

Introducción al procesamiento de señales

Trabajo con Utilitario

Emilio Daniel Ballardini 1677/3

Sofia, Martin Peinetti 1459/4

Introducción:

Este trabajo con utilitario se enfoca en la utilización de Matlab u Octave como herramienta para el análisis y descripción de sistemas discretos, en particular de tipo lineal e invariante al desplazamiento (o en el tiempo).

Tabla de contenidos:

1. Ejercicio primero

- a. Inciso 1: En este inciso se solicita a lxs estudiantes que grafiquen la señal entregada 'senial.m', calculen su TFTD y realicen un gráfico de la misma.
- b. Inciso 2: Este inciso solicita a lxs estudiantes calcular la respuesta impulsional y en frecuencia de cada uno de los sistemas dados.
- c. Inciso 3: En este inciso se solicita a lxs estudiantes que propongan de manera imaginativa señales de variables independientes discretas, para ingresarlas a los sistemas trabajados anteriormente, con la finalidad de verificar que las respuestas impulsionales calculadas coinciden con los resultados obtenidos, así mismo corroborar lo mismo con sus TFTDs correspondientes.
- d. Inciso 4: Este inciso es resuelto de manera unívoca con el anterior enunciado, ya que en el mismo se utilizó la señal generada en el inciso 1.
- e. Inciso 5: Este inciso solicita a lxs estudiantes calcular las TFTDs correspondientes a cada señal. Identificar los diferentes componentes en frecuencia obtenidos y generar una breve conclusión del sistema.

2. Ejercicio segundo:

“La señal $h_A[n]$ retornada por la función `hcanald.m` representa la respuesta impulsional de un sistema SLID FIR que modela el comportamiento de un canal digital con dos ecos.”

- a. Inciso 1: *Grafique esta respuesta al impulso e interprete en qué consiste el modelo mencionado. Escriba la ecuación en diferencias que representa el sistema. ¿Qué puede decir acerca de la estabilidad del sistema?*
- b. Inciso 2: *Escriba en MATLAB una función que implemente un sistema con esta ecuación en diferencias. Verifique que este sistema tiene la misma respuesta impulsional que la obtenida en 1.*

“Por otra parte, el comando `1 [x, fs] = audioread('audio.wav');` permite cargar un archivo de audio (en este caso, el provisto entre los archivos de la

presente práctica) en la variable x . La variable fs indica la frecuencia de muestreo de la señal, concepto que estudiaremos más adelante. Como cualquier otra variable, se puede graficar utilizando el comando `plot(x)`. Es posible escuchar la señal mediante el comando: `sound(x,fs);`”

- c. [Inciso 3:](#) Usando la función escrita en 2. halle la salida del sistema (que llamaremos y) cuando se aplica a su entrada la señal de audio x . ¿Cómo se escucha esta señal? Grafique la señal de entrada y salida en el tiempo.

Una forma de intentar corregir el efecto que introduce el canal consiste en colocar un filtro FIR en cascada con el canal intentando que la respuesta impulsional total se aproxime lo más posible a una única delta de Kronecker. ¿Por qué queríamos lograr esto? Por ejemplo, si la respuesta impulsional del canal fuese $h_c[n] = \delta[n] + \alpha \delta[n - n_0]$, un filtro con una sola rama de retardo sería de la forma $y[n] = x[n] - \alpha x[n - n_0]$. ¿Qué efecto tiene este sistema? ¿Por qué la corrección no es perfecta? ¿Cómo añadir a una segunda rama de retardos que mejore esto?

- d. [Inciso 4:](#) “Con esta idea implemente un sistema que intente corregir el efecto del canal. Pruebe con 2 y 3 ramas de retardo. Explícite la ecuación en diferencias. En cada caso grafique la respuesta impulsional de la cascada (canal + filtro).”
- e. [Inciso 5:](#) “Filtre la señal y con estos sistemas y escuche como suena. Compare cada una de estas señales con la señal x original.

Ejercicio primero:

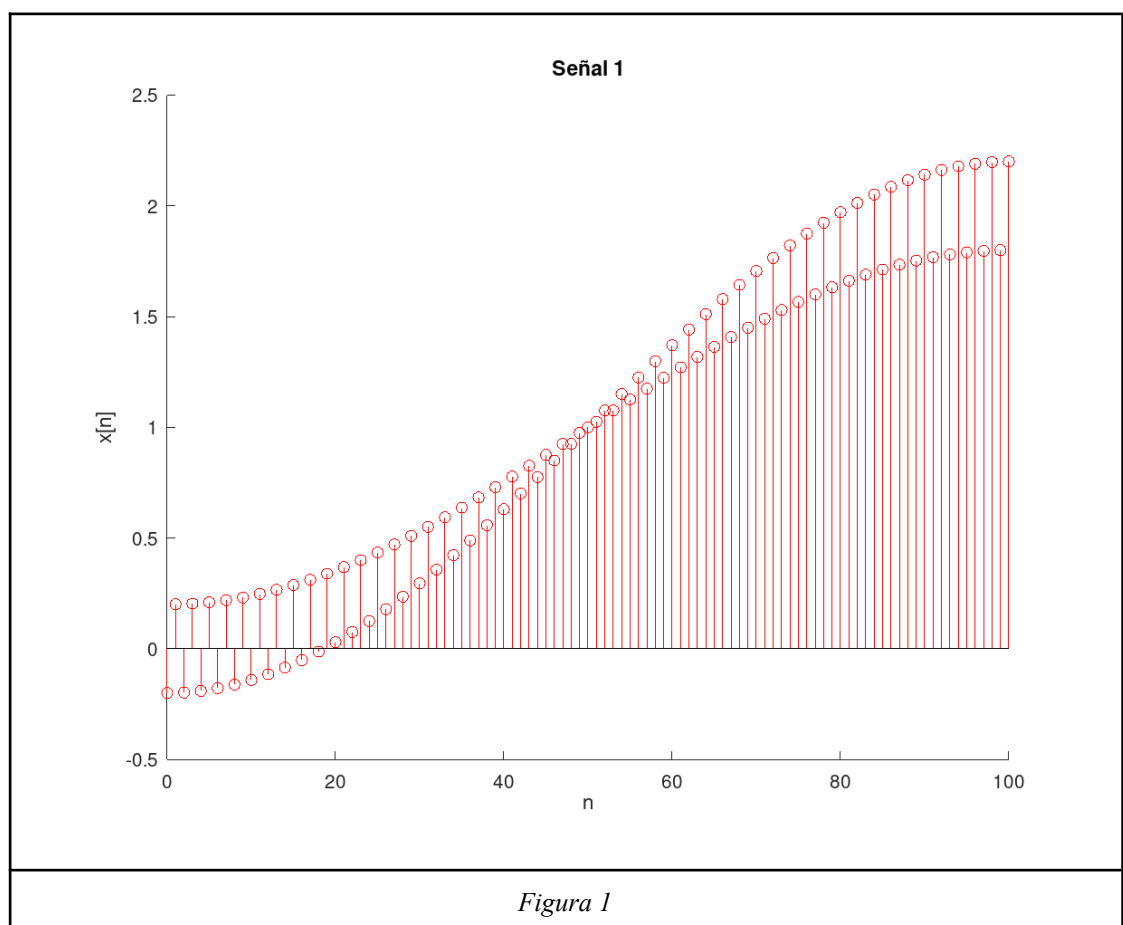
Inciso primero:

“La función `senal.m` retorna una señal de variable independiente discreta, y los instantes n en los que está definida la misma.

Grafique esta señal. Obtenga la TFTD de la misma y grafíquela. ¿Qué componentes de frecuencia identifica?”

Para representar la señal dada por la cátedra, se utilizó el siguiente comando:

`[n, x] = senal(16773);`



Para hacer la TFTD se creó un script con el siguiente código:

```
function [ X,s ] = TFTD( x,n );
```

```
ds = 0.001;
```

```
s = [-0.5:ds:0.5];
```

```

X = zeros(size(s));

for k = 1:length(s)

    X(k)=sum(x.*exp((-1)*i*2*pi*s(k)*n));

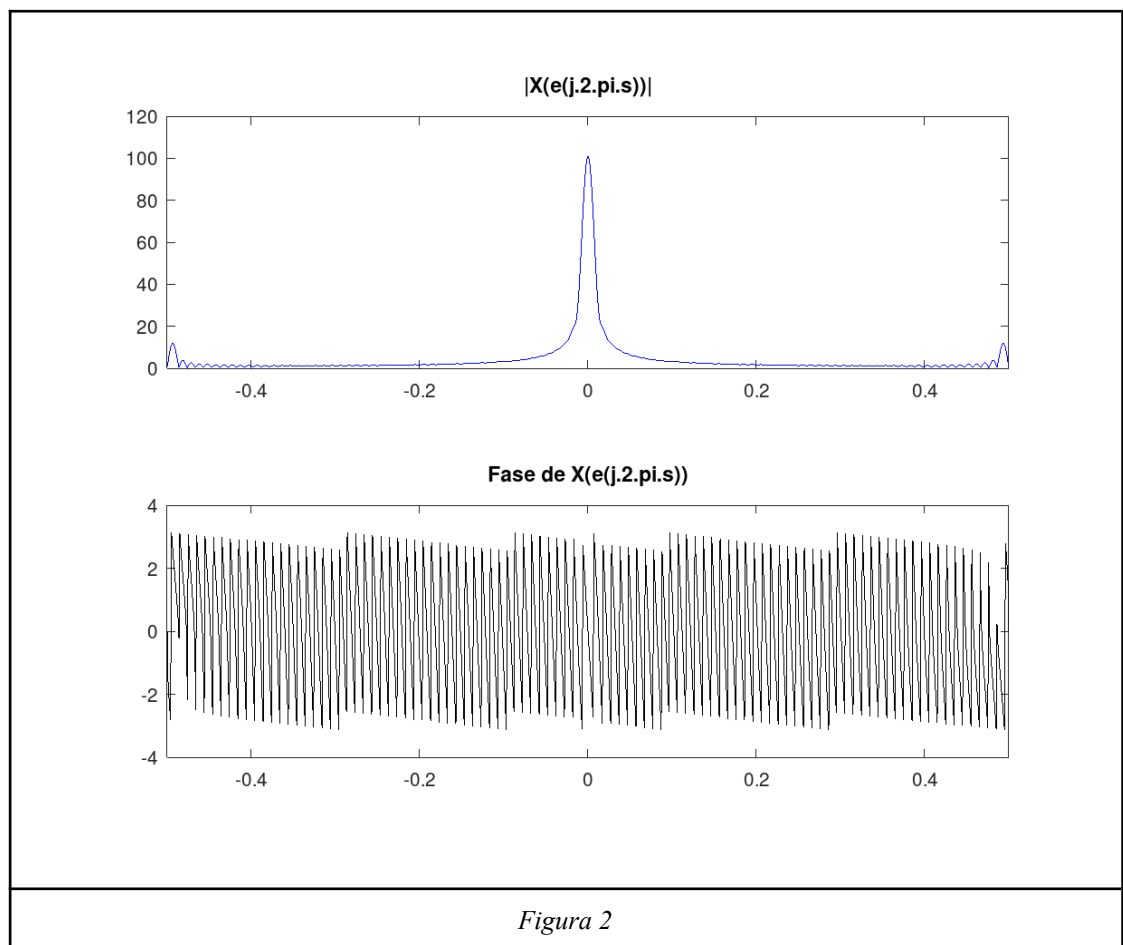
end

end

```

Una vez calculada la TFTD de la señal.

Graficamos módulo y fase ya que es una función compleja.

