

# ELO 314 - Procesamiento Digital de Señales

## Lab. 3 - Parte I: Filtros Digitales en MatLab

### Preparado por

Dr. Matías Zañartu, e-mail: Matias.Zanartu@usm.cl

Dr. Christian Rojas, e-mail: Christian.Rojas@usm.cl

### I. INTRODUCCIÓN

En este laboratorio se trabajará con conceptos básicos de filtros FIR (Finite Impulse Response) e IIR (Infinite Impulse Response), y herramientas de diseño y análisis de MATLAB. Los filtros son sistemas LTI (lineales e invariantes en el tiempo) que pueden tomar diferentes respuestas en frecuencia, y son descritos como:

$$y[n] = \sum_{i=0}^{N-1} b_i x[n-i] - \sum_{k=1}^M a_k y[n-k]. \quad (1)$$

Un filtro FIR es aquel que puede ser expresado sin términos recurrentes, i.e., pueden ser descritos considerando los coeficientes  $a_k = 0$  para  $k = 1, 2, \dots, N$  (el coeficiente  $a_0$  es siempre igual a 1). Los filtros IIR son recursivos y cuentan con ambos tipos de coeficientes.

La transformada  $\mathcal{Z}$  permite analizar la respuesta de frecuencia  $H(z = e^{jw})$  de los filtros. La magnitud de esta respuesta  $|H(w)|$  determina cómo se altera la amplitud en función de la frecuencia, y su respuesta de fase  $\angle H(w)$  define los retardos en el tiempo producto del filtro en función de la frecuencia.

Los filtros FIR son siempre estables (tienen solo ceros<sup>1</sup>), pueden ser diseñados con fase lineal, y no sufren de mayores problemas por errores de redondeo de sus coeficientes. Los filtros IIR suelen ser más eficientes que los FIR (requieren menos cálculos para obtener una misma respuesta), pero son muy sensibles a errores de redondeo en sus coeficientes. Estos filtros pueden ser inestables y tienen una respuesta de fase no lineal. Todo filtro compuesto por polos/ceros conjugados tiene una respuesta a impulso con valores reales.

MATLAB cuenta con una serie de herramientas para el diseño y análisis de filtros que serán estudiados en este laboratorio, entre ellos, *filter*, *fdatool*, *freqz*, *fir1*, *fir2*, *firpm*, *ellip*, *butter*, *cheby1*, *cheby2*, etc. Otras funciones diseñadas para este laboratorio serán utilizadas.

### II. CONCEPTOS BÁSICOS DE FILTROS FIR E IIR

- 1) Basado en la Fig. 4, diseñe un filtro que tenga dos ceros conjugados sobre el círculo unitario, ubicados en un ángulo  $\theta$  en el plano  $\mathcal{Z}$ . Escriba la expresión analítica en función de  $z$  y luego use  $z = e^{jwT_s}$  para obtener  $H(w)$ . Escriba también la ecuación de diferencia del filtro y grafique la respuesta a impulso del filtro. Grafique la magnitud de la respuesta de frecuencia  $|H(w)|$  para tres distintos ángulos  $\theta = \frac{\pi}{6}, \frac{\pi}{3}, \frac{\pi}{2}$  en el intervalo  $[-\pi, \pi]$  y amplitud en dB ( $20\log_{10}(|H(w)|)$ ). Explique cómo el valor de  $\theta$  se relaciona con la respuesta de frecuencia del filtro dada una frecuencia de muestreo  $f_s$ . (5 Ptos.)
- 2) Basado en la Fig. 5, diseñe un filtro que tenga dos polos conjugados dentro del círculo unitario, ubicados en un ángulo  $\theta$  en el plano  $\mathcal{Z}$ . Considere que el filtro tiene además una ganancia sobre la entrada de  $b_0 = 1 - r$ . Escriba la expresión analítica en función de  $z$  y luego use  $z = e^{jw}$  para obtener  $H(w)$ . Escriba también la ecuación de diferencia del filtro y grafique la respuesta a impulso del filtro. Grafique la magnitud de la respuesta de frecuencia  $|H(w)|$  para  $\theta = \frac{\pi}{3}$  y tres distintas amplitudes  $r = 0.99, 0.9, 0.7$ . Explique cómo el valor de  $r$  afecta la respuesta de frecuencia y el ancho de banda del filtro. (5 Ptos.)

