

Scilab Textbook Companion for
Signals And Systems
by V. Krishnaveni And A. Rajeswari¹

Created by
S D Vinodhini
Bachelor of Engineering
Electronics Engineering
PSG College of Technology
College Teacher
V. Krishnaveni
Cross-Checked by
Bhavani

July 31, 2019

¹Funded by a grant from the National Mission on Education through ICT, <http://spoken-tutorial.org/NMEICT-Intro>. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website <http://scilab.in>

Book Description

Title: Signals And Systems

Author: V. Krishnaveni And A. Rajeswari

Publisher: Wiley India Private Limited, New Delhi-110 002

Edition: 1

Year: 2012

ISBN: 9788126522897

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes	4
1 Overview of Signals and systems	5
2 Continuoustime and discretetime signals	11
3 continuoustime and discretetime systems	15
4 Linear Time Invariant Systems	23
5 Fourier Analysis of Continuoustime signals and systems	41
6 Sampling	63
7 Fourier Analysis of discretetime signals and systems	70

List of Scilab Codes

Exa 1.1	Plotting a dc signal	5
Exa 1.2	plotting basic signals	5
Exa 1.3	Plotting Basic signals	6
Exa 1.4	Plotting basic signals	6
Exa 1.5	Plotting basic signals	7
Exa 1.6	Plotting basic signals	8
Exa 1.7	Plotting basic signals	9
Exa 1.8	Plotting basic signals	9
Exa 1.10	Plotting basic signals	9
Exa 2.1	Signal Operations	11
Exa 2.2	Signal operations	12
Exa 2.3	Signal operations	14
Exa 3.10	properties of a system	15
Exa 3.11	properties of a system	16
Exa 3.13	properties of a system	17
Exa 3.14	properties of a system	18
Exa 3.15	properties of a system	20
Exa 3.16	properties of a system	20
Exa 3.17	properties of a system	21
Exa 3.18	properties of a system	22
Exa 4.1	Convolution	23
Exa 4.2	Convolution	24
Exa 4.5	Convolution	24
Exa 4.6	convolution	25
Exa 4.10	Convolution	26
Exa 4.11	Convolution	26
Exa 4.13	Convolution	27
Exa 4.14	Convolution	28

Exa 4.15	Convolution	29
Exa 4.16	convolution	30
Exa 4.17	Convolution	30
Exa 4.18	Convolution	31
Exa 4.19	Convolution	32
Exa 4.20	Convolution	33
Exa 4.21	Convolution	34
Exa 4.23	Convolution	35
Exa 4.24	Convolution	36
Exa 4.26	Convolution	37
Exa 4.27	Convolution	39
Exa 4.30	Convolution	39
Exa 5.1	Fourier Series representation	41
Exa 5.2	Fourier Series representation	43
Exa 5.3	Fourier series representation	44
Exa 5.4	Fourier series representation	45
Exa 5.5	Fourier Series Coefficients	46
Exa 5.6	Fourier series Coefficients	48
Exa 5.7	Fourier series coefficients of halfwave rectified signal	48
Exa 5.8	Fourier series coefficients	50
Exa 5.14	Continuoustime Fourier Transform	51
Exa 5.15	Continuoustime Fourier Transform	52
Exa 5.16	Continuoustime Fourier Transform	53
Exa 5.17	Continuoustime Fourier Transform	54
Exa 5.18	Inverse Fourier transform	55
Exa 5.19	Continuoustime Fourier Transform	56
Exa 5.20	Inverse Fourier transform	57
Exa 5.21	Inverse Fourier Transform	58
Exa 5.22	Inverse fourier transform	58
Exa 5.24	Fourier Transform of periodic sinusoid	59
Exa 5.25	Fourier Transform of periodic signal	60
Exa 5.32	Fourier transform of impulse train	60
Exa 5.37	Frequency response of the system	61
Exa 6.1	Sampling	63
Exa 6.2	Sampling	64
Exa 6.3	Sampling	64
Exa 6.4	Sampling	65

Exa 6.6	Sampling	66
Exa 6.7	Sampling	67
Exa 7.3	Fourier series representation of DT signal . .	70
Exa 7.4	Fourier series representation of DT signal . .	71
Exa 7.5	Fourier series representation of DT signal . .	72
Exa 7.6	Fourier series representation of DT signal . .	73
Exa 7.7	Fourier series representation of DT signal . .	73
Exa 7.8	Fourier series representation of DT signal . .	74
Exa 7.9	Fourier series representation of DT signal . .	75
Exa 7.10	DTFSrepresentation	76
Exa 7.16	Discretetime fourier transform	77
Exa 7.17	Discretetime fourier transform	78
Exa 7.18	Discretetime fourier transform	79
Exa 7.19	Discretetime fourier transform	80
Exa 7.24	Fourier transform	81
Exa 7.26	Discretetime fourier transform	82

Chapter 1

Overview of Signals and systems

Scilab code Exa 1.1 Plotting a dc signal

```
1 clf
2 clc
3 clear
4 t=[-20:0.01:20];
5 for i=1:length(t)
6     x(i)=2;
7 end
8 plot(t,x);
9 xtitle("x(t)=2 for all t","time","amplitude");
10 xgrid(5)
```

Scilab code Exa 1.2 plotting basic signals

```
1 clf
2 clear
3 clc
```



```

4 t=[-20:0.01:20];
5 for i=1:length(t)
6     x=2*t;
7 end
8 plot(t,x);
9 xtitle("x(t)=2*t for all t","time","amplitude");
10 xgrid(5)

```

Scilab code Exa 1.3 Plotting Basic signals

```

1 clc
2 clear
3 clf
4 interval=input('enter the value of time interval T
    between two samples');
5 t=(-20*interval):interval:(20*interval);
6 for i=1:length(t)
7     if t(i)<0 then
8         x(i)=-1;
9     elseif t(i)>0 then
10        x(i)=1;
11    else
12        x(i)=0;
13    end
14 end
15 plot(t,x,".");
16 xtitle("x(t)=1 for positive values of t..., x(t)=0
    for t=0...., x(t)=-1 for negative values of t","
    time","amplitude");
17 xgrid(5)

```

Scilab code Exa 1.4 Plotting basic signals

```

1 clear
2 clf
3 clc
4 t=-20:0.01:20;
5 for i=1:length(t)
6     if t(i)>0 then
7         x1(i)=0.5;
8     else
9         x1(i)=-0.5;
10    end
11 end
12 subplot(3,1,1)
13 plot(t,x1);
14 xtitle("x1(t)=-0.5 for t<0 and x1(t)=0.5 for t>0", "
    time", "amplitude");
15 xgrid(5);
16 subplot(3,1,2)
17 for i=1:length(t)
18     x2(i)=-t(i);
19 end
20 plot(t,x2);
21 xtitle("x2(t)=-t for all t", "time", "amplitude");
22 xgrid(5);
23 subplot(3,1,3)
24 for i=1:length(t)
25     x3(i)=t(i).^2;
26 end
27 plot(t,x3);
28 xtitle("x3(t)=t^2 for all t", "time", "amplitude");
29 xgrid(5);

```

Scilab code Exa 1.5 Plotting basic signals

```

1 clear
2 clf

```

```

3  clc
4  n=-20:1:20;
5  for i=1:length(n)
6      if n(i)>=0 then
7          x(i)=2;
8      else
9          x(i)=0;
10     end
11 end
12 plot(n,x,".");
13 xtitle("x(n)=0 for n<0 and x(n)=2 for n>=0","number
        of samples","amplitude");
14 xgrid(5)

```

Scilab code Exa 1.6 Plotting basic signals

```

1  clc
2  clear
3  clf
4  n1=-2:1:2;
5  x1=-2:1:2;
6  subplot(3,1,1)
7  plot(n1,x1,".");
8  xtitle("x1(n)","n","x1(n)");
9  xgrid(5)
10 n=-5:1:5;
11 for i=1:length(n)
12     x2(i)=n(i);
13     x3(i)=2-n(i);
14 end
15 subplot(3,1,2);
16 plot(n,x2,".");
17 xtitle("x2(n)","n","x2(n)");
18 xgrid(5);
19 subplot(3,1,3);

```

```
20 plot(n,x3, ".");
21 xtitle("x3(n)", "n", "x3(n)");
22 xgrid(5);
```

Scilab code Exa 1.7 Plotting basic signals

```
1 clear
2 clf
3 clc
4 interval=input('enter the sampling interval');
5 n=[-20:1:20];
6 t=n*interval
7 for i=1:length(t)
8     x(i)=2*t(i);
9 end
10 plot(t,x, ".");
11 xtitle("sampled function of x(t)=2*t for all t",
    "number of samples", "amplitude");
```

Scilab code Exa 1.8 Plotting basic signals

```
1 x=poly([-4 2 1], 't', 'c')
2 a=horner(x,0)
3 b=horner(x,-2)
4 disp(a)
5 disp(b)
```

Scilab code Exa 1.10 Plotting basic signals

```
1 clear
```

```
2 clf
3 clc
4 n=-20:1:20;
5 for i=1:length(n)
6     x(i)=0.5;
7 end
8 subplot(2,1,1)
9 plot(n,x,".");
10 xtitle("x(n)=0.5 for all n","number of samples",
        "amplitude");
11 xgrid(5)
12 y=0.5*x;
13 subplot(2,1,2)
14 plot(n,y,".");
15 xtitle("y(n)=0.5*x(n) for all n","number of samples"
        ,"amplitude");
16 xgrid(5)
```

Chapter 2

Continuoustime and discretetime signals

Scilab code Exa 2.1 Signal Operations

```
1 //Example 2.1
2 clf
3 clear
4 clc
5 t=[-10:0.01:10];
6 for i=1:length(t)
7     if t(i)>= -0.5 & t(i)<= 0.5 then
8         x(i)=t(i)+0.5;
9     elseif t(i)>0.5 & t(i)<=1.5 then
10        x(i)=1.5-t(i);
11    else
12        x(i)=0;
13    end
14 end
15 subplot(3,1,1);
16 plot2d(t,x,rect=[-4 0 4 2]);
17 xtitle("x(t) vs t", "t in sec", "x(t)");
18 subplot(3,1,2);
19 plot2d(t-1,x,rect=[-4 0 4 2]);
```

```

20 xtitle("x(t+1) vs t", "t in sec", "x(t+1)");
21 subplot(3,1,3);
22 plot2d(t+2,x,rect=[-4 0 4 2]);
23 xtitle("x(t-2) vs t", "t in sec", "x(t-2)");
24 xset('window',1);
25 subplot(3,1,1);
26 plot2d(-t,x,rect=[-4 0 4 2]);
27 xtitle("x(-t) vs t", "t in sec", "x(-t)");
28 subplot(3,1,2);
29 plot2d(t/2,x,rect=[-4 0 4 2]);
30 xtitle("x(2t) vs t", "t in sec", "x(2t)");
31 subplot(3,1,3);
32 plot2d(t*2,x,rect=[-4 0 4 2]);
33 xtitle("x(t/2) vs t", "t in sec", "x(t/2)");
34 xset('window',2);
35 subplot(3,1,1);
36 plot2d(-t-1,x,rect=[-4 0 4 2]);
37 xtitle("x(-t+1) vs t", "t in sec", "x(-t+1)");
38 subplot(3,1,2);
39 plot2d(-t+2,x,rect=[-4 0 4 2]);
40 xtitle("x(-t-2) vs t", "t in sec", "x(-t-2)");
41 subplot(3,1,3);
42 plot2d(-t/2,x,rect=[-4 0 4 2]);
43 xtitle("x(-2t) vs t", "t in sec", "x(-2t)");
44 xset('window',3);
45 subplot(3,1,1);
46 plot2d(-t*2,x,rect=[-4 0 4 2]);
47 xtitle("x(-t/2) vs t", "t in sec", "x(-t/2)");
48 subplot(3,1,2);
49 plot2d(-(t-1)/2,x,rect=[-4 0 4 2]);
50 xtitle("x(-2t+1) vs t", "t in sec", "x(-2t+1)");
51 subplot(3,1,3);
52 plot2d(-(t+2)/2,x,rect=[-4 0 4 2]);
53 xtitle("x(-2t-2) vs t", "t in sec", "x(-2t-2)");

