

Matlab 程序设计与应用

(第 3 版)

源程序代码



源程序目录

第 1 章	MATLAB 系统环境.....	3
第 2 章	MATLAB 数据及其运算.....	3
第 3 章	MATLAB 矩阵处理.....	4
第 4 章	MATLAB 程序流程控制.....	5
第 5 章	MATLAB 绘图.....	12
第 6 章	MATLAB 数据分析与多项式计算.....	21
第 7 章	MATLAB 数值微分与积分.....	24
第 8 章	MATLAB 方程数值求解.....	27
第 9 章	MATLAB 符号计算.....	32
第 10 章	MATLAB 图形对象句柄.....	38
第 11 章	MATLAB 图形用户界面设计.....	43
第 12 章	MATLAB Simulink 系统仿真.....	50
第 13 章	MATLAB 外部程序接口技术.....	52
第 14 章	MATLAB 的应用.....	56
14.1	电路分析.....	56
14.2	控制系统分析.....	56
14.3	分形曲线的绘制.....	57
14.4	最优化问题求解.....	57
	证券投资组合问题.....	57
	生产决策问题.....	58
14.5	工程结构分析.....	58

静不定问题.....	58
直梁的自由振动频率.....	58

第 1 章 MATLAB 系统环境

例 1-1 分别绘制函数 $y = 2^{-|x|}$ 和 $y = \sin x$ 的曲线。

```
>> x=-2*pi:pi/180:2*pi;
>> y1=2.^(-abs(x));
>> y2=sin(x);
>> plot(x,y1,':',x,y2)
```

例 1-2 求方程 $2x^5-3x^3+71x^2-9x+13=0$ 的全部根。

```
>> p=[2,0,-3,71,-9,13];
>> x=roots(p)
```

例 1-3 求解线性方程组。

```
>> a=[2,3,-1;8,2,3;45,3,9];
>> b=[2;4;23];
>> x=inv(a)*b
```

通过符号计算来解此方程。

```
>> syms x y z
>> [x,y,z]=solve(2*x+3*y-z-2,8*x+2*y+3*z-4,45*x+3*y+9*z-23)
>> eval([x,y,z])
```

例 1-4 求定积分 $\int_0^1 x \ln(1+x) dx$ 。

```
>> f=@(x) x.*log(1+x);    %定义被积函数，log 是 MATLAB 提供的对数函数
>> integral(f,0,1)         %求定积分
ans =
    0.2500
```

通过符号计算来求符号定积分。

```
>> syms x
>> int(x*log(1+x),0,1)
```

第 2 章 MATLAB 数据及其运算

例 2-1 计算表达式的值。

```
>> x=sqrt(7)-2i;
>> y=exp(pi/2);
>> z=(5+cos(47*pi/180))/(1+abs(x-y))
```

例 2-2 产生 5 阶随机方阵 A，其元素为[10, 90]区间的随机整数，然后判断 A 的元素是否能被 3 整除。

```
>> A=fix((90-10+1)*rand(5)+10)
```

```
>> P=rem(A,3)==0           %等价于 P=eq(rem(A,3),0)
```

例 2-3 建立矩阵 A，然后找出大于 4 的元素的位置，并输出相应位置的元素。

```
>> A=[4,-65,-54,0,6;56,0,67,-45,0]
>> k=find(A>4)
>> A(k)
```

例 2-4 建立一个字符串向量，然后对该向量做如下处理。

```
>> ch='ABc123d4e56Fg9';
>> subch=ch(1:5)           %取子字符串
subch =
ABc12
>> revch=ch(end:-1:1)      %将字符串倒排
revch =
9gF65e4d321cBA
>> k=find(ch>='a'&ch<='z'); %找小写字母的位置
>> ch(k)=ch(k)-('a'-'A')   %将小写字母变成相应的大写字母
ch =
ABC123D4E56FG9
>> length(k)               %统计小写字母的个数
ans =
4
```

第 3 章 MATLAB 矩阵处理

例 3-1 建立随机矩阵：

```
>> x=20+(50-20)*rand(5)
>> y=0.6+sqrt(0.1)*randn(5)
```

例 3-2 将[101, 125]范围内的 25 个整数填入一个 5 行 5 列的表格中，使其每行每列及对角线的和均为 565。

```
>> M=100+magic(5)
```

例 3-3 求 4 阶希尔伯特矩阵及其逆矩阵。

```
>> format rat           %以有理形式输出
>> H=hilb(4)
>> H=invhilb(4)
>> format              %恢复默认输出格式
```

例 3-4 求 $(x+y)^5$ 的展开式。

```
>> pascal(6)
```

例 3-5 先建立 5×5 矩阵 A，然后将 A 的第一行元素乘以 1，第二行乘以 2，…，第五行乘以 5。

```
>> A=[17,0,1,0,15;23,5,7,14,16;4,0,13,0,22;10,12,19,21,3;...
11,18,25,2,19];
>> D=diag(1:5);
>> D*A           %用 D 左乘 A，对 A 的每行乘以一个指定常数
```

例 3-6 求方阵 A 的逆矩阵，且验证 A 与 A^{-1} 是互逆的。

```
>> A=[1,-1,1;5,-4,3;2,1,1];
>> B=inv(A);
>> A*B
>> B*A
```

例 3-7 用求逆矩阵的方法解线性方程组。

```
>> A=[1,2,3;1,4,9;1,8,27];
>> b=[5,-2,6]';
>> x=inv(A)*b
```

例 3-8 用求特征值的方法解方程。

```
>> p=[3,-7,0,5,2,-18];
>> A=compan(p);           %A 的伴随矩阵
>> x1=eig(A)              %求 A 的特征值
>> x2=roots(p)            %直接求多项式 p 的零点
```

例 3-9 将 X 转化为稀疏存储方式。

```
>> X=[2,0,0,0,0;0,0,0,0,0;0,0,0,5,0;0,1,0,0,-1;0,0,0,0,-5];
>> A=sparse(X)
```

例 3-10 根据表示稀疏矩阵的矩阵 A，产生一个稀疏存储矩阵 B。

```
>> A=[2,2,1;3,1,-1;4,3,3;5,3,8;6,6,12];
>> B=spconvert(A)
```

例 3-11 求下列三对角线性方程组的解。

```
>> B=[1,2,0;1,4,3;2,6,1;1,6,4;0,1,2]; %产生非 0 对角元素矩阵
>> d=[-1;0;1];
```

生非 0 对角元素位置向量

```
>> A=spdiags(B,d,5,5)
      %产生稀疏存储的系数矩阵
```

% 产

第 4 章 MATLAB 程序流程控制

例 4-1 建立一个脚本文件将变量 a、b 的值互换，然后运行该脚本文件。

程序 1:

首先建立脚本文件并以文件名 `exch.m` 存盘。

```
clear;
a=1:10;
b=[11,12,13,14;15,16,17,18];
c=a;a=b;b=c;
a
b
```

然后在 MATLAB 的命令行窗口中输入 `exch`，将会执行该脚本文件。

```
>> exch
```

程序 2:

首先建立函数文件 `fexch.m`。

```
function [a,b]=exch(a,b)
c=a;a=b;b=c;
```

然后在 MATLAB 的命令行窗口调用该函数文件:

```
>> clear;
>> x=1:10;
>> y=[11,12,13,14;15,16,17,18];
>> [x,y]=fexch(x,y)
```

例 4-2 求一元二次方程 $ax^2 + bx + c = 0$ 的根。

```
a=input('a=?');
b=input('b=?');
c=input('c=?');
d=b*b-4*a*c;
x=[(-b+sqrt(d))/(2*a),(-b-sqrt(d))/(2*a)];
disp(['x1=',num2str(x(1)),' ',x2=',num2str(x(2))'])
```

例 4-3 计算分段函数的值。

$$y = \begin{cases} \cos(x+1) + \sqrt{x^2+1}, & x=10 \\ x\sqrt{x+\sqrt{x}}, & x \neq 10 \end{cases}$$

```
x=input('请输入 x 的值:');
if x==10
    y=cos(x+1)+sqrt(x*x+1);
else
    y=x*sqrt(x+sqrt(x));
end
y
```

也可以用单分支 if 语句来实现，程序如下:

```
x=input('请输入 x 的值:');
if x==10
    y=cos(x+1)+sqrt(x*x+1);
end
if x~=10
    y=x*sqrt(x+sqrt(x));
end
y
```

第一个 if 语句可以不用，而直接求函数值即可，改用以下程序实现。

```
x=input('请输入 x 的值:');
y=cos(x+1)+sqrt(x*x+1);
if x~=10
    y=x*sqrt(x+sqrt(x));
end
y
```

例 4-4 输入一个字符，若为大写字母，则输出其对应的小写字母；若为小写字母，则输出其对应的大

写字母；若为数字字符则输出其对应数的平方，若为其他字符则原样输出。

```
c=input('请输入一个字符: ','s');
if c>='A' && c<='Z'
    disp(lower(c))
elseif c>='a' && c<='z'
    disp(upper(c))
elseif c>='0' && c<='9'
    disp(str2double(c)^2)
else
    disp(c)
end
```

例 4-5 某商场对商品实行打折销售，求其实际销售价格。

```
price=input('请输入商品价格');
switch fix(price/100)
    case {0,1} %价格小于 200
        rate=0;
    case {2,3,4} %价格大于等于 200 但小于 500
        rate=3/100;
    case num2cell(5:9) %价格大于等于 500 但小于 1000
        rate=5/100;
    case num2cell(10:24) %价格大于等于 1000 但小于 2500
        rate=8/100;
    case num2cell(25:49) %价格大于等于 2500 但小于 5000
        rate=10/100;
    otherwise %价格大于等于 5000
        rate=14/100;
end
price=price*(1-rate) %输出商品实际销售价格
```

例 4-6 矩阵乘法运算要求两矩阵的维数相容，否则会出错。先求两矩阵的乘积，若出错，则提示错误信息。

```
A=input('请输入 A 矩阵: ');
B=input('请输入 B 矩阵: ');
try
    C=A*B
catch
    lasterr
end
```

例 4-7 输出全部水仙花数。

```
shu=[]; %用于存放结果，先赋空值
for m=100:999
    m1=fix(m/100); %求 m 的百位数字
    m2=rem(fix(m/10),10); %求 m 的十位数字
    m3=rem(m,10); %求 m 的个位数字
    if m==m1*m1*m1+m2*m2*m2+m3*m3*m3
        shu=[shu,m]; %存入结果
    end
end
disp(shu)
```

例 4-8 已知 y ，当 $n=100$ 时，求 y 的值。

```
y=0;
n=100;
for i=1:n
    y=y+1/(i*i);
end
y
```

例 4-9 设 $f(x) = e^{-0.5x} \sin(x + \frac{\pi}{6})$ ，求 $s = \int_0^{3\pi} f(x)dx$ 。

程序一：

```
a=0;
b=3*pi;
n=1000;
h=(b-a)/n;
x=a;
s=0;
f0=exp(-0.5*x)*sin(x+pi/6);
for i=1:n
    x=x+h;           %下一个 x 坐标
    f1=exp(-0.5*x)*sin(x+pi/6); %求新的函数值
    s=s+(f0+f1)*h/2; %求小梯形面积并累加
    f0=f1;           %更新函数值
end
s
```

程序二：

```
a=0;
b=3*pi;
n=1000;
h=(b-a)/n;
x=a:h:b;           %生成自变量向量 x
f=exp(-0.5*x).*sin(x+pi/6); %求函数值向量 f
for i=1:n
    s(i)=(f(i)+f(i+1))*h/2; %求各个小梯形面积，组成面积向量 s
end
s=sum(s)            %求面积向量 s 元素之和
```

例 4-10 写出下列程序的执行结果。

```
s=0;
a=[12,13,14;15,16,17;18,19,20;21,22,23];
for k=a
    s=s+k;
end
disp(s')
```

例 4-11 从键盘输入若干个数，当输入 0 时结束输入，求这些数的平均值和它们之和。

```
sum=0;
n=0;
x=input('Enter a number (end in 0):');
while x~=0
    sum=sum+x;
```



```

        n=n+1;
        x=input('Enter a number (end in 0):');
end
if n>0
    sum
    mean=sum/n
end

```

例 4-12 根据矩阵指数的幂级数展开式求矩阵指数。

```

X=input('Enter X:');
E=zeros(size(X));           %生成与 x 同样大小的零矩阵
F=eye(size(X));             %生成与 x 同样大小的单位矩阵
n=1;
while norm(F,1)>0
    E=E+F;                   %实现累加
    F=F*X/n;                 %求累加项
    n=n+1;
end
E
expm(X)                      %调用 MATLAB 矩阵指数函数求矩阵指数

```

例 4-13 求[100, 200]之间第一个能被 21 整除的整数。

```

for n=100:200
    if rem(n,21)~=0
        continue
    end
    n
    break
end

```

例 4-14 若一个数等于它的各个真因子之和，则称该数为完数，如 $6=1+2+3$ ，所以 6 是完数。求[1, 500]之间的全部完数。

```

for m=1:500
    s=0;
    for k=1:m/2
        if rem(m,k)==0
            s=s+k;
        end
    end
    if m==s
        disp(m)
    end
end

```

例 4-15 用筛选法求某自然数范围内的全部素数。

```

m=input('m=');
p=1:m;
p(1)=0;
for i=2:sqrt(m)
    for j=2*i:i:m
        p(j)=0;
    end
end

```

```

        end
    end
    n=find(p~=0);
    p(n)
    更为简洁的程序:
    m=input('m=');
    p=2:m;
    for i=2:sqrt(m)
        n=find(rem(p,i)==0 & p~=i);
        p(n)=[];
    end
    p

```

例 4-16 编写函数文件，求半径为 r 的圆的面积和周长。

函数文件如下：

```

function [s,p]=fcircle(r)
%CIRCLE calculate the area and perimeter of a circle of radii r
%r        圆半径
%s        圆面积
%p        圆周长

```

%2016 年 9 月 20 日编

s=pi*r*r;

p=2*pi*r;

将以上函数文件以文件名 **fcircle.m** 存盘，然后在 **MATLAB** 命令行窗口调用该函数。

```
>> [s,p]=fcircle(10)
```

```
s =
```

```
314.1593
```

```
p =
```

```
62.8319
```

例 4-17 利用函数文件，实现直角坐标 (x, y) 与极坐标 (ρ, θ) 之间的转换。

首先编写函数文件 **tran.m**。

```

function [rho,theta]=tran(x,y)
rho=sqrt(x*x+y*y);
theta=atan(y/x);

```

再在脚本文件 **main1.m** 中调用函数文件 **tran.m**。

```

x=input('Please input x=');
y=input('Please input y=');
[rho,the]=tran(x,y);

```

```
disp(['rho=',num2str(rho)])
```

```
disp(['the=',num2str(the)])
```

最后在命令行窗口运行脚本文件 **main1.m**。

```
>> main1
```

例 4-18 利用函数的递归调用，求 $n!$ 。

采用递归调用的函数文件 **factor.m** 如下：

```
function f=factor(n)
```

```
if n<=1
```

```
    f=1;
```

```
else
```

```

        f=factor(n-1)*n;    %递归调用求 (n-1) !
    end
在脚本文件 main2.m 中调用函数文件 factor.m, 求  $s = 1! + 2! + 3! + 4! + 5!$ 。
s=0;
n=input('Please input n=');
for i=1:n
    s=s+factor(i);
end
disp(['1 到', num2str(n), '的阶乘和为: ', num2str(s)])
在命令行窗口运行脚本文件:
>> main2

