EEE 321 - Signals and Systems - Lab 4 - Tuna Şahin 22201730

Fourier Series Expansion

Question 1:

The signal can be expressed as: $y = 8 \times u(mod(t,18) - 7) \times u(10 - mod(t,18))$ which will make it easy to compute.

Q(1-a): Discretise the given signal:

```
subplot(1,5,1)
n1 = linspace(-4,31.9,360); %sample times with an interval of 0.1s
y1 = 8.*arrayfun(@u,mod(n1,18)-7).*arrayfun(@u,10-mod(n1,18)); %sample y for every value
of n
fineplot(n1,y1,'a) Discretised signal','n','y[n]',[-4 32],[-1 10],'on',[2000 400])
```

Q(1-b):

Analytical Fourier Series Expansion calculation:

$$a_k = \frac{1}{T}\int_{o.o.p} y(t)e^{-j\frac{2\pi}{T}kt}dt \Rightarrow \frac{1}{18}\int_0^{18} y(t)e^{-j\frac{\pi}{9}kt}dt \Rightarrow \frac{8}{18}\int_7^{10} e^{-j\frac{\pi}{9}kt}dt = \frac{-8}{j2\pi k}\left(e^{-j\frac{10\pi}{9}k} - e^{j\frac{-7\pi}{9}k}\right),\ \forall k \in \mathbb{Z} - \{0\}$$

$$a_0 = \frac{1}{18}\int_{o.o.p.} y(t)dt = \frac{24}{18}$$

We can then write the FSE as:

$$y_a(t) = \sum_{k=-\infty}^{\infty} a_k e^{j\frac{2\pi}{18}\mathrm{kt}} = \sum_{k=-\infty}^{-1} \frac{-8}{\mathrm{j}2\pi k}\left(e^{-j\frac{10\pi}{9}k}e^{j\frac{2\pi}{18}\mathrm{kt}} - e^{-j\frac{7}{9}k}e^{j\frac{2\pi}{18}\mathrm{kt}}\right) + a_0 + \sum_{k=1}^{\infty} \frac{-8}{\mathrm{j}2\pi k}\left(e^{-j\frac{10\pi}{9}k}e^{j\frac{2\pi}{18}\mathrm{kt}} - e^{-j\frac{7}{9}k}e^{j\frac{2\pi}{18}\mathrm{kt}}\right)$$

$$y_a(t) = \sum_{k=-\infty}^{-1} \frac{-8}{\mathrm{j}2\pi k}\left(e^{\mathrm{jk}\pi\left(\frac{t-10}{9}\right)} - e^{\mathrm{jk}\pi\left(\frac{t-7}{9}\right)}\right) + a_0 + \sum_{k=1}^{\infty} \frac{-8}{\mathrm{j}2\pi k}\left(e^{\mathrm{jk}\pi\left(\frac{t-10}{9}\right)} - e^{\mathrm{jk}\pi\left(\frac{t-7}{9}\right)}\right)$$

Q(1-c): Calculate and plot the spectrum

The signal is real valued. Therefore, its $a_k's$ are conjuagte symmetric. We can plot the spectrum for $\omega > 0$ as this will also include information for $\omega < 0$.

```
subplot(1,5,2)
ks1 = linspace(-300,300,601); %frequency values from -300 to 300
aks1 = arrayfun(@a1,ks1);      %calculate the coefficients for every frequec
finestem(ks1,real(aks1),'c) Spectrum(Real)','n','y3[n]',[-20 20],[-1.5 1.5],'on',[2000
400])
set(gca,'XTick',[0 10 20 30],'XTickLabel',{0,'10\pi/9','20\pi/9','30\pi/9'})
subplot(1,5,3)
finestem(ks1,imag(aks1),'c) Spectrum(Imag)','n','y3[n]',[-20 20],[-1.5 1.5],'on',[2000
400])
set(gca,'XTick',[0 10 20 30],'XTickLabel',{0,'10\pi/9','20\pi/9','30\pi/9'})
subplot(1,5,4)
```

```
finestem(ks1,abs(aks1),'c) Spectrum(Absolute)','n','y3[n]',[-20 20],[-0.5 1.5],'on',[2000
400])
set(gca,'XTick',[0 10 20 30],'XTickLabel',{0,'10\pi/9','20\pi/9','30\pi/9'})
```

