HOMEWORK 6
CPE381

Canvas: `hw06`

Due: 4th November 2024, 11:59 PM
65 points

You are allowed to use a generative model-based AI tool for your assignment. However, you must submit an accompanying reflection report on how you use the AI tool, what was the query for the tool, and how it improved your understanding of the subject. You must also add your thoughts on how you would tackle the assignment if there was no such tool available. Failure to provide a reflection report for every single assignment where an AI tool was used may result in a penalty and subsequent actions will be taken in line with plagiarism policy.

Submission instruction:

Upload a .pdf on Canvas with the format {firstname.lastname}_cpe381_hw06.pdf. If there is a programming assignment, then you should include your source code along with your PDF files in a zip file firstname.lastname}_cpe381_hw06.zip. If a plot is being asked, your PDF file must also contain plots generated by your MATLAB code. Your submission must contain your name, and UAH Charger ID or the UAH email address. Please number your pages as well.

---

1 Bandlimited (10 points)

State whether the following functions and band-limited or not. If they are band-limited, determine the Nyquist frequency **(5 points each)**.

1. $\textbf{rect}\left(\frac{t}{3}\right)$

   where

   $$\textbf{rect}\left(\frac{t}{a}\right) = \begin{cases} 1, & |t| \le \frac{a}{2}, \\ 0, & \text{otherwise} \end{cases}$$

   for a positive number a.

   Hint: sketch the function.

2. $\text{sinc}\left(\frac{x}{3}\right)$

Solution

1. Its Fourier transform is

$$
\begin{aligned}
X(\Omega) &= \int_{-\infty}^{\infty} x(t)e^{-j\Omega t}dt \\
&= \int_{-\frac{3}{2}}^{0} 1 \cdot e^{-j\Omega t}dt + \int_{0}^{\frac{3}{2}} e^{-j\Omega t}dt \\
&= \frac{e^{-j\Omega t}}{-j\Omega}\bigg|_{-\frac{3}{2}}^{0} + \frac{e^{-j\Omega t}}{-j\Omega}\bigg|_{0}^{\frac{3}{2}} \\
&= \frac{1 - e^{j\Omega\frac{3}{2}}}{-j\Omega} + \frac{e^{-j\Omega\frac{3}{2}} - 1}{-j\Omega} \\
&= \frac{-e^{j\Omega\frac{3}{2}} + e^{-j\Omega\frac{3}{2}}}{-j\Omega} = \frac{e^{j\Omega\frac{3}{2}} - e^{-j\Omega\frac{3}{2}}}{j\Omega} = \frac{e^{j\Omega\frac{3}{2}} - e^{-j\Omega\frac{3}{2}}}{2j\Omega} \cdot 2 \\
&= \frac{2\sin(\Omega\frac{3}{2})}{\Omega} \cdot \frac{\frac{3}{2}}{\frac{3}{2}} = 2\tfrac{3}{2}\text{sinc}(\Omega\tfrac{3}{2}) = 3\text{sinc}(\Omega\tfrac{3}{2}) \quad \text{Here, } \text{sinc}(x) = \frac{\sin(x)}{x}
\end{aligned}
$$

and the $3\text{sinc}(\ )$ function has non-zero values out to infinity, so the function is not band-limited.

2. Its Fourier transform has $2\pi[u(\Omega + \frac{1}{3}) - u(\Omega - \frac{1}{3})]$ which is zero for $|\Omega| > \dfrac{1}{3}$, hence the function is band-limited and the Nyquist frequency is $\Omega_{\max} = \dfrac{1}{3}$.

2 Nyquist Sampling Rate Condition (25 points)

1. Consider the signal $x(t) = 2\cos(2\pi t + \pi/4)$, $0 - \infty < t < \infty$, determine if it is bandlimited. **(5 points)**

2. Use $T_s = 0.1, 0.2, 0.3$, and 2 seconds/sample as sampling periods, and for each of these write down their sampled version of the signal $x_s(t)$, and then find out whether the Nyquist sampling rate condition is satisfied and if the sampled signal looks like the original signal or not. **(10 points)**

3. Write a MATLAB code to plot continuous signal $x(t)$. Vary time from $t = 0$ to $t = 3.0, s$ Choose the sampling time of $T_s = 0.10.2, 0.3$, and 2 seconds/sample and create 4 subplots in a single plot, showing original $x(t)$ and corresponding sampled signal $x_s(t)$ for each of the four $T_s s$. Make sure you plot the sampled signal using a stem plot.

