2),b2,'same');
handles.R=R;
guidata(hObject,handles);
else
R=conv(s(:,1),b2,'same');
handles.R=R;
guidata(hObject,handles);
end

stereo=handles.stereo;
stereo=[L,R];
handles.stereo=stereo;
guidata(hObject,handles);

player2=audioplayer(stereo,fs);
handles.player2=player2;
guidata(hObject,handles);

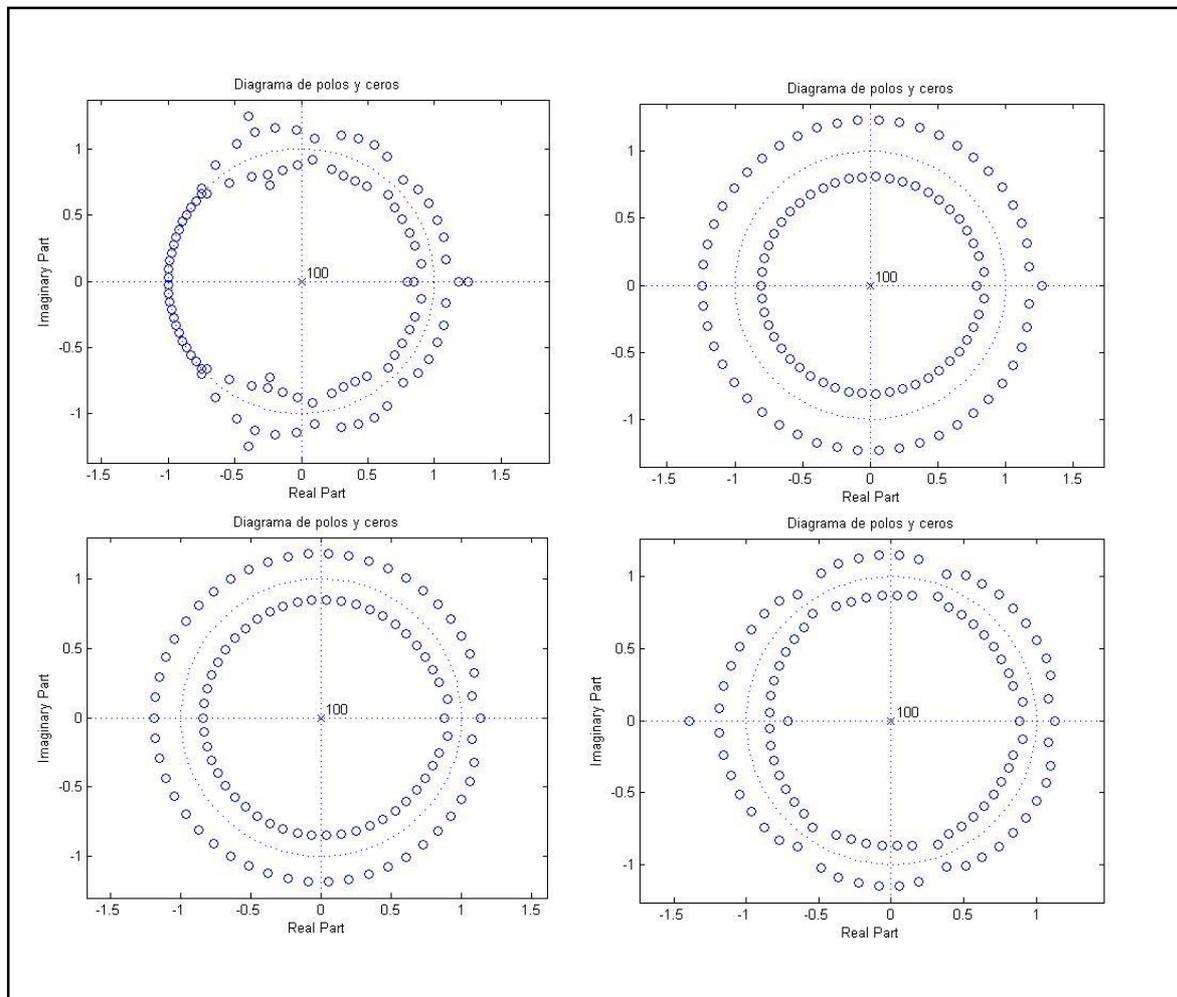
axes(handles.axes4);

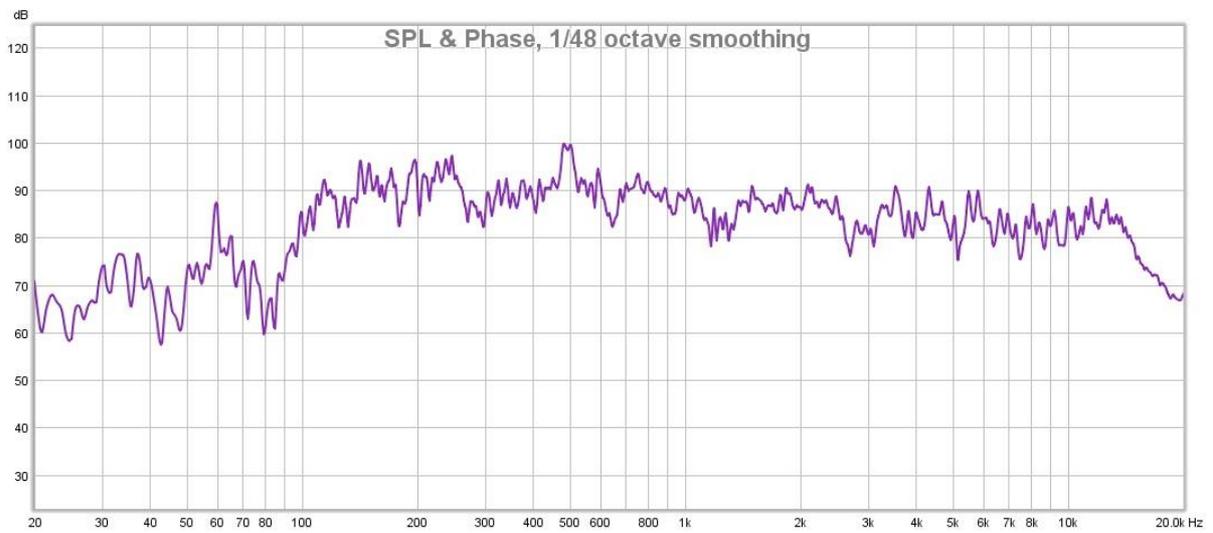
%grafica de respuesta del filtro
semilogx(f*(fs/2),20.*log10(m),(w/pi)*(fs/2),20.*log10(abs(h2)));
axis([20 20000 -12 12])
xlabel('\omega / \pi');
lgs = {'Ideal','fir2 default'};
legend(lgs);

function checkbox10_Callback(hObject, eventdata, handles)

function checkbox11_Callback(hObject, eventdata, handles)
