EE430 DIGITAL SIGNAL PROCESSING

HOMEWORK 3

Name: Osman ALENBEY

Student ID: 1875616

2)

a) For k=0,1,2

$X_3[k] = \sum_{n=0}^{2} x[n]e^{-j\frac{2\pi}{3}kn} = x[0] + x[1]\, e^{-j\frac{2\pi}{3}k} + x[2]\, e^{-j\frac{4\pi}{3}k} = 1 + 3\, e^{-j\frac{2\pi}{3}k} + e^{-j\frac{4\pi}{3}k}$

$X_3[0] = 5$ $X_3[1] = 1 + 3\, e^{-j\frac{2\pi}{3}} + e^{-j\frac{4\pi}{3}} = -1-j1.7321$

$X_3[2] = 1 + 3\, e^{-j\frac{4\pi}{3}} + e^{-j\frac{8\pi}{3}} = -1+j1.7321$

Otherwise $X_3[k]=0$.

$X_5[k] = \sum_{n=0}^{4} x[n]e^{-j\frac{2\pi}{5}kn} = x[0] + x[1]\, e^{-j\frac{2\pi}{5}k} + x[2]\, e^{-j\frac{4\pi}{5}k} + x[3]\, e^{-j\frac{6\pi}{5}k} + x[4]\, e^{-j\frac{8\pi}{5}k} =$

$= 1 + 3\, e^{-j\frac{2\pi}{5}k} + e^{-j\frac{4\pi}{5}k}$

$X_5[0] = 5$ $X_5[1] = 1 + 3\, e^{-j\frac{2\pi}{5}} + e^{-j\frac{4\pi}{5}} = 1.1180-j3.4410$

$X_5[2] = 1 + 3\, e^{-j\frac{4\pi}{5}} + e^{-j\frac{8\pi}{5}} = -1.1180-j0.8123$ $X_5[3] = 1 + 3\, e^{-j\frac{6\pi}{5}} + e^{-j\frac{12\pi}{5}} = -1.1180+j0.8123$

$X_5[4] = 1 + 3\, e^{-j\frac{8\pi}{5}} + e^{-j\frac{16\pi}{5}} = 1.1180+j3.4410$

b) $X_3[k]= \tilde{X}_3[k]$ for k=0,1,2 $X_3[k]=0$ otherwise

$X_5[k]= \tilde{X}_5[k]$ for k=0,1,2,3,4 $X_5[k]=0$ otherwise

c) $x[n]=\frac{1}{3}\sum_{k=0}^{2} X_3[k]e^{j\frac{2\pi}{3}kn}$ for k=0,1,2 x[n]=0 otherwise

$x[n]=\frac{1}{5}\sum_{k=0}^{4} X_3[k]e^{j\frac{2\pi}{5}kn}$ for k=0,1,2,3,4 x[n]=0 otherwise

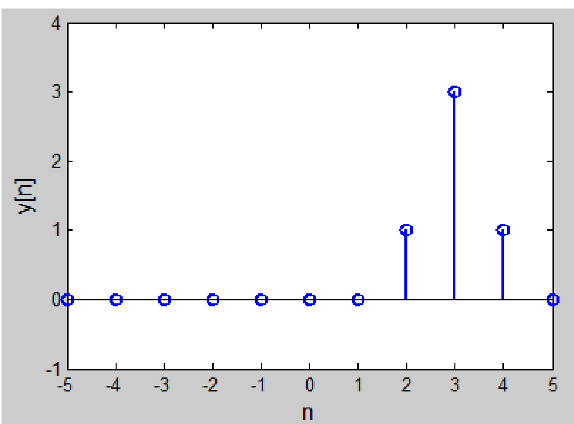
d)