```

例 4-19 任意排列问题。

① 函数用循环结构。

```

function Y=rdndprm1(X)
%RDNDPRM1    用 for 循环产生一个行向量的任意排列
%RDNDPRM1(X) 产生行向量 X 的任意排列
[m,n]=size(X);
if m>1
    error('RDNDPRM1 accepts as inputs only vectors');
end
Y=[];                                %从一个空矩阵开始
l=n;                                %X 的元素个数
for i=1:n
    k=1+fix(l*rand);                %随机选择 Y 的下一个元素的位置
    x=X(k);                        %被选择的元素
    Y=[Y,x];                        %将 x 添加到 Y 中
    X(k)=[];                        %从 X 中删除 x 元素
    l=l-1;                          %更新 X 的元素个数
end

```

② 函数用递归调用。

```

function Y=rdndprm2(X)
%RDNDPRM2    用递归调用产生一个行向量的任意排列
%RDNDPRM2(X) 产生一个 X 的任意排列
[m,n]=size(X);
l=n;
if m>1
    error('RDNDPRM2 accepts as inputs only vectors')
end
if n<=1
    Y=X;
else
    k=1+fix(l*rand);                %随机选择 Y 的下一个元素的位置
    x=X(k);                        %被选择的元素
    X(k)=[];                        %从 X 中删除 x 元素
    Z=rdndprm2(X);                 %将剩下的元素随机排列
    Y=[Z,x];                        %构造输出向量
    l=l-1;
end

```

③ 在命令行窗口调用所定义的函数, 命令如下:

```

>> rdndprm1([34,6,3,54,2,5,454])

```

```
>> rndprn2([34,6,3,54,2,5,454])
>> rndprn1('apple')
>> rndprn2('apple')
```

例 4-20 nargin 用法示例。

建立函数文件 **chararray.m**。

```
function fout=chararray(a,b,c)
if nargin==1
    fout=a;
elseif nargin==2
    fout=a+b;
elseif nargin==3
    fout=(a*b*c)/2;
end
```

建立脚本文件 **mydemo.m**。

```
a=1:3;b=a';
x=chararray(a);
y=chararray(a,b');
z=chararray(a,b,3);
disp(['x=      ',num2str(x)])
disp(['y=      ',num2str(y)])
disp(['z=      ',num2str(z)])
```

例 4-21 全局变量应用示例。

先建立函数文件 **wadd.m**，该函数将输入的参数加权相加。

```
function f=wadd(x,y)
global ALPHA BETA
f=ALPHA*x+BETA*y;
```

在命令行窗口中输入命令并得到输出结果。

```
>> global ALPHA BETA
>> ALPHA=1;
>> BETA=2;
>> s=wadd(1,2)
```

第 5 章 MATLAB 绘图

例 5-1 在 $0 \leq x \leq 2\pi$ 区间内，绘制曲线 $y=2e^{-0.5x}\sin(2\pi x)$ 。

程序如下：

```
x=0:pi/100:2*pi;
y=2*exp(-0.5*x).*sin(2*pi*x);
plot(x,y)
```

例 5-2 绘制曲线。

```
t=-pi:pi/100:pi;
x=t.*cos(3*t);
y=t.*sin(t).*sin(t);
plot(x,y)
```

例 5-3 用不同线型和颜色在同一坐标内绘制曲线 $y=2e^{-0.5x}\sin(2\pi x)$ 及其包络线。

```
x=(0:pi/100:2*pi)';
y1=2*exp(-0.5*x)*[1,-1];
```

```

y2=2*exp(-0.5*x).*sin(2*pi*x);
x1=(0:12)/2;
y3=2*exp(-0.5*x1).*sin(2*pi*x1);
plot(x,y1,'k:',x,y2,'b--',x1,y3,'rp')

```

例 5-4 用不同标度在同一坐标内绘制曲线 $y=e^{-0.5x}\sin(2\pi x)$ 及曲线 $y=\sin x$ 。

```

x=0:pi/100:2*pi;
y1=exp(-0.5*x).*sin(2*pi*x);
y2=sin(x);
plotyy(x,y1,x,y2)

```

例 5-5 绘制分段函数曲线并添加图形标注。

```

x=linspace(0,10,100); %产生自变量向量 x
y=[]; %y 的初始值为空
for x0=x
    if x0>=8 %将函数值追加到向量 y
        y=[y,1];
    elseif x0>=6
        y=[y,5-x0/2];
    elseif x0>=4
        y=[y,2];
    elseif x0>=0
        y=[y,sqrt(x0)];
    end
end
plot(x,y)
axis([0,10,0,2.5]) %设置坐标轴
title('分段函数曲线') %加图形标题
xlabel('Variable X') %加 X 轴说明
ylabel('Variable Y') %加 Y 轴说明
text(2,1.3,'y=x^{1/2}') %在指定位置添加图形说明
text(4.5,1.9,'y=2')
text(7.3,1.5,'y=5-x/2')
text(8.5,0.9,'y=1')

```

例 5-6 用图形保持功能在同一坐标内绘制曲线 $y=2e^{-0.5x}\sin(2\pi x)$ 及其包络线。

```

x=(0:pi/100:2*pi)';
y1=2*exp(-0.5*x)*[1,-1];
y2=2*exp(-0.5*x).*sin(2*pi*x);
plot(x,y1,'b:') %绘制两根包络线
axis([0,2*pi,-2,2]) %设置坐标
hold on %设置图形保持状态
plot(x,y2,'k') %绘制曲线
legend('包络线','包络线','曲线 y') %加图例
hold off %关闭图形保持
grid %网格线控制

```

例 5-7 在一个图形窗口中以子图形式同时绘制正弦、余弦、正切、余切曲线。

程序如下：

```

x=linspace(0,2*pi,60);
y=sin(x);

```

```

z=cos(x);
t=sin(x)./(cos(x)+eps);
ct=cos(x)./(sin(x)+eps);
subplot(2,2,1)      %选择 2×2 个区中的 1 号区
plot(x,y)
title('sin(x)');axis([0,2*pi,-1,1])
subplot(2,2,2)      %选择 2×2 个区中的 2 号区
plot(x,z)
title('cos(x)');axis([0,2*pi,-1,1])
subplot(2,2,3)      %选择 2×2 个区中的 3 号区
plot(x,t)
title('tangent(x)');axis([0,2*pi,-40,40])
subplot(2,2,4)      %选择 2×2 个区中的 4 号区
plot(x,ct)
title('cotangent(x)');axis([0,2*pi,-40,40])

```

实际上，还可以作更灵活的分割。请看下面的程序。

```

x=linspace(0,2*pi,60);
y=sin(x);
z=cos(x);
t=sin(x)./(cos(x)+eps);
ct=cos(x)./(sin(x)+eps);
subplot(2,2,1)      %选择 2×2 个区中的 1 号区
plot(x,y-1)
title('sin(x)-1');axis ([0,2*pi,-2,0])
subplot(2,1,2)      %选择 2×1 个区中的 2 号区
plot(x,z-1)
title('cos(x)-1');axis ([0,2*pi,-2,0])
subplot(4,4,3)      %选择 4×4 个区中的 3 号区
plot(x,y)
title('sin(x)');axis ([0,2*pi,-1,1])
subplot(4,4,4)      %选择 4×4 个区中的 4 号区
plot(x,z)
title('cos(x)');axis ([0,2*pi,-1,1])
subplot(4,4,7)      %选择 4×4 个区中的 7 号区
plot(x,t)
title('tangent(x)');axis ([0,2*pi,-40,40])
subplot(4,4,8)      %选择 4×4 个区中的 8 号区
plot(x,ct)
title('cotangent(x)');axis ([0,2*pi,-40,40])

```

例 5-8 用 `fplot` 函数绘制 $f(x)=\cos(\tan(\pi x))$ 的曲线。

命令如下：

```
>> fplot(@ (x) cos(tan(pi*x)), [0,1])
```

例 5-9 绘制 $y=10x^2$ 的对数坐标图并与直角线性坐标图进行比较。

程序如下：

```

x=0:0.1:10;
y=10*x.*x;
subplot(2,2,1);plot(x,y) %直角坐标曲线
title('plot(x,y)');grid on

```

```

subplot(2,2,2);semilogx(x,y)    %x 半对数坐标曲线
title('semilogx(x,y)');grid on
subplot(2,2,3);semilogy(x,y)    %y 半对数坐标曲线
title('semilogy(x,y)');grid on
subplot(2,2,4);loglog(x,y)      %全对数坐标曲线
title('loglog(x,y)');grid on

```

例 5-10 蝴蝶曲线是一种富有美感的平面曲线。

① 绘制蝴蝶曲线。

② 调整 θ 的大小可以改变曲线形状及其方向，将 θ 减 $\pi/2$ ，使图形旋转 90° ，绘制蝴蝶曲线。

程序如下：

```

t=0:pi/50:20*pi;
r1=exp(cos(t))-2*cos(4*t)+sin(t/12).^5;
r2=exp(cos(t-pi/2))-2*cos(4*(t-pi/2))+sin((t-pi/2)/12).^5;
subplot(1,2,1)
polar(t,r1)    %绘制蝴蝶曲线
subplot(1,2,2)
polar(t,r2)    %旋转 90 度的蝴蝶曲线

```

例 5-11 条形图应用示例。

程序如下：

```

x=-1:1;
y=[1,2,3,4,5;1,2,1,2,1;5,4,3,2,1];
subplot(2,2,1);bar(x,y,'grouped')
title('Group');axis([-3,3,0,6])
subplot(2,2,2);barh(x,y,'stacked')
title('Stack')

```

例 5-12 绘制服从高斯分布的直方图，再将这些数据分到指定范围的区间中，并绘制直方图中。

程序如下：

```

y=randn(500,1);
subplot(2,2,1)
hist(y);title('高斯分布直方图')
x=-4:0.1:4;
subplot(2,2,2)
hist(y,x);title('指定范围的高斯分布直方图')

```

例 5-13 绘制例 5-12 中高斯分布数据在极坐标下的直方图。

```

y=randn(500,1);
theta=y*pi;
rose(theta)
title('在极坐标下的直方图')

```

例 5-14 某次考试优秀、良好、中等、及格、不及格的人数分别为：7、17、23、19、5，试用扇形统计图作成绩统计分析。

程序如下：

```
pie([7,17,23,19,5],[0,0,0,0,1])    %对应第 5 分量部分从饼图中心分离
title('饼图')
legend('优秀','良好','中等','及格','不及格')
```

例 5-15 绘制一个红色的正八边形。

程序如下：

```
t=0:2*pi/8:2*pi; %取正八边形坐标点
t=[t,t(1)];      %数据向量的首尾重合，使图形封闭。
x=sin(t);
y=cos(t);
fill(x,y,'r')    %x、y 是圆周坐标，数据间隔足够小时可以画圆
axis equal;axis([-1.5,1.5,-1.5,1.5])
```

例 5-16 分别以散点图、阶梯图和杆图形式绘制曲线 $y=2e^{-0.5x}$ 。

程序如下：

```
x=0:0.35:7;
y=2*exp(-0.5*x);
subplot(2,3,1);scatter(x,y,'g')
title('scatter(x,y,\'g\')');axis([0,7,0,2])
subplot(2,3,2);stairs(x,y,'b')
title('stairs(x,y,\'b\')');axis([0,7,0,2])
subplot(2,3,3);stem(x,y,'k')
title('stem(x,y,\'k\')');axis([0,7,0,2])
```

例 5-17 分别以罗盘图、羽毛图和箭头图形式绘制绘制正弦曲线。

程序如下：

```
x=-pi:pi/8:pi;
y=sin(x);
subplot(2,2,1);compass(x,y)
title('罗盘图')
subplot(2,2,2);feather(x,y)
title('羽毛图')
subplot(2,1,2);quiver(x,y)
title('箭头图')
```

程序运行结果如图 5-19 所示。

例 5-18 绘制空间曲线。

程序如下：


```

t=0:pi/50:2*pi;
x=8*cos(t);
y=4*sqrt(2)*sin(t);
z=-4*sqrt(2)*sin(t);
plot3(x,y,z,'p')
title('Line in 3-D Space')
text(0,0,0,'origin')
xlabel('X'),ylabel('Y'),zlabel('Z');grid

```

例5-19 已知 $6 < x < 30$, $15 < y < 36$, 求不定方程 $2x+5y=126$ 的整数解。

程序如下:

```

x=7:29;
y=16:35;
[x,y]=meshgrid(x,y); %在[7,29]×[16,35]区域生成网格坐标
z=2*x+5*y;
k=find(z==126); %找出解的位置
x(k)',y(k)' %输出对应位置的 x, y 即方程的解

```

例5-20 绘制三维曲面图 $z = \sin y \cos x$ 。

为便于分析各种三维曲面的特征, 下面画出了3种不同形式的曲面。

程序1:

```

x=0:0.1:2*pi;
[x,y]=meshgrid(x);
z=sin(y).*cos(x);
mesh(x,y,z)
xlabel('x-axis'),ylabel('y-axis'),zlabel('z-axis')
title('mesh')

```

程序2:

```

x=0:0.1:2*pi;
[x,y]=meshgrid(x);
z=sin(y).*cos(x);
surf(x,y,z)
xlabel('x-axis'),ylabel('y-axis'),zlabel('z-axis')
title('surf')

```

程序3:

```

x=0:0.1:2*pi;
[x,y]=meshgrid(x);
z=sin(y).*cos(x);
plot3(x,y,z)
xlabel('x-axis'),ylabel('y-axis'),zlabel('z-axis')
title('plot3');grid

```

例 5-21 绘制两个相互垂直且直径相等的圆柱体的相交图形。

程序如下:

```

m=30;
z=1.2*(0:m)/m;
r=ones(size(z));
theta=(0:m)/m*2*pi;
x1=r'*cos(theta);           %生成第一个圆柱体的坐标矩阵
y1=r'*sin(theta);
z1=z'*ones(1,m+1);
x=(-m:2:m)/m;
x2=x'*ones(1,m+1);          %生成第二个圆柱体的坐标矩阵
y2=r'*cos(theta);
z2=r'*sin(theta);
surf(x1,y1,z1)               %绘制垂直的圆柱体
axis equal,axis off
hold on
surf(x2,y2,z2)               %绘制水平的圆柱体
axis equal,axis off
title('两个圆柱体的相交图形')
hold off

```

例5-22 分析由函数 $z=x^2-2y^2$ 构成的曲面形状及与平面 $z=a$ 的交线。

程序如下：

```

[x,y]=meshgrid(-10:0.2:10);
z1=(x.^2-2*y.^2)+eps;       %第一个曲面坐标
a=input('a=?');
z2=a*ones(size(x));         %第二个曲面坐标
subplot(2,2,1)
mesh(x,y,z1);hold on;mesh(x,y,z2) %分别画出两个曲面
v=[-10,10,-10,10,-100,100];axis(v);grid %第一子图的坐标设置
hold off
r0=abs(z1-z2)<=1;           %求两曲面 z 坐标差小于 1 的点
xx=r0.*x;yy=r0.*y;zz=r0.*z2; %求这些点上的 x、y、z 坐标，即交线坐标
subplot(2,2,2)
plot3(xx(r0~=0),yy(r0~=0),zz(r0~=0),'*') %在第二子图画上交线
axis(v);grid                %第二子图的坐标设置

```

例5-23 在xy平面内选择区域 $[-8, 8] \times [-8, 8]$ ，绘制函数的4种三维曲面图（墨西哥帽子图形）。

程序如下：

```

[x,y]=meshgrid(-8:0.5:8);
z=sin(sqrt(x.^2+y.^2))./sqrt(x.^2+y.^2+eps);
subplot(2,2,1)
meshc(x,y,z)
title('meshc(x,y,z)')
subplot(2,2,2)
meshz(x,y,z)

