CURSO:

CODIFICACIÓN DE FUENTE Y CANAL (EL271) 2023-01

LABORATORIO DE SIMULACIÓN 2

Profesor:

Dr. Carlos Rafael Valdez Velasquez Lopez

Integrantes:

Eliu Bonilla Nuñez

Walter Ricardo Avila Camacho

Anthony Gabriel Muchaypiña Palomino

Sección: EL85

LIMA - Perú

2023

I. Teoría

I.I Introducción

En el presente trabajo se llevará a cabo la simulación de la etapa que corresponde a la codificación y decodificación de canal con un codificador de bloque, se calculará el BER (Bit Error Rate) correspondiente a diferentes valores de relación señal a ruido γb, y se comparará estos resultados con los valores teóricos de las probabilidades de error Pb del sistema codificado, y Pub del sistema no codificado. Para este laboratorio se empleó un modulador BPSK para ambos casos.

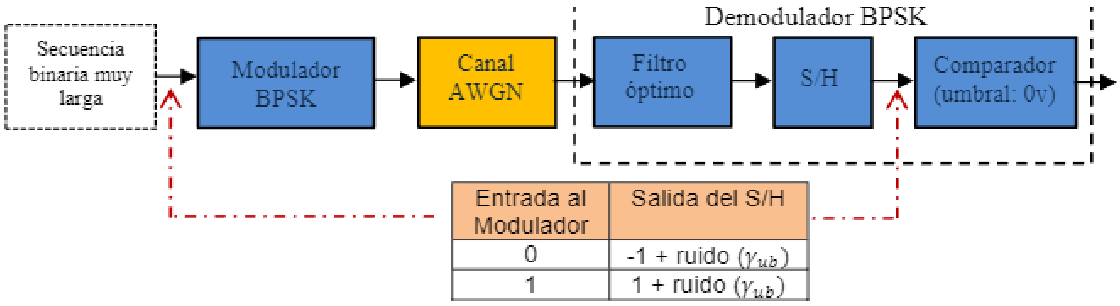


Figura 1. Diagrama de bloques del sistema de comunicación no codificado con modulador BPSK

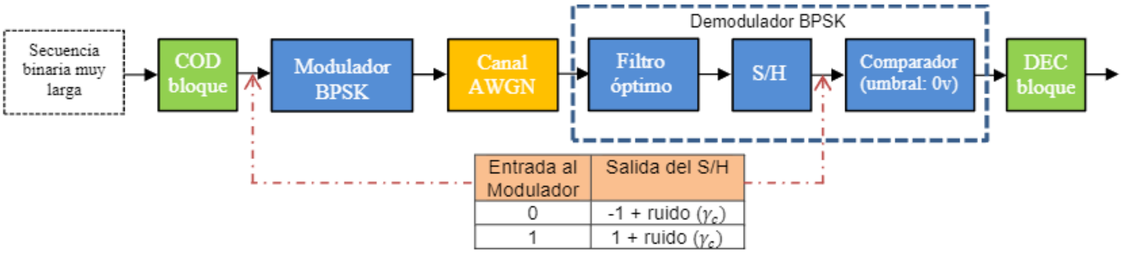


Figura 2. Diagrama de bloques con modulador BPSK y codificador de bloque

I.II Codificación de canal

En los sistemas de comunicaciones hay dos causas principales que influyen en el deterioro de la señal recibida. La primera es el ruido introducido por el propio canal de comunicaciones. Para el desarrollo del presente laboratorio se empleó un canal AWGN (ruido blanco aditivo gaussiano), el cual contiene ruido produciendo errores de transmisión, que hacen que la señal reconstruida por el receptor no sea la misma que la señal transmitida. La fidelidad de la transmisión de información se mide en términos de la probabilidad de error (tasa de error), es decir, la probabilidad de que el símbolo a la salida del receptor sea diferente al símbolo transmitido.

