Autumn Semester 2011-12 (2.0 hours)

EEE6440 Advanced Signal Processing

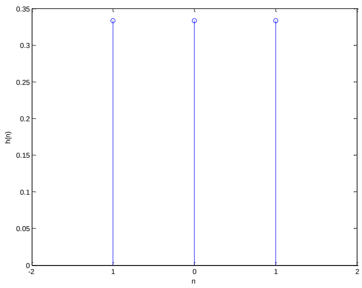
Solutions for Part A:

1.

a. y(n) = 1/3(x[n+1]+x[n]+x[n+1]]

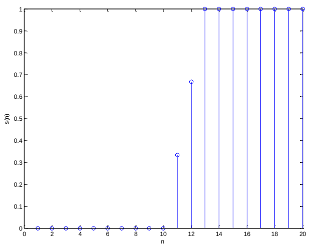
h(n) = { 1/3, 1/3, 1/3} the second element represents n=0.

Or a graphical solution.



(1)

b.



Convolve the h(n) with step function u(n). In other words taking the discrete integral of h(n). Results in { ... 0, 1/3, 2/3, 1, ...}

Time-domain performance:

- Fast response, No overshoot, linear phase
- Filter kernel is short. Therefore,
  - Average performance on smoothing a signal corrupted
  - can retain the edge information in the signal

(2)

c. y(n) = 1/3(x[n+1] +x[n] +x[n+1])

h(n) = 1/3 1/3 1/3

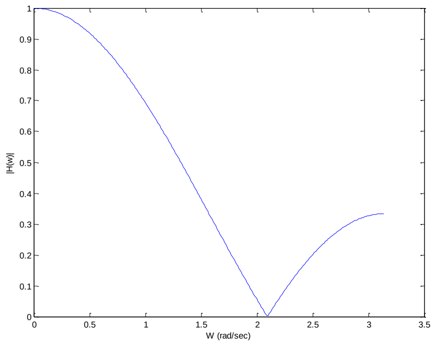
$h(z) = 1/3 (z + 1 + z^{-1})$

$z= e^{j\omega}$,

$H(j\omega) = 1/3 (e^{j\omega} + 1 + e^{-j\omega}) = (1+2 \cos \omega)/3$

$|H(j\omega)| = | (1+2 \cos \omega)/3|$

(3)



d.

$$y(i) = \frac{1}{M}\sum_{k=\frac{1-M}{2}}^{\frac{M-1}{2}} x(i+k)$$

$$y(i+1) = \frac{1}{M}\sum_{k=\frac{1-M}{2}}^{\frac{M-1}{2}} x(i+1+k)$$

$$y(i+1) = \frac{1}{M}\left(\left(\sum_{k=\frac{1-M}{2}}^{\frac{M-1}{2}-1} x(i+1+k)\right) + x(i+1+\frac{M-1}{2})\right)$$

$$y(i+1) = \frac{1}{M}\left(-x(i+\frac{1-M}{2}) + \left(\sum_{k=\frac{1-M}{2}}^{\frac{M-1}{2}} x(i+k)\right) + x(i+1+\frac{M-1}{2})\right)$$

$$y(i+1) = \frac{1}{M}\left(-x(i+\frac{1-M}{2}) + My(i) + x(i+1+\frac{M-1}{2})\right)$$

That means, for any M, y[i] is computed as

y[i] = y[i-1]+(x[i+p]-x[i-q]) (division by M can be done at the end)

where p=(M-1)/2 and q=p+1

Number of multiplications: M (to compute y[0]) -----------------(A)

Number of additions: (M-1) + 2L (L is the signal length) -------------(B)

For non-recursive implementation

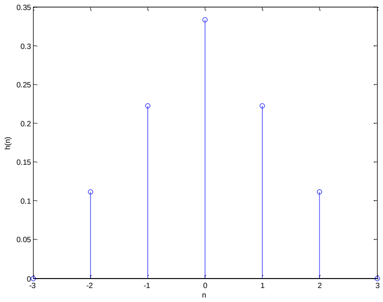
Number of multiplications: ML ----------------(C)

