Universidad Nacional de Córdoba

Variables Aleatorias y Procesos Estocásticos

Trabajo Práctico 1

Alumno: Ing. Daniel R. Garcia
Profesor: Dr. Ing. Damián A. Morero

January 4, 2019

1 Transmisor:

Para la simulación de los distintos sistemas de transmisión se designa a la variable $M = 10000$ como la cantidad de símbolos a enviar o cantidad de tiempo simulado.

El transmisor modula en QPSK (quadrature phase-shift keying), es decir que en cada instante de tiempo se transmite un símbolo del conjunto $\{(1,1),(1,-1),(-1,1),(-1,-1)\}$. El transmisor se codificará en matlab por medio de matrices de dimensión de 2xM, designando a S_{TX} como la matriz de símbolos y x e y como las señales en cuadratura I, Q.

$$S_{TX} = \begin{bmatrix} 1-2b_0^{(x)} & 1-2b_1^{(x)} & \cdots & 1-2b_M^{(x)} \\ 1-2b_0^{(y)} & 1-2b_1^{(y)} & \cdots & 1-2b_M^{(y)} \end{bmatrix} = I - 2B \tag{1}$$

En matlab su codificación es la siguiente:

```
% Generar Tx bits
tx_bits    = rand(2,M) > 0.5;
% Modulacion QPSK
tx_symbols = 1-2*tx_bits;
```

2 Canal:

El canal solo agrega ruido Gaussiano. Al igual que el Tx, la entrada y salida del canal se puede representar mediante dos matrices de 2xM, matrices de entrada y salida del canal (S_{TX} y S_{RX}). Podemos modelar el canal como:

$$S_{RX} = S_{TX} + G \tag{2}$$

siendo G la matriz de ruido:

$$G = \begin{bmatrix} g_0^{(x)} & g_1^{(x)} & \cdots & g_M^{(x)} \\ g_0^{(y)} & g_1^{(y)} & \cdots & g_M^{(y)} \end{bmatrix} \tag{3}$$

Donde el par de muestras $g_k^{(x)}$ y $g_k^{(y)}$ poseen distribución gaussiana de media $\mu_G = 0$ y matriz de covarianza:

$$C_G = K \begin{bmatrix} 17 & 11 \\ 11 & 13 \end{bmatrix} \tag{4}$$

Donde K es una constante que depende de la relación señal ruido (SNR) del canal. La SNR medida en dB se define como:

$$SNR = 10 \cdot log_{10}\left(\frac{Var\{S_{TX}\}}{Var\{G\}}\right) = 10 \cdot log_{10}\left(\frac{2}{\sigma^2_{g^{(x)}} + \sigma^2_{g^{(y)}}}\right) = 10 \cdot log_{10}\left(\frac{2}{30 \cdot K}\right) \tag{5}$$

El factor $(30 \cdot K)$ en la ecuación (5) es la traza de la matriz de covarianza (suma de los elementos de la diagonal). Despejando K de la ecuación:

$$K = \frac{2}{30 \cdot 10^{\frac{SNR}{10}}} \tag{6}$$

2.1 Ruido

La generación del ruido G se hace en dos pasos. El 1er paso consiste en generar una matriz Z de dimensiones 2xM cuyas componentes son todas muestras independientes de ruido gaussiano de media 0 y varianza 1. El 2do paso consiste en generar G a partir de Z multiplicando por una *matriz adecuada*, esta matriz se puede calcular mediante la descomposición en valores singulares de la matriz de covarianza de G.

```
C       = [17 11; 11 13];
% Cg     = K*C Matriz de Covarianza
C       = K*C;
[U,D,V] = svd(C);
% Queremos calcular U tal que C = U*U'
U       = U*(D.^0.5);
% Vector normal Z con mu_Z=0 y C_Z=indentidad
z_noise = randn(2,M);
g_noise = U*z_noise;
% Señal trasmistida
rx_symbols = tx_symbols + g_noise;
```

La señal captada en el lado del receptor es el resultado de la suma entre los símbolos transmitidos y el ruido que ha sido coloreado debido a la multiplicación de una matriz U, la cual provoco una deformación en el ruido blanco gaussiano.

3 Receptor

Los siguientes receptores implementarán tres formas diferentes de decidir que símbolo fue transmitido a partir del símbolo recibido.