```

Scilab code Exa 2.2 Signal operations

```
1 //Example 2.2
2 clf
3 clear
4 clc
5 t=[-10:0.01:10];
6 for i=1:length(t)
7     if t(i)>= -2 & t(i)<=4 then
8         x(i)=(t(i)+2)/6;
9     else
10        x(i)=0;
11    end
12 end
13 subplot(3,1,1);
14 plot2d(t,x,rect=[-10 0 10 2]);
15 xtitle("x(t) vs t","t in sec","x(t)");
16 subplot(3,1,2);
17 plot2d(t-1,x,rect=[-10 0 10 2]);
18 xtitle("x(t+1) vs t","t in sec","x(t+1)");
19 subplot(3,1,3);
20 plot2d(t+1,x,rect=[-10 0 10 2]);
21 xtitle("x(t-1) vs t","t in sec","x(t-1)");
22 xset('window',1);
23 subplot(3,1,1);
24 plot2d(t/2,x,rect=[-10 0 10 2]);
25 xtitle("x(2t) vs t","t in sec","x(2t)");
26 subplot(3,1,2);
27 plot2d(2*t,x,rect=[-10 0 10 2]);
28 xtitle("x(t/2) vs t","t in sec","x(t/2)");
29 subplot(3,1,3);
30 plot2d(-t/3,x,rect=[-10 0 10 2]);
31 xtitle("x(-3t) vs t","t in sec","x(-3t)");
32 xset('window',2);
33 subplot(3,1,1);
34 plot2d(-(t-3),x,rect=[-10 0 10 2]);
35 xtitle("x(3-t) vs t","t in sec","x(3-t)");
36 subplot(3,1,2);
```



```
37 plot2d(-(t-2)/2,x,rect=[-10 0 10 2]);
38 xtitle("x(-2t+2) vs t","t in sec","x(-2t+2)");
```

Scilab code Exa 2.3 Signal operations

```
1 //Example 2.3
2 clear
3 clc
4 clf
5 n=-20:1:20;
6 x=[zeros(1,19),1,1,2,3,4,0.5,zeros(1,16)];
7 subplot(3,1,1);
8 plot(n,x,".");
9 xtitle("x(n) vs n","n","x(n)");
10 subplot(3,1,2);
11 plot(n+3,x,".");
12 xtitle("x(n-3) vs n","n","x(n-3)");
13 subplot(3,1,3);
14 plot(n-2,x,".");
15 xtitle("x(n+2) vs n","n","x(n+2)");
16 figure(1)
17 subplot(3,1,1);
18 plot(-n,x,".");
19 xtitle("x(-n) vs n","n","x(-n)");
20 subplot(3,1,2);
21 plot(-n+2,x,".");
22 xtitle("x(-n+2) vs n","n","x(-n+2)");
23 subplot(3,1,3);
24 plot(-n-3,x,".");
25 xtitle("x(-n-3) vs n","n","x(-n-3)");
```

Chapter 3

continuoustime and discretetime systems

Scilab code Exa 3.10 properties of a system

```
1 //Example 3.10
2 clc
3 clear all
4 t=0:0.01:20;
5 function y=signal(x)
6     y=x
7 endfunction
8 //Assume v(t) as ramp signal
9 v1=t;
10 v2=t+2;
11 //Assume R=1
12 i1=signal(v1)
13 i2=signal(v2)
14 a=2;
15 b=2;
16 subplot(4,2,1)
17 plot(t,a*v1)
18 xtitle("a*v1(t)", "t in sec", "a*v1(t)");
19 subplot(4,2,2)
```

```

20 plot(t,signal(a*v1))
21 xtitle("a*i1(t)","t in sec","a*i1(t)");
22 subplot(4,2,3)
23 plot(t,b*v2)
24 xtitle("b*v2(t)","t in sec","b*v2(t)");
25 subplot(4,2,4)
26 plot(t,signal(b*v2))
27 xtitle("b*i2(t)","t in sec","b*i2(t)");
28 c=(a*v1)+(b*v2);
29 subplot(4,2,5)
30 plot(t,c)
31 xtitle("v3(t)","t in sec","v3(t)");
32 subplot(4,2,6)
33 plot(t,signal(c))
34 xtitle("i3(t)","t in sec","i3(t)");
35 subplot(4,2,8)
36 plot(t,signal(a*v1)+signal(b*v2))
37 xtitle("LINEAR SYSTEM","t in sec","a*i1(t)+b*i2(t)")
    ;

```

Scilab code Exa 3.11 properties of a system

```

1 //Example 3.11
2 clc
3 clear all
4 t=0:0.001:0.5;
5 function i=signal(v)
6     i=exp(v);
7 endfunction
8 //Assume v(t) as ramp signal
9 x1=2*ones(1,length(t));
10 x2=t+2;
11 //Assume R=1
12 y1=signal(x1)
13 y2=signal(x2)

```

```

14 a=2;
15 b=2;
16 subplot(4,2,1)
17 plot(t,a*x1)
18 xtitle("a*x1(t)","t in sec","a*x1(t)");
19 subplot(4,2,2)
20 plot(t,signal(a*x1))
21 xtitle("a*y1(t)","t in sec","a*y1(t)");
22 subplot(4,2,3)
23 plot(t,b*x2)
24 xtitle("b*x2(t)","t in sec","b*x2(t)");
25 subplot(4,2,4)
26 plot(t,signal(b*x2))
27 xtitle("b*y2(t)","t in sec","b*y2(t)");
28 c=(a*x1)+(b*x2);
29 subplot(4,2,5)
30 plot(t,c)
31 xtitle("x3(t)","t in sec","x3(t)");
32 subplot(4,2,6)
33 plot(t,signal(c))
34 xtitle("y3(t)","t in sec","y3(t)");
35 subplot(4,2,8)
36 plot(t,signal(a*x1)+signal(b*x2))
37 xtitle("NON-LINEAR SYSTEM","t in sec","a*y1(t)+b*y2(
    t)");

```

Scilab code Exa 3.13 properties of a system

```

1 //Example 3.13
2 clear;
3 clc;
4 x1 = [1,1,1,1];
5 x2 = [2,2,2,2];
6 a = 1;
7 b = 1;

```

```

8 for n = 1:length(x1)
9     x3(n) = a*x1(n)+b*x2(n);
10 end
11 for n = 1:length(x1)
12     y1(n) = x1(n)^2;
13     y2(n) = x2(n)^2;
14     y3(n) = x3(n)^2;
15 end
16 for n = 1:length(y1)
17     z(n) = a*y1(n)+b*y2(n);
18 end
19 count = 0;
20 for n =1:length(y1)
21     if(y3(n)== z(n))
22         count = count+1;
23     end
24 end
25 if(count == length(y3))
26     disp('Since It satisfies the superposition
27         principle ')
27     disp('The given system is a Linear system')
28     y3
29     z
30 else
31     disp('Since It does not satisfy the
32         superposition principle ')
32     disp('The given system is a Non-Linear system')
33 end

```

Scilab code Exa 3.14 properties of a system

```

1 //Example 3.14
2 clear;
3 clc;
4 to = 2; //Assume the amount of time shift =2

```

```

5 T=10;
6 t=0:0.1:T;
7 for i=1:length(t)
8     if (t(i)>=0 & t(i)<1)
9         x1(i) = t(i);
10        x2(i)=0;
11        elseif (t(i)>=1 & t(i)<2) then
12            x1(i)=1;
13            x2(i)=t(i)-1;
14        elseif (t(i)>=2 & t(i)<3) then
15            x1(i)=2;
16            x2(i)=1;
17        elseif (t(i)>=3 & t(i)<4)
18            x1(i)=0;
19            x2(i)=2;
20        else
21            x1(i)=0;
22            x2(i)=0;
23        end
24 y1(i) = 2*(x1(i));
25 y2(i)=2*x2(i);
26 end
27 figure(0);
28 subplot(2,1,1);
29 plot(t,x1);
30 xtitle("x1(t) ","t","x1(t)");
31 subplot(2,1,2);
32 plot(t,y1);
33 xtitle("y1(t)=2*x1(t) ","t","y1(t)");
34 figure(1);
35 subplot(2,1,1);
36 plot(t,x2);
37 xtitle("x2(t) ","t","x2(t)");
38 subplot(2,1,2);
39 plot(t,y2);
40 xtitle("y2(t)=2*x2(t)=2*x1(t-1)=y1(t-1) ","t","y2(t)");
41 //First shift the input signal only

```

```

42 Input_shift = 2*(x1(T-to));
43 Output_shift = y1(T-to);
44 if(Input_shift == Output_shift)
45     disp('The given system is a Time In-variant system
         ');
46 else
47     disp('The given system is a Time Variant system');
48 end

```

Scilab code Exa 3.15 properties of a system

```

1 //Example 3.15
2 clear;
3 clc;
4 to = 2; //Assume the amount of time shift =2
5 T=10;
6 for t = 1:0.01:T
7     x(t) = sin(t);
8     y(t) = sin(2*t);
9 end
10 //First shift the input signal only
11 Input_shift = x(T-to);
12 Output_shift = y(T-to);
13 if(Input_shift == Output_shift)
14     disp('The given system is a Time In-variant system
         ');
15 else
16     disp('The given system is a Time Variant system');
17 end

```

Scilab code Exa 3.16 properties of a system

```

1 //Example 3.16

```

```

2 clear;
3 clc;
4 no = 2; //Assume the amount of time shift =2
5 L = 10; //Length of given signal
6 for n = 1:L
7     x(n)=sin(n);
8 end
9 n=2;
10 for i=1:L
11     y(i)=x(n-1);
12     n=n+1;
13 end
14 //First shift the input signal only
15 Input_shift = x(L-no);
16 Output_shift = y(L-no);
17 if(Input_shift == Output_shift)
18     disp('The given discrete system is a Time In-
19         variant system');
20 else
21     disp('The given discrete system is a Time Variant
22         system');
23 end

```

Scilab code Exa 3.17 properties of a system

```

1 //Example 3.17
2 clear;
3 clc;
4 no = 2; //Assume the amount of time shift =2
5 L = 10; //Length of given signal
6 for n = 1:L
7     x(n)=sin(n);
8     y(n)=-sin(n);
9 end
10

```



```
11 //First shift the input signal only
12 Input_shift = x(L-no);
13 Output_shift = y(L-no);
14 if(Input_shift == Output_shift)
15     disp('The given discrete system is a Time In-
        variant system');
16 else
17     disp('The given discrete system is a Time Variant
        system');
18 end
```

