

4.- DISEÑO DE FILTROS IIR

4.1.- Introducción.

Son sistemas cuya salida depende además de salidas anteriores y que, estando en reposo, al ser estimulados con una entrada impulsional su salida no vuelve al reposo, de ahí el calificativo de filtros de respuesta impulsional infinita (IIR). La ecuación en diferencias general es de la forma:

$$y(n) = b_0x(n) + b_1x(n-1) + \dots + b_Mx(n-M) - a_1y(n-1) - a_2y(n-2) - \dots - a_Ny(n-N) =$$

$$= \sum_{k=0}^M b_k \cdot x(n-k) - \sum_{k=1}^N a_k \cdot y(n-k)$$

donde el orden es igual al máximo de M y N.

La función de transferencia en Z del filtro es:

$$H(z) = \frac{\sum_{k=0}^M b_k \cdot z^{-k}}{\sum_{k=0}^N a_k \cdot z^{-k}}$$

No todo sistema que tenga esta forma es IIR. Por ejemplo:

$$y(n) = y(n-1) + 0.2 \cdot x(n) - 0.2 \cdot x(n-5) \quad H(z) = 0.2 \cdot \frac{1 - z^{-5}}{1 - z^{-1}}$$

es aparentemente un filtro IIR pues presenta términos recursivos, sin embargo vemos que ésta no es sino una forma distinta de representar el sistema:

$$y(n) = 0.2 \cdot x(n) + 0.2 \cdot x(n-1) + 0.2 \cdot x(n-2) + 0.2 \cdot x(n-3) + 0.2 \cdot x(n-4)$$

que es claramente de fase lineal.

Comparado con un FIR, un filtro IIR requiere un orden mucho menor para cumplir las especificaciones de diseño, sin embargo estos últimos no pueden diseñarse para tener fase lineal. Existen técnicas de compensación de fase mediante la utilización de filtros pasa todo, sin embargo esto aumenta la longitud total del filtro. Si no es necesario que el sistema sea causal (no funcionará en tiempo real) se puede conseguir fase lineal mediante filtros IIR realizando un filtrado BIDIRECCIONAL este consiste en filtrar la señal, invertir el orden de las muestras obtenidas y volver a filtrar de nuevo. La señal obtenida no tendrá distorsión de fase. (Matlab instrucción *filtfilt()*)

Existen dos filosofías de diseño de filtros IIR.

INDIRECTA Se basa en aplicar a filtros analógicos diseñados previamente, transformaciones que los conviertan en digitales con las mismas características. Hay tres métodos fundamentales:

- Diseño por impulso invariante
- Diseño por analogía o aproximación de derivadas
- Diseño por transformación bilineal

DIRECTA Se propone el diseño de filtros digitales imponiendo una serie de condiciones a la respuesta para determinar los coeficientes. Nos centraremos en dos métodos simples como son:

- Diseño por la aproximación de Padé
- Diseño por aproximación de mínimos cuadrados.

También podemos considerar como método directo aunque de uso limitado el diseño por ubicación de ceros y polos.

4.2.- Localización de ceros y polos en filtros IIR.

Los filtros IIR más generales (ARMA) contienen ceros y polos. Si los coeficientes del filtro son reales, si los ceros o polos son complejos siempre aparecen como pares complejos conjugados.

La condición de estabilidad, para sistemas causales implica que los POLOS se encuentran en el interior de la circunferencia unidad. Los ceros no tienen efecto sobre la estabilidad del sistema y pueden encontrarse en el interior o en el exterior de dicha circunferencia.

Cuando los ceros y polos de un sistema se encuentran en el interior de la circunferencia unidad se dice que el sistema es de FASE MÍNIMA.

Cuando todos los ceros y polos están en el exterior de la circunferencia unidad se dice que el sistema es de FASE MÁXIMA.

En general, cuando tenemos ceros y polos en el exterior y en el interior se dice que el sistema es de FASE MIXTA.

Es sencillo verificar que si un sistema tiene un cero en el exterior de la circunferencia unidad ($z_k = a \quad |a| > 1$) y éste se sustituye por su recíproco conjugado ($z'_k = \left(\frac{1}{a}\right)^*$ $|a| > 1$), el sistema tiene la misma respuesta en frecuencia en módulo multiplicado por un factor constante igual al modulo de cero, (probar como Ejercicio), si bien la respuesta en fase sí experimenta cambios.

Un sistema de FASE MÍNIMA también se define como aquel que experimenta un cambio de fase neto nulo; es decir, $\Phi(\pi) - \Phi(0) = 0$

4.3.- Métodos de diseños de filtros IIR a partir de prototipos analógicos.

4.3.1.- Aproximación del impulso invariante.

Se diseña un filtro digital IIR cuya respuesta impulsional, $h(n)$, sea la versión muestreada de la respuesta impulsional del filtro analógico equivalente: $h_d(n) = h_a(nT)$, con $n=0,1,2,\dots$

Veamos el caso de partir de un filtro analógico con N polos simples.

$$H(s) = \sum_{k=1}^N \frac{A_k}{s - s_k}$$

su respuesta impulsional $h_a(t)$ es:

$$h_a(t) = \sum_{k=1}^N A_k \cdot e^{s_k \cdot t} \quad \Rightarrow \quad h_d(n) = \sum_{k=1}^N A_k \cdot e^{s_k \cdot nT}$$

con lo que, considerando que el filtro es causal, la transformada Z del filtro digital queda:

$$H(z) = TZ[h(n)] = \sum_{n=0}^{\infty} \left[\sum_{k=1}^N A_k \cdot e^{s_k \cdot nT} \right] \cdot z^{-n} = \sum_{k=1}^N A_k \cdot \frac{1}{1 - e^{s_k \cdot T} \cdot z^{-1}}$$

Por tanto, el procedimiento es claro, se detectan los polos del filtro analógico $\{s_k\}$, se descompone $H(s)$ como suma de monomios y se obtiene $H(z)$.

Comparando $H(s)$ y $H(z)$ observamos que por cada polo en s_k tenemos un polo en

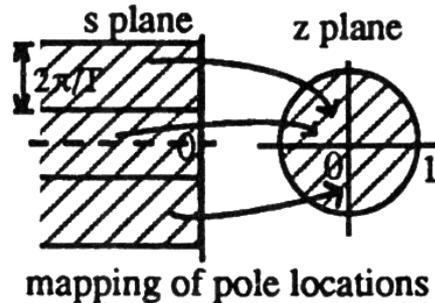
$$z_k = e^{s_k T}$$

Veamos qué efecto tiene esta transformación sobre la posición de los polos al pasar del dominio de Laplace al Z.

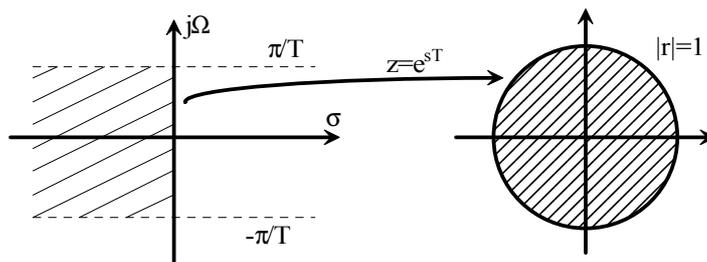
Transform

$$H_a(s) = \sum_{i=1}^N \frac{b_i}{(s - p_i)} \quad \text{to}$$

$$H(z) = \sum_{i=1}^N \frac{b_i}{[1 - z^{-1} \exp(p_i T)]}$$



Extraído de: The Circuits and Filters Handbook. Editor Wai-Kay Chen



Características:

Se mantiene la respuesta impulsional de ambos filtros.

La estabilidad se mantiene (Filtros analógicos estables dan lugar a filtros digitales estables).

