

# Laboratorio 2: Interferencia Intersímbolo (ISI)

Felipe Ulloa 1, felipe.ulloa1@mail.udp.cl  
 Alex Parada 2, email alex.parada@mail.udp.cl  
 Fernando Vergara 3, fernando.vergara1@mail.udp.cl  
 Escuela de Informática y Telecomunicaciones  
 Universidad Diego Portales

## I. INTRODUCCIÓN

La interferencia intersímbolo es una forma de distorsión de una señal, cuando un pulso se fuga a los símbolos adyacentes causa interferencia, a esta se le denomina interferencia intersímbolo o intersimbólica (ISI). Esto tiene un efecto similar al ruido por lo que nuestras señales serán menos fiables en nuestras transmisiones digitales ya que comprometería la integridad de los datos transmitidos. Para contrarrestar esto se utiliza el filtro de coseno elevado de Nyquist filtro que será utilizado y desarrollado a lo largo de la experiencia, bajo este filtro se generará un diagrama de ojo con la finalidad de ver las repercusiones de este al cambiar parámetros o aplicar un filtro denominado "AWGN".

## II. METODOLOGÍA

### A. Actividad previa

La experiencia comenzó con la realización de una actividad previa que consistió en la creación de la gráfica de la respuesta al impulso para un pulso coseno alzado, y además se realizó otro gráfico para la respuesta en frecuencia para este mismo pulso.

1) **Gráfica coseno alzado:** Se gráfica a través de la herramienta "Matlab", el pulso coseno alzado con una frecuencia de ancho de banda de 6db, tal pulso nos permitirá analizar y crear posteriormente el diagrama de ojo.

Para realizar tal gráfica es necesario establecer un factor *roll-off*, tal factor denomina cual es el porcentaje de ancho de banda que excede la señal con respecto al ancho de banda que ocuparía un pulso rectangular [3], además de este factor, se establece los parámetros que nos permitirán realizar la señal.

```
1 fo=1000
2 fs=fo*10;
3 alpha = 0.22;
4 cosNum = cos(2*pi*alpha*fo*t);
5 cosDen = (1-(4*alpha*fo*t).^2);
6 cosDenZero = find(abs(cosDen)< 10^-10);
7 cosOp = cosNum./cosDen;
8 cosOp(cosDenZero) = pi/4;
```

2) **Respuesta al impulso:** Para poder graficar y obtener la respuesta al impulso es necesario esclarecer que significa como tal la respuesta al impulso. Se define como la respuesta en el dominio del tiempo del sistema [2], esto quiere decir que nos va a mostrar información tanto de la amplitud como

del tiempo.

En temas de procedimiento, se establece la multiplicación con las variables anteriormente utilizadas para la creación del coseno alzado con el fin de establecer la respuesta al impulso.

```
1 h_et = 2*fo*sincOp.*cosOp;
```

3) **Respuesta en frecuencia:** Para poder graficar y obtener la respuesta en frecuencia, se utilizó el comando "fft" proporcionado por la herramienta "Matlab", con esta herramienta nos permite calcular la transformada de fourier que nos permite transformar señales entre el dominio del tiempo al dominio de la frecuencia, para poder analizar la señal desde una perspectiva de la frecuencia.

```
1 H_et=fft(h_et);
2 f=fs*(-(length(H_et)-1)/2:(length(H_et)-1)/2)/length(H_et);
```

### B. Laboratorio Presencial

A continuación se realizó el diagrama de ojo para el pulso coseno alzado utilizando la modulación BPSK, además de establecer un canal perfecto AWGN.

