

第 8 章 基于 MATLAB 的信号与系统实验

实验 1 信号的时域描述与运算

(基础型实验)

一、实验目的

1. 掌握信号的 MATLAB 表示及其可视化方法。
2. 掌握信号基本时域运算的 MATLAB 实现方法。
3. 利用 MATLAB 分析常用信号，加深对信号时域特性的理解。

二、实验原理与方法

1. 连续时间信号的 MATLAB 表示

连续时间信号指的是在连续时间范围内有定义的信号，即除了若干个不连续点外，在任何时刻信号都有定义。在 MATLAB 中连续时间信号可以用两种方法来表示，即向量表示法和符号对象表示法。

从严格意义上来说，MATLAB 并不能处理连续时间信号，在 MATLAB 中连续时间信号是用信号等时间间隔采样后的采样值来近似表示的，当采样间隔足够小时，这些采样值就可以很好的近似表示出连续时间信号，这种表示方法称为向量表示法。表示一个连续时间信号需要使用两个向量，其中一个向量用于表示信号的时间范围，另一个向量表示连续时间信号在该时间范围内的采样值。例如一个正弦信号可以表示如下：

```
>> t = 0:0.01:10;
```

```
>> x = sin(t);
```

利用 `plot(t,x)`命令可以绘制上述信号的时域波形如图 8-1 所示。

如果连续时间信号可以用表达式来描述，则还可以采用符号表达式来表示信号。例如对于上述正弦信号，可以用符号对象表示如下：

```
>> syms t;
```

```
>> x = sin(t);
```

利用 `ezplot(x)`命令可以绘制上述信号的时域波形如图 8-2 所示。

MATLAB 提供了一些函数用于常用信号的产生，例如阶跃信号、脉冲信号、指数信号、正弦信号等，表 8-1 中列出了一些常用的基本函数。

表 8-1 常用的信号产生函数

函数名	功能	函数名	功能
heaviside	单位阶跃函数	rectpuls	门函数
sin	正弦函数	tripuls	三角脉冲函数
cos	余弦函数	square	周期方波
sinc	sinc 函数	sawtooth	周期锯齿波或三角波
exp	指数函数		

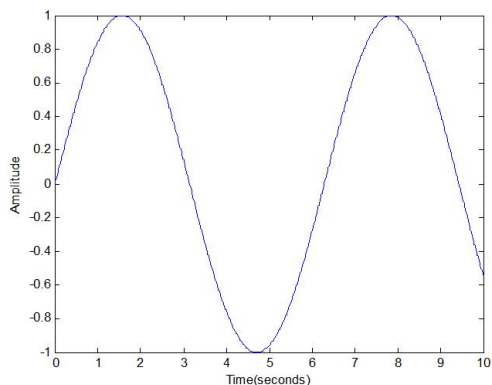


图 8-1 利用向量表示连续时间信号

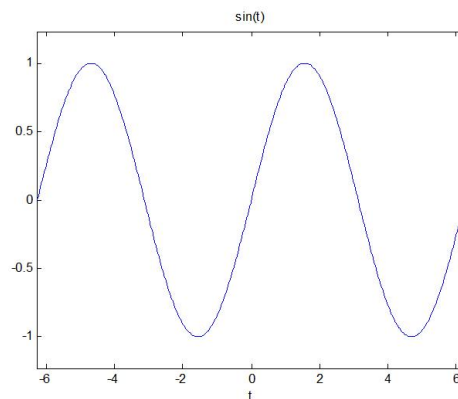


图 8-2 利用符号对象表示连续时间信号

2. 连续时间信号的时域运算

对连续时间信号的运算包括两信号相加、相乘、微分、积分，以及移位、反转、尺度变换（尺度伸缩）等。

1) 相加和相乘

信号相加和相乘指两信号对应时刻的值相加和相乘，对于两个采用向量表示的可以直接使用算术运算的运算符“+”和“*”来计算，此时要求表示两信号的向量时间范围和采样间隔相同。采用符号对象表示的两个信号，可以直接根据符号对象的运算规则运算。

2) 微分和积分

对于向量表示法表示的连续时间信号，可以通过数值计算的方法计算信号的微分和积分的。这里微分是用差分来近似求取的，由时间向量 $[t_1, t_2, \dots, t_N]$ 和采样值向量 $[x_1, x_2, \dots, x_N]$ 表示的连续时间信号，其微分可以通过式（8-1）求得

$$x'(t)|_{t=t_k} \approx \frac{x_{k+1} - x_k}{\Delta t} \quad (8-1)$$

其中 Δt 表示采样间隔。MATLAB 中用 diff 函数来计算差分 $x_{k+1} - x_k$ 。

连续时间信号的定积分可以由 MATLAB 的 quad 函数实现，调用格式为

quad('function_name',a,b)

其中，function_name 为被积函数名，a、b 为积分区间。

对于符号对象表示的连续时间信号，MATLAB 提供了 diff 函数和 int 函数分别用于求微分和积分。

例 8-1 计算定积分 $f = \int_0^{3\pi} e^{-t} \sin(t + \frac{\pi}{2}) dt$ 。

解：可以分别采用两种方法计算如下：

```
>> y = inline('exp(-t).*sin(t+pi/2)');
>> f = quad(y,0,3*pi)
f =
    0.5000
>> f = int('exp(-t)*sin(t+pi/2)',0,3*pi)
f =
    1/(2*exp(3*pi)) + 1/2
>> double(f)
ans =
    0.5000
```

上述两种方法的计算结果相同。其中，`inline` 命令用于由 MATLAB 表达式创建 `inline` 函数对象。

例 8-2 绘制图 8-3 所示三角波信号的一阶导数和积分的波形图。

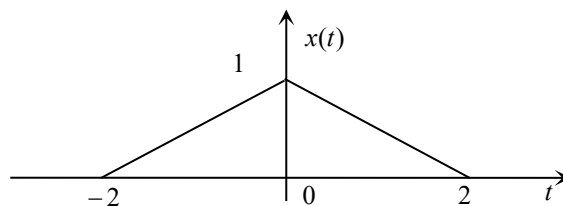


图 8-3 三角波

解：我们分别采用数值计算的方法和符号计算的方法实现。

(a) 数值计算方法的程序如下：

```
%Eg_8_2_a.m
ts = -3;
te = 3;
dt = 0.01;
t = ts:dt:te;
x = tripuls(t,4);           %生成三角波信号
subplot(311);
plot(t,x);                  %绘制 x(t)波形
xlabel('t(s)');
title('x(t)');
grid on;

dx = diff(x)/dt;           %求一阶导数
subplot(312);
plot(t(1:length(t)-1),dx);
xlabel('t(s)');
title('Derivative of x(t)');
grid on;

fun = inline('tripuls(t,4)'); %定义 inline 函数对象
intx = zeros(size(x));
for i = 1:length(t)
    intx(i) = quad(fun,-3,t(i)); %求积分
end
subplot(313);
plot(t,intx);
xlabel('t(s)');
title('Integral of x(t)');
grid on;
```

程序运行结果如图 8-4 所示。

(b) 符号计算方法的程序如下：

```
%Eg_8_2_b.m
syms t;
```

```
x = sym('(1/2*t+1)*(heaviside(t+2)-heaviside(t))+(-1/2*t+1)*(heaviside(t)-heaviside(t-2))');
subplot(311);
ezplot(x,[-3,3]);
title('x(t)');
grid on;
```

```
dx = diff(x); %求一阶导数
subplot(312);
ezplot(dx,[-3,3]);
title('Derivative of x(t)');
grid on;
```

```
intx = int(x); %求积分
subplot(313);
ezplot(intx,[-3,3]);
title('Integral of x(t)');
grid on;
```

程序运行结果如图 8-5 所示。

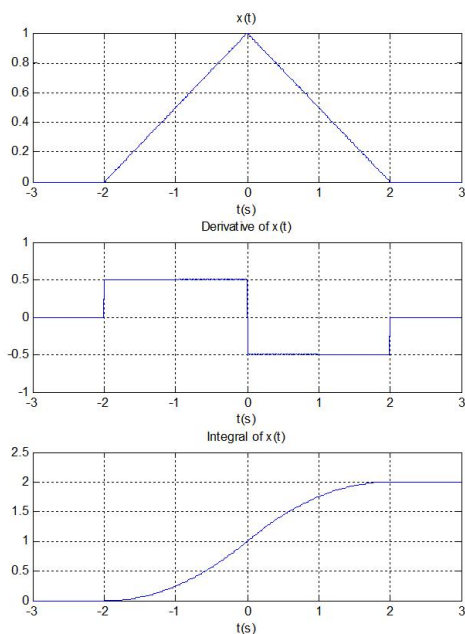


图 8-4 用数值计算方法得到的结果

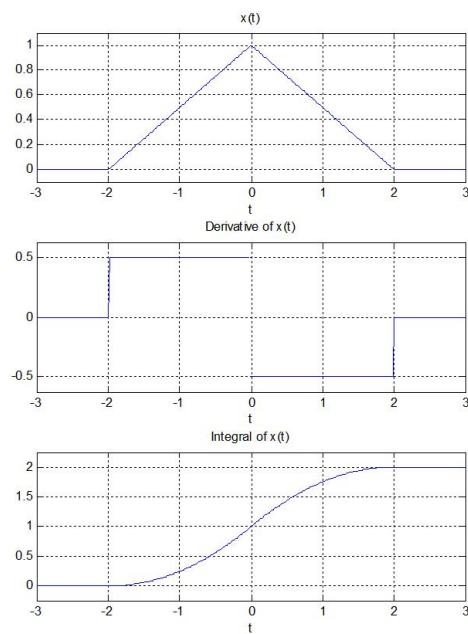


图 8-5 用符号计算方法得到的结果

3) 移位、反转和尺度变换

信号的移位：信号 $x(t)$ 的自变量 t 更换为 $(t-t_0)$ ，表示 $x(t)$ 波形在 t 轴上整体移动，当 $t_0 > 0$ 整体右移，当 $t_0 < 0$ 整体左移。

信号的反转：信号 $x(t)$ 的自变量 t 更换为 $-t$ ， $x(t)$ 的波形相当的波形以 $t=0$ 为轴反转过来。

信号的尺度变换：信号 $x(t)$ 的自变量 t 更换为 at ， $x(at)$ 表示信号压缩 ($a > 1$) 或拉伸 ($a < 1$)。

这里，我们通过一个实例来学习 MATLAB 中这些变换的实现方法。

例 8-3 已知 $x(t) = \sin(3\pi t^2)[u(t) - u(t-2)]$ ，绘制 $x(t)$ 和 $x(6-2t)$ 波形。

解：这里我们还是分别使用数值计算的方法和符号计算的方法来实现。

(a) 数值计算的方法：

```
%Eg_8_3_a.m
t=-0.5:0.001:3.5;
x=(sin(3*pi*t.^2)).*(heaviside(t)-heaviside(t-2));
x1=(sin(3*pi*(6-2*t).^2)).*(heaviside(6-2*t)-heaviside(6-2*t-2));
subplot(211);
plot(t,x); %绘制 x(t)波形
xlabel('t'); title('x(t)');
subplot(212);
plot(t,x1); %绘制 x(3-2t)波形
xlabel('t'); title('x(6-2t)');
程序运行结果如图 8-6 所示。
```

(b) 符号计算的方法：

```
%Eg_8_3_b.m
syms t
x=sin(3*pi*t^2)*(heaviside(t)-heaviside(t-2));
x1=subs(x,t,6-2*t); %使用 subs 命令替换变量
subplot(211);
ezplot(x,[0,3]);
xlabel('t'); title('x(t)');
subplot(212);
ezplot(x1,[0,3]);
xlabel('t'); title('x(6-2t)');
程序运行结果如图 8-7 所示。
```

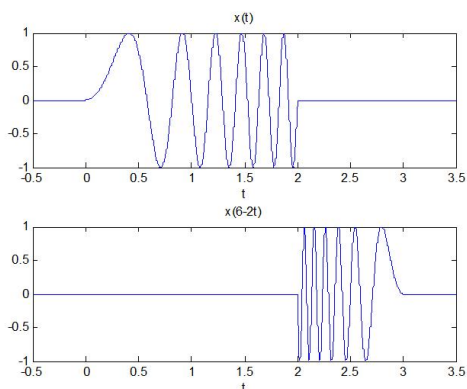


图 8-6 用数值计算的方法实现信号变换

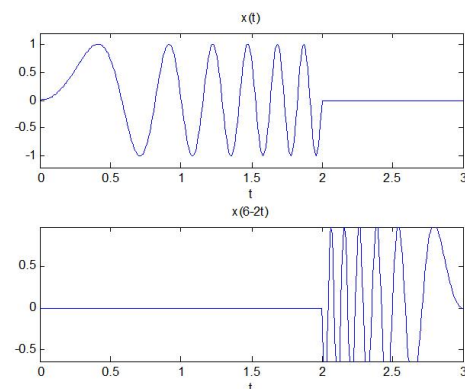


图 8-7 用符号计算的方法实现信号变换

3. 离散时间信号的 MATLAB 表示

离散时间信号仅在一些离散时刻有定义。在 MATLAB 中离散时间信号需要使用两个向量来表示，其中一个向量用于表示离散的时间点，另一个向量表示在这些时间点上的值。例如对于如下离散时间信号

$$x(n) = \{-3, 2, -1, 2, 1, -1, 2, 3\}$$

采用 MATLAB 可以表示如下：

```
>> n = -3:4;  
>> x = [-3 2 -1 2 1 -1 2 3];  
>> stem(n,x,'filled');  
>> xlabel('n');  
>> title('x(n)');
```

stem 函数用于绘制离散时间信号波形，为了与我们表示离散时间信号的习惯相同，在绘图时一般需要添加'filled'选项，以绘制实心的杆状图形。上述命令绘制的信号时域波形如图 8-8 所示。

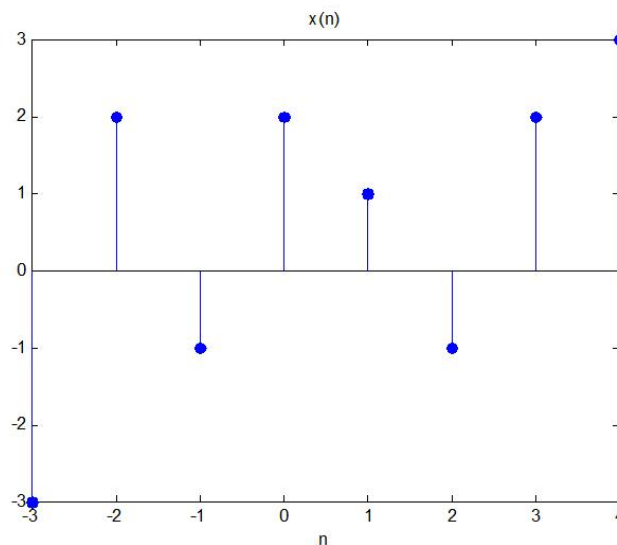


图 8-8 离散时间信号示例

4. 离散时间信号的时域运算

离散时间信号的相加相乘是将两个信号对应的时间点上的值相加或相乘，可以直接使用算术运算的运算符“+”和“*”来计算。

离散时间信号的移位，则可看作是将表示时间的向量平移，而表示对应时间点上的值的向量不变。

离散时间信号的反转，则可看作是将表示时间的向量和表示对应时间点上的值的向量以零点为基准点，以纵轴为对称轴反折，向量的反折可以利用 MATLAB 的 flipr 函数实现。

