Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

EEE 321 Lab 5

Part 1

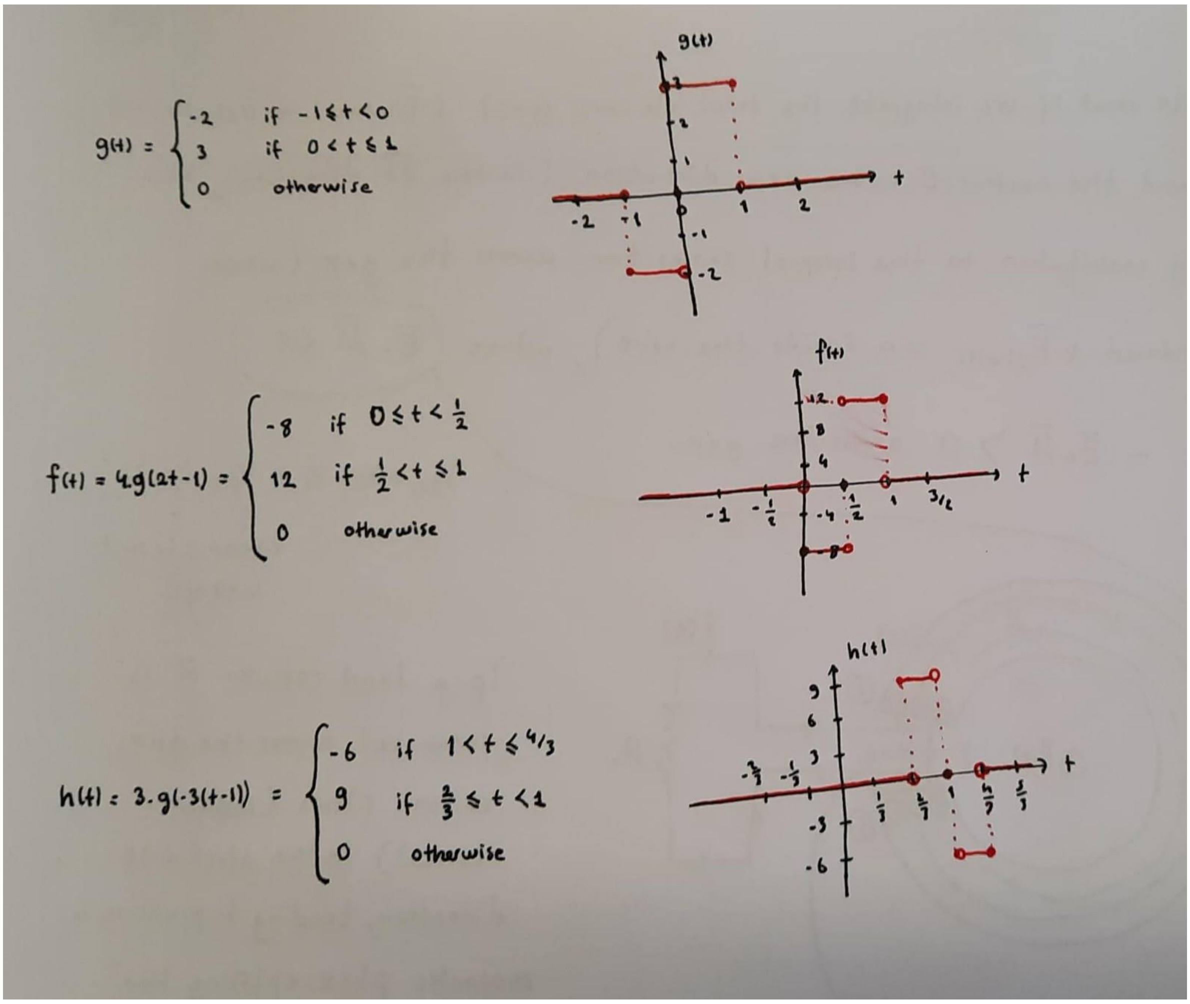

Figure 1 – f(t) and h(t) graphs

While I generate the new graphs, first I pay attention to determine the side points. Since, starting and ending points are the points that have only changes on the graph, after determined these points, rest is only assembling the points with lines.