Se establecieron los siguientes parámetros

```
1 %Frecuencia y periodo de sincronizacion
2 fs=1000;
3 Ts=1/fs;
4 Frecuencia de muestreo de Nyquist fn>=2
   Frecuencia max
5 fm=10*fs;
6 Tm=1/fm;
```

Luego se realizó la generación de los  $10^5$  junto con las muestras y un vector de tiempo.

```
1%a_n, valores que tendrán los pulsos [-1,1]
   ant poda binaria
2valores=(2*randi([0,1],[1,10^5]))-1;
3
4%Muestras para un periodo de sincronización
```

```

sm=Ts/Tm;
6
7%Vector de tiempo, respecto a los impulsos que
  tenga el Tren.
8t=[-(length(valores)/2)*Ts:Tm:(length(valores)
  /2)*Ts];

```

Luego se generó el tren de impulsos para las  $m$  muestras anteriores.

```

1%Tren de Impulsos separados en Ts => por m
  muestras
2TrenImpulsos=t;
3contador=m-1;
4indiceValores=1;
5for index=1 : (length(t)-1)
6    if(indiceValores<=length(valores))
7        if(contador == (m-1) )
8            TrenImpulsos(index)=valores(
              indiceValores);
9            indiceValores=indiceValores+1;
10           contador=0;
11        else
12            TrenImpulsos(index)=0;
13            contador=contador+1;
14        end
15    else
16        TrenImpulsos(index)=0;
17    end
18end

```

A continuación se generó una onda con período  $t_s$  asignado anteriormente. Junto a una función de transferencia filtro transmisor, y la creación de la señal de entrada.

```

1%Onda cajon con periodo Ts, funcion
  transferencia filtro trasmisor h_t(t)
2h_t=t;
3for index = 1 : length(h_t)
4    if( (length(h_t)/2)- (m/2) <= index &&
        index <= (length(h_t)/2)+(m/2)) %
        Bonito
5        h_t(index)=1;
6    else
7        h_t(index)=0;
8    end
9end
10
11%Se al de entrada x(t)
12x_t=conv(TrenImpulsos,h_t);
13t2=[-((length(x_t)/2)/m)*Ts:Tm:((length(x_t)
  /2)/m)*Ts];

```

Luego se realiza el pulso coseno alzado, utilizando lo realizado en la actividad previa. La única modificación fue de asignar  $\alpha$  (alpha) como 0.22

```

1%Coseno alzado
2%Frecuencia de ancho de banda de 6dB, mitad de
  velocidad de simbolos
3fo=fs/2;% fo=1000/2=500 Hz
4sincNum = sin(2*pi*fo*t);

```

```

5sincDen = (2*pi*fo*t);
6sincDenZero = find(abs(sincDen) < 10^-10);
7sincOp = sincNum./sincDen;
8sincOp(sincDenZero) = 1;
9
10alpha = 0.22;
11cosNum = cos(2*pi*alpha*fo*t);
12cosDen = (1-(4*alpha*fo*t).^2);
13cosDenZero = find(abs(cosDen)< 10^-10);
14cosOp = cosNum./cosDen;
15cosOp(cosDenZero) = pi/4;

```

Luego de esto se asigna la respuesta de impulso equivalente con las fracciones del seno y el coseno creadas en el paso anterior, luego mediante la función `fft` que retorna la transformada de fourier de un vector se asigna el parámetro y calcula la función equivalente de transferencia.

```

1%Respuesta impulso equivalente h_e(t)
2h_et = 2*fo*sincOp.*cosOp;
3%Funcion equivalente de tranferencia H_e(t)
4H_et=fft(h_et);
5f=fs*(-(length(H_et)-1)/2:(length(H_et)-1)/2)/
  length(H_et);

```

Luego se calcula la convolución del "Tren de Impulsos" y  $h_{et}$  esto con la finalidad de transmitir una señal por pulso de Nyquist, evitando la interferencia intersimbólica mientras se cumpla la frecuencia de sincronización para luego crear el canal perfecto AWGN.

```

1SecuenciaFiltrada=conv(TrenImpulsos,h_et);
2tfiltrada=[-((length(SecuenciaFiltrada)/m)/2*
  Ts):Tm:(length(SecuenciaFiltrada)/m)/2*Ts
  ];
3
4SecuenciaFiltradaRecortada=[1:length(valores)*
  m];
5shift=length(SecuenciaFiltrada)-length(x_t);
6for i=1:length(valores)*m
7    SecuenciaFiltradaRecortada(i)=
      SecuenciaFiltrada(shift+i);
8end
9
10filtro = reshape(SecuenciaFiltradaRecortada,2*
  m,[]);
11y=awgn(filtro,18,'measured');