例 8-4 已知 $x(n) = \{-1, 0, 1, 2\}$ ，求 $x_1(n) = x(n-2)$ 。

解： MATLAB 实现方法如下

```
>> n = -1:2;  
>> x = -1:2;  
>> n1 = n + 2; %时间向量平移  
>> subplot(121);  
>> stem(n,x,'filled');  
>> xlabel('n');  
>> title('x(n)');  
>> subplot(122);  
>> stem(n1,x,'filled');  
>> xlabel('n');
```

>> title('x(n-2)');

结果如图 8-9 所示。

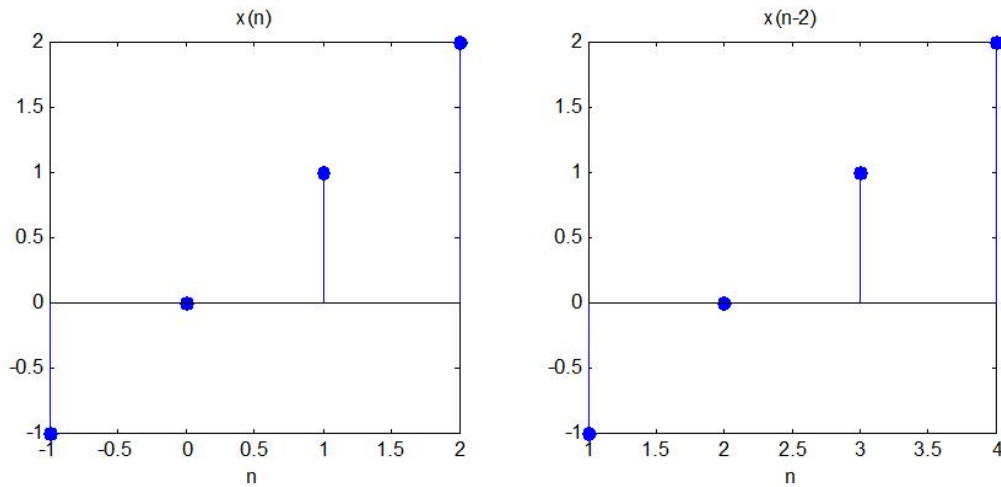


图 8-9 信号移位示例

例 8-5 已知 $x(n] = \{-1, 0, 1, 2\}$ ，求 $x_1(n) = x(-n)$ 。

解： MATLAB 实现方法如下

```
>> n = -1:2;
>> x = -1:2;
>> n1 = -fliplr(n);
>> x1 = fliplr(x);
>> subplot(121);
>> stem(n,x,'filled');
>> xlabel('n');
>> title('x(n)');
>> subplot(122);
>> stem(n1,x1,'filled');
>> xlabel('n');
>> title('x(-n)');
```

结果如图 8-10 所示。

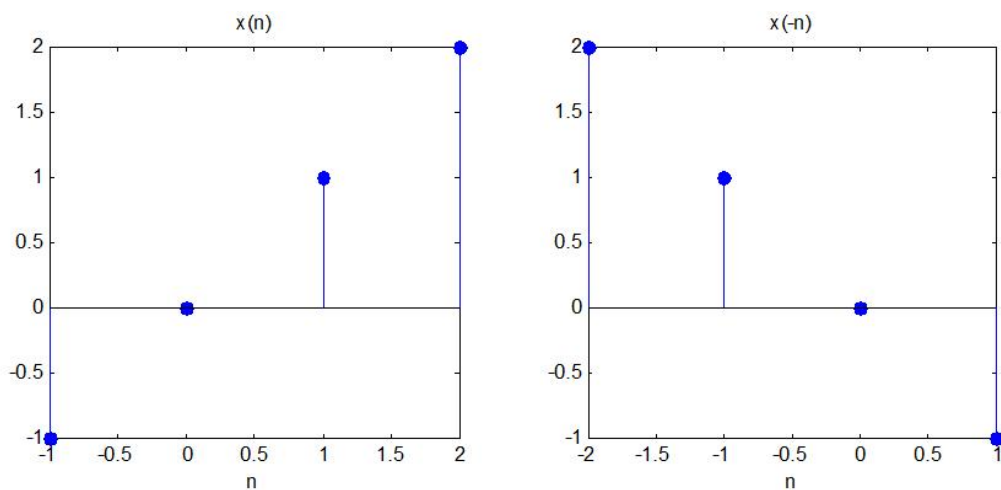


图 8-10 信号反转示例

三、实验内容

1. 利用 MATLAB 绘制下列连续时间信号波形。

(1) $x(t) = (1 - e^{-1.5t})u(t)$

(2) $x(t) = \cos(2\pi t)[u(t) - u(t - 4)]$

(3) $x(t) = \frac{|t|}{2} \cos(\pi t)[u(t + 3) - u(t - 3)]$

(4) $x(t) = e^{-t} \sin(3\pi t)[u(t) - u(t - 3)]$

2. 利用 MATLAB 绘制下列离散时间信号波形。

(1) $x(n) = 2u(n - 4)$

(2) $x(n) = (-1/3)^n u(n)$

(3) $x(n) = 2n[u(n - 1) - u(n - 6)]$

(4) $x(n) = \cos(n\pi/2)u(n)$

3. 利用 MATLAB 生成并绘制连续周期矩形波信号，要求周期为 2，峰值为 3，显示 4 个周期的波形。

4. 已知如图 8-11 所示信号 $x_1(t)$ ，及信号 $x_2(t) = \sin(3\pi t)$ ，用 MATLAB 绘出下列信号的波形：

(1) $x_3(t) = x_1(t) + x_2(t)$

(2) $x_4(t) = x_1(t) \times x_2(t)$

(3) $x_5(t) = x_1(-t) + 2x_1(t)$

(4) $x_6(t) = x_2(t) * x_3(t - 1)$

5. 已知离散时间信号 $x(n]$ 波形如图 8-12 所示，用 MATLAB 绘出 $x(n]$ 、 $x(-n]$ 、 $x(n+3]$ 和 $x(n-4]$ 的波形。

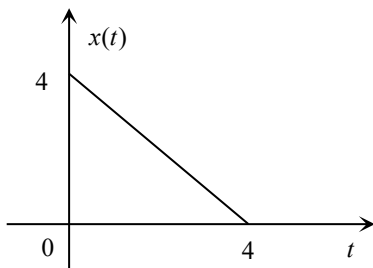


图 8-11

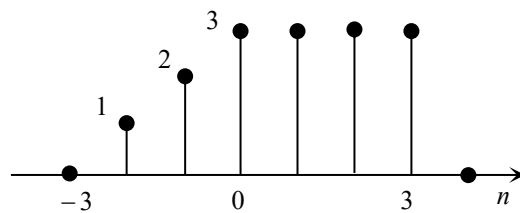


图 8-12

6. 用 MATLAB 编程绘制下列信号的时域波形，观察信号是否为周期信号？若是周期信号，周期是多少？若不是周期信号，请说明原因。

$$(1) \quad x(t) = 1 + 2 \cos\left(\frac{\pi}{4}t - \frac{\pi}{3}\right) + 2 \cos\left(\frac{\pi}{2}t - \frac{\pi}{4}\right) + \cos(2\pi t)$$

$$(2) \quad x(t) = 2 \sin(t) + \sin(\pi t)$$

$$(3) \quad x(n) = 3 + 3 \sin\left(\frac{2n\pi}{4} - \frac{\pi}{8}\right)$$

$$(4) \quad x(n) = 2 \cos\left(\frac{n\pi}{4}\right) + \sin\left(\frac{n\pi}{3}\right) + \cos\left(\frac{n\pi}{2}\right)$$

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

实验 2 LTI 系统的时域分析

(基础型实验)

一、实验目的

1. 掌握利用 MATLAB 对系统进行时域分析的方法。
2. 掌握连续时间系统零状态响应、冲激响应和阶跃响应的求解方法。
3. 掌握求解离散时间系统响应、单位抽样响应的方法。
4. 加深对卷积积分和卷积和的理解。掌握利用计算机进行卷积积分和卷积和计算的方法。

二、实验原理与方法

1. 连续时间系统时域分析的 MATLAB 实现

1) 连续时间系统的 MATLAB 表示

LTI 连续系统通常可以由系统微分方程描述，设描述系统的微分方程为：

$$a_N y^{(N)}(t) + a_{N-1} y^{(N-1)}(t) \cdots + a_0 y(t) = b_M x^{(M)}(t) + b_{M-1} x^{(M-1)}(t) \cdots + b_0 x(t) \quad (8-2)$$

则在 MATLAB 里，可以建立系统模型如下：

$$b = [b_M, b_{M-1}, \cdots, b_0];$$

$$a = [a_N, a_{N-1}, \cdots, a_0];$$

$$\text{sys} = \text{tf}(b, a);$$

其中，tf 是用于创建系统模型的函数，向量 a 与 b 的元素是以微分方程求导的降幂次序来排列的，如果有缺项，应用 0 补齐，例如由微分方程

$$2y''(t) + y'(t) + 3y(t) = x(t)$$

描述的系统可以表示为：

```
>> b = [1];
>> a = [2 1 3];
>> sys = tf(b,a);
```

而微分方程由

$$y''(t) + y'(t) + y(t) = x''(t) - x(t)$$

描述的系统则要表示成：

```
>> b = [1 0 -1];
>> a = [1 1 1];
>> sys = tf(b,a);
```

2) 连续时间系统的零状态响应

零状态响应指系统的初始状态为零，仅由输入信号所引起的响应。MATLAB 提供了一个用于求解零状态响应的函数 `lsim`，其调用格式如下：

`lsim(sys,x,t)` 绘出输入信号及响应的波形，`x` 和 `t` 表示输入信号数值向量及其时间向量。

`y = lsim(sys,x,t)` 这种调用格式不绘出波形，而是返回响应的数值向量。

例 8-6 绘出由微分方程 $y''(t) + 2y'(t) + 2y(t) = 2x'(t) + x(t)$ 描述的连续系统，对输入信号 $x(t) = e^{-2t}u(t)$ 的零状态响应波形。

解：这里通过 MATLAB 的 `lsim` 函数解决该问题，代码如下：

```
>> b = [2 1];
>> a = [1 2 2];
>> sys = tf(b,a);    %创建系统模型
>> t = 0:0.01:10; x = exp(-2*t);    %输入信号
>> lsim(sys,x,t);    %绘制零状态响应波形
```

执行上述命令绘制的零状态响应时域波形图如图 8-13 所示。

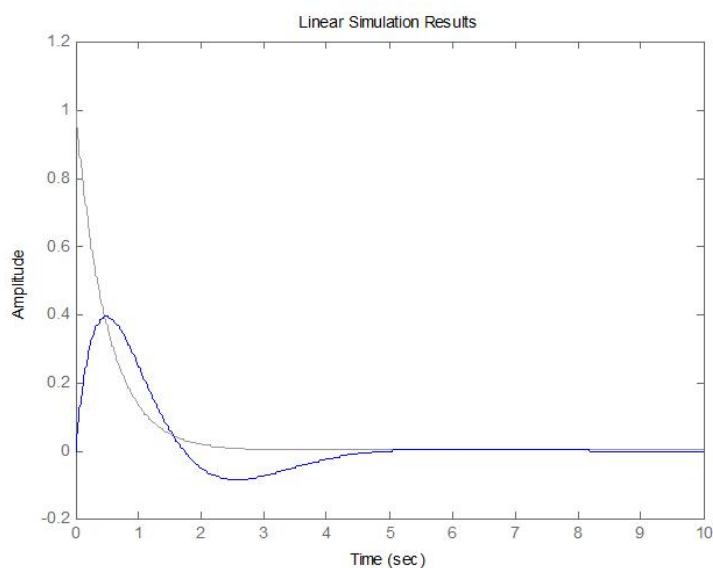


图 8-13 零状态响应时域波形

3) 连续时间系统的冲激响应与阶跃响应

MATLAB 提供了函数 `impulse` 来求指定时间范围内, 由模型 `sys` 描述的连续时间系统的单位冲激响应。`impulse` 函数的基本调用格式如下

`impulse(sys)` 在默认的时间范围内绘出系统冲激响应的时域波形

`impulse(sys,T)` 绘出系统在 $0 - T$ 范围内冲激响应的时域波形

`impulse(sys,ts:tp:te)` 绘出系统在 $ts - te$ 范围内, 以 `tp` 为时间间隔取样的冲激响应波形

`[y,t] = impulse(...)` 该调用格式不绘出冲激响应波形, 而是返回冲激响应的数值向量及其对应的时间向量。

函数 `step` 用于求解单位阶跃响应, 函数 `step` 同样也有如下几种调用格式:

`step(sys)`

`step(sys,T)`

`step(sys,ts:tp:te)`

`[y,t] = step(...)`

各种调用格式参数所代表的意思可参考上述 `impulse` 函数。

例 8-7 绘出例 8-6 微分方程所描述连续系统的冲激响应波形。

解: 利用 MATLAB 的 `impulse` 函数可以绘出系统的冲激响应波形, 具体代码如下:

```
>> b = [2 1];  
>> a = [1 2 2];  
>> sys = tf(b,a);    %创建系统模型  
>> subplot(121);  
>> impulse(sys);      %绘制冲激响应波形  
>> subplot(122);  
>> step(sys);          %绘制阶跃响应波形
```

