به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

تمرین کامپیوتری ۱

پردازش سیگنال های دیجیتال (DSP)

دکتر بدیعی

عرفان پناهی ۸۱۰۱۹۸۳۶۹

نیمسال اول ۰۱-۱۴۰۰

فهرســت:

*** فایل مربوط به این تمرین کامپیوتری با نام **CA1_MatlabFile_810198369.mlx** پیوست شده است.

چکیده : هدف از تمرین کامپیوتری ۱

هدف از انجام این تمرین کامپیوتری طراحی برخی بلوک های پردازش سیگنال گسسته زمان و بررسی بعضی کاربرد های آن است. در بخش اول بلوک های ابتدایی پردازش سیگنال را شبیه سازی می‌کنیم. در بخش دوم و چهارم با استفاده از بلوک های ساخته شده در بخش اول، یکی از کاربرد های تغییر نرخ نمونه برداری و نصف کننده فرکانسی را بررسی می‌کنیم. در بخش سوم تبدیل فوریه یک سیگنال متناوب را تحلیل می‌کنیم. در بخش پنجم تغییر فاز یک فایل صوتی را مورد بررسی قرار میدهیم. در نهایت و در بخش ششم نیز با استفاده از مطالب آموخته شده در درس یک معادله تفاضلی را به دو روش متفاوت حل می‌کنیم.

بخش اول : شبیه سازی بلوک های پردازش سیگنال

سوال ۱) تابع DTFT : تصویر ۱-۱ ، کد مربوط به این تابع را نشان میدهد. تابع DTFTfast سرعت بیشتری در محاسبه تبدیل فوریه دارد اما تعداد عناصر ورودی آن محدود میباشد. **(در سراسر بخش اول سیگنال های پیوسته با $f_s = 1000$ نمونه برداری شده است)**

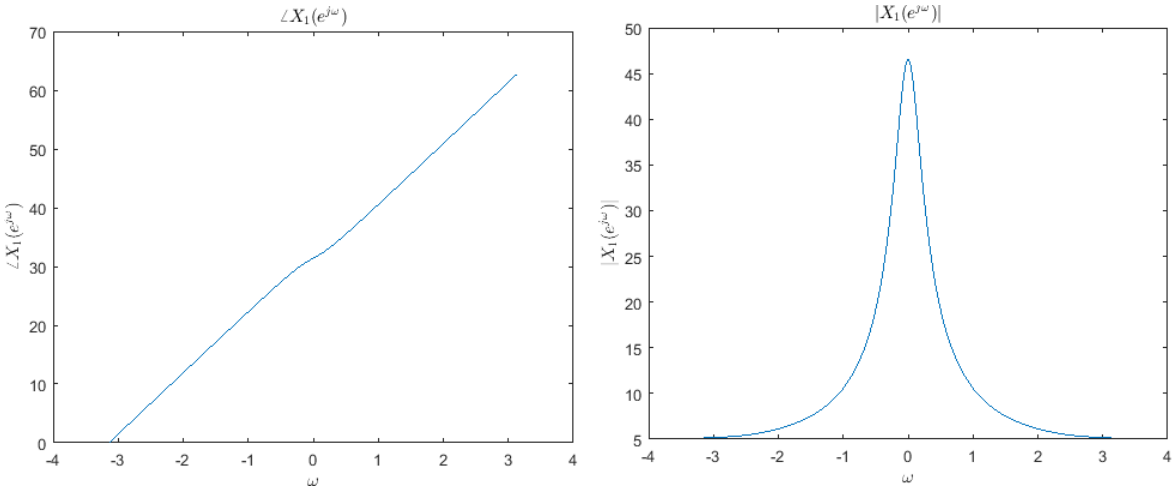
```
function Xejw = DTFT (x,n,w)
    Xejw = zeros(1,length(w));
    for i = 1:length(w)
        Xejw(i) = sum(x.*exp(-1i*w(i)*n));
    end
end

function Xejw = DTFTfast (x,n,w)
    Xejw = x * exp(-1i*n'*w);
end
```

تصویر ۱-۱ : کد **تابع DTFT**

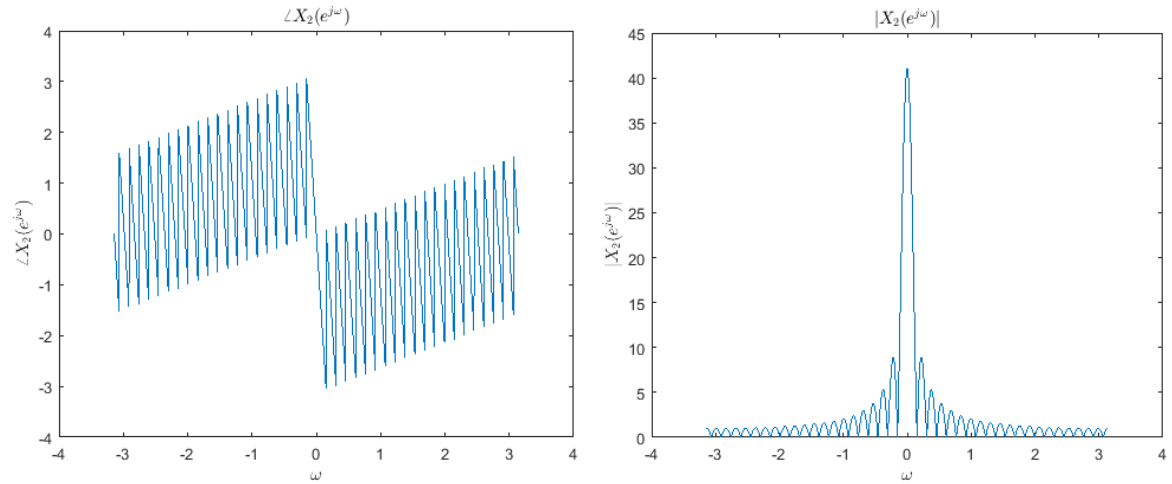
تصویر ۱-۲ اندازه و فاز تبدیل فوریه سیگنال $x_1[n]$ را نشان میدهد:

*** تمامی نمودار های فاز تبدیل فوریه در این پروژه با استفاده از دستور phase() رسم شده است.



تصویر ۱-۲ : اندازه و فاز تبدیل فوریه سیگنال $x_1[n] = (0.8)^n, -10 \le n \le 20$

تصویر ۱-۳ اندازه و فاز تبدیل فوریه سیگنال $x_2[n]$ را نشان میدهد:



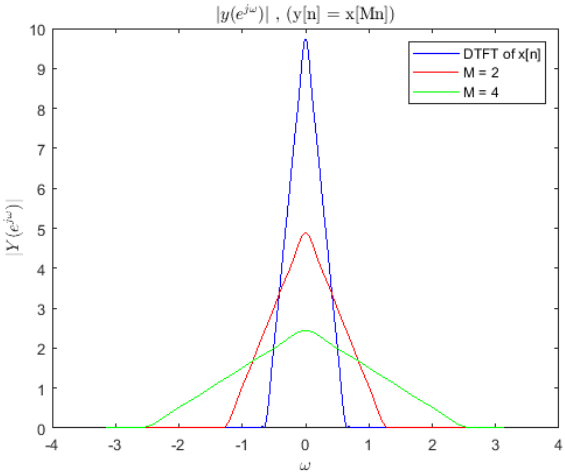
تصویر ۱-۳ : اندازه و فاز تبدیل فوریه سیگنال $x_2[n] = 1{,}0 \le n \le 40$

سوال ۲) تابع Compressor : تصویر ۱-۴ ، کد مربوط به این تابع را نشان میدهد.

```
function [y,ny] = Compressor (x,nx,M)
    for j = 1:M
        if (mod(nx(j),M) == 0)
            i = j;
        end
    end
    y = x(i:M:length(x));
    ny = nx(i:M:length(nx))/M;
end
```

تصویر ۱-۴ : کد **تابع Compressor**

تصویر ۱-۵ اندازه تبدیل فوریه سیگنال $x_d[n]$ و $y[n] = x_d[Mn]$ به ازای $M\epsilon\{2,4\}$ را نشان میدهد:



تصویر ۱-۵ : **اندازه تبدیل فوریه سیگنال $x_d[n]$ و $y[n] = x_d[Mn]$ به ازای $M\epsilon\{2,4\}$**

همانطور که انتظار میرفت، طبق روابط آموخته شده در درس انتظار داشتیم بهرهٔ اندازه تبدیل $\frac{1}{M}$ برابر شده و همچنین محور فرکانسی نیز با ضریب M منبسط می‌شود. روابط زیر این موضوع را تأیید می‌کند:

$$y[n] = x_d[Mn] \rightarrow Y(e^{j\omega}) = \frac{1}{M}\sum_{l=0}^{M-1} X_d(e^{j(\frac{\omega-2\pi l}{M})}$$

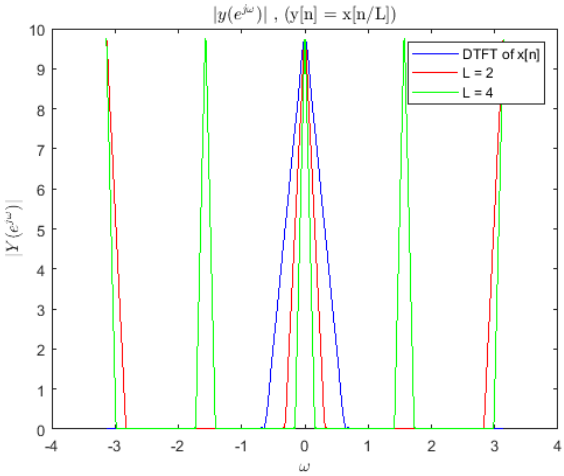
سوال ۳) تابع Compressor : تصویر ۱-۶ ، کد مربوط به این تابع را نشان میدهد.

```
function [y,ny] = Expander (x,nx,L)
    i = 1:L:(L*length(nx));
    y(i) = x;
    ny = (L*nx(1)):(L*nx(length(nx)));
end
```

تصویر ۱-۶ : کد **تابع Expander**

تصویر ۱-۷ اندازه تبدیل فوریه سیگنال $x_d[n]$ و $y[n]$ به ازای $L\epsilon\{2,4\}$ را نشان میدهد:

$$y[n] = \begin{cases} x\left[\frac{n}{L}\right], n = 0, \pm L, \pm 2L, \dots \\ 0 \quad\quad\quad , \ otherwise \end{cases}$$



تصویر ۱-۷ : **اندازه تبدیل فوریه سیگنال $x_d[n]$ و $y[n]$ به ازای $L\epsilon\{2,4\}$**

همانطور که انتظار میرفت، طبق روابط آموخته شده در درس انتظار داشتیم بهرهٔ اندازه تبدیل تغییر نکرده و همچنین محور فرکانسی نیز با ضریب L منقبض می‌شود. روابط زیر این موضوع را تأیید می‌کند:

$$Y(e^{j\omega}) = X_d(e^{j\omega L})$$

سوال ۴) قسمت الف : برای بازیابی سیگنال پیوسته با استفاده از یک بلوک بازکننده تعداد سمپل های حذف شده در نمونه برداری را اضافه می‌کنیم. سیگنال خروجی بازکننده در سمپل های اضافه شده مقدار صفر دارد. برای مقدار دهی به این سمپل ها از درونیاب استفاده می‌کنیم.

