EE-391
Matlab Assignement 1

Section: 1

Barkın Saday

21902967

Question 1:

a)

-Below is a sketch for the calculations by hand can be seen. However, the real calculations that are used for plotting is done in MATLAB. Which can also be seen in the code.

- MATLAB PROJECT 1 -

Q.1.

• The signal $X(t)$ has a period $T_0 = 2$.

• Than we have $a_k = \frac{1}{2} \cdot \int_0^2 x(t) \cdot e^{-j \cdot \pi k t} dt$

We need to split the integral to two part $[0,1]$ and $[1,2]$ to solve it. Hence we get:

$$a_k = \frac{1}{2} \cdot \left(\int_0^1 -1 \cdot e^{-j\pi k t} dt + \int_1^2 e^{-j\pi k t} dt \right) =$$

$$\frac{1}{2} \cdot \left(\int_1^2 e^{-j\pi k t} dt - \int_0^1 e^{-j\pi k t} dt \right) =$$

$$\frac{j e^{-2j\pi k} \cdot \left(-1 + e^{j\pi k}\right)^2}{2\pi k}$$ we can use Eulers Formula

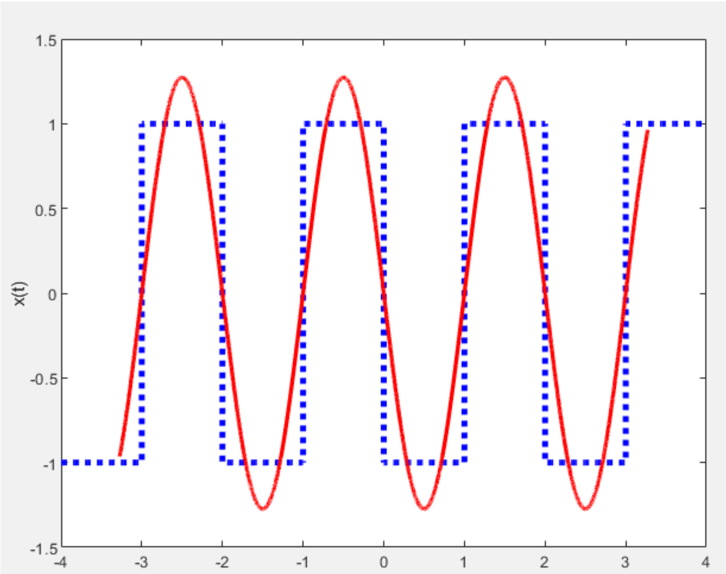
to write in the following way:

$$a_k = \frac{j e^{-j\pi k} \cdot (\cos(\pi k) - 1)}{\pi k}$$

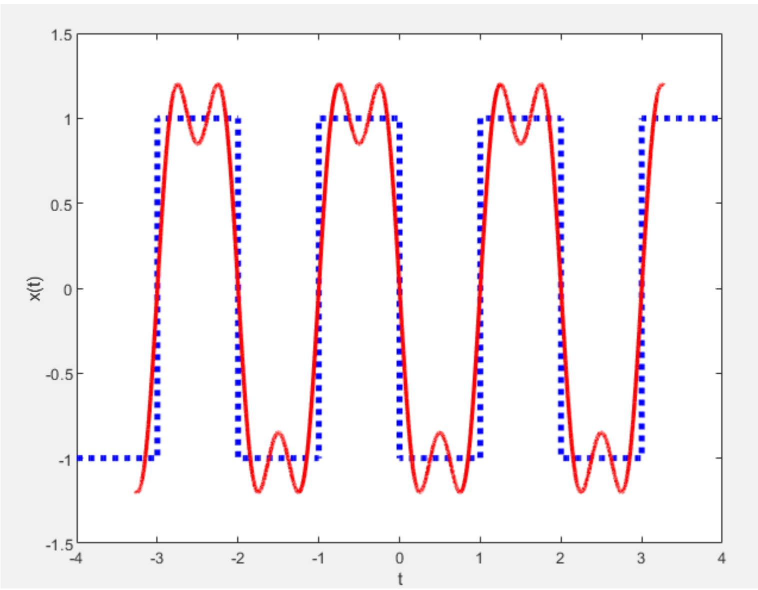
→ next page

CS CamScanner ile tarandı

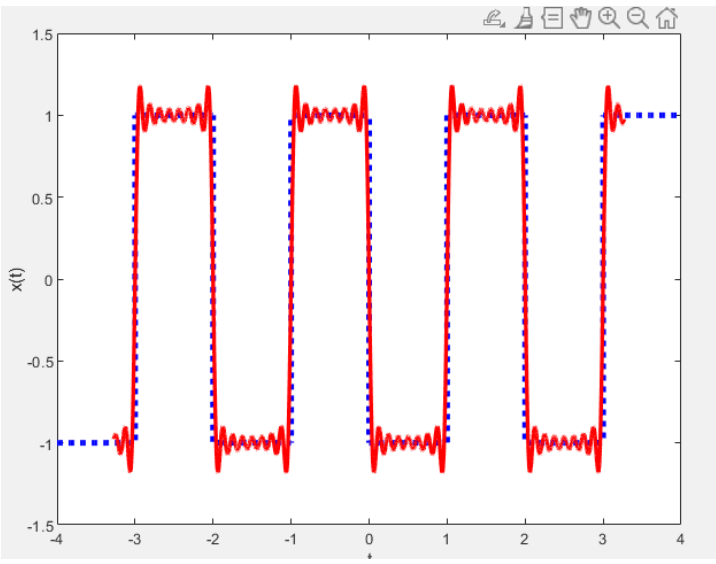
-The square wave function we have has a period T = 2. Below the plots with different range of k values can be seen on top of our square wave function so that we can visualize how the Fourier Series converges to the square wave function.



k between [-2, 2]



k between [-4, 4]

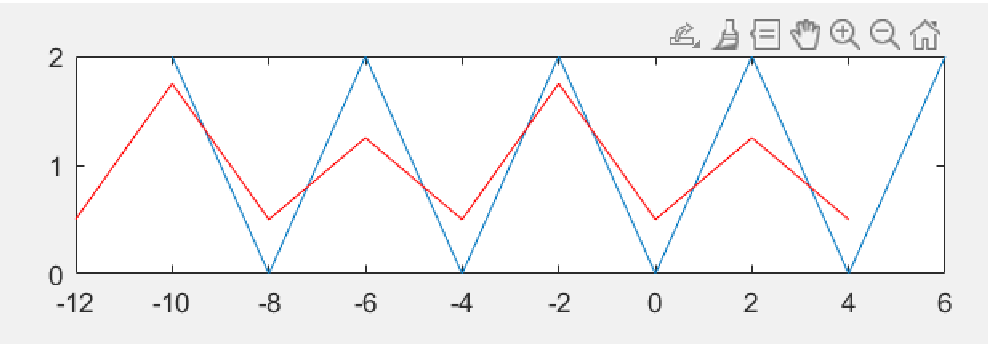


k between [-16, 16]

-As we can see in the figures as we increase the range of k (increase the number of summations) the Fourier Series gets closer to to the actual square wave function. This is what we were expecting since with infinite many summation the series converges to the square wave function.

b)

-This part is similar to “part a”. Instead of a square wave function we have a triangle wave function which goes between [0, 2] and has a period T = 4. Below is plotted graph with the Fourier series representation with k in range [-2, 2] on top of our actual triangle wave function.



Question 2:

a)

-The closed form of the given signals are found using Euler’s Formula.