执行上述命令绘制的波形如图 8-14 所示。

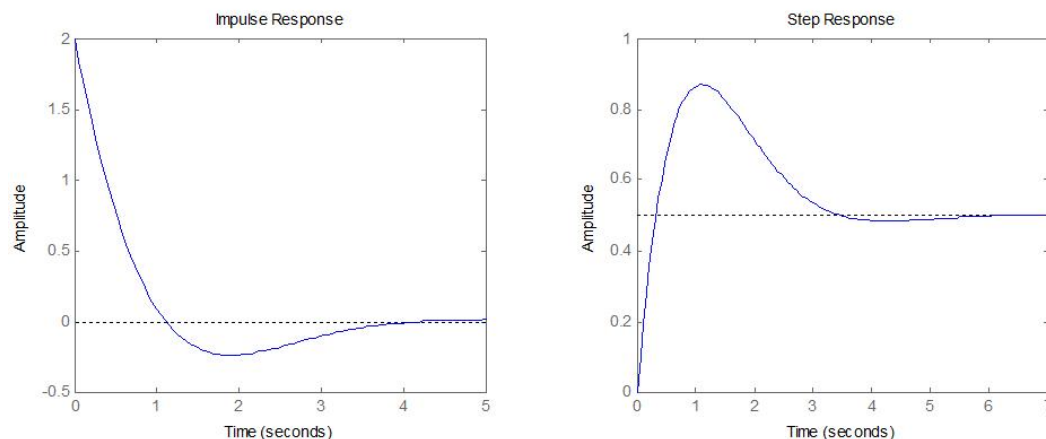


图 8-14 系统的冲激响应和阶跃响应时域波形

2. 离散时间系统时域分析的 MATLAB 实现

1) 离散时间系统的 MATLAB 表示

LTI 离散系统通常可以由系统差分方程描述, 设描述系统的差分方程为:

$$a_0 y(n) + a_1 y(n-1) + \cdots + a_N y(n-M) = b_0 x(n) + b_1 x(n-1) + \cdots + b_M x(n-N) \quad (8-3)$$

则在 MATLAB 里, 我们可以用如下两个向量来表示这个系统:

$$b = [b_0, b_1, \dots, b_M];$$

$$a = [a_0, a_1, \dots, a_N];$$

2) 离散时间系统对任意输入的响应

MATLAB 提供了求 LTI 离散系统响应的专用函数 `filter`，该函数用于求取由差分方程所描述的离散时间系统在指定时间范围内对输入序列所产生的响应，该函数基本调用格式为

$$y = \text{filter}(b, a, x)$$

其中 x 为输入序列， y 为输出序列，输出序列 y 对应的时间区间与 x 对应的时间区间相同。

例 8-8 已知描述离散系统的差分方程为

$$6y(n) - 5y(n-1) + 2y(n-2) = x(n) + x(n-2)$$

且该系统的输入序列为

$$x(n) = \left(\frac{3}{4}\right)^n u(n)$$

试用 MATLAB 绘出输入序列 $x(n)$ 的时域波形，求出系统响应 $y(n)$ 在 0~20 时间点上的值，并绘出系统响应的时域波形。

解：这个问题可以调用 `filter` 函数来实现，实现上述过程的 MATLAB 代码如下：

```
%Eg_8_8.m
b=[1 0 1];
a=[6 5 2];           %创建系统模型
n=0:20;
x=(3/4).^n;          %输入序列
disp('输出序列 0~20 时间点上的值如下：')
y=filter(b,a,x)       %计算输出序列
subplot(121);
stem(n,x,'filled');   %绘制输入序列波形
xlabel('n');
title('x(n)');
subplot(122);
stem(n,y,'filled');   %绘制输出序列波形
xlabel('n');
title('y(n)');
```

运行上述代码，在命令窗口中将显示输出序列 0~20 时间点上的值如下：

y =

Columns 1 through 9

0.1667 -0.0139 0.2164 0.0196 0.0580 0.0550 0.0172 0.0291 0.0163

Columns 10 through 18

0.0114 0.0111 0.0065 0.0056 0.0042 0.0029 0.0024 0.0017 0.0013

Columns 19 through 21

0.0010 0.0007 0.0005

绘出的系统输入及输出序列时域波形如图 8-15 所示。

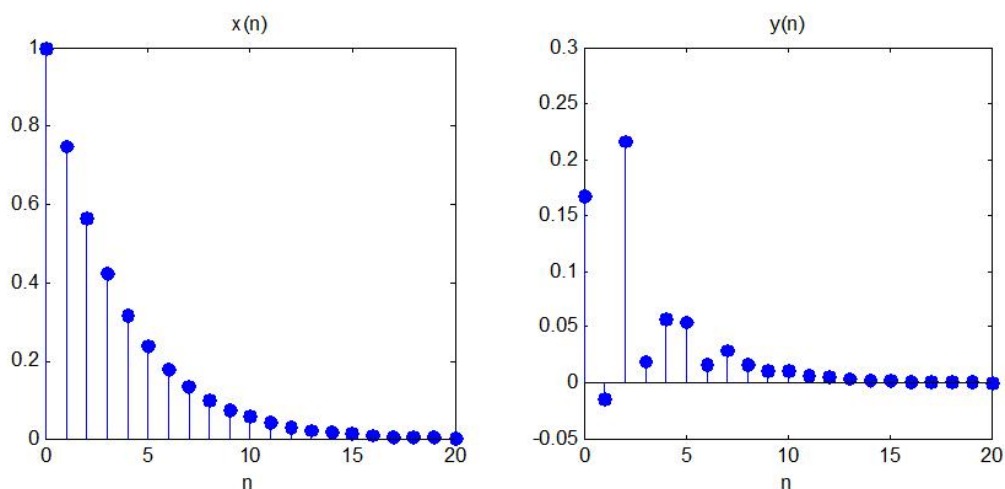


图 8-15 系统输入及输出序列时域波形

3) 离散时间系统的单位抽样响应

MATLAB 提供了函数 `impz` 来求指定时间范围内, 由向量 **b** 和 **a** 描述的离散时间系统的单位抽样响应, 具体调用格式如下:

`impz(b,a)` 在默认的时间范围内绘出系统单位抽样响应的时域波形

`impz(b,a,N)` 绘出系统在 $0 \sim N$ 范围内单位抽样响应的时域波形

`impz(b,a,ns:ne)` 绘出系统在 $ns \sim ne$ 范围内的单位抽样响应波形

`[y,t] = impz(...)` 该调用格式不绘出单位抽样响应波形, 而是返回单位抽样响应的数值向量及其对应的时间向量。

例 8-9 已知描述离散系统的差分方程为

$$y(n) + 3y(n-1) + 2y(n-2) = x(n)$$

试用 MATLAB 绘出系统单位抽样响应在 $0 \sim 10$ 之间的时域波形。

解: 利用用 `impz` 函数来实现, 实现上述过程的 MATLAB 代码如下:

```
>> b=[1];
```

```
>> a=[1 3 2];
```

```
>> impz(b,a,0:10);
```

代码运行结果如图 8-16 所示。

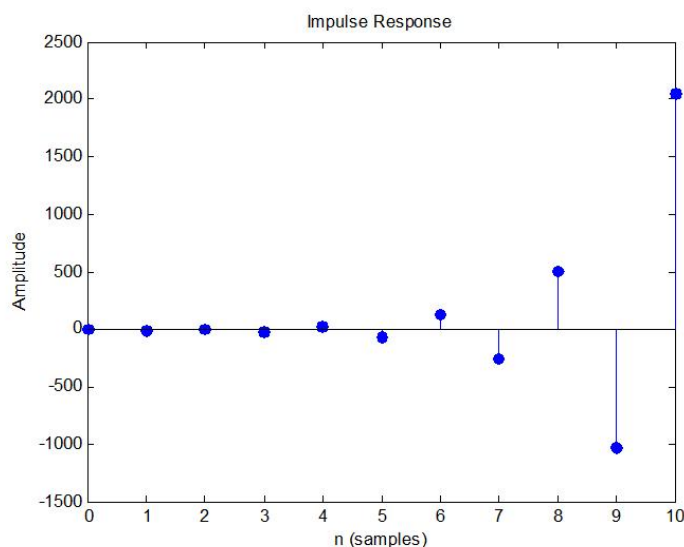


图 8-16 系统的单位抽样响应

3. 卷积和与卷积积分

1) 离散时间序列的卷积和

卷积和是离散系统时域分析的基本方法之一，离散时间序列 $x_1(n)$ 和 $x_2(n)$ 的卷积和 $x(n)$ 定义如下

$$x(n) = x_1(n) * x_2(n) = \sum_{k=-\infty}^{+\infty} x_1(k)x_2(n-k) \quad (8-4)$$

对已离散 LTI 系统，设其输入信号为 $x(n)$ ，单位抽样响应为 $h(n)$ ，则其零状态响应 $y(n)$ 为

$$y(n) = x(n) * h(n) \quad (8-5)$$

即离散 LTI 系统的零状态响应可以表示成输入信号 $x(n)$ 和单位抽样响应 $h(n)$ 的卷积。因此，离散时间序列的卷积和计算对我们进行离散系统时域分析具有重要的意义。

MATLAB 的 conv 函数可以用来求两个离散序列的卷积和，调用格式为 $x=\text{conv}(x1,x2)$ 。例如：

```
>> x1 = ones(1,3);  
>> x2 = [1 2 3 4];  
>> x = conv(x1,x2)  
x =
```

1 3 6 9 7 4

然而，我们不难发现，此例中序列 $x1$ 、 $x2$ 、 x 都没有时间信息，而实际上我们要考察的信号不仅需要知道各时刻的取值，还需要知道其对应的时间序列。因此，我们还需要根据序列 $x1$ 、 $x2$ 对应的时间序列确定卷积结果 x 对应的时间序列。

设 $x1$ 、 $x2$ 为两个在有限时间区间内非零的离散时间序列，即序列 $x1$ 在区间 $n1 \sim n2$ 内非零，序列 $x2$ 在区间 $m1 \sim m2$ 内非零，则序列 $x1$ 的时域宽度为 $L1=n2-n1+1$ ，序列 $x2$ 的时域宽度为 $L2=m2-m1+1$ 。由卷积和的定义可知，卷积和序列 x 的时域宽度为 $L=L1+L2-1$ ，且只在区间 $(n1+m1) \sim (n1+m1) + (L1+L2-2)$ 非零。

2) 连续时间信号的卷积积分

卷积积分是连续系统是与分析的有效方法和工具之一，连续时间信号 $x_1(t)$ 和 $x_2(t)$ 的卷积积分 $x(t)$ 定义如下

$$x(t) = x_1(t) * x_2(t) = \int_{-\infty}^{+\infty} x_1(\tau)x_2(t-\tau)d\tau \quad (8-6)$$

对已连续 LTI 系统，设其输入信号为 $x(t)$ ，单位冲激响应为 $h(t)$ ，其零状态响应为 $y(t)$ ，则有

$$y(t) = x(t) * h(t) \quad (8-7)$$

即连续 LTI 系统的零状态响应可以表示成输入信号和单位冲激响应的卷积。因此，连续时间信号卷积积分对连续系统的时域分析具有非常重要的意义。

利用 MATLAB 可以采用数值计算的方法近似计算卷积积分。卷积积分可用求和运算来实现：

$$x(t) = x_1(t) * x_2(t) = \int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau = \lim_{\Delta \rightarrow 0} \sum_{k=-\infty}^{\infty} x_1(k\Delta)x_2(t-k\Delta) \cdot \Delta \quad (8-8)$$

现在考虑只求 $t = n\Delta$ 时 $x(t)$ 的值 $x(n\Delta)$ ，则由式 (2-3) 可得：

$$x(n\Delta) = \Delta \cdot \sum_{k=-\infty}^{\infty} x_1(k\Delta)x_2((n-k)\Delta) \quad (8-9)$$

当 Δ 足够小， $x(n\Delta)$ 就是 $x(t)$ 的数值近似。我们可以利用计算离散序列卷积和的 `conv` 函数来计算卷积积分，具体步骤如下：

①将连续时间信号 $x_1(t)$ 和 $x_2(t)$ 以时间间隔 Δ 进行取样，得到离散序列 $x_1(n\Delta)$ 和 $x_2(n\Delta)$ 。

②构造离散序列 $x_1(t)$ 和 $x_2(t)$ 对应的时间向量 t_1 和 t_2 。

③调用函数 `conv` 计算卷积积分在 $t = n\Delta$ 时的近似采样值 $x(n\Delta)$ 。

④构造离散序列 $x(n\Delta)$ 对应的时间向量 n 。

根据上述过程可以自定义一个用于计算卷积积分的通用函数 `sconv`，函数源代码如下：

```
function [x,t]=sconv(x1,x2,t1,t2,dt)
%计算连续信号卷积积分 x(t)=x1(t)*x2(t)
%-----数值方法-----
% x: 卷积积分 x(t)对应的非零样值向量
% t: f(t)的对应时间向量
% x1: x1(t)非零样值向量
% x2: x2(t)的非零样值向量
% t1: x1(t)的对应时间向量
% t2: x2(t)的对应时间向量
% dt: 取样时间间隔
%
x = conv(x1,x2); %计算序列 x1 与 x2 的卷积和 x
x = x*dt;
t0 = t1(1)+t2(1); %计算序列 x 非零样值的起点位置
l = length(x1)+length(x2)-2; %计算卷积和 f 的非零样值的宽度
t = t0:dt:(t0+ l*dt); %确定卷积和 x 非零样值的时间向量
```

下面我们通过一个例子来说明这个函数的使用方法。

例 8-10 已知两个连续时间信号如图 8-17 所示，绘出这两个信号卷积的图形。

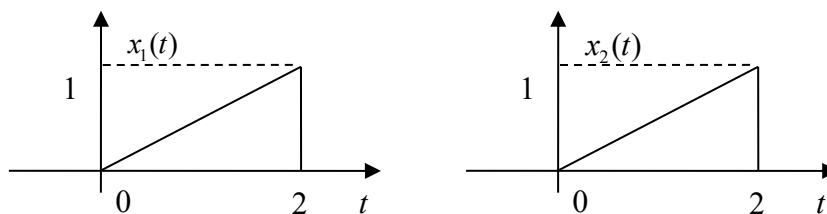


图 8-17 例 8-10 已知连续时间信号

解：绘制上述两个信号卷积的图形的 MATLAB 代码如下：

```
%Eg_8_10.m
dt=0.01;
t1=0:dt:2;
x1=0.5*t1;
```

```

x2=x1;t2=t1;
subplot(321);
plot(t1,x1);
xlabel('t(s)'); title('x_1(t)');
subplot(322);
plot(t2,x2);
xlabel('t(s)'); title('x_2(t)');
[x,t]=sconv(x1,x2,t1,t2,dt);    %调用自定义函数 sconv 计算卷积积分
subplot(312);
plot(t,x); xlabel('t(s)');
title('x(t)=x_1(t)*x_2(t)    \Deltat=0.01');
%改变采样间隔，再次计算
dt=0.5;
t1=0:dt:2;
x1=0.5*t1;
x2=x1;t2=t1;
[x,t]=sconv(x1,x2,t1,t2,dt);    %调用自定义函数 sconv 计算卷积积分
subplot(313);
plot(t,x);
xlabel('t(s)'); title('x(t)=x_1(t)*x_2(t)    \Deltat=0.5');