3.1 Receptor A

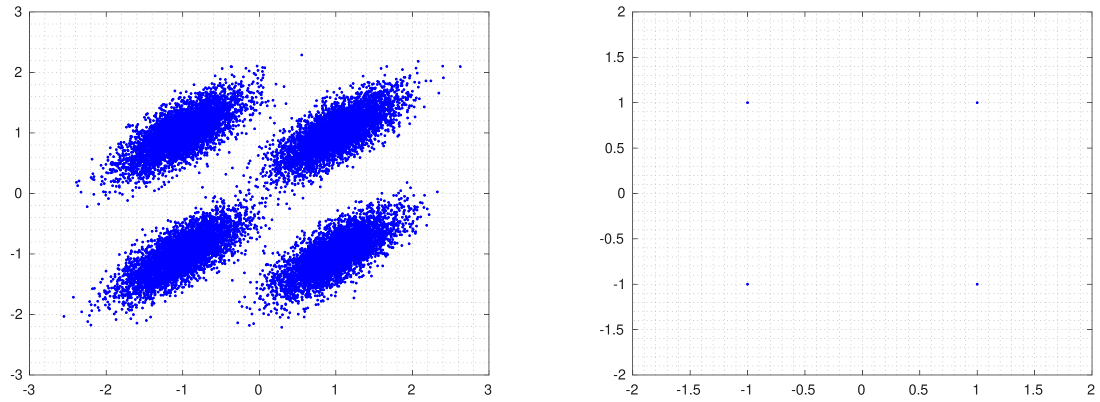
El receptor A elige como símbolo detectado a aquel con menor distancia Euclidiana al símbolo recibido. Es decir se procederá a evaluar el sumatorio del cuadrado de la diferencia entre la señal recibida (símbolos más el ruido) y una constelación de referencia con valores $[\pm 1, \pm j1]$.

```
% Receiver A
RxA_Const       = [1 1; 1 -1; -1 1; -1 -1];
RxA_dt_symbols  = detector(rx_symbols, RxA_Const);
RxA_er_symbols  = sum(sum(abs(RxA_dt_symbols - tx_symbols)) ~= 0);
RxA_SER(SNR_dB) = RxA_er_symbols/M
```

La función que implementa el detector para encontrar la menor distancia o error entre la señal recibida *rx_ symbols* y la de referencia *RxA_ Const* es:

```
dists2(n) = sum( (rx_symbols(:,k) - const(n,:)' ).^2 );
```

Una vez que la señal recibida pase por la función *detector*, se procede a calcular los errores producidos para diferentes valores de *SNR*.



(a) Constelación enviada al receptor (b) Constelación de referencia

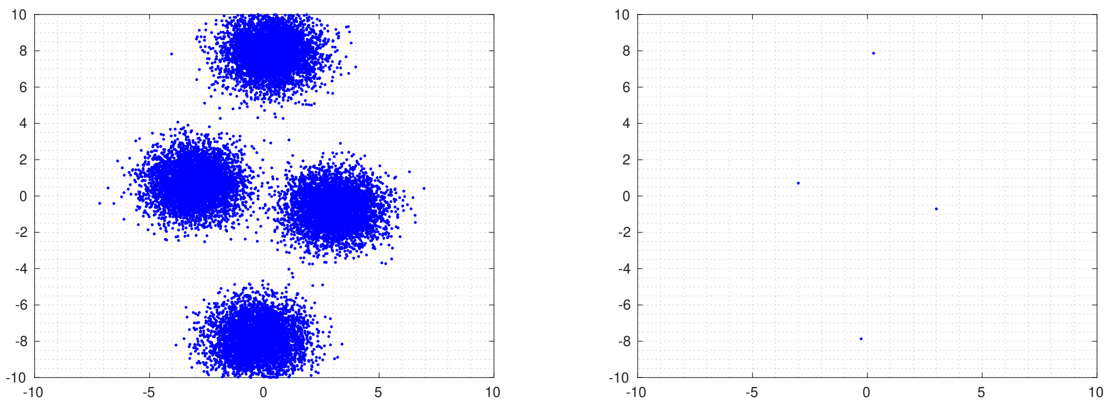
Figure 1: *Constelaciones a detectar*

3.2 Receptor B

El receptor B ecualiza los símbolos recibidos multiplicando a los mismos por la matriz inversa de U igual a: $U \cdot \sqrt{D}$, que se la obtiene al multiplicar la matrices U y D producidas por descomposición en valores singulares de la matriz de covarianza C_G.

$$S_{EQ} = U^{-1} \cdot S_{RX} = U^{-1} \cdot (S_{TX} + U \cdot Z) = U^{-1} \cdot S_{TX} + Z \tag{7}$$

Esto último elimina la correlación del ruido (convirtiéndolo en ruido gaussiano normalizado) pero modifica la posición de los símbolos de referencia según la transformación $S_{TX}^{(U)} = U^{-1} S_{TX}$. El proceso de detección consiste, al igual que el receptor A, en determinar cual es el símbolo $S_{TX}^{(U)}$ con menor distancia Euclidiana al símbolo S_{EQ}.



(a) Constelación Ecualizada S_{EQ} (b) Constelación de referencia $S_{TX}^{(U)}$

Figure 2: *Constelaciones a detectar*

El siguiente código en Matlab describe el proceso de ecualización y detección mencionado anteriormente. Una vez detectado los símbolos ecualizados, es necesario multiplicarlos de nuevo por la Matriz U para volver a llevarlos a su posición original y de esta manera lograr calcular el error con respecto a los símbolos generados originalmente *tx_symbols*.

```
% Receiver B
RxB_Const          = (inv(U)*(RxA_Const'))';
RxB_eq_rx_symbols = inv(U)*rx_symbols;
RxB_eq_dt_symbols = detector(RxB_eq_rx_symbols, RxB_Const);
RxB_dt_symbols     = U * RxB_eq_dt_symbols;
RxB_er_symbols     = sum(sum(abs(RxB_dt_symbols - tx_symbols)) > 1e-10);
RxB_SER(SNR_dB)    = RxB_er_symbols/M
```