Q(1-d): Calculate the Fourier Series Expansion approximation for $N = 150$

```
subplot(1,5,5)
sum1 = zeros([1 length(n1)]); %create a blank array to add our harmonics
for k = -150:150
    sum1 = sum1 + exp(2j*pi*k*n1/18) * a1(k); %add a complex sinusoidal multiplied with
its coefficient
end
fineplot(n1,real(sum1),'d) Fourier Approximation','n','z[n]',[-6 32],[-1 10],'on',[2000
400])
```

Since our signal is continous in nature; the signals were plotted using a continous graph style. Also, the values of the x axis was adjusted to display time instead of the discrete indexes. This way, the values of the signal represent our time. The signal that we obtained from the fourier expansion clearly resembles the original signal. However, there are some clear oscillations present that we not visible in the original signal. Plotting the real and imaginary components of the spectrum is sufficent. To get a better understanding about the nature of the signal, we can also plot the magnitude of the spectrum. We can see a sinc function emerging in the absolute case.

Q(1-e,f,g,h,i): Calculate the Fourier Series Expansion approximation for given N values.

```
clf;
letters = ['e','f','g','h','i']; %Question name for every iteration that is present in
the respective titles
bounds = [75 30 5 3 1]; %N value for every loop
for i = 1:5
    subplot(1,5,i)
    sum1 = zeros([1 length(n1)]); %blank array to add our harmonics
    for k = -1*bounds(i):bounds(i)
       sum1 = sum1 + exp(2j*pi*k*n1/18) * a1(k); %add the complex sinusoidals
    end
    fineplot(n1,real(sum1),strcat(letters(i),') N=',int2str(bounds(i))),'n','z[n]',[-6
32],[-2 10],'on',[1600 300])
end
```

The rising and falling edges of the fourier approximated signal converges to the original signal as N increases. However, as we converge we can not completely prevent spikes at the rising and falling edges. These spikes are called Gibbs Phonemenon and the area under those spikes converge to zero as $\lim_{N \to \infty}$. Gibbs Phonemenon is present when there is a discontinuity in the signal. Therefore, only leading to a pointwise inequality and a weak equality.

Q(1-i): Calculate and Plot the given Harmonics of the signal.

```
clf;
t1 = linspace(-30,30,601);
subplot(1,4,1)
fineplot(t1,a1(0)+t1.*0,'Zeroth Harmonic','n','z[n]',[-30 30],[-3 3],'on',[1600 300])
```

```
subplot(1,4,2)
fineplot(t1,a1(1)*exp(2j*pi*t1/18) + a1(-1)*exp(-2j*pi*t1/18),'First Harmonic','n','z[n]',
[-30 30],[-3 3],'on',[1600 300])
subplot(1,4,3)
fineplot(t1,a1(2)*exp(4j*pi*t1/18) + a1(-2)*exp(-4j*pi*t1/18),'Second
Harmonic','n','z[n]',[-30 30],[-3 3],'on',[1600 300])
subplot(1,4,4)
fineplot(t1,a1(3)*exp(6j*pi*t1/18) + a1(-3)*exp(-6j*pi*t1/18),'Third Harmonic','n','z[n]',
[-30 30],[-3 3],'on',[1600 300])
```

Question 2:

$y_a(t) = |5\cos\left(\frac{\pi}{9}t\right)|$ → The function has a fundamental period of 9. Without the absolute value the fundamental period would be 18.

