IEE239 - Procesamiento de Señales e Imágenes Digitales
Laboratorio 1 - Ejercicios Propuestos
Primer Semestre 2018

1. El señor Jürgen se compromete a un préstamo e incurre en una deuda con el Soft Rock Bank. La deuda restante en cada mes, denotada por $d[m]$, depende de la deuda del mes anterior y del sueldo del señor Jürgen en cada mes, denotado por $s[m]$. $d[m]$ es igual a 90 % de la deuda del mes anterior (por los pagos de capital que se realizan) menos el 20 % del sueldo del señor Jürgen (cantidad que usa para pagar intereses). Se pide:

   a. Plantear la ecuación en diferencias para la deuda en el mes m.

   b. Considerar que el señor Jürgen no percibió sueldo antes de incurrir en el préstamo y que empezó a percibir 1500 dólares cada mes desde aquel momento. Asumiendo que el sistema es causal y que la deuda inicial es de 25000 dólares, usar un bucle `for` para hallar en `Matlab` la deuda en $m = 4$. Usar `subplot` y `stem` para graficar $s[m]$ y $d[m]$.

   c. Suponer que, a sugerencia del señor Jürgen, el señor Wohlberg incurre en el mismo ṕrestamo y en las mismas condiciones iniciales salvo que su sueldo mensual ($s[m]$) es el doble, es decir 3000 dólares mensuales. Asumiendo que el sistema es causal y que la deuda inicial es de 25000 dólares, usar un bucle `for` para hallar en `Matlab` la deuda en $m = 4$. Usar `subplot` y `stem` para graficar $s[m]$ y $d[m]$ ¿Se trata de un sistema lineal? Justificar su respuesta.

   d. Hallar la transformada z del sistema. Halle los polos y ceros del sistema. Comentar sobre la estabilidad del sistema. Asuma que $s[m]$ siempre está acotado y que un $d[m]$ no acotado equivale a que el Soft Rock Bank denuncie al señor Jürgen con Infocorp ¿Será el señor Jürgen alguna vez denunciado por esta deuda?

   e. Ahora suponer las mismas condiciones, excepto que $d[m]$ es igual a 110 % de la deuda del mes anterior, debido a que no se realiza el suficiente pago de intereses. Hallar los polos de este nuevo sistema y comentar sobre la estabilidad.

2. Se brinda una función `sistemaEjemplo.p`, la cual devuelve la respuesta de un sistema LTI causal a una entrada `x` en forma vectorial (El primer elemento de `x` corresponde al tiempo $n = 0$). El comando es ejecutado escribiendo `sistemaEjemplo(x)`. Se pide:

   a. Hallar en Matlab la respuesta al impulso y graficarla. Usar la función `stem` para graficar la secuencia y las funciones `xlabel`, `ylabel` y `title` para rotular el gráfico apropiadamente.

   Solución: El código se muestra a continuación. La imagen resultante se muestra en la Figura 1.

```
        % generar el impulso
```

```
        impulso = [1; zeros(9,1)];
        % obtener la salida al sistema
        h = sistemaEjemplo(impulso);
        figure
        stem([0:9],h,'lineWidth',2)
        title('Respuesta al impulso del sistema')
        xlabel('Tiempo')
        ylabel('h[n]')
```

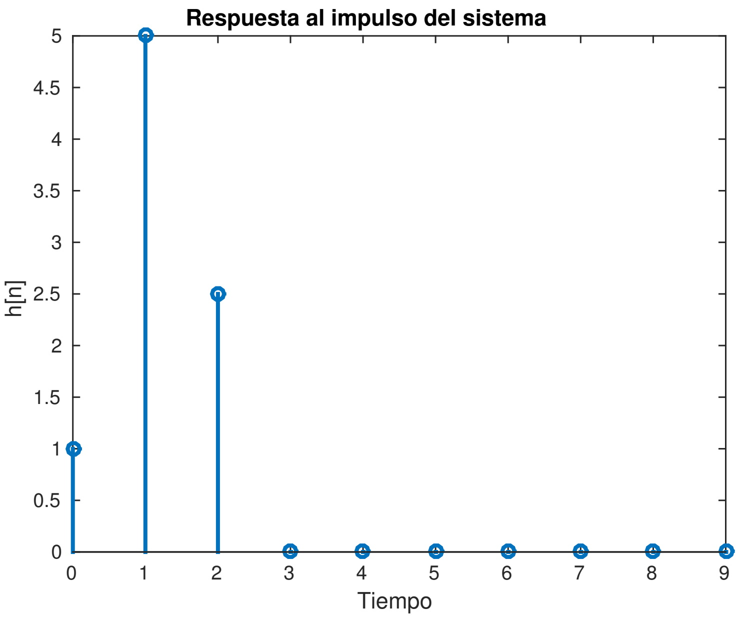


Figura 1: Respuesta al impulso del sistema implementado en `sistemaEjemplo`.

b. Se sabe que el sistema es de la forma $y[n] = a_1x[n] + a_2x[n-1] + a_3x[n-2]$ ¿Cuáles son los valores de a_1, a_2 y a_3? Hallar la respuesta al impulso $h[n]$ y graficarla y roturla apropiadamente. Comentar sobre la estabilidad del sistema.

   Solución: Como el sistema es FIR (es decir la salida solo depende de valores de la entrada), los coeficientes y por tanto la respuesta al impulso son iguales. Del gráfico obtenido se tiene $h[n] = [\underset{\uparrow}{1} \quad 5 \quad 2.5]$

c. Usar la respuesta de la parte anterior para hallar, mediante el comando `conv`, la respuesta a un escalón unitario y una rampa unitaria. Compararlos con las de la función que se le brinda. Graficar las respuestas usando `subplot` y `stem`[1].