3.3 Receptor C

El receptor C elige el símbolo transmitido que posea mayor probabilidad de haber sido transmitido dado el símbolo recibido. Para realizar este cálculo es necesario utilizar la PDF conjunta de G (ruido gaussiano que ha sido modificado por la matriz U). La función de distribución de probabilidad esta definida como:

$$f_{\bar{X}}(\bar{x}) = (2\pi)^{-n/2}[det(C)]^{-1/2} exp\left[-\frac{1}{2}(\bar{x}-\mu)^H C^{-1}(\bar{x}-\mu)\right]$$

Por medio de Matlab se realiza un a función de detección optima que evalúa solo el exponente de la función $f_{\bar{X}}(\bar{x})$, en este caso $\bar{X}$ representará el el vector aleatorio de ruido G, y C la matriz de covarianza del ruido.

```
% Calculo de la PDF
x          = rx_symbols(:,k) - const(n,:)';
dists2(n) = x' * invC * x;
```

La señal recibida por el receptor C es igual a la recibida por el receptor A que luego es comparado con los símbolos de referencia [1 1; 1 -1; -1 1; -1 -1], la única diferencia consiste en la forma de detección. Evaluando los tres receptores en una única gráfica:

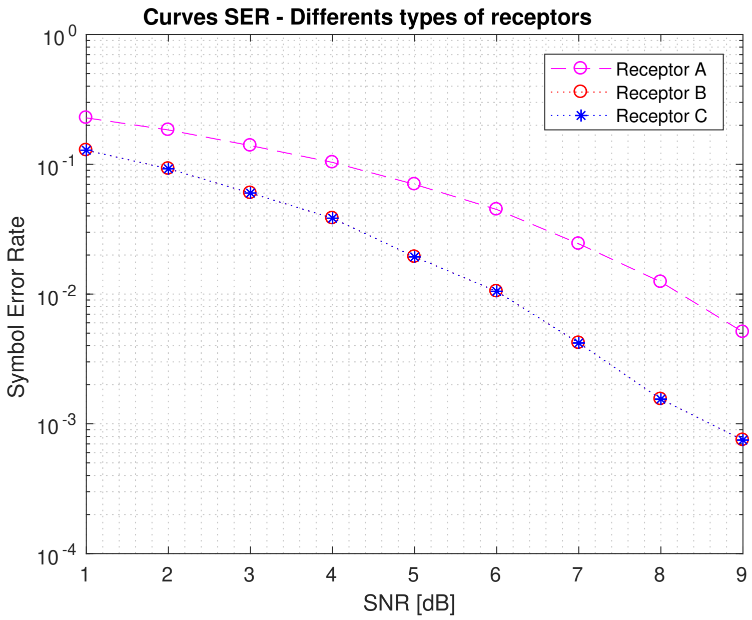


Figure 3: *Symbol Error Rate de los tres receptores*

Anexo

• Una descomposición en valores singulares de una matriz A es una factorización $A = U\Sigma V^T$, donde U es matriz unitaria ($U.U^T = I$), Σ es matriz diagonal y V es matriz unitaria, a U y V también se las llama matrices ortogonales. Una Matriz Ortogonal es aquella matriz que multiplicada por su traspuesta da como resultado la matriz identidad o unidad: $M.M^T = I$.
• Interpretación geométrica de los valores singulares: consideremos una esfera unitaria S en R^2, la transformación ortogonal V sólo produce la rotación de la esfera S, Σ la deforma en una hiperelipse y finalmente U rota la hiperelipse.
• La matriz de covarianza C asociada a un vector $\bar{X}$.

$$C_{\bar{X}} = \begin{bmatrix} \sigma^2_{X_1} & \sigma^2_{X_1,X_2} \\ \sigma^2_{X_2,X_1} & \sigma^2_{X_2} \end{bmatrix} \tag{8}$$

• Sea $\bar{X} \sim \mathcal{N}(\mu, \mathbf{C})$, donde $\bar{X} = [X_1, X_2, \ldots, X_n]^T$ es un vector aleatorio de media μ y covarianza $\mathbf{C}$. Con función de distribución de densidad:

$$f_{\bar{X}}(\bar{x}) = (2\pi)^{-n/2}[det(C)]^{-1/2} exp\left[-\frac{1}{2}(\bar{x}-\mu)^H C^{-1}(\bar{x}-\mu)\right] \tag{9}$$

• La distancia euclidiana entre los puntos $P = (p_1, p_2, \ldots, p_n)$ y $Q = (q_1, q_2, \ldots, q_n)$, del espacio euclídeo n-dimensional, se define como:

$$d_E(P, Q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2} \tag{10}$$

Una vista matemática para calcular la mínima distancia en el receptor A:

$$\sum(rxA - SymbComp')^2$$

Para el caso del receptor C, se emplea sólo el exponente de la función de distribución de probabilidad:

$$(rxC - SymbComp')' \cdot inv(C_G) \cdot (rxC - SymbComp')$$