

[Área personal](#) / [Mis cursos](#) / [SMC 21028](#) / [Práctica 7: Procesadores del sonido](#) / [Cuestionario de la primera sesión P7 \(P\)](#).

Comenzado el	miércoles, 25 de marzo de 2020, 10:14
Estado	Finalizado
Finalizado en	miércoles, 25 de marzo de 2020, 13:09
Tiempo empleado	2 horas 55 minutos
Puntos	11,5/14,0
Calificación	8,2 de 10,0 (82%)

Información

Procesadores del sonido

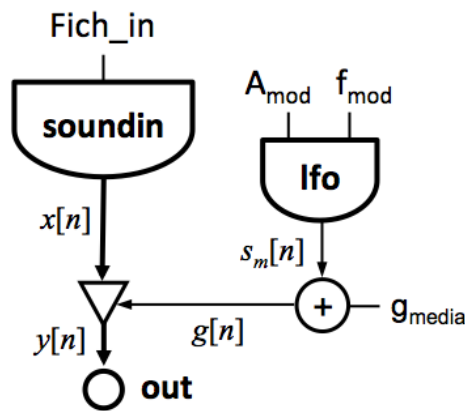
En esta práctica vamos a diseñar y programar en Csound procesadores de los diferentes parámetros psicoacústicos vistos en teoría. El material previo publicado es, de nuevo, muy importante para realizar estos ejercicios.

Información

Procesadores de la dinámica

Ejercicio 7.1: Trémolo

Vamos a implementar un trémolo que actúe sobre un sonido de entrada almacenado en un fichero, según el esquema que aparece a continuación.



Efecto sobre la envolvente del procesador de trémolo.

La señal $s[n]$ leída desde un fichero por **soundin** es modulada en amplitud por la ganancia variable $g[n]$. Esta ganancia es un valor variable (por tanto, de tipo k-) generado por el oscilador de baja frecuencia **lfo** que sumaremos a una ganancia media, lo cual podemos expresar como

$$g[n] = g_{media} + A_{mod} \sin(2\pi f_{mod}n / f_s)$$

que oscilará f_{mod} veces por segundo entre $g_{media} - A_{mod}$ y $g_{media} + A_{mod}$.

Condiciones para un buen diseño:

- $g_{media} + A_{mod}$ debe ser tal que la onda no se salga del rango de amplitudes; y
- $g_{media} - A_{mod} > 0$ siempre para que la ganancia sea siempre positiva.

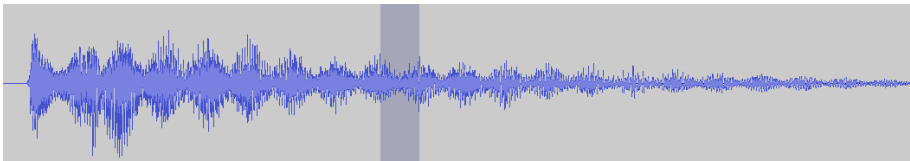
Desarrollo del instrumento:

- Utiliza como punto de partida la siguiente [PLANTILLA](#) (botón derecho → guardar como)
- Al instrumento **tremolo** le deberás suministrar desde la partitura, en este orden:
 - Una cadena con el nombre del fichero WAV a procesar (**p4**),
 - La ganancia media del efecto (en **p5**),
 - La amplitud de la modulación de la ganancia (en **p6**),
 - La frecuencia de esa modulación (**p7**), y
- Utiliza el esquema de arriba para implementar el instrumento.

¿Cómo saber si el efecto está bien aplicado?

Cuando tengas tu programa preparado, aplícalo sobre el sonido propuesto y abre con **Audacity** el fichero procesado, en el que se verá el sonido original y luego el procesado, en el que debe verse con claridad la modificación periódica de la envolvente (ver Figura abajo). Usarás esa visualización para calcular el periodo de modulación, como se indica.

