

ANÁLISIS DE CANALES CON INTERFERENCIA INTERSÍMBOLO

Trabajo Preparatorio N°6
Laboratorio de Comunicación Digital-GR9

1st Melanny Dávila

Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Electrónica
Quito, Ecuador
melanny.davila@epn.edu.ec

2nd Ronaldo Almachi

Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Electrónica
Quito, Ecuador
ronaldo.almachi@epn.edu.ec

Abstract—En el presente documento se revisará conceptos referentes a la canales con interferencia intersímbolo, en especial se analizará como afecta el factor de roll-off a la respuesta en frecuencia de un filtro. Todo esto será analizado en base a scripts en Matlab utilizando el toolbox de comunicaciones. A continuación se presenta el fundamento teórico para hacer uso de estas herramientas y scripts que hacen uso de las mismas.

Index Terms—Filtro coseno levantado, factor de roll-off, ISI, SNR.

I. INTRODUCCIÓN

Un canal de comunicaciones es el medio físico por el cual se realiza la comunicación entre una fuente de información con un destino determinado, de manera más técnica provee una conexión eléctrica entre la fuente y el destino, dicha conexión puede ser en base a un medio guiado (cable UTP,STP, coaxial, fibra óptica) o un medio no guiado(aire,el vacío, agua de mar); en ambos casos se producen impedimentos o limitaciones que ocasionan un error o bajo rendimiento.

Uno de los parámetros más importantes que causa problemas es el ancho de banda limitado, pues puede producir distorsión en amplitud y fase de la señal a transmitir. Uno de los principales problemas que se encuentra en un canal de transmisión es la interferencia intersímbolo (ISI), que afecta al rendimiento de la transmisión. Es un fenómeno en el cual algunas componentes de un pulso determinado interfiere con el o los pulsos adyacentes, puede ser causado por: inexactitudes de la señal de reloj o ancho de banda insuficiente. La interferencia intersímbolo se puede observar con la ayuda del diagrama del ojo y se la puede notar cuando el ojo tiende a cerrarse, lo que provoca que sea más difuso [2].

II. OBJETIVOS

- Familiarizar al estudiante con la interferencia intersímbolo ISI que sufren las señales digitales en los sistemas de comunicación.
- Utilizar MATLAB para analizar la reducción del ISI empleando un filtro de coseno levantado [1].

III. PREGUNTAS

A. *Revisar los conceptos de transmisión en banda base y banda ancha junto con sus aplicaciones*

• Transmisión en banda base:

Implica que se transmite información en forma digital sin la necesidad de realizar una modulación, usando un medio de transmisión físico guiado, como no se usa ninguna técnica de modulación el ancho de banda se desperdicia, pero varias señales pueden ser transmitidas a la vez usando multiplexación de división por tiempo, este tipo de métodos es ideal en situaciones en que no se tenga que cubrir una área muy extensa. A pesar que no se usa ningún tipo de modulación se ha desarrollado técnicas para que dichas señales se adecuen de forma mas óptima al medio guiado sin modificar la naturaleza de la señal digital, estas técnicas son conocidas como códigos de línea [2].

La códigos de línea son parte fundamental para una buena transmisión en banda base, pues se encargan de: dar una apropiada forma al espectro de la señal, una adecuada información de temporización para no perder sincronismo y una componente DC nula con el fin de facilitar el acoplamiento AC en el medio de transmisión [3].

Aplicaciones:

- Son usadas en redes de área local (LAN) debido a su bajo costo de implementación.
- Son usados para el diseño de ecualizadores, incluyendo forzamiento cero y ecualizadores MMSE.
- Sirven para el diseño de receptores óptimos, incluidos filtros y correladores emparejados.
- Detector de probabilidad a posteriori máximo (MAP), estimación de máxima verosimilitud (MLE) de símbolos y receptor RAKE [2].