Number of additions: (M-1)L --------------------------(D)

(A) and (B) are much smaller than (C) and (D). **(4)**

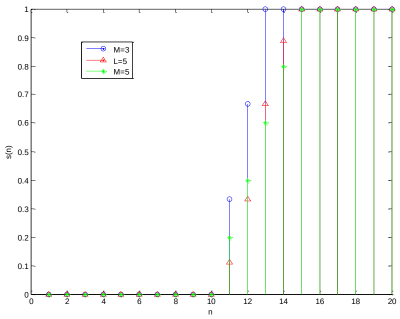
e. Convolve h(n)={1/3, 1/3, 1/3} with itself.

{1/3, 1/3, 1/3}*{1/3, 1/3, 1/3} = {1/9 2/9 1/3 2/9 1/9}



L=5

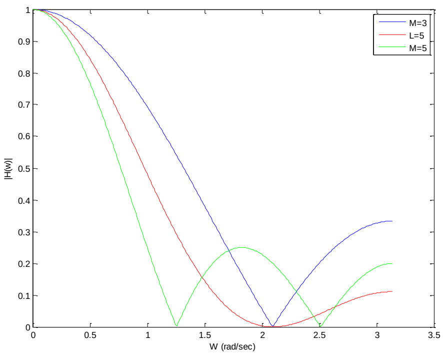
Now compare M=3, M=5 and L=5



For M=3, $|H(j\omega)| = |(1+2\cos\omega)/3|$

For M=5, $|H(j\omega)| = |(1+2\cos(\omega) + 2\cos(2\omega))/5|$

For L=5, $|H(j\omega)| = |(3+4\cos(\omega) + 2\cos(2\omega))/9|$



(Only an estimated sketch is required to explain the performance difference.)

L=5 provides a faster time-domain response compared to M=5 with better stop-band attenuation and compared to both M=3 and M=5. It has better smoothing performance (compared to M=3 and M=5) and ability to retain edges compared to M=5.

(5)

2.

a. In T each row corresponds to a basis vector f_v.

$f_0 = [h \quad h \quad 0 \quad 0] \qquad f_1 = [0 \quad 0 \quad h \quad h]$

$f_2 = [h \quad -h \quad 0 \quad 0] \quad f3 = [0 \quad 0 \quad h \quad h]$ **(2)**

b. For the orthogonality condition

If the inner product $\langle f_n, f_m \rangle$ = 1 when n = m and

= 0 when n $\neq$ m.

In other words $\sum_{i=0}^{3} f_{ni} f_{nm} = \delta_{nm}$

<f0,f0> =<f1,f1> =<f2,2> =<f3,f3> = ((hxh)+(hxh)+(0)+(0)) = $2h^2$=1

h = +/- 1/sqrt(2)

<f1,f2> = 0

<f1,f3> = ((hxh)+(-hxh)+(0)+(0)) = 0

<f1,f4> = 0

Similarly, <f2,f2>=<f3,f3>=<f4,f4>=1 and

<f2,f3>=<f2,f4>=<f4,f3>=0

(3)

c. low pass [1 1]/sqrt(2)

high pass [1 -1]/sqrt(2) **(1)**

d. F is orthogonal. Therefore, the inverse matrix is the transpose.

$$T^{-1} = \begin{bmatrix} h & 0 & h & 0 \\ h & 0 & -h & 0 \\ 0 & h & 0 & h \\ 0 & h & 0 & -h \end{bmatrix}$$

Compute the $T^{-1}T$ and show it is the Identity matrix (I)

$$TT^{-1} = \begin{bmatrix} h & h & 0 & 0 \\ 0 & 0 & h & h \\ h & -h & 0 & 0 \\ 0 & 0 & h & -h \end{bmatrix} \begin{bmatrix} h & 0 & h & 0 \\ h & 0 & -h & 0 \\ 0 & h & 0 & h \\ 0 & h & 0 & -h \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(3)

e.