Deberás seleccionar el audio procesado entre 2 máximos consecutivos de la modulación:



(recomendación: mide 3 veces el periodo en diferentes ciclos y quédate con un valor promedio)

Pregunta **1**

Finalizado

Puntúa 1,0
sobre 1,0

- Descarga el siguiente fichero para aplicarle el efecto:

0:00 / 0:00

- Tempo = 60 BPM
- Programa dos activaciones seguidas (sin pausas entre ellas) del instrumento **"tremolo"** :
 - Una primero para oír el sonido del fichero completo (8 segundos) sin procesar. Para esto, le pasarás una ganancia media = 0.8, y una amplitud y frecuencia de modulación = 0.
 - Crea otra activación al terminar la primera para aplicar el trémolo al sonido, con la misma ganancia media, amplitud de modulación = 0.4 y frecuencia 7.0 Hz.

Usa el procedimiento descrito anteriormente para medir el periodo de modulación en milisegundos y escríbe el resultado a continuación (sin decimales):

Respuesta: 147

Información

Ejercicio 7.2: Compresor y limitador

El objetivo es implementar un compresor con un umbral Θ_C y una tasa de compresión R_C .

- Descarga la plantilla del proyecto: [PLANTILLA](#) (botón derecho → guardar como)

El instrumento:

- Recibirá desde la partitura:
 - El nombre del fichero a procesar en **p4**.
 - El umbral de compresión: Θ_C en dBFS en **p5**, Debes convertir el valor recibido en dBFS a Θ_C en PCM. Una orden **print** imprime los valores del umbral en dBFS y el PCM calculado (ya está en la plantilla).
 - La tasa de compresión R_C en **p6** y
- Primero se leerá la señal del fichero con **soundin**.
- Calcula el valor de la amplitud de la señal con el valor absoluto (función **abs(·)**) y almacénalo en una variable de tipo **k-** para usarla en la condición y en el cálculo de la ganancia.
- El instrumento debe calcular la ganancia $g[n]$ (también en una variable tipo **k-**) a aplicar a cada muestra, según la ecuación de la ganancia en PCM dada en el material previo de la práctica.
- Una vez calculada, se aplica $y[n] = x[n] * g[n]$ y se escribe el valor procesado en la señal de salida.
- Sólo falta enviarla fuera.

El comportamiento acústico de estos procesadores es muy sutil, por lo que te pediremos que generes el WAV con **Render** y compares con **Audacity** las ondas, antes y después de actuar el compresor.

Pregunta **2**
Finalizado
Puntúa 2,0 sobre 2,0

Para la partitura:

- Comprueba su funcionamiento con el fichero

0:00 / 0:00

($A_{\text{pico}} = -0.27$ dBFS). La partitura proporcionada en la plantilla lleva una primera nota en la que no se aplica ninguna compresión.

- Añade una segunda nota (que empiece al terminar la primera) de 8 segundos para que aplique el compresor con un umbral $\Theta_C = -9$ dBFS y una tasa de compresión $R_C = 1.6:1$.

La compresión modifica el margen disponible. Para comprobarlo sigue los siguientes pasos en **Audacity**:

- Selecciona la onda sin comprimir y calcula el margen sin comprimir con **Efecto** → **Amplificar**. El valor de amplificación (dB) dirá la distancia entre pico y máximo posible (margen, positivo):

0,3 dB (con 1 decimal)

- Selecciona la onda comprimida y repite el procedimiento para conocer el nuevo margen:

3,5 dB (con 1 decimal)

Información

Uso como limitador

- La diferencia entre un compresor y un limitador es que suelen trabajar con umbrales mayores y con tasas muy grandes.
- Usaremos el mismo instrumento **compresor** pero ahora como un limitador.
- Para ello crea una nueva nota de 8 segundos que empiece cuando acabe la anterior y cambia los parámetros de la activación para procesar el mismo fichero de audio con los parámetros que se indican abajo.
- Comprobarás que ahora la señal procesada apenas superará el valor del umbral.

Pregunta **3**
Finalizado
Puntúa 2,0 sobre 2,0

- Los parámetros de la nueva activación serán: umbral $\Theta_L = -6.5$ dBFS y una tasa de limitación $R_L = 12:1$.
- Vuelve a hacer **Render** y abre el resultado con **Audacity**.
- Selecciona la onda limitada y calcula el margen igual que antes en esta nueva situación:

6,0 dB (con 1 decimal)

- ¿Cuánto margen hemos ganado al limitar la onda respecto al margen de la onda inicial?:

5,7 dB (con 1 decimal)

Información

Ejercicio 7.3: Expansor y puerta

Se trata de implementar un expansor, de una manera similar al anterior compresor.