Scilab code Exa 3.18 properties of a system

```
1 //Example 3.18
2 clc
3 clear
4 t=0:0.01:10;
5 for i=1:length(t)
6     x(i)=sin(i);
7     y(i)=2*x(i);
8     z(i)=0.5*y(i);
9 end
10 if (x==z) then
11     disp("The given system is invertible");
12 else
13     disp("the Given system is non-invertible");
14 end
```

Chapter 4

Linear Time Invariant Systems

Scilab code Exa 4.1 Convolution

```
1 //Example 4.1
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n=[0 1 2];
6 n1=0:4;
7 x=[0.5 0.5 0.5];
8 h=[3 2 1];
9 y=convol(h,x)
10 disp("Convolution of x[n] and h[n] is ...")
11 disp(y)
12 subplot(3,1,1)
13 xtitle(""," ..... n","x[n]");
14 plot2d3('gnn',n,x,5);
15 subplot(3,1,2)
16 xtitle(""," ..... n","h[n]");
17 plot2d3('gnn',n,h,5);
18 subplot(3,1,3)
19 xtitle(""," ..... n","y[n]");
20 plot2d3('gnn',n1,y,5);
```

Scilab code Exa 4.2 Convolution

```
1 //Example 4.2
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n=-1:1;
6 n1=-2:2;
7 x=[0.5 0.5 0.5];
8 h=[3 2 1];
9 y=coeff(poly(h,'z','c')*poly(x,'z','c'))
10 disp("Convolution of x[n] and h[n] is ...")
11 disp(y)
```

Scilab code Exa 4.5 Convolution

```
1 //Example 4.5
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n=0:2;
6 n1=0:4;
7 x=[0.5 0.5 0.5];
8 h=[3 2 1];
9 y=coeff(poly(h,'z','c')*poly(x,'z','c'))
10 disp("Convolution of x[n] and h[n] is ...")
11 disp(y)
12 subplot(3,1,1)
13 xtitle("input signal x(n)", "..... n", "
    x[n]");
14 plot(n,x,'. ');
15 subplot(3,1,2)
```

```

16 xtitle("system response h(n)", " ..... n
    ", "h[n]");
17 plot(n,h, '. ');
18 subplot(3,1,3)
19 xtitle("output signal y(n)", "
    ..... n", "y[n]");
20 plot(n1,y, '. ');

```

Scilab code Exa 4.6 convolution

```

1 //Example 4.6
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n=-1:1;
6 n1=-2:2;
7 x=[0.5 0.5 0.5];
8 h=[3 2 1];
9 y=coeff(poly(h, 'z', 'c')*poly(x, 'z', 'c'))
10 disp("Convolution of x[n] and h[n] is ...")
11 disp(y)
12 subplot(3,1,1)
13 xtitle("input signal x(n)", " ..... n", "
    x[n]");
14 plot(n,x, '. ');
15 subplot(3,1,2)
16 xtitle("system response h(n)", " ..... n
    ", "h[n]");
17 plot(n,h, '. ');
18 subplot(3,1,3)
19 xtitle("output signal y(n)", "
    ..... n", "y[n]");
20 plot(n1,y, '. ');

```

Scilab code Exa 4.10 Convolution

```
1 //Example 4.10
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n=0:2;
6 n1=0:4;
7 x=[0.5 0.5 0.5];
8 h=[3 2 1];
9 A=[x 0 0;0 x 0; 0 0 x];
10 y=A'*h'
11 disp("Convolution of x[n] and h[n] is...")
12 disp(y)
13 subplot(3,1,1)
14 xtitle("input signal x(n)", ".....n", "
    x[n]");
15 plot(n,x, '. ');
16 subplot(3,1,2)
17 xtitle("system response h(n)", ".....n
    ", "h[n]");
18 plot(n,h, '. ');
19 subplot(3,1,3)
20 xtitle("output signal y(n)", "
    .....n", "y[n]");
21 plot(n1,y, '. ');
```

Scilab code Exa 4.11 Convolution

```
1 //Example 4.11
2 //Convolution sum of x[n] and h[n]
3 clc
```

```

4  clear
5  n=-1:1;
6  n1=-2:2;
7  x=[0.5 0.5 0.5];
8  h=[3 2 1];
9  A=[x 0 0;0 x 0; 0 0 x];
10 y=A'*h'
11 disp("Convolution of x[n] and h[n] is ...")
12 disp(y)
13 subplot(3,1,1)
14 xtitle("input signal x(n)", "..... n", "
      x[n]");
15 plot2d3('gmn',n,x,5);
16 subplot(3,1,2)
17 xtitle("system response h(n)", "..... n
      ", "h[n]");
18 plot2d3('gmn',n,h,5);
19 subplot(3,1,3)
20 xtitle("output signal y(n)", "
      ..... n", "y[n]");
21 plot2d3('gmn',n1,y,5);

```

Scilab code Exa 4.13 Convolution

```

1  //Example 4.13
2  //Convolution sum of x[n] and h[n]
3  clc
4  clear
5  n=0:100;
6  n1=0:200;
7  for i=1:length(n)
8      x(i)=n(i);
9      h(i)=1;
10 end
11 y=convol(x,h);

```

```

12 disp(y)
13 subplot(3,1,1)
14 xtitle("input signal x(n)", " ..... n", "
      x[n]");
15 plot(n,x, '. ');
16 subplot(3,1,2)
17 xtitle("system response h(n)", " ..... n
      ", "h[n]");
18 plot(n,h, '. ');
19 subplot(3,1,3)
20 xtitle("output signal y(n)", "
      ..... n", "y[n]");
21 plot(n1,y, '. ');

```

Scilab code Exa 4.14 Convolution

```

1 //Example 4.14
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n=0:100;
6 n1=0:200;
7 a=0.7//assume the constant a=0.7
8 for i=1:length(n)
9     x(i)=a^n(i);
10    h(i)=1;
11 end
12 y=convol(x,h);
13 subplot(3,1,1)
14 xtitle("input signal x(n)", " ..... n", "
      x[n]");
15 plot(n,x, '. ');
16 subplot(3,1,2)
17 xtitle("system response h(n)", " ..... n
      ", "h[n]");

```

```

18 plot(n,h, '. ');
19 subplot(3,1,3)
20 xtitle("output signal y(n)", "
    ..... n", "y[n]");
21 plot(n1,y, '. ');

```

Scilab code Exa 4.15 Convolution

```

1 //Example 4.15
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n1=-100:1:0;
6 n2=0:100;
7 n3=-100:100;
8 a=0.5//assume the constant a=0.5
9 for i=1:length(n1)
10     x(i)=a^-n1(i);
11     h(i)=a^n1(i);
12 end
13 y=convol(x,h);
14 subplot(3,1,1)
15 xtitle("input signal x(n)", " ..... n", "
    x[n]");
16 plot(n1,x, '. ');
17 subplot(3,1,2)
18 xtitle("system response h(n)", " ..... n
    ", "h[n]");
19 plot(n2,h, '. ');
20 subplot(3,1,3)
21 xtitle("output signal y(n)", "
    ..... n", "y[n]");
22 plot(n3,y, '. ');

```

Scilab code Exa 4.16 convolution

```
1 //Example 4.16
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n1=-100:-2;
6 n2=2:100;
7 n3=-98:98;
8 a=1/2//assume the constant a=1/2
9 for i=1:length(n1)
10     x(i)=a^-n1(i);
11     h(i)=1;
12 end
13 y=convol(x,h);
14 subplot(3,1,1)
15 xtitle("input signal x(n)"," ..... n",
        x[n]);
16 plot(n1,x,'. ');
17 subplot(3,1,2)
18 xtitle("system response h(n)"," ..... n
        ", "h[n]");
19 plot(n2,h,'. ');
20 subplot(3,1,3)
21 xtitle("output signal y(n)","
        ..... n", "y[n]");
22 plot(n3,y,'. ');
```

Scilab code Exa 4.17 Convolution

```
1 //Example 4.17
2 //Convolution sum of x[n] and h[n]
```

```

3  clc
4  clear
5  n1=2:12;
6  n2=4:14;
7  n3=6:26;
8  a=1/3//assume the constant a=1/3
9  for i=1:length(n1)
10     x(i)=a-n1(i);
11     h(i)=1;
12 end
13 y=convol(x,h);
14 subplot(3,1,1)
15 xtitle("input signal x(n)", " ..... n", "
    x[n]");
16 plot(n1,x, '. ');
17 subplot(3,1,2)
18 xtitle("system response h(n)", " ..... n
    ", "h[n]");
19 plot(n2,h, '. ');
20 subplot(3,1,3)
21 xtitle("output signal y(n)", "
    ..... n", "y[n]");
22 plot(n3,y, '. ');

```

Scilab code Exa 4.18 Convolution

```

1  //Example 4.18
2  //Convolution sum of x[n] and h[n]
3  clc
4  clear
5  n1=-100:0;
6  n2=0:100;
7  n3=-100:100;
8  b=0.8//assume the constant b=0.4
9  a=0.8//assume the constant a=0.8

```

```

10 for i=1:length(n1)
11     x(i)=a^n1(i);
12     h(i)=b^n1(i);
13 end
14 y=convol(x,h);
15 subplot(3,1,1)
16 xtitle("input signal x(n)", ".....n", "
    x[n]");
17 plot(n1,x, '. ');
18 subplot(3,1,2)
19 xtitle("system response h(n)", ".....n
    ", "h[n]");
20 plot(n2,h, '. ');
21 subplot(3,1,3)
22 xtitle("output signal y(n)", "
    .....n", "y[n]");
23 plot(n3,y, '. ');

```

Scilab code Exa 4.19 Convolution

```

1 //Example 4.19
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n1=0:9;
6 n2=0:100;
7 n3=0:109;
8 b=0.8//assume the constant b=0.4
9 a=0.8//assume the constant a=0.8
10 for i=1:length(n1)
11     x(i)=a^n1(i);
12 end
13 for j=1:length(n2)
14     h(j)=b^n2(j);
15 end

```

```

16 y=convol(x,h);
17 subplot(3,1,1)
18 xtitle("input signal x(n)"," ..... n",
        x[n]);
19 plot(n1,x,'. ');
20 subplot(3,1,2)
21 xtitle("system response h(n)"," ..... n
        ", "h[n]");
22 plot(n2,h,'. ');
23 subplot(3,1,3)
24 xtitle("output signal y(n)","
        ..... n", "y[n]");
25 plot(n3,y,'. ');

```

Scilab code Exa 4.20 Convolution

```

1 //Example 4.20
2 //Convolution sum of x[n] and h[n]
3 clc
4 clear
5 n1=0:5;
6 n2=0:7;
7 n3=0:12;
8 a=0.8//assume the constant a=0.8
9 for i=1:length(n1)
10     x(i)=1;
11 end
12 for j=1:length(n2)
13     h(j)=a^n2(j);
14 end
15 y=convol(x,h);
16 subplot(3,1,1)
17 xtitle("input signal x(n)"," ..... n",
        x[n]);
18 plot(n1,x,'. ');

```

```

19 subplot(3,1,2)
20 xtitle("system response h(n)", " ..... n
    ", "h[n]");
21 plot(n2,h, '. ');
22 subplot(3,1,3)
23 xtitle("output signal y(n)", "
    ..... n", "y[n]");
24 plot(n3,y, '. ');

