In the Name of God

University of Tehran

Electrical and Computer Engineering faculty

Signals and Systems, Fall 95

Computer Assignment #2

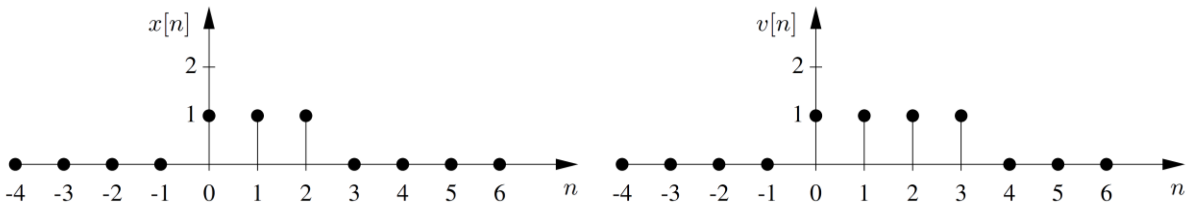
Due Date: Saturday, 29 Abaan 1395, 11:55 AM

Problem 1

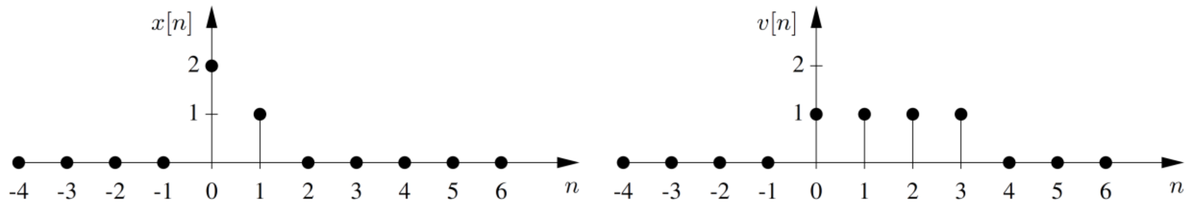
Important: Read two PDF files (6 pages) about ‘*conv*’ command in MATLAB first. Then answer the questions.

1. plot the convolution $x[n] * v[n]$ MATLAB by first computing it with ‘*conv*’ command and then plotting it with the ‘*stem*’ command.

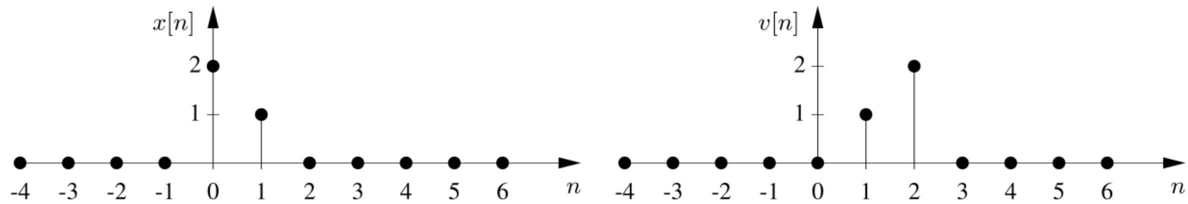
   1.a)



   1.b)

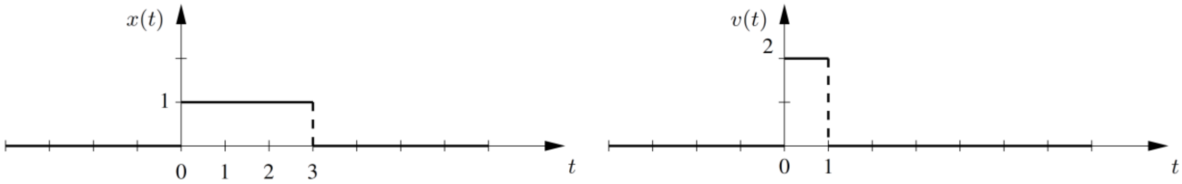


   1.c)

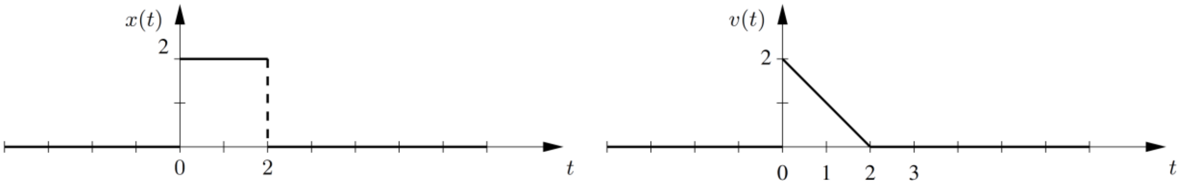


2. For the continuous-time signals $x(t)$ and $v(t)$ shown below, compute the convolution $x(t) * v(t)$ and plot the resulting signal in MATLAB.