Shannon-Nyquist criteria offers that the sampling frequency should be higher than twice of the highest frequency in the signal. As perfect square waves have infinitely many frequencies in it. So, the sample frequency cannot be determined as high as twice of highest frequency. As we cannot obtain useful sample frequency, reconstructed signals will be just approximation of the original one.

Let highest frequency in the signal be x and sample frequency be f_s,

$$f_s > 2x$$

So, it is impossible to reconstruct this signal with interpolation functions.

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

Part 2

$$x_R(t) = \sum_{n'=-\infty}^{\infty} \bar{x}[n'] . p(t - n'T_s)$$

$$x_R(nT_s) = \sum_{n'=-\infty}^{\infty} \bar{x}[n'] . p(nT_s - n'T_s)$$

since $p(kT_s) = 0$ for all non-zero integers k, the only term that survives in the summation is when $n' = n$

$x_R(nT_s) = \bar{x}[n] . p(nT_s - nT_s)$ $\quad p(0) = 1$, so we get

$x_R(nT_s) = \bar{x}[n] . 1 = \bar{x}[n]$

a) $p_Z(0) = \text{rect}\left(\frac{0}{T_s}\right) = \text{rect}(0) = 1$

$p_L(0) = \text{tri}\left(\frac{0}{T_s}\right) = \text{tri}(0) = 1$

$p_I(0) = \text{sinc}\left(\frac{0}{T_s}\right) = \text{sinc}(0) = 1$

b)
$$\left.\begin{aligned} p_Z(kT_s) &= \text{rect}\left(\frac{kT_s}{T_s}\right) = \text{rect}(k) = 0 \quad \text{for } k \neq 0 \\ p_L(kT_s) &= \text{tri}\left(\frac{kT_s}{T_s}\right) = \text{tri}(k) = 0 \quad \text{for } k \neq 0 \\ p_I(kT_s) &= \text{sinc}\left(\frac{kT_s}{T_s}\right) = \frac{\sin(\pi k)}{\pi k} = 0 \quad \text{for } k \neq 0 \end{aligned}\right\} \ k \text{ is non-zero integer}$$

c) $p_Z(0) = 1$ and $\text{rect}(k) = 0$ for $k \in \mathbb{Z}^+, \mathbb{Z}^-$ consistent ✓

$p_L(0) = 1$ and $\text{tri}(k) = 0$ for $k \in \mathbb{Z}^+, \mathbb{Z}^-$ consistent ✓

$p_Z(0) = 1$ and $p_I(kT_s) = 0$ for $k \in \mathbb{Z}^+, \mathbb{Z}^-$ consistent ✓