```

Scilab code Exa 4.21 Convolution

```

1 //Example 4.21
2 //Convolution of x(t) and h(t)
3 clc
4 clear
5 t1=0:0.01:5;
6 t2=0:0.01:2;
7 t3=0:0.01:7;
8 for i=1:length(t1)
9     x(i)=1;
10 end
11 for j=1:length(t2)
12     h(j)=1;
13 end
14 y=convol(x,h);
15 subplot(3,1,1)
16 xtitle("input signal x(t)", " ..... t", "
    x[t]");
17 plot(t1,x);
18 subplot(3,1,2)
19 xtitle("system response h(t)", " ..... t
    ", "h[t]");
20 plot(t2,h);
21 subplot(3,1,3)
22 xtitle("output signal y(t)", "

```

```

..... t", "y[t]");
23 plot(t3,y);

```

Scilab code Exa 4.23 Convolution

```

1 //Example 4.23
2 //Convolution of x(t) and h(t)
3 clc
4 clear
5 t1=0:0.01:20;
6 t2=0:0.01:20;
7 t3=0:0.01:40;
8 a1=0.5; //constants a and b are equal
9 b1=0.5;
10 a2=0.8; // constants a and b are unequal
11 b2=0.3;
12 for i=1:length(t1)
13     x1(i)=exp(-a1*t1(i));
14     x2(i)=exp(-a2*t1(i));
15 end
16 for j=1:length(t2)
17     h1(j)=exp(-b1*t2(j));
18     h2(j)=exp(-b2*t2(j));
19 end
20 //case 1: a & b are equal
21 y1=convol(x1,h1);
22 subplot(3,1,1)
23 xtitle("input signal x(t)", "..... t", "
    x[t]");
24 plot(t1,x1);
25 subplot(3,1,2)
26 xtitle("system response h(t)", "..... t
    ", "h[t]");
27 plot(t2,h1);
28 subplot(3,1,3)

```

```

29 xtitle("output signal y(t)", "
    ..... t", "y[t]");
30 plot(t3,y1);
31 //case 2: a& b are unequal
32 figure(1)
33 y2=convol(x2,h2);
34 subplot(3,1,1)
35 xtitle("input signal x(t)", " ..... t", "
    x[t]");
36 plot(t1,x2);
37 subplot(3,1,2)
38 xtitle("system response h(t)", " ..... t
    ", "h[t]");
39 plot(t2,h2);
40 subplot(3,1,3)
41 xtitle("output signal y(t)", "
    ..... t", "y[t]");
42 plot(t3,y2);

```

Scilab code Exa 4.24 Convolution

```

1 //Example 4.24
2 //Convolution of x(t) and h(t)
3 clc
4 clear
5 t1=-3:0.01:10;
6 t2=1:0.01:10;
7 t3=-2:0.01:20;
8 a=0.5//assume a=0.5
9 for i=1:length(t1)
10     x(i)=exp(-a*t1(i));
11 end
12 for j=1:length(t2)
13     h(j)=exp(-a*t2(j));
14 end

```

```

15 y=convol(x,h);
16 subplot(3,1,1)
17 xtitle("input signal x(t)"," ..... t",
        x[t]);
18 plot(t1,x);
19 subplot(3,1,2)
20 xtitle("system response h(t)"," ..... t
        ", "h[t]");
21 plot(t2,h);
22 subplot(3,1,3)
23 xtitle("output signal y(t)","
        ..... t", "y[t]");
24 plot(t3,y);

```

Scilab code Exa 4.26 Convolution

```

1 //Interconnectiuiou of LTI systems
2 n=-10:10;
3 for i=1:length(n)
4     if(n(i)==0)
5         h1(i)=2;
6     else
7         h1(i)=1;
8     end
9 end
10 for i=1:length(n)
11     if(n(i)==2)
12         h2(i)=1;
13     else
14         h2(i)=0;
15     end
16 end
17 for i=1:length(n)
18     if(n(i)>=1)
19         h3(i)=1;

```



```

20     else
21         h3(i)=0;
22     end
23 end
24 for i=1:length(n)
25     if(n(i)>= -1)
26         h4(i)=1;
27     else
28         h4(i)=0;
29     end
30 end
31 for i=1:length(n)
32     h5(i)=n(i);
33     h6(i)=1;
34 end
35 h23=h2.*h3;
36 h234=h4+h23;
37 h1234=h1.*h234;
38 h56=h5.*h6;
39 h=h56+h1234;
40 x=[1 -0.5];
41 n1=[0 1];
42 y=convol(x,h);
43 n2=-10:11;
44 subplot(3,1,1)
45 xtitle("input signal x(n)", "..... n", "
    x[n]");
46 plot(n1,x, '. ');
47 subplot(3,1,2)
48 xtitle("system response h(n)", "..... n
    ", "h[n]");
49 plot(n,h, '. ');
50 subplot(3,1,3)
51 xtitle("output signal y(n)", "
    ..... n", "y[n]");
52 plot(n2,y, '. ');

```

Scilab code Exa 4.27 Convolution

```
1 //Example 4.27
2 //Interconnectiuiion of LTI systems
3 n2=0:18;
4 h1=[1 5 10 11 8 4 1];
5 h2=[1 1 zeros(1,5)];
6 h3=[1 1 zeros(1,5)];
7 a=convol(h1,h2);
8 h=convol(a,h3);
9 x=[1 -1];
10 n1=[0 1];
11 n3=0:19;
12 y=convol(x,h);
13 subplot(3,1,1)
14 xtitle("input signal x(n)", ".....n", "
    x[n]");
15 plot(n1,x, '. ');
16 subplot(3,1,2)
17 xtitle("system response h(n)", ".....n
    ", "h[n]");
18 plot(n2,h, '. ');
19 subplot(3,1,3)
20 xtitle("output signal y(n)", "
    .....n", "y[n]");
21 plot(n3,y, '. ');
```

Scilab code Exa 4.30 Convolution

```
1 //Example 4.30
2 //Cascade connection of systems
3 clc
```

```

4 clear
5 n=0:10;
6 h11=[1 -0.5];
7 for i=1:length(n)
8     h2(i)=0.5^n(i);
9     if (n(i)==0) then
10        h1(i)=1;
11    elseif n(i)==1 then
12        h1(i)=-0.5
13    else
14        h1(i)=0;
15    end
16 end
17 h=convol(h11,h2);
18 n2=0:11;
19 //assume x[n]=[1 1 1]
20 n1=0:2;
21 x=[1 1 1];
22 n3=0:13;
23 y=convol(x,h);
24 subplot(3,1,1);
25 plot(n1,x, '. ');
26 xtitle("Input Signal x[n]", "n", "x[n]")
27 subplot(3,1,2);
28 plot(n2,h, '. ');
29 xtitle("Impulse Response h[n]", "n", "h[n]")
30 subplot(3,1,3);
31 plot(n3,y, '. ');
32 xtitle("Output Signal y[n]", "n", "y[n]")
33 disp("the given system is an invertible system");

```

Chapter 5

Fourier Analysis of Continuoustime signals and systems

Scilab code Exa 5.1 Fourier Series representation

```
1
2
3
4
5
6 //Continuous Time Fourier Series Coefficients of
7 //a periodic signal  $x(t) = \sin(2*Wot)$ 
8 clear;
9 close;
10 clc;
11 t = 0:0.01:1;
12 T = 1;
13 Wo = 2*%pi/T;
14 xt = sin(2*Wo*t);
15 for k =0:4
16     C(k+1,:) = exp(-sqrt(-1)*Wo*t.*k);
17     a(k+1) = xt*C(k+1,:)/length(t);
```

```

18     if(abs(a(k+1))<=0.01)
19         a(k+1)=0;
20     end
21 end
22 a =a';
23 ak = [-a,a(2:$)]
24 for i=1:length(ak)
25     if real(ak(i))== 0 then
26         phase(i)=0;
27     else
28         if i<length(ak)/2 then
29             phase(i)= atan(imag(ak(i))/real(ak(i)));
30         else
31             phase(i)= -atan(imag(ak(i))/real(ak(i)))
32                 ;
33         end
34     end
35 disp("The fourier series coefficients are...")
36 disp(ak)
37 disp("magnitude of Fourier series coefficient")
38 disp(abs(ak))
39 disp("Phase of Fourier series coefficient in radians
40     ")
41 disp(phase)
42 n=-4:4;
43 subplot(2,1,1)
44 plot(n,abs(ak),'.');
45 xtitle("|ak|","k","|ak|");
46 subplot(2,1,2)
47 for i=1:length(n)
48     if n(i)== -2 then
49         phase(i)=3.14/2;
50     elseif n(i)== 2 then
51         phase(i)= -3.14/2;
52     else
53         phase(i)=0;
54     end
55 end

```

```

54 end
55 plot(n,phase, '. ');
56 xtitle("/_ak", "k", "/_ak");

```

Scilab code Exa 5.2 Fourier Series representation

```

1 //Continuous Time Fourier Series Coefficients of
2 //a periodic signal x(t) = cos(Wot)
3 clear;
4 close;
5 clc;
6 t = 0:0.01:1;
7 T = 1;
8 Wo = 2*%pi/T;
9 xt = cos(Wo*t);
10 x1t=cos(Wo*-t);
11 for k =0:2
12     C(k+1,:) = exp(-sqrt(-1)*Wo*t.*k);
13     a(k+1) = xt*C(k+1,:)'/length(t);
14     if(abs(a(k+1))<=0.01)
15         a(k+1)=0;
16     end
17 end
18 a =a';
19 ak = [-a,a(2:$)]
20 disp("The fourier series coefficients are...")
21 disp(ak)
22 disp("magnitude of Fourier series coefficient")
23 disp(abs(ak))
24 n=-2:2;
25 subplot(2,1,1)
26 plot(n,abs(ak), '. ');
27 xtitle("Magnitude Spectrum", "k", "|ak|");
28 if xt== x1t then
29     disp("The Given signal is even. It has no phase

```

```

        spectrum");
30 else
31 for i=1:length(ak)
32     if real(ak(i))== 0 then
33         phase(i)=0;
34     else
35         phase(i)=atan(imag(ak(i))/real(ak(i)));
36     end
37 end
38 disp("Phase of Fourier series coefficient in radians
    ")
39 disp(phase)
40 subplot(2,1,2)
41 plot(n,phase, '. ');
42 xtitle("Phase Spectrum", "k", "ak in radians");
43 end

```

Scilab code Exa 5.3 Fourier series representation

```

1 //Continuous Time Fourier Series Coefficients of
2 //a periodic signal  $x(t) = 5\cos((\pi/2)t + (\pi/6))$ 
3 clear;
4 close;
5 clc;
6 t = 0:0.01:1;
7 T = 1;
8 Wo = 2*\pi/T;
9 xt = cos((\pi/2*t) + (\pi/6))
10 x1t=cos((\pi/2*-t) + (\pi/6))
11
12 //x(t) is expanded according to Euler's theorem
13 x=5/2*(exp(i*(\pi/2*t + \pi/6)) + exp(-i*(\pi/2*t + \pi
    /6)));
14 a1=5/2*exp(i*\pi/6);
15 a_1=5/2*exp(-i*\pi/6);

