PRÁCTICA 2 Análisis de sistemas de tiempo discreto en el dominio del tiempo.

2.1 Objetivos

1. Saber construir un sistema descrito por su relación entrada-salida.
2. Analizar un sistema descrito por una ecuación en diferencias.
3. Aprender a calcular la respuesta al impulso de un sistema.
4. Familiarizarse con la respuesta en frecuencia de los sistemas y su representación.
5. Crear y utilizar funciones.

2.2 Introducción

2.2.1 Sistemas LTI discretos

Se define un sistema discreto como un proceso que transforma una señal discreta de entrada en otra de salida, igualmente discreta. Los sistemas considerados en esta práctica serán siempre LTI (lineales e invariantes en el tiempo).

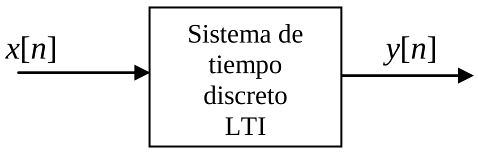


Figura 2-1. Sistema lineal e invariante en el tiempo

Un sistema LTI queda completamente caracterizado por su respuesta al impulso o, dicho de otra forma, por la señal de salida obtenida cuando la entrada es la delta de Kronecker, δ[n].

$$\delta[n] \rightarrow h[n] \quad (2\text{-}1)$$

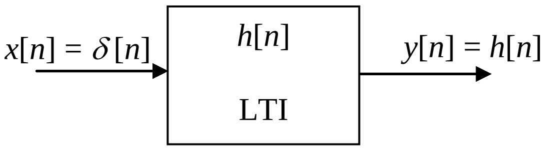


Figura 2-2. Sistema LTI caracterizado por su respuesta al impulso

La respuesta al impulso se suele denominar $h[n]$. Como el sistema es lineal e invariante en el tiempo, la respuesta al impulso caracteriza completamente el sistema. Por lo tanto, se puede calcular la salida del sistema como la convolución de la entrada con la respuesta al impulso

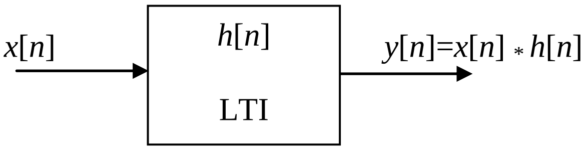


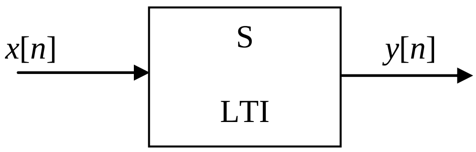
Figura 2-3. Respuesta de un Sistema LTI. La suma de convolución.

(2-2) $$x[n]=\sum_{k=-\infty}^{\infty} x[k]\delta[n-k] \Rightarrow y[n]=\sum_{k=-\infty}^{\infty} x[k]h[n-k]=x[n]*h[n]$$

Analizando la respuesta al impulso resulta inmediato determinar algunas propiedades del sistema, tal y como indica la siguiente tabla.

Propiedades de un sistema LTI	
No tiene memoria si y sólo si	$h[n]=k\cdot\delta[n]$
Es estable si y sólo si	$\sum_{n=-\infty}^{\infty} \lvert h[n] \rvert < \infty$
Es causal si y sólo si	$h[n]=0 \quad \forall n<0$

Una clase importante de sistemas discretos LTI es la de aquéllos cuyo comportamiento puede ser descrito mediante una ecuación en diferencias con condiciones iniciales nulas:



$$\begin{cases} a_0 y[n]+a_1 y[n-1]+a_2 y[n-2]+L = b_0 x[n]+b_1 x[n-1]+b_2 x[n-2]+L \\ C.I.Nulas \quad y[0]=0,\ y[-1]=0,\ y[-2]=0,... \end{cases}$$

Figura 2-4. Sistema LTI caracterizado por una ecuación en diferencias.

