



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

UNIVERSIDAD POLITÉCNICA DE
VALENCIA

**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
TELECOMUNICACIÓN**

**Máster Universitario en Ingeniería de
Telecomunicación**

Práctica 1-2

Modulaciones ópticas avanzadas

ESC

Autor:

Andrés Ruz Nieto
Gerardo Arias Martínez

VALENCIA, 2022

— **TELECOM**
UPV VLC

Transmisor - OOK

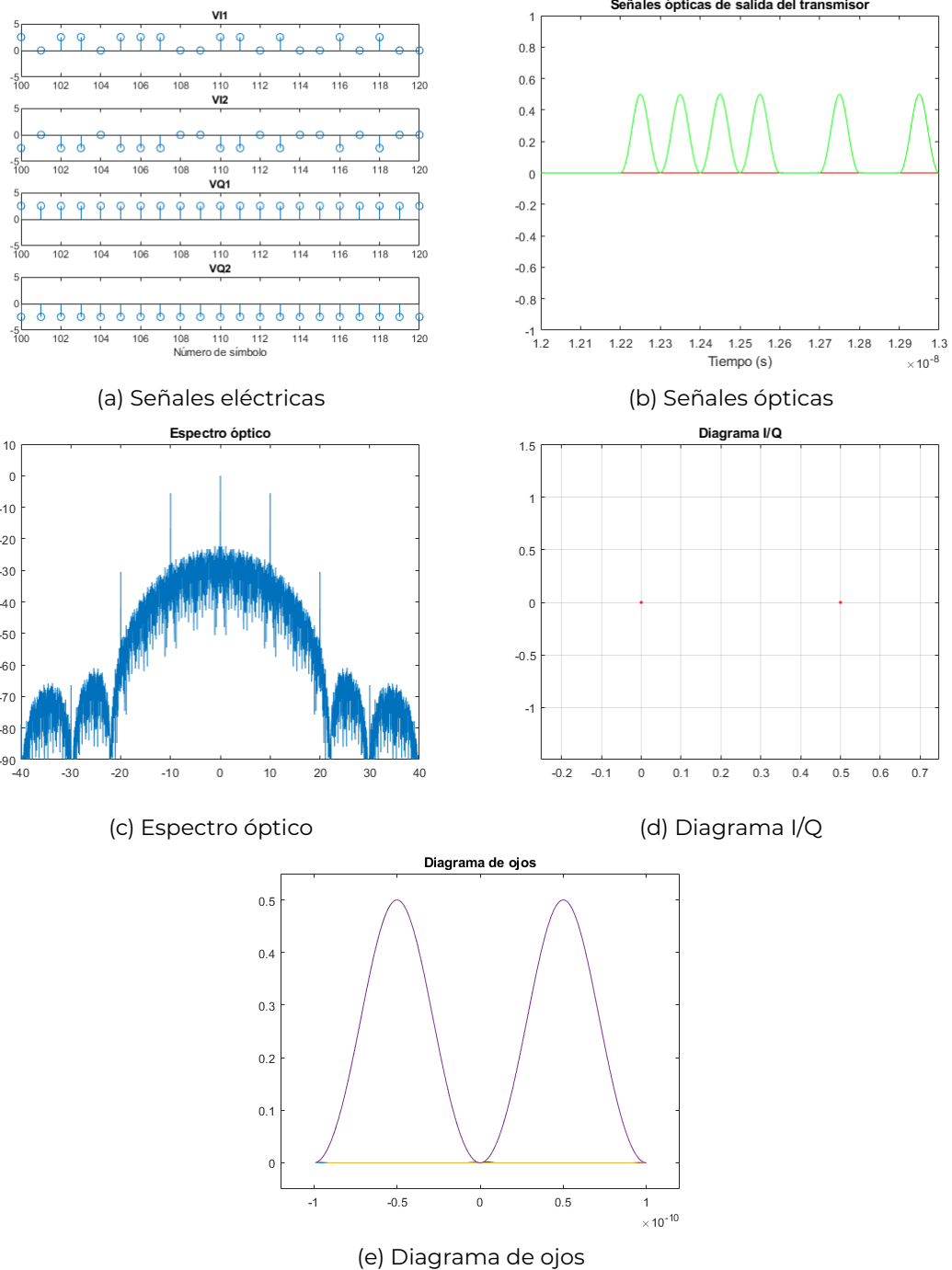


Figura 1: Modulación OOK

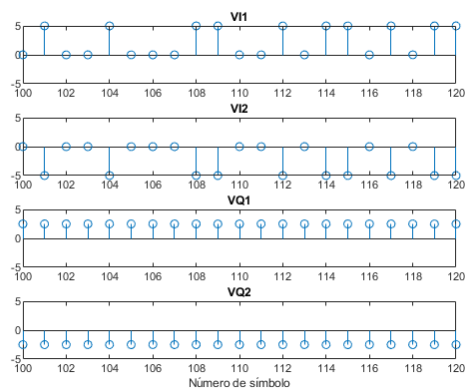
Al no tener parte imaginaria, se puede cómo VQ1 y VQ2 son constantes. Lo mismo pasará en BPSK. Las constelaciones (Diagrama I/Q) representan el número de símbolos que puede representar cada modulación.

Observando el diagrama de ojos, para la modulación OOK aparece una línea con amplitud máxima la unidad, representando las combinaciones de ceros y unos. A medida que se aumenta la complejidad de las modulaciones, el número de trazas aumenta.

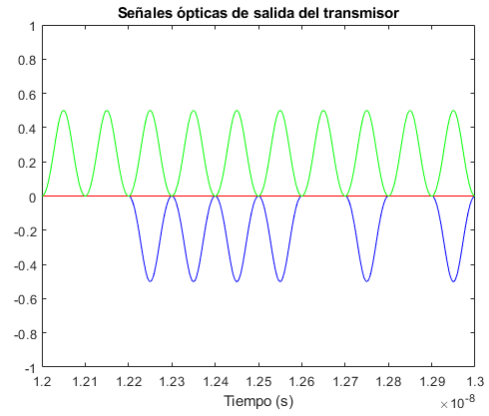
Esta modulación es la más fácil de analizar ya que únicamente está compuesta por dos símbolos reales,

las siguientes modulaciones irán incrementando su complejidad.