   (10 points)

Solution

1. Since $x(t)$ only has frequency 2π, it is bandlimited with $\Omega_{\text{max}} = 2\pi$ rad/s.

2. For any T_s, the sampled signal can be written as

$$x_s(t) = \sum_n 2\cos\left(2\pi n T_s + \frac{\pi}{4}\right)\delta(t - nT_s)$$

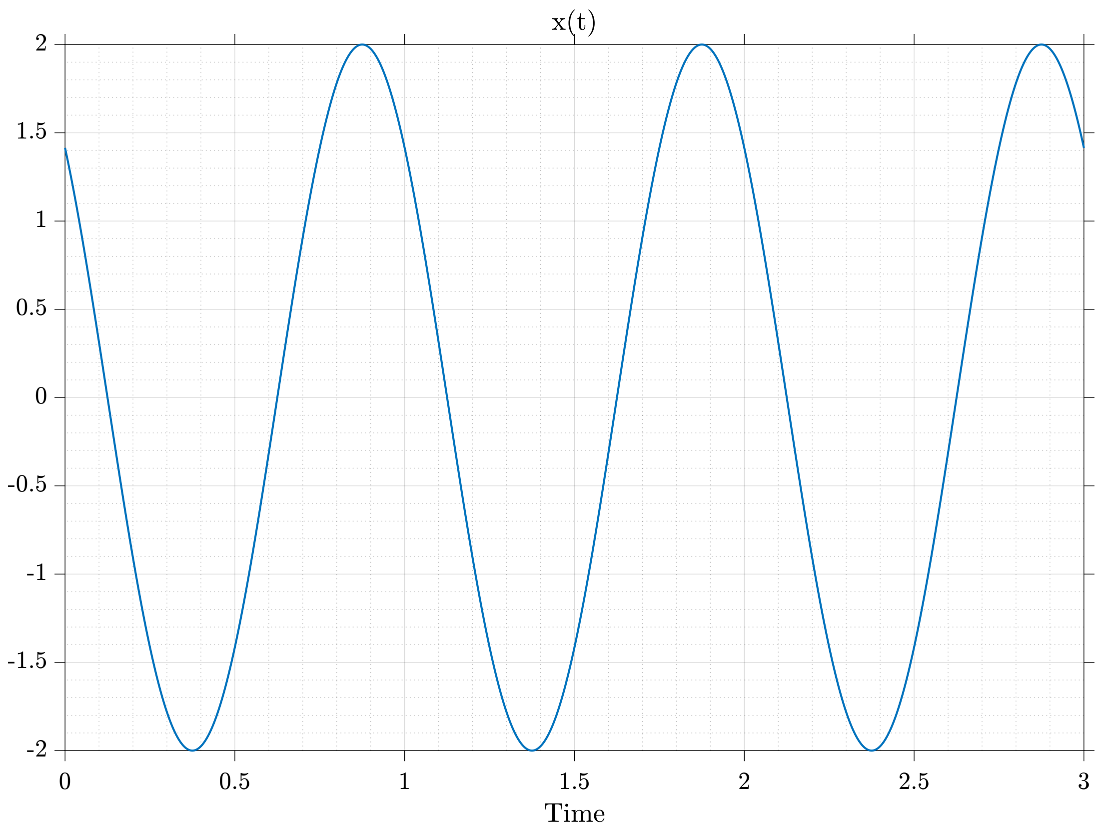
Using $T_s = 0.1$ s/sample, the sampling frequency is $\Omega_s = 2\pi/0.1 = 20\pi > 2\Omega_{\text{max}}$ which satisfies the Nyquist sampling rate condition.