```

```

16 ak=[zeros(1,5) a_1 0 a1 zeros(1,5)];
17 k=-6:6;
18 disp("The fourier series coefficients are...")
19 disp(ak)
20 disp("magnitude of Fourier series coefficient")
21 disp(abs(ak))
22 subplot(2,1,1)
23 plot(k,abs(ak),'.');
24 xtitle("Magnitude Spectrum","k","|ak|");
25 if xt== x1t then
26     disp("The Given signal is even. It has no phase
           spectrum");
27 else
28     phase=[zeros(1,5) atan(imag(a_1)/real(a_1)) 0
            atan(imag(a1)/real(a1)) zeros(1,5)];
29     disp("Phase of Fourier series coefficient in
           radians")
30     disp(phase)
31     subplot(2,1,2)
32     plot(k,phase, '. ');
33     xtitle("Phase Spectrum","k","ak in radians");
34 end

```

Scilab code Exa 5.4 Fourier series representation

```

1 //Continuous Time Fourier Series Coefficients of
2 //a periodic signal  $x(t) = 1 + \sin(6t) + \cos(4t)$ 
3 clear;
4 close;
5 clc;
6 t = 0:0.01:1;
7 xt = 1+sin(6*t)+cos(4*t);
8 x_t = 1+sin(6*-t)+cos(4*-t);
9
10 //x(t) is expanded according to Euler's theorem

```



```

11 x=1+(1/2)*exp(%i*4*t)+(1/2)*exp(-%i*4*t)+(1/(2*i))*
    exp(%i*6*t)-(1/(2*i))*exp(-%i*6*t);
12 a0=1;
13 a2=(1/2)
14 a_2=(1/2)
15 a3=(1/(2*i));
16 a_3=-(1/(2*i));
17 ak=[zeros(1,5) a_3 a_2 0 a2 a3 zeros(1,5)];
18 k=-7:7;
19 disp("The fourier series coefficients are...")
20 disp(ak)
21 disp("magnitude of Fourier series coefficient")
22 disp(abs(ak))
23 subplot(2,1,1)
24 plot(k,abs(ak),'.');
25 xtitle("Magnitude Spectrum","k","|ak|");
26 if xt== x_t then
27     disp("The Given signal is even. It has no phase
           spectrum");
28 else
29     phase=[zeros(1,6) %pi/2 0 -%pi/2 zeros(1,6)];
30     disp("Phase of Fourier series coefficient in
           radians")
31     disp(phase)
32     subplot(2,1,2)
33     plot(k,phase,'.');
34     xtitle("Phase Spectrum","k","ak in radians");
35 end

```

Scilab code Exa 5.5 Fourier Series Coefficients

```

1 //Fourier Series coefficients of the signal x(t)
2 //Assume the period of the signal T=10
3 clc
4 clear

```

```

5 close
6 T=1;
7 To=1/4;
8 //Assume the magnitude of the signal A=1
9 A=1;
10 t=-10:0.01:10;
11 for i=1:length(t)
12     if t>To & t<-To then
13         x(i)=0;
14     else
15         x(i)=A;
16     end
17 end
18
19 Wo=2*%pi;
20
21 k=-5:5
22 for i=1:length(k)
23     if k(i)==0 then
24         ak(i)=1.5;
25     else
26         ak(i)=(sin(k(i)*%pi/2))/(k(i)*%pi);
27     end
28 end
29
30 disp("The fourier series coefficients are...")
31 disp(ak)
32 disp("magnitude of Fourier series coefficient")
33 disp(abs(ak))
34 disp("the givem signal is even and so it has no
    phase spectrum")
35 //Plotting frequency spectrum
36 subplot(2,1,2)
37 plot(k,abs(ak),'.');
38 xtitle("Magnitude Spectrum","k","|ak|");
39 subplot(2,1,1)
40 plot(k,ak,'.');
41 xtitle("Ak","k","ak");

```

Scilab code Exa 5.6 Fourier series Coefficients

```
1 //Fourier Series coefficients for Impulse train
2 clc
3 clear
4 close
5 //Assume period of the impulse train T=2
6 T=2;
7 t=-5*T:T:5*T;
8 for i=1:length(t)
9     x(i)=1;
10 end
11 //Using sifting property of the impulse signal
12 k=-10:10
13 for i=1:length(k)
14     ak(i)=1/T;
15 end
16 subplot(2,1,1)
17 plot(t,x, '. ')
18 xtitle("Impulse train", "t", "x(t)")
19 subplot(2,1,2)
20 plot(k,ak, '. ')
21 xtitle("Fourier coefficients of impulse train", "k", "
    ak")
```

Scilab code Exa 5.7 Fourier series coefficients of halfwave rectified signal

```
1 //Fourier Series coefficients of half-wave rectifier
  output
2 //Assume the period of the signal T=1
3 t=-0.5:0.01:1;
```

```

4 T = 1;
5 for i=1:length(t)
6     if t(i)<T/2 then
7         x(i)=sin(2*%pi*t(i));
8     else
9         x(i)=0;
10    end
11 end
12 k=-10:10;
13 for i=1:length(k)
14     if k(i)==1 then
15         ak(i)=1/(4*%i);
16     elseif k(i)==-1
17         ak(i)=-1/(4*%i);
18     else
19         ak(i)=(cos(k(i)*%pi/2)*exp(-k(i)*%pi/2*-%i))
                /(%pi-(%pi*k(i)*k(i)));
20    end
21 end
22
23
24 disp("The fourier series coefficients are...")
25 disp(ak)
26 disp("magnitude of Fourier series coefficient")
27 disp(abs(ak))
28 //Plotting frequency spectrum
29 subplot(2,1,1)
30 plot(k,abs(ak),'.');
31 xtitle("Magnitude Spectrum","k","|ak|");
32 for i=1:length(k)
33     if k(i)==0 | k(i)==3 | k(i)==-3 | k(i)==-5 |k(i)
        ==5 then
34         phase(i)=0;
35     elseif k(i)==-1 then
36         phase(i)=%pi/2;
37     elseif k(i)==1 then
38         phase(i)=-%pi/2;
39     elseif k(i)==-2 | k(i)==-4

```

```

40         phase(i)=%pi;
41     elseif k(i)==2 | k(i)==4
42         phase(i)=-%pi;
43     else
44         phase(i) = 0;
45     end
46 end
47 subplot(2,1,2)
48 plot(k,phase, '. ');
49 xtitle("Phase Spectrum", "k", "angle(ak)");
50 disp(phase)

```

Scilab code Exa 5.8 Fourier series coefficients

```

1 //Fourier Series coefficients of half-wave rectifier
  output
2 //Assume the period of the signal T=1
3 t=-0.5:0.01:0.5;
4 for i=1:length(t)
5     if t(i)<-0.25 & t(i)>0.25 then
6         x(i)=-1;
7     else
8         x(i)=1;
9     end
10 end
11 k=-10:10;
12 for i=1:length(k)
13     if k(i)==0 then
14         ak(i)=0;
15     else
16         ak(i)=(%i*((2-(-1)^k(i))*exp(-%i*k(i)*%pi/2)
17             -exp(%i*k(i)*%pi/2)))/(k(i)*2*%pi);
18     end
19 end

```

```

20 disp("The fourier series coefficients are...")
21 disp(ak)
22 plot(k,ak, '. ')
23 xtitle("Fourier Coefficients", "k", "ak")

```

Scilab code Exa 5.14 Continuoustime Fourier Transform

```

1 //Continuous Time Signal  $x(t) = \exp(-B*t)u(t)$ ,  $t > 0$ 
2 clear;
3 clc;
4 close;
5 B = 1;
6 Dt = 0.005;
7 t = 0:Dt:10;
8 xt = exp(-B*t);
9 Wmax = 2*%pi*1;
10 K = 4;
11 k = 0:(K/1000):K;
12 W = k*Wmax/K;
13 XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
14 XW_Mag = abs(XW);
15 W = [-mtlbfliplr(W), W(2:1001)];
16 XW_Mag = [mtlbfliplr(XW_Mag), XW_Mag(2:1001)];
17 [XW_Phase,db] = phasemag(XW);
18 XW_Phase = [-mtlbfliplr(XW_Phase),XW_Phase(2:1001)
];
19 //Plotting Continuous Time Signal
20 figure(1)
21 plot(t,xt);
22 xlabel('t in sec. ');
23 ylabel('x(t)')
24 title('Continuous Time Signal')
25 figure(2)
26 //Plotting Magnitude Response of CTS
27 subplot(2,1,1);

```

```

28 plot(W,XW_Mag);
29 xlabel('Frequency in Radians/Seconds—> W');
30 ylabel('abs(X(jW))')
31 title('Magnitude Response (CTFT)')
32 //Plotting Phase Reponse of CTS
33 subplot(2,1,2);
34 plot(W,XW_Phase*%pi/180);
35 xlabel('                                Frequency in
        Radians/Seconds—> W');
36 ylabel('                                <X
        (jW)')
37 title('Phase Response(CTFT) in Radians')

```

Scilab code Exa 5.15 Continuoustime Fourier Transform

```

1 //Continuous Time Signal  $x(t) = \exp(B*t)u(-t)$ ,  $t > 0$ 
2 clear;
3 clc;
4 close;
5 B = 1;
6 Dt = 0.005;
7 t = -10:Dt:0;
8 xt = exp(B*t);
9 Wmax = 2*%pi*1;
10 K = 4;
11 k = 0:(K/1000):K;
12 W = k*Wmax/K;
13 XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
14 XW_Mag = abs(XW);
15 W = [-mtlbfliplr(W), W(2:1001)];
16 XW_Mag = [mtlbfliplr(XW_Mag), XW_Mag(2:1001)];
17 [XW_Phase,db] = phasemag(XW);
18 XW_Phase = [-mtlbfliplr(XW_Phase),XW_Phase(2:1001)
];

```

```

19 //Plotting Continuous Time Signal
20 figure(1)
21 plot(t,xt);
22 xlabel('t in sec. ');
23 ylabel('x(t) ');
24 title('Continuous Time Signal')
25 figure(2)
26 //Plotting Magnitude Response of CTS
27 subplot(2,1,1);
28 plot(W,XW_Mag);
29 xlabel('Frequency in Radians/Seconds —> W');
30 ylabel('abs(X(jW)) ');
31 title('Magnitude Response (CTFT) ');
32 //Plotting Phase Reponse of CTS
33 subplot(2,1,2);
34 plot(W,XW_Phase*%pi/180);
35 xlabel('                                Frequency in
        Radians/Seconds —> W');
36 ylabel('                                <X
        (jW) ');
37 title('Phase Response(CTFT) in Radians ')

```

Scilab code Exa 5.16 Continuoustime Fourier Transform

```

1 //Continuous Time Signal x(t)= exp(-B*abs(t))
2 clear;
3 clc;
4 close;
5 B =1;
6 Dt = 0.005;
7 t = -4.5:Dt:4.5;
8 xt = exp(-B*abs(t));
9 Wmax = 2*%pi*1;
10 K = 4;

```



```

11 k = 0:(K/1000):K;
12 W = k*Wmax/K;
13 XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
14 XW = real(XW);
15 W = [-mtlbfliplr(W), W(2:1001)];
16 XW = [mtlbfliplr(XW), XW(2:1001)];
17 disp("The given signal is even and it has no phase
       spectrum")
18 subplot(2,1,1);
19 plot(t,xt);
20 xlabel('t in sec. ');
21 ylabel('x(t) ');
22 title('Continuous Time Signal')
23 subplot(2,1,2);
24 plot(W,XW);
25 xlabel('Frequency in Radians/Seconds W');
26 ylabel('X(jW) ');
27 title('Continuous-time Fourier Transform')

