

Canal de adición de ruido blanco Gaussiano y bloque para cálculo de SER

Grupo B1B.g4

Castillo Bohorquez Jeyson Arley

Mancipe Arias Henry Mancipe

Padilla Arrieta Astrid Carolina

Universidad Industrial de Santander

Bucaramanga, Santander, Colombia

11 de mayo de 2018

Resumen

En toda transmisión de datos, sea analógica o digital, se presentan no idealidades, dadas por las condiciones en las cuales se realiza la transmisión, principalmente por el medio por el cual se transmite, estas no idealidades se generalizan en este caso a la presencia de ruido gaussiano blanco en el canal. En transmisiones digitales una forma de medir la calidad de la información recibida, es por medio de la tasa de error binario, la cual permite comparar distintos tipos de modulaciones según la probabilidad de error que estos presenten. En este trabajo se presenta documentación acerca del bloque QT GUI Bercurve para generar las curvas de probabilidad de error en la transmisión de bits con respecto a la amplitud del ruido blanco presente en el canal.

1. Introducción

La tasa de errores de bits es un registro empírico (histórico) del funcionamiento real del sistema en cuanto a errores. Una tasa de error de bits se mide y a continuación se compara con la probabilidad esperada de error, para evaluar el desempeño de un sistema. El ruido del canal se modela como ruido blanco aditivo de media cero y densidad espectral $N_0/2$. Dado que se modela de esta manera, es posible caracterizar una variable aleatoria. [1]

La probabilidad de error es una función de la relación de potencia de portadora a ruido (o en forma más específica, el promedio de la relación de energía por bit entre la densidad de potencia de ruido) y de la cantidad de condiciones posibles de codificación que se usan (M-aria). La relación de potencia de portadora a ruido es la de la potencia promedio de la portadora (la potencia combinada de la portadora con sus bandas laterales asociadas) entre la potencia del ruido térmico. En particular, la probabilidad de error disminuye muy rápido cuando el cociente E_0/N_0 aumente, por lo cual un aumento pequeño en la energía de la señal transmitida hará que la recepción de datos binarios tenga

muchos menos errores.

La relación de energía por bit a densidad de potencia de ruido se usa para comparar dos o más sistemas digitales de modulación que usen distintas rapidezces de transmisión (frecuencias de bits), esquemas de modulación (FSK, PSK, QAM) o técnicas M-arias de codificación. La relación de energía por bit a densidad de potencia de ruido es tan sólo la relación de la energía de un solo bit a la potencia de ruido presente en 1 Hz de ancho de banda. Así, E_b/N_0 normaliza todos los esquemas multifásicos de modulación a un ancho de banda común, permitiendo una comparación más sencilla y más exacta de su desempeño con errores. [2]

2. Desarrollo en GNU

2.1. Bloque de adición de ruido

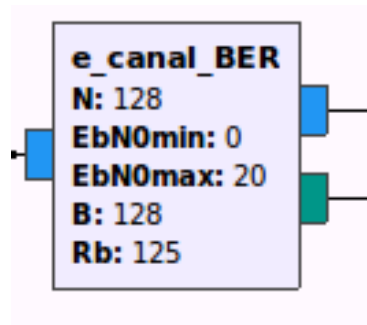


Figura 1: Bloque OOT e canal BER: añade ruido blanco Gaussiano a la señal

La función principal del bloque es agregar ruido AWGN con diferentes valores de potencia promedio a partir del numero de datos que el usuario desea, usando para el calculo los valores de la relación E_b/N_0 ingresado por el usuario. A su vez el bloque entrega el valor de SNR que esta siendo usado en ese momento. El bloque internamente detecta el numero de datos enviados en el paquete y posteriormente convierte los valores de E_b/N_0 a escala lineal y los transforma en valores de relación señal a ruido. Usando estos valores aplica un valor de ruido a cada uno de los datos de la señal.

El bloque tiene una sola entrada; esa entrada es la envolvente compleja de la señal codificada transmitida. A su vez el bloque se compone de dos salidas; por la Salida 1 (Out1 - verde) sale el valor de E_b/N_0 que fue añadido a cada muestra, y la Salida 0 (Out0 - azul) es la salida de la envolvente completa de la señal transmitida mas un ruido AWGN. Los parámetros del bloque son:

- N: representa el numero de datos que el usuario desea calcular en ese rango.
- E_b/N_{0min} : es el valor mínimo definido por el usuario de la relación entre la energía del bit y la potencia promedio del ruido.
- E_b/N_{0max} : es el rango superior definido por el usuario de la relación entre la energía del bit y la potencia promedio del ruido.