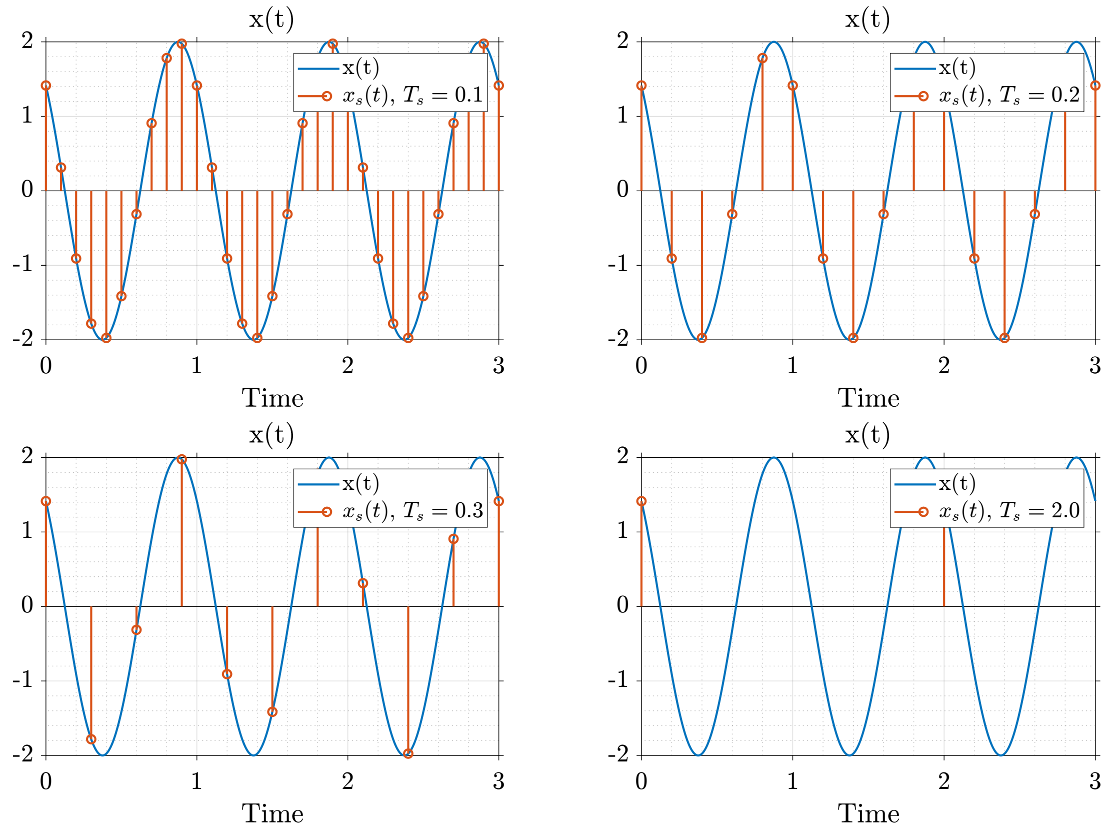
For $T_s = 0.2$ s/sample, the sampling frequency is $\Omega_s = 2\pi/0.2 = 10\pi > 2\Omega_{\text{max}}$ which satisfies the Nyquist sampling rate condition.

For $T_s = 0.3$ s/sample, the sampling frequency is $\Omega_s = 2\pi/0.3 = 6.67\pi > 2\Omega_{\text{max}}$ which satisfies the Nyquist sampling rate condition.

For $T_s = 2$ s/sample, the sampling frequency is $\Omega_s = 2\pi/2 = \pi < 2\Omega_{\text{max}}$ which **doesn't** satisfy the Nyquist sampling rate condition.

3.