Tanto los sistemas de transmisión como de grabación y reproducción están sujetos a errores. En la transmisión y grabación de señales digitales de audio, vídeo o datos en general, igual que en cualquier sistema digital de comunicaciones, la secuencia binaria recibida o reproducida debe ser, en la medida posible, igual que el transmitido o grabado, por lo que la información debe protegerse al máximo contra las degradaciones que inevitablemente introduce el medio de transmisión o los circuitos de procesado de la señal.

Por los motivos explicados, uno de los objetivos de cualquier sistema de comunicación debe ser lograr el mínimo de degradación en una transmisión. Por ejemplo, si en una empresa se tiene una relación señal a ruido y una probabilidad de error inaceptable, se puede optar por dos opciones Una opción es aumentando la potencia de la señal lo cual resulta ser muy costoso, o se puede codificar la señal lo cual resulta ser mucho más económico y una técnica más "elegante".

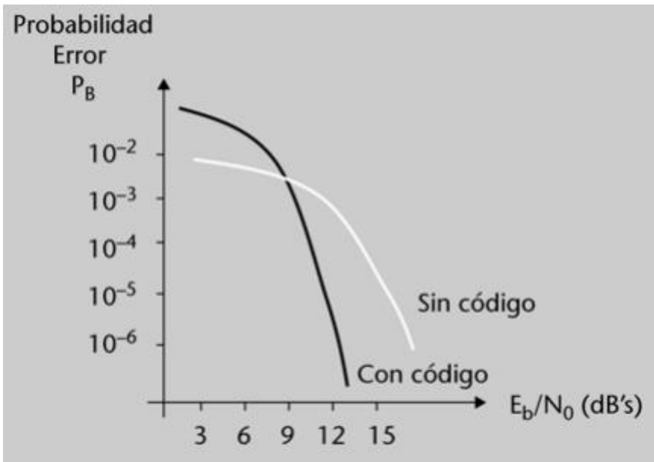


Figura 3. Probabilidad de error de un sistema codificado y sin codificar

A través de la codificación es posible la detección y corrección de errores, esto se logra agregando bits redundantes, los cuales no contienen ninguna información, solo sirven para proteger al mensaje de cualquier tipo de ruido ocasionados por el canal.

En el presente laboratorio se usará un modulador BPSK binario, por ello para hallar la probabilidad de error en codificación de bloque se obtiene de la siguiente manera:

$$P_b \approx \binom{n-1}{t} \alpha^{t+1}$$

Probabilidad de error con codificación de bloque junto con un modulador BPSK

$$P_{ub} = Q(\sqrt{2\gamma_{ub}})$$

Probabilidad de error sin codificación de bloque junto con un modulador BPSK

I.IV Probabilidad de error del modulador BPSK

El modulador BPSK (Binary Phase Shift Keying, en inglés) es una técnica de modulación digital que se utiliza comúnmente en sistemas de comunicación. La probabilidad de error en un sistema BPSK depende de varios factores, como la relación señal-ruido (SNR, por sus siglas en inglés), la presencia de interferencia y el ruido en el canal de comunicación. La probabilidad de error, generalmente, se expresa como la probabilidad de que se reciba un símbolo transmitido incorrecto.

La probabilidad de error para un sistema BPSK ideal en un canal AWGN se puede calcular utilizando la función $Q(x)$, que es la función de distribución acumulativa de una variable normal estándar. Por el momento, aún se aplica la codificación a dicho sistema. En un canal AWGN, la probabilidad de error para el sistema mencionado se puede aproximar utilizando la siguiente expresión:

$$\alpha = Q(\sqrt{2\gamma_c})$$

Probabilidad de error para el sistema BPSK (sin codificar)

Se pude deducir que $x = \sqrt{2} \cdot \gamma_c$.

La función $Q(x)$ se relaciona con la función de error complementario erfc(·), de la siguiente manera:

$$Q(x) = \frac{1}{2} \cdot erfc\left(\frac{x}{\sqrt{2}}\right)$$

I.III Canal AWGN y BER

El ruido AWGN se caracteriza por ser blanco, lo que significa que su espectro de potencia es uniforme en todo el ancho de banda; y gaussiano, lo que implica que sigue una distribución normal. Este tipo de ruido se utiliza a menudo como una aproximación realista del ruido en muchos sistemas de comunicación, incluyendo la transmisión inalámbrica y por cable.

