

Práctica con Utilitario N° 1

Tomás Vidal

Análisis de Sistemas y Señales

Facultad de Ingeniería, UNLP, La Plata, Argentina.

21 de Mayo, 2023.

I. INTRODUCCIÓN

En el presente se desarrollan detalladamente los **métodos** y **técnicas** empleadas para la *caracterización y análisis* de **sistemas discretos**, en particular de tipo lineal e invariante al desplazamiento (o en el tiempo), haciendo uso de la herramienta **Octave**¹. Estos sistemas están descritos en ecuaciones en diferencias², y son implementados con algoritmos en **scripts**³. Además se analizan y solucionan los problemas presentes en un canal de transmisión cuya respuesta impulsional es dada como información, para poder enviar por el mismo una señal sin que se distorsione tanto.

II. IMPLEMENTACIÓN DE LA TFTD EN OCTAVE

II-A. Marco Teórico

La **transformada de Fourier de tiempo discreto (TFTD)** de una señal dada se define como ([2] y [3])

$$TFTD\{x[n]\} = X(e^{j2\pi s}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j2\pi sn}$$

Pero como no se pueden computar infinitos términos se debe hacer una aproximación de la siguiente manera

$$\hat{X}(e^{j2\pi s}) \approx \sum_{n=-K}^K x[n]e^{-j2\pi sn}$$

Aproximar desde $n = -K$ hasta $n = K$ es lo mismo que multiplicar toda la señal por un cajón y sumarla desde $-\infty$ hasta ∞ , es decir que el efecto de aproximar es desprestigiar información de la señal.

$$\sum_{n=-K}^K x[n]e^{-j2\pi sn} = \sum_{n=-\infty}^{\infty} x[n] \Pi\left[\frac{n}{2K}\right]e^{-j2\pi sn}$$

II-B. Algoritmo en Octave

```
function [s, tftd] = TFTD(  
    n, signal,  
    defaultStep=1e-3)  
    ds = defaultStep;  
    s = [-.5:ds:.5];  
    tftd = zeros(size(s));
```

¹Octave o GNU Octave es un programa y lenguaje de programación para realizar cálculos numéricos. Como su nombre indica, Octave es parte del proyecto GNU. Es considerado el equivalente libre de MATLAB

²ver ASyS2023.PU1-1.pdf.

³En informática, un script, refiere a un archivo que contiene una secuencia de comandos de un lenguaje de programación

```
for k = 1:length(s)  
    tftd(k) = sum(signal.*exp(-1j*2*pi*s  
        (k)*n));  
end  
end
```

II-C. Prueba del Algoritmo

Para poder hacer uso del algoritmo primero se realizó el cálculo analítico de la **TFTD** de las siguientes señales a partir de propiedades y pares transformandos conocidos

$$\Delta = \delta[n - n_o] \Rightarrow e^{-j2\pi sn_o}$$

$$\text{Triángulo} = \Lambda\left[\frac{n-b}{a}\right] \Rightarrow |a| \cdot e^{-j2\pi s \frac{a}{b}} \cdot \text{sinc}^2(as)$$

Y luego se empleó el siguiente código que calcula la **TFTD** de manera numérica con el algoritmo (ver **II-B**) y la **TFTD** analítica, para verificar que las aproximaciones sean lo “suficientemente” buenas

```
N = [1:100];
```

```
% Delta desplazada 20.
```

```
Senial1 = zeros(size(N));
```

```
Senial1(20) = 1;
```

```
[s1, tftd1] = TFTD(N0, Senial1);
```

```
% Triangulo desplazado 50 y de base 5.
```

```
Senial2 = tri((N0-length(N0)/2)/5);
```

```
[s2, tftd2] = TFTD(N0, Senial2);
```

```
% Transformada analitica senial1
```

```
tftdAnaliticSenial1 = @(s) exp(-j*2*pi  
    *20.*s);
```

```
% Transformada analitica senial2
```

```
tftdAnaliticSenial2 = @(s) 5.*exp(-2*pi  
    *10.*s).*(sinc(5.*s).^2);
```

De lo cual se obtuvieron los siguientes resultados (Fig. 1 y 2), que son acordes por la similitud a los cálculos analíticos (Fig. 3 y 4 respectivamente), por lo que se concluye que el algoritmo implementado es adecuado para esta aplicación. Cabe aclarar que si bien hay variaciones en las fases, éstas son muy susceptibles a cambios en las aproximaciones y, además no son de nuestro interés principal, ya que se quieren analizar los módulos de las **TFTD**.

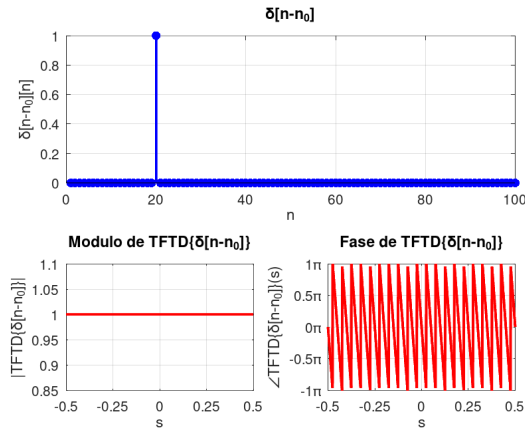


Fig. 1. Señal delta y el módulo y fase de su TFTD numérica.