<sup>1</sup>Pueden tener polos que se cancelan con ceros y/o polos ubicados en el origen.

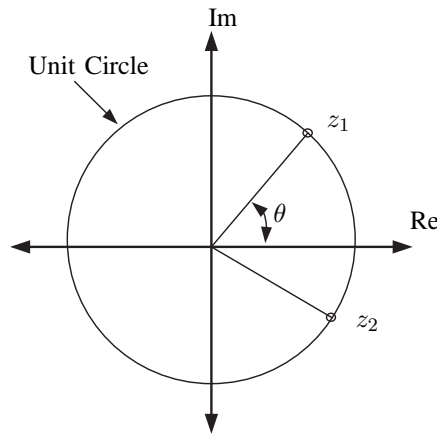


Figure 1. Ubicación de dos ceros conjugados para un filtro FIR simple.

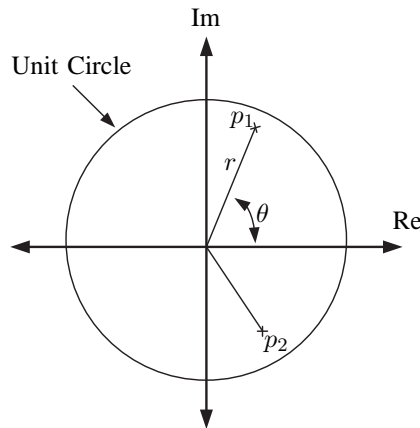


Figure 2. Ubicación de dos polos conjugados para un filtro IIR simple.

### III. ANÁLISIS Y FILTRADO SIMPLE DE SEÑALES

- 1) Aplique el filtro FIR diseñado en el punto II.1 a una señal de audio. Utilice el archivo *nspeech.mat* el cual contiene una señal de voz junto con la presencia no deseada de un tono puro. Utilizando la función *DTFT.m* obtenga  $X(w)$ , identifique la frecuencia del tono, y ajuste los parámetros de su filtro para eliminar dicha componente (la salida de la función DTFT, divídala por el número de muestras). Escriba una función llamada *FIR\_filter.m* donde implemente el filtrado mediante convolución. Grafique la señal resultante en el dominio del tiempo (20 primeros ms) y la frecuencia (magnitud de  $X(w)$  en el rango  $[0, \frac{f_s}{2}]$  y en dB, con  $\text{dB} = 20\log_{10}(|X(w)|)$ . Escuche la salida del filtro. Compare la señal filtrada con los obtenidos utilizando el comando *filter* de MATLAB. ¿Tienen el mismo largo?, ¿Por qué?. (5 Ptos.)
- 2) Aplique el filtro IIR diseñado en el punto II.2 a una señal de audio. Utilice el archivo *pcm.mat* el cual contiene una señal modulada en presencia de ruido. Utilizando la función *DTFT.m* obtenga  $X(w)$ , identifique la frecuencia de la señal modulada, y ajuste los parámetros de su filtro para eliminar el ruido. Escriba una función llamada *IIR\_filter.m* donde implemente el filtrado mediante la ecuación de diferencia. Grafique la señal resultante en el dominio del tiempo (500 primeros ms) y la frecuencia (magnitud de  $X(w)$  en el rango  $[0, \frac{f_s}{2}]$  y en dB, con  $\text{dB} = 20\log_{10}(|X(w)|)$ . Escuche la salida del filtro. Utilice un valor de  $r = 0.99$ . Compare sus resultados con los obtenidos utilizando el comando *filter* de MATLAB. (5 Ptos.)

### IV. DISEÑO DE FILTROS FIR USANDO VENTANAS

Considere el filtro pasa-bajo ideal con una frecuencia de corte  $w_c$ , el cual tiene una respuesta de frecuencia dada por:

$$H_{ideal}(w) = \begin{cases} 1, & |w| \leq w_c \\ 0, & w_c \leq |w| \leq \pi \end{cases}, \quad (2)$$

y una respuesta a impulso dada por:

$$h_{ideal}[n] = \frac{w_c}{\pi} \text{sinc}\left(\frac{nw_c}{\pi}\right) \quad (3)$$

para  $-\infty \leq n \leq \infty$ . Una aproximación razonable a este filtro es truncar su respuesta a impulso y asumir que es cero fuera de un rango  $n \in [-M, M]$ . Junto a esto se suele considerar un desplazamiento en el tiempo de la respuesta de impulso para lograr que el filtro resultante sea causal. La respuesta de frecuencia de dicho filtro truncado está dada por la convolución de  $H_{ideal}(w)$  con la DTFT de la ventana usada para truncar la respuesta a impulso y un retardo lineal en la fase dado por el desplazamiento.

