ELEC3004 Problem Set 3: Filters (Digital and Analogue)

Kerri Wait, ELEC3004 Tutor, BEng(Hons I) MIEEE

07/05/2015

Revision 0.41

Contents

1	Que	estion 1: Dancing around Poles and Filters 3						
	1.1	$H_a(z)$						
		1.1.1 Poles and Zeroes						
		1.1.2 High or Low Pass Filter						
		1.1.3 Filter order						
		1.1.4 FIR or IIR Filter $\ldots \ldots \ldots$						
	1.2	$H_b(z)$						
		1.2.1 Poles and Zeroes $\ldots \ldots 5$						
		1.2.2 High or Low Pass Filter						
		1.2.3 Filter order $\ldots \ldots \ldots$						
		1.2.4 FIR or IIR Filter $\ldots \ldots \ldots$						
	1.3	MATLAB Code						
2	Question 2: Towards Digital Filters 7							
	2.1	Proof that this system is a discrete time low-pass filter						
		2.1.1 In the z domain $\ldots \ldots \ldots$						
		2.1.2 Evaluating $H(e^{j\omega})$						
	2.2	Magnitude and Phase Response						
	2.3	Magnitude of filter at DC, and frequency at which the magnitude is equal to 1 9						
	2.4	Periodicity of response						
	2.5	MATLAB Code						
3	Question 3: A Simple FIR Filter							
0	3.1	Selection of window						
	3.2	Window order						
	0	3.2.1 Hamming window						
		3.2.2 Blackman window						
		3.2.3 Kaiser window and other niche windows						
	3.3	Is this the lowest possible order?						
1	011	astion 4. A Poor Man's FIR Jackot 13						
4	Que 4 1	System function $H(q)$ 13						
	4.1	$\begin{array}{c} \text{System function } \Pi(2) \\ \text{Polos and Zeroos} \end{array} $						
	4.2 13	Impulse response and Pole Zero plot						
	- T . U	$-\mathrm{III} \rho \mathrm{ub} \mathrm{r} \mathrm{o} \mathrm{p} \mathrm{o} \mathrm{ub} \mathrm{r} \mathrm{o} \mathrm{o} \mathrm{u} \mathrm{d} \mathrm{r} \mathrm{o} \mathrm{o} \mathrm{d} \mathrm{o} \mathrm{o} \mathrm{p} \mathrm{o} \mathrm{o} \mathrm{o} \mathrm{o} \mathrm{o} \mathrm{o} \mathrm{o} o$						
		4.3.1 Notes on the Impulse response and Pole zero plot 14						
		4.3.1 Notes on the Impulse response and Pole zero plot $\dots \dots \dots$						
	4.4	4.3.1Notes on the Impulse response and Pole zero plot144.3.2Results for $M = 1:10$ 14Magnitude and Phase response25						

5	Que	estion 5: More FIRmly Filtered	28
	5.1	MATLAB and Normalised Frequencies	28
	5.2	MATLAB Commands	28
	5.3	Pole Zero plots	29
	5.4	Magnitude and Phase Responses	31
		5.4.1 Linear magnitude response	31
		5.4.2 Magnitude (dB) response	32
		5.4.3 Phase response	32
	5.5	Group Delay	33
	5.6	Numerical Stability	34

1 Question 1: Dancing around Poles and Filters

Before attempting this question, it is useful to remind ourselves that the transfer function of a general LTI system H(z) can be represented as:

$$H(z) = \frac{\beta_o + \beta_1 z^{-1} + \beta_2 z^{-2} + \ldots + \beta_m z^{-m}}{1 + \alpha_1 z^{-1} + \alpha_2 z^{-2} + \ldots + \alpha_n z^{-n}}$$
(1)

It may be useful to represent H(z) using positive powers of z:

$$H(z) = \frac{z^{-m}}{z^{-n}} \frac{\beta_o z^m + \beta_1 z^{m-1} + \beta_2 z^{m-2} + \ldots + \beta_m}{z^n + \alpha_1 z^{n-1} + \alpha_2 z^{n-2} + \ldots + \alpha_n}$$
(2)

Now let us consider each of the transfer functions given below:

$$H_a(z) = \frac{(1-a)(1+z^{-1})}{2(1-az^{-1})}$$
(3)

$$H_b(z) = \frac{(1+b)(1-z^{-1})}{2(1-bz^{-1})} \tag{4}$$

1.1 $H_a(z)$

Let us begin by representing $H_a(z)$ using positive powers of z.

$$H_a(z) = \frac{(1-a)(1+z^{-1})}{2(1-az^{-1})} = \frac{(1-a)+(1-a)z^{-1}}{2-2az^{-1}}$$
$$= \frac{z^{-1}}{z^{-1}}\frac{(1-a)z+(1-a)}{2z-2a} = \frac{(1-a)z+(1-a)}{2z-2a}$$

1.1.1 Poles and Zeroes

The zeroes of a system can be found by solving the equation in the numerator when it equals zero.

numerator =
$$(1 - a)z + (1 - a) = 0$$

 $(1 - a)z = -(1 - a)$
 $z = \frac{-(1 - a)}{(1 - a)} = -1$

The poles of a system can be found by solving the equation in the denominator when it equals zero.

$$denominator = 2z - 2a = 0$$
$$2z = 2a$$
$$z = a$$

Therefore, there is a zero at -1 and a pole at a.