قسمت ب : درونیاب ایده آل در حقیقت یک فیلتر پایین گذر با بهرهٔ L است که با تبدیل فوریه معکوس میتوانیم پاسخ ضربه این فیلتر را بدست آوریم:

$$H_{ideal}(e^{j\omega}) = L\ , |\omega| \le \frac{\pi}{L} \rightarrow\ h_{idaael}[n] = sinc\left(\frac{n}{L}\right)$$

$$\rightarrow x_i[n] = x_e[n] * h_{ideal}[n] \rightarrow x_i[n] = \sum_{k=-\infty}^{+\infty} x_e[n] sinc(k - \frac{n}{L})$$

درونیاب خطی عملی روی سیگنال باز شده انجام میدهد تا نمونه ها به طور خطی به همدیگر وصل شوند.

$$h_{linear}[n] = \begin{cases} 1 - \frac{|n|}{L} \ , & |n| \le L \\ 0\ , & otherwise \end{cases}$$

$$\rightarrow x_i[n] = x_e[n] * h_{linear}[n] \rightarrow x_i[n] = \sum_{k=n-L+1}^{n+L-1} x_e[n](1 - \frac{|n|}{L})$$

درونیاب اسپلاین مکعبی بصورت زیر عمل می‌کند:

$$\tilde{h}_i[n] = \begin{cases} (a+2)|n/L|^3 - (a+3)|n/L|^2 + 1 & 0 \le n \le L \\ a|n/L|^3 - 5|n/L|^2 + 8a|n/L| - 4a & L \le n \le 2L \\ 0 & \text{otherwise} \end{cases}$$

$$\rightarrow x_i[n] = x_e[n] * \hat{h_i}[n]$$

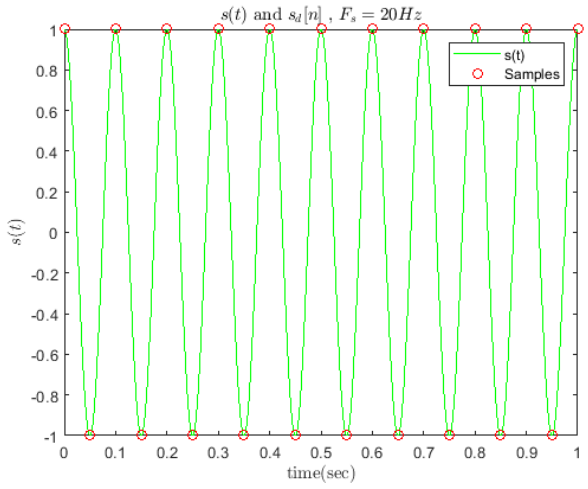
قسمت ج : با توجه به اینکه امکان ساخت سیگنال $sinc$ در حقیقت وجود ندارد (زیرا باید تا بینهایت ادامه داشته باشد)، پس نمیتوان فیلتر کاملاً ایده آل در حقیقت ساخت. البته میتوان فیلتر را بسته به نیاز به ایده آل نزدیک کرد اما هزینه بر است.

قسمت د : تابع Interpolate : کد مربوط به این تابع در تصویر ۱-۸ نشان داده شده است.

```
function [y] = Interpolate (x,n,L,fs,mode)
    n1 = n(1:L:length(n))/L;
    xc = x(1:L:length(x));
    t = n / (fs*L);
    if mode == 1
        y = xc * sinc(ones(length(n1),1)*n/L - n1'*ones(1,length(n)));
    elseif mode == 2
        x_lin = [zeros(1,L-1) x zeros(1,L-1)];
        n_lin = [-L+1:0 n length(n):length(n)+L-1];
        y = zeros(1,length(n));
        for i = n
            y(i+1) = sum(x_lin((i+1):(i+2*L-1)).*(1-abs(i-n_lin((i+1):(i+2*L-1)))/L));
        end
    elseif mode == 3
        y = spline(n1/fs,xc,t);
    else
        'invalid mode'
    end
end
```

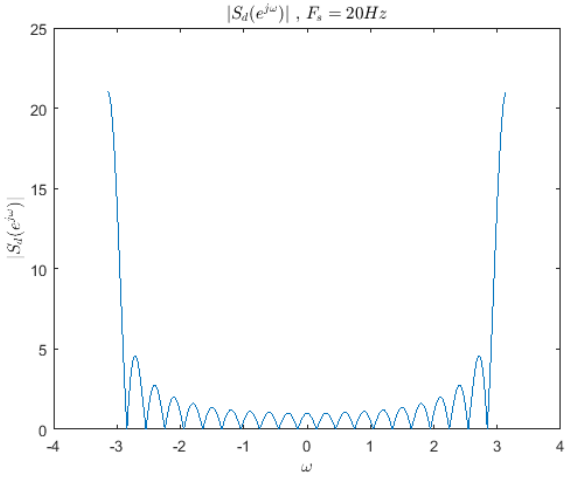
تصویر ۱-۸ : **کد مربوط به تابع Interpolate**

قسمت ه : تصویر ۱-۹ سیگنال $s(t)$ با به همراه نمونه ها را شنان میدهد:



تصویر ۱-۹ : **سیگنال $s(t)$ با به همراه نمونه ها**

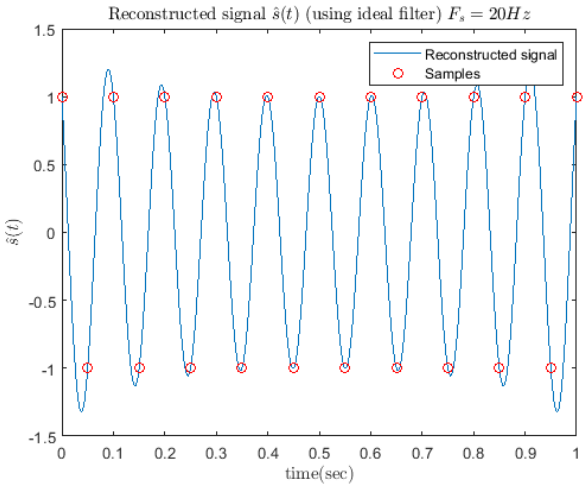
تصویر ۱-۱۰ پاسخ فرکانسی سیگنال گسسته زمان $s_d[n]$ را نشان می‌دهد:



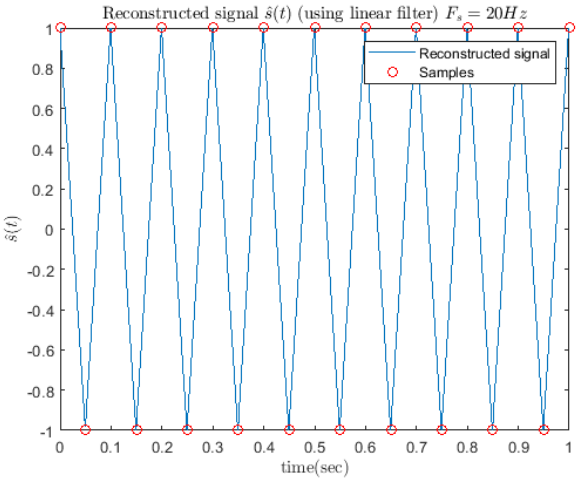
تصویر ۱-۱۰ : **پاسخ فرکانسی سیگنال $s_d[n]$**

با توجه به اینکه سیگنال پیوسته با نرخ $f_{s_{CT}} = 1000Hz$ تعریف شده است و نرخ نمونه برداری $F_s = 20Hz$ است پس باید بین هر دو نمونه گسسته $\frac{f_{s_{CT}}}{F_s} = \frac{1000}{20} = 50$ سمپل تعریف کنیم تا سیگنال را آماده پیوسته سازی کنیم. به این منظور با استفاده از $L = 50$ سیگنال را آماده درونیابی میکنیم.

تصویر ۱-۱۱ ، ۱-۱۲ و ۱-۱۳ سیگنال های بازیابی شده $\hat{s}(t)$ با روش های به ترتیب ایده آل ، خطی و اسپلاین مکعبی را نشان می‌دهد.

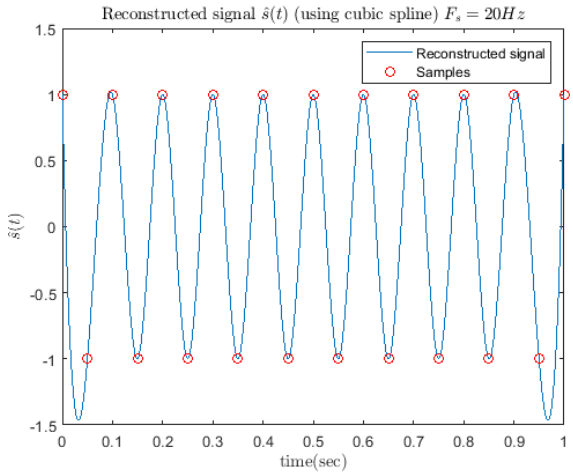


تصویر ۱-۱۱ : **سیگنال بازیابی شده با روش ایده آل**



تصویر ۱-۱۲ : **سیگنال بازیابی شده با روش خطی**

تصویر ۱-۱۴ میانگین خطای مربعات روش های بازیابی را نشان میدهد. همانطور که مشاهده می‌شود ، در این بازیابی درونیابی خطی میانگین مرعات خطای کمتری نسبت به دو درونیاب دیگر دارد. در بیشتر اوقات زمانی که تعداد نمونه های گسسته بسیار کمتر از کل باشد، تقریب خطی برای بازیابی مناسب تر است.