```

Anexo 2. Diagrama de polos y ceros del ecualizador de fase lineal con configuraciones aleatorias de frecuencia, ganancia y ancho de banda



Anexo 3. Resultado de cada una de las mediciones con ambos altavoces encendidos***Ecualizador de fase lineal***

Gráfica 1. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase lineal medido en la posición 1.



Gráfica 2. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase lineal medido en la posición 2.



Gráfica 1. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase lineal medido en la posición 3.



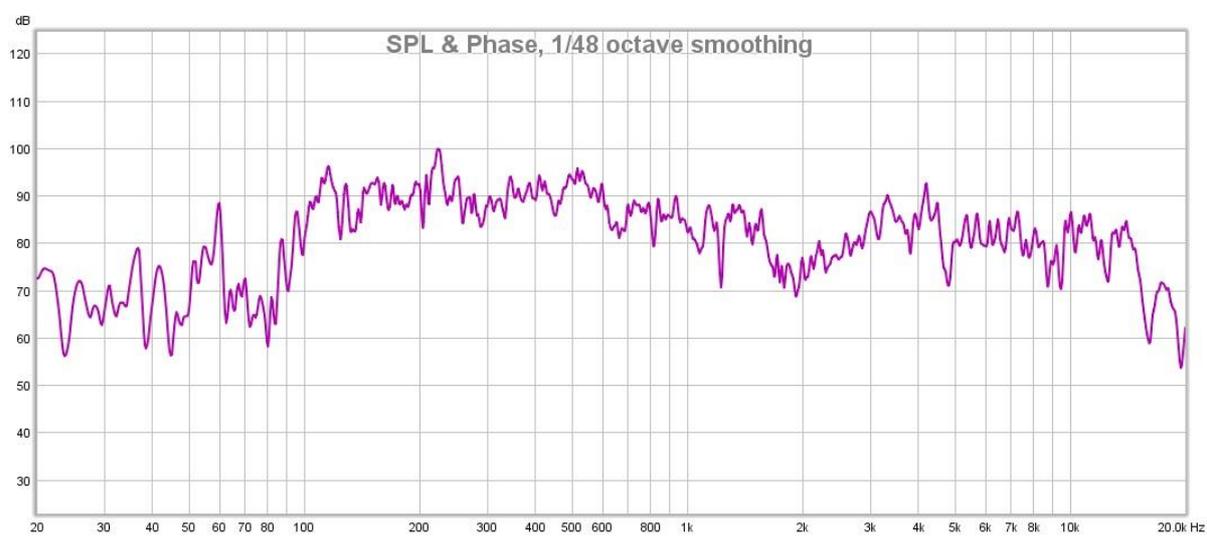
Gráfica 4. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase lineal medido en la posición 4.