 $H_a(z)$ is represented on a pole zero map in Figure 1. The value of a serves to set the pole location; it could vary from -1 to 1 without causing system instability.



Figure 1: Pole zero plot for $H_a(z)$, a = 0.5.

Figure 2: Frequency response of $H_a(z)$, a = 0.5

1.1.2 High or Low Pass Filter

 $H_a(z)$ is the transfer function for a low pass filter (LPF). This can be determined in one of two ways:

1. Examining the pole and zero locations. As the system zero is at -1, high frequency content will be attenuated. The location of the pole, set by the value of a, determines the cutoff frequency of the filter.

2. Examining the frequency response. The frequency response of $H_a(z)$ is shown in Figure 2. This is clearly a low pass filter (LPF), albeit not a very precise one.

1.1.3 Filter order

This filter is a 1st order filter, as z^{-1} is the highest order component. In a general case, the filter order is dictated by the largest value of m or n from Equation 1.

1.1.4 FIR or IIR Filter

Infinite impulse response (IIR) filters generally take the form:

$$H(z) = \frac{\sum_{i=0}^{M} \beta_i z^{-i}}{1 + \sum_{i=1}^{N} \alpha_i z^{-i}}$$

Whilst finite impulse response (FIR) filters generally take the form:

$$H(z) = \sum_{i=0}^{M} \beta_i z^{-i}$$

 $H_a(z)$ is an IIR filter, as there are coefficients in the denominator.

1.2 $H_b(z)$

Let us begin by representing $H_b(z)$ using positive powers of z.

$$H_a(z) = \frac{(1+b)(1-z^{-1})}{2(1-bz^{-1})} = \frac{(1+b)-(1+b)z^{-1}}{2-2bz^{-1}}$$
$$= \frac{z^{-1}}{z^{-1}}\frac{(1+b)z-(1+b)}{2z-2b} = \frac{(1+b)z-(1+b)}{2z-2b}$$

1.2.1 Poles and Zeroes

The zeroes of a system can be found by solving the equation in the numerator when it equals zero.

numerator =
$$(1 + b)z - (1 + b) = 0$$

 $(1 + b)z = (1 + b)$
 $z = \frac{(1 + b)}{(1 + b)} = 1$

The poles of a system can be found by solving the equation in the denominator when it equals zero.

$$denominator = 2z - 2b = 0$$
$$2z = 2b$$
$$z = b$$

Therefore, there is a zero at 1 and a pole at b.

 $H_b(z)$ is represented on a pole zero map in Figure 3. The value of b serves to set the pole location; it could vary from -1 to 1 without causing system instability.



Figure 3: Pole zero plot for $H_b(z)$, b = 0.5.

Figure 4: Frequency response of $H_b(z)$, b = 0.5

1.2.2 High or Low Pass Filter

 $H_b(z)$ is the transfer function for a high pass filter (HPF). This can be determined in one of two ways:

1. Examining the pole and zero locations. As the system zero is at 1, low frequency content will be attenuated. The location of the pole, set by the value of b, determines the cutoff frequency of the filter.

2. Examining the frequency response. The frequency response of $H_b(z)$ is shown in Figure 4. This is clearly a high pass filter (HPF), albeit not a very precise one.

1.2.3 Filter order

This filter is a 1st order filter, as z^{-1} is the highest order component. In a general case, the filter order is dictated by the largest value of m or n from Equation 1.

1.2.4 FIR or IIR Filter

Infinite impulse response (IIR) filters generally take the form:

$$H(z) = \frac{\sum_{i=0}^{M} \beta_i z^{-i}}{1 + \sum_{j=1}^{N} \alpha_i z^{-i}}$$

Whilst finite impulse response (FIR) filters generally take the form:

$$H(z) = \sum_{i=0}^{M} \beta_i z^{-i}$$

 $H_b(z)$ is an IIR filter, as there are coefficients in the denominator.