تصویر ۱-۱۳ : **سیگنال بازیابی شده با روش اسپلاین مکعبی**

Mean Squared Error:

```
% Fs = 20Hz
```

1. mode = 1 (ideal filter)

```
Er20_ideal = immse(s_hat1_20,s)
```

```
Er20_ideal = 0.1125
```

2. mode = 2 (linear filter)

```
Er20_linear = immse(s_hat2_20,s)
```

```
Er20_linear = 0.0243
```

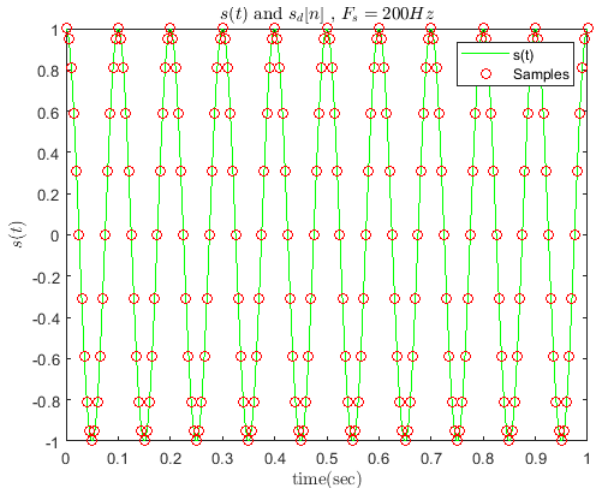
3. mode = 3 (cubic spline)

```
Er20_spline = immse(s_hat3_20,s)
```

```
Er20_spline = 0.1132
```

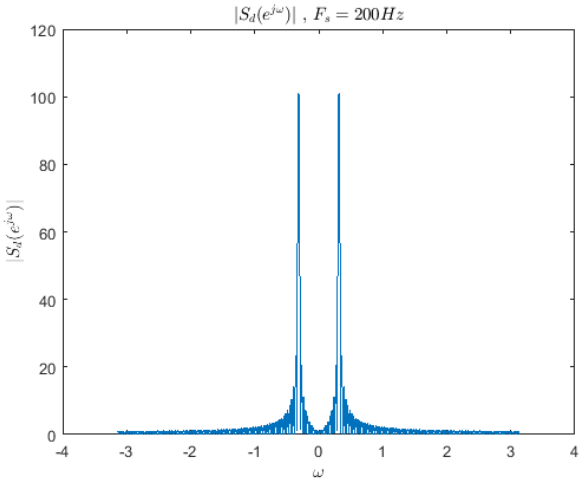
تصویر ۱-۱۴ : **میانگین خطای مربعات برای بازیابی سیگنال با $F_s = 20Hz$**

قسمت و : تصویر ۱-۱۵ سیگنال $s(t)$ با به همراه نمونه ها را برای $F_s = 200Hz$ نشان میدهد:



تصویر ۱-۱۵ : **سیگنال $s(t)$ با به همراه نمونه ها با $F_s = 20Hz$**

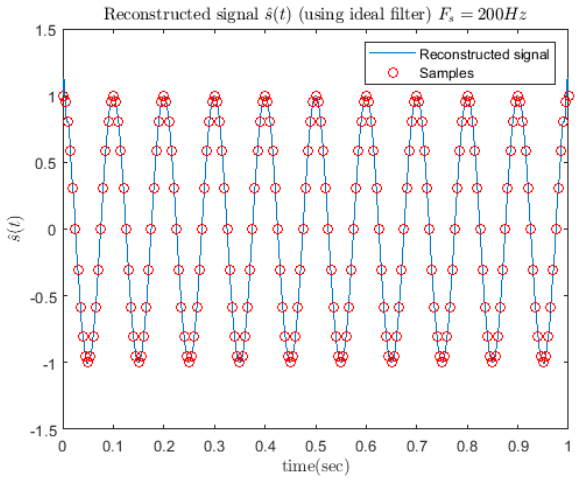
تصویر ۱-۱۰ پاسخ فرکانسی سیگنال گسسته زمان $s_d[n]$ را نشان می‌دهد:



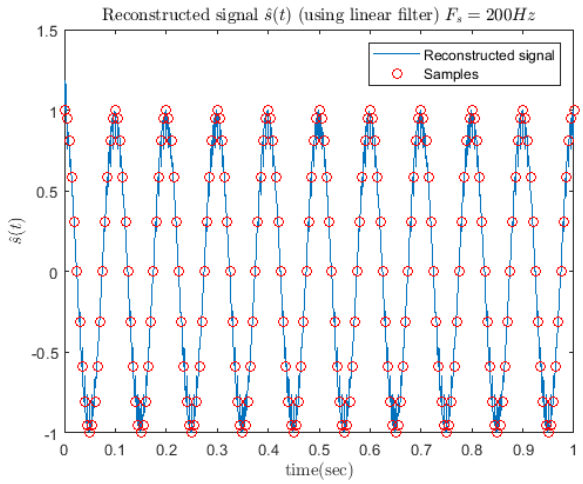
تصویر۱-۱۶ : **پاسخ فرکانسی سیگنال $s_d[n]$**

با توجه به اینکه سیگنال پیوسته با نرخ $f_{s_{CT}} = 1000Hz$ تعریف شده است و نرخ نمونه برداری $F_s = 200Hz$ است پس باید بین هر دو نمونه گسسته $\frac{f_{s_{CT}}}{F_s} = \frac{1000}{200} = 5$ سمپل تعریف کنیم تا سیگنال را آماده پیوسته سازی کنیم. به این منظور با استفاده از $L = 5$ سیگنال را آماده درونیابی میکنیم.

تصویر ۱-۱۷ ، ۱-۱۸ و ۱-۱۹ سیگنال های بازیابی شده $\hat{s}(t)$ با روش های به ترتیب ایده آل ، خطی و اسپلاین مکعبی را نشان می‌دهد.



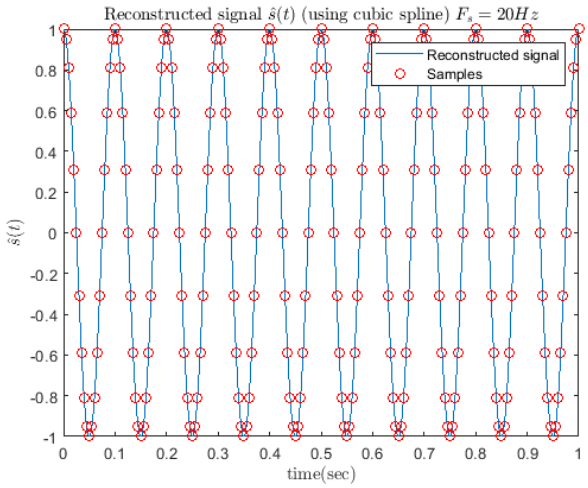
تصویر۱-۱۷ : **سیگنال بازیابی شده با روش ایده آل**



تصویر۱-۱۸ : **سیگنال بازیابی شده با روش خطی**

تصویر ۱-۲۰ میانگین خطای مربعات روش های بازیابی را نشان میدهد. همانطور که مشاهده می‌شود ، در این بازیابی درونیابی خطی میانگین مرعات خطای کمتری نسبت به دو درونیاب دیگر دارد. در بیشتر اوقات زمانی که تعداد نمونه های گسسته زیاد باشد ، تقریب اسپلاین مکعبی برای بازیابی مناسب تر است.

با مقایسه خطای بخش ه و بخش و متوجه میشویم که هر چه تعداد نمونه های گسسته بزرگتر باشد (با فرکانس بیشتری نمونه برداری اولبه را انجام دهیم) دقت در بازیابی نیز بیشتر خواهد بود.



تصویر۱-۱۹ : **سیگنال بازیابی شده با روش اسپلاین مکعبی**

Mean Squared Error:

```
% Fs = 20Hz
```

1. mode = 1 (ideal filter)

```
Er200_ideal = immse(s_hat1_200,s)
```

```
Er200_ideal = 1.6600e-04
```

2. mode = 2 (linear filter)

```
Er200_linear = immse(s_hat2_200,s)
```

```
Er200_linear = 0.0057
```

3. mode = 3 (cubic spline)

```
Er200_spline = immse(s_hat3_200,s)
```

```
Er200_spline = 5.0559e-10
```

تصویر۱-۲۰: **میانگین خطای مربعات برای بازیابی سیگنال با $F_s = 200Hz$**

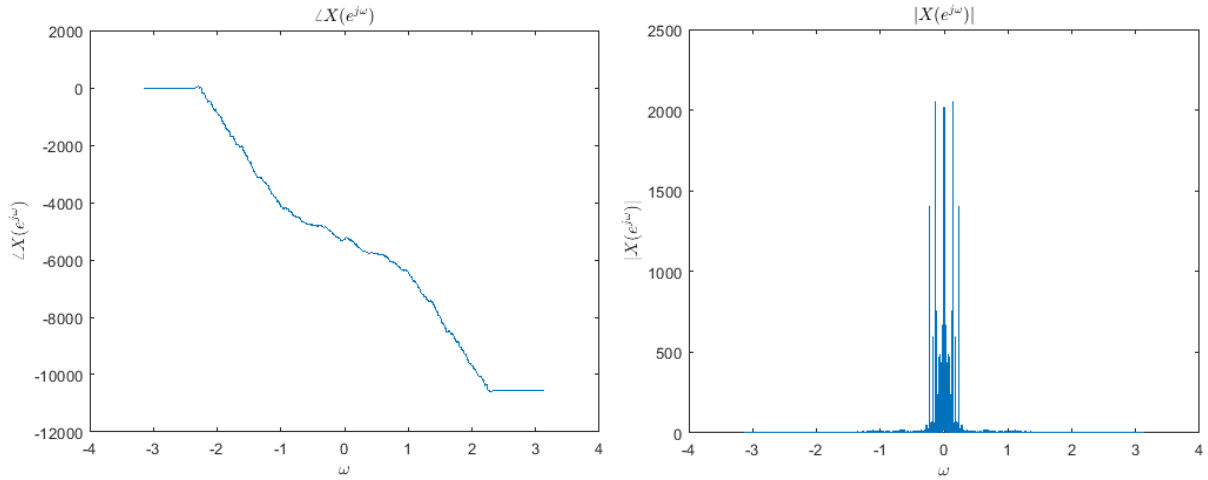
بخش دوم : سیستم تغییر دهنده نرخ نمونه برداری

