实验内容和要求：

1、以下程序中分别使用conv和filter函数计算h和x的卷积y和y1，运行程序，并分析y和y1是否有差别，为什么要使用x[n]补零后的x1来产生y1；具体分析当h[n]有i个值，x[n]有j个值，使用filter完成卷积功能，需要如何补零？

程序如下:

%conv函数和filter对比

clf;

h = [3 2 1 -2 1 0 -4 0 3]; % impulse response

x = [1 -2 3 -4 3 2 1]; % input sequence

y = conv(h,x);

n = 0:14;

subplot(2,1,1);

stem(n,y);

xlabel('Time index n'); ylabel('Amplitude');

title('Output Obtained by Convolution'); grid;

x1 = [x zeros(1,8)];

y1 = filter(h,1,x1);

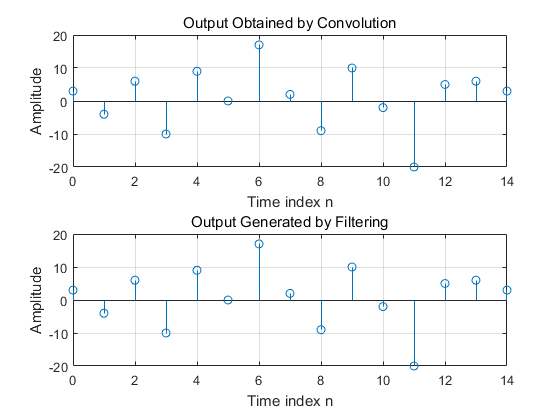
subplot(2,1,2);

stem(n,y1);

xlabel('Time index n'); ylabel('Amplitude');

title('Output Generated by Filtering'); grid;

结果:



由图可看出，y与y1并无差别。

使用x[n]补零后的x1来产生y1,是因为存在边界效应，只要脉冲响应采样点部分位于输入信号采样值之外，输出就不确定，如：

x 1 -2 3 -4 3 2 1 2

h 3 0 -4 0 1 -2 1 2 3 2 4 1

需变换成如下才能确定输出：

x 1 -2 3 -4 3 2 1 2 0 0 0 0 0 0 0 0 0 0 0

h 3 0 -4 0 1 -2 1 2 3 2 4 1

此时n＝18。

用conv函数计算能再输入序列后自动补零，而filter函数不能。

分析:

%使用filter函数进行线性卷积

% 线性卷积结果长度为i+j-1 (0~i+j-2) 至少要补11个零

%下面的代码是补充10个零的结果 可以看到无法完成全部卷积

h = [1 4 2 3 2 1 -2 1 0 -4 0 3]; % impulse response i=12

x = [1 -2 3 -4 3 2 1 2]; % input sequence j=8

n = 0:17;

x1 = [x zeros(1,10)]; %补十个零值

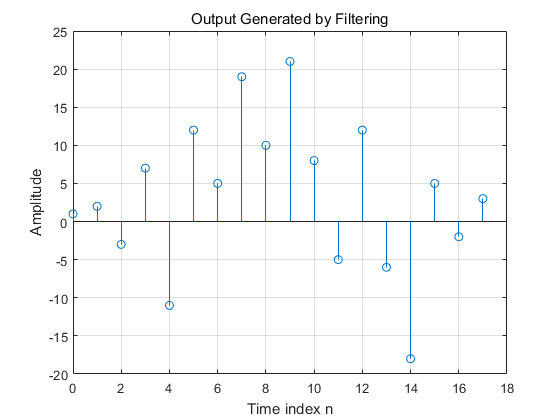
y1 = filter(h,1,x1);

stem(n,y1);

xlabel('Time index n'); ylabel('Amplitude');

title('Output Generated by Filtering');

grid



(1)

%使用filter函数进行线性卷积

% 线性卷积结果长度为i+j-1 (0~i+j-2) 至少要补11个零

%下面的代码是补充11个零的结果 可以看到刚好能够完成全部卷积

h = [1 4 2 3 2 1 -2 1 0 -4 0 3]; % impulse response i=12

x = [1 -2 3 -4 3 2 1 2]; % input sequence j=8

n = 0:18;