• Transmisión en banda ancha:

Es una tecnología de señalización que sirve para enviar señales simultáneamente en diferentes rangos de frecuencias como ondas electromagnética, un sistema de banda ancha es capaz de llevar múltiples señales de datos al

mismo tiempo, estas viajan en una sola dirección, por lo que se puede transmitir o recibir, pero no ambas al mismo tiempo, para usar esta tecnología a largas distancias es necesario usar regeneradores de señales para contrarrestar la atenuación a la que se ven afectadas [4]. En una transmisión de banda ancha usamos la multiplexación por división de frecuencias, en este caso cada canal tiene una frecuencia de portadora que modula cada señal para llevarla a través de del canal correspondiente, adicional tiene bandas de guarda que nos sirven para evitar la interferencia las señales moduladas [4].

Aplicaciones:

- La transmisión de banda ancha se utiliza normalmente para entornos en los que el vídeo, el audio y los datos deben transmitirse simultáneamente.
- Los sistemas de televisión por cable se basan en tecnologías de transmisión de banda ancha.
- Otros ejemplos de servicios de banda ancha incluyen los servicios de portadora de T, el modo de transferencia asíncrona (ATM) y las variantes de la línea de suscriptor digital (DSL) [4].

B. Consultar y describir los dos criterios de Nyquist empleados para eliminar la interferencia intersímbolo ISI. Revisar los conceptos relacionados al filtro coseno levantado y como afecta el factor de roll-off en el ISI.

• Primer criterio de Nyquist

Este criterio dice que la función de transferencia del sistema de la señal a transmitir debe ser empleada en el dominio del tiempo. La misma que debe ser una señal Sampling, que está definida como lo muestra la ecuación 1.

$$S_{am}(t) = \frac{\sin(2\pi ft)}{2\pi ft} \quad (1)$$

Con la finalidad de presentar una transmisión sin distorsión, pues en el dominio de la frecuencia dicha función se refleja teóricamente como un pulso rectangular como se puede ver en la figura 1 [5].

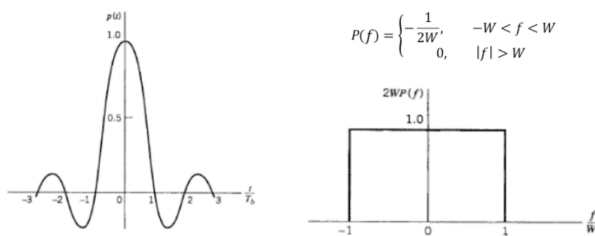


Fig. 1. Función samplin en dominio del tiempo y la frecuencia [6]

Para usar este primer método se debe tener consideraciones tales como el ancho de la señal debe ser la mitad de la velocidad de transmisión que matemáticamente se puede representar como $V_{tx} = 2AB$, además que la sincronización debe ser muy precisa, y si no lo es existirá

una interferencia debido a los lóbulos de la señal. Esto produciría un efecto adverso ya que los dichos lóbulos se sobrelaparan unos con otros, para evitar esto se debe muestrear a la señal en la mitad del tiempo de bit. Como se muestra en la figura 2 cuando una componente tiene el valor de 1, todas las demás componentes tienen el valor de 0 [6].

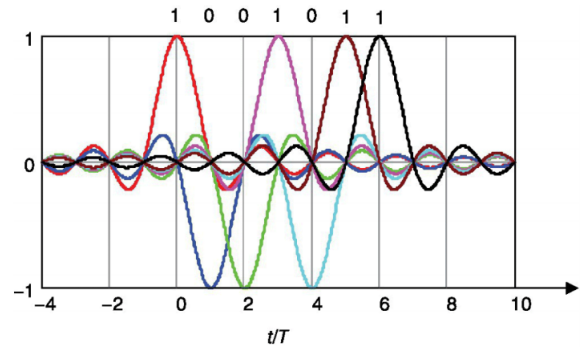


Fig. 2. Sincronización de la señal usando el primer criterio de Nyquist [6]

• Segundo criterio de Nyquist

Este criterio parte del primer criterio de Nyquist en específico del parámetro de sincronización de la señal, en base a la dificultad que se tiene para tener una buena sincronización durante toda la transmisión, lo que produce una interferencia debido a los lóbulos de la señal. Este criterio dice que se debe tratar de que dichos lóbulos sean casi despreciables, de tal manera que si no se puede conseguir una sincronización óptima esta no afecte mucho a la señal durante la transmisión, esta característica se ve mejor representada en la figura 3 [6].