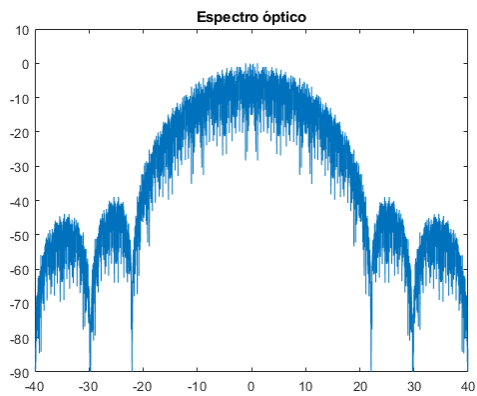
Transmisor - BPSK



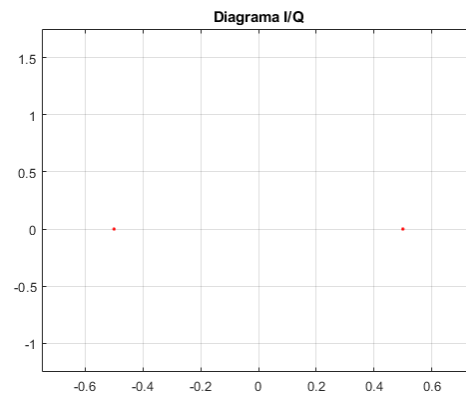
(a) Señales eléctricas



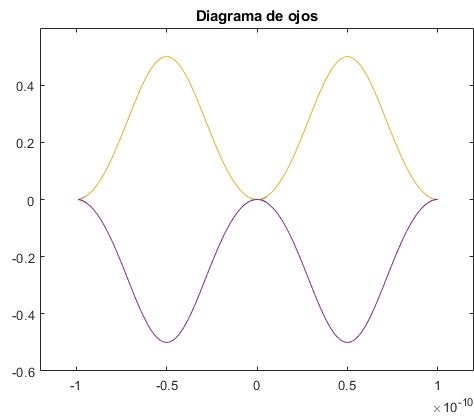
(b) Señales ópticas



(c) Espectro óptico



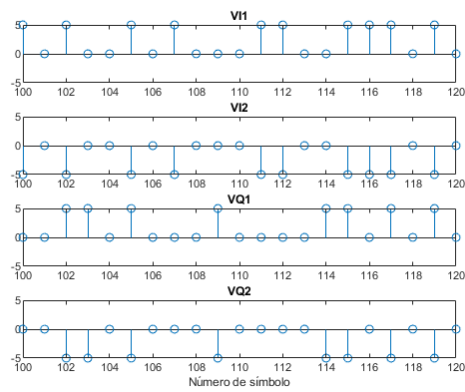
(d) Diagrama I/Q



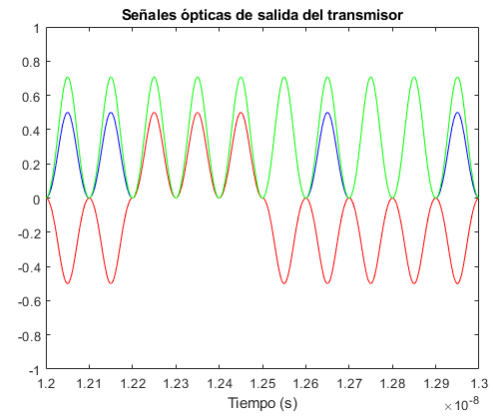
(e) Diagrama de ojos

Figura 2: Modulación BPSK

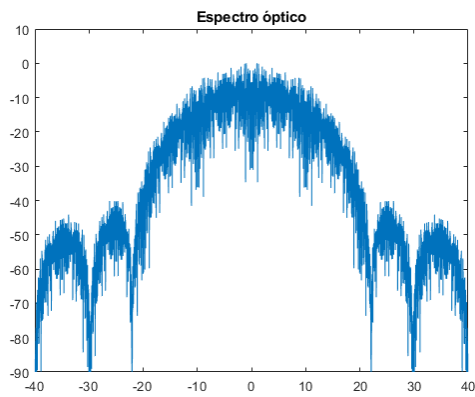
Transmisor - QPSK



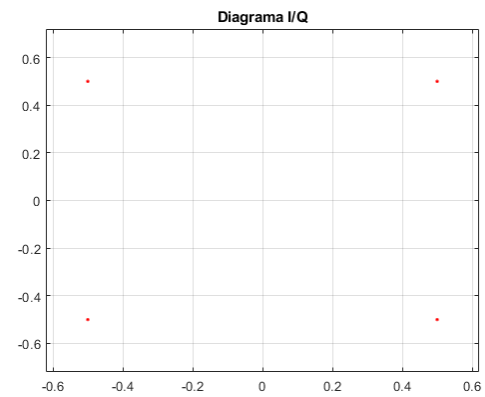
(a) Señales eléctricas



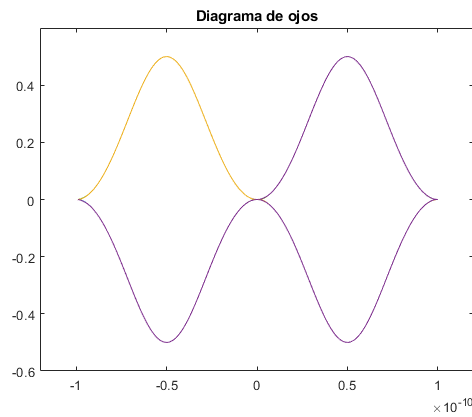
(b) Señales ópticas



(c) Espectro óptico



(d) Diagrama I/Q

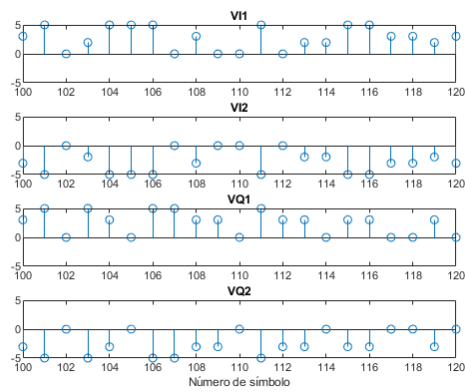


(e) Diagrama de ojos

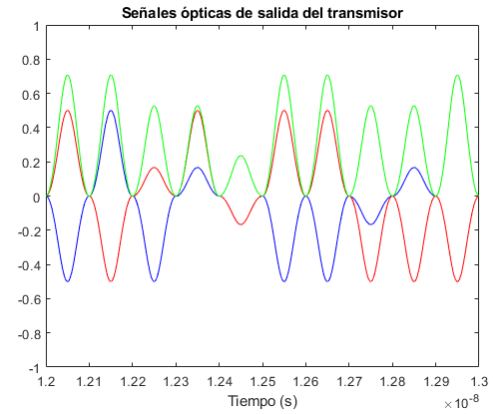
Figura 3: Modulación QPSK

Aquí ya se puede apreciar la componente imaginaria de la modulación QPSK.