La relación entre frecuencias es lineal ya que:

$$\left. \begin{array}{l} \text{Analógico: } s_k = j\Omega \\ \text{Digital: } \left. \begin{array}{l} z = e^{s_k T} = e^{j\Omega T} \\ z = e^{j\omega} \end{array} \right\} \rightarrow \omega = \Omega T \end{array} \right\}$$

Inconveniente:

Dado que este método produce un “mapeado muchos a uno”; se va a producir *aliasing*. Esto es sencillo de ver si tenemos en cuenta que estamos muestreando la respuesta impulsional del filtro analógico. El contenido frecuencial de h(n), es la respuesta en frecuencia del filtro

analógico. Sabemos que en los filtros ideales su respuesta no puede ser cero en una banda por lo que independientemente del valor de T , siempre habrá una contribución de $H(\Omega)$ que no verificará el teorema de muestreo. Si la frecuencia de muestreo es $\Omega_s = 2\pi F_s$, todas las frecuencia analógicas por encima de $\Omega_s/2$ producirán *aliasing*. Para que el método funcione adecuadamente el filtro analógico debe verificar $H(\Omega) \cong 0 \quad |\Omega| > \frac{\Omega_s}{2}$. Por esta razón este método no es adecuado para el diseño de filtros pasa-alta ni elimina-banda. Además, si tenemos en cuenta los prototipos analógicos considerados, los filtros de Tchebyshev II y los Elípticos presentarán peor comportamiento debido al rizado en la banda no pasante.

4.3.2.- Aproximación de derivadas (Analogía).

Se basa en establecer una analogía entre el operador diferencial en el dominio de Laplace y el Z.

En sistemas analógicos, el diferencial es: $H(s)=s$

En sistemas discretos podemos realizar una primera aproximación de la derivada que es la llamada '*backward difference*' y que se define como:

$$\left. \frac{dy(t)}{dt} \right|_{t=nT} = \frac{y(nT) - y(nT - T)}{T} = \frac{y(n) - y(n-1)}{T}$$

con lo que su transformada Z es:

$$H(z) = \frac{1 - z^{-1}}{T}$$

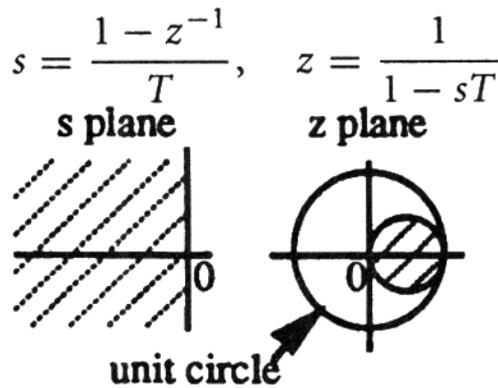
con lo que, identificando términos:

$$s = \frac{1 - z^{-1}}{T} \quad \Rightarrow \quad z = \frac{1}{1 - s \cdot T}$$

Veamos qué efecto tiene esta transformación sobre la posición de los polos al pasar del dominio de Laplace al Z.

Si hacemos el cambio $s = \sigma + j\Omega \quad z = \frac{1}{1 - (\sigma + j\Omega)T}$

$$\begin{aligned} \sigma = 0 \quad \Omega = 0 &\rightarrow z = 1 \\ \sigma = 0 \quad \Omega = +\infty &\rightarrow z = 0^- \\ \sigma = 0 \quad \Omega = -\infty &\rightarrow z = 0^+ \\ \sigma < 0 \quad \Omega &\rightarrow |z| < 1 \end{aligned}$$



Extraído de: The Circuits and Filters Handbook. Editor Wai-Kay Chen

Como se observa, la limitación es evidente puesto que, partiendo de filtros analógicos estables, no pueden diseñarse filtros pasa alta, al no poder ubicar polos fuera del círculo sombreado.

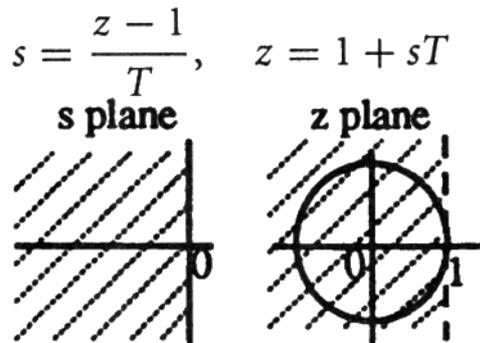
Otra posibilidad sería plantear alguna otra aproximación al operador diferencial como por ejemplo el cálculo de la derivada a partir de la 'forward difference':

$$\left. \frac{dy(t)}{dt} \right|_{t=nT} = \frac{y(nT+T) - y(nT)}{T} = \frac{y(n+1) - y(n)}{T}$$

que genera una relación de la forma:

$$s = \frac{z-1}{T} \quad \Rightarrow \quad z = 1 + s \cdot T$$

que, observando el efecto que esta transformación tiene sobre la posición de los polos al pasar del dominio de Laplace al Z,



Extraído de: The Circuits and Filters Handbook. Editor Wai-Kay Chen

no da lugar a un método de interés puesto que filtros analógicos estables no siempre da lugar a filtros digitales estables.

4.3.3.- Transformación bilineal.

Ninguno de los métodos anteriores parece resolver de forma adecuada la transformación del dominio analógico al digital. Veamos qué condiciones debería cumplir una transformación para ser óptima:

1. Dada una función real y racional en S, la función resultante en Z, será también racional y con coeficientes reales.
2. Dado un filtro analógico estable, el filtro digital resultante también lo será (el semiplano izquierdo del dominio de Laplace debe transformarse dentro del círculo de radio unidad en el dominio Z).
3. En particular, interesa una función reactancia que haga corresponder al eje imaginario del plano de Laplace, la circunferencia de radio unidad.
4. Dado un orden del filtro analógico, el filtro digital resultante será del mismo orden.

Existe una relación que cumple todas estas condiciones, la llamada transformación bilineal:

$$s = \frac{2}{T} \cdot \left(\frac{1 - z^{-1}}{1 + z^{-1}} \right) = \frac{2}{T} \cdot \frac{z - 1}{z + 1}$$

Así como en la transformación del impulso invariante el filtro analógico y digital tienen la misma respuesta impulsional, la transformación bilineal asegura que ambos filtros tienen una respuesta aproximadamente idéntica ante cualquier excitación.

Uno de los procedimientos para justificar esta expresión es obtener el equivalente digital de un integrador analógico.

$$H(s) = \frac{1}{s} \quad \rightarrow \quad h(t) = L^{-1}\{H(s)\} = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}$$

La salida ante una excitación cualquiera viene dada por $y(t) = x(t) * h(t)$

La convolución continua viene dada por:

$$y(t) = \int_0^t x(\tau)h(t-\tau) d\tau \quad \text{para } 0 < \tau < t_1, t_2 \text{ tenemos:}$$

$$y(t_2) - y(t_1) = \int_0^{t_2} x(\tau)h(t_2-\tau) d\tau - \int_0^{t_1} x(\tau)h(t_1-\tau) d\tau$$

$$\text{como } 0 < \tau < t_1, t_2 \quad h(t_2-\tau) = h(t_1-\tau) = 1 \quad \rightarrow \quad y(t_2) - y(t_1) = \int_{t_1}^{t_2} x(\tau) d\tau$$

Si aproximamos esta integral utilizando el método de los trapecios:

$$y(t_2) - y(t_1) \cong (t_2 - t_1)x(t_1) + \frac{1}{2}(t_2 - t_1)(x(t_2) - x(t_1))$$

$$y(t_2) - y(t_1) \cong \left(\frac{t_2 - t_1}{2}\right)(x(t_2) + x(t_1))$$

Si hacemos $t_2 = nT$ $t_1 = nT - T$

$$y(nT) - y(nT - T) \cong \left(\frac{T}{2}\right)(x(nT) + x(nT - T))$$

tomando transformadas Z

$$H(z) = \frac{T}{2} \frac{1 + z^{-1}}{1 - z^{-1}}$$

Comparando con la expresión del integrador analógico obtenemos que la relación entre las variables s y z viene dada por:

TRANSFORMACIÓ BILINEAL:

$$s = \frac{2}{T} \frac{z-1}{z+1} \quad z = \frac{\frac{2}{T} + s}{\frac{2}{T} - s}$$

Dado que hemos obtenido esta expresión considerando el mismo comportamiento del integrador para cualquier entrada, al filtro digital se le supone el mismo comportamiento. La diferencia entre el filtro analógico y el digital decrece a medida que $T \rightarrow 0$, ya que la aproximación de los trapecios es más exacta.

