

Tratamiento y publicación de audio. Ejercicios

Autores: Enrique Guaus Termens y Francisco Martí Pérez

El encargo y la creación de este recurso de aprendizaje UOC han sido coordinados por el profesor: Javier Melenchón Maldonado (2019)

PID_00260379

Ejercicios resueltos módulo 1

Ejercicio 1

Enunciado

Solución

Ejercicio 2

Enunciado

Solución

Ejercicio 3

Enunciado

Solución

Ejercicio 4

Enunciado

Solución

Ejercicio 5

Enunciado

Solución

Ejercicio 6

Enunciado

Solución

Ejercicio 7

Enunciado

Solución

Ejercicio 8

Enunciado

Solución

Ejercicio 9

Enunciado

Solución

Ejercicio 10

Enunciado

Solución

Ejercicio 11

Enunciado

Solución

Ejercicios resueltos módulo 2

Ejercicio 1

Enunciado

Solución

Ejercicio 2

Enunciado

Solución

Ejercicio 3

Enunciado

Solución

Ejercicio 4

Enunciado

Solución

Ejercicio 5

Enunciado

Solución

Ejercicio 6

Enunciado

Solución

Ejercicios propuestos módulo 2

Ejercicio 1

Ejercicio 2

Ejercicio 3

Ejercicio 4

Ejercicio 5

Ejercicio 6

Ejercicio 7

Ejercicio 8

Ejercicio 9

Ejercicio 10

Ejercicio 11

Ejercicio 12

Ejercicio 13

Ejercicio 14

Ejercicio 15

Ejercicio 16

Ejercicios resueltos módulo 3

Ejercicio 1

Enunciado

Solución

Ejercicio 2

Enunciado

Solución

Ejercicio 3

Enunciado

Solución

Ejercicio 4

Enunciado

Solución

Ejercicio 5

Enunciado

Solución

Ejercicios resueltos módulo 4

Ejercicio 1

Enunciado

Solución

Ejercicio 2

Enunciado

Solución

Ejercicio 3

Enunciado

Solución

Ejercicio 4

Enunciado

Solución

Ejercicio 5

Enunciado

Solución

Ejercicio 6

Enunciado

Solución

Ejercicios resueltos módulo 5

Ejercicio 1

Enunciado

Solución

Ejercicio 2

Enunciado

Solución

Ejercicio 3

Enunciado

Solución

Ejercicio 4

Enunciado

Solución

Ejercicios resueltos módulo 7

Ejercicio 1

Enunciado

Solución

Ejercicio 2

Enunciado

Solución

Ejercicio 3

Enunciado

Solución

Ejercicios propuestos módulo 7

Ejercicio 1

Ejercicio 2

Ejercicio 3

Recursos

Ejercicios módulo 1

Ejercicio 1

Enunciado

A partir de la forma de onda sinusoidal de las figuras 1, 2 y 3, determina el periodo, la frecuencia y la amplitud de la señal.

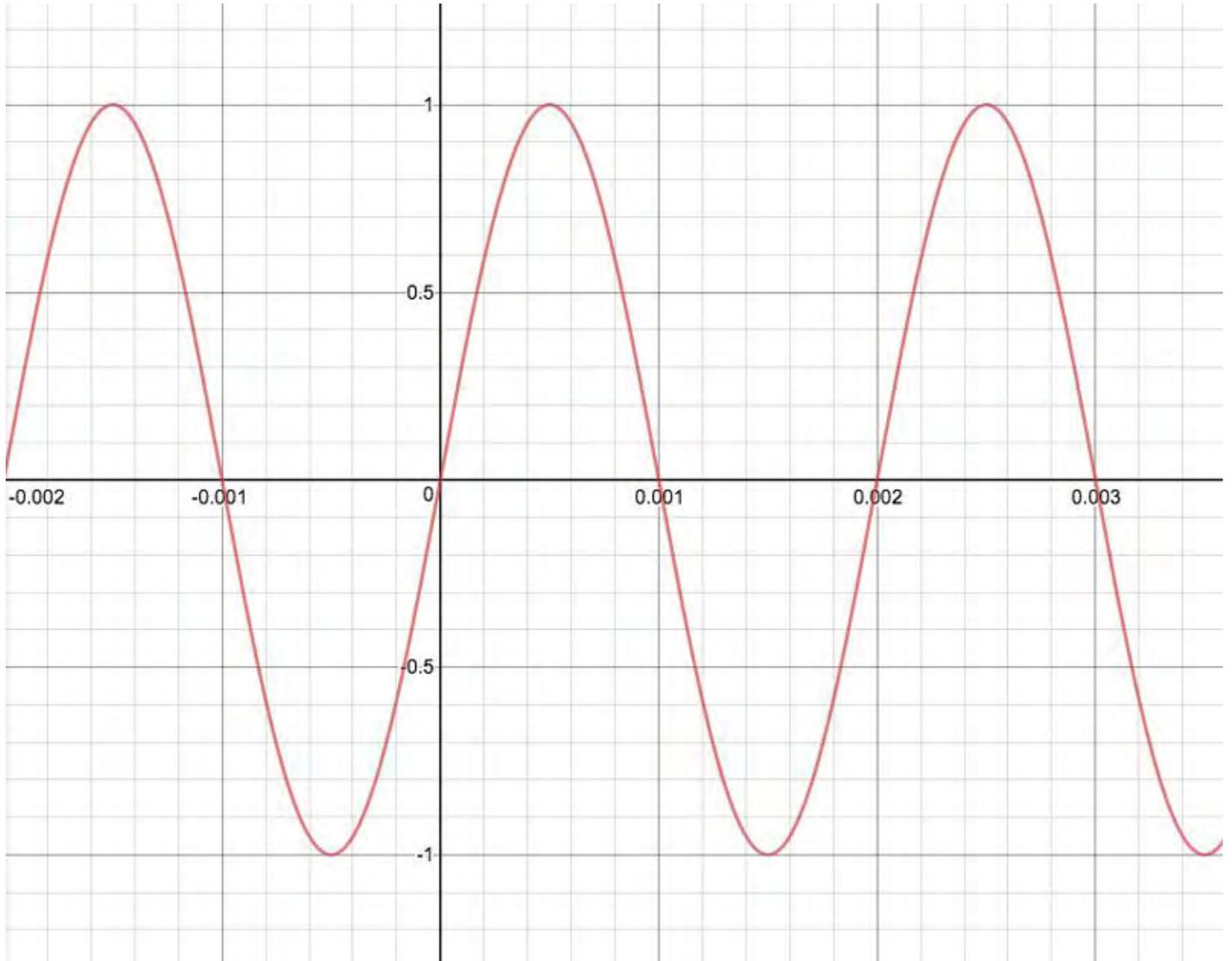


Figura 1

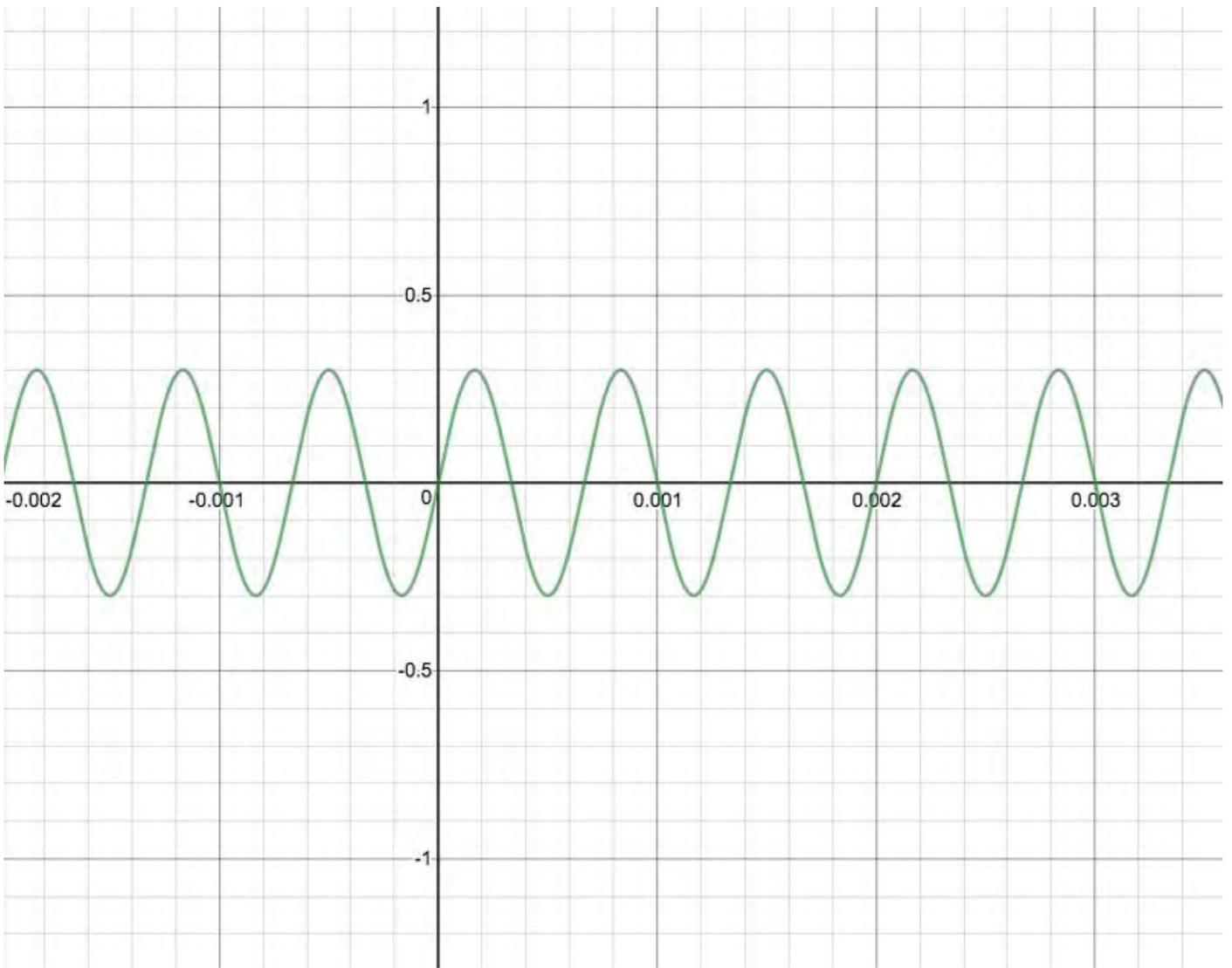


Figura 2

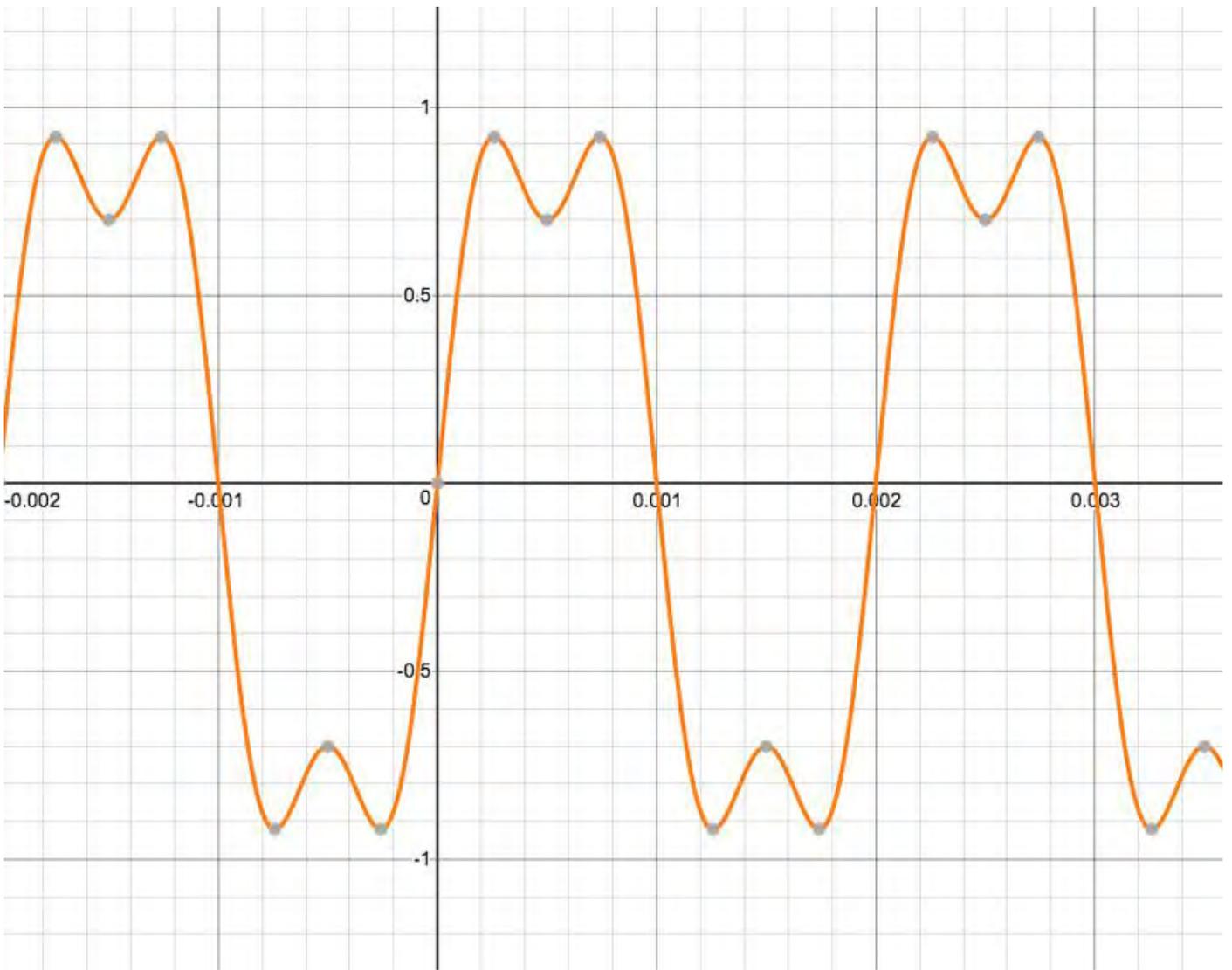


Figura 3

Ejercicios módulo 1

Ejercicio 1

Solución

$$F = 500 \text{ Hz}$$

$$y_1 = 1 * \sin(2 * \pi * f * t)$$

$$y_2 = 0.3 * \sin(2 * \pi * 3 * f * t)$$

$$y_3 = 1 * \sin(2 * \pi * f * t) + 0.3 * \sin(2 * \pi * 3 * f * t)$$

Ejercicios módulo 1

Ejercicio 2

Enunciado

1. ¿Cuánto tiempo tarda un sonido de $f = 100$ Hz en llegar de un punto A a un punto B si los puntos están separados por una distancia de $d = 50$ m y la onda se propaga por el aire a una velocidad de $v = 340$ m/s?
2. ¿Cuánto tiempo tarda un sonido de $f = 200$ Hz en llegar de un punto A a un punto B si los puntos están separados por una distancia de $d = 50$ m y la onda se propaga por el aire a una velocidad de $v = 340$ m/s?
3. ¿Cuánto tiempo tarda un sonido de $f = 100$ Hz en llegar de un punto A a un punto B si los puntos están separados por una distancia de $d = 50$ m y la onda se propaga por el agua a una velocidad de $v = 1.450$ m/s?
4. ¿Cuánto tiempo tarda un sonido de $f = 200$ Hz en llegar de un punto A a un punto B si los puntos están separados por una distancia de $d = 50$ m y la onda se propaga por el agua a una velocidad de $v = 1.450$ m/s?

Ejercicios módulo 1

Ejercicio 2

Solución

1. 0,147 s

2. 0,147 s

3. 0,034 s

4. 0,034 s

Ejercicios módulo 1

Ejercicio 3

Enunciado

1. Calcular la longitud de onda de una onda de $f = 100$ Hz que recorre una distancia de $d = 50$ m si la onda se propaga por el aire a una velocidad de $v = 340$ m/s.
2. Calcular la longitud de onda de una onda de $f = 200$ Hz que recorre una distancia de $d = 50$ m si la onda se propaga por el aire a una velocidad de $v = 340$ m/s.
3. Calcular la longitud de onda de una onda de $f = 100$ Hz que recorre una distancia de $d = 50$ m si la onda se propaga por el agua a una velocidad de $v = 1.450$ m/s.
4. Calcular la longitud de onda de una onda de $f = 200$ Hz que recorre una distancia de $d = 50$ m si la onda se propaga por el agua a una velocidad de $v = 1.450$ m/s.

Ejercicios módulo 1

Ejercicio 3

Solución

1. 3,4 m
2. 1,7 m
3. 14,5 m
4. 7,25 m

Ejercicios módulo 1

Ejercicio 4

Enunciado

Si el oído humano es capaz de percibir frecuencias entre $f = 20$ Hz y $f = 20$ KHz, calcula las longitudes de onda máxima y mínima correspondientes, asumiendo la velocidad de propagación del sonido en el aire de $v = 340$ m/s. ¿Las frecuencias máxima y mínima serán las mismas bajo el agua? ¿Y las longitudes de onda?

Ejercicios módulo 1

Ejercicio 4

Solución

$\lambda_1 = 17 \text{ m}$

$\lambda_2 = 0,017 \text{ m}$

Los límites frecuenciales serán los mismos, pero las longitudes de onda correspondientes serán diferentes, puesto que cambia la velocidad de propagación.

Ejercicios módulo 1

Ejercicio 5

Enunciado

Se dispone de dieciocho archivos de audio organizados tal como muestran la tabla 1 y la tabla 2.

Tabla 1

f = 300 Hz	Duración = 50 ms	Duración = 250 ms	Duración = 1.000 s
Amplitud = 0 dB	M1_e05_0300hz_0050ms_-00dB.w <u>av</u>	M1_e05_0300hz_0250ms_-00dB.w <u>av</u>	M1_e05_0300hz_1000ms_-00dB.w <u>av</u>
Amplitud = -6 dB	M1_e05_0300hz_0050ms_-06dB.w <u>av</u>	M1_e05_0300hz_0250ms_-06dB.w <u>av</u>	M1_e05_0300hz_1000ms_-06dB.w <u>av</u>
Amplitud = -12 dB	M1_e05_0300hz_0050ms_-12dB.w <u>av</u>	M1_e05_0300hz_0250ms_-12dB.w <u>v</u>	M1_e05_0300hz_1000ms_-12dB.w <u>v</u>

Tabla 2

f = 5.000 Hz	Duración = 50 ms	Duración = 250 ms	Duración = 1.000 s
Amplitud = 0 dB	M1_e05_5000hz_0050ms_-00dB.w <u>av</u>	M1_e05_5000hz_0250ms_-00dB.w <u>av</u>	M1_e05_5000hz_1000ms_-00dB.w <u>av</u>
Amplitud = -6 dB	M1_e05_5000hz_0050ms_-06dB.w <u>av</u>	M1_e05_5000hz_0250ms_-06dB.w <u>av</u>	M1_e05_5000hz_1000ms_-06dB.w <u>av</u>
Amplitud = -12 dB	M1_e05_5000hz_0050ms_-12dB.w <u>v</u>	M1_e05_5000hz_0250ms_-12dB.w <u>v</u>	M1_e05_5000hz_1000ms_-12dB.w <u>v</u>

Asumiendo que el comportamiento del oído es diferente para cada persona y que el equipo de reproducción (ordenador, tarjeta de sonido, auriculares, etc.) puede afectar los resultados, comenta si las diferentes situaciones de duración y amplitud provocan que percibas un *pitch* diferente en el primer caso ($f = 300$ Hz) y en el segundo ($f = 5.000$ Hz). ¿Es esto acorde con la teoría?

- **Nota 1:** En los extremos de la tabla se debería percibir alguna diferencia, aunque bastante sutil.
- **Nota 2:** En las posiciones intermedias de la tabla se aprecian cambios muy sutiles (es decir, puede ser que un oído no entrenado no los perciba).

Ejercicios módulo 1

Ejercicio 5

Solución

Los cambios que más se notan son:

[M1_e05_0300hz_0050ms_-00dB.wav](#)

suenan más agudo que:

[M1_e05_0300hz_1000ms_-12dB.wav](#)

[M1_e05_5000hz_0050ms_-00dB.wav](#)

suenan más grave que:

[M1_e05_5000hz_1000ms_-12dB.wav](#)

Ejercicios módulo 1

Ejercicio 6

Enunciado

El archivo:

[M1_e06_440hz_400mV_000deg.wav](#)

contiene un tono puro de $f = 440$ Hz y amplitud $V = 0,4$ V (o lo que es lo mismo, $V = 400$ mV).

Por otro lado, el archivo:

[M1_e06_440hz_400mV_180deg.wav](#)

contiene el mismo tono puro pero invirtiendo la fase. Dibuja el sonido resultante de sumar:

[M1_e06_440hz_400mV_000deg.wav](#)

y

[M1_e06_440hz_400mV_000deg.wav](#)

[M1_e06_440hz_400mV_000deg.wav](#)

y

[M1_e06_440hz_400mV_180deg.wav](#)

Ejercicios módulo 1

Ejercicio 6

Solución

Las figuras 4 y 5 muestran el resultado de las operaciones, respectivamente.

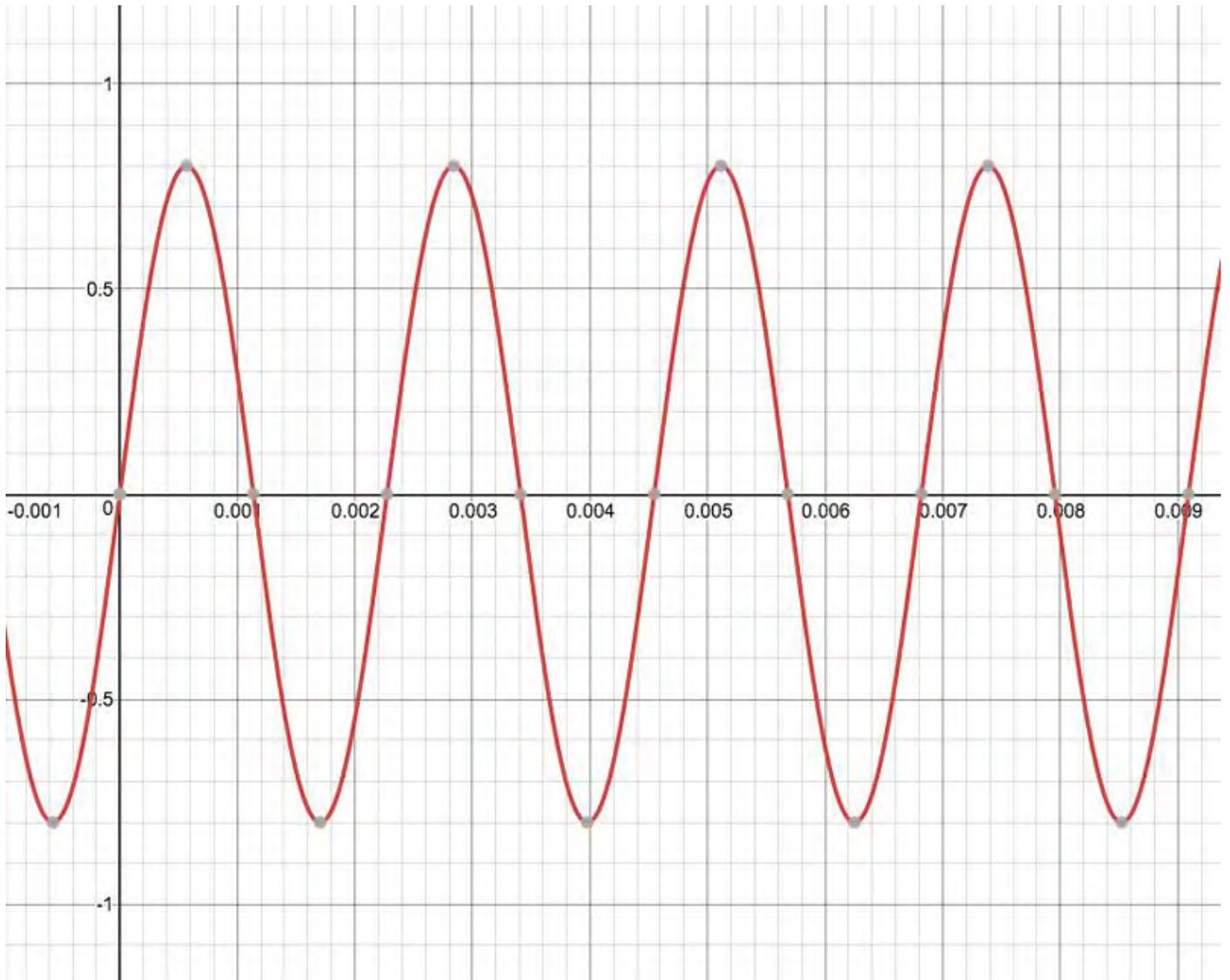


Figura 4

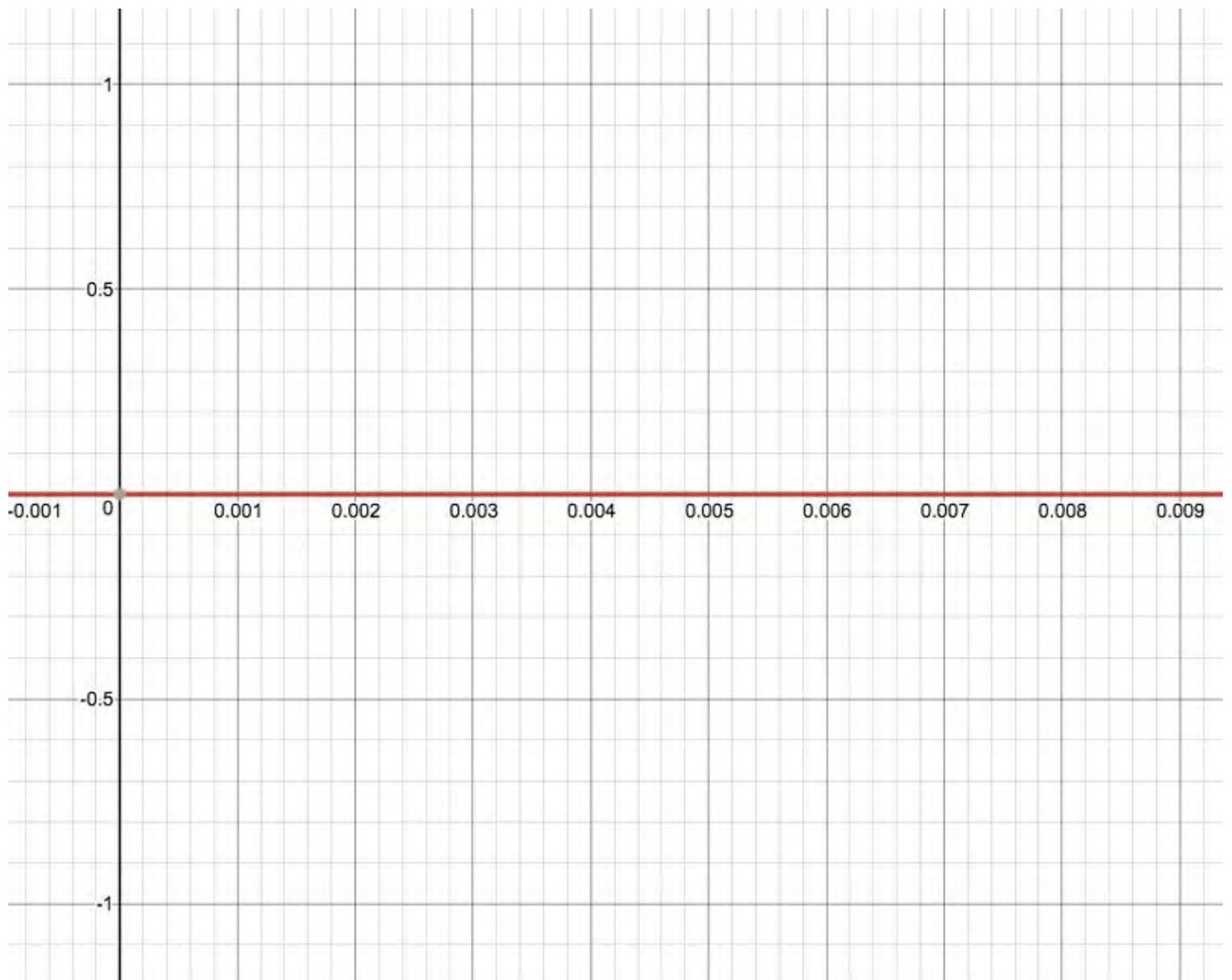


Figura 5

Los respectivos resultados sonoros se puede escuchar en los archivos:

[M1_e06_440hz_constr.wav](#)

[M1_e06_440hz_destr.wav](#)

Ejercicios módulo 1

Ejercicio 7

Enunciado

Se dispone de los archivos de audio:

[M1_e07_0998hz_400mv_5s.wav](#)

[M1_e07_1000hz_400mv_5s.wav](#)

[M1_e07_1002hz_400mv_5s.wav](#)

que contienen un tono puro de amplitud $V = 0,4$ V, duración $d = 5$ s y frecuencias de $f_1 = 998$ Hz, $f_2 = 1.000$ Hz y $f_3 = 1.002$ Hz, respectivamente.