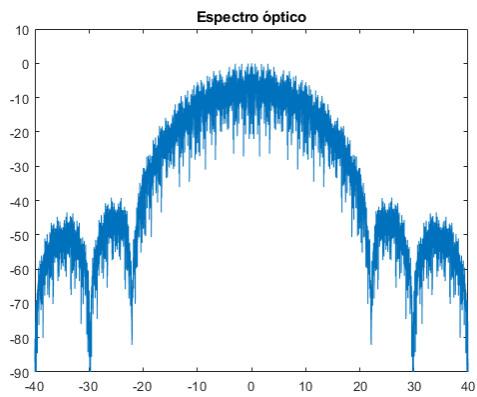
Transmisor - 16-QAM



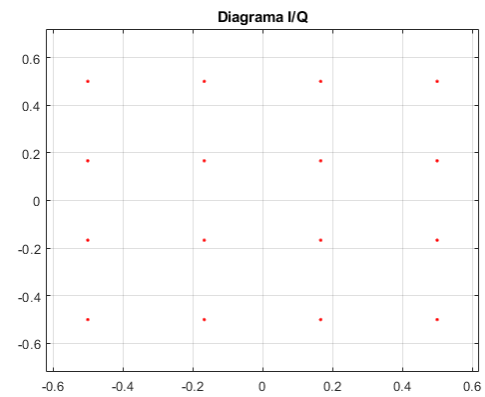
(a) Señales eléctricas



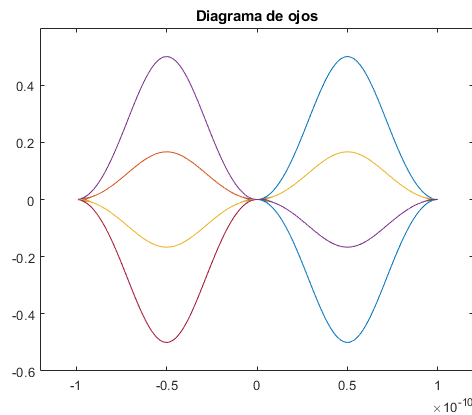
(b) Señales ópticas



(c) Espectro óptico



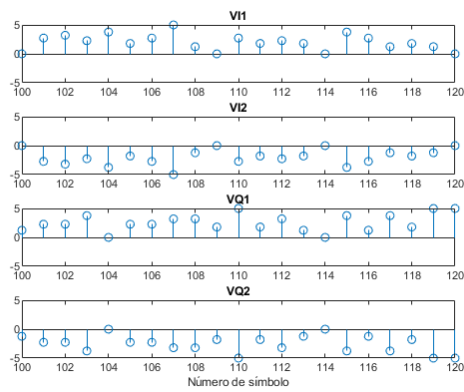
(d) Diagrama I/Q



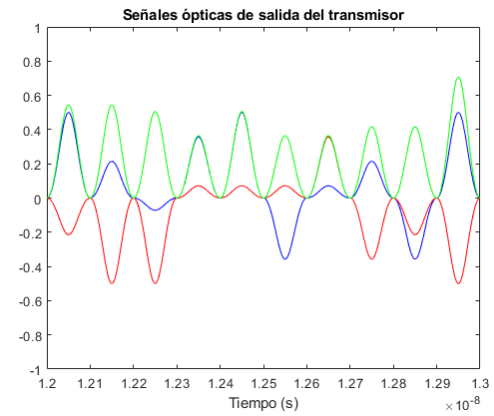
(e) Diagrama de ojos

Figura 4: Modulación 16-QAM

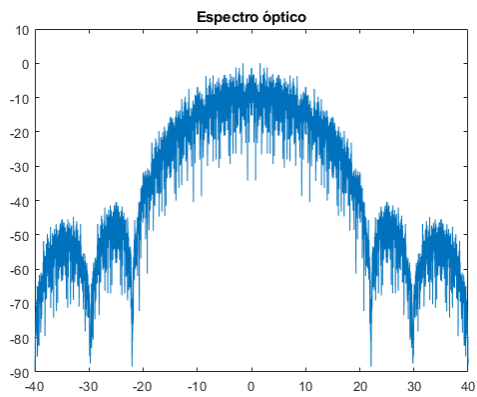
Transmisor - 64-QAM



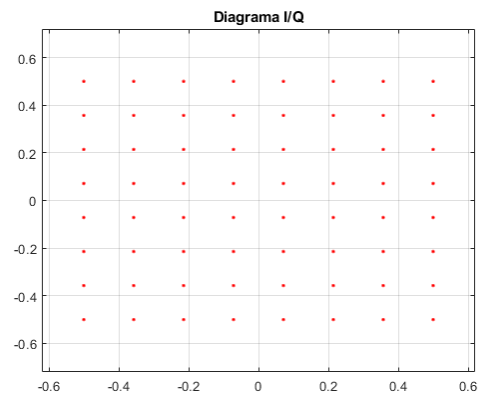
(a) Señales eléctricas



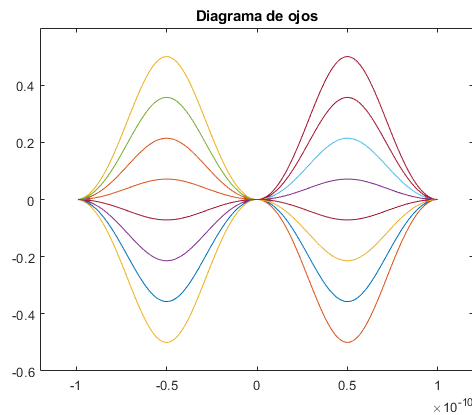
(b) Señales ópticas



(c) Espectro óptico



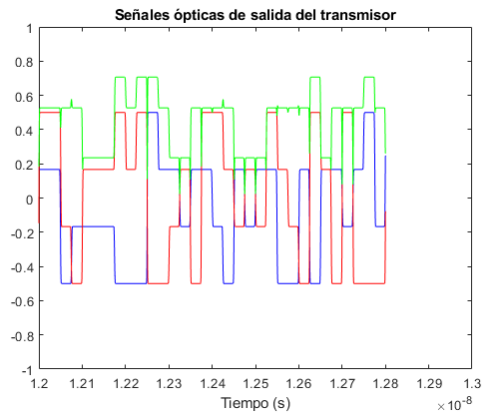
(d) Diagrama I/Q



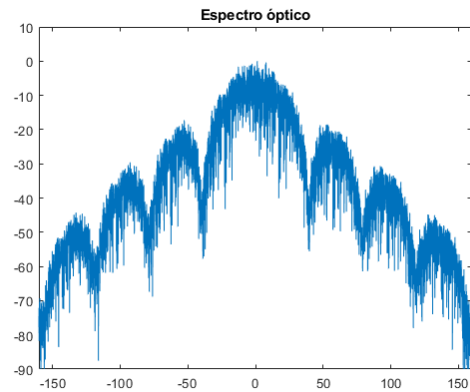
(e) Diagrama de ojos

Figura 5: Modulación 64-QAM

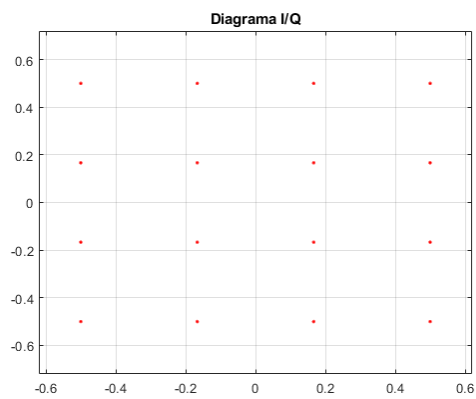
Transmisor - 16-QAM - Sin retorno a cero



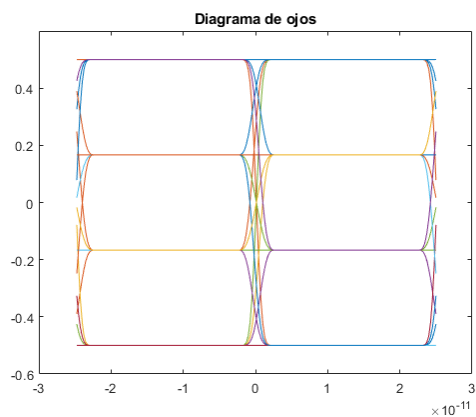
(a) Señales ópticas



(b) Espectro óptico



(c) Diagrama I/Q



(d) Diagrama de ojos

Figura 6: Modulación 16-QAM sin retorno a cero

En la codificación NRZ se puede observar como entre dos bits consecutivos del mismo valor no hay un retorno a 0, esta es la principal diferencia con las modulaciones anteriores. En el diagrama de ojos podemos ver cómo se mantiene constante el valor.

Receptor - OOK

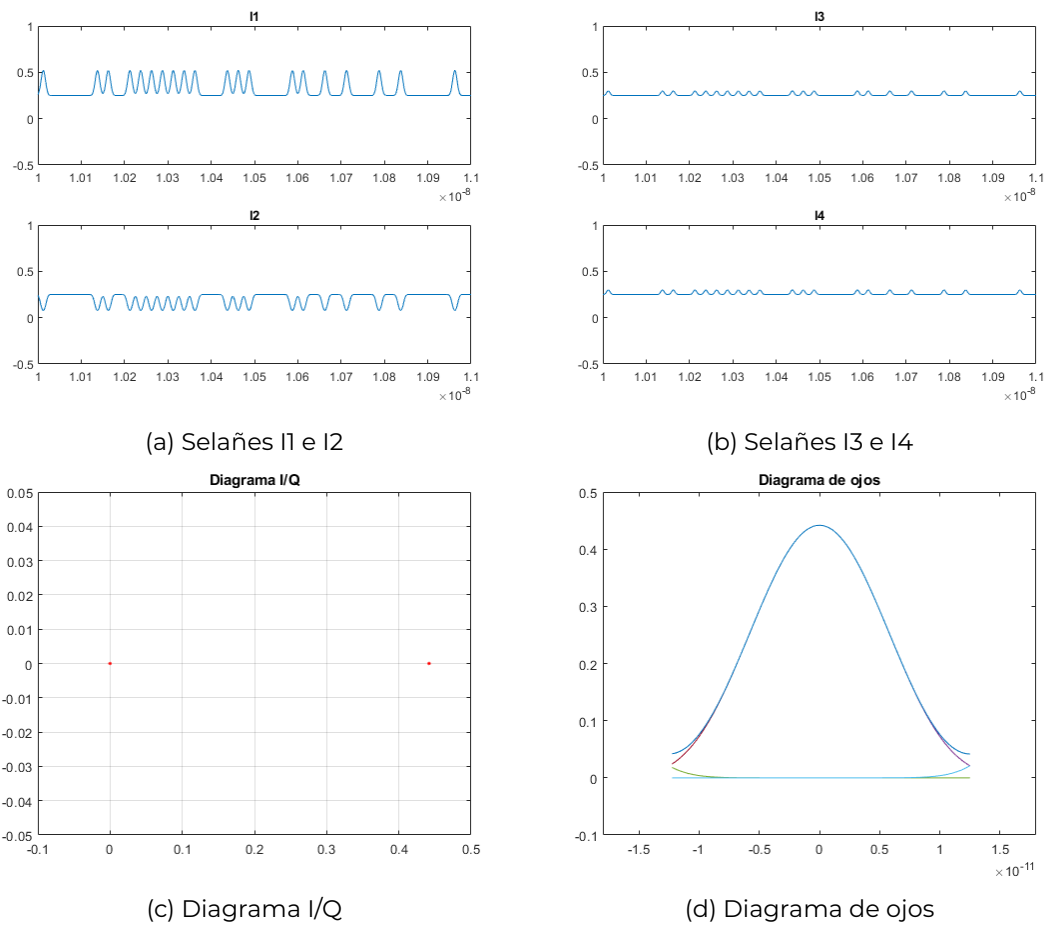
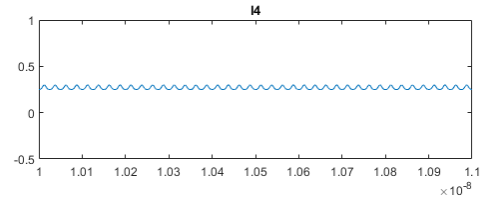
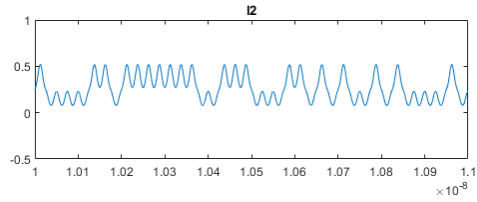
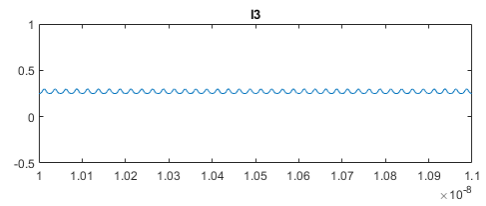
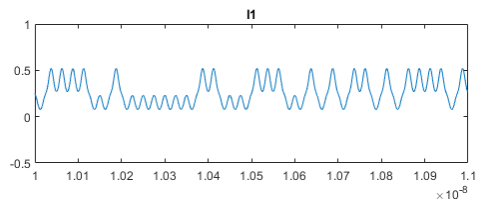