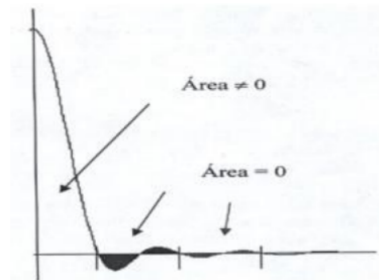


Fig. 3. Reducción de los lóbulos de la señal samplin [6]

• Filtro coseno levantado:

Basándose en los dos criterios de Nyquist dado que la señal en el dominio de la frecuencia se asemeja a un pulso rectangular es fundamental usar un filtro en el canal que permita pasar el rango de frecuencias necesario, pero en un escenario real es difícil de conseguir la caída perpendicular en un filtro por lo se usa el filtro coseno levantado.

Este filtro ayuda a cumplir ambos criterios, el factor

de roll-off afecta directamente la caída en el filtro lo que influye en la señal y en consecuencia afecta a la interferencia intersímbolo; es decir un factor de roll-off cercano a cero permite cumplir los dos criterios de Nyquist para eliminar el ISI. Debido a este suceso se debe la gran importancia del factor de roll-off en un filtro de coseno levantado, en la figura 4 se puede apreciar todo lo ya mencionado. [6]

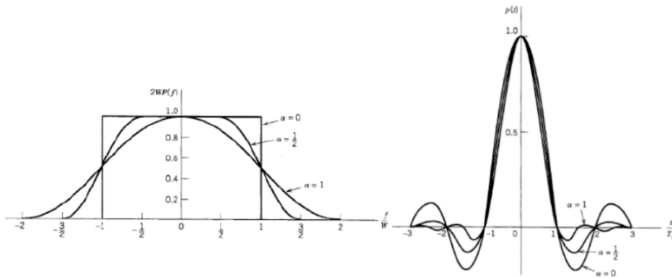


Fig. 4. Filtro coseno levantado con diferentes factores de roll-off [6]

C. Consultar las propiedades de las funciones de MATLAB `comm.RaisedCosineTransmitFilter` y `comm.RaisedCosineReceiveFilter` y cómo se aplican a un flujo de datos digital.

- **Función `comm.RaisedCosineTransmitFilter`:** El objeto de sistema de filtro de transmisión de coseno elevado aplica la forma de pulso al interpolar una señal de entrada utilizando un filtro FIR de coseno elevado.

Para realizar la interpolación de la señal de entrada se deben seguir los siguientes pasos:

- Definir y configurar el objeto de filtro de transmisión de coseno elevado.
- Colocar el paso para interpolar la señal de entrada de acuerdo con las propiedades de `comm.RaisedCosineTransmitFilter`. El comportamiento del paso es específico para cada objeto en la caja de herramientas.

Además, la construcción de este objeto se puede realizar de la siguiente manera:

- $H = \text{comm.RaisedCosineTransmitFilter}$ devuelve un objeto de sistema de filtro de transmisión de coseno elevado, H. Interpola la señal de entrada utilizando un filtro FIR de coseno elevado. El filtro utiliza una estructura de interpolación FIR polifásica eficiente y tiene energía unitaria.
- $H = \text{comm.RaisedCosineTransmitFilter}(\text{PropertyName}, \text{PropertyValue}, \dots)$ devuelve un objeto de filtro de transmisión de coseno elevado, H, con cada una de las propiedades especificadas con un valor establecido.

Adicionalmente, este objeto tiene las siguientes propiedades que pueden ser especificadas en la construcción de un script que utilice dicho objeto.