```

Scilab code Exa 5.17 Continuoustime Fourier Transform

```

1 //Frequency Response of a Rectangular Waveform
2 // x(t)= A, from -T1 to T1
3 clear;
4 clc;
5 close;
6 A =1;
7 Dt = 0.005;
8 T1 = 4;
9 t = -T1/2:Dt:T1/2;
10 for i = 1:length(t)
11     xt(i) = A;
12 end
13 Wmax = 2*%pi*1;
14 K = 4;

```

```

15 k = 0:(K/1000):K;
16 W = k*Wmax/K;
17 xt = xt';
18 XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
19 XW_Mag = real(XW);
20 W = [-mtlbfliplr(W), W(2:1001)];
21 XW_Mag = [mtlbfliplr(XW_Mag), XW_Mag(2:1001)];
22 subplot(2,1,1);
23 plot(t,xt);
24 xlabel('t in sec. ');
25 title('Contiuous Time Signal x(t)')
26 subplot(2,1,2);
27 plot(W,XW_Mag);
28 xlabel('Frequency in Radians/Seconds');
29 title('Continuous-time Fourier Transform X(jW)')

```

Scilab code Exa 5.18 Inverse Fourier transform

```

1 // Inverse Continuous Time Fourier Transform
2 // X(jW)= 1, from -T1 to T1
3 clear;
4 clc;
5 close;
6 // CTFT
7 A =1;
8 Dw = 0.005;
9 W1 = 4;
10 w = -W1/2:Dw:W1/2;
11 for i = 1:length(w)
12     XW(i) = A;
13 end
14 XW = XW';
15 //Inverse Continuous-time Fourier Transform
16 t = -3*pi:pi/length(w):3*pi;
17 xt =(1/(2*pi))*XW *exp(sqrt(-1)*w'*t)*Dw;

```

```

18 xt = real(xt);
19 figure
20 plot(t,xt);
21 xlabel('          t Sec');
22 title('Time domain signal x(t)')

```

Scilab code Exa 5.19 Continuoustime Fourier Transform

```

1 //frequency response of impulse signal
2 clear;
3 clc;
4 close;
5 A =1;
6 Dt = 0.005;
7 T1 = 4;
8 Wo=2//Assume Wo=2
9 t = -T1/2:Dt:T1/2;
10 for i = 1:length(t)
11     xt(i)=sin(Wo*t(i));
12 end
13 Wmax = 2*%pi*1;
14 K = 4;
15 k = 0:(K/1000):K;
16 W = k*Wmax/K;
17 xt = xt';
18 XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
19 XW_Mag = real(XW);
20 W = [-mtlbfliplr(W), W(2:1001)];
21 XW_Mag = [mtlbfliplr(XW_Mag), XW_Mag(2:1001)];
22 subplot(2,1,1);
23 plot(t,xt);
24 xlabel('t in sec. ');
25 title('Contiuous Time Signal x(t)')
26 subplot(2,1,2);
27 plot(W,XW_Mag);

```

```
28 xlabel('Frequency in Radians/Seconds');
29 title('Continuous-time Fourier Transform X(jW)')
```

Scilab code Exa 5.20 Inverse Fourier transform

```
1 // Inverse Continuous Time Fourier Transform
2 // X(jW)= 2*pi, at W=0
3 clear;
4 clc;
5 close;
6 // CTFT
7 A =1;
8 Dw = 0.005;
9 W1 = 4;
10 w = -W1/2:Dw:W1/2;
11 for i = 1:length(w)
12     if w(i)==0 then
13         XW(i) = 2*%pi;
14     else
15         XW(i)=0;
16     end
17 end
18 XW = XW';
19 subplot(2,1,1)
20 plot(w,XW)
21 //Inverse Continuous-time Fourier Transform
22 t = -3*%pi:%pi/length(w):3*%pi;
23 xt =(1/(2*%pi))*XW *exp(sqrt(-1)*w'*t)*Dw;
24 xt = real(1+xt);
25 subplot(2,1,2)
26 plot(t,xt);
27 xlabel('          t Sec');
28 title('Time domain signal x(t)')
```

Scilab code Exa 5.21 Inverse Fourier Transform

```
1 // Inverse Continuous Time Fourier Transform
2 // X(jW)= 2*pi, at W=Wo
3 clear;
4 clc;
5 close;
6 // CTFT
7 A =1;
8 Dw = 0.005;
9 W1 = 4;
10 Wo=2//Assume Wo=2
11 w = -W1/2:Dw:W1/2;
12 for i = 1:length(w)
13     if w(i)==Wo then
14         XW(i) = 2*pi;
15     else
16         XW(i)=0;
17     end
18 end
19 XW = XW';
20 //Inverse Continuous-time Fourier Transform
21 t = -3*pi:%pi/length(w):3*pi;
22 xt =(1/(2*pi))*XW *exp(sqrt(-1)*w'*t)*Dw;
23 xt = real(1+xt);
24 plot(t,xt);
25 xlabel('          t Sec');
26 title('Time domain signal x(t)')
```

Scilab code Exa 5.22 Inverse fourier transform

```
1 // Inverse Continuous Time Fourier Transform
```

```

2 // X(jW)= 2*pi , at W=-Wo
3 clear;
4 clc;
5 close;
6 // CTFT
7 A =1;
8 Dw = 0.005;
9 W1 = 4;
10 Wo=2//Assume Wo=2
11 w = -W1/2:Dw:W1/2;
12 for i = 1:length(w)
13     if w(i)==-Wo then
14         XW(i) = 2*pi;
15     else
16         XW(i)=0;
17     end
18 end
19 XW = XW';
20 //Inverse Continuous-time Fourier Transform
21 t = -3*pi:pi/length(w):3*pi;
22 xt =(1/(2*pi))*XW *exp(sqrt(-1)*w'*t)*Dw;
23 xt = real(1+xt);
24 plot(t,xt);
25 xlabel('          t Sec');
26 title('Time domain signal x(t)')

```

Scilab code Exa 5.24 Fourier Transform of periodic sinusoid

```

1 // Continuous Time Fourier Transforms of
2 // Sinusoidal waveforms sin(Wot)
3 clear
4 clc;
5 close;
6 T1 = 2;
7 T = 4*T1;

```

```

8  Wo = 2*%pi/T;
9  W = [-Wo,0,Wo];
10 ak = (2*%pi*Wo*T1/%pi)/sqrt(-1);
11 XW = [-ak,0,ak];
12 plot(W,-imag(XW),'.');
13 xlabel('
      W');
14 xtitle('CTFT of sin(Wot)', 'W', 'X(jW)')

```

Scilab code Exa 5.25 Fourier Transform of periodic signal

```

1  // Continuous Time Fourier Transforms of
2  // Sinusoidal waveforms cos(Wot)
3  clear;
4  clc;
5  close;
6  // CTFT
7  T1 = 2;
8  T = 4*T1;
9  Wo = 2*%pi/T;
10 W = [-Wo,0,Wo];
11 ak = (2*%pi*Wo*T1/%pi);
12 XW =[ak,0,ak];
13 plot(W,abs(XW),'.');
14 xlabel('
      W');
15 xtitle('CTFT of cos(Wot)', 'W', 'X(jW)')

```

Scilab code Exa 5.32 Fourier transform of impulse train

```

1  //CTFT of Periodic Impulse Train
2  clear;
3  clc;

```

```

4 close;
5 // CTFT
6 T = -4:4;;
7 T1 = 1;
8 xt = ones(1,length(T));
9 ak = 1/T1;
10 XW = 2*%pi*ak*ones(1,length(T));
11 Wo = 2*%pi/T1;
12 W = Wo*T;
13 figure
14 subplot(2,1,1)
15 plot2d3('gnn',T,xt);
16 xlabel('t');
17 title('Periodic Impulse Train')
18 subplot(2,1,2)
19 plot2d3('gnn',W,XW);
20 xlabel('t');
21 title('CTFT of Periodic Impulse Train')

```

Scilab code Exa 5.37 Frequency response of the system

```

1 //Continuous Time Signal  $x(t) = 0.5 \cdot \exp(-B \cdot t \cdot 0.5) u(t)$ 
  ,  $t > 0$ 
2 clear;
3 clc;
4 close;
5 B = 1;
6 Dt = 0.005;
7 t = 0:Dt:10;
8 h = 0.5*exp(-B*t*0.5);
9 Wmax = 2*%pi*1;
10 K = 4;
11 k = 0:(K/1000):K;
12 W = k*Wmax/K;
13 XW = h* exp(-sqrt(-1)*t'*W) * Dt;

```



```

14 XW_Mag = abs(XW);
15 W = [-mtlbfliplr(W), W(2:1001)];
16 XW_Mag = [mtlbfliplr(XW_Mag), XW_Mag(2:1001)];
17 [XW_Phase,db] = phasemag(XW);
18 XW_Phase = [-mtlbfliplr(XW_Phase),XW_Phase(2:1001)
    ];
19 //Plotting Continuous Time Signal
20 figure(1)
21 plot(t,h);
22 xlabel('t in sec. ');
23 ylabel('x(t)')
24 title('Continuous Time Signal')
25 figure(2)
26 //Plotting Magnitude Response of CTS
27 subplot(2,1,1);
28 plot(W,XW_Mag);
29 xlabel('Frequency in Radians/Seconds—> W');
30 ylabel('abs(X(jW))')
31 title('Magnitude Response (CTFT)')
32 //Plotting Phase Reponse of CTS
33 subplot(2,1,2);
34 plot(W,XW_Phase*%pi/180);
35 xlabel('                                Frequency in
    Radians/Seconds—> W');
36 ylabel('                                <X
    (jW)')
37 title('Phase Response(CTFT) in Radians')

```

Chapter 6

Sampling

Scilab code Exa 6.1 Sampling

```
1 //Sampling the CT signals
2 clc
3 clear
4 close
5 t=-0.3:0.0001:0.3;
6 x1=2*cos(2*pi*20*t); //F1=20Hz
7 x2=2*cos(2*pi*80*t); //F2=80Hz
8 figure(1)
9 subplot(2,1,1)
10 plot(t,x1);
11 xtitle("CT Signal X1(t)","t","x1(t)");
12 subplot(2,1,2)
13 plot(t,x2)
14 xtitle("CT Signal X2(t)","t","x2(t)");
15 //Given Sampling frequency Fs=60Hz
16 Fs=60;
17 n=-10:1:10;
18 Ts=1/60; //Sampling interval Ts=1/Fs
19 x1_n=2*cos(2*pi*20*n*Ts);
20 x2_n=2*cos(2*pi*80*n*Ts);
21 figure(2)
```

```

22 subplot(2,1,1)
23 plot2d3('gmn',n,x1_n,3);
24 xtitle("Sampled signal x1[n]","n","x1[n]")
25 subplot(2,1,2)
26 plot2d3('gmn',n,x2_n,3);
27 xtitle("Sampled signal x2[n]","n","x2[n]")

```

Scilab code Exa 6.2 Sampling

```

1 //Sampling the CT signals
2 clc
3 clear
4 close
5 t=-10:0.01:10;
6 x=sin(%pi*t);
7 figure(1)
8 subplot(2,1,1)
9 plot(t,x);
10 xtitle("CT Signal sin(pi*t)","t","x(t)");
11 Wb=%pi;//Given Sampling frequency is Pi radians
12 Ws=2*Wb;
13 Fs=Ws/(2*%pi);
14 n=-100:1:100;
15 Ts=1/Fs;//Sampling interval Ts=1/Fs
16 x_n=sin(%pi*n*Ts);
17 subplot(2,1,2)
18 plot2d(n,x_n,rect=[-100 -2 100 2]);
19 xtitle("Sampled signal x[n]","n","x[n]")