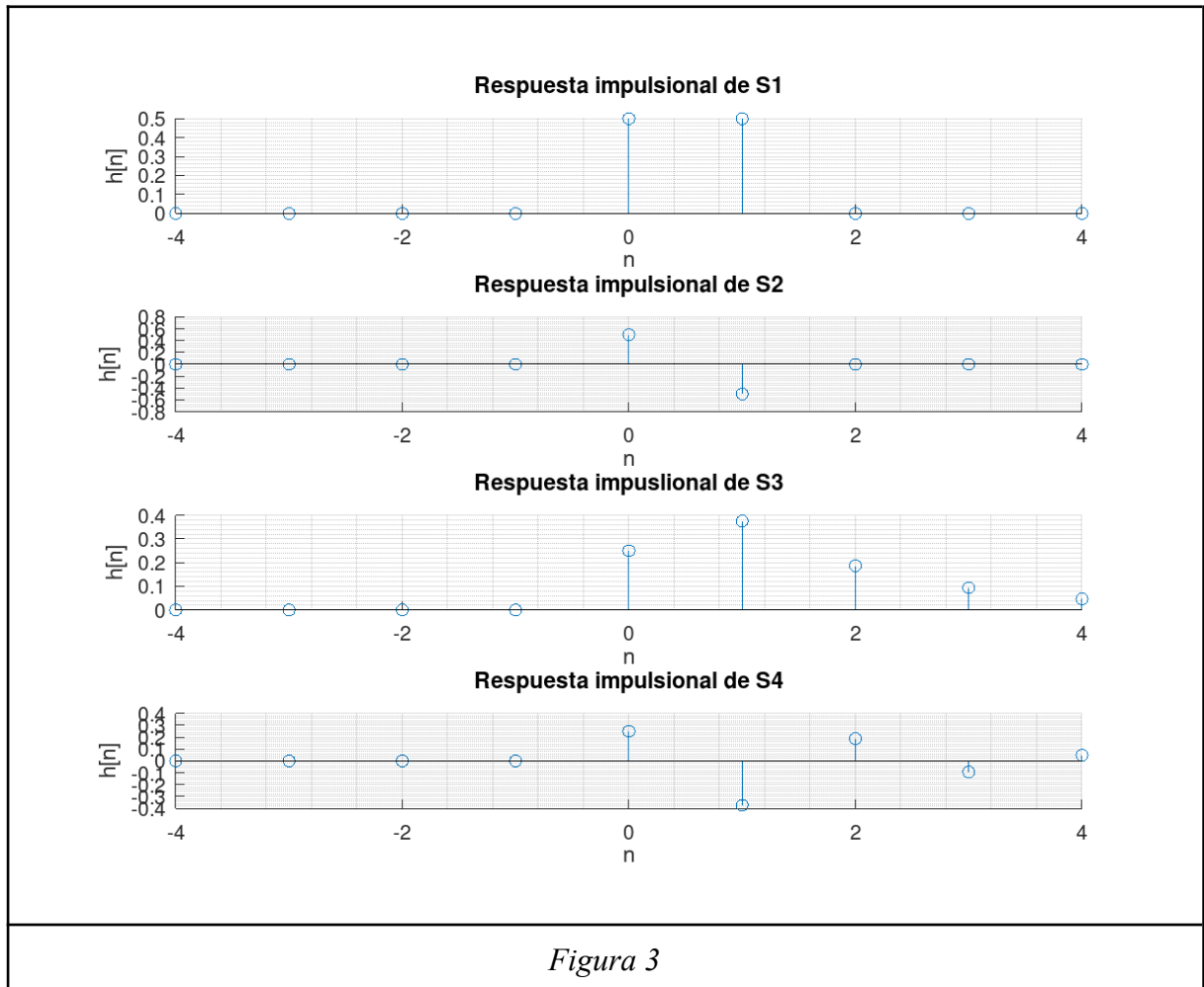


En la gráfica podemos ver que tenemos un pico (gran amplitud) en componentes de baja frecuencia (cerca de 0) y componentes con alta frecuencia de menor amplitud en las altas frecuencias (cerca de -0,5 y 0,5).

Inciso segundo:

Para esto definimos la entrada como una Delta de Kronecker.

$\Delta = [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0];$



$$Y1[n] = \frac{1}{2} x[n] + \frac{1}{2} x[n - 1]$$

$$Y2[n] = \frac{1}{2} x[n] - \frac{1}{2} x[n - 1]$$

$$Y3[n] = \frac{1}{4} x[n] + \frac{1}{4} x[n - 1] + \frac{1}{2} y3[n - 1]$$

$$Y4[n] = \frac{1}{4} x[n] - \frac{1}{4} x[n - 1] - \frac{1}{2} y4[n - 1]$$

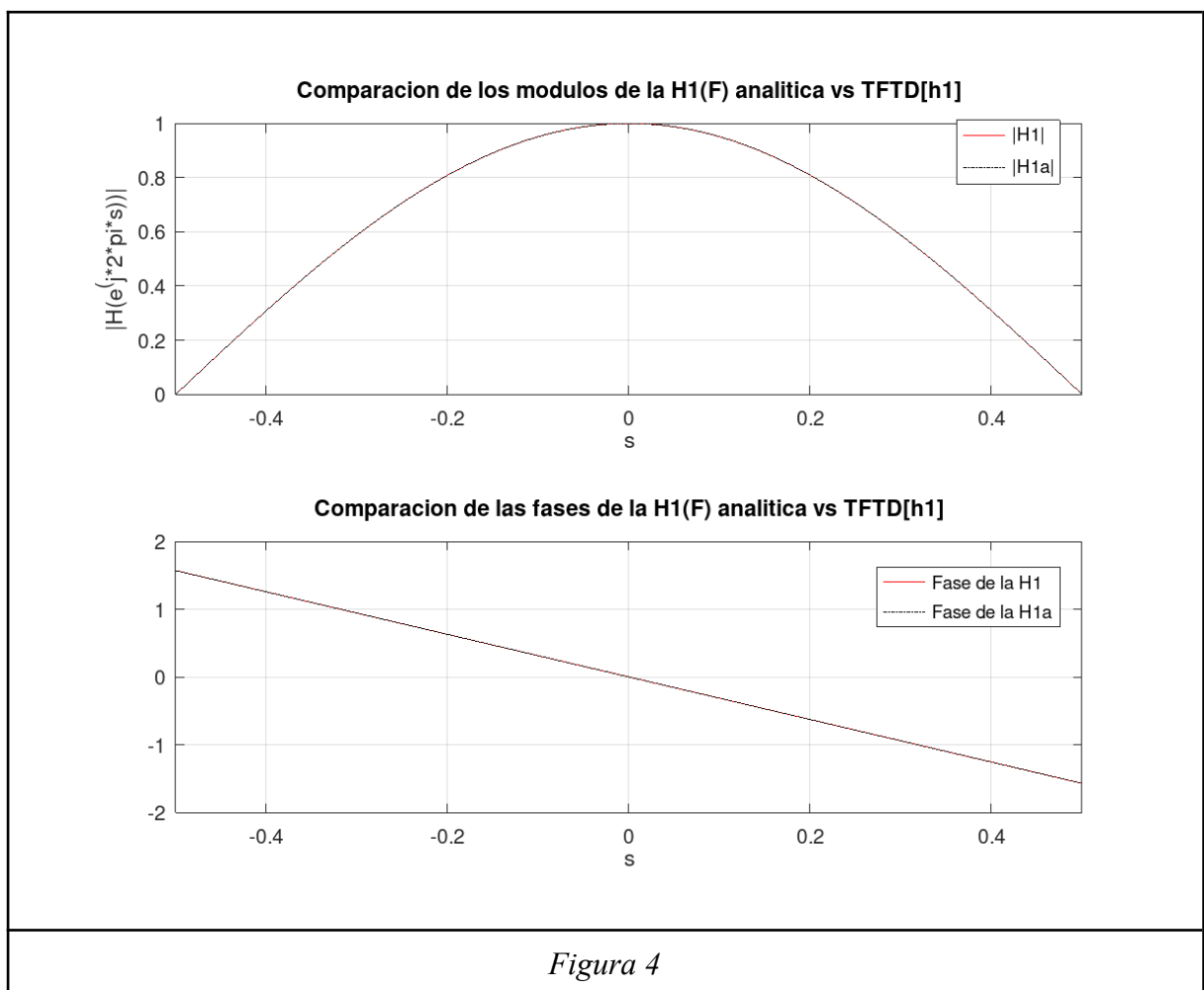
Mirando las gráficas y las ecuaciones podemos sacar las siguientes conclusiones:

Para el sistema Y_1 , podemos decir que se trata de un filtro FIR (respuesta impulsional finita). Este filtro se conoce como promedio móvil, va calculando el promedio (en este caso entre dos muestras) y da el resultado como salida. Eso va disminuyendo todo lo que sean variaciones rápidas y mantiene lo que esté más a baja frecuencia, por lo que tendría un comportamiento de tipo “pasabajos”. Se podría decir que es una aproximación a la integral.

El sistema Y_2 también es un FIR, pero hace lo contrario al Y_1 , ya que resta las dos muestras, por lo que si son iguales se cancelarían. También ocurre lo mismo para las muestras “constantes”, se cancelan. Esto quiere decir que es un “pasaaltos”. Este sistema es una aproximación a la derivada.

Los sistemas Y_3 e Y_4 son sistemas IIR (respuesta impulsional infinita) por la realimentación que se le agrega ($\frac{1}{2} Y[n-1]$). Podríamos pensar que dicha realimentación haría que el sistema se puede “inestabilizar”, pero el $\frac{1}{2}$ que multiplica al $Y[n-1]$ soluciona esto. Con esto quiero decir que esta realimentación mejora los filtros anteriores Y_1 e Y_2 .

Ahora veremos los gráficos en los cuales se muestra una comparación entre las respuestas en frecuencia de los sistemas del inciso 2 que fueron obtenidas de manera analítica por un lado y con nuestro script TFTD.m del inciso 1:



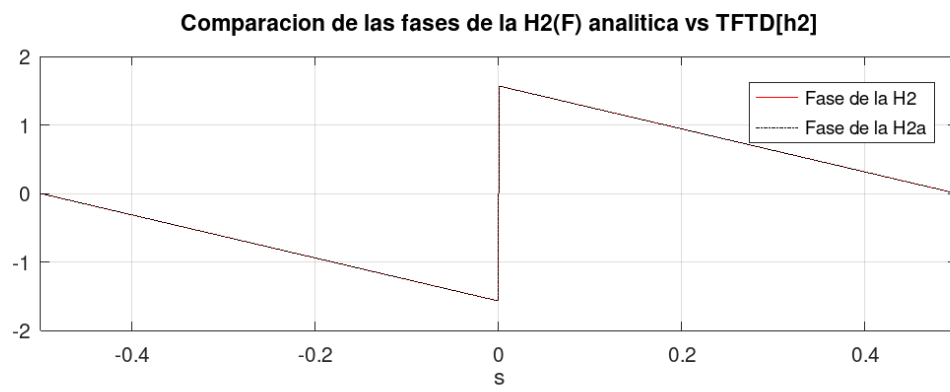
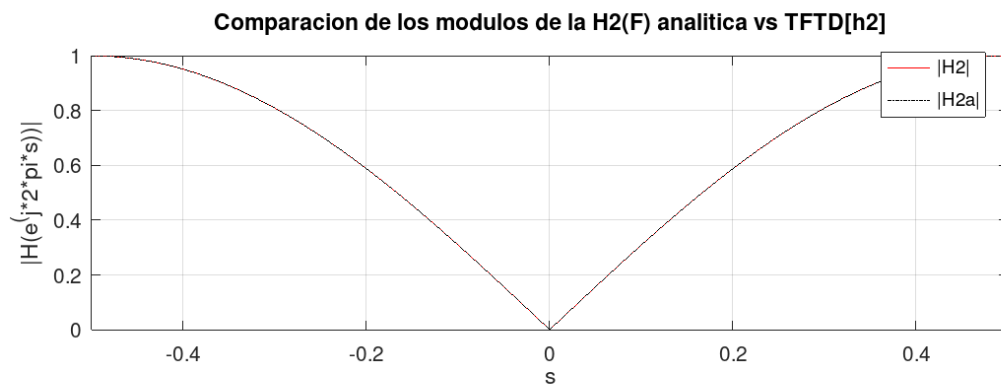