- **Shape:** Especifica la forma del filtro como una raíz normal o cuadrada. El valor predeterminado es raíz cuadrada
- **RolloffFactor:** Especifica el factor de caída como un escalar entre 0 y 1. El valor predeterminado es 0.2.
- **FilterSpanInSymbols:** Especifica el número de símbolos que abarca el filtro como un escalar positivo de valor entero. El valor predeterminado es 10. Debido a que el filtro de coseno elevado ideal tiene una respuesta de impulso infinita, el objeto trunca la respuesta de impulso al valor que se especifica para esta propiedad.
- **OutputSamplesPerSymbol:** Especifica el número de muestras en la salida para cada símbolo de entrada. El valor predeterminado es 8, sin embargo, acepta un cualquier valor escalar mientras sea un entero positivo.
- **Gain:** Especifica la ganancia lineal del filtro como un escalar numérico positivo. El valor predeterminado es 1. El objeto diseña un filtro de coseno elevado que tiene energía unitaria y luego aplica la ganancia lineal para obtener los valores finales de derivación [7].

- **Función `comm.RaisedCosineReceiveFilter`:** El objeto de sistema de filtro de recepción de coseno elevado aplica la forma de pulso diezmando una señal de entrada utilizando un filtro FIR de coseno elevado.

Para diezmar la señal de entrada:

- Definir y configurar el objeto de filtro de recepción de coseno elevado.
- Llamar al paso para diezmar la señal de entrada de acuerdo con las propiedades de `comm.RaisedCosineReceiveFilter`. El comportamiento del paso es específico para cada objeto en la caja de herramientas.

La construcción de este objeto se puede ser realizada de las siguientes maneras:

- $H = \text{comm.RaisedCosineReceiveFilter}$ devuelve un objeto del sistema de filtro de recepción de coseno elevado, H, que diezma la señal de entrada. El filtro utiliza una estructura eficaz de diezmando FIR polifásico y tiene energía unitaria.
- $H = \text{comm.RaisedCosineReceiveFilter}(\text{PropertyName}, \text{PropertyValue}, \dots)$ devuelve un objeto de filtro de recepción de coseno elevado, H, con cada propiedad especificada establecida en el valor especificado.

Las siguientes propiedades pueden ser definidas en dicho objeto:

- **Shape:** Especifica la forma del filtro como una raíz normal o cuadrada. El valor predeterminado es raíz cuadrada.
- **RolloffFactor:** Especifica el factor de caída como un escalar entre 0 y 1. El valor predeterminado es 0.2.

- **FilterSpanInSymbols:** Especifica el número de símbolos que abarca el filtro como un escalar positivo con valor entero. El valor predeterminado es 10. Debido a que el filtro de coseno elevado ideal tiene una respuesta de impulso infinita, el objeto trunca la respuesta de impulso al valor que se especifica para esta propiedad.
- **InputSamplesPerSymbol:** Especifica el número de muestras de entrada que representan un símbolo. El valor predeterminado es 8. Esta propiedad acepta un valor escalar positivo o doble con valor entero.
- **DecimationFactor:** Especifica el factor por el cual el objeto reduce la frecuencia de muestreo de la señal de entrada. El valor predeterminado es 8. Esta propiedad acepta un valor escalar entero positivo entre 1 y InputSamplesPerSymbol. Si su valor se establece en 1, el objeto solo aplica el filtrado sin disminución de resolución.
- **DecimationOffset:** Especifica el número de muestras filtradas que el objeto del sistema descarta antes de reducir el muestreo. El valor predeterminado es 0. Esta propiedad acepta un escalar con valor entero entre 0 y DecimationFactor - 1.
- **Gain:** Especifica la ganancia lineal del filtro como un escalar numérico positivo. El valor predeterminado es 1. El objeto diseña un filtro de coseno elevado que tiene energía unitaria y luego aplica la ganancia lineal para obtener los valores finales de derivación [8].

D. Codificar un script empleando las funciones comm.RaisedCosineTransmitFilter y fvtool para observar la respuesta en frecuencia de tres filtros con factor de roll-off de 0.2, 0.5 y 0.9 respectivamente. Comentar las diferencias observadas.

El siguiente script realiza una modulación 4-PSK en base a 100 datos aleatorios que pueden tomar valor entre 0 y 15 en este caso.

```
SeFil=FCos(SeTx); %Aplicacion del filtro
Coseno levantado a la senal tx
fvtool(SeFil)% Visualizacion de la
respuesta en frecuencia del filtro
```

A esta señal modulada, se la pasa por filtro Coseno Levantado con un factor de roll-off específico, además, para visualizar la respuesta de frecuencia de dicho filtro se utiliza la función “fvtool”. A continuación en las figuras 5, 6 y 7 se muestran los resultados obtenidos cuando el factor de roll-off toma diferentes valores.