¿Se percibe alguna diferencia de amplitud entre ellos? ¿Y de *pitch*?

Dibuja el sonido resultante de sumar:

[M1_e07_0998hz_400mv_5s.wav](#)

y

[M1_e07_1000hz_400mv_5s.wav](#)

[M1_e07_0998hz_400mv_5s.wav](#)

y

[M1_e07_1002hz_400mv_5s.wav](#)

[M1_e07_1000hz_400mv_5s.wav](#)

y

[M1_e07_1002hz_400mv_5s.wav](#)

Ejercicios módulo 1

Ejercicio 7

Solución

No hay ninguna diferencia de amplitud, pero sí que puede llegar a percibirse, en oídos muy entrenados, una ligera diferencia de *pitch*.

Las figuras 6, 7 y 8 muestran el resultado de las sumas propuestas.

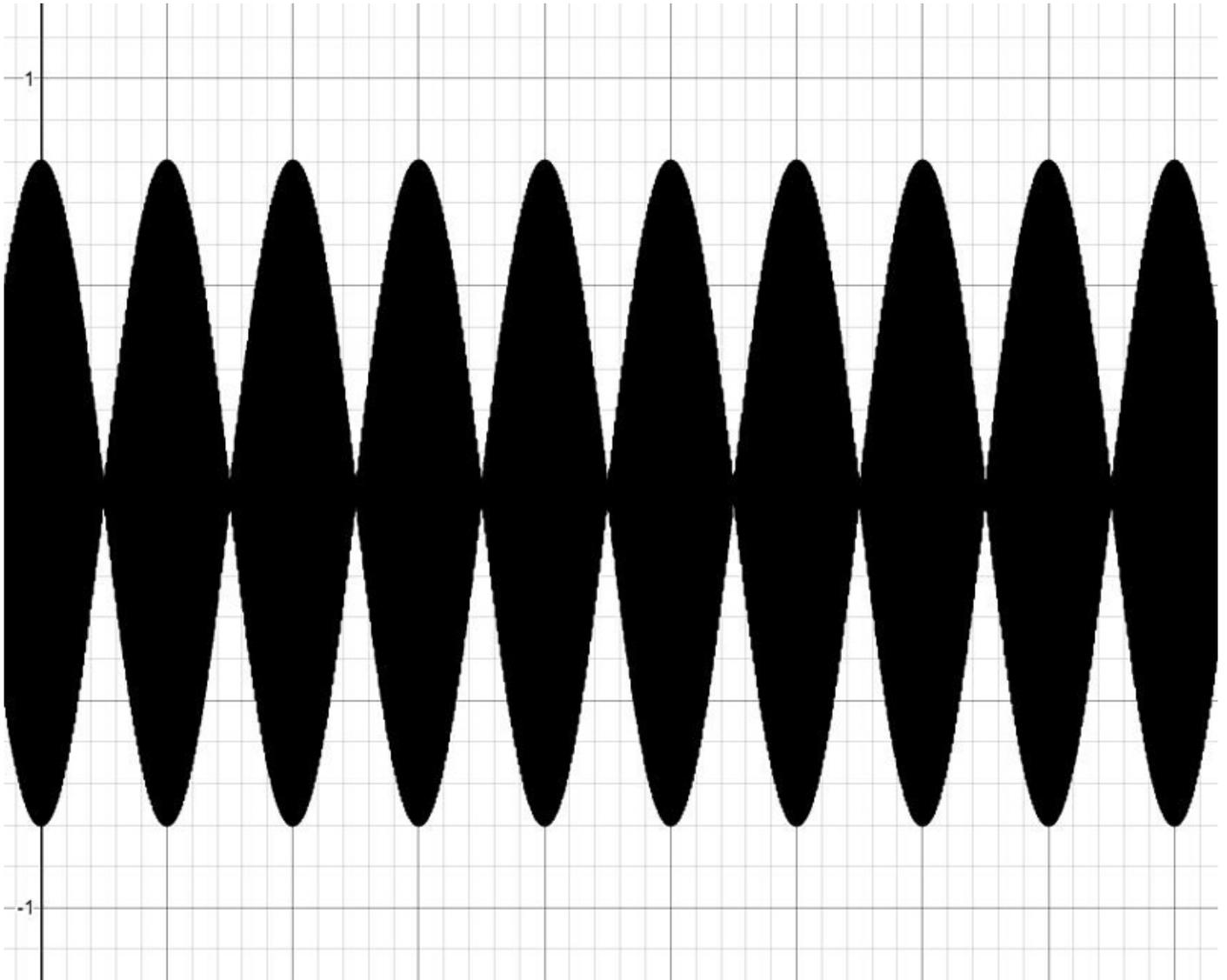


Figura 6

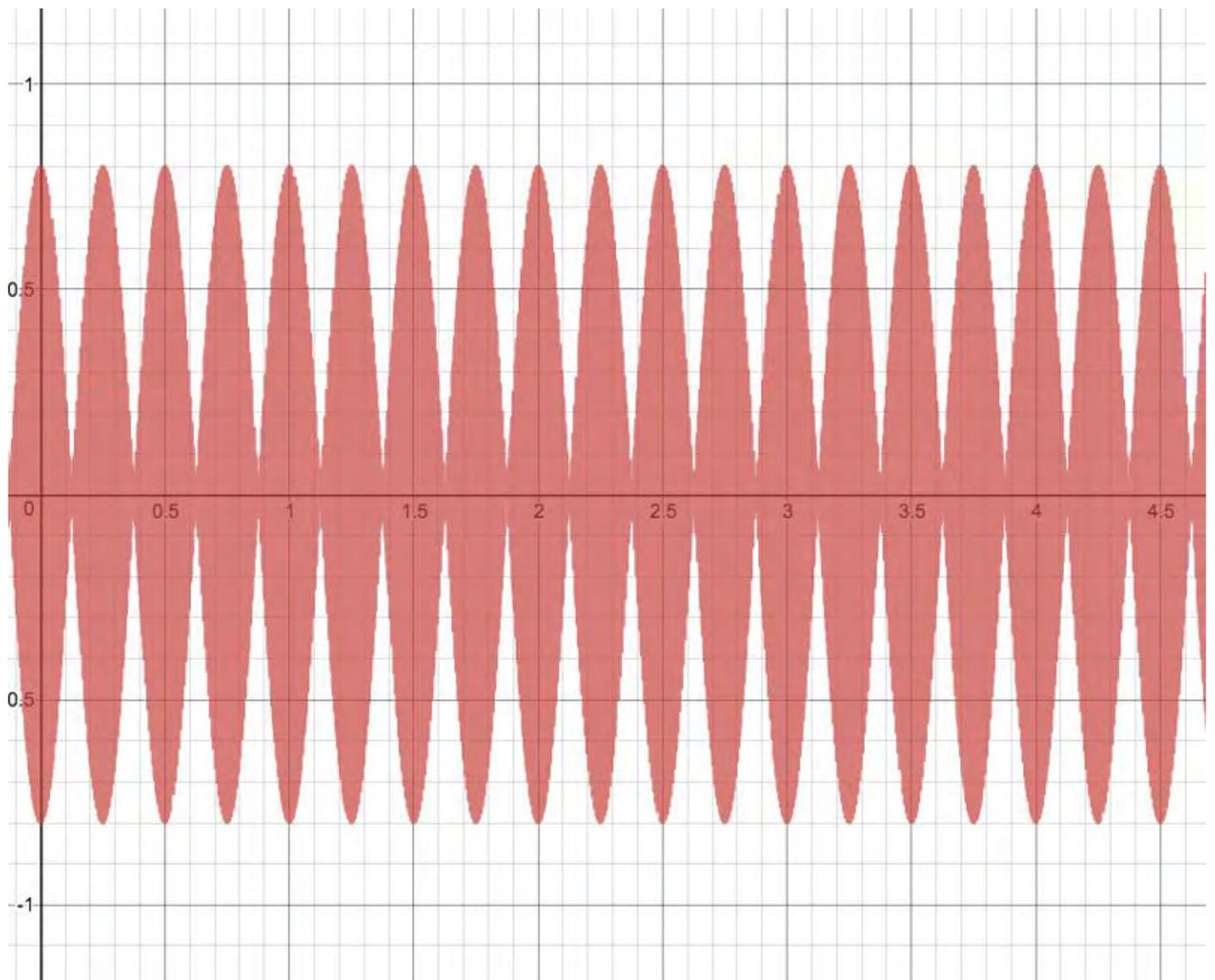


Figura 7

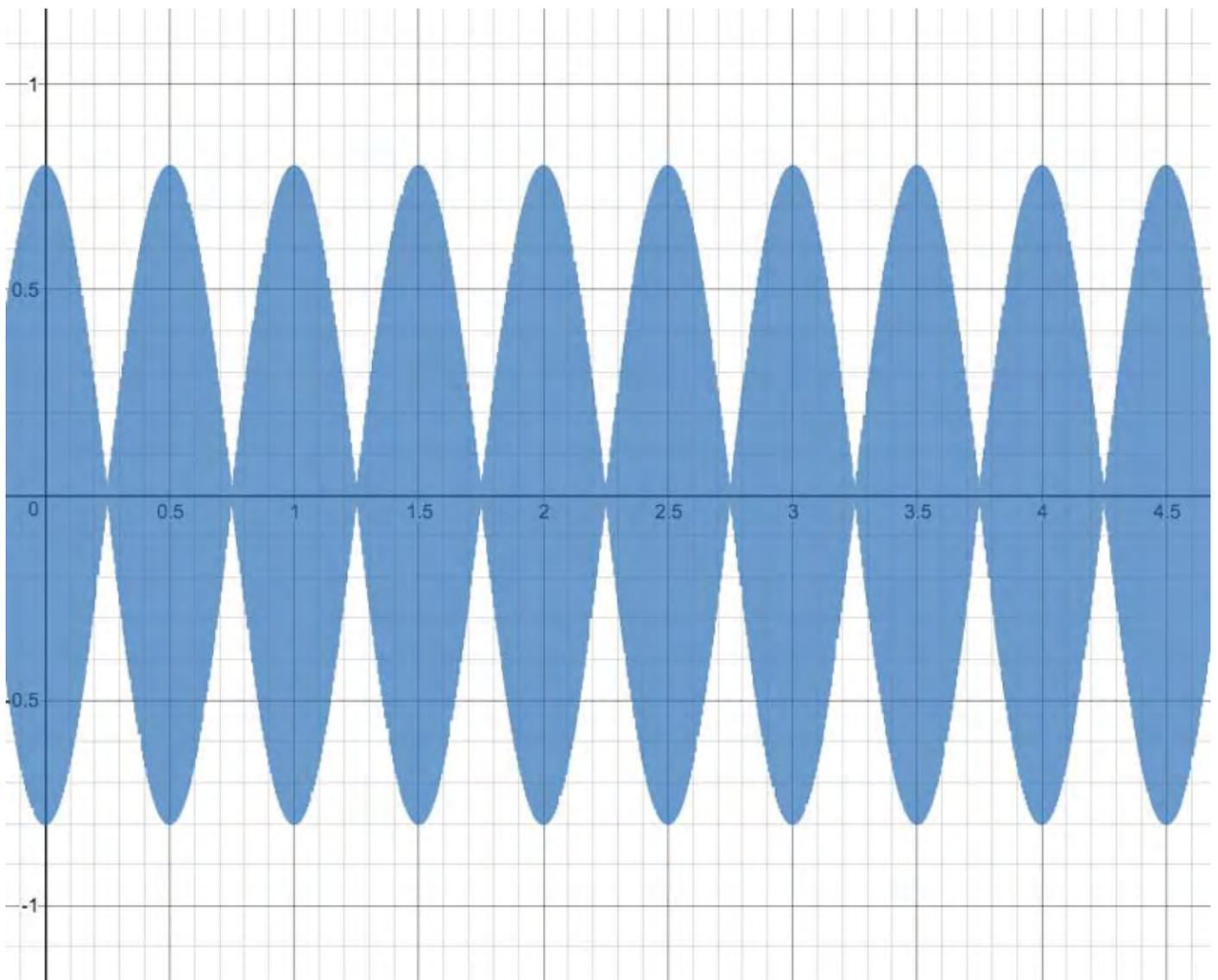


Figura 8

Los respectivos resultados sonoros se pueden escuchar en los archivos:

[M1_e07_0998_1000hz_5s.wav](#)

[M1_e07_0998_1002hz_5s.wav](#)

[M1_e07_1000_1002hz_5s.wav](#)

Ejercicios módulo 1

Ejercicio 8

Enunciado

Se dispone de los ficheros de audio:

[M1_e08_0660hz_400mv_5s.wav](#)

[M1_e08_0680hz_400mv_5s.wav](#)

que contienen un tono puro de amplitud $V = 0,4$ V, duración $d = 5$ s y frecuencias de $f_1 = 660$ Hz y $f_2 = 680$ Hz, respectivamente.

¿Se percibe alguna diferencia de amplitud entre ellos? ¿Y de *pitch*?

Razona cuál es el resultado sonoro de sumar los dos sonidos.

Ejercicios módulo 1

Ejercicio 8

Solución

La amplitud de los dos sonidos es la misma, pero se percibe una diferencia de *pitch* entre ellos. Ahora bien, al escuchar la suma de los dos, se percibe un sonido con una alta rugosidad fruto de tener unos batimientos tan rápidos que no se pueden percibir como tales. Aun así, las dos notas empiezan a diferenciarse, a diferencia de lo que sucede en el ejercicio 7.

El resultado sonoro se puede escuchar en el archivo:

[M1_e08_0680_0680hz_400mv_5s.wav](#)

Ejercicios módulo 1

Ejercicio 9

Enunciado

El archivo:

[M1_e09_gaussian+sweep_30s.wav](#)

contiene la mezcla de un ruido gaussiano centrado en la frecuencia de $f_c = 1.000$ Hz y un tono puro que evoluciona desde una frecuencia $f_1 = 220$ Hz hasta $f_2 = 2.200$ Hz, de manera exponencial. ¿Hay algún momento en que sea difícil percibir el tono puro? ¿Porqué?

Ejercicios módulo 1

Ejercicio 9

Solución

El efecto de enmascaramiento puede provocar que, debido a frecuencias cercanas a los 1.000 Hz del ruido, el tono puro quede enmascarado, hasta que la frecuencia aumente lo suficiente como para salir de la zona de enmascaramiento.

Ejercicios módulo 1

Ejercicio 10

Enunciado

Los archivos:

[M1_e10_0500hz_400mv_5s_sine.wav](#)

[M1_e10_0500hz_400mv_5s_square.wav](#)

[M1_e10_0500hz_400mv_5s_saw.wav](#)

contienen señales de $f = 500$ Hz, amplitud de $V = 400$ mV y formas de onda sinusoidales, cuadradas y en diente de sierra, respectivamente.

¿Escucharemos el mismo *pitch* en los tres casos? ¿Y el mismo timbre? Dibujar cómo es el espectro de estas tres señales.

Ejercicios módulo 1

Ejercicio 10

Solución

Dado que la frecuencia fundamental de las tres señales es la misma, el *pitch* percibido será siempre el mismo. Aun así, el timbre diferirá, puesto que el contenido en armónicos es diferente. La señal sinusoidal es la más pobre, mientras que las señales triangulares y la señal en diente de sierra son ricas en armónicos. La señal cuadrada sólo contiene armónicos noes, mientras que la señal en diente de sierra los contiene todos, por eso suena más rica.

Las figuras 9, 10 y 11 muestran el espectro de las tres señales.

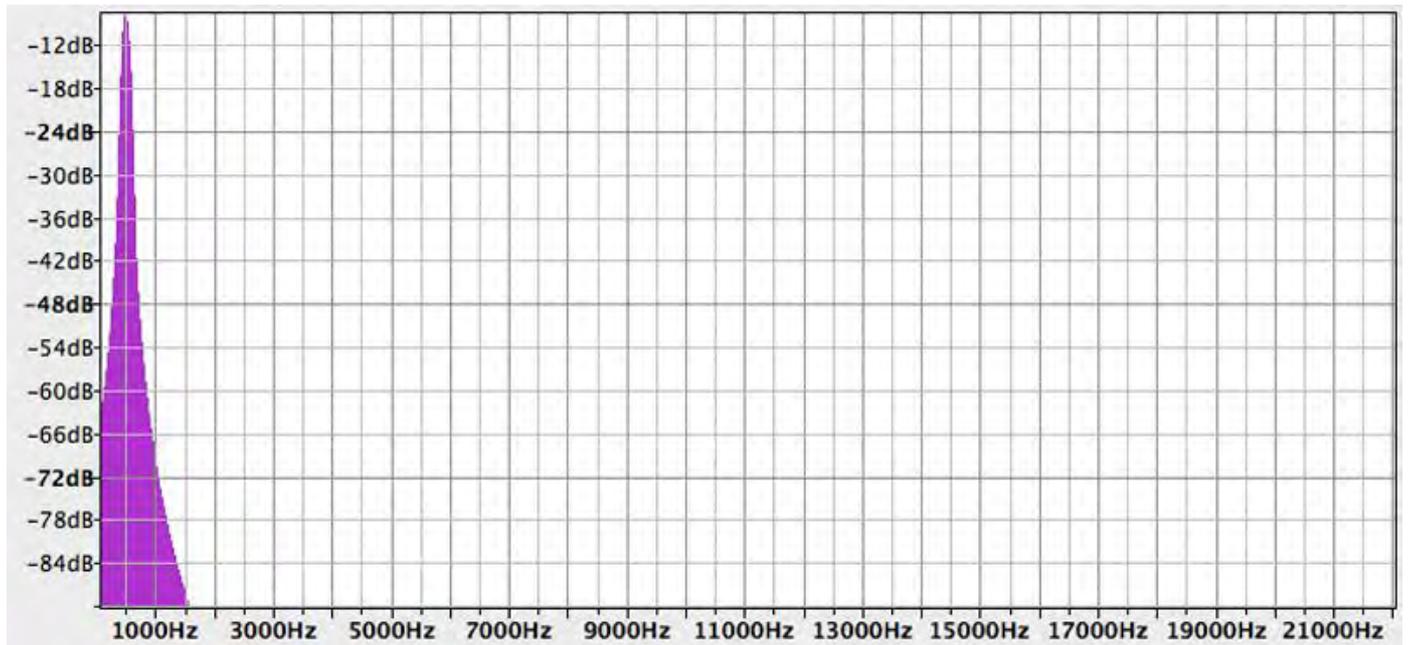


Figura 9

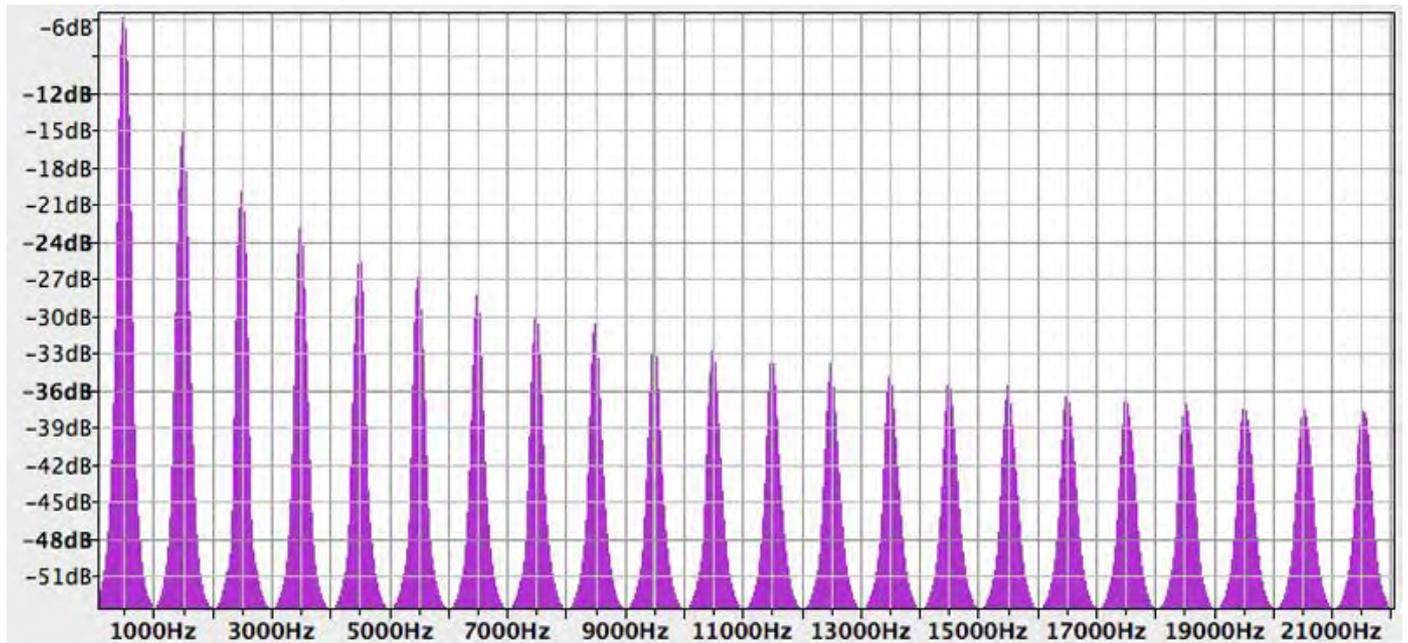


Figura 10

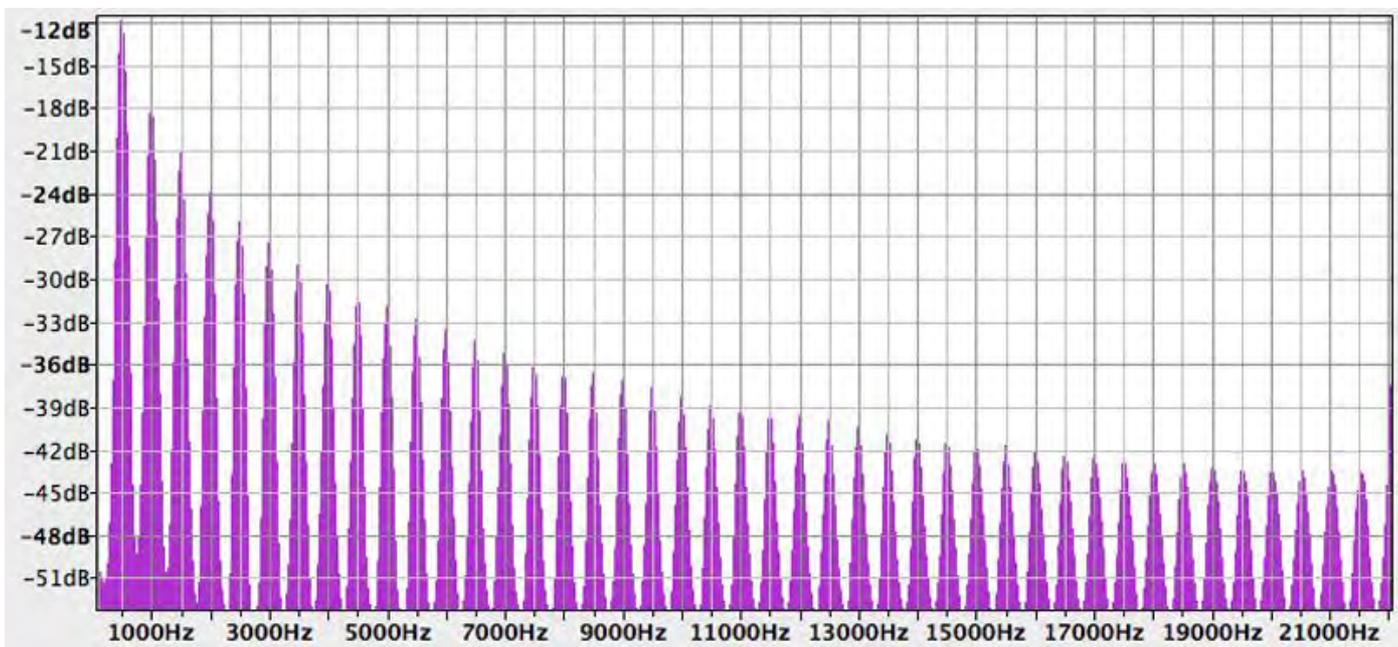


Figura 11

Ejercicios módulo 1

Ejercicio 11

Enunciado

Dibujar el sonograma de un barrido frecuencial lineal con inicio en la frecuencia de $f_1 = 100$ Hz, final en la frecuencia de 1.000 Hz, amplitud $V = 400$ mV y duración = 10 s. ¿Cuál será la frecuencia correspondiente al instante $t = 4,5$ s?

