**Lab 2：Convolution and LTI system**

|  |  |
| --- | --- |
| **Author** | Name：秦庆福 宣逸凡 Student ID: 11910103 11910101 |
| **Introduction**  Based on the exercises of 2.4 2.5 and 2.10 on the Computer Explorations in Signals and Systems Using MATLAB.  We have the following aims:   1. Use MATLAB filter() and conv() function to compute convolution sum of discrete time signals with infinite length or finite length. 2. Use MATLAB to verify the commutative, associative and distributive properties of convolution for a special set of signals. 3. Use MATLAB to compute convolution sum to verify linearity and time-invariant properties or discrete-time signals. 4. Finish exercises and strengthen the knowledge studying in the theory course.   **Lab results & Analysis**：        2.4a  Fig 2.4(a)  **Analysis(a):**  x1[n], h1[n], h2[n] are shown.  2.4b  Fig 2.4(b)  **Analysis(b):**  According to the fig, we find x1[n]\*h1[n]=h1[n]\*x1[n], so the output of conv() is the same regardless of the order of the input.    2.4c  Fig 2.4(c)  **Analysis(c):**  According to the fig, we find x1[n]\*(h1[n]+h2[n])=x1[n]\*h1[n]+x1[n]\*h2[n], so the distributive property is verified.    2.4d  Fig d.4(d)  **Analysis(d):**  According to the fig, we find (x1[n]\*h1[n])\*h2[n] =x1[n]\*(h1[n]\*h2[n]), so the associative property is verified.    2.4e  Fig 2.4(e)  **Analysis(e):**  From fig 2.4(e), we find ye1[n]=ye2[n], so that the outputs will be the same if we interchange the input and impulse response of each system.        Fig 2.4(f)  **Analysis(f):**  (f) From Fig 2.4(f) we can see that yf1[n] dose not equal to yf2[n], but the result dose not violate the associative property of convolution as system 1 is not a LTI system since if we input x1[n] = n and we let y1[n] = y[n-1] = nx[n-1], if we input x2[n] = x[n-1], and y2 will be (n+1)x2[n] = (n+1)x[n-1]. Clearly, we can see that y1[n] is not equal to y2[n], which means that the system is time-varying, so that we can’t apply associative law to it.      Fig 2.4(g)  **Analysis(g):**  (g) From the fig we got, clearly yg1[n] is not equal to yg2[n]. However, it still not violated the distributive property of convolution as system 1 here is not a LTI system. For system 1, if we take x1 = n as our first input, y1 will be n2. If we take x2 = 2x1 = 2n, then the output will be y2 = 4n2. Since y2 is not equal to two times of y1, system 1 is not linear, so we can’t apply distributive property to it.  In the last figure, if the system is invertible, for is not equal should not equal ut the figure show it is wrong.  2.5    **Lab results(a)(b)(c):**  2.5a-w  Fig2.5(a)  2.5a-y  Fig2.5(b)  2.5a-z  Fig2.5(c)  **Analysis(a)(b)(c):**   1. The fig2.5(a)(b)(c) show the w,y,z. 2. The w[n] is liner, because:   From fig2.5(a), we can see that the subplot 3 is the same as subplot 4, meaning that y[n] is a linear system.  The y[n] is nonlinear, because:  Similarly with above, the y3 isn’t equal y4,so the y[n] is nonlinear.  The z[n] is linear, because:  From fig2.5(b), we can see that the subplot 3 is the same as subplot 4, meaning that y[n] is a linear system.   1. The w[n] is time-invariant, because:   From fig2.5(a), subplot 1 and 2, we learn that w[n] is a time-invariant system, because the figure doesn’t change after translation motion.  