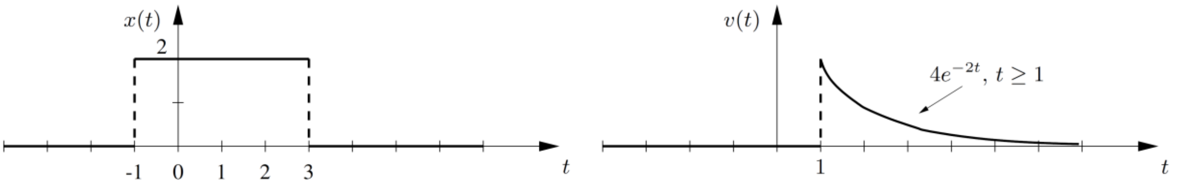
2.a)



2.b)



2.c)



- You can solve by hand (at least in easier ones) and compare to make sure you are getting the correct answer.

Problem 2

1. Write a MATLAB function which calculates the Fourier series coefficients of an arbitrary periodic signal $s(t)$.

The function takes a period (for example the [0,T]) of signal i.e., x, number of harmonics (from DC to N-th harmonic) you need i.e., N, and the period of signal i.e., T_0, and returns Fourier series coefficients (from DC till the number you asked for) in an array i.e.,a_k, and their corresponding harmonic frequency i.e., w:

$$\text{function } [a_k, w] = fourierseries\,(x, T_0, N)$$

here is an example of a code which finds complex exponential Fourier series coefficients.

```
function [a, w]=fourierseries_complex(x,T0,N)
% function fourierseries
% Computes harmonics of the Fourier series of a continuous-time signal
% symbolically
% input: periodic signal x(t), its period (T0), number of harmonics (N)
% output: harmonics a and corresponding harmonic frequency w
% use: [a, w]=fourier(x,T0,N)
syms t
```

```
t0=0;
omega0 = 2*pi/T0;
k=0:N-1,
a=int(x*exp(−j*k*omega0*t),t,0,T0)/T0;
a=eval(a);
w=k*omega0;
```

I want you to write a function, $[a_0, b_k, c_k, w] = fourierseries\_tri\,(x, T_0, N)$ which finds trigonometric Fourier series coefficients and another function which finds coefficients of Fourier Series in the cosine with phase Form, $[A_k, \theta_k, w] = fourierseries\_cosine\,(x, T_0, N)$

- Different Fourier Series Forms
  - Complex Exponential Fourier Series

$$x(t) = \sum_{k=-\infty}^{\infty} a_k\, e^{jk\Omega_0 t}, \mathrm{t} \in [t_0, t_0 + T]$$

$$a_k = \frac{1}{T}\int_{t0}^{t0+T} x(t) e^{jk\Omega_0 t}\, dt$$

  - Trigonometric Fourier Series

$$x(t) = a_0 + \sum_{k=1}^{\infty} b_k \cos(k\Omega_0 t) + \sum_{k=1}^{\infty} c \sin(k\Omega_0 t)$$

$$a_0 = \frac{1}{T}\int_{t0}^{t0+T} x(t)\, dt$$

$$b_k = \frac{2}{T}\int_{t0}^{t0+T} x(t)\cos(k\Omega_0 t)\, dt\,, k = 1,2, \ldots$$

$$c_k = \frac{2}{T}\int_{t0}^{t0+T} x(t)\sin(k\Omega_0 t)\, dt\,, k = 1,2, \ldots$$