- B: es el valor del ancho de banda de la señal el cual esta definido por el tiempo de bits. para el caso de GNU radio como la fuente random source, $SampRate/2$
- Rb: es la rata de bits que ingresa al bloque.

2.2. Bloque de cálculo de la SER

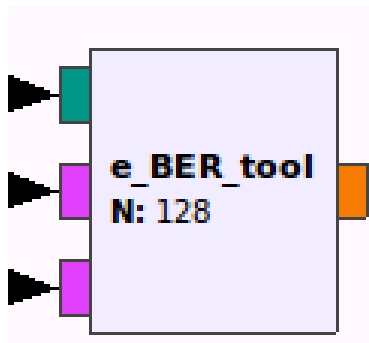


Figura 2: Bloque OOT e BER tool: calcula la SER para el rango de valores de EbN0 que el usuario definió

El bloque sirve para calcular la SER para el rango de valores de EbN0 que el usuario definió. La Entrada 0 (In0 - verde) del bloque recibe el valor de EbN0 actual que usa el canal para imprimir el ruido en la señal. La entrada entrada 1 (In1 - morada) recibe los bit transmitidos. La entrada 2 (In2 - morada) es la entrada que recibe la retroalimentación de los bits recibidos. El bloque consta de una única salida tipo float y representa la salida del valor actual de la probabilidad de error basado en el calculo de los bits enviados y transmitidos. El bloque solo necesita conocer el número de puntos que se están calculando, es decir, N, el cual lo ingresa el usuario. Es importante que este valor sea igual para los dos bloques y no se generen errores en la gráfica de la curva.

2.3. Instalación de los bloques

Desde la terminal, se abre la ubicación de la carpeta gr-SERcurvas2, se abre la carpeta build y desde la misma se ingresa el comando para la instalación de los bloques. A continuación se muestra la lista de comandos:

```
cd gr-SERcurvas2
cd build
cmake ../ && make
sudo make install
sudo ldconfig
```

Luego de instalados los bloques, se actualiza el entorno en GNURadio con el botón actualizar para que los bloques aparezcan listados en la barra ubicada a la derecha del entorno de GNURadio, en la carpeta SERcurvas2.