```

Para finalizar se realizaron los gráficos de la siguiente manera.

```

1plot(filtro,'b'); title('Diagrama de ojo');
  xlabel('time');ylabel('amplitude');grid on
2
3
4
5plot(y,'b');title('Diagrama de ojo con AWGN');

```

### III. RESULTADOS

En el siguiente apartado se muestran los resultados de ejecutar el código planteado para la creación de la respuesta, tanto de impulso como de frecuencia del pulso coseno alzado para distintos factores *roll-off*, además de la creación del diagrama de ojo junto con la aplicación del filtro AWGN.

Para realizar lo anterior mencionado, se utilizaron de manera general los siguientes parámetros.

$$Frecuencia = 1000$$

$$Periodo\ de\ sincronizacion = \frac{1}{1000}[Ts]$$

$$Frecuencia\ de\ muestreo\ de\ Nyquist = 10000$$

$$Periodo\ de\ muestreo = \frac{1}{10000}[tm]$$

$$Muestras\ para\ un\ periodo\ de\ sincronizacion = \frac{ts}{tm}$$

$$Factores\ roll-off = 0 - 0.25 - 0.75 - 1$$

(1)

#### A. Actividad 1

En las siguientes imágenes, se muestra el resultado de aplicar distintos factor de *roll-off* a la respuesta de impulso del coseno alzado y a la frecuencia del coseno alzado:

##### 1) Impulso roll-off 0:

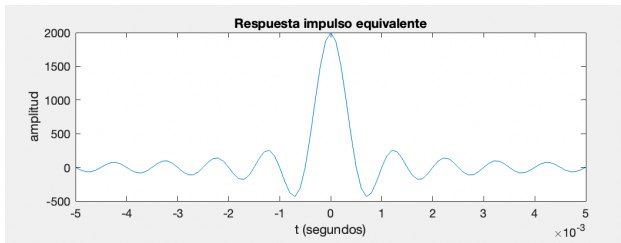


Fig. 1. Respuesta en impulso con roll-off de 0

##### 2) Frecuencia roll-off 0:

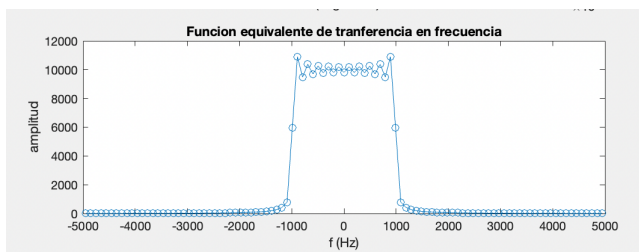


Fig. 2. Respuesta en impulso con roll-off de 0

##### 3) Impulso roll-off 0.25:

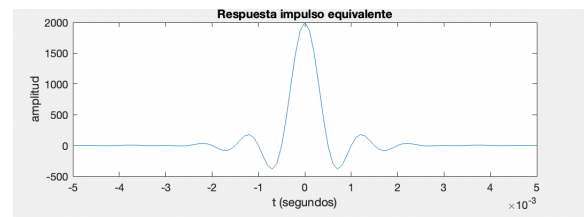


Fig. 3. Respuesta al impulso con roll-off de 0.25

##### 4) Frecuencia roll-off 0.25:

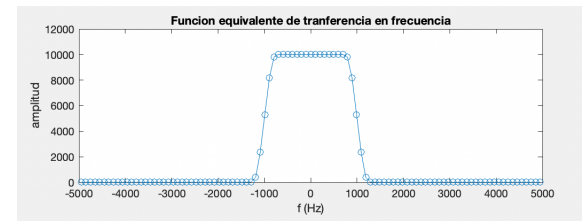


Fig. 4. Respuesta en frecuencia con roll-off de 0.25

##### 5) Impulso roll-off 0.75:

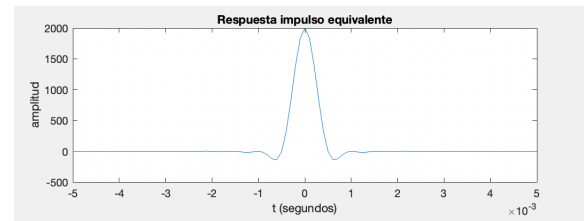


Fig. 5. Respuesta al impulso con roll-off de 0.75

##### 6) Frecuencia roll-off 0.75:

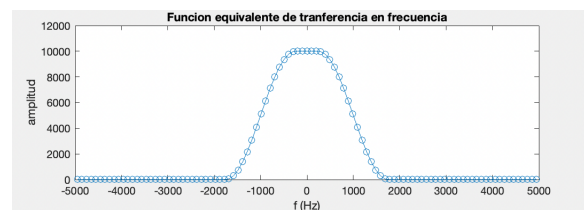


Fig. 6. Respuesta en frecuencia con roll-off de 0.75

##### 7) Impulso roll-off 1:

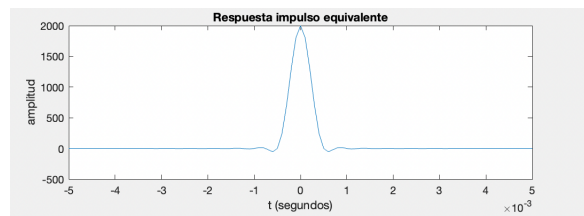


Fig. 7. Respuesta al impulso con roll-off de 1

##### 8) Frecuencia roll-off 1:

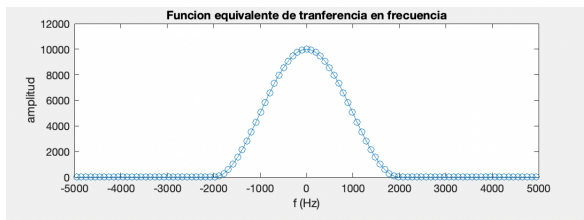


Fig. 8. Respuesta en frecuencia con roll-off de 1

## B. Actividad 2

En el siguiente apartado se muestra los resultados al ejecutar el código planteado para la creación del diagrama de ojo para el pulso de coseno alzado, por otra parte se vera las consecuencias de modificar las frecuencias de muestreo y el factor roll-off de este pulso.