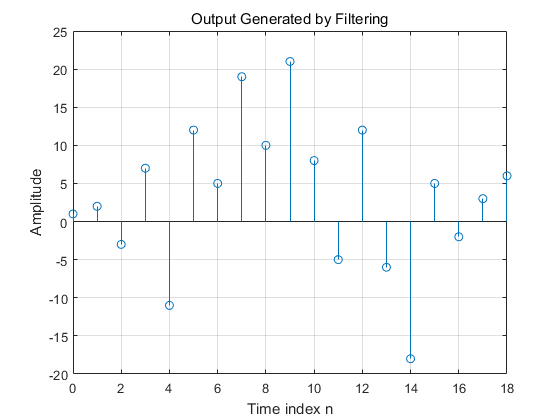
x1 = [x zeros(1,11)]; %补11个零值

y1 = filter(h,1,x1);

stem(n,y1);

xlabel('Time index n'); ylabel('Amplitude');

title('Output Generated by Filtering'); grid



(2)

%使用filter函数进行线性卷积

% 线性卷积结果长度为i+j-1 (0~i+j-2) 至少要补11个零

%下面的代码是补充23个零的结果 可以看到能够完成全部卷积,但是多了一些零

h = [1 4 2 3 2 1 -2 1 0 -4 0 3]; % impulse response i=12

x = [1 -2 3 -4 3 2 1 2]; % input sequence j=8

n = 0:30;

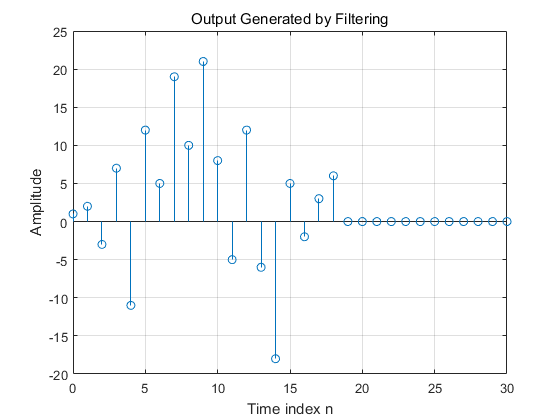
x1 = [x zeros(1,23)]; %补23个零值

y1 = filter(h,1,x1);

stem(n,y1);

xlabel('Time index n'); ylabel('Amplitude');

title('Output Generated by Filtering'); grid;



(3)

对照（1）、（2）图，当n＝18时两图有区别，(2)图能完全卷积，当补零数少于j－1，就不能完全卷积。

对照（2）、（3）图可知，不零数可大于 j－1，须满足n的长度与补零后x1的长度相等。

h[n]有i个值，x[n]有j个值, 以n为x轴，n=0:N，使用x[n]补零后的x1来产生y1，由上述图可知，要完全卷积，x1至少需补j－1个零值。其中N>=(i＋j－1)-1， n的长度与补零后x1的长度相等，若x1中补a个零值（a>=j-1），则N＝i＋a－1。

2编制程序求解下列两个系统的单位冲激响应和阶跃响应，并绘出其图形。要求分别用 filter、conv、impz三种函数完成。





给出理论计算结果和程序计算结果并讨论。

程序计算结果:

I.

****

a. 单位冲激响应：

% y[n]+0.75y[n-1]+0.125y[n-2]=x[n]-x[n-1]

%单位冲击响应 filter函数

% 这个函数一步到位 直接把x序列仿真到输出

%相当于 先进行计算该系统的单位脉冲响应,然后计算输入序列与

%单位脉冲响应的卷积

a1=[1,0.75,0.125];% y[n]系数

b1=[1,-1];%x[n]系数

n=0:20;

x1=[1 zeros(1,20)];%单位抽样序列

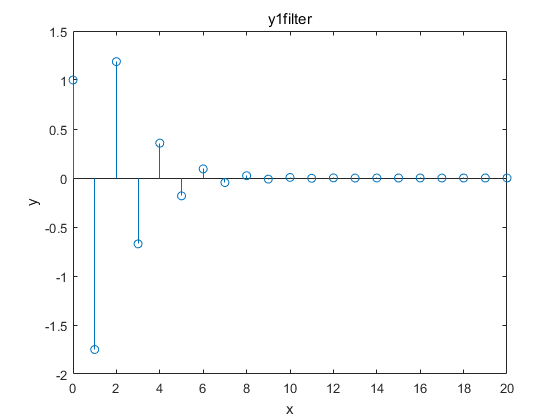
y1filter=filter(b1,a1,x1);

stem(n,y1filter);

title('y1filter');

xlabel('x');

ylabel('y');