The y[n] is time-invariant, because:  From fig2.5(a), subplot 1 and 2, we learn that w[n] is a time-invariant system, because the figure doesn’t change after translation motion.  The z[n] is time-varying, because:  From from subplot 1 and subplot 2 in fig2.5(c), we learn that z[n] is a time-varying system as the graph changes after translation motion.      **Lab results(d):**  2.5d  Fig 2.5(d)  **Analysis(d):** for h2[n], we use h2(i+1)=(3/5)^(i)\*h2(i)+x1(i+1) to do loop to calculus the h2[n].    **Lab results(e):**  2.5e  Fig 2.5(e)  **Analysis(d):**  for s2[n], we use s2(i+1)=(3/5)^(i)\*s2(i)+x2(i+1) to do loop to calculus the h2[n].  **Question(f):**    **Lab results(f):**  2.5f  Fig2.5(f)  **Analysis(f):**  After convolution process, we select the first 20 items of the results, so the result is above.  **Question(g):**    **Lab results(g):**  2.5g-1  Fig2.5(g)-1  **2.5g-2**  Fig2.5(g)-2  **Analysis(g):**  s1[n] and z1[n] are same, so the system 1 is a LTI system.  s2[n] and z2[n] are different, so the system 2 is a nonlinear system.    **Question(a)**      **Lab results(a):**  **D:\SUSTECH2020春\信号和系统\lab\HW2\2.10a.png2.10a**  Figure 2.10(a)  **Analysis(a):**  The vector he as shown in Figure 2.10(a).  **Question(b):**    **Lab results(b):**  **2.10b**  Figure 2.10(b)  **Analysis(b):**  The overall difference equation is . By using unit impulse as input, then we have the output is the same as the input. Therefore, we’ve verified that is a valid solution to the overall difference equation.  **Question(c):**    **Lab results(c):**  **2.10c**  Figure 2.10(c)  **Analysis(c):**  The unit impulse response is above.  **Question(d):**    **Lab results(d):**  **2.10d**  Figure 2.10(d)  **Analysis(d):**  Figure 2.10(d) shows us the output z without echo.  **Question(e):**    **Lab results(e):2.10e**  Figure 2.10(e)  **Analysis(e):**  From Figure 2.10e-1, we find that the result is indeed not a unit impulse response. Because the truncation effect. In theoretically, the response should be infinitely long. However, the actual length is finite. So the removed signal cause the error.  **Question(f):**      **Lab results(f):**  **2.10f-1**  Figure 2.10(f)-1  because  use autocorrelation principle, determine N = 501 (The x-coordinate of the peak value other than x=0)  Then use α=(y2(502)-z(502))/(z(1)) to calculus α.  Finally, bring N and α to the origin equation y2i=filter(1,[1,zeros(1,500),0.70],y2), and the signal without echo is the result.  2.10f-2  Fig2.10(f)-2  2.10f-3  Figure 2.10(f)-3  Like the y2,      determine the N1=751,N2=2252;  And using the same method to find that the interval of theα1 andα2. Then we average them and find that:α1=0.75 and α2=0.6.  Finally, bring N1, N2 and α1, α2 to the origin equation y3i=filter(1,[1,zeros(1,750),0.75,zeros(1,1500),0.60],y3), and the signal without echo is the result.  2.10f-4  Figure 2.10(f)-4 | |
|  | |
| **Experience**   1. We learnt about how to use MATLAB filter() and conv() to compute convolution of discrete time signals and systems. 2. We’ve gained deeper understanding about properties of convolution and its constraint: LTI system. 3. We applied Auto-correlation to echo canceling problems and managed to obtain the signal with echo removed without know the accurate parameters of the echo. 4. We now know how to set xlabel/ylabel/title of our figure into better. | |
| **Score** |  |