Figura 5

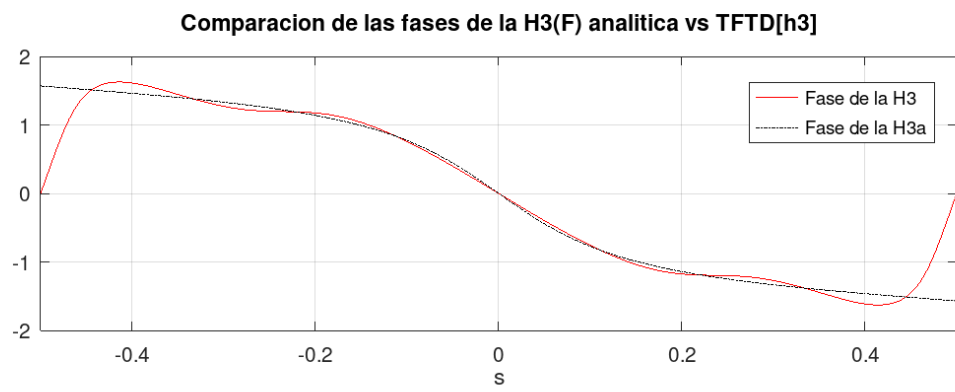
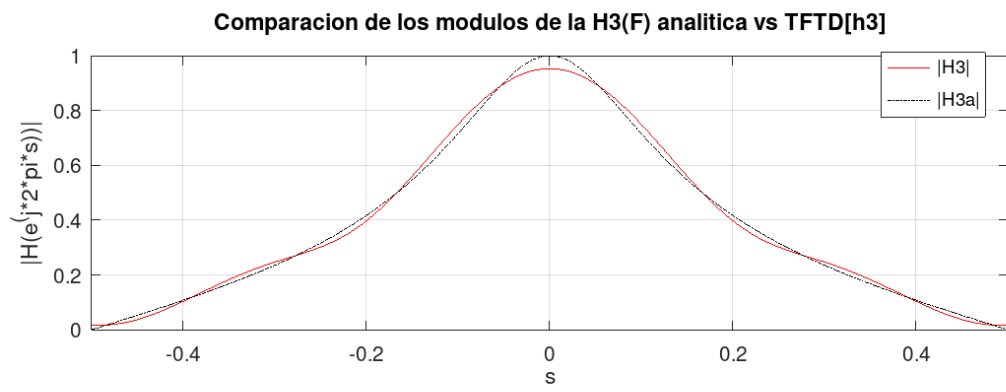
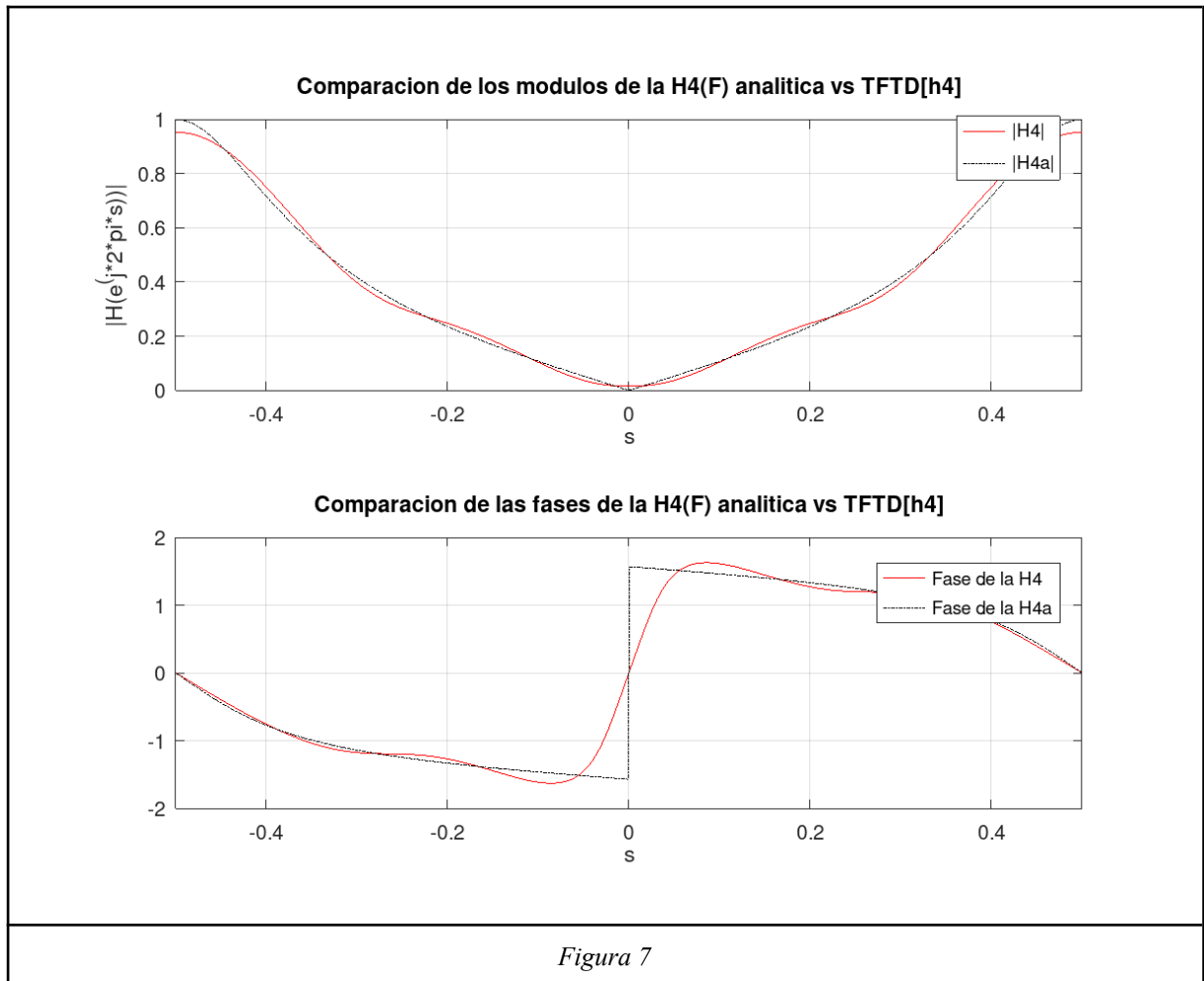


Figura 6



El hecho de que las gráficas de las respuestas en frecuencia de los sistemas IIR calculadas de manera analítica, en comparación con las calculadas de manera numérica, se debe a que el cálculo de la TFTD requiere infinitos valores, y dado que trabajamos con valores truncados, la aproximación de esta no es precisa; en caso de querer un mayor detalle se debería utilizar mas valores.

Inciso tercero:

Las respuestas en frecuencia de manera analítica fueron obtenidas con la propiedad siguiente:

$$Y(e^{j2\pi s}) = H(e^{j2\pi s}) \cdot X(e^{j2\pi s}) \rightarrow H(e^{j2\pi s}) = Y(e^{j2\pi s}) / X(e^{j2\pi s}).$$

$$y_1[n] = \frac{1}{2} * x[n] + \frac{1}{2} * x[n - 1]$$

$$y_2[n] = \frac{1}{2} * x[n] - \frac{1}{2} * x[n - 1]$$

$$y_3[n] = \frac{1}{4} * x[n] + \frac{1}{4} * x[n - 1] + \frac{1}{2} * y_3[n - 1]$$

$$y_4[n] = \frac{1}{4} * x[n] + \frac{1}{4} * x[n - 1] - \frac{1}{2} * y_4[n - 1]$$

Para calcular la respuesta impulsional de los dos primeros sistemas basta con ingresar deltas en las entradas y obtendremos la respuesta impulsional, ya que son sistemas FIR (Finite Impulse Response):

$$h_1[n] = \frac{1}{2} * \delta[n] + \frac{1}{2} * \delta[n - 1] \rightarrow \text{Dos deltas, una centrada en 0 y la otra en 1}$$

$$h_2[n] = \frac{1}{2} * \delta[n] - \frac{1}{2} * \delta[n - 1] \rightarrow \text{Dos deltas, una centrada en 0 y la otra en 1}$$

Para los otros dos sistemas no es tan sencillo ya que son sistemas IIR (Infinite Impulse Response), para esto lo que hacemos es primero ingresar deltas al sistema, y luego ir dando valores al mismo para ver qué resultado obtenemos; se asume condiciones iniciales nulas.

$$h_3[n] = \frac{1}{4} * \delta[n] + \frac{1}{4} * \delta[n - 1] + \frac{1}{2} h_3[n - 1]$$

$$n = 0 \rightarrow h_3[0] = \frac{1}{4} * 1 + \frac{1}{4} * 0 + \frac{1}{2} * 0 = \frac{1}{4}$$

$$n = 1 \rightarrow h_3[1] = \frac{1}{4} * 0 + \frac{1}{4} * 1 + \frac{1}{2} * \frac{1}{4} = \frac{3}{8}$$

$$n = 2 \rightarrow h_3[2] = \frac{1}{4} * 0 + \frac{1}{4} * 0 + \frac{1}{2} * \frac{3}{8} = \frac{3}{16}$$

$$n = 3 \rightarrow h_3[3] = \frac{1}{4} * 0 + \frac{1}{4} * 0 + \frac{1}{2} * \frac{3}{16} = \frac{3}{32}$$

$$n = 4 \rightarrow h_3[4] = \frac{1}{4} * 0 + \frac{1}{4} * 0 + \frac{1}{2} * \frac{3}{32} = \frac{3}{64}$$

$$h_4[n] = \frac{1}{4} * \delta[n] - \frac{1}{4} * \delta[n - 1] - \frac{1}{2} h_4[n - 1]$$

$$n = 0 \rightarrow h_4[0] = \frac{1}{4} * 1 - \frac{1}{4} * 0 - \frac{1}{2} * 0 = \frac{1}{4}$$

$$n = 1 \rightarrow h_4[1] = \frac{1}{4} * 0 - \frac{1}{4} * 1 - \frac{1}{2} * \frac{1}{4} = -\frac{3}{8}$$

$$n = 2 \rightarrow h_4[2] = \frac{1}{4} * 0 - \frac{1}{4} * 0 - \frac{1}{2} * \left(-\frac{3}{8}\right) = \frac{3}{16}$$

$$n = 3 \rightarrow h_4[3] = \frac{1}{4} * 0 - \frac{1}{4} * 0 - \frac{1}{2} * \frac{3}{16} = -\frac{3}{32}$$

$$n = 4 \rightarrow h_4[4] = \frac{1}{4} * 0 - \frac{1}{4} * 0 - \frac{1}{2} * \left(-\frac{3}{32}\right) = \frac{3}{64}$$

Se puede verificar en Octave al descomentar la línea 14 del script 'Inciso2.m'