Figure 2 – Answers of questions

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

Part 3

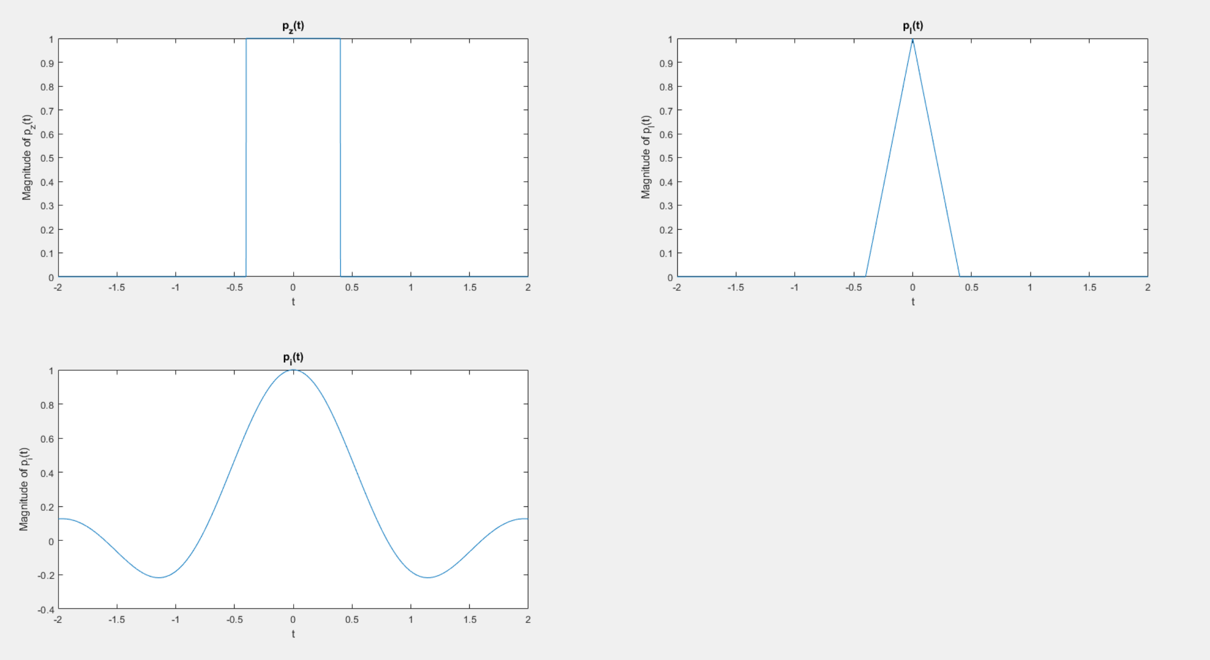

Figure 3 – Interpolation functions

Matlab Code

```
ID=22002075;
dur=mod(ID,7);
Ts=dur/5;

subplot(2,2,1)
P = generateInterp(0,Ts,dur);
title('p_z(t)')
xlabel('t')
ylabel('Magnitude of p_z(t)')

subplot(2,2,2)
P = generateInterp(1,Ts,dur);
title('p_l(t)')
xlabel('t')
ylabel('Magnitude of p_l(t)')

subplot(2,2,3)
P = generateInterp(2,Ts,dur);
title('p_i(t)')
xlabel('t')
ylabel('Magnitude of p_i(t)')
```

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

```
function P=generateInterp(type,Ts,dur)
    Ts1=Ts/500;
    t= (-dur/2):Ts1:(dur/2);
    P=zeros(size(t));

%%%% type 0 is for zero order hold interpolation

    if type == 0
        for i=1:size(t,2)
            if abs(t(i)) <= Ts/2
                P(i)=1;
            else
                P(i)=0;
            end
        end

%%%% type 1 is for linear interpolation

    elseif type ==1
        for i=1:size(t,2)
            if t(i)<=Ts/2 && t(i)>0
                P(i)=-((2*t(i))/Ts)+1;
            elseif t(i)>=-Ts/2 && t(i)<=0
                P(i)=((2*t(i))/Ts)+1;
            else
                P(i)=0;
            end
        end

%%%% type 2 is for ideal bandlimited interpolation

    else
         for i=1:size(t,2)
             if t(i)==0
                 P(i) = 1;
             Else
                 P(i)=(sin(pi*t(i)/Ts))/((pi*t(i))/Ts);
             end
         end
    end
end
```

