Capítulo 5: Transmisión Digital Paso de Banda

1. Introducción

En el capítulo anterior, hemos considerado que la transmisión de la información digital se realiza a través de canales en banda base. En ese caso, la información digital se transmite directamente por el canal sin necesidad de usar una portadora sinusoidal.

Sin embargo la mayoría de las comunicaciones son a través de canales paso de banda, con lo cual, la única forma de transmitir señales en estos canales es realizando un traslado en frecuencia de la información de la señal a la frecuencia de la banda del canal.

En éste capítulo consideraremos tres tipos de señales moduladas con portadora, usadas para canales paso de banda: Modulación por desplazamiento de fase (PSK), modulación diferencial de fase(DPSK), modulación de amplitud de cuadratura (QAM), modulación por desplazamiento en frecuencia (FSK).

2. Modulación por Desplazamiento de Fase (PSK).

En una modulación por desplazamiento de fase la información que se transmite sobre el canal de comunicación se encuentra en la fase de la portadora. Puesto que el rango de la fase de la portadora es: $0 \le \phi \le 2\pi$, las fases de las portadoras usadas para transmitir la información digital a través de la modulación por desplazamiento de fase son: $\phi_i = 2\pi i / M$ para $i = 0,1,..,M-1$. Así, por ejemplo para una modulación binaria por desplazamiento de fase (BPSK)., las dos fases de la portadoras son: $\phi_0 = 0$ y $\phi_1 = \pi$ radianes. En general, en una modulación por desplazamiento de fase, las M señales transmitidas, pueden representarse de la forma:

$$S_i(t) = g(t)\cos\left(\omega_c t + \frac{2\pi i}{M}\right) \quad i = 0,1,..,M-1 \tag{5.2.1}$$

donde $g(t)$ es un pulso rectangular de amplitud A y duración T

Se observa que todas las señales PSK tienen la misma energía:

$$E_i = \int_0^T S_i^2(t)dt = \frac{1}{2}E_g \tag{5.2.2}$$

Estas señales pueden expresarse como combinación lineal de dos señales ortonormales, en la forma:

$$S_i(t) = S_{i1}\phi_1(t) + S_2\phi_2(t) \qquad i = 0,1,....,M-1 \tag{5.2.3}$$

con:

$$\phi_1(t) = \sqrt{\frac{2}{E_g}} g(t)\cos(\omega_c t)$$
$$\phi_2(t) = \sqrt{\frac{2}{E_g}} g(t)\sin(\omega_c t) \tag{5.2.4}$$

En este caso podemos expresar: $S_i(t) = \sqrt{\frac{2E_g}{T}} \cos\left(\omega_c t + \frac{2\pi i}{M}\right) \quad i = 0,1,..,M-1$, de esta ecuación se observa que la envolvente es constante y que es la fase de la portadora la que cambia bruscamente al principio de cada intervalo de señal.

La asignación de los k bits de información en las $M = 2^k$ posibles fases se pueden hacer de muchas maneras. La asignación preferible es usar codificación Gray, en donde las fases difieren en un digito binario, como se ilustra en la figura 5.2. Por lo tanto, solamente solo un único bit tendrá error en una secuencia de l-bits con codificación Gray cuando el ruido causa una selección errónea en una fase adyacente a la fase transmitida.

La señal recibida paso de banda tras pasar por el canal AWGN puede ser expresada como:

$$r(t) = S_i(t) + n(t) \tag{5.2.5}$$

a la salida de los dos correladores tendremos:

$$r = \left[S_{i1} + n_c \quad S_{i2} + n_s\right] = \left[\sqrt{E_s}\cos\frac{2\pi i}{M} + n_c \,, \sqrt{E_s}\cos\frac{2\pi i}{M} + n_s\right] \tag{5.2.6}$$

Donde n_c y n_s están definidas como:

$$n_c = \int_0^T n(t)\cos(\omega_c t)dt$$
$$n_s = \int_0^T n(t)\sin(\omega_c t)dt \tag{5.2.7}$$

Se puede demostrar que: n_c y n_s son variables aleatorias gaussianas correspondientes a las componentes en fase y en cuadratura del ruido cuyos valores esperados y varianzas son:

$$E(n_c) = E(n_s) = 0$$
$$E(n_c n_s) = 0 \tag{5.2.8}$$
$$E(n_c^2) = E(n_s^2) = \frac{N_0}{2}$$

El detector óptimo proyecta el vector de señal recibido en los M posibles vectores de señal $\{S_i\}$ y selecciona el vector correspondiente con mayor proyección. Así obtenemos las siguientes métricas:

$$C(r, S_i) = r \cdot S_i \qquad i = 0,1,..,M-1 \tag{5.2.9}$$

Debido a que todas las señales tienen la misma energía, una equivalente métrica que puede hacer el detector de una modulación en fase es calcular la fase del vector recibido $r = (r_1, r_2)$ como:

$$\phi_r = \tan^{-1}\frac{r_2}{r_1} \tag{5.2.10}$$

y se elegirá la señal cuya fase esté mas cercana a ϕ.

Se puede comprobar que la modulación de fase binaria es idéntica a una PAM binaria, la probabilidad de error es :

$$P_2 = Q\left(\frac{2E_b}{N_0}\right) \tag{5.2.11}$$

Para M=4, puede comprobarse tal y como se ve en teoría que:

$$P_4 = 2Q\left(\sqrt{\frac{2E_b}{N_0}}\right)\left[1-\frac{1}{2}Q\left(\sqrt{\frac{2E_b}{N_0}}\right)\right] \tag{5.2.12}$$

Para M>4, se puede hacer la siguiente aproximación:

$$\boxed{P_M \approx 2Q\left(\sqrt{\frac{2E_s}{N_0}}sen\frac{\pi}{M}\right) \approx 2Q\left(\sqrt{\frac{2kE_b}{N_0}}sen\frac{\pi}{M}\right)} \tag{5.2.13}$$

Cuando se usa código gray, dos símbolos correspondientes a dos fases de señal adyacentes difieren únicamente en un único bit. Por ello, los errores mas probables debido al ruido resulta en la elección errónea de una fase adyacente a la fase verdadero, así la mayoría de los símbolos erróneos tendrán un único bit en error. Con lo cual, la probabilidad de error de bit para una modulación M-aria por desplazamiento de fase se puede aproximar:

$$P_b \approx \frac{1}{k}P_M \tag{5.2.14}$$

Ejercicio 5.1.

Enunciado

Ejecutar la simulación de Monte-Carlo para un sistema comunicación M-PSK cuyo modelo de detector realiza las métricas de correlación vistas anteriormente. El modelo del sistema a simular se muestra en la figura 5.1.

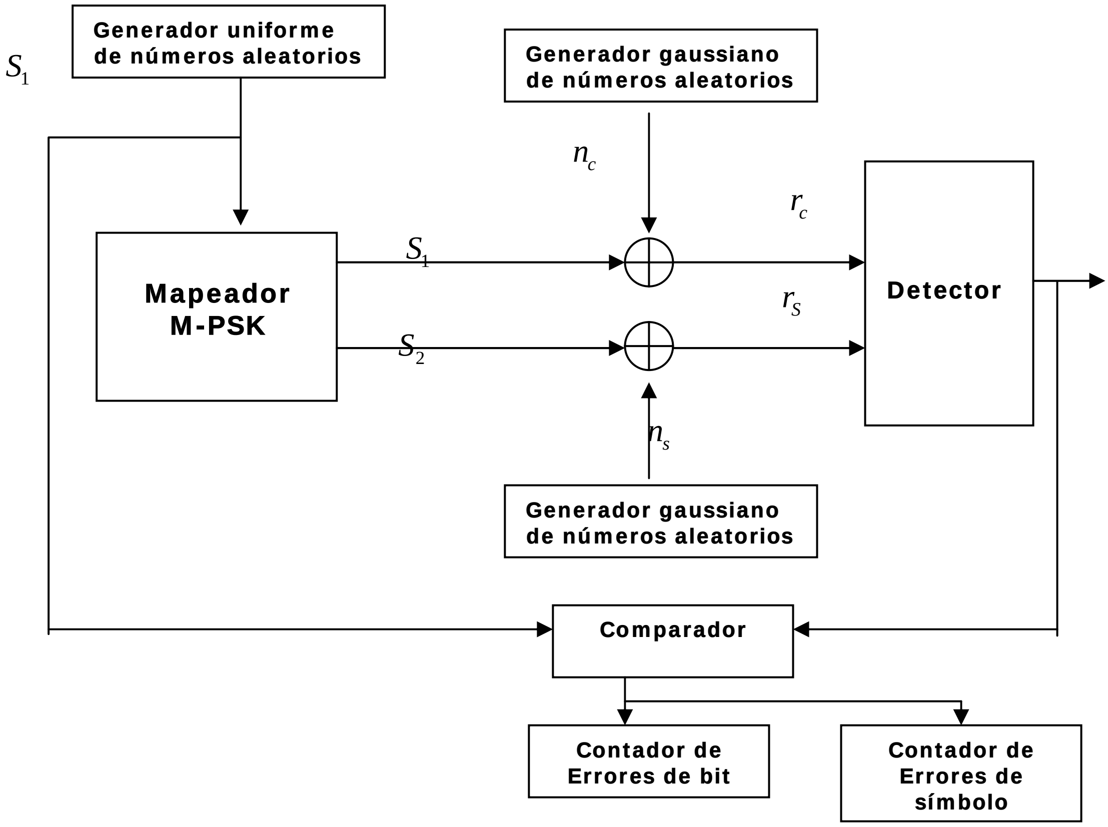

Figura 5.1. Diagrama en bloque de un sistema M-PSK para una simulación de Monte Carlo

Dibujar la probabilidad de Error de bit simulada y teórica, junto con la probabilidad de error de símbolo frente a la relación señal a ruido en decibelios ($10\log\frac{E_b}{N_0}$)

Solución

Tal y como se muestra en la figura, simulamos la generación de un vector aleatorio r*, el cual es la salida del correlador de señal y la entrada al detector. Empezamos generando un símbolo aleatorio (entre M posibles), este símbolo se corresponderá con una secuencia de* $K = \log_2(M)$ *dígitos binarios con codificación gray, a la que* *llamaremos fuente. El Mapeador M-PSK generará todas las componentes en fase y cuadratura:*

$$S_{i,1} = \cos\left(\frac{2*(i-1)*\pi}{M}\right) \qquad i = 1,..,M$$
$$S_{i,2} = sen\left(\frac{2*(i-1)*\pi}{M}\right)$$

y la enviada :
$$M_1 = \cos\left(\frac{2*simbolo*\pi}{M}\right)$$
$$M_2 = sen\left(\frac{2*simbolo*\pi}{M}\right)$$