$$T = \begin{bmatrix} h & h & 0 & 0 \\ h & -h & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(2)

f. $y_{0=}$(x0+x1)*h

y1=(x2+x3)*h

(x0+x1+x2+x3)=(y0+y1)/h

mean(x0+x1+x2+x3)=(y0+y1)/(4h)

OR

Perform the second level transform on y to get z=(z0,z1,z2,z3). Then mean= z0/2. **(2)**

g. Transform matrix for 2 data points

$$T = \begin{bmatrix} h & h \\ h & -h \end{bmatrix}$$

Create low pass transform matrix L (64x128) and H (64x128)

$$L = \begin{bmatrix} h & h & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h & h & & 0 & 0 \\ \vdots & \vdots & & & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & h & h \end{bmatrix}$$

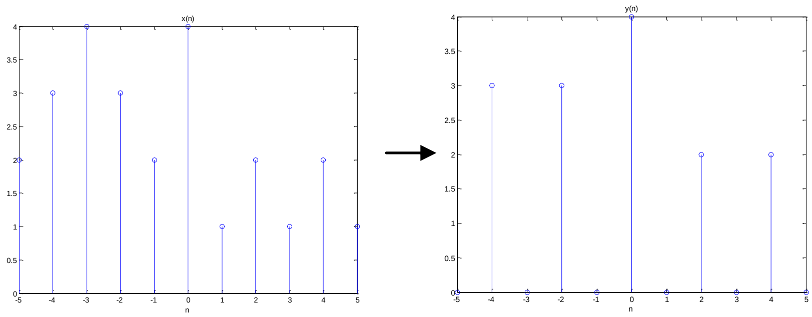
$$H = \begin{bmatrix} h & -h & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & h & -h & & 0 & 0 \\ \vdots & \vdots & & & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & h & -h \end{bmatrix}$$

$$T = \begin{bmatrix} L \\ H \end{bmatrix}$$

Apply T on the data vector [X] as a matrix multiplication TX. **(2)**

3. a. (i)

$y(n) = \{0, 3, 0, 3, 0, 4, 0, 2, 0, 2, 0\}$



(ii) This represents a situation where x(n) is decimated by a factor of 2 followed by interpolation by a factor of 2.

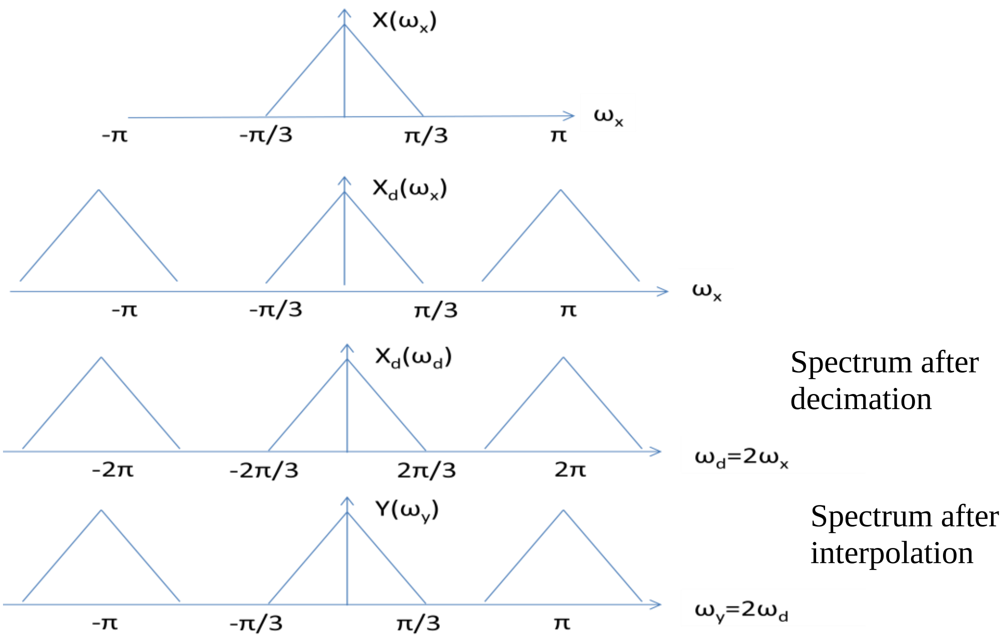
Starting from:

$$Y(j\omega_x) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(j(\omega_x - 2\pi k / M)\right)$$