2.4. Creación del bloque Jerárquico para añadirlo en librería E3TRadio

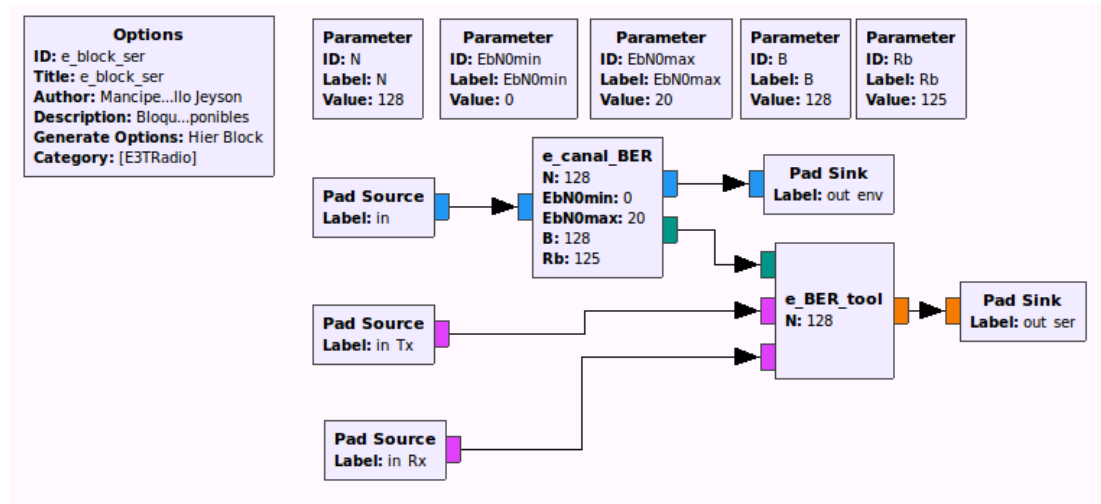


Figura 3: Bloque OOT que calcula el error en escala logarítmica

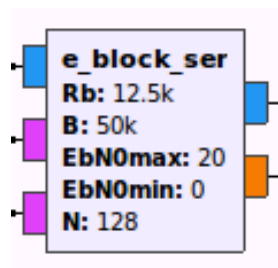


Figura 4: Bloque jerárquico para el cálculo de los puntos de la curva de SER

Figura 5: Parámetros a ingresar por el usuario en el bloque e_block_ser

3. Flujograma de prueba

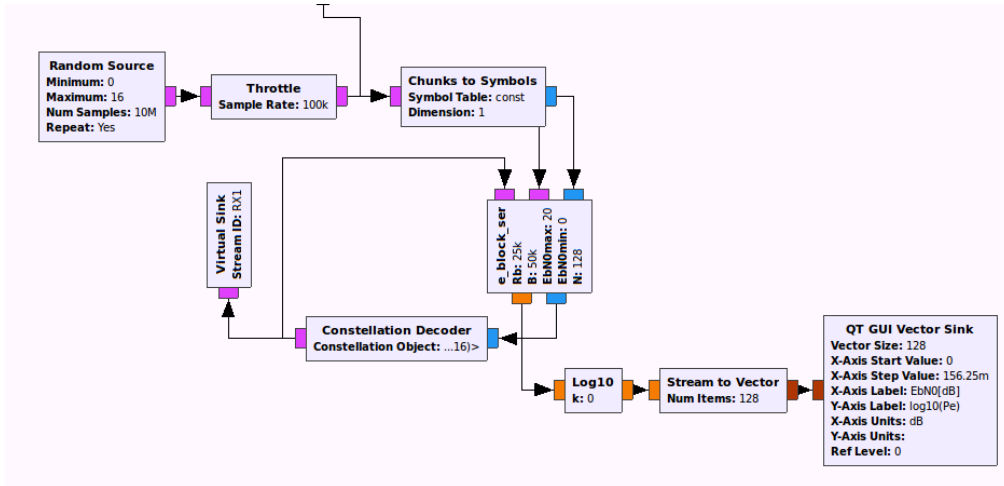


Figura 6: Flujograma para la implementación del bloque Jerarquico que calcula los puntos de la SER para una modulación determinada por el usuario

En este flujograma se genera a partir de un Random Source una señal binaria aleatoria, la cual se modula y pasa por el bloque e block SER, el cual calcula el número de puntos que desea el usuario de la curva de SER para la modulación escogida. El bloque presenta una realimentación, dado que este adiciona ruido blanco a la señal transmitida, emulando el ruido que en la vida real es añadido a una señal que se transmite por un canal, esta señal con ruido blanco adicionado sale del bloque para luego ser demodulada, la señal demodulada es una de las entradas del bloque, dado a que es necesaria para compararse con la señal transmitida y así comparar la probabilidad de error en la transmisión para distintos valores de EbN0. Del bloque salen cada uno de los valores de la probabilidad de error, los cuales para efectos de graficación son sometidos a escala logritmica con el bloque log10, luego se convierten los datos de Stream to Vector para poder graficarlos desde un QT GUI Vector Sink, el cual permite modificar los ejes y así obtener Pe vs EbNo (dB).

Para hacer uso de este flujograma de prueba, el usuario solo debe habilitar la variable que corresponde a la modulación de la cual desea visualizar la curva de SER y deshabilitar las otras, como se puede ver en la figura 7.

Variable ID: const Value: -1, 1	Note Note: Modulación BPSK
Variable ID: const Value: digital.qam constel...	Note Note: Modulación 16QAM
Variable ID: const Value: 1, 1j, -1, -1j	Note Note: Modulación 4PSK