- 1) Considere un truncamiento con una ventana rectangular, de modo que el filtro resultante está dado por:

$$h[n] = \begin{cases} \frac{w_c}{\pi} \text{sinc}\left(\frac{w_c}{\pi}\left(n - \frac{N-1}{2}\right)\right), & n = 0, 1, \dots, N-1 \\ 0, & \text{otro caso} \end{cases}, \quad (4)$$

Obtenga la magnitud  $H(w)$  y la fase  $\angle H(w)$  para un filtro de frecuencia de corte  $w_c = \frac{2\pi}{3}$ , con largos  $N = 21, 101, 1001$ , utilizando la función *DTFT.m* con una resolución no menor a 512 para cada caso. ¿En qué afecta el largo de la ventana en la respuesta de frecuencia del filtro y la fase? (3 Ptos.)

- 2) Compare las propiedades temporales y espectrales de filtros FIR construidos con los siguientes tipos de ventanas: rectangular, hamming, hanning, blackman, y bartlett. Utilice una frecuencia de muestreo de 1 kHz y una ventana con una duración de 100 ms. Grafique en un figura usando subplots las cinco ventanas. Grafique la magnitud y fase con respecto a la frecuencia de estas ventanas. Cree una tabla con el ancho del mainlobe y su amplitud relativa al primer sidelobe. Comente las ventajas y desventajas de cada ventana y discuta posibles ejemplos donde cada filtro podría ser mejor que los demás. Utilice los comandos *freqz*, *rectwin*, *hann*, *hamming*, *blackman*, y *bartlett*. (5 Ptos.)

## V. DISEÑO DE FILTROS FIR USANDO HERRAMIENTAS DE MATLAB

Estudiar los comandos *fir1*, *fir2* y *firpm* de MATLAB. Comente las principales características de cada uno de ellos.

- 1) Diseñe un filtro pasa-bajos de orden 70 con frecuencia de corte en 1.5 kHz y frecuencia de muestreo 8 kHz, empleando el método por ventanas usando el comando *fir1*, (use ventanas blackman, y rectangular). Analice similitudes y diferencias empleando las herramientas de *fdatool* que estime más pertinentes. Grafique usando la función de Matlab *freqz*. (3 Ptos.)
- 2) Diseñe el mismo filtro anterior pero utilizando *firpm*. ¿Qué diferencias nota con respecto al diseño anterior? Analice similitudes y diferencias empleando las herramientas de *fdatool* que estime más pertinentes. Asegúrese de considerar los parámetros adecuados para el comando *firpm* (se le recomienda revisar primero el comando *firpmor*). (2 Ptos.)
- 3) Obtenga los coeficientes para un filtro de orden 70 y de orden 150 para el filtro de la Fig. 6. utilizando el comando *fir2*. Compare la respuesta de magnitud obtenida con las especificaciones establecidas para cada caso. (2 Ptos.)

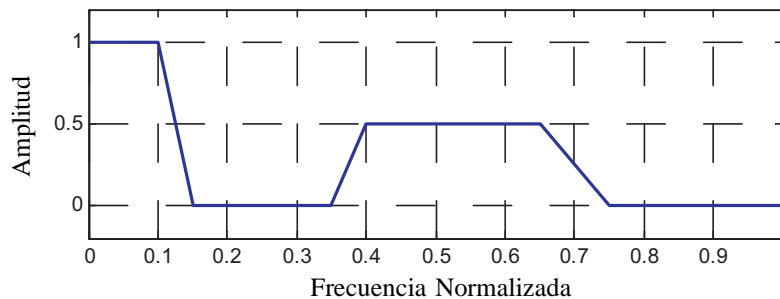


Figure 3. Respuesta del filtro a diseñar con *fir2*.