Para encontrar la $H(e^{j*2*\pi*s})$ de cada sistema, se calcula de manera directa:

$$y[n] \rightarrow Y(e^{j*2*\pi*s})$$

$$x[n] \rightarrow X(e^{j*2*\pi*s})$$

$$h[n] \rightarrow H(e^{j*2*\pi*s})$$

$$y_1[n] = \frac{1}{2} * x[n] + \frac{1}{2} * x[n - 1] \rightarrow$$

$$Y_1(e^{j*2*\pi*s}) = \frac{1}{2} * X_1(e^{j*2*\pi*s}) + \frac{1}{2} * X_1(e^{j*2*\pi*s}) * e^{j*2*\pi*s*(-1)}$$

$$Y_1(e^{j*2*\pi*s}) = X_1(e^{j*2*\pi*s}) * (\frac{1}{2} + \frac{1}{2} * e^{j*2*\pi*s*(-1)})$$

$$\frac{Y_1(e^{j*2*\pi*s})}{X_1(e^{j*2*\pi*s})} = (\frac{1}{2} + \frac{1}{2} * e^{j*2*\pi*s*(-1)}) = H_1(e^{j*2*\pi*s})$$

$$y_2[n] = \frac{1}{2} * x[n] - \frac{1}{2} * x[n - 1] \rightarrow$$

$$Y_2(e^{j*2*\pi*s}) = \frac{1}{2} * X_2(e^{j*2*\pi*s}) - \frac{1}{2} * X_2(e^{j*2*\pi*s}) * e^{j*2*\pi*s*(-1)}$$

$$Y_2(e^{j*2*\pi*s}) = X_2(e^{j*2*\pi*s}) * (\frac{1}{2} - \frac{1}{2} * e^{j*2*\pi*s*(-1)})$$

$$\frac{Y_2(e^{j*2*\pi*s})}{X_2(e^{j*2*\pi*s})} = (\frac{1}{2} - \frac{1}{2} * e^{j*2*\pi*s*(-1)}) = H_2(e^{j*2*\pi*s})$$

$$y_3[n] = \frac{1}{4} * x[n] + \frac{1}{4} * x[n - 1] + \frac{1}{2} * y_3[n - 1] \rightarrow$$

$$Y_3(e^{j*2*\pi*s}) = X_3(e^{j*2*\pi*s}) * (\frac{1}{4} + \frac{1}{4} * e^{j*2*\pi*s*(-1)}) + \frac{1}{2} * Y_3(e^{j*2*\pi*s}) * e^{j*2*\pi*s*(-1)}$$

$$Y_3(e^{j*2*\pi*s}) - \frac{1}{2} * Y_3(e^{j*2*\pi*s}) * e^{j*2*\pi*s*(-1)} = X_3(e^{j*2*\pi*s}) * (\frac{1}{4} + \frac{1}{4} * e^{j*2*\pi*s*(-1)})$$

$$Y_3(e^{j*2*\pi*s}) * (1 - \frac{1}{2} * e^{j*2*\pi*s*(-1)}) = X_3(e^{j*2*\pi*s}) * (\frac{1}{4} + \frac{1}{4} * e^{j*2*\pi*s*(-1)})$$

$$\frac{Y_3(e^{j*2*\pi*s})}{X_3(e^{j*2*\pi*s})} = \frac{(\frac{1}{4} + \frac{1}{4} * e^{j*2*\pi*s*(-1)})}{(1 - \frac{1}{2} * e^{j*2*\pi*s*(-1)})} = H_3(e^{j*2*\pi*s})$$

$$y_4[n] = \frac{1}{4} * x[n] - \frac{1}{4} * x[n - 1] - \frac{1}{2} * y_4[n - 1] \rightarrow$$

$$Y_4(e^{j*2*\pi*s}) = X_4(e^{j*2*\pi*s}) * (\frac{1}{4} - \frac{1}{4} * e^{j*2*\pi*s*(-1)}) - \frac{1}{2} * Y_4(e^{j*2*\pi*s}) * e^{j*2*\pi*s*(-1)}$$

$$Y_4(e^{j*2*\pi*s}) + \frac{1}{2} * Y_4(e^{j*2*\pi*s}) * e^{j*2*\pi*s*(-1)} = X_4(e^{j*2*\pi*s}) * (\frac{1}{4} - \frac{1}{4} * e^{j*2*\pi*s*(-1)})$$

$$Y_4(e^{j*2*\pi*s}) * (1 + \frac{1}{2} * e^{j*2*\pi*s*(-1)}) = X_4(e^{j*2*\pi*s}) * (\frac{1}{4} - \frac{1}{4} * e^{j*2*\pi*s*(-1)})$$

$$\frac{Y_4(e^{j*2*\pi*s})}{X_4(e^{j*2*\pi*s})} = \frac{(\frac{1}{4} - \frac{1}{4} * e^{j*2*\pi*s*(-1)})}{(1 + \frac{1}{2} * e^{j*2*\pi*s*(-1)})} = H_4(e^{j*2*\pi*s})$$

Despejando H, en Octave nos quedó el siguiente código:

$$H1a = 0.5 + 0.5 * \exp((-1) * i * 2 * \pi * s);$$

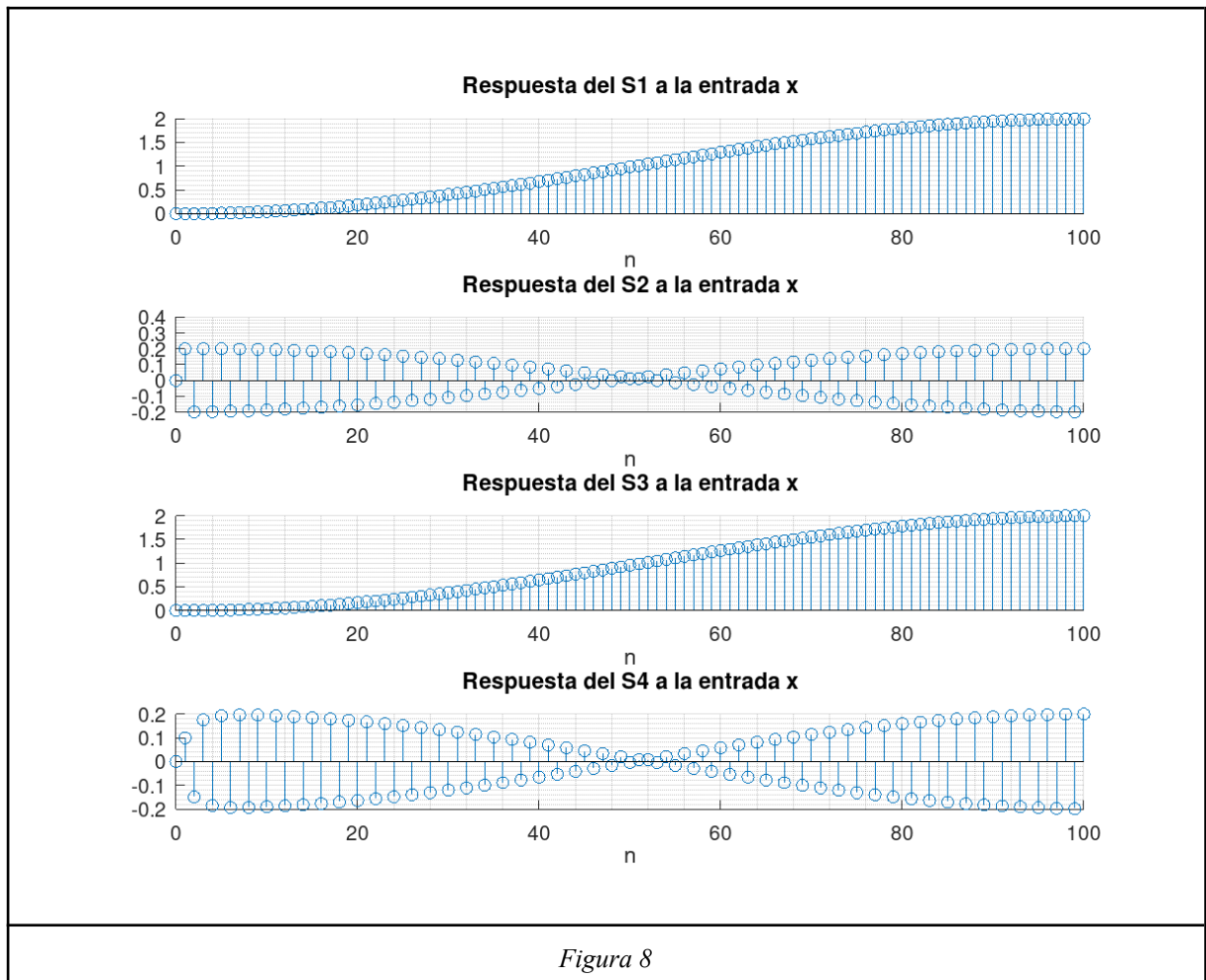
$$H2a = 0.5 - 0.5 * \exp((-1) * i * 2 * \pi * s);$$

$$H3a = (0.25 + 0.25 * \exp((-1) * i * 2 * \pi * s)) / (1 - 0.5 * \exp((-1) * i * 2 * \pi * s));$$

$$H4a = (0.25 - 0.25 * \exp((-1) * i * 2 * \pi * s)) / (1 + 0.5 * \exp((-1) * i * 2 * \pi * s));$$

Inciso cuarto:

Gráficos de las salidas de los sistemas nombrados a la señal del inciso 1.



Inciso quinto:

Gráficos de las TFTD de las salidas encontradas en el inciso anterior, utilizando nuestro script TFTD.m

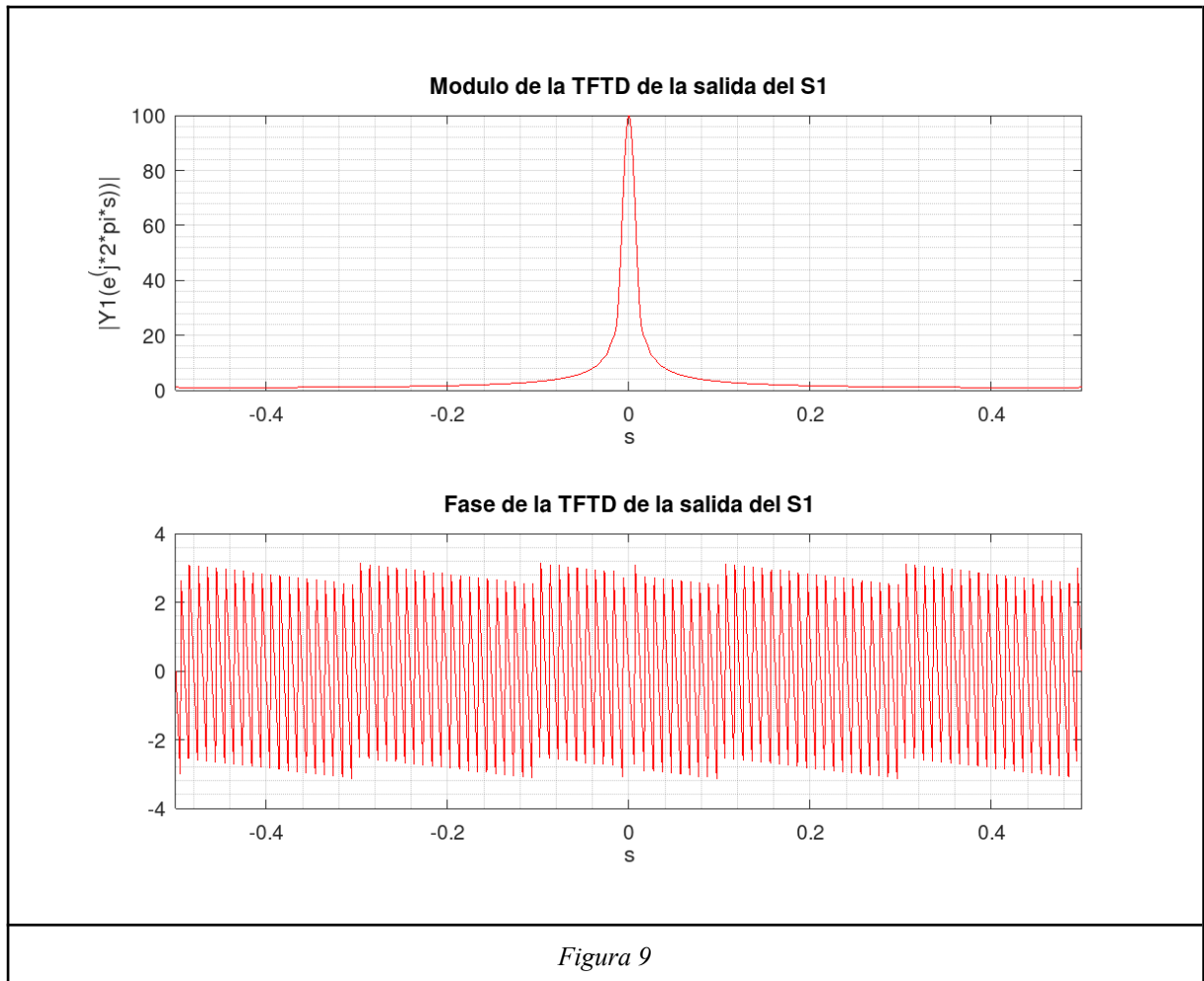


Figura 9

El filtro que se observa en la figura 9, elimina los componentes en alta frecuencia que se observaban en la gráfica de la señal de la figura 1, lo cual se corresponde al comportamiento de filtros “pasa bajos”.