For M=2,

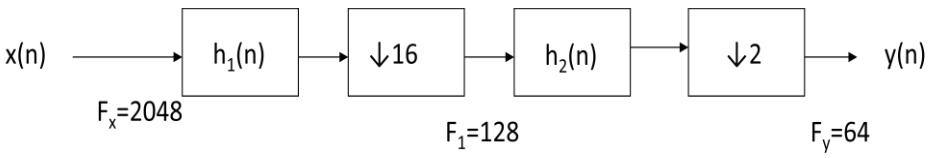
$$Y(j\omega_x) = \frac{1}{2}\sum_{k=0}^{1} X\left(j(\omega_x - \pi k)\right)$$

(5)



(iii) an anti-imaging filter required with transition band from π/3 to 2π/3.

b.



Passband deviation: 0.01dB → 0.00115

Stopband atteneuation: 80dB → 0.0001

For both filters we choose

δ_p=0.00115/2 =0.00058

δ_s=0.0001

Filter length given by $N \approx \frac{-10\log(\delta_p\,\delta_s)-13}{14.6(\Delta f)} + 1$

$$N \approx \frac{-10\log(0.00058 \text{ x } 0.0001) - 13}{14.6(\Delta f)} + 1$$

$$N \approx \frac{4.066}{14.6(\Delta f)} + 1$$

For h_2:

Passband 0 - 30 Hz

Stopband 32-64 Hz

Transition band 30Hz – 32Hz **(8)**

Normalised transition bandwidth   (32-30)/128 = 2/128

Therefore $N_2 \approx \frac{4.066}{14.6\left(\frac{2}{128}\right)} + 1 = 261$

For h_1:

Passband   0 - 30 Hz

Stopband   (128-64/2) -1024 Hz = 96-1024

Transition band   30Hz – 96Hz

Normalised transition bandwidth   (96-30)/2048 = 66/2048

Therefore $N_1 \approx \frac{4.066}{14.6\left(\frac{66}{2048}\right)} + 1 = 127$

(ii) MPS = $\sum_{i=1}^{2} F_i N_i$ =   128x127 + 261x64 = 32 960

(iii) N is inversely proportion to Δf. If a single-stage was used Δf would have been (32-30)/2048. To make this value larger, we need to make the numerator bigger and the denominator smaller. This can be achieved by factoring F into a product of several smaller sampling rates. Each of the early stage filetrs the transition bandwidth is large because the correson=ding sampling rates are closer to F.

c. Sampling rate 8kHz means 8000 samples/sec.

- Subband 5 represents ½ of the total bandwidth, while each of the other subbands represent 1/8th of the total bandwidth.
- Therefore the data rate
  ((5+4+4+2)/8 + ½) x 8000
  =19 kHz

(2)

Part B

Q4 a.

The variance $\sigma_x^2(n)$ of a random variable x(n) is the mean-square variation about the mean $m_x(n)$.

$$\sigma_x^2(n) = E[(x(n) - m_x(n))^2]$$
$$= E[x^2(n) - x(n)m_x(n) - x(n)m_x(n) + m_x^2(n)]$$
$$= E[x^2(n)] - 2E[x(n)]m_x(n) + m_x^2(n)$$
$$= E[x^2(n)] - 2m_x^2(n) + m_x^2(n) = E[x^2(n)] - m_x^2(n)$$

Q4 b.

