

## EE422G Homework #13 (12 points)

1. (5 points) In this problem, you are asked to explore an important application of FFT: efficient computation of convolution. The impulse response of a system is given by

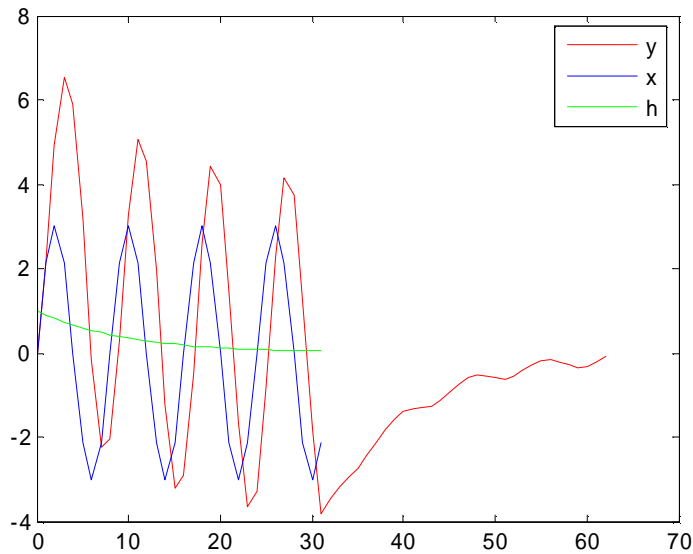
$$h(nT) = (0.9)^n \quad n = 0, 1, 2, \dots, 31$$

and the input to the system is given by

$$x(nT) = 3\sin(\frac{1}{4}\pi n) \quad n = 0, 1, 2, \dots, 31$$

You are asked to use the `fft` and `ifft` commands in Matlab to compute and plot the output  $y(nT) = h(nT) * x(nT)$ .

```
>> n=0:31;
>> h = 0.9.^n;
>> x = 3*sin(1/4*pi*n);
>> H = fft(h,63);           % fft pads zero
>> X = fft(x,63);
>> Y = X.*H;
>> y = ifft(Y);
>> plot(0:62,y,'r',0:31,x,'b',0:31,h,'g')
```



In addition, please answer the following questions:

- a. What is the minimum value of  $N$  in the  $N$ -point DFT (FFT) needed to compute this linear convolution?

$N$  must be large enough so that it exceeds the length of the output sequence, which is the sum of the lengths of the input and impulse sequence minus one, or 63.

- b. If the impulse response has  $N$  non-zero samples and the input has  $M$  non-zero samples, how many arithmetic operations (additions and multiplications) are needed to compute the linear operation from its definition, i.e.

$$y(nT) = \sum_{m=0}^{N-1} h(mT)x(nT - mT) \quad \text{for } n = 0, 1, \dots, M-1?$$

Each sample in the output sequence requires at most  $2 \cdot \max(N, M)$  arithmetic operations – the max operator is needed as less operations are needed in the beginning and at the end samples as the amount of overlap between the input and the impulse response is smaller. Since there are  $N+M-1$  non-zero output samples, the total number of operation is roughly  $2 \cdot \max(N, M) \cdot (N+M-1)$ . In the case when  $N=M$  and they are significantly bigger than 1, the above expression can be simplified as roughly equal to  $4N^2$ .

- c. Repeat b) but using the FFT method.

If FFT is used, both the input sequence and the impulse response need to first zero-padded to  $N+M-1$ . This process does not use any arithmetic operations (except perhaps one to compute what  $N+M-1$  is). Then the two forward FFT costs about  $2(N+M-1)\log(N+M-1)$  operations. The frequency samples are then multiplied with each other and this takes  $N+M-1$  operations. The final inverse FFT takes  $(N+M-1)\log(N+M-1)$ , making the total count of operations  $(N+M-1)[3\log(N+M-1)+1]$ . For large  $N=M$ , we have roughly  $6N \cdot \log N$ . As one can see this is significantly faster than the straightforward linear convolution.