1.3 MATLAB Code

```
Listing 1: MATLAB Code for Question 1
```

```
1
   clear all; close all; clc
 2
 3
   % Ha(z)
 4
   a = 0.5;
                                % Define a
 5
   beta = [(1-a) (1-a)];
                                % Define filter coefficients
 6
   alpha = [2 - 2*a];
 7
8
   figure()
                                % Plot poles and zeros, as well as magnitude response
9
   zplane(beta,alpha)
10 | hline = findobj(gcf, 'type', 'line');
   set(hline, 'LineWidth', 1.5)
11
12
   title(['Pole zero map of H_a(z) for a = ', num2str(a)])
13
14
   figure()
   freqz(beta,alpha)
   title(['Frequency response of H_a(z) for a = ', num2str(a)])
16
17
18
   % Hb(z)
19
   b = 0.5;
                                % Define b
20
    beta = [(1+b) - (1+b)];
                                % Define filter coefficients
21
   alpha = [2 - 2*b];
22
23
   figure()
                                % Plot poles and zeros, as well as magnitude response
24
   zplane(beta,alpha)
25
   title(['Pole zero map of H_a(z) for b = ', num2str(b)])
26
27
   figure()
28
   freqz(beta,alpha)
   title(['Frequency response of H_b(z) for b = ', num2str(b)])
29
```

2 Question 2: Towards Digital Filters

The system under consideration in this question is the first-order FIR filter $H(z) = 1 + z^{-1}$.

2.1 Proof that this system is a discrete time low-pass filter

Note: The focus of this question is on proving that it is a low-pass filter, rather than proving that it is a discrete time low-pass filter.

Here we will examine two methods that can be used to demonstrate that H(z) is a low-pass filter.

2.1.1 In the z domain

Let us first represent H(z) using positive powers of z.

$$H(z) = \frac{1+z^{-1}}{1} = \frac{z^{-1}}{z^0} \frac{z+1}{1} = \frac{z+1}{z}$$

Looking at the location of the poles and zeros of this filter is one way of determining the type of filter. The zeroes of a system can be found by solving the equation in the numerator when it equals zero.

$$numerator = z + 1 = 0$$
$$z = -1$$

The poles of a system can be found by solving the equation in the denominator when it equals zero.

$$denominator = z = 0$$

Therefore, there is a zero at -1 and a pole at 0. As the system zero is at -1, high frequency content will be attenuated; thus this is a low pass filter. The pole zero map for this system is shown in Figure 5.

2.1.2 Evaluating $H(e^{j\omega})$

To evaluate H(z) as $H(\omega)$, we substitute $z = e^{-j\omega}$.

$$H(\omega) = H(z)|_{z=e^{-j\omega}}$$
$$= 1 + e^{-j\omega}$$

Evaluating $|H(\omega)|$ at $\omega = 0, \pi/2, \pi$ will allow us to approximate the magnitude response and determine which type of filter this is.

$$\begin{aligned} H(0) &= 1 + e^{-j0} & H(\pi/2) = 1 + e^{-j\pi/2} & H(\pi) = 1 + e^{-j\pi} \\ &= 1 + 1 = 2 & = 1 - 1i = \sqrt{2} = 1.4142 & = 1 - 1 = 0 \end{aligned}$$

As the magnitude response decreases as ω increases, it can be said that this is a low-pass filter.

2.2 Magnitude and Phase Response

To evaluate H(z) as $H(\omega)$, we substitute $z = e^{-j\omega}$.

$$H(\omega) = H(z)|_{z=e^{-j\omega}}$$
$$= 1 + e^{-j\omega}$$

Using Euler's formula, $e^{j\theta} = \cos(\theta) + j\sin(\theta)$, it is possible to represent H(w) as a complex number. Bearing in mind that the expression of interest is $e^{-j\omega}$, cosine is an even function, and sine an odd function, the outcome is:

$$e^{-j\omega} = \cos(-\omega) + j\sin(-\omega)$$
$$= \cos(\omega) - j\sin(\omega)$$

Therefore $H(\omega)$ is:

$$H(\omega) = 1 + \cos(\omega) - j\sin(\omega)$$

As $H(\omega)$ is a complex variable, it is possible to determine the phase and magnitude response of $H(\omega)$ using the following equations:

$$\begin{aligned} |H(\omega)| &= \sqrt{\Re^2 + \Im^2} = \sqrt{Re^2 + Im^2} = \sqrt{(1 + \cos(\omega))^2 + (-\sin(\omega))^2} \\ \angle H(\omega) &= \tan^{-1}\left(\frac{\Im}{\Re}\right) = \tan^{-1}\left(\frac{Im}{Re}\right) = \tan^{-1}\left(\frac{-\sin(\omega)}{1 + \cos(\omega)}\right) \end{aligned}$$

Testing at various points allows us to explore the Magnitude and Phase response of this filter.

Magnitude

Phase

The results of the Magnitude testing indicate that the magnitude decreases as ω increases. For the Phase, the testing indicates that the phase shift becomes an increasing negative number as ω increases. This can be depicted graphically, as shown in Figure 6.



