# 第3章 MATLAB 绘图

在 MATLAB 语言中,学会使用程序画图十分重要。图形的绘制不仅在整个通信工程领域具有非常重要的意义,在自然科学和管理科学几乎所有的领域都占有非常重要的意义,因此通过图形绘制可以很清晰地表示想要表达的观点,也可以很方便地说明相关实验的效果。MATLAB 包含很多用来显示各种图形的函数,提供了丰富的修饰方法,可以绘制出更加美观、精确满足用户需求的各种图形。因此,本章将对 MATLAB 程序中的二维绘图和 三维绘图进行详细的介绍,以期能够为后续的学习奠定坚实的基础。本章不仅介绍图形的 绘制,还介绍如何设置曲线样式、图形标注与坐标控制、图形的可视化编辑和图形窗口的 分割,这是利用图像来解决实际问题的基础,在医疗和通信工程领域非常重要。

# 3.1 二维数据曲线图

MATLAB 不仅具有强大的数值计算能力,同时具备非常便利的绘图功能,尤其擅长将数据、函数等各种科学运算结构可视化,从而将枯燥乏味的数字变成赏心悦目的图片,而二维图形绘制是 MATLAB 语言处理图形的基础。本节将较为全面地介绍二维绘图函数的种类和格式,以及如何设置线条属性和标注图形的方法。

## 3.1.1 绘制单根二维曲线

#### 1. Plot()函数

在 MATLAB 中,绘制图像时使用最频繁的函数是 plot()函数,一般称之为画图函数。 Plot()函数的基本调用格式如下。

- □ plot(y): 对于只含有一个输入参数的 plot()函数,如果输入参数 y 为向量,则绘图时的横坐标从 1 开始到向量 y 的长度结束,纵坐标为向量 y 中的元素;如果输入参数 y 为矩阵,则绘图时,按列绘制每列元素的曲线,每条曲线的纵坐标为该列矩阵的元素值,横坐标从 1 开始,与矩阵元素的行坐标一一对应,绘制曲线的条数为输入矩阵 y 的列数,多条曲线在默认的情况下,一般通过不同的颜色进行区分。
- □ plot(*x*, *y*): 对于含有两个输入参数的 plot()函数,如果 *x* 是向量,*y* 也为向量,则向量*x*,*y* 的长度必须相同,在绘图时,将以 *x* 为横坐标,以 *y* 为纵坐标进行画图;如果 *x* 为向量时,*y* 为矩阵,则矩阵 *y* 必须有一维的大小与 *x* 向量的长度相同,在进行绘图的时候,将以 *x* 为横坐标,绘制出不同颜色的曲线,曲线的条数与矩阵 *y* 的另一维的大小相同;如果 *x* 和 *y* 都是矩阵,则 *x* 与 *y* 的维数相同,在进行绘图

时,分别以矩阵 x 和 y 对应的列元素为横、纵坐标,绘制曲线,曲线的条数与 x 或 y 矩阵的列数相同,并且不同的曲线以不同的颜色进行区分。

【例 3-1】 在 0 ≤ t ≤ 2pi 区间内, 绘制曲线 y=3cost+1。MATLAB 代码如下:

```
t=0:pi/100:2*pi;
y=3*cos(t)+1
figure
plot(y)
```

MATLAB运行结果如图 3.1 所示。

在例 3-1 中,先设定好 x 的范围, y 是给定的函数值,由函数表达式 y=3\*cos(t)+1 可以 计算出绘图的数据 y,从而可以使用 plot(y)绘制出想要的图形。下面再举一个例子,不仅 将 x 的值用函数表达式表示,也将 y 的值同样用表达式表达。

【例 3-2】 已知 *t* 的范围是 0~2π, *x*=2\**t*; *y*=*t*\*sin(*t*)\*sin(*t*),请画出关于 *x*、*y* 的曲线图。 MATLAB 代码如下:

```
t=0:0.1:2*pi;
x=2 * t;
y=t*sin(t)*sin(t);
plot(x, y);
```

MATLAB运行结果如图 3.2 所示。



【例 3-3】已知 *t* 的范围是 0~10, *y*=rand(4, 10), 请画出关于 *x*、*y* 的曲线图。MATLAB 代码如下:

```
t =0:1:10;
y = rand(4, 10);
y1 = rand(10, 4);
% y为矩阵
figure
plot(y)
% 绘制 x 和 y
figure
plot(t, y);
figure
plot(t, y1)
```

