TRABAJO PRÁCTICO 2

Procesos Estocásticos

Ejercicio 1 Dado el archivo *signal_exercise_1.mat*, subido en el Classroom junto con este TP, calcular y dibujar:

a) Media como función del tiempo

b) Varianza como función del tiempo

c) Función de autocorrelación $R_X(t1, t2) = E\{x(t_1)x^*(t_2)\}$

d) En base a los resultados obtenidos, ¿considera que el proceso es WSS? Justifique

La estructura de la señal es la siguiente: cada fila es un nuevo experimento, mientras que las columnas indican el indice de tiempo de cada experimento.

TIP: para calcular la autocorrelacion, barrer t_1 y t_2 para varios valores (posiblemente todos), y calcular la expresion de $R_X(t1, t2)$ como se muestra en el siguiente ejemplo:

```
t1=1; % indice de tiempo a barrer
term1=x(:,t1);
z = term1.*conj(x);
correlation_for_t1 = sum(z,1)/nexp;

plot(correlation_for_t1);
hold all
```

Finalmente, dibujar en la misma figura $R_X(t1, :)$ superponiendo todos los valores elegidos de t_1.

TIP 2: Notar que las correlaciones calculadas para distintos valores de t_1 podrían estar "desfasadas"temporalmente. Sin embargo, se considera que dos auto-correlaciones son iguales aun si existe entre ellas un desfasaje temporal, esto es $R_X(t1, t2) = R_X(t1, t2 + \Delta t)$

Ejercicio 2 Idem Ejercicio 1, pero usando la señal *signal_exercise_2.mat*

Ejercicio 3 Genere un proceso blanco gaussiano (WGN, White Gaussian Noise) $x(n)$ de 100k muestras de longitud, con media $\mu_X = 1$ y varianza $sigma_X^2 = 2$. Notar que el proceso WGN es estacionario y ergódico, con función de autocovarianza $C_X(m) = \sigma_X^2\delta(m)$. Notar que la función de autocovarianza es la función de autocorrelación $R_X(m) = C_X(m) + \mu_X^2$. Luego, filtrar el proceso $x(n)$ con el filtro pasa-bajos de media móvil realizado en el primer trabajo práctico, con un tamaño de ventana de 25 muestras.

a) Comprobar la expresión de $R_X(m)$ y $C_X(m)$ usando simulaciones

b) Comprobar por simulación que $R_Y(m) = R_X(m) * h(m) * h^*(-m)$

c) Calcular $E\{y(m)\}$ y justificar el valor que se obtiene

d) OPCIONAL. Si haces este, te convertis en héroe. Calcular por simulación $E\{|y(m) - \mu_Y|^2\}$ y justificar por qué da ese valor. Tip: demostrar matemáticamente el valor de $E\{|y(m) - \mu_Y|^2\}$ y encontrar la relación entre la varianza del proceso de entrada y del proceso de salida del filtro. Te va a ayudar bastante tener en cuenta que el proceso de entrada es blanco :).

TIP en GENERAL: tene en cuenta que los procesos en cuestión son estacionarios y ergódicos. Esto te va a simplificar el calculo de la media y la varianza.
TIP en GENERAL 2: usa como ejemplo el script *process_white_colored.m* que está subido al Classroom

Ejercicio 4 Usando las mismas señales del ejercicio anterior, hacé lo siguiente:

a) Computá la PSD del proceso de entrada $x(n)$ y verificá que la potencia del proceso es la integral debajo de la PSD

b) Computá la PSD del proceso de entrada $y(n)$ y verificá que la potencia del proceso es la integral debajo de la PSD. Además, verificá que $S_Y(f) = S_X(f)|H(f)|^2$

TIP en GENERAL: usa como ejemplo el script *psd_process_white_colored.m* que está subido al Classroom

Ejercicio 5 En un modelo de simulación de canal eléctrico se requiere generar ruido blanco gaussiano para simular el ruido que introduce un amplificador analógico. El ancho de banda de la señal de datos es de 1GHz, y se decide utilizar un factor de sobre-muestreo 8, es decir, el modelo de canal estará muestreando a $f_s = 8GHz$. Se sabe que la potencia del ruido térmico medido en el ancho de banda de 8GHz es de 100mW.

a) Generá en Matlab un vector de muestras WGN con la potencia adecuada y demostrá que el vector está correctamente generado usando una PSD. Es importante que consideres la frecuencia de muestreo en el dibujo de la PSD. ¿Qué valor numérico tiene la densidad de potencia de ruido para este caso?

b) Suponé que el usuario del modelo quiere llevar la frecuencia de muestreo a 16GHz. Repetí el inciso anterior y compará la PSD del ruido entre ambos casos. Interpretá el resultado

c) Suponé que en el ejercicio inicial, el usuario quiere modelar un ruido blanco gaussiano complejo de potencia 100mW. ¿Cómo hay que modificar el código para generar este nuevo vector de ruido?