Veamos qué efecto tiene esta transformación sobre la posición de los polos al pasar del dominio de Laplace al Z.

Supongamos:

$$z = r \cdot e^{j\omega} \quad y \quad s = \sigma + j \cdot \Omega$$

entonces:

$$s = \frac{2}{T} \cdot \frac{z-1}{z+1} = \frac{2}{T} \cdot \left(\frac{r^2 - 1}{1 + 2 \cdot r \cdot \cos \omega + r^2} + j \cdot \frac{2 \cdot r \cdot \sin \omega}{1 + 2 \cdot r \cdot \cos \omega + r^2} \right)$$

con lo cual,

$$\sigma = \frac{2}{T} \cdot \frac{r^2 - 1}{1 + 2 \cdot r \cdot \cos \omega + r^2} \quad y \quad \Omega = \frac{2}{T} \cdot \frac{2 \cdot r \cdot \sin \omega}{1 + 2 \cdot r \cdot \cos \omega + r^2}$$

O bien

$$r = \sqrt{\frac{\left(\frac{2}{T} + \sigma\right)^2 + \Omega^2}{\left(\frac{2}{T} - \sigma\right)^2 + \Omega^2}} \quad \omega = \tan^{-1} \left(\frac{\Omega}{\frac{2}{T} + \sigma} \right) + \tan^{-1} \left(\frac{\Omega}{\frac{2}{T} - \sigma} \right)$$

De estas expresiones deducimos que:

$$\begin{aligned} \sigma > 0 &\rightarrow r > 1 \\ \sigma = 0 &\rightarrow r = 1 \\ \sigma < 0 &\rightarrow r < 1 \end{aligned}$$

Es decir;

- El semiplano izquierdo ($\sigma < 0$) se transforma en el interior de la circunferencia unidad, luego un sistema analógico estable da lugar a un sistema digital estable.
- El eje imaginario ($\sigma = 0$) se transforma en la circunferencia unidad

$$\sigma = 0 \rightarrow r = 1 \quad \omega = 2 \cdot \tan^{-1}\left(\frac{T\Omega}{2}\right) \quad \begin{array}{l} \Omega \rightarrow +\infty \Rightarrow \omega \rightarrow \pi \\ \Omega \rightarrow -\infty \Rightarrow \omega \rightarrow -\pi \end{array}$$

Esta relación entre frecuencias nos asegura que si $\Omega_1 < \Omega_2 \rightarrow \omega_1 < \omega_2$; es decir un pasa-baja analógico dará lugar a un pasa-baja digital, y análogamente con el resto de prototipos.

$$s = \frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}}, \quad z = \frac{(2/T) + s}{(2/T) - s}$$

Extraído de: The Circuits and Filters Handbook. Editor Wai-Kay Chen

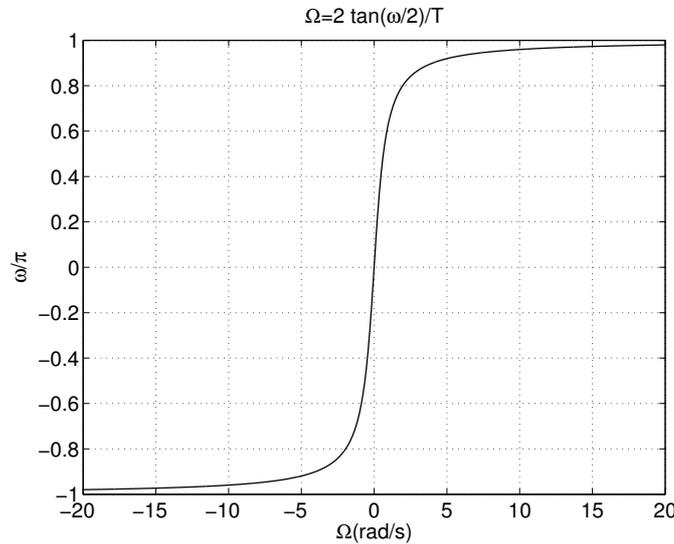
La relación entre las frecuencias analógica y digitales es **no lineal** y viene dada por:

$$\omega = 2 \cdot \tan^{-1}\left(\frac{T\Omega}{2}\right) \quad \Omega = \frac{2}{T} \cdot \text{tg} \frac{\omega}{2} \quad \text{donde} \quad \omega = 2\pi \frac{F}{F_s}$$

Es decir, se lleva a cabo una compresión de las frecuencias analógicas. Este proceso de “Deformación” es lo que se denomina *warping*.

Si tenemos que diseñar un filtro digital con una frecuencia de corte ω_c deberemos partir de un filtro analógico de frecuencia de corte $\Omega_c = \frac{2}{T} \cdot \text{tg} \frac{\omega_c}{2}$, de manera que al realizar la transformación bilineal la frecuencia de corte sea la correcta. El proceso de obtención de las frecuencias analógicas adecuadas se denomina *prewarping*.

Como consecuencia de esta relación no lineal entre frecuencias los filtros diseñados por la Transformación bilineal verán modificada su respuesta en fase; es decir, si el filtro analógico presenta fase lineal el filtro digital no la presentará.



$$\omega = 2 \cdot \operatorname{tg}^{-1} \frac{\Omega \cdot T}{2} \quad \text{donde} \quad \omega = 2 \cdot \pi \cdot \frac{f}{F_s}$$

La transformación bilineal realiza un “mapeado”, entre los dominios de Laplace y Z con una correspondencia 1 a 1 debido a la compresión de frecuencias (*prewarping*), por esta razón no se produce *aliasing* como ocurre con el método del impulso invariante.

Ejemplo:

ENUNCIADO: Se desea diseñar un filtro digital pasa-baja de orden 1, con frecuencia de corte de 1 kHz, frecuencia de muestreo 10 kHz y ganancia en continua de 5 mediante la T. Bilineal.

El filtro analógico correspondiente es: $H(s) = \frac{G \cdot \Omega_0}{s + \Omega_0}$, con $\Omega_0 = 2 \cdot \pi \cdot 1000$ y $G = 5$

Si aplicamos directamente la transformación, el filtro digital resultante será un pasa-baja pero su frecuencia de corte no estará en 1 kHz si no que se transformará según la ecuación anteriores. Por tanto, si lo que queremos es obtener un filtro digital con $F_c = 1$ kHz

he de partir de un filtro analógico con la frecuencia de corte adecuada que es la resultante de resolver:

$$\Omega = \frac{2}{T} \cdot \operatorname{tg} \frac{\omega}{2} = \frac{2}{T} \cdot \operatorname{tg} \frac{2 \cdot \pi \cdot 1000 / 10000}{2} = \frac{2}{T} \cdot \operatorname{tg}(0.1 \cdot \pi)$$

con lo cual:

$$H(z) = \frac{5 \frac{2}{T} \cdot \operatorname{tg}(0.1 \cdot \pi)}{\frac{2}{T} \cdot \frac{z-1}{z+1} + \frac{2}{T} \cdot \operatorname{tg}(0.1 \cdot \pi)} = \frac{5 \cdot \operatorname{tg}(0.1 \pi) \cdot (z+1)}{(\operatorname{tg}(0.1 \cdot \pi) + 1) \cdot z + (\operatorname{tg}(0.1 \cdot \pi) - 1)}$$