Q(2-a): Discretise the given signal:

```
clf;
subplot(1,3,1)
n2 = linspace(-4,31.9,360); %sample times with an interval of 0.1s
y2 = 5*cos(((mod(n2-4.5,9)-4.5)*(pi/9)));
fineplot(n2,y2,'a) Discretised signal','n','y[n]',[-4 32],[-1 10],'on',[1600 400])
```

Q(2-b):

Analytical Fourier Series Expansion calculation:

$$\frac{5}{T}\int_{-4.5}^{4.5} cos\left(\frac{\pi}{9}t\right)e^{-j\frac{2\pi}{T}kt}dt \Longrightarrow \frac{5}{18}\int_{-4.5}^{4.5}\left(e^{j\frac{\pi}{9}t}+e^{-j\frac{\pi}{9}t}\right)e^{-j\frac{2\pi}{9}kt}dt \Longrightarrow \frac{5}{18}\int_{-4.5}^{4.5}e^{j\frac{\pi}{9}t}e^{-j\frac{2\pi}{9}kt}dt + \frac{5}{18}\int_{-4.5}^{4.5}e^{-j\frac{\pi}{9}t}e^{-j\frac{2\pi}{9}kt}dt$$

$$\frac{5}{18}\int_{-4.5}^{4.5}e^{j\frac{\pi}{9}(1-2k)t}dt + \frac{5}{18}\int_{-4.5}^{4.5}e^{j\frac{\pi}{9}(-1-2k)t}dt \Longrightarrow \frac{5}{18}\frac{1}{j\frac{\pi}{9}(1-2k)}\left(e^{j\frac{\pi}{2}(1-2k)}-e^{j\frac{-\pi}{2}(1-2k)}\right) + \frac{5}{18}\frac{1}{j\frac{\pi}{9}(-1-2k)}\left(e^{j\frac{\pi}{2}(-1-2k)}-e^{-j\frac{\pi}{2}(-1-2k)}\right)$$

$$a_k = \frac{5}{\pi(1-2k)}\sin\left(\frac{\pi}{2}(1-2k)\right) + \frac{5}{\pi(-1-2k)}\sin\left(\frac{\pi}{2}(-1-2k)\right), \forall k \in \mathbb{Z}$$

After realising $\sin\left(\frac{\pi}{2}(1-2k)\right) = -\sin\left(\frac{\pi}{2}(-1-2k)\right)$ we can further simplify the expression as:

$$a_k = \left[\frac{5}{\pi(1-2k)} + \frac{5}{\pi(1+2k)}\right]\sin\left(\frac{\pi}{2}(1-2k)\right)$$

Then we can write the FSE as:

$$y_a(k) = \sum_{k=-\infty}^{\infty}\left[\frac{5}{\pi(1-2k)} + \frac{5}{\pi(1+2k)}\right]\sin\left(\frac{\pi}{2}(1-2k)\right)e^{j\frac{2\pi}{9}\mathrm{kt}}$$

Q(2-c): Calculate and plot the spectrum

Since our signal is real valued and conjugate symmetric (even) , the $a_k's$ are also real valued and conjugate symmetric (even) . Since we know the imaginary component is always 0. We do not need to plot the imaginary component.

Since, the spectrum is even, we can plot values for $\omega > 0$ to contain information for $\omega < 0$.

```
subplot(1,3,2)
ks2 = linspace(-300,300,601);
aks2 = arrayfun(@a2,ks2);
hold on
finestem(ks2,real(aks2),'c) Spectrum','n','y3[n]',[0 20],[-1 4],'on',[1600 400])
set(gca,'XTick',[0 5 10 15],'XTickLabel',{0,'10\pi/9','20\pi/9','30\pi/9'})
```

Q(2-d): Calculate the Fourier Series Expansion approximation for $N = 150$

```
subplot(1,3,3)
sum2 = zeros([1 length(n2)]);
for k = -150:150
    sum2 = sum2 + exp(2j*pi*k*n2/9) * a2(k);
end
fineplot(n2,real(sum2),'d) Fourier Approximation','n','z[n]',[-6 32],[-1 10],'on',[1600
400])
```

Like the first question, our signal is continous in nature; the signals were plotted using a continous graph style. Also, the values of the x axis was adjusted to display time instead of the discrete indexes. Our Fourier Series Approximation looks nearly identical to our original function. The FSE converged to our original function much faster then the square wave did in the first question. This is because, compared to the spectrum of the first question, the spectrum of the second converges to zero much faster. The Gibbs Phonemenon cannot be observed in this signal's Fourier Series Approximation since the original function is continous.