La BER es una medida de la cantidad de bits erróneos con respecto a un número total de bits transmitidos. En presencia de ruido AWGN, la probabilidad de error de bit depende de la relación señal a ruido (SNR, por sus siglas en inglés), que es la relación entre la potencia de la señal recibida y la potencia del ruido.

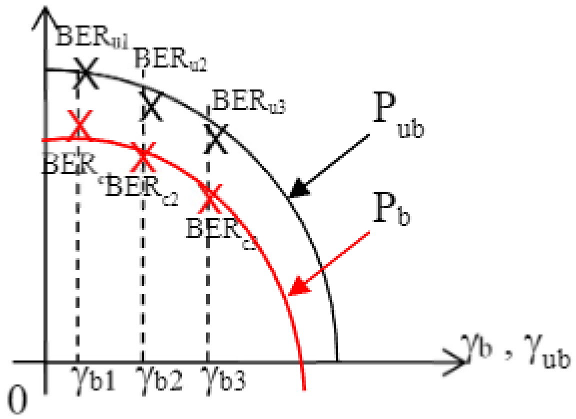


Figura 3. P_b y BER_c (c: coded), P_{ub} y BER_u (u: uncoded)

En el caso sin codificar, para generar la fuente de ruido gaussiano se halló la varianza para cada valor de gamma (relación señal-ruido). Se utilizó un rango de 0 a 10 dB con pasos de 2 dB. Por cada gamma, se calculó un determinado BER. Por otro lado, en el caso codificado se aplica la misma estrategia.

$$\sigma^2 = 1/2\gamma_{\mathrm{ub}}$$

Relación de la varianza y γub

$$BER = \frac{N^{\circ}\text{ de bits errados (compare la secuencia entrante al MOD con la secuencia de salida del comparador)}}{N^{\circ}\text{ total de bits transmitidos}}$$

Calculo del BER

$$\sigma^2 = 1/2\gamma_{\mathrm{b}}$$

Relación de la varianza y γ_{b}

Algoritmo de Código Hamming (31,26):

1. La palabra resultante de este código tendrá 31 posiciones desde la posición 1 hasta la posición 31. Los bits que pertenezcan a las posiciones que sean potencias de 2 serán utilizados como bits de paridad. Los bits del resto de posiciones serán utilizados como bits de datos.
2. Al tener un bloque de mensaje de 26 bits y una palabra codificada de 31 bits, se obtiene 5 bits redundantes. Por lo tanto, obtendrá un bloque de código de 5 bits. Los bits de código serán denominados como C1, C2, C3, C4 y C5. C1 sería el bit más significativo y el C5 sería el bit menos significativo en el vector de código.

3. Para hallar C1, partiendo desde el bit de paridad 1, se selecciona un bit y se deja un bit y así sucesivamente hasta llegar a la última posición de la palabra codificada. Al terminar de seleccionar los bits, se procede a realizar una suma XOR de todos esos bits. El resultado de la suma XOR será el valor para el bit C1. Entonces, para hallar C2, se realiza el mismo proceso, pero con ciertas diferencias. En lugar de partir desde el bit de paridad 1 seleccionando un bit y dejando un bit, se selecciona dos bits y se deja dos bits y así sucesivamente a partir del bit de paridad 2. Luego, se procede con la suma XOR para obtener C2. La regla es: Para Cx, se selecciona 2^(x-1) bits y se deja x bits. Sólo se seleccionan bits de datos a partir del bit de paridad x. Además, los bits de datos también pueden ser llamados bits de mensaje (dx->mx).

