

ELO313 - Certamen #3

Procesamiento de Señales con Aplicaciones

Primer Semestre 2021

Profesor: Matías Zañartu, Ph.D.

Fecha: 13 de Agosto de 2021, 9:35 am, por Zoom

NOTA IMPORTANTE:

Este certamen es individual, donde sus respuestas deben reflejar su exclusivo conocimiento y trabajo, sin el apoyo de ningún método o material que le entregue ventaja por sobre el resto de sus compañeros y compañeras. El no cumplimiento de esta regla será investigado y fuertemente sancionado, para todas las partes involucradas, en base a las normas éticas establecidas en el “Reglamento de Derechos y Deberes de los Alumnos en Casa Central y Campus Santiago”, DR 182/2016. El certamen deberá ser entregado en formato PDF junto con sus archivos de MATLAB y un link a sus archivos de audio mediante la plataforma AULA en la fecha y hora indicada cuidando de no exceder los 400 MB en su archivo. No se aceptan entregas atrasadas ni por email.

1 Filtrado de señales

- a. Un filtro digital está caracterizado por las siguientes propiedades: (1) Es un pasa-alto y tiene un polo y un cero. (2) El polo está ubicado a una distancia $r = 0.9$ del origen en el plano- z . (3) Las señales constantes no pasan por el sistema.
 - i. Determine la función de transferencia y grafique el diagrama de polos y ceros.
 - ii. Calcule la magnitud y fase de la respuesta de frecuencia (DTFT) del sistema
 - iii. Normalice la respuesta de frecuencia de modo que tenga ganancia unitaria en $\omega = \pi$, y luego determine la ecuación de diferencia del filtro.
 - iv. Calcule la salida del sistema si la entrada es $x[n] = 2 \cos(\frac{\pi}{6}n + \frac{\pi}{4})$
- b. El archivo `ecg2x60.dat` contiene una señal de un electrocardiograma (ECG) muestreada a 200 Hz con un ruido no menor de alimentación a 60 Hz. Vea también `ecg2x60.m` y lea sobre este tipo de señales aquí: <http://goo.gl/2MdpSj>.
 - i. Diseñe un filtro notch con dos ceros que elimine este ruido e impleméntelo en MATLAB. Estudie los efectos en magnitud y fase que produce su filtro.
 - ii. Agregue dos polos a la misma frecuencia que los ceros, pero con un radio menor a una unidad. Estudie los efectos de los polos tanto en magnitud como fase a medida que el radio de ellos varía entre 0.8 y 0.99.
- c. La señal de ECG del archivo `ecg_1fn.dat` contiene fluctuaciones en su nivel DC a lo largo del tiempo (vea también `ecg_1fn.m`). Diseñe filtros pasa-alto Butterworth de ordenes 2-8 con frecuencias 0.5-5 Hz. Estudie la eficacia de estos filtros para eliminar la fluctuación DC y los efectos que produce en la señal. ¿Cómo se asocian estos resultados con la respuesta de magnitud y fase de los filtros? ¿Hay algún filtro optimo” en algún sentido entre ellos?

2 Identificación de Vocales

Cargue el archivo `vowels.mat` en MATLAB. Obtenga la DFT de cada vocal con una resolución espectral adecuada.

- a. Compare la DFT de las vocales considerando el largo completo y aquellas porciones en que solo existe señal de voz (entre las muestras 5000 y 10000 en cada caso). Grafique la magnitud del espectro de cada vocal entre $[0, f_s/2]$ y en su amplitud en dB. ¿A qué se deben dichas diferencias? ¿Qué caso es más adecuado para analizar la voz?.
- b. Considere solo las porciones en que existe señal de voz para cada vocal. ¿Cuál es la frecuencia fundamental y las frecuencias de resonancia (formantes) del tracto vocal en cada caso? Escriba los valores de los dos primeros formantes en una tabla y grafique el triángulo vocálico (frecuencia del primer formante v/s frecuencia del segundo formante) para las vocales consideradas.
- c. Utilizando el comando $y = lpc(x, p)$ de MATLAB (para $y = a, e, i, o, u$), obtenga el filtro IIR que simula el efecto modulador del tracto vocal en cada caso. Utilice $p = 15$ y solo las porciones con voz en este caso. Utilizando `freqz` grafique la magnitud de la respuesta de cada filtro, la cual deberá ser superpuesta con la DFT con la frecuencia entre $[0, f_s/2]$ y en su amplitud en dB. Asegúrese que las amplitudes de las magnitudes estén superpuestas en el mismo rango de amplitud. Obtenga nuevamente los valores de los formantes e inclúyalos en una tabla y dibuje el triángulo vocálico en función a los datos obtenidos vía LPC. ¿Varía su estimación de los formantes con esta nueva representación espectral?

