Glasgow College, UESTC

# Digital Signal Processing

# Homework 3

Author: Changgang Zheng
UoG ID: 2289258Z
UESTC ID: 2016200302027
E-mail: chnaggangzheng@std.uestc.edu.cn

University of Glasgow

电子科技大学
University of Electronic Science and Technology of China

# HOMEWORK 3

CHANGGANG ZHENG[1]

April 15, 2019

## CONTENTS



## LIST OF FIGURES



## INTRODUCTION

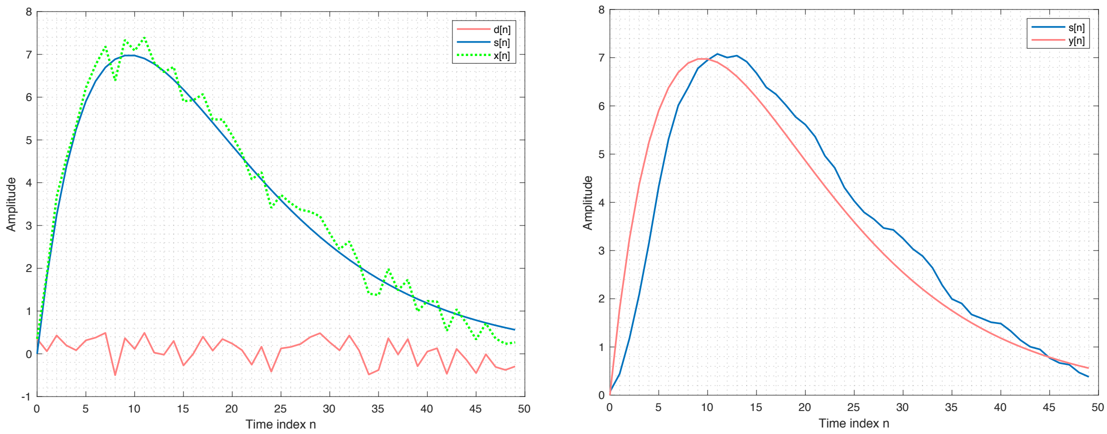
This report is the homework that should be finished on the MATLAB, there are three questions about Digital Signal Processing. The first question is to introduce filter and let us understand the property of the filter. The seconde question is about a system, and how to use the initial condition to find the result on it. The last questions of the homework require us to generate and explain a filter.

1*Glasgow College, University of Electronic Science and Technology of China, ChengDu, China*

# 1 PROBLEM M4.1

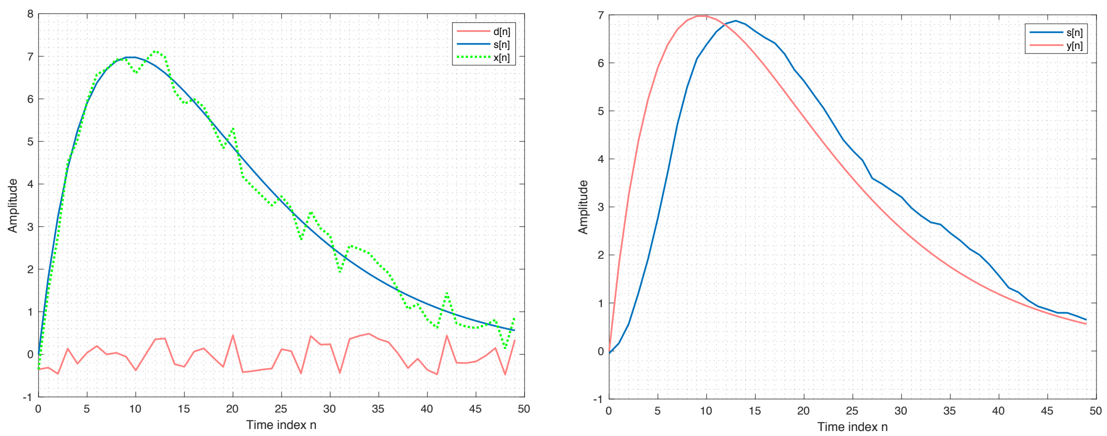
**Using the program provided by this book, investigate the smoothing effect of the signal, find the smoothing result and how is the condition of the delay by using a different length of the sliding windows.**