- Para ello, haz una copia del proyecto anterior en un nuevo `ej7.3.csd`.

En la orquesta:

- Cambia en las opciones el fichero a generar para que sea `ej7.3.wav` y el nombre del instrumento, de **compresor** a **expansor**.
- La única diferencia en el nuevo instrumento es que debes cambiar el signo de la condición, porque estos procesadores actúan no por encima, sino *por debajo* del umbral.

En la partitura:

- Descarga el fichero

0:00 / 0:00

- Borra las notas del ejercicio anterior y escribe una primera nota para oír el fichero original:

i "expansor" 0 8.0 "percusion.wav" -96 [1/1.1] ; no aplica el expansor

- Cuando compile correctamente y oigas el sonido (que contiene ruido de fondo añadido), añade una segunda nota de 8 segundos al terminar la primera, para que active de nuevo el instrumento y procese el fichero usando el umbral de expansión $\Theta_E = -24$ dBFS y una tasa de expansión $R_E = 1:2.0$
- Compila y escucha el resultado, antes y después de la expansión.

Información

Problemas con la amplitud:

- Percibirás un efecto desagradable sobre el sonido procesado. Esto es debido a que la estimación de la amplitud con $|x[n]|$ es muy inestable (el expansor actúa cada vez que la señal se acerca al cero, en cada oscilación) lo que provoca que la reducción de ganancia se esté aplicando intermitentemente. Para evitar esto hay que cambiar $|x[n]|$ por un cálculo más estable de la envolvente:
 - Para ello, sustituye el cálculo de la amplitud con la función **abs(.)** por el uso del operador **rms**
- Compila ahora con **Render** y abre el resultado con **Audacity** para comprobar que el efecto se ha aplicado con más limpieza.
- La expansión aumenta la relación entre la señal y el ruido, manteniendo el volumen de la señal y reduciendo el del ruido. Comprueba si ha funcionado, viendo y oyendo las ondas.

Información

Uso como puerta de ruido

- Usa ahora el expansor como una puerta de ruido. La puerta hace lo mismo que el expansor pero más drásticamente.
- Para ello, crea en la partitura una nueva activación del instrumento **"expansor"** que empiece cuando acabe la anterior y de la misma duración, aplicada sobre el mismo fichero. Los parámetros de puerta están en la siguiente pregunta.
- Cuando esté terminado el programa, vuelve a hacer **Render** y a abrir el resultado con **Audacity**.
- Verás los 3 sonidos seguidos:
 - El sonido original,
 - El sonido procesado con el expansor, y
 - El sonido procesado con la puerta de ruido.
- Veremos si ha funcionado analizando la relación entre la señal y el ruido.

Pregunta **4**
Finalizado
Puntúa 2,2 sobre 4,0

Parámetros de la puerta

- Umbral de puerta $\Theta_P = -30$ dBFS y tasa de expansión $R_P = 1:21$

Relación señal/ruido del sonido original:

- Selecciona de la onda sin comprimir la zona entre 5,5 y 7,0 s (donde hay señal) (puedes ayudarte con la ventana de "inicio y final de la selección" en la parte inferior del interfaz de **Audacity**).
- Abre en el menú **Analizar** → **Contraste**.
- Pulsa el botón [Selección de medida] superior. Aparecerá el valor de amplitud RMS medio de esa selección.
- Sin cerrar esta ventana selecciona en la onda sin comprimir la zona entre 7,0 y 7,6 s (donde predomina el ruido de fondo).
- Pulsa el botón [Selección de medida] inferior. Aparecerá el valor de amplitud RMS medio de esa selección y la diferencia entre ambas amplitudes en la parte inferior.
- Indica los valores que has medido para la señal, el ruido y la relación señal / ruido:

Señal: dB – Ruido: dB = Diferencia: dB RMS

Relación señal/ruido tras la expansión:

- Repite este procedimiento sobre la señal expandida, con la zona de señal entre 13,5 y 15,0 s y la zona de ruido de fondo entre 15,0 y 15,6 s. Anota los nuevos valores, para ver qué ha ocurrido con el ruido y la relación señal/ruido:

Señal: dB – Ruido: dB = Diferencia: dB RMS

Relación señal/ruido tras la puerta:

- Repite este procedimiento sobre la señal puerteadada, con la zona de señal entre 21,5 y 23,0 s y la zona de ruido entre 23,0 y 23,6 s. Apunta a continuación los valores medidos para la señal, el ruido y la relación señal / ruido:

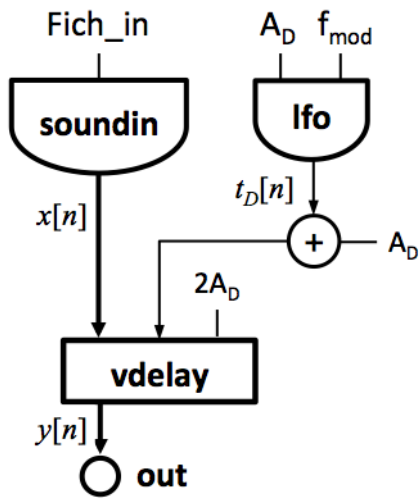
Señal: dB – Ruido: dB = Diferencia: dB RMS

Información

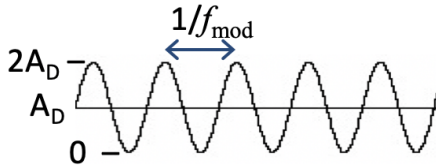
Procesadores de la altura

Ejercicio 7.4: Vibrato

Implementa un vibrato según el concepto explicado en el material previo de la práctica: aplicando un efecto Doppler sobre una onda mediante un retardo variable. El esquema en Csound de este procesador es el que se muestra en la figura.



Señal de modulación de t_D :



La señal de modulación del t_D que crea el **lfo** y oscila a f_{mod} Hz entre $-A_D$ y $+A_D$. Al sumarle A_D (ver figura) tendremos un tiempo de retardo variable $t_D[n]$ que oscilará entre 0 y $2A_D$ ms.

- Usa la siguiente plantilla para empezar: [PLANTILLA](#) (botón derecho → guardar como).
- La plantilla lleva programado un operador **frec_max** que aplicado a un sonido detecta máximos de frecuencia y los escribe en la consola.

En el instrumento:

- La señal $x[n]$ la carga **soundin** desde el fichero enviado en **p4**.
- La desviación máxima del tiempo de retardo A_D se recibirá en **p5** (en ms).
- La frecuencia de modulación del vibrato f_{mod} se recibirá en **p6** (en Hz).
- Añade al esquema una condición para que aplique el vibrato sólo a partir de que pase 1 segundo del sonido de entrada. Para ello tendrás que llevar a cabo dos acciones:
 1. tomar valores de tiempos con **timeinsts** (que empieza a contar desde cero en cada activación del instrumento) y
 2. aplicar un **if-then-else** para que:
 - si el tiempo es menor de 1 segundo, la variable de la modulación = 0
 - en caso contrario, la variable de modulación se calcula con **lfo** (según indica la figura).
- No olvides poner la variable de audio que estés utilizando en la entrada del operador **frec_max** para que haga las mediciones.

Pregunta **5**
Finalizado
Puntúa 2,0 sobre 2,0

En la partitura:

- Descarga el siguiente fichero para aplicarle el efecto:

0:00 / 0:00

- El tempo está establecido a 60 BPM.
- Programa dos activaciones seguidas (sin pausas entre ellas) del instrumento **"vibrato"** :
 - Una primero para oír el sonido del fichero completo (5 segundos) sin procesar. Para esto, le pasarás una amplitud máxima de desviación de 0.1 ms y una frecuencia de modulación = 0 Hz.
 - Crea otra activación al terminar la primera para aplicar el vibrato al sonido, con una amplitud máxima de desviación de 0.9 ms y una frecuencia de modulación = 5 Hz.
- Compila en tiempo real y escucha para comparar el sonido con y sin la modulación aplicada.