Q(2-e,f,g,h,i): Calculate the Fourier Series Expansion approximation for given N values.

```
letters = ['e','f','g','h','i'];
bounds = [75 30 5 3 1];
for i = 1:5
    subplot(1,5,i)
    sum2 = zeros([1 length(n2)]);
    for k = -1*bounds(i):bounds(i)
       sum2 = sum2 + exp(2j*pi*k*n2/9) * a2(k);
    end
    fineplot(n2,real(sum2),strcat(letters(i),') N=',int2str(bounds(i))),'n','z[n]',[-6
32],[-2 10],'on',[1600 300])
end
```

Q(2-i): Calculate and Plot the given Harmonics of the signal.

Since the $a_k's$ are even, $a_{-k} = a_k$. Therefore, $a_{-k}e^{-j\frac{2\pi}{T}\text{kt}} + a_k e^{j\frac{2\pi}{T}\text{kt}}$ is real valued. We don't need to plot the imaginary parts of the harmonics.

```
clf;
```

```
t2 = linspace(-30,30,601);
subplot(1,4,1)
fineplot(t2,a2(0)+t2.*0,'Zeroth Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
subplot(1,4,2)
fineplot(t2,real(a2(1)*exp(2j*pi*t2/18) + a2(-1)*exp(-2j*pi*t2/18)),'First
Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
subplot(1,4,3)
fineplot(t2,real(a2(2)*exp(4j*pi*t2/18) + a2(-2)*exp(-4j*pi*t2/18)),'Second
Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
subplot(1,4,4)
fineplot(t2,real(a2(3)*exp(6j*pi*t2/18) + a2(-3)*exp(-6j*pi*t2/18)),'Third
Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
```

Question 3:

Q(3-a): Discretise the given signal:

```
clf
subplot(1,3,1)
n3 = linspace(-4,31.9,360); %sample times with an interval of 0.1s
y3 = (y2 + 5*cos(pi*n3/9))/2; %add normal cosine with rectified to obtain half rectified
fineplot(n3,y3,'a) Discretised signal','n','y[n]',[-4 32],[-1 10],'on',[1600 400])
```

Q(3-b):

Analytical Fourier Series Expansion calculation:

$$\frac{5}{18}\int_{-4.5}^{13.5}\cos\left(\frac{\pi}{9}t\right)u(4.5-t)e^{-j\frac{2\pi}{18}\text{kt}}\,\text{dt} \Longrightarrow \frac{5}{36}\int_{-4.5}^{4.5}\left(e^{j\frac{2\pi}{18}t}+e^{-j\frac{2\pi}{18}t}\right)e^{-j\frac{2\pi}{18}\text{kt}}\,\text{dt} \Longrightarrow \frac{5}{36}\int_{-4.5}^{4.5}e^{j\frac{2\pi}{18}t}e^{-j\frac{2\pi}{18}\text{kt}}\,\text{dt}+\frac{5}{36}\int_{-4.5}^{4.5}e^{-j\frac{2\pi}{18}t}e^{-j\frac{2\pi}{18}\text{kt}}\,\text{dt}$$

$$\frac{5}{36}\int_{-4.5}^{4.5}e^{j\frac{\pi}{18}t(2-2k)}\,\text{dt}+\frac{5}{36}\int_{-4.5}^{4.5}e^{j\frac{\pi}{18}t(-2-2k)}\,\text{dt} \Longrightarrow a_k=\frac{5}{4}\frac{1}{j\pi\,(1-k)}\left(e^{j\frac{\pi}{2}(1-k)}-e^{-j\frac{\pi}{2}(1-k)}\right)\frac{2j}{2j}+\frac{5}{4}\frac{1}{j\pi\,(-1-k)}\left(e^{j\frac{\pi}{2}(-1-k)}-e^{-j\frac{\pi}{2}(-1-k)}\right)\frac{2j}{2j}$$

$$a_k=\frac{5}{2\pi(1-k)}\sin\left(\frac{\pi}{2}(1-k)\right)+\frac{5}{2\pi(-1-k)}\sin\left(\frac{\pi}{2}(-1-k)\right),\ \forall\, k\in\mathbb{Z}-\{-1,1\}$$