A partir de la ecuación en diferencias, y expresando la salida $y[n]$ en función de la entrada y de posibles salidas en instantes anteriores:

(2-3) $$y[n]=\frac{1}{a_0}\left[b_0 x[n]+b_1 x[n-1]+b_2 x[n-2]+\cdots-a_1 y[n-1]-a_2 y[n-2]-\cdots\right]$$

Resulta inmediato calcular la evolución de la salida para los distintos instantes n=0, n=1,…

$$y[0]=\frac{1}{a_0}\left[b_0x[0]+b_1x[-1]+b_2x[-2]+\cdots-a_1y[-1]-a_2y[-2]-\cdots\right] \tag{2-4}$$

$$y[1]=\frac{1}{a_0}\left[b_0x[1]+b_1x[0]+b_2x[-1]+\cdots-a_1y[0]-a_2y[-1]-\cdots\right] \tag{2-5}$$

...

2.2.2 Desarrollo de funciones en Matlab

El usuario de Matlab puede definir sus propias funciones y asignarles el nombre que quiera. Para definir una función propia simplemente hay que crear un fichero de texto con la extensión `.m` que ha de tener por nombre el mismo nombre que se quiera dar a la función y cuya primera línea tenga la forma:

```
function [variables_salida] = nombre_funcion (variables_entrada)
```

Por ejemplo, sea un archivo de texto llamado `suma_valor_absoluto.m` con el siguiente contenido:

```
function resultado = suma_valor_absoluto (operando1, operando2)
resultado = abs(operando1) + abs(operando2);
```

Si el archivo está guardado en el espacio de trabajo, entonces la función se puede usar desde el intérprete de comandos de Matlab:

```
>> a = 1+i
a =
   1.0000 + 1.0000i
>> b = -5
b =
    -5
>> c = suma_valor_absoluto(a,b)
c =
    6.4142
```

De igual modo, una función desarrollada por el usuario puede ser invocada desde un fichero `.m` de Matlab.

Puede crear una función propia para resolver un mismo tipo de ejercicio (la convolución, por ejemplo) repetidas veces.

2.2.3 Bibliografía

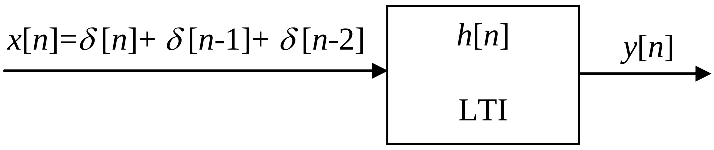
[1] "Señales y Sistemas". Oppenheim, Willsky, Young. Editorial Prentice-Hall 1998.

[2] "Señales y Sistemas". M. J. Roberts. Editorial Mc Graw Hill, 2004.

2.3 Desarrollo de la práctica

2.3.1 Estudio de un sistema LTI caracterizado por su respuesta al impulso

En este apartado se emplea la convolución para obtener la salida $y[n]$ de un sistema LTI a partir de la entrada $x[n]$ y su respuesta al impulso $h[n]$.



$$h[n]=\delta[n]+\delta[n-1]+\delta[n-2]+\delta[n-3]$$

Figura 2-5.Ejemplo de sistema LTI caracterizado por su respuesta al impulso

Para determinar la salida $y[n]=x[n] * h[n]$, en el ejemplo de la figura 2-5, se puede utilizar la instrucción `conv()` que calcula la convolución de dos vectores.

```
» x=[1 1 1];
» h=[1 1 1 1];
» y=conv(x,h);
```

Una de las *carencias* de la función `conv()` de Matlab reside en que no tiene en cuenta la posición de las secuencias en el eje "*n*" (recuérdese que Matlab no trabaja con señales, sino con vectores). Se limita a calcular un vector, con los valores numéricos de sus componentes, dejando para el usuario la tarea de *situarlo* en el eje de tiempos.