A estas componentes le sumamos el ruido aleatorio gaussiano de media 0 y varianza σ^2*.*

$$r_1 = s_1 + n_1$$
$$r_2 = s_2 + n_2$$

El detector calculará la proyección (producto escalar) de r *con todos los posibles vectores de señal, nos quedaremos con aquel s cuyo producto escalar sea mayor. Este* s *lo volvemos a transformar en una secuencia binaria de* $K = \log_2(M)$ *dígitos a la que llamaremos decisión, compararemos decisión con fuente y calcularemos la probabilidad de error de bit y la probabilidad de error de símbolo.*

```
function [pb,ps]=psk(snr_db,M)
echo on;
E=1;
SNR=10^(snr_db/10);
R=log2(M);
sigma=sqrt(E/(2*SNR*R));
N=10000;
%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:M
    t=(2*(i-1)*pi)/M;
    s(i,1)=feval('cos',t);

    s(i,2)=feval('sin',t);
end;

%%%%%FUENTE%%%%%%%%%%%%%%%%%%%%%%%%%%%%
grayencod = bitxor ( 0: M-1, floor((0:M-1)/2));
codea=de2bi(grayencod,R,'left-msb');
[fila,columna]=size(codea);

for i=1:N
    aux=rand(1,1);
    simbolo(i)=floor(M*aux);
    for j=1:columna
        fuente(i,j)=codea((simbolo(i)+1),j);
    end;
    mapeo(i,1)=s(simbolo(i)+1,1);
    mapeo(i,2)=s(simbolo(i)+1,2);
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
error_simbolo=0;
error=0;
for i=1:N
    n(1)=sigma*randn(1,1);
    n(2)=sigma*randn(1,1);
    for j=1:2
        r(j)=mapeo(i,j)+n(j);
    end;
    for j=1:M
        c(j)=dot(r,s(j,:));
    end;
    max_c=max(c(:));
    posicion=strfind(c,max_c);
    for j=1:columna
        decision(1,j)=codea(posicion,j);
    end;
    if(isequal(fuente(i,:),decision))
        error_simbolo=error_simbolo;
    else
        error_simbolo=error_simbolo+1;
    end;
    for j=1:columna
        if(fuente(i,j)~=decision(j))
            error=error+1;
        end;
    end;
end;
ps=error_simbolo/N;
```

```
echo on;

snr_db1=0:2:12;
snr_db2=0:0.1:12;
M_16=16;
M_8=8;
M_4=4;
for i=1:length(snr_db1)
    [simuladab_4(i),simuladas_4(i)]=psk(snr_db1(i),M_4);
    [simuladab_8(i),simuladas_8(i)]=psk(snr_db1(i),M_8);
    [simuladab_16(i),simuladas_16(i)]=psk(snr_db1(i),M_16);
end;

for i=1:length(snr_db2)
    SNR=10^(snr_db2(i)/10);
    K_4=log2(M_4);
    K_8=log2(M_8);
    K_16=log2(M_16);
    teoricas_4(i)=2*Q(sqrt(2*SNR));
    teoricas_8(i)=(2)*Q(sqrt(2*K_8*SNR)*sin(pi/M_8));
    teoricas_16(i)=(2)*Q(sqrt(2*K_16*SNR)*sin(pi/M_16));
    teorica_4(i)=teoricas_4(i)/K_4;
    teorica_8(i)=teoricas_8(i)/K_8;
    teorica_16(i)=teoricas_16(i)/K_8;
end;

figure(1);
semilogy(snr_db1,simuladab_4,'g*',snr_db2,teorica_4,'g');
hold on;
semilogy(snr_db1,simuladab_8,'r*',snr_db2,teorica_8,'r');
semilogy(snr_db1,simuladab_16,'b*',snr_db2,teorica_16,'b');
xlabel('10log(Eb/No)');
ylabel('Pe');
legend('M=4','','M=8','','M=16','');
grid on;
hold off;
figure(2);
semilogy(snr_db1,simuladas_4,'g*',snr_db2,teoricas_4,'g');
hold on;
semilogy(snr_db1,simuladas_8,'r*',snr_db2,teoricas_8,'r');
semilogy(snr_db1,simuladas_16,'b*',snr_db2,teoricas_16,'b');
xlabel('10log(Eb/No)');
ylabel('Pm');
legend('M=4','','M=8','','M=16','');

hold off;
echo off;
```

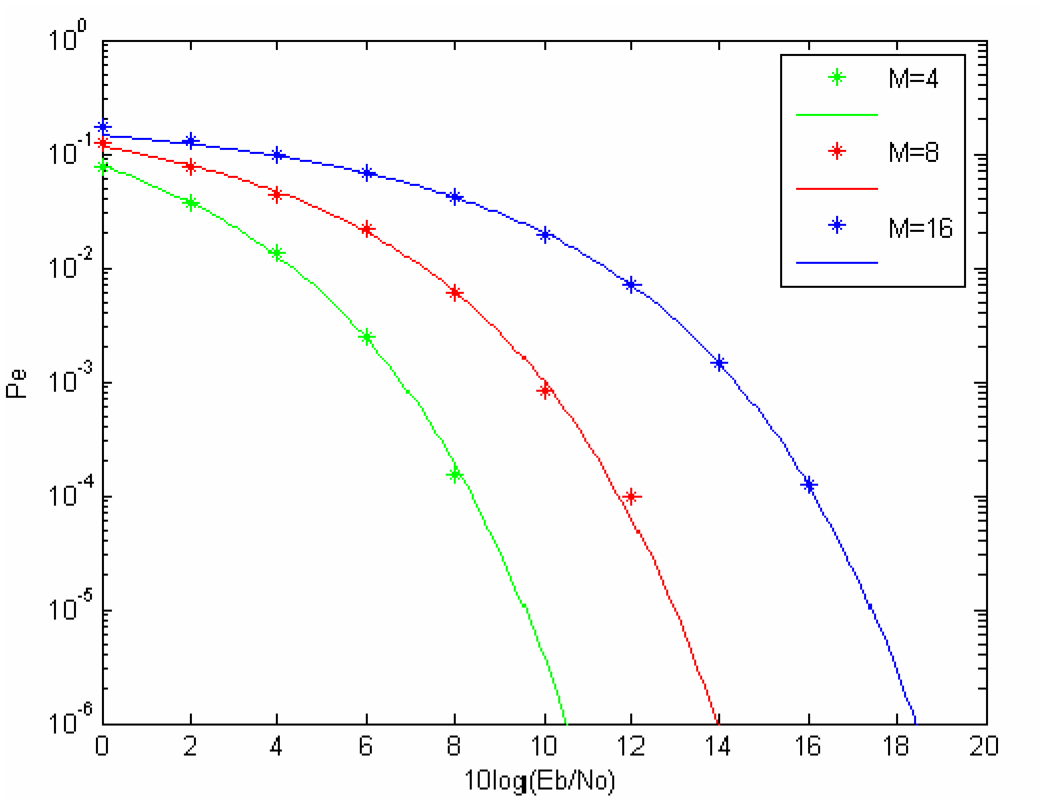

Figura 5.2. Probabilidad de Error de bit vs SNR_bit con simulación Monte carlo comparada con la Probabilidad de error teórica utilizando sistema M-ario PSKM=16, M=8 y M=4

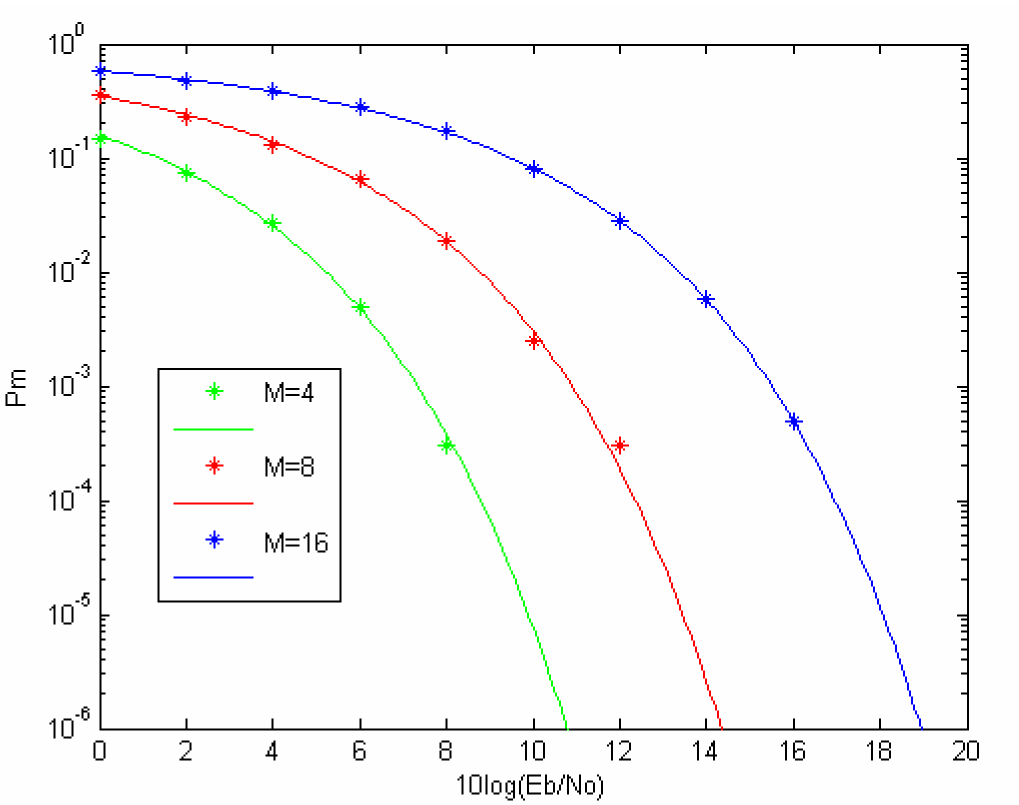

Figura 5.3. Probabilidad de Error de símbolo vs SNR_bit con simulación Monte carlo comparada con la Probabilidad de error teórica utilizando sistema M-ario PSKM=16, M=8 y M=4

3. Modulación Diferencial de Fase(DPSK)

En la demodulación de la señal modulada en fase, descrita anteriormente hemos supuesto que el modulador puede hacer una estimación perfecta de la fase de la señal recibida. En general, esto implica que el receptor debe estimar el offset de la componente en fase de la señal recibida debido al retraso en la transmisión a través del canal. La estimación del offset de la fase normalmente se hace usando un PLL. Y así, llegamos a una demodulación coherente.

Otro tipo de modulación de Fase es la modulación diferencial de fase, en la cual se codifica la información del símbolo que se desea transmitir en la diferencia de fase de la señal transmitida entre dos símbolos consecutivos. Así la señal transmitida viene dada por:

$$s_n(t) = g(t)\cos(\omega_c t + \phi_n) \qquad nT \le t \le (n+1)T \tag{5.3.1}$$

la fase ϕ_n dependerá de la fase transmitida en el intervalo anterior y de un incremento de fase $\Delta\phi_n$

$$\phi_n = \phi_{n-1} + \Delta\phi_n \tag{5.3.2}$$

Por ejemplo en una modulación binaria, el bit de información 1 puede ser transmitido con un desplazamiento de fase de 180º con respecto a la señal del intervalo anterior y para el bit de información 0 puede ser transmitido con un desplazamiento de fase 0 con respecto a la señal del intervalo anterior. En una transmisión M-aria con M=4, incrementos de fase entre los sucesivos intervalos son 0º, 90º, 180º y 270º correspondientes a los bits de información 00, 01, 11, y 10 respectivamente. La generalización para la codificación diferencial para M>4 es directa. La ejecución de la codificación se ejecuta con un circuito lógico simple que se sitúa antes del modulador.

La demodulación y la detección de la señal modulada en fase diferencial puede ejecutarse como sigue. Se calcula la fase de la señal recibida $\phi_r = \tan^{-1} r_2/r_1$ en el detector y se compara con las M posibles fases transmitidas $\{\phi_m\}$, se escogerá aquella ϕ_m que esté más cerca a ϕ_r. Así el detector compararía la fase de la señal detectada sobre dos intervalos consecutivos para extraer la información transmitida.