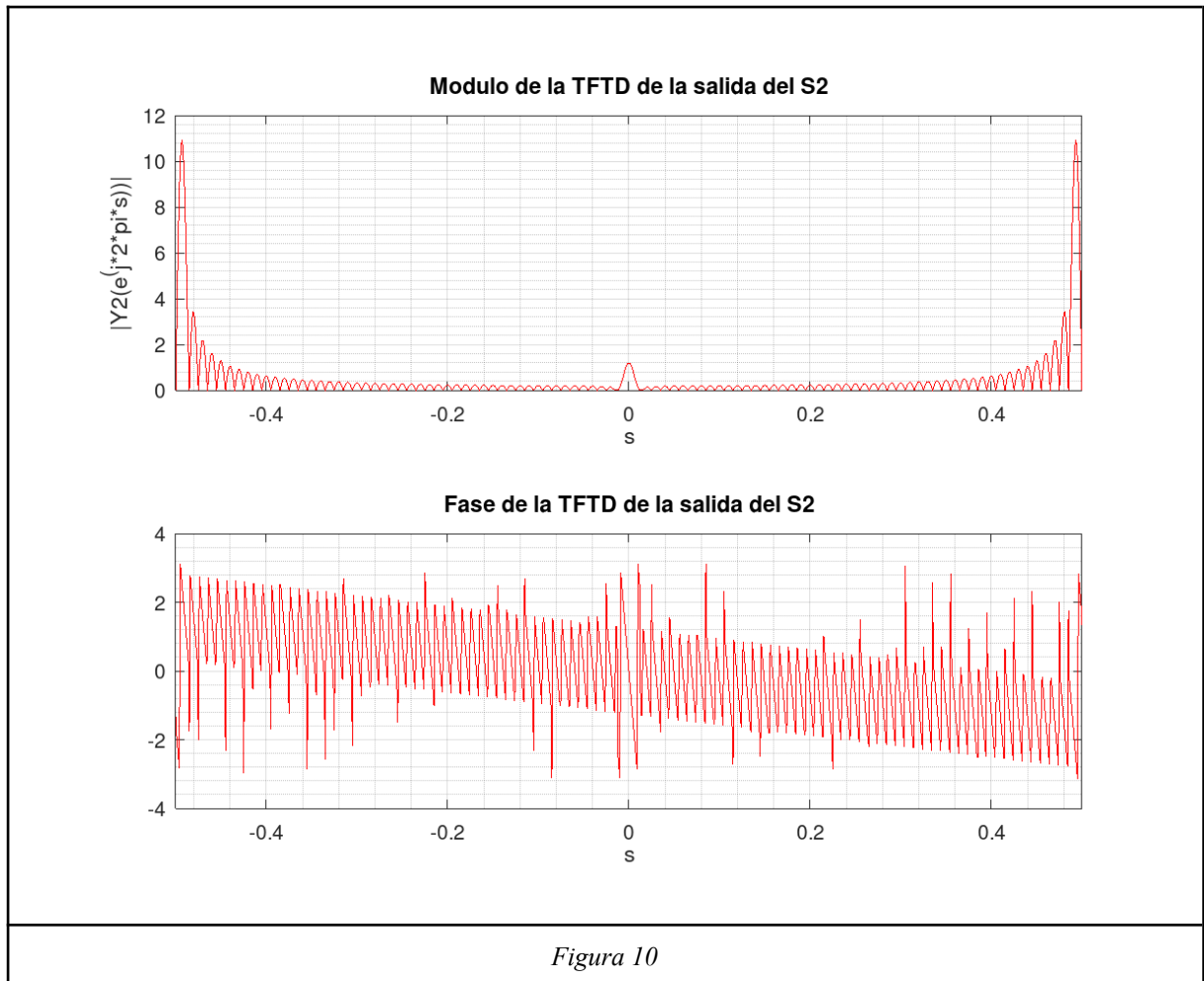


Figura 10

En la segunda señal podemos ver (figura 10) que se eliminó el pico en baja frecuencia, lo que nos permite ver con mayor detalle lo que ocurre en las altas frecuencias, siendo así, un filtro “pasa altos”

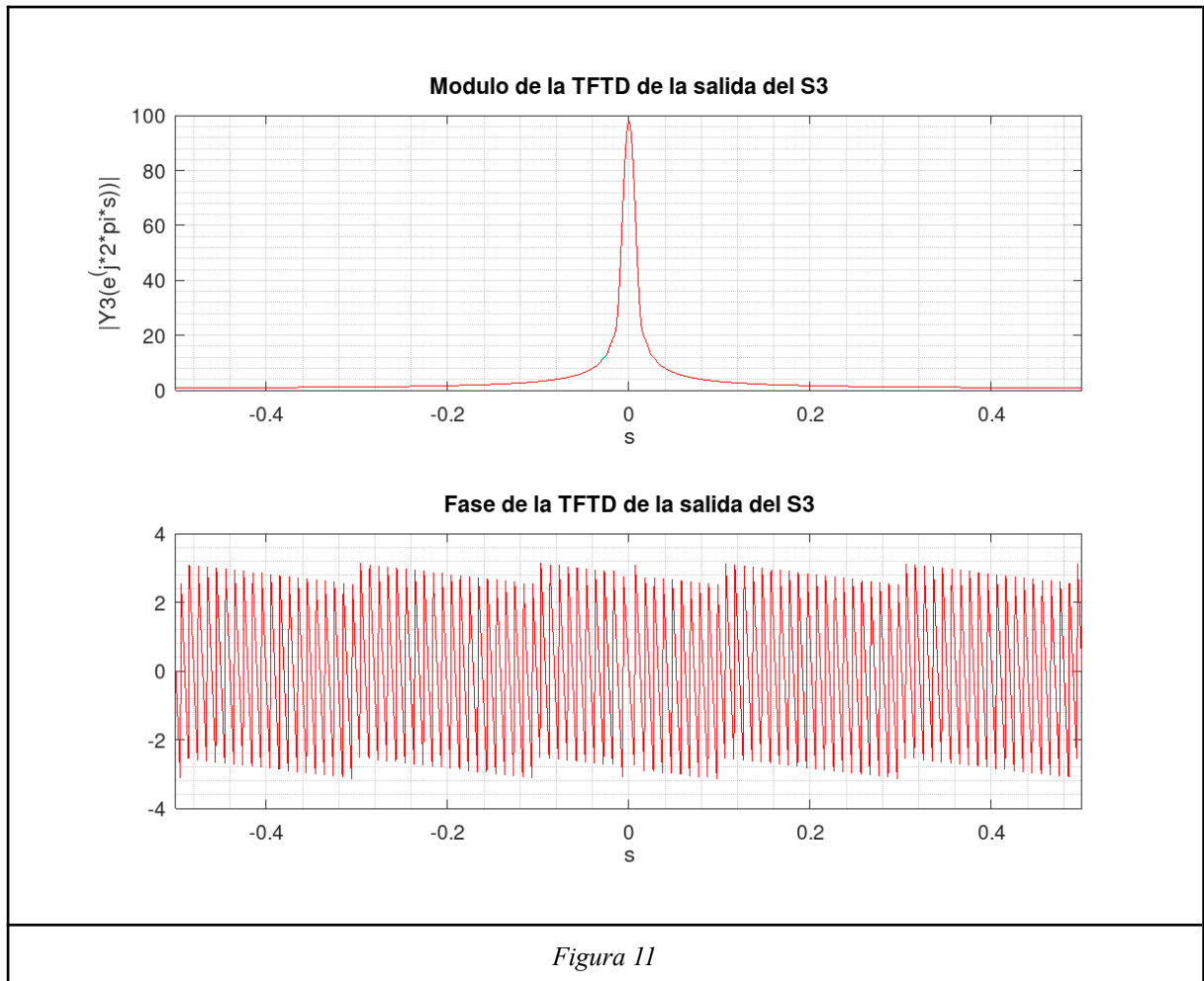


Figura 11

El filtro que se observa en la figura 11, elimina los componentes en alta frecuencia, lo cual se corresponde al comportamiento de filtros “pasa bajos”.