   Solución: El código se muestra a continuación. La imagen resultante se muestra en la Figura 2.

```
        % general escalon y rampa
    u = [ones(10,1)];
    r = [1:10]';
```

[1]Para secuencias causales, la función brindada solo da la respuesta en el dominio $0 : N-1$ donde N es la longitud de la secuencia de entrada. Por el contrario, la convolución entrega la respuesta en el dominio $0 : N+M-1$ donde N y M son las longitudes de las dos secuencias que son convolucionadas.

```
% respuestas con la funcion dada
y1 = sistemaEjemplo(u);
y2 = sistemaEjemplo(r);
figure
subplot(2,2,1)
stem([0:9],y1,'lineWidth',2)
xlabel('Tiempo')
ylabel('Respuesta al escalon')
title('Rpta con funcion dada al escalon')
subplot(2,2,2)
stem([0:9],y2,'lineWidth',2)
xlabel('Tiempo')
ylabel('Respuesta a la rampa')
title('Rpta con funcion dada a la rampa')
% respuestas por convolucion
h = [1 5 2.5];
g1 = conv(h,u);
g2 = conv(h,r);
subplot(2,2,3)
stem([0:length(u)+length(h)-2],g1,'lineWidth',2)
xlabel('Tiempo')
ylabel('Respuesta al escalon')
title('Rpta con convolucion al escalon')
subplot(2,2,4)
stem([0:length(u)+length(h)-2],g2,'lineWidth',2)
xlabel('Tiempo')
ylabel('Respuesta a la rampa')
title('Rpta con convolucion a la rampa')
```

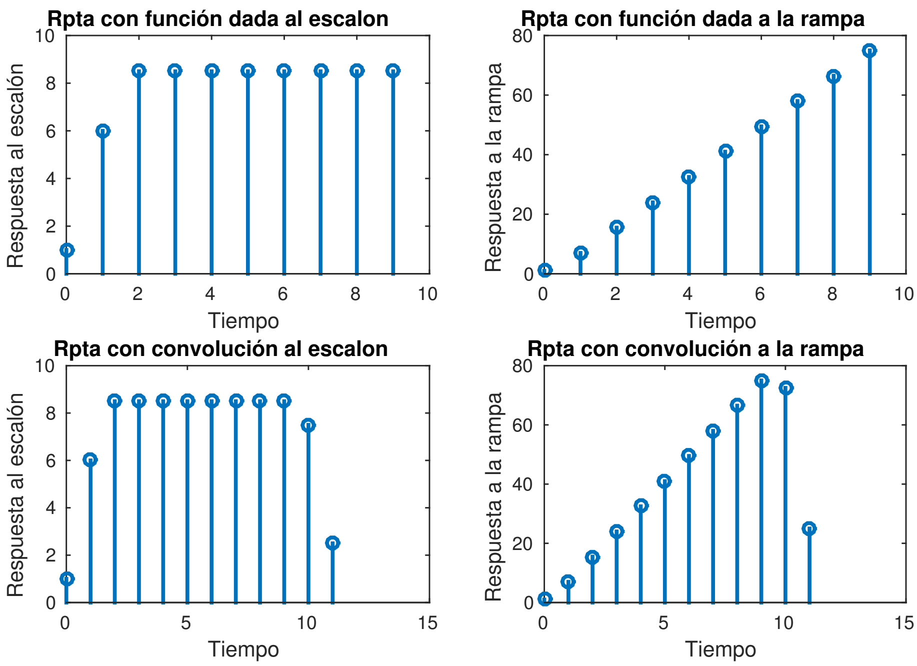


Figura 2: Respuesta a escalón y rampa unitaria usando `sistemaEjemplo` y `conv`

d. Con los coeficientes del filtro hallados anteriormente, encontrar la matriz convolucional H, tal que $y = Hx$ ¿Qué patrón sigue cada columna de la matriz y cómo se relaciona a las propiedades LTI del sistema? Encontrar, además, una matriz G inversa que permita obtener x a partir de y ¿El proceso inverso sería posible para un sistema lineal pero variante en el tiempo?

Solución:
Para este caso, considerando una entrada de 10 muestras se tendría la matriz de 12×10 dada por:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2.5 & 5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.5 & 5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.5 & 5 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.5 & 5 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2.5 & 5 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.5 & 5 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2.5 & 5 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.5 & 5 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.5 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.5 \end{bmatrix}$$

Se observa que, por tratase de un sistema LTI, cada columna es un versión desplazada de la primera columna (en otras palabras, en cada instante de tiempo se aplica la misma función pero desplazada). A esta estructura se le conoce como matriz *Toeplitz*. La matriz inversa G puede hallarse ejecutando el comando `pinv(H)`, el cual halla la matriz pseudo-inversa de H. En el código adjunto se muestra un ejemplo y una demostración de como se puede recuperar una entrada aleatoria. Este proceso también sirve para sistemas variantes en el tiempo, con la única diferencia que H no tendría un patrón Toeplitz.

```
H = toeplitz([1;5;2.5;zeros(9,1)],[1 zeros(1,9)]');
G = pinv(H);
x = rand(10,1)
y = H*x;
x_estimado = G*y

 % Ejemplo de sistema variante en el tiempo
H2 = randn(12,10);
G2 = pinv(H2);
x2 = rand(10,1)
y2 = H2*x2;
x2_estimado = G2*y2
```

3. Dado el sistema $T_{\mathrm{d}2}$ en su Forma Directa I descrito en la Figura 3 y la señal en tiempo contínuo $x_c(t)$:

$$x_c(t) = \begin{cases} 10(t-0.5) & , 0.5 \leq t < 1.5 \\ 10(2.5-t) & , 1.5 \leq t < 2.5 \\ 0 & , \text{otros casos} \end{cases}.$$

a. Crear la secuencia $x[n] \triangleq x_c(nT_s)$ para $n \in \{0, 1, \ldots, 47\}$ y $T_s = 0.1$ s. Luego, graficar la secuencia resultante y rotularla adecuadamente.