2. (4 points) During lecture, we discussed the decimation-in-time method of implementing FFT in which we partition the sequence into even and odd time samples. There is an alternative approach called *decimation-in-frequency* which we are going to explore in this problem. Recall that a 8-point DFT is given by

$$X(k) = \sum_{n=0}^7 x(nT)W_8^{nk} \quad k = 0, 1, \dots, 7 \quad (1)$$

where  $W_N^{nk} = \exp\left(\frac{j2\pi nk}{N}\right)$ .

- a. Show that

$$X(k) = \sum_{n=0}^3 [x(n) + (-1)^k x(4+n)]W_8^{nk} \quad (2)$$

It is easy to see that  $W_8^{4k} = (-1)^k$ , thus the right hand side of (2) becomes

$$\begin{aligned} & \sum_{n=0}^3 [x(n) + W_8^{4k} x(4+n)]W_8^{nk} \\ &= \sum_{n=0}^3 [x(n)W_8^{nk} + x(4+n)W_8^{(n+4)k}] \\ &= \sum_{n=0}^7 x(n)W_8^{nk} = X(k) \end{aligned}$$

- b. For even  $k$ , show that equation (2) is equivalent to the 4-point DFT of the sequence  $y(n) = x(n) + x(4+n)$

For even  $k$ , we have  $W_8^{n2k} = W_4^{nk}$

Using (2) and the given substitution, we have

$$\begin{aligned}
 X(2k) &= \sum_{n=0}^3 [x(n) + (-1)^{2k} x(4+n)] W_8^{n2k} \\
 &= \sum_{n=0}^3 y(n) W_4^{nk} = Y(k)
 \end{aligned}$$

- c. For odd k, show that equation (2) is equivalent to the 4-point DFT of the sequence  $y(n) = [x(n) + x(4+n)] W_8^n$

There is a typo in this question so everyone is getting full credit for this part:

The substitution should be  $y(n) = [x(n) - x(4+n)] W_8^n$

$$\begin{aligned}
 X(2k+1) &= \sum_{n=0}^3 [x(n) - x(4+n)] W_8^{n(2k+1)} \\
 &= \sum_{n=0}^3 [x(n) - x(4+n)] W_8^n W_8^{n2k} \\
 &= \sum_{n=0}^3 y(n) W_4^{nk} = Y(k)
 \end{aligned}$$

- d. Draw the butterfly network of the 8-point DFT in terms of the preprocessing stage followed by two 4-point DFTs.

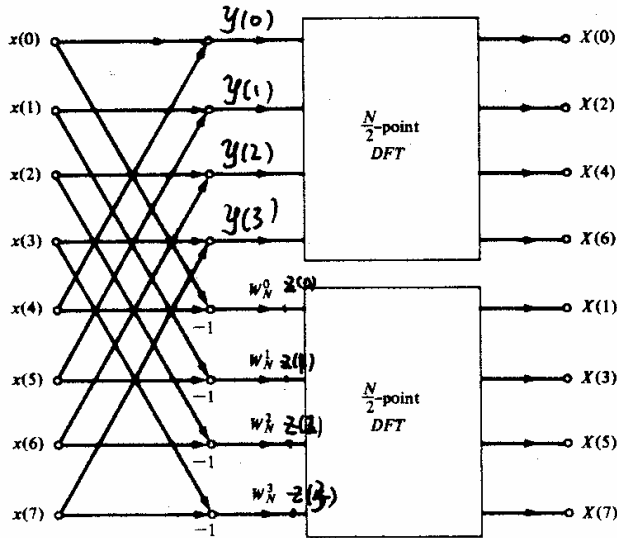


FIGURE 10-10. Flow graph after a single decimation.