（2）用conv函数

% y[n]+0.75y[n-1]+0.125y[n-2]=x[n]-x[n-1]

%单位冲击响应 用conv函数

%需要先用impz()函数计算系统的单位脉冲响应

%再使用conv()函数将输入序列与系统单位进行卷积

a1=[1,0.75,0.125];% y[n]系数

b1=[1,-1];%x[n]系数

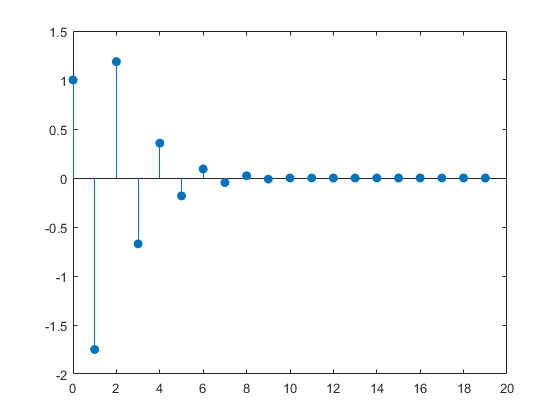
x1=[1 zeros(1,10)];%单位抽样序列

[h]=impz(b1,a1,10);%得到10个长度的单位脉冲响应

y1conv=conv(h,x1);%进行卷积

n=0:19;%卷积长度为:10+11-1共20个

stem(n,y1conv,'filled')



（3）用impz函数

% y[n]+0.75y[n-1]+0.125y[n-2]=x[n]-x[n-1]

%用impz()函数求解单位冲击响应

%需要先用impz()函数计算系统的单位脉冲响应

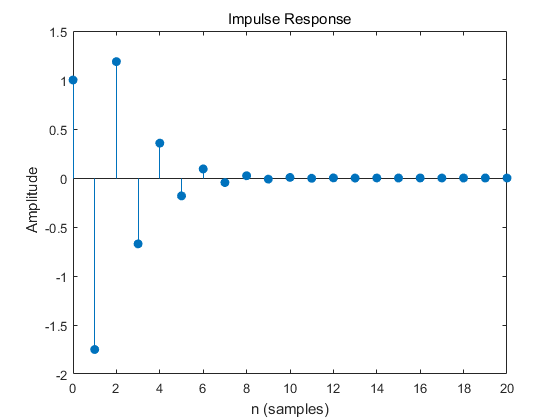
%再使用conv()函数将输入序列与系统单位进行卷积

%注意到本题只需要系统单位脉冲响应,所以可以不需要卷积

a1=[1,0.75,0.125];

b1=[1,-1];

impz(b1,a1,21);%这里改成20 则与上conv\_2\_1\_2结果一样



b. 单位阶跃响应：

(1)用filter函数

% y[n]+0.75y[n-1]+0.125y[n-2]=x[n]-x[n-1]

%单位阶跃响应 filter函数

% 这个函数一步到位 直接把x序列仿真到输出

%相当于 先进行计算该系统的单位脉冲响应,然后计算输入序列与

%单位脉冲响应的卷积

a1=[1,0.75,0.125];

b1=[1,-1];

n=0:20;

x2=ones(1,21);