Since the value of 1+k and 1-k are 0 for -1,1. The seperated integral on the left and right will be equal to 9 for -1,1 respectively. Therefore a_{-1}, a_1 :

$a_{-1}=\frac{5}{4}+\frac{5}{2\pi(1-k)}\sin\left(\frac{\pi}{2}(1-k)\right)=\frac{5}{4}$ and $a_1=\frac{5}{4}+\frac{5}{2\pi(-1-k)}\sin\left(\frac{\pi}{2}(-1-k)\right)=\frac{5}{4}$

After realising $\sin\left(\frac{\pi}{2}(1-k)\right)=-\sin\left(\frac{\pi}{2}(-1-k)\right)$ we can further simplify the expression as:

$$a_k=\left[\frac{5}{2\pi(1-k)}+\frac{5}{2\pi(1+k)}\right]\sin\left(\frac{\pi}{2}(1-1k)\right)$$

$$y_a(t) = \sum_{k=-\infty}^{-2} \frac{5}{2\pi}\sin\left(\frac{\pi}{2}(1-k)\right)\frac{1}{1-k^2}e^{j\frac{2\pi}{18}\text{kt}} + a_{-1} + a_0 + a_1 + \sum_{k=-\infty}^{-2} \frac{5}{2\pi}\sin\left(\frac{\pi}{2}(1-k)\right)\frac{1}{1-k^2}e^{j\frac{2\pi}{18}\text{kt}}$$

$$y_a(t) = \left[\sum_{k=-\infty}^{-2} \frac{5}{2\pi}\sin\left(\frac{\pi}{2}(1-k)\right)e^{j\frac{2\pi}{18}\text{kt}}\frac{1}{1-k^2}\right] + \frac{5}{4} + \frac{5}{\pi} + \frac{5}{4} + \left[\sum_{k=-\infty}^{-2} \frac{5}{2\pi}\sin\left(\frac{\pi}{2}(1-k)\right)\frac{1}{1-k^2}e^{j\frac{2\pi}{18}\text{kt}}\right]$$

Q(3-c): Calculate and plot the spectrum

Since our signal is real valued and conjugate symmetric (even) , the $a_k's$ are also real valued and conjugate symmetric (even) . Since we know the imaginary component is always 0. We do not need to plot the imaginary component.

Since, the spectrum is even, we can plot values for $\omega > 0$ to contain information for $\omega < 0$.

```
subplot(1,3,2)
ks3 = linspace(-300,300,601);
aks3 = arrayfun(@a3,ks3);
hold on
finestem(ks3,real(aks3),'c) Spectrum','n','y3[n]',[0 20],[-0.5 2],'on',[1600 400])
set(gca,'XTick',[0 5 10 15],'XTickLabel',{0,'10\pi/9','20\pi/9','30\pi/9'})
```

Q(3-d): Calculate the Fourier Series Expansion approximation for $N = 150$

```
subplot(1,3,3)
sum3 = zeros([1 length(n3)]);
for k = -150:150
    sum3 = sum3 + exp(2j*pi*k*n3/18) * a3(k);
end
fineplot(n3,real(sum3),'d) Fourier Approximation','n','z[n]',[-6 32],[-1 10],'on',[1600
400])
```

Just like the first two questions, our signal is continous in nature; the signals were plotted using a continous graph style. Also, the values of the x axis was adjusted to display time instead of the discrete indexes.The Fourier Series Approximation looks nearly identical to our original function. The FSE converged to our original function much faster then the square wave did in the first question. This is because, compared to the spectrum of the first question, the spectrum of the third converges to zero much faster. The Gibbs Phonemenon cannot be observed in this signal's Fourier Series Approximation too, since, the original function is again continous.