Figura 7: Modulaciones para visualización de curvas de SER

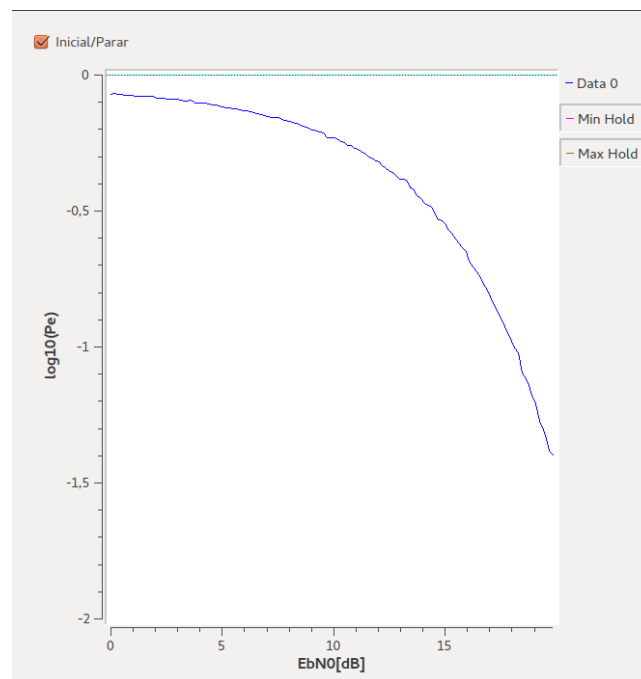


Figura 8: Prueba modulación 16-QAM

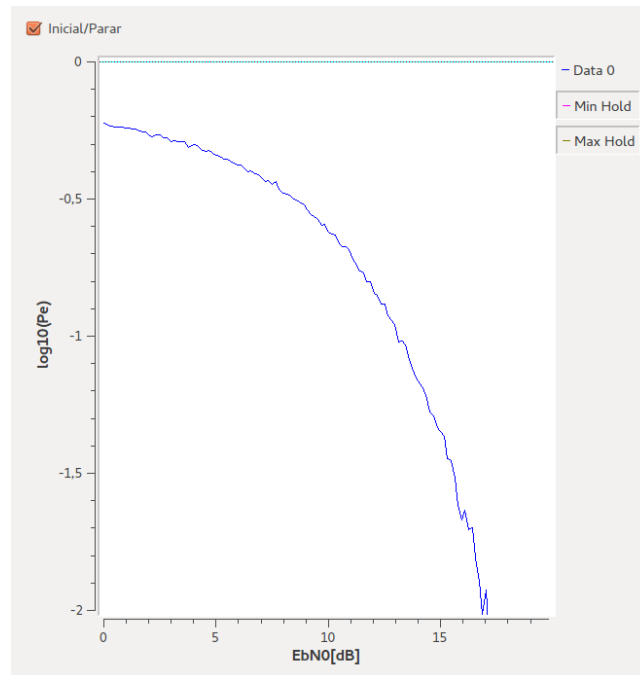


Figura 9: Prueba modulación 4PSK

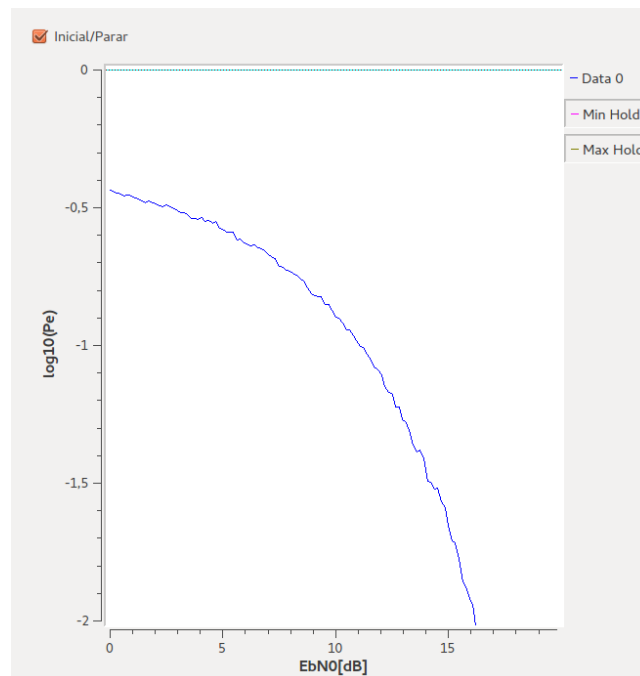


Figura 10: Prueba modulación BPSK

4. Anotaciones

- Para la traducción de EbN0 a SNR se hizo uso de la ecuación $SNR = EbN \cdot \frac{fb}{BW}$, donde fb es la tasa de transmisión de bits o Rb (para el caso de flujograma de prueba) y BW es el ancho de banda de la señal, por efectos de uso se escogió BW como $sample/2$.
- Es necesario revisar esa traducción, dado que aunque las curvas que se obtienen muestran un comportamiento similar a curvas de SER teóricas, al parecer se presenta un problema con el cálculo de la amplitud del ruido. Se hicieron pruebas con distintas formas de pasar de EbN0 a SNR, y en este trabajo se presenta la que mejor resultado tuvo.
- Para la creación del bloque jerárquico, fue necesario hacerlo de manera tradicional, como un flujograma, dado a que se presentaron problemas para la creación del mismo desde la herramienta de creación de módulos OOT, dado a que este bloque presenta dos tipos diferentes de entradas y dos tipos diferentes de salida, la estructura de bloques jerárquicos OOT (según la documentación disponible por el momento) no permiten combinar diferentes tipos de entradas ni salidas. Aún así, se requiere investigar un poco sobre el tema.

Referencias

- [1] W. TOMASI, “Sistemas de comunicaciones electrónicas, México: Pearson education, 2003.” 2003.
- [2] C. L. W, *Sistemas de Comunicaciones digitales y analogicos. Séptima edición.*, México: PEARSON EDUCATION, 2008.