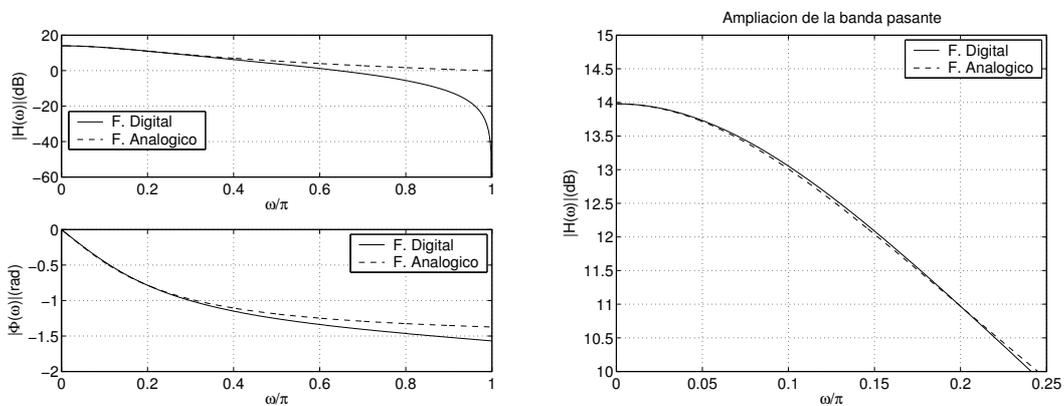
la ganancia en continua es:

$$|H(z)|_{z=1} = 5$$

y, por tanto:

$$H(z) = \frac{5 \cdot 0.3249 \cdot (z+1)}{1.3249z - 0.6751} \rightarrow H(z) = \frac{1.2261 \cdot (1+z^{-1})}{1 - 0.5095z^{-1}}$$

cuya respuesta en frecuencia aparece en la figura. Observamos que a medida que la frecuencia aumenta la diferencia entre la respuesta del filtro analógico y digital es mayor.



4.3.4.- Transformada Z adaptada. (*Matched Z transformation*)

Este procedimiento lo podemos considerar como una variación del impulso invariante. Dada una función de transferencia en el dominio de Laplace, factorizada

$$H(s) = G \frac{\prod_{i=1}^M (s - q_i)}{\prod_{i=1}^M (s - p_i)}$$

realizamos la transformación a digital de acuerdo al método utilizado por el impulso invariante para transformar cada uno de los polos. Da lugar a una expresión del tipo

$$H(z) = b_o (z + 1)^L \frac{\prod_{i=1}^M (z - e^{q_i T})}{\prod_{i=1}^M (z - e^{p_i T})} \quad L \text{ es el número de ceros en el infinito de } H(s)$$

Esta transformación da lugar a filtros digitales estables pero produce *aliasing* como ocurre con el método del impulso invariante.

4.3.5.- Mapeado de polos y ceros (*Pole/Zero mapping*)

La transformación bilineal, si hacemos los cálculos manualmente es tediosa de aplicar, por lo que se suele recurrir a una aproximación más sencilla. Dada H(s)

$$H(s) = G \frac{\prod_{i=1}^M (s - q_i)}{\prod_{i=1}^M (s - p_i)}$$

como la relación entre s y z para la transformación bilineal viene dada

por

$$z = \frac{\frac{2}{T} + s}{\frac{2}{T} - s}$$

podemos determinar las posiciones correspondientes de los ceros y los polos en el plano z

$$z_i = \frac{\frac{2}{T} + q_i}{\frac{2}{T} - q_i} \quad r_i = \frac{\frac{2}{T} + p_i}{\frac{2}{T} - p_i} \quad \text{y obtener la funci3n de transferencia en Z como:}$$

$$H(z) = b_o (z+1)^L \frac{\prod_{i=1}^M (z - z_i)}{\prod_{i=1}^M (z - r_i)} \quad \text{donde el t3rmino } (z+1)^L \text{ se origina de los ceros en el infinito.}$$

Ejemplo: Utilizando la transformaci3n ‘‘Pole Zero mapping’’, obt3n el filtro digital correspondiente al ejemplo anterior. $F_s=10\text{kHz}$, $F_c=1\text{kHz}$, $G=5$

Sabemos que $H(s) = \frac{5 \cdot \Omega_0}{s + \Omega_0}$, con $\Omega_0 = 2 \cdot \pi \cdot 1000$. El sistema tiene un cero en el infinito y un

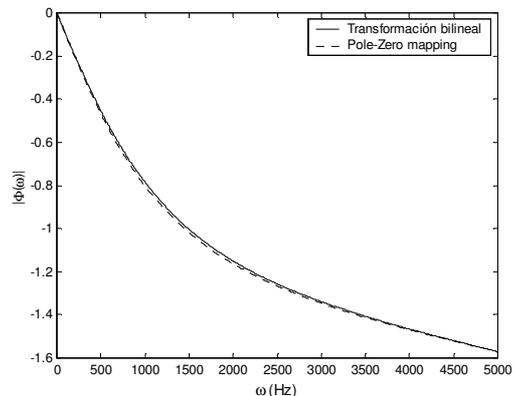
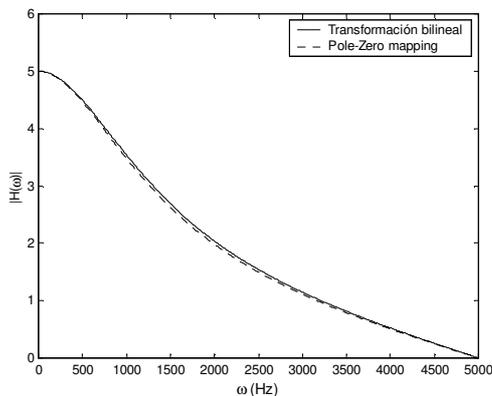
polo en $s = -\Omega_0$ que en el plano z se corresponden con un cero en $z_k = -1$

$$\text{y un polo en } r_i = \frac{\frac{2}{T} + p_i}{\frac{2}{T} - p_i} = \frac{2 \cdot 10000 - 2 \cdot \pi \cdot 1000}{2 \cdot 10000 + 2 \cdot \pi \cdot 1000} = 0.5219$$

Si imponemos adem3s que la ganancia en continua sea 5 para el filtro digital obtenido es f3cil determinar que el factor de ganancia necesario es $G = 1.1953$ y

$$H(z) = 1.1953 \frac{1 + z^{-1}}{1 - 0.5219z^{-1}}$$

En las siguientes gr3ficas mostramos los resultados de la respuesta en m3dulo y fase del filtro dise~nado por este m3todo y mediante la transformaci3n bilineal.



Resumen: Transformaciones Continuo/Discreto

TABLE 83.1 Various Types of Analog-to-Digital Transformation

Type of Transformation	Principle	Resulting mapping	Advantages	Disadvantages
Backward difference approximation	$\frac{dy}{dt} \approx \frac{y(n) - y(n-1)}{T}$	$s = \frac{1-z^{-1}}{T}, \quad z = \frac{1}{1-sT}$ 	1. Stable analog filter yields stable digital filter.	1. Left-half s plane is not mapped onto unit circle on z plane. Pole locations will be in the circle centered at 0.5 with radius 0.5 on the z plane.
Forward difference approximation	$\frac{dy}{dt} \approx \frac{y(n+1) - y(n)}{T}$	$s = \frac{z-1}{T}, \quad z = 1+sT$ 		1. Stable analog filter does not yield stable digital filter. 2. Left-half s plane is not mapped onto unit circle on z plane.
Impulse invariant method	Sample the analog impulse response: $h(n) = h_a(nT)$	Transform $H_a(s) = \sum_{i=1}^N \frac{b_i}{(s-p_i)}$ to $H(z) = \sum_{i=1}^N \frac{b_i}{[1-z^{-1} \exp(p_i T)]}$ 	1. Preserve shape of impulse response. 2. Stable analog filter yields stable digital filter. 3. The analog frequency and digital frequency are linearly related, $\lambda T = 2\pi f, -0.5 \leq f \leq 0.5$.	1. Aliasing in the frequency domain (due to many-to-one mapping from s plane to z plane). $T \cdot H(f) = \sum_{k=-\infty}^{\infty} H_a\left(2\pi \frac{f+k}{T}\right)$
Bilinear transformation	Approximate $y(t) = \int_{t-T}^t y'(\tau) d\tau + y(t-T)$ by $y(n) = \frac{T}{2} [y'(n) + y'(n-1)] + y(n-1)$	$s = \frac{2}{T} \frac{1-z^{-1}}{1+z^{-1}}, \quad z = \frac{(2/T)+s}{(2/T)-s}$ 	1. Stable analog filter yields stable digital filter. 2. Left-half s plane is mapped onto unit circle on z plane, a one-to-one mapping.	1. Frequency warping—nonlinear relation between analog frequency and digital frequency.
Matched z transformation	Map each pole and zero on s plane directly to pole and zero on the z plane.	Transform $H_a(s) = \prod_{i=1}^N \frac{(s-q_i)}{(s-p_i)}$ to $H(z) = \prod_{i=1}^N \frac{[1-z^{-1} \exp(q_i T)]}{[1-z^{-1} \exp(p_i T)]}$ 	1. Stable analog filter yields stable digital filter.	1. Aliasing in the frequency domain (due to many-to-one mapping from s plane to z plane).