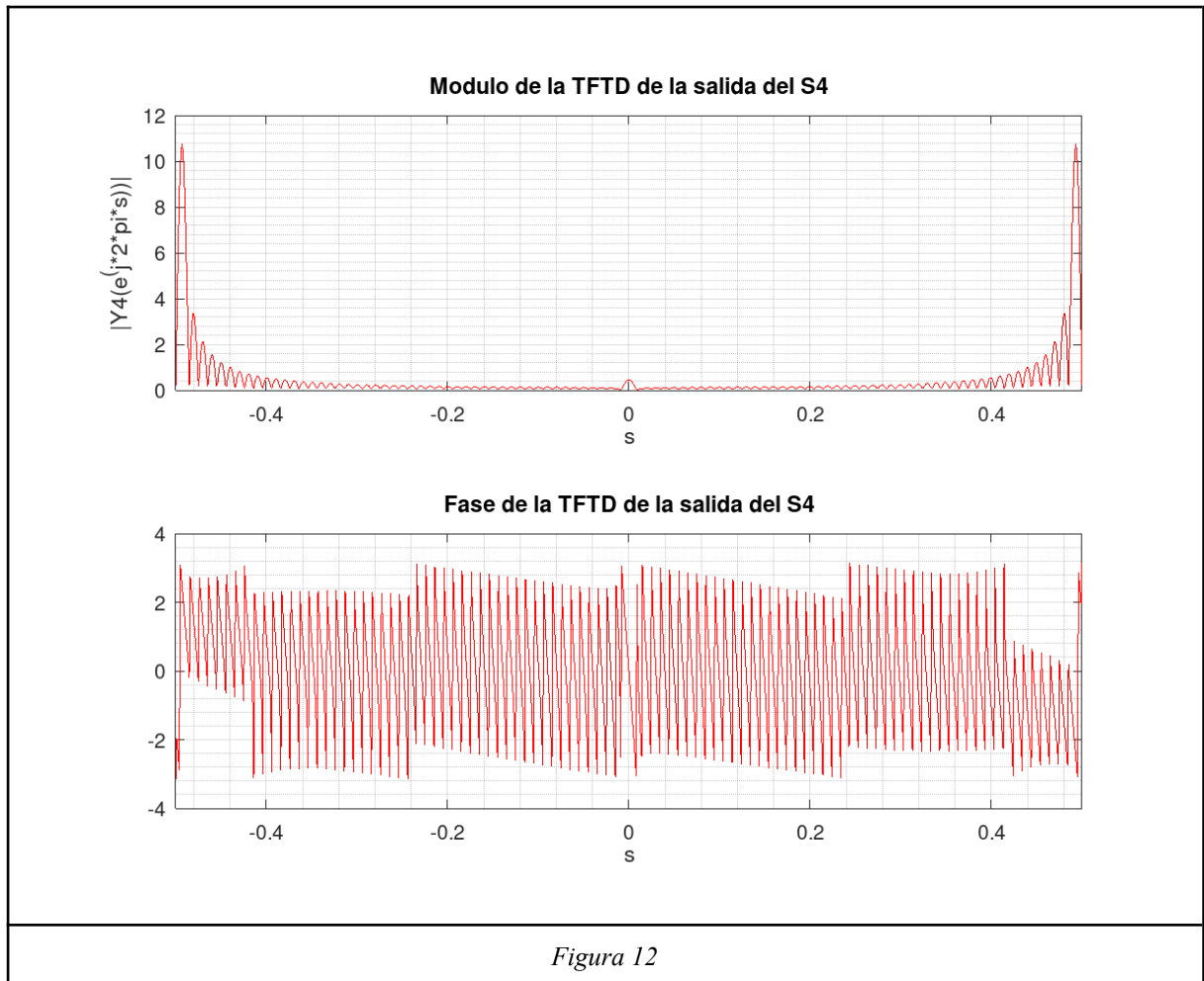


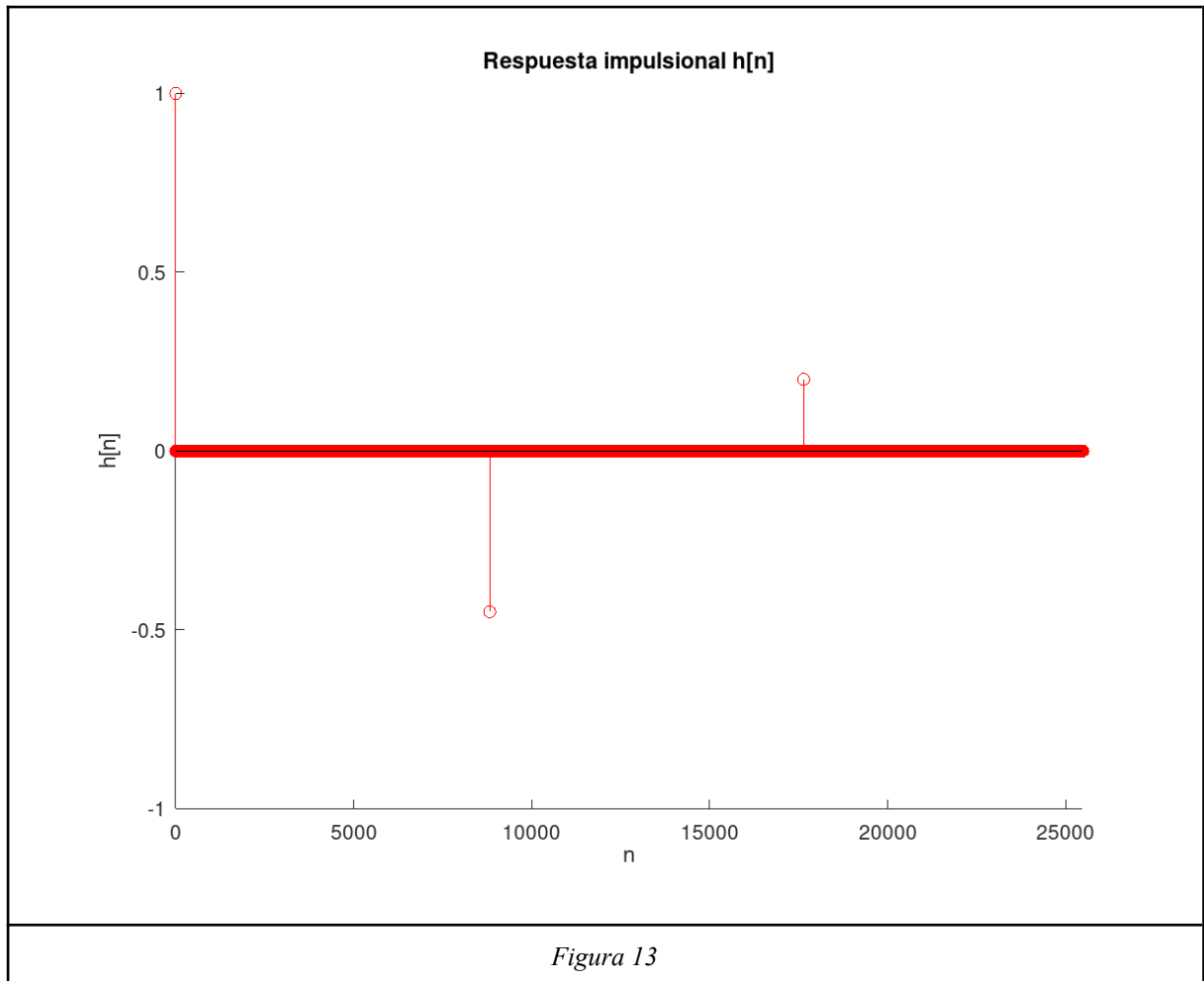
Figura 12

El filtro que se observa en la figura 12, elimina los componentes en bajafrecuencia, lo cual se corresponde al comportamiento de filtros “pasa altos”.

Ejercicio segundo:

Inciso primero II:

Calculamos y graficamos la respuesta impulsional:



La cual puede representarse con la ecuación:

$$h[n] = \delta[n] - 0,45 * \delta[n - 8820] + 0,2 * \delta[n - 17640]$$

La ecuación en diferencias que representa es la siguiente:

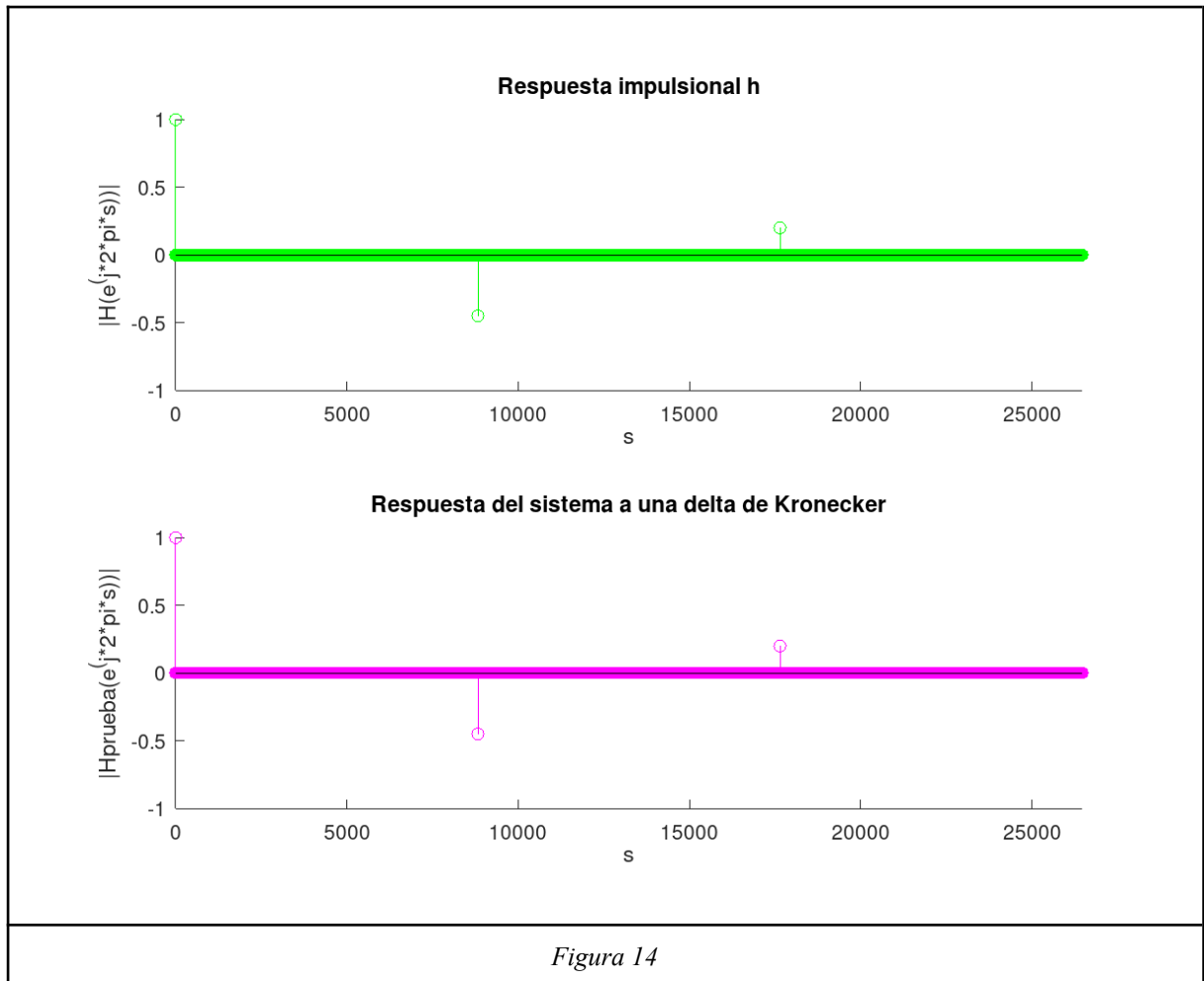
$$y[n] = x[n] - 0,45 * x[n - 8820] + 0,2 * x[n - 17640]$$

Sobre la estabilidad del sistema, podemos asegurar que es estable, ya que su respuesta impulsional es absolutamente sumable.

Es un sistema FIR.

Inciso segundo II:

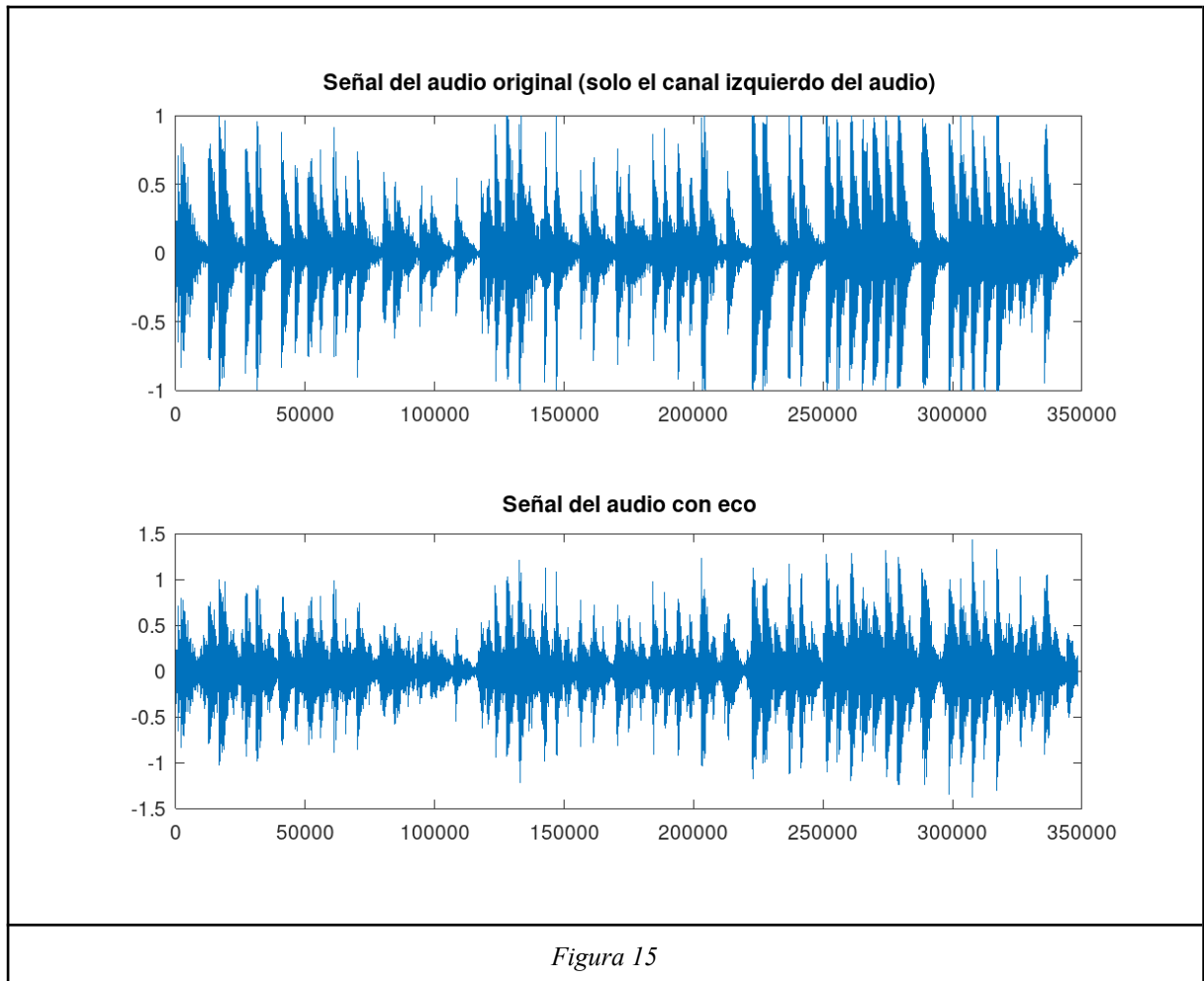
En la siguiente gráfica (figura 14) podemos observar una comparación entre las respuestas impulsionales de una función generada en Octave, y la del inciso anterior.