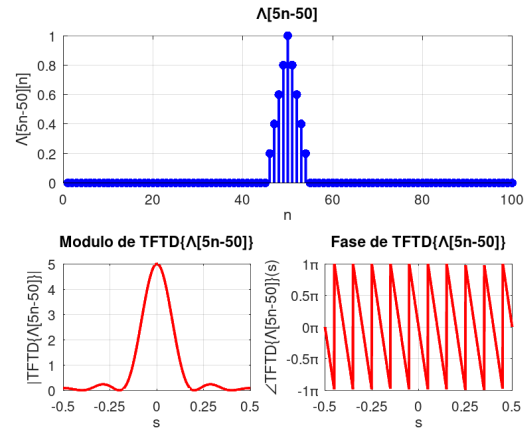


Fig. 4. Señal triángulo y el módulo y fase de su TFTD analítica.

III. TFTD DE LA SEÑAL PROVISTA

Se obtuvo la señal dada a partir de los comandos provistos⁴, y como se puede apreciar en el gráfico (fig. 5) la misma está compuesta de sinusoides superpuestas de componentes de alta y baja frecuencia.

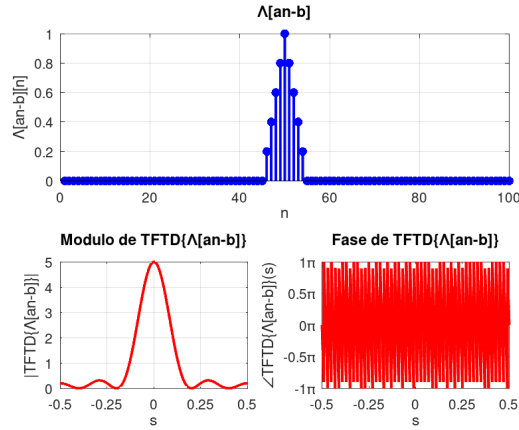


Fig. 2. Señal triángulo y el módulo y fase de su TFTD numérica.

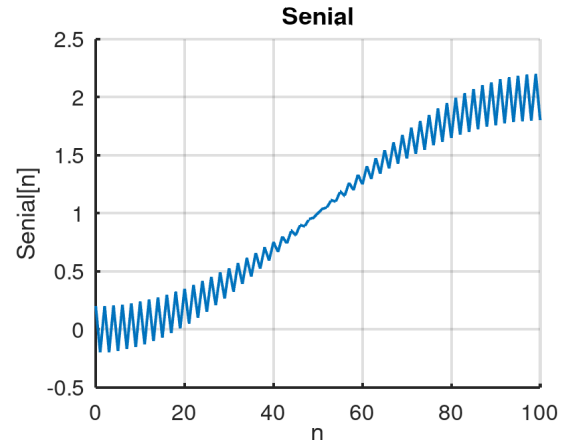


Fig. 5. Señal provista.

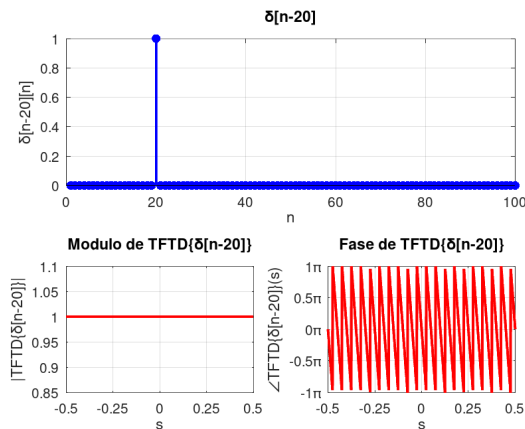


Fig. 3. Señal delta y el módulo y fase de su TFTD analítica.

Entonces haciendo uso del algoritmo desarrollado previamente (ver II-B) se aproximó la TFTD de la señal dada (ver 6), de lo cual se puede observar que efectivamente lo hipotetizado sobre la señal es cierto, pues la misma está compuesta por dos componentes principales: una de baja frecuencia, preponderante, y otra de alta de menor peso (ver Fig. 7).

⁴use el comando 'help senial' para más información.

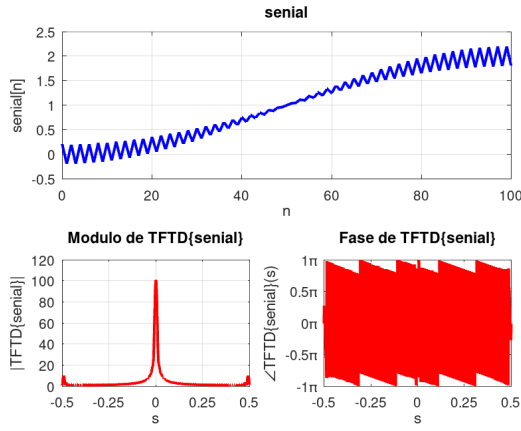


Fig. 6. Plot de la Señal y el módulo y fase de su TFTD.