MATLAB运行结果如图 3.3 所示。

• 70 •



【**例 3-4**】 已知 *x*=rand(4, 4), *y*=randn(4,4), 请画出关于 *x*、*y* 的曲线图。MATLAB 代 码如下:

x = rand(4, 4); y = randn(4, 4); figure plot(x,y, 'LineWidth', 3)

MATLAB运行结果如图 3.4 所示。





关于 MATLAB 单根二维曲线,大致分为以下 3 种类型。

(1)确定好一个坐标轴的范围,输入另一个坐标轴变量的表达式。

(2)要画出的图像和设置的基本变量本身是间接关系,都是关于已知范围变量的函数,只要掌握这两种方法即可。在进行绘图时,如果仅是简单绘制一幅图像,在 MATLAB 中可以省略打开图的程序,也就是 figure。如果 figure 在程序的最开始,这时如果是在命令窗口,就会弹出一个空白框。当然也可以省略它,因为系统检测 plot()函数也会自动打开画图的界面。

(3)最简单的调用格式。plot()函数最简单的调用格式是只包含一个输入参数,代码如下:

plot(x)

在这种情况下,当*x*是实向量时,以该向量元素的下标为横坐标,元素值为纵坐标画出一条连续曲线,这实际上是绘制折线图。

#### 2. line()函数

在 MATLABB 中,用 line()函数画出直线型的图,其调用格式如下。

□ line(x, y): 以向量x 和y的元素为坐标点绘制直线。

□ line(x, y, z): 以向量  $x, y \neq z$  的元素为坐标点绘制三维直线。

【例 3-5】 已知 x=[0:pi]; y=2×sin(4×x), 试用 line()函数绘图。MATLAB 代码如下:

```
x=0:1/pi:pi;
y=2*sin(4*x);
% 绘制直线
figure
line(x, y)
```

MATLAB运行结果如图 3.5 所示。



【例 3-6】 已知 *x*=[0:pi]; *y*=[0:pi]; *z*=2×sin(4.×*x*.×*y*),试用 line()函数绘图。MATLAB 代码如下:

• 72 •

```
x=[0:pi];
y=[0:pi];
z=2*sin(4.*x.*y);
% 绘制直线
figure
line(x, y, z)
```

MATLAB运行结果如图 3.6 所示。

## 3. 极坐标下的polar()函数

在极坐标下,MATLAB利用 polar()函数绘制曲线,其调用方式如下。

- □ polar(theta, rho): 在极坐标系中, 以角度 theta 和半径 rho 进行绘图。
- polar(theta, rho, s): 在极坐标系中,以线形
   s 绘制角度为 theta 和半径为 rho 的曲线。

【例 3-7】 利用 polar()函数绘制轮胎图。 MATLAB 代码如下:

```
%设定角度
theta=0:45;
%设定对应角度的半径
rho=ones(1,length(theta));
%绘图
```

figure

polar(theta, rho)



图 3.6 绘制三维直线

MATLAB运行结果如图 3.7 所示。

【例 3-8】 利用 polar()函数在极坐标下进行绘图。MATLAB 代码如下:

```
% 角度
t = 0:.01:2*pi;
%设定对应角度的半径
r = sin(2*t).*cos(2*t);
% 绘图
figure
polar(t, r, '--r')
MATLAB 运行结果如图 3.8 所示。
```

图 3.7 轮胎图



图 3.8 polar()函数运行窗口

## 【例 3-9】 利用 polar()函数绘制笛卡尔心形图。MATLAB 代码如下:

```
%设定角度
t = -2*pi:.001:2*pi;
%设定对应角度的半径
r = (1-sin(t));
%绘图
figure
polar(t,r, 'r')
```