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

Part 4

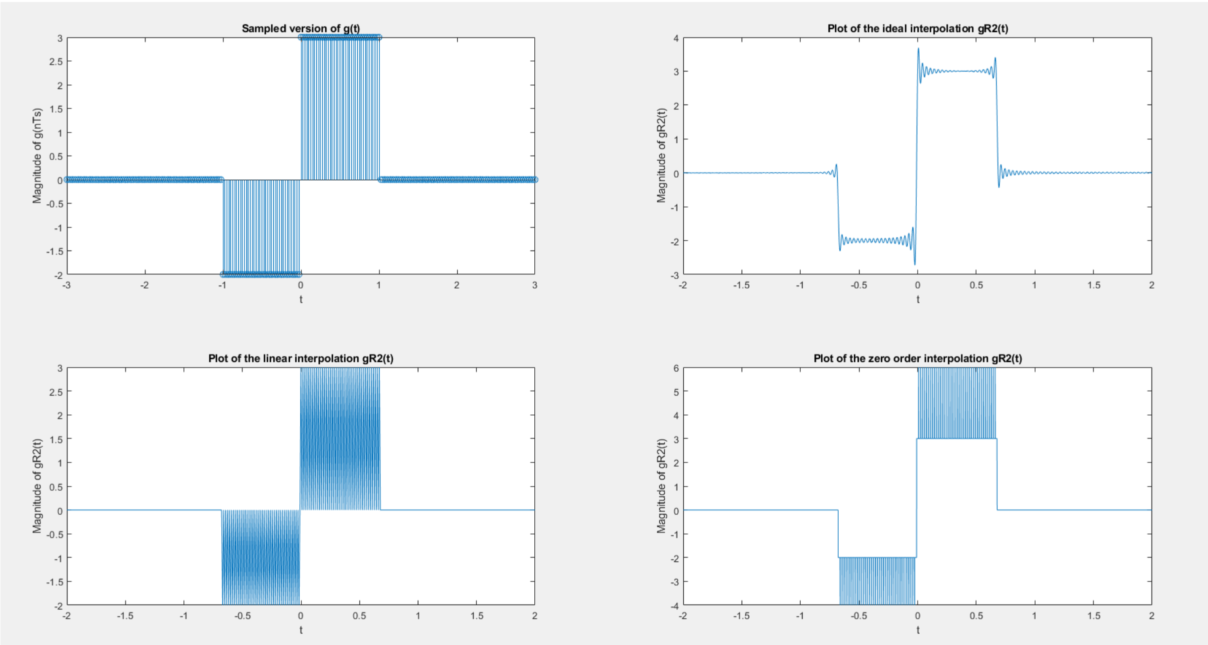

Figure 3 – Sampled and interpolated graphs

As shown in the figures, the interpolated signals are not exactly the same with the original because square waves have infinitely many frequencies in it.

Matlab Code

```
Ts = 0.1;
dur = 8;
t = -dur/2:Ts:dur/2;

ID=22002075;
duration=mod(ID,7);
TS=duration/5;

g = zeros(size(t));
g(1+2/Ts:1+3/Ts) = -2;
g(1+3/Ts:1+4/Ts) = 3;
h = generateInterp(1,Ts,dur)

gR0 = DtoA(0,Ts,dur,g,h);
gR1 = DtoA(1,Ts,dur,g,h);
gR2 = DtoA(2,Ts,dur,g,h);
```

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

```
x_axis2 = linspace(-2,2,length(gR2));
x_axis1 = linspace(-2,2,length(gR1));
x_axis0 = linspace(-2,2,length(gR0));


subplot(2,2,1)
plot(t,g);
title('Sampled version of g(t)')
xlabel('t')
ylabel('Magnitude of g(nTs)')

subplot(2,2,2)
plot(x_axis0,gR0);
title('Plot of the zero order interpolation, gR2(t)')
xlabel('t')
ylabel('Magnitude of gR2(t)')

subplot(2,2,3)
plot(x_axis1,gR1);
title('Plot of the linear interpolation, gR2(t)')
xlabel('t')
ylabel('Magnitude of gR2(t)')

subplot(2,2,4)
plot(x_axis2,gR2);
title('Plot of the ideal interpolation, gR2(t)')
xlabel('t')
ylabel('Magnitude of gR2(t)')


function [xR] = DtoA(type,Ts,dur,Xn,h)
    ratio = 500;
    sizexR = 1+size(h);
    xR = zeros(sizexR);

    for i = 1:length(Xn)
    xR(1+rate*(i-1):rate*(i-1)+length(h)) = Xn(i)*h + xR(1+ratio *(i-1):ratio *(i-
1)+length(h));
    end
    xR = xR((rate/2)*length(Xn)+1:end-(rate/2)*length(Xn));
end
```