Se observa que la demodulación de la señal modulada en fase diferencial no requiere de la estimación de la fase de la portadora. Para comprobarlo, suponemos que demodulamos la señal diferencial codificada multiplicando $r(t)$ con $g(t)\cos(\omega_c t)$ y con $-g(t)\sin(\omega_c t)$. El k-intervalo de señal, las dos componentes a la salida del demodulador se pueden expresar de manera compleja como:

$$r_k = \sqrt{E_s}\, e^{j(\phi_k - \phi)} + n_k \tag{5.3.3}$$

donde ϕ_k es la fase de la señal transmitida en el intervalo k, ϕ es la fase de la portadora y $n_k = n_{kc} + n_{ks}$ es el ruido. De forma similar, el vector de señal recibido a la salida del demodulador en el intervalo anterior es el siguiente valor complejo:

$$r_{k-1} = \sqrt{E_s}\, e^{j(\phi_{k-1}-\phi)} + n_{k-1} \tag{5.3.4}$$

La variable de decisión en el detector es la diferencia de fase de estos dos valores complejos. También, se puede realizar la proyección de r_k en r_{k-1} y usar la fase del complejo resultante. Así:

$$r_k r_{k-1}^* = E_s e^{j(\phi_k-\phi_{k-1})} + \sqrt{E_s}\, e^{j(\phi_k-\phi)} n_{k-1} + \sqrt{E_s}\, e^{j(\phi_{k-1}-\phi)} n_k + n_k n_{k-1}^* \tag{5.3.5}$$

en donde, en ausencia de ruido, la diferencia de fase es $\phi_k - \phi_{k-1}$. Esto implica que la media de $r_k r_{k-1}^*$ es independiente de la fase de la portadora.

Se puede demostrar que la probabilidad de error de símbolo, usando la función de Marcum se puede aproximar por:

$$\boxed{P_m \le 1 + Q_1\left(\sqrt{\frac{E_s}{N_o}\left(1-\sin\frac{\pi}{M}\right)}, \sqrt{\frac{E_s}{N_o}\left(1+\sin\frac{\pi}{M}\right)}\right) - Q_1\left(\sqrt{\frac{E_s}{N_o}\left(1+\sin\frac{\pi}{M}\right)}, \sqrt{\frac{E_s}{N_o}\left(1-\sin\frac{\pi}{M}\right)}\right)} \tag{5.3.6}$$

donde $Q_1(x)$ es la función de Marcum definida más adelante.
Se pueden realizar las siguientes aproximaciones:

$$M = 2 \qquad P_2 = \frac{1}{2} e^{-E_b/N_o} \tag{5.3.7}$$

$$M \uparrow\uparrow\uparrow \qquad P_M \approx 2Q\left(\sqrt{\frac{E_s}{N_o}}\sin\frac{\pi}{M}\right) \tag{5.3.8}$$

La función Marcum

Definiremos el primer orden de la función Marcum como :

$$Q_1(a,b) = \int_b^{+\infty} x e^{-\frac{x^2+a^2}{2}} I_0(ax)\,dx \tag{5.3.9}$$

donde *Io* es la función modificada de Bessel de tipo 1 y orden cero, , tenemos dos casos particulares

$$Q_1(0,b) = e^{-\frac{b^2}{2}} \qquad Q_1(a,0) = 1 \tag{5.3.10}$$

Cuando $b >> 1$ y $b >> b-a$ se puede hacer la siguiente aproximación:

$$Q_1(a,b) \approx Q(b-a) \tag{5.3.11}$$

Una aproximación útil cuando $b >> 1,\ a >> 1,\ b >> b-a > 0$, es la siguiente:

$$1 + Q_1(a,b) - Q_1(b,a) \approx 2Q(b-a) \tag{5.3.12}$$

Ejercicio 5.2.

Enunciado

Ejecutar la simulación de Monte-Carlo para un sistema comunicación M-DPSK. El modelo del sistema a simular se muestra en la figura 5.4.

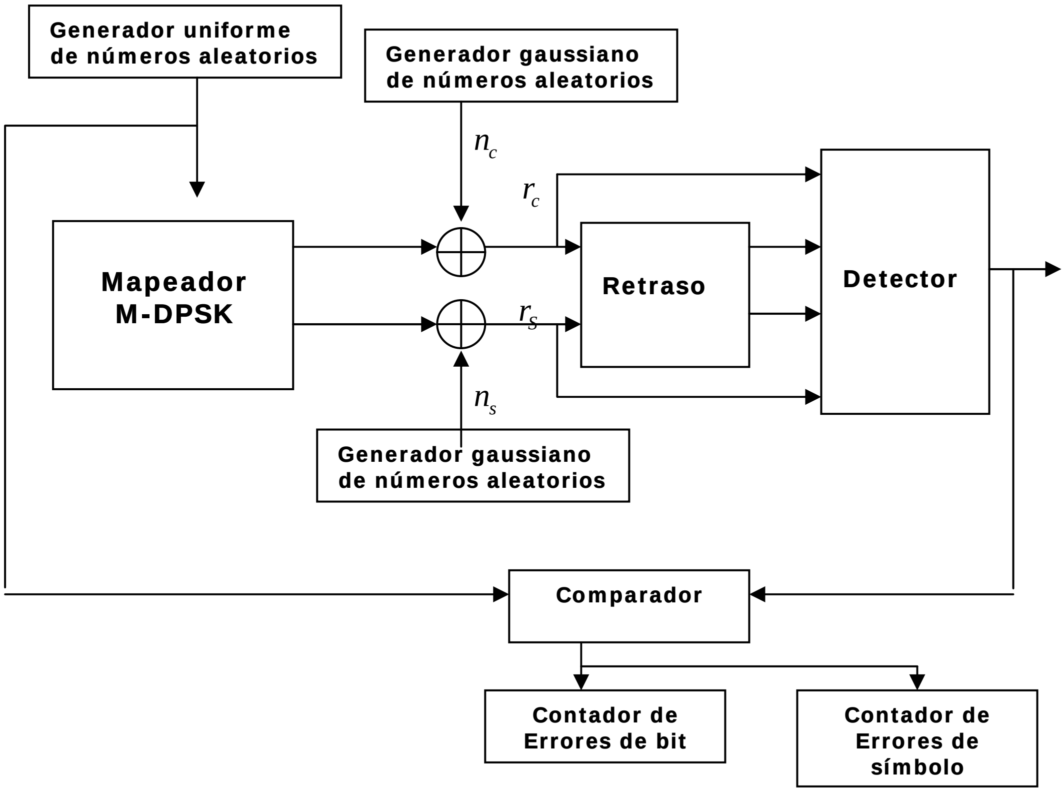

Figura 5.4. Diagrama en bloque de un sistema M-DPSK para una simulación de Monte Carlo

Para implementar el Codificador diferencial se proporciona la función cod_dpsk.m

a) Dibujar la probabilidad de Error de bit simulada y teórica, junto con la probabilidad de error de símbolo frente a la relación señal a ruido en decibelios ($10\log\frac{E_b}{N_0}$*)*

b) Comparar las dos modulaciones en fase (PSK, DPSK)

Resultado

a)

```
% script en Matlab para el problema 5_2 --Apartado a----
echo on

tolerance=1e-7;
plus_inf=20;
snrdb1=0:2:12;
snrdb2=0:0.1:12;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Tasas de error Simuladas
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:length(snrdb1),
  [simuladas_2(i)]=dpsk(snrdb1(i),2);
  [simuladas_4(i)]=dpsk(snrdb1(i),4);
  [simuladas_8(i)]=dpsk(snrdb1(i),8);
  echo off ;
end;
echo on ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Tasas de error Teóricas
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:length(snrdb2),
  SNR=exp(snrdb2(i)*log(10)/10);

    a1_8=sqrt(SNR*(log2(8))*(1-sin(pi/8)));
    b1_8=sqrt(SNR*(log2(8))*(1+sin(pi/8)));

    a1_4=sqrt(SNR*(log2(4))*(1-sin(pi/4)));
    b1_4=sqrt(SNR*(log2(4))*(1+sin(pi/4)));


teoricam_8(i)=1+quadl('dpsk_int',b1_8,plus_inf,tolerance,[],a1_
8)-quadl('dpsk_int',a1_8,plus_inf,tolerance,[],b1_8);

teoricam_4(i)=1+quadl('dpsk_int',b1_4,plus_inf,tolerance,[],a1_
4)-quadl('dpsk_int',a1_4,plus_inf,tolerance,[],b1_4);
    teoricam_2(i)=(1/2)*exp(-SNR);
    echo off ;
end;
echo on ;

semilogy(snrdb1,simuladas_2,'g*',snrdb2,teoricam_2,'g');
hold on;
semilogy(snrdb1,simuladas_4,'r*',snrdb2,teoricam_4,'r');
semilogy(snrdb1,simuladas_8,'b*',snrdb2,teoricam_8,'b');
legend('M=2','','M=4','','M=8');
xlabel('10log(Eb/No)');
ylabel('Pm')
hold off;
```

```
function [p]=dpsk(snr_in_dB,M)
N=10000;
E=1;                          % energia por simbolo
snr=10^(snr_in_dB/10);              % razon señal a ruido
R=log2(M);
sgma=sqrt(E/(2*R*snr));             % varianza del ruido
%%%%%%%FUENTE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
for i=1:R*N,
  temp=rand;
  if (temp<0.5),
    dsource(i)=0;
  else
    dsource(i)=1;
  end;
end;

m=1;
for i=1:N
    for l=1:R
     fuente(i,l)=dsource(m+(l-1));
    end;
    m=m+R;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%Codificación de la señal diferencial
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
if(M~=8)
    mapping= bitxor ( 0: M-1, floor((0:M-1)/2));
elseif(M==8)
    mapping=[0 1 3 2 7 6 4 5];
end;

[diff_enc_output] = cod_dpsk(E,M,mapping,dsource);
for i=1:N,
  n(1)=sgma*randn(1,1);
  n(2)=sgma*randn(1,1);
  r(i,:)=diff_enc_output(i,:)+n;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
% Regiones de Decisión y posibles Decisiones
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
xinicial=(pi/(M));
for i=1:M
    x(i)=xinicial+((i-1)*(2*pi/M));
end;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Detección y cálculo de la probabilidad de error
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
error=0;
prev_theta=0;
for i=1:N,
  theta=angle(r(i,1)+j*r(i,2));
  delta_theta=mod(theta-prev_theta,2*pi);
  for k=1:M-1
    if(delta_theta<x(1) | delta_theta>x(M))
        decis=d(1,:);
    elseif((delta_theta<x(k+1)) & (delta_theta>x(k)))
        decis=d(k+1,:);
    end;
  end;
  prev_theta=theta;
  if (isequal(decis,fuente(i,:)))
      error=error;
  else
      error=error+1;
  end;
end;
p=error/N;
```

```
function [enc_comp] = cod_dpsk(E,M,mapping,sequence);

k=log2(M);
N=length(sequence);
% If N is not divisible by k, append zeros, so that it is...
remainder=rem(N,k);
if (remainder~=0),
  for i=N+1:N+k-remainder,
    sequence(i)=0;
  end;
  N=N+k-remainder;
end;
theta=0;                      % Initially, assume that theta=0.

for i=1:k:N,
  index=0;
  for j=i:i+k-1,
    index=2*index+sequence(j);
  end;
  index=index+1;
  theta=mod(2*pi*mapping(index)/M+theta,2*pi);
  enc_comp((i+k-1)/k,1)=sqrt(E)*cos(theta);
  enc_comp((i+k-1)/k,2)=sqrt(E)*sin(theta);
end;
```