From the left hand side graph of each figure, we can se that the longer the filter length is, the soomther, it will be. This means the smoothing is improved. Also, it will gain a longer delay with the increase of the filter size.




how original signal is distorted    smoothing signal

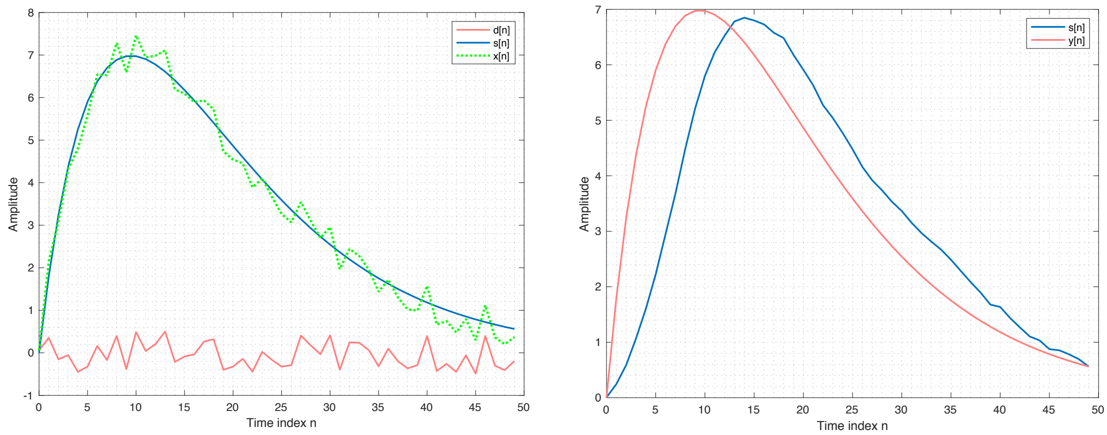
**Figure 1:** filter length equals to 5




how original signal is distorted    smoothing signal

**Figure 2:** filter length equals to 7

```
% Program M4.1
% Signal Smoothing by a Moving-Average Filter %
R = 50;
```




how original signal is distorted

smoothing signal

**Figure 3:** filter length equals to 9

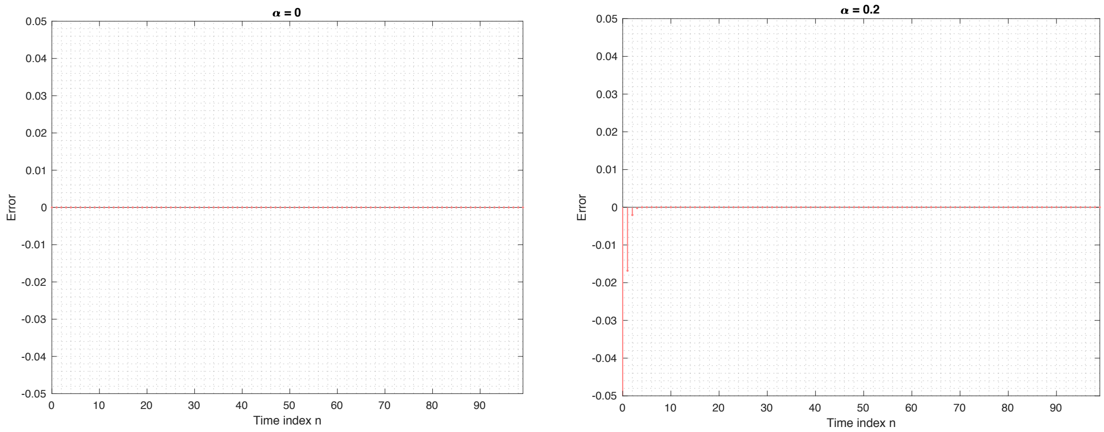
```
d = rand(R,1)-0.5;
m = 0:1:R-1;
s = 2*m.*(0.9.^m);
x =s+d';
figure;
plot(m,d,'r-','LineWidth',1.5,'color',[1,0.5,0.5])
hold on
plot(m,s,'LineWidth',1.5)
hold on
plot(m,x,'g:','LineWidth',2)
xlabel('Time index n');
ylabel('Amplitude')
legend('d[n]','s[n]','x[n]');
grid minor


figure;
M = input('Number of input samples = ');
b = ones(M,1)/M;
y = filter(b,1,x);
plot(m,y,'LineWidth',1.5)
hold on
plot(m,s,'LineWidth',1.5,'color',[1,0.5,0.5])
hold on
legend('s[n]','y[n]');
xlabel ('Time index n');ylabel('Amplitude')
grid minor
```