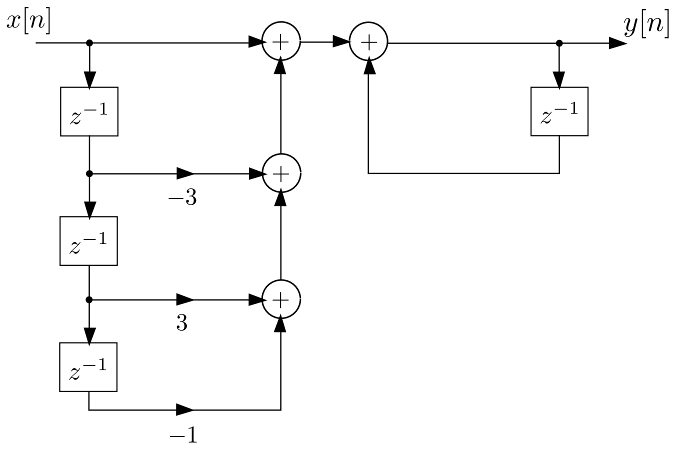


Figura 3: Sistema recursivo en tiempo discreto $T_{\text{d}2}$.

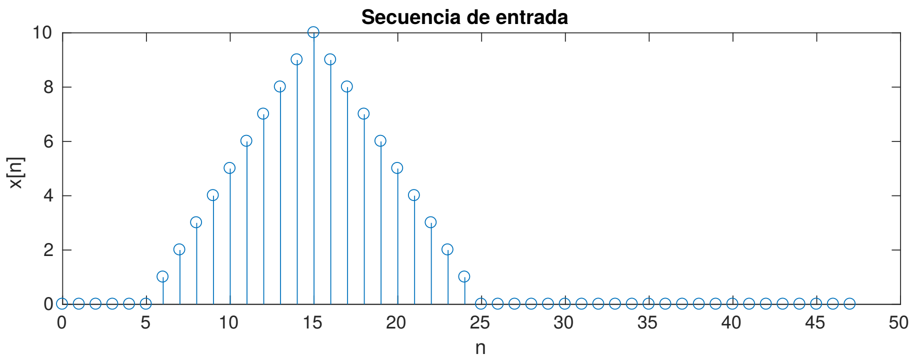


Figura 4: Secuencia de entrada $x[n]$.

Solución: discretizando para $T_s = 0.1$ s, la secuencia $x[n]$ corresponde a:

$$\begin{aligned} x[n] &\triangleq x_c(n \cdot 0.1) \\ &= \begin{cases} 10(n \cdot 0.1 - 0.5) & , 0.5 \leq n \cdot 0.1 < 1.5 \\ 10(2.5 - n \cdot 0.1) & , 1.5 \leq n \cdot 0.1 < 2.5 \\ 0 & , \text{otros casos} \end{cases} \\ &= \begin{cases} n - 5 & , 5 \leq n < 15 \\ 25 - n & , 15 \leq n < 25 \\ 0 & , \text{otros casos} \end{cases} . \end{aligned}$$

El Script 1 muestra las instrucciones para generar la secuencia triangular de entrada y la Figura 4 describe el resultado.

Script 1: Secuencia triangular

```
% --- a. secuencia de interes --- %
N= 48;   % longitud del dominio
n_v= 0: N-1;     % espacio de muestras
x01_v= ( n_v- 5).*( n_v≥ 5 & n_v< 16)+ ( 25- n_v).* ( n_v≥ 16 & n_v< ...
    25); % secuencia triangular

```

```
fig01= figure; stem( n_v, x01_v); xlabel( 'n'); ylabel( ...
    '\hat{x}[n]'); title( 'Secuencia Triangular');
```

b. Determinar la ecuacion de diferencias del sistema T_{d2} e incluirla en comentarios. Luego, crear una función `func_Td2()` que reciba la secuencia de entrada y las condiciones iniciales del sistema, y a partir de ello genere la salida del sistema recursivo usando un bucle `for`. Finalmente, asumiendo sistema en reposo, obtener $T\{x[n]\}$, graficar las secuencias de entrada y salida en una misma ventana y rotularlas adecuadamente. Cuál es el efecto del sistema? Justificar su respuesta e incluirla en comentarios.

Solución: a partir del diagrama de bloques, la ecuación de diferencias de T_{d2} corresponde a la siguiente expresión:

$$y[n] - y[n-1] = x[n] - 3x[n-1] + 3x[n-2] - x[n-3].$$

De la expresión, se requiere de la condición auxiliar $y[-1]$ para conocer la respuesta ante una entrada causal[2], por lo tanto, la implementación de `func_Td2()` debe recibir la secuencia de entrada y un valor escalar correspondiente a $y[-1]$. La función, implementada a partir de `for-loops`, corresponde al archivo **func_Td2.m**. El Script 2 muestra el uso de la subrutina y la Figura 5 muestra la respuesta del sistema ante la entrada $x[n]$ y la condición inicial $y[-1] = 0$ correspondiente a estado de reposo.