MATLAB运行结果如图 3.9 所示。

# 3.1.2 绘制多根二维曲线

在 MATLAB 中绘制多个二维曲线主要有以下 3 种方式。

- (1) 通过对多维数组进行图形绘制。
- (2) 通过特殊的 MATLAB 函数进行图像绘制。
- (3) 通过对多个绘图函数叠加得到多根二维曲线图。

首先来看第一种方法,这涉及多维数组的建立,在 MATLAB 中,多维数组的建立与矩阵的建立方法类似,而且

MATLAB 还为多维数组的建立提供了专门的函数,常用的多维数组建立的方式有下面4种。

- (1)利用下标建立多维数组。
- (2)利用 MATLAB 函数产生多维数组。
- (3)利用 cat()函数建立多维数组。

(4) 用户自己编写 M 文件产生多维数组,即用户自己编写代码产生多维数组。

本书在第6章对这4种方法进行了逐一介绍,这里主要是要求读者会使用这些方法产 生多维数组,然后利用这些多维数组进行绘图即可。

## 1. plot()函数的输入参数是矩阵形式

将 3.1.1 节 plot()函数的参数为矩阵形式的情况进行总结,可以得到如下情况。

- □ 若*x*是向量,*y*是矩阵,当*y*矩阵中的一维与向量*x*同维时,能绘出多条不同颜色的曲线,且当曲线条数与*y*矩阵的另一维相同时,*x*可作为多条曲线的共同横坐标。
- □ 若*x、y*均为矩阵且维数相同时,以*x*和*y*对应的列元素分别作为横、纵坐标绘制 曲线,此时,曲线条数与矩阵列数相同。
- □ 当 plot()函数中只有一个输入参数时,若该输入参数为实矩阵,按列绘制曲线,此 时曲线条数与输入矩阵列数相同。

当输入参数为复数矩阵时,绘制曲线时需按列分别将元素实部、虚部为横、纵坐标绘制曲线。大部分的情况在 3.1.1 节都已经进行了举例说明,本节仅举一个例子进行说明。

【例 3-10】 以单位阵 A 和全 1 阵 B 为输入进行图形绘制。MATLAB 代码如下:

% 生成单位阵 A = eye(4); % 生成全1阵 B = ones(4); %绘图





图 3.9 笛卡尔心形图

figure plot(A,B)

MATLAB运行结果如图 3.10 所示。

#### 2. 含多个输入参数的plot()函数

含多个输入参数的 plot()函数的调用格式如下。

plot(x1,y1,x2,y2,...,xn,yn): 当输入参数是向量形式时,x1 与 y1,x2 与 y2, ..., xn 与 yn 组成 n 组向量对(每组向量对的长度可以不同),可以绘制出 n 条曲线; 当输入参数是矩 阵形式时,矩阵 x1 与 y1, x2 与 y2, ..., xn 与 yn 对应列元素为横、纵坐标分别绘制曲线,此时曲线条数与矩阵列数相同。

【例 3-11】 尝试绘制正余弦双曲线图。MATLAB 代码如下:

```
x=-pi:0.01:pi;
% 产生数据
y=sin(2*x+pi/3);
y1=cos(3*x+pi/3);
% 绘图
plot(x,y,x,y1, 'LineWidth',3)
```

MATLAB运行结果如图 3.11 所示。





```
t1 = -pi:0.01:pi;
t2 = 0:0.01:pi/2;
% 产生数据
y1 = sin(2*t1+pi/3);
y2 = sin(3*t2+pi/3);
% 绘图
plot(t1, y1, t2, y2,'k-', 'LineWidth', 3)
```

## MATLAB运行结果如图 3.12 所示。

上面介绍的是通过对多维数组进行图形绘制,下面介绍第二种方法,即通过特殊的

• 75 •

MATLAB 函数进行图像绘制,这些函数主要 有 plotyy()函数、contour()函数、clabel()函数、 contourc()函数等。

## 3. 具有两个纵坐标标度的图形

在 MATLAB 中,如果需要绘制出具有不同纵坐标标度的两个图形,可以使用 plotyy() 函数绘图函数,其调用格式如下。

□ plotyy(x1,y1,x2,y2): 其中 x1, y1 对应 一条曲线, x2, y2 对应另一条曲线。 此时,横坐标有相同的标度,而纵坐 标标度不同,左纵坐标对应 x1,y1, 右纵坐标对应 x2,y2。



- □ plotyy(x1,y1,x2,y2, fun): 其中 x1, y1 对应一条曲线, x2, y2 对应另一条曲线。横坐标的标度相同, 纵坐标有两个, 左纵坐标利用函数 fun()函数将 x1 和 y1 数据对绘制成图形, 右纵坐标利用函数 fun()函数将 x2, y2 数据对绘制成图形。其中 fun()函数可以用 MATLAB 中的绘图函数来表示, 例如 plot(), semilogx(), semilogy(), loglog(), stem()等函数,也可以使用自定义的函数来表示,其调用格式一般为 plotyy (X1,Y1,X2,Y2,@loglog) 或 plotyy (X1,Y1,X2,Y2,'loglog')。
- □ plotyy(x1,y1,x2,y2, fun1, fun2): 其中 x1, y1 对应一条曲线, x2, y2 对应另一条曲线。 此时,横坐标有相同的标度,而纵坐标标度不同。左纵坐标下,是使用函数 fun1() 函数将 x1 和 y1 数据绘成曲线,右纵坐标下,是使用函数 fun2()将 x2, y2 数据绘 成曲线。其中 fun1()函数和 fun2()函数可以用 MATLAB 中的绘图函数来表示,例 如 plot(), semilogx(), semilogy(), loglog(), stem()等函数,也可以使用自定义的函数 来表示。