3. (4 points) Filter design

- a. Response of the analog filter

i.  $w(t) = \delta(t) \Rightarrow W(s) = 1 \Rightarrow Y(s) = H_a(s)W(s) = \frac{5}{s+4} \Rightarrow y(t) = 5e^{-4t}u(t)$

ii.  $w(t) = u(t) \Rightarrow W(s) = \frac{1}{s} \Rightarrow Y(s) = \frac{5}{s(s+4)} = \frac{5/4}{s} + \frac{-5/4}{s+4} \Rightarrow y(t) = 5/4 [1 - e^{-4t}]u(t)$

iii.  $w(t) = e^{-4t}u(t) \Rightarrow W(s) = \frac{1}{s+4} \Rightarrow Y(s) = \frac{5}{(s+4)^2} \Rightarrow y(t) = 5te^{-4t}u(t)$

- b. For the low pass filter in part a), design a digital filter  $H(z)$  with  $T_s=10$  msec using the following techniques:

i)

### Impulse Invariant Design

$$\begin{aligned}y(t) &= \mathcal{L}^{-1}\{H_a(s)\} = \mathcal{L}^{-1}\left\{\frac{5}{s+4}\right\} = 5e^{-4t}u(t) \Rightarrow y(kT) = 5e^{-4kT}u(kT) \\ \Rightarrow Y(z) &= Z\{y(kT)\} = Z\left\{5[e^{-4T}]^k u(kT)\right\} = \frac{5}{1 - e^{-4T}z^{-1}} = \frac{5}{1 - e^{-0.04}z^{-1}} \\ \Rightarrow W(z) &= Z\{\delta_{\text{practical}}(kT)\} = Z\left\{\frac{1}{T}\delta(kT)\right\} = \frac{1}{T} \\ \Rightarrow H(z) &= \frac{Y(z)}{W(z)} = \frac{5T}{1 - e^{-4T}z^{-1}} = \frac{0.05}{1 - e^{-0.04}z^{-1}}\end{aligned}$$

ii)

### Step Invariant Design:

$$\begin{aligned}w(t) &= u(t) \Rightarrow W(s) = \frac{1}{s} \\ \Rightarrow y(t) &= \mathcal{L}^{-1}\{H_a(s)W(s)\} = \mathcal{L}^{-1}\left\{\frac{5}{s(s+4)}\right\} = 5/4[1 - e^{-4t}]u(t) \Rightarrow y(kT) = 5/4[1 - e^{-4kT}]u(kT) \\ \Rightarrow Y(z) &= Z\{y(kT)\} = Z\left\{5/4[1 - (e^{-4T})^k]u(kT)\right\} = \frac{5/4}{1 - z^{-1}} + \frac{5/4}{1 - e^{-0.04}z^{-1}}, W(z) = Z\{u(kT)\} = \frac{1}{1 - z^{-1}} \\ \Rightarrow H(z) &= \frac{Y(z)}{W(z)} = 5/4\left[1 + \frac{1 - z^{-1}}{1 - e^{-0.04}z^{-1}}\right] = \frac{5/4(1 - e^{-0.04})z^{-1}}{1 - e^{-0.04}z^{-1}}\end{aligned}$$

iii)

### Bilinear Transformation:

$$\begin{aligned}H(z) &= H_a(s)\Big|_{s=\frac{2}{T}\frac{z-1}{z+1}} = \frac{5}{\frac{2}{T}\frac{z-1}{z+1} + 4} = \frac{5}{\frac{2}{T}\frac{z-1}{z+1} + 4} = \frac{5(z+1)}{\frac{2}{T}z - \frac{2}{T} + 4z + 4} = \frac{5(z+1)}{\left(\frac{2}{T} + 4\right)z - \left(\frac{2}{T} - 4\right)} \\ &= \frac{5}{\left(\frac{2}{T} + 4\right)} \frac{(z+1)}{z - \frac{\frac{2}{T}-4}{\frac{2}{T}+4}} = \frac{5}{204} \frac{z+1}{z - \frac{196}{204}} = \frac{5}{204} \frac{1+z^{-1}}{1 - \frac{196}{204}z^{-1}}\end{aligned}$$