```

```

title('meshz(x,y,z)')
subplot(2,2,3)
surfc(x,y,z)
title('surfc(x,y,z)')
subplot(2,2,4)
surfl(x,y,z)
title('surfl(x,y,z)')

```

例5-24 绘制标准三维曲面图形。

程序如下：

```

t=0:pi/20:2*pi;
[x,y,z]=cylinder(2+sin(t),30);
subplot(2,3,1);surf(x,y,z) %生成一个正弦型柱面
[x,y,z]=sphere;
subplot(2,3,2);surf(x,y,z);axis equal %生成一个球面
[x,y,z]=peaks(30);
subplot(2,3,3);meshz(x,y,z) %生成一个多峰曲面

```

例 5-25 绘制以下三维图形。

- ① 绘制魔方阵的三维条形图。
- ② 已知 $x=[2347, 1827, 2043, 3025]$ ，绘制三维饼图。
- ③ 用随机的顶点坐标值画出五个黄色三角形。
- ④ 以三维杆图形式绘制曲线 $y=2\sin x$ 。

程序如下：

```

subplot(2,2,1);bar3(magic(4));title('bar3')
subplot(2,2,2);pie3([2347,1827,2043,3025]);title('pie3')
a=rand(3,5);b=rand(3,5);c=rand(3,5);
subplot(2,2,3);fill3(a,b,c,'y');title('fill3')
y=2*sin(0:pi/10:2*pi);
subplot(2,2,4);stem3(y);title('stem3')

```

例5-26 绘制多峰函数的瀑布图和等高线图。

程序如下：

```

[X,Y,Z]=peaks(30);
subplot(2,2,1);waterfall(X,Y,Z)
xlabel('X-axis'),ylabel('Y-axis'),zlabel('Z-axis')
subplot(2,2,2);contour3(X,Y,Z,12,'k') %其中 12 代表高度的等级数
xlabel('X-axis'),ylabel('Y-axis'),zlabel('Z-axis')

```

例 5-27 隐函数绘图应用举例。

程序如下：

```

subplot(2,2,1);ezplot('x^2+y^2-9');axis equal
subplot(2,2,2);ezplot(@(x,y)x.^3+y.^3-5.*x.*y+1/5)
subplot(2,2,3);ezplot('cos(tan(pi*x))',[0,1])

```

```
subplot(2,2,4);ezplot('8*cos(t)','4*sqrt(2)*sin(t)',[0,2*pi])
```

例 5-28 绘制下列曲面。

命令如下：

```
>> ezsurf('exp(-s)*cos(t)','exp(-s)*sin(t)','t',[0,8,0,5*pi])
```

例 5-29 从不同视点绘制多峰函数曲面。

程序如下：

```
subplot(2,2,1);mesh(peaks)
view(-37.5,30)      %指定子图 1 的视点
title('azimuth=-37.5,elevation=30')
subplot(2,2,2);mesh(peaks)
view(0,90)          %指定子图 2 的视点
title('azimuth=0,elevation=90')
subplot(2,2,3);mesh(peaks)
view(90,0)          %指定子图 3 的视点
title('azimuth=90,elevation=0')
subplot(2,2,4);mesh(peaks)
view(-7,-10)        %指定子图 4 的视点
title('azimuth=-7,elevation=-10')
```

例 5-30 采用不同着色方式的效果展示。

程序如下：

```
t=0:pi/20:2*pi;
z=peaks(20);
colormap(jet)
subplot(2,2,1);surf(z)
subplot(2,2,2);surf(z);shading flat
subplot(2,2,3);surf(z);shading interp
subplot(2,2,4);mesh(z);shading interp
```

例5-31 绘制两个球面，其中一个球在另一个球里面，将外面的球裁掉一部分，使得能看见里面的球。

程序如下：

```
[x,y,z]=sphere(20);
%生成外面的大球
z1=z;
z1(:,1:4)=NaN;    %将大球裁掉一部分
c1=ones(size(z1));
surf(3*x,3*y,3*z1,c1)
%生成里面的小球
hold on
z2=z;
c2=2*ones(size(z2));
c2(:,1:4)=3*ones(size(c2(:,1:4)));
```

```
surf(1.5*x,1.5*y,1.5*z2,c2)
colormap([0,1,0;0.5,0,0;1,0,0])
grid on
hold off
```

例 5-32 播放一个直径不断变化的球体。

程序如下：

```
[x,y,z]=sphere(50);
m=moviein(30); %建立一个 30 列大矩阵
for i=1:30
    surf(i*x,i*y,i*z) %绘制球面
    m(:,i)=getframe; %将球面保存到 m 矩阵
end
movie(m,10) %以每秒 10 幅的速度播放球面
```

例 5-33 生成一个三维运动图形轨迹。

```
x=0:pi/250:10*pi;
y=sin(x);
z=cos(x);
comet3(x,y,z)
```

第 6 章 MATLAB 数据分析与多项式计算

例 6-1 求矩阵 A 每行及每列的最大值，并求整个矩阵的最大值。

```
>> A=[13,-56,78;25,63,-235;78,25,563;1,0,-1];
>> max(A,[],2) %求每行最大值
>> max(A) %求每列最大值
>> max(max(A)) %求整个矩阵的最大值，也可使用 max(A(:))
```

例 6-2 求矩阵 A 的每行元素之和和全部元素之和。

```
>> A=[1,2,3,4;5,6,7,8;9,10,11,12];
>> S=sum(A,2)
>> sum(S)
```

例 6-3 求 $s=1!+2!+\cdots+6!$ 的值。

```
>> x=cumprod(1:6)
>> s=sum(x)
```

例 6-4 对二维矩阵 x，从不同维方向求出其标准差和方差。

```
>> x=[4,5,6;1,4,8];
>> y1=std(x,0,1) %求标准差
>> v1=var(x,0,1) %求方差
>> y2=std(x,1,1)
>> v2=var(x,1,1)
>> y3=std(x,0,2)
>> v3=var(x,0,2)
```

```
>> y4=std(x,1,2)
>> v4=var(x,1,2)
```

例 6-5 生成满足正态分布的 10000×5 随机矩阵，然后求各列元素的均值和标准差，再求这 5 列随机数据的相关系数矩阵。

```
>> X=randn(10000,5);
>> M=mean(X)
>> D=std(X)
>> R=corrcoef(X)
>> R=corrcoef(X(:,1),X(:,2)) %X 前两列的相关系数
```

例 6-6 对下列矩阵做各种排序。

```
>> A=[1,-8,5;4,12,6;13,7,-13];
>> sort(A) %对 A 的每列按升序排序
>> sort(A,2,'descend') %对 A 的每行按降序排序
>> [X,I]=sort(A) %对 A 按列排序，并将每个元素所在行号送矩阵 I
```

例 6-7 求多项式 x^4+8x^3-10 与多项式 $2x^2-x+3$ 的乘积。

```
>> A=[1,8,0,0,-10];
>> B=[2,-1,3];
>> C=conv(A,B)
```

例 6-8 求多项式 x^4+8x^3-10 除以多项式 $2x^2-x+3$ 的结果。

```
>> A=[1,8,0,0,-10];
>> B=[2,-1,3];
>> [P,r]=deconv(A,B)
```

例 6-9 求有理分式的导数。

```
>> P=1;
>> Q=[1,0,5];
>> [p,q]=polyder(P,Q)
```

例 6-10 已知多项式 x^4+8x^3-10 ，分别取 $x=1.2$ 和一个 2×3 矩阵为自变量，计算该多项式的值。

```
>> A=[1,8,0,0,-10]; %4 次多项式系数
>> x=1.2; %取自变量为一数值
>> y1=polyval(A,x)
>> x=[-1,1.2,-1.4;2,-1.8,1.6]; %给出一个矩阵 x
>> y2=polyval(A,x) %分别计算矩阵 x 中各元素为自变量的多项式之值
```

例 6-11 仍以多项式 x^4+8x^3-10 为例，以 2×2 矩阵为自变量分别用 polyval 和 polyvalm 计算该多项式的值。

```
>> A=[1,8,0,0,-10]; %多项式系数
>> x=[-1,1.2;2,-1.8]; %给出一个矩阵 x
>> y1=polyval(A,x) %计算代数多项式的值
>> y2=polyvalm(A,x) %计算矩阵多项式的值
```

例 6-12 求多项式 x^4+8x^3-10 的根。

```
>> A=[1,8,0,0,-10];  
>> x=roots(A)
```

例 6-13 已知:

$$f(x)=3x^5+4x^3-5x^2-7.2x+5$$

① 计算 $f(x)=0$ 的全部根。

② 由方程 $f(x)=0$ 的根构造一个多项式 $g(x)$, 并与 $f(x)$ 进行对比。

```
>> P=[3,0,4,-5,-7.2,5];  
>> X=roots(P)           %求方程 f(x)=0 的根  
>> G=poly(X)            %求多项式 g(x)
```

例 6-14 给出以下概率积分的数据表如表 6-1 所示, 用不同的插值方法计算 $f(0.472)$ 。

这是一个一维插值问题, 命令如下:

```
>> x=0.46:0.01:0.49;           %给出 x 和 f(x)  
>> f=[0.4846555,0.4937542,0.5027498,0.5116683];  
>> format long  
>> interp1(x,f,0.472)          %用默认方法, 即线性插值计算 f(0.472)  
>> interp1(x,f,0.472,'nearest') %用最近点插值计算 f(0.472)  
>> interp1(x,f,0.472,'pchip')   %用 3 次埃尔米特插值计算 f(0.472)  
>> interp1(x,f,0.472,'spline')  %用 3 次样条插值计算 f(0.472)  
format short
```

例 6-15 某检测参数 f 随时间 t 的采样结果如表 6-2, 用数据插值法计算 $t=2, 12, 22, 32, 42, 52$ 时的 f 值。

这是一个一维数据插值问题, 命令如下:

```
>> T=0:5:65;  
>> X=2:10:52;  
>> F=[3.2015,2.2560,879.5,1835.9,2968.8,4136.2,5237.9,6152.7,...  
6725.3,6848.3,6403.5,6824.7,7328.5,7857.6];  
>> F1=interp1(T,F,X)           %用线性插值方法插值  
>> F2=interp1(T,F,X,'nearest') %用最近点插值方法插值  
>> F3=interp1(T,F,X,'pchip')   %用 3 次埃尔米特插值方法插值  
>> F4=interp1(T,F,X,'spline')  %用 3 次样条插值方法插值
```

例 6-16 设 $z=x^2+y^2$, 对 z 函数在 $[0, 1] \times [0, 2]$ 区域内进行插值。

```
>> x=0:0.1:1;y=0:0.2:2;  
>> [X,Y]=meshgrid(x,y);       %产生自变量网格坐标  
>> Z=X.^2+Y.^2;               %求对应的函数值  
>> interp2(x,y,Z,0.5,0.5)      %在 (0.5,0.5) 点插值  
>> interp2(x,y,Z,[0.5 0.6],0.4) %在 (0.5,0.4) 点和 (0.6,0.4) 点插值  
>> interp2(x,y,Z,[0.5 0.6],[0.4 0.5]) %在 (0.5,0.4) 点和 (0.6,0.5) 点插值  
%下一命令在 (0.5,0.4), (0.6,0.4), (0.5,0.5) 和 (0.6,0.5) 各点插值
```

```
>> interp2(x,y,z,[0.5 0.6]','[0.4 0.5])
>> interp2(x,y,z,[0.5 0.6]','[0.4 0.5'],'spline')
```

例 6-17 某实验对一根长 10m 的钢轨进行热源的温度传播测试。用 d 表示测量点距离 (m)，用 t 表示测量时间 (s)，用 c 表示测得各点的温度 ($^{\circ}\text{C}$)，测量结果如表 6-3 所示。

试用线性插值求出在一分钟内每隔 20s、钢轨每隔 2m 处的温度。

程序如下：

```
d=0:2.5:10;
t=(0:30:60)';
c=[95,14,0,0,0;88,48,32,12,6;67,64,54,48,41];
di=0:2:10;
ti=(0:20:60)';
ci=interp2(d,t,c,di,ti)
```

例 6-18 用一个 3 次多项式在区间 $[0, 2\pi]$ 内逼近函数 $\sin x$ 。

在给定区间上，均匀地选择 50 个采样点，并计算采样点的函数值，然后利用 3 次多项式逼近。命令如下：

```
>> X=linspace(0,2*pi,50);
>> Y=sin(X);
>> P=polyfit(X,Y,3) %得到 3 次多项式的系数和误差
下面利用绘图的方法将多项式 p(x)和 sinx 进行比较，继续执行下列命令：
>> X=linspace(0,2*pi,20);
>> Y=sin(X);
>> Y1=polyval(P,X);
>> plot(X,Y,':o',X,Y1,'-*')
```

第 7 章 MATLAB 数值微分与积分

例 7-1 设 x 由 $[0, 2\pi]$ 间均匀分布的 6 个点组成，求 $\sin x$ 的 1~3 阶差分。

```
>> X=linspace(0,2*pi,6);
>> Y=sin(X)
>> DY=diff(Y) %计算 Y 的一阶差分
>> D2Y=diff(Y,2) %计算 Y 的二阶差分，也可用命令 diff(DY) 计算
>> D3Y=diff(Y,3) %计算 Y 的三阶差分，也可用 diff(D2Y) 或 diff(DY,2)
```

例 7-2 用不同的方法求函数 $f(x)$ 的数值导数，并在同一个坐标系中做出 $f(x)$ 的图像。

```
f=@(x) sqrt(x.^3+2*x.^2-x+12)+(x+5).^(1/6)+5*x+2;
g=@(x) (3*x.^2+4*x-1)./sqrt(x.^3+2*x.^2-x+12)/2+1/6./(x+5).^(5/6)+5;
x=-3:0.01:3;
p=polyfit(x,f(x),5); %用 5 次多项式 p 拟合 f(x)
dp=polyder(p); %对拟合多项式 p 求导数 dp
dpx=polyval(dp,x); %求 dp 在假设点的函数值
dx=diff(f([x,3.01]))/0.01; %直接对 f(x) 求数值导数
gx=g(x); %求函数 f 的导函数 g 在假设点的导数
```



```
plot(x,dpx,x,dx,'.',x,gx,'-') %作图
```

例 7-3 求 $I = \int_0^1 e^{-x^2} dx$ 。

先建立一个函数文件 **fex.m**。

```
function f=fex(x)
```

```
f=exp(-x.^2);
```

接下来调用数值积分函数 **quad** 求定积分，命令如下：

```
>> [I,n]=quad(@fex,0,1)
```

也可不建立关于被积函数的函数文件，而使用匿名函数（或内联函数）求解，命令如下：

```
>> f=@(x) exp(-x.^2); %用匿名函数 f(x) 定义被积函数
```

```
>> [I,n]=quad(f,0,1) %注意函数句柄 f 前面不加 @ 号
```

例 7-4 分别用 **quad** 函数和 **quadl** 函数求积分的近似值，并在相同的积分精度下，比较函数的调用次数。

```
>> format long
```

```
>> f=@(x) 4./(1+x.^2);
```

```
>> [I,n]=quad(f,0,1,1e-8) %调用函数 quad 求定积分
```

```
>> [I,n]=quadl(f,0,1,1e-8) %调用函数 quadl 求定积分
```

```
>> (atan(1)-atan(0))*4 %理论值
```

```
>> format
```

例 7-5 求 $I = \int_1^e \frac{1}{x\sqrt{1-\ln^2 x}} dx$ 。

先建立被积函数文件 **fe.m**。

```
function f=fe(x)
```

```
f=1./(x.*sqrt(1-log(x).^2));
```