Figura 7: Receptor OOK

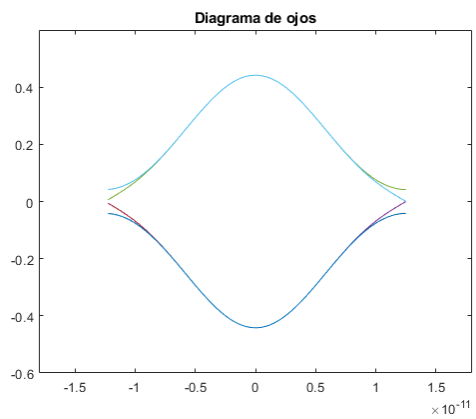
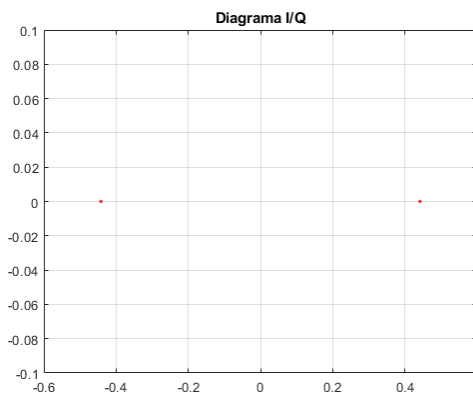
Al ser un receptor ideal (sin pérdidas en transmisión) se puede ver cómo se recupera la señal perfectamente. Esto pasará para el resto de modulaciones.

Receptor - BPSK



(a) Selaños I_1 e I_2

(b) Selaños I_3 e I_4

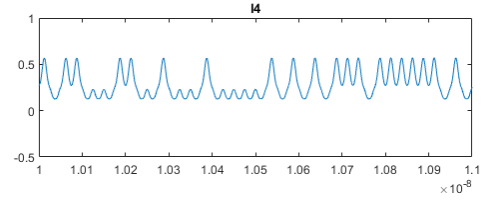
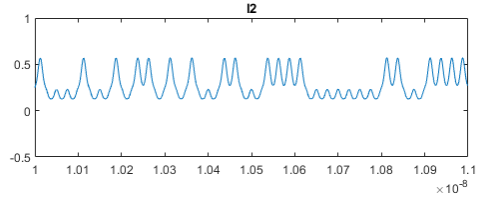
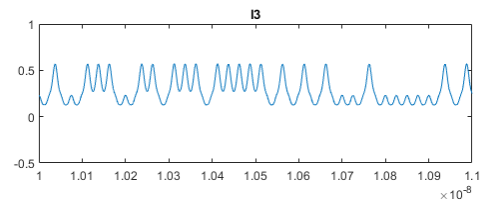
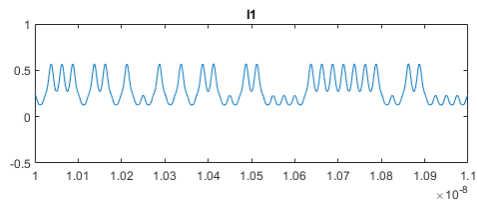


(c) Diagrama I/Q

(d) Diagrama de ojos

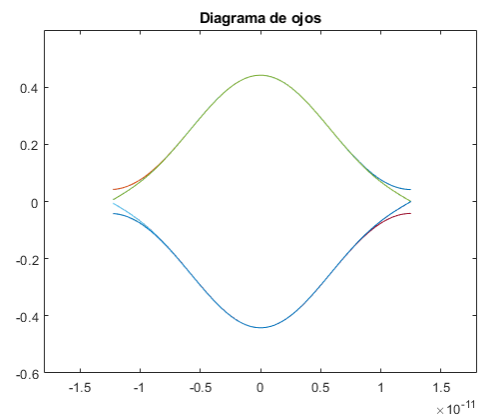
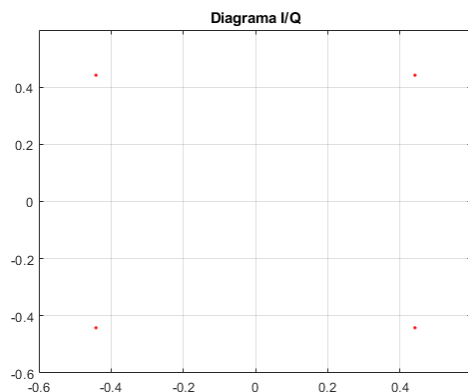
Figura 8: Receptor BPSK

Receptor - QPSK



(a) Selañes I1 e I2

(b) Selañes I3 e I4

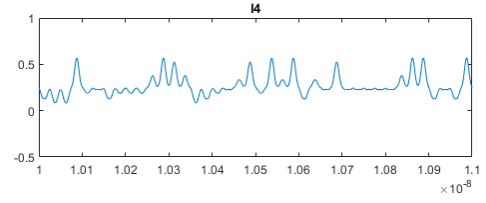
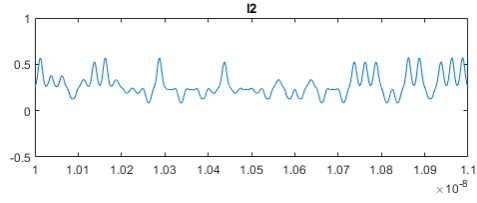
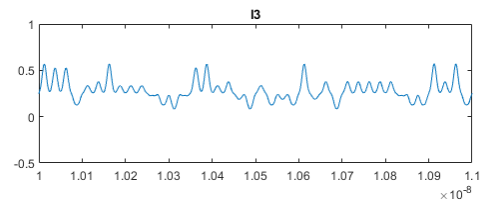
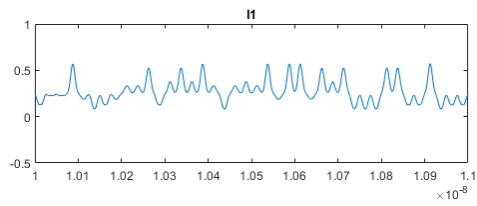