Usa el medidor de frecuencia ya programado para obtener la frecuencia máxima (el último valor que se escriba en la consola) alcanzada por el vibrato (sin decimales):

Frecuencia sin vibrato: Hz → Frecuencia máxima con vibrato: Hz

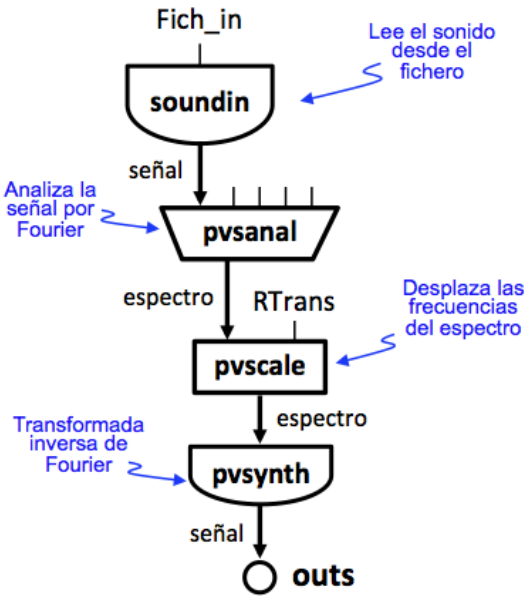
Ejercicio 7.5: Transposición

En este ejercicio se trata de leer un sonido y transponer su altura en un intervalo dado. Lo vamos a implementar mediante la técnica del escalado de frecuencias en el espectro, según se indica en el material previo: leeremos la señal del fichero, calcularemos su espectrograma, desplazaremos sus frecuencias y reconstruiremos la señal desde el espectro modificado.

- Descarga la plantilla para el proyecto: [PLANTILLA](#) (botón derecho → guardar como)
- La plantilla incluye los valores de los parámetros de cabecera, un operador (llamado **F0**) para medir la frecuencia fundamental al cabo de 1 segundo, la estructura básica del instrumento y una orden de partitura.

En el instrumento:

- Implementa el instrumento **transpon**. Como ves en el instrumento, ya hay definidas varias cosas:
 - El nombre del fichero que se recibirá desde la partitura en **p4**.
 - La relación de transposición que se recibirá como una fracción en **p5**.
 - Los valores del tamaño de la ventana, de su desplazamiento, solapamiento y tipo de ventana (ya definidos).
 - La lectura del sonido desde un fichero con **soundin**.
 - Y, al final del instrumento, una llamada al operador **F0** al que debes pasar el nombre de la variable de tipo f- que contenga el espectro transpuesto.
- Procesa el sonido según el esquema siguiente (los *opcodes* están descritos en el material previo) y emite el sonido procesado con **out**.



Las relaciones de frecuencias que usarás para transponer le sonido serán relaciones justas y calculando todos los intervalos respecto a la inicial, según la siguiente tabla:

Intervalo	I	II	iii	III	IV	V	vi	VI	VII	VIII
Relación	1/1	9/8	6/5	5/4	4/3	3/2	8/5	5/3	15/8	2/1

No olvides poner las fracciones de las relaciones de transposición entre corchetes en la partitura.

Pregunta 6

Finalizado

Puntúa 2,3 sobre 3,0

Descarga el siguiente sonido:

0:00 / 0:00

(4.6 segundos)

Partitura:

- Aplica el instrumento sobre ese sonido cuatro veces (4 activaciones consecutivas de 4.6 s). Se da ya la primera sin trasponer (relación de unísono $R_I = 1:1$).
- Haz las otras 3 activaciones transponiendo una VII y una VIII ascendentes y finalmente una vi descendente (todas calculadas desde la inicial).
- El instrumento debe imprimir en la consola el valor de la frecuencia fundamental que suena cada vez al cabo de un segundo. Anota a continuación los valores analizados:
 - Sin transponer: Hz
 - Primera transposición (+VII): Hz
 - Segunda transposición (+VIII): Hz
 - Tercera transpsición (−vi): Hz