Figure 5: Pole zero plot for H(z).

Figure 6: Frequency response of H(z).

2.3 Magnitude of filter at DC, and frequency at which the magnitude is equal to 1

As calculated above, the magnitude of the filter at DC (when $\omega = 0$) is 2. To calculate the frequency at which the magnitude is equal to 1, the following equation is solved:

$$|H(\omega)| = 1 = \sqrt{\Re^2 + \Im^2} = \sqrt{Re^2 + Im^2} = \sqrt{(1 + \cos(\omega))^2 + (-\sin(\omega))^2}$$
$$1 = \sqrt{(1 + \cos^2(\omega) + 2\cos(\omega) + \sin^2(\omega))^2}$$

As $\cos^2(\omega) + \sin^2(\omega) = 1$:

$$1 = \sqrt{2\cos(\omega) + 2}$$

$$1 = 2\cos(\omega) + 2$$

$$-1 = 2\cos(\omega)$$

$$\cos(\omega) = -\frac{1}{2}$$

$$\omega = \cos^{-1}(-0.5)$$

$$\omega = 2.094395rad$$

$$= \frac{2\pi}{3}$$

Therefore, the frequency ω at which the magnitude is equal to 1 is $\omega = \frac{2\pi}{3}$.

2.4 Periodicity of response

If a signal is periodic if it satisfies the condition

$$x(t+T) = x(t)$$

for some nonzero T and $\forall t$. Let us evaluate the following expression

$$H(\omega) \stackrel{?}{=} H(\omega + 2\pi)$$

From 6-16 of http://www.eas.uccs.edu/wickert/ece2610/lecture_notes/ece2610_chap6.pdf we get:

$$H\left(e^{j(\omega+2\pi)}\right) = \sum_{k=0}^{M} b_k e^{-j\omega} \cdot \underbrace{e^{-j2\pi}}_{1} = H\left(e^{j\omega}\right)$$

Thus

$$H(\omega) = H(\omega + 2\pi)$$

The response of this filter is periodic.

2.5 MATLAB Code

Listing 2: MA	TLAB C	Code for	Question	2
---------------	--------	----------	----------	---

```
1
    clear all; close all; clc
 2
 3
   % H(z)
                           % Define filter coefficients
 4
   beta = [1 1];
 5
   alpha = 1;
 6
 7
   figure()
                            % Plot poles and zeros, as well as magnitude response
8
   zplane(beta, alpha)
9
   title('Pole zero map of H(z)')
                            % Plot linear, log magnitudes and phase
11
   handle = figure();
12
   [h, w] = freqz(beta, alpha);
13
   [phi, w2] = phasez(beta, alpha);
   set(gcf, 'PaperPositionMode', 'auto')
14
15 set(handle, 'Position', [680 558 560 420*1.5])
16
   subplot(3,1,1), plot(w/pi,abs(h))
17
   title('Frequency response of H(z)'), xlabel('Normalised frequency (\pi rad)')
   ylabel('Magnitude'), grid
18
   subplot(3,1,2), plot(w/pi,20*log10(abs(h)))
19
20
   title('Frequency response of H(z)'), xlabel('Normalised frequency (\pi rad)')
21
   ylabel('Magnitude(db)'), grid
22
   subplot(3,1,3), plot(w2/pi,phi)
   title('Frequency response of H(z)'), xlabel('Normalised frequency (\pi rad)')
23
24
   ylabel('Phase (degrees)'), grid
25
26
   figure()
                            % Plot just log magnitude and phase
27
   freqz(beta, alpha);
28
   title('Frequency response of H(z)')
```

3 Question 3: A Simple FIR Filter

This task requires the use of the windowing method to design an FIR filter with the following specifications:

- Sampling rate: 1 kHz Stopband: > 250 Hz
- Passband cutoff: 150 Hz

Let us define

$$\omega_p = \frac{150}{1000} \times 2\pi = 0.3\pi \qquad \qquad \omega_s = \frac{250}{1000} \times 2\pi = 0.5\pi$$

• Stopband attenuation: > 50 dB

3.1 Selection of window

If stopband attenuation of greater than 50 dB is required, a Hamming window or a Blackman window should be used as they have Peak Approximation errors of approximately -53 dB and -74 dB respectively. The Blackman has much greater attenuation however this comes at the expense of a larger main lobe. A Kaiser window may also be used, as could any other niche window reported in the literature.