1	2	3	4	5	6	7	8	9	10
p1	p2	d1	p3	d2	d3	d4	p4	d5	d6

11	12	13	14	15	16	17	18	19	20
d7	d8	d9	d10	d11	p5	d12	d13	d14	d15

21	22	23	24	25	26	27	28	29	30	31
d16	d17	d18	d19	d20	d21	d22	d23	d24	d25	d26

px: bit de paridad

dx: bit de dato

Leyenda:

Cx	Color
C1	
C2	
C3	
C4	

C5	

C1 = d1 xor d2 xor d4 xor d5 xor d7 xor d9 xor d11 xor d12 xor d14 xor d16 xor d18 xor d20 xor d22 xor d24 xor d26

C2 = d1 xor d3 xor d4 xor d6 xor d7 xor d10 xor d11 xor d13 xor d14 xor d17 xor d18 xor d21 xor d22 xor d25 xor d26

C3 = d2 xor d3 xor d4 xor d8 xor d9 xor d10 xor d11 xor d15 xor d16 xor d17 xor d18 xor d23 xor d24 xor d25 xor d26

C4 = d5 xor d6 xor d7 xor d8 xor d9 xor d10 xor d11 xor d19 xor d20 xor d21 xor d22 xor d23 xor d24 xor d25 xor d26

C5 = d12 xor d13 xor d14 xor d15 xor d16 xor d17 xor d18 xor d19 xor d20 xor d21 xor d22 xor d23 xor d24 xor d25 xor d26

C = [C1 C2 C3 C4 C5]

Se recuerda que los bits de datos se consideran los bits de mensaje (dx -> mx). Se sabe que el bloque de mensaje debe tener la siguiente estructura:

M = [m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 m12 m13 m14 m15 m16 m17 m18 m19 m20 m21 m22 m23 m24 m25 m26]

El M tiene una dimensión 1x26 y C tiene una dimensión 1x5. Entonces, P debería tener una dimensión 26x5 conteniendo 130 bits.

$$P = \begin{bmatrix} p1 & p2 & p3 & p4 & p5 \\ p6 & p7 & p8 & p9 & p10 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ p126 & p127 & p128 & p129 & p130 \end{bmatrix}$$

En este caso, los px representan a los bits dentro de la matriz P.

Entonces, para que se cumpla C=MxP, P se forma de la siguiente manera:

$$P = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

De esta forma, al multiplicar M por P, se obtendrán las mismas sumatorias xor para C1, C2, C3, C4 y C5 observadas anteriormente.

C = [C1 C2 C3 C4 C5]

C = [**d1 xor d2 xor d4 xor d5 xor d7 xor d9 xor d11 xor d12 xor d14 xor d16 xor d18 xor d20 xor d22 xor d24 xor d26 , d1 xor d3 xor d4 xor d6 xor d7 xor d10 xor d11 xor d13 xor d14 xor d17 xor d18 xor d21 xor d22 xor d25 xor d26 , d2 xor d3 xor d4 xor d8 xor d9 xor d10 xor d11 xor d15 xor d16 xor d17 xor d18 xor d23 xor d24 xor d25 xor d26 , d5 xor d6 xor d7 xor d8 xor d9 xor d10 xor d11 xor d19 xor d20 xor d21 xor d22 xor d23 xor d24 xor d25 xor d26 , d12 xor d13 xor d14 xor d15 xor d16 xor d17 xor d18 xor d19 xor d20 xor d21 xor d22 xor d23 xor d24 xor d25 xor d26]**

Con ello, ya se puede obtener los 5 bits redundantes para cada bloque mensaje de 26 bits.

II. Código del programa

El código del programa se adjuntó junto con este informe. Fue desarrollado en MATLAB. Tiene el nombre de LAB2v2.m .