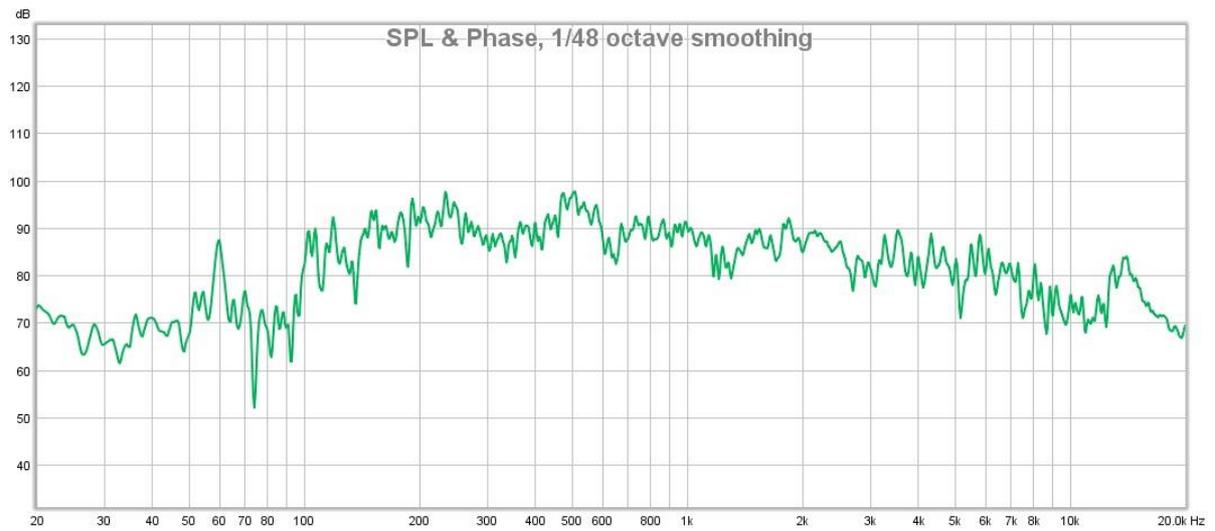
Gráfica 5. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase lineal medido en la posición 5.



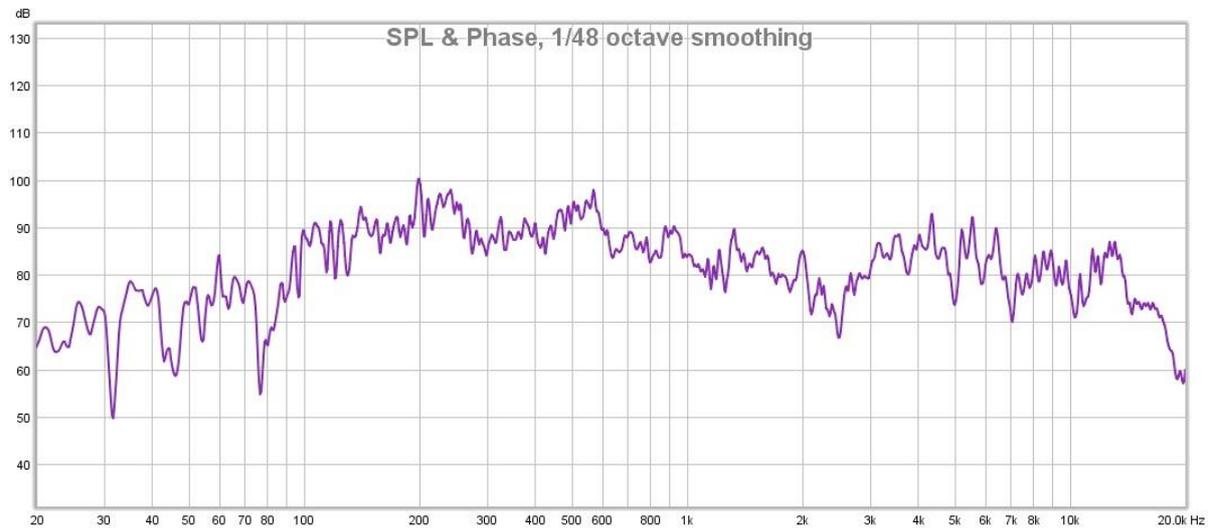
Gráfica 6. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase lineal medido en la posición 6.