## 2 PROBLEM M4.3

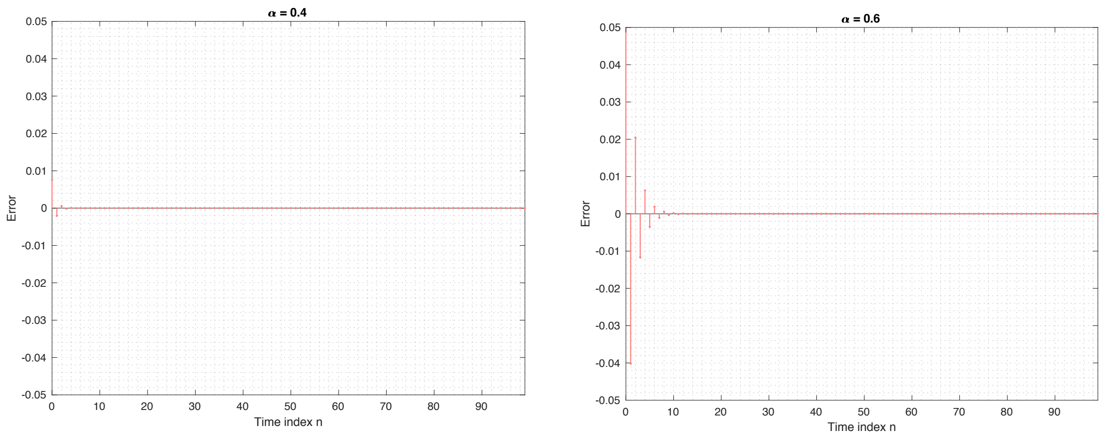
**Using the program to calculate the y[n] in the euqation 4.105 in problem 4.12, suppose the y[-1]=1 and x=$\alpha$u[n]. under this condition y[n] would converge to**

**$\sqrt{\alpha}$ when n approach infinity. Show and plot these results by using different $\alpha$ and think about how to compute the square root of $\alpha$ when it is bigger then 1.**




