Problema sobre filtros digitais

Suponha que dispõe de um sinal de fala de boa qualidade em formato digital resultado da amostragem a 48 KHz de um sinal contínuo previamente filtrado a 24 kHz. Suponha que pretende aplicar este sinal a um altifalante modesto de largura de banda 4 KHz (qualidade semelhante à existente na rede telefónica analógica). Uma vez que vai usar uma largura de banda muito limitada suponha que pretende compactar a representação do sinal de fala, o que pode ser feito por decimação mas obriga a uma filtragem do tipo passa baixo. Sabe-se que o ouvido humano é pouco sensível a variações de fase mas muito sensível a variações de amplitude.

a) Pretende-se que projecte para o efeito um filtro digital de 2ª ordem que tenha o melhor desempenho possível no âmbito do problema.

b) Suponha exactamente o mesmo problema em que o sinal agora em jogo é um sinal de vídeo onde os KHz agora são MHz e se sabe que a qualidade da imagem é seriamente afectada por distorções de fase. Resolva a alínea a) mas agora adaptada a este caso.

Resolução:

a) Como a distorção de fase não é importante para a aplicação em causa e a distorção de amplitude é muito importante o filtro mais adequado é um Butterworth ou Chebyshev tipo II, no entanto este último apresenta "equiripple" na banda de rejeição o que pode ser importante pois o sinal remanescente na banda de rejeição vai causar aliasing no processo de decimação. Apesar de para a mesma ordem o filtro de chebyshev apresentar um maior decaimento que o filtro de Butterrworth este último parece ser mais adequado ao problema em causa. Finalmente como o processo da decimação é sensível ao aliasing o método da transformação bilinear será o mais indicado para o problema. Temos então

$$\Omega_c = \frac{4}{24}\pi = \frac{\pi}{6}$$

Como o método da transformação bilinear requer compensação do modelo analógico podemos então calcular a freq. angular de corte como

$$w = \frac{2}{T}\tan(\Omega/2) \Rightarrow w_c = 2\tan\left(\frac{\pi}{12}\right) = 0{,}536$$

Como se trata de um filtro de segunda ordem os 2 pólos estão situados sobre a circunferência de raio w_c e a distâncias angulares de $3\pi/4$ e $5\pi/4$ (ver acetato 18). Os pólos são então:

$$p_{1,2} = w_c\left[\cos\left(3\pi/4\right) \pm j sen\left(3\pi/4\right)\right] = 0{,}379 \pm j0{,}379$$

Como

$$(s - p_1)(s - p_2) = s^2 - s(p_1 + p_2) + p_1 p_2 = s^2 + 0{,}7580s + 0{,}2873$$

Se pretendemos ganho DC unitário então a função de transferência do filtro analógico será

$$H_c(s) = \frac{0{,}2873}{s^2 + 0{,}7580s + 0{,}2873}$$

A função de transferência do filtro digital será

$$H(z) = H_c(s)\Big|_{s=2\frac{1-z^{-1}}{1+z^{-1}}} = \frac{0{,}2873}{s^2 + 0{,}7580s + 0{,}2873} = \frac{0{,}2873 + 0{,}5746z^{-1} + 0{,}2873z^{-2}}{5{,}8033 - 7{,}4254z^{-1} + 2{,}7713z^{-2}}$$

Ou normalizando para o y[n] (dividir tudo por 5,8033) obtém-se

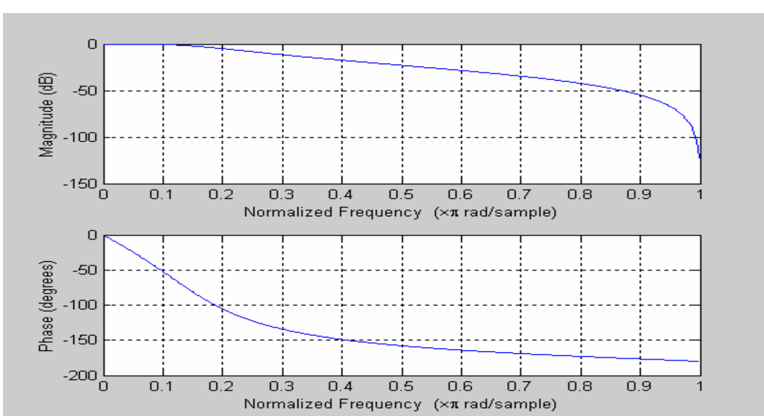
$$H(z) = \frac{0.0495 + 0.0990\mathrm{z}^{-1} + 0.0495\mathrm{z}^{-2}}{1.0000 - 1.2795\mathrm{z}^{-1} + 0.4775z^{-2}}$$

ou em termos de equação diferenças tem-se

$$y[n] = 1.2795\mathrm{y}[\mathrm{n}-1] - 0.4775\mathrm{y}[\mathrm{n}-2] + 0.0495\mathrm{x}[\mathrm{n}] + 0.0990\mathrm{x}[\mathrm{n}-1] + 0.0495\mathrm{x}[\mathrm{n}-2]$$