(c) Diagrama I/Q

(d) Diagrama de ojos

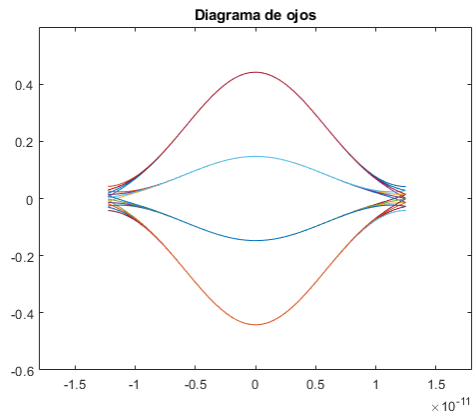
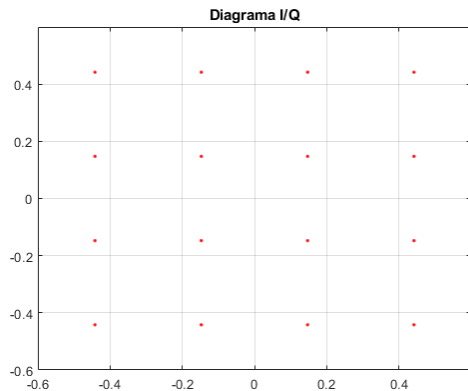
Figura 9: Receptor QPSK

Receptor - 16-QAM



(a) Selaños I_1 e I_2

(b) Selaños I_3 e I_4

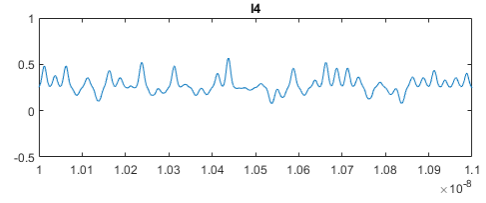
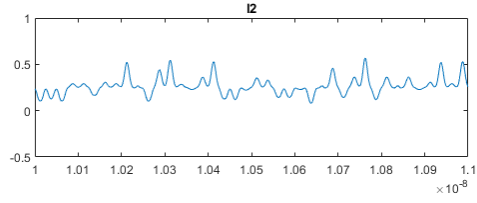
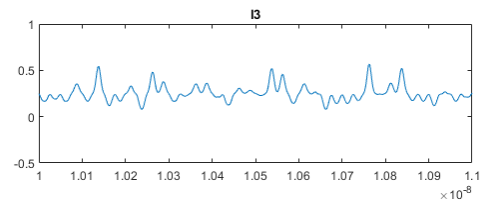
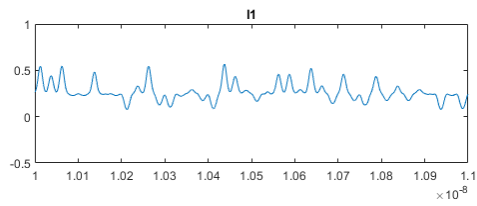


(c) Diagrama I/Q

(d) Diagrama de ojos

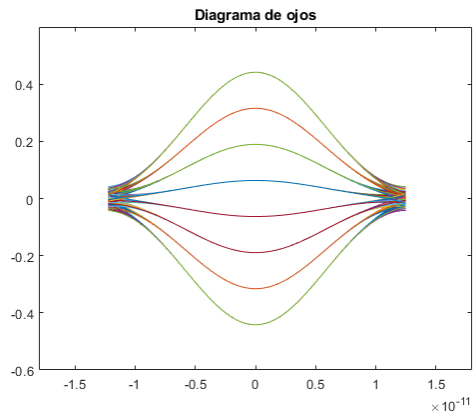
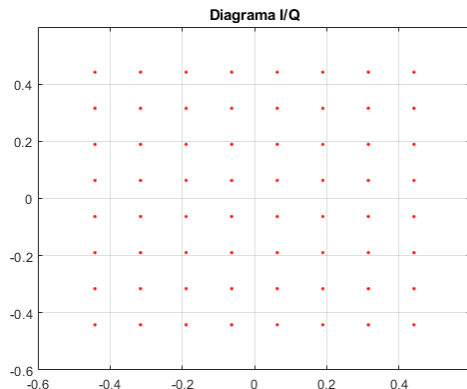
Figura 10: Receptor 16-QAM

Receptor - 64-QAM



(a) Selañes I1 e I2

(b) Selañes I3 e I4



(c) Diagrama I/Q

(d) Diagrama de ojos

Figura 11: Modulación OOK

Diferencias de fase

Las técnicas de detección coherente necesitan tener ajustado su oscilador local (en caso de que este se emplee) para poder demodular la señal correctamente. En los apartados anteriores la señal se recibía correctamente ya que el oscilador local estaba ajustado (Figura 12a).

Al desajustarlo $\pi/4$ (Figura 12b) y $\pi/10$ (Figura 12c), se puede apreciar como los valores de I y Q han rotado en los ejes.

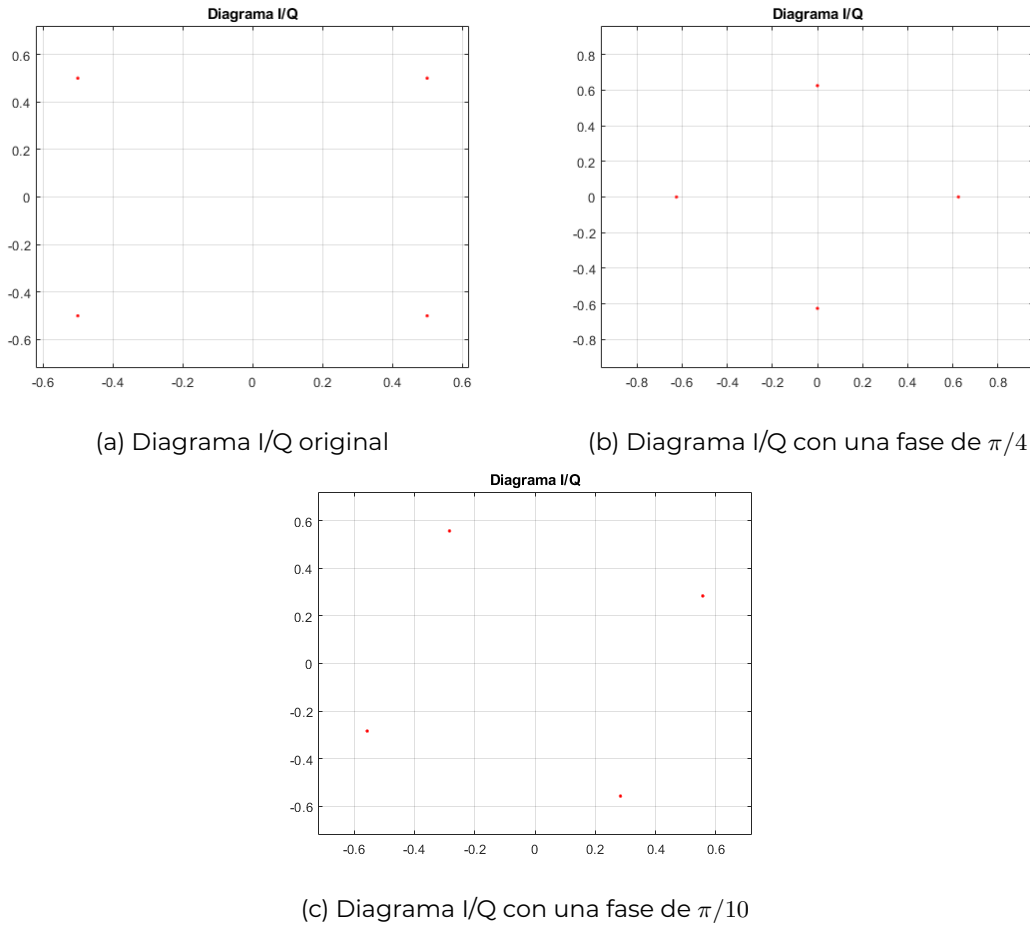


Figura 12: Diferencias de fase de $\pi/4$ y $\pi/10$

Dispersiones cromáticas

A continuación se estudiarán las dispersiones cromáticas dadas por la fórmula

$$\beta(w) = \beta_0 + \beta_1(w - w_0) + \frac{1}{2}\beta_2(w - w_0)^2 + \frac{1}{6}\beta_3(w - w_0)^3 \quad (1)$$