Ejercicios módulo 1

Ejercicio 11

Solución

El sonograma es el que se muestra en la figura 12.

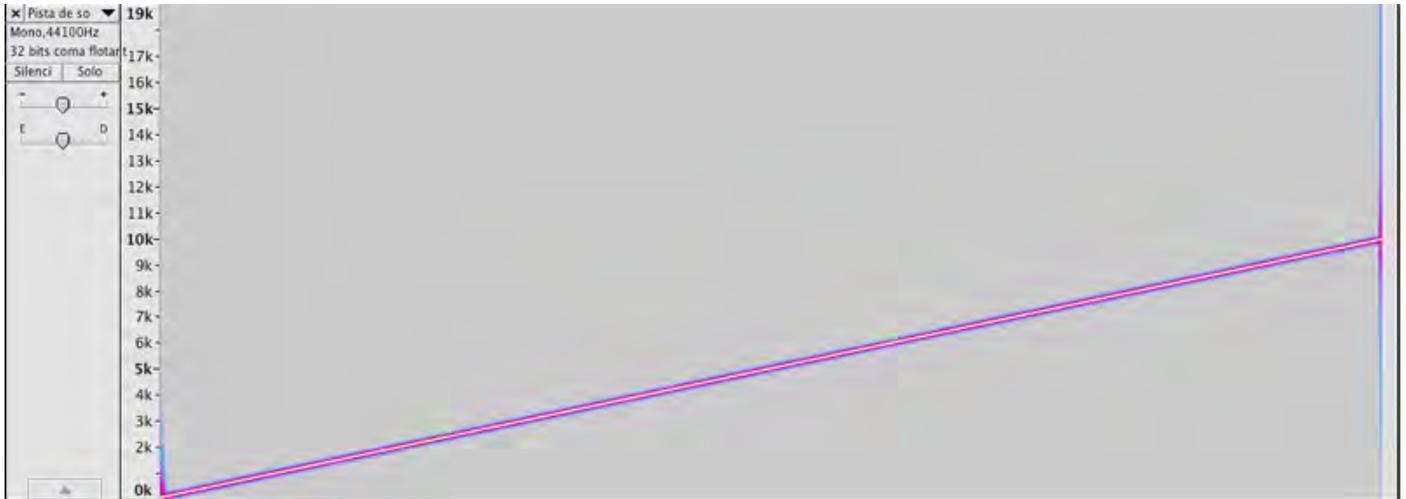


Figura 12

Para saber la frecuencia correspondiente al instante $t = 4,5$ s hay que buscar la ecuación de la recta. A partir de los dos puntos conocidos $(0 \text{ s}, 100 \text{ Hz})$ y $(100 \text{ s}, 10.000 \text{ Hz})$ se puede encontrar la ecuación de la recta $f = 990 t + 100$. Con esto, para $t = 4,5$ s se obtiene una $f = 4.555 \text{ Hz}$.

Ejercicios resueltos módulo 2

Ejercicio 1

Enunciado

Descarga el archivo de audio <http://freesound.org/people/schulle4u/sounds/70143/> de la página Freesound (deberás darte de alta en el portal si todavía no lo has hecho) y usa este sonido para:

1. Generar un sonido de exactamente 14 segundos de duración.
2. Este sonido debe contener los números 0-9, enumerados de mayor a menor. Es decir, usando herramientas de edición básicas debéis usar el inicio del audio original –el fragmento que contiene los números– y convertir el audio original «one, two, three...» en «nine, eight... zero».
3. El archivo resultante no debe contener clics ni ningún otro tipo de incidencia que muestre que es un audio editado.

Nota: Aunque estos ejercicios hacen referencia principalmente a los conceptos estudiados en el módulo 2, algunos de los apartados requerirán que el estudiante también consulte el módulo 6 para resolverlos.

Ejercicios resueltos módulo 2

Ejercicio 1

Solución

1. Como se puede observar, los primeros 14 segundos del archivo ya contienen los números que pide el enunciado. Así que sólo tenemos que seleccionar los primeros 14 segundos del sonido y usar Audition –o Audacity– para recortar este fragmento ([Ctrl] + [T]).

Ahora, para facilitar el trabajo de ir cortando los fragmentos y pegarlos en un nuevo archivo, creamos un marcador entre cada número.

Un concepto muy importante que siempre debe tenerse presente es que los puntos del sonido digitalizado en los que la amplitud es cero son los mejores lugares para realizar selecciones. Las selecciones que empiezan y acaban en intersecciones cero reducen la posibilidad de que se produzcan clics u otros ruidos no deseados.

Así que empezamos situando el cursor entre el número «zero» y el «one», y con cualquiera de las opciones del menú «Intersecciones Cero», que buscan valores de amplitud cero a la derecha e izquierda del cursor, buscar uno de estos puntos.

Una vez que el cursor esté sobre una muestra de amplitud 0, pulsamos la tecla M y creamos el marcador.

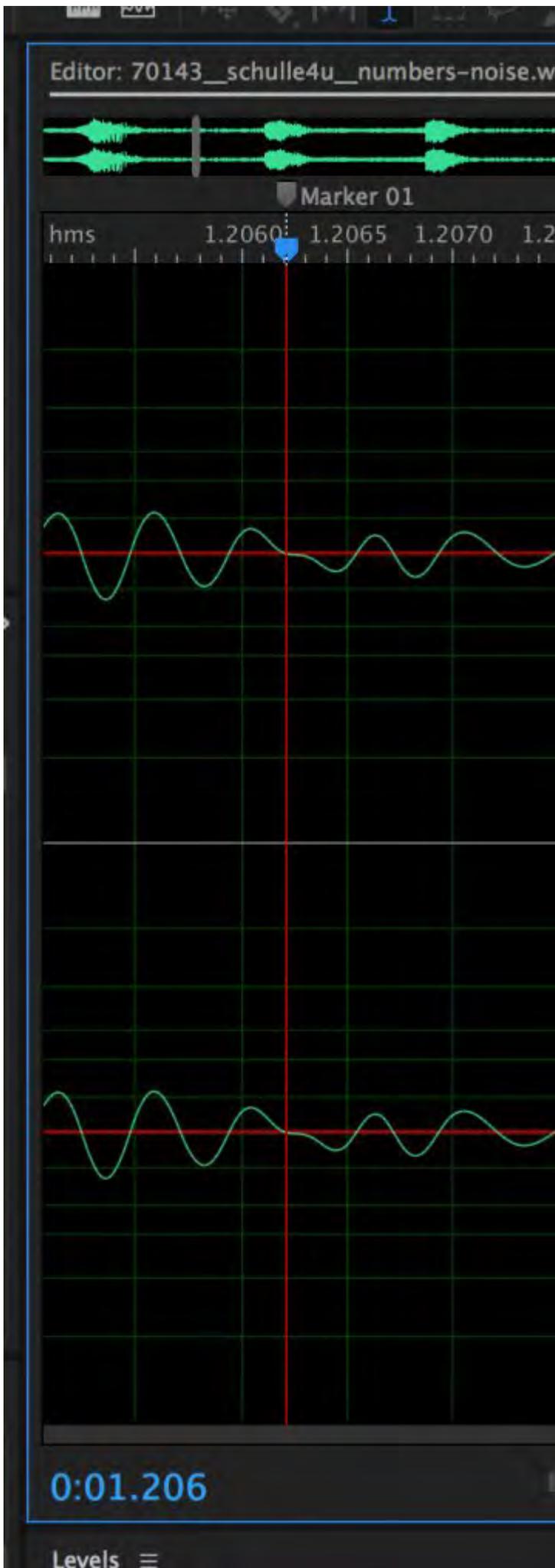


Figura 13. Creación de un marcador en un punto de intersección cero.

Repetim els mateixos passos fins a crear un marcador entre cada nombre, tal com es mostra en la figura 14.



Figura 14

2. Una vez creados los marcadores, el siguiente paso es muy sencillo: solo debemos crear un nuevo archivo e ir pegando los fragmentos, pero en orden inverso. Es decir, seleccionamos el fragmento de audio que hay entre el último marcador y el final y pulsamos [Ctrl] + [X] o «cortar». Vamos al nuevo archivo y pegamos este fragmento en el inicio.

Volvemos a seleccionar el último fragmento del sonido original, lo cortamos y lo pegamos en el nuevo archivo, y así hasta completar todos los fragmentos.

3. Una de las ventajas de haber trabajado con los marcadores es que ahora sabemos exactamente dónde hemos hecho los cortes y podemos revisar mejor si se ha generado algún clic. Cortar y pegar los fragmentos en muestras de valor 0 previene la creación de clics, pero no asegura completamente que no se hayan creado.

Para empezar, tenemos que crear un *fade in* al inicio del sonido y un *fade out* al final para asegurarnos de que el sonido no tendrá ningún problema en los extremos. Después, revisamos las zonas de corte para ver si hay problemas con alguno de los cortes.

Si en el primer paso hemos seleccionado los 14 primeros segundos del sonido original, muy probablemente tendremos un clic entre el «nine» y el «eight», puesto que la muestra final del fragmento «nine» no tiene valor cero.



Figura 15

Una herramienta muy útil de Audition se encuentra en «Favoritos» > «Limpieza automática». Seleccionamos el fragmento donde está el problema, aplicamos este efecto y el problema parece solucionado. Podemos consultar la teoría para ver otros procedimientos para fijar clics y otros tipos de ruidos puntuales.



Figura 16

En caso de usar algún efecto de reparación automática de clics, es mejor aplicarlo únicamente sobre la zona de corte; así podemos comprobar si realmente ha solucionado el problema y también nos aseguramos de que el efecto no haya modificado otras partes del sonido que pueda haber detectado como clic sin realmente serlo (puede pasar).

Ejercicios resueltos módulo 2

Ejercicio 2

Enunciado

Con un micrófono, un técnico de sonido ha grabado el sonido de un bajo y el bombo de una batería con el nombre «gravacio.aif». De todos modos, como puedes escuchar, el micrófono también ha captado, a un nivel más bajo, el sonido agudo del *hi-hat*.

[gravacio.aif](#)

1. Si el técnico de sonido quisiera eliminar el sonido del *hi-hat* de la grabación y dejar solo el sonido del bajo y el del bombo sin perder calidad, ¿qué filtros o efectos debería que aplicar?
2. Intenta efectuar esta operación con los filtros y efectos de tu editor de audio, y guarda el resultado con el nombre de «gravacio2.wav». Incluye una captura de pantalla de los filtros o efectos escogidos y comenta los parámetros utilizados.
3. Con las herramientas del editor o los editores de audio que hayas usado, haz un análisis espectral de los dos sonidos y compara los resultados. Incluye capturas de pantalla de los análisis para complementar tus razonamientos.

Ejercicios resueltos módulo 2

Ejercicio 2

Solución

1. Este ejercicio está pensado para que el alumno ponga en práctica alguno de los procesos más básicos que se tratan en la teoría.

Un filtro de paso bajo es uno de estos procesos básicos y puede ser una buena opción para resolver este ejercicio. El sonido del *hi-hat* es un sonido agudo; por tanto, un filtro de paso bajo nos puede ayudar a eliminar las frecuencias altas del *hi-hat* sin modificar las frecuencias de los sonidos más graves: bajo y bombo.

Una puerta de ruido también podría ser un efecto a considerar, puesto que permite silenciar las muestras que se encuentran por debajo de un determinado valor umbral. En nuestro caso, el sonido del *hi-hat* parece tener una intensidad menor que el bombo y el bajo, por lo que esta opción sería una opción a estudiar.

Una opción muy similar a la puerta de ruido –pero menos radical– sería un expansor. Seguramente no eliminaría por completo el sonido del *hi-hat*, pero, por otro lado, respetaría más las frecuencias agudas del bombo y el bajo.

Estos efectos pueden aplicarse por separado o de forma conjunta. También sería posible aplicar otros procesos más complicados. De todos modos, en el punto 2 aplicaremos la opción más sencilla –pero válida–: el filtro de paso bajo.

2. Al hacer las primeras pruebas con el filtro de paso bajo detectamos –y este es un concepto importante– que no es posible eliminar completamente el sonido del *hi-hat* sin modificar también –ni que sea un poco– el sonido del bombo y el bajo.

Cualquier sonido que encontramos en la naturaleza tiene infinitas frecuencias. Si, por ejemplo, percibimos el sonido del bombo como un sonido grave es porque este sonido tiene más energía en las frecuencias graves que en las agudas, pero esto no quiere decir que no tenga también frecuencias agudas.

Así que el bombo, el bajo y el *hi-hat* comparten algunas frecuencias y, al eliminar las frecuencias del *hi-hat*, también eliminaremos frecuencias del bombo y del bajo.

Para aplicar el filtro de paso bajo podemos usar la herramienta «Low Pass Filter» de Audacity. Es un efecto muy sencillo en el que simplemente debemos configurar el *Cutoff* o frecuencia de corte, y el *Rolloff* o pendiente del filtro.

La frecuencia de corte determina la frecuencia a la que se empieza a atenuar el sonido, y el *Rolloff* controla la pendiente de esta atenuación. Cuanto mayor sea el valor *Rolloff*, más acentuada será la pendiente de atenuación.

Haciendo pruebas con diferentes configuraciones, podemos llegar a la conclusión que con *Rolloff* = 36 dB y *Cutoff* = 4.100 Hz conseguimos eliminar prácticamente el sonido del *hi-hat* sin cortar del todo las frecuencias más altas de los otros sonidos. Se puede eliminar completamente el sonido del *hi-hat* con valores más pequeños de *Cutoff*, pero entonces veríamos comprometida la calidad de los otros sonidos, puesto que también estaríamos filtrando muchas de sus frecuencias.

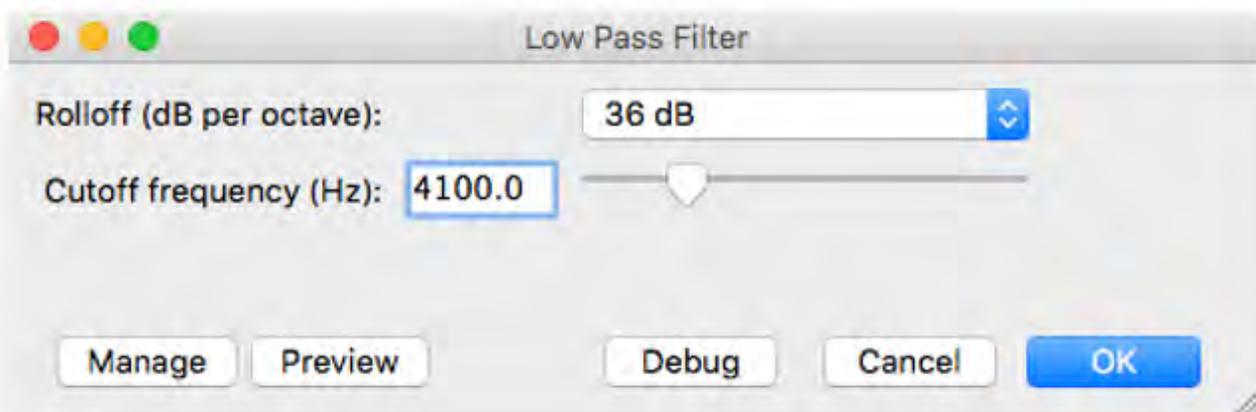


Figura 17

3. En este apartado es importante remarcar que no se debe confundir el concepto *forma de onda* con el concepto *espectrograma*: son dos formas distintas de representar visualmente un sonido.

Para ver el análisis espectral de un sonido en Audacity, debemos hacer clic en «Analyze» > «Plot Spectrum».

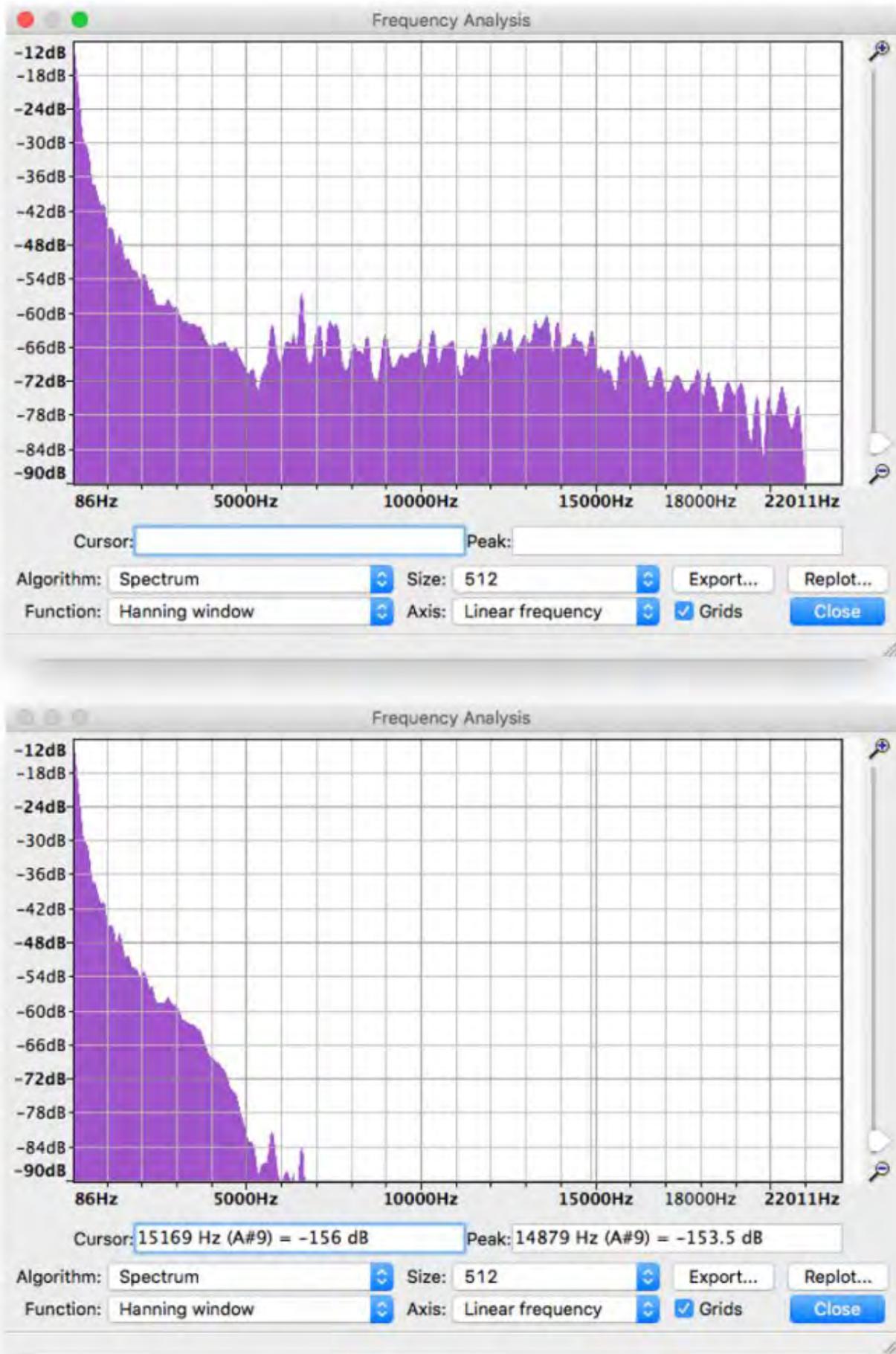


Figura 18. Comparación de los espectros.

En la primera figura tenemos el análisis espectral del sonido original, mientras que en la segunda imagen tenemos el análisis espectral del sonido filtrado.

Como se puede observar en el eje horizontal (que muestra las frecuencias presentes en el sonido), el filtro de paso bajo ha dejado pasar todas las frecuencias que se encontraban por debajo de la frecuencia de corte (4.100), y ha ido atenuando progresivamente las frecuencias que se encontraban por encima de la frecuencia de corte hasta filtrarlas completamente.

Ejercicios resueltos módulo 2

Ejercicio 3

Enunciado

En los estudios de grabación, es típico encontrarse con grupos que tienen muchas dificultades para tocar correctamente sus instrumentos. El siguiente audio sería una simulación de la grabación de un batería que tiene dificultades para tocar su instrumento a un nivel constante.

[bateria_gravacio.wav](#)

Si abris el archivo con un editor de audio y observáis la forma de onda de la grabación, veréis que algunos golpes de percusión tienen un nivel mucho más alto que otros.

1. ¿Qué operación u operaciones deberá efectuar el técnico de sonido para nivelar los sonidos de la batería?
2. Intenta efectuar esta operación con los filtros y efectos de tu editor de audio y guarda el resultado con el nombre de «bateriaCorregida.wav».

Ejercicios resueltos módulo 2

Ejercicio 3

Solución

1. Un compresor es un efecto que sirve para reducir el rango dinámico de una señal. Un compresor típico reduce los valores que se encuentran por encima de un valor máximo, lo que hace que la diferencia entre los valores máximos y mínimos se reduzca.

Este sería el efecto más adecuado para intentar nivelar un poco los sonidos de la batería. Aunque también es posible usar un limitador, sería una opción demasiado radical. Debemos pensar que tampoco nos interesa que todos los golpes de batería sean exactamente iguales y perfectos, puesto que entonces quedaría un sonido poco natural.

Conviene anotar que el audio del ejercicio es un pequeño fragmento de la grabación. Por lo tanto, sería realmente costoso para el técnico de sonido corregir manualmente los 45 o 50 minutos de batería que puede haber en un disco.

Otro punto a tener en consideración al modificar manualmente los sonidos de la batería es que aplicar un efecto en un fragmento puede generar clics. En la figura 19 se puede ver cómo se genera un clic con el programa Audacity al aplicar un efecto «Normalización» sobre un fragmento del audio. En cambio, Audition aplica de forma progresiva los efectos a los extremos de los fragmentos, lo que impide que se generen clics.

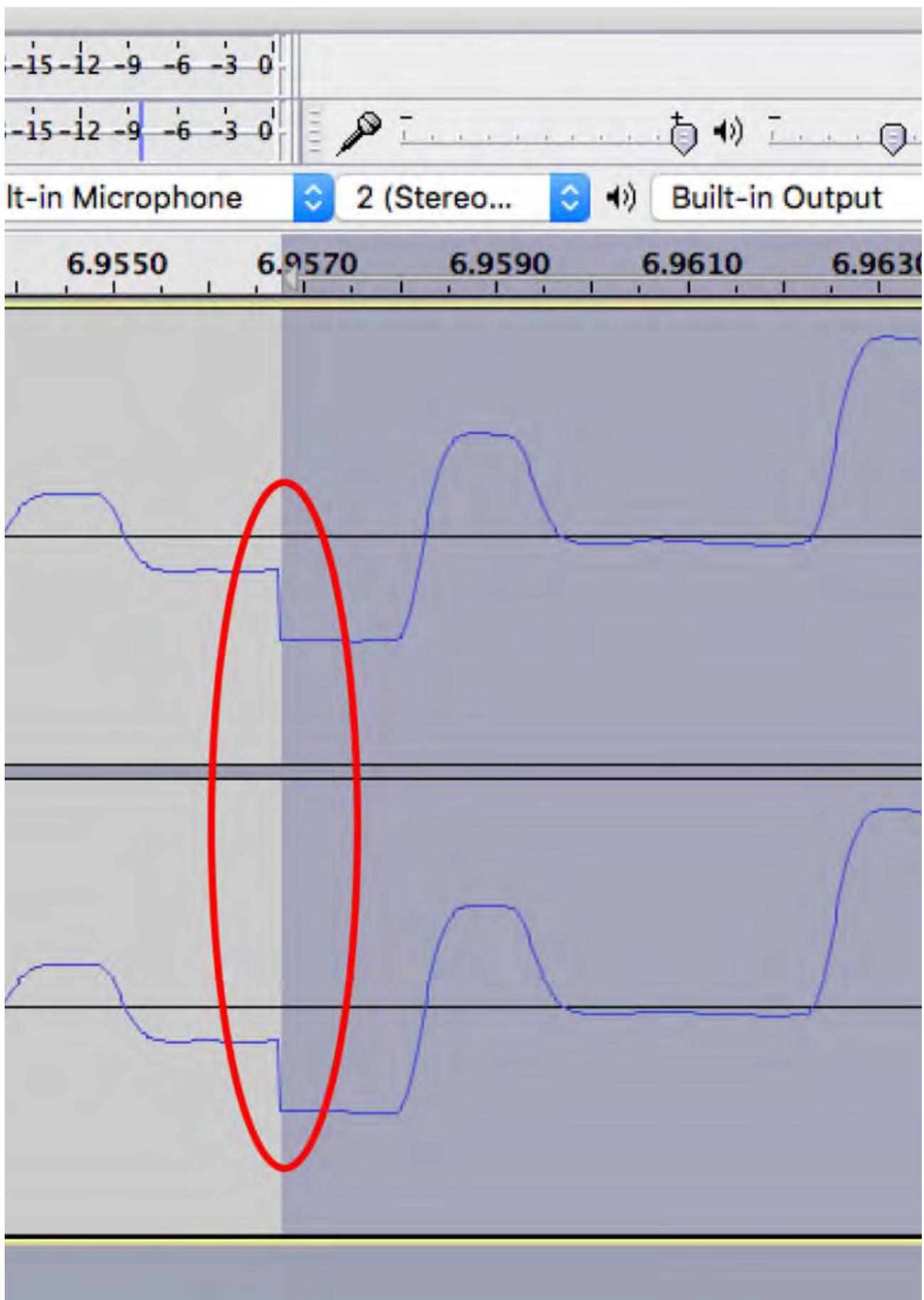


Figura 19. Clic generado en Audacity al aplicar una «Normalización» a un fragmento de audio.