سوال ۱) با استفاده از دستورات نشان داده شده در تصویر ۲-۱ ، فایل صوتی در متلب باز شده و ۱ ثانیه نخست آن در پخش می‌کنیم:

```
clear;
[music,Fs1] = audioread('sound.mp3');
x = music(1:Fs1);
sound(x,Fs1);
```

تصویر۲-۱: **باز کردن فایل صوتی و پخش ۱ ثانیه نخست**

سوال ۲) تصویر ۲-۲ اندازه و فاز تبدیل فوریه سیگنال $x[n]$ را نشان میدهد:



تصویر۲-۲ : **اندازه و فاز تبدیل فوریه سیگنال $x[n]$**

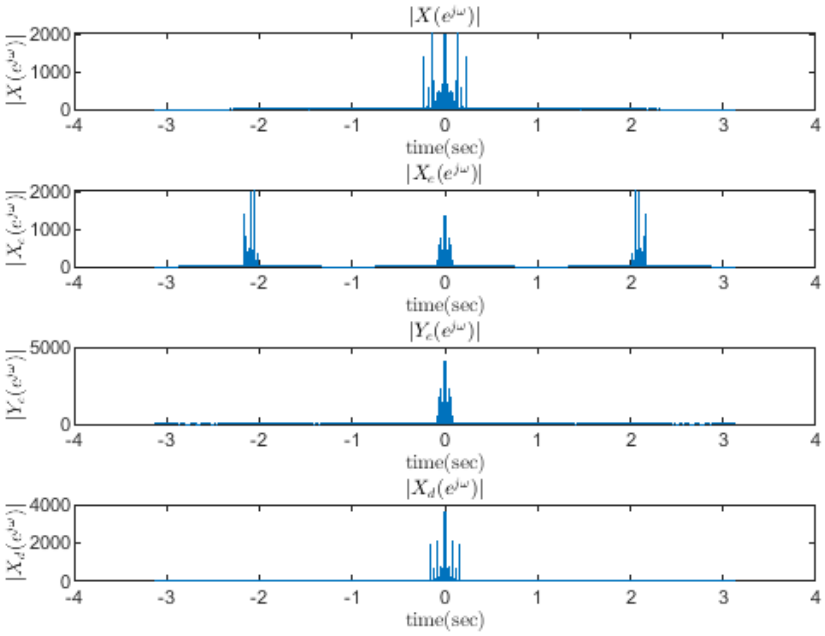
سوال ۳) با توجه به اینکه می‌خواهیم فرکانس $F_{s_1} = 44100Hz$ را برای ورودی گرفته و خروجی را با $F_{s_2} = 66150Hz$ تولید کنیم، مقدار L و M بصورت زیر تعیین می‌شود:

$$\frac{L}{M} = \frac{66150}{44100} = \frac{3}{2} \rightarrow L = 3, M = 2$$

همچنین فرکانس قطع فیلتر با توجه به مقدار L=3 تعیین می‌شود:

$$L = 3 \rightarrow |\omega| \leq \frac{\pi}{L} = \frac{\pi}{3} \rightarrow \omega_c = \frac{\pi}{3}$$

سوال ۴) تصویر ۲-۳ اندازه تبدیل فوریه سیگنال های $x_e[n]$ ، $x_d[n]$ ، $y_e[n]$ و $x[n]$ را نشان میدهد. همانطور که مشاهده می‌شود، پس از عبور از باز کننده اندازه تبدیل فشرده شده و با استفاده از فیلتر پایین گذر باند پایه تبدیل بدست می آید. در نهایت نیز با استفاده از فشرده ساز ، طیف کمی باز تر شده و بهره دو برابر میشود.



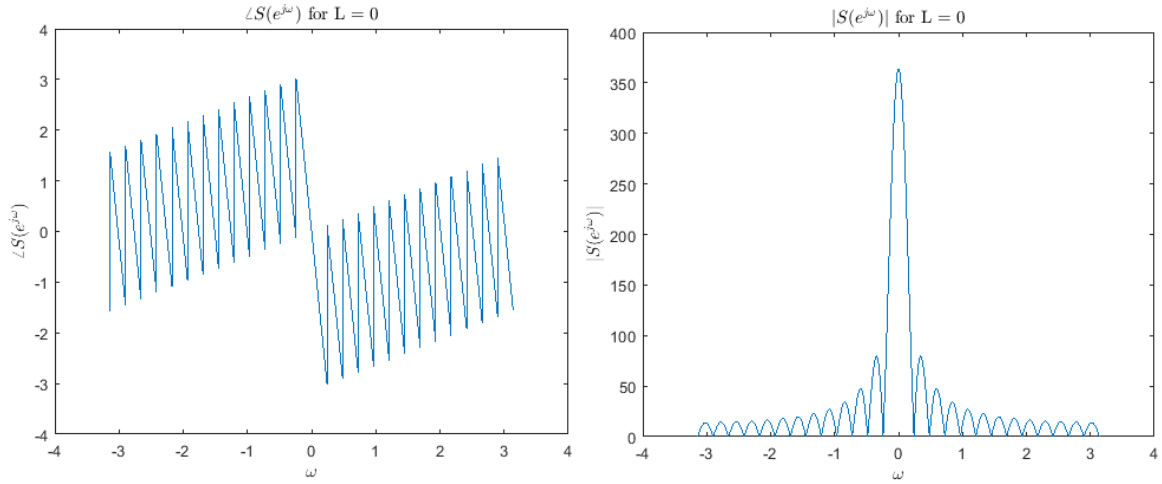
تصویر۲-۳ : **اندازه تبدیل فوریه سیگنال های $x_e[n]$ ، $x_d[n]$ ، $y_e[n]$ و $x[n]$**

سوال ۵) پس از دریافت سیگنال خروجی و شنیدن آن متوجه شباهت این صوت با صوت ابتدایی می‌شویم. دلیل این امر محسوس نبودن تغییرات میباشد. برای اینکه تغییرات را بیشتر حس کنیم باید تفاوت بیشتری بین فرکانس ورودی و خروجی وجود داشته باشد. در حالت کلی دلیل افزایش کیفیت صدا افزایش تعداد سمپل ها با افزایش فرکانس خروجی است. البته در فرکانس های بالا مثل این مسئله این موضوع چندان حس نمی‌شود.

بخش سوم : تحلیل تبدیل فوریه

سوال ۱) تصویر ۳-۱ اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 0$ را نشان میدهد:

$$s(t) = \sum_{i=0}^{L} x(t - iT) = x(t)$$



تصویر۳-۱ : **اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 0$**

سوال ۲) تصویر ۳-۲ پهنای باند سیگنال $s(t)$ به ازای $L = 0$ را نشان میدهد.

```
ABS_S0 = abs(S0ejw);
Peak_S0 = max(ABS_S0);
i0 = min(find(ABS_S0 > Peak_S0/sqrt(2)));
j0 = max(find(ABS_S0 > Peak_S0/sqrt(2)));
BW_L0 = w(j0)-w(i0)
```

```
BW_L0 = 0.2142
```

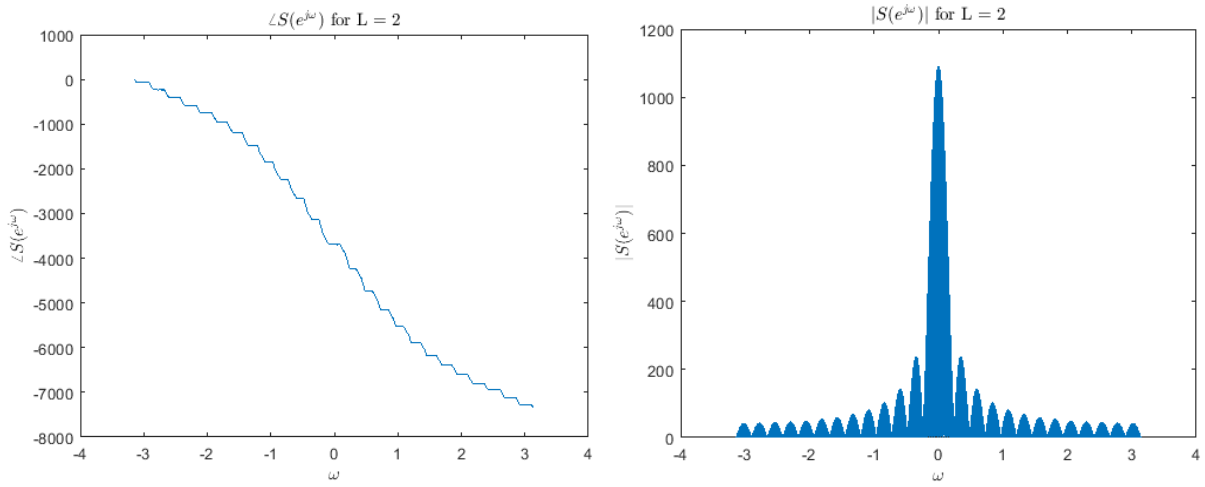
```
BW_L0_per100 = BW_L0/(2*pi)
```

```
BW_L0_per100 = 0.0341
```