【例 3-13】用不同标度在同一坐标内绘制曲线 y1=0.2e-0.5xcos(4πx) 和 y2=2e-0.5xcos(πx)。 MATLAB 代码如下:

```
x=0:pi/100:2*pi;
% 生成曲线
y1=0.2*exp(-0.5*x).*cos(4*pi*x);
y2=2*exp(-0.5*x).*cos(pi*x);
% 绘图
figure
plotyy(x,y1,x,y2);
plot(x, y1, 'k-', x, y2, 'k-', 'LineWidth', 3)
```

MATLAB运行结果如图 3.13 所示。

【例 3-14】 一个简单的调用 plotyy(x1,y1,x2,y2, fun1, fun2)函数的例子。MATLAB 代码 如下:

```
% 生成横轴数据
x1=1:0.1:100;
x2=x1;
% 生成纵轴数据
```

• 76 •



#### 4. Contour()函数

Contour()函数主要用来绘制曲面的等高线图,其调用方法如下。

- □ contour(*z*): 把矩阵 *z* 中的值作为一个二维函数的值,等高曲线在一个平面内,平面的高度 *v* 由 MATLAB 自动选取,绘制等高线。
- □ contour(*x*,*y*,*z*): (*x*,*y*)是平面 *z*=0 上点的坐标矩阵, *z* 为相应点的高度值矩阵, 绘制 等高线。
- **□** contour(z,n): 画出矩阵z的n条等高线。
- □ contour(x,y,z,n): 画出矩阵 z 的 n 条等高线。
- **□** contour(z,v): 在指定的高度v上画出等高线。

【例 3-15】 用 contour()函数画 peaks()函数。MATLAB 代码如下:

```
z = peaks(40);
figure
contour(z,'k')
```

MATLAB运行结果如图 3.15 所示。

【例 3-16】 用 contour()函数画 sinxcosy 的等高线。MATLAB 代码如下:

```
x = 0:0.1:pi;
y = 0:0.1:pi;
% 生成坐标系
[xx,yy] = meshgrid(y,x);
% 生成曲面
z = sin(xx).*cos(yy);
```

• 77 •

% 绘制等高线 figure contour(xx,yy,z,'k');



MATLAB运行结果如图 3.16 所示。



图 3.16 绘制 sinxcosy 的等高线

#### 5. Clabel()函数

Clabel()函数主要的功能是在二维等高线图中添加高度标签。在下列形式中,若有 h 出现,则会对标签进行恰当的旋转,否则标签会竖直放置,且在恰当的位置显示个一个+号。该函数的调用格式如下。

- □ clabel(*C*, *h*): 把标签旋转到恰当的角度,再插入到等高线中。只有等高线之间有足够的空间时才加入,当然这决定于等高线的尺度。
- □ clabel(*C*, *h*, *v*): 在指定的高度 *v* 上显示标签 *h*, 当然要对标签做恰当的处理。
- □ clabel(*C*, *h*, 'manual'):手动设置标签。用户用鼠标左键或 Space 键在最接近指定的 位置上放置标签,用键盘上的 Enter 键结束该操作,当然要对标签做恰当的处理。
- □ clabel(C): 在从命令 contour 生成的等高线结构 c 的位置上添加标签。此时标签的

放置位置是随机的。

【例 3-17】 对例 3-16 中的等高线加上适当的标准。MATLAB 代码如下:

```
x = 0:0.1:pi;
y = 0:0.1:pi;
% 生成坐标系
[xx,yy] = meshgrid(y,x);
% 生成曲面
z = sin(xx).*cos(yy);
% 绘图,加标注
figure
[C, h] = contour(xx,yy,z);
clabel(C, h);
```

MATLAB运行结果如图 3.17 所示。

#### 6. Contourc()函数

Contourc()函数计算等高线矩阵 *c*,该矩阵 可用于命令 contour、contour3 和 contourf 等, 其调用格式如下。

> □ C = contourc(z): 从矩阵 z 中计算等高 矩阵,其中 z 的维数至少为 2×2 阶, 等高线为矩阵 z 中数值相等的单元。 等高线的数目和相应的高度值是自动 选择的。