O projecto deste filtro pode ser feito em Matlab como segue

```
>> wc=2*tan(pi/12)
wc =
   0.5359
>> [B,A]=butter(2,wc,'s')
B =
        0         0    0.2872
A =
   1.0000    0.7579    0.2872
>> [Numd, Dend]=bilinear(B,A,1)
Numd =
   0.0495    0.0990    0.0495
Dend =
   1.0000   -1.2796    0.4776
>>freqz(Numd,Dend)
```



b) No caso de uma imagem a sensibilidade à fase é muito significativa pelo que temos que ir para o caso dos filtros FIR devido à característica de fase linear. Temos que optar entre o método das janelas e o método óptimo (algoritmo de McClellan). O método óptimo tem 2 vantagens importantes:

1) Permite “ripple” diferente na banda de transição e na banda de rejeição

2) Minimiza a capacidade de cálculo necessária à obtenção do sinal de saída do filtro, pois para o mesmo desempenho a largura do filtro é a mínima possível, ou seja a convolução do sinal de entrada com a resposta impulsional do filtro tem o menor número possível de somas e multiplicações.

O filtro óptimo centrado na origem será do tipo

$$A_e(\Omega) = \sum_{k=0}^{L} a_k (\cos\Omega)^k$$

Sendo um filtro de 2ª ordem temos L=2 (polinómio do 2º grau). A função a optimizar será

$$W(\Omega_i)[H_d(\Omega_i) - A_e(\Omega_i)] = (-1)^{i+1}\delta;\ \ i = 1,2,3,4. \qquad (L+2=4) \tag{1}$$

A função de erro pode permitir o mesmo ripple em ambas as bandas ou seja

$$W_p(\cos\Omega) = \begin{cases} \dfrac{1}{K}; & \cos\Omega_p \le \cos\Omega \le 1 \\ 1; & -1 \le \cos\Omega \le \cos\Omega_s \end{cases}$$

Com $K=\delta_1/\delta_2$. Para facilidade de cálculos vamos supôr $\delta_1=\delta_2$, ou seja K=1.

Vamos agora dividir o eixo Ω num número de pontos que garanta pelo menos L+2=4 alternâncias do erro, sabendo que as frequências de topo de banda (Ωp e Ωs) são pontos de alternância do erro. Como se pretende uma freq. de corte (Ωp= π/6) muito menor que π/2 espera-se mais alternâncias do erro acima de Ωs que abaixo de Ωp. O mínimo valor de alternâncias abaixo de Ωp ocorre se em Ωp existir a segunda alternância, ou seja se A(Ω) for uma função monotónica decrescente desde Ω=0 até Ω=Ωp. Mesmo assim em Ωs existirá a 3ª alternância, haverá mais uma a inverter a função e já só poderá existir mais uma que deverá aparecer antes de π pois o espaço em Ω é muito maior para a direita de Ωp que para a esquerda. Claro que neste caso Ωs será grande pois forçámos uma ordem pequena (2) para o filtro ou seja esperamos uma largura de banda de transição significativa. Podemos então considerar os Ωi tal que as condições acima sejam mais ou menos satisfeitas o que leva a menos iterações noi algoritmo até se atingirem os valores óptimos. Consideremos então por exemplo (note que esperamos uma largura de banda de transição grande pelo que suspeitamos que possa ser por exemplo o dobro da banda passante (Ω3) e como estamos em π/2 ainda nos resta mais π/2 e só podemos ter mais uma alternância do erro então ela estará antes de π, daí o valor de (Ω4).

$$\begin{cases} \Omega_1 = 0 \\ \Omega_2 = \Omega_p = \dfrac{\pi}{6} \\ \Omega_3 = \dfrac{\pi}{2} \\ \Omega_4 = \dfrac{2\pi}{3} \end{cases} \quad ==\!\!\Rightarrow \quad \begin{cases} H_d(\Omega_1) = 1 \\ H_d(\Omega_2) = 1 \\ H_d(\Omega_3) = 0 \\ H_d(\Omega_4) = 0 \end{cases}$$

H_d (Ω_i) são os valores desejados da resposta em frequência do filtro. Vamos então escrever a equação do sistema de acordo com a equação (1)

$$\begin{cases} H_d(\Omega_1) = \dfrac{1}{W(\Omega_1)}\delta + a_0 + a_1 \cos(\Omega_1) + a_2(\cos(\Omega_1))^2 \\ H_d(\Omega_2) = \dfrac{1}{W(\Omega_2)}\delta + a_0 + a_1 \cos(\Omega_2) + a_2(\cos(\Omega_2))^2 \\ H_d(\Omega_3) = \dfrac{1}{W(\Omega_3)}\delta + a_0 + a_1 \cos(\Omega_3) + a_2(\cos(\Omega_3))^2 \\ H_d(\Omega_4) = \dfrac{1}{W(\Omega_4)}\delta + a_0 + a_1 \cos(\Omega_4) + a_2(\cos(\Omega_4))^2 \end{cases}$$