error for α=0 error for α=0.2

**Figure 4:** Figure of error for different $\alpha$

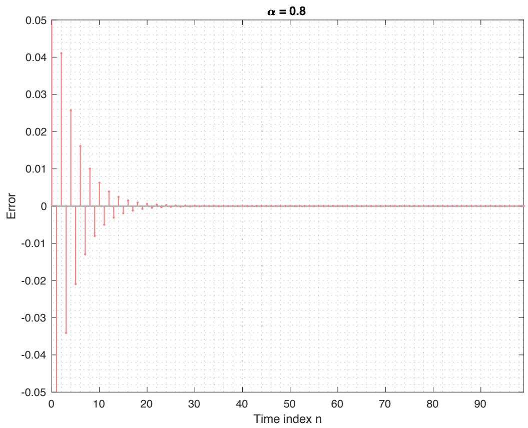



error for α=0.4 error for α=0.6

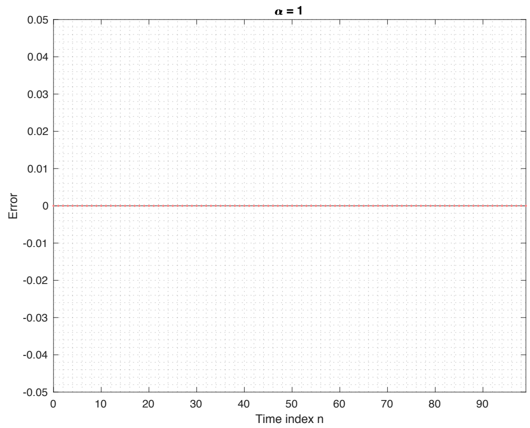
**Figure 5:** Figure of error for different $\alpha$

```
% Problem M4.3
clear
clc

format long
alpha = input('Alpha = ');
y0 = alpha;
y = zeros(1,61);
L = length(y) - 1;
y(1) = alpha - y0*y0 + y0;
n = 2;
```




error for $\alpha$=0.8




error for $\alpha$=1

**Figure 6:** Figure of error for different $\alpha$

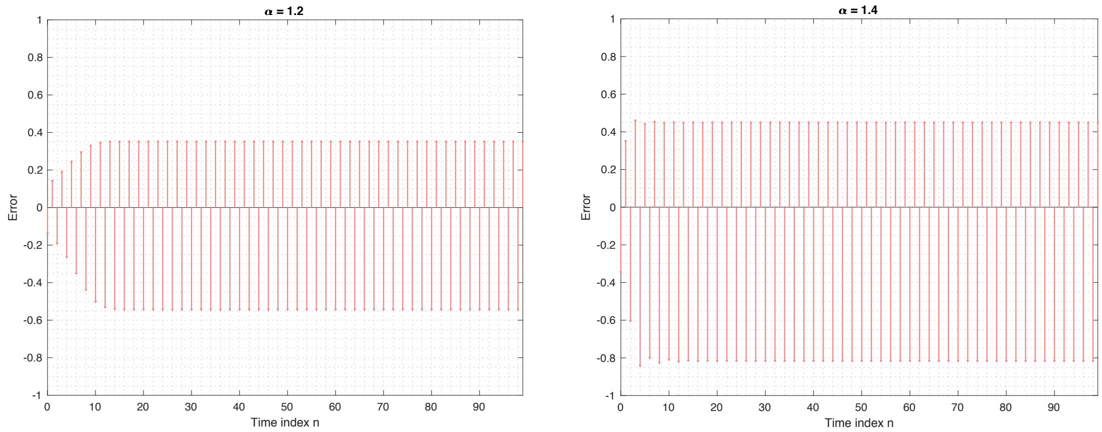
```
while (abs(y(n-1) - y0) > 0.001 || n⩽100)&&n⩽100
    y2 = alpha - y(n-1)*y(n-1) + y(n-1);
    y0 = y(n-1);
    y(n) = y2;
    n = n+1;
end
disp('Square root of alpha is');
disp(y(n-1));
m = 0:n-2;
err = y(1:n-1) - sqrt(alpha);

figure
fig1 = plot(m,y(1:n-1),'.','LineWidth',1,'color',[1,0.5,0.5]);
axis([0 n-2 -1.5 1.5]);
xlabel('Time index n');
ylabel('y');
title(['result of y when \alpha = ',num2str(alpha)]);
grid minor
saveas(fig1,['y_for_a_is_',num2str(alpha),'.pdf'],'pdf')

figure
fig2=stem(m,err,'.','LineWidth',1,'color',[1,0.5,0.5]);
axis([0 n-2 -0.05 0.05]);
xlabel('Time index n');
ylabel('Error');
title(['\alpha = ',num2str(alpha)]);
grid minor
saveas(fig2,['error_for_a_is_',num2str(alpha),'.pdf'],'pdf')
%
%
```

**However for that function, we can find if a bigger than 1, the error is not chonverge and the result is not correct, so we use another function.**