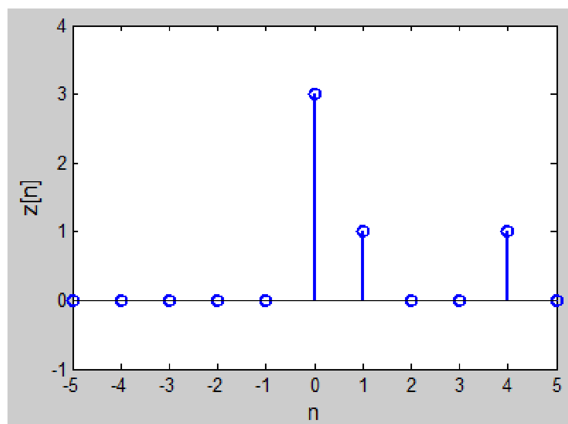
```
>> x

x =

     1     3     1

>> X3=fft(x,3)

X3 =

   5.0000 + 0.0000i  -1.0000 - 1.7321i  -1.0000 + 1.7321i

>> X5=fft(x,5)

X5 =

   5.0000 + 0.0000i   1.1180 - 3.4410i  -1.1180 - 0.8123i  -1.1180 + 0.8123i   1.1180 + 3.4410i
```

Results are same with the ones obtained in part a.

3)

a)





b) $y[n]=x[n-2]$ $z[n]=x[((n-4))_5]$

c) For k=0, 1, 2, 3, 4

$Y_5[k]=\sum_{n=0}^{4} y[n]e^{-j\frac{2\pi}{5}kn} = y[2]e^{-j\frac{4\pi}{5}k} + y[3]e^{-j\frac{6\pi}{5}k} + y[4]e^{-j\frac{8\pi}{5}k} = e^{-j\frac{4\pi}{5}k} + 3e^{-j\frac{6\pi}{5}k} + e^{-j\frac{8\pi}{5}k}$

$Y_5[0]=5$ $Y_5[1]= e^{-j\frac{4\pi}{5}} + 3e^{-j\frac{6\pi}{5}} + e^{-j\frac{8\pi}{5}}= -2.9271 + j2.1266$

$Y_5[2]= e^{-j\frac{8\pi}{5}} + 3e^{-j\frac{12\pi}{5}} + e^{-j\frac{16\pi}{5}}= 0.4271 - j1.3143$

$Y_5[3]= e^{-j\frac{12\pi}{5}} + 3e^{-j\frac{18\pi}{5}} + e^{-j\frac{24\pi}{5}}= 0.4271 + j1.3143$

$Y_5[4]= e^{-j\frac{16\pi}{5}} + 3e^{-j\frac{24\pi}{5}} + e^{-j\frac{32\pi}{5}}= -2.9271 - j2.1266$

Otherwise $Y_5[k]=0$.

For k=0, 1, 2, 3, 4

$Z_5[k]=\sum_{n=0}^{4} z[n]e^{-j\frac{2\pi}{5}kn} = z[0] + z[1]e^{-j\frac{2\pi}{5}k} + z[4]e^{-j\frac{8\pi}{5}k} = 3 + e^{-j\frac{2\pi}{5}k} + e^{-j\frac{8\pi}{5}k}$

$Z_5[0]=5$ $\quad$ $Z_5[1]= 3 + e^{-j\frac{2\pi}{5}} + e^{-j\frac{8\pi}{5}}=3.618$ $\quad$ $Z_5[2]= 3 + e^{-j\frac{4\pi}{5}} + e^{-j\frac{16\pi}{5}}= 1.382$

$Z_5[3]= 3 + e^{-j\frac{6\pi}{5}} + e^{-j\frac{24\pi}{5}}= 1.382$ $\quad$ $Z_5[4]= 3 + e^{-j\frac{8\pi}{5}} + e^{-j\frac{32\pi}{5}}= 3.618$

Otherwise $Z_5[k]=0$.

The sequences have 3-point DFTs. However, these are 5-point length sequences. So if we take DFTs with less than 5 points when we apply inverse DFT we cannot get the original sequence back. There will be some missing points and some values can be erroneous.