图 3.17 加标注的等高线

- **口** C = contourc(z, n): 在矩阵 z 中计算出 n 个高度的等高线。
- □ C = contour(*z*, *v*): 在矩阵 *z* 中计算给定高度向量 *v* 上的等高线,当然向量 *v* 的维数 决定了等高线的数目。若只要计算一条高度为 *a* 的等高线,输入: contourc(*z*,[a,a])。
- □ C = contourc(x, y, z) 在矩阵 z 中, 参量 x, y 确定的坐标轴范围内计算等高线。
- □ C = contourc(x, y, z, n) 从矩阵 z + , 参量 x = y 确定的坐标范围内画出 n 条等高线。
- □ C = contourc(x, y, z, v) 从矩阵 z 中, 参量 x 与 y 确定的坐标范围内, 画在 v 指定的 高度上指定的等高线。
- 【例 3-18】 计算矩阵 A 等高线为 2 和 3 的坐标位置。MATLAB 代码如下:

A = [1 2 3 4;2 3 4 5;3 4 5 6]; contourc(A,[2 3])

MATLAB 运行结果如下:

ans = 2 2 1 3 3 2 2 1 2 1 2 3 2 4 2 1

这里A的[23]有两层意思:

□ 寻找矩阵 A 中值为 2 和 3 的等值线(坐标)。

□ 值为2和3自然就有两层等值线。

分析 MATLAB 的运行结果,由于 MATLAB 中矩阵的存储按列进行,因此得到的数据 要竖着看,即第一列为(2,2),上面的2代表的数值为2的等值线,下面的2为等值线对

• 79 •

应的坐标个数。接下来的两(对应2)列自然就是数值为2的坐标了,即(2,1)和(1,2)。

同理,接下来的列(3,3),第一个代表数值为3的等值线,第二个代表的个数为3个。 自然地,后面的3个为相应的坐标。

△注意: "坐标"就是说的矩阵的行列,例(2,1)即第二行第三列。

最后介绍在 MATLAB 中通过对多个绘图函数叠加的方式得到多根二维曲线图,该种 方式是 MATLAB 中最常用的多维曲线绘制方法。下面进行举例说明。

【例 3-19】 设矩阵 *A*=randn(5, 5),在 MATLAB 中将 *A* 的每一列绘制出一条曲线。 MATLAB 代码如下:

```
% 生成矩阵
A=randn(5, 5);
% 绘图
figure
plot(A(1,:),'-*');
hold on
plot(A(2,:),'-v');
hold on
plot(A(3,:),'-+');
hold on
plot(A(4,:),'-.');
hold on
plot(A(5,:),'-');
```

MATLAB运行结果如图 3.18 所示。



注意,在画多条曲线时,需要加上 hold on 命令,相对的也应有 hold off。hold on 使当前轴及图形保持而不被刷新,准备接受新的绘图命令后进行绘制; hold off 使当前轴及图形不再具备被刷新的性质。所以我们在 plot(*A*(1,:),'-\*');后加上了 hold on,这样多条曲线就可以被同时显示了。关于线型和颜色,会在后面进行详细讲解。

## 3.1.3 设置曲线样式

在使用 plot()函数的时候, MATLAB 提供了一些绘图选项, 用于确定所绘曲线的线型、颜色和数据点标记符号, 它们可以组合使用。例如, "b-."表示蓝色点划线, "y:d"表示

• 80 •

黄色虚线并用菱形符标记数据点。当选项省略时,MATLAB规定,线型一律用实线,颜色将根据曲线的先后顺序依次进行轮换。所以在MATLAB中要设置曲线样式,只需要在plot()函数中加绘图选项即可,其调用格式如下。

- plot(x1, y1, LineSpec):用于对图形的线型、数据点的样式、颜色进行控制,LineSpec 为控制线型、点型、颜色的字符串。在MATLAB中,线型、点型和颜色的控制符 分别如表 3.1、表 3.2 和表 3.3 所示。在MATLAB中还可以将多个控制字符连为一 个字符串以实现对图形样式的控制,并且线型、点型和颜色控制符的位置不影响 最终的绘图结果。当然,也可以默认设置任意的一个或多个控制符。例如,"b-." 表示蓝色点画线,"r--p"则表示红色虚线并用五角星标记数据点。如果使用 plot() 函数的数据参数为矩阵数据时,一般不设置图形的样式,这是因为 plot()函数数据 参数为矩阵时,表示 MATLAB将要绘制多条曲线,在设置了图形样式的情况下, 各条曲线的样式将变成统一的,因而变得难以区分。
- □ plot(*x*1, *y*1, 'propertyName', 'propertyName'): 在进行绘图时对绘制的图形属性进一步设置。其中 propertyName 为需要设置的曲线的属性名称,而 propertyValue 为要设置属性的值。属性和属性值一般是成对出现的,且不同属性之间排列的先后顺序不影响最终绘制的图形样式。如表 3.1~表 3.4 列出了一些常用的需要设置的图形属性。