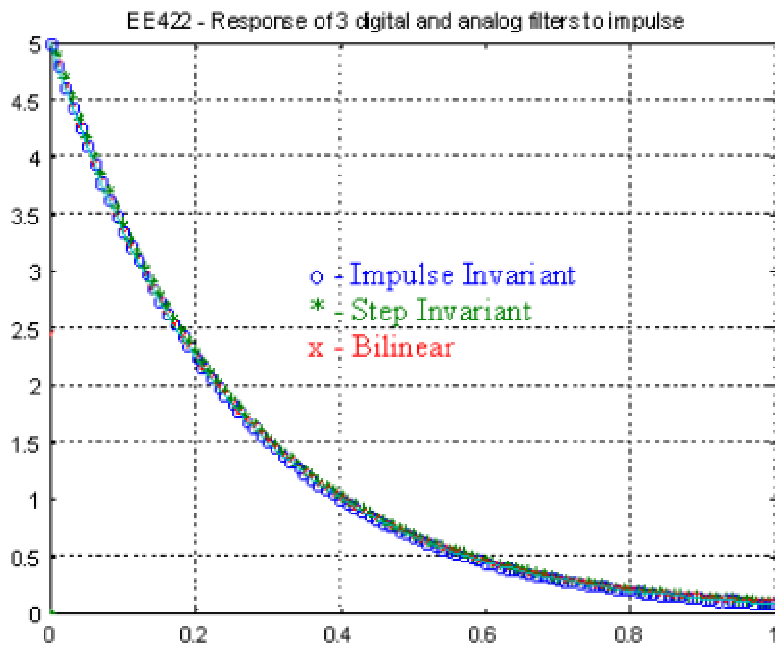
### **c) Matlab Code to obtain the plots:**

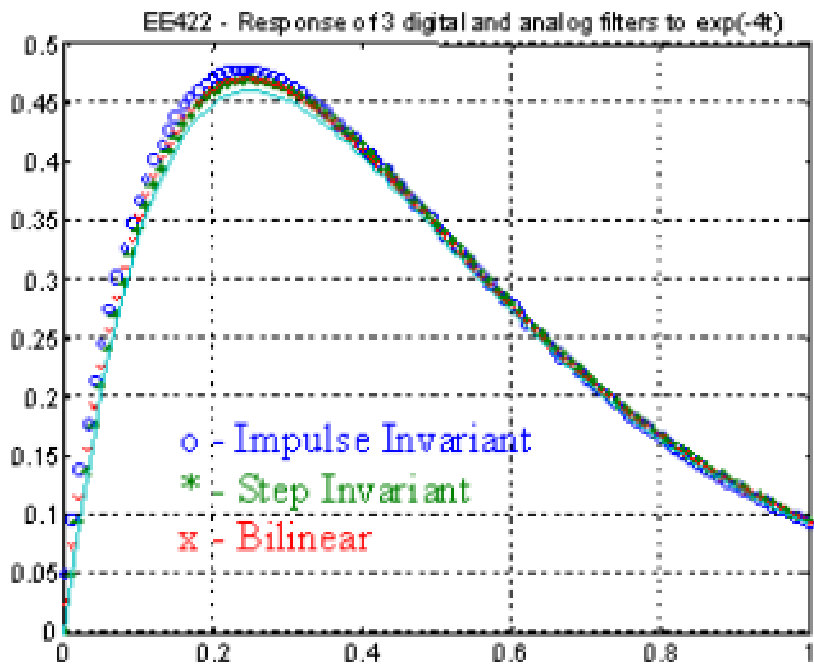
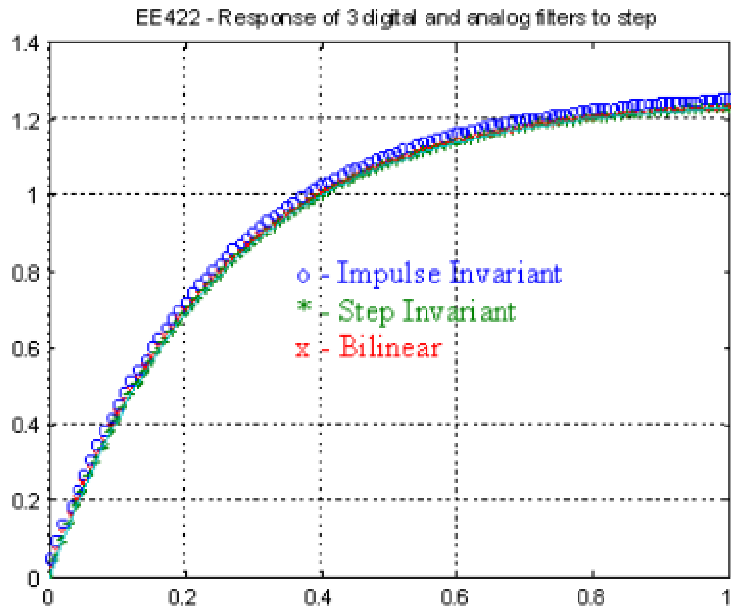
```
> T=0.01;
> w=1/T*[1 zeros(1,100)];
> w=1/T*[1 zeros(1,100)];
> w_impulse=1/T*[1 zeros(1,100)];
> w_step=ones(1,101);
> t=[0:T:1];
> w_exponential=exp(-4*t);
> num_impulse=5*T;
> den_impulse=[1 -exp(-0.04)];
> num_step=5/4*(1-exp(-0.04))*[0 1];
```

```

» den_step=[1 -exp(-0.04)];
» num_bilinear=5/204*[1 1];
» den_bilinear=[1 -196/204];
» y_impulse_impulse=filter(num_impulse,den_impulse,w_impulse);
» y_impulse_step=filter(num_impulse,den_impulse,w_step);
» y_impulse_exponential=filter(num_impulse,den_impulse,w_exponential);
» y_step_impulse=filter(num_step,den_step,w_impulse);
» y_step_step=filter(num_step,den_step,w_step);
» y_step_exponential=filter(num_step,den_step,w_exponential);
» y_bilinear_impulse=filter(num_bilinear,den_bilinear,w_impulse);
» y_bilinear_step=filter(num_bilinear,den_bilinear,w_step);
» y_bilinear_exponential=filter(num_bilinear,den_bilinear,w_exponential);
» y_analog_impulse=5*exp(-4*t);
» y_analog_step=5/4*(1-exp(-4*t));
» y_analog_exponential=5*t.*(exp(-4*t));
» plot(t,y_impulse_impulse,'o',t,y_step_impulse,'*',t, ...
y_bilinear_impulse,'x',t,y_analog_impulse)
» grid,title('EE422 - Response of 3 digital and analog filters to
impulse')
» plot(t,y_impulse_step,'o',t,y_step_step,'*',t,y_bilinear_step,'x',...
t,y_analog_step)
» grid,title('EE422 - Response of 3 digital and analog filters to step')
» plot(t,y_impulse_exponential,'o',t,y_step_exponential,'*',...
t,y_bilinear_exponential,'x',t,y_analog_exponential)
» grid,title('EE422 - Response of 3 digital and analog filters to exp(-
4t)')