**Code:**

**2.4**

**(a)**

figure(1)

X1=0:9;

X2=0:4;

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

h1=[1 -1 3 0 1];

h2=[0 2 5 4 -1];

subplot(3,1,1),stem(X1,x1),grid on,xlabel('n'),ylabel('x\_1[n]'),title('x[n]');

subplot(3,1,2,'r'),stem(X2,h1),grid on,xlabel('n'),ylabel('h\_1[n]'),title('h\_1[n]');

subplot(3,1,3),stem(X2,h2),grid on,xlabel('n'),ylabel('h\_2[n]'),title('h\_2[n]');

**(b)**

y1=conv(x1,h1);

y2=conv(h1,x1);

xx=(X1(1)+X2(1)):(X1(end)+X2(end));

subplot(2,1,1),stem(xx,y1),grid on,xlabel('n'),ylabel('y[n]'),title('x\_1[n]\*h\_1[n]');

subplot(2,1,2),stem(xx,y2,'r'),grid on,xlabel('n'),ylabel('y[n]'),title('h\_1[n]\*x\_1[n]');

**(c)**

y3=conv(x1,h1)+conv(x1,h2);

y4=conv(x1,h1+h2);

subplot(2,1,1),stem(xx,y3),grid on,xlabel('n'),ylabel('y[n]'),title('y\_1[n]=x\_1[n]\*h\_1[n]+x\_1[n]\*h\_2[n]');

subplot(2,1,2),stem(xx,y4,'r'),grid on,xlabel('n'),ylabel('y[n]'),title('y\_2[n]=x\_1[n]\*(h\_1[n]+h\_2[n])');

**(d)**

xh1=conv(x1,h1);

y5=conv(xh1,h2);

h1h2=conv(h1,h2);

y6=conv(x1,h1h2);

xx2=(X1(1)+X2(1)+X2(1)):(X1(end)+X2(end)+X2(end));

subplot(2,1,1),stem(xx2,y5),grid on,xlabel('n'),ylabel('y[n]'),title('y\_{d1}[n]=(x\_1[n]\*h\_1[n])\*h\_2[n]');

subplot(2,1,2),stem(xx2,y6,'r'),grid on,xlabel('n'),ylabel('y[n]'),title('y\_{d2}[n]=x\_1[n]\*(h\_1[n]\*h\_2[n])');

**(e)**

h1=[1 -1 3 0 1 0 0 0 0];

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

h2=[0 0 1 -1 3 0 1];

x2=[0 0 ones(1,5) 0];

ye1=conv(x2,h1);

ye2=conv(x1,h2);

X1=[0:15];

stem(X1,ye1);

hold on;

stem(X1,ye2,'r-x'),grid on,xlabel('n'),ylabel('y[n]'),title('y\_{e1}[n-2]andy\_{e2}[n]');

legend('y\_{e1}[n-2]','y\_{e2}[n]=x\_1[n]\*h\_1[n-2]');

**(f)**

w=(nx+1).\*x1;

nyf1=nx(1)+nh(1):nx(end)+nh(end);

nyf2=nx(1)+nh(1)\*2:nx(end)+nh(end)\*2;

yf1=conv(w,h1);

hf1=(nh+1).\*(nh==0);

hf2=h1;

hseries=conv(hf1,hf2);

yf2=conv(x1,hseries);

figure(6)

stem(nyf1,yf1,'-rx');

hold on;

stem(nyf2,yf2,'bo');

hold off;

legend('conv by series', 'conv by impulse responses');

title('2.4 (f) "Verify Associative Property of The System"');

xlabel('n');

ylabel('y[n]');

ylim([0,20]);

print('-dpng','Lab2 2.4(f).png');

**(g)**

xg=2\*(nh==0);

yga=xg.^2;

hg2=h2;

ygb=conv(xg,hg2);

ng1=nh(1)\*2:nh(end)\*2;

yg1=[yga zeros(1,4)]+ygb;

hg1=(nh==0).^2;

hparallel=hg1+hg2;

yg2=conv(xg,hparallel);

ng2=ng1;

stem(ng1,yg1,'-rx');

hold on;

stem(ng2,yg2,'bo');

hold off;

legend('parallel connection', 'sum up impulse responses');

title('2.4 (g) "Verify Distributive Property of The System"');

xlabel('n');

ylabel('y[n]');

xlim([-1,8])

ylim([0,11]);

print('-dpng','Lab2 2.4(g).png');