  - Fourier Series in the Cosine with Phase Form

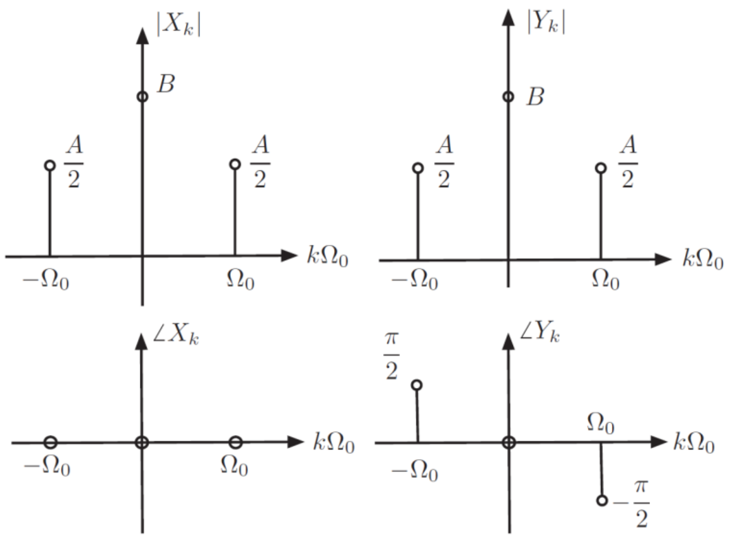
$$x(t) = a_0 + \sum_{k=1}^{\infty} A_k \cos(k\Omega_0 t + \theta_k)$$

$$a_0 = \frac{1}{T}\int_{t0}^{t0+T} x(t)\, dt$$

$$A_k = \sqrt{b_k^2 + c_k^2}, k = 1,2, \ldots$$

$$\theta_k = \begin{cases} tan^{-1}\left(-\frac{c_k}{b_k}\right) = -tan^{-1}\left(\frac{c_k}{b_k}\right), k = 1,2, \ldots, when\ b_k \geq 0 \\ \pi + tan^{-1}\left(-\frac{c_k}{b_k}\right) \qquad\qquad\qquad , k = 1,2, \ldots, when\ b_k < 0 \end{cases}$$

2. Using the previous function, *fourierseries_complex*, find the exponential Fourier series of a raised cosine signal $x(t) = B + A\cos(\Omega_0 t + \theta)$ for $\theta = 0$ and also for raised sin, $y(t) = B + A\sin(\Omega_0 t)$ (which can be obtained by letting $\theta = -\pi/2$ in $x(t)$). Then plot the line spectrum (magnitude and phase of Their complex exponential Fourier series) of $x(t)$ and $y(t)$. Bellow you can see the line spectrum of raised cosine (left) and of raised sine (right) which you have to find.



Let $B = 2$, $A = 1$, $\Omega_0 = 100$ and $\theta = 0, -\pi/2$.

Hint 1: The above code only gives the positive coefficients of complex exponential Fourier series. Use the following code to obtain negative half of it together with the positive part.

```
a=[conj(fliplr(a(2:N))) a];w=[-fliplr(w(2:N))  w];
```

Explain why this is correct.

- Use '*subplot*' command to plot the wanted signals in one figure.
- Use '*stem*' command for plotting.
- Use '*abs*' and '*angle*' to obtain magnitude and phase
- Use appropriate Labels and titles.

3. The periodic signal $x(t)$ in one period is given by

$$x(t) = \begin{Bmatrix} 1, 0 \le t \le 1 \\ 0, 1 \le t \le 2 \end{Bmatrix}$$

Use *fourierseries_complex*, *fourierseries_tri* or *fourierseries_cosine* in pervious part in new scripts you write to solve each of following parts.

3.a) Plot in 5 period the approximate signals using 5, 11 and 41 terms of the complex exponential Fourier series (you can let N=21 in given code and use Hint 1 to get 41 terms or you can change k=0:N-1 to k=-N:N in first place and let N=20 to get 41 terms and do the same for 11 and 5 terms). Plot the magnitudes and the angles of the coefficients a_k only for 41 terms. (5 plots in one figure)

Hint 2: Use the following code to obtain approximate signals

```
xx=sum(a.*exp(j*k* omega0*t));
```

3.b) Plot in 5 period the approximate signals using 5, 11 and 41 terms of the trigonometric Fourier series (you can let N=41 in your own code which returns k=0:40 , k=0 is DC and the rest 40 to get 41 terms do the same for 5 and 11). Plot The coefficients b_k and c_k only for 41 terms. (5 plots in one figure)

Hint 3: `xx=a0+sum(b.*cos(k*omega0*t))+sum(c.*sin(k*omega0*t))`

3.c) Plot in 5 period the approximate signals using 5, 11 and 41 terms of the Fourier series in the cosine with phase form. (you can let N=41 in your own code which returns k=0:40 , k=0 is DC and the rest 40 to get 41 terms do the same for 5 and 11). Plot The coefficients A_k and θ_k only for 41 terms. (5 plots in one figure)

Hint 3: `xx=a0+sum(A.*cos(n*omega0*thita))`

- Use '*subplot*' command to plot the wanted signals in one figure.
- Use '*sezplot*' command for plotting the signal.
- Use '*stem*' command for plotting the coefficients.
- Use '*abs*' and '*angle*' to obtain magnitude and phase
- Use appropriate Labels and titles.

