

# Práctica 4 de Señales y Sistemas

## Señales y sistemas discretos en el dominio de la frecuencia

Francisco Javier Mercader Martínez

Rubén Gil Martínez

### 1. Transformada de Fourier de secuencias

#### Cuestiones

Realice un script en Matlab para representar una secuencia  $\mathbf{x}$  y su transformada de Fourier  $\mathbf{X}$ . Las secuencias están muestreadas a una frecuencia  $\mathbf{fs}$ . Para ello siga los siguientes pasos:

- Inicialice `ft = inline('fftshift(fft(x))');`
- Las secuencias de audio y su frecuencia de muestreo son accesibles desde Matlab- Para ello tiene que cargar los ficheros indicados más adelante mediante la instrucción `load`.
- Realice la representación gráfica de la secuencia empleando el eje de tiempo discreto. Etiquete adecuadamente cada uno de los ejes indicando claramente su contenido y en caso de que sea necesario las unidades.
- Empleando las instrucciones `subplot` represente la transformada de Fourier de la señal en dominio continuo y la transformada de Fourier de la secuencia. De nuevo incluya el etiquetado de los ejes.
- Reproduzca empleando la instrucción `sound` a la frecuencia de muestreo indicada la secuencia de audio.

Se muestran las secuencias a continuación que se han de estudiar empleando el código programado. Se estudiará la representación en tiempo y frecuencia. De manera cualitativa establezca relaciones entre las frecuencias que escucha y las que se pueden observar en la transformada de Fourier representada:

- Secuencia 'splat' muestreada a 8 KHz:

```
load splat.mat
```

- Secuencia 'laughter' muestreada a 8 KHz:

```
load laughter.mat
```

- Secuencia 'handel' muestreada a 8 KHz:

```
load handel.mat
```

## 2. Comparativa del coste computacional DFT y FFT

### Cuestiones

Implemente dos funciones en Matlab:

- La primera debe emplear la implementación directa con Matlab y su entrada será la secuencia a transformar. La función proporcionará la transformada. Llame a la función **mdft.m**
- La segunda debe implementar la versión eficiente basada en el algoritmo de diezmado en tiempo. El nombre de la función será **ffdt.m**

Redacte un script en Matlab que obtenga una secuencia real  $x$  de longitud  $N$  con muestras aleatorias empleando el comando **rand**.

Empleando los comandos **tic** y **toc** establezca el tiempo empleado en calcular su DFT utilizando la función de Matlab **mdft.m** y mediante la versión eficaz del mismo denominada **ffdt.m**

Empleando los siguientes valores para  $N = 2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}$  y  $2^{17}$  represente en una misma gráfica los tiempos de cálculo empleados por las dos implementaciones de la DFT respecto al valor de  $N$ . Etiquete correctamente ambos ejes e incluya la correspondiente leyenda para diferenciar ambas curvas.

## 3. Comparativa de convolución lineal vs convolución circular

### Cuestiones

Implemente el código necesario para generar y representar la convolución lineal de:

- Una secuencia de números aleatorios de 64 muestras definida en  $n = 0..63$ .
- Un filtro FIR casual que realice el promedio deslizante de 8 muestras.

Utilice la función **convcirc(x1,x2,N)** para realizar la convolución circular de  $N$  puntos de secuencia y la respuesta al impulso del filtro. Considere el mínimo  $N$  posible y represente gráficamente el resultado, etiquetando correctamente el eje temporal,  $n = 0..N - 1$ .

Defina la convolución circular de la siguiente manera

```
convcirc = inline('real(ifft(fft(x1, N) .* fft(x2, N), N))', 'x1', 'x2', 'N');
```

- Especifique si proporcionan el mismo resultado la convolución lineal y la convolución circular. Indique las muestras que difieren.
- Incremente el tamaño  $N$  de la convolución circular. Indique el valor de  $N$  para el que se obtiene un mismo resultado para ambas convoluciones.