```

代码运行结果如图 8-18 所示。可见利用 sconv 函数计算连续时间信号的卷积积分的效果取决于信号采样时间间隔的大小，当采样间隔足够小时，计算结果才是连续时间信号卷积积分的较好近似。

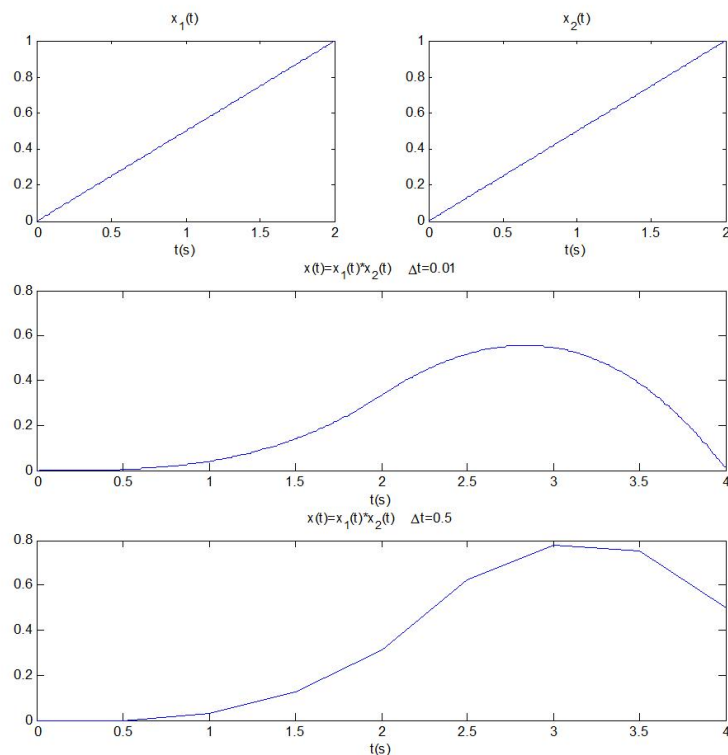


图 8-18 例 8-10 连续时间信号卷积积分的结果

三、实验内容

1. 已知描述模拟低通、高通、带通和带阻滤波器的微分方程如下，试采用 MATLAB 绘出各系统的单位冲激响应和单位阶跃响应波形。

$$(1) \quad y''(t) + \sqrt{2}y'(t) + y(t) = 2x(t)$$

$$(2) \quad y''(t) + \sqrt{2}y'(t) + y(t) = 2x''(t)$$

$$(3) \quad y''(t) + y'(t) + y(t) = 2x'(t)$$

$$(4) \quad y''(t) + y'(t) + y(t) = 2x''(t) + 2x(t)$$

2. 已知某系统可以由如下微分方程描述

$$y''(t) + y'(t) + 6y(t) = 2x(t)$$

(1) 请利用 MATLAB 绘出该系统冲激响应和阶跃响应的时域波形；

(2) 根据冲激响应的时域波形分析系统的稳定性；

(3) 如果系统的输入为 $x(t) = e^{-2t}u(t)$ ，求系统的零状态响应。

3. 已知描述离散系统的微分方程如下，试采用 MATLAB 绘出各系统的单位抽样响应，并根据单位抽样响应的时域波形分析系统的稳定性。

$$(1) \quad y(n) + y(n-1) + 2y(n-2) = x(n)$$

$$(2) \quad y(n) - 0.5y(n-1) + 0.8y(n-2) = 3x(n) - 2x(n-1)$$

4. 已知系统可以由如下差分方程描述

$$y(n) + y(n-1) + 0.25y(n-2) = 2x(n)$$

试采用 MATLAB 绘出该系统的单位抽样响应波形和单位阶跃响应波形。

5. 采用 MATLAB 计算如下两个序列的卷积，并绘出图形

$$x_1(n) = \{1, \underset{\uparrow}{2}, 1, 1\} \quad x_2(n) = \begin{cases} 1 & -3 \leq n \leq 3 \\ 0 & \text{其他} \end{cases}$$

6. 已知某 LTI 离散系统，其单位抽样响应 $h(n) = \sin(0.2n)$ ， $n \geq 0$ ，系统的输入为 $x(n) = \sin(0.4n)$ ， $n \geq 0$ ，计算当 $n = 0, 1, 2, \dots, 40$ 时系统的零状态响应 $y(n)$ ，绘出 $x(n)$ 、 $h(n)$ 和 $y(n)$ 时域波形。

7. 已知两个连续时间信号如图 8-19 所示，试采用 MATLAB 求这两个信号的卷积。

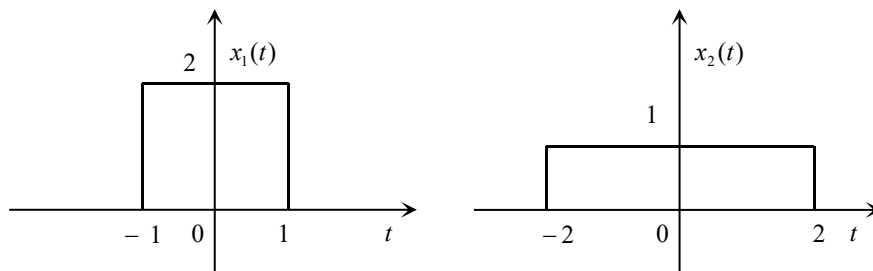


图 8-19

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

实验3 信号的频域分析

(综合型实验)

一、实验目的

1. 深入理解信号频谱的概念，掌握信号的频域分析方法。
2. 观察典型周期信号和非周期信号的频谱，掌握其频谱特性。

二、实验原理与方法

1. 连续周期信号的频谱分析

如果周期信号满足狄里赫利（Dirichlet）条件，就可以展开为傅里叶级数的形式，即

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{jk\omega_0 t} \quad (8-10)$$

$$c_k = \frac{1}{T_0} \int_{T_0} x(t) e^{-jk\omega_0 t} dt \quad (8-11)$$

其中 T_0 表示基波周期， $\omega_0 = 2\pi/T_0$ 为基波频率， $\int_{T_0} (\cdot)$ 表示任一个基波周期内的积分。

式（8-10）和式（8-11）定义为周期信号复指数形式的傅里叶级数，系数 c_k 称为 $x(t)$ 的傅里叶系数。周期信号的傅里叶级数还可以由三角函数的线性组合来表示，即

$$x(t) = a_0 + \sum_{k=1}^{\infty} a_k \cos k\omega_0 t + \sum_{k=1}^{\infty} b_k \sin k\omega_0 t \quad (8-12)$$

其中

$$a_0 = \frac{1}{T_0} \int_{T_0} x(t) dt, \quad a_k = \frac{2}{T_0} \int_{T_0} x(t) \cos k\omega_0 t dt, \quad b_k = \frac{2}{T_0} \int_{T_0} x(t) \sin k\omega_0 t dt \quad (8-13)$$

式（8-12）中同频率的正弦项和余弦项可以合并，从而得到三角函数形式的傅里叶级数，即

$$x(t) = A_0 + \sum_{k=1}^{\infty} A_k \cos(k\omega_0 t + \theta_k) \quad (8-14)$$

其中

$$A_0 = a_0, \quad A_k = \sqrt{a_k^2 + b_k^2}, \quad \theta_k = -\arctg \frac{b_k}{a_k} \quad (8-15)$$

可见,任何满足狄里赫利条件的周期信号都可以表示成一组谐波关系的复指数函数或三角函数的叠加。一般来说周期信号表示为傅里叶级数时需要无限多项才能完全逼近原信号,但在实际应用中经常采用有限项级数来替代,所选项数越多就越逼近原信号。

例 8-11 周期矩形波如图 8-20 所示, 求出该信号的傅里叶级数, 并采用 MATLAB 实现其各次谐波叠加。

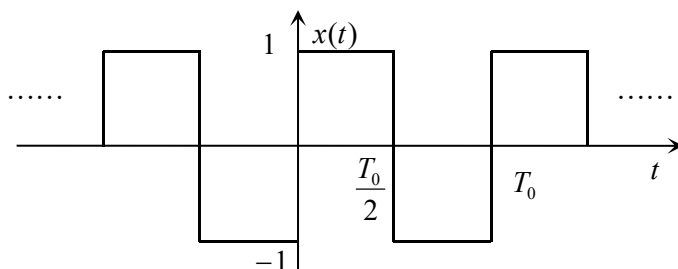


图 8-20 周期矩形波信号

解: 根据式 (8-12) 和式 (8-13) 可以求得

$$a_k = 0, \quad b_k = \begin{cases} \frac{4}{k\pi} & k \text{ 为奇数} \\ 0 & k \text{ 为偶数} \end{cases}$$

因此, 信号 $x(t)$ 的傅里叶级数为

$$x(t) = \frac{4}{\pi} \left(\sin \omega_0 t + \frac{1}{3} \sin 3\omega_0 t + \frac{1}{5} \sin 5\omega_0 t + \dots \right)$$

利用 MATLAB 可以形象的观察前 N 次谐波合成信号的近似波形, 具体程序代码如下:

```
%Eg_8_11.m
t = -1.5:0.01:1.5;
N = input('N='); %从键盘输入 N 的取值
x = zeros(size(t)); %初始化 x 变量
for n = 1:2:N
    x = x + (4/(pi*n))*sin(2*pi*n*t);
end
plot(t,x);
xlabel('Time(sec)');
title(['N=' num2str(N)]);
```

图 8-21 分别画出了 $N = 7$ 和 $N = 28$ 时的波形, 这里取 $T_0 = 1$, 可见傅里叶级数所取的项数越多, 合成的波形越接近与原来的矩形波信号。

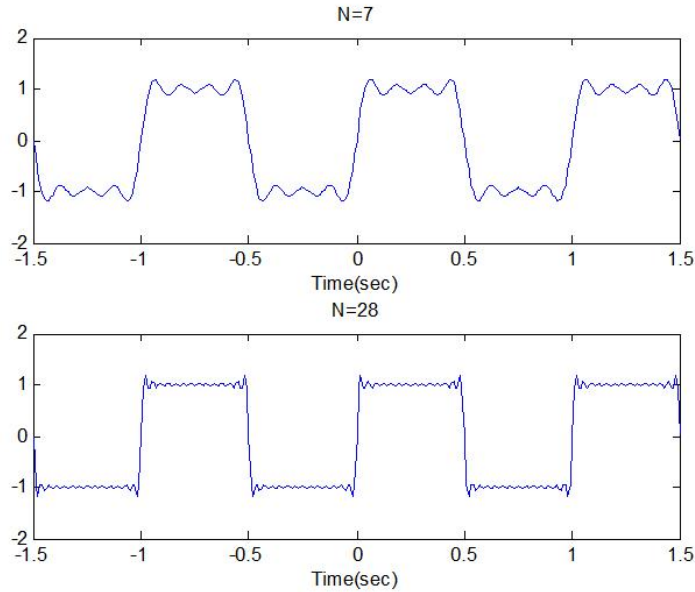


图 8-21 前 N 次谐波合成的近似波形

如上所述，周期信号可以表示为如式 (8-10) 或 (8-14) 所示的傅里叶级数形式，我们将各次谐波的幅度 A_k 和相位 θ_k 或 c_k 的幅度 $|c_k|$ 和相位 $\arg c_k$ 随频率变化的规律画出来，这就是频谱图。

例 8-12 利用 MATLAB 画出如图 8-22 所示的周期三角波信号的频谱。

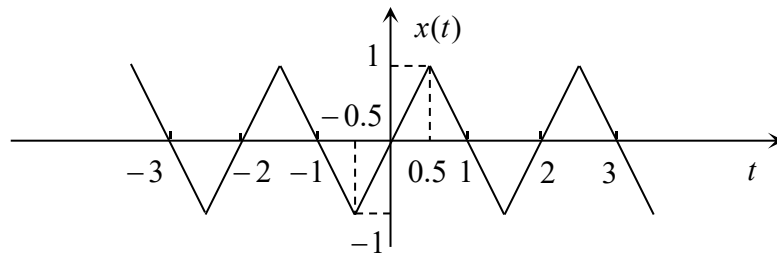


图 8-22 周期三角波信号

解：根据式 (8-11) 可以计算出，该周期三角波信号的傅里叶系数为：

$$c_k = \begin{cases} \frac{-4j}{k^2 \pi^2} \sin\left(\frac{k\pi}{2}\right) & n \neq 0 \\ 0 & n = 0 \end{cases}$$

则画出该信号频谱的 MATLAB 代码如下：

```
%Eg_8_12.m
N = 10;
%计算 n 从 -N 到 -1 的傅里叶系数
n1 = -N:-1;
c1 = -4*j*sin(n1*pi/2)/pi^2./n1.^2;
%计算 n=0 时的傅里叶系数
c0 = 0;
%计算 n 从 1 到 N 的傅里叶系数
n2 = 1:N;
c2 = -4*j*sin(n2*pi/2)/pi^2./n2.^2;
```

```

cn = [c1 c0 c2];
n = -N:N;
subplot(211);
stem(n,abs(cn),'filled');
xlabel('\omega/\omega_0');
title('Magnitude of ck');
subplot(212);
stem(n,angle(cn),'filled');
xlabel('\omega/\omega_0');
title('Phase of ck');

```

以上程序运行结果如图 8-23 所示。

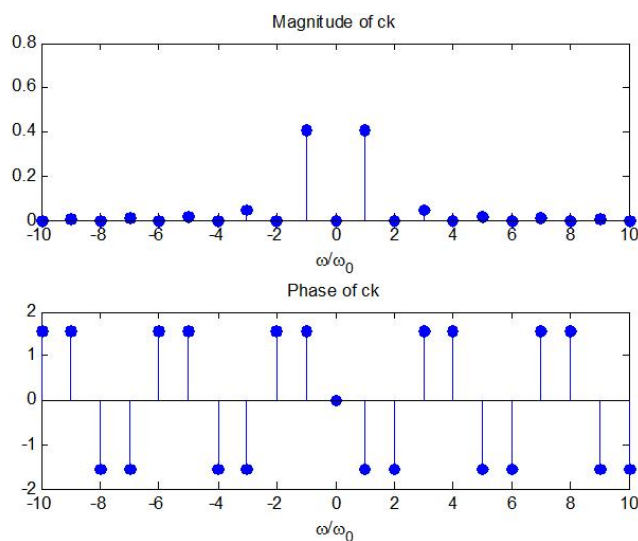


图 8-23 周期三角波信号的频谱

2. 连续非周期信号的频谱分析

对于非周期连续时间信号，信号的傅里叶变换和傅里叶逆变换定义为：

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt \quad (8-16)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega)e^{j\omega t} d\omega \quad (8-17)$$