注意本节的内容非常重要,在进行学术论文的写作过程中,通常应该在实验结果和实 验分析中用不同的颜色来表示不同的曲线,更甚一步,应该考虑论文出版后打印输出的问 题,论文很有可能是灰度的,因此这时对实验结果的线型要求更重要,也就是说,必须能 够在灰度的情况下,通过不同的线型区分出不同的实验结果曲线。

控制符	线条样式
-	实线
:	点线
	点画线
	虚线

表 3.1 plot()函数线型的控制符

控制符	数据点样式
	点号
+	十字符
*	* -
X	叉号
0	空心圆
s 或者'square'	正方形
p 或者'pentagram'	五角星
d 或者'diamond'	菱形
h 或者'hexagram'	六角形
^	上三角形
v	下三角形
<	左三角
>	右三角

表 3.2 plot()函数点型的控制符

• 81 •

控制符	数据点样式
r	红色
m	粉色
g	绿色
c	青色
b	蓝色
W	白色
У	黄色
k	黑色

## 表 3.3 plot()函数颜色的控制符

#### 表 3.4 plot()函数常用的属性

属性名称	属性的描述
LineWidth	设置线的宽度
MarkerSize	设置标记点的大小
MarkerEdgeColor	设置标记点的边缘颜色
MarkerFaceColor	设置标记点的填充颜色

【例 3-20】 在同一坐标内,分别用不同线型和颜色绘制曲线 y1=0.2e-0.5xcos(4πx) 和 y2=2e-0.5xcos(πx)。MATLAB 代码如下:

```
% 横坐标轴
x = linspace(0, 2*pi, 100);
% 生成数据点, 纵坐标轴
y1 = 0.2 * exp(-0.5 * x).* cos(4 * pi * x);
y2 = 2 * exp(-0.5 * x) .* cos(pi * x);
% 绘图
figure
plot(x, y1, 'r-.', x, y2, 'k:','LineWidth',2);
```

MATLAB运行结果如图 3.19 所示。



观察图 3.19 可知,当图形为灰度图形时,两条曲线的区分还是有些困难,为此加入标记来突出两条曲线的不同。

• 82 •

【例 3-21】 对例 3-20 中的曲线 y1 中的数据点使用圆圈标记,对曲线 y2 使用十字符号标记。MATLAB 代码如下:

```
% 横坐标轴
x = linspace(0, 2*pi, 100);
% 生成数据点, 纵坐标轴
y1 = 0.2 * exp(-0.5 * x).* cos(4 * pi * x);
y2 = 2 * exp(-0.5 * x) .* cos(pi * x);
% 绘图
figure
plot(x, y1, 'r-.o', x, y2, 'k:+');
```

MATLAB运行结果如图 3.20 所示。

在有些情况下,需要在图形绘制中突出两条曲线交点的地方,以分析各实验过程的不 同趋势,从而得到更客观的实验分析,下面举例说明。

【例 3-22】 加密例 3-20 种两曲线的数据点,并利用五角星标记例 3-20 中两曲线的交 叉点。MATLAB 代码如下:

```
% 横坐标轴
x = linspace(0, 2*pi, 1000);
% 生成数据点, 纵坐标轴
y1 = 0.2 * exp(-0.5 * x).* cos(4 * pi * x);
y2 = 2 * exp(-0.5 * x) .* cos(pi * x);
% 查找 y1 与 y2 相等点(近似相等)的下标
k = find( abs(y1-y2) < 1e-2 );
%取 y1 与 y2 相等点的 x 坐标
x1 = x(k);
% 求 y1 与 y2 值相等点的 y 坐标
y3 = 0.2 * exp(-0.5 * x1) .* cos(4 * pi * x1);
% 绘图
figure
plot(x, y1, 'r-.', x, y2, 'k:', x1, y3, 'bp','LineWidth',2);
```

MATLAB运行结果如图 3.21 所示。



【例 3-23】 设置正弦曲线的线宽为 3,设置上三角形进行数据点的标记,并设置标记 点边缘为黑色,设置标记点填充颜色为红色,设置标记点的尺寸为 10,则 MATLAB 代码 如下:

MATLAB运行结果如图 3.22 所示。

## 3.1.4 图形标注

在完成一个图形的绘制后,在后续的处理中,可以通过添加图形标注的方式让读者更 好地理解图形的意义,从而进一步完善图形的含义,更好地说明图形中数据的含义。 MATLAB 中图形标注包括图形标题、标轴、文本和图例等标注。