Extraído de: The Circuits and Filters Handbook. Editor Wai-Kay Chen

4.4.- **Métodos directos de diseño de filtros IIR.**

Aunque lo incluyamos en el capítulo de filtros IIR, los métodos directos de diseño sirven tanto para filtros AR, como MA, y en general sistemas ARMA. Estos métodos se basan en la teoría de la aproximación, imponiendo restricciones en cuanto a la desviación de la respuesta en frecuencia del filtro respecto del filtro ideal. Estos métodos son especialmente útiles cuando el filtro se debe obtener a partir de datos experimentales.

El objetivo es determinar los coeficientes del filtro digital en base a minimizar algún parámetro de diseño. El número de procedimientos existentes es muy numeroso por lo que únicamente analizaremos los más sencillos.

4.4.1.- **Aproximación de Padé.**

Dado un sistema con función de transferencia:

$$H(z) = \frac{\sum_{k=0}^M b_k \cdot z^{-k}}{1 + \sum_{k=1}^N a_k \cdot z^{-k}} = \sum_{k=0}^{+\infty} h(k) \cdot z^{-k}$$

pretendemos obtener los $L=M+N+1$ coeficientes, $\{a_k\}$ y $\{b_k\}$, a partir de la minimización de algún criterio de error. Éste método se plantea minimizar la suma de los errores cuadráticos entre la respuesta impulsional ideal (deseada) y la real:

$$\mathcal{E} = \sum_{n=0}^U [h_d(n) - h(n)]^2$$

donde $h_d(n)$ es la respuesta deseada del filtro y $h(n)$ la real; U es un límite superior seleccionado por el diseñador.

En general, $h(n)$ es una función no lineal de $\{a_k\}$ y $\{b_k\}$, sin embargo, si $U=N+M$, es posible hacer coincidir perfectamente las respuestas real y deseada para $0 \leq n \leq M+N$.

Si hacemos $h_d(n) = h(n)$ $0 \leq n \leq M + N$, el error cometido será:

$$\mathcal{E} = \sum_{n=N+M+1}^{\infty} [h_d(n) - h(n)]^2$$

Planteemos el problema a través de la ecuación en diferencias:

$$y(n) = -\sum_{k=1}^N a_k y(n-k) + \sum_{k=0}^M b_k x(n-k)$$

cuando $x(n)=\delta(n)$, tenemos:

$$h(n) = -\sum_{k=1}^N a_k h(n-k) + \sum_{k=0}^M b_k \delta(n-k)$$

Si vamos dando valores a n y tenemos en cuenta que $\delta(n-k)=0$ excepto para $n=k$, podemos plantear el siguiente sistema de ecuaciones:

$$h(n) = -\sum_{k=1}^N a_k h(n-k) \quad n > M \quad (\text{N ecuaciones con N incógnitas})$$

$$h(n) = -a_1 h(n-1) - a_2 h(n-2) - \dots - a_N h(n-N) + b_n \quad 0 \leq n \leq M$$

Con estas ecuaciones lineales se pueden hallar los coeficientes $\{a_k\}$ y $\{b_k\}$:

1. Haciendo $h(n)=h_d(n)$ con $M < n \leq M+N$, obtengo $\{a_k\}$, y
2. Haciendo $h(n)=h_d(n)$ con $0 < n \leq M$, obtengo $\{b_k\}$.

El cálculo de los coeficientes a_k se puede realizar matricialmente de acuerdo con la siguiente expresión:

$$\begin{bmatrix} h(M+1) \\ \vdots \\ h(M+N) \end{bmatrix} = - \begin{bmatrix} h(M) & h(M-1) & \dots & h(M-N+1) \\ \vdots & \vdots & \ddots & \vdots \\ h(M+N-1) & h(M+N-2) & \dots & h(M) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix}$$

$$\text{Matricialmente: } \mathbf{a} = \mathbf{H}^{-1} \mathbf{h}_{(M+1, M+N)}$$

que podemos resolver directamente sin más que invertir la matriz \mathbf{H} . Conocidos los valores de

$$a_k, \text{ podemos calcular } b_n = \sum_{k=0}^N a_k h(n-k) \quad 0 \leq n \leq M$$

El grado de fiabilidad de este método depende del número de coeficientes seleccionado. Aproxima perfectamente cuando el sistema buscado presenta una función de transferencia $H(z)$ racional y sabemos el número de ceros y polos (orden del numerador y denominador), lo que, en la práctica resulta problemático.

Ejemplo: Supongamos un sistema con una respuesta impulsional $h_d(n) = 2\left(\frac{1}{2}\right)^n u(n)$

Observando la $h(n)$, deducimos que tiene un solo polo por lo que asumimos una función de transferencia de la forma:

$$H(z) = \frac{b_0 + b_1 \cdot z^{-1}}{1 + a_1 \cdot z^{-1}}$$

Reproduciendo las ecuaciones anteriores:

$$h(n) = -a_1 \cdot h(n-1) + b_0 \cdot \delta(n) + b_1 \cdot \delta(n-1)$$

para $n > 1$, $h(n) = -a_1 \cdot h(n-1)$

$$\text{igualando } h(n) \text{ a } h_d(n), 2 \cdot \left(\frac{1}{2}\right)^n \cdot u(n) = -a_1 \cdot 2 \cdot \left(\frac{1}{2}\right)^{n-1} \cdot u(n-1) \Rightarrow a_1 = -\frac{1}{2}$$

Resolviendo la ecuación para $n=0$ obtenemos que $b_0=2$ y para $n=1$ tenemos que $b_1=0$, con lo que:

$$H(z) = \frac{2}{1 - 0.5 \cdot z^{-1}}$$

que es lo que esperábamos obtener.