Code:

```
%(C) Rahul Bhadani
close all;
t = 0:0.001:3;
x = 2*cos(2.*pi.*t + pi./4);
f = figure(1);
f.Position = [265, 377, 1290, 849];

plot(t, x,'LineWidth', 1.5);
grid on;
xlabel('Time','Interpreter', 'latex');
hold on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', ...
    [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';
```

```
title('x(t)', 'Interpreter','latex');
set(gca, 'FontSize', 18);
exportgraphics(f, '../figures/HW06_Q2_FA24_CPE381_FIG1.pdf');

%%

f = figure(2);
f.Position = [265, 377, 1290, 849];

subplot(2, 2,1)
T_s = 0.1;
[ts1, xs1] = sampled_cosine(T_s);
plot(t, x,'LineWidth', 1.5, 'DisplayName','x(t)');
hold on;
stem(ts1, xs1,'LineWidth', 1.5, ...
    'DisplayName','$x_s(t)$, $T_s = 0.1$');
grid on;
xlabel('Time','Interpreter', 'latex');
hold on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', ...
    [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';
title('x(t)', 'Interpreter','latex');
set(gca, 'FontSize', 18);
legend('Interpreter','latex');

subplot(2, 2,2)
T_s = 0.2;
[ts1, xs1] = sampled_cosine(T_s);
plot(t, x,'LineWidth', 1.5, 'DisplayName','x(t)');
hold on;
stem(ts1, xs1,'LineWidth', 1.5, ...
    'DisplayName','$x_s(t)$, $T_s = 0.2$');
grid on;
xlabel('Time','Interpreter', 'latex');
hold on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], ...
    'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
```

```
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';
title('x(t)', 'Interpreter','latex');
set(gca, 'FontSize', 18);
legend('Interpreter','latex');

subplot(2, 2,3)
T_s = 0.3;
[ts1, xs1] = sampled_cosine(T_s);
plot(t, x,'LineWidth', 1.5, 'DisplayName','x(t)');
hold on;
stem(ts1, xs1,'LineWidth', 1.5, ...
    'DisplayName','$x_s(t)$, $T_s = 0.3$');
grid on;
xlabel('Time','Interpreter', 'latex');
hold on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], ...
    'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';
title('x(t)', 'Interpreter','latex');
set(gca, 'FontSize', 18);
legend('Interpreter','latex');

subplot(2, 2,4)
T_s = 2.0;
[ts1, xs1] = sampled_cosine(T_s);
plot(t, x,'LineWidth', 1.5, 'DisplayName','x(t)');
hold on;
stem(ts1, xs1,'LineWidth', 1.5, ...
    'DisplayName','$x_s(t)$, $T_s = 2.0$');
grid on;
xlabel('Time','Interpreter', 'latex');
hold on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], ...
    'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
```

```
yaxis.TickLabelInterpreter = 'latex';
title('x(t)', 'Interpreter','latex');
set(gca, 'FontSize', 18);
legend('Interpreter','latex');
exportgraphics(f, '../figures/HW06_Q2_FA24_CPE381_FIG2.pdf');

% Function to quantize signal
function [t, x] = sampled_cosine(T_s)
    t = 0:T_s:3.0;
    x = 2*cos(2.*pi.*t + pi./4);
end
```

3 Draw Again (10 points)

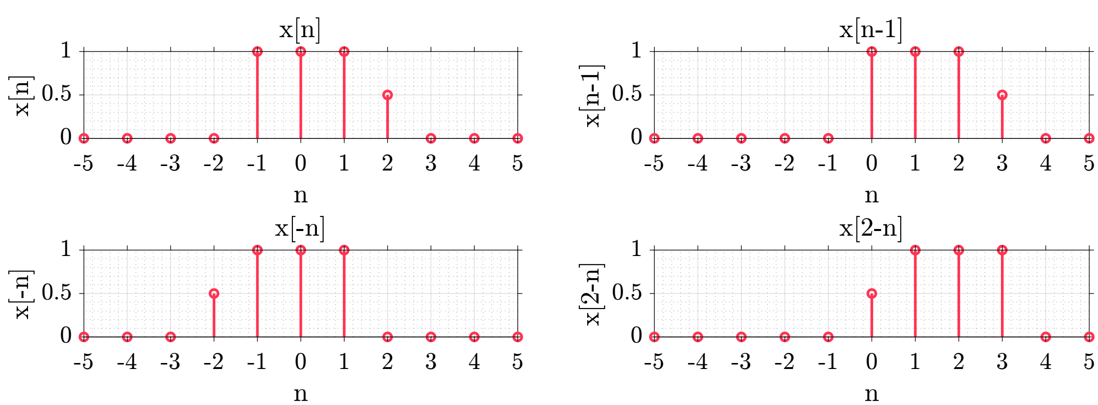
For the discrete-time signals,

$$x[n] = \begin{cases} 1, & n = -1, 0, 1 \\ 0.5, & n = 2 \\ 0, & \text{otherwise} \end{cases}$$