#### 1. 图形的标题标注

在 MATLAB 中, title()函数用于给当前绘制的图形坐标轴的正上方添加标题。具体调用格式如下。

□ title('string'): 在图形窗口中为当前绘制的图形添加标题 string。

□ title('string', 'propertyName', propertyValue, …): 在图形窗口中为当前绘制的图形添加标题,并对标题的属性进行设置。

【例 3-24】 绘制正弦曲线,设置图形的标题为 y=sin(x)。MATLAB 代码如下:

```
% 横坐标轴
x = linspace(0, 2*pi, 50);
% 生成数据点,纵坐标轴
y = 2 * sin(pi * x);
% 绘图
figure
% 设置线宽为 3
plot(x, y, 'k--^', 'LineWidth', 3)
% 设置标题
title('y=sin(x)的曲线')
```

MATLAB运行结果如图 3.23 所示。

## 2. 图形的坐标轴标注

在 MATLAB 中, xlabel()函数和 ylabel()函数分别用于对当前绘制的图形的 x 轴和 y 轴 坐标轴进行标注,其调用格式。

• 84 •

- □ xlabel('string'):为当前绘制的图形的 x 轴添加标注 string。
- □ xlabel ('string', 'propertyName', propertyValue, ...): 为当前绘制的图形的 x 轴添加标 注 string, 并对标注的属性进行设置。



- □ ylabel('string'):为当前绘制的图形的 y 轴添加标注 string。
- □ ylabel ('string', 'propertyName', propertyValue, ...): 为当前绘制的图形的 *y* 轴添加标 注 string, 并对标注的属性进行设置。

【例 3-25】 绘制正弦曲线,设置图形的标题为 y=sin(x),设置 x 轴的标注为 x,设置 y 轴的标注为 sin(x)。MATLAB 代码如下:

```
% 横坐标轴
x = linspace(0, 2*pi, 50);
% 生成数据点, 纵坐标轴
y = 2 * sin(pi * x);
% 绘图
figure
% 设置线宽为 3
plot(x, y, 'k--^', 'LineWidth', 3)
% 设置标题
title('y=sin(x)的曲线')
% 设置 x 轴的标注
xlabel('x')
% 设置 y 轴的标注
ylabel('sin(x)')
```

MATLAB运行结果如图 3.24 所示。

3. 图形的文本标注

在 MATLAB 中, text()函数和 gtext()函数都可以用于对当前绘制的图形进行文本标注, 其中 text()函数需要设置文本标注的位置,而 gtext()函数可用于交互式的文本标注,即在函 数运行后由用户在当前绘制的图形窗口中选择标注的位置。两个函数的调用格式如下。

- **□** text(x, y, 'string'): 为当前绘制的图形在坐标(x, y)处添加标注 string。
- □ gtext('string'): 该函数可以让用户交互式地在当前绘制的图形上标注字符串 string, 函数执行后,当前图形中会出现+字型交叉线让用户选择待标注的位置。

• 85 •



【例 3-26】 绘制正弦和余弦曲线,设置图形的标题、*x* 轴和 *y* 轴的标注,设置曲线标 准。MATLAB 代码如下:

```
% 橫轴
x=0:pi/50:2*pi;
% 曲线数据
y1=sin(x);
y2=cos(x);
% 绘图
figure
plot(x, y1, 'k-', x, y2, 'k-.')
% 文本标注
text(pi, 0.05, '\leftarrow sin(\alpha)')
text(pi/4-0.05, 0.05, 'cos(\alpha)')
text(pi/4-0.05, 0.05, 'cos(\alpha))')
% 标题标注
title('sin(\alpha) and cos(\alpha)')
% 坐标轴标注
xlabel('\alpha')
ylabel('sin(\alpha) and cos(\alpha)')
```

```
MATLAB运行结果如图 3.25 所示。
```



• 86 •

会注意:在标准中的字符串要使用转义字符,另外本例子也可以使用 gtext()函数进行标注,gtext()函数和 text()函数的使用方法类似,只是可以利用鼠标交互地放置文本标注。

#### 4. 图形的图例标注

在 MATLAB 中, legend()函数可以用于对当前绘制的图形进行图例标注,图例主要用 来设置不同图形颜色、线型等所代表的数据的含义,在 MATLAB 中几乎所有的图形都可 以生成图例。该函数极大地方便了数据曲线的说明,函数调用格式如下。