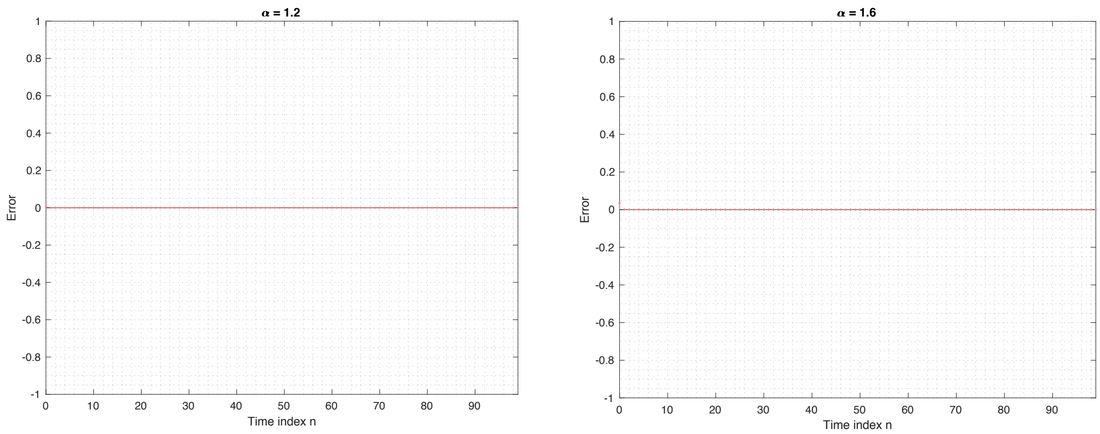
$$y[n] = 0.5y[n-1] + \frac{x[n]}{2y[n-1]} \tag{1}$$




error for $\alpha$=1.2 error for $\alpha$=1.4

**Figure 7**: Figure of error for different $\alpha$

We can find when x[n]=$\alpha$x[n], $\alpha$ is the number we want to calculate, the result can be shown as the following figures. Also from the output, we can see Alpha = 1.6 Square root of alpha is 1.264911064067352 and Alpha = 1.2 Square root of alpha is 1.095445115010332.




error for $\alpha$=1.2 error for $\alpha$=1.6

**Figure 8**: Figure of error for different $\alpha$

```
% Problem M4.3
clear
clc
format long
alpha = input('Alpha = ');
y0 = alpha;
y = zeros(1,61);
L = length(y) - 1;
y(1) = alpha/(2*y0) + 0.5*y0;
n = 2;
```

```
while (abs(y(n-1) - y0) > 0.001 || n≤100)&&n≤100
    y2 = alpha/(2*y(n-1)) + 0.5*y(n-1);
    y0 = y(n-1);
    y(n) = y2;
    n = n+1;
end
disp('Square root of alpha is');
disp(y(n-1));
m = 0:n-2;
err = y(1:n-1) - sqrt(alpha);

figure
fig1 = plot(m,y(1:n-1),'.','LineWidth',1,'color',[1,0.5,0.5]);
axis([0 n-2 -1.5 1.5]);
xlabel('Time index n');
ylabel('y');
title(['result of y when \alpha = ',num2str(alpha)]);
grid minor
saveas(fig1,['pro_y_for_a_is_',num2str(alpha),'.pdf'],'pdf')

figure
fig2=stem(m,err,'.','LineWidth',1,'color',[1,0.5,0.5]);
axis([0 n-2 -1 1]);
xlabel('Time index n');
ylabel('Error');
title(['\alpha = ',num2str(alpha)]);
grid minor
saveas(fig2,['pro_error_for_a_is_',num2str(alpha),'.pdf'],'pdf')
```

## 3 PROBLEM M4.5

**Generate the filter shown in problem 4.105, and verify its filtering operation.**

From the following program, we can calculated the spectrum of the filter. On the frequency spectrum of the filter, we can see that the spectrum for higher frequency is high and the for the lower frequency is dicreasing and close to zero. From the above analysis, we can understand this filter is a highpass filter.