Inciso tercero II:

Como la señal entra a un sistema con ecuaciones en diferencias, esto produce eco.

Podemos observar lo dicho en la siguiente gráfica (figura 15)



La primera gráfica de la figura 15 presenta una sola señal ya que se utilizó un solo canal de la señal original, que es estéreo.

La segunda señal, es descrita con Eco, porque así es como se escucha al procesar la señal utilizando la misma como entrada por un sistema el sistema dado en el script:

```
function [ y ] = funcion_prueba( x )
%% Escribimos una ecuación en diferencias para verificar la resp impulsional
y = zeros(size(x));
y(1) = x(1);
for i=1:length(x)
    if(i < 8821)
        y(i) = x(i);
    elseif(i < 17641)
        y(i) = x(i) - (0.45)*x(i-8820);
    else
        y(i) = x(i) - (0.45)*x(i-8820) + (0.2)*x(i-17640);
    end
end
end
```

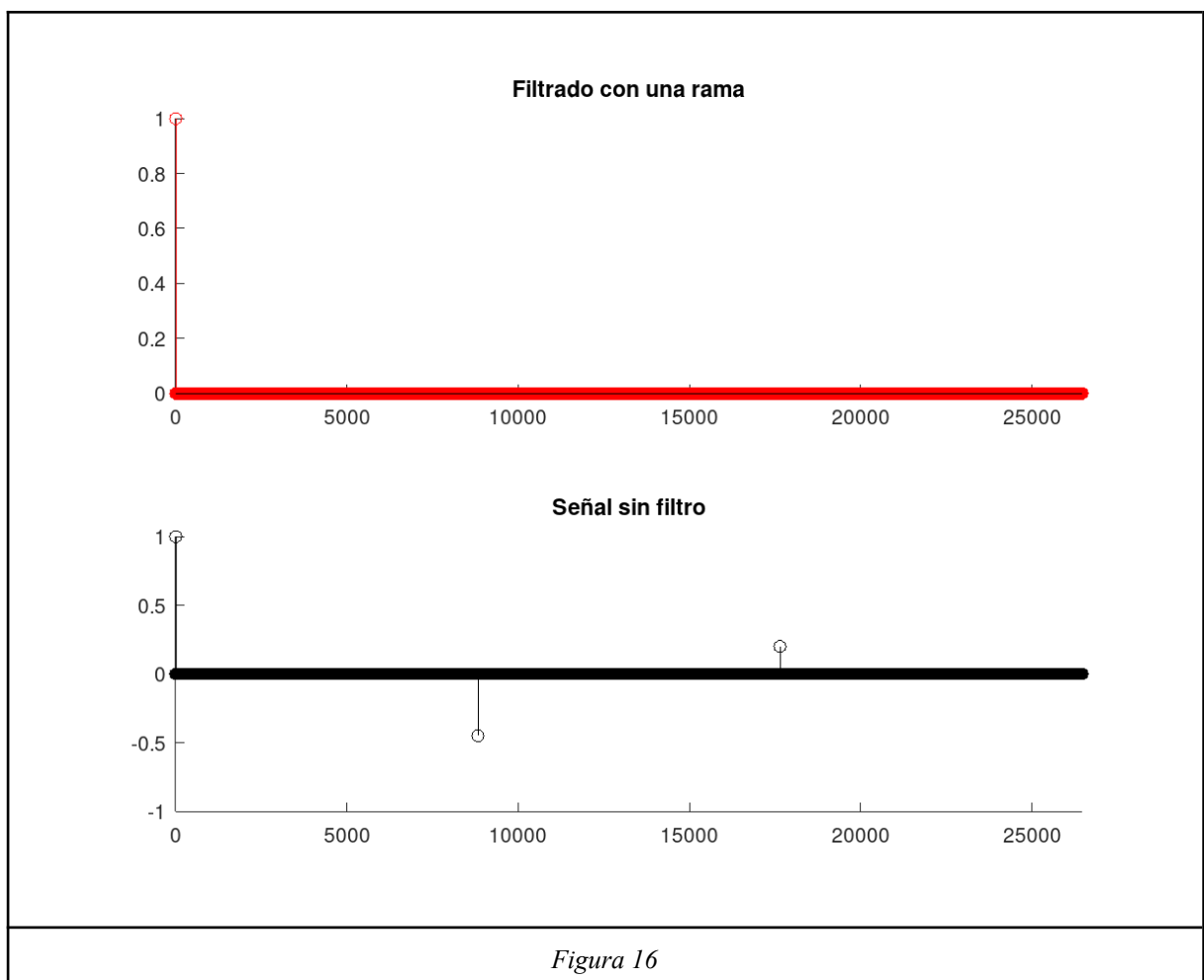
La señal procesada con eco presenta un solo canal ya que utiliza uno de los dos canales de la señal estéreo.

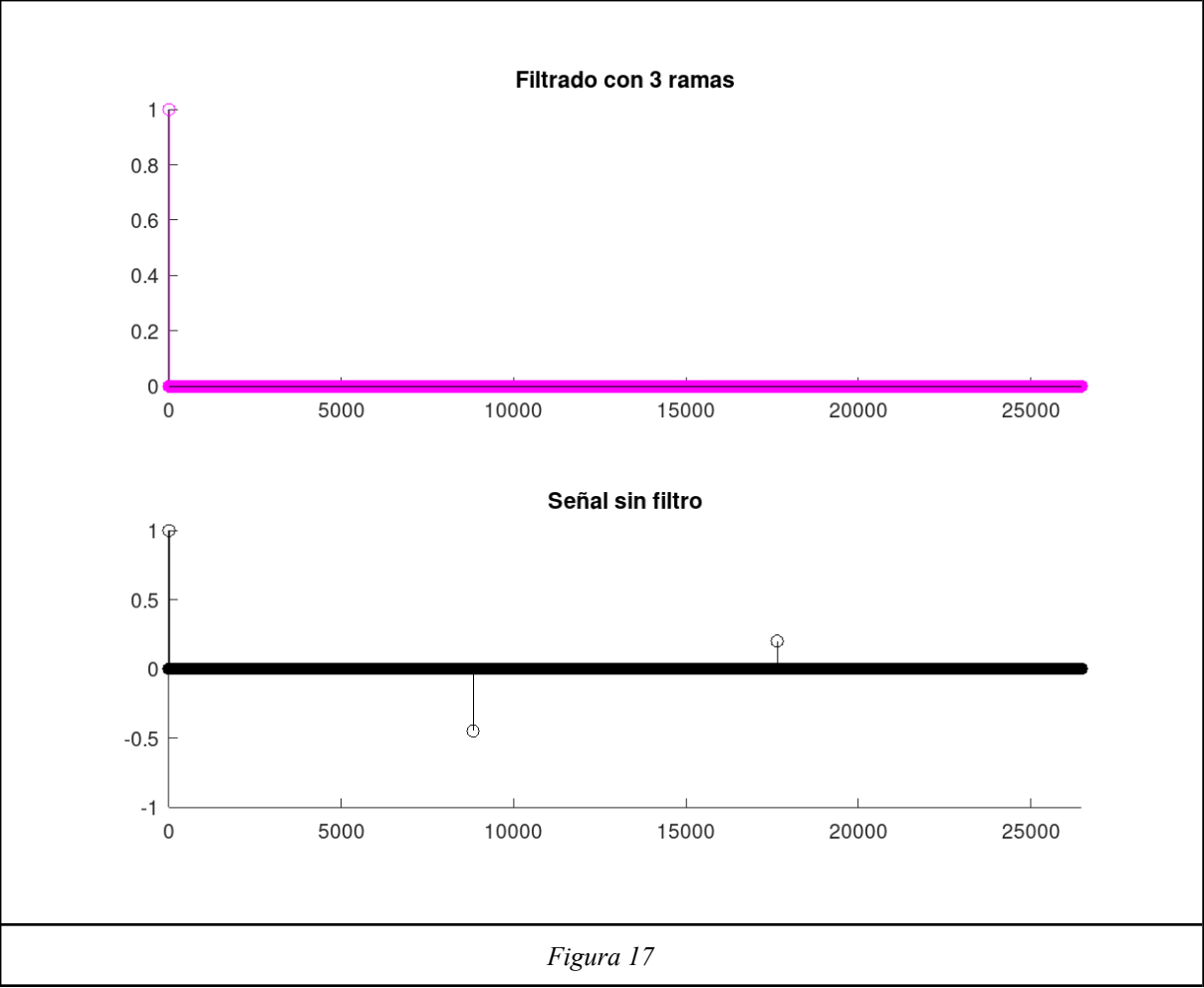
Inciso cuarto II:

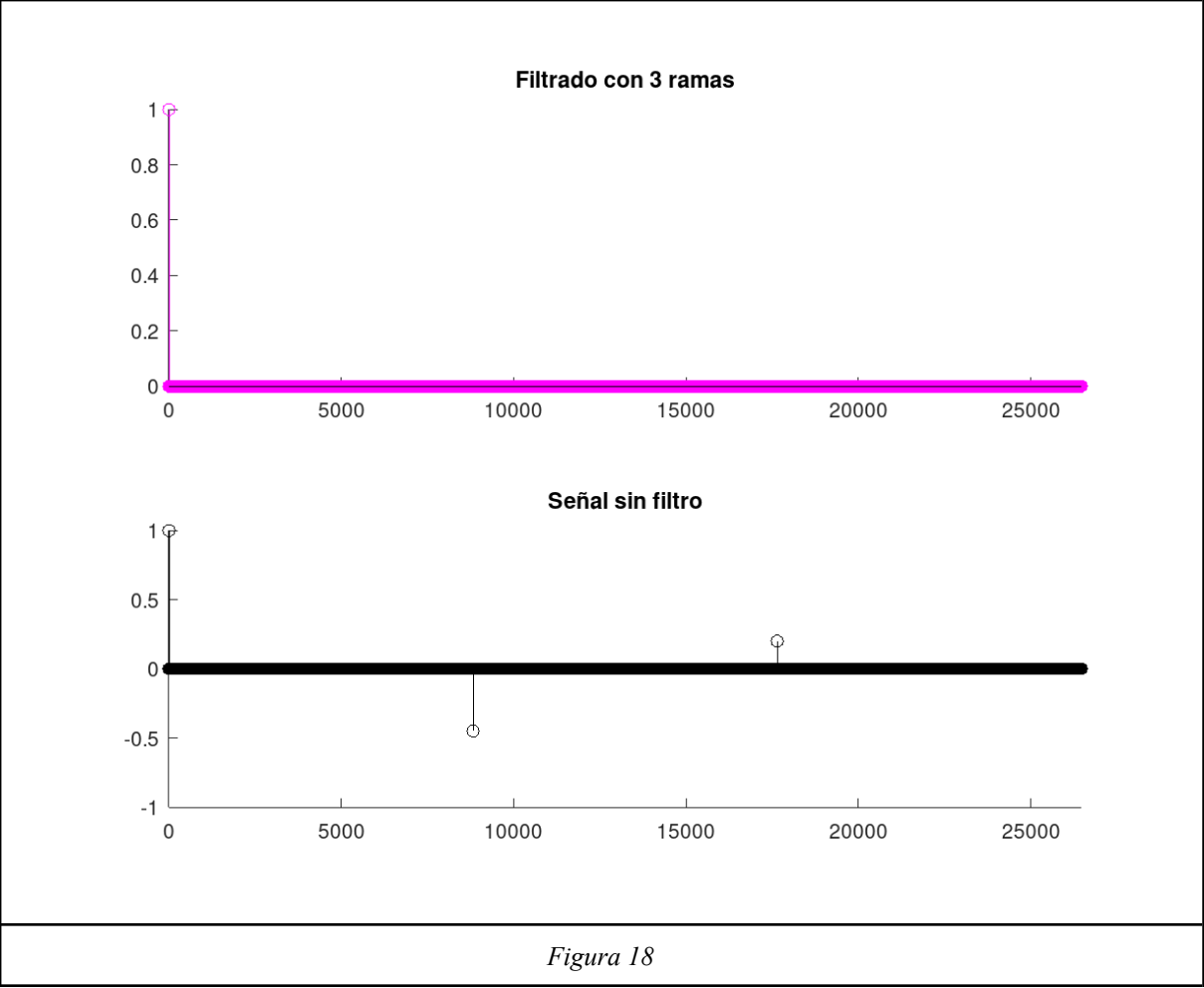
Si aplicamos lo que se explica anterior al inciso cuarto del ejercicio 2, obtenemos que un filtro de una sola rama, que se expresa con la ecuación:

$$y[n] = x[n] + 0.45 * x[n - 8820]$$

Podemos apreciar en las siguientes gráficas (Figuras de la 16 a la 18), que con utilizar un filtrado de una sola rama es suficiente para eliminar las deltas que corresponden a la respuesta impulsional de la señal con ruido.