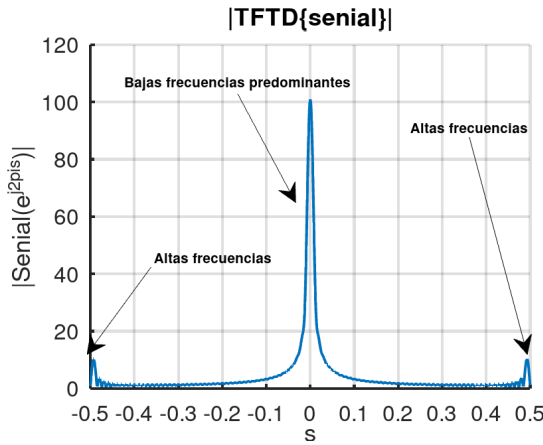


Fig. 7. Se indica las frecuencias predominantes en la TFTD de la señal.

IV. SISTEMAS DADOS

IV-A. Ecuaciones en diferencias

Las siguientes ecuaciones en diferencias dadas representan sistemas

$$y[n] = \frac{1}{2}x[n] + \frac{1}{2}x[n-1] \quad (1)$$

$$y[n] = \frac{1}{2}x[n] - \frac{1}{2}x[n-1] \quad (2)$$

$$y[n] = \frac{1}{4}x[n] + \frac{1}{4}x[n-1] + \frac{1}{2}y[n-1] \quad (3)$$

$$y[n] = \frac{1}{4}x[n] - \frac{1}{4}x[n-1] - \frac{1}{2}y[n-1] \quad (4)$$

Las respuestas impulsionales de los sistemas 1 y 2 se obtienen analíticamente cuando la entrada es $x[n] = \delta[n]$.

$$h_1[n] = \frac{1}{2}\delta[n] + \frac{1}{2}\delta[n-1] \quad (5)$$

$$h_2[n] = \frac{1}{2}\delta[n] - \frac{1}{2}\delta[n-1] \quad (6)$$

Y las respuestas en frecuencia de los mismos es:

$$H(e^{j2\pi s}) = \frac{TFTD\{y[n]\}}{TFTD\{x[n]\}} = \frac{Y(e^{j2\pi s})}{X(e^{j2\pi s})}$$

$$H_1(e^{j2\pi s}) = \frac{1}{2} + \frac{1}{2}e^{-j2\pi s} \quad (7)$$

$$H_2(e^{j2\pi s}) = \frac{1}{2} - \frac{1}{2}e^{-j2\pi s} \quad (8)$$

Para poder conseguir las respuestas impulsionales de los sistemas 3 y 4 cambia el procedimiento, porque hay una realimentación de la salida ($y[n]$), entonces se obtiene de la siguiente manera calculando la TFTD del sistema 3.

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

$$Y(e^{j2\pi s}) = \frac{1}{4}X(e^{j2\pi s}) + \frac{1}{4}e^{-j2\pi s}X(e^{j2\pi s}) + \frac{1}{2}e^{-j2\pi s}Y(e^{j2\pi s})$$

$$Y(e^{j2\pi s})[1 - \frac{1}{2}e^{-j2\pi s}] = X(e^{j2\pi s})[\frac{1}{4} + \frac{1}{4}e^{-j2\pi s}]$$

$$H_3(e^{j2\pi s}) = \frac{Y(e^{j2\pi s})}{X(e^{j2\pi s})} = \frac{[\frac{1}{4} + \frac{1}{4}e^{-j2\pi s}]}{[1 - \frac{1}{2}e^{-j2\pi s}]} \quad (9)$$

Entonces a partir de la siguiente transformada

$$u[n]a^n \supset \frac{1}{1 - ae^{-j2\pi s}}, |a| < 1$$

y de la propiedad

$$x[n - n_o] \supset e^{-j2\pi sn_o} X(e^{-j2\pi s})$$

Se antitransforma la respuesta en frecuencia del sistema 3 (ec. 9) para obtener la respuesta impulsional del mismo (ec. 10)

$$H_3(e^{j2\pi s}) = \frac{1}{4} \left[\frac{1}{1 - \frac{1}{2}e^{-j2\pi s}} + \frac{e^{-j2\pi s}}{1 - \frac{1}{2}e^{-j2\pi s}} \right]$$

$$H_3(e^{j2\pi s}) \supset h_3[n] = \frac{1}{4} \left[u[n] \left(\frac{1}{2} \right)^n + u[n-1] \left(\frac{1}{2} \right)^{n-1} \right] \quad (10)$$

Ahora se procede de la misma manera con el sistema 4, y así se obtienen la respuesta en frecuencia (ec. 11) y la respuesta impulsional (ec. 12) del mismo.

$$H_4(e^{j2\pi s}) = \frac{1}{4} \left[\frac{1}{1 + \frac{1}{2}e^{-j2\pi s}} - \frac{e^{-j2\pi s}}{1 + \frac{1}{2}e^{-j2\pi s}} \right] \quad (11)$$

$$H_4(e^{j2\pi s}) \supset h_4[n]$$

$$h_4[n] = \frac{1}{4} \left[u[n] \left(\frac{-1}{2} \right)^n - u[n-1] \left(\frac{-1}{2} \right)^{n-1} \right] \quad (12)$$

IV-B. Conclusiones sobre los sistemas

Observando las respuestas impulsionales podemos concluir que corresponden a sistemas del tipo FIR⁵.