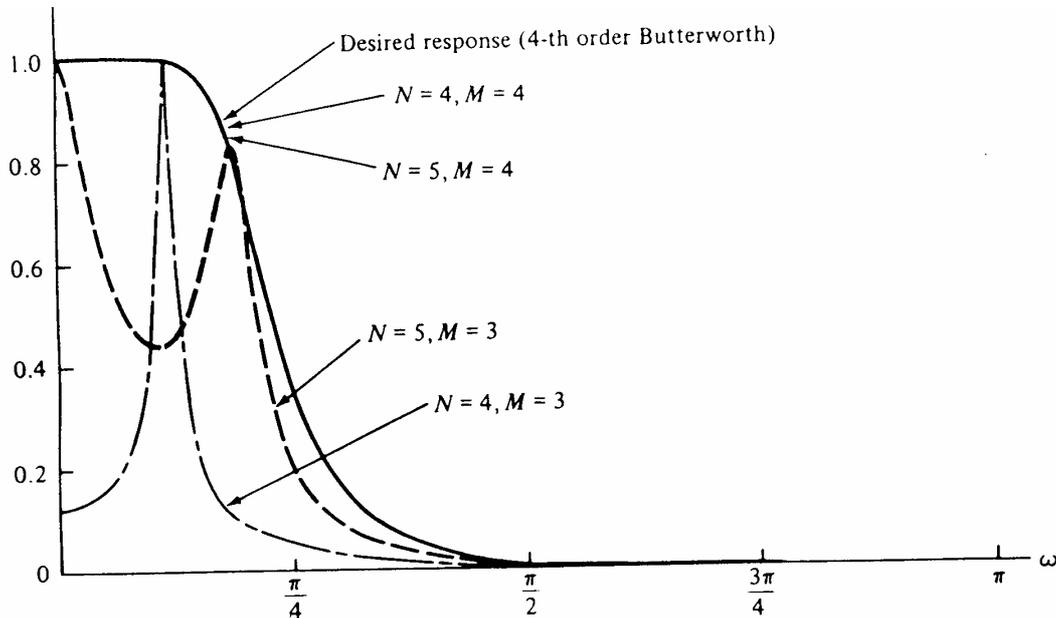
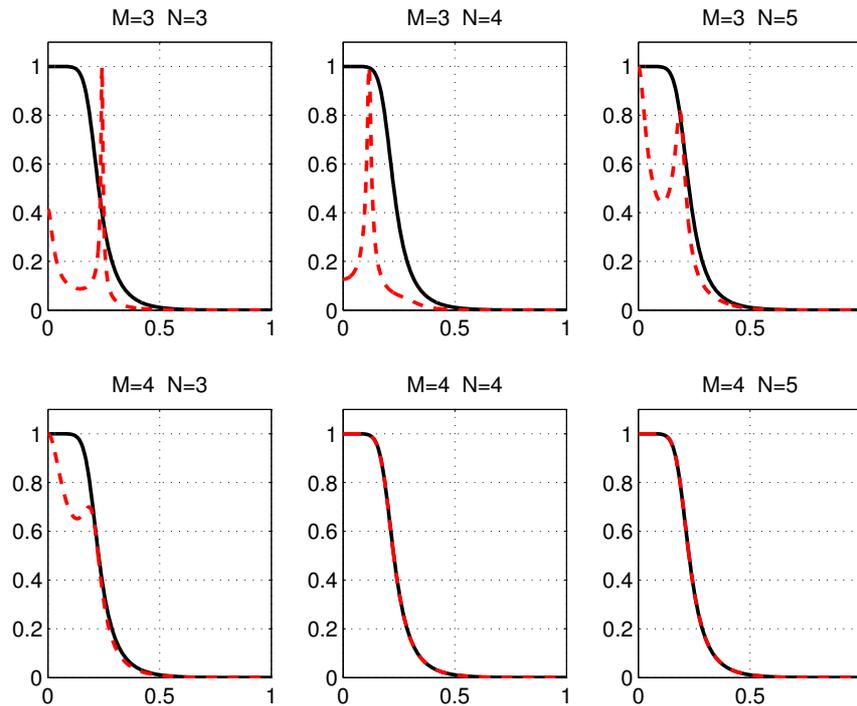
En el caso más real de disponer de la respuesta impulsional como una serie de puntos y desconocer el orden del filtro buscado, podríamos plantearnos qué ocurre al considerar distintos valores para el orden del numerador y denominador.

Ejemplo: consideremos que el filtro buscado tiene la respuesta función de transferencia

$$\text{siguiente: } H_d(z) = \frac{4.8334 \cdot 10^{-3} \cdot (z+1)^4}{(z^2 - 1.3205 \cdot z + 0.6326) \cdot (z^2 - 1.0482 \cdot z + 0.2959)}$$

Podemos aplicar el método anterior, y analizar cual es la función de transferencia obtenida para diversas elecciones de N y M, como se observa en las gráficas siguientes.

Obsérvese en la siguiente gráfica que la respuesta en magnitud es más sensible al número de polos que de ceros (aunque “acertemos” el número de ceros la respuesta del filtro diseñado es bastante pobre) y que si sobredimensionamos el orden no aparecen efectos secundarios.



Extraído de: Tratamiento Digital de Señales. J.G. Proakis

El método de la aproximación de Padé consigue un ajuste perfecto entre la respuesta impulsional del filtro obtenido y la del filtro real en los $L=M+N+1$ puntos utilizados. Sólo funciona adecuadamente para filtros con muchos ceros y polos

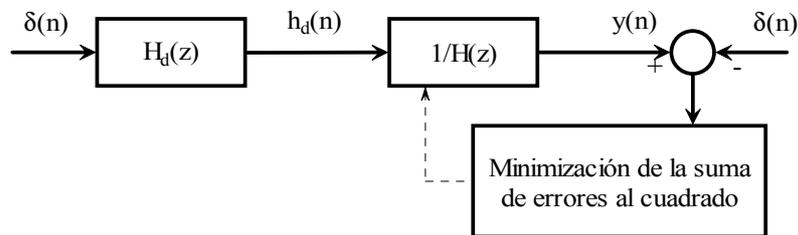
4.4.2.- Aproximación por mínimos cuadrados (Métodos Prony y de Shanks).

Un método alternativo para diseñar filtros IIR es afrontar el problema de la minimización de la diferencia al cuadrado entre la respuesta del filtro ideal y la real, es decir, plantear el problema de la identificación de sistemas. La figura siguiente muestra dicha estructura. La idea es determinar los coeficientes del sistema $H(z)$ de manera que colocado en cascada con el sistema que queremos modelizar obtengamos como resultado una señal $(y(n))$ que en el dominio temporal debería ser un impulso, cuando la modelización es exacta. Si definimos como criterio de error la suma de $y^2(n)$, una podemos obtener los coeficientes imponiendo de dicho error sea mínimo.

Supongamos el caso más simple de considerar un filtro con solo polos:

$$H(z) = \frac{b_0}{1 + \sum_{k=1}^N a_k \cdot z^{-k}}$$

Consideremos un esquema de ecualización de sistemas:



donde modificamos los coeficientes del filtro $1/H(z)$ para anular el efecto de $H_d(z)$, es decir, que la salida, $y(n)$, sea lo más parecida posible a la entrada, $\delta(n)$. (Idealmente $\frac{H_d(z)}{H(z)} = 1$ e

$$y(n) = \delta(n))$$

La ecuación en diferencias del sistema inverso será:

$$y(n) = \frac{1}{b_0} \cdot \left[h_d(n) + \sum_{k=1}^N a_k \cdot h_d(n-k) \right]$$

Para $n=0$,

$$y(0) = \frac{1}{b_0} \cdot \left[h_d(0) + \sum_{k=1}^N a_k \cdot h_d(0-k) \right] = \frac{1}{b_0} \cdot h_d(0) = 1 \quad \Rightarrow \quad b_0 = h_d(0)$$

Para $n > 0$, la salida, $y(n)$ debe ser cero, lo que minimizaremos será la suma de los errores al cuadrado:

$$\mathcal{E} = \sum_{n=1}^{+\infty} y^2(n) = \frac{1}{h_d^2(0)} \cdot \sum_{n=1}^{+\infty} \left[h_d(n) + \sum_{k=1}^N a_k \cdot h_d(n-k) \right]^2$$

Para que la función de error sea mínima, las derivadas parciales respecto de los coeficientes del filtro deben ser cero y las derivadas segundas mayores que cero.

$$\frac{\partial \mathcal{E}}{\partial a_p} = 0 \quad \text{para } p = 1, 2, \dots, N$$

$$\frac{\partial^2 \mathcal{E}}{\partial a_p^2} > 0$$

$$\frac{\partial \mathcal{E}}{\partial a_p} = \frac{1}{h_d^2(0)} \cdot \sum_{n=1}^{+\infty} 2 \cdot \left[h_d(n) + \sum_{k=1}^N a_k \cdot h_d(n-k) \right] \cdot h_d(n-p) = 0$$

$$\sum_{n=1}^{+\infty} h_d(n) \cdot h_d(n-p) + \sum_{k=1}^N a_k \cdot \sum_{n=1}^{+\infty} h_d(n-k) \cdot h_d(n-p) = 0$$

definiendo la variable auxiliar: $r_{dd}(p) = \sum_{n=1}^{+\infty} h_d(n) \cdot h_d(n-p)$

obtenemos: $-r_{dd}(p) = \sum_{k=1}^N a_k \cdot r_{dd}(p-k) \quad 1 \leq p \leq N$

Si desarrollamos el sumatorio:

$$- \begin{bmatrix} r_{dd}(1) \\ \vdots \\ r_{dd}(N) \end{bmatrix} = \begin{bmatrix} r_{dd}(0) & r_{dd}(-1) & \dots & r_{dd}(-(N-1)) \\ r_{dd}(1) & r_{dd}(0) & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ r_{dd}((N-1)) & r_{dd}((N-2)) & \dots & r_{dd}(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix}$$

Expresión que puede escribirse en forma matricial como: $\mathfrak{R} \cdot \mathbf{a} = \mathbf{r}$, donde:

$\Rightarrow \mathfrak{R}$ es la matriz de correlación de $N \times N$ elementos

$\Rightarrow \mathbf{a}$ es un vector $N \times 1$ que contiene los coeficientes del filtro

$\Rightarrow \mathbf{r}$ es el vector de correlación con $N \times 1$ elementos, $\mathbf{r} = -\mathbf{r}_{dd}(n)$

El sistema de ecuaciones puede resolverse haciendo: $\mathbf{a} = \mathfrak{R}^{-1} \cdot \mathbf{r}$.