2. Para resolver este ejercicio usaremos el compresor de una banda de Audition con los siguientes parámetros:

- **Threshold (umbral):** -6 dB. Es decir, el compresor solo trabajará cuando la amplitud del sonido sea, como mínimo, de -6 dB.
- **Ratio (proporción):** 10:1. Indica que un cambio de 10 dB en la entrada produce un cambio de 1 dB en la salida.

- **Attack (ataque):** 1 ms. El ataque es el tiempo que tarda el compresor en completar la reducción de ganancia. 1 ms seguramente sea un ataque demasiado rápido, pero el resultado visual final será más fácil de entender para el alumno.
- **Release (decaimiento):** 200 ms. Es el tiempo que tarda el compresor en dejar de comprimir tras detectar que la amplitud del sonido está por debajo del umbral.
- **Output Gain (ganancia de salida):** 3 dB. El compresor reduce los valores que se encuentran por encima de un valor máximo. Por tanto, en muchas situaciones es conveniente darle ganancia a todo el sonido para compensar la reducción de nivel aplicada.

[bateriaCorregida.wav](#)

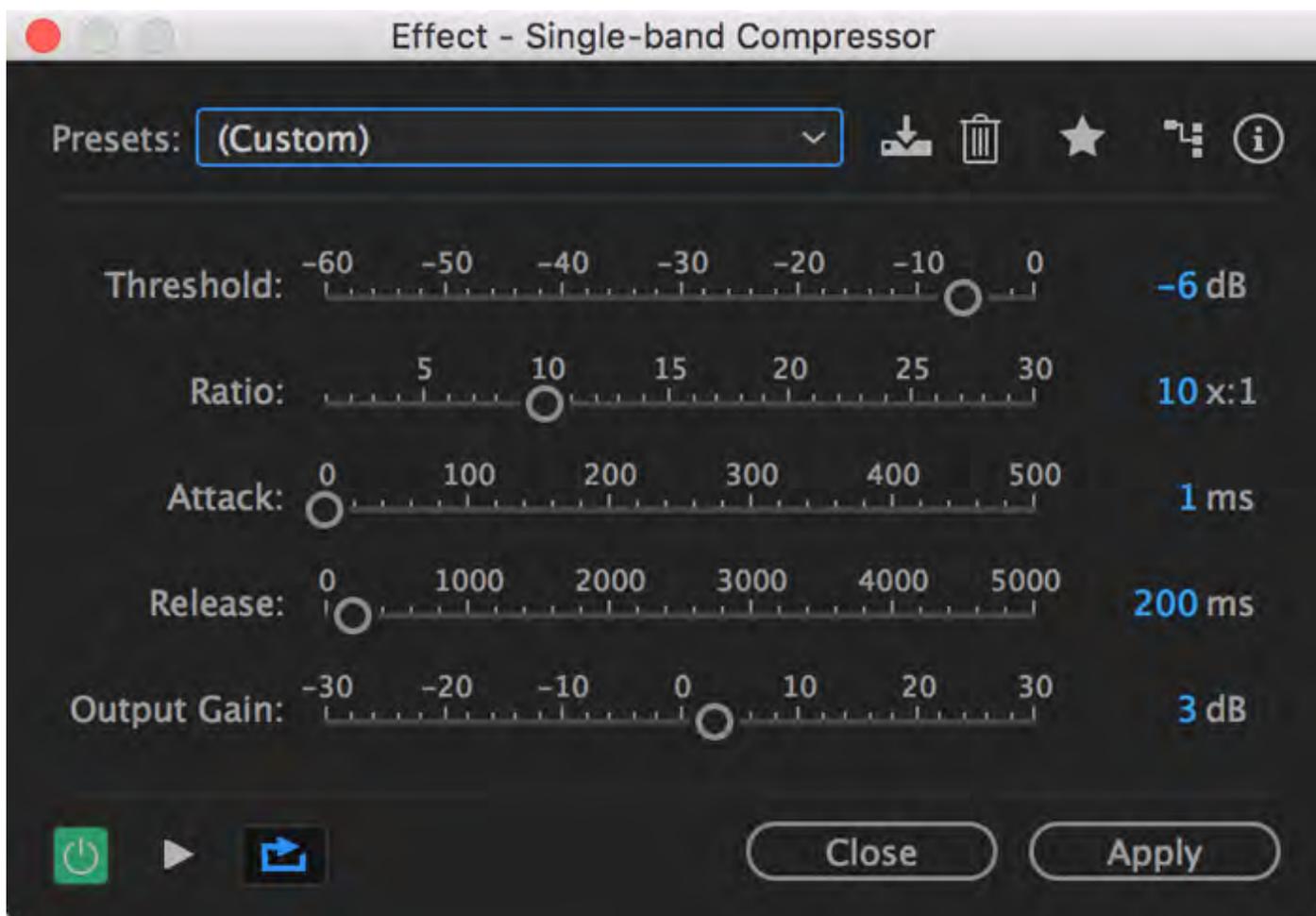


Figura 20. El compresor de una banda de Audition.

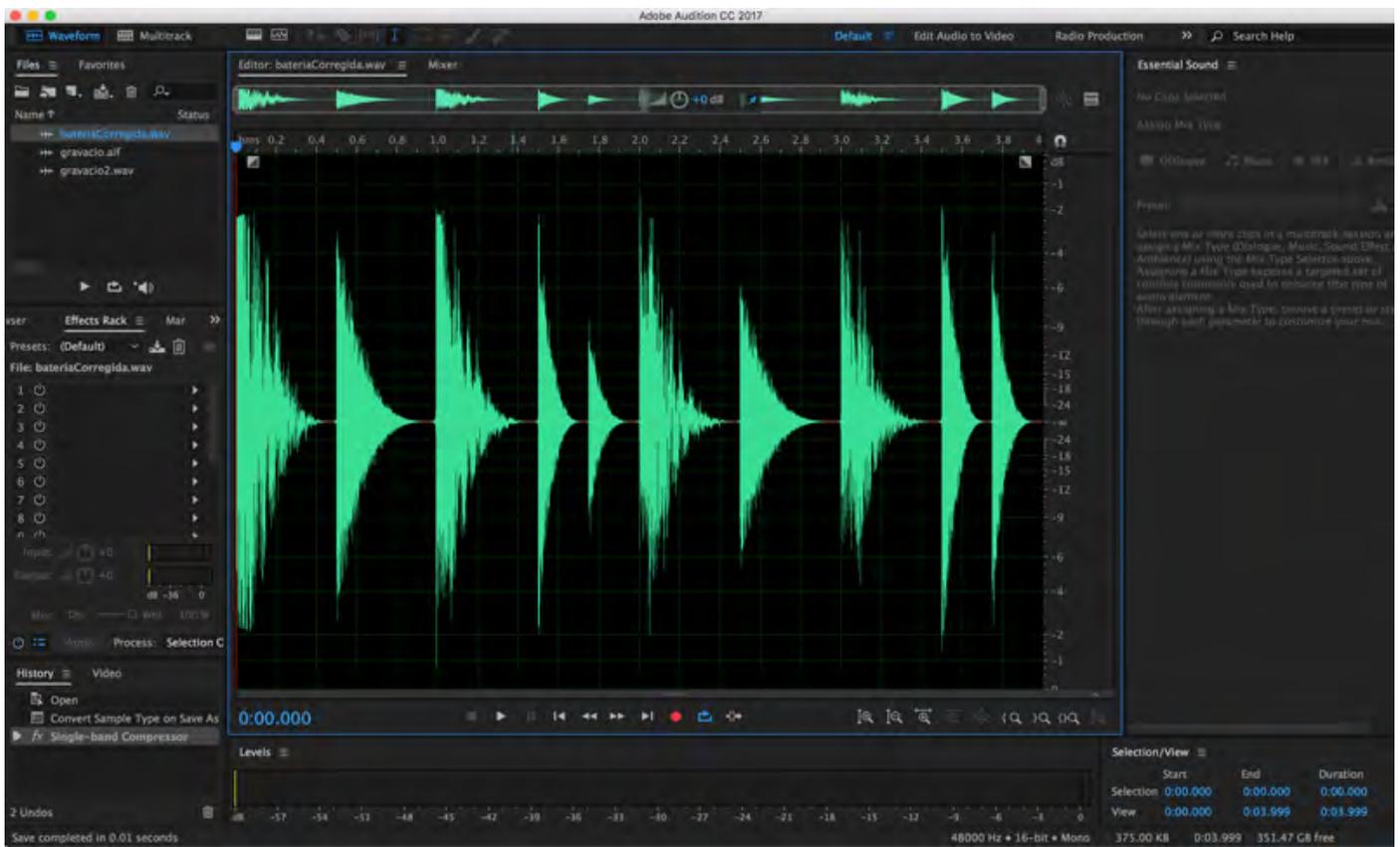


Figura 21. Forma de onda del fichero «bateriaCorregida.wav».

Ejercicios resueltos módulo 2

Ejercicio 4

Enunciado

Abre el archivo «[273177_xserra_la-vaca-cega-eva.wav](#)» con tu editor de audio. Con este fichero, genera otro fichero de audio con una o dos de las estrofas del poema (escoge la estrofa o las estrofas que permitan percibir mejor los efectos aplicados) y, sobre este fragmento, realiza las siguientes operaciones:

1. Aplícale un efecto que simule que la mujer está hablando por teléfono.
2. Aplícale un efecto de dinámica que acentúe los sonidos de los movimientos bucales (lengua, saliva, etc.) y de la respiración de la mujer.
3. Aplícale un efecto que haga lo contrario. Es decir, que atenúe o elimine los sonidos bucales y de respiración.
4. Aplícale un efecto que le cambie el tono de la voz para que suene más grave, sin cambiar la velocidad a la que recita la mujer.
5. Modifica la frecuencia de muestreo a 22.050 Hz sin modificar la frecuencia del sonido (remuestreo).
6. Aplícale un efecto para que el sonido original suene como si hubiera varias voces recitando al mismo tiempo.

Ejercicios resueltos módulo 2

Ejercicio 4

Solución

1. El sonido que recibimos por un teléfono se caracteriza por estar muy comprimido y ligeramente distorsionado. Pero si hay un filtro que nos puede ayudar a recrear el sonido de un teléfono, este es un filtro de paso de banda. Como veremos, solo con el filtro ya obtendremos un resultado bastante aproximado al que se pide.

[poemaA.wav](#)

Las frecuencias presentes en la voz hablada van aproximadamente desde 100 Hz hasta 6 kHz. Pero el canal telefónico solo permite frecuencias entre 300 y 3.400 Hz.

Un «Filtro FFT» nos permite configurar con precisión un filtro de paso de banda que recorte las frecuencias inferiores a los 300 Hz y superiores a los 3.400 Hz.

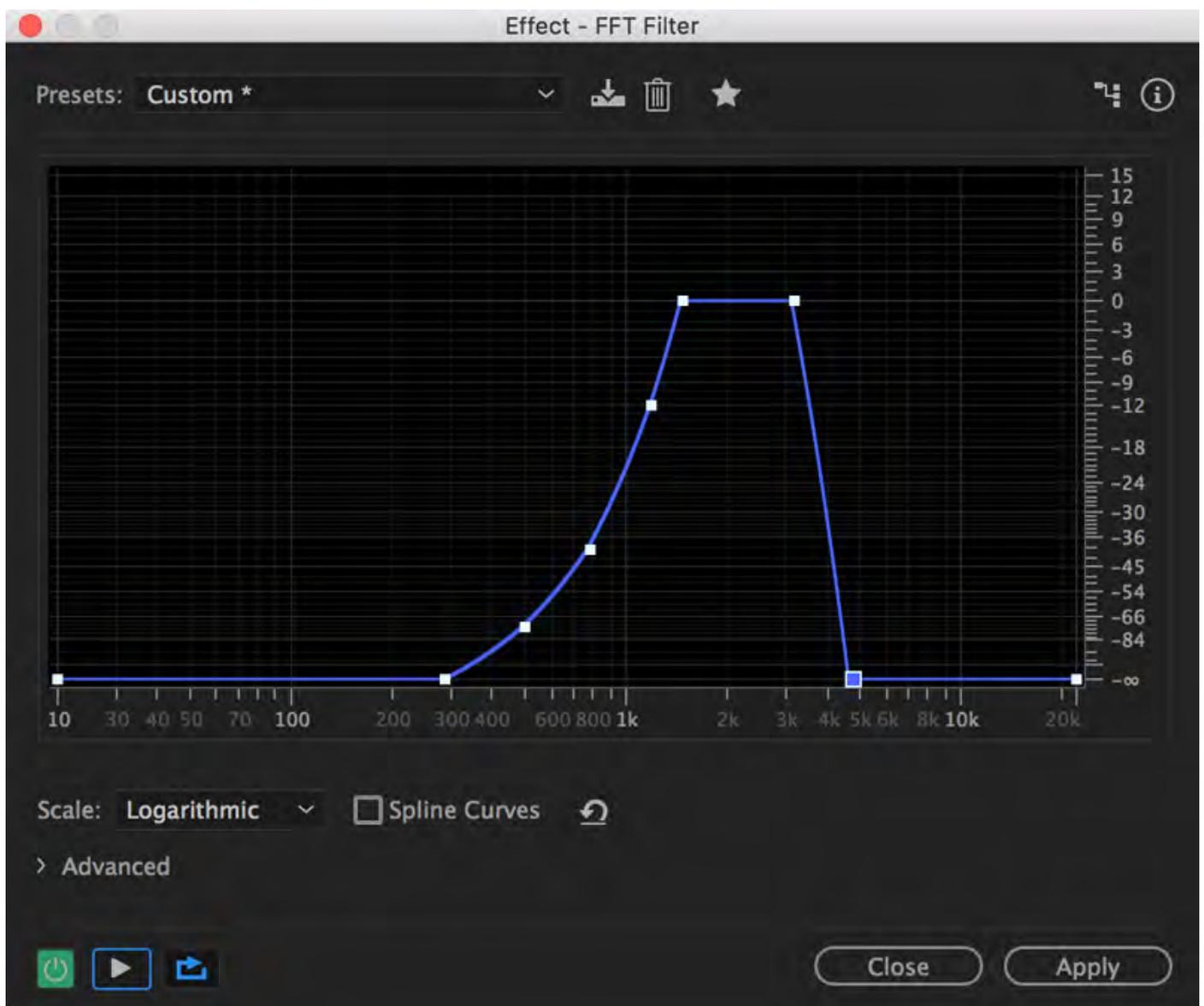


Figura 22

2. Como hemos visto, un compresor es un efecto que sirve para reducir el rango dinámico de una señal. Comprimiendo una señal podemos hacer que sonidos que quedaban a un nivel muy bajo en la grabación original pasen a estar mucho más presentes.

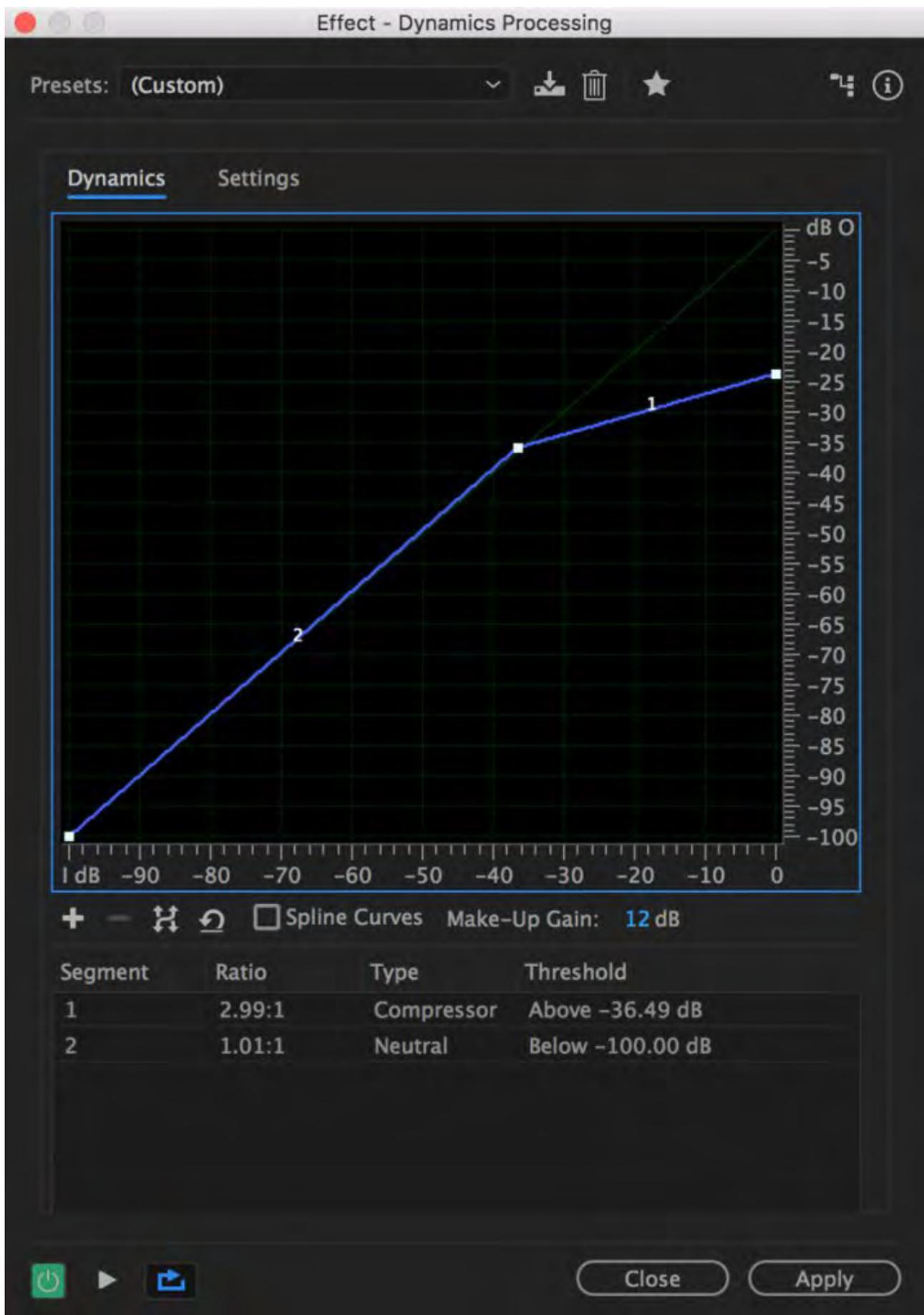


Figura 23

Con un procesador de dinámica configuramos el compresor para reducir los valores que se encuentran por encima de aproximadamente -40 dB. Al resultado final le damos una ganancia de 12 dB para compensar la pérdida de amplitud.

Debe quedar claro que este efecto y el efecto «Compresor» de una banda del ejercicio 3 son básicamente el mismo: un compresor. La diferencia más importante entre ellos es su interfaz.

Al escuchar el resultado final es fácil oír que los sonidos bucales son más evidentes que en el sonido original.

[poemaB.wav](#)

3. Si queremos aplicar un efecto de dinámica que haga lo contrario, debemos aplicar un expansor. Un expansor acentúa las diferencias disminuyendo los niveles débiles y aumentando los fuertes, que es lo que se pide.

Volvemos a usar el Procesador de Dinámica. Los dos primeros segmentos de la curva permiten disminuir el nivel de los niveles débiles, mientras que el último segmento aumenta el nivel de las muestras que ya tenían un nivel alto.

[poemaC.wav](#)

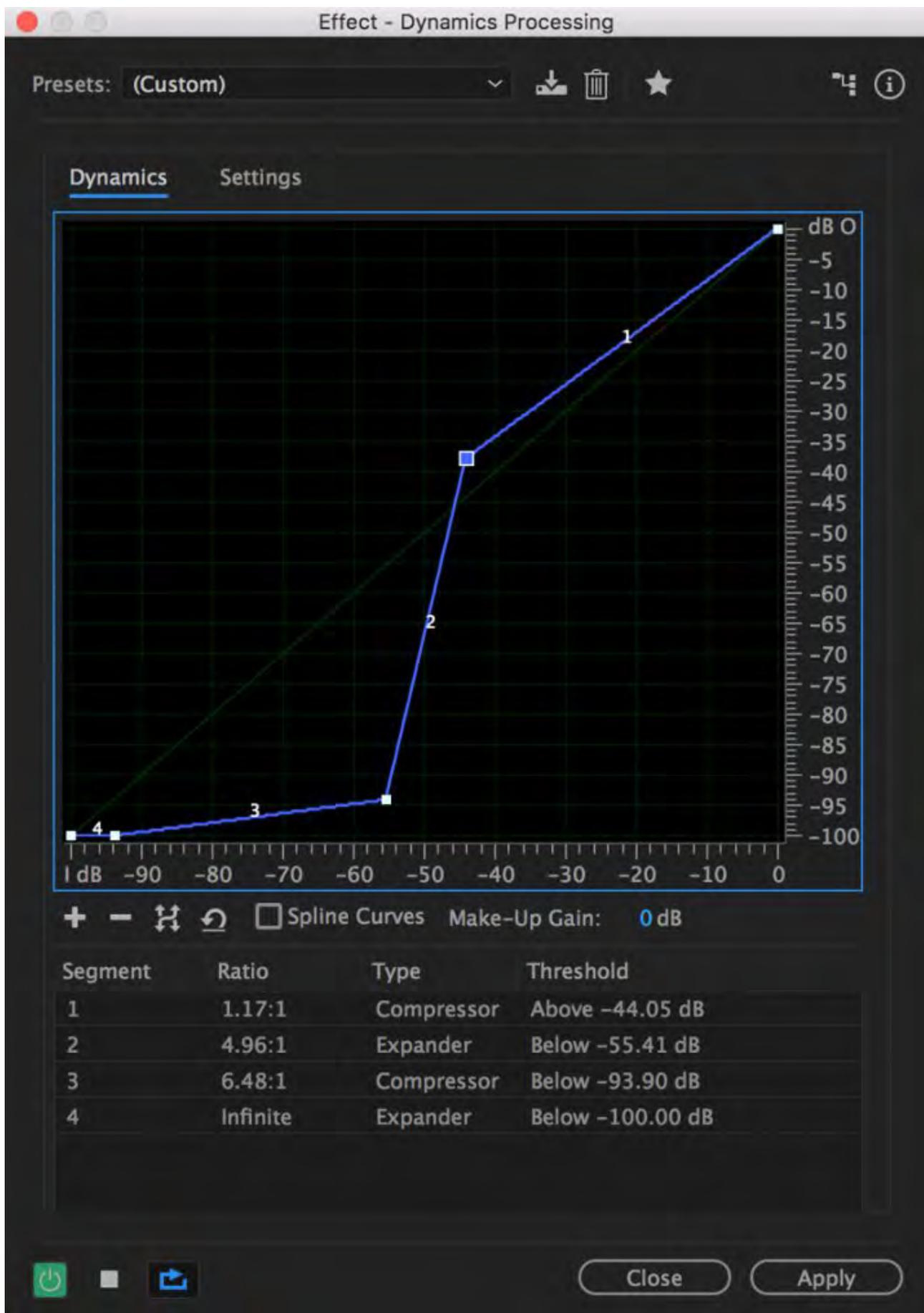


Figura 24

4. Para realizar este apartado podemos escoger el efecto «Pitch Shifter» y asignar un valor de semitonos < 0. Por ejemplo, -5.

El resultado se puede consultar en el archivo siguiente:

[poemaD.wav](#)

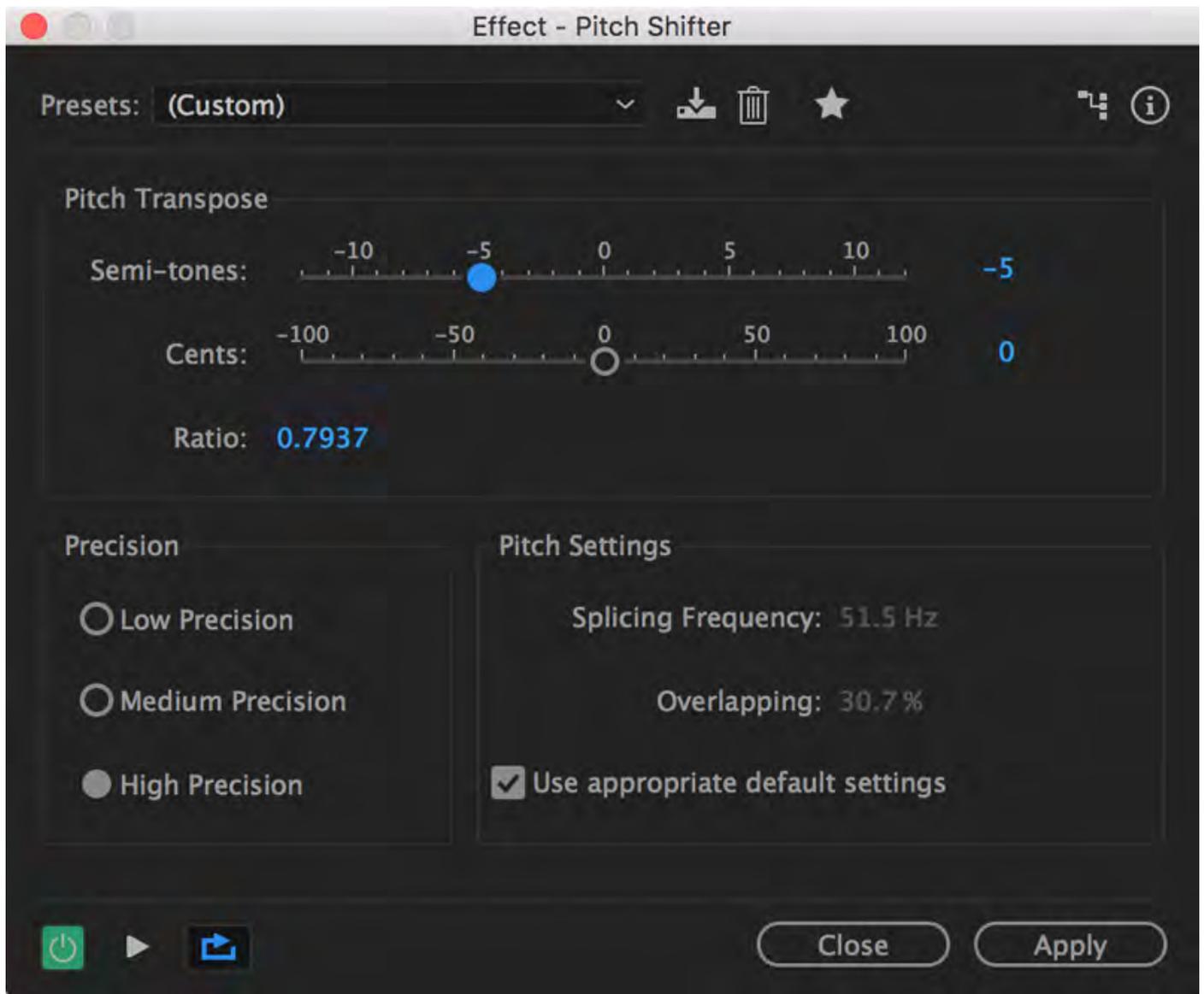


Figura 25

5. Para realizar este apartado simplemente debemos seleccionar «Guardar como» y cambiar el formato del audio dentro de las opciones. En el desplegable de la frecuencia debemos escoger 22.050 Hz.