Además, analizando las respuestas en frecuencias, el **sistema 1** (Fig. 8) se comporta como un filtro pasa bajos, el **sistema 2** (Fig. 9) como uno pasa altos, el **3** (Fig. 10) como un pasa bajos con un ancho de banda más selectivo que el sistema 1, y por último el **sistema 4** (Fig. 11) como un filtro pasa altos con una banda más selectiva que el sistema 3.

⁵Finite Impulse Response (Respuesta Impulsional Finita).

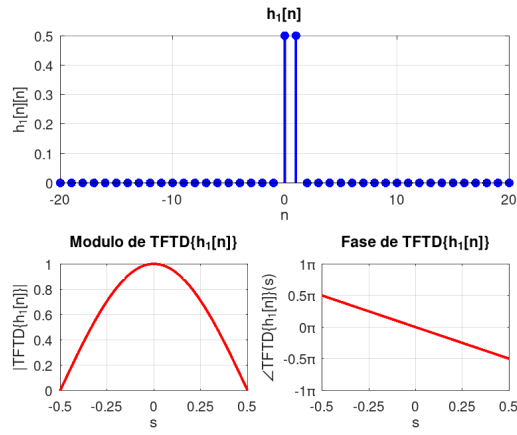


Fig. 8. Respuesta impulsional y en frecuencia del sistema 1.

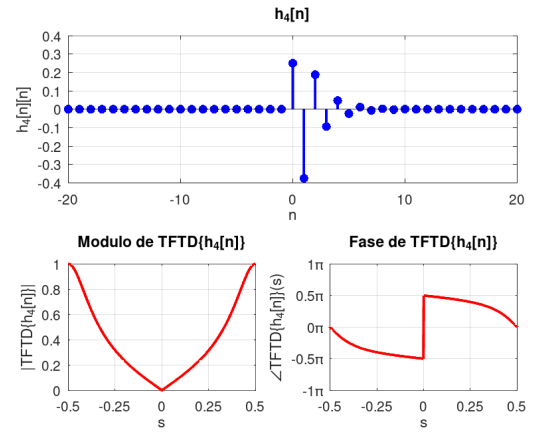


Fig. 11. Respuesta impulsional y en frecuencia del sistema 4.

V. CÁLCULO NUMÉRICO DE LAS RESPUESTAS IMPULSIONALES Y EN FRECUENCIAS DE LOS SISTEMAS

A partir de las ecuaciones de los sistemas (1, 2, 3 y 4) se obtuvieron, en forma numérica, las respuestas impulsionales, tomando como entrada a los sistemas una delta ($\delta[n]$); y también la respuesta en frecuencia a partir de usar el algoritmo de la TFTD (ver II-B) en la respuesta impulsional obtenida previamente.

V-A. Implementación de los sistemas en Octave

Para poder implementar cada sistema se codificaron funciones que describen a cada uno, luego se definió una función anónima delta que representa una delta de kronocker ($\delta[n]$) y, por último, un vector N que define el instante de la señal

V-B. Código

```
y1 = @(x, n) (.5)*x(n) + (.5)*x(n-1);
y2 = @(x, n) (.5)*x(n) - (.5)*x(n-1);
function [y] = Y3(x, n)
    y(1) = 0; % Condicion inicial.
    for (k = 2:length(n))
        y(k) = (.25).*x(n(k)) + (.25).*x(n(k)-1)) + (.5).*y(k-1);
    end
end
y3 = @(x, n) Y3(x, n);
function [y] = Y4(x, n)
    y(1) = 0; % Condicion inicial.
    for (k = 2:length(n))
        y(k) = (.25).*x(n(k)) - (.25).*x(n(k)-1)) - (.5).*y(k-1);
    end
end
y4 = @(x, n) Y4(x, n);

% Defino un impulso.
delta = @(n) (n==0);
```

N = [-20:20];

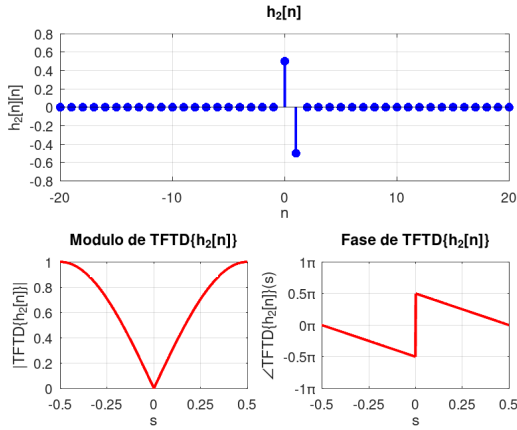


Fig. 9. Respuesta impulsional y en frecuencia del sistema 2.

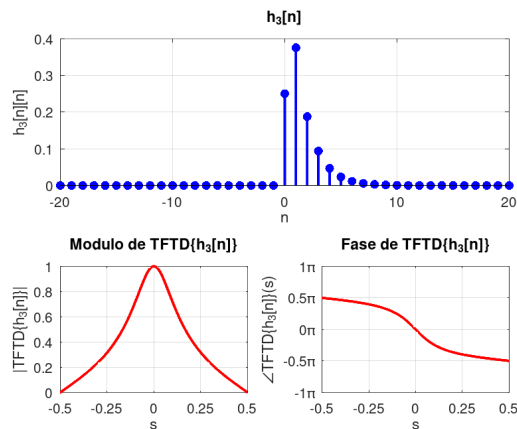


Fig. 10. Respuesta impulsional y en frecuencia del sistema 3.