Donde:

$$\beta_2 = \frac{-D\lambda^2}{2\pi c} \cdot 10^{24} \quad (2)$$

$$\beta_3 = \frac{S\lambda^2}{2\pi c} \cdot 10^{36} \quad (3)$$

Dispersión cromática de primer orden. $D = 17$

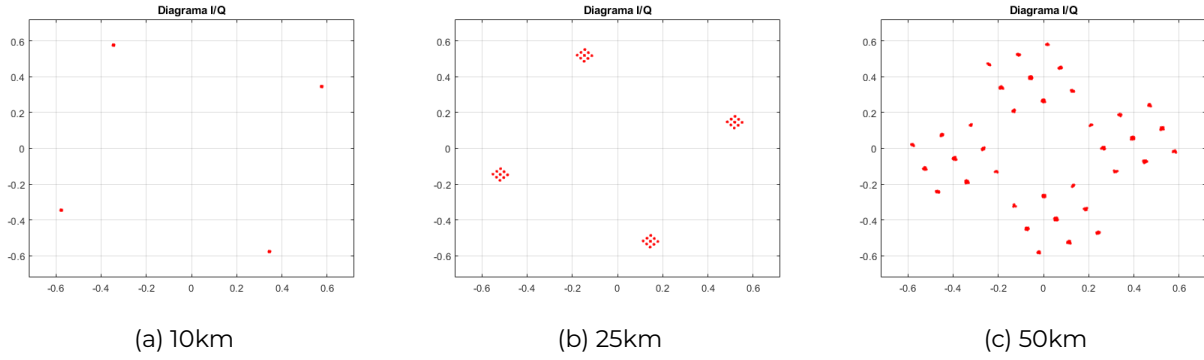


Figura 13: QPSK - Diagrama I/Q

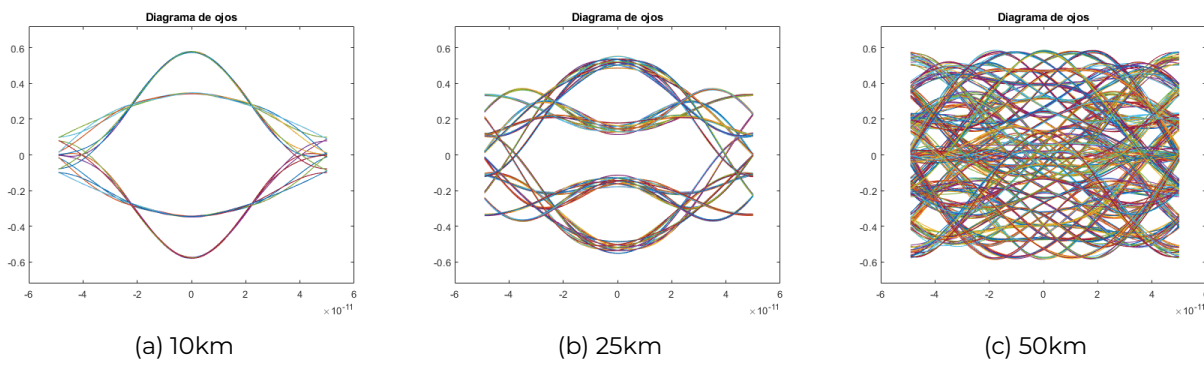


Figura 14: QPSK - Diagrama de ojos

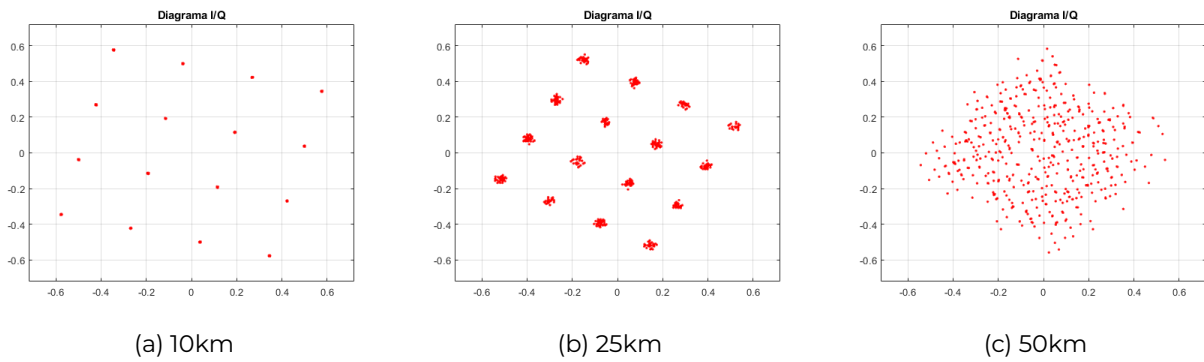


Figura 15: 16QAM - Diagrama I/Q

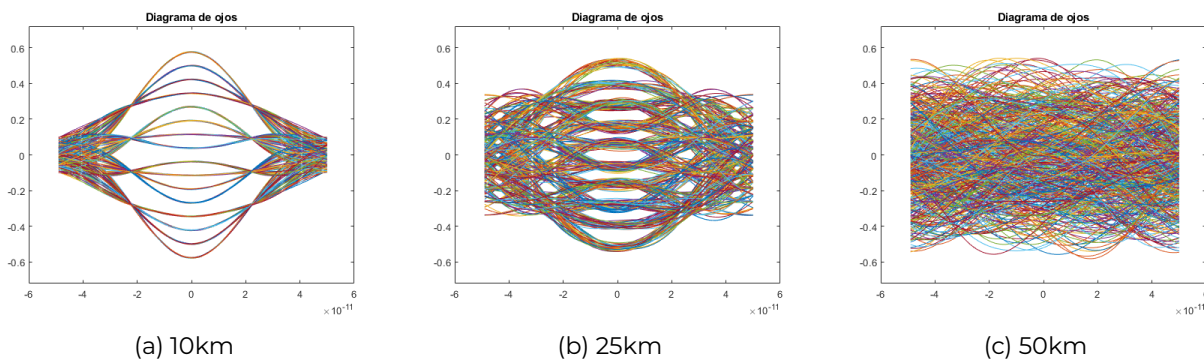
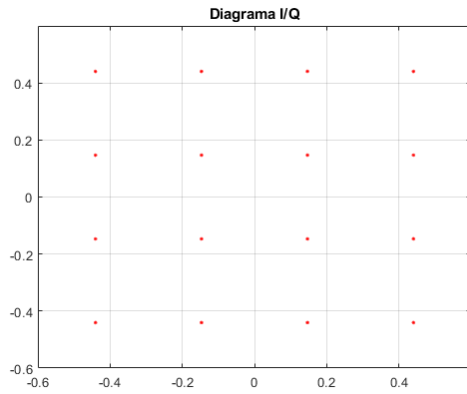
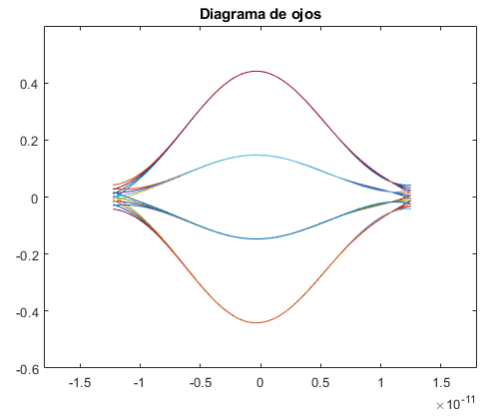