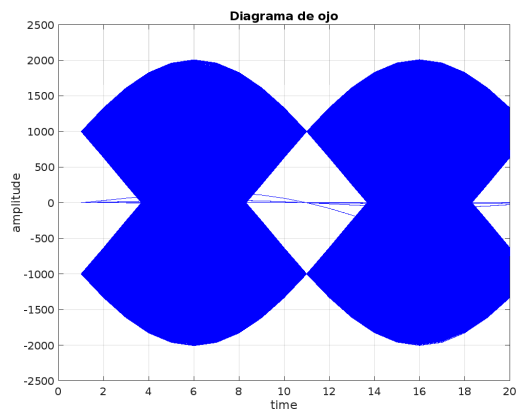


Fig. 9. Diagrama ojo a partir de roll-off 0.22

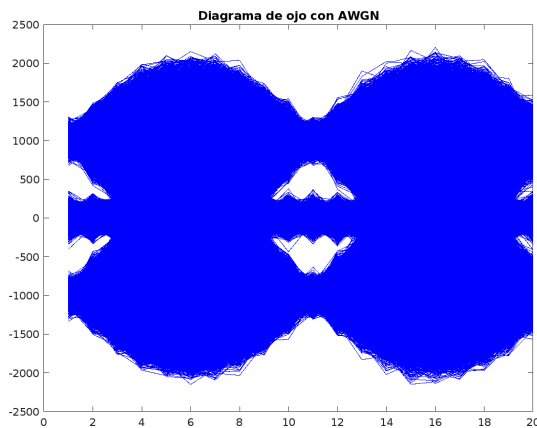


Fig. 10. Diagrama ojo AWGN a partir del roll-off 0.22

## C. Cuestionario

### 1) ¿Que pasa si disminuye la frecuencia de muestreo?

**R:** Al disminuir la frecuencia de muestreo, se generó el siguiente diagrama de ojo.

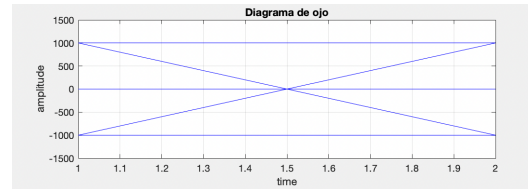


Fig. 11. Diagrama ojo con frecuencia de muestreo equivalente a 1000

Donde se observa que si, al disminuir considerablemente la frecuencia de muestreo del pulso coseno alzado se estaría restringiendo considerablemente el ancho de banda de este mismo, esto ocurre siempre y cuando el cambio de frecuencia de muestreo sea considerable. En caso contrario de que la modificación sea menor, la restricción al ancho de banda no será significativa, sin embargo no influye en que genere una pérdida de muestras.

### 2) En forma general, ¿Qué sucede al incrementar el valor de $\alpha$ ?

**R:** Al ir incrementando el factor roll-off, el diagrama de ojo generado por el pulso coseno alzado poseerá un mayor ancho de banda en comparación a pulsos de menor valor de factor roll-off, además el coseno alzado poseerá un ancho de banda superior al pulso cuadrado (dependiendo del factor que se utilice), ya que por definición el factor roll-off ( $\alpha$ ) establece el porcentaje de ancho de banda que supera el pulso del coseno alzado con respecto al ancho de banda que ocupa el pulso rectangular. [3]

Por otra parte al ir aumentando el roll-off, se va solucionando el problema de "jitter" que es la fluctuación del retardo de señales [1], esto a mayor roll-off la fluctuación es casi nula permitiendo que la señal pase por 0 en el mismo instante relativo.

## IV. CONCLUSIONES

Al alterar de manera que vaya aumentando el factor roll-off, se demuestra que el ancho de banda será superior en comparación al pulso rectangular, esto provoca que se requiera abarcar más en el espectro de frecuencia con el fin de transmitir la señal y disminuir la interferencia intersimbólica.

A pesar que al usar un roll-off de 1 (que es el máximo que se puede ocupar) ocupe el doble de ancho de banda mínimo, esto lo solventa al poder solucionar el problema denominado "jitter", permitiendo sincronizar la señal recibida en el instante 0.

Otra forma de afectar el ancho de banda del pulso coseno alzado es modificar la frecuencia de muestreo, como se vio en el apartado de "análisis de resultados", al modificar significativamente la frecuencia de muestreo, el pulso tiende a perder una cantidad considerable de muestras y cambiando su forma abruptamente pasando a restringir el ancho de banda de dicha señal.

Finalmente si se ocupa un filtro "AWGN" y se aplica ruido al diagrama de ojo, esto provocará que el diagrama de ojo posea una notable cantidad de interferencia intersimbólica provocando un error en los flujos de bits en el diagrama de ojos.