式 (8-16) 和式 (8-17) 把信号的时域特性和频域特性联系起来，确立了非周期信号 $x(t)$ 和频谱 $X(\omega)$ 之间的关系。

采用 MATLAB 可以方便的求取非周期连续时间信号的傅里叶变换，这里我们介绍常用的几种方法。

1) 符号运算法

MATLAB 的符号数学工具箱提供了直接求解傅里叶变换和反变换的函数，fourier 函数和 ifourier 函数，基本调用格式为

$$X = \text{fourier}(x)$$

$$x = \text{ifourier}(X)$$

默认的时域变量为 t ，频域变量为 w 。

例如求 $x(t) = e^{-2|t|}$ 的傅里叶变换，MATLAB 代码和运行结果如下：

```
>> syms t
>> x = exp(-2*abs(t));
>> X = fourier(x)
X =
4/(w^2 + 4)
```

因此，傅里叶变换的结果为

$$X(\omega) = \frac{4}{4 + \omega^2}$$

也可以利用 `int` 函数直接根据式 (8-16) 求解傅里叶变换，我们以三角波为例来学习这种方法。

例 8-13 利用 MATLAB 的符号运算方法计算如下三角波信号的频谱

$$x(t) = \begin{cases} 1-|t| & |t| \leq 1 \\ 0 & |t| > 1 \end{cases}$$

解：可由如下命令计算傅里叶变换：

```
%Eg_8_13.m
syms t w
X=int((1-abs(t))*exp(-j*w*t),t,-1,1);
ezplot(abs(X),[-6*pi,6*pi]);
grid on;
xlabel('\omega');
ylabel('Magnitude');
title('|X(\omega)|');
```

程序运行结果如图 8-24 所示。

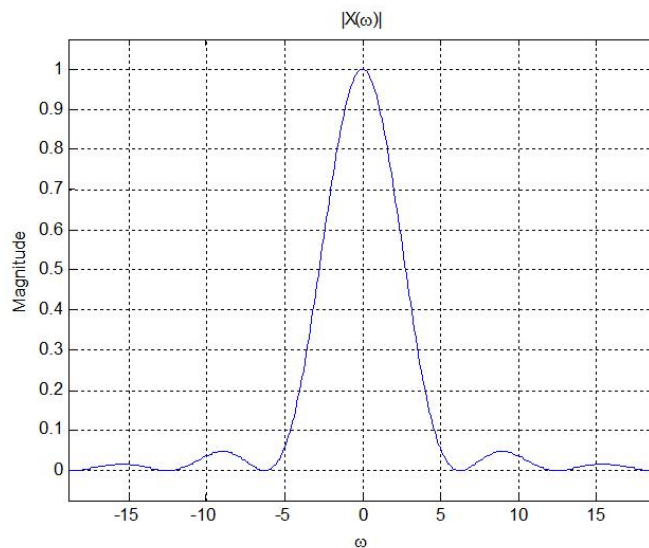


图 8-24 符号运算法计算三角波信号频谱

2) 数值积分法

除了采用符号运算的方法外，我们还可以利用 MATLAB 的 `quad` 函数，采用数值积分的方法来进行连续信号的频谱分析。`quad` 函数是一个用来计算数值积分的函数。利用 `quad` 函数可以计算非周期连续时间信号的频谱。`quad` 函数的一般调用格式为：

```
y = quad(fun,a,b)
```

```
y = quad(fun,a,b,TOL,TRACE,p1,p2,...)
```

其中 `fun` 指定被积函数，可以采用 `inline` 命令来创建，也可以通过传递函数句柄的形式来指定，`a`、`b` 表示定积分的下限和上限，`TOL` 表示允许的相对或绝对积分误差，`TRACE` 表示以被积函数的点绘图形式来跟踪该函数的返回值，如果 `TOL` 和 `TRACE` 为空矩阵，则使用缺省值，“`p1,p2,...`”表示被积函数除时间 `t` 之外所需的其他额外输入参数。

我们通过一个例子来学习 `quad` 函数的使用方法。

例 8-14 求 $f(x) = 1/(x^3 - 2x - 5)$ 从 0 到 2 的积分。

解：我们可以使用 `inline` 命令来创建被积函数然后计算积分，MATLAB 代码及运行结果如下：

```
>> f = inline('1./(x.^3-2*x-5)');  
>> F = quad(f,0,2)  
F =  
    -0.4605
```

我们还可以通过传递函数句柄的形式来指定被积函数，调用格式如下

```
F = quad(@myfun,0,2)
```

其中 `@` 表示取函数句柄，`myfun` 指根据被积函数编写的 MATLAB 函数，例如本例中可以将被积函数写成一个函数：

```
function y = myfun(x)  
y = 1./(x.^3-2*x-5);
```

将该函数保存到 `myfun.m` 文件中，然后调用如下命令：

```
>> F = quad(@myfun,0,2)  
F =  
    -0.4605
```

两种方法计算的结果一样。

利用 MATLAB 的 `quad` 函数，我们可以根据式（8-16）采用数值积分的方法来近似计算连续时间傅里叶变换。

例 8-15 利用 MATLAB 的数值积的方法近似计算例 8-13 中三角波信号的频谱。

解：为了利用 `quad` 函数计算频谱，先定义如下被积函数：

```
function y = sf(t,w)  
y = (abs(t)<=1).*(1-abs(t)).*exp(-j*w*t);
```

该函数表示被积函数，该函数保存在 `sf.m` 文件中。接下来我们计算被积函数的积分，即可得到该三角波信号频谱的近似数值解。

计算三角波信号频谱近似数值解的 MATLAB 程序如下：

```
%Eg_8_15.m  
w = linspace(-6*pi,6*pi,512);    %对频率轴取样  
N = length(w);  
X = zeros(1,N);    %初始化频谱序列  
%每个频率取样点上对被积函数积分  
for k = 1:N  
    X(k) = quad(@sf,-1,1,[],[],w(k));  
end  
plot(w,abs(X));  
grid on;  
xlabel('\omega');
```

ylabel('Magnitude');

title('|X(\omega)|');

程序运行结果如图 8-25 所示，与例 8-13 的计算结果一致。

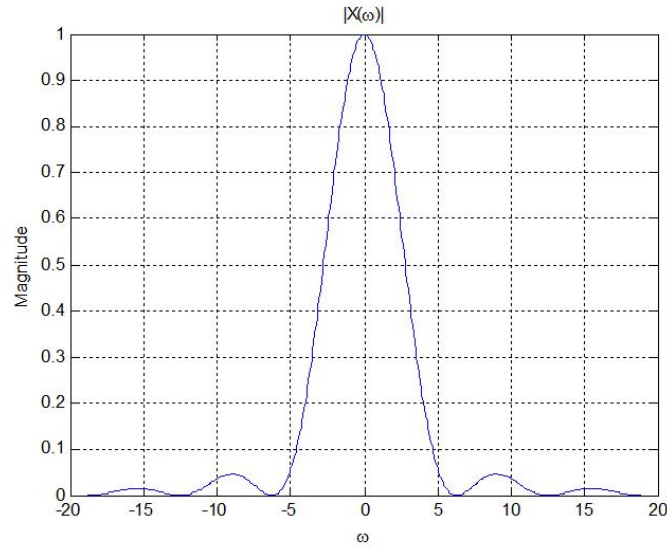


图 8-25 数值积分法近似计算三角波信号频谱

3) 数值近似法

我们还可以利用 MATLAB 的数值计算的方法近似计算连续时间傅里叶变换。傅里叶变换 $X(\omega)$ 可以由式 (8-18) 近似计算

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt = \lim_{\Delta \rightarrow 0} \sum_{k=-\infty}^{\infty} x(k\Delta)e^{-j\omega k\Delta} \Delta \quad (8-18)$$

当 $x(t)$ 为时限信号，且 Δ 足够小，式 (8-18) 可以演变为

$$X(\omega) = \Delta \sum_{k=a}^b x(k\Delta)e^{-jk\omega\Delta} \quad (8-19)$$

而式 (8-19) 中求和部分又可以表示成一个行向量和一个列向量的乘积

$$\sum_{k=a}^b x(k\Delta)e^{-jk\omega\Delta} = [x(a\Delta), x((a+1)\Delta), \dots, x(b\Delta)] \cdot \begin{bmatrix} e^{-j \cdot a\Delta \cdot \omega} \\ e^{-j \cdot (a+1)\Delta \cdot \omega} \\ \vdots \\ e^{-j \cdot b\Delta \cdot \omega} \end{bmatrix} \quad (8-20)$$

式 (8-20) 可以很方便的利用 MATLAB 实现。

例 8-16 利用 MATLAB 的数值近似方法计算例 8-13 中三角波信号的频谱。

解：根据上述介绍的方法，可以采用如下 MATLAB 代码计算三角波信号的频谱

```
%Eg_8_16.m
```

```
%对频率轴取样
```

```
w = linspace(-6*pi,6*pi,512);
```

```
%对信号取样
```

```
dt=0.001;
```

```
t=-1:dt:1;
```

```
x=1-abs(t);
```



```
%计算傅里叶变换
X=x*exp(-j*t*w)*dt;
plot(w,abs(X)); grid on;
xlabel('\omega');
ylabel('Magnitude');
title('|X(\omega)|');
```

程序运行结果如图 8-26 所示，与例 8-13、例 8-15 的计算结果一致。

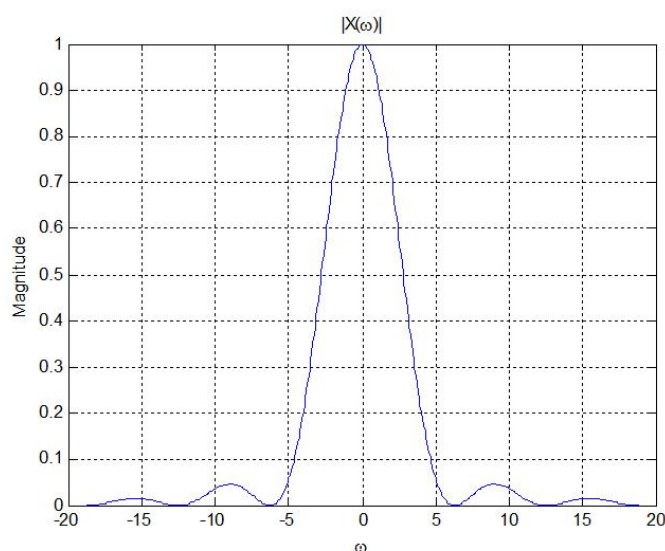


图 8-26 数值近似法计算三角波信号频谱

3. 离散周期时间信号的频域分析

基波周期为 N 的周期序列 $x(n)$ 可以用 N 个成谐波关系的复指数序列的加权和表示,即

$$x(n) = \sum_{k=\langle N \rangle} c_k e^{jk(2\pi/N)n} \quad (8-21)$$

这里 $k=\langle N \rangle$ 表示求和仅需包括一个周期内的 N 项,周期序列在一个周期内的求和与起点无关。将周期序列表示成式 (8-21) 的形式,称为离散傅里叶级数,而系数 c_k 则称为离散傅里叶系数。离散傅里叶系数 c_k 可以由式 (4-2) 确定

$$c_k = \frac{1}{N} \sum_{k=\langle N \rangle} x(n) e^{-jk(2\pi/N)n} \quad (8-22)$$

傅里叶系数 c_k 也称为 $x(n)$ 的频谱系数,而且可以证明 c_k 是以 N 为周期的离散频率序列。这说明了周期的离散时间函数对应于频域为周期的离散频率。

这里,我们用周期 N 与傅里叶系数 c_k 的乘积来表示周期离散时间信号的频谱,即

$$X(k) = N \cdot c_k = \sum_{k=\langle N \rangle} x(n) e^{-jk(2\pi/N)n} \quad (8-23)$$

$X(k)$ 可以利用 MATLAB 提供的函数 `fft` 用来计算,调用格式为

$$X = \text{fft}(x)$$

该函数返回 $X(k)$ 一个周期内的值,其中 x 表示 $x(n)$ 一个周期内的样本值。

例 8-17 求周期序列 $x(n) = \cos(n\pi/6)$ 的频谱。

解：该信号基波周期为 12，可在 $0 \leq k \leq 11$ 一个周期内进行计算，实现代码如下：

```
%Eg_8_17.m
n = 0:11;
x = cos(pi*n/6);
%使用 fft 函数计算频谱一个周期的值
X = fft(x);
subplot(211);
stem(n,x,'filled');
xlabel('n');
title('x(n)');
subplot(212);
stem(n,X,'filled');
xlabel('k');
title('X(k)');
```

运行结果如图 8-27 所示，图中分别绘制了 $x(n)$ 和 $X(k)$ 一个周期内的波形。

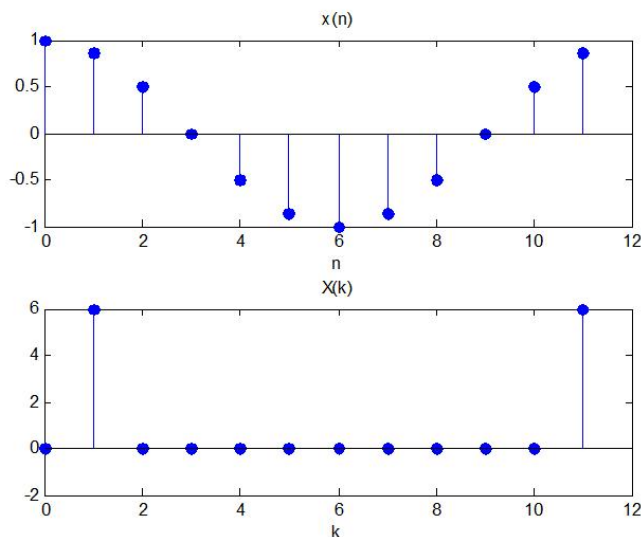


图 8-27 周期余弦序列及其频谱（一个周期内的波形）

4. 离散非周期时间信号的频域分析

非周期序列 $x(n)$ 可以表示成一组复指数序列的连续和

$$x(n) = \frac{1}{2\pi} \int_{2\pi} X(e^{j\Omega}) e^{j\Omega n} d\Omega \quad (8-24)$$

其中

$$X(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\Omega n} \quad (8-25)$$

式（8-25）称为 $x(n)$ 的离散时间傅里叶变换，式（8-24）和式（8-25）确立了非周期离散时间信号 $x(n)$ 及其离散时间傅里叶变换 $X(e^{j\Omega})$ 之间的关系。 $X(e^{j\Omega})$ 是连续频率 Ω 的函数，称为频谱函数，且 $X(e^{j\Omega})$ 是周期的连续频率函数，其周期为 2π 。可见，非周期离散时间函数对应于频域中是一个连续的周期的频率函数。