V-C. Conclusiones sobre los resultados numéricos

Observando los resultados que se pueden apreciar en los siguiente gráficos (Fig. 12, 13, 14 y 15) se concluye que las implementaciones numéricas son tan buenas como las analíticas y, además, indican que los cálculos analíticos son correctos.

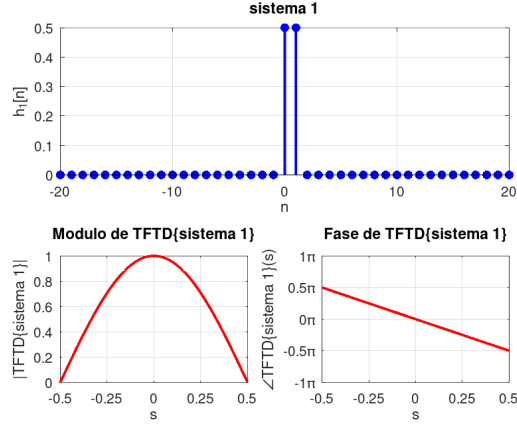


Fig. 12. Respuesta impulsional y en frecuencia del sistema 1 (numérica).

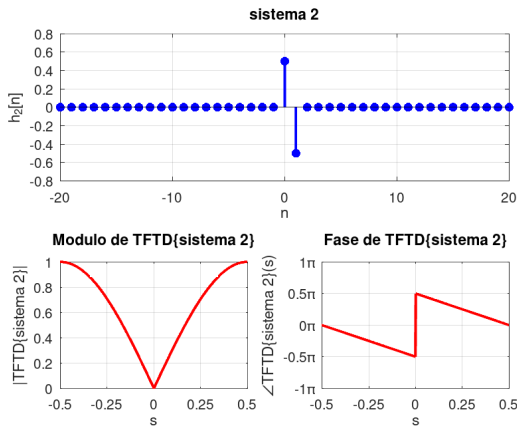


Fig. 13. Respuesta impulsional y en frecuencia del sistema 2 (numérica).

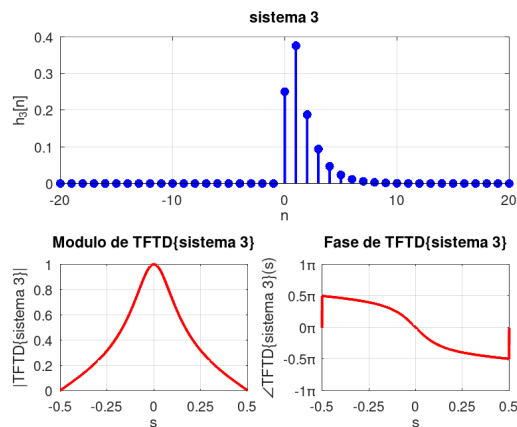


Fig. 14. Respuesta impulsional y en frecuencia del sistema 3 (numérica).

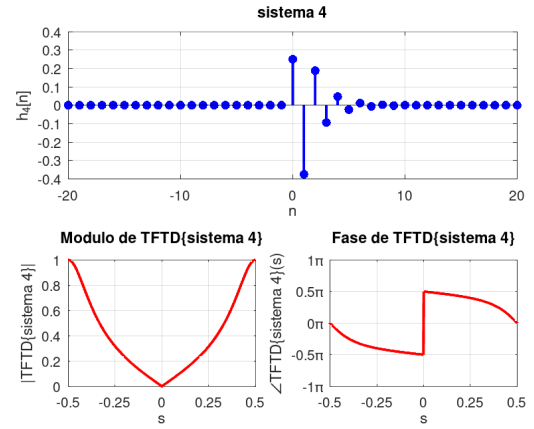


Fig. 15. Respuesta impulsional y en frecuencia del sistema 4 (numérica).

VI. PROCESAMIENTO DE “senial” CON LOS SISTEMAS

Se emplean los sistemas 1, 2, 3 y 4 como filtros para procesar la señal dada “senial”; para ello se usa como entrada a los mismos “senial”; luego de esto se calculan las TFTD de las señales filtradas. Los resultados obtenidos se pueden observar en los gráficos 16, 17, 18 y 19, donde “senial_i” refiere a la señal filtrada con el sistema “i”.

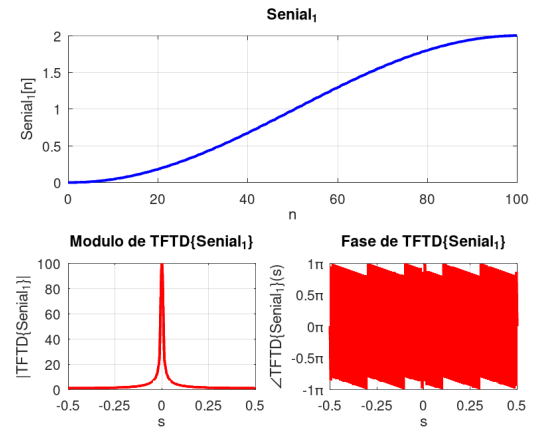


Fig. 16. Senial filtrada con el sistema 1.