III. Gráficos

1. Sin codificar:

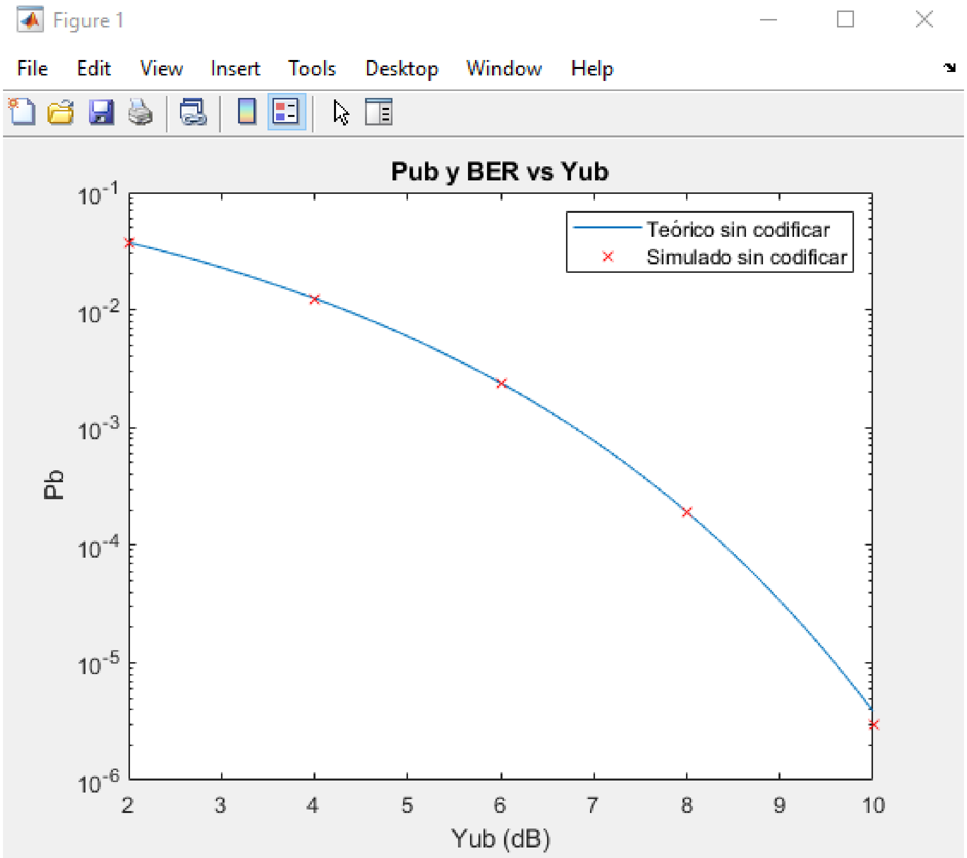


Figura 1 Gráfica N°1 Pub y BER vs Yub

Respecto a esta figura, se puede visualizar la curva teórica Pub vs. Yb (gamma b) y la curva (formada por 5 aspas) BER vs. Yb proveniente de la simulación. Estas curvas se obtuvieron al analizar (teóricamente) y simular un sistema de comunicación sin usar ningún codificador. Para la curva teórica, se utilizó un rango continuo de gamma Yub de 2 dB a 10 dB. Para la curva de la simulación, se utilizó valores puntuales de Yub para obtener determinados valores de BER. Se recuerda que la probabilidad de error (Pub) es igual a BER. La curva teórica presenta una curvatura descendente a medida que el gamma (relación señal a ruido) asciende. Se puede observar que un gamma bajo ofrece una probabilidad de error más alto (mayor error en la palabra codificada a la salida del demodulador) y un gamma alto otorga una probabilidad de error menor (menor error en la palabra codificada a la salida del demodulador). Los valores de Pb para los gamma b utilizados en la curva de simulación deberían ser iguales o cercanas a los valores Pb de la curva teórica. Esto demuestra que la teoría es muy cercana a lo que se podría ver en una simulación o implementación.