## VI. IMPLEMENTACIÓN DE FILTROS IIR: FORMAS DIRECTAS I Y II

Existen varios tipos de estructuras para la implementación de filtros IIR. Considere el siguiente filtro IIR:

$$y[n] = \sum_{i=0}^{N-1} b_i x[n-i] - \sum_{k=1}^M a_k y[n-k]. \quad (5)$$

Una forma alternativa de escribir una ecuación de diferencia de un filtro IIR es por medio de 2 funciones de transferencia individuales en cascada, esto quiere decir:

$$H(z) = H_1(z) \cdot H_2(z), \quad (6)$$

donde  $H_1(z)$  posee solo ceros de  $H(z)$  y  $H_2(z)$  solo polos <sup>2</sup>.

$$H_1(z) = \sum_{k=0}^M b_k z^{-k}, \quad (7)$$

$$H_2(z) = \frac{1}{1 + \sum_{k=1}^N a_k z^{-k}}. \quad (8)$$

Según sea el orden en que se ubican las funciones de transferencia de las ecuaciones (7) y (8), se llegará a uno de 2 tipos de realizaciones en formas directas. La ubicación de  $H_1(z)$  antes que  $H_2(z)$  llevará a la llamada *forma directa I*, la cual requiere de  $M + N + 1$  multiplicaciones y posiciones de memoria, y  $M + N$  sumas. Esta implementación es la más intuitiva, ya que el filtro se implementa como su ecuación de diferencia original, sin ningún tipo de reordenamiento. Por otro lado, si se ubica  $H_2(z)$  antes que  $H_1(z)$ , la realización obtenida es conocida como *forma directa II*. En este caso, se requiere de una variable intermedia  $w(n)$  para implementación del filtro, lo que agrega una nueva ecuación de diferencia. Como es de esperar, esta variable adicional solo es una herramienta de implementación y no genera cambios en el filtro. La principal ventaja de esta implementación es que necesita menos posiciones de memoria (usa  $\max\{M, N\}$  en vez de  $M + N + 1$ ), aunque el número de multiplicaciones y sumas sigue siendo el mismo. A partir de las ecuaciones 8 y 8 se obtienen las ecuaciones de diferencias correspondientes:

$$w(n) = x(n) - \sum_{k=1}^N a_k w(n-k), \quad (9)$$

$$y(n) = \sum_{k=0}^M b_k w(n-k), \quad (10)$$

$$H(z) = H_2(z) \cdot H_1(z), \quad (11)$$

y

$$H(z) = \frac{W(z)}{X(z)} \frac{Y(z)}{W(z)}. \quad (12)$$

Es posible visualizar claramente estas diferencias mediante un filtro IIR de segundo orden. Las Fig. 4 y Fig. 5 lo muestran uno en su estructura de la *forma directa I* y *II*, respectivamente.

<sup>2</sup>Sin contar con los polos/ceros en el origen en cada caso

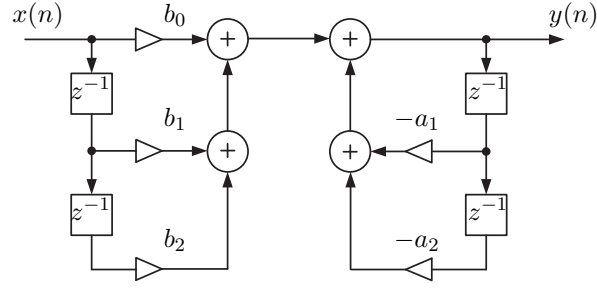


Figure 4. Forma directa I de un filtro de segundo orden IIR.

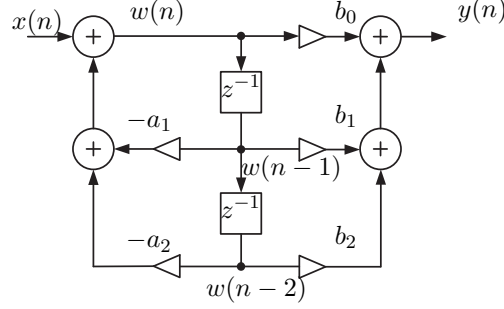


Figure 5. Forma directa II de un filtro de segundo orden IIR.