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

Part 5

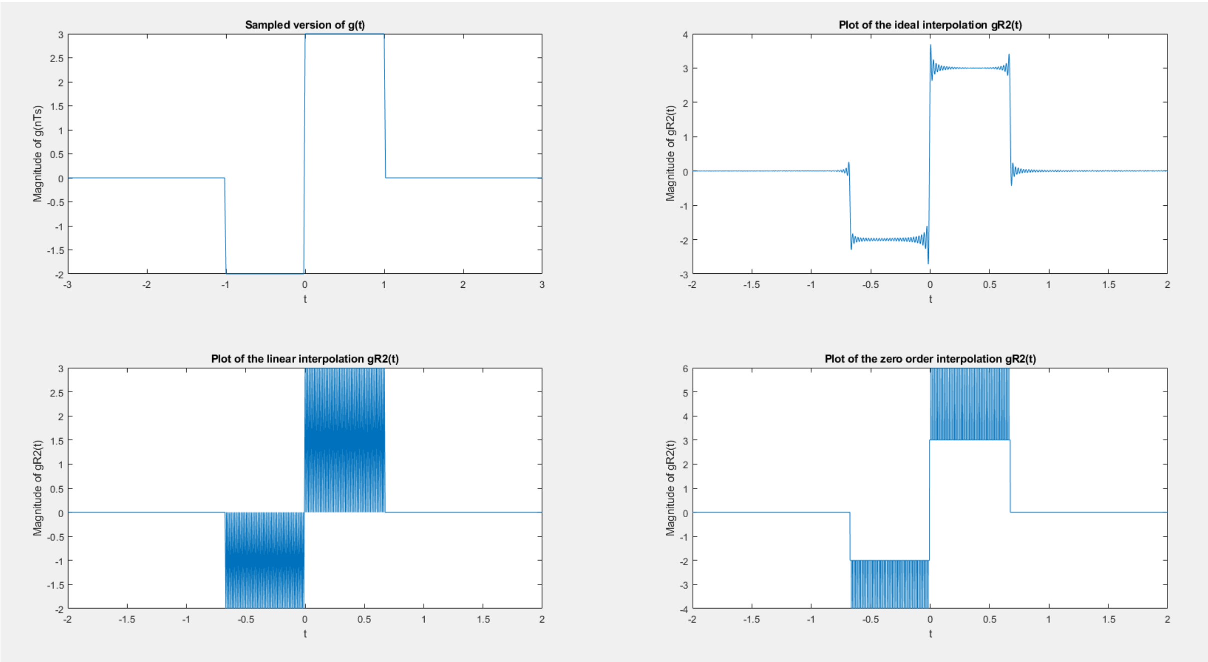

Figure 4 – Comparison of reconstructed signals

While Ts increase, the reconstructed signals become less successful. Because the signal has infinitely many frequencies in it and if we decrease the period of sampling, we can take more successful reconstructed signals.