Es fácil ver que la matriz \mathfrak{R} es simétrica ya que el vector de autocorrelación es simétrico

$$(r_{dd}(-l) = r_{dd}(l) \quad 0 \leq l \leq N-1)$$

La expresión matricial $\mathbf{a} = \mathfrak{R}^{-1} \cdot \mathbf{r}$ se denomina Ecuación de Yule-Walker, y el procedimiento utilizado para su resolución es el algoritmo de Levinson-Durbin (función *levinson()*, en Matlab)

En un caso real, disponemos de un número finito, L, de muestras de $h_d(n)$, que debería ser mucho mayor que el orden del filtro, N. Por ello, se debe trabajar con estimadores de la correlación:

$$\hat{r}_{dd}(k) = \sum_{n=0}^{L-|k|} h_d(n) \cdot h_d(n-k) \quad 0 \leq k \leq N$$

con lo que: $\mathbf{a} = \mathfrak{R}^{-1} \cdot \hat{\mathbf{r}}$

Ejemplo: Métodos directos de diseño IIR

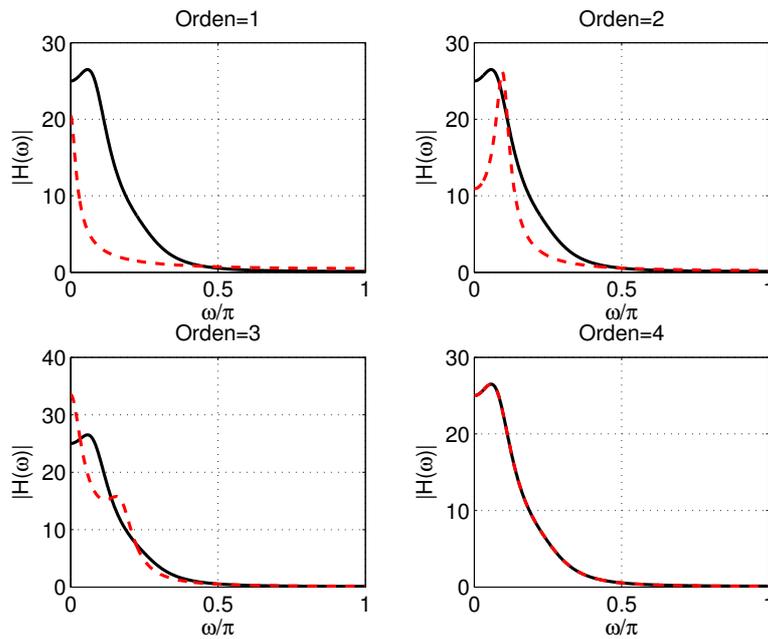
Consideremos que el sistema buscado tiene la siguiente función de transferencia. Utilizando el método anterior vamos a determinar los valores de los coeficientes para varios órdenes.

$$H(z) = \frac{1}{1 - 2.6z^{-1} + 2.78z^{-2} - 1.48z^{-3} + 0.34z^{-4}}$$

Resolvemos con Matlab

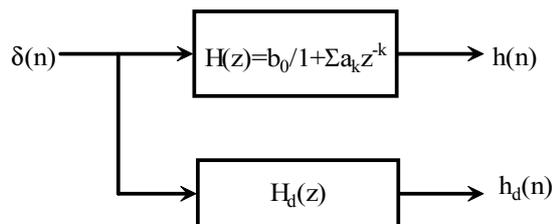
```
L=50; %Número de muestras de las respuesta impulsional
consideradas.
b=1
a=[1 -2.6 2.78 -1.48 0.34];
h=impz(b, a, L); %Vector de muestras h_d(n)
r=xcorr(h, 'none'); %vector de autocorrelación [-L:L]
r=r(L:end); %Cogemos desde lag=0
orden=2; % Repetimos para órdenes 2,3,4,5
A=levinson(r, orden) %El vector A contiene los coeficientes del
%filtro
%También se puede hacer directamente con lpc, a partir de h(n)
```

Orden	Coefficientes
1	1, -0.9511
2	1, -1.7821, 0.8737
3	1, 2.3709, 2.0746, -0.6739
4	1, 2.6, 2.78, 1.48, 0.34
5	1, 2.6, 2.78, 1.48, 0.34, 0



El procedimiento anterior sólo es válido para sistemas AR, veamos un método aplicable a sistemas ARMA.

Una aproximación alternativa emplea el concepto de predicción lineal por mínimos cuadrados, y da lugar a las mismas ecuaciones que fueron descritas anteriormente. Veamos su esquema:



Esta aproximación permite extender el tratamiento al análisis de un sistema con ceros y polos:

$$h(n) = -\sum_{k=1}^N a_k \cdot h(n-k) + \sum_{k=0}^M b_k \cdot \delta(n-k) \quad n \geq 0$$

$$h(n) = -\sum_{k=1}^N a_k \cdot h(n-k) + b_n \quad 0 \leq n \leq M$$

$$h(n) = -\sum_{k=1}^N a_k \cdot h(n-k) \quad n > M$$

Si consideramos que $H_d(z)$ tiene una estructura similar podemos emplear las salidas $h_d(n-k)$ $n > M$ para estimar $h_d(n)$.

Definimos el predictor lineal de $h_d(n)$ como:

$$\hat{h}_d(n) = -\sum_{k=1}^N a_k \cdot h_d(n-k) \quad n > M$$

de forma que la suma del cuadrado de los errores de predicción serán:

$$\mathcal{E} = \sum_{n=M+1}^{+\infty} [h_d(n) - \hat{h}_d(n)]^2 = \sum_{n=M+1}^{+\infty} \left[h_d(n) + \sum_{k=1}^N a_k \cdot h_d(n-k) \right]^2$$

que, como en el caso anterior, minimizando respecto a los $\{a_k\}$, obtenemos:

$$-\sum_{n=M+1}^{+\infty} h_d(n) \cdot h_d(n-p) = \sum_{k=1}^N a_k \cdot \sum_{n=M+1}^{+\infty} h_d(n-k) \cdot h_d(n-p)$$

definiendo la variable auxiliar: $r_{dd}(k, p) = \sum_{n=M+1}^{+\infty} h_d(n-k) \cdot h_d(n-p)$

obtenemos: $-r_{dd}(0, p) = \sum_{k=1}^N a_k \cdot r_{dd}(k, p) \quad 1 \leq p \leq N$

Cuando no haya ceros, M se reduce a cero y las expresiones coinciden con las del método anterior.