**2.5**

**(a)**

clc;

clear;

figure(1);

X=[0:5];

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

x2=[0 1 zeros(1,4)];

x3=[1 2 zeros(1,4)];

x4=[0 0 1 0 0 0];

x5=[0 0 0 1 0 0];

x6=[0 1 2 0 0 0];

x7=[0 0 1 2 0 0];

w1=x1-x2-x4;

w2=x2-x4-x5;

w3=x3-x6-x7;

w4=w1+2\*w2;

subplot(4,1,1),stem(X,w1),grid on,xlabel('n'),ylabel('w\_1[n]'),title('w\_1[n]');

subplot(4,1,2),stem(X,w2),grid on,xlabel('n'),ylabel('w\_2[n]'),title('w\_2[n]');

subplot(4,1,3),stem(X,w3),grid on,xlabel('n'),ylabel('w\_3[n]'),title('w\_3[n]');

subplot(4,1,4),stem(X,w4),grid on,xlabel('n'),ylabel('w\_1[n]+2\*w\_2[n]'),title('w\_1[n]+2\*w\_2[n]');

figure(2);

y1=cos(x1);

y2=cos(x2);

y3=cos(x3);

y4=y1+2\*y2;

subplot(4,1,1),stem(X,y1),grid on,xlabel('n'),ylabel('y\_1[n]'),title('y\_1[n]');

subplot(4,1,2),stem(X,y2),grid on,xlabel('n'),ylabel('y\_2[n]'),title('y\_1[n]');

subplot(4,1,3),stem(X,y3),grid on,xlabel('n'),ylabel('y\_3[n]'),title('y\_1[n]');

subplot(4,1,4),stem(X,y4),grid on,xlabel('n'),ylabel('y\_1[n]+2\*y\_2[n]'),title('y\_1[n]+2\*y\_2[n]');

figure(3);

z1=X.\*x1;

z2=X.\*x2;

z3=X.\*x3;

z4=z1+2\*z2;

subplot(4,1,1),stem(X,z1),grid on,xlabel('n'),ylabel('z\_1[n]'),title('z\_1[n]');

subplot(4,1,2),stem(X,z2),grid on,xlabel('n'),ylabel('z\_2[n]'),title('z\_2[n]');

subplot(4,1,3),stem(X,z3),grid on,xlabel('n'),ylabel('z\_3[n]'),title('z\_3[n]');

subplot(4,1,4),stem(X,z4),grid on,xlabel('n'),ylabel('z\_1[n]+2\*z\_2[n]'),title('z\_1[n]+2\*z\_2[n]');

**(d)**

X=[0:19];

x1=[1 zeros(1,19)];

A=[1 -0.6];

B=[1];

h1=filter(B,A,x1);

h2(1)=1;

for i=1:19

h2(i+1)=(3/5)^(i)\*h2(i)+x1(i+1);

end;

subplot(2,1,1),stem(X,h1),grid on,xlabel('n'),ylabel('h\_1[n]'),title('h\_1[n]');

grid on;

subplot(2,1,2),stem(X,h2),grid on,xlabel('n'),ylabel('h\_2[n]'),title('h\_2[n]');

grid on;

**(e)**

x2=[ones(1,20)];

s1=filter(B,A,x2);

s2(1)=1;

for i=1:19

s2(i+1)=(3/5)^(i)\*s2(i)+x2(i+1);

end;

subplot(2,1,1),stem(X,s1),grid on,xlabel('n'),ylabel('s\_1[n]'),title('s\_1[n]');

grid on;

subplot(2,1,2),stem(X,s2),grid on,xlabel('n'),ylabel('s\_2[n]'),title('s\_2[n]');

grid on;

**(f)**

u=[ones(1,20)];

nu=0:19;

nv=0:19;

nw=(nu(1)+nv(1)):(nu(end)+nv(end));

z1=conv(h1,u);

z2=conv(h2,u);

for k=1:20

Z1(k)=z1(k);