Gráfica 7. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase lineal medido en la posición 7.

Ecualizador de fase no lineal

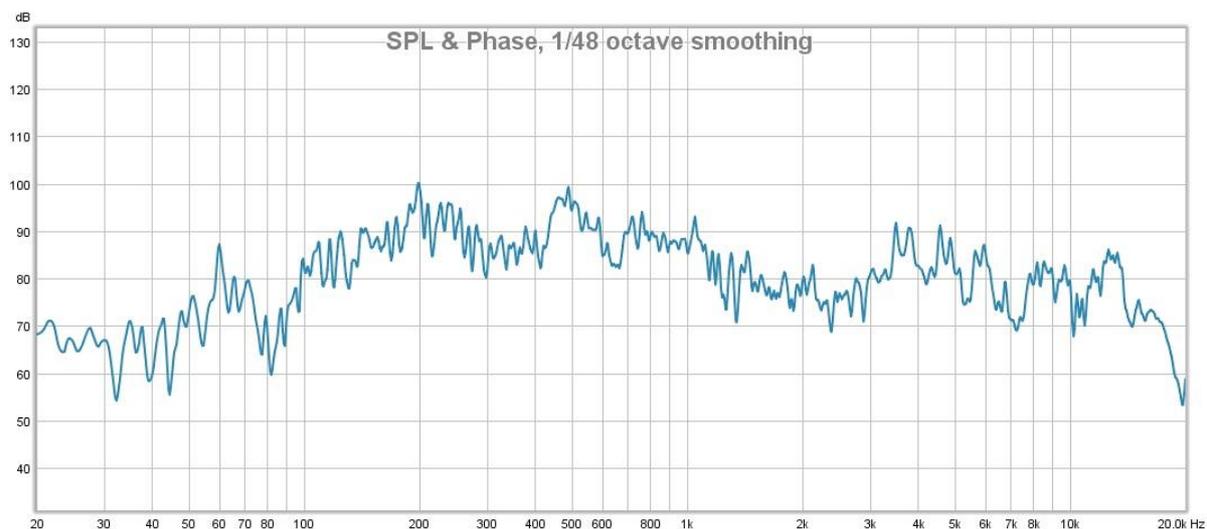
Gráfica 8. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase no lineal medido en la posición 1.



Gráfica 9. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase no lineal medido en la posición 2.



Gráfica 10. Respuesta estéreo de la sala luego de aplicar el ecualizador de fase no lineal medido en la posición 3.



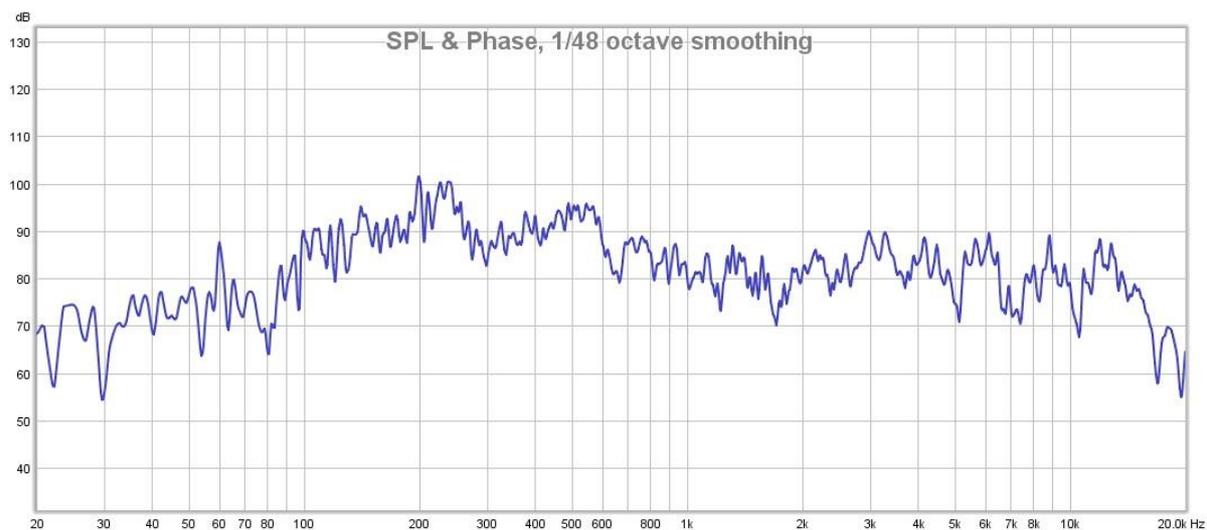
Gráfica 11. Respuesta estereo de la sala luego de aplicar el ecualizador de fase no lineal medido en la posición 4.