再调用数值积分函数 **integral** 求定积分，命令如下：

```
>> I=integral(@fe,1,exp(1))
```

例 7-6 求 $\int_{\frac{2}{\pi}}^{+\infty} \frac{1}{x^2} \sin \frac{1}{x} dx$ 。

建立被积函数文件 **fsx.m**。

```
function f=fsx(x)
```

```
f=sin(1./x)./x.^2;
```

调用函数 **quadgk** 求定积分，命令如下：

```
>> I=quadgk(@fsx,2/pi,+Inf)
```

例 7-7 用 **trapz** 函数计算定积分。

```
>> format long
```

```
>> x=0:0.001:1;
```

```
>> y=4./(1+x.^2); %生成函数向量
```

```
>> trapz(x,y)
```

```
>> format
```

5. 累计梯形积分

在 **MATLAB** 中，提供了对数据积分逐步累计的函数 **cumtrapz**。该函数调用格式如下。

```
Z=cumtrapz(Y)
```

```
Z=cumtrapz(X,Y)
```

函数其他参数的含义和用法与 **trapz** 函数的相同。例如：

```
>> S=cumtrapz([1:5;2:6]')
```

例 7-8 计算二重定积分。

建立一个函数文件 **fxym**。

```
function f=fxym(x,y)
```

```
global ki;
```

```
ki=ki+1; %ki 用于统计被积函数的调用次数
```

```
f=exp(-x.^2/2).*sin(x.^2+y);
```

调用函数求解，命令如下：

```
>> global ki;
```

```
>> ki=0;
```

```
>> I=integral2(@fxym,-2,2,-1,1) %调用 integral2 函数求解
```

```
>> ki=0;
```

```
>> I=quad2d(@fxym,-2,2,-1,1) %调用 quad2d 函数求解
```

```
>> ki
```

```
>> ki=0;
```

```
>> I=dblquad(@fxym,-2,2,-1,1) %调用 dblquad 函数求解
```

例 7-9 计算三重定积分

$$\int_0^1 \int_0^\pi \int_0^\pi 4xyz^{-z^2y-x^2} dx dy dz$$

```
>> fxyz=@(x,y,z) 4*x.*z.*exp(-z.*z.*y-x.*x); %定义被积函数
```

```
>> integral3(fxyz,0,pi,0,pi,0,1)
```

```
>> triplequad(fxyz,0,pi,0,pi,0,1,1e-7)
```

例 7-10 给定数学函数

$$x(t)=12\sin(2\pi \times 10t+\pi/4)+5\cos(2\pi \times 40t)$$

取 **N=128**，试对 **t** 从 **0~1s** 采样，用 **fft** 函数作快速傅里叶变换，绘制相应的振幅—频率图。

程序如下：

```
N=128;
```

```
% 采样点数
```

```
T=1;
```

```
% 采样时间终点
```

```
t=linspace(0,T,N);
```

```
% 给出 N 个采样时间
```

```
x=12*sin(2*pi*10*t+pi/4)+5*cos(2*pi*40*t); % 求各采样点样本值 x
```

```
dt=t(2)-t(1);
```

```
% 采样周期
```

```
f=1/dt;
```

```
% 采样频率 (Hz)
```

```
X=fft(x);
```

```
% 计算 x 的快速傅里叶变换 X
```

```
F=X(1:N/2+1);
```

```
% F(k)=X(k) (k=1:N/2+1)
```

```
f=f*(0:N/2)/N;
```

```
% 使频率轴 f 从零开始
```

```
plot(f,abs(F),'-')
```

```
% 绘制振幅—频率图
```

```
xlabel('Frequency');
```

```
ylabel('|F(k)|')
```

求 **x** 的快速傅里叶逆变换，并与原函数进行比较，命令如下：

```
>> ix=real(ifft(X)); %求逆变换，结果只取实部
```

```
>> plot(t,x,t,ix,':') %逆变换结果和原函数的曲线
```

```
>> norm(x-ix) %逆变换结果和原函数之间的距离
```

第8章 MATLAB 方程数值求解

例 8-1 用直接解法求解下列线性方程组。

程序如下：

```
A=[2,1,-5,1;1,-5,0,7;0,2,1,-1;1,6,-1,-4];
b=[13,-9,6,0]';
x=A\b
```

例 8-2 用 LU 分解求解例 8-1 中的线性方程组。

程序如下：

```
A=[2,1,-5,1;1,-5,0,7;0,2,1,-1;1,6,-1,-4];
b=[13,-9,6,0]';
[L,U]=lu(A);
x=U\ (L\b)
```

例 8-3 用 QR 分解求解例 8-1 中的线性方程组。

程序如下：

```
A=[2,1,-5,1;1,-5,0,7;0,2,1,-1;1,6,-1,-4];
b=[13,-9,6,0]';
[Q,R]=qr(A);
x=R\ (Q\b)
```

例 8-4 用 Cholesky 分解求解例 8-1 中的线性方程组。

命令如下：

```
>> A=[2,1,-5,1;1,-5,0,7;0,2,1,-1;1,6,-1,-4];
>> b=[13,-9,6,0]';
>> R=chol(A)
```

Jacobi 迭代法的 MATLAB 函数文件 jacobi.m 如下：

```
function [y,n]=jacobi(A,b,x0,ep)
if nargin==3
    ep=1.0e-6;
elseif nargin<3
    error
    return
end
D=diag(diag(A)); %求 A 的对角矩阵
L=-tril(A,-1); %求 A 的下三角阵
U=-triu(A,1); %求 A 的上三角阵
B=D\ (L+U);
f=D\b;
y=B*x0+f;
n=1; %迭代次数
while norm(y-x0)>=ep
    x0=y;
    y=B*x0+f;
    n=n+1;
end
```

例 8-5 用 Jacobi 迭代法求解下列线性方程组。设迭代初值为 0，迭代精度为 10^{-6} 。

在程序中调用函数文件 `jacobi.m`，程序如下：

```
A=[10,-1,0;-1,10,-2;0,-2,10];
b=[9,7,6]';
[x,n]=jacobi(A,b,[0,0,0]',1.0e-6)
```

Gauss-Serdel 迭代法的 MATLAB 函数文件 `gauseidel.m` 如下：

```
function [y,n]=gauseidel(A,b,x0,ep)
if nargin==3
    ep=1.0e-6;
elseif nargin<3
    error
    return
end
D=diag(diag(A));      %求 A 的对角矩阵
L=-tril(A,-1);        %求 A 的下三角阵
U=-triu(A,1);         %求 A 的上三角阵
G=(D-L)\U;
f=(D-L)\b;
y=G*x0+f;
n=1;                  %迭代次数
while norm(y-x0)>=ep
    x0=y;
    y=G*x0+f;
    n=n+1;
end
```

例 8-6 用 Gauss-Serdel 迭代法求解例 8-5 中的线性方程组。

在程序中调用函数文件 `gauseidel.m`，程序如下：

```
A=[10,-1,0;-1,10,-2;0,-2,10];
b=[9,7,6]';
[x,n]=gauseidel(A,b,[0,0,0]',1.0e-6)
```

例 8-7 分别用 Jacobi 迭代和 Gauss-Serdel 迭代法求解下列线性方程组，看是否收敛。

命令如下：

```
>> a=[1,2,-2;1,1,1;2,2,1];
>> b=[9;7;6];
>> [x,n]=jacobi(a,b,[0;0;0])
>> [x,n]=gauseidel(a,b,[0;0;0])
```

有了上面这些讨论，下面设计一个求解线性方程组的函数文件 `line_solution.m`。

```
function [x,y]=line_solution(A,b)
[m,n]=size(A);
y=[];
if norm(b)>0 %非齐次方程组
    if rank(A)==rank([A,b])
        if rank(A)==n %有唯一解
            disp('原方程组有唯一解 x')
            x=A\b;
```

```

else %方程组有无穷多个解，基础解系
    disp('原方程组有无穷个解，特解为 x，其齐次方程组的基础解系为 y')
    x=A\b;
    y=null(A,'r');
end
else
    disp('方程组无解') %方程组无解
    x=[];
end
else %齐次方程组
    disp('原方程组有零解 x')
    x=zeros(n,1); %0 解
    if rank(A)<n
        disp('方程组有无穷个解，基础解系为 y') %非 0 解
        y=null(A,'r');
    end
end
end

```

例 8-8 求解方程组。

程序如下：

```

A=[1,-2,3,-1;3,-1,5,-3;2,1,2,-2];
b=[1;2;3];
[x,y]=line_solution(A,b)

```

例 8-9 求方程组的通解。

$$\begin{cases} x_1 + x_2 - 3x_3 - x_4 = 1 \\ 3x_1 - x_2 - 3x_3 + 4x_4 = 4 \\ x_1 + 5x_2 - 9x_3 - 8x_4 = 0 \end{cases}$$

程序如下：

```

format rat %指定有理式格式输出
A=[1,1,-3,-1;3,-1,-3,4;1,5,-9,-8];
b=[1,4,0]';
[x,y]=line_solution(A,b);
x,y
format short %恢复默认的短格式输出

```

例 8-10 求 $f(x)=x-\frac{1}{x}+5$ 在 $x_0=-5$ 和 $x_0=1$ 作为迭代初值时的根。

先建立函数文件 fz.m。

```

function f=fz(x)
f=x-1./x+5;

```

然后调用 **fzero** 函数求根，命令如下：

```

>> fzero(@fz,-5) %以-5 作为迭代初值
>> fzero(@fz,1) %以 1 作为迭代初值

```

例 8-11 求下列方程组在 (1, 1, 1) 附近的解并对结果进行验证。

首先建立函数文件 myfun.m。

```
function F=myfun(X)
x=X(1);
y=X(2);
z=X(3);
F(1)=sin(x)+y+z^2*exp(x);
F(2)=x+y+z;
F(3)=x*y*z;
```

在给定的初值 $x_0=1$, $y_0=1$, $z_0=1$ 下, 调用 **fsolve** 函数求方程的根, 命令如下:

```
>> option=optimset('Display','off');
>> X=fsolve(@myfun,[1,1,1],option)
>> q=myfun(X)
```

例 8-12 求圆和直线的两个交点。

先建立方程组函数文件 **fxyz.m**。

```
function F=fxyz(X)
x=X(1);
y=X(2);
z=X(3);
F(1)=x^2+y^2+z^2-9;
F(2)=3*x+5*y+6*z;
F(3)=x-3*y-6*z-1;
```

再在 MATLAB 命令行窗口, 输入如下命令:

```
>> X1=fsolve(@fxyz,[-1,1,-1],optimset('Display','off')) %求第一个交点
>> X2=fsolve(@fxyz,[1,-1,1],optimset('Display','off')) %求第二个交点
```

例 8-13 求函数在区间 $(-10, -1)$ 和 $(1, 10)$ 上的最小值点。

命令如下:

```
>> f=@(x) x-1./x+5;
>> [x,fmin]=fminbnd(f,-10,-1) %求函数在 (-10, -1) 内的最小值点和最小值
>> fminbnd(f,1,10) %求函数在 (1, 10) 内的最小值点
```

例 8-14 求函数 f 在 $(0.5, 0.5, 0.5)$ 附近的最小值。

建立函数文件 **fxyz0.m**。

```
function f=fxyz0(u)
x=u(1);y=u(2);z=u(3);
f=x+y.^2./x/4+z.^2./y+2./z;
```

在 MATLAB 命令行窗口, 输入如下命令:

```
>> [U,fmin]=fminsearch(@fxyz0,[0.5,0.5,0.5]) %求函数的最小值点和最小值
```

例 8-15 求解有约束最优化问题。

$$\min_X f(X) = 0.4x_2 + x_1^2 + x_2^2 - x_1x_2 + \frac{1}{30}x_1^3$$

$$s.t. \begin{cases} x_1 + 0.5x_2 \geq 0.4 \\ 0.5x_1 + x_2 \geq 0.5 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

首先编写定义目标函数函数文件 **fop.m**。

```
function f=fop(x)
```

```
f=0.4*x(2)+x(1)^2+x(2)^2-x(1)*x(2)+1/30*x(1)^3;
再设定约束条件，并调用fmincon函数求解此约束最优化问题，程序如下：
x0=[0.5;0.5];
A=[-1,-0.5;-0.5,-1];
b=[-0.4;-0.5];
lb=[0;0];
option=optimset; option.LargeScale='off'; option.Display='off';
[x,f]=fmincon(@fop,x0,A,b,[],[],lb,[],[],option)
```

例8-16 求解线性规划问题。

$$\begin{array}{ll}
 \min & f(X) = -5x_1 - 4x_2 - 6x_3 \\
 X & s.t. \begin{cases} x_1 - x_2 + x_3 \leq 20 \\ 3x_1 + 2x_2 + 4x_3 \leq 42 \\ 3x_1 + 2x_2 \leq 30 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 \end{cases}
 \end{array}$$

命令

如下：

```
f=[-5;-4;-6];
A=[1,-1,1;3,2,4;3,2,0];
b=[20;42;30];
Aeq=[];
Beq=[];
LB=zeros(3,1);
[x,favl]=linprog(f,A,b,Aeq,Beq,LB)
```

例 8-17 设有初值问题，试求其数值解，并与精确解相比较（精确解为 $y(t)=\sqrt{t+1}+1$ ）。

建立函数文件 funt.m。

```
function yp=funt(t,y)
yp=(y^2-t-2)/4/(t+1);
求解微分方程，程序如下：
t0=0;
tf=10;
y0=2;
[t,y]=ode23(@funt,[t0,tf],y0); %求数值解
y1=sqrt(t+1)+1; %求精确解
plot(t,y,'b.',t,y1,'r-') %通过图形来比较
```

例 8-18 已知一个二阶线性系统的微分方程为：

取 $a=2$ ，绘制系统的时间响应曲线和相平面图。

函数 `ode23` 和 `ode45` 是对一阶常微分方程组设计的，因此对高阶常微分方程，需先将它转化为一阶常微分方程组，即状态方程。令 $x_2=x$ ， $x_1=x'$ ，则得到系统的状态方程：

$$\begin{cases} \dot{x}_2 = x_1 \\ \dot{x}_1 = -ax_2 \\ x_2(0) = 0, \quad x_1(0) = 1 \end{cases}$$

建立函数文件 sys.m。

```
function xdot=sys(t,x)
```

```
xdot=[-2*x(2);x(1)];
```

取 t0=0, tf=20, 求微分方程的解, 程序如下:

```
t0=0;
```

```
tf=20;
```

```
[t,x]=ode45(@sys,[t0,tf],[1,0]);
```

程序运行后, 查看结果。

```
>> [t,x]
```

为直观地表示方程的解, 可以绘制方程的时间响应曲线及相平面曲线 (如图 8-3 所示), 程序如下:

```
subplot(2,2,1);plot(t,x(:,2))
```

%系统时间响应曲线, 即 t-x

```
subplot(2,2,2);plot(x(:,2),x(:,1))
```

%系统相平面曲线, 即 x-x'

```
axis equal
```

第9章 MATLAB 符号计算

例 9-1 当 λ 取何值时, 以下齐次线性方程组有非零解。

对于齐次线性方程组 $Ax=0$, 当 $\text{rank}(A)<n$ 或 $|A|=0$ 时, 齐次方程组有非零解。程序如下:

```
syms lamda
```

```
A=[1-lamda,-2,4;2,3-lamda,1;1,1,1-lamda];
```

```
D=det(A);
```

```
factor(D)
```