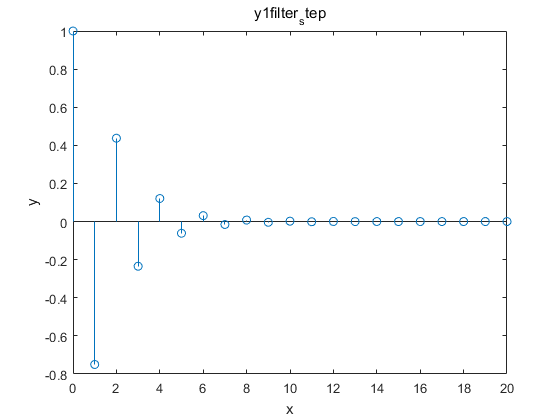
y1filter=filter(b1,a1,x2);

stem(n,y1filter);

title('y1filter\_step');

xlabel('x');

ylabel('y');



（2）用conv函数

% y[n]+0.75y[n-1]+0.125y[n-2]=x[n]-x[n-1]

%单位阶跃响应 用conv函数

%需要先用impz()函数计算系统的单位脉冲响应

%再使用conv()函数将输入序列与系统单位进行卷积

a1=[1,0.75,0.125];

b1=[1,-1];

x2=ones(1,21);

[h]=impz(b1,a1,20);%得到系统的单位脉冲响应

y1=conv(h,x2);

%本题理论上得到的卷积长度是无限长的(因为单位阶跃序列是无限长)

%为了画图(研究)方便 单位阶跃序列只取了有限值

%于是在conv()卷积的过程中 会"补0"

%卷积结果就代码来说 长度是 20+21-1 = 40 (0~39)

%则在卷积计算过程中会补 19个0 ,但是理论上单位阶跃序列全是1

%所以卷积结果后19位理论上应该是用输入序列的后19位

%即用1和系统单位脉冲响应进行卷积

%但实际上由于取有限位并补"0"的操作后

%导致后19位是在用0和系统单位脉冲响应进行卷积

%因此得到的卷积结果 应该抛弃后19位

%即只要前21位

y1conv=y1(1:21);

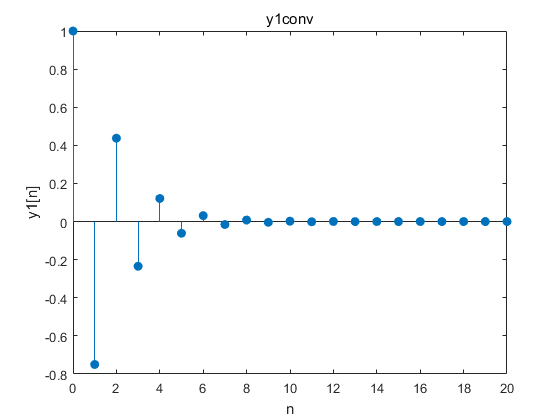
n1=0:20;

stem(n1,y1conv,'filled');

title('y1conv');

xlabel('n');

ylabel('y1[n]');



（3）用impz函数

% y[n]+0.75y[n-1]+0.125y[n-2]=x[n]-x[n-1]

%用impz()函数求解单位阶跃响应

%先将x[n]=单位阶跃序列带入

%得到y[n]+0.75y[n-1]+0.125y[n-2] = delta[n]

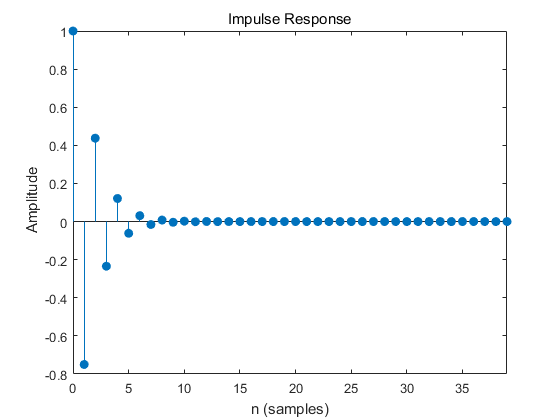
%得到的序列就是单位脉冲序列表示的结果,而impz函数求得就是单位脉冲响应

%所以此时系统的单位脉冲响应结果就是

b=1;%x前边的系数

a=[1,0.75,0.125];%y前边的系数

impz(b,a,40)