Ejercicios:

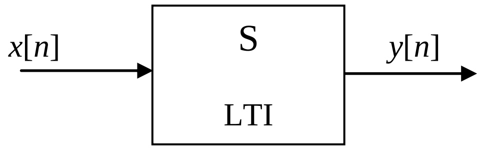
1. Calcular y representar la salida del sistema anterior, cuando la entrada es:

$$x[n]=\delta[n+3]+\delta[n+2]+\delta[n+1]$$

2. Cree dos vectores ("h" y "x", por ejemplo) que contengan las muestras no nulas de las dos señales.
3. Defina dos variables (hi y xi, por ejemplo) que contengan el instante de tiempo que corresponde al primer valor almacenado en los vectores "h" y "x", respectivamente.
4. Defina un eje de tiempo que comience en "hi" y tenga tantos elementos como el vector "h".
5. Repita el paso 4 para la señal $x[n]$
6. Calcule los valores del resultado de la convolución $y[n]=x[n]*h[n]$, con la función "conv" y almacénelo en un vector ("y", por ejemplo). Calcule cuál es el instante de tiempo ("yi", por ejemplo) que corresponde al primer valor del vector "y".
7. Repita el paso 4 para la señal $y[n]$.
8. Represente las tres secuencias ($x[n]$, $h[n]$, $y[n]$) implicadas.

2.3.2 Análisis de un sistema LTI descrito por una ecuación en diferencias

Una forma muy habitual de describir el comportamiento de un sistema LTI consiste en relacionar la entada y la salida mediante una ecuación en diferencias con condiciones iniciales nulas.



$$y[n] - 0.5 \cdot y[n-1] + 0.5 \cdot y[n-2] = x[n] + 0.2 \cdot x[n-1] + 0.5 \cdot x[n-2]$$

C. I. Nulas

Figura 2-6. Ejemplo de sistema LTI caracterizado por una ecuación en diferencias

Para ilustrar este punto se toma como ejemplo el sistema LTI de la figura 2-6 descrito por la ecuación en diferencias indicada y que parte con condiciones iniciales nulas.

El siguiente código de Matlab permite obtener las 30 primeras muestras de la respuesta al impulso del sistema empleando la función `filter()` (se puede emplear `help filter` para obtener más detalles):

```
a=[1 -0.5 0.5];           %definición del vector de coeficientes ak
b=[1 0.2 0.5];      %definición del vector de coeficientes bk
n=0:29;             %definición del intervalo de tiempo
imp=[1 zeros(1,29)];%Impulso unidad en el intervalo de trabajo
h=filter(b,a,imp);  %la función filter calcula la salida del sistema
                    %definido por una ecuación en diferencias para
                    %una entrada dada, imp en este caso.

%Representación de señales
subplot(211)        % Divide la figura en varias gráficas (2 filas,
                    % 1 columna) y selecciona la primera de izquierda
                    % a derecha y de arriba abajo.
stem(n,imp);
title('Impulso unidad');
subplot(212)
stem(n,h)
title('Respuesta al impulso');
```

Obsérvese que la salida de `filter()` tiene el mismo tamaño que la entrada que se le pasa (en este caso `imp`).

Ejercicios:

9. Obtener la salida del sistema de la figura 2-6 en el intervalo n=0:49 cuando la entrada es

$$x[n] = 2\delta[n] - 2\delta[n-1] + 2\delta[n-2] - 2\delta[n-3]$$

Escribir un fichero de comandos en Matlab que realice esta operación, usando tanto la función `conv()` como `filter()`. Suponer que las 30 primeras muestras de la respuesta al impulso, obtenidas anteriormente, son suficientes para caracterizar $h[n]$. Compare los resultados obtenidos con cada una de las dos funciones.

2.3.3 Funciones utilizadas

Además de las funciones utilizadas en la práctica 1, para la práctica 2 se han de utilizar:

`conv()` `filter()` `subplot()`

así como las funciones definidas por el propio alumno.

2.4 Ejercicios propuestos

2.4.1 Enunciados

1. Empleando la función desarrollada en el apartado 2.3.1 y dadas las siguientes secuencias:

$$x_1[n] = A_1(\mathrm{u}[n + N_1] - \mathrm{u}[n - N_1])$$

$$x_2[n] = A_2\mathrm{u}[n + N_2] \cdot \mathrm{u}[-n + N_2]$$

$$x_3[n] = A_3(\mathrm{u}[n + N_1] - \mathrm{u}[n - N_2]) \cdot \mathrm{e}^{\mathrm{j}\Omega_1 n}$$

Obtener y representar las siguientes convoluciones (represente también las señales que se convolucionan):

1.1. $y_1[n] = x_1[n] * x_1[n + N_3]$

1.2. $y_2[n] = x_1[n] * x_2[n]$

1.3. $y_3[n] = x_1[n] * x_1[n] * x_2[n]$

1.4. $y_4[n] = \mathrm{Re}\,\{x_3[n]\} * \mathrm{Im}\,\{x_3[n - N_4]\}$

2. Obtener y representar, para n=0,1…N_5, la respuesta al impulso de los sistemas LTI descritos por las siguientes ecuaciones en diferencias. Observando la respuesta al impulso, determinar si los sistemas son estables.