i) $H_1(z)=1-3z^{-1}$
z-transform of the autocorrelation at the output
$$S_{y_1y_1}(z) = H_1(z)\ H_1(z^{-1})\ \sigma_x^2$$
$=(1-3z^{-1})(1-3z)=1-3z^{-1}-3z+9=-3z+10-3z^{-1}$
Inverse z-transform by inspection to give autocorrelation sequence:

$$\phi_{y_1y_1}(m) = Z^{-1}[\ S_{y_1y_1}(z)\ ]$$

Autocorrelation sequence: -3 for m=-1, 10 for m=0, -3 for m=1 and zero for other values of m

ii)
$H_1(z)=1-3z^{-1}$
$H_2(z)=1-2z^{-2}$
Cross-correlation sequence $\phi_{y_1y_2}(m) = E[\ y_1(n)\ y_2(n+m)\ ]$.
z-transform of the cross-correlation at the outputs
$$S_{y_1y_2}(z) = H_1(z^{-1})\ H_2(z)\ \sigma_x^2$$
$=(1-3z)(1-2z^{-2})=1-2z^{-2}-3z+6z^{-1}$
$=-3z+1+6z^{-1}-2z^{-2}$

Inverse z-transform yields: $\phi_{y_1y_2}$
-3 for m=-1, 1 for m=0, 6 for m=1, -2 for m=2 and zero for other values of m
The second cross-correlation is most easily obtained by using the property that $\phi_{xy}(m) = \phi_{yx}(-m)$ i.e.

$$\phi_{y_2y_1}(m) = \phi_{y_1y_2}(-m)$$

-2 for m=-2, 6 for m=-1, 1 for m=0, -3 for m=1, zeros for other values of m

Q4 c.

i)
For cosine wave input, the dynamic range R_D of the quantiser can be calculated from the equation in Section 7.5.2 since sine wave and cosine wave have the same power given the same amplitude.
Then, for a 20-bit A/D converter (M=20):

$R_D=1.76+6M$ dB$=1.76+20*6=121.76$dB,

ii)
For uniformly distributed input signal, its power is given by $P_i=(2A)^2/12$, where A is its amplitude.
The stepsize δx is given by $\delta x=2A/2^M$
The quantisation noise also has a uniform distribution, then its power is $P_n=(\delta x)^2/12$.
Then its dynamic range is given by

$R_D=10\log_{10}(P_i/P_n)=6M=120$dB

Q5 a.

There are four zeros for $S_{yy}(z)$: ½, 3, 2, 1/3. So $S_{yy}(z)$ can be formed by passing a zero-mean white signal through four possible filters:
$H_0(z)=(z-1/2)(z-3)$
$H_1(z)=(z-1/2)(z-1/3)$
$H_2(z)=(z-2)(z-3)$
$H_3(z)=(z-2)(z-1/3)$
The inverse of any of the above four filters will whiten the signal y(n). The inverse of $H_1(z)$ will be the one with minimum phase since all of its zeros are inside the unit circle.

Q5 b.

$e(n) = x(n) - \hat{x}(n)$

The mean-square error (MSE) cost function

$$\xi(n) = E[\, e^2(n)\,]$$

$$\hat{x}(n) = \sum_{i=0}^{N-1} h_i\, y(n-i)$$

$$= [\, h_0\, h_1 \cdots h_{N-1}\,] \begin{bmatrix} y(n) \\ y(n-1) \\ \cdot \\ \cdot \\ \cdot \\ y(n-N+1) \end{bmatrix}$$

$$= \mathbf{h}^T\, \mathbf{y}(n) = \mathbf{y}^T(n)\, \mathbf{h}$$

Differentiate

$$\frac{\partial \xi}{\partial h_j} = \frac{\partial}{\partial h_j} E[\, \{ e^2(n) \}\,]$$

$$= E[\, \frac{\partial}{\partial h_j} \{ e^2(n) \}\,]$$

$$= E[\, 2\, e(n) \frac{\partial e(n)}{\partial h_j}\,]$$

$$= E[\, 2\, e(n) \frac{\partial}{\partial h_j} \{ x(n) - \mathbf{h}^T\, \mathbf{y}(n) \}\,]$$

$$= E[\, 2\, e(n) \frac{\partial}{\partial h_j} \{ -h_j\, y(n-j) \}\,]$$

$$= E[\, 2\, e(n)\, y(n-j) \}\,]$$

$$= 0$$

for j=0, 1, …, N-1.