3.2 Window order

The main lobe width of the magnitude response is approximately equal to the transition bandwidth $(\omega_s - \omega_p)$. Therefore it is possible to determine the filter length required to achieve a given transition width. The transition bandwidth for this filter is:

$$\omega_s - \omega_p = 0.5\pi - 0.3\pi = 0.2\pi$$

3.2.1 Hamming window

For the Hamming window, the main lobe width is $\frac{8\pi}{M}$ where M is the filter length. Therefore:

$$0.2\pi = \frac{8\pi}{M}$$
$$M = \frac{8\pi}{0.2\pi} = 40$$

This corresponds to a 39th order filter. This is $\therefore M = N + 1$ where N is the filter order.

3.2.2 Blackman window

For the Blackman window, the main lobe width is $\frac{12\pi}{M}$ where M is the filter length. Therefore:

$$0.2\pi = \frac{12\pi}{M}$$
$$M = \frac{12\pi}{0.2\pi} = 60$$

This corresponds to a 59th order filter. This is :: M = N + 1 where N is the filter order.

3.2.3 Kaiser window and other niche windows

If a Kaiser window is used, the length reduces to $11.2\pi/0.2\pi = 56$, therefore a 55th order filter would be required. Other niche windows may yield lower filter orders.

3.3 Is this the lowest possible order?

Note: there is a degree of ambiguity in this question; how you answer is influenced by your interpretation of the question. Thanks to Surya for this answer!

No, a windowing approach is not, in general, the lowest possible order for a given filter performance specification. Some of the reasons for this are:

- 1. The windowing process is, by definition, a truncation process. This leaves less terms to be determined and adjusted than if a non-windowed approach was used on the input signal
- 2. As noted in class lower filter orders may be found by treating the design of the filter coefficients as an optimisation problem. Unlike the window approach, these approaches treat the problem as a linear program that has to be solved, which returns the desired performance.
- 3. Another approach is to use a multi-stage filtering approach. Here, the two (or more) filters are combined to give better performance. For more information, see: http://au.mathworks.com/help/dsp/examples/designing-low-pass-fir-filters.html#zmw57dd0e2491

4 Question 4: A Poor Man's FIR Jacket

The system under consideration in this question is given by:

$$y[n] = \frac{1}{M+1} \sum_{k=0}^{M} x[n-k]$$

4.1 System function H(z)

It is useful to remind ourselves that $H(z) = \frac{Y(z)}{X(z)}$. With this in mind, solving for H(z) can be done as follows. Express y[n] as Y(z):

$$y[n] = \frac{1}{M+1} \sum_{k=0}^{M} x[n-k]$$
$$Y(Z) = \frac{1}{M+1} \sum_{k=0}^{M} X(z) z^{-k} = X(z) \frac{1}{M+1} \sum_{k=0}^{M} z^{-k}$$

H(z) then becomes:

$$H(Z) = \frac{Y(z)}{X(z)} = \frac{1}{M+1} \sum_{k=0}^{M} z^{-k}$$
$$= \frac{1}{M+1} \left(1 + z^{-1} + z^{-2} + \dots + z^{-M} \right)$$

This is a geometric series, and the sum of the first n terms of a geometric series is expressed as:

$$(c + cx^{-1} + cx^{-2} + \ldots + cx^{-n}) = \sum_{k=0}^{n} cx^{-k} = c\frac{1 - x^{n+1}}{1 - x}$$

Applying this to H(z) yields:

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

4.2 Poles and Zeroes

The zeroes of a system can be found by solving the equation in the numerator when it equals zero.

numerator =
$$1 - z^{-(M+1)} = 0$$

 $z^{-(M+1)} = 1$

To gain a more insightful understanding of the zeroes of a moving average filter, let us solve for $|H(\omega)| = 0$. To simplify the problem, substitute N = M + 1.

$$\begin{split} H(z) &= \frac{1}{M+1} \frac{1-z^{-(M+1)}}{1-z^{-1}} = \frac{1}{N} \frac{1-z^{-N}}{1-z^{-1}} \\ H(\omega) &= \frac{1}{N} \frac{1-e^{-j\omega N}}{1-e^{-j\omega}} \\ &= \frac{1}{N} \frac{e^{-j\omega N/2} \left(e^{j\omega N/2} - e^{-j\omega N/2}\right)}{e^{-j\omega/2} \left(e^{j\omega/2} - e^{-j\omega/2}\right)} \\ &= \frac{1}{N} e^{-j\omega (N-1)/2} \frac{2jsin(\frac{\omega N}{2})}{2jsin(\frac{\omega}{2})} \\ &= \frac{1}{N} \frac{sin(\frac{\omega N}{2})}{sin(\frac{\omega}{2})} e^{-j\omega (N-1)/2} \\ |H(\omega)| &= \frac{1}{N} \frac{|sin\frac{\omega N}{2}|}{|sin\frac{\omega}{2}|} \end{split}$$

To solve for $|H(\omega)| = 0$, solve for $\sin \frac{\omega N}{2} = 0$.

$$sin\frac{\omega N}{2}=0$$

The sinusoid is 0 at π , 2π , 3π , $4\pi = k\pi$, where k = 1, 2, 3, 4, ..., N. Therefore, ω values corresponding to zeroes can be expressed as:

$$\omega = \frac{2\pi k}{N} = \frac{2\pi k}{M+1}$$

There are M zeroes, located at $z = e^{j^{2\pi k}/(M+1)}$.