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

Part 6

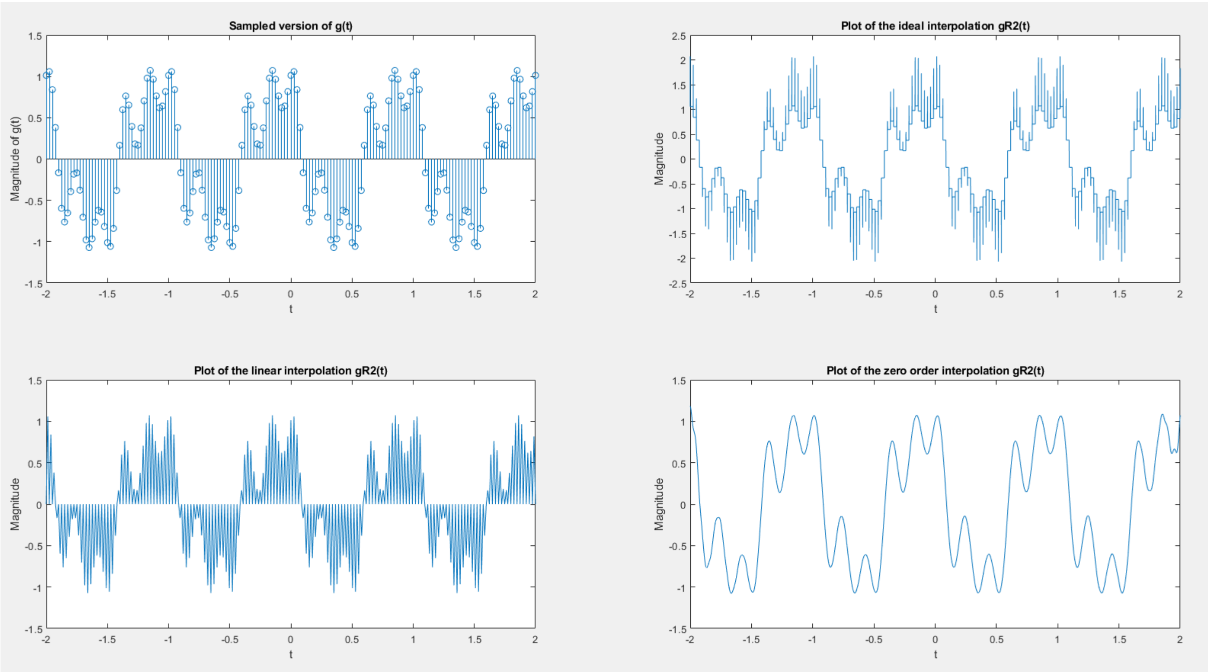

Figure 5 – Ts = 0.005 x (D7 + 1)

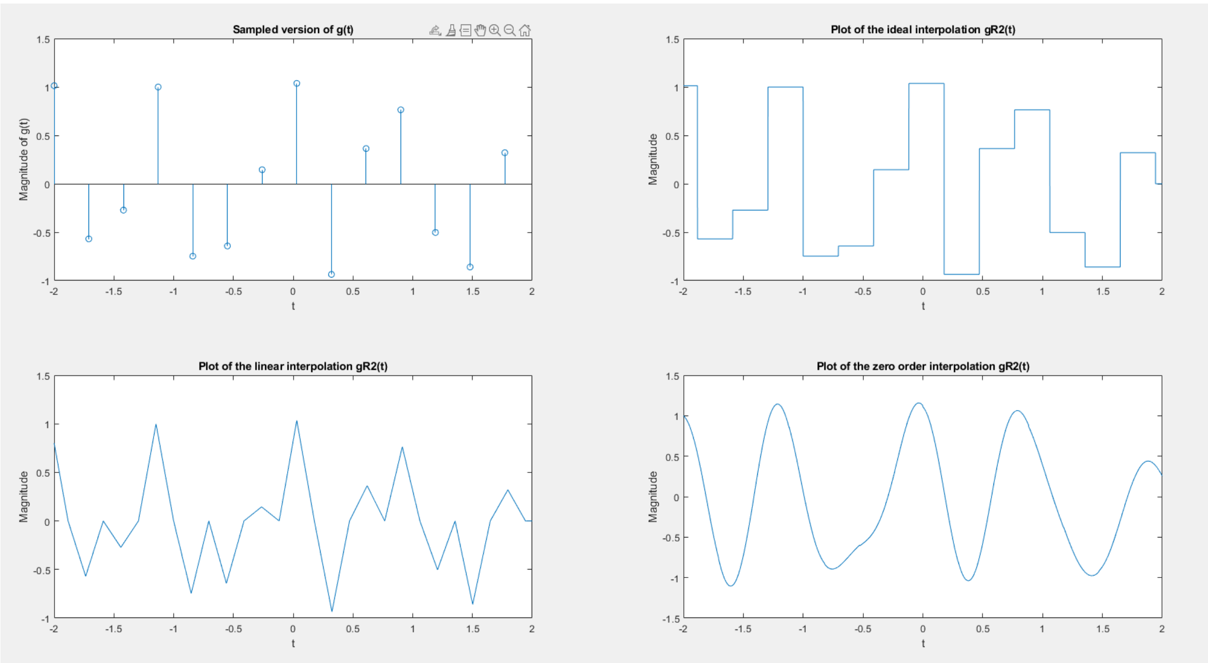

Figure 6 – Ts = 0.25 + (0.01 x D7)