## ANEXOS

## A. Código Explicación

```

1 fo=1000;
2 T=1/fo;
3 fs = fo*10;
4 Ts=1/fs;
5 %a_n, valores que tendran los pulsos [-1,1]
6 valores=(2*randi([0,1],[1,10^1]))-1;
7 %Muestras para un periodo T
8 m=T/Ts;
9 %Tren de pulsos separados en T <=> m muestras
10 TrenImpulsos=upsample(valores,m)
11 t=[-(length(TrenImpulsos)/m)/2*T:Ts:(length(TrenImpulsos)/m)/2*T];
12
13 %Onda cajon con periodo T, funcion transferencia filtro trasmisor h_t(t)
14 h_t=t;
15 for index = 1 : length(h_t)
16     if (length(h_t)/2) - (m/2) <= index && index <= (length(h_t)/2) + (m/2)
17         h_t(index)=1;
18     else
19         h_t(index)=0;
20     end
21 end
22
23 %Se al de entrada x(t)
24 x_t=conv(valores,h_t);
25 t2=[-(length(x_t)/2)/m)*T:Ts:+((length(x_t)/2)/m)*T];
26
27 figure(1)
28 subplot(3,1,1); stem(t, [TrenImpulsos 0]);title('Tren de Impulsos a distancias T=0,001(s)');
29     xlabel('t (segundos)'); ylabel('amplitud');
30 subplot(3,1,2); plot(t, h_t, '-o'); title('h_t(t) funcion transferencia filtro transmisor');
31 subplot(3,1,3); plot(t2,[x_t 0], '-o'); title('Se al de entrada x(t) = TrenPulsos * h_t(t)');
32
33 %Coseno alzado
34 sincNum = sin(2*pi*fo*t);
35 sincDen = (2*pi*fo*t);
36 sincDenZero = find(abs(sincDen) < 10^-10);
37 sincOp = sincNum./sincDen;
38 sincOp(sincDenZero) = 1;
39
40 alpha = 0.22;
41 cosNum = cos(2*pi*alpha*fo*t);
42 cosDen = (1-(4*alpha*fo*t).^2);
43 cosDenZero = find(abs(cosDen)< 10^-10);
44 cosOp = cosNum./cosDen;
45 cosOp(cosDenZero) = pi/4;
46
47 %Respuesta impulso equivalente h_e(t)
48 h_et = 2*fo*sincOp.*cosOp;
49 %Funcion equivalente de tranferencia H_e(t)
50 H_et=fft(h_et);
51 f=fs*(-(length(H_et)-1)/2:(length(H_et)-1)/2)/length(H_et)
52 figure(2)
53 subplot(2,1,1); plot(t, h_et);title('Respuesta impulso equivalente'); xlabel('t (segundos)');
54     ylabel('amplitud')
55 subplot(2,1,2); plot(f,abs(fftshift(H_et)), '-o'); title('Funcion equivalente de tranferencia
56     en frecuencia');xlabel('f (Hz)'); ylabel('amplitud')
57
58
59 %Secuencia Filtrada=conv(TrenImpulsos,h_et);
60
61 tfiltrada=[-(length(SecuenciaFiltrada)/m)/2*T:Ts:(length(SecuenciaFiltrada)/m)/2*T];
62 figure(3)

```

```

61 subplot(3,1,1); stem(t, [TrenImpulsos 0]);title('Tren de Impulsos a distancias T=0,001(s)');
    xlabel('t (segundos)'); ylabel('amplitud');
62 subplot(3,1,2); plot(tfiltrada, [SecuenciaFiltrada 0]);title('Secuencia filtrada con Coseno
    alzado, recortada en tiempo'); xlabel('t (segundos)');xlim([-5*T 5*T]); ylabel('amplitud');
63 subplot(3,1,3); plot(tfiltrada, [SecuenciaFiltrada 0]);title('Secuencia filtrada con Coseno
    alzado'); xlabel('t (segundos)'); ylabel('amplitud');
64 filtro = reshape(SecuenciaFiltrada,2*m,[]);
65 figure(4);
66 plot(filtro,'b');
67 title('eye diagram with alpha=0.22');
68 xlabel('time')
69 ylabel('amplitude')
70 grid on
71
72 y=awgn(filtro,20,'measured');
73 figure(5);
74 plot(y,'b');
75 title('eye diagram with alpha=0.22');
76 xlabel('time')
77 ylabel('amplitude')
78 grid on

```

## REFERENCES

- [1] Definición jitter. <https://techlib.net/definition/jitter.html>. Consultado: 2022-05-19.
- [2] Definición respuesta a impulso. <https://www.produccioneselotano.com/la-respuesta-de-impulso-esa-gran-desconocida/>. Consultado: 2022-05-19.
- [3] Filtro adaptado. <https://aholab.ehu.eus/prj/modem/pulsos/filtro.html>. Consultado: 2022-05-20.
- [4] H. Kopka and P. W. Daly. A Guide to  $\text{\LaTeX}$ , 3rd ed. Harlow. England: Addison-Wesley, 1999.