例 9-2 求下列极限。

$$(1) \lim_{x \rightarrow a} \frac{\sqrt[m]{x} - \sqrt[m]{a}}{x - a}$$

$$(2) \lim_{x \rightarrow 0} \frac{\sin(a+x) - \sin(a-x)}{x}$$

$$(3) \lim_{x \rightarrow +\infty} x(\sqrt{x^2+1} - x)$$

$$(4) \lim_{x \rightarrow a^+} \frac{\sqrt{x} - \sqrt{a} + \sqrt{x-a}}{\sqrt{x^2-a^2}}$$

命令如下:

```
>> syms a m x;
```

```
>> f=(x^(1/m)-a^(1/m))/(x-a);
```

```
>> limit(f,x,a)
```

%求极限 (1)

```
ans =
```

```
a^(1/m - 1)/m
```

```
>> f=(sin(a+x)-sin(a-x))/x;
```

```
>> limit(f)
```

%求极限 (2)

```
ans =
```

```
2*cos(a)
```



```
>> f=x*(sqrt(x^2+1)-x);
>> limit(f,x,inf,'left')           %求极限(3)
ans =
1/2
>> f=(sqrt(x)-sqrt(a)-sqrt(x-a))/sqrt(x*x-a*a);
>> limit(f,x,a,'right')           %求极限(4)
ans =
-1/(2*a)^(1/2)
```

例 9-3 求下列函数的导数。

- (1) $y = \sqrt{1+e^x}$, 求 y' 。 (2) $y=x \cos x$, 求 y'' 、 y''' 。
- (3) $\begin{cases} x = a \cos t \\ y = a \sin t \end{cases}$, 求 y'_x 、 y''_x 。 (4) $z = \frac{xe^y}{y^2}$, 求 z'_x 、 z'_y 。

(5) $z=f(x,y)$ 由方程 $x^2+y^2+z^2=a^2$ 定义, 求 z'_x 、 z'_y 。

命令如下:

```
>> syms a b t x y z;
>> f=sqrt(1+exp(x));
>> diff(f)           %求(1)。未指定求导变量和阶数, 按默认规则处理
ans =
exp(x)/(2*(exp(x) + 1)^(1/2))
>> f=x*cos(x);
>> diff(f,x,2)       %求(2)。求 f 对 x 的二阶导数
ans =
-2*sin(x)-x*cos(x)
>> diff(f,x,3)       %求(2)。求 f 对 x 的三阶导数
ans =
x*sin(x) - 3*cos(x)
>> f1=a*cos(t);f2=b*sin(t);
>> diff(f2)/diff(f1) %求(3)。按参数方程求导公式求 y 对 x 的导数
ans =
-(b*cos(t))/(a*sin(t))
%求(3)。求 y 对 x 的二阶导数
>> (diff(f1)*diff(f2,2)-diff(f1,2)*diff(f2))/(diff(f1))^3
ans =
-(a*b*cos(t)^2 + a*b*sin(t)^2)/(a^3*sin(t)^3)
>> f=x*exp(y)/y^2;
>> diff(f,x)         %求(4)。z 对 x 的偏导数
ans =
exp(y)/y^2
>> diff(f,y)         %求(4)。z 对 y 的偏导数
ans =
```

```

(x*exp(y))/y^2 - (2*x*exp(y))/y^3
>> f=x^2+y^2+z^2-a^2;
>> zx=-diff(f,x)/diff(f,z)      %求(5)。
zx =
-x/z
>> zy=-diff(f,y)/diff(f,z)      %求(5)。
zy =
-y/z

```

例 9-4 在曲线 $y=x^3+3x-2$ 上哪一点的切线与直线 $y=4x-1$ 平行。

依题意，即求曲线哪一点的导数值为 4。程序如下：

```

x=sym('x');
y=x^3+3*x-2;          %定义曲线函数
f=diff(y);             %对曲线求导数
g=f-4;
solve(g)                %求方程 f-4=0 的根，见 9.4.1 小节

```

例 9-5 求下列不定积分。

$$(1) \int (3-x^2)^3 dx \qquad (2) \int \sin^2 x dx$$

$$(3) \int e^{\alpha} dt \qquad (4) \int \frac{5xt}{1+x^2} dt$$

命令如下：

```

>> x=sym('x');
>> f=(3-x^2)^3;
>> int(f)          %求(1)
ans =
-x^7/7 + (9*x^5)/5 - 9*x^3 + 27*x
>> f=sin(x)^2;
>> int(f)          %求(2)
ans =
x/2 - sin(2*x)/4
>> syms alpha t;
>> f=exp(alpha*t);
>> int(f)          %求(3)
ans =
exp(alpha*t)/alpha
>> f=5*x*t/(1+x^2);
>> int(f,t)        %求(4)
ans =
(5*t^2*x)/(2*(x^2 + 1))

```

例 9-6 求下列定积分。

$$(1) \int_1^2 |1-x| dx$$

$$(2) \int_{-\infty}^{+\infty} \frac{1}{1+x^2} dx$$

$$(3) \int_2^3 \frac{x^3}{(x-1)^{10}} dx$$

$$(4) \int_2^{\sin x} \frac{4x}{t} dt$$

命令如下:

```
>> x=sym('x');t=sym('t');
```

```
>> int(abs(1-x),1,2) %求(1)
```

```
ans =
```

```
1/2
```

```
>> f=1/(1+x^2);
```

```
>> int(f,-inf,inf) %求(2)
```

```
ans =
```

```
pi
```

```
>> f=x^3/(x-1)^10;
```

```
>> I=int(f,2,3) %求(3)
```

```
I =
```

```
138535/129024
```

```
>> double(I) %将上述符号结果转换为数值
```

```
ans =
```

```
1.0737
```

```
>> int(4*x/t,t,2,sin(x)) %求(4)
```

```
ans =
```

```
4*x*(log(sin(x)) - log(2))
```

例 9-7 求椭球的体积。

程序如下:

```
syms a b c z;
```

```
f=pi*a*b*(c^2-z^2)/c^2;
```

```
V=int(f,z,-c,c)
```

例 9-8 求下列级数之和。

$$(1) s_1 = 1 + \frac{1}{4} + \frac{1}{9} + \frac{1}{16} + \cdots + \frac{1}{n^2} + \cdots$$

$$(2) s_2 = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots + (-1)^{n+1} \frac{1}{n} + \cdots$$

$$(3) s_3 = x + 2x^2 + 3x^3 + \cdots + nx^n + \cdots$$

$$(4) s_4 = 1 + 4 + 9 + 16 + \cdots + 10000$$

命令如下:

```
>> n=sym('n');
```

```
>> s1=symsum(1/n^2,n,1,inf) %求 s1
```

```
s1 =
```

```
pi^2/6
```

```
>> s2=symsum((-1)^(n+1)/n,1,inf)           %求 s2。未指定求和变量，默认为 n
s2 =
log(2)
>> s3=symsum(n*x^n,n,1,inf)                 %求 s3。此处的求和变量 n 不能省略。
s3 =
piecewise([abs(x) < 1, x/(x - 1)^2])
>> s4=symsum(n^2,1,100)                     %求 s4。计算有限级数的和
s4 =
338350
```

例 9-9 求函数在指定点的泰勒级数展开式。

(1) 求 $\sqrt{1-2x+x^3} - \sqrt[3]{1-3x+x^2}$ 的 5 阶泰勒级数展开式。

(2) 将 $\frac{1+x+x^2}{1-x+x^2}$ 在 $x=1$ 处按 5 次多项式展开。

命令如下：

```
>> x=sym('x');
>> f1=sqrt(1-2*x+x^3)-(1-3*x+x^2)^(1/3);
>> f2=(1+x+x^2)/(1-x+x^2);
>> taylor(f1)                                %求 (1)。
ans =
(239*x^5)/72 + (119*x^4)/72 + x^3 + x^2/6
>> taylor(f2,x,1,'Order',6)                 %求 (2)。展开到 x-1 的 5 次幂时应选择 n=6
ans =
2*(x - 1)^3 - 2*(x - 1)^2 - 2*(x - 1)^5 + 3
```

例 9-10 解下列方程。

$$(1) \frac{1}{x+2} + \frac{4x}{x^2-4} = 1 + \frac{2}{x-2}$$

$$(2) x - \sqrt[3]{x^3 - 4x - 7} = 1$$

$$(3) 2\sin(3x - \frac{\pi}{4}) = 1$$

$$(4) x + xe^x - 10 = 0$$

命令如下：

```
>> syms x
>> x=solve(1/(x+2)+4*x/(x^2-4)==1+2/(x-2),x) %解方程 (1)
x =
1
>> syms x
>> f=x-(x^3-4*x-7)^(1/3)==1;
>> x=solve(f)                                %解方程 (2)
x =
3
>> syms x
>> x=solve(2*sin(3*x-pi/4)==1)                %解方程 (3)
x =
(5*pi)/36
(13*pi)/36
>> syms x
>> x=solve(x+x*exp(x)-10,x)                  %解方程 (4)。仅标出方程的左端
x =
```

1.6335061701558463841931651789789

例 9-11 求下列方程组的解。

$$(1) \begin{cases} \frac{1}{x^3} + \frac{1}{y^3} = 28 \\ \frac{1}{x} + \frac{1}{y} = 4 \end{cases}$$

$$(2) \begin{cases} x + y = 98 \\ \sqrt[3]{x} + \sqrt[3]{y} = 2 \end{cases}$$

$$(3) \begin{cases} u^3 + v^3 = 98 \\ u + v = 2 \end{cases}$$

$$(4) \begin{cases} x^2 + y^2 = 5 \\ 2x^2 - 3xy - 2y^2 = 0 \end{cases}$$

命令如下：

```
>> syms x y
>> [x y]=solve([1/x^3+1/y^3==28,1/x+1/y==4],[x,y]) %解方程组(1)
x =
    1
    1/3
y =
    1/3
    1
```

```
>> syms x y
>> [x y]=solve([x+y-98,x^(1/3)+y^(1/3)-2],[x,y]) %解方程组(2)
x =
Empty sym: 0-by-1
y =
Empty sym: 0-by-1
```

对方程组(2)，MATLAB 给出了无解的结论，显然错误，请看完全与其同构的方程组(3)。输入命令如

下：

```
>> syms u v
>> [u,v]=solve([u^3+v^3-98,u+v-2],[u,v]) %解方程组(3)
u =
    5
   -3
v =
   -3
    5
```

```
>> syms x y z
>> [x v]=solve([x^2+y^2-5,2*x^2-3*x*y-2*y^2]) %解方程组(4)
x =
    1
   -2
    2
   -1
v =
   -2
   -1
    1
    2
```

例 9-12 求下列微分方程的解。

(1) 求 $\frac{dy}{dx} = \frac{x^2 + y^2}{2x^2}$ 的通解。

(2) 求 $x^2 \frac{dy}{dx} + 2xy = e^x$ 的通解。

(3) 求 $\frac{dy}{dx} = \frac{x^2}{1+y^2}$ 的特解, $y(2)=1$ 。

(4) 求 $\begin{cases} \frac{dx}{dt} = 4x - 2y \\ \frac{dy}{dt} = 2x - y \end{cases}$ 的通解。

命令如下:

```
>> y=dsolve('Dy-(x^2+y^2)/x^2/2','x') %解(1)。方程的右端为0时可以不写
y =
      x
-x*(1/(C5 + log(x)/2) - 1)
>> y=dsolve('Dy*x^2+2*x*y-exp(x)','x') %解(2)
y =
-(C7 - exp(x))/x^2
>> y=dsolve('Dy-x^2/(1+y^2)','y(2)=1','x') %解(3)
y =
(((x^3/2 - 2)^2 + 1)^(1/2) + x^3/2 - 2)^(1/3) - 1/(((x^3/2 - 2)^2 + 1)^(1/2)
+ x^3/2 - 2)^(1/3)
>> [x,y]=dsolve('Dx=4*x-2*y','Dy=2*x-y','t') %解方程组(4)
x =
C12/2 + 2*C11*exp(3*t)
y =
C12 + C11*exp(3*t)
```

第10章 MATLAB 图形对象句柄

例10-1 在同一坐标下绘制红、绿两根不同曲线, 希望获得绿色曲线的句柄, 并对其进行设置。

程序如下:

```
x=0:pi/50:2*pi;
y=sin(x);
z=cos(x);
plot(x,y,'r',x,z,'g') %绘制两根不同的曲线
Hl=get(gca,'Children'); %获取两曲线句柄向量Hl
for k=1:size(Hl)
    if get(Hl(k),'Color')==[0,1,0] %[0,1,0]代表绿色
        Hlg=Hl(k); %获取绿色线条句柄
    end
end
pause %便于观察设置前后的效果
set(Hlg,'LineStyle',':','Marker','p') %对绿色线条进行设置
```

例 10-2 建立一个图形窗口。该图形窗口没有菜单条, 标题名称为“图形窗口示例”, 起始于屏幕左下角、宽度和高度分别为 300 像素点和 150 像素点, 背景颜色为绿色, 且当用户从键盘按下任意一个键时, 将显示“Hello,World!”字样。

命令如下:

```
>> hf=figure('Color',[0,1,0],'Position',[1,1,300,150],...
    'Name','图形窗口示例','NumberTitle','off','MenuBar','none',...
    'KeyPressFcn','disp(''Hello,World!'')');
```

例10-3 分别在4个不同的图形窗口绘制出正弦、余弦、正切、余切曲线。要求先建立一个图形窗口并绘图，然后每关闭一个再建立下一个，直到建立第4个窗口并绘图。

程序如下：

```
x=linspace(0,2*pi,60);
y=sin(x);
z=cos(x);
t=tan(x);
ct=1./(t+eps);;
%命令组待用
C4=['figure(''Name'', 'cotangent(x)', 'NumberTitle','', ...
    ''off'');plot(x,ct);axis([0,2*pi,-40,40]);'];
C3=['figure(''Name'', 'tangent(x)', 'DeleteFcn',C4,', ...
    ''NumberTitle'', 'off'');plot(x,t);axis([0,2*pi,-40,40]);'];
C2=['figure(''Name'', 'cos(x)', 'DeleteFcn',C3,', ...
    ''NumberTitle'', 'off'');plot(x,z);axis([0,2*pi,-1,1]);'];
%先创建一个图形窗口并绘制曲线
figure('Name', 'sin(x)', 'DeleteFcn',C2, 'NumberTitle', 'off')
plot(x,y)
axis([0,2*pi,-1,1])
```

例 10-4 利用坐标轴对象实现图形窗口的任意分割。

```
clf; %清除当前图形窗口的内容
x=linspace(0,2*pi,20);
y=sin(x);
axes('Position',[0.2,0.2,0.2,0.7], 'GridLineStyle', '-.')
plot(y,x);title('sin(x)-1')
axes('Position',[0.4,0.5,0.2,0.1])
stairs(x,y);title('sin(x)-2')
axes('Position',[0.55,0.6,0.25,0.3])
stem(x,y);title('sin(x)-3')
axes('Position',[0.55,0.2,0.25,0.3])
[x,y]=meshgrid(-8:0.5:8);
z=sin(sqrt(x.^2+y.^2))./sqrt(x.^2+y.^2+eps);
mesh(x,y,z); title('mesh(x,y,z)')
```

例 10-5 利用曲线对象绘制曲线。

程序如下：

```
t=0:pi/100:pi/2;
y1=sin(2*pi*t);
y2=sqrt(3)/2*exp(-4*t).*sin(4*sqrt(3)*t+pi/3);
axes('GridLineStyle',':','XLim',[0,pi/2], 'YLim', [-1,1])
line('XData',t, 'YData',y1, 'LineWidth',1)
line(t,y2)
```

```
grid on
```