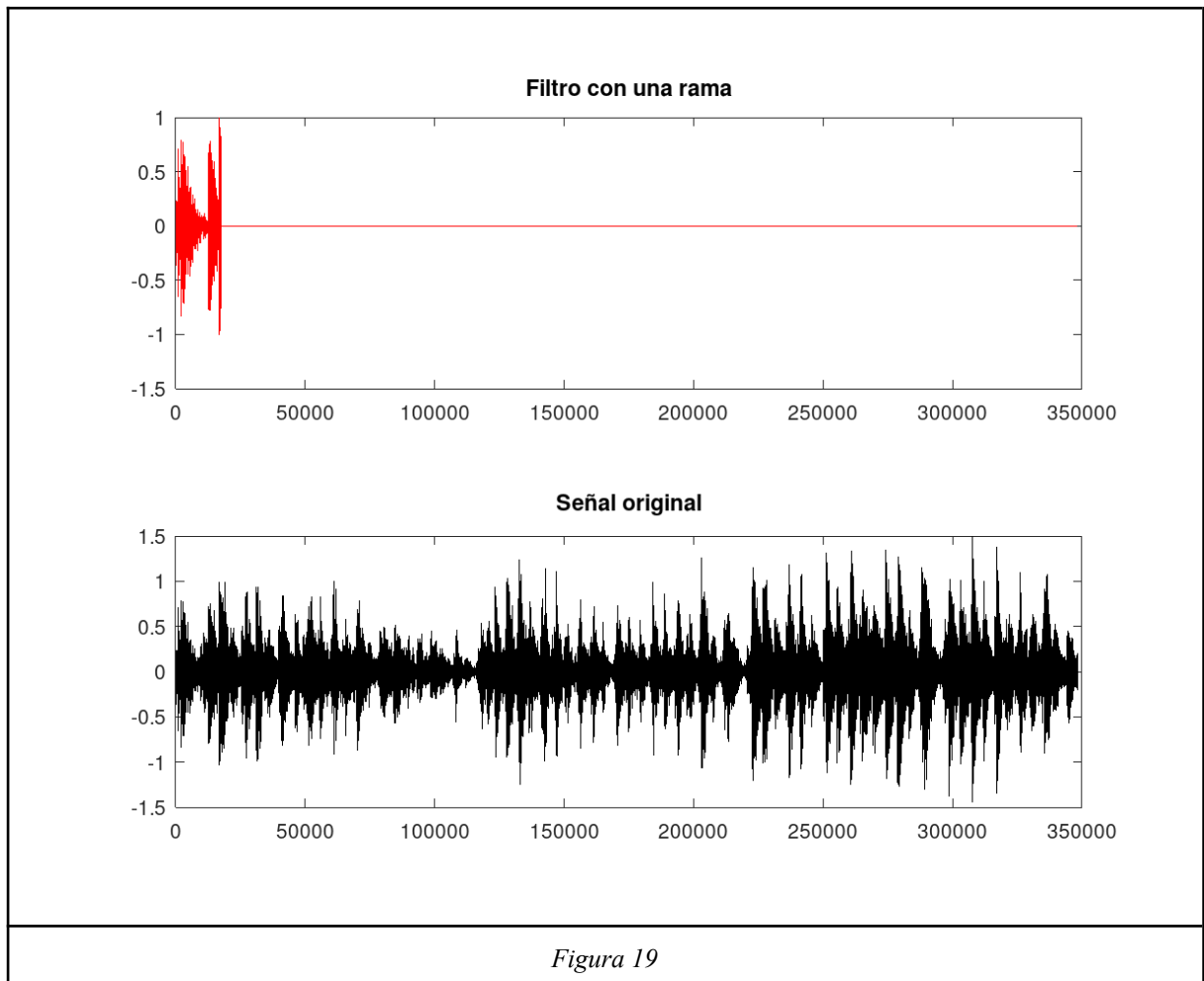






Inciso quinto II:

Por el funcionamiento mismo del filtro de una sola rama, podemos ver (figura 19), que procesar la señal por este sistema hace que la misma sea truncada al límite en el que trabaja el filtro. Esto mismo se observa en la figura 21 con el filtro de 3 ramas. El mejor resultado, se obtiene en el filtrado de dos ramas, ya que escuchando la señal en Octave, apreciamos que se elimina el eco, a la vez que se mantiene la totalidad de la señal.



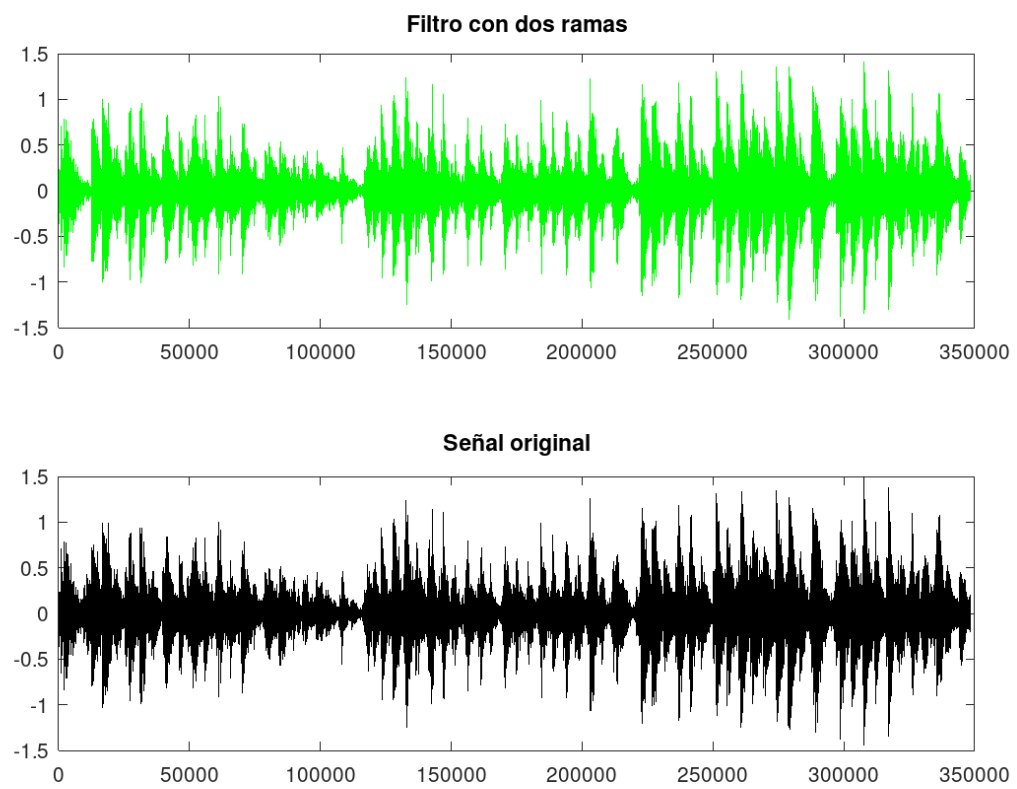


Figura 20

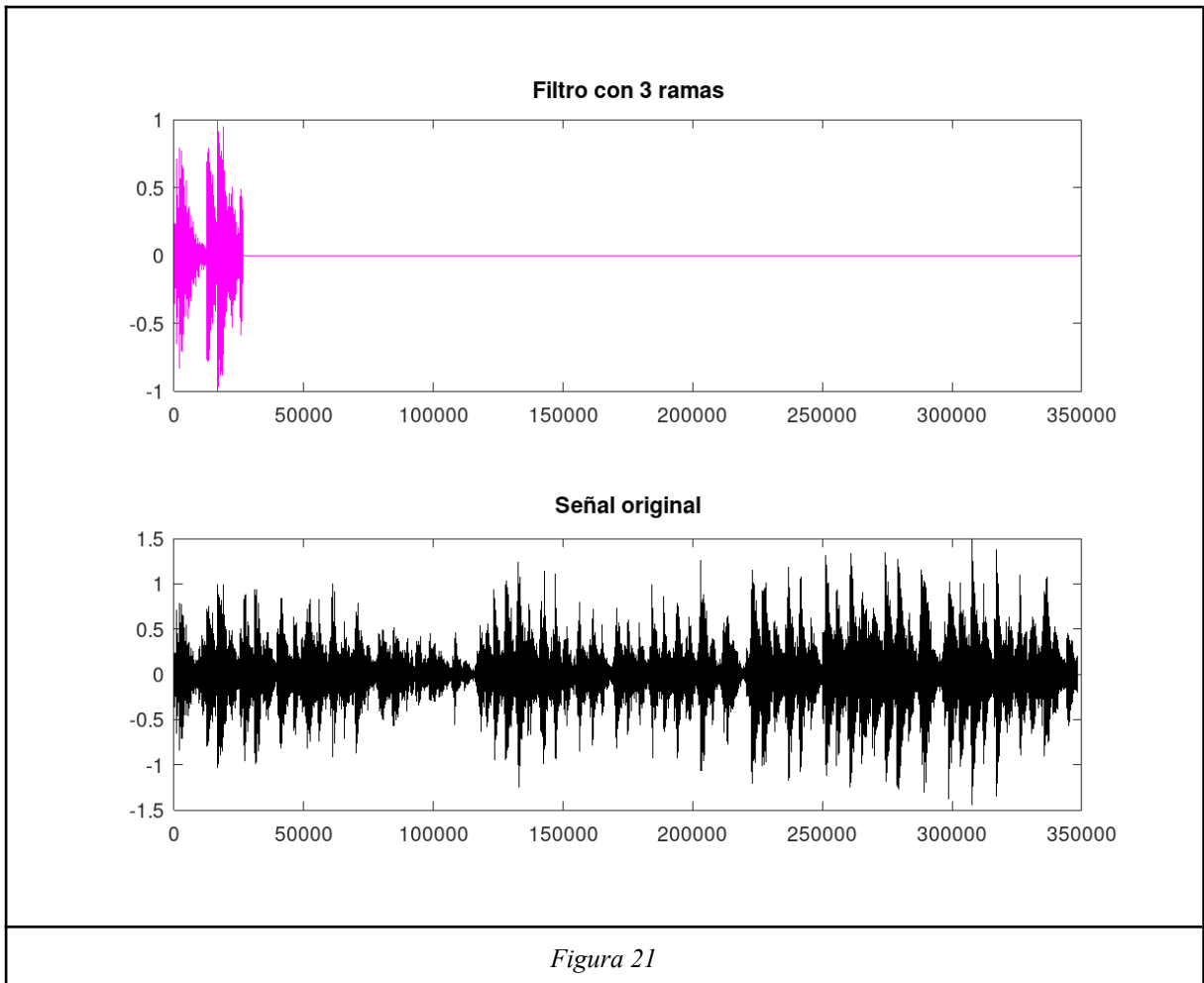


Figura 21

Como conclusión agregamos que si bien un filtro de una sola rama, es suficiente para eliminar el eco, éste mismo trunca la señal, es por esto que se opta por buscar otras soluciones, de las cuales, el filtro de dos ramas es la óptima.