• Notice $\cos(\pi k) = 1$ when k is an even number

so $\cos(\pi k) - 1 = 0$ when k is even. For odd k's we can calculate by Euler's Formula:

$k=-4$, $a_{-4} = 0$

$k=-3$, $a_{-3} = \frac{-2j}{3\pi}$

$k=-2$, $a_{-2} = 0$

$k=-1$, $a_{-1} = \frac{-2j}{\pi}$

$k=0$, in this case we have $0/0$ so we need to check the limit. $a_0 = 0$

$k=1$, $a_1 = \frac{2j}{\pi}$

$k=2$, $a_2 = 0$

$k=3$, $a_3 = \frac{2j}{3\pi}$

$k=4$, $a_4 = 0$

CamScanner ile tarandı

• $x(t) = \begin{cases} -x & -2 \le x < 0 \\ x & 0 \le x < 2 \end{cases}$ has a period of $T=4$

We can write as $x(t) = a_k \cdot e^{jk2\pi \cdot \frac{1}{4} \cdot t}$

where $a_k = \frac{1}{4} \int_{-2}^{2} x(t) \cdot e^{-j(\frac{\pi}{2}) \cdot k t} \, dt$

$= \int_{0}^{2} t \cdot e^{-j \cdot \pi/2 \cdot k t} \, dt - \int_{-2}^{0} t \cdot e^{-j \cdot \pi/2 \cdot k t} \, dt$

$= \frac{8 \cdot (\pi k \sin(\pi k) + \cos(\pi k) - 1)}{\pi^2 \cdot k^2} \rightarrow$ odd function

$k=-4$, $a_{-4} = 0$

$k=-3$, $a_{-3} = \frac{-16}{9\pi^2}$

$k=-2$, $a_{-2} = 0$

$k=-1$, $a_{-1} = \frac{-16}{\pi^2}$

$k=0$, $a_0 =$ indeterminate

as $k \rightarrow 0$, $a_0 \rightarrow 4$

$k=1$, $a_1 = \frac{-16}{\pi^2}$

$k=2$, $a_2 = 0$

$k=3$, $a_3 = \frac{-16}{9\pi^2}$

$k=4$, $a_4 = 0$

next page

CamScanner ile tarandı

Using Euler's Formula: $e^{j\theta} = \cos\theta + j\sin\theta$

and $\cos\theta = \frac{e^{j\theta} + e^{-j\theta}}{2}$ we have

$Signal_1 = \cos(2\pi \cdot 220 \cdot 2^{10/12} \cdot \cancel{t8})$ 1/8000

$Signal_2 = \cos(2\pi \cdot 220 \cdot 2^{6/12} \cdot t2)$

$Signal_3 = \cos(2\pi \cdot 220 \cdot 2^{8/12} \cdot t8)$

$Signal_4 = \cos(2\pi \cdot 220 \cdot 2^{5/12} \cdot t2)$

CamScanner ile tarandı

b)

-The closed forms written in MATLAB as:

```
s_1 = cos( 2*pi*220*(2^(10/12))* t8 );
s_2 = cos( 2*pi*220*(2^(6/12))* t2 );
s_3 = cos( 2*pi*220*(2^(8/12))* t8 );
s_4 = cos( 2*pi*220*(2^(5/12))* t2 );
```

c and d)

-With the provided lines our full MATLAB code for question 2 is:

```
%Provided frequencies and not lengths
fs = 8000;
n1 = 2;
t8 = 1/fs: 1/fs : n1/8;
t2 = 1/fs: 1/fs : n1/2;
sd = zeros(1,round(length(t8)/10));
rest = zeros(1,length(t8));

%Signals
s_1 = cos( 2*pi*220*(2^(10/12))* t8 );
s_2 = cos( 2*pi*220*(2^(6/12))* t2 );
s_3 = cos( 2*pi*220*(2^(8/12))* t8 );
s_4 = cos( 2*pi*220*(2^(5/12))* t2 );

%Signals as an array in the given form
arr = [s_1 sd s_1 sd s_1 sd s_2 sd rest sd s_3 sd s_3 sd s_3 sd s_4];

%Sound command to listen the signal
sound(arr);
```

e)

Using sound command on the array allowed me to listen the signals in the given combination which turns out to be the 5th symphony from Beethoven.