II. y[n]=0.25(x[n-1]+x[n-2]+x[n-3]+x[n-4])

a. 单位冲激响应：

% y[n]=0.25（x[n-1]+x[n-2]+x[n-3]+x[n-4]）

%单位冲击响应 filter函数

% 这个函数一步到位 直接把x序列仿真到输出

%相当于 先进行计算该系统的单位脉冲响应,然后计算输入序列与

%单位脉冲响应的卷积

a2=1;

b2=[0 0.25\*ones(1,4)];

n=0:9;

x1=[1 zeros(1,9)];

y2filter=filter(b2,a2,x1);

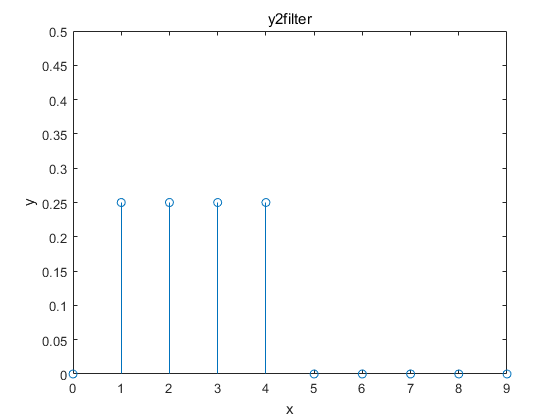
stem(n,y2filter);

title('y2filter');

xlabel('x');

ylabel('y')

set(gca,'YLim',[0 .5]);%Y轴的数据显示范围



（2）用conv函数

% y[n]=0.25（x[n-1]+x[n-2]+x[n-3]+x[n-4]）

%单位冲击响应 用conv函数

%需要先用impz()函数计算系统的单位脉冲响应

%再使用conv()函数将输入序列与系统单位进行卷积

a2=1;

b2=[0 0.25\*ones(1,4)];

x1=[1 zeros(1,5)];

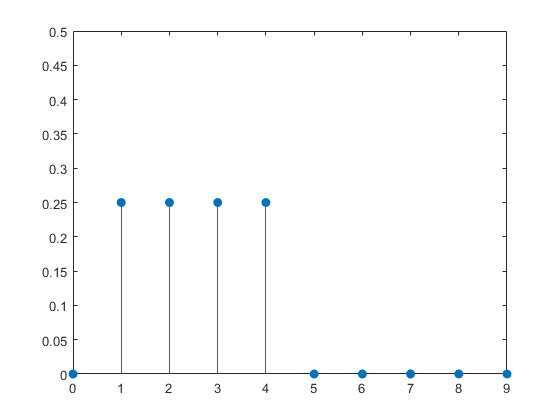
[h]=impz(b2,a2,5);

y2conv=conv(h,x1);

n=0:9;

stem(n,y2conv,'filled')

set(gca,'YLim',[0 .5]);%Y轴的数据显示范围



（3）用impz函数

% y[n]=0.25（x[n-1]+x[n-2]+x[n-3]+x[n-4]）

%单位冲击响应 用impz函数

%需要先用impz()函数计算系统的单位脉冲响应

%再使用conv()函数将输入序列与系统单位进行卷积

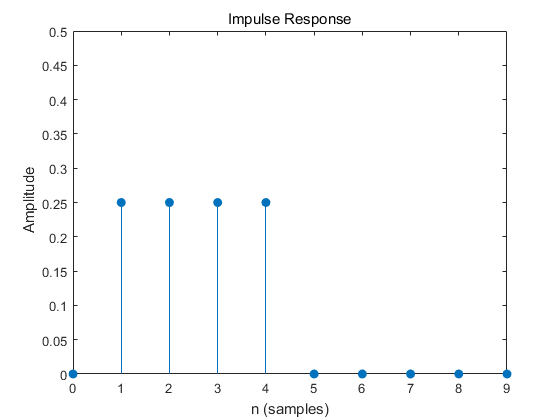
%注意到本题只需要系统单位脉冲响应,所以可以不需要卷积

b2=[0 0.25\*ones(1,4)];%x前边的系数

a2=1;

impz(b2,a2,10);

set(gca,'YLim',[0 .5]);%Y轴的数据显示范围



b. 单位阶跃响应：

(1)用filter函数

% y[n]=0.25（x[n-1]+x[n-2]+x[n-3]+x[n-4]）

%求单位阶跃响应 filter函数

% 这个函数一步到位 直接把x序列仿真到输出

%相当于 先进行计算该系统的单位脉冲响应,然后计算输入序列与

%单位脉冲响应的卷积

a2=1;

b2=[0 0.25\*ones(1,4)];

n=0:20;

x2=ones(1,21);

y2filter=filter(b2,a2,x2);