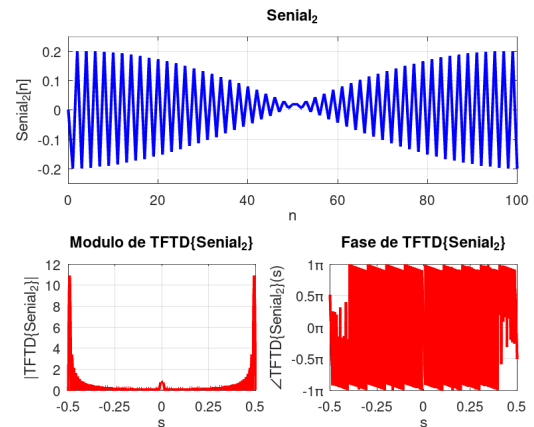


Fig. 17. Senial filtrada con el sistema 2.

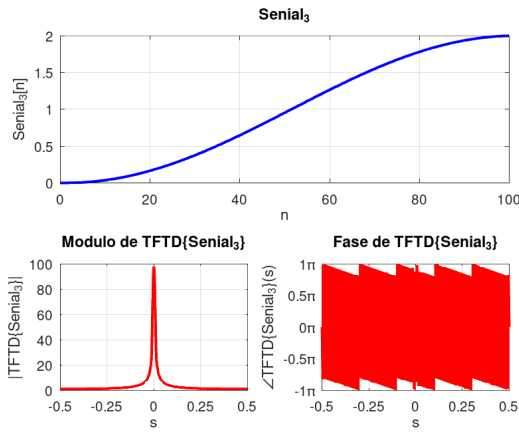


Fig. 18. Señal filtrada con el sistema 3.

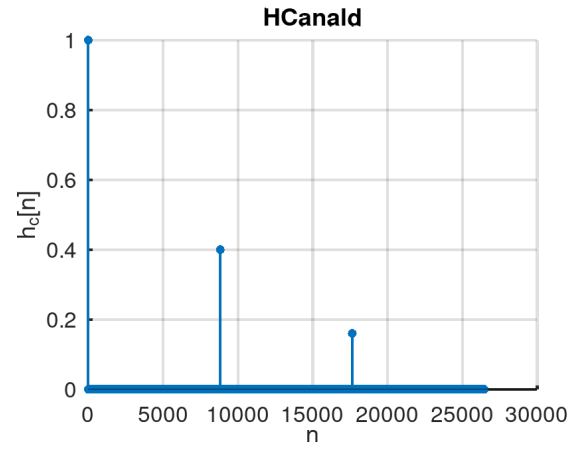


Fig. 20. Señal "hcanald".

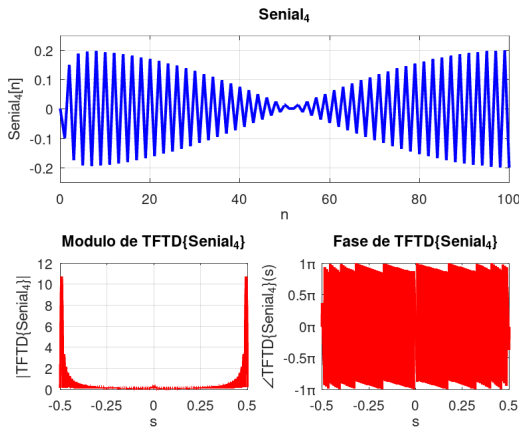


Fig. 19. Señal filtrada con el sistema 4.

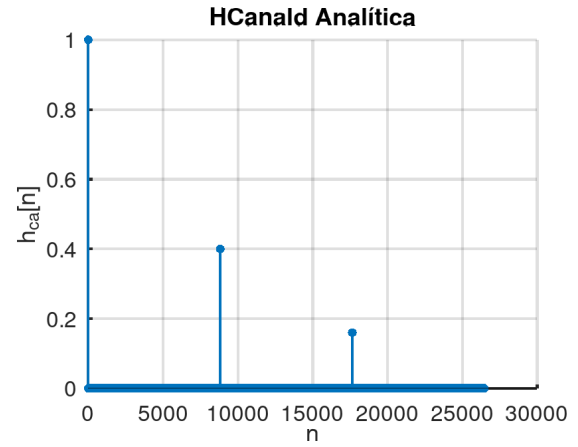


Fig. 21. Señal 13.

VI-A. Conclusiones de las señales filtradas

A partir de los gráficos 16, 17, 18 y 19 se puede concluir que las hipótesis planteadas en IV-B son correctas, pues en los mismos se puede observar que las componentes de alta frecuencia son eliminadas con los sistemas 1 y 3 (Fig. 16 y 18 respectivamente), que se comportaban como **filtros pasa bajos**; y de manera similar las frecuencias bajas son eliminadas con los sistemas 2 y 4 (Fig. 17 y 19 respectivamente), que actuaban como **filtros pasa altos**.