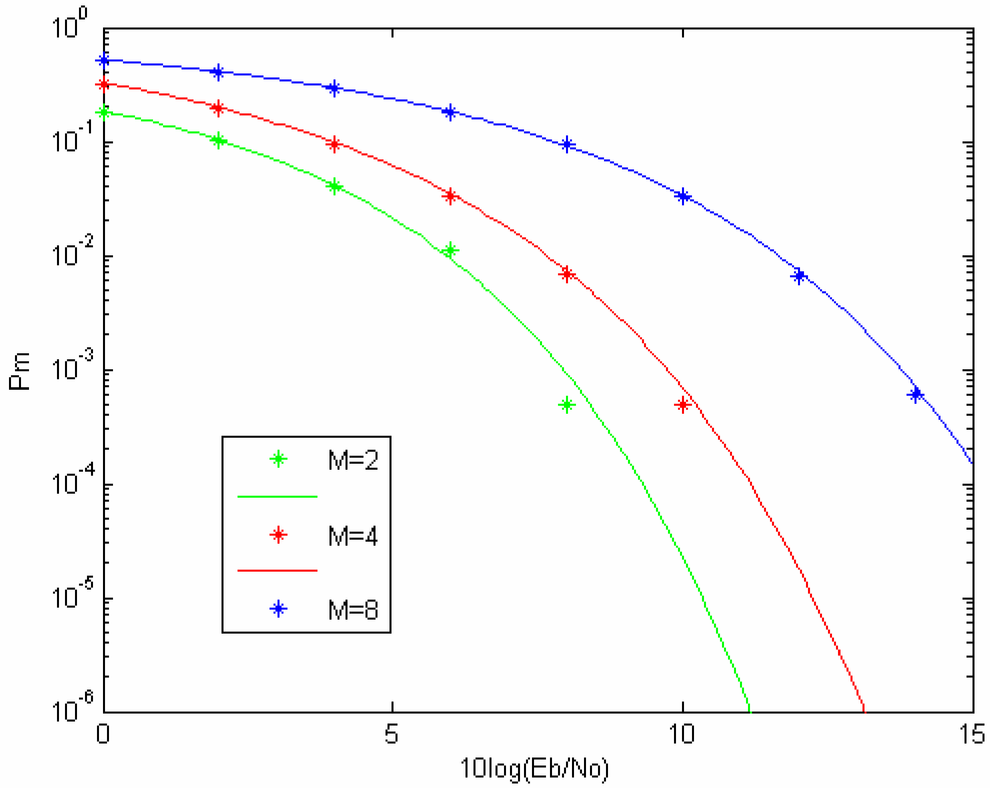

Figura 5.5. Probabilidad de Error de símbolo vs SNR_bit con simulación Monte carlo comparada con la Probabilidad de error teórica utilizando sistema M-ario DPSK M=8, M=4 y M=2

b)

```
echo on;
M_8=8;
M_4=4;
tolerance=1e-7;
plus_inf=20;
snrdb=5:0.1:15;
snrdb2=5:0.1:17;
for i=1:length(snrdb)
    SNR=10^(snrdb(i)/10);
   %%%%%%%%%%%%%%%%%%%%%%
   % M=2
   %%%%%%%%%%%%%%%%%%%%%%
    teoricab2_psk(i)=2*Q(sqrt(2*SNR));
    teoricab2_dpsk(i)=(1/2)*exp(-SNR);
    %%%%%%%%%%%%%%%%%%%%%
    %M=4
    %%%%%%%%%%%%%%%%%%%%%
    a1_4=sqrt(SNR*(log2(4))*(1-sin(pi/4)));
    b1_4=sqrt(SNR*(log2(4))*(1+sin(pi/4)));
    teoricab4_psk(i)=Q(sqrt(2*SNR))*(1-((1/2)*Q(sqrt(2*SNR))));

teoricab4_dpsk(i)=(1/log2(4))*(1+quadl('dpsk_int',b1_4,plus_inf
,tolerance,[],a1_4)-
quadl('dpsk_int',a1_4,plus_inf,tolerance,[],b1_4));
    echo off;
end;
echo on;
%%%%M=8%%%%
for i=1:length(snrdb2)
    SNR=10^(snrdb2(i)/10);

teoricab8_psk(i)=(2/log2(8))*Q(sqrt(2*log2(8)*SNR)*sin(pi/8));
    teoricam8_psk(i)=(2)*Q(sqrt(2*log2(8)*SNR)*sin(pi/8));
    a1_8=sqrt(SNR*(log2(8))*(1-sin(pi/8)));
    b1_8=sqrt(SNR*(log2(8))*(1+sin(pi/8)));

teoricab8_dpsk(i)=(1/log2(8))*(1+quadl('dpsk_int',b1_8,plus_inf
,tolerance,[],a1_8)-
quadl('dpsk_int',a1_8,plus_inf,tolerance,[],b1_8));
    echo off;
end;
semilogy(snrdb,teoricab2_psk,'b',snrdb,teoricab2_dpsk,'r');
hold on;
semilogy(snrdb,teoricab2_psk,'b',snrdb,teoricab4_dpsk,'r');
semilogy(snrdb2,teoricab8_psk,'b',snrdb2,teoricab8_dpsk,'r');
xlabel('10log(Eb/No)');
ylabel('Pb');
legend('PSK','DPSK');
axis([5 18 1e-6 1]);
title('Comparación PSK - DPSK');
hold off;
echo off;
```

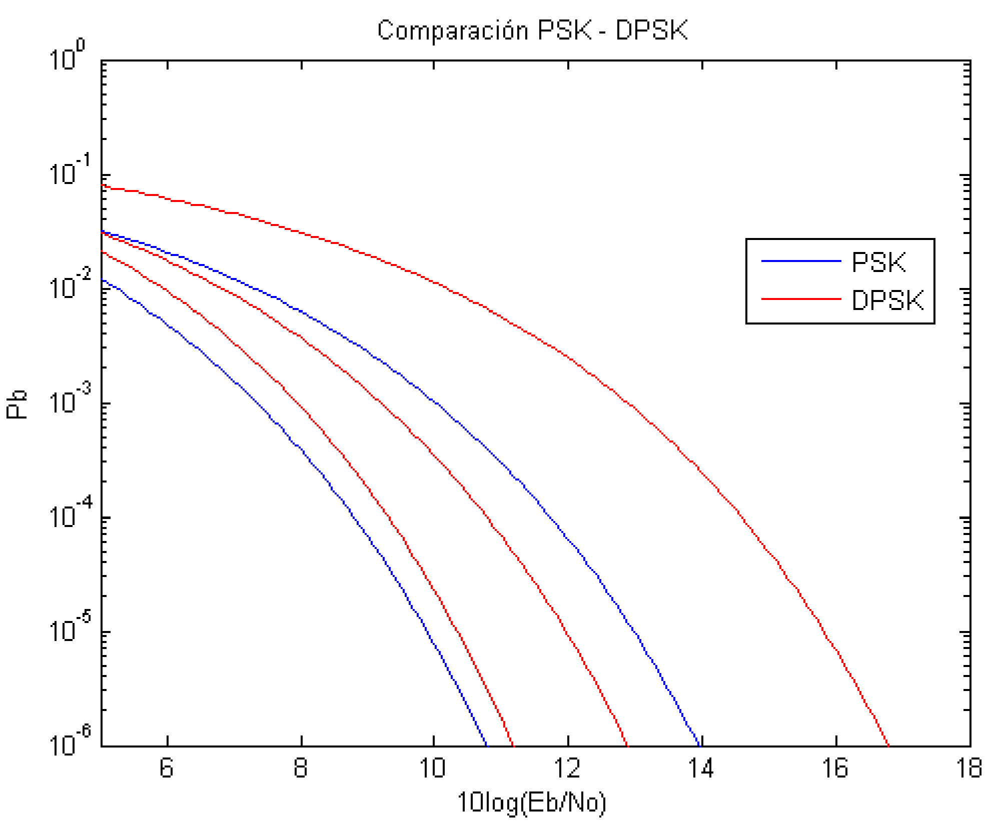

Figura 5.6. Probabilidad de Error de bit vs SNR_bit teórica
para un sistema M-ario DPSK y PSK

Un receptor DPSK es más simple en comparación con un receptor coherente PSK, ya que no requiere de la recuperación de la fase de la portadora, así que para M=2 se prefiere a DPSK que la PSK.

Utilizar una modulación DPSK da peores resultados que la PSK. Se puede concluir que la ejecución de DPSK es 3dB peor que la PSK cuando M>4. Así que para M>4 se prefiere DPSK.

4. Modulación QAM

La Modulación de Amplitud en Cuadratura o QAM es una modulación digital en la que el mensaje está contenido tanto en la amplitud como en la fase de la señal transmitida.

Si definimos un conjunto de $C_i = V_i e^{j\phi_i}$, $i = 1, 2, ..., M$ de manera que si se cumple: $A_{ic} = \mathrm{Re}\left[C_i\right], \quad A_{is} = \mathrm{Im}\left[C_i\right]$.

Las M señales transmitidas en una modulación QAM pueden representarse de la forma:

$$
\begin{aligned}
S_i(t) &= \mathrm{Re}\left[g_{li}(t) e^{j\omega_c t} \right] = \mathrm{Re}\left[g(t)(A_{ic} + A_{is}) e^{j\omega_c t} \right] = \\
&= A_{ic} g(t)\cos(\omega_c t) - A_{is} g(t) sen(\omega_c t) \quad i = 1, 2, ..., M \quad 0 \le t \le T
\end{aligned}
\tag{5.4.1}
$$

Podemos comprobar que efectivamente la información está contenida en la amplitud y la fase de la señal transmitida observando que la señal puede expresarse alternativamente de la siguiente manera:

$$
S_i(t) = \mathrm{Re}\left[V_i e^{j\phi_i} g(t) e^{j\omega_c t} \right] = V_i g(t)\cos(\omega_c t + \phi) \qquad i = 1, 2, ..., M \tag{5.4.2}
$$

Las señales QAM pueden representarse mediante una combinación lineal de dos señales ortonormales, definidas de la forma:

$$
\begin{aligned}
\phi_1(t) &= \sqrt{\frac{2}{E_g} g(t)} \cos(\omega_c t) \\
\phi_2(t) &= \sqrt{\frac{2}{E_g}} g(t) \sin(\omega_c t)
\end{aligned}
\tag{5.4.3}
$$

de forma que: $S_i(t) = S_{i1}(t)\phi_1(t) + S_{i2}(t)\phi_2(t) \qquad i = 1, 2, ..., M$ (5.4.4)

$$
\vec{S}_i = \begin{bmatrix} S_{i1} \\ S_{i2} \end{bmatrix} = \sqrt{\frac{E_g}{2}} \begin{bmatrix} A_{ic} \\ A_{is} \end{bmatrix} \qquad i = 1, 2, ..., M \tag{5.4.5}
$$

Como ejemplo de las distintas constelaciones, tenemos:

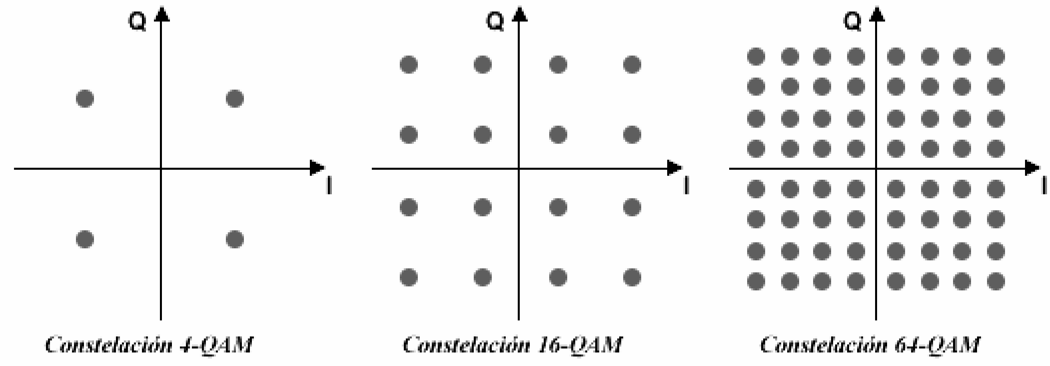

Figura 5.7. Ejemplo de Constelaciones QAM

Para calcular la probabilidad de error del esquema de modulación QAM, es imposible dar un resultado genérico por lo que nos centraremos, al igual que en teoría en el caso rectangular, en el que M_1 y M_2 (M_1 símbolos verticales y M_2 símbolos horizontales) son números pares. se puede demostrar que:

La energía promedio de la constelación:

$$E_{av} = \frac{d^2(M_1^2 + M_2^2 - 2)}{12} \quad \Rightarrow \quad \boxed{E_{avb} = \frac{d^2(M_1^2 + M_2^2 - 2)}{12}\log_2 M} \qquad (5.4.6)$$

y la probabilidad de error de símbolo:

$$P_M \le 4\left(1 - \frac{M_1 + M_2}{2M}\right) Q\left(\frac{1}{\sqrt{2N_0}}\right) = 4\left(1 - \frac{M_1 + M_2}{2M}\right) Q\left(\frac{3E_{av}}{\left(\frac{M_1^2 + M_2^2}{2} - 1\right) N_0}\right) \qquad (5.4.7)$$

Para el caso particular $M_1 = M_2 = \sqrt{M}$:

$$\boxed{P_M \le 4Q\left(\sqrt{\frac{3k}{M-1}\frac{E_{avb}}{N_0}}\right)} \qquad (5.4.8)$$

Ejercicio 5.3.