2. Con codificar:

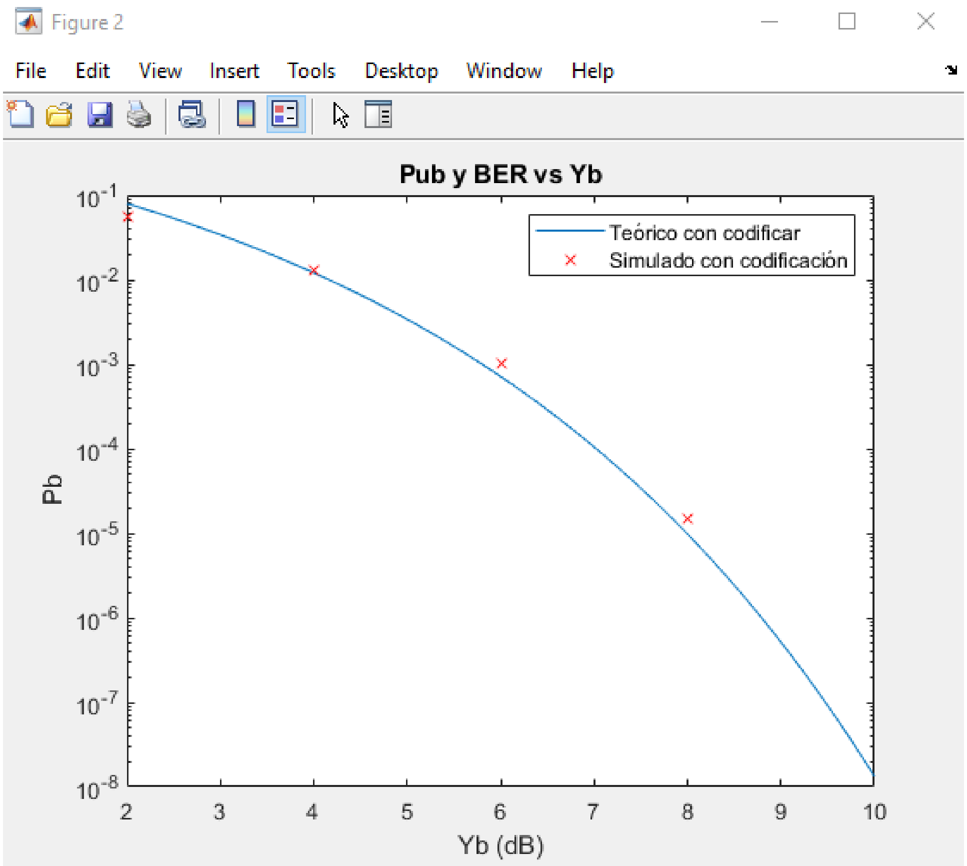
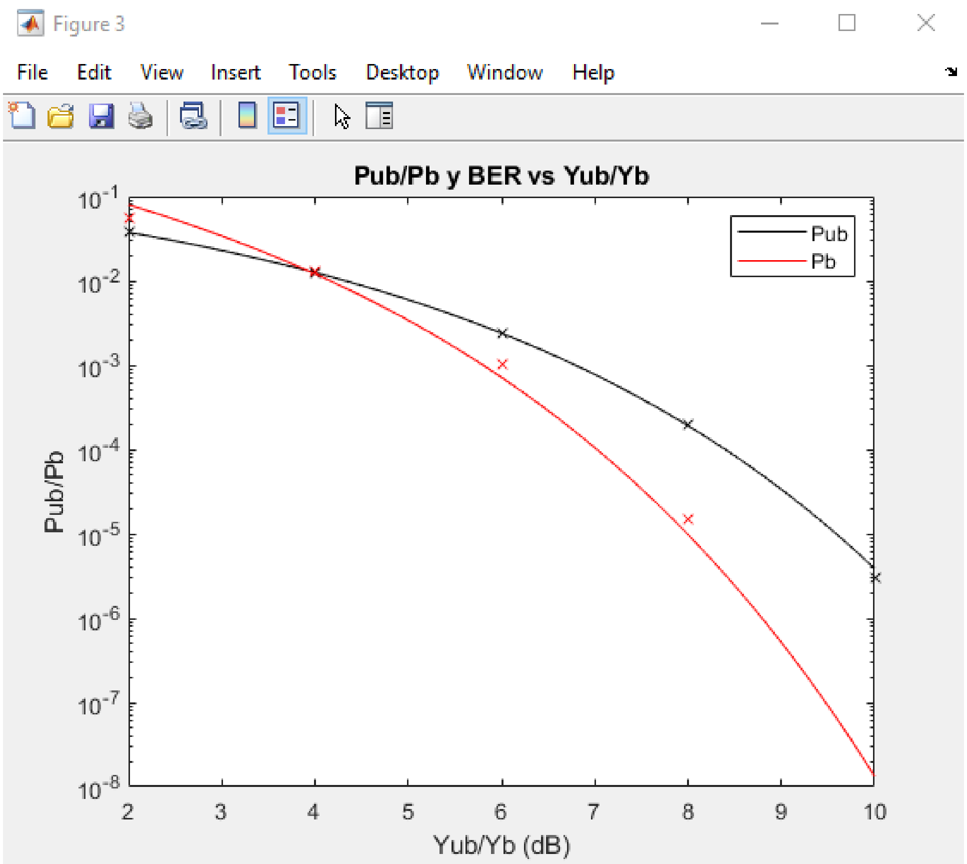