VII. PROCESAMIENTO DE LA SEÑAL PROVISTA "hcanald"

La señal provista "hcanald" se compone de tres deltas de Kronocker de diferentes amplitudes, una ubicada en 0 y de amplitud 1 ($\delta[n]$), otra en $n_0 = 8820$ y de amplitud $\alpha = \frac{2}{5}$ ($\alpha\delta[n - n_0]$), y la tercera en $n_1 = 2n_0 = 17640$ y de amplitud $\beta = \alpha^2 = \frac{4}{25}$ ($\beta\delta[n - n_1]$). Por lo que la respuesta impulsional puede ser descripta analíticamente como:

$$h[n] = \delta[n] + \alpha\delta[n - n_0] + \beta\delta[n - n_1] \quad (13)$$

Y como se puede observar en los gráficos (Fig. 20 y Fig. 21), la ecuación 13 es correcta.

VII-A. Salida del canal digital

A partir de la ecuación 13 implementada como un sistema en Octave (ver VII-B) y de la señal de audio provista, se obtuvo la salida de lo que simula ser un canal digital (Fig. 22).

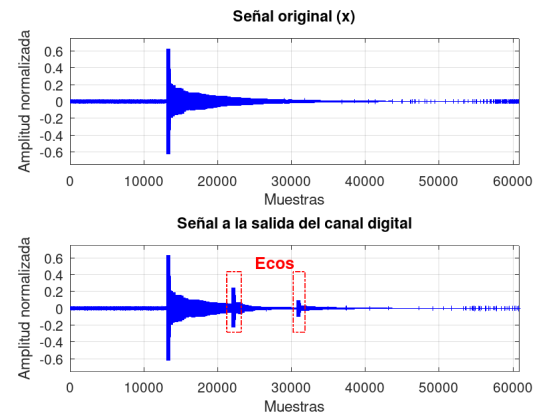


Fig. 22. Audio provisto como entrada y salida al canal digital.

VII-B. Implementación del canal digital en Octave

```
function y = ecDiff(x)
% El algoritmo solo acepta que x sea
vector.
if (
    strcmp(typeinfo(x), 'bool matrix')
        == 0 &&
    strcmp(typeinfo(x), 'matrix') == 0
)
    disp('ERROR: x debe ser vector.\n',
        'red');
    exit(1);
end

%{
    La ecuacion a resolver es:
    y[n] = x[n] + a.x[n-n0] + b.x[n-n1]
%}

% Constantes.
n0 = 8820;
n1 = 2*n0;
a = (2/5);
b = a^2;

% Algoritmo.
for (k = 1:length(x))
    term1=0;term2=0;
    if (k-n0 > 0)
        term1 = a*x(k-n0);
    end
    if (k-n1 > 0)
        term2 = b*x(k-n1);
    end
    y(k) = x(k) + term1 + term2;
end

end
```

VII-C. Conclusiones sobre el canal digital

De los resultados previos (Fig. 22) el lector puede inferir correctamente (de manera visual y auditiva), que el canal digital distorsiona la señal de entrada creando dos réplicas de *amplitudes significativas*; y estas son debidas a las deltas descriptas anteriormente (ver ec. 13). Para poder solucionar estos efectos indeseados se procede aplicando un filtro del tipo FIR, que busca asemejar la respuesta impulsional del canal a una única delta de Kronecker ($h[n] = \delta[n]$), pues de esta manera las réplicas no aparecen, ya que las deltas $\alpha\delta[n - n_0]$ y $\beta\delta[n - n_1]$ son la causa de las mismas. El problema que ocurre al aplicar el filtro, es que siempre permanece una delta indeseada, cuya amplitud es proporcional a la cantidad de ramas del filtro. Esto se puede ver claramente en las siguientes ecuaciones, donde se calculan las repuestas impulsionales de los sistemas equivalentes en cascada $h_{fi}[n]$, donde i corresponde a la cantidad de ramas

del filtro.

Si suponemos la respuesta impulsional del canal como $h_c[n]$

$$h_c[n] = \delta[n] + \alpha\delta[n - n_0]$$

Entonces el sistema equivalente tiene una respuesta impulsional $h_{eqi}[n]$:

$$h_{eqi}[n] = \text{conv}\{h_c[n] * h_{fi}[n]\}[n]$$

$$h_{f1}[n] = \delta[n] - \alpha\delta[n - n_0]$$

$$h_{eq1}[n] = \delta[n] - (\alpha)^2\delta[n - n_0]$$

Para una rama se tiene una delta indeseada de amplitud α^2

$$h_{f2}[n] = \delta[n] - \alpha\delta[n - n_0] + (\alpha)^2\delta[n - 2n_0]$$

$$h_{eq2}[n] = \delta[n] + (\alpha)^3\delta[n - 3n_0]$$

Para dos ramas se tiene una delta indeseada de amplitud α^3

$$h_{f3}[n] = \delta[n] - \alpha\delta[n - n_0] + (\alpha)^2\delta[n - 2n_0] - (\alpha)^3\delta[n - 3n_0]$$

$$h_{eq3}[n] = \delta[n] - (\alpha)^4\delta[n - 4n_0]$$

Para tres ramas se tiene una delta indeseada de amplitud α^4 . Y así sucesivamente, concluyendo que siempre se tiene un eco.