```

Scilab code Exa 6.3 Sampling

```

1 //Sampling the CT signals
2 clc

```

```

3 clear
4 close
5 t=-0.3:0.0001:0.3;
6 x=5*sin(10*pi*t);
7 figure(1)
8 plot(t,x);
9 xtitle("CT Signal x(t)", "t", "x(t)");
10 //Given Sampling frequency (a) Fs=15Hz (b) Fs=6Hz
11 Fs1=15;
12 Fs2=6;
13 n=-10:1:10;
14 Ts1=1/Fs1; //Sampling interval Ts=1/Fs
15 Ts2=1/Fs2;
16 x1=5*sin(pi*10*n*Ts1);
17 x2=5*sin(pi*10*n*Ts2);
18 figure(2)
19 subplot(2,1,1)
20 plot2d3('ggn',n,x1);
21 xtitle("Sampled signal Fs=15Hz", "n", "x1[n]")
22 subplot(2,1,2)
23 plot2d3('ggn',n,x2);
24 xtitle("Sampled signal Fs=6Hz", "n", "x2[n]")

```

Scilab code Exa 6.4 Sampling

```

1 // Continuous Time Fourier Transforms of
2 // Sinusoidal waveforms 3cos(2*pi*t)
3 clear;
4 clc;
5 close;
6 // CTFT
7 t=-10:0.01:10;
8 x=3*cos(2*pi*t);
9 subplot(2,1,1)
10 plot(t,x);

```

```

11 xtitle("CT signal x(t)", "t", "x(t)");
12 T1 = 2;
13 T = 4*T1;
14 Wo = 6*%pi/T;
15 W = [-Wo, 0, Wo];
16 ak = (2*%pi*Wo*T1/%pi);
17 XW =[ak, 0, ak];
18 subplot(2,1,2)
19 plot2d3('ggn', W, real(XW));
20 xlabel('          W');
21 xtitle('CTFT of cos(Wot)', 'W', 'X(jW)')
22 n=-10:10;
23 W1=4*%pi;
24 W2=8*%pi;
25 W3=3*%pi;
26 T1=(2*%pi)/W1;
27 T2=(2*%pi)/W2;
28 T3=(2*%pi)/W3;
29 x1=3*cos(2*%pi*n*T1);
30 x2=3*cos(2*%pi*n*T2);
31 x3=3*cos(2*%pi*n*T3);
32 figure(1)
33 subplot(3,1,1)
34 plot2d3('ggn', n, x1);
35 xtitle("X(t) sampled at Ws=4*pi", "n", "x1[n]");
36 subplot(3,1,2)
37 plot2d3('ggn', n, x2);
38 xtitle("X(t) sampled at Ws=8*pi", "n", "x2[n]");
39 subplot(3,1,3)
40 plot2d3('ggn', n, x3);
41 xtitle("X(t) sampled at Ws=3*pi", "n", "x3[n]");

```

Scilab code Exa 6.6 Sampling

```

1 //Sampling the signal at nyquist rate

```

```

2  clear;
3  clc;
4  close;
5  t=-1:0.01:1;
6  x=2*cos(200*%pi*t)+3*sin(100*%pi*t)-4*sin(500*%pi*t)
    ;
7  f1=100;
8  f2=50;
9  f3=250;
10 fb=max(f1,f2,f3);
11 Fs=2*fb;
12 Ts=1/Fs;
13 n=-10:10;
14 x_n=2*cos(200*%pi*n*Ts)+3*sin(100*%pi*n*Ts)-4*sin
    (500*%pi*n*Ts);
15 plot2d3('gmn',n,x_n)
16 xtitle("DT Signal x(n) sampled at nyquist rate","n",
    "x[n]");

```

Scilab code Exa 6.7 Sampling

```

1  //Determining nyquist rate for the signals
2  clc
3  clear
4  close
5  Wb1=4*%pi;
6  Wb2=10*%pi;
7  Wbs=max(Wb1,Wb2);
8  Ws=2*Wbs;
9  //Bandlimited frequency doesnt change by Amplitude
    scaling
10 // (a) 2*x1(t)
11 Wa=2*Wb1
12 disp("Wa=")
13 disp(Wa)

```

```

14 //Timing shifting doesnt affect the magnitude
    spectrum
15 //(b) x1(t-1)
16 Wb=2*Wb1
17 disp("Wb=")
18 disp(Wb)
19 //Adding two band-limited spectrums will not
    sampling frequency
20 //(c) 2*x1(t)+x1(t-1)
21 Wc=2*Wb1
22 disp("Wc=")
23 disp(Wc)
24 //Compressing time axis expands frequency axis by
    the same factor
25 //(d) x2(2t)
26 Wd=2*2*Wb2
27 disp("Wd=")
28 disp(Wd)
29 //Expanding the time axis compresses the frequency
    axis by same factor
30 //(e) x2(t/2)
31 We=1/2*2*Wb2
32 disp("We=")
33 disp(We)
34 //(f) x2(2t)+x2(t/2)
35 Wf=max(Wd, We)
36 disp("Wf=")
37 disp(Wf)
38 //x1(t)x2(t)
39 Wg=2*(Wb1+Wb2)
40 disp("Wg=")
41 disp(Wg)
42 //x1(t)*x2(t)
43 Wh=2*min(Wb1, Wb2)
44 disp("Wh=")
45 disp(Wh)
46 //x1(t)*cos(2*%pi*t)
47 Wi=2*(Wb1+2*%pi)

```

```
48 disp("Wi=")
49 disp(Wi)
50 //x1'(t)
51 Wj=2*Wb1
52 disp("Wj=")
53 disp(Wj)
```

Chapter 7

Fourier Analysis of discretetime signals and systems

Scilab code Exa 7.3 Fourier series representation of DT signal

```
1 //DTFS of  $x[n] = 2\cos((\pi/3)n + (\pi/6))$ 
2 clear;
3 close;
4 clc;
5 n = -3:3;
6 N = 6;
7 Wo = 2*%pi/N;
8 xn = 2*cos((%pi/3)*n+(%pi/6));
9 //By euler's theorem  $X[n]$  can be represented
10 x_n=exp(%i*(%pi*n/3)+%pi/6)+exp(-%i*(%pi*n/3)+%pi/6)
11 for i=1:length(n)
12     if n(i)==1
13         a(i)=exp(%i*%pi/6);
14     elseif n(i)==-1
15         a(i)=exp(-%i*%pi/6);
16     else
17         a(i)=0;
18     end
19 end
```

```

20 for i=1:length(a)
21     if real(a(i))==0 then
22         phase(i)=0;
23     else
24         phase(i)=atan(imag(a(i))/real(a(i)));
25     end
26 end
27 subplot(2,1,1)
28 plot2d3('ggn',n,abs(a))
29 xtitle("MAgnitude spectrum","k","|ak|")
30 subplot(2,1,2)
31 plot2d3('ggn',n,phase)
32 xtitle("Phase spectrum","k","angle(ak)")

```

Scilab code Exa 7.4 Fourier series representation of DT signal

```

1 //Fouries series representation of combination of
  signals
2 //x[n]=1+sin(pi*n/2)+cos(%pi*n/4)
3 clc
4 clear
5 close
6 n=-3:3;
7 x=1+sin(%pi*n/2)+cos(%pi*n/4);
8 w1=%pi/2;
9 w2=%pi/4;
10 N1=2*%pi/w1;
11 N2=2*%pi/w2;
12 N=max(N1,N2);
13 wo=2*%pi/N;
14 //Expanding x[n] by Euler's theorem
15 xn=1+0.5*exp(%i*wo*n)+0.5*exp(-%i*wo*n)-0.5*%i*exp(
    %i*2*wo*n)-0.5*%i*exp(-%i*2*wo*n);
16 a0=1;
17 a1=0.5;

```

```

18 a_1=0.5;
19 a2=1/2*%i;
20 a_2=-1/2*%i;
21 a=[a_2 a_1 a0 a1 a2];
22 a1=[0 a 0];
23 phase=[%pi/2 0 0 0 -%pi/2]
24 phase=[0 phase 0]
25 subplot(2,1,1)
26 plot(n,abs(a1),'.')
27 xtitle("magnitude spectrum","k","ak")
28 subplot(2,1,2)
29 plot(n,phase,'.')
30 xtitle("Phase spectrum","k","ak")

```

Scilab code Exa 7.5 Fourier series representation of DT signal

```

1 //DTFS coefficients of periodic square wave
2 clear;
3 close;
4 clc;
5 N = 10;
6 N1 = 2;
7 Wo = 2*%pi/N;
8 xn = ones(1,length(N));
9 n = -(2*N1+1):(2*N1+1);
10 a(1) = (2*N1+1)/N;
11 for k =1:2*N1
12     a(k+1) = sin((2*%pi*k*(N1+0.5))/N)/sin(%pi*k/N);
13     a(k+1) = a(k+1)/N;
14     if(abs(a(k+1))<=0.1)
15         a(k+1) =0;
16     end
17 end
18 a =a';
19 a_conj =conj(a);

```

```

20 ak = [a_conj($:-1:1),a(2:$)];
21 k = -2*N1:2*N1;
22 plot2d3('gmn',k,abs(ak))
23 xtitle('Magnitude spectrum','k','|ak|')

```

Scilab code Exa 7.6 Fourier series representation of DT signal

```

1 //DTFS of a periodic sequence
2 clc
3 clear
4 close
5 n=-4:3;
6 x=[0 1 2 3 0 1 2 3];
7 N=4;
8 k=0:3;;
9 wo=2*%pi/N;
10 a0=1.5;
11 a1=-0.5+0.5*%i;
12 a2=-0.5;
13 a3=-0.5-0.5*%i;
14 a=[a0,a1,a2,a3]
15 for i=1:length(a)
16     phase(i)=atan(imag(a(i))/real(a(i)));
17 end
18 subplot(2,1,1)
19 plot(k,abs(a),'.'');
20 xtitle("magnitude spectrum","k","ak");
21 subplot(2,1,2)
22 plot(k,phase,'.'');
23 xtitle("phase spectrum","k","ak");

```

Scilab code Exa 7.7 Fourier series representation of DT signal

```

1 //DTFS of discrete periodic signal
2 clc
3 clear
4 close
5 N=2//assume N=2
6 n=-2*N:2*N
7 for i=1:length(n)
8     if modulo(n(i),N)==0 then
9         x(i)=1;
10    else
11        x(i)=0;
12    end
13 end
14 subplot(2,1,1)
15 plot(n,x, '. ')
16 xtitle("Input signal x[n]", "n", "x[n]");
17 k=-5:5;
18 for i=1:length(k)
19     ak(i)=1/N;
20 end
21 subplot(2,1,2)
22 plot(k,ak, '. ')
23 xtitle("Frequency spectrum", "k", "ak")

```

Scilab code Exa 7.8 Fourier series representation of DT signal

```

1 //x[n] = 1+sin(2*%pi/N)n+3cos(2*%pi/N)n+cos[(4*%pi/N
    )n+%pi/4]
2 clear;
3 close;
4 clc;
5 N = 10;
6 n = 0:0.01:N;
7 Wo = 2*%pi/N;
8 xn =ones(1, length(n))+sin(Wo*n)+3*cos(Wo*n)+cos(2*Wo