$$H(e^{jw}) = h[0](1 - e^{1j2w}) + h[1]e^{-1} = e^{-jw}(2h[0]\cos(w) + h[1]) \tag{2}$$

$$H(e^{j0.2}) = 2h[0]\cos(0.2) + h[1] = 0 \tag{3}$$

$$H(e^{j0.5}) = 2h[0]\cos(0.5) + h[1] = 1 \tag{4}$$

$$h[0] = -4.8788 and h[1] = 9.5631 \tag{5}$$

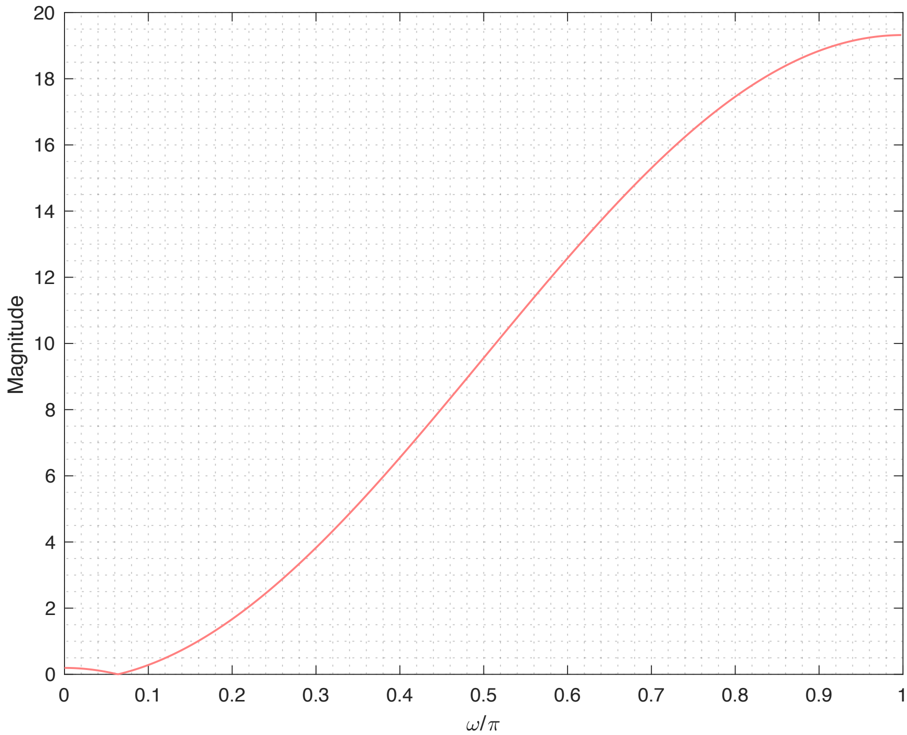



**Figure 9**: High pass filter for Problem M4.5

From the above equation, we can get h[0]=-4.8788 and h[1]=9.5631

```
% Problem M4.5
clear
clc

h = [-4.8788 9.5631 -4.8788];
[H,w] = freqz(h,1,512);
plot(w/pi,abs(H),'LineWidth',1,'color',[1,0.5,0.5]);
grid minor
xlabel('\omega/\pi');
ylabel('Magnitude');
```

# 4 SUMMARY

For this Homework, I understand more about Digital Signal Processing, as well as how to use the MATLAB to plot and analysis series and how to smooth the signals. I also understand what will be the effect if we use the filter to smooth the signal. I also know more about the property of the filter and how to generate the system to help us impliment some operation like square root.

## REFERENCES

[1] Changgang-Zheng/Signals-and-Systems/report.https://github.com/Changgang-Zheng/Signals-and-Systems

[2] Supplementary materials to the text book 'Digital Signal Processing: A Computer-Based Approach', 4th Edition. by S.K. Mitra, ISBN 0077320670. http://www.bb9.uestc.edu.cn/webapps/portal/frameset.jsp?tab_tab_group_id=_2_1&url=%2Fwebapps%2Fblackboard%2Fexecute%2Flauncher%3Ftype%3DCourse%26id%3D_13014_1%26url%3D

[3] Digital Signal Processing: A Computer-Based Approach, 4th Edition. by S.K. Mitra, ISBN 0077320670.