Por lo que para solucionar los ecos del canal digital, se comienza implementando un filtro de dos ramas “f1”, $f_1[n] = x[n] - (\alpha)x[n - n_0] + (\alpha)^2x[n - 2n_0]$, con $n_0 = 8820$ y $\alpha = \frac{2}{5}$, el resultado se puede ver en la figura 24.

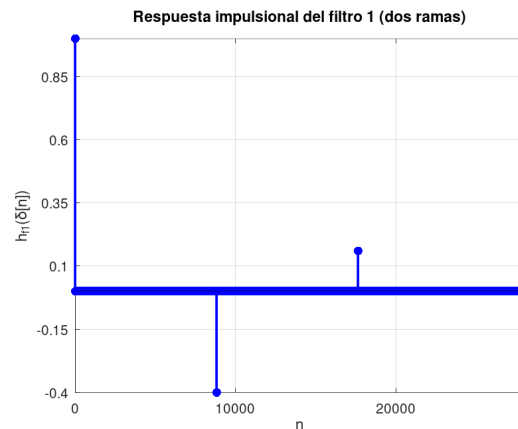


Fig. 23. Respuesta impulsional del filtro 1.

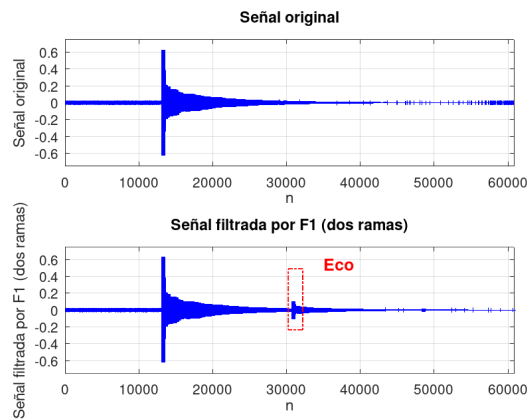


Fig. 24. Salida del filtro 1.

Por lo explicado previamente la salida del filtro “ f_1 ” todavía posee un poco de distorsión, lo cual se puede mejorar implementando un filtro “ f_2 ” que posee tres ramas de retardo: $f_2[n] = x[n] - (\alpha)x[n - n_0] + (\alpha)^2x[n - 2n_0] + (\alpha)^3x[n - 3n_0]$.

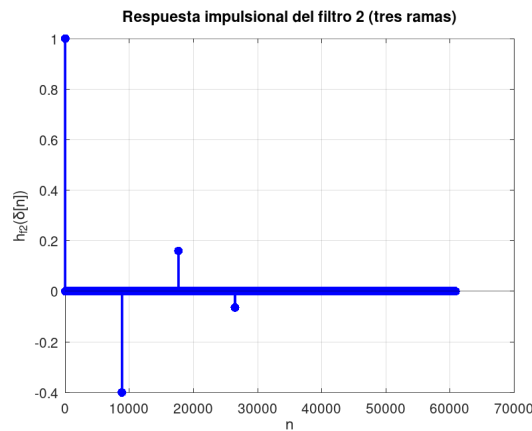


Fig. 25. Respuesta impulsional del filtro 2.

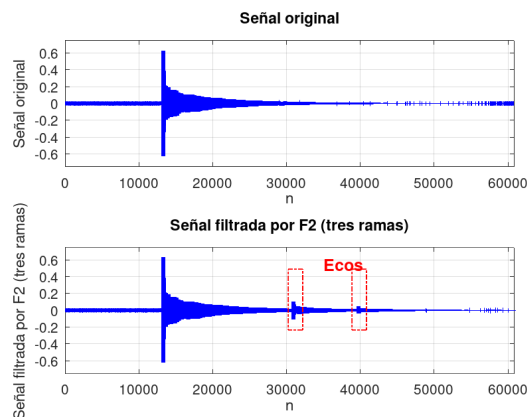


Fig. 26. Salida del filtro 2.

El lector puede corroborar, auditivamente u observando los gráficos provistos que, la salida del filtro 2, es mucho mejor que la salida del canal digital, pero *no idéntica a la señal original* antes de introducirla en el canal digital; por lo que se concluye que *se pudieron mejorar*, a un nivel aceptable, las distorsiones que produce el canal; pero que también, se podría seguir mejorando aún más, *implementando filtros con más ramas*.

VIII. ACLARACIONES SOBRE DATOS AJENOS AL PRESENTE

Todos los archivos y datos a los que refieren en el presente se pueden encontrar en las carpetas adjuntas al mismo; algunas de éstas carpetas se generarán después de ejecutar “*main.m*” con **Octave** (*no se recomienda usar Matlab⁶, pues no ha sido diseñado para el mismo*).

REFERENCIAS

- [1] A.V. Oppenheim, A.S. Willsky, Señales y Sistemas.
- [2] ANÁLISIS DE SISTEMAS Y SEÑALES - Año 2023, Práctica 5 Transformada de Fourier de Tiempo Discreto (TFTD), Serie Discreta Fourier (SDF).
- [3] ANÁLISIS DE SISTEMAS Y SEÑALES - Año 2023, Transparencias de teoría 5: Transformada de Fourier de Tiempo Discreto (TFTD).

⁶MATLAB (abreviatura de MATrix LABoratory, “laboratorio de matrices”) es un sistema de cómputo numérico que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M).