The poles of a system can be found by solving the equation in the denominator when it equals zero.

denominator =
$$1 - z^{-1} = 0$$

 $z^{-1} = 1$
 $z = 1$

Therefore, there are M+1 zeroes at $z = e^{j^{2\pi k}/(M+1)}$ and a pole at 1. This pole is cancelled out by the first zero, which is also located at 1.

4.3 Impulse response and Pole Zero plot

The impulse responses and pole zero plots for M = 1:10 are depicted in Figures 7 - 26

4.3.1 Notes on the Impulse response and Pole zero plot

When marking this, please pay attention to both the number of elements in the impulse response, and their amplitude. The $\frac{1}{M+1}$ term must have been applied to the coefficients in order to perform the average, otherwise this is a summing filter. Similarly, please ensure that there are M zeroes on the pole zero plot (**not** including the pole-zero cancellation at z = 1).

4.3.2 Results for M = 1:10



Figure 8: Pole zero plot ${\rm M}=01$



Figure 10: Pole zero plot ${\rm M}=02$



Figure 12: Pole zero plot ${\rm M}=03$



Figure 14: Pole zero plot M = 04



Figure 16: Pole zero plot M = 05

-0.5

-1

0 Real Part 0.5

1



Figure 17: Impulse response M = 06



Figure 18: Pole zero plot ${\rm M}=06$







Figure 20: Pole zero plot ${\rm M}=07$



Figure 21: Impulse response M = 08



Figure 22: Pole zero plot M = 08



Figure 23: Impulse response M = 9



Figure 24: Pole zero plot M = 9







Figure 26: Pole zero plot M = 10

4.4 Magnitude and Phase response

The magnitude and phase responses for M = 1:10 are depicted in Figures 27 - 36. Please ensure that there are an appropriate number of zeros in the magnitude response for the M value given.

4.4.1 Results



Figure 27: Frequency response M = 01



Frequency response of moving average FIR filter with M = 1

Figure 28: Frequency response M = 02



Figure 29: Frequency response M = 03



Figure 30: Frequency response M = 04



Figure 31: Frequency response M = 05



Figure 32: Frequency response M = 06



Figure 33: Frequency response M = 07



Figure 34: Frequency response M = 08



Figure 35: Frequency response M = 09



Figure 36: Frequency response M = 10

5 Question 5: More FIRmly Filtered

There are a number of factors to consider when evaluating this question.

5.1 MATLAB and Normalised Frequencies

Prior to any discussion of specific filter design approaches, it is useful to review conversion of frequencies from Hertz to normalised radian frequencies. Given a signal with frequency f_x , and a sampling frequency F_s , the normalised frequency ω_x is given by:

$$\omega_x = \frac{f_x}{F_s} 2\pi$$

It is worth exploring the notation that MATLAB uses prior to progressing any further. From http://au.mathworks.com/help/signal/ref/fir1.html?searchHighlight=fir1#input_ argument_wn, the values in the input vector ω_n are characterised as:

Frequency constraints, specified as a scalar, a two-element vector, or a multi-element vector. All elements of Wn must be strictly greater than 0 and strictly smaller than 1, where 1 corresponds to the Nyquist frequency: 0 < Wn < 1. The Nyquist frequency is half the sample rate or $\pi rad/sample$.

Converting the frequencies required for this filter to their normalised representation gives:

Thus the values in the ω_n vector will be [0.2, 0.3, 0.5, 0.6, 0.8, 0.9].

5.2 MATLAB Commands

Once these values have been obtained, creating a filter in MATLAB is a relatively simple process. The code snippet below demonstrates a generic approach using the fir1 command.

Listing 3: MATLAB Code for Multiband FIR Filter Design

```
1
   clear all; close all; clc
2
3
   % Define Fs — Sampling Frequency
   Fs = 2000;
4
5
6
   % Define M = 20
7
   M = 20;
8
9
   % Define pass bands using normalised frequencies - omit pi for simplicity
   wn = [200 300 500 600 900]*2/Fs;
11
12
   % Create multiband bandpass filter of order M
13
   bM = fir1(M, wn, 'DC-0');
14
   % Plot all of the filter magnitude responses
15
   hfvt = fvtool(bM, 1);
16
17
   legend(hfvt, ['Hamming M = ', M])
```

5.3 Pole Zero plots

The pole zero plots on the left, which vary in scale, demonstrate the numerical instability present in the filter coefficients. This will be covered in a later section. The pole zero plots on the right have been scaled to \pm 1.5 and provide more detail about the frequency response of each filter. Notice that as M increases, the pass band regions become more and more defined.



