Guía Laboratorio 11
Procesamiento Digital de Señales

Juan Camilo Vásquez, Tomás Arias

Mayo 2017 - 2017-1

1. Introducción

- La práctica se debe desarrollar de manera individual.
- Por cada punto genere una celda en el *jupyter-notebook*.
- Concluya en cada punto.
- El informe deben seguir tener el siguiente formato para el nombre de archivo: *Nombre_Apelido.ipynb*.

2. Filtros con respuesta infinita al impulso (IIR)

Este tipo de filtros digitales se caracterizan porque el tamaño de su respuesta a la señal impulso tiene un número infinito de componentes.

Los filtros IIR se pueden expresar mediante la siguiente ecuación en diferencias:

$$y[n] = \frac{1}{a_0}\left(\sum_{i=0}^{P} b_i x[n-i] - \sum_{j=1}^{Q} a_j y[n-j]\right) \tag{1}$$

Como se puede observar, estos filtros tienen realimentación de valores pasados de la salida ponderados por los coeficientes a_j. Esta realimentación indica que estos filtros poseen polos diferentes a $z = 0$. Esto trae como ventaja que se puede ajustar mucho mejor la respuesta del filtro, por lo que pueden ser utilizados en un mayor número de aplicaciones y lograr buenos resultados con un orden menor al filtro FIR, es decir, pueden ser más eficientes computacionalmente. La desventaja es que estos filtros pueden llegar a ser inestables y que su fase no es lineal. Una de las formas para el diseño de filtros IIR es mediante la transformación de filtro análogo, por ejemplo Butterworth o Chebyshev, a su equivalente digital.

2.1. Comparación filtros FIR e IIR

1. Diseñe un filtro FIR con las siguientes características. Puede usar la función *firwin* de la libreria scipy.signal. No olvide normalizar la frecuencia de corte del filtro respecto a la mitad de la frecuencia de muestreo. Recuerde además que el filtro FIR tiene N polos en el origen.
   - Tipo: Pasabajas
   - Orden: 12
   - Ventana: Hamming
   - Frecuencia de muestreo: 1000 Hz
   - Frecuencia de corte: 100 Hz
2. Diseñe un filtro IIR tipo Chebyshev con las siguientes características. Puede usar la función *cheby2* de la libreria scipy.signal. No olvide normalizar la frecuencia de corte del filtro respecto a la mitad de la frecuencia de muestreo.

- Tipo: Pasabajas
- Método de diseño: Chebyshev Tipo II
- Orden: 12
- Frecuencia de muestreo: 1000 Hz
- Frecuencia de corte: 100 Hz
- Atenuación en banda rechazada(rs): 30 dB

3. Compare las respuestas al impulso de ambos filtros. ¿Qué puede concluir?

   Para la respuesta al impulso utilice la función *dimpulse* de la librería scipy.signal

4. Compare las respuestas al escalon de ambos filtros. ¿Qué puede concluir?

   Para la respuesta al impulso utilice la función *dstep* de la librería scipy.signal

5. Compare las respuestas de magnitud frecuencia de ambos filtros. ¿Qué puede concluir? ¿Cuál considera mejor filtro?

6. Utilice la siguiente función para comparar la respuesta de los filtros FIR e IIR ante tres señales sinusoidales con frecuencias de 10 Hz, 80 Hz, y 101 Hz. *b_fir* y *b_iir* son los numeradores de la función de transferencia de los filtros FIR e IIR. *a_fir* y *a_iir* corresponde a los denominadores.

```
def compare_filters(b_fir, a_fir, b_iir, a_iir, fs):
    t=np.arange(0,0.16, 1./fs)
    f1=20
    f2=80
    f3=101
    y1=np.sin(2*np.pi*f1*t)
    y2=np.sin(2*np.pi*f2*t)
    y3=np.sin(2*np.pi*f3*t)

    y1iir=sp.lfilter(b_iir, a_iir, y1)
    y2iir=sp.lfilter(b_iir, a_iir, y2)
    y3iir=sp.lfilter(b_iir, a_iir, y3)

    y1fir=sp.lfilter(b_fir, a_fir, y1)
    y2fir=sp.lfilter(b_fir, a_fir, y2)
    y3fir=sp.lfilter(b_fir, a_fir, y3)

    plt.figure(figsize=(20,8))
    plt.subplot(311)
    plt.plot(t,y1, label='Senal de entrada')
    plt.plot(t,y1fir, label='Senal filtrada FIR')
    plt.plot(t,y1iir, label='Senal filtrada IIR')
    plt.subplot(312)
    plt.plot(t,y2, label='Senal de entrada')
    plt.plot(t,y2fir, label='Senal filtrada FIR')
    plt.plot(t,y2iir, label='Senal filtrada IIR')
    plt.subplot(313)
    plt.plot(t,y3, label='Senal de entrada')
    plt.plot(t,y3fir, label='Senal filtrada FIR')
    plt.plot(t,y3iir, label='Senal filtrada IIR')
    plt.legend(loc=4, fontsize=16, ncol=3)
    plt.show()
```

7. Compare la atenuación producida por los filtros en cada frecuencia. ¿Qué puede concluir?

8. Compare el retraso producido por los filtros en cada frecuencia. ¿Qué puede concluir?

2.2. Filtrado de música con filtros FIR e IIR

1. Cargue el audio que acompaña esta guia

```
fs, s = read('gtr-jazz.wav')
s=s[:,0]
IPython.display.Audio(s, rate=fs)
```

Suponga que estamos interesados en extraer solamente la parte del bajo; Sabemos que el bajo tiene la mayor parte de su energía en la zona de baja frecuencia. Se puede tratar de filtrar el audio con un filtro pasa-bajas con ua frecuencia de corte en 200 Hz.

2. Diseñe un filtro IIR tipo butterworth de orden 6 con una frecuencia de corte de 200 Hz, y grafique su respuesta en frecuencia. Recuerde normalizar la frecuencia de corte respecto a la mitad de la frecuencia de muestreo

```
b, a = sp.butter(6, wc)
wb, Hb = sp.freqz(b, a, 1024);
plt.plot(wb[0:100]/np.pi * (fs/2), np.abs(Hb[0:100]))
```

3. Pase la señal de audio por el filtro diseñado, y escúchela

```
y = sp.lfilter(b, a, s)
IPython.display.Audio(data = y, rate = fs)
```

Se puede tratar de alcanzar el mismo efecto usando un filtro FIR. Es posible, pero se necesita un orden del filtro muy grande

4. Diseñe un filtro FIR de orden 500, con una frecuencia de corte en 200 Hz, con una banda de transición de 100 Hz

```
tb = 100
M = 500
h = sp.remez(M, [0, fc, fc+tb, fs/2], [1, 0], [1, 1], Hz=fs, maxiter=50)

w, H = sp.freqz(h, 1, 1024)
plt.plot(w[0:100]/np.pi * (SF/2), np.abs(H[0:100]), label='FIR')
plt.plot(wb[0:100]/np.pi * (SF/2), np.abs(Hb[0:100]), 'green', label='IIR');
plt.legend()
```

5. Filtre y escuche la señal filtrada por el filtro FIR, y concluya.

```
y = sp.lfilter(h, 1, s)
IPython.display.Audio(y, rate=SF)
```

6. Cambie el orden del filtro FIR por 1600, repita los dos puntos anteriores, y concluya.

7. Ahora si se quiere obtener solamente la parte de la percusión de platillos, deben pasar solamente las altas altas frecuencias, por encima de 7 kHz. Diseñe un filtro IIR que cumpla con este objetivo