[poemaE.wav](#)

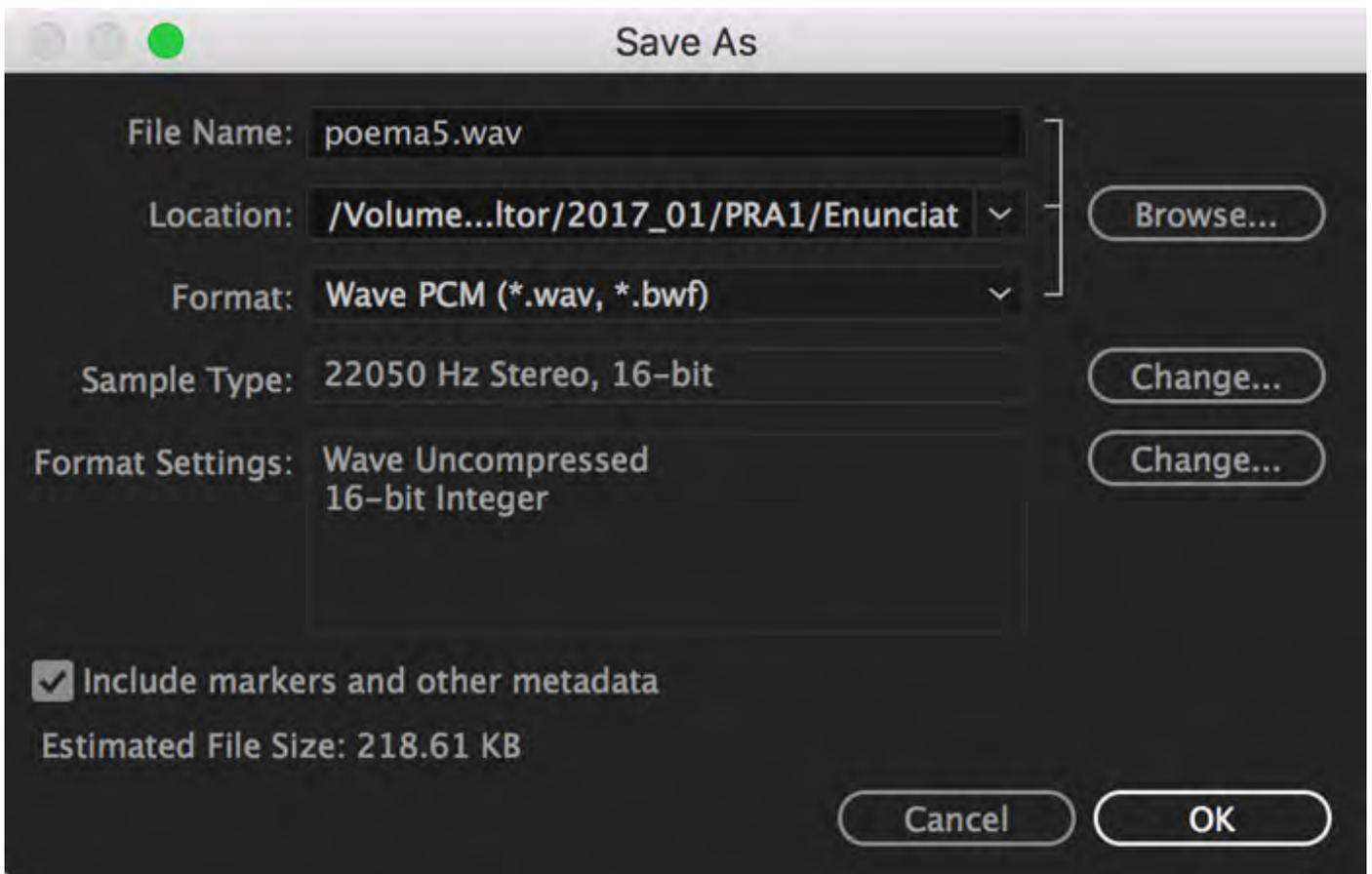


Figura 26

6. Hay distintas formas de realizar este punto. Seguramente la más sencilla sea usar el efecto «Chorus». Como podemos leer en la teoría: «El *chorus* es un efecto muy utilizado, con el cual se intenta simular que un solo instrumento (o una sola voz) suene como varios instrumentos a coro.».

Configuramos 5 voces y, sobre todo con los parámetros que controlan el retraso o *delay*, podemos intentar diferenciar las distintas voces. Debemos ir con cuidado con los parámetros, puesto que unos valores demasiado grandes pueden dar como resultado sonidos poco naturales.

[poemaF.wav](#)



Figura 27

Ejercicios resueltos módulo 2

Ejercicio 5

Enunciado

Al sonido de la figura 28 le hemos aplicado un compresor y un expansor para generar los sonidos de las figuras 29 y 30. Es decir, primero hemos aplicado un compresor al sonido de la figura 28 y hemos generado uno de los sonidos, y después hemos aplicado un expansor al sonido original de la figura 28 y hemos generado el otro sonido.

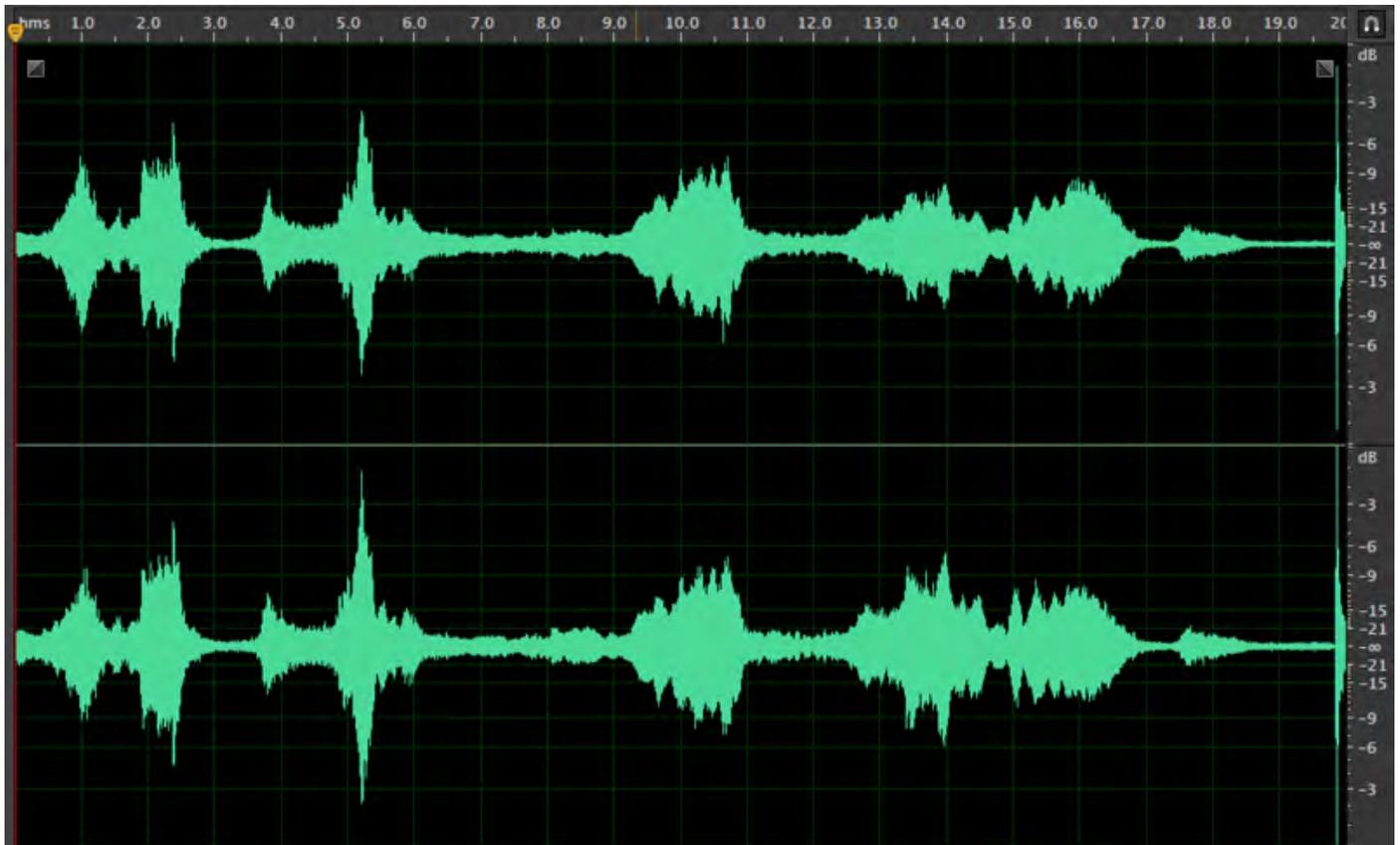


Figura 28

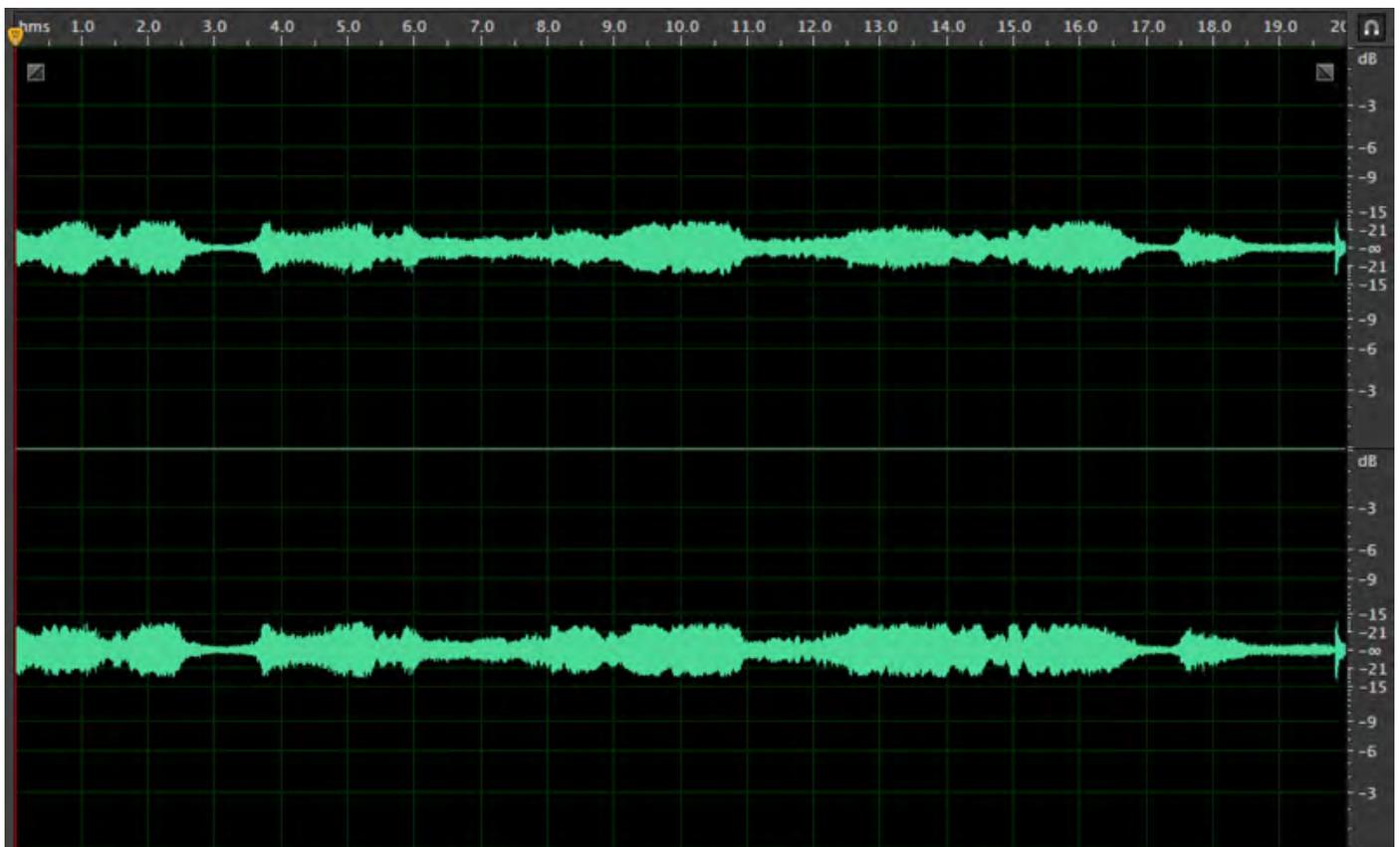


Figura 29

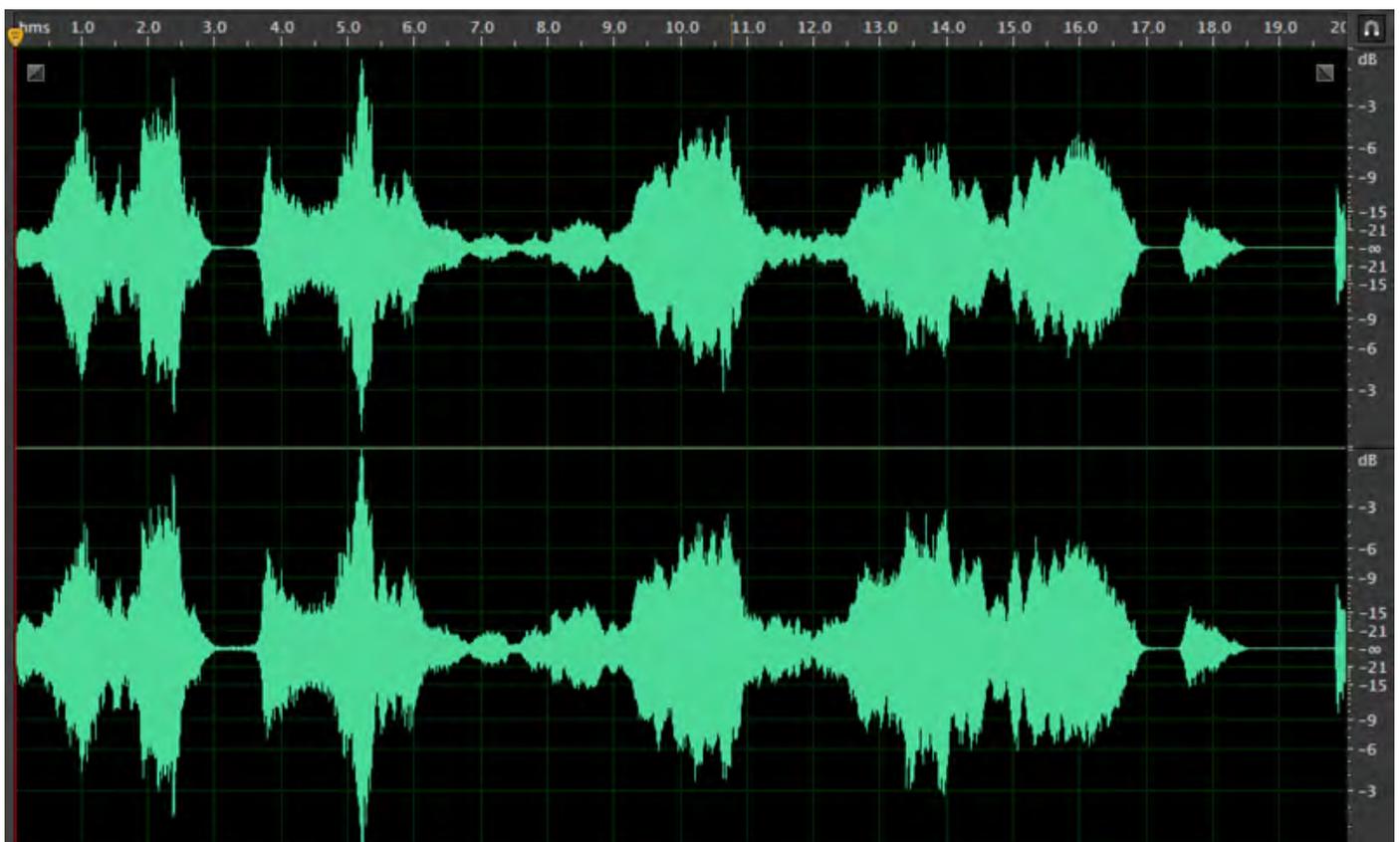


Figura 30

1. A partir de sus formas de onda, ¿qué sonido crees que ha sido generado con el compresor y cuál con el expansor? Razona la respuesta.
2. ¿En qué casos se utiliza un compresor y en cuáles es más conveniente utilizar un expansor?

Ejercicios resueltos módulo 2

Ejercicio 5

Solución

Como ya sabemos, el rango dinámico de un sonido está determinado por la diferencia entre la intensidad más fuerte y la más débil.

La figura 31 nos permite ver claramente que el resultado del primer procesador de dinámica (imagen central) es un sonido con un rango dinámico más pequeño que el rango dinámico del sonido original (imagen izquierda), mientras que el segundo procesador (imagen derecha) nos ha generado un sonido con un rango dinámico más elevado.

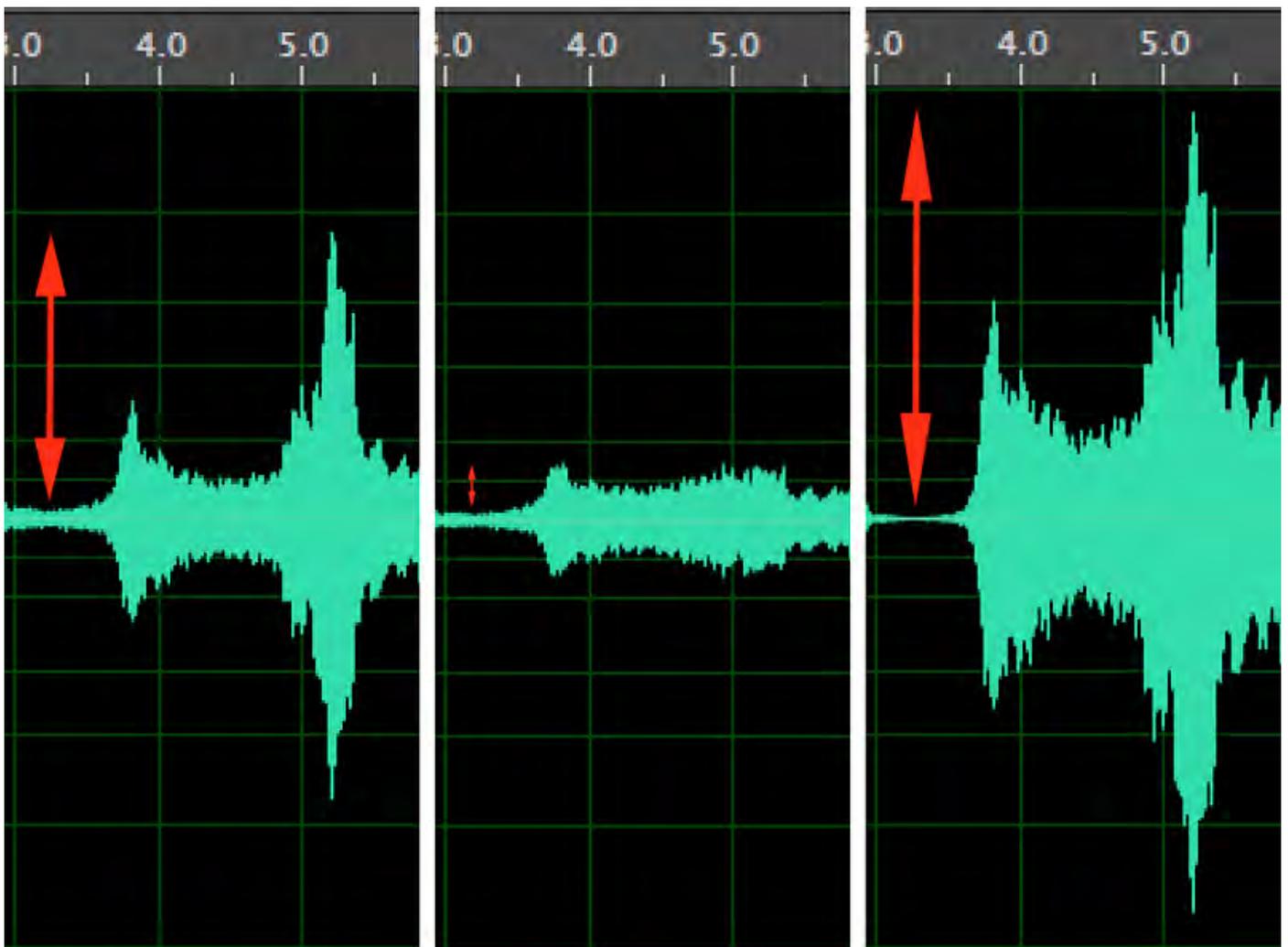


Figura 31. Comparación del rango dinámico en el mismo punto de los tres sonidos.

Por lo tanto, el primer procesador de dinámica es un compresor (reduce el rango dinámico), mientras que el segundo es un expansor (aumenta el rango dinámico).

Los compresores se utilizan mucho en la música rock y pop. Al reducir el rango dinámico de las canciones, es posible situar tanto los niveles fuertes como los débiles muy cerca del máximo, dando la sensación de «pegada».

A veces, cuando escuchamos música clásica, debemos ajustar constantemente el volumen de nuestro equipo, ya que algunos fragmentos que suenan muy fuerte y otros son difícilmente perceptibles. En cambio, cuando escuchamos música rock o pop, normalmente ya no deberemos regular el volumen del equipo, puesto que –por norma general– todo suena muy fuerte.

Los compresores también se usan mucho en grabaciones de voces, percusiones, bajos, etc. A veces, la diferencia entre los niveles mínimos y máximos es demasiado elevada o presenta irregularidades, y el compresor nos permite conseguir un resultado más equilibrado dinámicamente.

Finalmente, también es muy habitual usar los compresores durante el proceso de masterización (*). Como ya hemos comentado, es muy habitual aplicar compresiones muy elevadas en algunos géneros musicales o en productos sonoros destinados a la publicidad en radio y televisión.

Si bien no siempre es necesario ni aconsejable, aplicar niveles de compresión excesivos para no perder naturalidad es un proceso muy habitual durante la masterización de cualquier producto sonoro, puesto que permite equilibrar y conjuntar dinámicamente los diferentes elementos sonoros que lo componen.

Dado que aumentan el rango dinámico, los expansores se usan mucho, por ejemplo, sobre material proveniente de cintas magnéticas analógicas que, con el tiempo, han perdido definición. También se pueden usar para reducir el ruido de fondo de una grabación, puesto que permiten aumentar la diferencia de nivel entre el sonido grabado –una voz, por ejemplo– y el ruido de fondo que se haya podido captar.

Ejercicios resueltos módulo 2

Ejercicio 6

Enunciado

En trabajos de esta asignatura o en otros del grado de multimedia, es habitual leer sentencias como «el sonido tendrá más calidad cuanto mayor sea su resolución» o –el equivalente en el ámbito visual– «cuanto mayor sea la resolución, más calidad tendrá la imagen». Para probar la validez de estas sentencias efectuaremos las operaciones siguientes:

1. Utiliza tu ordenador, tu teléfono o cualquier dispositivo al que tengas acceso, para grabar, con una calidad de 11.025 Hz y 8 bits y añadiendo tu nombre y apellido, la frase siguiente: «Un sonido digitalizado siempre tendrá más calidad cuanto mayor sea su resolución».
 2. Guarda el resultado de la grabación (en formato MP3 o WAV, o en el formato que asigne por defecto tu dispositivo de grabación) con el nombre «soBaixaResolucio.xxx».
 3. Con tu editor de audio, convierte el sonido a 48 KHZ y 32 bits. Guarda el resultado en formato WAV, sin comprimir, y con el nombre «soAltaResolucio.wav».
 4. Compara el espacio que ocupan los dos archivos en tu disco duro. ¿Cuál de los dos ocupa más espacio? ¿Por qué?
 5. Compara auditivamente los dos sonidos: el sonido con «baja resolución» y el sonido con «alta resolución». Si entendemos la calidad de una grabación como su similitud con el sonido real, razona si es cierto decir que «un sonido digitalizado siempre tendrá más calidad cuanto mayor sea su resolución».
- **Nota 1:** para hacer los ejercicios en los que se pida efectuar una grabación se aconseja usar el teléfono móvil. Se debe configurar el teléfono móvil de forma que la grabación se realice en la máxima calidad que el dispositivo permita. El resultado se debe importar a un editor de audio para realizar las operaciones, comparaciones y análisis que se piden.
 - **Nota 2:** para hacer los ejercicios se aconseja llevar a cabola escucha con auriculares si no se dispone de un buen sistema de altavoces.