例10-6 利用曲面对象绘制三维曲面 $z=\sin y \cos x$ 。

程序如下：

```
x=0:0.1:2*pi;
[x,y]=meshgrid(x);
z=sin(y).*cos(x);
axes('view',[-37.5,30])
hs=surface(x,y,z,'FaceColor','w','EdgeColor','flat');
grid on
xlabel('x-axis'),ylabel('y-axis'),zlabel('z-axis')
title('mesh-surf')
pause
set(hs,'FaceColor','flat')
```

例 10-7 利用曲线对象绘制 $y=\sin \theta$ 和 $y=\cos \theta$ 并利用文字对象完成标注。

程序如下：

```
theta=-pi:.1:pi;
y1=sin(theta);
y2=cos(theta);
h=line(theta,y1,'LineStyle',':','Color','g');
line(theta,y2,'LineStyle','--','Color','b')
xlabel('-\pi \leq \theta \leq \pi')
ylabel('sin\theta')
title('Plot of sin\theta')
text(-pi/4,sin(-pi/4),'<math>\leftarrow \sin(-\pi/4)</math>','FontSize',12)
set(h,'Color','r','LineWidth',2) %改变曲线 1 的颜色和线宽
```

例 10-8 绘制两个相互垂直且直径相等的圆柱体的交线。

%PLINE 绘制给定点和属性的直线。

% pline(P,w,c,s) 是绘制直线的函数

% 直线的坐标在矩阵 P 中描述，宽度是 w，颜色是 c，线型是 s

% 对于二维曲线，矩阵 P 的格式如下：

```
% [x1 x2 ... xn
```

```
% y1 y2 ... yn]
```

% 其中 x1、y1 是坐标的第一个点，其余坐标点类推。三维曲线相应改为 3 行

```
function pline(P,w,c,s)
```

```
[m,n]=size(P);
```

```
if m==2 %二维曲线
```

```
H1=plot(P(1,:),P(2,:));
```

```
set(H1,'LineWidth',w,'Color',c,'LineStyle',s)
```

```
elseif m==3 %三维曲线
```

```
H2=plot3(P(1,:),P(2,:),P(3,:));
```

```
set(H2,'LineWidth',w,'Color',c,'LineStyle',s)
```

```
else
```

```
error('输入参数 P 的维数不正确!')
```


end

下面是绘制两个相互垂直且直径相等的圆柱体的交线的程序。

```
%前视图
P0=[0;0];P1=[-5;5];P2=[-13;5];P3=[-13;-5];    %水平圆柱体坐标设置
P4=[13;-5];P5=[13;5];P6=[5;5];
patch([P0(1),P6(1),P5(1),P4(1),P3(1),P2(1),P1(1)],...
      [P0(2),P6(2),P5(2),P4(2),P3(2),P2(2),P1(2)], 'y')    %绘制水平圆柱体
axis([-15,30,-27,18]),axis equal,axis off
hold on
P7=[5;15];P8=[-5;15];    %垂直圆柱体坐标设置
patch([P0(1),P6(1),P7(1),P8(1),P1(1)],...
      [P0(2),P6(2),P7(2),P8(2),P1(2)], 'g')    %绘制垂直圆柱体
pline([P0,P1,P2,P3,P4,P5,P6,P0],2.5,'k','-')    %绘制水平圆柱体的外围线
pline([P6,P7,P8,P1],2.5,'k','-')    %绘制垂直圆柱体的外围线
%侧视图
r=5;    %两个圆柱体的半径
t=0:pi/100:2*pi;
x=23+r*cos(t);y=r*sin(t);    %水平圆柱体坐标设置
patch(x,y,'y')    %绘制水平圆柱体
P11=[28;0];P12=[28;15];P13=[18;15];P14=[18;0]; %垂直圆柱体坐标设置
t=0:pi/100:pi;
x1=23+r*cos(t);y1=r*sin(t);
patch([x1,P11(1),P12(1),P13(1),P14(1)],...
      [y1,P11(2),P12(2),P13(2),P14(2)], 'g')    %绘制垂直圆柱体
pline([x;y],2.5,'k','-')    %绘制水平圆柱体的外围线
pline([P11,P12,P13,P14],2,'k','-')    %绘制垂直圆柱体的外围线
%轴线
ha=plot([-14,29],[0,0],'k-.');    %绘制水平轴
ha.LineWidth=1.5;
ha=plot([0,0],[-6,16],'k-.');    %绘制前视图的垂直轴
ha.LineWidth=1.5;
ha=plot([23,23],[-6,16],'k-.');    %绘制侧视图的垂直轴
ha.LineWidth=1.5;
%标题及标注
ht=title('两个圆柱体的交线');
ht.FontSize=12;
ht=text(-3,-10,'水平圆柱体:  $y^2+z^2=r^2$ ');
ht.FontSize=12;
ht=text(-3,-15,'垂直圆柱体:  $x^2+y^2=r^2$ ');
ht.FontSize=12;
ht=text(-3,-20,'交线的投影:  $z^2-x^2=0$ ');
ht.FontSize=12;
hold off
```

例 10-9 在同一个图形窗口中创建矩形、圆角矩形、椭圆和圆各一个，并使用了不同的线型和线宽。

程序如下：

```
subplot(2,2,1)
rectangle('Position',[2,3,25,15],'LineWidth',3)
subplot(2,2,2)
rectangle('Position',[3,5,15,8],'Curvature',0.4,'LineWidth',2)
subplot(2,2,3)
rectangle('Position',[5,3,10,15],'Curvature',[1,1])
subplot(2,2,4)
rectangle('Position',[5,3,10,10],'Curvature',[1,1],...
        'LineStyle','--','FaceColor','r')
axis equal
```

例 10-10 绘制光照处理后的球面并观察不同光照模式下的效果。

程序如下：

```
[x,y,z]=sphere(20);
subplot(1,4,1)
surf(x,y,z);axis equal
shading interp
hold on
subplot(1,4,2)
surf(x,y,z);axis equal
light('Position',[0,1,1])
shading interp;lighting flat
hold on
plot3(0,1,1,'p');text(0,1,1,' light')
subplot(1,4,3)
surf(x,y,z);axis equal
light('Position',[0,1,1])
shading interp;lighting gouraud
hold on
subplot(1,4,4)
surf(x,y,z);axis equal
light('Position',[0,1,1])
shading interp;lighting phong
```

程序运行结果如图 10-10 所示，4 个球分别是没有使用光照、使用 flat 光照、使用 gouraud 光照和使用 phong 光照时的显示效果，第二个球还标出了光源的位置。

例 10-11 生成一个球体和一个立方体，观察表面反射特性设置效果。

程序如下：

```
sphere(36)
h=findobj('Type','surface');
set(h,'FaceLighting','phong','FaceColor','interp',...
    'EdgeColor',[0.4,0.4,0.4],'BackFaceLighting','lit')
```

```

hold on
vert=[2,0,-1;2,1,-1;3,0,0;3,0,-1;2,0,0;2,1,0;3,1,0;3,0,0];
fac=[1,2,3,4;2,6,7,3;4,3,7,8;1,5,8,4;1,2,6,5;5,6,7,8];
patch('Faces',fac,'Vertices',vert,'FaceColor','y')
light('Position',[1,3,2])
light('Position',[-3,-1,3])
material shiny
axis equal
hold off

```

第 11 章 MATLAB 图形用户界面设计

3. 建立控件对象举例

(1) 建立按钮和双位按钮

程序如下：

```

pbstart=uicontrol(gcf,'Style','push','Position',...
    [50,5,60,25],'String','Start Plot',...
    'CallBack','t=-pi:pi/20:pi;plot(t,sin(t))');
ptgrid=uicontrol(gcf,'Style','toggle','Position',...
    [150,5,60,25],'String','Grid','CallBack','grid');

```

(2) 建立单选按钮

程序如下：

```

set(gcf,'Color','R') %设置默认的背景颜色
htxt=uicontrol(gcf,'Style','text','String',...
    'Color Options','Position',[200,130,150,20]);
%建立单选按钮
hr=uicontrol(gcf,'Style','radio','String',...
    'Red','Position',[200,100,150,25],'Value',1,...
    'CallBack',['set(hr,'Value',1);','set(hb,'Value',0);','...
    'set(hy,'Value',0);','set(gcf,'Color','R')']);
hb=uicontrol(gcf,'Style','radio','String',...
    'Blue','Position',[200,75,150,25],...
    'CallBack',['set(hb,'Value',1);','set(hr,'Value',0);','...
    'set(hy,'Value',0);','set(gcf,'Color','B')']);
hy=uicontrol(gcf,'Style','radio','String',...
    'Yellow','Position',[200,50,150,25],...
    'CallBack',['set(hy,'Value',1);','set(hr,'Value',0);','...
    'set(hb,'Value',0);','set(gcf,'Color','Y')']);

```

(3) 建立复选框

程序如下：

```

htxt=uicontrol(gcf,'Style','text','Position',[200,125,150,25],...
    'String','Set Windows Properties');
hp=uicontrol(gcf,'Style','check','Position',...
    [200,100,150,25],'String','MyPosition',...
    'CallBack',['set(gcf,'Position',[10,10,300,250]);','...
    'if get(hp,'Value')==1,','...

```

```

        'set(gcf,'Position',[10,10,600,500]),','end']);
hc=uicontrol(gcf,'Style','checkbox','Position',...
    [200,75,150,25],'String','MyColor',...
    'Callback',['set(gcf,'color','w');',...
    'if get(hc,'Value')==1,'set(gcf,'color','g'),','end']);
hn=uicontrol(gcf,'Style','checkbox','Position',...
    [200,50,150,25],'String','MyName',...
    'Callback',['set(gcf,'Name','复选框未选中');',...
    'if get(hn,'Value')==1,'...
    'set(gcf,'Name','复选框被选中'),','end']);

```

(4) 建立弹出框

程序如下:

```

hpop=uicontrol(gcf,'Style','popup','String',...
    'red|blue|green|yellow|white','Position',[100,100,100,80],...
    'Callback',['cbcol=['R','B','G','Y'];',...
    'set(gcf,'Color',cbcol(get(hpop,'Value')))]);

```

(5) 建立列表框

界面中列表框的作用与(4)同,程序如下:

```

hl=uicontrol(gcf,'Style','list',...
    'String','red|blue|green|yellow|white',...
    'Position',[100,100,100,80],'Callback',...
    ['cbcol=['r','b','g','y','w','k'];',...
    'set(gcf,'color',cbcol(get(hl,'value')))]);

```

(6) 建立编辑框及边框

程序如下:

```

ftdir=uicontrol(gcf,'Style','frame',...
    'back','y','Position',[30,180,120,100]);
edmulti=uicontrol(gcf,'Style','edit',...
    'String','MATLAB is a very useful language.',...
    'Position',[50,200,75,55],'Max',2,'back','w');

```

(7) 建立滑动条

程序如下:

```

fig=figure('Position',[20,20,400,150]);
hsl1=uicontrol(fig,'Style','slider','Position',...
    [50,50,120,20],'Min',200,'Max',800,'Value',400,...
    'Callback',['set(azmcur,'String','...
    'num2str(get(hsl1,'Value')));',...
    'set(gcf,'Position',[20,20,get(hsl1,'Value'),300]);']);
hsl2=uicontrol(fig,'Style','slider','Position',...
    [240,50,120,20],'Min',100,'Max',600,'Value',300,...
    'Callback',['set(elvcur,'String','...
    'num2str(get(hsl2,'Value')));',...
    'set(gcf,'Position',[20,20,400,get(hsl2,'Value')]);']);
%用静态文本标出最小值
azmmin=uicontrol(fig,'Style','text','Position',...
    [20,50,30,20],'String',num2str(get(hsl1,'Min')));
elvmin=uicontrol(fig,'Style','text','Position',...

```

```

    [210,50,30,20], 'String', num2str(get(hsli2, 'Min')));
%用静态文本标出最大值
azmmx=uicontrol(fig, 'Style', 'text', 'Position', ...
    [170,50,30,20], 'String', num2str(get(hsli1, 'Max')));
elvmax=uicontrol(fig, 'Style', 'text', 'Position', ...
    [360,50,30,20], 'String', num2str(get(hsli2, 'Max')));
%用静态文本标出当前设置的宽度和高度
azmLabel=uicontrol(fig, 'Style', 'text', 'Position', ...
    [50,80,65,20], 'String', 'Width');
elvLabel=uicontrol(fig, 'Style', 'text', 'Position', ...
    [240,80,65,20], 'String', 'Height');
azmcur=uicontrol(fig, 'Style', 'text', 'Position', ...
    [120,80,50,20], 'String', num2str(get(hsli1, 'Value')));
elvcur=uicontrol(fig, 'Style', 'text', 'Position', ...
    [310,80,50,20], 'String', num2str(get(hsli2, 'Value')));

```

例 11-1 建立如图 11-8 所示的数制转换对话框，在左边输入一个十进制整数和 2~16 之间的数，单击“转换”按钮能在右边得到十进制数所对应的 2~16 进制字符串，单击“退出”按钮退出对话框。



图 11-8 数制转换对话框

程序如下：

```

hf=figure('Color',[0,1,1],'Position',[100,200,400,200],...
    'Name','数制转换','NumberTitle','off','MenuBar','none');
uicontrol(hf,'Style','Text','Position',[80,160,40,20],...
    'Horizontal','center','String','输入框','Back',[0,1,1])
uicontrol(hf,'Style','Text','Position',[280,160,40,20],...
    'Horizontal','center','String','输出框','Back',[0,1,1])
uicontrol(hf,'Style','Frame','Position',[20,65,165,90],'Back',[1,1,0])
uicontrol(hf,'Style','Text','Position',[25,110,85,25],...
    'Horizontal','center','String','十进制数','Back',[1,1,0])
uicontrol(hf,'Style','Text','Position',[25,75,85,25],...
    'Horizontal','center','String','2~16进制','Back',[1,1,0])
he1=uicontrol(hf,'Style','Edit','Position',[100,115,60,25],...
    'Back',[0,1,0]);
he2=uicontrol(hf,'Style','Edit','Position',[100,80,60,25],...
    'Back',[0,1,0]);
uicontrol(hf,'Style','Frame','Position',[215,65,165,90],'Back',[1,1,0])
ht=uicontrol(hf,'Style','Text','Position',[255,95,90,30],...