Para analizar el efecto del sistema ante la secuencia de entrada triangular, cabe recalcar lo siguiente:

- La derivada de primer orden de una secuencia triangular corresponde a pulsos positivos en intervalos con pendientes positivas y pulsos negativos en intervalos con pendientes negativas. Por lo tanto, para la señal de interés, su derivada de primer orden debería ser un pulso positivo seguido un pulso negativo, ambas con la misma magnitud.
- La derivada de un pulso corresponde a dos impulsos unitarios. Un impulso positivo indica el inicio del pulso y un impulso negativo indica su fin.

Script 2: Respuesta del sistema ante secuencia triangular

```
% --- b. determinar ecuacion de diferencias del sistema y respuesta ...
    ante triangular --- %
% Del diagrama de bloques, obtenemos la siguiente relacion implicita:
% y[ n]- y[ n- 1]= x[ n]- 3x[ n- 1]+ 3x[ n- 2]- x[ n- 3]

y01_v= func_Td2( x01_v, 0);   % sistema a partir de for-loops, ...
    inicialmente en reposo
b_v= [ 1 -3 3 -1];     % coeficientes de entrada
a_v= [ 1 -1]; % coeficientes de salida

fig02= figure;
subplot( 2, 1, 1); stem( n_v, x01_v); xlabel( 'n'); ylabel( 'x[n]'); ...
    title( 'Secuencia de entrada') % descripcion grafica
subplot( 2, 1, 2); stem( n_v, y01_v, 'r'); xlabel( 'n'); ylabel( ...
    'T\{x[n]\}'); title( 'Respuesta del sistema ante x[n]') % ...
    descripcion grafica
```

[2] De forma general, para una entrada cuyos valores diferentes de 0 aparecen desde la posición n_0, la condición inicial requerida es $y[n_0 - 1]$.

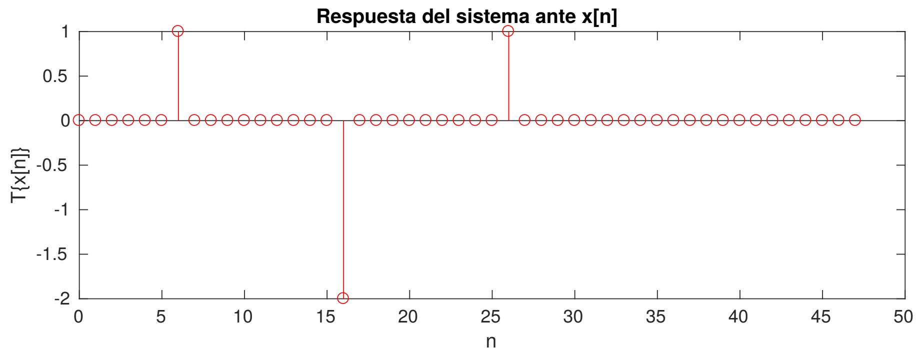


Figura 5: Respuesta del sistema ante $x[n]$.

Analizando la secuencia de salida, es claro que impulsos unitarios aparecen en los cambios de la onda triangular, cada uno con signos que indican el cambio de pendiente. Se puede notar además que el impulso que corresponde al cambio de pendiente positiva a negativa tiene el doble de magnitud. Por lo tanto, el sistema es un **derivador de segundo orden**.

c. De forma alternativa, obtener $y[n] = T\{x[n]\}$ a partir de `filter()`. Graficar el resultado y rotular adecuadamente. Verificar que ambos resultados son similares a partir de su distancia euclidiana[3]:

$$d_2\{\mathbf{a}, \mathbf{b}\} \triangleq \sqrt{\sum_{i=0}^{N-1} (a_i - b_i)^2},$$

donde $\mathbf{a}$ y $\mathbf{b}$ corresponden a los vectores a comparar, con N elementos cada uno.

Solución: la rutina `filter()` recibe $\mathbf{a}$ y $\mathbf{b}$, los vectores de coeficientes de ambos miembros de la ecuación de diferencias[4] y la secuencia de entrada. De la forma general de la ecuacion de diferencias:

$$\sum_{k=0}^{N-1} a_k y[n-k] = \sum_{k=0}^{M-1} b_k x[n-k],$$

el sistema de interés se caracteriza por:

$$\mathbf{a} = \{\underset{\uparrow}{1}, -1\}, \qquad \mathbf{b} = \{\underset{\uparrow}{1}, -3, 3, -1\}.$$

El Script 3 muestra el uso de `filter()` y la Figura 6 describe la secuencia resultante. La distancia euclidiana entre ambas secuencias de salida es muy cercana a 0, indicando que ambos vectores son muy similares. Esto implica que las dos subrutinas implementan el mismo sistema.

Script 3: Respuesta del sistema ante secuencia triangular a partir de `filter()`

```
% —— c. verificar secuencia de salida a partir de filter() —— %
y01_alt_v= filter( b_v, a_v, x01_v);
errval= norm( y01_v— y01_alt_v);     % distancia euclidiana entre ...
    salida por la rutina func_Td2 y por filter

```

[3] Revisar la rutina `norm()`.