Enunciado

Usar la simulación de Monte Carlo, vista en la práctica 2 para estimar y dibujar la probabilidad de error de símbolo frente a la SNR por bit en decibelios para una transmisión M-QAM que emplea demodulador de correlación o de filtros acoplados. Dibujar para M=4 , M=16 y M=64. El modelo del sistema se muestra en la siguiente figura:

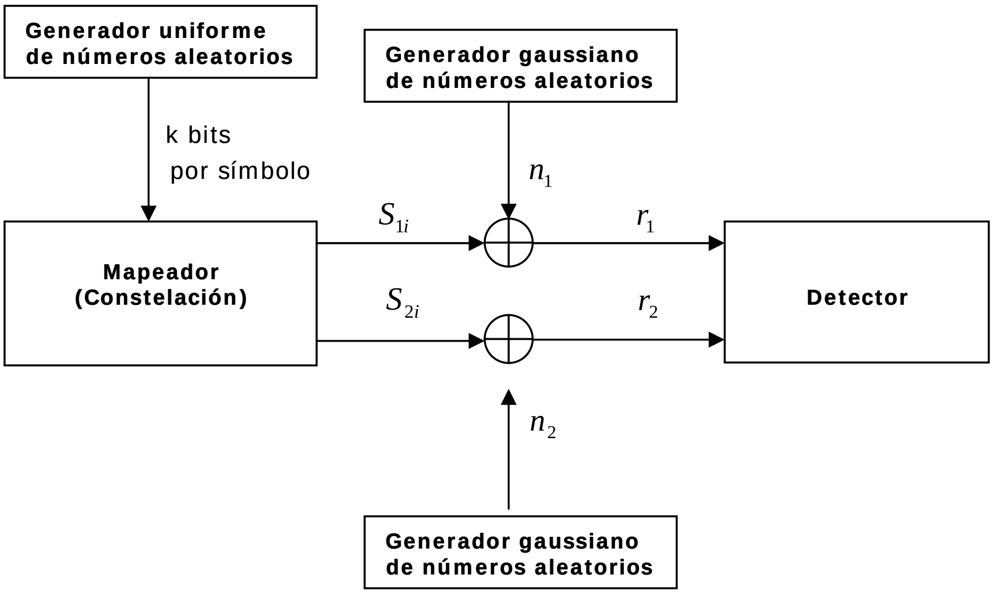

Figura 5.8. Diagrama de bloques para un sistema M-QAM para la realización de la simulación de Monte Carlo

Resultados

```
function [p]=qam(snr_in_dB,M1,M2)

M=M1*M2;
K=log2(M);
N=10000;
d=1;                          % minima distancia entre
simbolos
Eav=(K*d^2*((M1^2)+(M2^2)-2))/12;
snr=10^(snr_in_dB/10);        % SNR por bit (dada)
sgma=sqrt(Eav/(2*K*snr));     % varianza del ruido
M=M1*M2;
%%%%---- Generacion de N simbolos del 1 al M----%
for i=1:N,
  aux=rand;                   % VA uniforme de (0,1)
  fuente(i)=1+floor(M*aux);   % un numero aleatorio de 1 a M
uniforme
end;
%%%%%%%Mapeador de la constelación, suponiendo
cuadrada%%%%%%%
vertical0=-M1*d+d;
for j=1:M2
   vertical(j)=vertical0+(2*d*(j-1));
end;
horizontal0=M2*d-d;
for j=1:M1
    horizontal(j)=horizontal0-(2*d*(j-1));
end;
k=1;
for j=1:M2
    for i=1:M1
        mapeador(k,:)=[vertical(i) horizontal(j)];
        k=k+1;
    end;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
for i=1:N,
  s(i,:)=mapeador(fuente(i),:);
end;
% received signal
for i=1:N,
    n(1)=sgma*randn(1,1);
    n(2)=sgma*randn(1,1);
    for j=1:2
        r(i,j)=s(i,j)+n(j);
    end;
end;
% detection and error probability calculation
error=0;
for i=1:N,
  % Metric computation follows.
  for j=1:M,
    metrica(j)=(r(i,1)-mapeador(j,1))^2+(r(i,2)-
mapeador(j,2))^2;
  end;
  [min_m decision] = min(metrica);
  if (decision~=fuente(i)),
    error=error+1;
```

```
echo on
snrdb1=0:2:18;
snrdb2=0:0.1:18;
M_4=4;
K_4=log2(M_4);
M_16=16;
K_16=log2(M_16);
M_64=64;
K_64=log2(M_64);
for i=1:length(snrdb1)
  simuladam_4(i)=qam(snrdb1(i),K_4,K_4);       % tasa de error
simulada M=4
  simuladam_16(i)=qam(snrdb1(i),K_16,K_16);    % tasa de error
simulada M=16
  simuladam_64(i)=qam(snrdb1(i),K_64,K_64);    % tasa de error
simulada M=64
  echo off;
end;


for i=1:length(snrdb2),
  SNR=exp(snrdb2(i)*log(10)/10);
  % tasas de simbolos teóricas
  teoricam_4(i)=4*(1-(1/sqrt(M_4)))*Q(sqrt(3*K_16*SNR/(M_4-1)));
  teoricam_16(i)=4*(1-(1/sqrt(M_16)))*Q(sqrt(3*K_16*SNR/(M_16-
1)));
  teoricam_64(i)=4*(1-(1/sqrt(M_64)))*Q(sqrt(3*K_64*SNR/(M_64-
1)));
  echo off ;
end;
echo on ;

semilogy(snrdb1,simuladam_64,'b*',snrdb2,teoricam_64,'b');
hold on;
semilogy(snrdb1,simuladam_16,'r*',snrdb2,teoricam_16,'r');
semilogy(snrdb1,simuladam_4,'g*',snrdb2,teoricam_4,'g');

xlabel('10log(Eb/No)');
ylabel('Pm');
```

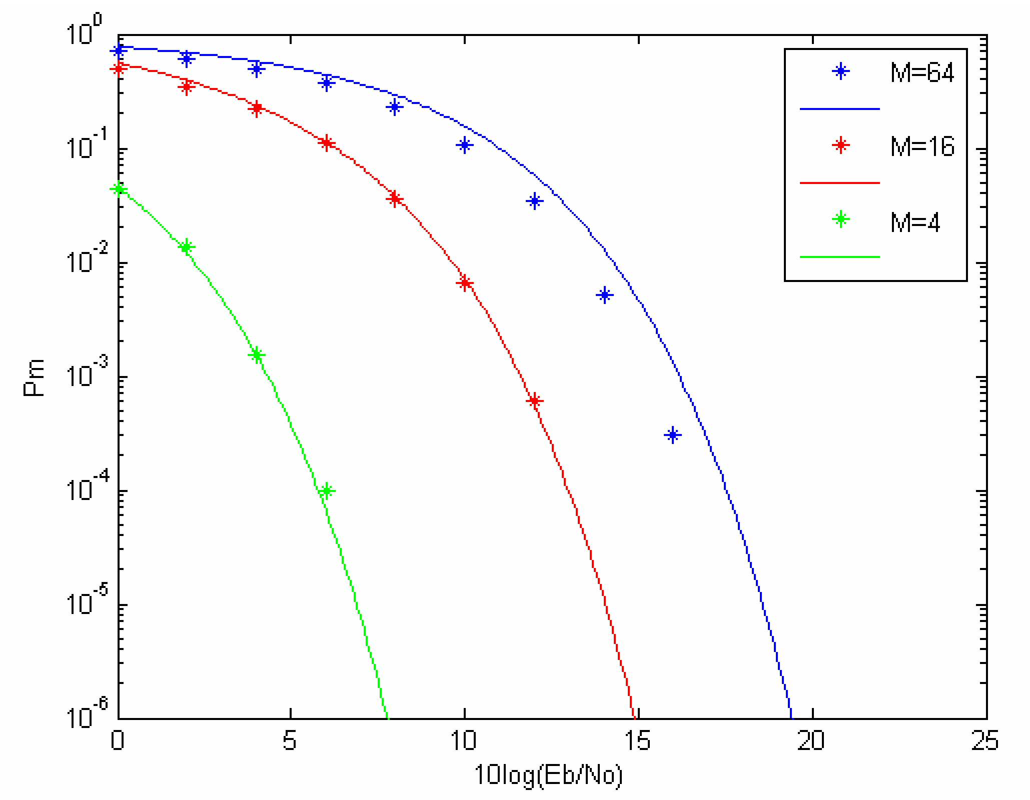

Figura 5.9. Probabilidad de Error de símbolo vs SNR_bit con simulación Monte carlo comparada con la Probabilidad de error teórica utilizando sistema M-ario QAM con M=64, M=16 y M=4

5. Modulación Ortogonal

Hemos descrito métodos para la transmisión de información digital modulando con la amplitud de la portadora, con la fase de la portadora, o con una combinación de la amplitud y la fase.

Como observaremos en este apartado la transmisión por modulación en frecuencia es un método de modulación apropiado para canales que carezcan de estabilidad en la fase y que sea necesario una estimación de la fase de la portadora. A diferencia de los métodos de modulación lineales que hemos introducido (PAM, PSK coherente y QAM), esta modulación requiere de la estimación de la fase de la portadora para ejecutar la detección coherente de la fase.

5.1. Modulación por desplazamiento en Frecuencia (FSK)

La forma más simple de la modular en frecuencia es una modulación FSK binaria (BFSK). En una BFSK empleamos dos frecuencias diferentes, llamadas, f_1 y $f_2 = f_1 + \Delta f$, para transmitir una secuencia de información binaria. La elección de la frecuencia de separación $\Delta f = f_2 - f_1$ se trata después. Las dos señales se pueden expresar como:

$$\begin{aligned} s_1(t) &= \sqrt{\frac{2E_b}{T_b}} \cos 2\pi f_1 t \qquad & 0 \le t \le T_b \\ s_2(t) &= \sqrt{\frac{2E_b}{T_b}} \cos 2\pi f_2 t \qquad & 0 \le t \le T_b \end{aligned} \tag{5.5.1}$$

donde E_b es la energía por bit de la señal y T_b la duración del intervalo de bit.