- Factor de roll-off: 0.2

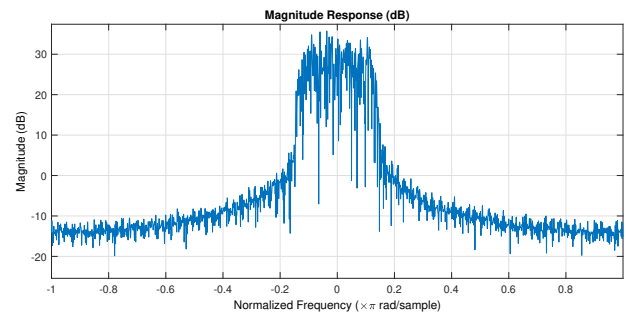


Fig. 5. Modulación 4-PSK con filtro Coseno Levantado

- Factor de roll-off: 0.5

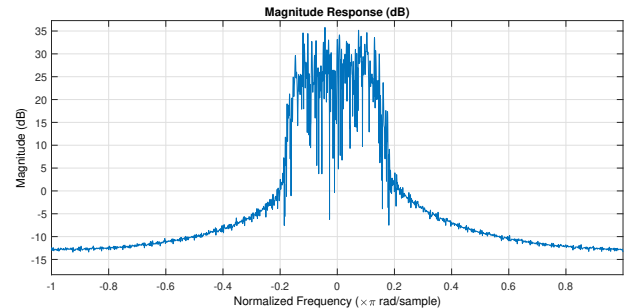


Fig. 6. Modulación 4-PSK con filtro Coseno Levantado

- Factor de roll-off: 0.9

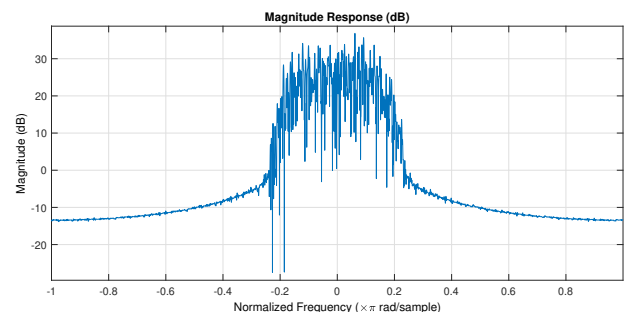


Fig. 7. Modulación 4-PSK con filtro Coseno Levantado

```
1 %% M-PSK
2 clear all
3 close all
4 clc
5
6 M=4; %Numero de fases;
7 datos=randi([0 M-1], 100,1); %Vector de
8   datos aleatorio con
9   %valores entre 0 y M-1
10 SeTx=pskmod(datos, M); %Senal modulada
11   con PSK
12 FCos= comm.RaisedCosineTransmitFilter('
13   RolloffFactor',0.9); %Creacion del
14   objeto que
15   %describe el filtro de coseno levantado
16   con un valor especifico de factor
17   %de Roll off
```

E. Codificar un script que permita modular y demodular una secuencia de datos aleatorios que atraviesa un canal AWGN como se presenta en el diagrama de bloques de la Figura 8. El script deberá presentar los símbolos en el tiempo y mostrar los diagramas del ojo en el bloque de recepción una vez que la señal ha sido recibida y filtrada, de acuerdo con los parámetros asignados a cada grupo.