Figura 16: 16QAM - Diagrama de ojos

Dispersión cromática de segundo orden. $S = 0.08$

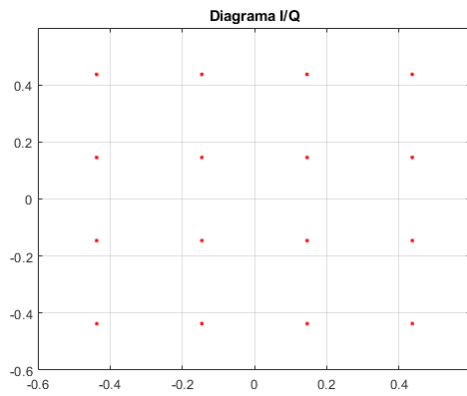


(a) Diagrama I/Q

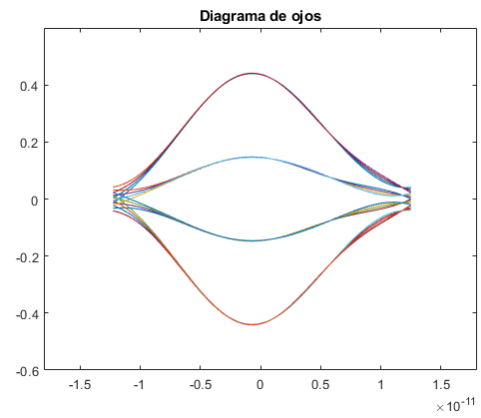


(b) Diagrama de ojos

Figura 17: 16QAM - a 200km

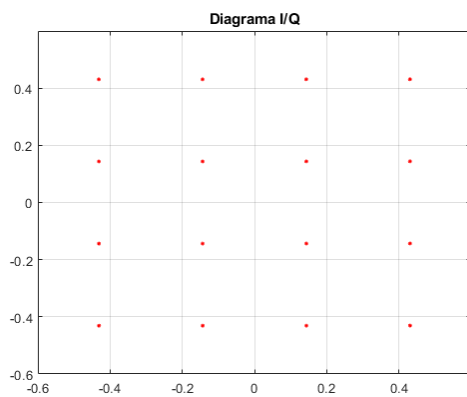


(a) Diagrama I/Q

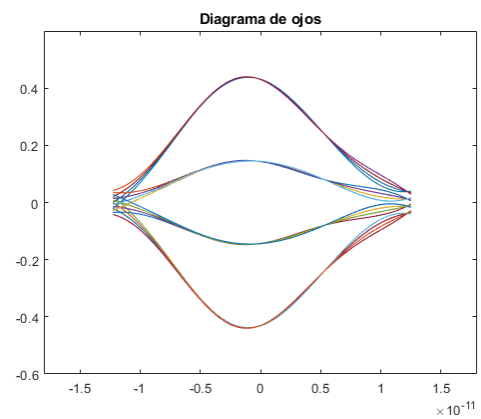


(b) Diagrama de ojos

Figura 18: 16QAM - a 400km



(a) Diagrama I/Q



(b) Diagrama de ojos

Figura 19: 16QAM - a 600km

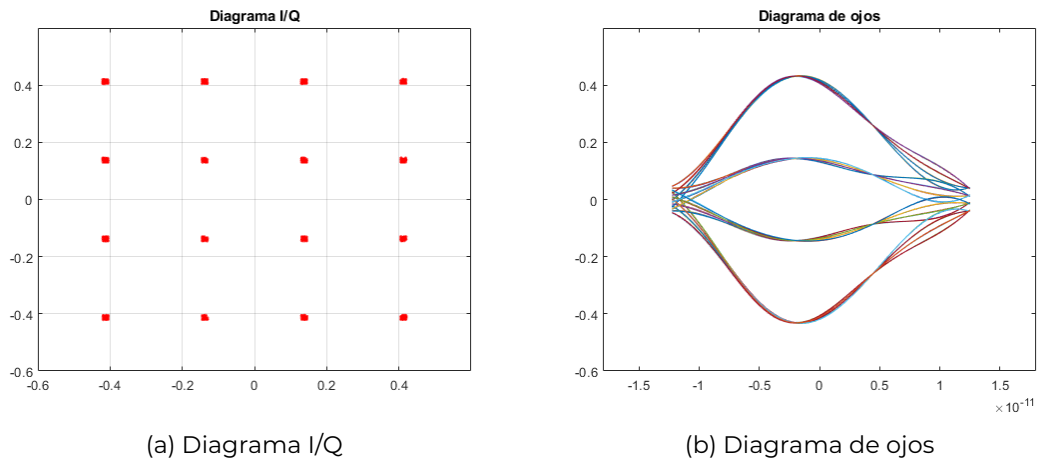


Figura 20: 16QAM - a 1000km

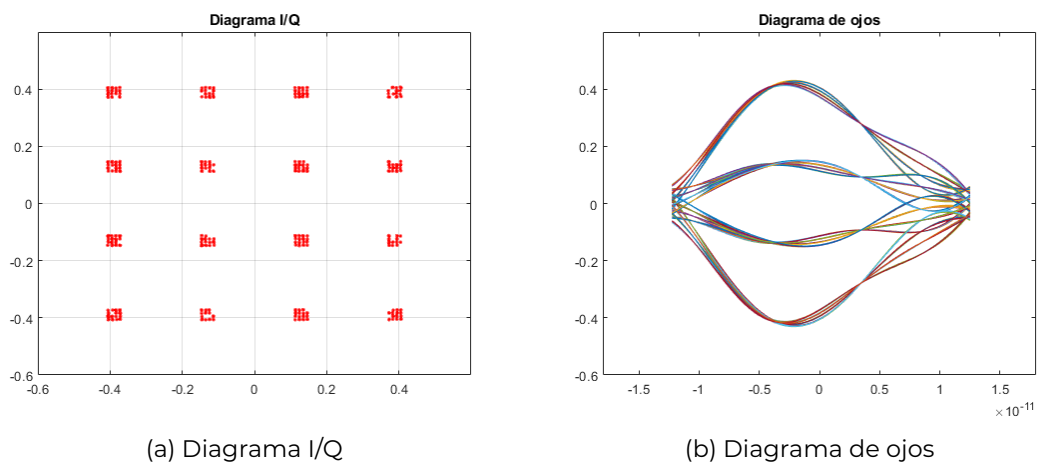


Figura 21: 16QAM - a 1500km

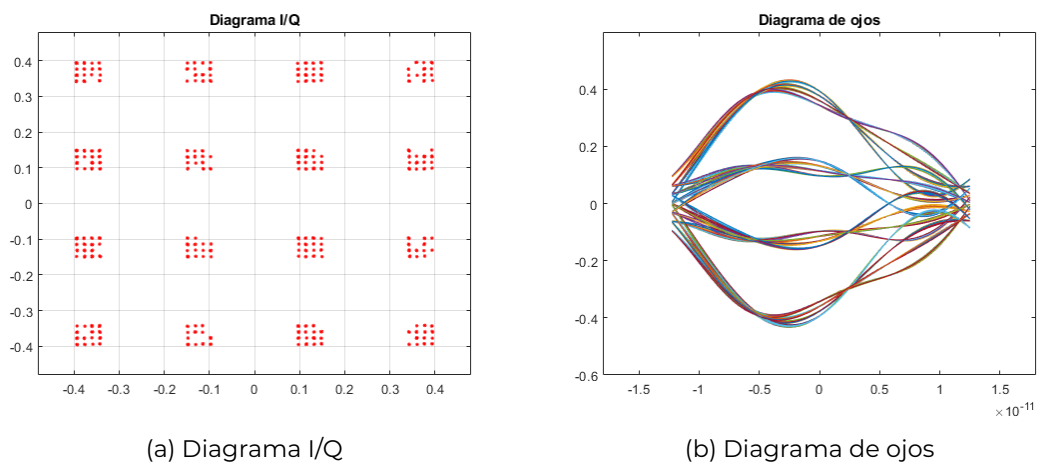


Figura 22: 16QAM - a 2000km

Como se puede ver, en las dispersiones cromáticas de primer orden (Figuras 13, 14, 15, 16) conforme se aumenta la distancia, se incrementa la dispersión en cada punto y las líneas del diagrama de ojos se empiezan a hacer "más densas" al aparecer nuevos puntos. Lo mismo pasa para las de segundo orden (Figuras 17, 18, 19, 20, 21, 22) pero a distancias más altas.