Ejercicio 6 En los modelos de simulación de ruido blanco para sistemas analógicos, EN GENERAL, no se usa como dato de entrada la potencia total del ruido, sino su densidad espectral de potencia. Por ejemplo, en la hoja de datos del amplificador operacional que subo al Classroom se puede leer en la página 3 que el `Input Noise Voltage Spectral Density` es de $1nV/\sqrt{Hz}$. Suponé que quiero incluir este amplificador en un modelo de simulación. Asumí que el ruido indicado es blanco*. Considerá que el ancho de banda de la simulación es 50MHz.

a) Generá la señal de ruido necesaria para que se cumplan las condiciones especificadas anteriormente, y comprobá mediante PSD que los resultados están bien.

b) Si el ancho de banda de simulación sube a 100MHz, ¿qué cambios hay que hacer en la generación de ruido para que el modelo siga siendo fiel al ampplificador elegido?

* El ruido en los amplificadores no es blanco. En general hay un ruido llamado *Filker noise* o *1/f noise* que aparece en la región de baja frecuencia. A modo informativo, subo una nota de aplicación que explica un poco mejor cómo hacen los fabricantes de amplificadores para generar modelos de ruido que incluyan ruido blanco y ruido 1/f.

Ejercicio 7 Un sistemas de comunicaciones digital muy simple puede modelarse de la siguiente manera:

(i) Se genera una señal de datos aleatorios X(n) que solo contiene los valores {1;-1}. Esto se suele denominar modulación PAM2 (Pulse Amplitude Modulatio, 2 porque solo tiene dos niveles). Es la forma más elemental de una modulación digital.

(ii) En el canal, se agregan muestras de ruido N(n) con media cero, y con cierta varianza σ_N^2. Notar que la varianza es la potencia del ruido (y en este caso coincide con la PSD del ruido porque asumimos que la frecuencia de muestreo del sistema es 1Hz). Se define a la relación señal ruido SNR como $SNR = /fracP_SP_N$, donde P_S es la potencia de la señal de datos, y P_N es la potencia del ruido aditivo **dentro del ancho de banda de la señal**. Notar que en este ejemplo, el ancho de banda de la señal es 1Hz, al igual que el ancho de banda de simulación, al igual que el ancho de banda del ruido. Además, $P_S = 1$, por lo que la SNR resulta $SNR = 1/sigma_N^2$.

(III) La señal recibida resulta Y(n)=X(n)+N(n), es decir, es una versión ruidosa de la señal transmitida. A este modelo se lo conoce como AWGN (Additive White Gaussian Noise).

Tarea:

a) Realizar el histograma de X, N e Y, y explicar los resultados obtenidos. Asumir que la potencia del ruido es $\sigma_N^2 = 0{,}1$

b) Calcular la SNR para este caso (teoría y simulación)

c) El objetivo de un receptor digital es determinar el valor de los datos que se mandaron en el lado transmisor (en comunicaciones digitales, a los datos transmitidos los llamamos símbolos). En este caso, el receptor debe determinar si el símbolo que se transmitió en cada instante es un 1 o un -1. Para ello, se podría seguir la siguiente estrategia: si la muestra recibida es mayor a 0, entonces el receptor asume que se transmitió un 1. Si la muestra recibida es menos a 0, entonces se asume que se transmitió un -1. A esto se lo llama **tomar una decisión**. Dado que las muestras recibidas están contaminadas con ruido, hay una posibilidad de que la decisión se tome de forma incorrecta. Por ejemplo, se transmite un 1, pero la muestra de ruido que se suma en el canal es -1.23. Luego, la muestra recibida es Y=-0.23. El receptor interpretará que se transmitió un -1, ya que $Y < 0$, pero vemos que en realidad se ha transmitido un 1. A esto se lo llama error de bit. Una métrica importante es la BER (Bit Error Rate, o tasa de error de bits). Para medir la BER, se transmiten M símbolos, se cuentan los F errores que hubo luego de tomar la decisión y se computa BER=F/M. Tarea: medir la BER para este ejemplo. Acá te dejo un modelo simple para armar este simulador:

```
X = randi([0 1], Lsim,1)*2-1; % Lsim es el número de símbolos a transmitir
N = sqrt(sigma2).*randn(Lsim,1); % sigma2 es la varianza del ruido
Y=X+N; % Señal recibida, son los símbolos de TX contaminados con ruido

% Toma de decision en el receptor
Y_decidido = Y;
Y_decidido(Y>0)=1;
Y_decidido(Y<0)=-1;
```

Te queda como tarea determinar cuántos de los Y decididos son errores y contar la BER.

d) Qué pasa si la SNR es más grande? Y si es más chica? Justificá tu respuesta usando simulaciones, o una explicación teórica

Referencias

[1] Roy Yates, "Probability and Stochastic Processes, A Friendly Introduction for Electrical and Computer Engineers", Second Edition, JOHN WILEY & SONS, 2005