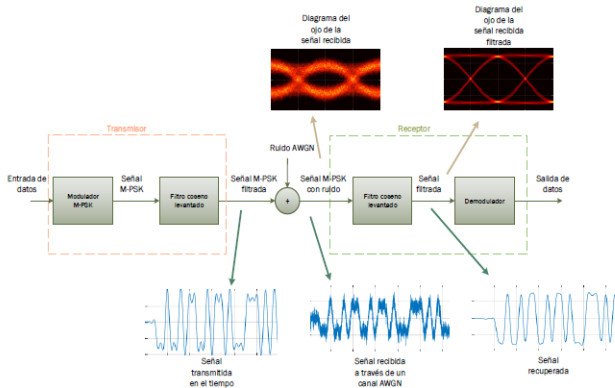


Fig. 8. Modulación M-PSK con filtros sobre un canal AWGN [1]

- En este caso se implementará un script que cumpla con las siguientes condiciones: modulación BPSK, factor de $roll-off = 0.9$ y $SNR = 10dB$. El segundo script de Matlab que se presenta a continuación cumple con dichas características.

```
1 %% BPSK
2 clear all
3 close all
4 clc
5 M=2; %Numero de fase de BPSK
6 SNR=10; %Valor de SNR en dB
7 K=100; %Numero datos
8 datos=randi([0 M-1],K,1); %vector
    aleatorio k datos entre 0 y M-1]
9 t=1:1:8*K; %Vector de tiempo para
    graficar la senal Tx y y senal Rx
10 SeTx=pskmod(datos, M); %Senal de datos
    modulada con M-PSK
11 FCos= comm.RaisedCosineTransmitFilter(
    'RolloffFactor',0.9); %Creacion del
    objeto que
12 %describe el filtro de coseno
    levantado con un valor especifico
    de factor
13 %de Roll off
14 SeFil=FCos(SeTx); %Aplicacion del
    filtro Coseno levantado a la senal
    tx
15 %Grafico senal tx en el tiempo
16 subplot(3,1,1)
17 plot(t,real(SeFil))
```

```
18 %stairs((0:length(SeFil)-1),real(SeFil
    ));
19 title('Senal transmitida en tiempo');
20 xlabel('Tiempo[s]');
21 ylabel('Amplitud [V]');
22 grid on;
23
24 SN=awgn(SeFil,SNR); %Paso a traves de
    un canal AWGN
25
26 %Grafico senal recibida a traves de un
    canal AWGN
27 subplot(3,1,2)
28 plot(t,real(SN))
29 %stairs((0:length(SN)-1),real(SN));
30 title('Senal recibida a traves de un
    canal AWGN');
31 xlabel('Tiempo[s]');
32 ylabel('Amplitud [V]');
33 grid on;
34 FCos1=comm.RaisedCosineReceiveFilter('
    RolloffFactor',0.9);
35 SeFill=FCos1(SN); %Aplicacion del
    filtro a la senal con ruido AWGN
36 t2=1:1:K; %Vector de tiempo para
    graficar la senal recuperada
37 %Grafico senal recuperada en tiempo
38 subplot(3,1,3)
39 plot(t2,real(SeFill))
40 %stairs((0:length(SeFill)-1),real(
    SeFill));
41 title('Senal recuperada en tiempo'); %
    codigo binario de Entrada
42 xlabel('Tiempo[s]');
43 ylabel('Amplitud [V]');
44 grid on;
45 %% Diagramas de ojo
46
47 eyediagram(SN,2*M); %Diagrama del ojo
    de la senal recibida
48 eyediagram(SeFill,2*M); %Diagrama del
    ojo de la senal recibida filtrada
```

Como resultado de este script se obtienen todos los gráficos que se muestran en la figura 9. Es decir, en la figura 9 se puede apreciar la gráfica de la señal transmitida en tiempo, recibida a través de un canal AWGN y recuperada.

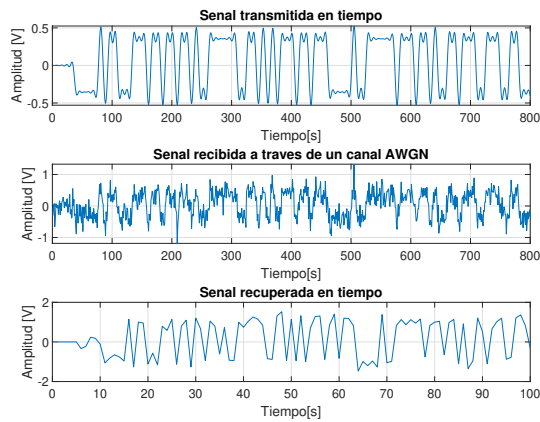


Fig. 9. Señales de datos transmitidas

En la figura 10, se aprecia el diagrama del ojo de la señal recibida luego de ser modulada con BPSK, la misma que atravesó un canal AWGN con un valor de relación señal a ruido de 10 dB.