```

```

    'Horizontal','center','Back',[0,1,0]);
COMM=['n=str2num(get(he1,'String'))';',...
    'b=str2num(get(he2,'String'))';',...
    'dec=trdec(n,b);','set(ht,'string',dec);'];
uicontrol(hf,'Style','Push','Position',[55,20,90,25],...
    'String','转换','Call',COMM)
uicontrol(hf,'Style','Push','Position',[255,20,90,30],...
    'String','退出','Call','close(hf)')

```

程序调用了 `trdec.m` 函数文件，该函数的作用是将任意十进制整数转换为 2~16 进制字符串。`trdec.m` 函数文件如下：

```

function dec=trdec(n,b)
ch1='0123456789ABCDEF';      %十六进制的 16 个符号
k=1;
while n~=0                    %不断除某进制基数取余直到商为 0
    p(k)=rem(n,b);
    n=fix(n/b);
    k=k+1;
end
k=k-1;
strdec='';
while k>=1                    %形成某进制数的字符串
    kb=p(k);
    strdec=strcat(strdec,ch1(kb+1:kb+1));
    k=k-1;
end
dec=strdec;

```

例 11-2 建立如图 11-9 所示的图形演示对话框，在编辑框中输入绘图命令，当单击“绘图”按钮时，能在左边坐标轴绘制出所对应的图形，弹出框提供色图控制，列表框提供坐标网格线和坐标边框控制。

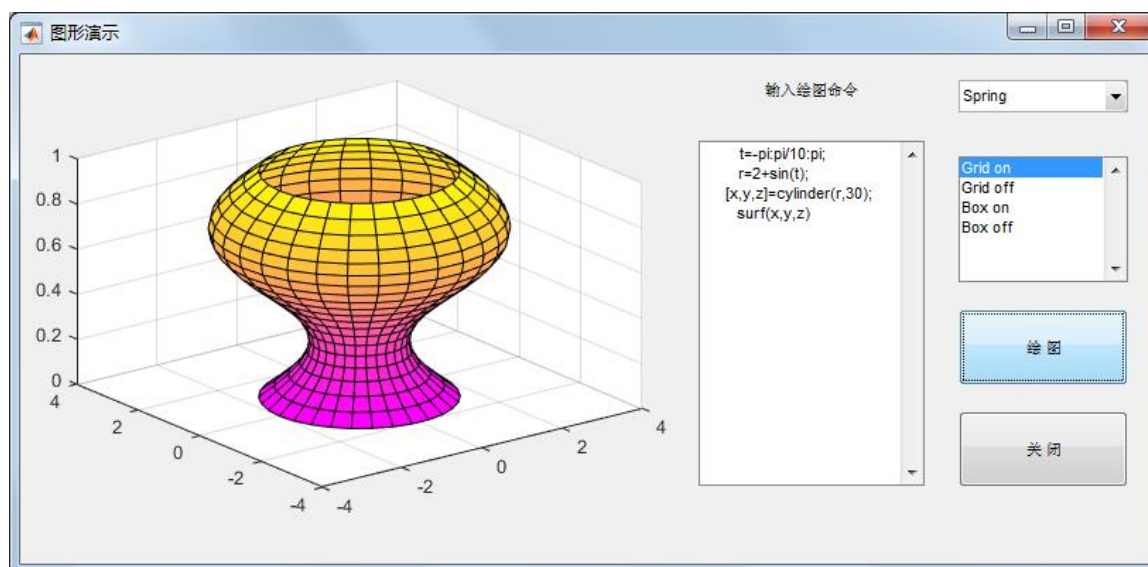


图 11-9 图形演示对话框

程序如下：

```

clf
set(gcf,'Unit','normalized','Position',[0.2,0.3,0.5,0.4])
set(gcf,'Menubar','none','Name','图形演示','NumberTitle','off')

```

```

axes('Position',[0.05,0.15,0.5,0.8])
uicontrol(gcf,'Style','text','Unit','normalized',...
    'Posi',[0.60,0.85,0.2,0.1],'String',...
    '输入绘图命令','Horizontal','center')
hedit=uicontrol(gcf,'Style','edit','Unit','normalized',...
    'Posi',[0.60,0.15,0.2,0.68],...
    'Max',2); %Max 取 2, 使 Max-Min>1, 从而允许多行输入
hpopup=uicontrol(gcf,'Style','popup','Unit','normalized',...
    'Posi',[0.83,0.8,0.15,0.15],'String',...
    'Spring|Summer|Autumn|Winter','Call',...
    'comm(hedit,hpopup,hlist)');
hlist=uicontrol(gcf,'Style','list','Unit','normalized',...
    'Posi',[0.83,0.55,0.15,0.25],'String',...
    'Grid on|Grid off|Box on|Box off','Call',...
    'comm(hedit,hpopup,hlist)');
hpush1=uicontrol(gcf,'Style','push','Unit','normalized',...
    'Posi',[0.83,0.35,0.15,0.15],'String',...
    '绘图','Call','comm(hedit,hpopup,hlist)');
uicontrol(gcf,'Style','push','Unit','normalized',...
    'Posi',[0.83,0.15,0.15,0.15],'String',...
    '关闭','Call','close all')

```

程序调用了 **comm.m** 函数文件，其定义如下：

```

function comm(hedit,hpopup,hlist)
com=get(hedit,'String');
n1=get(hpopup,'Value');
n2=get(hlist,'Value');
if ~isempty(com) %编辑框输入非空时
    eval(com); %执行从编辑框输入的命令
    chpop={'spring','summer','autumn','winter'};
    chlist={'grid on','grid off','box on','box off'};
    colormap(eval(chpop{n1}));
    eval(chlist{n2});
end

```

例 11-3 建立图形演示系统，系统菜单条中含有 3 个菜单项：Plot、Option 和 Quit。Plot 中有 Sine Wave 和 Cosine Wave 两个子菜单项，分别控制在本图形窗口画出正弦和余弦曲线。Option 菜单项的内容如图 11-10 所示，其中 Grid on 和 Grid off 控制给坐标轴加网格线，Box on 和 Box off 控制给坐标轴加边框，而且这 4 项只有在画有曲线时才是可选的。Window Color 控制图形窗口背景颜色。Quit 控制是否退出系统。

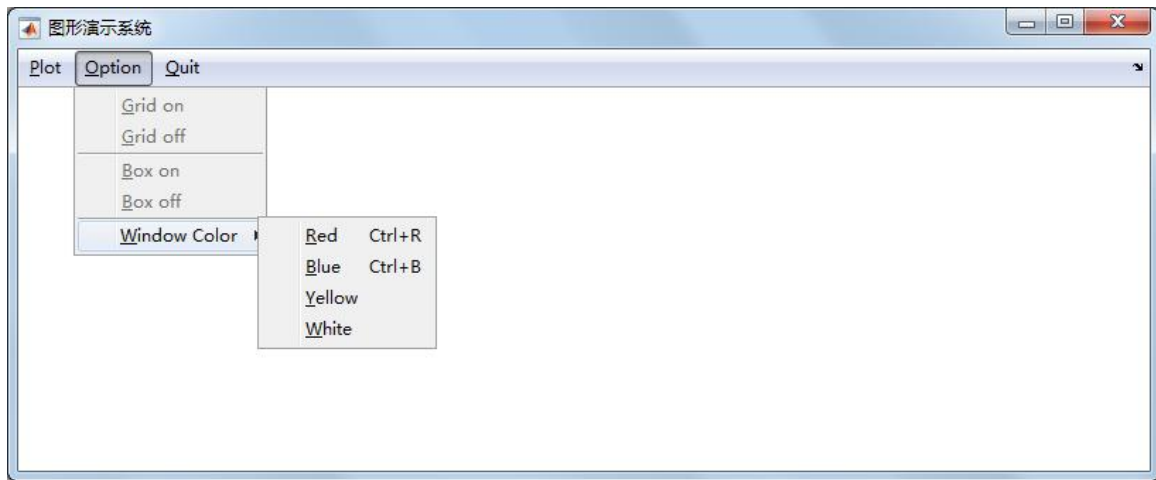


图 11-10 图形演示系统菜单

程序如下：

```
screen=get(0,'ScreenSize');
W=screen(3);
H=screen(4);
figure('Color',[1,1,1],'Position',[0.2*H,0.2*H,0.5*W,0.3*H],...
    'Name','图形演示系统','NumberTitle','off','MenuBar','none')
%定义 Plot 菜单项
hplot=uimenu(gcf,'Label','&Plot');
uimenu(hplot,'Label','Sine Wave','Call',...
    ['t=-pi:pi/20:pi;','plot(t,sin(t));',...
    'set(hgon,''Enable'', ''on'');',...
    'set(hgoff,''Enable'', ''on'');',...
    'set(hbon,''Enable'', ''on'');',...
    'set(hboff,''Enable'', ''on'')'])
uimenu(hplot,'Label','Cosine Wave','Call',...
    ['t=-pi:pi/20:pi;','plot(t,cos(t));',...
    'set(hgon,''Enable'', ''on'');',...
    'set(hgoff,''Enable'', ''on'');',...
    'set(hbon,''Enable'', ''on'');',...
    'set(hboff,''Enable'', ''on'')'])
%定义 Option 菜单项
hooption=uimenu(gcf,'Label','&Option');
hgon=uimenu(hooption,'Label','&Grid on',...
    'Call','grid on','Enable','off');
hgoff=uimenu(hooption,'Label','&Grid off',...
    'Call','grid off','Enable','off');
hbon=uimenu(hooption,'Label','&Box on',...
    'separator','on','Call','box on','Enable','off');
hboff=uimenu(hooption,'Label','&Box off',...
    'Call','box off','Enable','off');
hwincor=uimenu(hooption,'Label','&Window Color','Separator','on');
uimenu(hwincor,'Label','&Red','Accelerator','r',...
    'Call','set(gcf,''Color'', ''r'')')
uimenu(hwincor,'Label','&Blue','Accelerator','b',...
    'Call','set(gcf,''Color'', ''b'')')
uimenu(hwincor,'Label','&Yellow','Call',...
    'set(gcf,''Color'', ''y'')')
uimenu(hwincor,'Label','&White','Call',...
```



```

    'set(gcf,'Color','w'))
%定义 Quit 菜单项
uimenu(gcf,'Label','&Quit','Call','close(gcf)')

```

例 11-4 绘制曲线 $y=2e^{-0.5x}\sin(2\pi x)$ ，并建立一个与之相联系的快捷菜单，用以控制曲线的线型和曲线宽度。程序如下：

```

x=0:pi/100:2*pi;
y=2*exp(-0.5*x).*sin(2*pi*x);
hl=plot(x,y);
hc=uicontextmenu;                %建立快捷菜单
hls=uimenu(hc,'Label','线型');   %建立菜单项
hlw=uimenu(hc,'Label','线宽');
uimenu(hls,'Label','虚线','Call','set(hl,'LineStyle',':');')
uimenu(hls,'Label','实线','Call','set(hl,'LineStyle','-');')
uimenu(hlw,'Label','加宽','Call','set(hl,'LineWidth',2);')
uimenu(hlw,'Label','变细','Call','set(hl,'LineWidth',0.5);')
set(hl,'UIContextMenu',hc)      %将该快捷菜单和曲线对象联系起来

```

例 11-5 利用图形用户界面设计工具设计如图 11-22 所示的用户界面。该界面包括一个用于显示图形的轴对象，显示的图形包括表面图、网格图或等高线图。绘制图形的功能通过 3 个命令按钮来实现，用户通过单击相应的按钮，即可绘制相应图形。绘制图形所需要的数据通过一个弹出框来选取。在弹出框中包括 3 个选项，分别对应 MATLAB 的数据函数 Peaks、Membrane 和用户自定义的绘图数据 Sinc，用户可以通过选择相应的选项来载入相应的绘图数据。

(a) 为打开图形窗口事件编写响应代码。选择 MATLAB 编辑器的“编辑器”选项卡，在“导航”命令组中单击“转至”命令按钮，在弹出的菜单中选择 `guidemo_OpeningFcn` 函数，在以 `%varargin` 开头的注解语句下输入以下代码：

```

handles.peaks=peaks(35);
handles.membrane=membrane;
[x,y]=meshgrid(-8:0.3:8);
r=sqrt(x.^2+y.^2);
sinc=sin(r)./(r+eps);
handles.sinc=sinc;
handles.current_data=handles.sinc;
surf(handles.current_data)

```

(b) 为列表框编写响应代码。选择 `ChooseFun_Callback` 函数，在以 `%handles` 开头的注解语句下输入以下代码：

```

str=get(hObject,'String');
val=get(hObject,'Value');
%设置用所选函数产生当前数据集
switch str{val}
case 'Peaks'
    handles.current_data=handles.peaks;
case 'Membrane'
    handles.current_data=handles.membrane;
case 'Sinc'
    handles.current_data=handles.sinc;

```

```
end
% 保存句柄结构
guidata(hObject,handles)
```

例 11-6 在例 11-5 的图形窗口中添加图形窗口的默认菜单和一个自定义菜单项“设置”。“设置”菜单项的内容如图 11-23 所示，其中“网格线”控制给坐标轴加网格线，“边框”控制给坐标轴加边框，“背景色”控制图形窗口的背景颜色。

(a) 为“网格线”菜单项编写响应代码。选择 `hgong_Callback` 函数，在以 `%handles` 开头的注解语句下输入以下代码：

```
if strcmp(get(gcbo,'Checked'),'on')
    set(gcbo,'Checked','off')
    grid off
else
    set(gcbo,'Checked','on')
    grid on
end
```

(b) 为“边框”菜单项编写响应代码。选择 `hbon_Callback` 函数，在该区添加以下代码：

```
if strcmp(get(gcbo,'Checked'),'on')
    set(gcbo,'Checked','off')
    box off
else
    set(gcbo,'Checked','on')
    box on
end
```

(c) 为 3 个颜色菜单项编写响应代码。选择 `hwcYellow_Callback` 函数，在该区添加以下代码：

```
set(gcbf,'Color','y')
set(gcbo,'Checked','on')
set(handles.hwcBlue,'Checked','off')
set(handles.hwcWhite,'Checked','off')
```

选择 `hwcBlue_Callback` 函数，在该区添加以下代码：

```
set(gcbf,'Color','b')
set(gcbo,'Checked','on')
set(handles.hwcYellow,'Checked','off')
set(handles.hwcWhite,'Checked','off')
```

选择 `hwcWhite_Callback` 函数，在该区添加以下代码：

```
set(gcbf,'Color','w')
set(gcbo,'Checked','on')
set(handles.hwcYellow,'Checked','off')
set(handles.hwcBlue,'Checked','off')
```

第 12 章 MATLAB Simulink 系统仿真

例 12-8 采用 `S` 函数实现 $y = nx$ ，即把一个输入信号放大 n 倍。

① 利用 MATLAB 语言编写 `S` 函数，程序如下。

```
%*****
%S 函数 timesn.m, 其输出是输入的 n 倍
%*****
function [sys,x0,str,ts]=timesn(t,x,u,flag,n)
switch flag
    case 0
```

```

        [sys,x0,str,ts]=mdlInitializeSizes; %初始化
    case 3
        sys=mdlOutputs(t,x,u,n); %计算输出量
    case {1,2,4,9}
        sys=[];
    otherwise %出错处理
        error(num2str(flag))
end
%*****
%mdlInitializeSizes: 当 flag 为 0 时进行整个系统的初始化
%*****
function [sys,x0,str,ts]=mdlInitializeSizes()
%调用函数 simsizes 以创建结构 sizes
sizes=simsizes;
%用初始化信息填充结构 sizes
sizes.NumContStates=0; %无连续状态
sizes.NumDiscStates=0; %无离散状态
sizes.NumOutputs=1; %有一个输出量
sizes.NumInputs=1; %有一个输入信号
sizes.DirFeedthrough=1; %输出量中含有输入量
sizes.NumSampleTimes=1; %单个采样周期
%根据上面的设置设定系统初始化参数
sys=simsizes(sizes);
%给其他返回参数赋值
x0=[]; %设置初始状态为零状态
str=[]; %将 str 变量设置为空字符串
ts=[-1,0]; %假定继承输入信号的采样周期
%初始化子程序结束
%*****
%mdlOutputs: 当 flag 值为 3 时, 计算输出量
%*****
function sys=mdlOutputs(t,x,u,n)
sys=n*u;
%输出量计算子程序结束

```

例 12-9 采用 s 函数来构造非线性分段函数。

① 利用 MATLAB 语言编写 S 函数, 程序如下。

```

function [sys,x0,str,ts]=sfunction(t,x,u,flag)
switch flag
    case 0
        [sys,x0,str,ts]=mdlInitializeSizes;
    case 3
        sys=mdlOutputs(t,x,u);
    case {1,2,4,9}
        sys=[];
    otherwise
        error(['Unhandled flag=',num2str(flag)])
end