[4] Por convención, el peso correspondiente a $y[n]$ debe ser 1 para la rutina `filter()`.

```
% errval es cercano a 0 absoluto, lo que indica que ambos vectores son
% iguales.

fig03= figure;
subplot( 2, 1, 1); stem( n_v, y01_v); xlabel( 'n'); ylabel( ...
    'T\{x[n]\}'); title( 'Respuesta del sistema a partir de ...
    for—loops') % descripcion grafica
subplot( 2, 1, 2); stem( n_v, y01_alt_v, 'r'); xlabel( 'n'); ylabel( ...
    'T\{x[n]\}'); title( 'Respuesta del sistema a partir de filter') ...
    % descripcion grafica
```

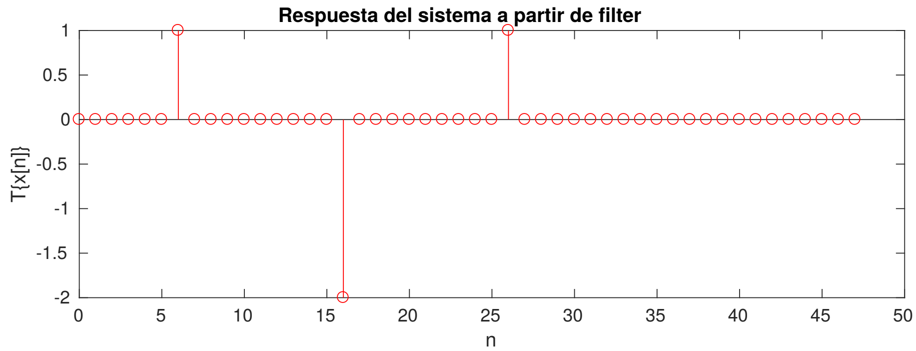


Figura 6: Respuesta del sistema ante $x[n]$ usando `filter()`.

d. Generar la secuencia $x_d[n]$ como un retardo de 10 muestras de $x[n]$[5] y determinar la respuesta del sistema ante dicha secuencia. Describir gráficamente ambas secuencias en una misma ventana y rotularlas adecuadamente. Podría tratarse de un sistema invariante en el tiempo? Justificar claramente su respuesta.

Solución: un sistema cumple con ser invariante ante desplazamientos en el tiempo si la siguiente relación es valida para toda secuencia de entrada $x[n]$. Dada la secuencia $y[n] = T\{x[n]\}$ y una posición $n_d \in \mathbb{Z}$:

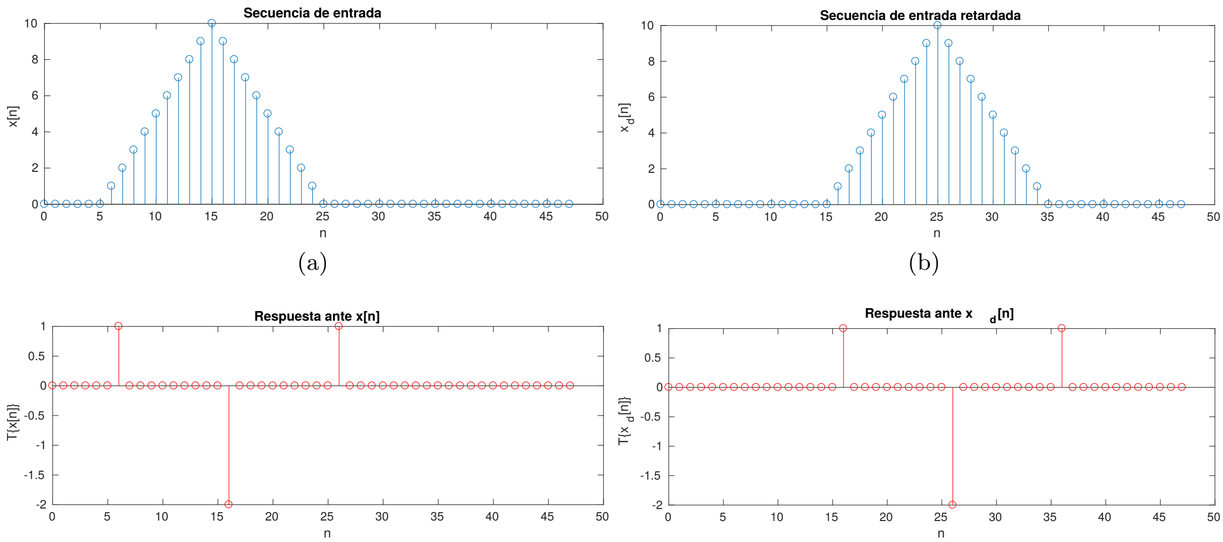
$$T\{x[n - n_d]\} = y[n - n_d].$$

El Script 4 muestra el uso de la subrutina `func_Td2()` para obtener la respuesta ante una secuencia con 10 muestras de retardo. Cabe recalcar que el retardo fue realizado a partir de `zeros()`. La Figura 7 muestra la comparación entre ambas secuencias. Se puede observar que, para el caso particular de la secuencia de entrada $x[n]$, el sistema cumple con la propiedad. Sin embargo, para afirmar que se trata de un sistema invariante en el tiempo, es necesario demostrar que esto es cierto para cualquier secuencia de entrada. La gráfica refuerza la posibilidad de que el sistema sea invariante en el tiempo, pero no es suficiente para afirmarlo.

Script 4: Analisis de respuesta ante desplazamientos en el tiempo

```
% —— d. respuesta ante secuencia con retardo —— %
D= 10;
x02_v= [ zeros( 1, D) x01_v( 1: end— D)];   % ingresar D ceros, ...
    descartar ultimas D muestras
```

[5] Para generar el retardo, incluir ceros al inicio de la secuencia y descartar la misma cantidad de muestras al final de ella para mantener la longitud del vector. Usar `zeros()`.