6)

a) It gives the first 3 samples of x[n].

$w_3[n]=3\delta[n]-2\ \delta[n-1]+ \delta[n-2]$

b) It gives the first 5 samples of x[n].

$w_5[n]=3\delta[n]-2\ \delta[n-1]+ \delta[n-2]+ \delta[n-3]-2\ \delta[n-4]$

c) Since DFT length is larger than the length of x[n], the result is x[n] itself in this case.

$w_8[n]=3\delta[n]-2\ \delta[n-1]+ \delta[n-2]+ \delta[n-3]-2\ \delta[n-4]- \delta[n-5]$

For further parts in this problem we must keep in mind that;

$y[n]=x[n] * h[n]=6\ \delta[n]-7\ \delta[n-1]+4\ \delta[n-2]+ \delta[n-3]-5\ \delta[n-4]+ \delta[n-6]$

d) Result will be a 3 point length sequence. The first point will be wrong with respect to original y[n].

$h_3[0]w_3[n]= 6\ \delta[n]-4\ \delta[n-1]+2\ \delta[n-2]$

$h_3[1]w_3[n-1]= -3\ \delta[n-1]+2\ \delta[n-2]- \delta[n-3]$

We normally expect the result to be the sum of two above expressions. But since we will obtain a three point sequence $-\delta[n-3]$ term in $h_3[1]w_3[n-1]$ will be added to the first term of resultant sequence. Hence;

$y_3[n]= 5\delta[n]-7\,\delta[n-1]+4\,\delta[n-2]$

A similar result will occur in part e.

e) $h_5[0]w_5[n]= 6\,\delta[n]-4\,\delta[n-1]+2\,\delta[n-2]+2\,\delta[n-3]-4\,\delta[n-4]$

$h_5[1]w_5[n-1]= -3\,\delta[n-1]+2\,\delta[n-2]-\delta[n-3]-\delta[n-4]+2\,\delta[n-5]$

$y_5[n]= 8\,\delta[n]-7\,\delta[n-1]+4\,\delta[n-2]+\delta[n-3]-5\,\delta[n-4]$

f) In this part we will have the correct convolution result. h[n] is a 2 point and x[n] is a 6 point sequence. We will need at least 6+2-1=7 points for the correct result which is the case in this part.

$y_8[n]=6\,\delta[n]-7\,\delta[n-1]+4\,\delta[n-2]+\delta[n-3]-5\,\delta[n-4]+\delta[n-6]$

g) As explained in parts d-f $y_3[n]$ and $y_5[n]$ have some missing points of y[n] and the first points of them are erroneous. But since there are enough DFT points in the evaluation of $y_8[n]$ it is directly equal to y[n].

7)

a) We know $X[k] = \sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi}{N}kn}$

The summation can be written by separating the sums of even and odd indexed elements.

$$X[k] = \sum_{n\text{ even}} x[n]e^{-j\frac{2\pi}{N}kn} + \sum_{n\text{ odd}} x[n]e^{-j\frac{2\pi}{N}kn} =$$

$$= \sum_{n=0}^{\frac{N}{2}-1} x[2n]e^{-j\frac{2\pi}{N}k2n} + \sum_{n=0}^{\frac{N}{2}-1} x[2n+1]e^{-j\frac{2\pi}{N}k(2n+1)} =$$

$$= \sum_{n=0}^{\frac{N}{2}-1} x[2n]e^{-j\frac{2\pi}{N}k2n} + e^{-j\frac{2\pi}{N}k}\sum_{n=0}^{\frac{N}{2}-1} x[2n+1]e^{-j\frac{2\pi}{N}k2n} = E[((k))]_{\frac{N}{2}} + e^{-j\frac{2\pi}{N}k}O[((k))]_{\frac{N}{2}}$$

Hence;

$$X[k] = E[((k))]_{\frac{N}{2}} + e^{-j\frac{2\pi}{N}k}O[((k))]_{\frac{N}{2}} \qquad \text{for } k=0, 1, 2\ldots\frac{N}{2}-1$$