```

```

    *n+%pi/4);
9  for k =0:N-2
10  C(k+1,:) = exp(-sqrt(-1)*Wo*n.*k);
11  a(k+1) = xn*C(k+1,:)/length(n);
12  if(abs(a(k+1))<=0.1)
13      a(k+1)=0;
14  end
15 end
16 a =a';
17 a_conj =conj(a);
18 ak = [a_conj($:-1:1),a(2:$)];
19 Mag_ak = abs(ak);
20 for i = 1:length(a)
21     Phase_ak(i) = atan(imag(ak(i))/(real(ak(i))
        +0.0001));
22 end
23 Phase_ak = Phase_ak'
24 Phase_ak = [Phase_ak(1:$-1) -Phase_ak($:-1:1)];
25 k = -(N-2):(N-2);
26 subplot(2,1,1)
27 plot2d3('gnn',k,Mag_ak,5)
28 xtitle('abs(ak)','k','ak')
29 subplot(2,1,2)
30 plot2d3('gnn',k,Phase_ak,5)
31 xtitle('phase(ak)','k','ak')

```

Scilab code Exa 7.9 Fourier series representation of DT signal

```

1 //x[n] = 1+sin(4*%pi/N)n+cos(10*%pi/N)n
2 clear;
3 close;
4 clc;
5 N = 21;
6 n = 0:0.01:N;
7 Wo = 2*%pi/N;

```

```

8 xn =ones(1,length(n))+sin(2*Wo*n)+cos(5*Wo*n);
9 for k =0:N-2
10   C(k+1,:) = exp(-sqrt(-1)*Wo*n.*k);
11   a(k+1) = xn*C(k+1,:)/length(n);
12   if(abs(a(k+1))<=0.1)
13     a(k+1)=0;
14   end
15 end
16 a =a';
17 a_conj =conj(a);
18 ak = [a_conj($:-1:1),a(2:$)];
19 Mag_ak = abs(ak);
20 for i = 1:length(a)
21   Phase_ak(i) = atan(imag(ak(i))/(real(ak(i))
    +0.0001));
22 end
23 Phase_ak = Phase_ak'
24 Phase_ak = [Phase_ak(1:$-1) -Phase_ak($:-1:1)];
25 k = -(N-2):(N-2);
26 subplot(2,1,1)
27 plot2d3('gnn',k,Mag_ak,5)
28 xtitle('abs(ak)','k','ak')
29 subplot(2,1,2)
30 plot2d3('gnn',k,Phase_ak,5)
31 xtitle('phase(ak)','k','ak')

```

Scilab code Exa 7.10 DTFSrepresentation

```

1 //x[n] = 0.5+0.5*cos(2*%pi/N)n
2 clear;
3 close;
4 clc;
5 N = 8;
6 n = 0:0.01:N;
7 Wo = 2*%pi/N;

```

```

8 xn =0.5*ones(1,length(n))+0.5*cos(Wo*n);
9 for k =0:N-2
10   C(k+1,:) = exp(-sqrt(-1)*Wo*n.*k);
11   a(k+1) = xn*C(k+1,:)/length(n);
12   if(abs(a(k+1))<=0.1)
13     a(k+1)=0;
14   end
15 end
16 a =a';
17 a_conj =conj(a);
18 ak = [a_conj($:-1:1),a(2:$)];
19 Mag_ak = abs(ak);
20 k = -(N-2):(N-2);
21 plot2d3('gnn',k,Mag_ak,5)
22 xtitle('abs(ak)', 'k', 'ak')

```

Scilab code Exa 7.16 Discretetime fourier transform

```

1 //Discrete Time Fourier Transform of discrete
  sequence
2 //x[n]= (a^n).u[n], |a|<1
3 clear;
4 clc;
5 close;
6 a1 = 0.5;
7 max_limit = 10;
8 for n = 0:max_limit-1
9   x1(n+1) = (a1^n);
10 end
11 n = 0:max_limit-1;
12 Wmax = 2*%pi;
13 K = 4;
14 k = 0:(K/1000):K;
15 W = k*Wmax/K;
16 x1 = x1';

```



```

17 XW1 = x1* exp(-sqrt(-1)*n'*W);
18 XW1_Mag = abs(XW1);
19 W = [-mtlbfliplr(W), W(2:1001)]; // Omega from -
    Wmax to Wmax
20 XW1_Mag = 2.5*[mtlbfliplr(XW1_Mag), XW1_Mag(2:1001)
    ];
21 [XW1_Phase,db] = phasemag(XW1);
22 XW1_Phase = (1/30)*[-mtlbfliplr(XW1_Phase),
    XW1_Phase(2:1001)];
23 subplot(3,1,1);
24 plot2d3('gmn',n,x1);
25 xtitle('Discrete Time Sequence x[n]')
26 subplot(3,1,2);
27 plot2d(W,XW1_Mag);
28 title('Magnitude Response abs(X(jW))')
29 subplot(3,1,3);
30 plot2d(W,XW1_Phase);
31 title('Phase Response <(X(jW))')

```

Scilab code Exa 7.17 Discretetime fourier transform

```

1 //Discrete Time Fourier Transform of discrete
    sequence
2 //x[n]= (a^n).u[-n], |a|>1
3 clear;
4 clc;
5 close;
6 a1 = 3;
7 min_limit = -20;
8 n = min_limit:0
9 for i=1:length(n)
10    x1(i) = (a1^n(i));
11 end
12 Wmax = 2*%pi;
13 K = 4;

```

```

14 k = 0:(K/1000):K;
15 W = k*Wmax/K;
16 x1 = x1';
17 XW1 = x1* exp(-sqrt(-1)*n'*W);
18 XW1_Mag = abs(XW1);
19 W = [-mtlbfliplr(W), W(2:1001)]; // Omega from -
    Wmax to Wmax
20 XW1_Mag = [mtlbfliplr(XW1_Mag), XW1_Mag(2:1001)];
21 [XW1_Phase,db] = phasemag(XW1);
22 XW1_Phase = [-mtlbfliplr(XW1_Phase),XW1_Phase
    (2:1001)];
23 subplot(3,1,1);
24 plot2d3('gnn',n,x1);
25 xtitle('Discrete Time Sequence x[n]', 'n', 'x[n]')
26 subplot(3,1,2);
27 plot2d(W,XW1_Mag);
28 xtitle('Magnitude Response abs(X(jW))', 'w', '|X(jW)|')
    )
29 subplot(3,1,3);
30 plot2d(W,XW1_Phase);
31 xtitle('Phase Response <(X(jW))', 'w', '<(X(jW))')

```

Scilab code Exa 7.18 Discretetime fourier transform

```

1 //Discrete Time Fourier Transform of
2 //x[n]= (a^abs(n)) |a|<1
3 clear;
4 clc;
5 close;
6 // DTS Signal
7 a = 0.5;
8 max_limit = 10;
9 n = -max_limit+1:max_limit-1;
10 x = a^abs(n);
11 // Discrete-time Fourier Transform

```

```

12 Wmax = 2*%pi;
13 K = 4;
14 k = 0:(K/1000):K;
15 W = k*Wmax/K;
16 XW = x* exp(-sqrt(-1)*n'*W);
17 XW_Mag = real(XW);
18 W = [-mtlbfliplr(W), W(2:1001)]; // Omega from -
    Wmax to Wmax
19 XW_Mag = [mtlbfliplr(XW_Mag), XW_Mag(2:1001)];
20 //plot for abs(a)<1
21 figure
22 subplot(2,1,1);
23 plot2d3('gmn',n,x);
24 xtitle('Discrete Time Sequence x[n] for a>0', 'n', 'x[
    n]')
25 subplot(2,1,2);
26 plot2d(W,XW_Mag);
27 xtitle('Discrete Time Fourier Transform X(exp(jW))',
    'w', '|X(exp(jW))|')

```

Scilab code Exa 7.19 Discretetime fourier transform

```

1 //Discrete Time Fourier Transform of
2 //x[n]= 1 , abs(n)<=M1
3 clear;
4 clc;
5 close;
6 // DTS Signal
7 M1 = 2;
8 n = -M1:M1;
9 x = ones(1,length(n));
10 Wmax = 2*%pi;
11 K = 4;
12 k = 0:(K/1000):K;
13 W = k*Wmax/K;

```

```

14 XW = x* exp(-sqrt(-1)*n'*W);
15 XW_Mag = real(XW);
16 W = [-mtlbfliplr(W), W(2:1001)]; // Omega from -
    Wmax to Wmax
17 XW_Mag = [mtlbfliplr(XW_Mag), XW_Mag(2:1001)];
18 //plot for abs(a)<1
19 figure
20 subplot(2,1,1);
21 plot2d3('gnn',n,x,2);
22 xtitle('Discrete Time Sequence x[n]', 'n', 'x[n]')
23 subplot(2,1,2);
24 plot2d(W,XW_Mag);
25 xtitle('Discrete Time Fourier Transform X(exp(jW))',
    'w', '|X(exp(jW))|')

```

Scilab code Exa 7.24 Fourier transform

```

1 //Discrete Time Fourier Transform of
2 // Periodic Impulse Train
3 clear;
4 clc;
5 close;
6 N = 5;
7 N1 = -3*N:3*N;
8 xn = [zeros(1,N-1),1];
9 x = [1 xn xn xn xn xn xn];
10 ak = 1/N;
11 XW = 2*%pi*ak*ones(1,2*N);
12 Wo = 2*%pi/N;
13 n = -N:N-1;
14 W = Wo*n;
15 figure
16 subplot(2,1,1)
17 plot2d3('gnn',N1,x,2);
18 xtitle('Periodic Impulse Train', 'n', 'x[n]')

```

```

19 subplot(2,1,2)
20 plot2d3('gmn',W,XW,2);
21 xtitle('DTFT of Periodic Impulse Train','w','|X(exp(
      jw))|')
22 disp(Wo)

```

Scilab code Exa 7.26 Discretetime fourier transform

```

1 //Discrete Time Fourier Transform of discrete
  sequence
2 //x[n]= 1, n=2
3 clear;
4 clc;
5 close;
6 a1 = 1/8;
7 max_limit = 10;
8 for n = 0:max_limit-1
9     if n==2 then
10        x1(n+1) = 1;
11    else
12        x1(n+1) = 0;
13    end
14 end
15 n = 0:max_limit-1;
16 Wmax = 2*%pi;
17 K = 4;
18 k = 0:(K/1000):K;
19 W = k*Wmax/K;
20 x1 = x1';
21 XW1 = x1* exp(-sqrt(-1)*n'*W);
22 XW1_Mag = abs(XW1);
23 W = [-mtlb_fliplr(W), W(2:1001)]; // Omega from -
  Wmax to Wmax
24 XW1_Mag = [mtlb_fliplr(XW1_Mag), XW1_Mag(2:1001)];
25 [XW1_Phase,db] = phasemag(XW1);

```

```
26 XW1_Phase = [-mtlbfliplr(XW1_Phase),XW1_Phase
    (2:1001)];
27 subplot(3,1,1);
28 plot2d3('gmn',n,x1);
29 xtitle('Discrete Time Sequence x[n]')
30 subplot(3,1,2);
31 plot2d(W,XW1_Mag);
32 title('Magnitude Response abs(X(jW))')
33 subplot(3,1,3);
34 plot2d(W,XW1_Phase);
35 title('Phase Response <(X(jW))')
```