Figura 2 Grafica N°2 Pub y BER vs Yub

La figura incluye las mismas curvas (la teórica y de simulación). Sin embargo, hay variaciones. Según las curvas, a partir de un gamma b igual a 4 dB, se puede obtener un Pb más pequeño utilizando valores de SNR (gamma b) más pequeños. Esto indica que sistema requeriría menor potencia para transmitir datos con una mayor confiabilidad, lo cual supone un gasto energético menor (menor gasto monetario). Esto es gracias al codificador de canal Hamming (31,26) debido a que es capaz de detectar y corregir errores simples. Es cierto que puede haber errores dobles, errores triples o más. Sin embargo, los errores simples son mucho más frecuentes (tiene mayor probabilidad de ocurrencia) que los errores dobles, triples, etc. Respecto a esta segunda gráfica es similar a la primera con ligeras variaciones. Como comentario adicional, en esta figura, las variaciones entre los datos simulados y teóricos son más evidentes que en la figura anterior. Se logró demostrar que la probabilidad de error en un sistema codificado (capaz de corregir errores) siempre será mayor que un sistema sin ningún tipo de codificación

Uniendo ambos gráficos, se puede apreciar mejor la diferencia:



IV. Conclusiones

- Un sistema codificado capaz de detectar y corregir errores suele tener una menor probabilidad de error a comparación de un sistema sin codificar.
- La cercanía (o margen de error) entre valores teóricos y simulados permite verificar la teoría aplicada.
- Una codificación capaz de corregir errores simples es la primera opción a utilizar debido a que la probabilidad de ocurrencia de errores simples es mucho mayor que errores dobles, triples, etc.
- La matriz P perteneciente a una determinada codificación de canal es la base para generar los bits redundantes (vector de código C) para los bloques de mensaje de k bits. Por otro lado, las matrices Iq y H^T permiten realizar la técnica del Síndrome para localizar los posibles errores simples dentro de la palabra codificada recibida.

- Se puede observar cómo varía el BER a medida que se modifican los valores de la relación señal a ruido. En general, a medida que la SNR aumenta, el BER disminuya, lo que indica una mejor calidad de la transmisión.

- La simulación permite evaluar el rendimiento del codificador de bloque utilizado en términos de la capacidad de corrección de errores. En la gráfica se logra observar como el codificador de bloque logra reducir la tasa de errores en comparación con el sistema no codificado.

- Este sistema permite solo corregir 1 bit errado en la palabra codificada. Por lo tanto, si el mensaje se encuentra en un canal con un gran ruido es posible que se generen 2 o más bits errados porque este sistema podría no resultar optimo, por lo cual, se debería considerar utilizar otros métodos de codificación como Hamming extendido.

- Se pudo evidenciar que antes un nivel de gama mejor de 4dB, la probabilidad de error del sistema codificado era un poco mayor frente al sin codificación. Esto es un poco contradictorio considerando lo que se explica en la teoría, pero es un fenómeno que puede ocurrir; a partir del gama de 4dB los valores de probabilidad de error del sistema codificado si alcanzaron valor muy por debajo frente al sistema sin codificar.