4. Increase the number of terms to, let's say 500. Can you still see the oscillations around discontinuity points? What is the reason of these oscillations? Can we eliminate these oscillations by further increasing the terms?
5. We wish to approximate the signal $x(t)$ by a Fourier series with a finite number of terms, i.e., k=-N:N (2N+1 terms) so that it has 80% of the power of $x(t)$. Find N.

Problem 3

AM envelope detector—Consider an *envelope detector* that is used to detect the message sent in the AM system shown in the examples. The envelopedetector as a system is composed of two cascaded systems: one which computes the absolute value of the input (implemented with ideal diodes), and a second that low-pass filters its input (implemented with an *RC* circuit). The following is an implementation of these operations in the discrete time so we can use numeric MATLAB. Let the input to the envelope detector be

$$x(t) = [p(t) + P]\cos(\Omega_0 t).$$

Use MATLAB to solve numerically this problem.

1. Consider first $p(t) = 20[u(t) - u(t-40)] - 10[u(t-40) - u(t-60)]$ and let $\Omega_0 = 2\pi$, $P = 1.1|\min(p(t))|$. Generate the signals $p(t)$ and $x(t)$ for $0 \le t \le 100$ with and interval of $T_s = 0.01$.
2. Consider then the subsystem that computes the absolute value of the input $x(t)$. Plot $y(t) = |x(t)|$.
3. Compute the low-pass filtered signal $(h * y)(t)$ by using an *RC* circuit with impulse response $h(t) = e^{-0.8\,t}u(t)$. To implement the convolution use '*conv*' function multiplied by T_s. Plot together the message signal $p(t)$, the modulated signal $x(t)$, the absolute value $y(t)$, and the envelope $z(t) = (h * y)(t) - P$. Does this envelope look like $p(t)$?
4. Consider the message signal $p(t) = 2\cos(0.2\pi t)$, $\Omega_0 = 10\pi$ and $P = |\min(p(t))|$and repeat the process. Scale the signal to get the original $p(t)$.

Problem 4

Optional. ***This part includes some interesting features of signals which can be discovered by MATLAB. You may try them. There is additional Points for those who solve it.***

Windowing and music sounds—In the computer generation of musical sounds, pure tones need to be windowed to make them more interesting. Windowing mimics the way a musician would approach the generation of a certain sound. Increasing the richness of the harmonic frequencies is the result of the windowing as we will see in this problem. Consider the generation of a musical note with frequencies around $f_A = 880\ Hz$. Assume our "musician" while playing this note uses three strokes corresponding to a window $w_1(t) = r(t) - r(t - T_1) - r(t - T_2) + r(t - T_0)$, so that the resulting sound would be the multiplication, or windowing, of a pure sinusoid $\cos(2\pi f_A t)$ by a periodic signal $w(t)$ with $w_1(t)$ a period that repeats every $T_0 = 5T$ where T is the period of the sinusoid. Let $T_1 = T_0/4$, and $T_2 = 3T_0/4$.

a) Analytically determine the Fourier series of the window $w(t)$ and plot its line spectrum using MATLAB. Indicate how you would choose the number of harmonics needed to obtain a good approximation to $w(t)$.

b) Use the modulation or the convolution properties of the Fourier series to obtain the coefficients of the product $s(t) = \cos(2\pi f_A t)\, w(t)$. Use MATLAB to plot the line spectrum of this periodic signal and again determine how many harmonic frequencies you would need to obtain a good approximation to $s(t)$.

c) The line spectrum of the pure tone $p(t) = \cos(2\pi f_A t)$ only displays one harmonic, the one corresponding to the $f_A = 880\ Hz$ frequency, how many more harmonics does $s(t)$ have? To listen to the richness in harmonics use the function *sound* to play the sinusoid $p(t)$ and $s(t)$ (use $F_s = 2\times 880\ Hz$ to play both).

d) Consider a combination of notes in a certain scale, for instance let $p(t) = \sin(2\pi\times 440t) + \sin(2\pi\times 550t) + \sin(2\pi\times 660t)$. Use the same windowing $w(t)$, and let $s(t) = p(t)w(t)$. Use to plot $p(t)$ and $s(t)$ and to compute and plot their corresponding line spectra. Use '*sound*' to play $p(nT_s)$ and $s(nT_s)$ using $F_s = 1000\ Hz$.