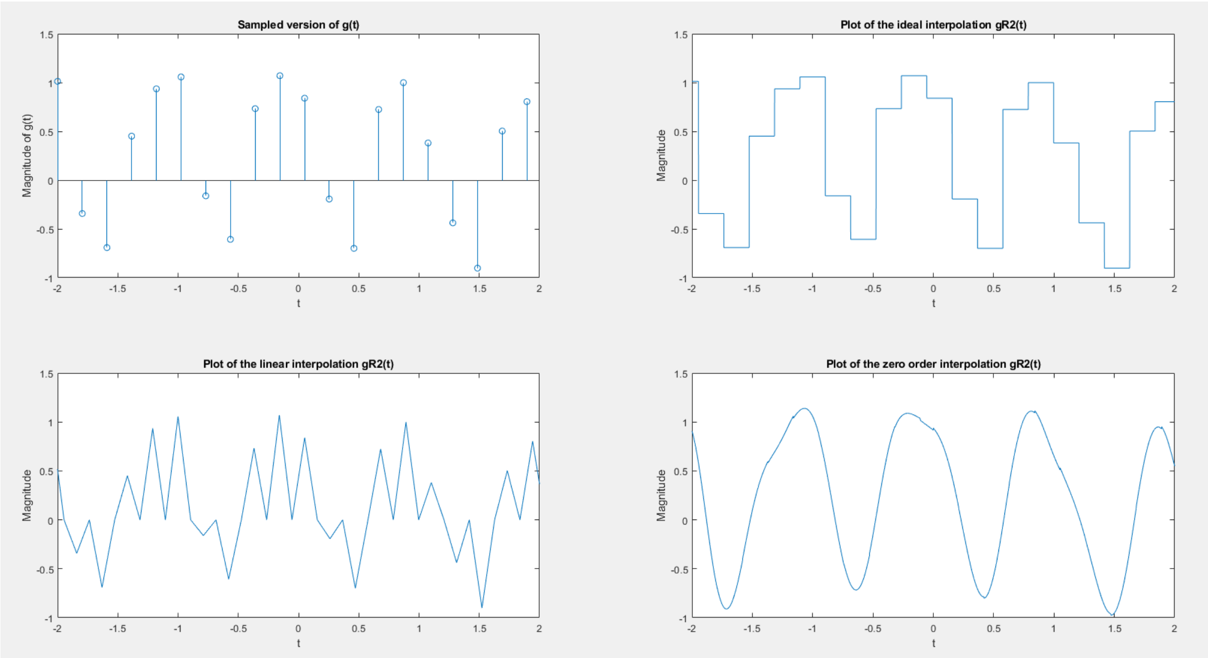

Figure 7 - *Ts* = 0.18+0.005(*D*7+1)