Figure 37: Pole zero plot for Hamming window Figure 38: Pole zero plot for Hamming window M = 20 M = 20



Figure 39: Pole zero plot for Hamming window Figure 40: Pole zero plot for Hamming window M = 50 M = 50



Figure 41: Pole zero plot for Hamming window Figure 42: Pole zero plot for Hamming window M=100 M=100



Figure 43: Pole zero plot for Hamming window Figure 44: Pole zero plot for Hamming window M = 150 M = 150



Figure 45: Pole zero plot for Hamming window Figure 46: Pole zero plot for Hamming window M = 200 M = 200

5.4 Magnitude and Phase Responses

5.4.1 Linear magnitude response

As the filter length increases, the transition bandwidth decreases and the response becomes sharper. Can you think of any advantages or disadvantages as a result of this?



Figure 47: Magnitude (linear) as a function of normalised frequency, for Hamming windowed FIR Filters of order 20, 50, 100, 150, 200.

5.4.2 Magnitude (dB) response

A similar decrease occurs in the log magnitude response as the filter length increases. The main lobe width decreases (given by $\frac{12\pi}{M}$), the sidebands narrow (due to the increased number of zeroes), and the stopband attenuation improves. Are there any other improvements or issues that you observed as you varied the filter length?



Figure 48: Magnitude (linear) as a function of normalised frequency, for Hamming windowed FIR Filters of order 20, 50, 100, 150, 200.

5.4.3 Phase response

The change in phase response as the filter length increases is quite interesting; as the filter length increases, so too does the phase distortion. Can you think of any particular situations where this would be problematic?



Figure 49: Phase as a function of normalised frequency, for Hamming windowed FIR Filters of order 20, 50, 100, 150, 200.

5.5 Group Delay

Group or envelope delay is the "time delay that a signal component of frequency ω undergoes as it passes from the input to the output of the system" [1]. The group delay pictured in Figure 47 is linear, meaning that all frequency components are subject to the same delay.



Figure 50: Group delay as a function of normalised frequency, for Hamming windowed FIR Filters of order 20, 50, 100, 150, 200.

Group delay is an important design consideration, particularly for control problems. Consider "Hamming 200" from Figure 47. It has a linear group delay of $\frac{M}{2} = 100$ samples, which is equivalent to $\frac{1}{20}$ th of a second when $F_s = 2$ kHz. This is approximately 50Hz, which is well below the Nyquist frequency of 1kHz. The design challenge in this situation has very little to do with the Nyquist frequency and a lot more to do with the latency of the controller. Any signal content above 50Hz will have observation latency, which is the difference between a disturbance and the subsequent response of the controller. This means that the controller is essentially unable to signal the plant to respond to any input signal content with a frequency greater than 50Hz. Filter design is largely application specific; clearly a compromise must be made between precise stop and pass bands, roll-off rates, and latency, amongst numerous other variables for each situation.

5.6 Numerical Stability

The optimisation routines MATLAB uses to calculate the filter coefficients result in some values that are extremely small numbers (in the order of 10^{-18}). These values are actually approximations of 0, and can be seen in Figure 51. The MATLAB code supplied below replaces these very small numbers with zeroes, and then plots the resulting frequency responses. The rounded (or truncated) coefficients are seen below in Figure 51. What impact do you think this will have on the filter performance? If you vary the truncation point from 10^{-10} to 10^{-4} , what do you expect would happen?



Figure 51: Fiilter coefficient bases, original data set, for Hamming windowed FIR Filters of order 20, 50, 100, 150, 200.



Figure 52: Filter coefficient bases, rounded data set, for Hamming windowed FIR Filters of order 20, 50, 100, 150, 200.