De forma más general, una modulación M-aria FSK puede ser usada para transmitir bloques de $k = \log_2 M$ bits por señal. En este caso las M señales se expresan como:

$$s_m(t) = \sqrt{\frac{2E_s}{T}} \cos(2\pi f_c t + 2\pi m \Delta f t), \qquad m = 0, \ldots, M-1 \qquad 0 \le t \le T \tag{5.5.2}$$

donde $E_s = kE_b$ es la energía por símbolo, $T = kT_b$ es el intervalo de símbolo y Δf es la separación en frecuencia entre frecuencias consecutivas, es decir $\Delta f = f_m - f_{m-1}$ para todas $m = 1, 2, \ldots, M-1$, donde $f_m = f_c + m\Delta f$.

Se observa que las M señales FSK tienen igual energía, E_s. La frecuencia de separación Δf entre frecuencias determina el grado en el cual podemos discriminar entre las M posibles señales transmitidas. Como medida de la semejanza entre un par de señales, usamos el coeficiente de correlación γ_{nm}:

$$\gamma_{nm} = \frac{1}{E_s}\int_0^T S_m(t)S_n(t)dt \qquad (5.5.3)$$

Sustituyendo $S_m(t)$, se obtiene:

$$\begin{aligned}\gamma_{nm} &= \frac{1}{E_s}\int_0^T \frac{2E_s}{T}\cos(2\pi f_c t + 2\pi m\Delta f t)\cos(2\pi f_c t + 2\pi n\Delta f t)dt = \\ &= \frac{1}{T}\int_0^T \cos 2\pi(m-n)\Delta f t dt + \frac{1}{T}\int_0^T \cos\left[4\pi f_c t + 2\pi n\Delta f\right] t dt = \\ &= \frac{\sin 2\pi(m-n)\Delta f T}{2\pi(m-n)\Delta f T}\end{aligned} \qquad (5.5.4)$$

donde la segunda integral desaparece cuando $f_c >> 1/T$. Se observa que $\gamma_{nm} = 0$ las señales son ortogonales cuando Δf es múltiplo de $1/2T$. Entonces la frecuencia mínima de separación entre frecuencias consecutivas ortogonales es de $1/2T$.

Los vectores de señal para una modulación M-aria FSK ortogonal son de dimensión $M \times M$ dados por:

$$\begin{aligned} S_0 &= (\sqrt{E_s}, 0, ..., 0) \\ S_1 &= (0, \sqrt{E_s} ..., 0) \\ &\dots . \\ S_{M-1} &= (0, ..., 0, \sqrt{E_s}) \end{aligned} \qquad (5.5.5)$$

donde las funciones bases son: $\phi_m(t) = \sqrt{2/T}\cos 2\pi(f_c + m\Delta f)t$. La distancia entre una pareja de vectores de señal es de $d = \sqrt{2E_s}$ para todo m, n, que coincide con la distancia mínima.

5.2. Demodulación y Detección de Señales FSK

Ahora suponemos que las señales FSK se transmiten a través de un canal con ruido aditivo, blanco gaussiano. Además, suponemos que cada señal es retrasada en la transmisión a través del canal. Consecuentemente la señal recibida a la entrada del demodulador puede ser expresada como

$$r(t) = \sqrt{\frac{2E_s}{T}}\cos(2\pi f_c t + 2\pi m\Delta f t + \varphi_m) + n(t) \qquad (5.5.6)$$

donde, φ_m es el desplazamiento de fase de la señal m (debido al retraso en la transmisión) y $n(t)$ representa el ruido paso de banda, que puede ser expresado como:

$$n(t) = n_c(t)\cos 2\pi f_c t - n_s(t)\sin 2\pi f_c t \qquad (5.5.7)$$

La demodulación y la detección de señales M-FSK se puede hacer de dos formas. La primera sería estimar los M desplazamientos de fase de la portadora $\{\varphi_m\}$ y ejecutar una demodulación y detección de fase coherente. Como forma alternativa, la fases de la portadora se ignoran en la demodulación y detección.

En una demodulación de fase coherente, la señal recibida $r(t)$ se multiplica con cada una de las M posibles señales recibidas,

$$\cos(2\pi f_c t + 2\pi m\Delta f t + \hat{\varphi}_m) \quad \text{para} \quad m = 0,1,...,M-1 \qquad (5.5.8)$$

donde $\{\hat{\varphi}_m\}$ son las estimaciones de la fase de la portadora. Un diagrama de bloques que ilustra este tipo de demodulación se muestra en la figura 5.10. Es interesante observar que si $\varphi_m \neq \hat{\varphi}_m$ para $m = 0,...,M-1$ (estimación de la fase imperfecta, la frecuencia de separación requerida para que las señales sean ortogonales en el demodulador es $\Delta f = 1/T$, la cual es dos veces la mínima separación para que las señales sean ortogonales cuando $\varphi_m = \hat{\varphi}_m$.

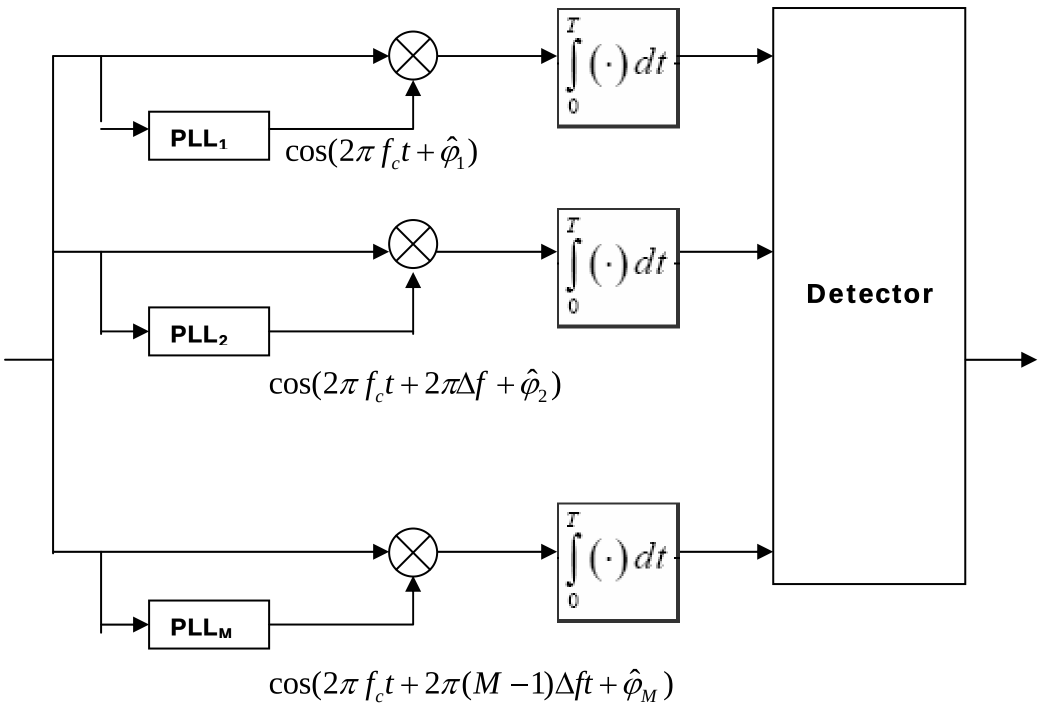

Figura 5.10. Demodulación coherente de señales M-FSK

La necesidad de estimar las M fases hace que la demodulación de las señales FSK sea muy compleja en la práctica, especialmente cuando el número de señales es muy grande. Entonces, no consideraremos una detección coherente para las señales FSK.

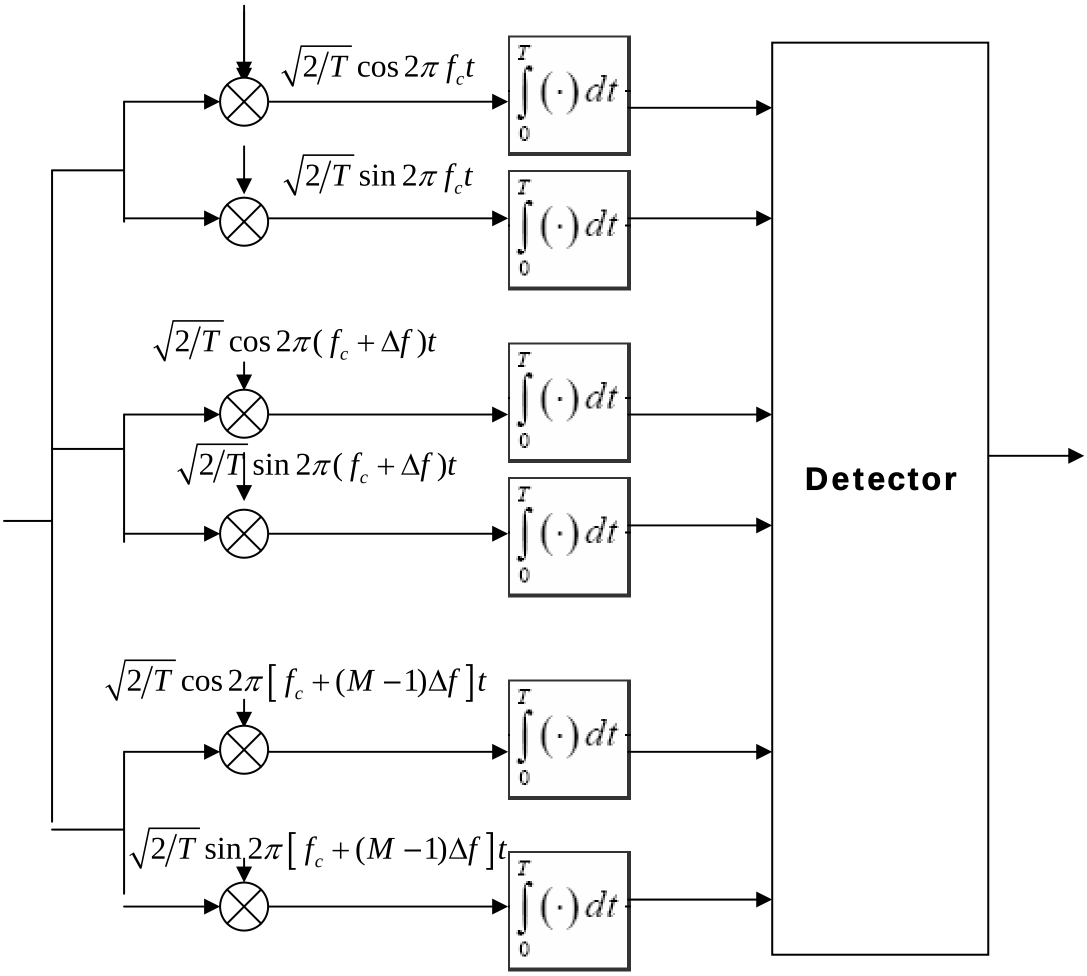

Figura 5.11. Demodulación no coherente de señales M-FSK

Entonces consideraremos un método de demodulación y detección que no requiere del conocimiento de las fases. La demodulación puede ser efectuada tal y como se muestra en la figura 5.11.En este caso hay dos correladores por señal, en total $2M$ correladores, en general. La señal recibida se multiplica por las funciones bases. Las $2M$ entradas al detector, se pueden expresar de la siguiente forma:

$$
\begin{aligned}
r_{kc} &= \sqrt{E_s}\left[\frac{\sin 2\pi(k-M)\Delta fT}{2\pi(k-M)\Delta fT}\cos\varphi_m - \frac{\cos 2\pi(k-M)\Delta fT}{2\pi(k-M)\Delta fT}\sin\varphi_m\right] + n_{kc} \\
r_{ks} &= \sqrt{E_s}\left[\frac{\cos 2\pi(k-M)\Delta fT}{2\pi(k-M)\Delta fT}\cos\varphi_m - \frac{\sin 2\pi(k-M)\Delta fT}{2\pi(k-M)\Delta fT}\sin\varphi_m\right] + n_{ks}
\end{aligned}
\qquad (5.5.9)
$$

donde n_{kc} y n_{ks} denotan las componentes gaussianas del ruido.