(a) (b) (c) (d)

Figura 7: Imagen reconstruida. (a) Secuencia de entrada original. (b) Secuencia de entrada con retardo de 10 muestras. (b) Respuesta ante secuencia de entrada original. (c) Respuesta ante secuencia de entrada con retardo de 10 muestras.

```
y02_v= func_Td2( x02_v, 0);   % sistema a partir de for-loops, ...
    inicialmente en reposo

fig04= figure;
subplot( 2, 2, 1); stem( n_v, x01_v); xlabel( 'n'); ylabel( 'x[n]'); ...
    title( 'Secuencia de entrada'); % descripcion grafica
subplot( 2, 2, 2); stem( n_v, x02_v); xlabel( 'n'); ylabel( ...
    'x_{d}[n]'); title( 'Secuencia de entrada retardada'); % ...
    descripcion grafica
subplot( 2, 2, 3); stem( n_v, y01_v, 'r'); xlabel( 'n'); ylabel( ...
    'T\{x[n]\}'); title( 'Respuesta ante x[n]'); % descripcion grafica
subplot( 2, 2, 4); stem( n_v, y02_v, 'r'); xlabel( 'n'); ylabel( ...
    'T\{x_{d}[n]\}'); title( 'Respuesta ante x_{d}[n]'); % ...
    descripcion grafica
```

e. Determinar de forma analítica la respuesta al impulso del sistema. Luego, verificar lo anterior a partir de la función `impz()`. Se trata de un sistema FIR o IIR? Se trata de un sistema BIBO estable? Justificar claramente su respuesta en comentarios.

   Solución: A partir de la ecuación de diferencias, la función de transferencia de T_{d2} corresponde a:

$$H(z) = \frac{1 - 3z^{-1} + 3z^{-2} - z^{-3}}{1 - z^{-1}}.$$

   A partir de división de polinomios, la expresión se simplifica a:

$$H(z) = 1 - 2z^{-1} + z^{-2}.$$

   Al tratarse de una respuesta al impulso conformada solo por ceros, se sabe que el sistema es de **respuesta al impulso finita (FIR)**. A partir de series de potencias, la respuesta al impulso puede ser facilmente calculada:

$$h[n] = \mathcal{Z}^{-1}\{1 - 2z^{-1} + z^{-2}\} = \delta[n] - 2\delta[n-1] + \delta[n-2] = \{\underset{\uparrow}{1}, -2, 1\}.$$

Luego, para analizar si se trata de un sistema BIBO estable, basta con determinar si $h[n]$ es absolutamente sumable:

$$\sum_{n=-\infty}^{+\infty} |h[n]| = |1| + |-2| + |1| = 4 < +\infty$$

Por lo tanto, se trata de un sistema **BIBO estable**. El Script 5 muestra el uso de la rutina `impz()` para determinar la respuesta al impulso del sistema a partir de los coeficientes de su ecuación de diferencias y la Figura 8 la muestra. La secuencia resultante es equivalente a aquella obtenida de forma analítica.

Script 5: Respuesta al impulso del sistema y salida a partir de convolucion

```
% —— e. respuesta al impulso del sistema —— %
h_v= impz( b_v, a_v, N);     % verificar expresion analitica con ...
    resultado de impz()
y_conv_v= conv( h_v, x01_v);     % salida por convolucion.
y_conv_v= y_conv_v( 1: length( x01_v)); % mantener longitud original

fig05= figure;
subplot( 3, 1, 1); stem( n_v, h_v); xlabel( 'n'); ylabel( 'h[n]'); ...
    title( 'Respuesta al impulso del sistema'); % descripcion grafica
subplot( 3, 1, 2); stem( n_v, y01_v, 'r'); xlabel( 'n'); ylabel( ...
    'y[n]'); title( 'Salida por implementacion recursiva'); % ...
    descripcion grafica
subplot( 3, 1, 3); stem( n_v, y_conv_v, 'g'); xlabel( 'n'); ylabel( ...
    'y[n]'); title( 'Salida por convolucion'); % descripcion grafica
```

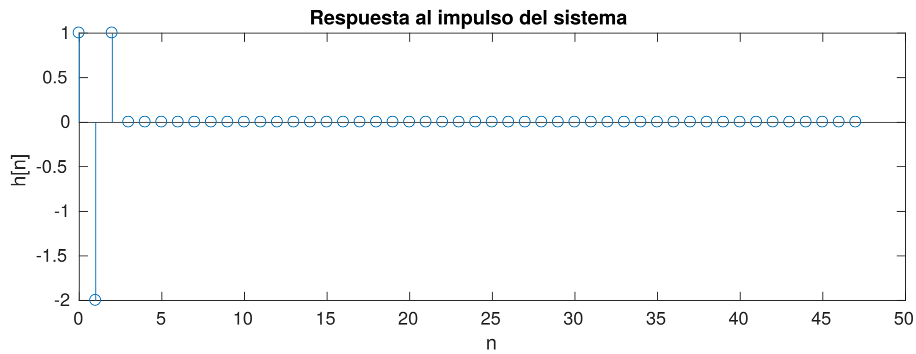


Figura 8: Respuesta al impulso del sistema T_{d2}.

f. A partir de transformada z, Determinar el sistema T_{inv} tal que:

$$T_{\mathrm{inv}}\{T\{x[n]\}\} = x[n].$$

Determinar su respuesta ante la secuencia obtenida en 3c y verificar que se obtiene $x[n]$. T_{inv} es un sistema FIR o IIR? Es BIBO estable? Incluir su respuesta en comentarios.