对于有限长的离散时间序列，式（8-25）可以表示为：

$$X(e^{j\Omega}) = \sum_{n=n_1}^{n_N} x(n)e^{-j\Omega n} = [x(n_1), x(n_2), \dots, x(n_N)] \cdot \begin{bmatrix} e^{-jn_1\Omega} \\ e^{-jn_2\Omega} \\ \vdots \\ e^{-jn_N\Omega} \end{bmatrix} \quad (8-26)$$

式 (8-26) 可以方便的利用 MATLAB 实现。

例 8-18 用 MATLAB 分析图 8-28 所示矩形序列的频谱。

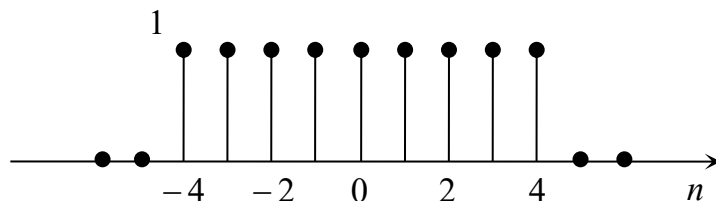


图 8-28 矩形序列

解：根据式 (8-26)，该矩形序列的频谱可以采用如下 MATLAB 代码实现：

```
%Eg_8_18.m
w=-pi:0.01*pi:pi;      %对频率轴取样
n=-4:4;
x=ones(size(n));
X=x*exp(-j*n*w);        %计算频谱
subplot(211);
stem(n,x,'filled');
xlabel('n');
title('x(n)');
subplot(212);
plot(w/pi,abs(X));
xlabel('\Omega/pi');
title('|X(e^{j\Omega})|');
```

代码运行结果如图 8-29 所示。

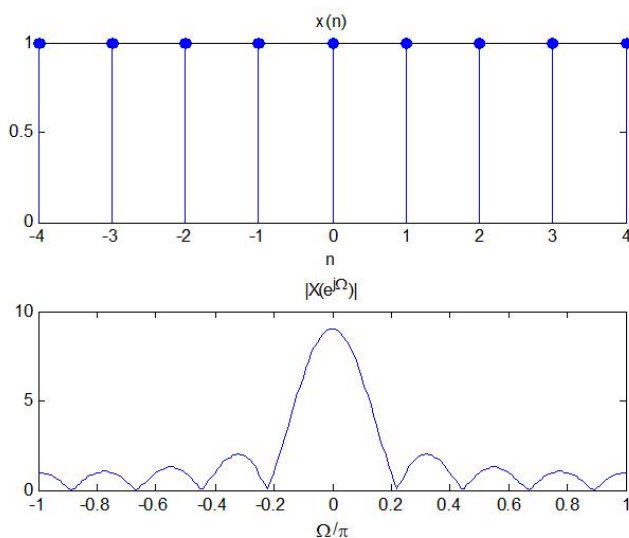


图 8-29 矩形序列及其频谱

三、实验内容

1. 已知 $x(t)$ 是一如图 8-30 所示的周期矩形脉冲信号

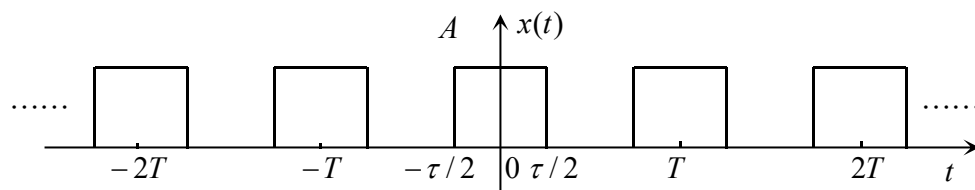


图 8-30

- (1) 计算该信号的傅里叶级数；
- (2) 利用 MATLAB 绘出由前 N 次谐波合成的信号波形，观察随着 N 的变化合成信号波形的变化规律；
- (3) 利用 MATLAB 绘出周期矩形脉冲信号的频谱，观察参数 T 和 τ 变化时对频谱波形的影响。

观察实验结果，思考如下问题：

- Q1-1. 什么是吉伯斯现象？产生吉伯斯现象的原因是什么？
- Q1-2. 以周期矩形脉冲信号为例，说明周期信号的频谱有什么特点。
- Q1-3. 周期矩形脉冲信号的有效频带宽度与信号的时域宽度之间有什么关系？
- Q1-4. 随着矩形脉冲信号参数 τ/T 的变化，其频谱结构（如频谱包络形状、过零点、谱线间隔等）如何变化？

2. 已知 $x(t)$ 是一如图 8-31 所示的矩形脉冲信号

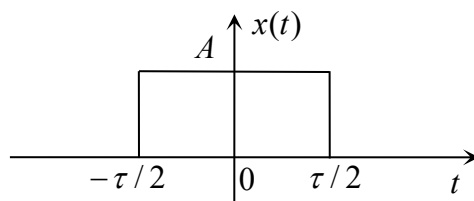


图 8-31

- (1) 求该信号的傅里叶变换；
- (2) 利用 MATLAB 绘出矩形脉冲信号的频谱，观察矩形脉冲宽度 τ 变化时对频谱波形的影响；
- (3) 让矩形脉冲的面积始终等于 1，改变矩形脉冲宽度，观察矩形脉冲信号时域波形和频谱随矩形脉冲宽度的变化趋势。

观察实验结果，思考如下问题：

- Q2-1. 比较矩形脉冲信号和周期矩形脉冲信号的频谱，两者之间有何异同？
- Q2-2. 根据矩形脉冲宽度 τ 变化时频谱的变化规律，说明信号的有效频带宽度与其时域宽度之间有什么关系。当脉冲宽度 $\tau \rightarrow 0$ ，脉冲的面积始终等于 1，其频谱有何特点？

3. 已知 $x(n)$ 是一如图 8-32 所示的周期方波序列

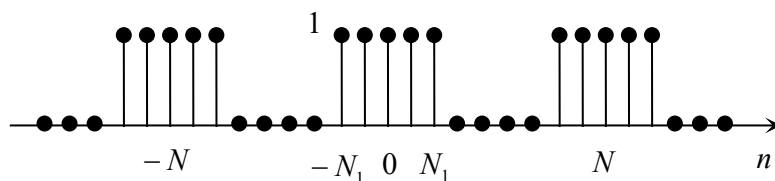


图 8-32

利用 MATLAB 绘制周期方波序列的频谱波形，改变参数 N 和 N_1 的大小，观察频谱波

形的变化趋势。

观察实验结果，思考如下问题：

Q3-1. 以周期方波序列为例，说明周期序列与连续周期信号的频谱有何异同。

Q3-2. 随着周期方波序列占空比的变化，其频谱如何随之变化？

4. 已知一矩形脉冲序列

$$x(n) = \begin{cases} 2 & |n| \leq N_1 \\ 0 & |n| > N_1 \end{cases}$$

利用 MATLAB 绘制周期方波序列的频谱波形，改变矩形脉冲序列的宽度，观察频谱波形的变化趋势。

观察实验结果，思考如下问题：

Q4-1. 随着矩形脉冲序列宽度的变化，其频谱如何随之变化？其宽度与频谱的有效频带宽度有何关系？

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

实验 4 LTI 系统的频域分析

(综合型实验)

一、实验目的

1. 加深对 LTI 系统频率响应基本概念的掌握和理解。
2. 学习和掌握 LTI 系统频率特性的分析方法。

二、实验原理与方法

1. 连续时间系统的频率响应

系统的频率响应定义为系统单位冲激响应 $h(t)$ 的傅里叶变换，即

$$H(\omega) = \int_{-\infty}^{\infty} h(\tau) e^{-j\omega\tau} d\tau \quad (8-27)$$

若 LTI 连续时间系统的单位冲激响应为 $h(t)$ ，输入信号为 $x(t)$ ，根据系统的时域分析可知系统的零状态响应为

$$y(t) = x(t) * h(t) \quad (8-28)$$

对式 (8-28) 等式两边分别求傅里叶变换，根据时域卷积定理得以得到

$$Y(\omega) = X(\omega)H(\omega) \quad (8-29)$$

因此，系统的频率响应还可以由系统的零状态响应和输入的傅里叶变换之比得到

$$H(\omega) = Y(\omega) / X(\omega) \quad (8-30)$$

$H(j\omega)$ 反映了 LTI 连续时间系统对不同频率信号的响应特性,是系统内在固有的特性,与外部激励无关。 $H(j\omega)$ 又可以表示为

$$H(\omega) = |H(j\omega)| e^{j\theta(\omega)} \quad (8-31)$$

其中 $|H(\omega)|$ 称为系统的幅度响应, $\theta(\omega)$ 称为系统的相位响应。

当虚指数信号 $e^{j\omega t}$ 作用于 LTI 系统时,系统的零状态响应 $y(t)$ 仍为同频率的虚指数信号,即

$$y(t) = e^{j\omega t} H(\omega) \quad (8-32)$$

由此还可以推导出正弦信号作用在系统上的响应如表 8-2 所示:

表 8-2 正弦信号作用与 LTI 系统的响应

输入信号	响应
$\sin(\omega_0 t + \varphi), -\infty \leq t \leq +\infty$	$ H(\omega_0) \sin(\omega_0 t + \theta(\omega_0) + \varphi)$
$\cos(\omega_0 t + \varphi), -\infty \leq t \leq +\infty$	$ H(\omega_0) \cos(\omega_0 t + \theta(\omega_0) + \varphi)$

对于由下述微分方程描述的 LTI 连续时间系统

$$\sum_{n=0}^N a_n y^{(n)}(t) = \sum_{m=0}^M b_m x^{(m)}(t) \quad (8-33)$$

其频率响应 $H(j\omega)$ 可以表示为式(8-34)所示的 $j\omega$ 的有理多项式

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} = \frac{b_M(j\omega)^M + b_{M-1}(j\omega)^{M-1} + \cdots + b_1 j\omega + b_0}{a_N(j\omega)^N + a_{N-1}(j\omega)^{N-1} + \cdots + a_1 j\omega + a_0} \quad (8-34)$$

MATLAB 的信号处理工具箱提供了专门的函数 `freqs`, 用来分析连续时间系统的频率响应, 该函数有下列几种调用格式:

`[h,w] = freqs(b,a)` 计算默认频率范围内 200 个频率点上的频率响应的取样值, 这 200 个频率点记录在 `w` 中。

`h = freqs(b,a,w)` `b`、`a` 分别为表示 $H(j\omega)$ 的有理多项式中分子和分母多项式的系数向量, `w` 为频率取样点, 返回值 `h` 就是频率响应在频率取样点上的数值向量。

`[h,w] = freqs(b,a,n)` 计算默认频率范围内 `n` 个频率点上的频率响应的取样值, 这 `n` 个频率点记录在 `w` 中。

`freqs(b,a,...)` 这种调用格式不返回频率响应的取样值, 而是以对数坐标的方式绘出系统的幅频响应和相频响应。

例 8-19 三阶归一化 Butterworth 低通滤波器的频率响应为

$$H(\omega) = \frac{1}{(j\omega)^3 + 2(j\omega)^2 + 2(j\omega) + 1}$$

利用 MATLAB 绘制该系统的频率响应特性曲线。

解: 实现上述系统频域分析 MATLAB 程序如下:

```
%Eg_8_19.m
```

```
b = [1];
```

```
a = [1 2 2 1];
```

```
[H,w] = freqs(b,a);
```

```
subplot(211);
```

```

plot(w,abs(H));
set(gca,'xtick',[0:10]);           %指定 x 轴的显示刻度
set(gca,'ytick',[0 0.4 0.707 1]); %指定 y 轴的显示刻度
xlabel('\omega(rad/s)');
ylabel('Magnitude');
title('|H(j\omega)|');
grid on;
subplot(212);
plot(w,angle(H));
set(gca,'xtick',[0:10]);
xlabel('\omega(rad/s)');
ylabel('Phase');
title('\phi(\omega)');
grid on;

```

程序运行结果如图 8-33 所示。

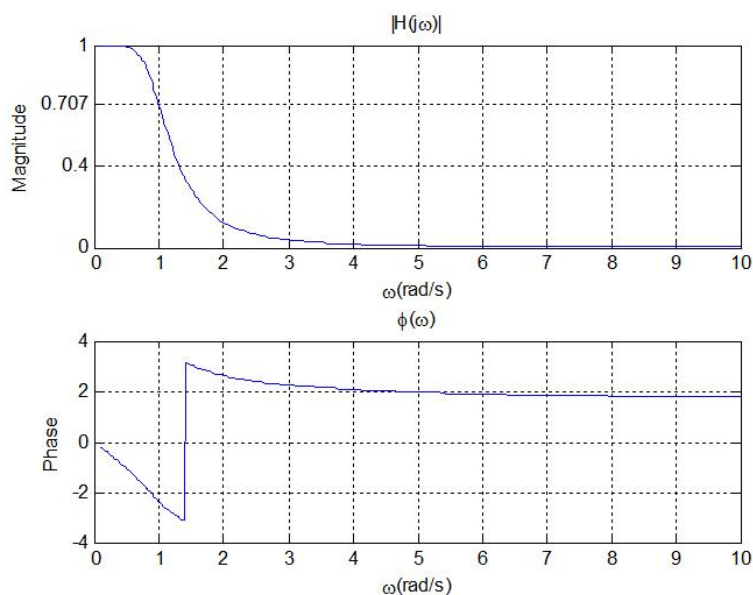


图 8-33 Butterworth 低通滤波器的频率响应

2. 离散时间系统的频率响应

LTI 离散时间系统的频率响应定义为单位抽样响应 $h(n)$ 的离散时间傅里叶变换

$$H(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} h(n)e^{-j\Omega n} \quad (8-35)$$

对于任意的输入信号 $x(n)$ ，输入与输出信号的离散时间傅里叶变换有如下关系

$$Y(e^{j\Omega}) = H(e^{j\Omega})X(e^{j\Omega}) \quad (8-36)$$

因此，系统的频率响应还可以表示为

$$H(e^{j\Omega}) = Y(e^{j\Omega}) / X(e^{j\Omega}) \quad (8-37)$$

当系统输入信号为 $x(n) = e^{j\Omega n}$ 时，系统的输出

$$y(n) = e^{j\Omega n} * h(n) = \sum_{k=-\infty}^{\infty} e^{j\Omega(n-k)} h(k) = e^{j\Omega n} H(e^{j\Omega}) \quad (8-38)$$

由式(8-38)可知,虚指数信号通过 LTI 离散时间系统后信号的频率不变,信号的幅度有系统频率响应的幅度值确定,所以 $H(e^{j\Omega})$ 表示了系统对不同频率信号的衰减量。