تصویر۳-۲ : **پهنای باند سیگنال $s(t)$ به ازای $L = 0$ ، 3.41% کل طیف**

سوال ۳) تصویر ۳-۳ اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 2$ را نشان میدهد:

$$s(t) = \sum_{i=0}^{2} x(t - iT) = x(t) + x(t - T) + x(t - 2T)$$



تصویر۳-۳ : **اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 2$**

تصویر ۳-۴ پهنای باند سیگنال $s(t)$ به ازای $L = 2$ را نشان میدهد.

```
ABS_S2 = abs(S2ejw);
Peak_S2 = max(ABS_S2);
i2 = min(find(ABS_S2 > Peak_S2/sqrt(2)));
j2 = max(find(ABS_S2 > Peak_S2/sqrt(2)));
BW_L2 = w(j2)-w(i2)
```

```
BW_L2 = 0.2112
```

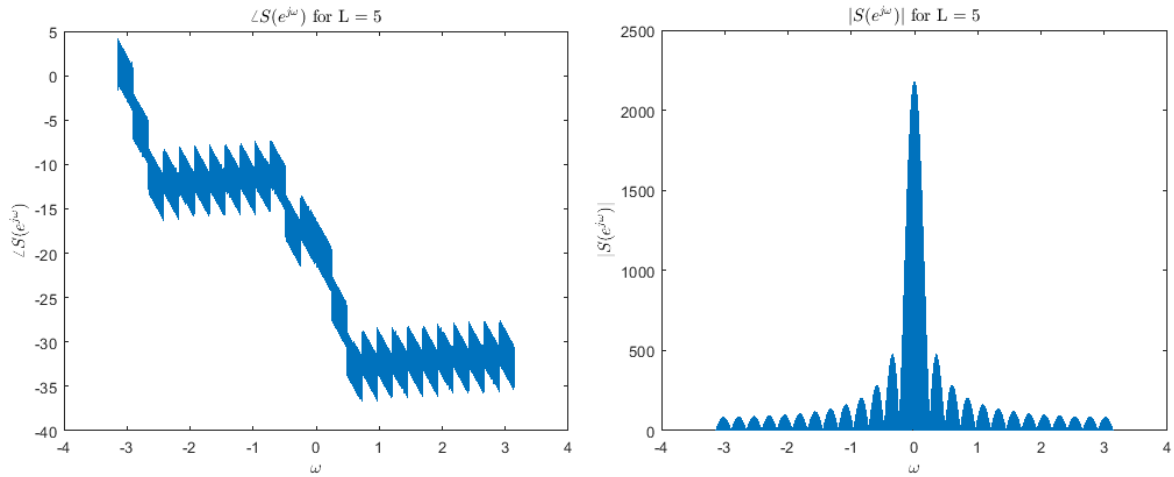
```
BW_L2_per100 = BW_L2/(2*pi)
```

```
BW_L2_per100 = 0.0336
```

تصویر۳-۴ : **پهنای باند سیگنال $s(t)$ به ازای $L = 2$ ، 3.36% کل طیف**

تصویر ۳-۵ اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 5$ را نشان میدهد:

$$s(t) = \sum_{i=0}^{5} x(t - iT) = x(t) + x(t - T) + x(t - 2T) + (t - 3T) + x(t - 4T) + x(t - 5T)$$



تصویر۳-۵ : **اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 5$**

تصویر ۳-۶ پهنای باند سیگنال $s(t)$ به ازای $L = 5$ را نشان میدهد.

```
ABS_S5 = abs(S5ejw);
Peak_S5 = max(ABS_S5);
i5 = min(find(ABS_S5 > Peak_S5/sqrt(2)));
j5 = max(find(ABS_S5 > Peak_S5/sqrt(2)));
BW_L5 = w(j5)-w(i5)
```

```
BW_L5 = 0.2061
```

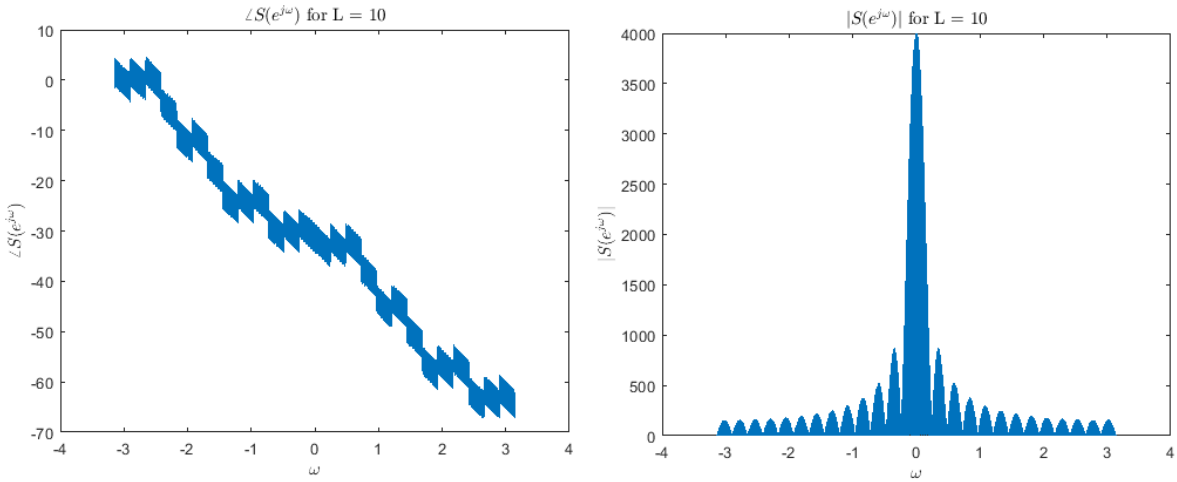
```
BW_L5_per100 = BW_L5/(2*pi)
```

```
BW_L5_per100 = 0.0328
```

تصویر۳-۶ : **پهنای باند سیگنال $s(t)$ به ازای $L = 5$ ، 3.28% کل طیف**

سوال ۴) همانطور که مشاهده شد با افزایش مقدار L ، اوج تبدیل فوریه نسبت به سایر نقاط رشد بیشتری دارد و به همین خاطر پهنای باند سیگنال کاهش پیدا می‌کند. با افزایش L پهنای باند آنقدر کم میشود تا به یک دلتا در صفر تبدیل شود پس سیگنال کماکان پایین گذر باقی می‌ماند. برای بررسی این موضوع اندازه و فاز تبدیل فوریه و پهنای باند سیگنال $s(t)$ به ازای $L = 10$ بدست می آوریم:

تصویر ۳-۷ اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 10$ را نشان میدهد:



تصویر۳-۷ : **اندازه و فاز تبدیل فوریه سیگنال $s(t)$ به ازای $L = 10$**

```
ABS_S10 = abs(S10ejw);
Peak_S10 = max(ABS_S10);
i10 = min(find(ABS_S10 > Peak_S10/sqrt(2)));
j10 = max(find(ABS_S10 > Peak_S10/sqrt(2)));
BW_L10 = w(j10)-w(i10)
```

```
BW_L10 = 0.1985
```

```
BW_L10_per100 = BW_L10/(2*pi)
```

```
BW_L10_per100 = 0.0316
```

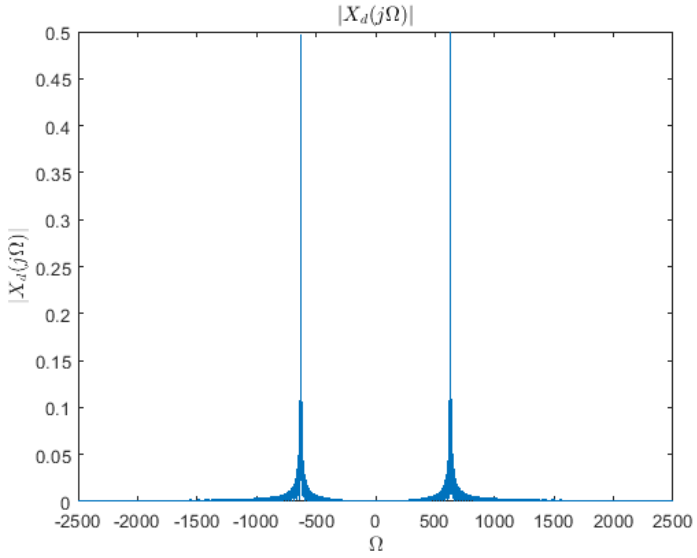
تصویر۳-۶ : **پهنای باند سیگنال $s(t)$ به ازای $L = 5$ ، 3.16% کل طیف**

سوال ۵) همانطور که در سوال قبل اشاره شد، پهنای باند سیگنال آنقدر کاهش پیدا می‌کند تا تبدیل فوریه به یک دلتا تبدیل شود. برای توجیه این قضیه میتوانیم به ارتباط ضرایب سری فوریه و تبدیل فوریه اشاره کنیم. اگر مقدار L به سمت بینهایت میل کند در حقیقت سیگنال $s(t)$ یک سیگنال متناوب خواهد بود.

$$X(j\Omega) = \sum 2\pi a_k \delta(\Omega - \Omega_k)$$

بخش چهارم : ساخت نصف کننده فرکانسی

سوال ۱) تصویر ۴-۱ اندازه تبدیل فوریه سیگنال $x(t)$ به ازای $f_c = 100Hz$ را نشان میدهد:



تصویر۴-۱ : **اندازه تبدیل فوریه سیگنال $x_c(t)$**

همانطور که انتظار میرفت پهنای باند این سیگنال در حدود $2\pi f_c = 200\pi$ میباشد. پس نرخ نایکوئیست این سیگنال بصورت زیر بدست می‌آید:

$$\Omega_N = 200\pi \rightarrow f_N = 100 \rightarrow Nyquist\ rate: 2f_N = 200Hz$$

سوال ۲) حال f_s را دو برابر نرخ نایکوئیست تعریف می‌کنیم. پس داریم: $f_s = 400Hz$

برای اینکه بتوانیم با استفاده از تابع *filter* خروجی را بدست آوریم به صورت زیر عمل میکنیم:

۱. ابتدا سیگنال *z[n]* را بصورت زیر تعریف می‌کنیم:

$$z[n] \triangleq y[n-1] \rightarrow z[n] = 0.5x[n] + x[n-1] + 0.5x[n-2]$$

۲. حال ضرایب این سیستم را به همراه ورودی $x_c(t) = \cos(200\pi t)$ به دستور فیلتر میدهیم.