Ejercicios resueltos módulo 2

Ejercicio 6

Solución

1 y 2. Se adjunta el archivo «soBaixaResolucio.wav» a 11.025 Hz y 8 bits.

[soBaixaResolucio.wav](#)

3. Algunos editores de audio, como Audition, graban por defecto en 32 bits. Es decir, aunque el usuario cree un archivo de 8 bits, el editor puede estar realmente grabando a 32 bits. Por lo tanto, en el caso de haber usado Audition para grabar el archivo «soBaixaResolucio.wav», es importante guardar el archivo, cerrarlo y abrirlo de nuevo antes de hacer la conversión, para asegurarnos de que se está trabajando sobre un archivo de 8 bits. Por ejemplo, el ruido de fondo que hay en el archivo «soBaixaResolucio.wav» también debe aparecer en el archivo «soAltaResolucio.wav». Si ha desaparecido, alguno de los pasos efectuados no ha sido correcto.

4. El tamaño de los archivos de audio digitalizados depende de la frecuencia de muestreo y la profundidad de bits de los archivos. Por lo tanto, lógicamente el tamaño del archivo «soAltaResolucio.wav» (48.000 Hz y 32 bits) debe ser superior al del archivo «soBaixaResolucio.wav» (11.025 Hz y 8 bits).

Si queremos calcular con más precisión el número de bytes que ocupan los archivos en el disco duro, podemos hacer el siguiente cálculo:

$(\text{Frecuencia de muestreo} \times \text{Profundidad de bits} \times \text{Duración en segundos} \times \text{Número de canales}) / 8$

Sin embargo, después de hacer estos cálculos vemos que nuestros archivos ocupan algo más de espacio en nuestro ordenador. ¿Por qué? Por una parte, un archivo puede tener 500 bytes de información, pero puede ser que el sistema operativo utilice 512 bytes de espacio para guardarlo. El espacio exacto que utilice el sistema operativo dependerá del sistema de archivos que utilice.

Y, por otra parte, además de las muestras de audio, los archivos de audio digital también incluyen una cabecera de información. En esta cabecera se indica la frecuencia de muestreo del sonido, su profundidad de bits, si es monofónico o estéreo, etc. También hace posible que el archivo sea interpretado correctamente en cualquier dispositivo.

5. El proceso de grabación es el que determina la calidad del sonido. Cualquier pérdida en el proceso de grabación es irreversible. Cualquier frecuencia, sonido, detalle, etc. que no se capte inicialmente no se podrá recuperar posteriormente.

Si al hacer una fotografía a un grupo de personas la cabeza de una de ellas queda cortada, esa cabeza no aparecerá aunque convirtamos la fotografía a la máxima calidad posible. Lo mismo sucede con el sonido. Los datos que no sean capturados en la grabación inicial no aparecerán posteriormente. Al modificar la frecuencia de muestreo o la profundidad de bits del archivo, simplemente estamos duplicando la información que ya existe o insertando nueva información que el programa genera, pero que en ningún caso aportará mayor calidad al sonido.

Por lo tanto, no es cierto que un sonido digitalizado siempre tendrá más calidad cuanto mayor sea su resolución.

Ejercicios propuestos módulo 2

Ejercicio 1

El siguiente archivo de audio digital corresponde a un archivo de audio de una voz muy saturada:

[344250_honest-cactus_normal-voice-and-intro_Saturation.wav](#)

1. Abre y reproduce el archivo en tu editor de audio y familiarízate con el sonido distorsionado de una señal saturada.
2. Al archivo de audio, aplícale un cambio de amplitud de -6 dB y vuélvelo a reproducir. Tras reducir la amplitud del archivo de audio, ¿ha desaparecido la distorsión?
3. En tu editor de audio, utiliza el zoom temporal para visualizar en detalle la forma de onda del archivo de audio, y estudia sus puntos de saturación para familiarizarte con la forma de onda de un archivo de sonido saturado.
4. Compara auditivamente el archivo de audio saturado con el archivo de audio original, sin saturar:

[344250_honest-cactus_normal-voice-and-intro.wav](#)

5. ¿Qué operación crees que se le ha aplicado al archivo original para generar el archivo saturado?
6. En tu editor de audio, utiliza el zoom temporal para visualizar en detalle la forma de onda del archivo sin saturar y compárala con la forma de onda del archivo saturado.

Nota: Aunque estos ejercicios hacen referencia principalmente a los conceptos estudiados en el módulo 2, algunos de los apartados requerirán que el estudiante también consulte el módulo 6 para resolverlos.

Ejercicios propuestos módulo 2

Ejercicio 2

Repite el ejercicio 1 con una grabación de tu voz. Es decir:

1. Con un dispositivo de grabación (teléfono, ordenador, etc.), graba tu voz de forma que la grabación quede muy saturada.
2. Abre y reproduce el archivo en tu editor de audio (si la grabación se ha realizado, por ejemplo, con un teléfono móvil, deberás importarla). Familiarízate con el sonido distorsionado de una señal saturada.
3. Aplica a tu grabación un cambio de amplitud de -6 dB y vuélvela a reproducir. Detecta auditivamente si la distorsión ha desaparecido o está menos presente.
4. Finalmente, estudia la forma de onda de la señal y familiarízate con la forma de onda de una señal saturada.
5. ¿Crees que es posible eliminar la saturación de una grabación una vez realizada?

Ejercicios propuestos módulo 2

Ejercicio 3

Con un dispositivo de grabación (teléfono, ordenador, etc.), graba tu voz con dos configuraciones diferentes: al máximo nivel posible pero sin llegar a saturar, y a un nivel muy bajo. Abre los dos archivos en tu editor de audio y:

1. Aplica una ganancia a la segunda señal, de forma que las dos grabaciones tengan un nivel similar.
2. Compara auditivamente las dos grabaciones. En particular, compara el ruido de fondo presente en las dos grabaciones. ¿En cuál de las dos grabaciones está más presente el ruido de fondo?

Ejercicios propuestos módulo 2

Ejercicio 4

Repita el ejercicio 3 pero grabando otros fondos sonoros: animales, naturaleza, tráfico, el ambiente de una estación de metro, etc.

Ejercicios propuestos módulo 2

Ejercicio 5

Abre en tu editor de audio el archivo de audio digital «[390514_tylean_counting-1-to-10.wav](#)» y realiza las siguientes operaciones:

1. Aplícale una inversión temporal (el resultado debe ser el mismo que obtendríamos si reprodujéramos un disco de vinilo al revés).
2. ¿Pueden generarse clics u otros ruidos al invertir todo el archivo de audio?
3. Sobre el archivo original, invierte solo los números impares. Es decir, el resultado sonoro debería ser 1 (invertido), 2 (sin invertir), 3 (invertido), etc.
4. ¿Pueden generarse clics al realizar esta última operación? Si la respuesta es afirmativa, ¿cómo evitarías o eliminarías estos clics generados?

Ejercicios propuestos módulo 2

Ejercicio 6

Edita el archivo «[390514__tylean__counting-1-to-10.wav](#)» de forma que:

1. La narración pase a ser una cuenta atrás. Es decir, los números pasen a estar ordenados de mayor a menor «10, 9, 8, etc.». El archivo resultante debe tener exactamente la misma duración que el archivo original.
2. ¿Qué técnicas usarías para evitar o eliminar los clics generados?

Ejercicios propuestos módulo 2

Ejercicio 7

Edita el archivo «[390514_tylean_counting-1-to-10.wav](#)» de forma que las pausas entre los diferentes números sean más cortas y el resultado final tenga una duración exacta de 20 segundos. El resultado final no debe contener ningún clic ni ningún ruido y no debe notarse que el archivo ha sido editado.

Ejercicios propuestos módulo 2

Ejercicio 8

Edita el archivo «[177269_sergeo_numbers-in-french.wav](#)» de forma que las pausas entre los diferentes números sean más cortas y el resultado final tenga una duración exacta de 20 segundos. El resultado final no debe contener ningún clic ni ningún ruido y no debe notarse que el archivo ha sido editado.

Ejercicios propuestos módulo 2

Ejercicio 9

Elimina el inicio y el final de los siguientes sonidos, de forma que sus duraciones pasen a ser la mitad de la duración original. Usa un *fade in* y un *fade out* (fundido de entrada y fundido de salida) para prevenir que se creen clics:

[Sine1200Hz.wav](#)

[126444_harregarre_compilatie.mp3](#)

[332769_emanuele-correani_ambience-train-station-inside.wav](#)

Ejercicios propuestos módulo 2

Ejercicio 10

La figura 32 muestra un fragmento de la forma de onda de un sonido con un clic, y un fragmento de la forma de onda de un sonido con puntos de saturación.

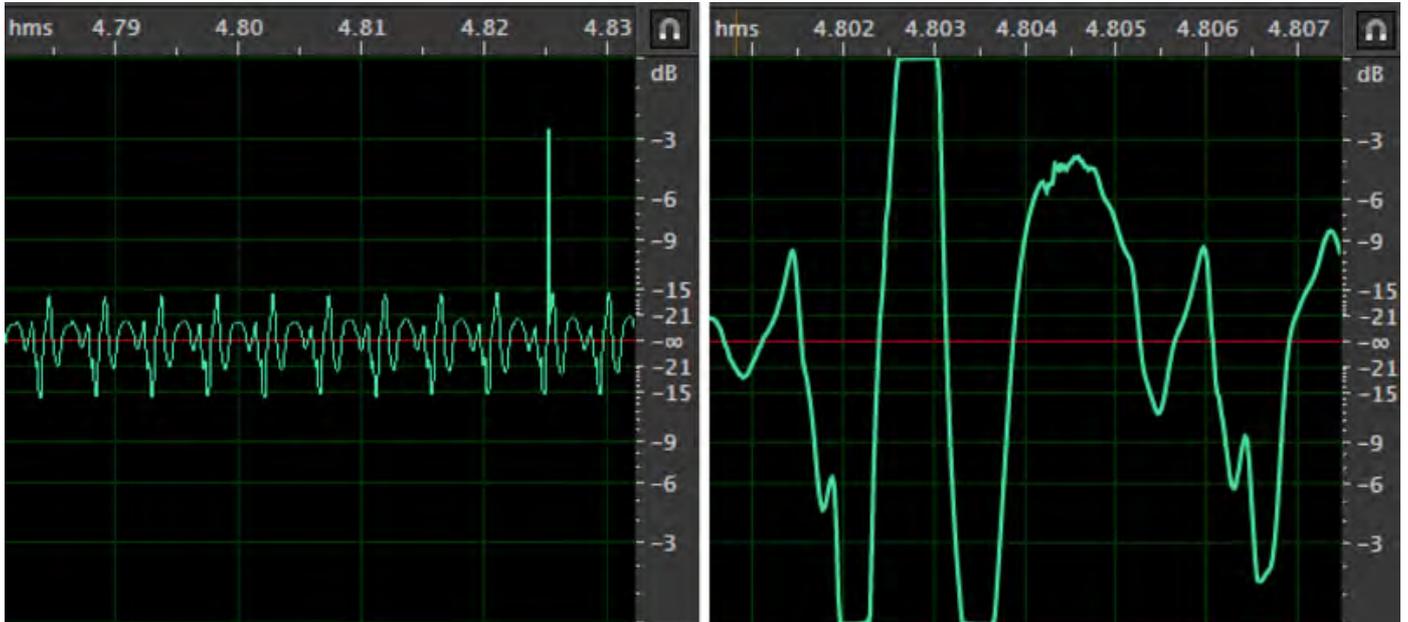


Figura 32

1. ¿Cuál de las imágenes corresponde al sonido con un clic y cuál al sonido con puntos de saturación?
2. Sonoramente, ¿en qué se diferencia un clic de un punto de saturación?
3. ¿En qué casos es posible que se genere un sonido con puntos de saturación?
4. Una vez generado un sonido muy saturado (es decir, con muchos puntos de saturación), ¿es posible, con la ayuda de algún efecto, recuperar el sonido original sin saturación?

Ejercicios propuestos módulo 2

Ejercicio 11

Abre el archivo «[AmbientSoundClics.mp3](#)» con tu editor de audio y:

1. Localiza auditivamente los clics presentes en el archivo.
2. Ajusta al máximo los dos zooms de tu editor de audio para estudiar estos clics con detalle y determinar, para cada uno de ellos y con la máxima precisión posible, sus posiciones, el número de muestras de clics y los valores máximos de estas muestras.
3. Usa alguna de las herramientas de tu editor de audio para eliminar completamente estos clics sin que se vea afectado el resto del archivo.

Ejercicios propuestos módulo 2

Ejercicio 12

Razona qué tipo de filtro digital básico usarías en las siguientes situaciones:

1. Tenemos una grabación de voz en un bar muy ruidoso. Se nos pide que modifiquemos el archivo original para hacerlo más inteligible.
2. Tenemos la grabación de un bombo de batería en la que se escucha, de fondo, el sonido de la guitarra. Se nos pide que solo se escuche el bombo.
3. Tenemos la mezcla final de la canción de un grupo y observamos que la voz del cantante ha quedado mucho más fuerte que el resto de los instrumentos. Queremos atenuar un poco la voz sin que afecte demasiado al resto de instrumentos.

Ejercicios propuestos módulo 2

Ejercicio 13

Supongamos que la forma de onda del archivo digitalizado de la grabación en estéreo de un concierto de música clásica es como la de la figura 33.

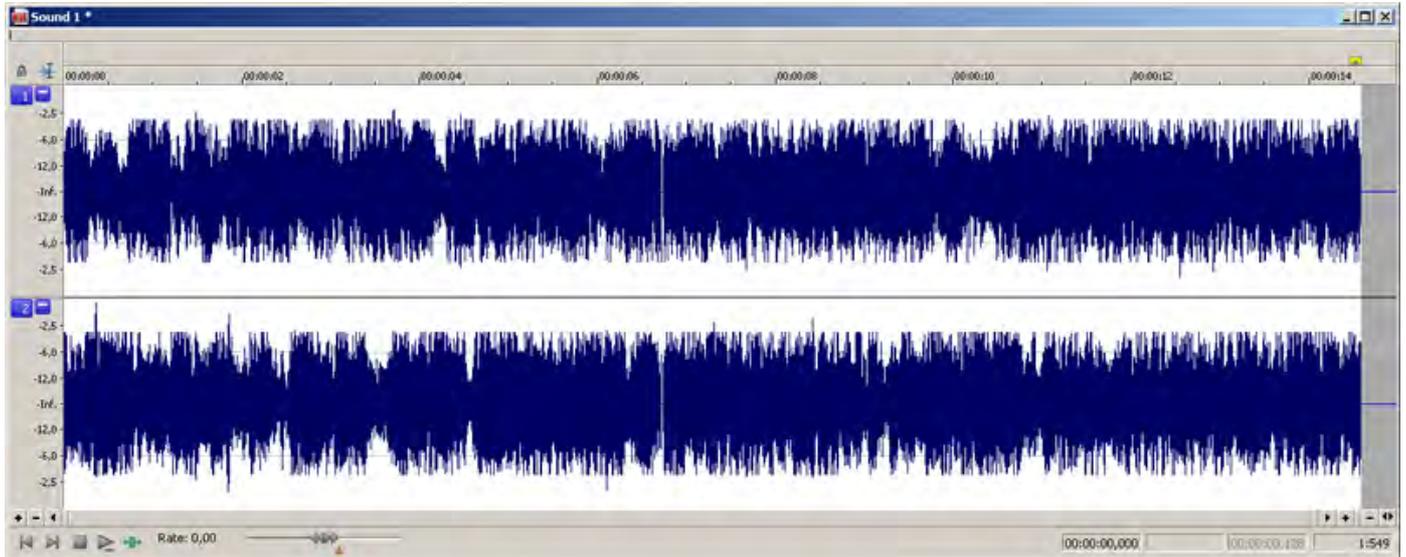


Figura 33

¿De qué efecto crees que se ha abusado?

1. Compresor
2. Expansor
3. Reverberación
4. Puerta de ruido

Ejercicios propuestos módulo 2

Ejercicio 14

Con el programa Audition, al sonido

[Ex3_original.wav](#)

le hemos aplicado tres filtros: un filtro de paso bajo, un filtro de paso alto y un filtro de paso de banda. A continuación encontraréis los sonidos resultantes de aplicar estos filtros al sonido original:

[Ex3_modificat 1.wav](#)

[Ex3_modificat 2.wav](#)

[Ex3_modificat 3.wav](#)

Teniendo en cuenta esto:

1. ¿Qué filtro crees que se ha aplicado a cada sonido?
2. ¿Qué particularidades del sonido resultante te han hecho decidir el filtro del punto 1?
3. ¿Crees que se podría deducir si se ha usado un ecualizador gráfico o paramétrico para generar los sonidos modificados?

Ejercicios propuestos módulo 2

Ejercicio 15

Abre el archivo de audio «[43555_dobroide_20071103-description-human.wav](#)» y realiza los ejercicios siguientes:

1. Convierte el archivo de audio «[43555_dobroide_20071103-description-human.wav](#)» al formato Mp3 con Bit rate = 128 Kbps, 44.100 Hz.
2. Compara el tamaño del archivo WAV original con el del Mp3 generado.
3. Compara los espectrogramas de los archivos WAV y Mp3 y comenta las diferencias que encuentras.
4. ¿Detectas alguna diferencia sonora entre el sonido WAV y el Mp3? Reprodúcelos en distintos equipos (altavoces y auriculares, por ejemplo).
5. Finalmente, cambia la frecuencia de muestreo del archivo WAV a 22.050 Hz y explica brevemente en qué consiste este proceso.

Ejercicios propuestos módulo 2

Ejercicio 16

¿Qué ventajas e inconvenientes tiene digitalizar un sonido a 48 KHZ y 16 bits? ¿Y a 96 KHZ y 24 bits?

Ejercicios módulo 3

Enunciado

Para conmemorar el 78 aniversario del nacimiento de Robert Moog, Google publicó un doodle inspirado en sus instrumentos: <https://www.google.com/doodles/robert-moogs-78th-birthday>.

1. ¿Qué tipo de síntesis crees que usa este sintetizador virtual?
2. ¿En qué se diferencian este tipo de síntesis y la síntesis por tabla de ondas?
3. Comenta cuáles son las secciones principales de este sintetizador y analiza en detalle las secciones «Oscillators» y «Filter». Relaciona sus controles y su funcionamiento con los conceptos estudiados en la teoría del curso.

Ejercicios módulo 3

Ejercicio 1

Solución

1. Este sintetizador virtual usa síntesis sustractiva. Sería un ejemplo de sintetizador digital que emula el sonido de un sintetizador analógico por *hardware*.

La opción síntesis aditiva la podemos descartar rápidamente al constatar que la sección «Oscillators» contiene solo dos osciladores. La síntesis aditiva parte del principio que cualquier sonido se puede constituir a partir de la suma de componentes básicos –que se pueden generar con osciladores–, pero para conseguir sonidos mínimamente interesantes necesita un número elevado de osciladores.

La síntesis sustractiva, en cambio, necesita menos osciladores (los sintetizadores por síntesis sustractiva acostumbran a tener entre uno y tres osciladores). Estos osciladores deben ser capaces de crear ondas de espectro rico, y estas ondas serán filtradas por un conjunto de filtros para generar el sonido final. Como podemos ver en la figura 34, este sintetizador desarrollado por Google contiene todos estos elementos. Además, si lo tocamos, podremos ver que el sonido que produce es el típico sonido de un sintetizador por síntesis sustractiva.



Tips

- Click on the doodle to activate "keyboard mode" and play sounds by typing.
- On US English keyboards, the qwerty and number rows emulate the order of synthesizer keys.
- To adjust knobs, click, click and drag up and down, or click and use the arrow keys.
- To record on a different track, click the track's volume meter.

Figura 34. Sintetizador propuesto por Google

<https://www.google.com/doodles/robert-moogs-78th-birthday>

2. La síntesis por tabla de ondas es un método de síntesis digital de sonido. Por lo tanto, también se puede descartar rápidamente que este sintetizador sea un sintetizador de tabla de ondas, puesto que –aunque técnicamente es un sintetizador digital– se trata de un sintetizador que emula a un sintetizador analógico.

Como hemos comentado, los componentes básicos de la síntesis sustractiva son los osciladores y los filtros. Como también hemos visto, aunque tiene un funcionamiento diferente, la síntesis aditiva también se basa en osciladores, capaces de generar ondas simples (ondas sinusoidales).

En cambio, la síntesis por tabla de ondas es un método de síntesis completamente diferente. Los sintetizadores basados en este tipo de síntesis tienen almacenados en la memoria pequeños fragmentos de audio digital de sonidos reales o electrónicos, y generan el sonido combinando, repitiendo, filtrando, etc. estos fragmentos.

Una característica importante de la síntesis por tabla de ondas es que, dado que tiene realmente grabaciones de instrumentos reales en su memoria, es un método ideal para recrear los sonidos de estos instrumentos con más facilidad y acierto.

3. Este sintetizador virtual se podría dividir en seis partes. Empezando por los controles superiores, de izquierda a derecha, encontramos las secciones:

- **Mixer:** Permite controlar los niveles de los osciladores por separado, así como el volumen general de la mezcla de los tres osciladores.
- **Oscillators:** Sección con tres osciladores.
- **Filter:** Sección con un filtro con seis controles.
- **Envelope:** Envoltorio de amplitud que se aplica sobre la mezcla de los tres osciladores una vez filtrada.
- **Modulation:** La rueda de modulación permite manipular el sonido generado. También dispone de un botón *on/off* para activar o desactivar esta opción.
- **Teclado:** Permite generar los sonidos. También es posible generar los sonidos desde el teclado del ordenador.

Además, como complemento, el sintetizador está conectado a una grabadora de cuatro pistas que nos permite grabar y reproducir posteriormente el material grabado.

La sección «Oscillators» contiene tres osciladores. El oscilador número 2 es el oscilador superior; el número 1, el oscilador de enmedio, y el número 3, el oscilador situado en la parte inferior de la sección.

Además del control del nivel ubicado en la sección «Mixer», cada uno de los osciladores dispone de tres controles (*Range*, *Frequency* y *Waveform*), excepto el primer oscilador, que no dispone de *Frequency* (ved figura 35).

En detalle, estos parámetros son:

- **Range:** permite cambiar la octava del oscilador. Es decir, permite configurar su altura, el *pitch*, pero octava a octava.
- **Frequency:** permite modificar la afinación del oscilador. La frecuencia del oscilador número 1 siempre estará determinada por la tecla que se presione, mientras que los osciladores 2 y 3 pueden desafinarse, emulando el sonido de los sintetizadores analógicos de *hardware* clásicos.
- **Waveform:** permite seleccionar la forma de onda del oscilador entre seis opciones posibles.

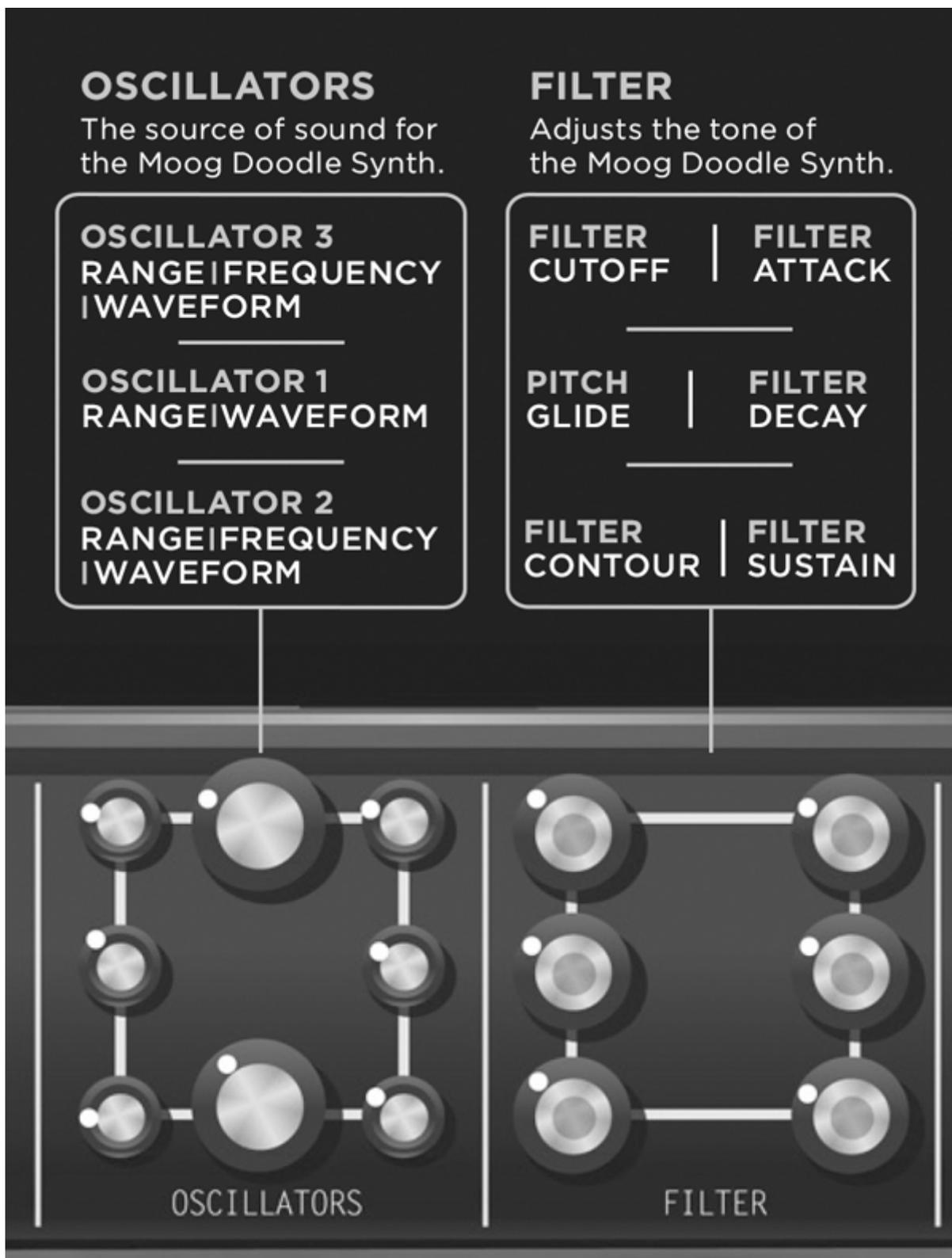


Figura 35. Osciladores y filtros.