```





As expected, the impulse invariant design response agrees exactly with the analog filter when the input is an impulse and similarly, the step invariant design response agrees exactly with the analog filter when the input is a step. The bilinear transformation does well when the analog response is relatively smooth.

4. (3 points) Design a digital filter for the analog prototype  $H(s) = \frac{1}{(s+5)(s+10)}$  using both the impulse-invariant and the bilinear methods. The sample period of the digital filter is 0.1 second. Plot the amplitude and phase responses of the analog filter as well as the two digital filters and comment on their differences.

```
>> num = 1;
>> den = conv([1 5],[1 10]);
>> freq = 1/0.1;
>> [bz,az] =impinvar(num,den,freq);
>> [bz1,az1] = bilinear(num,den,freq);      % Bilinear mapping
>> [ha,w] = freqs(num,den);
>> hd = freqz(bz,az,w/(2*pi),freq);
>> hd1 = freqz(bz1,az1,w/(2*pi),freq);
>> subplot(1,2,1); plot(w,abs(ha),'b',w,abs(hd),'r',w,abs(hd1),'g');
>> subplot(1,2,2); plot(w,phase(ha),'b',w,phase(hd),'r',w,phase(hd1),'g');
>> subplot(1,2,1); title('Magnitude'); xlabel('Freq(rad/s)');
>> subplot(1,2,2); title('Phase'); xlabel('Freq(rad/s)');
```