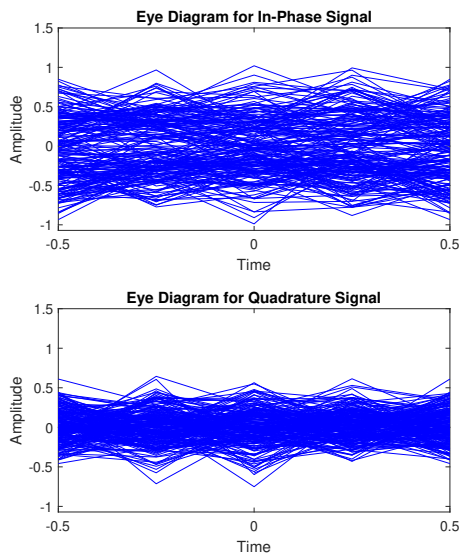


Fig. 10. Diagrama del ojo de la señal recibida

Por último, en la siguiente figura se presenta el diagrama del ojo de la señal recibida filtrada, con un filtro de coseno levantado cuyo factor de roll off es de 0.9.

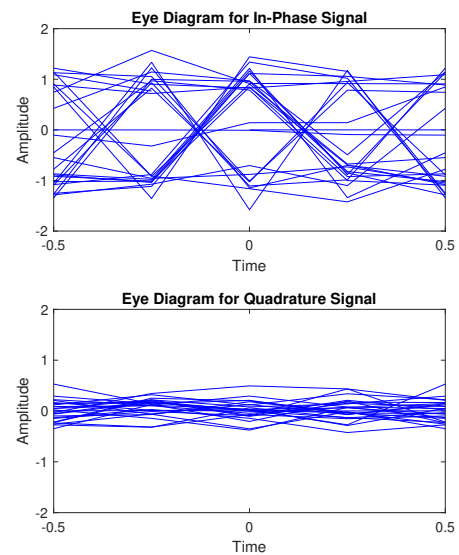


Fig. 11. Diagrama del ojo de la señal recibida filtrada

REFERENCES

- [1] E. Tatayo, "ANÁLISIS DE CANALES CON INTERFERENCIA INTERSÍMBOLO". C.P. COMUNICACIÓN DIGITAL, Accedido: jul. 17, 2020. [En línea].
- [2] "Baseband Data Transmission and Reception", en Telecommunications Engineering, 0 vols., WORLD SCIENTIFIC, 2018, pp. 509–578.
- [3] matias, "Señales en banda base", jul. 27, 2005. <https://www.textoscientificos.com/redes/senales/banda-base> (consultado jul. 22, 2020).
- [4] "Broadband Transmission", Network Encyclopedia. <https://networkencyclopedia.com/broadband-transmission/> (consultado jul. 22, 2020).
- [5] "The first Nyquist criterion - DSPIllustrations.com". <https://dspillustrations.com/pages/posts/misc/the-first-nyquist-criterion.html> (consultado jul. 22, 2020).
- [6] "Grupo 10 Resumen Criterio de Nyquist para la trasmision sin distorsión", StuDocu. <https://www.studocu.com/co/document/universidad-industrial-de-santander/comunicaciones-1/informe/grupo-10-resumen-criterio-de-nyquist-para-la-trasmision-sin-distorsion/2598576/view> (consultado jul. 22, 2020).
- [7] "Apply pulse shaping by interpolating signal using raised cosine filter - MATLAB - MathWorks América Latina". <https://la.mathworks.com/help/comm/ref/comm.raisedcosinetransmitfilter-system-object.html> (accedido jul. 17, 2020).
- [8] "Apply pulse shaping by decimating signal using raised cosine filter - MATLAB - MathWorks América Latina". <https://la.mathworks.com/help/comm/ref/comm.raisedcosinereceivefilter-system-object.html> (accedido jul. 17, 2020).