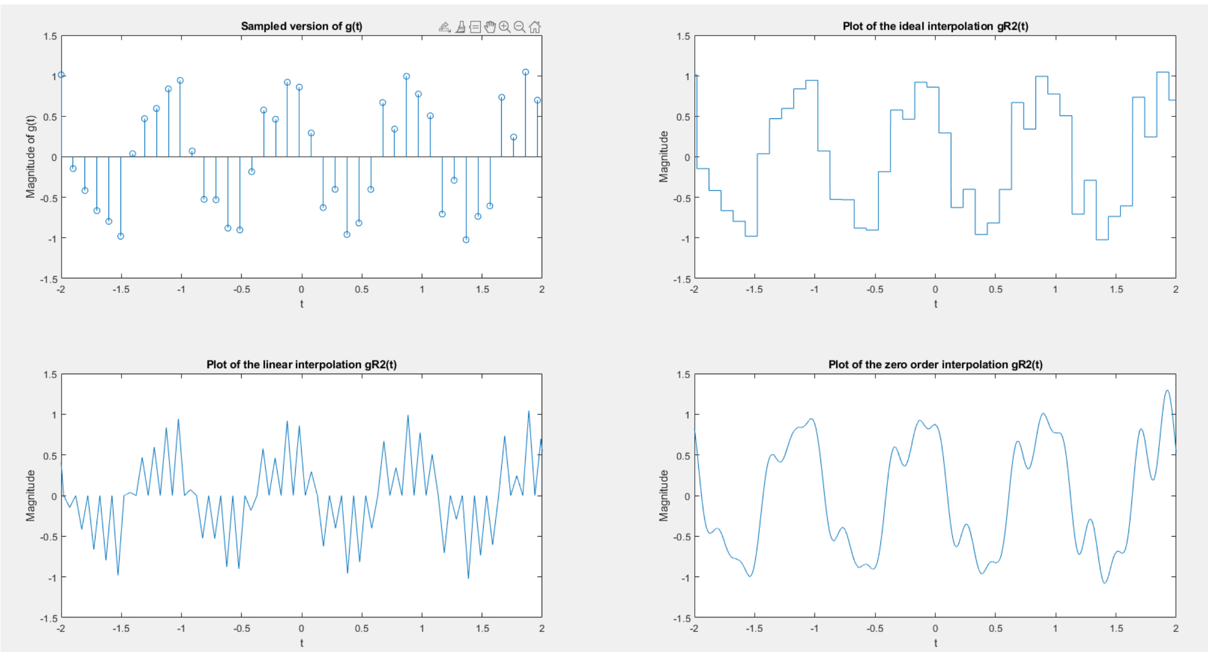

Figure 8 – Ts = 0.099

Satılmış Furkan Kahraman
22002075
24/11/2023
Report 5

Most successful one is ideal bandlimited interpolation because it has already sin component in it, it is very successful to imitate our original signal. When the sample frequency is higher than highest frequency the signal of interpolation can not differentiate the original one. Below the highest frequency, no graphs are not resembled to the original one.

Since there are 3 different period are consisted in main signal. If the Ts's interval is in the between the period of these frequencies the graph of ideal bandlimited interpolator is always changed. When our Ts lower than the half of the smallest period of the signal, then the signal becomes steady.

Matlab Code

```
D=22002075;
D7=mod(D,7);

%%% Ts = it depends current values
ts = -2:Ts:2;
t =-2:Ts/500: 2;
xs=zeros(size(ts));
x =zeros(size(t));

for i=1:size(ts,2)
    if abs(ts(i))<=2
        xs(i)=0.25*cos(2*pi*3*ts(i)+(pi/8))+0.4*cos(2*pi*5*ts(i)-
1.2)+0.9*cos(2*pi*ts(i)+(pi/4));
    else
        xs(i)=0;
    end
end

subplot(2,2,1)
plot(t,x)
stem(ts,xs)
title('Sampled version of g(t)')
xlabel('t')
ylabel('Magnitude')

subplot(2,2,2)
x_axis = linspace(-2,2,length(xR1));
xR1 = DtoA(0,Ts,dur,xs);

plot(x_axis,xR1);


title ('Plot of the ideal interpolation gR2(t))

xlabel('t')

ylabel('Magnitude')

subplot(2,2,3)

x_axis = linspace(-2,2,length(xR1));

xR1 = DtoA(1,Ts,dur,xs);
```

```
plot(x_axis,xR1);

title('Plot of the linear interpolation gR2(t)')

xlabel('t')

ylabel('Magnitude')

subplot(2,2,4)

x_axis = linspace(-2,2,length(xR1));

xR1 = DtoA(2,Ts,dur,xs);

plot(x_axis,xR1);

title('Plot of the zero order interpolation gR2(t)')

xlabel('t')

ylabel('Magnitude')
```