<http://googletricks.com/wp-content/google-doodles/moog-doodle-guide.pdf>

En cuanto a la sección «Filter», esta está compuesta de un único filtro de tipo paso de bajo. En detalle, sus controles son (ved figura 35):

- **Cutoff:** es la frecuencia de corte a partir de la cual actúa el filtro.
- **Glide:** controla la cantidad de portamento entre las notas que se vayan tocando. Un valor alto de *glide* genera una transición del *pitch* entre las diferentes notas (por ejemplo, la voz puede crear este efecto), mientras que un valor de *glide* igual a 0 genera notas discontinuas en términos de *pitch* (como sucede, por ejemplo, al tocar las teclas de un piano).
- **Contour:** controla la cantidad de envolvente que se aplica al filtro (ved el siguiente control).

- **Envelope ADS:** los tres controles que encontramos a la derecha del filtro son el *attack*, *decay* y *sustain* de una envolvente que modifica el valor del *Cutoff* del filtro a lo largo del tiempo. Un filtro que vaya modificando el valor del *Cutoff* a lo largo del tiempo acostumbra a generar sonidos más interesantes que un filtro con un valor de *Cutoff* fijo.

Ejercicios módulo 3

Ejercicio 2

Enunciado

Abre el sintetizador *online* <https://stuartmemo.com/synth/>.

Activa solo uno de los 3 osciladores. Analiza en detalle el funcionamiento del «Volume Filter» que genera una envolvente de amplitud del tipo ADSR.

Nota: El sintetizador no funciona con todos los navegadores. Para funcionar debe poder cargar un teclado en la parte inferior de los controles, que se puede tocar con el ratón o con las teclas de la «a» a la «l» y otras teclas colindantes.

Ejercicios módulo 3

Ejercicio 2

Solución

En la Figura 36 se muestra el sintetizador completamente cargado en un navegador.

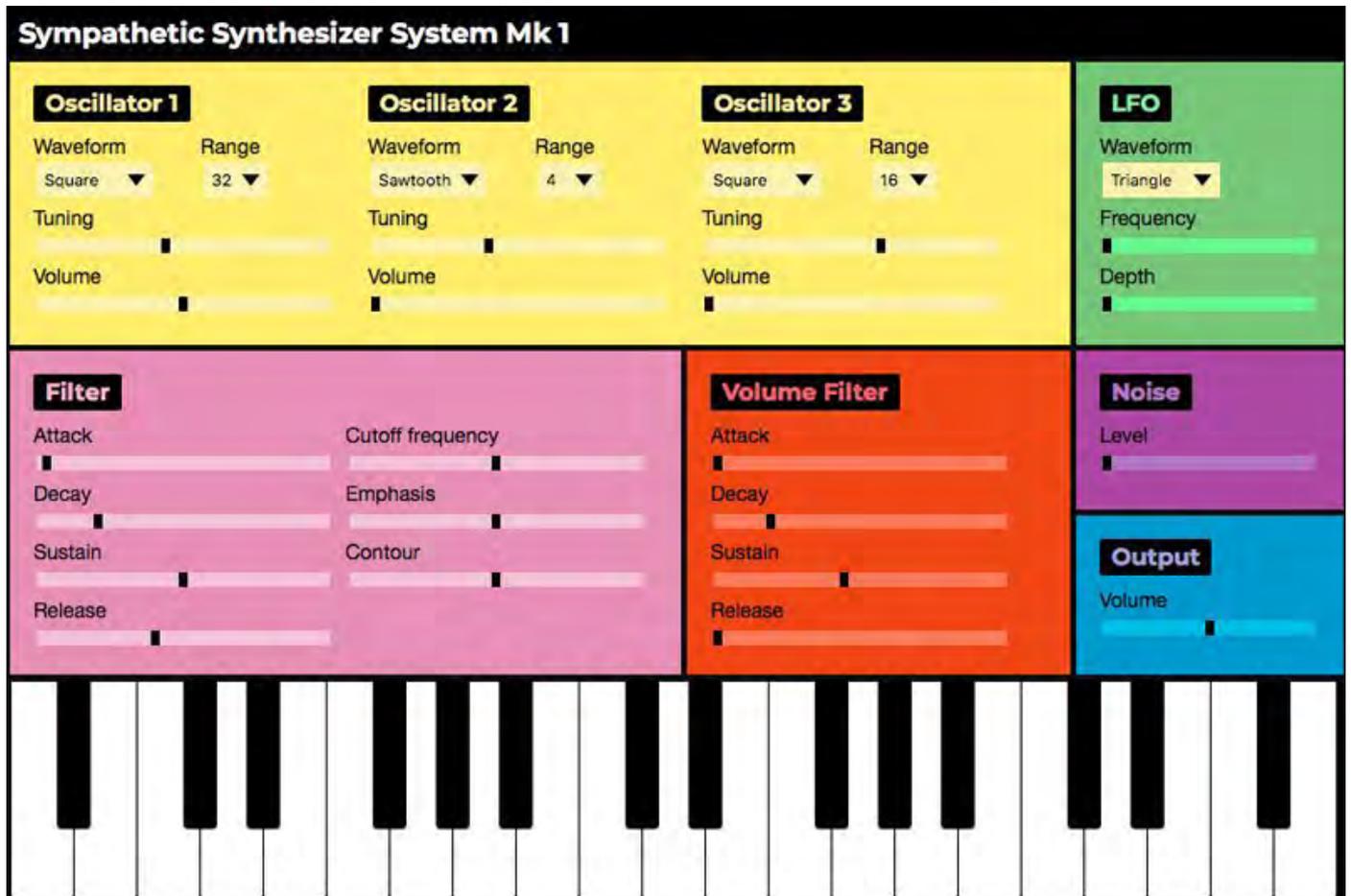


Figura 36. Sintetizador cargado correctamente.

Para activar solo uno de los sintetizadores, se debe poner el *fader* de los «Oscillator2» y «Oscillator3» a cero (a la izquierda del todo). A partir de aquí, si se pulsa la secuencia de teclas «a», «s», «d», «f», «g», «h», «j» y «k» se escucha la escala de do mayor. La secuencia de teclas «w», «e» y «t», «y», «u» hacen sonar las notas negras del teclado.

Para cambiar de octava, se debe cambiar el *range* de frecuencias y seleccionar uno de los números (32, 16, 8, 4 o 2), lo que hará que el teclado se mueva sucesivamente por octavas, de grave a agudo.

El menú de *waveform* permite seleccionar la forma de onda generada (triangular, diente de sierra y triangular) con las diferentes distribuciones espectrales que implica cada una de ellas (es decir, diferentes timbres).

Tal y como indica el enunciado, la sección «Volume Filter» no es más que un generador de envolventes ADSR. La curva ADSR, de forma genérica, contiene los 4 parámetros siguientes:

- **Ataque (Attack):** El ataque de un sonido de un instrumento se produce en el momento de empezar o atacar la nota. Se puede decir que es el tiempo que tarda el sonido desde que empieza hasta llegar a su máxima amplitud. Se mide en segundos.
- **Decaimiento (Decay):** Después del ataque se produce una atenuación del sonido. El decaimiento es el tiempo que tarda el sonido en pasar de su volumen máximo (final del ataque) al mantenimiento. Se mide en segundos.
- **Mantenimiento (Sustain):** Corresponde al nivel de sonido en la zona estable del sonido. Durante el mantenimiento, el timbre y el volumen del sonido permanecen constantes. La duración del mantenimiento no se puede conocer *a priori*, puesto que dependerá del tiempo durante el cual el usuario esté pulsando la tecla. Así pues, el mantenimiento es una medida de nivel (volumen) del sonido en su parte estable y continua, y se mide en porcentaje o en decibelios dBs. Los instrumentos percusivos no tienen parte de mantenimiento.

- **La extinción o liberación (Release):** Es el tiempo que tarda el sonido en desaparecer una vez liberada la tecla del sintetizador. En el caso de un órgano en una iglesia (tomando como instrumento el conjunto del órgano y la iglesia, con su *reverb*), sería el tiempo que pasa entre que el organista deja de pulsar una tecla y el sonido de la iglesia se atenúa totalmente, que puede tardar unos segundos.

Con los parámetros por defecto del «Volume Filter» se obtiene un sonido seco, con un ataque muy rápido, un nivel de mantenimiento bastante alto y una extinción también inmediata. Es la respuesta más directa de pulsar una tecla: cuando se aprieta, suena, y cuando se deja de apretar, desaparece.

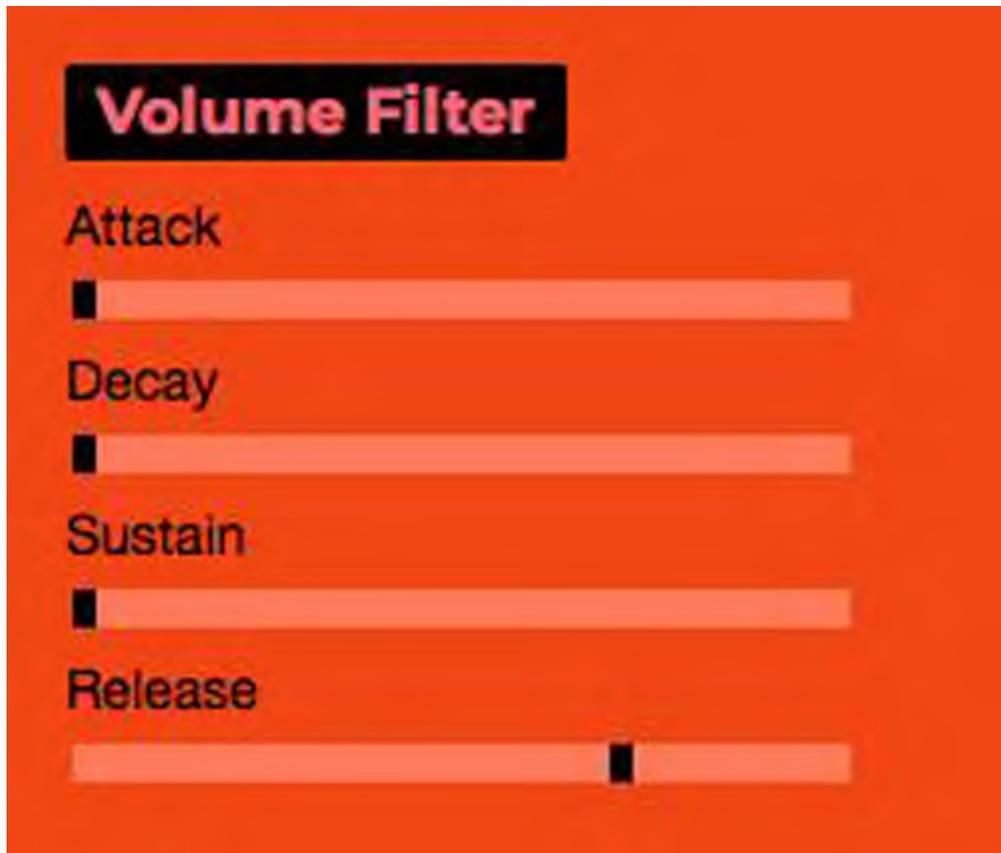


Figura 37. Configuración del «Volume Filter» para obtener un sonido percusivo.

Si se modifica la situación inicial y se propone un decaimiento y un mantenimiento nulos y una extinción larga, el comportamiento será como el de un instrumento percusivo: solo hay ataque y extinción. La Figura 37 muestra la configuración propuesta.

Ejercicios módulo 3

Ejercicio 3

Enunciado

¿Qué es un oscilador controlado por voltaje? ¿Cuál es su función en un sintetizador analógico?

Ejercicios módulo 3

Ejercicio 3

Solución

Un oscilador controlado por voltaje (VCO) es un generador de forma de onda cuya frecuencia viene controlada por la amplitud (tensión, en Voltios) de una señal «externa» (a pesar de que puede ser generado por el propio sintetizador en otro módulo). La frecuencia de este oscilador puede ser a frecuencia de LFO (baja frecuencia) o a frecuencia de audio (frecuencia audible). En el primero de los casos, que es el más común, el VCO se usa para modular, por ejemplo, la frecuencia de una señal portadora en lo que se denomina **Síntesis FM**, modificando el timbre del sonido generado. En el segundo de los casos, la tensión dada modifica directamente la altura de la señal sintetizada, lo que permite un control más directo de la frecuencia percibida. La figura 38 muestra un ejemplo de VCO.



Figura 38. Ejemplo de VCO virtual.

Fuente: imagen de synthesizeracademy.com/

Cuando se usa como modulador (ya sea de frecuencia o de amplitud, o hacia algún parámetro de un generador de envolvente ADSR), la amplitud de la señal del VCO se suele denominar **Depth** (o profundidad), puesto que indica con qué intensidad se modula el parámetro. Por ejemplo, si se usa para una modulación en frecuencia de un vibrato, la amplitud del LFO determina si el vibrato va medio, uno o dos semitonos arriba y abajo. Por otro lado, la frecuencia de la señal del VCO se suele denominar **Rate**, puesto que indica a qué ritmo se produce esta modulación. Por ejemplo, si se usa para hacer una modulación en frecuencia de un vibrato, la frecuencia del LFO determina cuántas subidas y bajadas de la frecuencia fundamental por segundo tendrá el sonido final.

Finalmente, cabe decir que estas modulaciones pueden ser sinusoidales (en el ejemplo anterior sonaría como un vibrato de un cantante), con señales cuadradas (se podría percibir un poco como el sonido de una sirena, a dos frecuencias diferentes y constantes), etc.

Ejercicios módulo 3

Ejercicio 4

Enunciado

¿Qué es un LFO? ¿En la síntesis aditiva, los diferentes armónicos son generados por LFOs?

Ejercicios módulo 3

Ejercicio 4

Solución

En un sintetizador, un LFO (*Low Frequency Oscillator*) es un oscilador que genera ondas (generalmente, sinusoidales, triangulares o cuadradas) a frecuencias más bajas que las frecuencias audibles. Estas señales de baja frecuencia se usan generalmente para modular otros parámetros, tales como una amplitud de otra señal (creando, por ejemplo, un efecto de Trémolo) o la frecuencia (creando, por ejemplo, un efecto de Vibrato o una modulación FM para conseguir diferentes timbres de un sonido). Generalmente, la amplitud de la señal proveniente del LFO se conoce como *Depth* de la modulación, mientras que la frecuencia se conoce como *Rate*. En función de la forma de onda, el *Depth* y el *Rate*, las posibilidades de generar diferentes sonidos usando LFOs son infinitas. La Figura 39 muestra un detalle de un sintetizador donde aparece un LFO.



Figura 39. Detalle de un sintetizador con un LFO.

Fuente: imagen de blog.landr.com

Los LFOs, a pesar de que pueden llegar a generar frecuencias de hasta 100Hz –es decir, frecuencias que ya entran dentro del margen audible–, no se utilizan en la síntesis aditiva. Se usan para controlar parámetros de los osciladores, pero no como osciladores propiamente dichos, que son los que se usan en la síntesis aditiva, generalmente sintonizados a frecuencias múltiples de una fundamental.

Ejercicios módulo 3

Ejercicio 5

Enunciado

¿Es posible generar el sonido de un oboe con sintetizadores de síntesis aditiva? ¿Y de síntesis por tabla de ondas?

Ejercicios módulo 3

Ejercicio 5

Solución

A priori, la respuesta a las dos preguntas planteadas sería afirmativa. Ahora bien, se pueden matizar bastantes detalles.

Si se analiza el espectro del sonido de una nota del oboe se observa que es un sonido rico en armónicos y que los tiene todos: pares y nones. La figura 40 muestra el espectro simplificado del sonido de una nota de oboe.

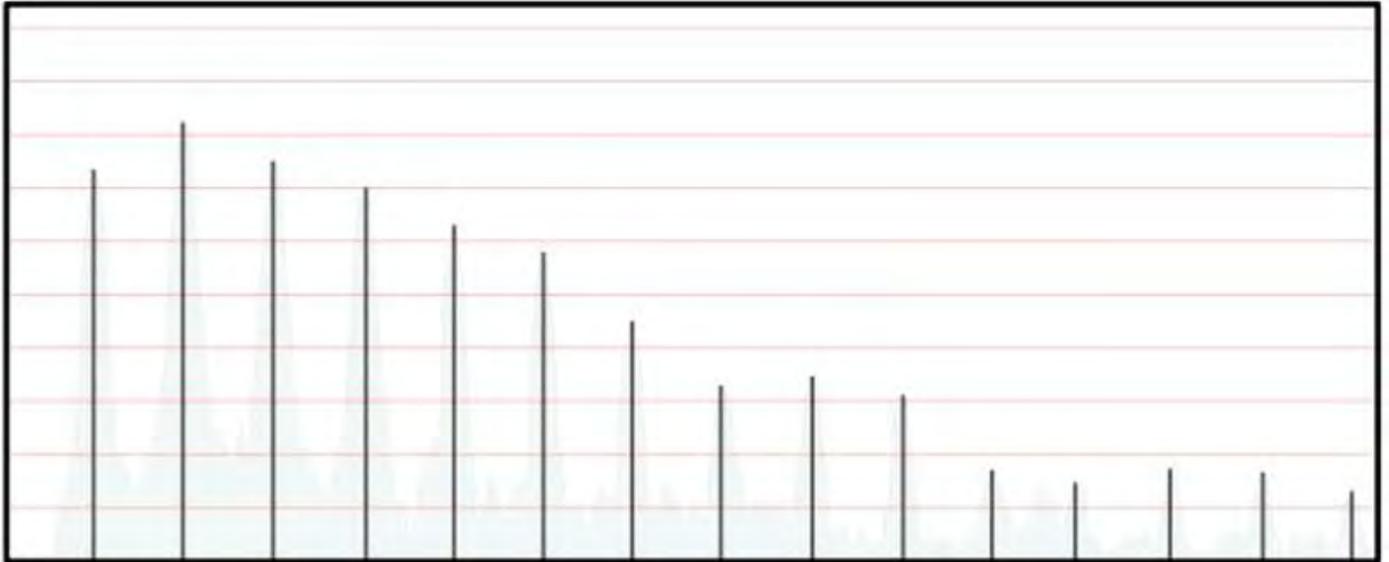


Figura 40. Espectro simplificado de la nota de un oboe tocando C5, *mezzo-forte*.

Fuente: Paul Clift, Adrien Mamou-Mande, René Caussé (2015). «Extending Brass & Woodwind instruments with acoustic – aggregate – synthesis». Proc. of the International Computer Music Conference (ICMC). Denton, TX, USA.

A modo de curiosidad y como pasa con muchos instrumentos, se observa como la fundamental no es la componente frecuencial con una mayor amplitud.

A pesar de que la amplitud de los armónicos decrece rápidamente, hay que sintetizarlos todos, desde la fundamental hasta un número suficientemente elevado, según la calidad final deseada. Por lo tanto, hace falta un número elevado de osciladores. Hoy en día es difícil, y es necesario poder disponer de muchos osciladores y sincronizarlos, salvo que se haga con un ordenador. La potencia de cálculo es suficientemente elevada como para generar el sonido de unos 15 o 20 osciladores, todos a frecuencias y amplitudes diferentes. Pero esto no quita que sea un proceso computacionalmente caro. Se puede conseguir, sí, pero consumirá bastantes recursos. A su favor tiene que la amplitud de los armónicos decae rápidamente y, de este modo, a pesar de que se deban sintetizar los armónicos pares y nones, bastará con unos «pocos» osciladores.

Por otro lado, para sintetizar el sonido del oboe mediante una mesa de onda hay que grabarlo y almacenar un (o más de un) periodo de la forma de onda y leerlo repetidamente (modo bucle) a una velocidad que dependerá de la nota deseada: si la nota que debe sonar es más aguda, la velocidad de lectura será más elevada, puesto que la frecuencia de la fundamental es más elevada). Se puede grabar desde un solo periodo (teoría de la síntesis portable de ondas entendida al pie de a el letra) hasta muchos periodos. En este caso, si por ejemplo el sonido grabado tiene una duración de 0,5s, es que en lugar de un sintetizador por tabla de ondas se está usando un *sampler*. Sea como fuere, con esta técnica se obtiene un sonido más realista que el anterior, y de forma más económica desde el punto de vista computacional. Ahora bien, no se tiene ningún control sobre el espectro en sí en caso de que se quieran implementar pequeñas modificaciones al espectro en función del volumen de la nota, hecho que pasa en la realidad y sí que se puede hacer fácilmente mediante la síntesis aditiva.

Como conclusión, cabe decir que el sonido de un oboe se puede conseguir con las dos técnicas propuestas por el enunciado, y que las dos tienen ventajas e inconvenientes. Aun así, quizás la mejor manera de sintetizar un sonido de oboe es con mediante la síntesis sustractiva o, incluso, mediante la síntesis por FM. Pero esa es harina de otro costal.

Ejercicios módulo 4

Ejercicio 1

Enunciado

Comenta un caso real de música objetiva y de música subjetiva de algún fragmento de película. Descríbelo brevemente y justifica la respuesta. (**Nota:** Los casos no pueden ser inventados. Debes citar la escena de la serie o película.)

Ejercicios módulo 4

Ejercicio 1

Solución



Figura 41. Fotograma de la película *El puente sobre el río Kwai*.

La música objetiva es la música que forma parte directa de la acción de una escena. Como ejemplo, se puede citar la escena de la película *El puente sobre el río Kwai* en la que se visualiza la entrada al campo de concentración japonés de las tropas del Coronel Nicholson mientras silban la conocida melodía. Se observa como la música se integra perfectamente a la acción y forma parte de la misma. La figura 41 muestra un fotograma de la escena propuesta.



Figura 42. Fotograma de la serie *Black Mirror*.

La música subjetiva es la música que acompaña al espectador y ayuda a situarle dentro de la escena, pero no forma parte directa de la acción. En este caso se puede tomar el ejemplo de la serie *Black Mirror*, en el primer episodio de la tercera temporada, cuyo título original es *Nosedive*, que en castellano se ha traducido por *Caída en picado*. En esta escena se observa la protagonista en una terraza pidiendo un café, tomándose un café y haciéndole una foto para compartirla en las redes sociales. La música que suena no está vinculada en ningún caso a la acción, y el único objetivo es crear un clima de falsa felicidad. La figura 42 muestra un fotograma de la escena propuesta.

Trecho do Episódio 01 da 3ª Temporada de Black Mirror



Ejercicios módulo 4

Ejercicio 2

Enunciado

Comenta un caso real de efecto objetivo y efecto subjetivo de algún fragmento de película. Descríbelo brevemente y justifica la respuesta. (**Nota:** Los casos no pueden ser inventados. Debes citar la escena de la serie o película.)

Ejercicios módulo 4

Ejercicio 2

Solución



Figura 43. Escena de la película *El bueno, el feo y el malo*.

Los efectos de sonido objetivos son los que forman parte directa de la acción de una escena. En la película *El bueno, el feo y el malo*, en la escena del duelo final, los ruidos de las pistolas –tanto los disparos como los ruidos de los cargadores– son efectos objetivos, puesto que forman parte de la acción directa de la escena. En teoría, los oiríamos si estuviéramos presentes en la escena real, a pesar de que quizás se han incorporado a la banda sonora *a posteriori*, en el estudio. Pero conceptualmente son efectos objetivos. La figura 43 muestra un fotograma de la escena propuesta.





Figura 44. Escena de la película *Lo imposible*.

Los efectos de sonido subjetivos no son consecuencia directa de una acción de una escena. En la película *Lo imposible* (*), justo antes de que llegue el tsunami se oyen unos sonidos graves que, a pesar de no surgir de la escena, indican al espectador que algo muy gordo está a punto de suceder. La figura 44 muestra un fotograma de la escena propuesta.



Ejercicios módulo 4

Ejercicio 3

Enunciado

Comenta la diferencia entre los conceptos de banda sonora, banda musical y banda internacional.

Ejercicios módulo 4

Ejercicio 3

Solución

La banda sonora propiamente dicha es la pista de audio donde se combinan todos los elementos de la producción, incluyendo la música, los efectos, las voces y otros elementos sonoros. La banda musical, estrictamente hablando, es la pista de audio que contiene solo la música. Sin embargo, es habitual confundir la banda musical con lo que coloquialmente denominamos como la banda **sonora de la producción**. Finalmente, la banda internacional hace referencia a la unión de las pistas de música, los efectos de sonido y otros elementos sonoros excluyendo la voz. Dicho de otro modo, en la banda internacional hay todos los elementos que componen la banda sonora excepto las voces, puesto que cada país incluye las suyas en el proceso de doblaje.

Ejercicios módulo 4

Ejercicio 4

Enunciado

¿Qué es y qué parámetros tiene en cuenta una licencia Creative Commons?

Ejercicios módulo 4

Ejercicio 4

Solución

La licencia Creative Commons protege la obra artística teniendo en cuenta los siguientes criterios, que pueden ser incluidos o no según la voluntad del autor:

- **Reconocimiento:** Debe reconocerse la autoría del autor. Este parámetro siempre debe estar presente.
- **Uso comercial:** El autor puede especificar si permite que su obra sea usada o no con finalidades comerciales.
- **Obras derivadas:** El autor puede especificar si permite la creación de nuevas obras partiendo de la suya, o si hay que mantener la obra original inmodificable.
- **Compartir igual:** en caso de permitir obras derivadas, el autor puede especificar si estas deben tener el mismo tipo de licencia creative commons o no.

Ejercicios módulo 4

Ejercicio 5

Enunciado

Enumera diferentes maneras de obtener la música para un producto audiovisual y comenta las ventajas y los inconvenientes de cada una de ellos.

Ejercicios módulo 4

Ejercicio 5

Solución

Los efectos de sonido se pueden obtener a partir de:

- librerías de sonido,
- grabaciones de ambiente en directo, o
- efectos reproducidos o sintetizados *ad-hoc*.