Si denotamos por $\{\hat{a}_k\}$ a los coeficientes calculados, para obtener los coeficientes del numerador $\{b_k\}$, que fijan los ceros del filtro, existen varios procedimientos:

APROXIMACIÓN DE PRONY

Utiliza el método de Padé para el cálculo de los coeficientes del numerador. Los parámetros $\{b_k\}$ se obtienen haciendo $h(n)=h_d(n)$ para $0 \leq n \leq M$, a partir de los valores estimados para los coeficientes del denominador, $\{\hat{a}_k\}$:

$$b_n = h_d(n) + \sum_{k=1}^N \hat{a}_k \cdot h_d(n-k) \quad 0 \leq n \leq M$$

Este método, aunque funciona bien para los términos del denominador, falla en los del numerador pues su cálculo no se basa en un método de optimización.

MÉTODO DE SHANKS

Shanks propuso una alternativa para determinar los $\{b_k\}$.

Con los $\{\hat{a}_k\}$ obtenidos según el procedimiento descrito, sintetizo un filtro con solo polos:

$$H_1(z) = \frac{1}{1 + \sum_{k=1}^N \hat{a}_k \cdot z^{-k}}$$

La respuesta de $H_1(z)$ frente a una entrada impulsional es:

$$v(n) = -\sum_{k=1}^N \hat{a}_k \cdot v(n-k) + \delta(n) \quad n \geq 0$$

Con $\{v(n)\}$ excitamos un filtro con solo ceros:

$$H_2(z) = \sum_{k=0}^M b_k \cdot z^{-k}$$

cuya respuesta es:

$$\hat{h}_d(n) = \sum_{k=0}^M b_k \cdot v(n-k)$$

con lo que la secuencia de error resulta:

$$e(n) = h_d(n) - \hat{h}_d(n) = h_d(n) - \sum_{k=0}^M b_k \cdot v(n-k)$$

Considerando como error el sumatorio de la secuencia de error al cuadrado:

$$\mathcal{E} = \sum_{n=0}^{+\infty} \left[h_d(n) - \sum_{k=0}^M b_k \cdot v(n-k) \right]^2$$

que minimizando respecto a b_k , conduce a:

$$\sum_{n=0}^{+\infty} h_d(n) \cdot v(n-p) = \sum_{k=0}^M b_k \cdot \sum_{n=0}^{+\infty} v(n-k) \cdot v(n-p)$$

Si definimos

$$r_{vv}(k, p) = \sum_{n=0}^{+\infty} v(n-k) \cdot v(n-p)$$

$$r_{vh}(0, p) = \sum_{n=0}^{+\infty} h_d(n) \cdot v(n-p)$$

Tenemos:

$$r_{vh}(0, p) = \sum_{k=0}^M b_k \cdot r_{vv}(k, p) \quad 0 \leq p \leq M$$

Ecuación análoga para el cálculo de los coeficientes b_k minimizando el error cuadrático medio. Los resultados son mejores que los proporcionados por el método de Prony.

Ejercicio propuesto:

Un sistema desconocido tiene una respuesta impulsional cuyas 5 primeras muestras son $h(n) = \{0.9, 1, 0.8, 0.7, 0.5, \dots\}$. Obtén un modelo de primer orden para este sistema utilizando la aproximación de Padé y la de Prony y Shanks

Solución:

$$H(z)_{\text{Padé}} = \frac{0.9 + 0.28z^{-1}}{1 - 0.8z^{-1}} \quad H(z)_{\text{Prony}} = \frac{0.9 + 0.2775z^{-1}}{1 - 0.8028z^{-1}} \quad H(z)_{\text{Shanks}} = \frac{0.9 + 0.2880z^{-1}}{1 - 0.8028z^{-1}}$$

4.5.- Transformaciones en frecuencia.

El diseño de filtros digitales se suele centrar en filtros pasa-baja. Análogamente al procedimiento empleado para el diseño analógico, sobre este filtro se llevarán a cabo transformaciones en frecuencia para obtener un filtro con la respuesta en frecuencia deseada. Si consideramos los métodos de diseño a partir de prototipos analógicos tenemos dos opciones para llevar a cabo dichas transformaciones:

1. Obtener un filtro analógico con las características deseadas, realizando las transformaciones en frecuencia en el dominio analógico, y luego aplicar alguno de los métodos de paso a filtro digital.

2. Transformar el filtro pasa bajo, analógico en uno digital, y luego aplicar unas transformaciones en frecuencia dentro del dominio digital.

Estos dos caminos solo resultan equivalentes en el caso de emplear la transformación bilineal para el paso a filtro digital.

Las funciones de transformación en el dominio digital deben cumplir las mismas condiciones que las especificadas para filtros analógicos y tratarse, por supuesto, de filtros pasa-todo, por tanto tendrán la forma:

$$g(z^{-1}) = \pm \prod_{k=1}^N \frac{z^{-1} - \alpha_k}{1 - \alpha_k \cdot z^{-1}}$$

Las transformaciones digitales son:

P. Baja → P. Baja	$z^{-1} \rightarrow \frac{z^{-1} - a}{1 - a \cdot z^{-1}}$	ω' nueva frecuencia de corte $a = \frac{\sin[(\omega - \omega')/2]}{\sin[(\omega + \omega')/2]}$
P. Baja → P. Alta	$z^{-1} \rightarrow -\frac{z^{-1} - a}{1 - a \cdot z^{-1}}$	ω' nueva frecuencia de corte $a = \frac{\cos[(\omega + \omega')/2]}{\cos[(\omega - \omega')/2]}$
P. Baja → P. Banda ω_l frecuencia inferior de la banda ω_u frecuencia superior de la banda ω frecuencia de corte del filtro pasa-baja	$z^{-1} \rightarrow -\frac{a_2 + a_1 \cdot z^{-1} + z^{-2}}{1 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2}}$	$K = \cot \frac{\omega_u - \omega_l}{2} \cdot \operatorname{tg} \frac{\omega}{2}$ $\alpha = \frac{\cos[(\omega_u + \omega_l)/2]}{\cos[(\omega_u - \omega_l)/2]}$ $a_1 = -\frac{2 \cdot \alpha \cdot K}{K + 1}$ $a_2 = \frac{K - 1}{K + 1}$
P. Baja → Elimina Banda ω_l frecuencia inferior de la banda ω_u frecuencia superior de la banda ω frecuencia de corte del filtro pasa-baja	$z^{-1} \rightarrow \frac{a_2 + a_1 \cdot z^{-1} + z^{-2}}{1 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2}}$	$K = \operatorname{tg} \frac{\omega_u - \omega_l}{2} \cdot \operatorname{tg} \frac{\omega}{2}$ $\alpha = \frac{\cos[(\omega_u + \omega_l)/2]}{\cos[(\omega_u - \omega_l)/2]}$ $a_1 = -\frac{2 \cdot \alpha}{K + 1}$ $a_2 = \frac{1 - K}{1 + K}$

8.4.- Discusión entre filtros FIR e IIR.

Para concluir el capítulo presentaremos, a modo de resumen, una tabla comparativa entre los filtros FIR e IIR:

FIR	IIR
<p>Son siempre estables</p> <p>Pueden diseñarse para que presente fase lineal, no distorsionando las características de fase y el retardo de grupo</p> <p>Requieren un mayor orden para valores de la atenuación y zonas de transición similares lo que supone un mayor retardo en la obtención de la respuesta y mayor consumo de memoria.</p>	<p>Aún cuando los polos estén situados dentro del círculo de radio unidad, pueden presentar inestabilidades si se implementan en sistemas con aritmética de coma fija.</p> <p>Presentan mayor error de cuantización en función del número de bits empleado para almacenar los coeficientes y las variables intermedias.</p> <p>Son más apropiados cuando deseamos obtener regiones de transición estrechas.</p>

En la actualidad, siguen apareciendo todos los meses métodos de diseño que se publican en las revistas especializadas, la proliferación de métodos de diseño se justifica por la diversidad de la casuística que se encuentra en su aplicación práctica, por ejemplo resulta obvio el diferente tratamiento que podemos dar a aplicaciones en las que preservar la fase es vital (audio) frente a otras que sólo se preocupan de la magnitud de la respuesta en frecuencia. Por otro lado, los métodos de diseño que entrañan una elevada carga computacional pueden resultar inadecuados en sistemas que requieren una variación de las características del filtro, y por tanto de sus coeficientes, en función de las condiciones del entorno.