一般情况下离散系统的频率响应 $H(e^{j\Omega})$ 是复值函数,可用幅度和相位表示

$$H(e^{j\Omega}) = |H(e^{j\Omega})| e^{j\theta(\Omega)} \quad (8-39)$$

其中 $|H(e^{j\Omega})|$ 称为系统的幅度响应, $\theta(\Omega)$ 称为系统的相位响应。

若 LTI 离散系统可以由如下差分方程描述

$$\sum_{i=0}^N a_i y(n-i) = \sum_{j=0}^M b_j x(n-j) \quad (8-40)$$

则由式(6-4)描述的离散时间系统的频率响应 $H(e^{j\Omega})$ 可以表示为 $e^{j\Omega}$ 的有理多项式

$$H(e^{j\Omega}) = \frac{Y(e^{j\Omega})}{X(e^{j\Omega})} = \frac{b_0 + b_1 e^{-j\Omega} + \dots + b_M e^{-jM\Omega}}{a_0 + a_1 e^{-j\Omega} + \dots + a_N e^{-jN\Omega}} \quad (8-41)$$

MATLAB 的信号处理工具箱提供了专门的函数 `freqz`, 用来分析连续时间系统的频率响应, 该函数有下列几种调用格式:

`[H,w] = freqz(b,a,n)` `b`、`a` 分别为有理多项式中分子和分母多项式的系数向量, 返回值 `H` 是频率响应在 0 到 π 范围内 `n` 个频率等分点上的数值向量, `w` 包含了这 `n` 个频率点。

`[H,w] = freqz(b,a,n,'whole')` 计算 0 到 2π 之间 `n` 个频率点上的频率响应的取样值, 这 `n` 个频率点记录在 `w` 中。

`H = freqz(b,a,w)` `w` 为频率取样点, 计算这些频率点上的频率响应的取样值。

`freqz(b,a,...)` 这种调用格式不返回频率响应的取样值, 而是直接绘出系统的幅频响应和相频响应。

例 8-20 离散时间系统的频率响应为

$$H(e^{j\Omega}) = \frac{0.0441e^{-j\Omega} + 0.0317e^{-j2\Omega}}{1 - 2.0204e^{-j\Omega} + 1.4641e^{-j2\Omega} - 0.3679e^{-j3\Omega}}$$

试利用 MATLAB 分析该系统的频率特性。

解: 实现上述系统频域分析 MATLAB 程序如下:

```
%Eg_8_20.m
b = [0 0.0441 0.0317];
a = [1 -2.0204 1.4641 -0.3679];
[H,w] = freqz(b,a);    %计算频率响应
%绘制波形
subplot(211);
plot(w/pi,abs(H));
xlabel('\omega(\pi)');
ylabel('Magnitude');
title('|H(e^{j\Omega})|');
grid on;
subplot(212);
```



```

plot(w/pi,angle(H)/pi);
xlabel('\omega(\pi)');
ylabel('Phase(\pi)');
title('\theta(\Omega)');
grid on;

```

程序运行结果如图 8-34 所示。

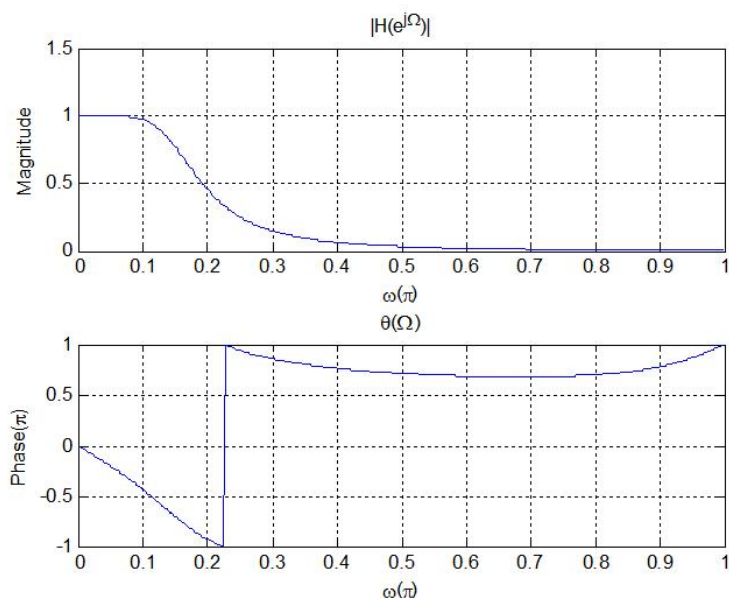


图 8-34 例 8-20 系统的频率响应

由系统频率特性曲线可知，该系统是一个低通滤波器，其中幅频特性采用了归一化的绝对幅度值。

三、实验内容

1. 已知一个 RLC 电路构造的二阶高通滤波器如图 8-35 所示，其中

$$R = \sqrt{\frac{L}{2C}}, \quad L = 0.4H, \quad C = 0.05F$$

- (1) 计算该电路系统的频率响应及高通截止频率；
 - (2) 利用 MATLAB 绘制幅度响应和相位响应曲线，比较系统的频率特性与理论计算的结果是否一致。

2. 已知一个 RC 电路如图 8-36 所示

(1) 对不同的 RC 值，用 MATLAB 画出系统的幅度响应曲线 $|H(\omega)|$ ，观察实验结果，分析图 8-36 所示 RC 电路具有什么样的频率特性（高通、低通、带通或带阻）？系统的频率特性随着 RC 值的改变，有何变化规律？

(2) 系统输入信号 $x(t) = \cos(100t) + \cos(2000t)$ ， $t = 0 \sim 0.2s$ ，该信号包含了一个低频分量和一个高频分量。试确定适当的 RC 值，滤除信号中的高频分量。并绘出滤波前后的时域信号波形及系统的频率响应曲线。

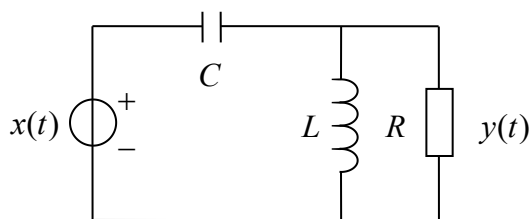


图 8-35

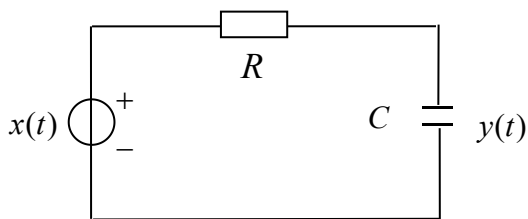


图 8-36

3. 已知离散系统的系统框图如图 8-37 所示

- (1) 写出 $M = 8$ 时系统的差分方程和系统函数；
- (2) 利用 MATLAB 计算系统的单位抽样响应；
- (3) 试利用 MATLAB 绘出其系统零极点分布图、幅频和相频特性曲线，并分析该系统具有怎样的频率特性。

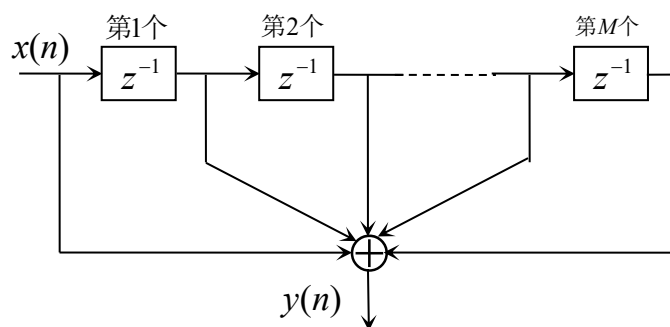


图 8-37

4. 已知一离散时间 LTI 系统的频率响应 $H(e^{j\Omega})$ 如图 8-38 所示，输入信号为 $x(n] = \cos(0.5\pi n) + 0.5\cos(\pi n)$ 。试根据式 (8-38) 分析正弦信号 $\sin(\Omega_0 t)$ 通过频率响应为 $H(e^{j\Omega})$ 的离散时间系统的响应，并根据分析结果计算系统对于 $x(n]$ 的响应 $y(n]$ ，用 MATLAB 绘出系统输入与输出波形。

观察实验结果，分析图 8-38 所示系统具有什么样的频率特性（高通、低通、带通或带阻）？从输入输出信号上怎么反映出系统的频率特性？

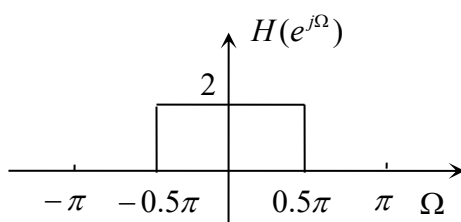


图 8-38

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

实验 5 连续时间系统的复频域分析

(综合型实验)

一、实验目的

1. 掌握拉普拉斯变换及其反变换的定义，并掌握 MATLAB 实现方法。
2. 学习和掌握连续时间系统系统函数的定义及复频域分析方法。
3. 掌握系统零极点的定义，加深理解系统零极点分布与系统特性的关系。

二、实验原理与方法

1. 拉普拉斯变换

连续时间信号 $x(t)$ 的拉普拉斯变换定义为

$$X(s) = \int_{-\infty}^{\infty} x(t)e^{-st} dt \quad (8-42)$$

拉普拉斯反变换定义为

$$x(t) = \frac{1}{2\pi j} \int_{\sigma-j\infty}^{\sigma+j\infty} X(s)e^{st} ds \quad (8-43)$$

在 MATLAB 中，可以采用了符号数学工具箱的 `laplace` 函数和 `ilaplace` 函数进行拉氏变换和拉氏反变换。

`L = laplace(F)` 符号表达式 F 的拉氏变换， F 中时间变量为 t ，返回变量为 s 的结果表达式。

`L = laplace(F,t)` 用 t 替换结果中的变量 s 。

`F = ilaplace(L)` 以 s 为变量的符号表达式 L 的拉氏反变换，返回时间变量为 t 的结果表达式。

`F = ilaplace(L,x)` 用 x 替换结果中的变量 t 。

例 8-21 求 $x(t) = e^{-t} \sin(at)u(t)$ 的拉氏变换。

解：可以利用 `laplace` 函数求得，MATLAB 代码和执行结果如下

```
>> f = sym('exp(-t)*sin(a*t)*heaviside(t)');
```

```
>> F = laplace(f)
```

```
F =
```

```
a/((s + 1)^2 + a^2)
```

即 $x(t)$ 的拉氏变换为

$$X(s) = \frac{a}{(s+1)^2 + a^2}$$

对于拉氏反变换，除了上述符号计算的方法之外，还可以采用部分分式法，当 $X(s)$ 为有理分式时，它可以表示为两个多项式之比

$$X(s) = \frac{N(s)}{D(s)} = \frac{b_M s^M + b_{M-1} s^{M-1} + \cdots + b_0}{a_N s^N + a_{N-1} s^{N-1} + \cdots + a_0} \quad (8-44)$$

式 (8-44) 可以用部分分式法展成以下形式

$$X(s) = \frac{r_1}{s-p_1} + \frac{r_2}{s-p_2} + \cdots + \frac{r_N}{s-p_N} \quad (8-45)$$

再通过查常用拉氏变换对, 很容易求得反变换。

利用 MATLAB 的 `residue` 函数可以将 $X(s)$ 展成式 (8-45) 所示的部分分式展开式, 该函数的调用格式为

`[r,p,k] = residue(b,a)` 其中 `b`、`a` 为分子和分母多项式系数向量, `r`、`p`、`k` 分别为上述展开式中的部分分式系数、极点和直项多项式系数。

例 8-22 用部分分式展开法求单边拉氏变换 $X(s)$ 的反变换

$$X(s) = \frac{s-2}{s(s+1)^3}$$

解: 首先, 将分母的因子相乘形式转换成为多项式形式, 这可以用 `conv` 函数实现, MATLAB 代码及运行结果如下

```
>> a = conv([1 0],[1 1]);
>> a = conv(a,[1 1]);
>> a = conv(a,[1 1])
a =
    1     3     3     1     0
```

所以 $X(s)$ 可以转换成如下形式:

$$X(s) = \frac{s-2}{s^4 + 3s^3 + 3s^2 + s}$$

用 `residue` 函数求 $X(s)$ 的部分分式展开式, MATLAB 代码及运行结果如下

```
>> b = [1 -2];
>> a = [1 3 3 1 0];
>> [r,p,k] = residue(b,a)
r =
    2.0000
    2.0000
    3.0000
   -2.0000
p =
   -1.0000
   -1.0000
   -1.0000
    0
k =
    []
```

$X(s)$ 的极点为 0 和 -1, 其中 -1 为三阶极点, 因此可以得到的部分分式展开式如下

$$X(s) = \frac{2}{s+1} + \frac{2}{(s+1)^2} + \frac{2}{(s+1)^3} - \frac{2}{s}$$

然后根据基本拉氏变换对可以得到的拉氏反变换为

$$x(t) = (2e^{-t} + 2te^{-t} + 1.5t^2e^{-t} - 2)u(t)$$

2. 连续时间系统的系统函数

连续时间系统的系统函数是系统单位冲激响应的拉氏变换：

$$H(s) = \int_{-\infty}^{\infty} h(t)e^{-st} dt \quad (8-46)$$

此外，连续时间系统的系统函数还可以由系统输入和输出信号的拉氏变换之比得到

$$H(s) = Y(s) / X(s) \quad (8-47)$$

单位冲激响应 $h(t)$ 反映了系统的固有性质，而 $H(s)$ 从复频域反应了系统的固有性质。由式 (8-33) 描述的连续时间系统，其系统函数为 s 的有理函数

$$H(s) = \frac{b_M s^M + b_{M-1} s^{M-1} + \cdots + b_0}{a_N s^N + a_{N-1} s^{N-1} + \cdots + a_0} \quad (8-48)$$

3. 连续时间系统的零极点分析

系统的零点指使式 (8-48) 的分子多项式为零的点，极点指使分母多项式为零的点，零点使系统的值为零，极点使系统函数的值无穷大。通常将系统函数的零极点绘在 s 平面上，零点用 \circ 表示，极点用 \times 表示，这样得到的图形称为零极点分布图。

由零极点的定义可知，零点和极点分别指式 (8-48) 的分子多项式和分母多项式的根。利用 MATLAB 求多项式的根可以通过函数 `roots` 来实现，该函数的调用格式为