Z2(k)=z2(k);

end

subplot(2,1,1),stem(nu,Z1),grid on,xlabel('n'),ylabel('z\_1[n]'),title('z\_1[n]');

subplot(2,1,2),stem(nu,Z2),grid on,xlabel('n'),ylabel('z\_2[n]'),title('z\_2[n]');

**(g)**

figure(4);

stem(nu,Z1);

hold on;

stem(X,s1,'r-v'),grid on,xlabel('n'),title('s\_1[n]andz\_1[n]');

legend('z\_1[n]','s\_1[n]');

grid on;

figure(5);

stem(nu,Z2);

hold on;

stem(X,s2,'r-<'),grid on,xlabel('n'),title('s\_2[n]andz\_2[n]');

legend('z\_2[n]','s\_2[n]');

grid on;

**2.10**

**(a)**

X=[0:4000];

B=[1 zeros(1,999) 0.5 zeros(1,3000)];

A=[1];

x1=[1 zeros(1,4000)];

he=filter(B,A,x1);

stem(X,he),grid on,xlabel('n'),ylabel('he'),title('The impulse response of the system');

**(b)**

B=[1];

A=[1 zeros(1,999) 0.5];

z=filter(B,A,he);

stem(X,z),grid on,xlabel('n'),ylabel('z[n]'),title('The impulse response of overall difference equation');

**(c)**

X3=[0:4000];

d=[1 zeros(1,4000)];

A=[1 zeros(1,999) 0.5];

B=[1];

her=filter(B,A,d);

stem(X3,her),grid on,xlabel('n'),ylabel('her'),title('her');

**(d)**

z=filter(B,A,y);

sound(z,8192);

plot(z),grid on,xlabel('n'),ylabel('z[n]'),title('the signal without echo');

**(e)**

hoa=conv(he,her);

stem(hoa),grid on,xlabel('n'),ylabel('hoa'),title('hoa')

%截断效应

**(f)**

clc;

clear;

clf;

load lineup.mat;

%sound(y,8192);

A=[1 zeros(1,999) 0.5];

B=[1];

z=filter(B,A,y);

y2\_t=flipud(y2);

NX=length(y2);

Ryy2=conv(y2,y2\_t);

plot([-NX+1:NX-1],Ryy2),title('Autp cprrelation of y2'),xlabel('n'),legend('Ryy2');

grid on;

%求出Ryy

figure(2);

[max1,index1]=max(Ryy2);

[max2,index2]=max(Ryy2(1:(index1-10)));

N2=index1-index2;

a=(y2(N2+1)-z(N2+1))/z(1);%求出α

B1=[1];

A1=[1 zeros(1,N2-1) a];

y2i=filter(B1,A1,y2);

plot(y2i,'--x'),grid on,legend('y\_2i');

hold on;

%sound(y2i,8192);

figure(3);

y3\_t=flipud(y3);

NX=length(y3);

Ryy3=conv(y3,y3\_t);

plot([-NX+1:NX-1],Ryy3),title('Autp cprrelation of y3'),xlabel('n'),legend('Ryy3');

grid on;

%求出Ryy3

figure(4);

[max31,index31]=max(Ryy3);

[max32,index32]=max(Ryy3(1:(index31-10)));

[max33,index33]=max(Ryy3(1:(index32-10)));

N31=index31-index32;

N32=index31-index33;

N32+1

a31=(y3(N31+1)-z(N31+1))/z(1);

y33=filter(1,[1 zeros(1,N31-1) a31],y3);

a32=(y33(N32+1)-z(N32+1))/z(1);

B3=[1];

A3=[1 zeros(1,N31-1) a31 zeros(1,N32-N31-1) 0.6];

y3i=filter(B3,A3,y3);

plot(y3i,'--x'),grid on,legend('y\_3i');

%sound(y3i,8192);