۳. در نهایت نیز خروجی فیلتر را یک واحد به سمت چپ شیفت میدهیم. برای شرایط اولیه داریم:

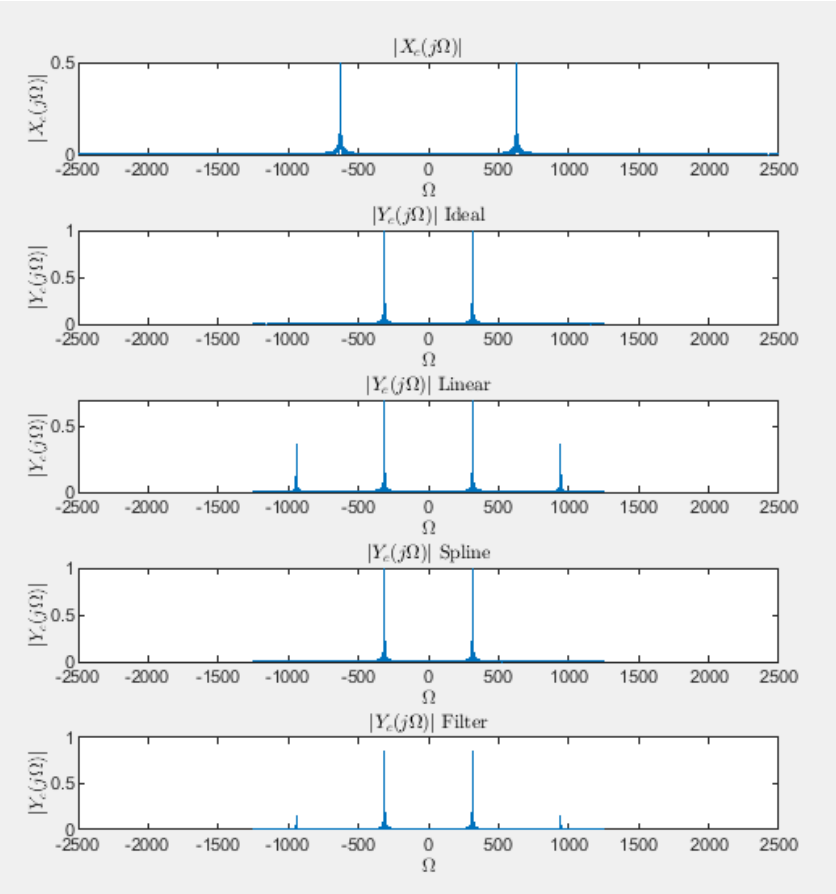
$$y[-1] = x[-1] + 0.5x[-2] + 0.5x[0] = 0$$

تمامی این مراحل را مطابق با تصویر ۴-۲ انجام می‌دهیم:

```
L = 2;
yc_ide = Interpolate(xe,ne,L,fs,1);
yc_lin = Interpolate(xe,ne,L,fs,2);
yc_spl = Interpolate(xe,ne,L,fs,3);
zc = filter([0.5 1 0.5],1,xe);
yc_fil = [0 zc(1:length(zc)-1)];
```

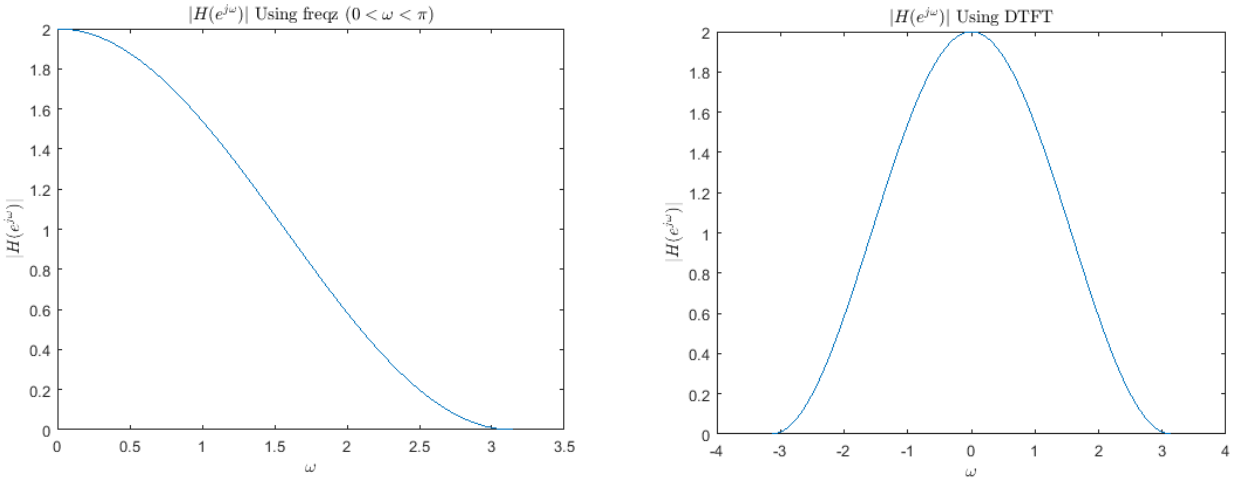
تصویر۴-۲ : **بدست آوردن خروجی در حوزه زمان**

در نهایت اندازه تبدیل فوریه سیگنال خروجی به هر ۴ روش درونیابی و همچنین سیگنال ورودی در تصویر ۴-۳ نشان داده شده است:



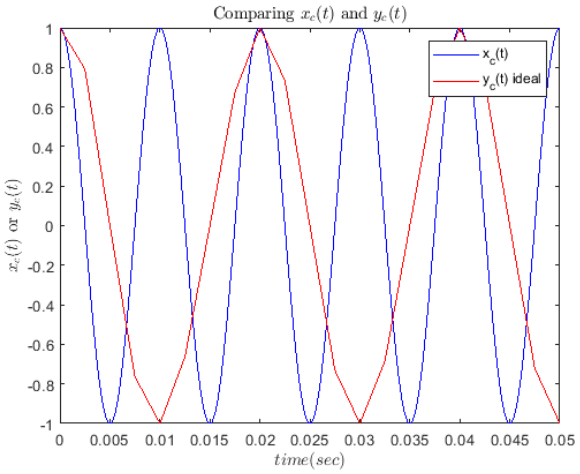
تصویر۴-۳ : **اندازه تبدیل فوریه سیگنال های $x_c(t)$ و $y_c(t)$ به هر ۴ روش**

سوال ۳) تصویر ۴-۴ پاسخ فرکانسی فیلتر را با دو روش freqz و DTFT نشان می‌دهد:



تصویر۴-۲ : **پاسخ فرکانسی فیلتر را با دو روش freqz و DTFT**

سوال ۴) تصویر ۴-۵ خروجی (روش ایده آل) سیستم را به همراه ورودی در حوزه زمان نشان میدهد:



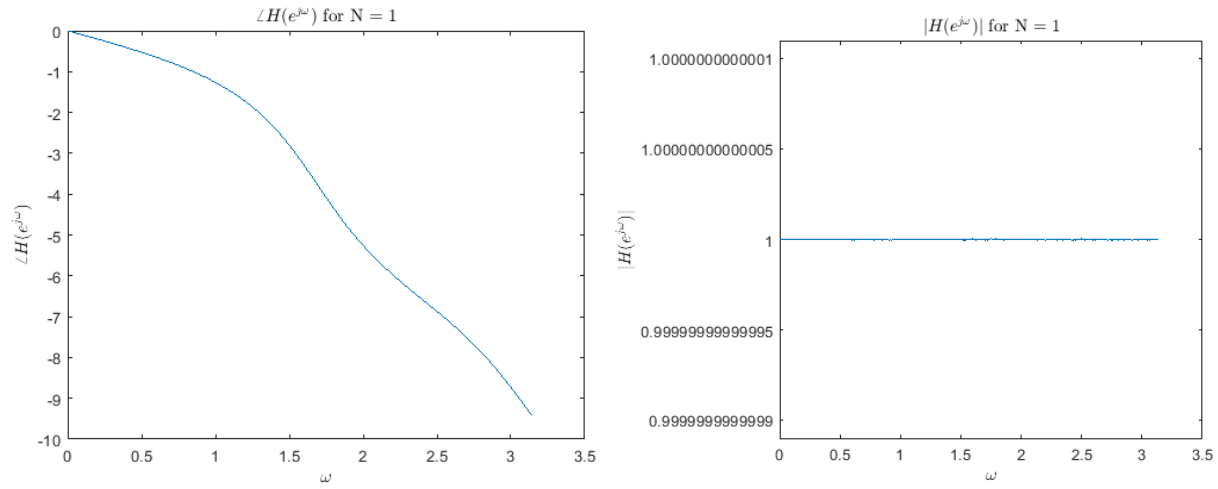
تصویر۴-۵ : **خروجی (روش ایده آل) سیستم را به همراه ورودی در حوزه زمان**

سوال ۵) همانطور که در تصویر ۴-۳ مشاهده میشود، تبدیل فوریه خروجیبا استفاده از هر سه روش درونیابی خروجی یکسانی داشته است. در حقیقت بخاطر وجود بازکننده با ضریب L=2 ، تبدیل فوریه سیگنال با ضریب ۲ فشرده میشود و ضربه ها در فرکانس نصف اتفاق می افتد و انگار فرکانس ورودی در خروجی نصف شده است. در روش آخر(فیلتر رابطه ۱۲) با توجه به پاسخ فرکانسی آن که در سوال ۳ نمایش داده شد، بهره ایده آل نیست و به همین خاطر ضربه ها ضعیف تر خود را نشان می‌دهند.

سوال ۶) همانطور که در سوال قبل اشاره شد ، بخاطر ایده آل نبودن این فیلتر اعوجاج دامنه در خروجی اتفاق می افتد و دامنه تضعیف میشود. همچنین این تضعیف برای فرکانس های مختلف متفاوت است. در نتیجه استفاده از این فیلتر در حالت کلی برای این سیستم مناسب نیست.

