

Apellidos:	Nombre:
Apellidos:	Nombre:
Apellidos:	Nombre:
Apellidos:	Nombre:

## PRÁCTICA 1

### ESTUDIO DEL MUESTREO DE UNA SEÑAL DE AUDIO

- Al finalizar la práctica debes entregar las respuestas de este cuestionario en el formulario web que se proporcionará al principio de la sesión.
- Justifica adecuadamente tus cálculos y razonamientos.
- Usa al menos cuatro cifras significativas en tus operaciones.

#### Ejercicio 1.1 Parámetros principales del muestreo

En esta práctica vamos a analizar una señal de audio digital, proveniente del muestreo de una señal analógica. En un caso real, dicho proceso se realiza mediante un dispositivo denominado *convertidor analógico/digital* (CA/D), el cual presenta dos parámetros básicos:

- El período de muestreo  $T_s$ , es decir, la separación temporal entre dos muestras consecutivas. El inverso del período de muestreo es la frecuencia de muestreo  $F_s$ . O sea,  $F_s = 1/T_s$ .
- El número de bits  $n$  con el que se almacena digitalmente la amplitud de cada muestra. El número de niveles de amplitud distintos que se pueden representar con  $n$  bits es  $2^n$ .

Si estamos tomando  $F_s$  muestras cada segundo y cada una contribuye con  $n$  bits, la velocidad binaria total (número de bits por segundo) será  $R_b = nF_s$ .

Aunque existen muchos tipos de formatos de audio, uno de los más elementales desde el punto de vista de la codificación es el *.wav*, donde las muestras de la señal se graban más o menos como se explicará en la teoría de la asignatura. Concretamente, vamos a fijarnos en el fichero *Entrada.wav*, el cual contiene una grabación en formato estéreo (es decir, se guardan de manera independiente dos canales de audio, izquierdo y derecho). Vamos a tratar de determinar los parámetros con que se ha muestreado.

Para ello, inicia Matlab y abre el programa `p1_1`. Este programa importa el fichero *Entrada.wav*, creando en tu espacio de trabajo una variable `Fs` correspondiente a la frecuencia de muestreo en Hz y una matriz `x` correspondiente a los datos de sonido. Esta matriz tiene dos columnas, correspondientes a los datos de sonido de cada uno de los dos canales estéreo. Por tanto, el número de datos es el doble del normal (se han realizado dos muestreos).

**Q1.1** ¿Cuánto vale la frecuencia de muestreo?

- Q1.2** A continuación, lee en Windows las propiedades de *Entrada.wav*. Concretamente, en la pestaña *Detalles* aparece la velocidad binaria, en kilobits por segundo. ¿Cuánto vale dicha velocidad?
- Q1.3** Determina el número de bits que se han usado en el muestreo de cada canal, es decir, el número de bits  $n$  de cada CA/D. (Recuerda que tienes dos canales en cada grabación por ser estéreo y que la velocidad hallada en la cuestión Q1.2 es la total considerando ambos canales, por provenir de las propiedades del fichero completo).
- Q1.4** Con el dato anterior, ¿cuántos niveles de amplitud diferentes tiene el archivo *Entrada.wav*?
- Q1.5** De nuevo en las propiedades de *Entrada.wav*, en la pestaña *General*, mira el tamaño exacto en bytes (no lo que se denomina “tamaño en disco”). ¿Cuál es dicho tamaño?
- Q1.6** Con los datos anteriores puedes determinar la duración de tu grabación. (Duración exacta, no redondeada a segundos como muestra Windows o cualquier reproductor). Calcúlala detalladamente. (Recuerda que 1 byte son 8 bits).

## Ejercicio 1.2 Análisis de la frecuencia de muestreo

Entre la documentación proporcionada en la práctica, además de *Entrada.wav* encontrarás otros muchos archivos de audio con extensión *.wav*. Su denominación genérica es de la forma “*Tipo de voz\_frecuencia de muestreo.wav*”, donde *Tipo de voz* puede ser grave o aguda, y la frecuencia de muestreo varía de 1000 a 10000 Hz. Estos archivos se han generado a partir de dos señales analógicas distintas, una con voz grave y otra aguda, en las cuales se dice: “aaaa, eeee, iiiii, ooooo, uuuuu”. Posteriormente, cada una de las dos señales se ha muestreado a diferentes frecuencias.

- Q1.7** Escucha alternativamente las grabaciones agudas y graves a la misma frecuencia (usa el Reproductor de Windows Media para ello, no VLC). Es decir, escucha el archivo *Aguda\_1000.wav* seguido de *Grave\_1000.wav*, a continuación *Aguda\_2000.wav* seguido de *Grave\_2000.wav*, y así sucesivamente para varios valores de la frecuencia de muestreo. En general, para una misma frecuencia, ¿se escucha mejor la voz aguda o la grave?
- Q1.8** Escucha la grabación *Aguda\_1000.wav*. Comprobarás que es imposible distinguir adecuadamente las vocales. A continuación, sigue escuchando las grabaciones de la voz aguda de frecuencias de muestreo cada vez mayores. ¿A partir de qué frecuencia eres capaz de distinguir bien todas las vocales? (La respuesta es subjetiva pero debe ser coherente con el resto de respuestas del ejercicio 2.2).