O sistema acima descrito é constituído por 4 equações e contém 4 incógnitas (δ e os a_k's) pelo que tem uma solução única. A solução pode ser obtida por minimização iterativa do erro. No entanto a interpolação polinomial é mais eficiente. Começa-se por calcular δ através de

$$\delta = \frac{\sum_{k=1}^{L+2} b_k H_d(\Omega_k)}{\sum_{k=1}^{L+2} \dfrac{b_k(-1)^{k+1}}{W(\Omega_k)}} \qquad\qquad b_k = \prod_{\substack{i=1 \\ i\neq k}}^{L+2} \frac{1}{x_k - x_i}$$

Onde x_i=cosΩ_i.

$$\begin{cases} x_1 = \cos\Omega_1 = 1 \\ x_2 = \cos\Omega_2 = 0.866 \\ x_3 = \cos\Omega_3 = 0 \\ x_4 = \cos\Omega_4 = -0.5 \end{cases} \qquad \begin{cases} b_1 = \dfrac{1}{1-0.866}\dfrac{1}{1-0}\dfrac{1}{1-(-0.5)} = 4.9751 \\ b_2 = \dfrac{1}{0.866-1}\dfrac{1}{0.866-0}\dfrac{1}{0.866-(-0.5)} = -6.3085 \\ b_3 = \ldots\ldots = 2.3095 \\ b_4 = \ldots\ldots = -0.9761 \end{cases}$$

Pelo que o erro óptimo será

$$\delta = \frac{\sum_{k=1}^{4} b_k H_d(\Omega_k)}{\sum_{k=1}^{4} \dfrac{b_k(-1)^{k+1}}{W(\Omega_k)}} = \frac{b_1 + b_2 + 0 + 0}{Kb_1 - Kb_2 + b_3 - b_4} = -0.0915$$

O passo seguinte é calcular o polinómio óptimo dado pela seguinte equação

$$A_e(\Omega) = P(\cos\Omega) = \frac{\sum_{k=1}^{L+1} d_k (x - x_k)^{-1} C_k}{\sum_{k=1}^{L+1} d_k (x - x_k)^{-1}}$$

$$C_k = H_d(\Omega_k) - \frac{(-1)^{k+1}\delta}{W(\Omega_k)}$$

$$d_k = \prod_{\substack{i=1 \\ i \neq k}}^{L+1} \frac{1}{x_k - x_i} = b_k (x_k - x_{L+2})$$

$$\begin{cases} C_1 = 1 - \delta = 1.0915 \\ C_2 = 1 + \delta = 0.9085 \\ C_3 = -\delta = 0.0915 \\ C_4 = \delta = -0.0915 \end{cases}$$

$$\begin{cases} d_1 = b_1(x_1 - x_4) = 4.9751 \ .(1.5) = 7.4627 \\ d_2 = b_2(x_2 - x_4) = -6.3085 \ .(0.866 + 0.5) = -8.6174 \\ d_3 = ... = 1.1547 \end{cases}$$

Dispomos agora de todos os dados para calcular o polinómio óptimo que aproxima o filtro passa baixo ideal por um sistema de 2ª ordem.

$$P(\cos\Omega) = \frac{\sum_{k=1}^{3} d_k (x - x_k)^{-1} C_k}{\sum_{k=1}^{3} d_k (x - x_k)^{-1}} = \frac{\left[7.4627/(x-1)\right]1.0915 + \left[-8.6174/(x-0.866)\right]0.9085 + \left[1.1547/(x)\right]0.0915}{\frac{7.4627}{x-1} - \frac{8.6174}{x-0.866} + \frac{1.1547}{x}}$$

Reduzindo as expressões do numerador e denominador da equação anterior ao mesmo deniminador obtém-se

$$P(x = \cos\Omega) = \frac{0.4223\ x^2 + 0.5777\ x + 0.1057}{1}$$

Ou seja

$$\begin{cases} a_0 = 0.1057 \\ a_1 = 0.5777 \\ a_2 = 0.4223 \end{cases}$$

A resposta impulsional do filtro não causal pode agora ser obtida de

$$A_e(\Omega) = \sum_{n=-L}^{L} h_e[n] e^{-j\Omega n} = h_e[0] + \sum_{n=1}^{L} 2h_e[n]\cos(\Omega n) = \sum_{k=0}^{L} a_k (\cos\Omega)^k$$