3 Espectrograma

3.1 Construcción de espectrogramas:

Cree una función de MATLAB $X = DFTwin(x, L, m, N)$ que permita calcular una DFT sobre un segmento de largo L sobre la señal x , que comience en la muestra m de la señal x , y que tenga N puntos de resolución espectral en su DFT. Utilice la función *fft* de MATLAB para calcular la DFT. Utilice una ventana Hamming de largo L para segmentar la señal.

- a. Utilizando su función *DFTwin*, obtenga la DFT de la vocal *a* del archivo *vowels.mat* considerando seis periodos de oscilación. Grafique la magnitud del espectro entre [0,4000] Hz y su amplitud en dB.
- b. Cree una función MATLAB $X = specgm(x, L, overlap, N)$ la cual permita generar un espectrograma de la señal x , con ventanas de largo L que se superpongan cada *overlap* puntos, y que tenga N puntos de resolución espectral en su DFT. Utilice su función *DFTwin* a lo largo del tiempo y almacene los espectros en columnas de igual largo. Utilice *imagesc* sobre la magnitud de los espectros en escala logarítmica para la generación del espectrograma. Presente el espectro etiquetado en sus ejes en segundos y de [0, 4000] Hz y su amplitud en dB. Grafique, interprete y comente las diferencias entre los espectrogramas de banda ancha y angosta de las siguientes señales:
 - a. La señal *chirp.mat* con una ventana de 5 ms y una de 100 ms, ambas sin overlap y una resolución espectral de 1024 puntos.
 - b. La señal *test_signal* del archivo *test_training_signals.mat* con una ventana de 5 ms y una de 50 ms, con un 50% de overlap y una resolución espectral de 1024 puntos.
 - c. La señal *glide* del archivo *vocal_gestures.mat* con una ventana de 5 ms y una de 50 ms, con un 50% de overlap y una resolución espectral de 1024 puntos.
 - d. La señal *vowels* del archivo *vocal_gestures.mat* con una ventana de 5 ms y una de 50 ms, con un 50% de overlap y una resolución espectral de 1024 puntos.
- c. Grafique manualmente sobre los espectrogramas de los puntos c y d anteriores (señales *glide* y *vowels*), los tres primeros formantes y sus transiciones temporales. ¿Cómo podría hacerse esta estimación de forma automática? ¿Qué largo de ventana es más ventajoso para identificar los formantes?. ¿Qué largo es mejor para identificar los armónicos?

Observaciones para la construcción de espectrogramas:

- Ubique el tiempo en la dirección horizontal con el origen ($t=0$ s) en la esquina inferior izquierda y la frecuencia en la dirección vertical con el origen ($f=0$ Hz) en la esquina inferior izquierda.
- La función *DFTwin* entrega la DFT entre $[0, 2\pi]$, de modo que se debe descartar la mitad de esta componente al construir el espectrograma.
- Utilice el comando *colormap(1 - gray)* para graficar las imágenes resultantes en escala de grises.

3.2 Síntesis de audio a partir de un espectrograma

Cree una función de MATLAB $x = spectrasound(Im, fs)$ que permita leer una imagen Im y que genere un archivo x de audio a partir de la imagen como si esta fuese un espectrograma sin overlap. Considere que lo que se visualiza es la magnitud del espectrograma y evalúe posibles formas de manejar la fase del espectro. El largo de la señal resultante será función de la frecuencia de muestreo. Utilice la función *imread* de MATLAB o similar para leer la imagen.

- Utilice su función para evaluar si logra recuperar la señal original a partir de los espectrograma de la Pregunta 3.1.a
- Utilice su función para evaluar si logra recuperar la señal original a partir de los espectrograma de la Pregunta 3.1.b, con el overlap y ventanas utilizadas.
- Utilice su función para crear un sonido a partir una imagen de su preferencia. Utilice este sonido para crear un espectrograma de banda ancha y angosta y compare las diferencias entre las imágenes.