- Q1.9** Si continuas aumentando la frecuencia de muestreo, conseguirás escuchar la grabación con mejor calidad, hasta llegar a cierto límite. ¿A partir de qué frecuencia ya no mejora la calidad de la grabación? (La respuesta es subjetiva pero debe ser coherente con el resto de respuestas del ejercicio 2.2).
- Q1.10** Repite la cuestión Q1.8 para el caso de la voz grave. (La respuesta es subjetiva pero debe ser coherente con el resto de respuestas del ejercicio 2.2).
- Q1.11** Repite la cuestión Q1.9 para el caso de la voz grave. (La respuesta es subjetiva pero debe ser coherente con el resto de respuestas del ejercicio 2.2).

Vamos a tratar de investigar por qué sucede lo anterior. Para ello usaremos el programa `p1_2`. Dicho programa representa las DTFTs de una voz grave y de una voz aguda<sup>1</sup> que han sido muestreadas con frecuencias de muestreo entre 1000 y 10000 Hz. Concretamente, se dibujan 9 figuras donde aparecen los espectros de las voces originales (con  $F_s = 10000$  Hz) y muestreadas a frecuencias decrecientes. Las DTFTs han sido calculadas ya *offline*.

Ejecuta el programa `p1_2`. Fíjate en primer lugar en las figuras de la voz grave, en orden decreciente de la frecuencia de muestreo. Comprobarás que, para una  $F_s$  de  $1000 \cdot k$  Hz, siendo  $k$  entero, el efecto producido es “estirar” en el eje horizontal el espectro de 10000 Hz en un factor  $10000/(1000 \cdot k) = 10/k$ . Por ejemplo, el espectro a 5000 Hz es el estiramiento al doble del original<sup>2</sup>. Puedes observarlo con mayor claridad con un zoom. Este estiramiento se produce siempre respetando la periodicidad de las transformadas, es decir, cada réplica periódica del espectro se estira por igual.

- Q1.12** Cuando la frecuencia de muestreo es muy baja, la DTFT sufre distorsión. ¿Por qué?
- Q1.13** Observa ahora las figuras de la voz aguda, que se comporta de manera similar. Sin embargo, ¿la distorsión empieza a hacerse patente para una frecuencia de muestreo mayor o menor que en la voz grave? ¿Por qué?
- Q1.14** De acuerdo con estos últimos razonamientos, explica tus resultados de las cuestiones Q1.7 a Q1.11. Haz referencia explícita tanto a la capacidad de entender las vocales como a la mejora de su calidad. (Aunque se trate de señales distintas a las de las cuestiones Q1.12 y Q1.13, el comportamiento es análogo).

---

<sup>1</sup> Con el fin de lograr unas visualizaciones más claras, estas voces no son las mismas que antes. Puedes escucharlas en *Voz aguda 2.wav* y *Voz grave 2.wav*.

<sup>2</sup> Teóricamente el estiramiento se efectuaría de manera ideal pero en la práctica puede haber pequeñas diferencias que son debidas a que, aunque la DTFT es continua, debe calcularse de manera discreta.

### Ejercicio 1.3 Análisis del número de bits

El programa `p1_3` tiene como objetivo modificar el número de bits con que se codifica la señal de audio contenida en el fichero *Entrada.wav* (siendo el número original el número de bits del convertidor analógico/digital, es decir, el determinado en la cuestión Q1.3). Como resultado, se genera el archivo de audio *Salida.wav*.

**Q1.15** Abre el programa `p1_3`. ¿Cuántos bits se utilizan para cada canal de audio en el archivo *Salida.wav*? (Míralo en el propio programa). ¿Cuántos niveles de amplitud tendrá dicho fichero de audio?

**Q1.16** Ejecuta el programa `p1_3`. (Ten cuidado de que no aparezcan errores de escritura<sup>3</sup>). Observa la figura. ¿Qué ha sucedido?

**Q1.17** Escucha el archivo *Salida.wav*. ¿Qué aprecias?

Edita el programa `p1_3`. Aumenta progresivamente el número de bits y compara los archivos resultantes con la grabación original, *Entrada.wav*. Llegará un momento en que no puedas diferenciar<sup>4</sup> el archivo original *Entrada.wav* y el modificado con el programa `p1_3`, *Salida.wav*. Detente en ese momento.

**Q1.18** ¿Con cuántos bits has generado en este último caso *Salida.wav*? ¿Cuántos niveles de amplitud son?

**Q1.19** Como conclusión de la práctica, habrás comprobado que es posible disminuir la tasa de muestreo y/o el número de bits de cuantificación sin que se aprecien diferencias con la señal original. ¿Esto significa que realmente se ha perdido información o no? ¿Por qué no se aprecian las diferencias?

**Q1.20** ¿Qué ventaja tendría desde un punto de vista práctico la situación observada en la cuestión anterior?

---

<sup>3</sup> Si aparece un error, es posible que se deba a que está abierto el reproductor de audio cuando ejecutas el programa.

<sup>4</sup> En ocasiones se aprecia un ligero nivel de ruido de fondo que no desaparece en ningún caso. No te fijas en eso, sino solo en cómo se escuchan las vocales.