Las librerías de sonido presentan la facilidad de poder ser usadas indiscriminadamente sin tener que grabar previamente y, al mismo tiempo, los sonidos están muy clasificados. Como inconvenientes, se puede remarcar que los sonidos de la librería son limitados y hay que pagar una licencia.

Las grabaciones en directo tienen la ventaja de poder corresponder al mismo escenario donde se ha rodado una escena, lo que crea un ambiente más realista. Sin embargo, hay que grabar el efecto y procesarlo, hecho que es más lento que usar un sonido de librería. Cabe decir que en las librerías también hay grabaciones de ambiente en directo.

Finalmente, los efectos reproducidos o sintetizados cuentan con la ventaja de no tener límites en cuanto a la creatividad, pero el proceso para la obtención de estos sonidos puede llegar a ser muy tedioso. Es relativamente fácil grabar y crear el efecto de la fricción de la tela de una persona al sentarse en un sofá, pero puede ser muy difícil sintetizar el sonido de un coche en movimiento de forma realista.

Ejercicios módulo 4

Ejercicio 6

Enunciado

Explica dos situaciones en las que sea necesario recurrir al doblaje por la necesidad de aplicarle algún efecto de sonido.

Ejercicios módulo 4

Ejercicio 6

Solución

El primero podría ser la grabación de una conversación telefónica: si bien uno de los dos interlocutores puede ser grabado en directo, el segundo puede ser grabado *a posteriori*, durante un proceso de doblaje, y se le puede aplicar un filtro telefónico para dar realismo a la escena. Otro caso podría ser la simulación de dos interlocutores que están a una cierta distancia. En este caso limitaremos el nivel de sonido y quizás aplicaremos un filtro para simular este efecto.

Ejercicios módulo 5

Ejercicio 1

Enunciado

¿Qué diferencias conceptuales hay entre SoundCloud y Audiotool?

Ejercicios módulo 5

Ejercicio 1

Solución

Son dos herramientas muy diferentes. Audiotool es una herramienta en línea que sirve para crear piezas musicales a partir de sintetizadores y muestras de audio, así como compartirlas a través de la red. En cambio, SoundCloud está pensado únicamente para compartir audio en la red, especialmente en la redes sociales. Dicho de otro modo, Audiotool se centra en el proceso de creación (que puedes compartir), mientras que SoundCloud se centra en el acto de compartir el audio.

Ejercicios módulo 5

Ejercicio 2

Enunciado

¿Qué diferencias conceptuales hay entre SoundCloud y Giss.tv?

Ejercicios módulo 5

Ejercicio 2

Solución

Son dos herramientas muy diferentes. Soundcloud es una herramienta en línea que permite compartir audio (producido *a priori*) en la red, especialmente en las redes sociales. En cambio, Giss es una herramienta pensada para reproducir audio a tiempo real mediante *streaming*, sin dedicar demasiados esfuerzos a la redes sociales.

Ejercicios módulo 5

Ejercicio 3

Enunciado

¿Cómo subirías una colección de 10 sonidos de 5 segundos en una página web?

Ejercicios módulo 5

Ejercicio 3

Solución

Para subir sonidos a la red hay diferentes opciones, como por ejemplo subirlas directamente al servidor, SoundCloud, o usar un servidor de *streaming*. Dado que son sonidos cortos, la idea del *streaming* pierde bastante, y podríamos debatir entre subirlos directamente a la web o ponerlos en SoundCloud. Si el usuario a quién va destinada la web no está acostumbrado a navegar por SoundCloud y no hay mucho tráfico en el servidor, quizás sea mejor subirlos directamente al servidor usando las herramientas que nos brinda el HTML5 (insistiendo en la corta duración de los archivos). Si prevemos mucho tráfico o se trata de archivos más grandes, SoundCloud sería la opción perfecta.

Ejercicios módulo 5

Ejercicio 4

Enunciado

Propone tres inconvenientes de usar archivos MP3 en lugar de WAV en el tratamiento de audio.

Ejercicios módulo 5

Solución

Solución

A pesar de que la principal ventaja del formato MP3 es su reducido tamaño, debido al uso de algoritmos de compresión de audio, no siempre son todas ventajas.

En primer lugar, la principal desventaja es que el formato MP3 es un formato con pérdidas. Es cierto que, en las codificaciones más elevadas (por ejemplo, a partir de 256 kbps), estas pérdidas son prácticamente despreciables, pero en estos casos el nivel de compresión es más bajo. En resumen: cuanto más comprimido, peor calidad. Asimismo, otro inconveniente es que el proceso de compresión y descompresión requiere un tiempo importante. Es cierto que comprimir o descomprimir un tema con los ordenadores de hoy en día es muy rápido, pero hay que pensar en qué sucede durante una sesión de grabación, donde es posible que haya 32 o más pistas de audio sonando a la vez y, al mismo tiempo, el ordenador esté reproduciendo un vídeo de manera sincrónica. En estos casos en los que el ordenador funciona al límite de su capacidad, es mejor no trabajar con ficheros MP3. Por último, debemos tener en cuenta que el formato MP3 es un formato propietario, es decir, no es un código abierto ni gratuito y, hasta hace poco, era necesario abonar unas tasas a sus creadores (Fraunhofer Institute) para distribuir música en este formato. Sin embargo, los derechos sobre la patente expiraron el pasado 16 de abril del 2017, así que ya no es necesario pagar tasas.

Ejercicios resueltos módulo 7

Ejercicio 1

Enunciado

En este ejercicio implementaremos un programa con Processing y Minim que permita experimentar con las interferencias constructivas y destructivas. Para hacer esto, empezaremos generando una señal sinusoidal de frecuencia 440 Hz que no será la misma en los dos canales:

1. En el canal izquierdo, este tono puro tendrá la frecuencia fija de 440Hz y amplitud 0,5. Podéis usar el objeto «Pan» para dirigir este tono puro al canal izquierdo.
2. En el canal derecho, el tono puro también tendrá la frecuencia de 440Hz, pero la amplitud podrá ir desde 0 a 0,5 según el movimiento vertical del ratón. También controlaremos la fase de este tono puro con el movimiento horizontal del ratón. El método «setPhase» del objeto «Oscil» puede ayudaros a configurar este comportamiento.
3. La aplicación debe mostrar simultáneamente tres formas de onda: una para cada canal y la suma de las dos señales.

Nota: En este ejercicio y en otros ejercicios de Processing, es probable que las formas de onda se muevan muy rápidamente y sea difícil apreciar su forma. Para corregir este comportamiento, añadid al programa una función «KeyPressed» y modificad ligeramente la frecuencia de los dos tonos puros en pasos de 0,5 o 1 Hz hacia arriba o hacia abajo con las teclas «+» y «-» . Estos pequeños cambios de frecuencia harán que la imagen quede más estable en la pantalla.

Ejercicios resueltos módulo 7

Ejercicio 1

Solución

```
/**
 * Minim. Ejercicio resuelto nº 1
 * Francesc Martí, martifrancesc@uoc.edu, 27-09-2018
 *
 */

// Importamos los paquetes necesarios de la biblioteca Minim
import ddf.minim.*;
import ddf.minim.ugens.*;
import ddf.minim.spi.*;

// Definimos los objetos y las variables del programa
Minim minim;
AudioOutput out;
Oscil wave1, wave2;
Pan pan1, pan2;
float ampWave2, phaseWave2;
float freq;
PFont f;

void setup() {
  size(512, 360);

  // Inicializamos los objetos y las variables
  minim = new Minim(this);
  out = minim.getLineOut(Minim.STEREO, 512);
  freq = 440.0;
  wave1 = new Oscil( freq, 0.5, Waves.SINE);
  wave2 = new Oscil( freq, 0.5, Waves.SINE);

  // Los objetos Pan nos permitirán enviar uno de los osciladores al canal derecho
  // y el otro al canal izquierdo
  pan1 = new Pan(-1);
  pan2 = new Pan(1);

  // Conectamos los objetos
  // Enviamos una señal al canal derecho y la otra al canal izquierdo
  wave1.patch(pan1).patch(out);
  wave2.patch(pan2).patch(out);

  // Creamos la fuente que usaremos para mostrar texto en la ventana del programa
  f = createFont("Arial", 12, true);
  textFont(f);

  fill(0);
}

void draw() {
```

```

background(255);

// El texto a mostrar en la ventana del programa
text("Canal izquierdo", 10, 20);
text("Canal derecho", 10, 120);
text("Suma de los dos canales", 10, 220);
// Dibujo de las formas de onda
for (int i = 0; i < out.bufferSize() - 1; i++)
{

// Dibujo del canal izquierdo
line(i, 60 + out.left.get(i)*50, i+1, 60 + out.left.get(i+1)*50);
}
for (int i = 0; i < out.bufferSize() - 1; i++)
{
// Dibujo del canal derecho
line(i, 160 + out.right.get(i)*50, i+1, 160 + out.right.get(i+1)*50);
}
for (int i = 0; i < out.bufferSize() - 1; i++)
{
// Dibujo de la suma del canal izquierdo y derecho
line(i, 280 + out.mix.get(i)*100, i+1, 280 + out.mix.get(i+1)*100);
}
}

// Con el ratón controlaremos la amplitud y la fase de wave2
void mouseMoved()
{
// Control de la amplitud de la señal wave2
ampWave2 = map (mouseY, 0, height, 0.5, 0);
wave2.setAmplitude(ampWave2);

// Control de la fase de la señal wave2
phaseWave2 = map (mouseX, 0, width, 0.5, 1);
wave2.setPhase(phaseWave2);
}

// Desde el teclado modificaremos ligeramente la frecuencia de los dos tonos puros.
// Esto permitirá estabilizar la imagen de las formas de onda en la pantalla
void keyPressed()
{
switch(key)
{
case '+':
freq=freq+1;
break;
case '-':
freq=freq-1;
break;
default:
break;
}
// Aplicamos los cambios a las dos ondas
wave1.setFrequency(freq);
wave2.setFrequency(freq);
}

```


Ejercicios resueltos módulo 7

Ejercicio 2

Enunciado

En este ejercicio realizaremos un programa que permita al usuario experimentar y entender el concepto de sensación de sonoridad. Generaremos un programa que emita un tono puro siempre con la misma intensidad, pero con una frecuencia diferente. Así el usuario podrá percibir cómo va variando la sensación sonora de sonidos con el mismo nivel de intensidad.

Para realizar el programa seguiremos los siguientes pasos:

1. Generamos un tono puro de frecuencia 1.000 Hz y amplitud 0,5.
2. La frecuencia de este tono puro la controlaremos con el movimiento horizontal del ratón, pudiendo variar desde 20 Hz (cuando el puntero del ratón esté a la izquierda de la ventana programa) hasta 20 Khz (cuando el puntero esté a la derecha).
3. La aplicación mostrará la forma de onda de la señal que se genere. Análogamente al primer ejercicio anterior, añadiremos en el programa una función «KeyPressed» que permita controlar el movimiento del gráfico de la forma de onda con las teclas «+» y «-» .
4. Finalmente, la aplicación también deberá mostrar el espectro de la señal. Esto permitirá al usuario ver que señales con el mismo nivel de intensidad tienen sonoridades diferentes.

Ejercicios resueltos módulo 7

Ejercicio 2

Solución

```
/**
 * Minim. Ejercicio resuelto nº 2
 * Francesc Martí, martifrancesc@uoc.edu, 27-09-2018
 *
 */

// Importamos los paquetes necesarios de la biblioteca Minim
import ddf.minim.*;
import ddf.minim.ugens.*;
import ddf.minim.spi.*;
import ddf.minim.analysis.*;

// Definimos los objetos y las variables del programa
Minim minim;
AudioOutput out;
Oscil wave;
float freq;
FFT fft;
PFont f;

void setup() {
  size(512, 450);

  // Inicializamos los objetos y las variables
  minim = new Minim(this);
  out = minim.getLineOut(Minim.STEREO, 512);
  freq = 1000;
  wave = new Oscil( freq, 0.5, Waves.SINE);
  fft = new FFT( out.bufferSize(), out.sampleRate());

  // Conectamos los objetos
  wave.patch(out);

  // Creamos la fuente que usaremos para mostrar texto en la ventana del programa
  f = createFont("Arial", 12, true);
  textFont(f);
  fill(0);
}

void draw() {
  background(255);

  // Dibujaremos la forma de onda y espectro en azul

  stroke(0, 0, 255);

  // El texto a mostrar en la ventana del programa
  text("Tono puro con intensidad constante: 0.5" , 10, 20);
```

```

text("Frecuencia: " + freq + " Hz.", 10, 36);

// Dibujo de la forma de onda
strokeWeight(1);
for (int i = 0; i < out.bufferSize() - 1; i++)
{
line(i, 180 + out.mix.get(i)*100, i+1, 180 + out.mix.get(i+1)*100);
}

// Esta función analizará y dibujará los espectros
drawFTP();
}

void drawFTP() {
// Hacemos el análisis fft sobre el buffer mix
fft.forward(out.mix);

// Dibujamos el espectro
strokeWeight(2);
for (int i = 0; i < fft.specSize(); i++)
{
line( 2.2*i, height, 2.2*i, height - fft.getBand(i) );
}
}

// Con el ratón controlaremos la frecuencia de la señal
void mouseMoved()
{
// Control de la frecuencia de la onda
// Esta puede ir de los 20 Hz hasta los 20 KHZ
freq = map(mouseX, 0, width, 20, 20000);
wave.setFrequency(freq);
}

// Desde el teclado modificaremos ligeramente la frecuencia de los dos tonos puros.
// Esto permitirá estabilizar la imagen de las formas de onda en la pantalla
void keyPressed()
{
switch(key)
{
case '+':
freq=freq+1;
break;
case '-':
freq=freq-1;
break;
default:
break;
}
wave.setFrequency(freq);
}
}

```


Ejercicios resueltos módulo 7

Ejercicio 3

Enunciado

En este ejercicio diferenciaremos entre los sonidos armónicos y el ruido. La diferencia principal es que los sonidos armónicos tienen claramente definida una frecuencia fundamental y unos armónicos, mientras que en el ruido no hay ninguna frecuencia predominante ni ningún contenido armónico con cierta relevancia.

Para visualizar este fenómeno abriremos dos archivos ya pregrabados: uno con una melodía armónica y el otro con un ruido:

1. Usando el objeto `AudioPlayer`, abre los ficheros «`music_box`» y «`noise.wav`» y reproducélos a la vez.
2. Con el movimiento horizontal del ratón, controla el volumen de los dos archivos: en el extremo izquierdo sonará sólo el fichero «`music_box`», mientras que en el extremo derecho sonará sólo el fichero «`noise.wav`».
3. Dibuja por pantalla las formas de onda y los espectros de los dos sonidos en relación a lo que está sonando. Es decir, cuando el ratón se encuentre en el extremo izquierdo de la ventana se deberá visualizar solo la forma de onda y espectro del sonido «`music_box`», mientras que cuando el ratón se encuentre en el extremo derecho, solo se deberá visualizar la forma de onda y espectro del sonido «`noise.wav`».

Ejercicios resueltos módulo 7

Ejercicio 3

Solución

```
/**
 * Minim. Ejercicio resuelto n° 3
 * Francesc Martí, martifrancesc@uoc.edu, 28-09-2018
 *
 */

// Importamos los paquetes necesarios de la biblioteca Minim
import ddf.minim.*;
import ddf.minim.ugens.*;
import ddf.minim.spi.*;
import ddf.minim.analysis.*;

// Definimos los objetos y las variables del programa
Minim minim;
AudioPlayer player1, player2;
float amp1, amp2, drawControl;
FFT fft1, fft2;
PFont f;

void setup() {
  size(512, 350);

  // Inicializamos los objetos y las variables
  minim = new Minim(this);
  player1 = minim.loadFile("music_box.wav");
  player2 = minim.loadFile("noise.wav");
  fft1 = new FFT( player1.bufferSize(), player1.sampleRate() );
  fft2 = new FFT( player2.bufferSize(), player2.sampleRate() );

  // Creamos la fuente que usaremos para mostrar texto en la ventana del programa
  f = createFont("Arial", 12, true);
  textFont(f);
  fill(0);

  // Reproducimos los dos archivos de audio en modo "loop", ajustando los volúmenes iniciales a
  player1.setGain(-35);
  player2.setGain(-35);
  player1.loop();
  player2.loop();
}

void draw() {
  background(255);
  stroke(0);

  // El texto a mostrar en la ventana del programa
  text("music_box.wav (espectro rojo)", 10, 20);
```

```

text("noise.wav (espectro verde)", 10, 120);

// Dibujo de las formas de onda
strokeWeight(1);
for (int i = 0; i < player1.bufferSize() - 1; i++)
{
//Espectro rojo para el player1
stroke(255, 0, 0);
line( i, 60 - drawControl*player1.mix.get(i)*50, i+1, 60 - drawControl*player1.mix.get(i+1)
}
for (int i = 0; i < player2.bufferSize() - 1; i++)
{
// Espectro verde por el player2
stroke(0, 255, 0);
line( i, 160 - (-drawControl+1)*player2.mix.get(i)*50, i+1, 160 - (-drawControl+1)*player2.mi
}

// Esta función analizará y dibujará los espectros
drawFTP();
}

void drawFTP() {

// Hacemos los análisis fft sobre los buffers mix
fft1.forward(player1.mix);
fft2.forward(player2.mix);

// Dibujamos el espectro
for (int i = 0; i < fft1.specSize(); i++)
{
//Espectro rojo para el player1
stroke(255, 0, 0);
strokeWeight(2);
line( 2.2*i, height, 2.2*i, height - 2*drawControl*fft1.getBand(i) );

// Espectro verde por el player2
stroke(0, 255, 0);
line( 2.2*i, height, 2.2*i, height - 2*(-drawControl+1)*fft2.getBand(i) );
}
}

// Con el ratón controlaremos las amplitudes de los dos sonidos
void mouseMoved()
{
amp1 = map( mouseX, 0, width, -80, 10);
player1.setGain(amp1);
amp2 = map( mouseX, 0, width, 10, -80);
player2.setGain(amp2);

// Esta variable nos ayuda a hacer los dibujos en función de la
// posición del ratón
drawControl = map( mouseX, 0, width, 0, 1);
}

```



Ejercicios propuestos módulo 7

Ejercicio 1

Con la ayuda de la librería «Mínimo de Processing», hay que programar una aplicación que implemente la síntesis sustractiva. Para realizar esta tarea seguiremos los siguientes pasos:

1. Con un objeto de tipo Noise, genera ruido blanco con amplitud 0,5.
2. Con las teclas «1», «2» y «3», el usuario debe poder escoger el tipo de ruido que generará el objeto «Noise». Los tipos de ruido disponibles serán blanco, rosa y marrón.
3. La amplitud del ruido se controlará con las teclas «4» (disminuyendo) y «5» (aumentando), y podrá variar entre 0 y 1.
4. A esta señal se le aplicará un filtro usando un objeto de tipo «MoogFilter».
5. Con las teclas «a», «s» y «d» podremos seleccionar el tipo de filtro: paso bajo, paso de banda o paso alto (por defecto, será un filtro de paso bajo).
6. La frecuencia de corte del filtro se controlará con las teclas «q» (disminuyendo) y «w» (aumentando), y podrá variar entre 200 y 1.000 Hz.
7. La resonancia del filtro se controlará con las teclas «z» (disminuyendo) y «x» (aumentando), y podrá variar entre 0 y 1.
8. Los valores del tipo de ruido generado y de la amplitud del ruido deben mostrarse en todo momento en la ventana de la aplicación. También se deberá mostrar el tipo de filtro, su resonancia y su frecuencia de corte.
9. Finalmente, además de emitir el sonido que se genera, el programa también deberá mostrar la forma de onda y espectro de la señal filtrada en la ventana de la aplicación.

Ejercicios propuestos módulo 7

Ejercicio 2

Desarrolla una aplicación que permita abrir, reproducir y aplicar un efecto «Vocoder» sobre un archivo de audio. Para realizar el programa seguiremos los siguientes pasos:

1. Graba con tu ordenador, tu teléfono o un dispositivo al que tengas acceso, con una calidad de 44.100 Hz, 16 bits y monofónico, la frase «Un sonido digitalizado no siempre tendrá más calidad cuanto mayor sea su resolución».
2. Usa el objeto «FilePlayer» de Minim para abrir y reproducir el archivo de audio generado.
3. Inicialmente, el archivo debe estar en modo «pausa», y el usuario podrá reproducir o parar el sonido con la tecla «p» de su teclado. Si el usuario no pausa la reproducción, el archivo de audio se deberá reproducir continuamente en modo «Loop».
4. Conecta la salida de audio del objeto «FilePlayer» a un objeto «Vocoder» (http://code.compartmental.net/minim/vocoder_class_vocoder.html).
5. Inicialmente el objeto «Vocoder» tendrá una longitud de ventana («Window size») de 1.024 muestras y un valor de solapamiento («Window count») de 8. El usuario podrá escoger valores de longitud de ventana de 128, 256, 512, 1.024 o 2.048 muestras, y valores de solapamiento 2, 4, 6, 8, 10, 12, 14 y 16. Se deja libertad al alumno para que decida qué teclas controlarán estas variables.
6. En la ventana de la aplicación se deberán mostrar simultáneamente los espectrogramas del sonido original y el sonido con el efecto del «Vocoder» aplicado, de forma que sea sencillo comparar sus diferencias.

Ejercicios propuestos módulo 7

Ejercicio 3

La aplicación a desarrollar en este ejercicio permitirá experimentar con el concepto de frecuencia fundamental y armónicos. La idea consiste en desarrollar un programa que permita añadir tonos puros a un tono puro inicial dado (que será la fundamental), de forma que el resultado tienda a una señal con forma triangular. Las características de la aplicación a diseñar son:

1. La aplicación generará inicialmente un tono puro «A» de frecuencia $f = 340\text{Hz}$, amplitud 0,4 y fase 0,5.
2. Si el usuario pulsa la tecla «1», a este tono puro «A» se le añadirá –sumará– otro tono puro. Si pulsa la tecla «2», al tono puro «A» se le añadirán dos tonos puros, y así sucesivamente hasta añadir 9 tonos puros al pulsar la tecla «9».
3. La relación entre el tono puro «A» y los tonos puros que vamos sumando debe ser tal que la forma de onda resultante tienda a una forma de onda triangular con el tono puro «A» como fundamental. Dicho de otro modo, al pulsar la tecla «1», la aplicación debe generar la fundamental y el primer armónico de una triangular con frecuencia fundamental $f = 340$. Al pulsar la tecla «2», la fundamental y los dos primeros armónicos, etc.
4. La aplicación debe mostrar en la ventana de la aplicación la forma de onda de la señal que se genera.
5. La aplicación también debe mostrar el espectro de la señal que se genera.
6. La aplicación debe incluir una función que permita estabilizar la imagen de la forma de onda con las teclas «+» y «-».

Recursos

Módulo 1

[M1_e05_0300hz_0050ms_-00dB.wav](#)

[M1_e05_0300hz_0050ms_-06dB.wav](#)

[M1_e05_0300hz_0050ms_-12dB.wav](#)

[M1_e05_0300hz_0250ms_-00dB.wav](#)

[M1_e05_0300hz_0250ms_-06dB.wav](#)

[M1_e05_0300hz_0250ms_-12dB.wav](#)

[M1_e05_0300hz_1000ms_-00dB.wav](#)

[M1_e05_0300hz_1000ms_-06dB.wav](#)

[M1_e05_0300hz_1000ms_-12dB.wav](#)

[M1_e05_5000hz_0050ms_-00dB.wav](#)

[M1_e05_5000hz_0050ms_-06dB.wav](#)

[M1_e05_5000hz_0050ms_-12dB.wav](#)

[M1_e05_5000hz_0250ms_-00dB.wav](#)

[M1_e05_5000hz_0250ms_-06dB.wav](#)

[M1_e05_5000hz_0250ms_-12dB.wav](#)

[M1_e05_5000hz_1000ms_-00dB.wav](#)

[M1_e05_5000hz_1000ms_-06dB.wav](#)

[M1_e05_5000hz_1000ms_-12dB.wav](#)

[M1_e06_440hz_400mV_000deg.wav](#)

[M1_e06_440hz_400mV_180deg.wav](#)

[M1_e06_440hz_constr.wav](#)

[M1_e06_440hz_destr.wav](#)

[M1_e07_0998hz_400mv_5s.wav](#)

[M1_e07_1000hz_400mv_5s.wav](#)

[M1_e07_1002hz_400mv_5s.wav](#)

[M1_e07_0998hz_400mv_5s.wav](#)

[M1_e07_1000hz_400mv_5s.wav](#)

[M1_e07_1002hz_400mv_5s.wav](#)

[M1_e07_0998+1000hz_5s.wav](#)

[M1_e07_0998+1002hz_5s.wav](#)

[M1_e07_1000+1002hz_5s.wav](#)

[M1_e08_0660hz_400mv_5s.wav](#)

[M1_e08_0680hz_400mv_5s.wav](#)

[M1_e08_0660+0680hz_400mv_5s.wav](#)

[M1_e09_gaussian+sweep_30s.wav](#)

[M1_e10_0500hz_400mv_5s_sine.wav](#)

[M1_e10_0500hz_400mv_5s_square.wav](#)

[M1_e10_0500hz_400mv_5s_saw.wav](#)

Módulo 2

[gravacio.aif](#)

[bateria_gravacio.wav](#)

[bateriaCorregida.wav](#)

[273177_xserra_la-vaca-cega-eva.wav](#)

[poemaA.wav](#)

[poemaB.wav](#)

[poemaC.wav](#)

[poemaD.wav](#)

[poemaE.wav](#)

[poemaF.wav](#)

[soBaixaResolucio.wav](#)

[344250_honest-cactus_normal-voice-and-intro_Saturation.wav](#)

[390514_tylean_counting-1-to-10.wav](#)

[177269_sergeo_numbers-in-french.wav](#)

[Sine1200Hz.wav](#)

[126444_harregarre_compilatie.mp3](#)

[332769_emanuele-correani_ambience-train-station-inside.wav](#)

[AmbientSoundClics.mp3](#)

[Ex3_original.wav](#)

[Ex3_modificat_1.wav](#)

[Ex3_modificat_2.wav](#)

[Ex3_modificat_3.wav](#)

[43555_dobroide_20071103-description-human.wav](#)

Módulo 7

[music_box.wav](#)

[noise.wav](#)

(*) Contenido disponible solo en web.