Observamos que cuando $k = m$,

$$\begin{aligned} r_{mc} &= \sqrt{E_s}\cos\varphi_m + n_{mc} \\ r_{ms} &= \sqrt{E_s}\cos\varphi_m + n_{ms} \end{aligned} \tag{5.5.10}$$

Además, observamos que cuando $k \neq m$, las componentes r_{kc} y r_{ks} se desvanecerán, independientemente de los valores del desplazamiento de fase φ_m, cuando la separación entre frecuencias consecutivas es de $\Delta f = 1/2T$. Con lo que cuando $k \neq m$;

$$\begin{aligned} r_{kc} &= n_{kc} \\ r_{ks} &= n_{ks} \end{aligned} \tag{5.5.11}$$

Para los siguientes pasos, supondremos que $\Delta f = 1/2T$, así pues las señales son ortogonales.

Las componentes, $\{n_{kc}\}$ y $\{n_{ks}\}$ son variables aleatorias gaussianas con media cero e incorreladas con varianza $\sigma^2 = N_0/2$. Consecuentemente, la función de densidad de probabilidad conjunta para r_{mc} y r_{ms} condicionadas a φ_m es:

$$f_{r_m}\left(r_{mc}, r_{ms} \mid \varphi_m\right) = \frac{1}{2\pi\sigma^2} e^{-\left[\left(r_{kc} - \sqrt{E_S}\cos\varphi_m\right)^2 + \left(r_{mc} - \sqrt{E_S}\sin\varphi_m\right)^2\right] / 2\sigma^2} \qquad k = m$$

$$f_{r_k}\left(r_{kc}, r_{ks} \mid \varphi_m\right) = \frac{1}{2\pi\sigma^2} e^{-\left(r_{kc}^2 + r_{ks}^2\right) / 2\sigma^2} \qquad k \neq m \tag{5.5.12}$$

Dadas las $2M$ variables aleatorias observadas $\{r_{kc}, r_{ks}\}_{k=0}^{M-1}$, el detector óptimo selecciona la señal que corresponda al máximo de las siguientes probabilidades:

$$P\left[s_m txda \mid r\right] = P(s_m \mid r) \qquad m = 0, \ldots, M-1 \tag{5.5.13}$$

cuando las señales son igualmente probables, el detector óptimo calcula la envolvente de la señal, definido como:

$$r_m = \sqrt{r_{mc}^{\ 2} + r_{ms}^{\ 2}} \qquad m = 0, \ldots, M-1. \tag{5.5.14}$$

y selecciona la señal correspondiente a la mayor envolvente del conjunto $\{r_m\}$. En este caso al detector óptimo se le llama detector de envolvente.

Un detector equivalente es el que calcula:

$$r_m^{\ 2} = r_{mc}^{\ 2} + r_{ms}^{\ 2} \qquad m = 0, \ldots, M-1 \tag{5.5.15}$$

5.3. Probabilidad de error para una detección no coherente de señales FSK

El desarrollo del detector de envolvente óptimo para señales M-arias FSK puede ser encontrado en la mayoría de los textos de comunicaciones digitales. La probabilidad de símbolo puede ser expresada como:

$$P_M = \sum_{n=1}^{M-1} (-1)^{n+1} \binom{M-1}{n} \frac{1}{n+1} e^{-nkE_b/N_0(n+1)} \tag{5.5.16}$$

Cuando $M = 2$, esta expresión se reduce a la probabilidad de error para una modulación binaria FSK:

$$P_2 = \frac{1}{2} e^{-E_b/N_0} \tag{5.5.17}$$

Para $M > 2$, la probabilidad de error de bit puede ser obtenida de la probabilidad de error de símbolo, mediante la siguiente relación:

$$P_b = \frac{2^{k-1}}{2^k - 1} P_M \tag{5.5.18}$$

Ejercicio 5.4.

Enunciado

Usar la simulación de Monte Carlo, vista en la práctica 2 para estimar y dibujar la probabilidad de error de símbolo frente a la SNR por bit en decibelios para una transmisión M-FSK que emplea demodulador de correlación o de filtros acoplados. Dibujar para M=2, M=4, M=16 y M=32 . El modelo del sistema se muestra en la siguiente figura:

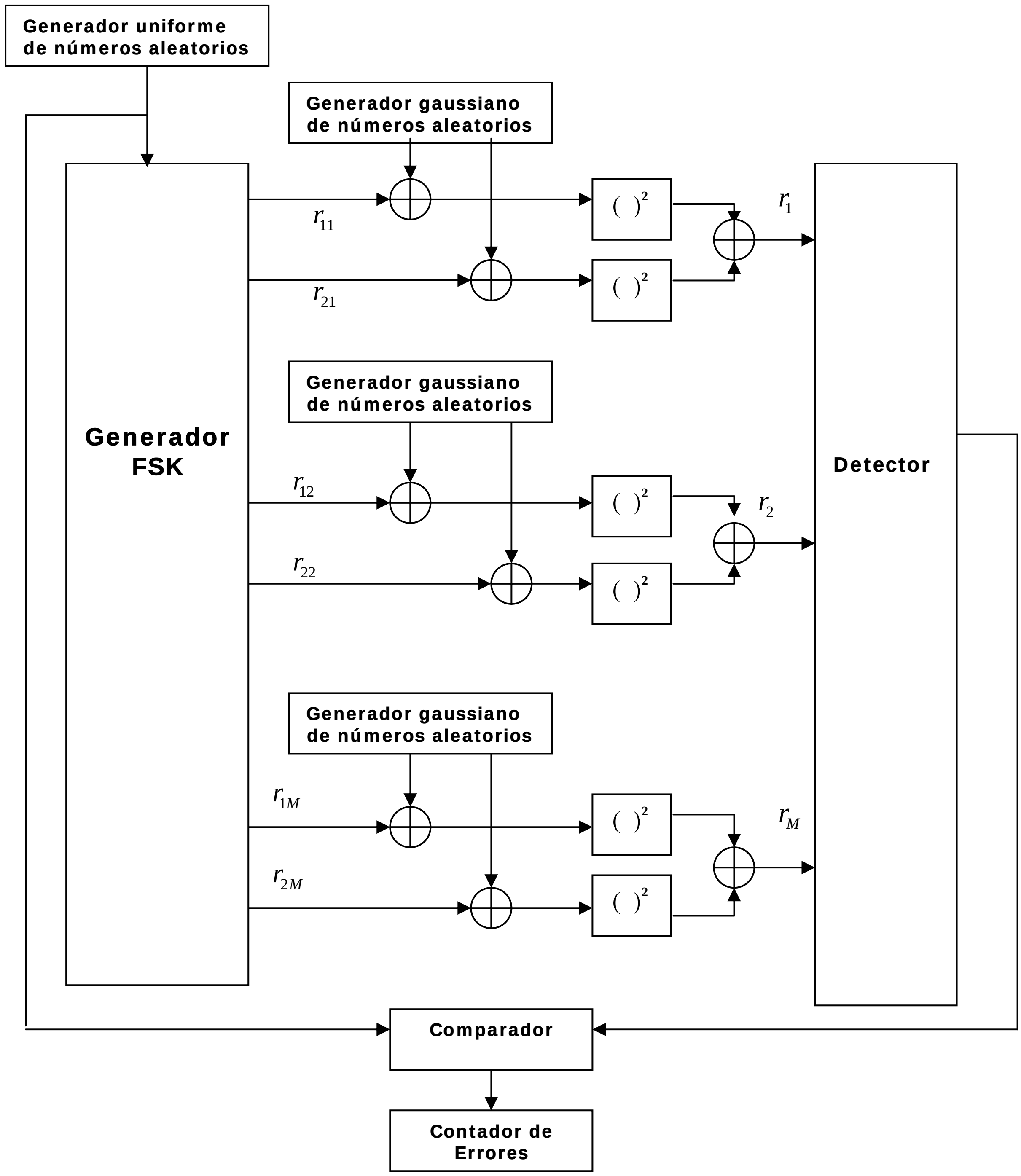

Figura 5.12. Modelo de Simulación para el ejercicio 5.4

Resultado

```
function [p]=fsk(snr_db,M)
N=10000;
Eb=1;
R=log2(M);
Es=Eb*R;
d=1;
snr=10^(snr_db/10);
sigma=sqrt(Es/(2*R*snr));
phi=0;

%%%%---- Generacion de N simbolos del 1 al M----%
for j=0:M-1
    code(j+1,1)=(j*1);
end
codea=de2bi(code,R,'left-msb');
[fila,columna]=size(codea);

for i=1:N
    aux=rand(1,1);
    simbolo(i)=floor(M*aux);
    for j=1:columna
        fuente(i,j)=codea((simbolo(i)+1),j);
    end;
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

error=0;

for i=1:N
    for j=1:M
        r(1,j)=sigma*randn(1,1);
        r(2,j)=sigma*randn(1,1);
        if((j-1)==simbolo(i))
            r(1,j)=r(1,j)+(sqrt(Es)*cos(phi));
            r(2,j)=r(2,j)+(sqrt(Es)*sin(phi));
        end;
    end;
    for j=1:M
        c(j)=((r(1,j)^2)+(r(2,j)^2));
    end;
    max_c=max(c(:));
    posicion=strfind(c,max_c);
    for j=1:columna
        decision(1,j)=codea(posicion,j);
    end;
    if(fuente(i)~=decision)
        error=error+1;
    end;
end;
p=error/N;
```

```
echo on
snrdb1=0:2:15;
snrdb2=0:0.1:15;
M_2=2;
M_4=4;
M_16=16;
M_32=32;
for i=1:length(snrdb1),
  [simulada_2(i)]=fsk(snrdb1(i),M_2);
  [simulada_4(i)]=fsk(snrdb1(i),M_4);
  [simulada_16(i)]=fsk(snrdb1(i),M_16);
  [simulada_32(i)]=fsk(snrdb1(i),M_32);
end;
echo on ;
for i=1:length(snrdb2),
  SNR=10^(snrdb2(i)/10);
  teorica_2(i)=(1/2)*exp(-SNR/2);
  teorica_4(i)=((M_4-1)*Q(sqrt(SNR*log2(M_4))));
  teorica_16(i)=((M_16-1)*Q(sqrt(SNR*log2(M_16))));
  teorica_32(i)=((M_32-1)*Q(sqrt(SNR*log2(M_32))));
end;


semilogy(snrdb1,simulada_2,'g*',snrdb2,teorica_2,'g');
hold on;
semilogy(snrdb1,simulada_4,'r*',snrdb2,teorica_4,'r');
semilogy(snrdb1,simulada_16,'b*',snrdb2,teorica_16,'b');
semilogy(snrdb1,simulada_32,'m*',snrdb2,teorica_32,'m');
legend('M=2','','M=4','','M=16','','M32','');
xlabel('10log(Eb/No)');
ylabel('Pm');
hold off;
echo off;
```