Listing 4: MATLAB Code for Multiband FIR Filter Design and Experimentation

```
1
    clear all; close all; clc
 2
 3
   % Define Fs — Sampling Frequency
   Fs = 2000;
 4
 5
 6
   % Define pass bands using normalised frequencies — omit pi for simplicity
 7
    wn = [200 300 500 600 800 900]*2/Fs;
 8
9
   % Create filters of various orders - Hamming window
   bHamm20 = fir1(20, wn, 'DC-0');
   bHamm50 = fir1(50, wn, 'DC-0');
12
   bHamm100 = fir1(100, wn, 'DC-0');
13
   bHamm150 = fir1(150, wn, 'DC-0');
   bHamm200 = fir1(200, wn, 'DC-0');
14
15
16
   % Plot all of the filter magnitude responses
17
   hfvt = fvtool(bHamm20, 1, bHamm50, 1, bHamm100, 1, bHamm150, 1, bHamm200, 1);
18
   legend(hfvt, 'Hamming 20', 'Hamming 50', 'Hamming 100', 'Hamming 150', 'Hamming 200
        ')
19
20 % Calculate log10 of filter coefficients in order to demonstrate numerical
21
    % instability
22
   lbHamm20 = log10(abs(bHamm20));
   lbHamm50 = log10(abs(bHamm50));
23
24
   lbHamm100 = log10(abs(bHamm100));
25
   lbHamm150 = log10(abs(bHamm150));
   lbHamm200 = log10(abs(bHamm200));
26
27
28
   % Plot log10 results
   figure()
29
   hold all
30
   plot(lbHamm20, '.')
31
32 plot(lbHamm50, '.')
33 plot(lbHamm100, '.')
34 plot(lbHamm150, '.')
   plot(lbHamm200, '.')
35
36
   logaxis = axis();
37
    title('Filter coefficients (base10) to illustrate numerical instability')
38
   xlabel('Coefficients'), ylabel('Base')
39
   legend('Hamming 20', 'Hamming 50', 'Hamming 100', 'Hamming 150', 'Hamming 200')
40
    % Correct the very small numbers and set them to a true zero
41
42
    rbHamm20=(abs(bHamm20)>1E-10).*(bHamm20);
43
    rbHamm50=(abs(bHamm50)>1E-10).*(bHamm50);
44
    rbHamm100=(abs(bHamm100)>1E-10).*(bHamm100);
45
    rbHamm150=(abs(bHamm150)>1E-10).*(bHamm150);
46
    rbHamm200=(abs(bHamm200)>1E-10).*(bHamm200);
47
48
   % Calculate log10 of rounded filter coefficients
49
   rlbHamm20 = log10(abs(rbHamm20));
50
   rlbHamm50 = log10(abs(rbHamm50));
51
   rlbHamm100 = log10(abs(rbHamm100));
52
   rlbHamm150 = log10(abs(rbHamm150));
   rlbHamm200 = log10(abs(rbHamm200));
53
54
```

```
55 |% Plot log10 results for rounded set, same axis as previous case.
56 figure()
57 hold all
58 axis(logaxis)
    plot(rlbHamm20, '.')
59
60 plot(rlbHamm50, '.')
61 plot(rlbHamm100, '.')
62 | plot(rlbHamm150, '.')
63 plot(rlbHamm200, '.')
64 | title('Filter coefficients (base10) to illustrate numerical stability')
    xlabel('Coefficients'), ylabel('Base')
65
66 | legend('Hamming 20', 'Hamming 50', 'Hamming 100', 'Hamming 150', 'Hamming 200')
67
68 % Plot pole zero plots
69 figure()
70 zplane(bHamm20, 1)
 71
    title('Pole Zero plot for Hamming window FIR filter M = 20')
 72
 73
    figure()
    zplane(bHamm20, 1), axis([-1.5 1.5 -1.5 1.5])
 74
    title(['Pole Zero plot for Hamming window FIR filter M = 20'])
 77
    figure()
 78
    zplane(bHamm50, 1)
 79
    title(['Pole Zero plot for Hamming window FIR filter M = 50'])
80
81
    figure()
    zplane(bHamm50, 1), axis([-1.5 1.5 -1.5 1.5])
 82
83
    title(['Pole Zero plot for Hamming window FIR filter M = 50'])
84
85 figure()
86
    zplane(bHamm100, 1)
    title(['Pole Zero plot for Hamming window FIR filter M = 100'])
87
88
89 figure()
    zplane(bHamm100, 1), axis([-1.5 1.5 -1.5 1.5])
90
91
    title(['Pole Zero plot for Hamming window FIR filter M = 100'])
92
93 figure()
94
    zplane(bHamm150, 1)
95
    title(['Pole Zero plot for Hamming window FIR filter M = 150'])
96
97
    figure()
98
    zplane(bHamm150, 1), axis([-1.5 1.5 -1.5 1.5])
99
    title(['Pole Zero plot for Hamming window FIR filter M = 150'])
100
    figure()
102
    zplane(bHamm200, 1)
103
    title(['Pole Zero plot for Hamming window FIR filter M = 200'])
104
105
    figure()
106
    zplane(bHamm200, 1), axis([-1.5 1.5 -1.5 1.5])
107
    title(['Pole Zero plot for Hamming window FIR filter M = 200'])
108
109 % Plot numerically stable filter responses
110 hfvt = fvtool(rbHamm20, 1, rbHamm50, 1, rbHamm100, 1, rbHamm150, 1, rbHamm200, 1);
```

References

[1] J. Proakis and D. Manolakis, *Digital signal processing*. Pearson Prentice Hall, 2007. [Online]. Available: http://books.google.com.au/books?id=H_5SAAAAMAAJ