`r = roots(c)` c 为多项式的系数向量，返回值 r 为多项式的根向量。

分别对式 (8-48) 的分子多项式和分母多项式求根即可求得零极点。

此外，在 MATLAB 中还提供了更简便的方法来求取零极点和绘制系统函数的零极点分布图，即利用 `pzmap` 函数，该函数的调用格式为：

`pzmap(sys)` 绘出由系统模型 `sys` 描述的系统的零极点分布图

`[p,z] = pzmap(sys)` 这种调用方法返回极点和零点，而不绘出零极点分布图

其中 `sys` 为系统传函模型，由 `t` 命令 `sys=tf(b,a)` 实现， b 、 a 为传递函数的分子多项式和分母多项式的系数向量。

MATLAB 还为用户提供了两个专用函数 `tf2zp` 和 `zp2tf` 来实现系统传递函数模型和零极点增益模型的转换，其调用格式为

$$[z,p,k] = \text{tf2zp}(b,a)$$

$$[b,a] = \text{tf2zp}(z,p,k)$$

其中 b 、 a 为传递函数的分子多项式和分母多项式的系数向量，返回值 z 为零点列向量， p 为极点列向量， k 为系统函数零极点形式的增益。

例 8-23 已知系统函数 $H(s)$ ，求取并画出零极点分布图

$$H(s) = \frac{s-1}{s^2+2s+2}$$

解：求取零极点的 MATLAB 代码及运行结果如下：

```
>> b = [1 -1];
```

```
>> a = [1 2 2];
>> sys = tf(b,a);
>> [p,z] = pzmap(sys) %求取零极点
p =
    -1.0000 + 1.0000i
    -1.0000 - 1.0000i
```

```
z =
```

```
1
```

所以系统的零点为 $z = 1$ ，极点为 $p = -1 \pm i$ 。

绘制零极点分布图的代码如下：

```
>> pzmap(sys); %绘制零极点分布图
```

绘制的零极点分布图如图 8-39 所示

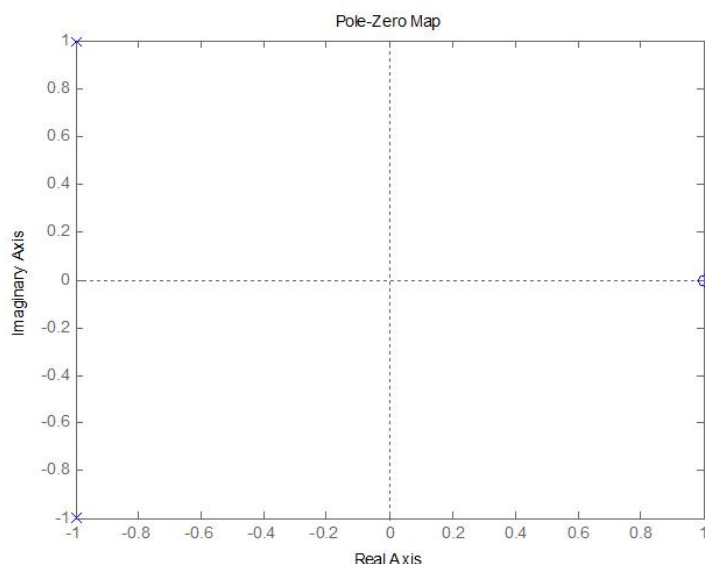


图 8-39 系统的零极点分布图

研究系统函数的零极点分布不仅可以了解系统冲激响应的形式，还可以了解系统的频率特性响应以及判断系统的稳定性。

1) 零极点分布与冲激响应的关系

系统的极点位置决定着系统冲激响应 $h(t)$ 的波形，冲激响应的幅值是由系统函数的零点和极点共同确定的，系统的零点位置只影响冲激响应的幅度和相位，不影响波形。

2) 零极点分布与系统频率响应的关系

系统函数的零极点分布不仅决定了系统函数 $H(s)$ ，也决定了系统的频率响应 $H(j\omega)$ ，根据系统的零极点分布情况，可以由几何矢量法分析系统的频率响应。

3) 零极点分布与系统稳定性的关系

稳定性是系统的固有性质，与激励信号无关，由于系统函数 $H(s)$ 包含了系统的所有固有性质，因而可以根据系统函数的零极点分布判断系统的稳定性。因果系统稳定的充要条件是 $H(s)$ 的全部极点位于 s 平面的左半平面。

三、实验内容

1. 已知系统的冲激响应 $h(t) = u(t) - 0.5u(t-2)$ ，输入信号 $x(t) = u(t)$ ，试采用复频域的方法求解系统的响应，编写 MATLAB 程序实现。

2. 已知因果连续时间系统的系统函数分别如下

$$(1) H(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}$$

$$(2) H(s) = \frac{s^2 + 1}{s^5 + 2s^4 - 3s^3 + 3s^2 + 3s + 4}$$

试采用 MATLAB 画出其零极点分布图，求解系统的冲激响应 $h(t)$ 和频率响应 $H(\omega)$ ，并判断系统是否稳定。

3. 已知连续时间系统系统函数的极点位置分别如下所示（设系统无零点）：

$$(1) p = 0$$

$$(2) p = -2$$

$$(3) p = 2$$

$$(4) p_1 = 2j, p_2 = -2j$$

$$(5) p_1 = -1 + 4j, p_2 = -1 - 4j$$

$$(6) p_1 = 1 + 4j, p_2 = 1 - 4j$$

试用 MATLAB 绘制上述 6 种不同情况下，系统函数的零极点分布图，并绘制相应冲激响应的时域波形，观察并分析系统函数极点位置对冲激响应时域特性的影响。

4. 已知连续时间系统的系统函数分别如下：

$$(1) H(s) = \frac{1}{s^2 + 2s + 16}$$

$$(2) H(s) = \frac{s + 8}{s^2 + 2s + 16}$$

$$(3) H(s) = \frac{s - 8}{s^2 + 2s + 16}$$

上述 3 个系统具有相同的极点，只是零点不同，试用 MATLAB 分别绘制系统的零极点分布图及相应冲激响应的时域波形，观察并分析系统函数零点位置对冲激响应时域特性的影响。

四、实验报告要求

1. 简述实验目的和实验原理。

2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。

3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

实验 6 离散时间系统的 z 域分析

(综合型实验)

一、实验目的

1. 掌握 z 变换及其反变换的定义，并掌握 MATLAB 实现方法。
2. 学习和掌握离散时间系统系统函数的定义及 z 域分析方法。
3. 掌握系统零极点的定义，加深理解系统零极点分布与系统特性的关系。

二、实验原理与方法

1. z 变换

序列 $x(n)$ 的 z 变换定义为

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \quad (8-49)$$

z 反变换定义为

$$x(n) = \frac{1}{2\pi j} \oint_r X(z)z^{n-1} dz \quad (8-50)$$

在 MATLAB 中，可以采用了符号数学工具箱的 `ztrans` 函数和 `iztrans` 函数计算 z 变换和 z 反变换：

`Z = ztrans(F)` 求符号表达式 F 的 Z 变换。

`F = ilaplace(Z)` 求符号表达式 F 的 Z 反变换。

例 8-24 求 $x(n) = (n-3)u(n)$ 的 Z 变换。

解：可以利用 `ztrans` 函数求得，MATLAB 代码和执行结果如下

```
>> x = sym('(n-3)');
```

```
>> Z = ztrans(x)
```

```
Z =
```

```
z/(z - 1)^2 - (3*z)/(z - 1)
```

```
>> Z = simplify(Z)
```

```
Z =
```

```
-(2*z - 3)/(z - 1)^2 - 3
```

即的 Z 变换为

$$X(z) = -\frac{2z-3}{(z-1)^2} - 3$$

与求解拉氏反变换的原理相同，我们可以用部分分式法求 z 反变换，即先采用 `residue` 函数部分分式展开，然后根据常用变换对逐项求其反变换。

例 8-25 用部分分式展开法求 $X(z)$ 的 z 反变换

$$X(z) = \frac{z^2}{(z-0.5)(z-0.25)}, \quad |z| > 0.5$$

解：由已知条件可得

$$\frac{X(z)}{z} = \frac{z^2}{z(z-0.5)(z-0.25)}$$

首先，利用 poly 函数将分母转换为多项式形式，MATLAB 代码和执行结果如下

```
>> a = poly([0 0.5 0.25])
```

```
a =
```

```
1.0000 -0.7500 0.1250 0
```

即

$$\frac{X(z)}{z} = \frac{z^2}{z^3 - 0.75z^2 + 0.125z}$$

然后，使用 residue 对对上式进行部分分式展开，MATLAB 代码和执行结果如下

```
>> b = [1 0 0];
```

```
>> [r,p,k] = residue(b,a)
```

```
r =
```

```
2
```

```
-1
```

```
0
```

```
p =
```

```
0.5000
```

```
0.2500
```

```
0
```

```
k =
```

```
[]
```

因此可得

$$\frac{X(z)}{z} = \frac{2}{z-0.5} + \frac{-1}{z-0.25}$$

则

$$X(z) = \frac{2z}{z-0.5} + \frac{-z}{z-0.25}$$

所以由基本变换对可得 z 反变换的结果为

$$x(n) = \left[2\left(\frac{1}{2}\right)^n - \left(\frac{1}{4}\right)^n \right] u(n)$$

2. 离散时间系统的系统函数

离散时间系统的系统函数 $H(z)$ 定义为单位抽样响应 $h(n)$ 的 Z 变换

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} \quad (8-51)$$

此外，连续时间系统的系统函数还可以由系统输入和输出信号的 z 变换之比得到

$$H(z) = Y(z) / X(z) \quad (8-52)$$

由式 (8-40) 描述的离散时间系统的系统函数可以表示为

$$H(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{a_0 + a_1 z^{-1} + \cdots + a_N z^{-N}} \quad (8-53)$$

3. 离散时间系统的零极点分析

离散时间系统的零点和极点分别指使系统函数分子多项式和分母多项式为零的点。在 MATLAB 中可以通过函数 `roots` 来求系统函数分子多项式和分母多项式的根，从而得到系统的零极点。

此外，还可以利用 MATLAB 的 `zplane` 函数来求解和绘制离散系统的零极点分布图，`zplane` 函数的调用格式为：

`zplane(b,a)` b 、 a 为系统函数的分子、分母多项式的系数向量（行向量）。

`zplane(z,p)` z 、 p 为零极点序列（列向量）

例 8-26 已知系统函数 $H(z)$ ，求取并画出零极点分布图

$$H(z) = \frac{z^2 - z}{z^2 - 0.75z + 0.125}$$

解： 求取零极点的 MATLAB 代码如下：

```
>> b = [1 -1 0];
```

```
>> a = [1 -0.75 0.125];
```

```
>> zplane(b,a);
```

绘制的零极点分布图如图 8-40 所示。

例 8-27 已知系统函数 $H(z)$ ，求取并画出零极点分布图

$$H(z) = \frac{z(z-1.2)}{(z-0.8)(z+0.8)}$$

解： 求取零极点的 MATLAB 代码如下：

```
>> z = [0 1.2]';
```

```
>> p = [0.8 -0.8]';
```

```
>> zplane(z,p);
```

绘制的零极点分布图如图 8-41 所示。

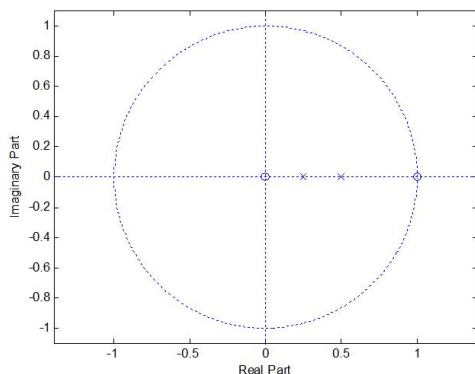


图 8-40 例 8-26 系统的零极点分布图

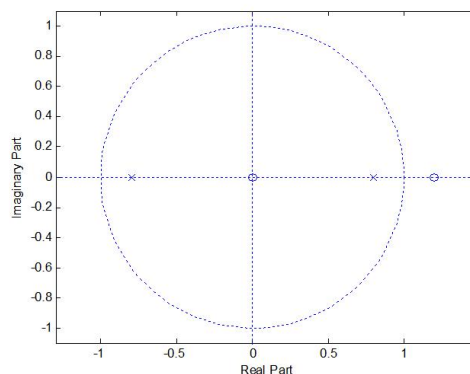


图 8-41 例 8-27 系统的零极点分布图

系统函数是描述系统的重要物理量，研究系统函数的零极点分布不仅可以了解系统单位抽样响应的变化，还可以了解系统的频率特性响应以及判断系统的稳定性：

1) 系统函数的极点位置决定了系统单位抽样响应 $h(n)$ 的波形, 系统函数零点位置只影响冲激响应的幅度和相位, 不影响波形。

2) 系统的频率响应取决于系统函数的零极点, 根据系统的零极点分布情况, 可以通过向量法分析系统的频率响应。

3) 因果的离散时间系统稳定的充要条件是 $H(z)$ 的全部极点都位于单位圆内。

三、实验内容

1. 已知因果离散时间系统的系统函数分别为

$$(1) H(z) = \frac{z^2 + 2z + 1}{z^3 - z^2 - 0.5z + 0.3}$$

$$(2) H(z) = \frac{z^3 - z^2 + 2}{3z^4 + 2z^3 - z^2 + 3z - 1}$$

试采用 MATLAB 画出其零极点分布图, 求解系统的冲激响应 $h(n)$ 和频率响应 $H(e^{j\Omega})$, 并判断系统是否稳定。

2. 已知离散时间系统系统函数的零点 z 和极点 p 分别为:

$$(1) z = 0, \quad p = 0.25$$

$$(2) z = 0, \quad p = 1$$

$$(3) z = 0, \quad p = -1.25$$

$$(4) z = 0, \quad p_1 = 0.8e^{j\frac{\pi}{6}}, \quad p_2 = 0.8e^{-j\frac{\pi}{6}}$$

$$(5) z = 0, \quad p_1 = e^{j\frac{\pi}{8}}, \quad p_2 = e^{-j\frac{\pi}{8}}$$

$$(6) z = 0, \quad p_1 = 1.2e^{j\frac{3\pi}{4}}, \quad p_2 = 1.2e^{-j\frac{3\pi}{4}}$$

试用 MATLAB 绘制上述 6 种不同情况下, 系统函数的零极点分布图, 并绘制相应单位抽样响应的时域波形, 观察分析系统函数极点位置对单位抽样响应时域特性的影响和规律。

3. 已知离散时间系统的系统函数分别为:

$$(1) H(z) = \frac{z(z+2)}{(z-0.8e^{j\frac{\pi}{6}})(z-0.8e^{-j\frac{\pi}{6}})}$$

$$(2) H(z) = \frac{z(z-2)}{(z-0.8e^{j\frac{\pi}{6}})(z-0.8e^{-j\frac{\pi}{6}})}$$

上述两个系统具有相同的极点, 只是零点不同, 试用 MATLAB 分别绘制上述两个系统的零极点分布图及相应单位抽样响应的时域波形, 观察分析系统函数零点位置对单位抽样响应时域特性的影响。

四、实验报告要求

1. 简述实验目的和实验原理。

2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。