Sketch and label properly the following signals:

1. $x[n-1]$, $x[-n]$, $x[2-n]$. **(3 points)**
2. The even components $x_e[n]$ of $x[n]$. **(3.5 points)**
3. The odd components $x_o[n]$ of $x[n]$. **(3.5 points)**

Solution

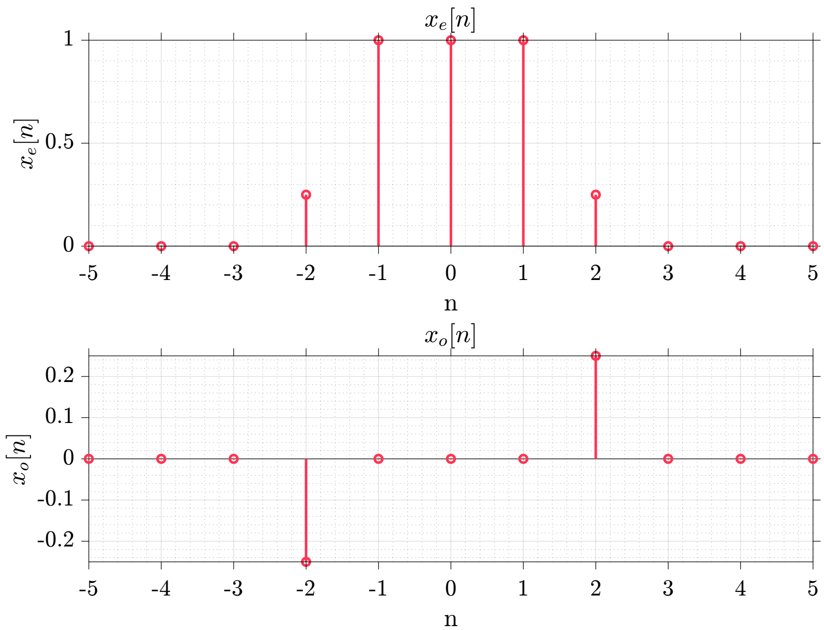
1. 

2. Even function:

$$x_e[n] = 0.5(x[n] + x[-n])$$
$$= \begin{cases} 1, & n = -1, 0, 1 \\ 0.25, & n = -2, 2 \\ 0, & \text{otherwise} \end{cases}$$

3. Odd function:

$$x_o[n] = 0.5(x[n] - x[-n])$$
$$= \begin{cases} -0.25, & n = -2 \\ 0.25, & n = 2 \\ 0, & \text{otherwise} \end{cases}$$




```
% Define the signal x[n] using an anonymous function
x = @(n) (n == -1 | n == 0 | n == 1) + 0.5 * (n == 2);

% Evaluate the signal at specific values of n
n = -5:1:5;
x_values = x(n);

% Plot the acceleration data
fig = figure;
fig.Position = [554, 456, 1300, 420];
subplot(2,2,1)
stem(n, x_values, 'LineWidth',2,'color', '#ff3453');
title('x[n]', 'Interpreter','latex');
grid on;
```

```
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
ylabel('x[n]', 'Interpreter','latex');
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
set(gca, 'XTick', n);

x_values = x(n-1);
subplot(2,2,2)
stem(n, x_values, 'LineWidth',2,'color', '#ff3453');
title('x[n-1]', 'Interpreter','latex');
grid on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
ylabel('x[n-1]', 'Interpreter','latex');
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
set(gca, 'XTick', n);

x_values = x(-n);
subplot(2,2,3)
stem(n, x_values, 'LineWidth',2,'color', '#ff3453');
title('x[-n]', 'Interpreter','latex');
grid on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
```

```
ylabel('x[-n]', 'Interpreter','latex');
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
set(gca, 'XTick', n);

x_values = x(2-n);
subplot(2,2,4)
stem(n, x_values, 'LineWidth',2,'color', '#ff3453');
title('x[2-n]', 'Interpreter','latex');
grid on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
ylabel('x[2-n]', 'Interpreter','latex');
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
set(gca, 'XTick', n);

exportgraphics(fig, ...
'../figures/CPE381_FA24_HW6_Q3_xn.pdf');

fig = figure;
fig.Position = [206   190   969   706];
subplot(2,1,1)

x_values = 0.5.*( x(n) + x(-n) );

stem(n, x_values, 'LineWidth',2,'color', '#ff3453');
title('$x_e[n]$', 'Interpreter','latex');
grid on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
ylabel('$x_e[n]$', 'Interpreter','latex');
```

```
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
set(gca, 'XTick', n);

subplot(2,1,2)


x_values = 0.5.*( x(n) - x(-n) );


stem(n, x_values, 'LineWidth',2,'color', '#ff3453');
title('$x_o[n]$', 'Interpreter','latex');
grid on;
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
ylabel('$x_o[n]$', 'Interpreter','latex');
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
set(gca, 'XTick', n);


exportgraphics(fig, ...
'../figures/CPE381_FA24_HW6_Q3_xn_even_odd.pdf');
```