Pelo que

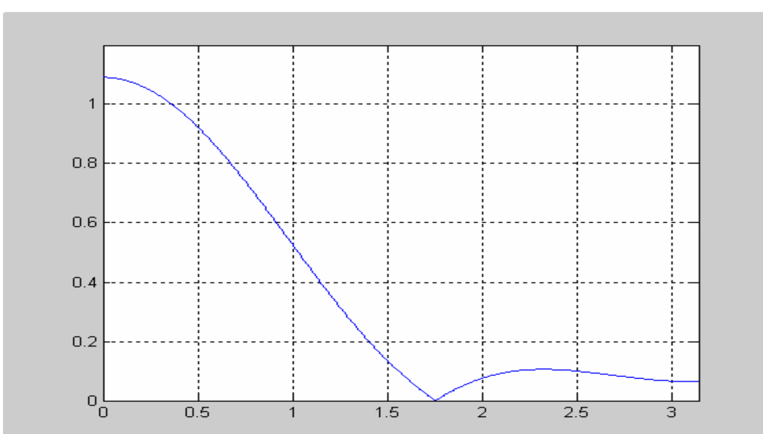
$$\begin{cases} h_e[0] = a_0 = 0.1057 \\ h_e[1] = \dfrac{a_0}{2} = 0.2889 \\ h_e[2] = \ldots = 0.3029 \end{cases}$$

A resposta impulsional do filtro final será a resposta calculada anteriormente simétrica relativamente à origem e deslocada à direita de M/2=2.

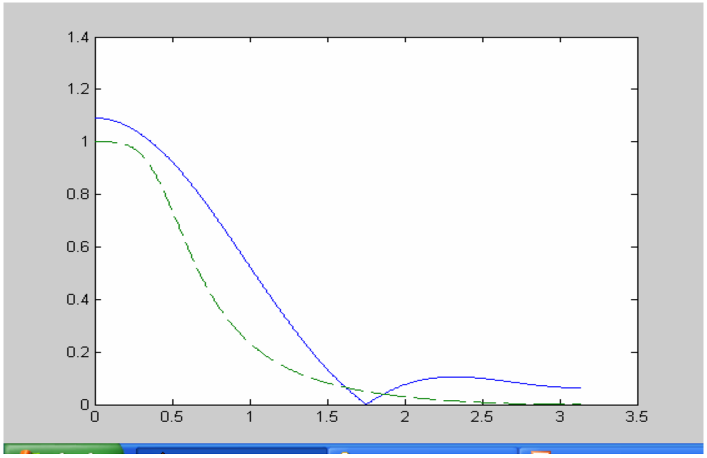
$$\begin{cases} h[0] = 0.1057 \\ h[1] = 0.2889 \\ h[2] = 0.3029 \\ h[3] = 0.2889 \\ h[4] = 0.1057 \end{cases}$$

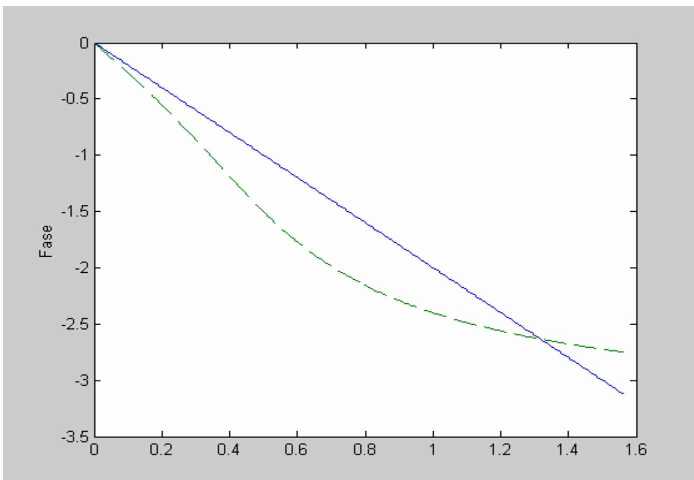
Este problema pode ser resolvido em Matlab com apenas 5 instruções:

```
>> wei=[1 1];
>> f=[0 1/6 0.5 2/3];
>> m=[1 1 0 0];
>> h=remez(4,f,m,wei)
h =
   0.1057    0.2887    0.3028    0.2887    0.1057
>> [H,W]=freqz(h,[1],1000,'whole');
>> mag=abs(H(1:1:501));
>> delta_w=2*pi/1000;
>> plot(0:delta_w:delta_w*500,mag)
>> axis([0 pi 0 1.2])
>> grid
```



Podemos a título de exemplo comparar as características dos filtros desenvolvidos





Ignorando a capacidade de cálculo necessária à implementação dos filtos a vantagem óbvia do filtro FIR é que apresenta fase linear enquanto o filtro IIR distorce em termos de fase.