In vector form, the gradient is given by

$$\underline{\nabla} = -2\ E[\ \mathbf{y}(n)\ e(n)\ ]$$

$$= -2\ E[\ \mathbf{y}(n)\ (\ x(n) - \mathbf{y}^T(n)\ \mathbf{h}\ )\ ]$$

$$= -2\ E[\ \mathbf{y}(n)\ x(n)\ ] + 2\ E[\ \mathbf{y}(n)\mathbf{y}^T(n)\ ]\ \mathbf{h}$$

$$= -2\ \Phi_{yx} + 2\ \Phi_{yy}\ \mathbf{h}$$

$$= \underline{0}$$

where

Autocorrelation matrix

$$\Phi_{yy} = E[\ \mathbf{y}(n)\ \mathbf{y}^T(n)\ ]$$

Cross-correlation vector

$$\Phi_{yx} = E[\ \mathbf{y}(n)\ x(n)\ ]$$

Optimal Solution

$$\Phi_{yy}\ \mathbf{h}_{opt} = \Phi_{yx}$$

Alternative formulation

$$\mathbf{h}_{opt} = \Phi_{yy}^{-1}\ \Phi_{yx}$$

Q6 a.

i)
Suppose the z-transform of the filter is given by H(z), then the relationship is given by

$$S_{xy}(z) = H(z)\ S_{xx}(z)$$

ii)
For a white input, we have

$$S_{xy}(z) = H(z)\ \sigma_x^2$$

where σ_x^2 is variance of the input.
Taking inverse transforms gives:

$$\phi_{xy}(m) = h_m\ \sigma_x^2$$

where h_m is the impulse response of the filter. It can be measured by estimating the cross-correlation directly from the data with the following three steps:

$$\hat{\phi}_{xy}(m) = \frac{1}{M}\sum_{n=0}^{M-1} x(n)\, y(n+m)$$

$$\hat{\sigma}_x^2 = \frac{1}{M}\sum_{n=0}^{M-1} x^2(n)$$

$$\hat{h}_m = \frac{\hat{\phi}_{xy}(m)}{\hat{\sigma}_x^2}$$

Q6 b.

i)

A Time Recursion

$$\mathbf{h}(n) = \mathbf{h}(n-1) - \mu\, \hat{\underline{\nabla}}(n-1)\,.$$

The Exact Gradient

$$\underline{\nabla}(n) = -\,2\,\mathrm{E}[\,\mathbf{y}(k)\,(\,x(k) - \mathbf{h}^T(n)\,\mathbf{y}(k)\,)\,]$$

$$= -\,2\,\mathrm{E}[\,\mathbf{y}(k)\,e(k)\,]$$

A Simple Estimate of the Gradient

$$\hat{\underline{\nabla}}(n) = -\,2\,\mathbf{y}(n+1)\,e(n+1)$$

The Error

$$e(n+1) = x(n+1) - \mathbf{h}^T(n)\,\mathbf{y}(n+1)$$

Then the updated equation of the LMS algorithm is given by

$\mathbf{h}(n)=\mathbf{h}(n-1)+2\mu\mathbf{y}(n)e(n)$

ii)

$e(4)=x(4)-\mathbf{h}^T(3)\mathbf{y}(4)=-0.27-[1\ 3][0.5\ 0.25]^T$

$=-1.52$

The impulse response is then updated by

$\mathbf{h}(4)=\mathbf{h}(3)+2\mu\mathbf{y}(4)e(4)$

$=[1\ 3]^T+0.2*(-1.52)*[0.5\ 0.25]^T$

$=[0.848\quad 2.924]^T$