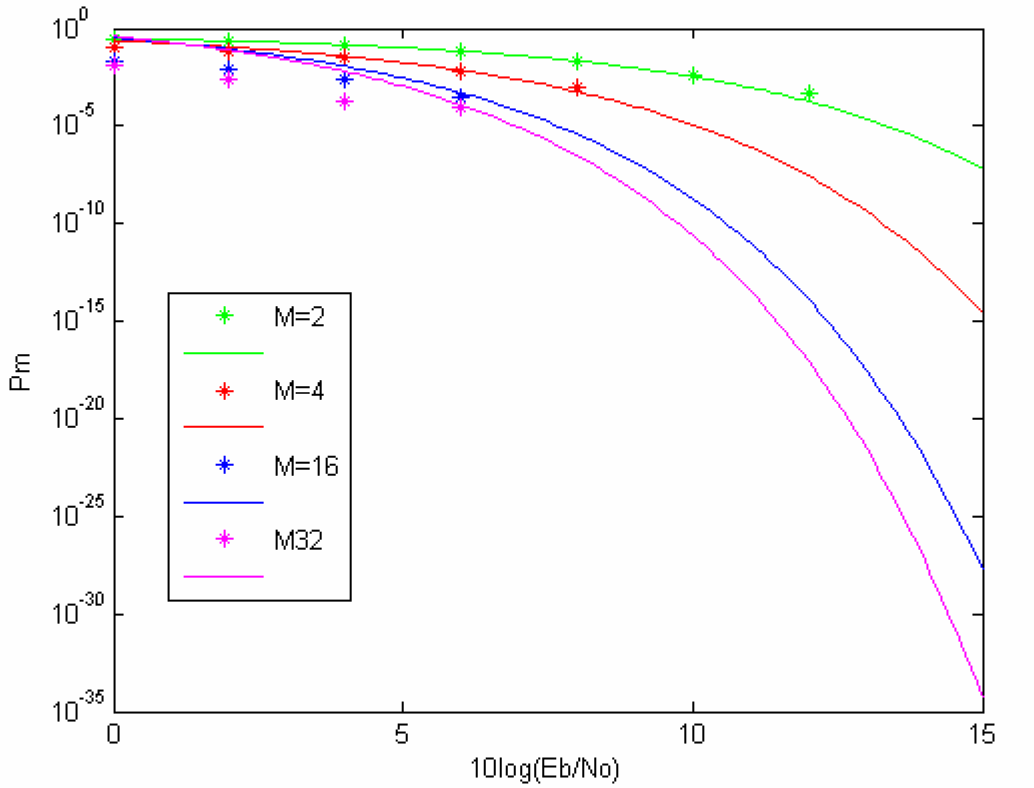

Figura 5.13. Probabilidad de Error de símbolo vs SNR_bit con simulación Monte carlo comparada con la Probabilidad de error teórica utilizando sistema M-ario FSK con M=64, M=16 y M=4

6. Comparación entre los diferentes esquemas de modulación digitales

Hay muchas formas de calcular el ancho de banda de los esquemas de modulación digitales M-arios. Una forma de hacer este cálculo sencillo es considerar como ancho de banda, el requerido para transmitir el lóbulo principal del espectro de la señal.

El ancho de banda en radio frecuencia de los esquemas de modulación que usan una única portadora sinusoidal para transmitir la información como los métodos MPSK, MDPSK y MASK es el siguiente:

$$B_{RF} = \frac{2}{T_s} = 2R_s \tag{5.6.1}$$

donde R_s es la tasa de símbolo. Sin embargo, para un esquema de modulación M-ario, el tiempo de símbolo está relacionado con el tiempo de bit por:

$$T_s = T_b \log_2 M = \frac{\log_2 M}{R_b} \tag{5.6.2}$$

Así pues, el ancho de banda en términos de tasa de bit para estos esquemas de modulación es:

$$B_{RF} = \frac{2R_b}{\log_2 M} \qquad \text{(MPSK, MDPSK, MQASK,QAM)} \tag{5.6.3}$$

Al cociente entre la tasa de transmisión $R_b \; {}^{b}\!/_{S}$ y el ancho de banda ocupado se le denomina Eficiencia Espectral, la cual para estos esquemas de modulación es la siguiente:

$$\rho = \frac{R_b}{B_{RF}} = \frac{\log_2 M}{2} \qquad \text{(MPSK, MDPSK, MASK,QAM)} \tag{5.6.4}$$

Para esquemas que usan múltiples frecuencias para transmitir la información como el esquema M-ario FSK, se utiliza a una forma más general. Cada símbolo se representa por frecuencias diferentes. En un esquema M-ario FSK coherente, la separación mínima requerida para mantener la ortogonalidad es de $1/2T_s$ Hz. Las dos frecuencias en los extremos del espectro usan solo $1/T_s$ (la mitad del lóbulo principal) para componer el espectro. Las $M-2$ frecuencias interiores requieren una separación mínima de $1/2T_s$ Hz (hay $M-1$ slots interiores), con todo esto se tiene:

$$B_{RF} = \frac{1}{T_s} + \frac{M-1}{2T_s} + \frac{1}{T_S} = \frac{M+3}{2T_S} \qquad \text{(MFSK coherente)} \tag{5.6.5}$$

$$\rho = \frac{R_b}{B_{RF}} = \frac{2\log_2 M}{M+3} \qquad \text{(MFSK coherente)} \tag{5.6.6}$$

Para una modulación FSK no coherente, la separación mínima de frecuencias usada para representar los símbolos es de $2/T_S$ Hz, con lo cual en ancho de banda:

$$B_{RF} = \frac{1}{T_S} + \frac{2(M-1)}{T_S} + \frac{1}{T_S} = \frac{2M}{T_S} \quad \text{(MFSK no coherente)} \quad (5.6.7)$$

$$\rho = \frac{R_b}{B_{RF}} = \frac{\log_2 M}{2M} \quad \text{(MFSK no coherente)} \quad (5.6.8)$$

Haciendo una comparación entre los distintos esquemas de modulación desde el punto de vista espectral, se tiene que para los métodos de modulación MPSK, MDPSK y MASK a medida que M aumenta, la eficiencia espectral también aumenta, no ocurre lo mismo sin embargo para el esquema de modulación MFSK que a medida que M aumenta, la eficiencia espectral disminuye. Una comparación entre los diferentes métodos de modulación se da en la siguiente tabla:

M	MPSK, MDPSK MASK	MFSK Coherente	MFSK No coherente
2	0.5	0.400	0.250
4	1.0	0.571	0.250
8	1.5	0.545	0.188
16	2.0	0.421	0.125
32	2.5	0.286	0.078
64	3.0	0.179	0.047

Tabla 5.1. Comparación Eficiencia Espectral

La comparación final la efectuaremos con un método que combina la eficiencia espectral de la modulación y la SNR por bit necesaria para obtener una determinada probabilidad de error en una gráfica que se conoce como plano eficiencia espectral/probabilidad de error.

A continuación se ilustra una tabla que resume los datos calculados para el trazado de la gráfica:

	BW(rad/s)	°B(Hz)	ρ	Pm
MPSK	$4\pi/T$	$2/T$	10^{-5}	$P_M \;;\; 2Q\left[\sqrt{2SNR_b \log_2 M}\,\sin\left(\frac{\pi}{M}\right)\right]$
QAM	$4\pi/T$	$2/T$	$\frac{\log_2 M}{2}$	$P_M \;;\; 4Q\left[\frac{3\log_2 M}{(M-1)} SNR_b\right]$
MFSK (no coh)	$M\pi/T$	$M/2T$	$\frac{2\log_2 M}{M}$	$P_M \;;\; (M-1)Q\left(\sqrt{SNR_b \log_2 M}\right)$

Tabla 5.2. Tabla Resumen

Si representamos en una gráfica la eficiencia espectral que se puede alcanzar para los distintos esquemas de modulación, frente a la relación señal a ruido que debemos tener para conseguir una determinada probabilidad de error; en este caso 10^{-5}, obtenemos la siguiente figura que resume de forma útil todo lo estudiado y permite una comparación entre las diferentes modulaciones.

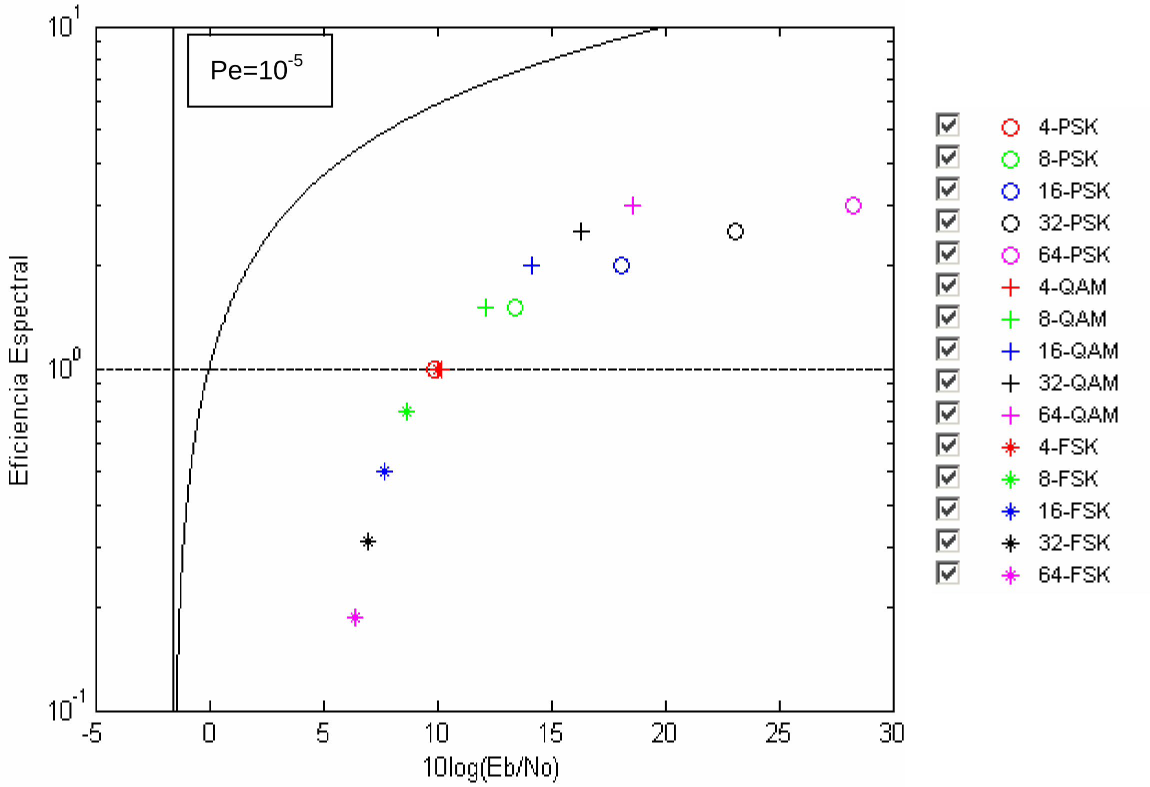

Figura 5.14. Comparación diferentes modulaciones para una Pe=10^{-5}

Observamos que en el caso de las modulaciones QAM y PSK, cuando M se incrementa la eficiencia espectral crece, pero a costa de un incremento en la SNR por bit. Consecuentemente estos sistemas son apropiados para canales de comunicación en los que el ancho de banda está limitado, se desea conseguir una eficiencia espectral mayor que la unidad y dónde es posible incrementar la potencia para soportar este incremento de M. El canal telefónico es un ejemplo típico de este caso.

Por otro lado, observamos que la modulación MFSK, posee una eficiencia espectral menor que la unidad. además cuando M se incrementa, la eficiencia espectral disminuye , debido a que necesita proporcionalmente un mayor ancho de banda. Sin embargo, la potencia necesaria para conseguir una determinada probabilidad de error de bit disminuye a medida que M aumenta. En consecuencia, tales sistemas son apropiados cuando nos encontramos con limitaciones en la potencia disponible y existe un ancho de banda suficiente para acomodar la señal.