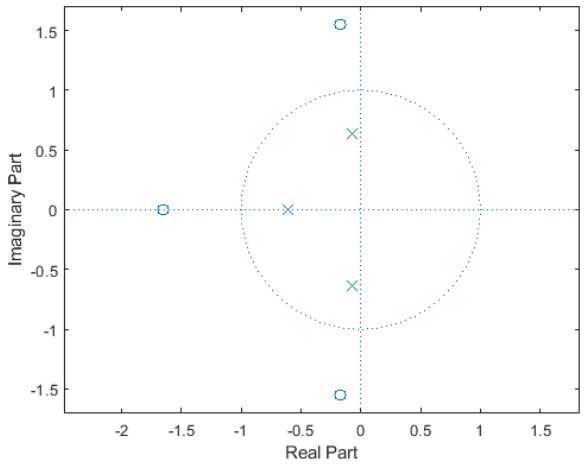
بخش پنجم : شناسایی فاز

سوال ۱) تصویر ۵-۱ نمودار اندازه و فاز پاسخ فرکانسی سیستم را نشان میدهد. همانطور که مشاهده میشود اندازه پاسخ فرکانسی ثابت بوده و فاز محدوده تقریبی ۰ تا -۱۰ دارد.



تصویر۵-۱ : **اندازه و فاز پاسخ فرکانسی فیلتر به ازای N=1**

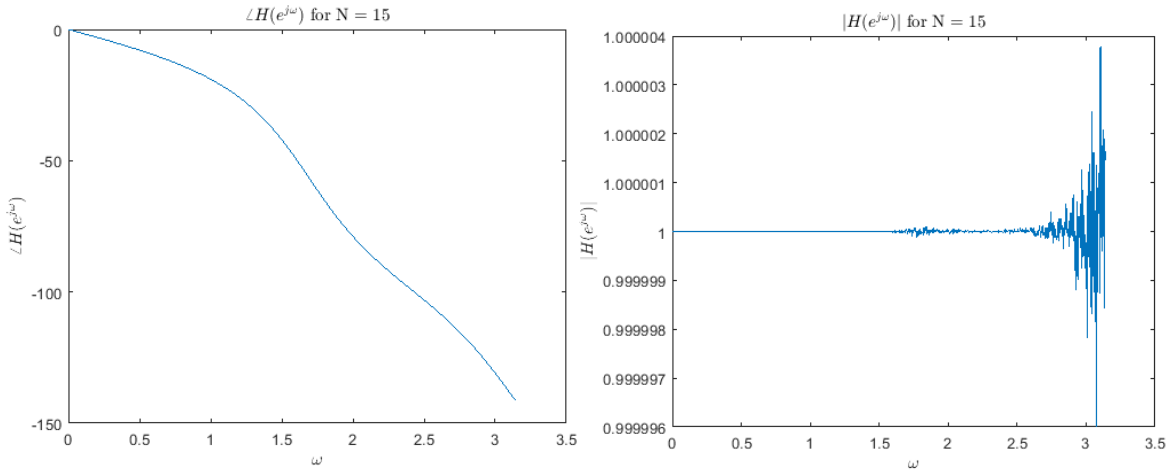
تصویر ۵-۲ نمودار صفر و قطب های سیستم را نشان می‌دهد.



تصویر۵-۲ : **نمودار صفر و قطب های سیستم به ازای N=1**

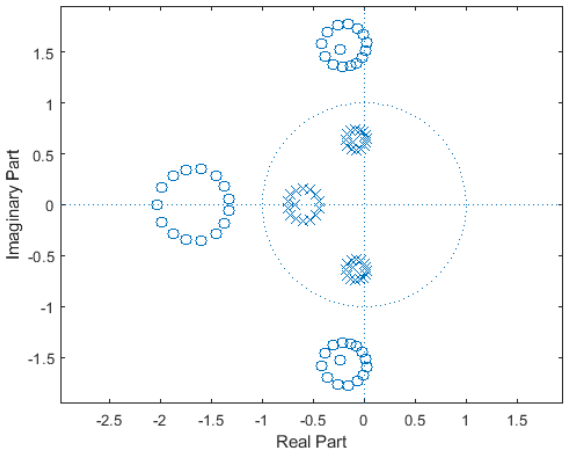
سوال ۲) با شنیدن فایل صوتی خروجی متوجه تغییری در آن نمیشویم. دلیل این امر عدم تغییر اندازه تبدیل است. دلیل اینکه تغییر فاز را حس نمیکنیم این است که در حقیقت فاز بیانگر جایجایی و شیفت زمانی است و گوش انسان قادر به تشخیص این موضوع نیست و صرفا اندازه آن شنیده میشود.

سوال ۳) تصویر ۵-۳ نمودار اندازه و فاز پاسخ فرکانسی سیستم را نشان میدهد. همانطور که مشاهده میشود اندازه پاسخ فرکانسی تقریباً ثابت و در حدود ۱ است و فاز محدوده تقریبی ۰ تا -۱۵۰ دارد. (نسبت به حالت قبل ۱۵ برابر شده است)



تصویر۵-۳ : **اندازه و فاز پاسخ فرکانسی فیلتر به ازای N=15**

تصویر ۵-۴ نمودار صفر و قطب های سیستم را نشان می‌دهد.

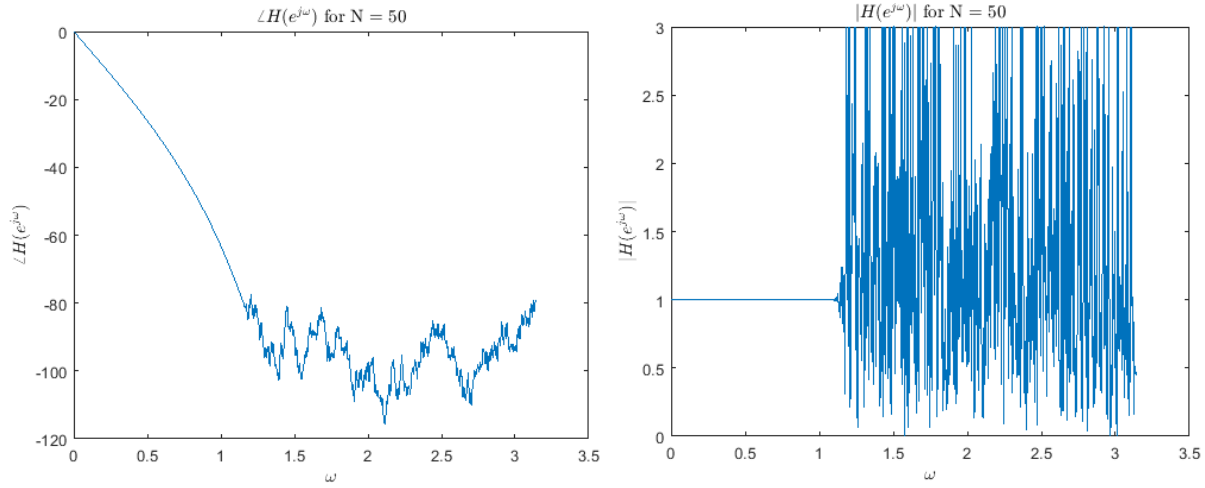


تصویر۵-۴ : **نمودار صفر و قطب های سیستم به ازای N=15**

نمودار صفر و قطب های سیستم به ازای $N = 15$ درست رسم نشده است. دلیل این امر ناتوانی متلب در بدست آوردن دقیق قطب و صفر هاست و به همین خاطر همه قطب و صفر ها دقیقا در یک نقطه نیستند.

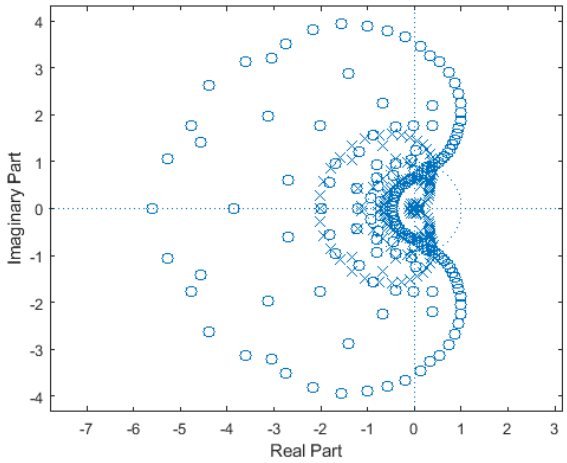
سوال ۴) کماکان با شنیدن فایل صوتی خروجی متوجه تغییری در آن نمیشویم. پس همانطور که در سوال ۲ اشاره شد گوش انسان قادر به درک تغییر فاز نیست. دلیل اینکه تغییر فاز را حس نمیکنیم این است که در حقیقت فاز بیانگر جایجایی و شیفت زمانی است و گوش انسان قادر به تشخیص این موضوع نیست و صرفا اندازه آن شنیده میشود.

سوال ۵) تصویر ۵-۵ نمودار اندازه و فاز پاسخ فرکانسی سیستم را نشان میدهد. همانطور که مشاهده میشود اندازه پاسخ فرکانسی دیگر قابت نیست و به دلیل ناپایدار شدن دچار اعوجاج میشود و فاز محدوده تقریبی نیز دچار اعوجاج در همین نقاط شده است .



تصویر۵-۵ : **اندازه و فاز پاسخ فرکانسی فیلتر به ازای N=50**

تصویر ۵-۶ نمودار صفر و قطب های سیستم را نشان می‌دهد.