2.1. $y[n] = A_1x[n] + A_3x[n - N_3]$, C.I. nulas.

2.2. $y[n] + y[n - 1] = x[n] + A_1\, x[n - 1] + A_2\, x[n - 2] + A_3x[n - 3]$, C.I. nulas.

3. Obtener y representar, para n=0,1…N_5, la salida de los anteriores sistemas cuando la entrada es:

3.1. $x_4[n] = A_1(\mathrm{u}[n] - \mathrm{u}[n - N_1])$

3.2. $x_5[n] = A_3 \cdot \mathrm{e}^{\mathrm{j}\Omega_1 n}\mathrm{u}[n]$

Compare los resultados usando la convolución y usando la función "filter"

4. Para la conexión de sistemas representada en la figura 2-7, y sabiendo que:

   - *S1* es un sistema LTI tal que $y[n] + A_4 y[n-2] = A_1 x[n - N_4]$ y que parte con C.I. nulas.
   - *S2* es un sistema LTI cuya respuesta al impulso es

   $$h[n] = \delta[n] + A_1\delta[n - N_4] + A_2\delta[n - N_2] + A_3\delta[n - N_1]$$

   - $x[n] = A_1 \cos(\Omega_2 n)(\mathrm{u}[n] - \mathrm{u}[n - N_6])$

   Represente gráficamente las señales de $s[n]$, $v[n]$ e $y[n]$ en el intervalo n=0:15.

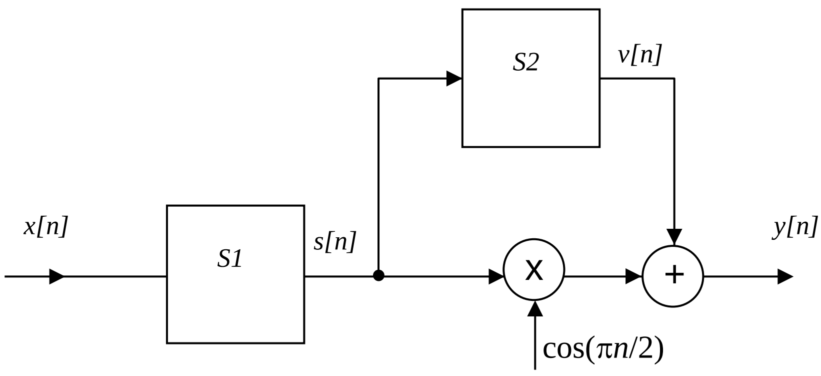


Figura 2-7. Interconexión de sistemas LTI.

2.4.2 Presentación de resultados.

La presentación de resultados será en formato electrónico y se deberá entregar en Moodle **un unico fichero en formato ".pdf"**. Deberá incluir como mínimo, los siguientes aspectos:

1. Copia de las figuras de Matlab con las representaciones de las señales para todos los ejercicios.
2. Copia de los "scripts" o ficheros ".m" (insertados como texto plano en el document a entregar) que realizan las operaciones pedidas.
3. Epígrafe con el ejercicio y apartado a que se refiere cada figura y script copiado.

NOTA: El alumno deberá guardar copia de los ficheros ".m" que haya generado porque le pueden ser requeridos en sesiones sucesivas (incluso en la sesión de evaluación del laboratorio).

2.4.3 Valores de las constantes

$A_1 = 2$; $A_2 = -2$; $A_3 = 3$; $A_4 = -1/2$

$N_1 = 6$; $N_2 = 4$; $N_3 = 3$; $N_4 = 2$

$N_5 = 20$; $N_6 = 12$;

$\Omega_1 = 1/3$; $\Omega_2 = 3\pi/8$