This time number of k's to be computed has been reduced since the separated expressions are periodic with $\frac{N}{2}$ in k.

b)

a. $X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}$ for k=0, 1, 2,…, N-1

For each value of k between 0 and N-1 there are N multiplications above. Since we are dealing with totally N number of k values this makes N x N = N^2 multiplications. We also need N additions for the final result.

b. With the modified DFT expression for each value of k from 0 to $\frac{N}{2} - 1$ we must perform $\frac{N}{2}$ multiplications for even and $\frac{N}{2}$ multiplications for odd part. We will repeat that procedure for $\frac{N}{2}$ number of k values. So total number of multiplications is $\frac{N}{2} \times \left(\frac{N}{2} + \frac{N}{2}\right) = \frac{N^2}{2}$. We need $\frac{N}{2}$ additions for each of even and odd parts. Thus this makes a total of N additions which is the same number in part a.

c. Number of arithmetic operations has been reduced significantly in the second case. By going further in the expansion of DFT due to the symmetries of the terms these operations can be reduced even more. There is a butterfly pattern related to that and we will solve the DFT equation with $\frac{N}{2}\log_2 N$ multiplications totally. This is the idea behind the fast Fourier transform to solve these equations in computer faster and more efficiently.

8)

a) Impulse response is a P=3 length sequence. Thus, we need L+P-1=6 point DFT.

b) x[n] is a 12 point length sequence. With L=4 we will have 4 segments. They are given below.

$x_1[n]$=[1 2 3 4] $x_2[n]$=[-1 -2 -3 -4] $x_3[n]$=[1 2 3 4]

c)

```
>> x1

x1 =

     1     2     3     4

>> x2

x2 =

    -1    -2    -3    -4

>> x3

x3 =

     1     2     3     4

>> h

h =

     2    -1     1
```

```
>> X1=fft(x1,6)

X1 =

  10.0000 + 0.0000i  -3.5000 - 4.3301i   2.5000 + 0.8660i  -2.0000 + 0.0000i   2.5000 - 0.8660i  -3.5000 + 4.3301i

>> X2=fft(x2,6)

X2 =

 -10.0000 + 0.0000i   3.5000 + 4.3301i  -2.5000 - 0.8660i   2.0000 + 0.0000i  -2.5000 + 0.8660i   3.5000 - 4.3301i

>> X3=fft(x3,6)

X3 =

  10.0000 + 0.0000i  -3.5000 - 4.3301i   2.5000 + 0.8660i  -2.0000 + 0.0000i   2.5000 - 0.8660i  -3.5000 + 4.3301i

>> H=fft(h,6)

H =

   2.0000 + 0.0000i   1.0000 + 0.0000i   2.0000 + 1.7321i   4.0000 + 0.0000i   2.0000 - 1.7321i   1.0000 + 0.0000i
```

```
>> y1=ifft(X1.*H,6)

y1 =

    2.0000    3.0000    5.0000    7.0000   -1.0000    4.0000

>> y2=ifft(X2.*H,6)

y2 =

   -2.0000   -3.0000   -5.0000   -7.0000    1.0000   -4.0000

>> y3=ifft(X3.*H,6)

y3 =

    2.0000    3.0000    5.0000    7.0000   -1.0000    4.0000
```

d) $y[n]=y_1[n]+y_2[n-4]+y_3[n-8]=[2\ 3\ 5\ 7\ -3\ 1\ -5\ -7\ 3\ -1\ 5\ 7\ -1\ 4]$

In order to verify that I give a MATLAB screenshot below which uses “conv” command directly.

```
>> x

x =

     1     2     3     4    -1    -2    -3    -4     1     2     3     4

>> h

h =

     2    -1     1

>> y=conv(x,h)

y =

     2     3     5     7    -3     1    -5    -7     3    -1     5     7    -1     4
```