4 Impulse Response (20 points)

1. Determine the impulse response $h[n]$ of an LTI system represented by the difference equation

$$y[n] = -0.5y[n-1] + x[n]$$

where $x[n]$ is the input , $y[n]$ is the output and the initial conditions are zero. **(5 points)**

2. Find $y[n]$ analytically when the input is

$$x[n] = \begin{cases} 1, & 0 \leq n \leq 2 \\ 0, & \text{otherwise} \end{cases}$$

3. Compute $y[n]$ using `for` loop by implementing in MATLAB and plot for n = 1:20. Use the analytical solution obtained from the previous part of this question to plot separately and compare your result. **(10 points)**

Solution

1. Impulse response $h[n]$ is the output $y[n]$ when $x[n] = \delta[n]$.

   Hence,
   $$h[n] = -0.5h[n-1] + \delta[n]$$

   To get a closed form, considering a system to be a causal system, $h[-1] = 0$, and $\delta[n] = 1$ for $n = 0$, and 0 for $n \neq 0$, we can write

   $$\begin{aligned} h[0] &= -0.5h[-1] + \delta[n] \\ &= 1 = u[0] \\ h[1] &= -0.5h[0] = -0.5 = -0.5u[1] \\ h[2] &= -0.5h[1] = -0.5 \times -0.5u[2] \\ h[3] &= -0.5h[2] = -0.5 \times -0.5 \times -0.5u[3] \\ &\vdots \\ h[n] &= (-1)^n 0.5^n u[n] \end{aligned}$$

2. $x[n]$ can be written as

   $x[n] = \delta[0] + \delta[1] + \delta[2] = \delta[n] + \delta[n-1] + \delta[n-2]$