Q(3-e,f,g,h,i): Calculate the Fourier Series Expansion approximation for given N values.

```
letters = ['e','f','g','h','i'];
bounds = [75 30 5 3 1];
for i = 1:5
    subplot(1,5,i)
    sum3 = zeros([1 length(n3)]);
    for k = -1*bounds(i):bounds(i)
       sum3 = sum3 + exp(2j*pi*k*n3/18) * a3(k);
    end
    fineplot(n3,real(sum3),strcat(letters(i),') N=',int2str(bounds(i))),'n','z[n]',[-6
32],[-2 10],'on',[1600 300])
```

```
end
```

Q(3-i): Calculate and Plot the given Harmonics of the signal.

Since the a_k's are even, $a_{-k} = a_k$. Therefore, $a_{-k}e^{-j\frac{2\pi}{T}\mathrm{kt}} + a_k e^{j\frac{2\pi}{T}\mathrm{kt}}$ is real valued. We don't need to plot the imaginary parts of the harmonics.

```
clf;
t3 = linspace(-30,30,601);
subplot(1,4,1)
fineplot(t3,a3(0)+t3.*0,'Zeroth Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
subplot(1,4,2)
fineplot(t3,real(a3(1)*exp(2j*pi*t3/18) + a3(-1)*exp(-2j*pi*t3/18)),'First
Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
subplot(1,4,3)
fineplot(t3,real(a3(2)*exp(4j*pi*t3/18) + a1(-2)*exp(-4j*pi*t3/18)),'Second
Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
subplot(1,4,4)
fineplot(t3,real(a3(3)*exp(6j*pi*t3/18) + a3(-3)*exp(-6j*pi*t3/18)),'Third
Harmonic','n','z[n]',[-30 30],[-4 4],'on',[1600 300])
```

Functions:

Functions for Fourier Series Coefficients:

Square Wave Fourier Series Coefficients:

```
function ak = a1(k)
    if k == 0
        ak = 24/18;
    else
        ak = -(exp((-10j/9)*pi*k)-exp((-7j/9)*pi*k))*8/(2j*pi*k);%*exp((2j*pi*k)/18);
    end
end
```

Full Wave Rectifier Fourier Series Coefficients:

```
function ak = a2(k)
    ak = 5*(1/(pi*(1-2*k)) * sin((pi / 2) * (1-2*k)) + 1/(pi*(-1-2*k)) * sin((pi / 2) *
(-1-2*k)));
end
```

Half Wave Rectifier Fourier Series Coefficients:

```
function ak = a3(k)
    if k == 1
        ak = (5/4) + (5/(2*pi*(-1-k)))*sin(0.5*pi*(-1-k));
    elseif k == -1
        ak = (5/(2*pi*(1-k)))*sin(0.5*pi*(1-k)) + (5/4);
```

```
    else
        ak = (5/(2*pi*(1-k)))*sin(0.5*pi*(1-k)) + (5/(2*pi*(-1-k)))*sin(0.5*pi*(-1-k));
    end
end
```

Unit Step Function:

```
function un = u(n)
    if n < 0
        un = 0;
    else
        un = 1;
    end
end
```

fineplot and finestem Functions:

The stem and plot functions with other configurations to get the same style as the book:

- titlename,xaxisname,yaxisname all take string values that determine the corresponding text.
- xlimits,ylimits take 1x2 arrays that hold the limits of the plot in arbitrary order
- holdstate should take holdstate = 'off' as an input if you don't want to plot to be held any other input will not hold the plot

```
function fineplot(n,x,titlename,xaxisname,yaxisname,xlimits,ylimits,holdstate,size)
    %even if the limits are reverse this section corrects the order
    if ylimits(1) > ylimits(2)
        ylimits = [ylimits(2) ylimits(1)];
    end
    if xlimits(1) > xlimits(2)
        xlimits = [xlimits(2) xlimits(1)];
    end

    %these measures are necessary for positioning arrows, labels etc.
    dx = (xlimits(2)-xlimits(1))/65;
    dy = (ylimits(2)-ylimits(1))/85;

    %this section is to not plot values that overlap with the arrows
    lowindex = 1;
    highindex = length(n);
    for i = n
        if (i > xlimits(1) + dx) && (lowindex == 1)
            lowindex = find(n==i);
        end
        if (i > xlimits(2) - dx) && (highindex == 1)
            highindex = find(n==i)-1;
        end
    end
    if n(lowindex) > 0 %we allow points to be drawn at the origin
        lowindex = lowindex - 1;
```

```
    end

    if n(highindex) < 0 %we allow points to be drawn at the origin
        highindex = highindex + 1;
    end


    n = n(lowindex:highindex);
    x = x(lowindex:highindex);

    %this section is responsible for axis configuration
    plot(n,x)
    xlim(xlimits)
    ylim(ylimits)
    set(get(gca,'XLabel'),'Visible','on')
    set(gca,'XAxisLocation','origin', 'box','off')
    set(gca,'YAxisLocation','origin')
    set(get(gca, 'XAxis'), 'FontWeight', 'bold')
    set(get(gca, 'YAxis'), 'FontWeight', 'bold');
    set(get(gca,'YLabel'),'Visible','on')
    set(gca,'Layer','top')
    set(gcf,'position',[(xlimits(2)-xlimits(1))/2 , (ylimits(2)-ylimits(1))/2 , size(1) ,
size(2)])

    %deletes the ticks that overlap with arrows
    xticks('auto');
    xt = xticks;
    xticks(xt(2:length(xt)-1));
    yticks('auto');
    yt = yticks;
    yticks(yt(2:length(yt)-1));

    % determining the ylevel to draw the arrows on the x axis
    if ((ylimits(1) * ylimits(2)) < 0)
        xal = 0;
    elseif ylimits(1) < 0
        xal = ylimits(2);
    else
        xal = ylimits(1);
    end

    %plotting the arrows
    hold on
    if xlimits(2) > 0
        plot([(xlimits(2)-dx) xlimits(2) (xlimits(2)-dx)],[xal+dy xal xal-dy],'k') %xaxis
right arrow
    end
    if xlimits(1) < 0
        plot([xlimits(1)+dx xlimits(1) xlimits(1)+dx],[xal+dy xal xal-dy],'k') %xaxis
left arrow
```

```
    end
    plot([0 0],[ylimits(2) ylimits(1)],'k')% y axis
    if ylimits(2) > 0
        plot([-dx/2 0 dx/2],[ylimits(2)-dy ylimits(2) ylimits(2)-dy],'k') %yaxis top
arrow
    end
    if ylimits(1) < 0
        plot([-dx/2 0 dx/2],[ylimits(1)+dy ylimits(1) ylimits(1)+dy],'k') %yaxis bottom
arrow
    end

    %repositioning title & label locations
    label_h1 = xlabel(xaxisname);
    label_h1.Position(1) = xlimits(2)+dx; % change horizontal position of xlabel.
    label_h1.Position(2) = dy; % change vertical position of xlabel.c5
    label_h2 = ylabel(yaxisname,rotation=0);
    label_h2.Position(1) = dx; % change horizontal position of ylabel.
    label_h2.Position(2) = ylimits(2)+dy; % change vertical position of ylabel.
    title(titlename);
    if ((xlimits(1) * xlimits(2)) < 0) && (xlimits(1)+xlimits(2) < ((xlimits(2)-
xlimits(1))/3))
        set(get(gca,'title'),'Position', [20*dx ylimits(2)-dy]) %prevents the title from