Ruido en los detectores

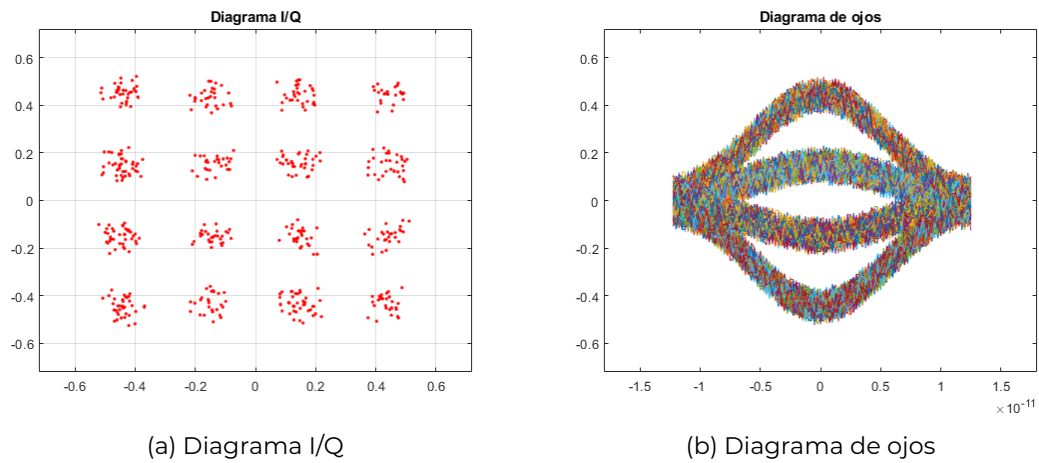


Figura 23: 16QAM - SNR 10

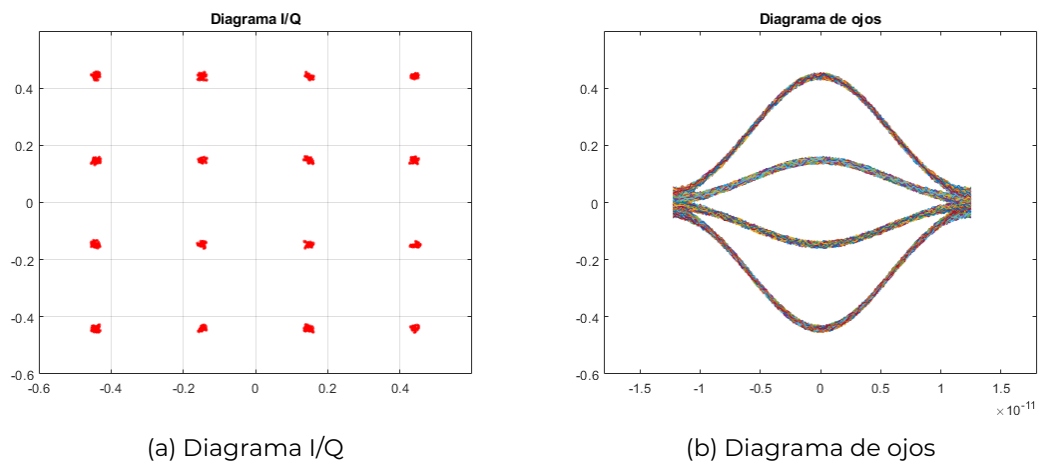


Figura 24: 16QAM - SNR 25

Como se puede observar, a menor SNR se obtiene un mayor ruido en el diagrama de ojos y aparecen puntos más dispersos en el diagrama I/Q.

Código Matlab empleado para realizar el transmisor

```
1 N_symbols=512;
2 N_bits=N_symbols*M;
3 data=idinput(N_bits,'PRBS',[0 1],[0 1]);
4 puntos_por_simbolo=100;
5 B=40e9;
6 Tb=1/B;
7 ts=Tb/puntos_por_simbolo;
8 puntos=puntos_por_simbolo* N_symbols;
9 t=(1:1:puntos)*ts;
10 RZ=1;
11 vI1 = [];
12 vQ1 = [];
13 signal=zeros(size(t));
14 Vpi = 5;
15 for n=1:1:N_symbols
16     simbol_index = bin2dec(num2str(data(M*(n-1)+1:1:M*(n-1)+M)));
17     valor_IQ = V(simbol_index+1);
18     vI1 = Vpi * acos(real(valor_IQ))/pi;
19     vQ1 = Vpi * acos(imag(valor_IQ))/pi;
20     vI2 = -vI1;
21     vQ2 = -vQ1;
22     vx(n,:)=[vI1 vI2 vQ1 vQ2];
23 end
24 e_signal_I1=[];
25 e_signal_I2=[];
26 e_signal_Q1=[];
27 e_signal_Q2=[];
28 unos=ones(size(1:1:puntos_por_simbolo));
29 for n=1:1:N_symbols
30     e_signal_I1=[e_signal_I1 unos*vx(n,1)];
31     e_signal_I2=[e_signal_I2 unos*vx(n,2)];
32     e_signal_Q1=[e_signal_Q1 unos*vx(n,3)];
33     e_signal_Q2=[e_signal_Q2 unos*vx(n,4)];
34 end
35 vent=10;
36 e_signal_I1=conv(e_signal_I1,1+cos(pi*(-vent:1:vent)/vent),'same');
37 e_signal_I1=e_signal_I1/sum(1+cos(pi*(-vent:1:vent)/vent));
38 e_signal_I2=conv(e_signal_I2,1+cos(pi*(-vent:1:vent)/vent),'same');
39 e_signal_I2=e_signal_I2/sum(1+cos(pi*(-vent:1:vent)/vent));
40 e_signal_Q1=conv(e_signal_Q1,1+cos(pi*(-vent:1:vent)/vent),'same');
41 e_signal_Q1=e_signal_Q1/sum(1+cos(pi*(-vent:1:vent)/vent));
42 e_signal_Q2=conv(e_signal_Q2,1+cos(pi*(-vent:1:vent)/vent),'same');
43 e_signal_Q2=e_signal_Q2/sum(1+cos(pi*(-vent:1:vent)/vent));
44 fi_I1 = pi*(e_signal_I1/Vpi);
45 fi_I2 = pi*(e_signal_I2/Vpi);
46 fi_Q1 = pi*(e_signal_Q1/Vpi);
47 fi_Q2 = pi*(e_signal_Q2/Vpi);
48 brazo_I = (1/2)*(exp(1i*fi_I1) + exp(1i*fi_I2));
49 brazo_Q = (1/2)*(exp(1i*fi_Q1) + exp(1i*fi_Q2));
50 carver=sin((pi/2)*(1+sin(pi*(t-Tb/2)/Tb)));
51 if (RZ == 1)
52     salida = (1/2)*(brazo_I+1i*brazo_Q).*carver;
53 else
54     salida = (1/2)*(brazo_I+1i*brazo_Q);
55 end
56
```

Script 1: Transmisor

Código Matlab empleado para modelar la fibra óptica

```
1 c = 3e5;
2 lambda = 1550;
3 D = 0*17;
4 S = 0*0.08;
5 L = 10;
6 beta2 = -D*(lambda^2/(2*pi*c))*1e-24;
7 beta3 = 1e-36*S*(lambda^2/(2*pi*c))^2;
8 p=length(t);
9 dw=2*pi*((( -p/2)+1:1:p/2)/p)*(1/ts));
10 FWHM = 100e9;
11 W_1_e = FWHM/1.665;
12 m = 1;
13 filtro = sqrt(exp(-(dw/(2*pi*W_1_e)).^(2*m)));
14 beta2L = beta2*L;
15 beta3L = beta3*L;
16 espectro_salida_fibra=fftshift(fft(salida)).*exp(1j*(1/2)*beta2L*dw.^2).*exp(1j*(1/6)*beta3L*dw.^3).*filtro;
17 tras_fibra = ifft(fftshift(espectro_salida_fibra));
```

Script 2: Fibra óptica

Código Matlab empleado para realizar el receptor

```
1 fi = 0*pi/10;
2 oscilador_local = exp(1i*fi)*ones(size(t));
3 df = 0*0.5e6;
4 L0 = oscilador_local.*exp(1i*2*pi*df*t);
5 Trx = tras_fibra;
6 b = L0;
7 i1 = abs((Trx + L0)/2).^2;
8 i2 = abs((Trx - L0)/2).^2;
9 i3 = abs((Trx + 1i*L0)/2).^2;
10 i4 = abs((Trx - 1i*L0)/2).^2;
11 i_I = i1-i2;
12 i_Q = i3-i4;
13 i_Total1 = i_I+1j*i_Q;
14
15 % Ruido
16 SNR=25;
17 SNRlin=10.^(SNR/10);
18 i_media=mean([mean(i1) mean(i2) mean(i3) mean(i4)]);
19 desv_ruido=sqrt((i_media.^2)/SNRlin);
20 rt1=(rand(size(t))-0.5*ones(size(t)))*desv_ruido;
21 rt2=(rand(size(t))-0.5*ones(size(t)))*desv_ruido;
22 rt3=(rand(size(t))-0.5*ones(size(t)))*desv_ruido;
23 rt4=(rand(size(t))-0.5*ones(size(t)))*desv_ruido;
24 i1=i1+rt1;
25 i2=i2+rt2;
26 i3=i3+rt3;
27 i4=i4+rt4;
28 i_I = i1-i2;
29 i_Q = i3-i4;
30 i_Total = i_I + 1j*i_Q;
```

Script 3: Receptor