```

```

function[sys,x0,str,ts]=mdlInitializeSizes
sizes=simsizes;
sizes.NumContStates=0;
sizes.NumDiscStates=0;
sizes.NumOutputs=1;
sizes.NumInputs=1;
sizes.DirFeedthrough=1;
sizes.NumSampleTimes=1;
sys=simsizes(sizes);
x0=[];
str=[];
ts=[0,0];
function sys=mdlOutputs(t,x,u)
if u<1
    sys=3*sqrt(u);
elseif u>=1&u<3
    sys=3;
elseif u>=3&u<4
    sys=3-(u-3)^2;
elseif u>=4&u<5
    sys=2;
elseif u>=5&u<6
    sys=2-(u-5)^2;
else
    sys=1;
end
end

```

第 13 章 MATLAB 外部程序接口技术

例 13-1 计算当 $x=[0.0, 0.1, 0.2, \dots, 1.0]$ 时, $f(x)=e^x$ 的值, 并将结果写入文件 **demo1.txt**。程序如下:

```

x=0:0.1:1;
Y=[x;exp(x)];
fid=fopen('demo1.txt','w');
fprintf(fid,'%6.2f %12.8f\n',Y);
fclose(fid);

```

例 13-2 读出例 13-1 生成的文件 **demo1.txt** 中的数据。程序如下:

```

fid=fopen('demo1.txt','r');
while 1
    line=fgetl(fid);
    if line<0
        break
    end
    disp(line)
end
fclose(fid);

```

例 13-3 假定文件 `textdemo.txt` 中有以下格式的数据：

Name	English	Chinese	Mathmatics
Wang	99	98	100
Li	98	89	70
Zhang	80	90	97
Zhao	77	65	87

此文件第一行为标题行，第 2~5 行的第 1 列为字符型，后 3 列为整型。从该文件中将前 3 个数据读入到 `grades` 的程序段如下：

```
fid=fopen('textdemo.txt','r');
grades=textscan(fid,'%s %d %d %d',3,'headerlines',1);
```

例 13-4 假设文件 `alphabet.txt` 的内容是按顺序排列的 26 个大写英文字母，读取前 5 个字母的 ASCII 和这 5 个字符。

程序如下：

```
fid=fopen('alphabet.txt','r');
c=fread(fid,5);
frewind(fid);
d=fread(fid,5,'*char');
fclose(fid);
```

例 13-5 建立一数据文件 `magic5.dat`，用于存放 5 阶魔方阵。

程序如下：

```
fid=fopen('magic5.dat','w');
cnt=fwrite(fid,magic(5),'int32');
fclose(fid);
```

上述程序段将 5 阶魔方阵以 32 位整数格式写入文件 `magic5.dat` 中。下列程序则可实现对数据文件 `magic5.dat` 的读操作。

```
fid=fopen('magic5.dat','r');
[B,cnt]=fread(fid,[5,inf],'int32')
fclose(fid);
>> cnt
```

例 13-6 下列程序执行后，变量 `four`、`position` 和 `three` 的值是多少？

```
a=1:5;
fid=fopen('fdat.bin','w'); %以写方式打开文件 fdat.bin
fwrite(fid,a,'int16'); %将 a 的元素以双字节整型写入文件 fdat.bin
fclose(fid);
fid=fopen('fdat.bin','r'); %以读数据方式打开文件 fdat.bin
status=fseek(fid,6,'bof'); %将文件指针从开始位置向尾部移动 6 个字节
four=fread(fid,1,'int16'); %读取第 4 个数据，并移动指针到下一个数据
position=ftell(fid); %ftell 的返回值为 8
status=fseek(fid,-4,'cof'); %将文件指针从当前位置往前移动 4 个字节
three=fread(fid,1,'int16'); %读取第 3 个数据
status=fclose(fid);
```

例 13-7 创建对 MAT 文件进行操作的 C 程序。

程序如下：

```
#include <stdio.h>
#include <stdlib.h> /* 定义 EXIT_FAILURE,EXIT_SUCCESS */
#include <mat.h>
```

```

int main()
{
    MATFile *pmat; /* 定义 MAT 文件指针*/
    mxArray *pa1,*pa2,*pa3;
    double data[9]={1.1,2.2,3.3,4.4,5.5,6.6,7.7,8.8,9.9};
    const char *file="c_matfile.mat";
    int status;
    /* 打开一个 MAT 文件, 如果不存在则创建一个 MAT 文件, 如果打开失败, 则返回到系统环境 */
    pmat=matOpen(file,"w");
    if (pmat==NULL) {
        printf("创建失败\n");
        return(EXIT_FAILURE); }
    /* 创建 3 个 mxArray 对象 */
    pa1=mxCreateDoubleScalar(1.234);
    if (pa1==NULL) {
        printf("创建失败\n");
        return(EXIT_FAILURE); }
    pa2=mxCreateDoubleMatrix(3,3,mxREAL);
    if (pa2==NULL) {
        printf("创建失败\n");
        return(EXIT_FAILURE); }
    memcpy((void *) (mxGetPr(pa2)), (void *) data, sizeof(data));
    pa3=mxCreateString("MAT 文件示例");
    if (pa3==NULL) {
        printf("创建失败\n");
        return(EXIT_FAILURE); }
    /* 向 MAT 文件中写数据, 失败则返回到系统环境 */
    status=matPutVariable(pmat,"LocalDouble",pa1);
    if (status!=0) {
        printf("赋值失败\n");
        return(EXIT_FAILURE); }
    status=matPutVariableAsGlobal(pmat,"GlobalDouble",pa2);
    if (status!=0) {
        printf("赋值失败\n");
        return(EXIT_FAILURE); }
    status=matPutVariable(pmat,"LocalString",pa3);
    if (status!=0) {
        printf("赋值失败\n");
        return(EXIT_FAILURE); }
    /* 清除矩阵 */
    mxDestroyArray(pa1);
    mxDestroyArray(pa2);
    mxDestroyArray(pa3);
    /* 关闭 MAT 文件 */
    if (matClose(pmat)!=0) {
        printf("关闭失败\n");
        return(EXIT_FAILURE); }
    printf("完成\n");
    return(EXIT_SUCCESS);
}

```

例 13-8 编写求两个数的最小公倍数的 C 语言 MEX 文件。
程序如下：

```

#include <mex.h>
/* 求最小公倍数子程序 */
void com_multi(double *z,double *x,double *y)
{
    int a,b,c;
    a=*x;
    b=*y;
    c=max(a,b);
    while(c%a!=0||c%b!=0)
        c=c+1;
    *z=c;
}
/* 入口程序 */
void mexFunction(int nlhs,mxArray *plhs[],int nrhs,const mxArray *prhs[])
{
    double *x,*y,*z;
    int m,n,i;
    /* 检查参数数目是否正确 */
    if (nrhs!=2)
        mexErrMsgTxt("Two inputs required.");
    if (nlhs!=1)
        mexErrMsgTxt("One output required.");
    /* 检查输入变量是否为单个的双精度数 */
    for(i=0;i<2;i++)
    {
        m=mxGetM(prhs[i]);
        n=mxGetN(prhs[i]);
        if (!mxIsDouble(prhs[i])||mxIsComplex(prhs[i])||!(m==1&&n==1))
            mexErrMsgTxt("Input must be a noncomplex scalar double.");
    }
    /* 读入输入数据 */
    x=mxGetPr(prhs[0]);
    y=mxGetPr(prhs[1]);
    /* 准备输出空间 */
    plhs[0]=mxCreateDoubleMatrix(m,n,mxREAL);
    z=mxGetPr(plhs[0]);
    /* 计算 */
    com_multi(z,x,y);
}

```

例 13-9 创建一个矩阵，然后将其送到 MATLAB 引擎的工作区中，绘制出结果图。

程序如下：

```

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <engine.h>
int main()
{
    Engine *ep; /* 定义 MATLAB 引擎变量 */
    mxArray *T=NULL,*result=NULL;
    double time[11]={-1,-0.8,-0.6,-0.4,-0.2,0,0.2,0.4,0.6,0.8,1.0};
    /* 启动 MATLAB 引擎。如果在本地启动，那么函数所带的参数字符串为空 */
    /* 如果在网络中启动，则需要提供服务器名，即 engOpen("服务器名") */

```

```

if (! (ep=engOpen("\0"))) {
    printf("不能启动 MATLAB 引擎\n");
    return EXIT_FAILURE; }
/* 向新启动的 MATLAB 工作区放置数据 */
T=mxCreateDoubleMatrix(1,11,mxREAL);
memcpy((void *)mxGetPr(T), (void *)time, sizeof(time));
engPutVariable(ep, "T", T);
/* 执行 MATLAB 命令 */
engEvalString(ep, "D=T.*T");
engEvalString(ep, "plot(T,D);");
/* 从 MATLAB 工作区获取计算结果 */
result=engGetVariable(ep, "D");
printf("按回车继续\n");
fgetc(stdin);
/* 释放内存空间, 关闭引擎 */
mxDestroyArray(T);
engEvalString(ep, "close;");
engClose(ep);
return EXIT_SUCCESS;
}

```

第 14 章 MATLAB 的应用

14.1 电路分析

基于以上状态方程建立函数文件 vdpol.m。

```

function ydot=vdpol(t,y)
ydot(1)=0.1*(1-y(2)^2)*y(1)-y(2);    %μ的值可以任意变化, 此处取 0.1
ydot(2)=y(1);
ydot=ydot';
求解微分方程, 并绘制振荡波形 (t, y) 和相轨迹 (y, dy/dt)。
t0=0;tf=60;                            %确定积分区间
y0=[0;0.25];                            %确定初始条件
[t,y]=ode45(@vdpol,[t0,tf],y0);        %求解微分方程
subplot(2,2,1);plot(t,y(:,2))          %绘制振荡波形
subplot(2,2,2);plot(y(:,2),y(:,1))     %绘制相轨迹

```

14.2 控制系统分析

解法 2: 利用 MATLAB 控制系统工具箱中已经定义的一些 LTI 仿真函数, 编写程序如下:

```

G=tf(4,[1,2,3,4]);
Gc=tf([1,-3],[1,3]);
H=tf(1,[0.01,1]);
G_o=Gc*G;                                %构造开环系统的传递函数
G_c=feedback(G_o,H);                    %构造闭环系统的传递函数
step(G_o)                                %求开环系统的阶跃响应并绘制相应的曲线
axis([0,10,-1.5,0.5]);grid on
figure
step(G_c)                                %求闭环系统的阶跃响应并绘制相应的曲线
axis([0,10,-6,1]);grid on

```


14.3 分形曲线的绘制

首先建立函数文件 koch.m。

```
function y=koch(ax,ay,bx,by,depth)
if depth<1
    plot([ax,bx],[ay,by],'k')
    hold on
else
    cx=ax+(bx-ax)/3;           %计算替换点坐标
    cy=ay+(by-ay)/3;
    dx=(ax+bx)/2+sqrt(3)*(ay-by)/6;
    dy=(ay+by)/2+sqrt(3)*(bx-ax)/6;
    ex=bx-(bx-ax)/3;
    ey=by-(by-ay)/3;
    koch(ax,ay,cx,cy,depth-1) %递归调用
    koch(cx,cy,dx,dy,depth-1)
    koch(dx,dy,ex,ey,depth-1)
    koch(ex,ey,bx,by,depth-1)
end
```

函数编写完成后，在命令行窗口输入以下命令：

```
>> depth=6;
>> koch(20,40,480,40,depth)
>> axis equal
>> axis([0,500,0,200])
```

在程序中 3 次调用 koch 函数，实现三角形 3 条边各自的科赫曲线，形成科赫雪花曲线效果。在命令行窗口输入以下命令，程序运行结果如图 14-10 所示。

```
>> depth=3;
>> koch(180,10,64.5,210,depth)
>> koch(64.5,210,295.5,210,depth)
>> koch(295.5,210,180,10,depth)
>> axis equal
```

14.4 最优化问题求解

证券投资组合问题

假定期望收益率 $r_p=10\%$ ，最优问题求解的程序如下：

首先建立函数文件 ef.m。

```
function f=ef(x)
v=zeros(3,3);
v(1,1)=10;
v(2,2)=1;
```

```

f=x'*v*x;
再建立主程序文件 ex1442.m。
format rat
ra=0.12;
rb=0.08;
rf=0.06;
rp=0.1;
x0=[1,1,1]'/3;
Aeq=[ra,rb,rf;1,1,1];
beq=[rp,1]';
Lb=[0,0,-100]';
options=optimset('LargeScale','off','Display','off');
x=fmincon('ef',x0,[],[],Aeq,beq,Lb,[],[],options)
format short

```

生产决策问题

程序如下：

```

f=[-7;-5];
A=[3,2;4,6;0,7];
b=[90;200;210];
lb=zeros(2,1);
[x,fval]=linprog(f,A,b,[],[],lb)

```

14.5 工程结构分析

静不定问题

设长度 $L_3=1000\text{mm}$ ，力 $F=25000\text{N}$ ，弹性模量 $E=205000\text{N/mm}^2$ ，横截面积 $A=100\text{mm}^2$ 。设 $AB=BC$ ， $AC=L_3/2$ ，程序如下：

```

alpha1=atan(1/2);
alpha2=atan(1/4);
L3=1000;
F=25000;
E=205000;
A=100;
L1=L3/cos(alpha1);L2=L3/cos(alpha2);
C=[2*cos(alpha1),2*cos(alpha2),1,0;L1/(E*A),0,0,-cos(alpha1);...
    0,L2/(E*A),0,-cos(alpha2);0,0,L3/(E*A),-1];
B=[F;0;0;0];
X=C\B

```

直梁的自由振动频率

将图 14-13 中的梁划分为 4 段，共 5 个结点 10 个自由度（其中 1 结点的竖向位移与 5 结点的竖向及转动位移受约束，即矩阵为 7×7 矩阵），程序如下：

```

E=2e11;
I=1.03e-7;
A=1.14e-3;
rho=7860;
L=1;
X=[12/L/L/L,-6/L/L,-12/L/L/L,-6/L/L];
Ke=E*I*[X;-6/L/L,4/L,6/L/L,2/L;-X;-6/L/L,2/L,6/L/L,4/L];
K=Ke;
KK=Ke;
for i=2:4 %生成整体刚度矩阵
    K=[K,zeros(2*i,1),zeros(2*i,1)];
    K=[K;zeros(1,2*i+2);zeros(1,2*i+2)];
    KK=[zeros(2*i,1),zeros(2*i,1),KK];
    KK=[zeros(1,2*i+2);zeros(1,2*i+2);KK];
    K=K+KK;
end
Me=[156,-22*L,54,13*L;-22*L,4*L*L,-13*L,-3*L*L;54,...
    -13*L,156,22*L;13*L,-3*L*L,22*L,4*L*L];
Me=rho*A*L/420*Me;
M=Me;MM=Me;
for i=2:4 %生成整体质量矩阵
    M=[M,zeros(2*i,1),zeros(2*i,1)];
    M=[M;zeros(1,2*i+2);zeros(1,2*i+2)];
    MM=[zeros(2*i,1),zeros(2*i,1),MM];
    MM=[zeros(1,2*i+2);zeros(1,2*i+2);MM];
    M=M+MM;
end
%处理约束条件
K(1,:)=[];K(:,1)=[]; K(8,:)=[];K(:,8)=[]; K(8,:)=[];K(:,8)=[];
M(1,:)=[];M(:,1)=[]; M(8,:)=[];M(:,8)=[];M(8,:)=[];M(:,8)=[];
AA=inv(M)*K;
W=eig(AA); %求特征值
W=sqrt(W')

```