colliding with ylabel
    end

    %holds or doesnt hold
    if strcmp(holdstate,'off')
        hold off
    else
        hold on
    end
end

function finestem(n,x,titlename,xaxisname,yaxisname,xlimits,ylimits,holdstate,size)
    %even if the limits are reverse this section corrects the order
    if ylimits(1) > ylimits(2)
        ylimits = [ylimits(2) ylimits(1)];
    end
    if xlimits(1) > xlimits(2)
        xlimits = [xlimits(2) xlimits(1)];
    end

    %these measures are necessary for positioning arrows, labels etc.
    dx = (xlimits(2)-xlimits(1))/65;
    dy = (ylimits(2)-ylimits(1))/85;

    %this section is to not plot values that overlap with the arrows
    lowindex = n(1);
    highindex = n(1);
    for i = n
```

```
        if (i > xlimits(1) + dx) * (lowindex == n(1))
            lowindex = find(n==i);
        end
        if (i > xlimits(2) - dx) * (highindex == n(1))
            highindex = find(n==i)-1;
        end
    end
    if n(lowindex) > 0 %we allow points to be drawn at the origin
        lowindex = lowindex - 1;
    end
    if n(highindex) < 0 %we allow points to be drawn at the origin
        highindex = highindex + 1;
    end
    n = n(lowindex:highindex);
    x = x(lowindex:highindex);

    %this section is responsible for axis configuration
    stem(n,x,'filled','MarkerSize',3)
    xlim(xlimits)
    ylim(ylimits)
    set(get(gca,'XLabel'),'Visible','on')
    set(gca,'XAxisLocation','origin', 'box','off')
    set(gca,'YAxisLocation','origin')
    set(get(gca, 'XAxis'), 'FontWeight', 'bold')
    set(get(gca, 'YAxis'), 'FontWeight', 'bold');
    set(get(gca,'YLabel'),'Visible','on')
    set(gca,'Layer','top')
    set(gcf,'position',[(xlimits(2)-xlimits(1))/2 , (ylimits(2)-ylimits(1))/2 , size(1) ,
size(2)])

    %deletes the ticks that overlap with arrows
    xticks('auto');
    xt = xticks;
    xticks(xt(2:length(xt)-1));
    yticks('auto');
    yt = yticks;
    yticks(yt(2:length(yt)-1));

    % determining the ylevel to draw the arrows on the x axis
    if ((ylimits(1) * ylimits(2)) < 0)
        xal = 0;
    elseif ylimits(1) < 0
        xal = ylimits(2);
    else
        xal = ylimits(1);
    end

    %plotting the arrows
    hold on
    if xlimits(2) > 0
```

```
        plot([(xlimits(2)-dx) xlimits(2) (xlimits(2)-dx)],[xal+dy xal xal-dy],'k') %xaxis
right arrow
    end
    if xlimits(1) < 0
        plot([xlimits(1)+dx xlimits(1) xlimits(1)+dx],[xal+dy xal xal-dy],'k') %xaxis
left arrow
    end
    plot([0 0],[ylimits(2) ylimits(1)],'k')% y axis
    if ylimits(2) > 0
        plot([-dx/2 0 dx/2],[ylimits(2)-dy ylimits(2) ylimits(2)-dy],'k') %yaxis top
arrow
    end
    if ylimits(1) < 0
        plot([-dx/2 0 dx/2],[ylimits(1)+dy ylimits(1) ylimits(1)+dy],'k') %yaxis bottom
arrow
    end

    %repositioning title & label locations
    label_h1 = xlabel(xaxisname);
    label_h1.Position(1) = xlimits(2)+dx; % change horizontal position of xlabel.
    label_h1.Position(2) = dy; % change vertical position of xlabel.c5
    label_h2 = ylabel(yaxisname,rotation=0);
    label_h2.Position(1) = dx; % change horizontal position of ylabel.
    label_h2.Position(2) = ylimits(2)+dy; % change vertical position of ylabel.
    title(titlename);
    if ((xlimits(1) * xlimits(2)) < 0) && (xlimits(1)+xlimits(2) < ((xlimits(2)-
xlimits(1))/3))
        set(get(gca,'title'),'Position', [20*dx ylimits(2)-dy]) %prevents the title from
colliding with ylabel
    end

    %holds or doesnt hold
    if strcmp(holdstate,'off')
        hold off
    else
        hold on
    end
end
```