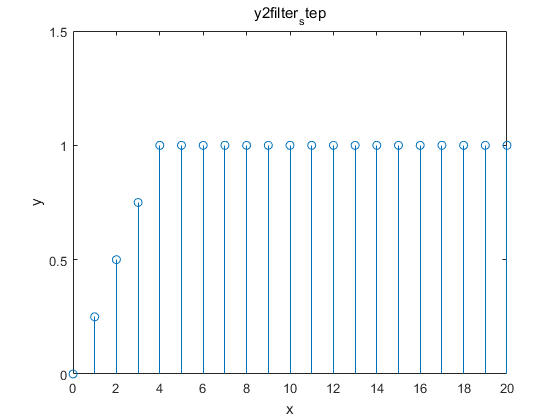
stem(n,y2filter);

title('y2filter\_step');

xlabel('x');

ylabel('y')

set(gca,'YLim',[0 1.5]);%Y轴的数据显示范围



（2）用conv函数

% y[n]=0.25（x[n-1]+x[n-2]+x[n-3]+x[n-4]）

%求单位阶跃响应 用conv函数

%需要先用impz()函数计算系统的单位脉冲响应

%再使用conv()函数将输入序列与系统单位进行卷积

h=[0 0.25\*ones(1,4)];

x2=ones(1,21);

n=0:20;

y2=conv(h,x2);

%本题理论上得到的卷积长度是无限长的(因为单位阶跃序列是无限长)

%为了画图(研究)方便 单位阶跃序列只取了有限值

%于是在conv()卷积的过程中 会"补0"

%卷积结果就代码来说 长度是 20+21-1 = 40 (0~39)

%则在卷积计算过程中会补 19个0 ,但是理论上单位阶跃序列全是1

%所以卷积结果后19位理论上应该是用输入序列的后19位

%即用1和系统单位脉冲响应进行卷积

%但实际上由于取有限位并补"0"的操作后

%导致后19位是在用0和系统单位脉冲响应进行卷积

%因此得到的卷积结果 应该抛弃后19位

%即只要前21位

y2conv=y2(1:21);

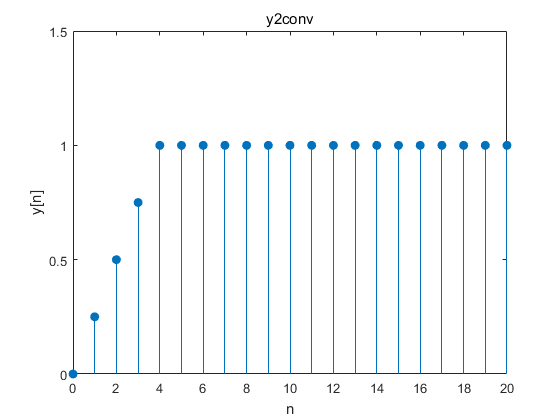
stem(n,y2conv,'filled');

title('y2conv');

xlabel('n');

ylabel('y[n]')

set(gca,'YLim',[0 1.5]);%Y轴的数据显示范围



（3）用impz函数

% y[n]=0.25（x[n-1]+x[n-2]+x[n-3]+x[n-4]）

%用impz()函数求解系统单位阶跃响应

%先将x[n]=单位阶跃序列带入

%得到S[n]=y[n] = 0.25\*delta[n-1]+0.5\*delta[n-2]+0.75\*delta[n-3] + u[n-4]

%得到的序列就是单位脉冲序列表示的结果,而impz函数求得就是单位脉冲响应

%所以此时系统的单位脉冲响应结果就是

b=[0,0.25,0.5,0.75,ones(1,17)];

a=1;

impz(b,a,21)

set(gca,'YLim',[0 1.5]);%Y轴的数据显示范围