Gráfica 12. Respuesta estereo de la sala luego de aplicar el ecualizador de fase no lineal medido en la posición 5.



Gráfica 13. Respuesta estereo de la sala luego de aplicar el ecualizador de fase no lineal medido en la posición 6.



Gráfica 14. Respuesta estereo de la sala luego de aplicar el ecualizador de fase no lineal medido en la posición 7.

Ecualizador analógico dbx 2231



Gráfica 15. Respuesta estereo de la sala luego de aplicar el ecualizador analógico medido en la posición 1.



Gráfica 16. Respuesta estéreo de la sala luego de aplicar el ecualizador analógico medido en la posición 2.



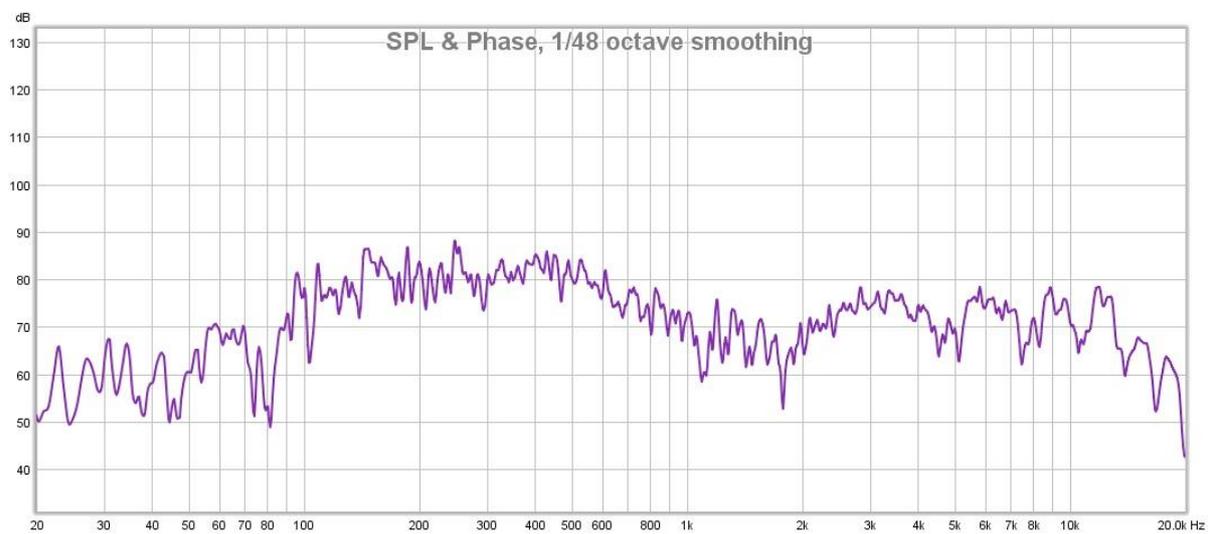
Gráfica 17. Respuesta estéreo de la sala luego de aplicar el ecualizador analógico medido en la posición 3.



Gráfica 18. Respuesta estéreo de la sala luego de aplicar el ecualizador analógico medido en la posición 4.



Gráfica 19. Respuesta estéreo de la sala luego de aplicar el ecualizador analógico medido en la posición 5.



Gráfica 20. Respuesta estéreo de la sala luego de aplicar el ecualizador analógico medido en la posición 6.



Gráfica 21. Respuesta estéreo de la sala luego de aplicar el ecualizador analógico medido en la posición 7.