## VII. IMPLEMENTACIÓN DE FILTROS IIR: CASCADAS DE SEGUNDO ORDEN

Desafortunadamente, las implementaciones de filtros IIR usando *forma directas* tienen la desventaja de ser altamente sensibles a la cuantificación de sus coeficientes cuando  $N$  es muy grande. Una solución más robusta es implementar los filtros mediante una cascada de filtros de segundo orden. Considere un filtro IIR de orden superior a 2 con la función de transferencia indicada en la ecuación 5. El filtro puede ser factorizado en la cascada de subsistemas de segundo orden tal que  $H(z)$  puede ser expresado como:

$$H(z) = \prod_{k=1}^K H_k(z), \quad (13)$$

donde  $K$  es la parte entera de  $(N + 1)/2$ .  $H_k$  tiene la forma general:

$$H_k(z) = \frac{b_{k0} + b_{k1}z^{-1} + b_{k2}z^{-2}}{1 + a_{k1}z^{-1} + a_{k2}z^{-2}}. \quad (14)$$

Como existen muchas formas de manejar los polos y ceros de la ecuación 13 en secciones de segundo orden, habrá muchas realizaciones en cascada y varias maneras de ordenar los subsistemas resultantes. Aunque todas las realizaciones en cascada son equivalentes en aritmética infinita, las diferentes realizaciones difieren significativamente cuando se implementan con aritmética de precisión finita.

La forma general de la estructura en cascada se muestra en la Fig. 6.

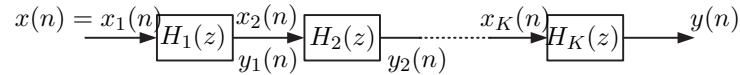


Figure 6. Estructura en cascada de sistemas de segundo orden.

## VIII. DISEÑO DE FILTROS IIR MEDIANTE HERRAMIENTAS DE MATLAB

Para todos los casos en este punto asuma una frecuencia de muestreo de 8 kHz. Utilice el comando *freqz* para obtener la respuesta en frecuencia de sus diseños de filtros.

1. Utilizando la función *ellip* de MATLAB diseñe un filtro de segundo orden para cada uno de los siguientes

casos (4 Ptos.):

- 1) Pasa bajos ( $f_c = 1$  kHz).
- 2) Pasa altos ( $f_c = 2$  kHz).
- 3) Pasa banda ( $f_1 = 1$  kHz,  $f_2 = 2$  kHz).
- 4) Elimina banda ( $f_1 = 1$  kHz,  $f_2 = 2$  kHz).

2. Utilizando la función *ChebyX* de MATLAB diseñe un filtro de segundo orden para cada uno de los siguientes casos (pasa banda  $f_1 = 1$  kHz,  $f_2 = 2$  kHz) (2 Ptos.):

- 1) 2 dB para la banda de paso (Cheby1).
- 2) 20 dB de atenuación (Cheby2).

3. Diseñe un filtro Butterworth pasabanda con frecuencias entre 400[Hz] y 800[Hz], y de orden 4, 10 y 20. ¿Qué sucede con los polos y ceros del sistema a medida que aumenta el orden del filtro? ¿Qué implicancias tiene esto en relación con la implementación en la tarjeta DSK?. (3 Ptos.)

## IX. COMPARACIÓN DE DISEÑOS ÓPTIMOS DE FILTROS FIR E IIR

Diseñe filtros IIR del tipo Butterworth, Chebyshev, y Elliptic, junto con un filtro FIR del tipo equiripple en MATLAB para las siguientes especificaciones de un filtro pasa-banda:

- Passband frequency,  $f_p = 1.2$  kHz,
- Stopband frequency,  $f_{sp} = 2$  kHz,
- Sampling frequency,  $f_s = 8$  kHz,
- Pass band ripple,  $R_p = 0.5$  dB,
- Stop band,  $R_s = 40$  dB o mayor.

- 1) Use los comandos de MATLAB buttord, cheb1ord, ellipord, y firpmord para determinar el orden óptimo más bajo para las especificaciones dadas. ¿Qué filtro IIR tiene el orden más bajo? ¿Cómo se compara el orden de los filtros FIR con los IIR? (6 Ptos.)
- 2) Comente las ventajas de cada filtro y presente ejemplos en los cuales sería conveniente usar los distintos diseños (2 Ptos.)

### **Informe de Laboratorio:**

Presente sus cálculos, ecuaciones y los gráficos requeridos. Comente sus observaciones y etiquete sus gráficos adecuadamente. Los códigos archivos generados para el informe deberán ser enviados vía email en un archivo comprimido.