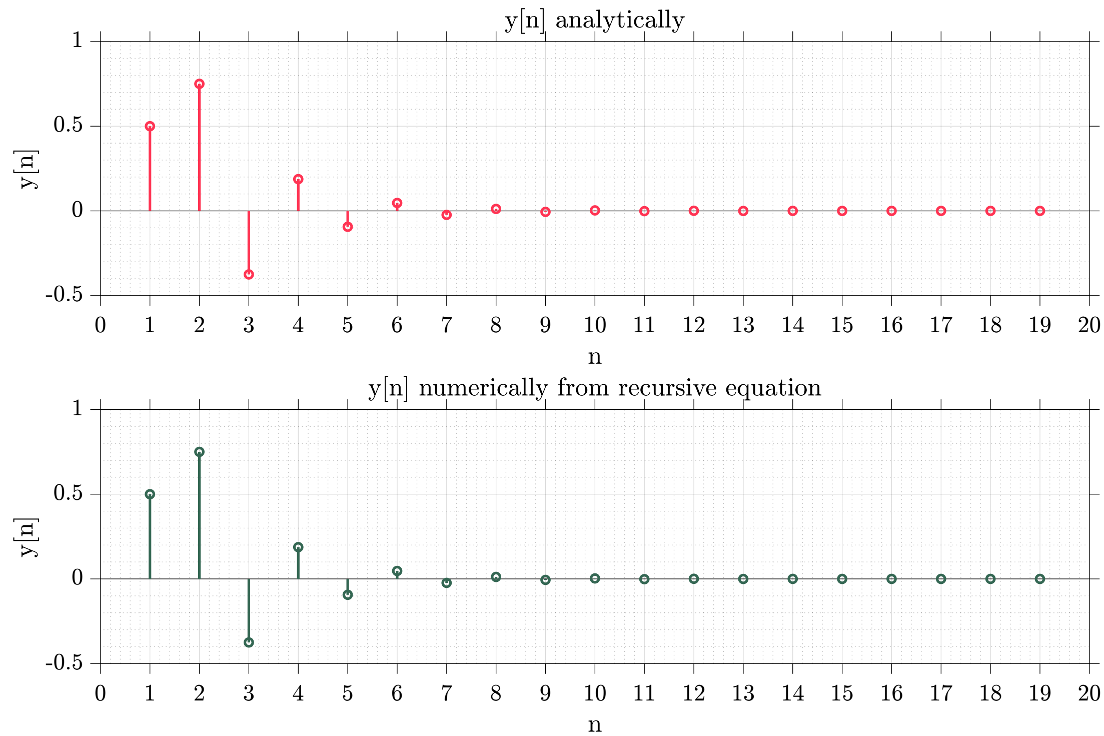
   then,

   by linearity,

   $y[n] = (-1)^n 0.5^n u[n] + (-1)^n 0.5^{n-1} u[n-1] + (-1)^n 0.5^{n-2} u[n-2]$

   How did we arrive at this:

   - If $\delta[n]$ (which is basically $\delta[0]$) is the input, then the output is $h[n]$, and since the system we study is an LTI system, if the input is $\delta[1]$ or $\delta[n-1]$, the output is $h[n-1]$. Additionally, due to the additivity property, if the input is $\delta[n] + \delta[n-1] + \delta[n-2]$, then then the output is $h[n] + h[n-1] + h[n-2]$



3. **Code:**

```
% Define the signal x[n] using an anonymous function
delta = @(n) (n==0);
u = @(n) (n>=0);

h = @(n)  (-1).^n.*0.5.^n .*u(n);

y = @(n) h(n) + h(n-1) + h(n-2);

% Evaluate the signal at specific values of n
n = 0:1:20;
y_values = y(n);

% Plot the acceleration data
fig = figure;
fig.Position = [549          66          1310          798];
subplot(2,1,1)
stem(n(2:20), y_values(2:20), 'LineWidth',2,'color', '#ff3453');
title('y[n] analytically', 'Interpreter','latex');
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
```

```
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
ylabel('y[n]', 'Interpreter','latex');
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
set(gca, 'XTick', n);

%%

% Define input x[n]
n = 0:1:20;  % Time index
x = zeros(size(n));
x(n >= 0 & n <= 2) = 1;

% Initialize output y[n]
y_recur = zeros(size(n));

% Set initial condition (y[-1] = 0)
y0 = 0;

% Implement difference equation using for loop
for i = 1:length(n)
    if i == 1
        y_recur(i) = x(i);  % y[0] = x[0]
    else
        y_recur(i) = -0.5*y_recur(i-1) + x(i);
    end
end

subplot(2,1,2)
stem(n(2:20), y_recur(2:20), 'LineWidth',2,'color', '#346753');
title('y[n] numerically from recursive equation', 'Interpreter','latex');
grid on;
grid minor;
set(gca, 'XColor', [0, 0, 0], 'YColor', [0, 0, 0], 'TickDir', 'out');
xaxis = get(gca, 'XAxis');
xaxis.TickLabelInterpreter = 'latex';
yaxis = get(gca, 'YAxis');
yaxis.TickLabelInterpreter = 'latex';

xlabel('n', 'Interpreter','latex');
ylabel('y[n]', 'Interpreter','latex');
set(gca, 'FontSize', 18);
% Display all ticks on the x-axis
```

```
set(gca, 'XTick', n);

exportgraphics(fig, ...
'../figures/CPE381_FA24_HW6_Q4.pdf');
```