Solución: Aplicando propiedades de transformada z, podemos expresar el problema de la siguiente forma:

$$\hat{X}(z) = H_{\mathrm{inv}}(z) \cdot H(z) \cdot X(z),$$

donde $X(z)$ es la transformada z de la secuencia de entrada $x[n]$, $H(z)$ la función de transferencia del sistema T_{d2}, $H_{\mathrm{inv}}(z)$ la función de transferencia del sistema T_{inv}

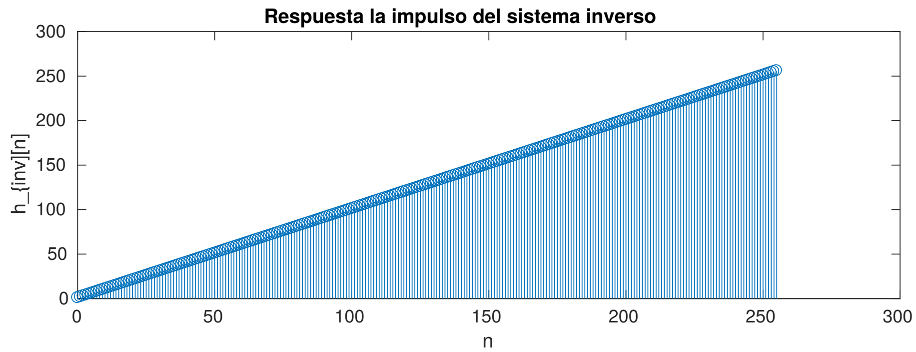


Figura 9: Respuesta al impulso del sistema inverso T_{inv}.

y $\hat{X}(z)$ es la transformada de la secuencia $\hat{x}[n]$. Se requiere que $\hat{x}[n] = x[n]$, lo cual implica que $\hat{X}(z) = X(z)$, entonces:

$$H_{\text{inv}}(z) = \frac{1}{H(z)} = \frac{1}{1 - 2z^{-1} + z^{-2}} = \frac{1}{(1 - z^{-1})^2}.$$

Teóricamente, se sabe que para obtener una secuencia de salida causal como respuesta ante una secuencia de entrada causal, el sistema debe ser causal. Adicionalmente, dado que la expresión se compone unicamente de polos, se sabe que el sistema es de **respuesta al impulso infinita (IIR)**. Esto puede verificarse aplicando pares de transformación conocidos para determinar su correspondiente respuesta al impulso $h_{\text{inv}}[n]$. Del siguiente par de transformación:

$$n\alpha^n u[n] \overset{z}{\longleftrightarrow} \frac{\alpha z^{-1}}{(1 - \alpha z^{-1})^2} \quad |z| > |\alpha|,$$

podemos expresar a $H_{\text{inv}}(z)$ como $z \cdot \frac{\alpha z^{-1}}{(1-\alpha z^{-1})^2}, |z| > 1$ para $\alpha = 1$. Luego, usando la propiedad de desplazamiento en espacio de muestras (**time shifting**):

$$\mathcal{Z}^{-1}\left\{ z \cdot \frac{z^{-1}}{(1 - z^{-1})^2} \right\} = (n+1)u[n+1] = (n+1)u[n].$$[6]

Finalmente, para determinar si T_{inv} corresponde a un sistema BIBO estable, basta con verificar que el diagrama de polos y ceros tiene un polo en $z = 1$. Dado que la región de convergencia de $H_{\text{inv}}(z)$ no incluye al circulo unitario, se puede afirmar que se trata de un **sistema inestable**. El Script 6 muestra el uso de `filter()` para obtener la secuencia original $x[n]$ a partir de la secuencia $y[n]$. La Figura 9 describe la respuesta al impulso del sistema inverso, la cual coincide con la expresión obtenida de forma analítica.

Script 6: Sistema inverso

```
% —— f. sistema inverso a partir de transformada Z —— %
a_inv= [ 1 -2 1];    % coeficientes del denominador para hinv
b_inv= 1;    % coeficientes dle numerador para hinv
z01_v= filter( b_inv, a_inv, y01_v);    % usar filter para ...
    implementar el sistema inverso y obtener su respuesta ante y[n]
```

[6]Dado que el escalón inicia en $n = -1$ y para dicho valor $(n+1) = 0$, entonces podemos simplificar la expresión y verificar que se trata de una secuencia causal.

```
fig06= figure;
subplot( 3, 1, 1); stem( n_v, x01_v); xlabel( 'n'); ylabel( 'x[n]'); ...
    title( 'Secuencia de entrada x');
subplot( 3, 1, 2); stem( n_v, y01_v, 'r'); xlabel( 'n'); ylabel( ...
    'y[n]'); title( 'Respuesta de sistema Td2 ante x');
subplot( 3, 1, 3); stem( n_v, z01_v, 'g'); xlabel( 'n'); ylabel( ...
    'z[n]'); title( 'Respuesta de sistema inverso ante y');

% --- analisis de sistema inverso --- %
M= 256;
m_v= 0: M- 1;
hinv_v= impz( b_inv, a_inv, M);

fig07= figure;
stem( m_v, hinv_v); xlabel( 'n'); ylabel( 'h_{inv}[n]'); title( ...
    'Respuesta la impulso del sistema inverso');

fig08= figure;
zplane( b_inv, a_inv);  % diagrama de polos y ceros del sistema inverso
```

4. Se cuenta con los sistemas T_1 y T_2, descritos en la Figura 10. El sistema T_1 es implementado de forma recursiva y el sistema T_2 de forma no recursiva a partir de bloques elementales, donde $a \in \mathbb{R}$. Se pide analizar las propiedades de ambos sistemas y modificar sus implementaciones:

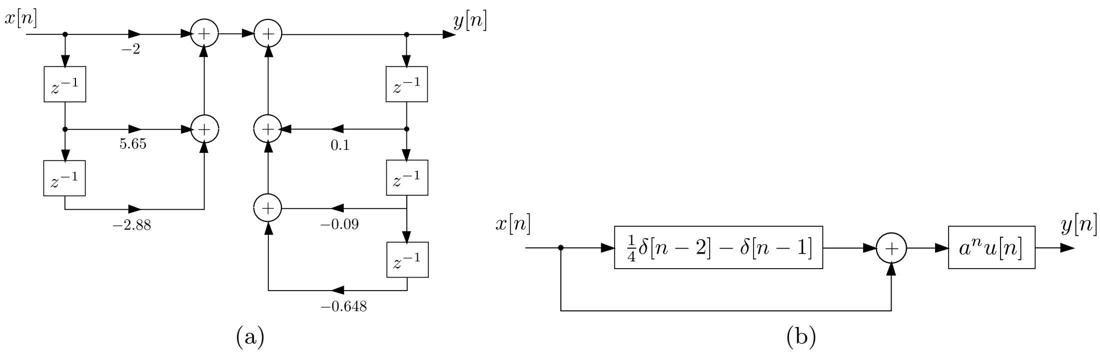


Figura 10: Sistemas en tiempo discreto. (a) Sistema T_1. (b) Sistema T_2.

a. Determinar analíticamente la ecuación de diferencias de T_1. Luego, determinar su función de transferencia $H_1(z)$ a partir de fracciones parciales usando `residuez()`. Incluir ambas expresiones en comentarios. La expresión en fracciones parciales debería tener la siguiente forma, donde x^* corresponde al complejo conjugado del escalar x:

$$H_1(z) = \frac{\alpha}{1-\beta z^{-1}} + \frac{\alpha^*}{1-\beta^* z^{-1}} + \frac{\gamma}{1-\delta z^{-1}}.$$

b. Expresar $H_1(z)$ como la suma de 2 términos racionales de la siguiente forma:

$$H_1(z) = \frac{c_0 + c_1 z^{-1}}{1 + d_1 z^{-1} + d_2 z^{-2}} + \frac{e_0}{1 + f_1 z^{-1}},$$

de tal manera que los coeficientes de ambos términos sean números reales[7]. Luego, **asumiendo sistema en reposo**, determinar analíticamente $\mathcal{Z}^{-1}\{H_1(z)\} = h_1[n]$ usando los siguientes pares de transformación:

$$\alpha^n u[n] \overset{z}{\longleftrightarrow} \frac{1}{1-\alpha z^{-1}}, \quad |z| > |\alpha|$$

$$A_c r^n \cos(\pi v_0 n)u[n] + A_s r^n \sin(\pi v_0 n)u[n] \overset{z}{\longleftrightarrow} \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \quad |z| > |r|,$$

donde:

- $b_0 = A_c$,
- $b_1 = r[A_s \sin(\pi v_0) - A_c \cos(\pi v_0)]$,
- $a_1 = -2r\cos(\pi v_0)$,
- $a_2 = r^2$.

Finalmente, crear $h_1[n]$ para $n \in \{0, 1, \ldots, 127\}$ según la expresión hallada y verificar que corresponde a aquella obtenida usando `impz()`. Graficar y tabular ambas secuencias en una sola ventana a partir de `subplot()` y `stem()`.

c. Determinar los polos y ceros del sistema T_1, así como sus posibles regiones de convergencia, a partir de `tf2zpk()` y graficar sus ubicaciones a partir de `zplane()`. Incluir en comentarios las propiedades de cada posible región de convergencia: (i) duración; (ii) BIBO estabilidad; (iii) sistema causal, anticausal o bilateral.

d. Para el sistema T_2, considerar que a puede tomar los dos siguientes valores: $a = \{\frac{1}{2}, \frac{1}{3}\}$. Para cada caso, determinar analíticamente la función de transferencia $H_2(z)$ y graficar su diagrama de polos y ceros a partir de `zplane()`. Existe cancelación de polos y ceros en algún caso? Incluir las expresiones obtenidas y su respuesta en comentarios.

e. **Asumiendo sistema en reposo**, determinar $h_2[n], n \in \{0, 1, \ldots, 127\}$ para cada posible valor de a usando `impz()` e indicar sus propiedades: (i) duración; (ii) BIBO estabilidad. Incluir su respuesta en comentarios.

f. Dada la secuencia de entrada $x[n]$ para $n \in \{0, 1, \ldots, 127\}$:

$$x[n] = 2\cos\left(\frac{2\pi n}{256}\right),$$

Determinar $T_2\{x[n]\}$ para ambos valores de a a partir de su forma recursiva usando `filter()`, graficar y rotular las secuencias adecuadamente. Es posible obtener la respuesta de forma no recursiva en algun caso? En caso sea cierto, obtener $T_2\{x[n]\}$ de forma no recursiva y verificar que es equivalente a aquella obtenida de forma recursiva. Caso contrario, justificar claramente por qué no es posible.

[7] La expresión obtenida en 4a solo permite una forma de representar $H_1(z)$ como la suma de 2 expresiones racionales de coeficientes reales. Una vez identificados los 2 términos a combinar, los coeficientes de la expresión resultante pueden ser obtenidos a partir de `[b, a]=residuez( r, p, k)`. Revisar la documentación de la subrutina indicada.