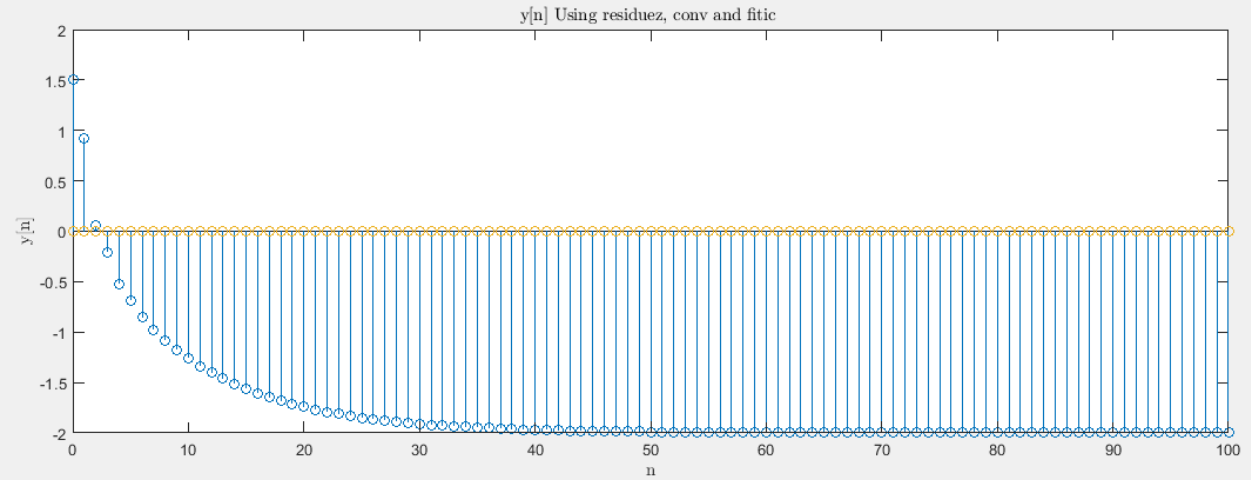
تصویر۵-۶ : **نمودار صفر و قطب های سیستم به ازای N=50**

همانطور که در تصویر ۵-۶ مشاهده میشود، با توجه به اینکه قطب های سیستم همه در دایره واحد نیستند پس سیستم پایدار نیست و در نتیجه باعث میشود تا صدایی غیرقابل انتظار بشنویم. البته این ناپایداری بخاطر اشتباه در محاسبات متلب است و درحقیقت وجود ندارد و سیستم کماکان باید پایدار باقی بماند.

بخش ششم : حل معادله تفاضلی

*** حل معادله تفاضلی با استفاده از دستورات residuez ، conv و filtic:

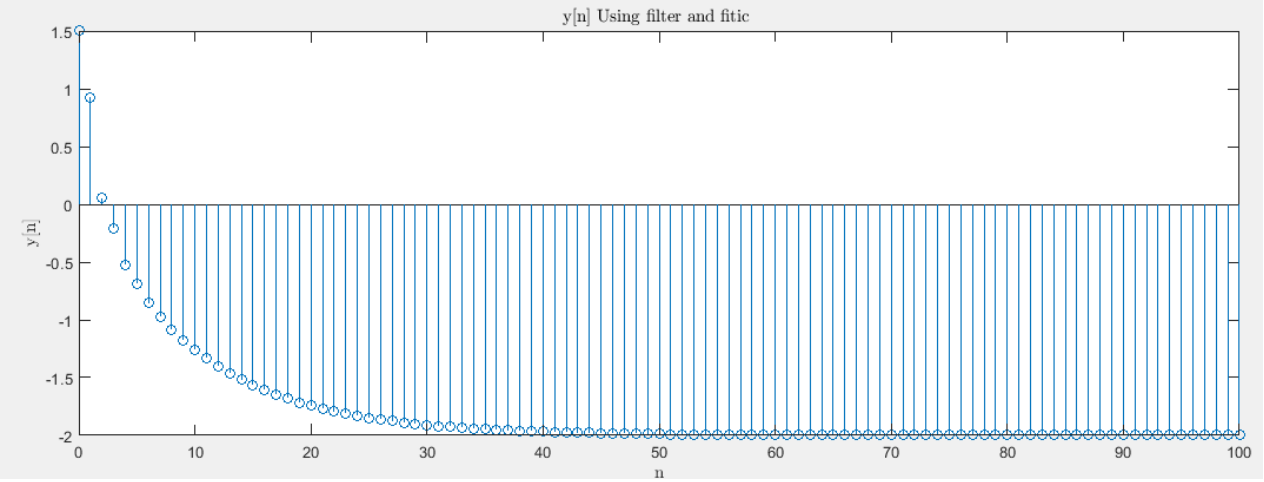
تصویر ۶-۱ پاسخ کامل را به این روش نشان می‌دهد.



تصویر۶-۱: پاسخ کامل با استفاده از دستورات residuez ، conv و filtic

*** حل معادله تفاضلی با استفاده از دستورات filter و filtic:

تصویر ۶-۲ پاسخ کامل را به این روش نشان می‌دهد.



تصویر۶-۲: پاسخ کامل با استفاده از دستورات filter و filtic

همانطور که شماهده میشود پاسخ کامل با استفاده از هر دو روش یکسان می‌باشد.

با استفاده از نتایج تصویر ۶-۳ میتوانیم فرم پاسخ کامل معادلهٔ تفاضلی را بنویسیم:

$$y[n] = -2u[n] + 2.116(0.9)^n u[n] + 1.7188(0.5)^n u[n] - 0.3304(-0.5)^n u[n]$$

```
M'

ans = 1x4
    1.0000    0.9000    0.5000    0.5000

A'

ans = 1x4
     0     0     0     1

r'

ans = 1x4
   -2.0000    2.1116    1.7188   -0.3304
```

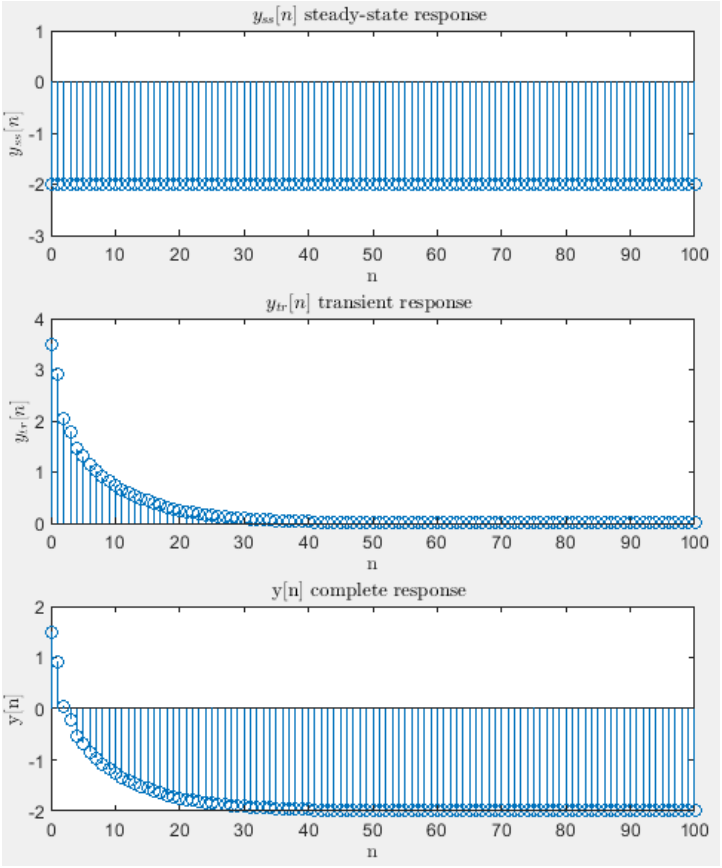
تصویر۶-۳: **نتایج روش اول**

در نتیجه میتوانیم پاسخ را به دو صورت حالت صفر-ورودی صفر و گذرا-ماندگار بنویسیم:

***پاسخ گذرا و ماندگار:

$$y_{ss}[n] = -2u[n]$$
$$y_{tr}[n] = 2.116(0.9)^n u[n] + 1.7188(0.5)^n u[n] - 0.3304(-0.5)^n u[n]$$

تصویر ۶-۴ نمودار پاسخ گذرا و ماندگار را نشان میدهد:



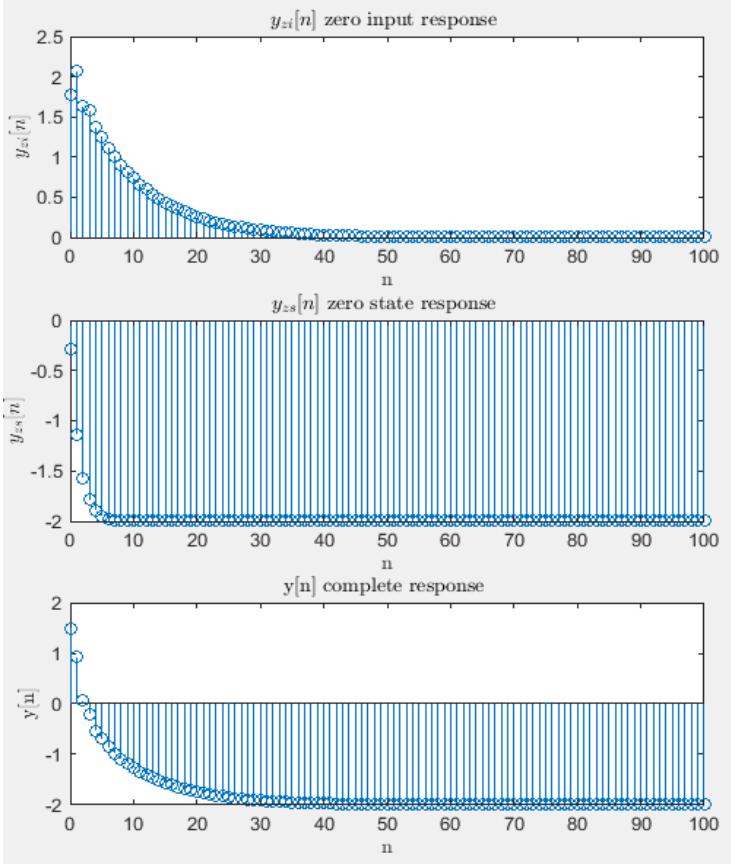
تصویر۶-۴: **پاسخ ماندگار و گذرا**

***پاسخ حالت صفر و ورودی صفر:

$$y_{zi}[n] = 2.116(0.9)^n u[n] - 0.3304(-0.5)^n u[n]$$

$$y_{zc}[n] = -2u[n] + 1.7188(0.5)^n u[n]$$

تصویر ۶-۵ نمودار پاسخ گذرا و ماندگار را نشان میدهد:



تصویر۶-۵: **پاسخ حالت صفر و وروید صفر**