- □ legned('str1', 'str2', 'str3', ...): 为当前绘制的图形中的各部分数据添加图列标注,字符串 str1、str2、str3...按照数据在绘图中显示的顺序依次标注图形中各部分数据的 图例。
- □ legned ('str1', 'str2', pos): 在默认的情况下,图形的图例标注很有可能把正在绘制的 图形中某一部分遮挡,为了解决这一问题,MATLAB 提供了这种调用格式,pos 参数用于设置添加的图例位置,其取值范围为-1~4之间的整数。pos 取-1 时,表 示图例标注在图形窗口的右边;pos 取 0 时,表示图例标注在图形窗口之内,并且 尽量不与图形重叠;pos 取 1 时,表示图例标注在图形窗口的右上角;pos 取 2 时, 表示图例标注在图形窗口的左上角;pos 取 3 时,表示图例标注在图形窗口的左下 角;pos 取 4 时,表示图例标注在图形窗口的右下角。图例标注还可以通过鼠标进 行直接移动,从而实现交互式地放置图例标注。

【例 3-27】 绘制抛物线和三次幂曲线,设置图形的标题、*x* 轴和 *y* 轴的标注,设置曲 线文本标准、图例标注。MATLAB 代码如下:

```
8 横轴
x = -2:.1:2;
% 曲线数据
y1 = x.^{2};
y^2 = x.^3;
8 绘图
figure
plot(x,y1, 'r-', x, y2, 'b:')
% 标题标注
title('y=x^2 和 y=x^3 曲线')
% 坐标轴标注
xlabel('x')
ylabel('y')
% 文本标注
gtext('y = x^2', 'FontName', 'Times New Roman', 'FontSize', 16)
gtext('y = x^3', 'FontName', 'Times New Roman', 'FontSize', 16)
% 图例标注
legend('ity=x^2', 'ity=x^3', 1)
```

MATLAB运行结果如图 3.26 所示。

## 3.1.5 坐标控制

在进行图形绘制时,一般 MATLAB 会 根据所要绘制的数据自动生成坐标轴,但是 默认状态下的坐标轴范围和精度往往不能 达到用户的需求,因此,在设计程序时需要 根据具体的应用进行坐标轴的范围、精度、 显示方式等方面的设置,这样才能更好地阐 述相应的图形,更好地说明所应用到的问 题。MATLAB 提供了对坐标轴的控制函数, 用于根据用户需要编辑完善坐标轴。



#### 1. 坐标轴范围的设置

在 MATLAB 中,坐标轴调整的函数为 axis()函数,该函数可以有效调整坐标轴的范围 和精度以及显示方式,其调用格式如下。

- □ axis([*x*min *x*max *y*min *y*max]):设置坐标轴的范围,指定当前坐标轴 *x* 轴和 *y* 轴的范围,其中 *x*min 为 *x* 轴下界,而 *x*max 为 *x* 轴的上界,而 *y*min 和 *y*max 分别为 *y* 轴的下界和上界。
- □ axis([*xmin xmax ymin ymax zmin zmax*]): 设置 *x*、 *y* 和 *z* 坐标轴的范围,其中 *xmin* 和 *xmax* 为 *x* 轴下界和上界, *ymin* 和 *ymax* 分别为 *y* 轴的下界和上界, *zmin* 和 *zmax* 分别为 *z* 轴的下界和上界。
- □ xlim([*x*min *x*max]): 仅设置 *x* 轴的范围。
- □ ylim([ymin ymax]): 仅设置 y 轴的范围。

除了上述的坐标轴范围的控制外,axis()函数的功能丰富,常用的调用格式还有如下 方式。

□ axis equal: 横、纵坐标轴采用等长刻度。

□ axis square: 产生正方形坐标系(默认为矩形)。

- □ axis auto: 使用默认设置。
- □ axis off: 取消坐标轴。
- □ axis on:显示坐标轴。
- □ axis tight: 按紧凑方式显示坐标轴范围,即坐标轴范围为绘图数据的范围。

【例 3-28】 绘制抛物线,产生等长刻度的坐标轴。MATLAB 代码如下:

```
% 横轴
x = -2:.1:2;
% 曲线数据
y1 = x.^2;
% 绘图
figure
plot(x,y1, 'r-')
```

• 88 •

```
% 标题标注
title('y=x^2 曲线')
% 产生等长刻度坐标轴
axis equal
```

MATLAB运行结果如图 3.27 所示。

一般在绘制曲线时,系统会根据所采用的数据自动生成适当的坐标轴刻度,但有时需要进行修改,比如在两个曲线对比时,应采用相同的比例因子,以便直观地比较大小。下 面举例说明。

【例 3-29】 绘制两条随机误差曲线,同时绘制出利用 axis()函数修改成相同比例后的 误差曲线。MATLAB 代码如下:

```
° x轴
t = 0:0.01:0.3;
% 生成误差曲线
e1 = 2-4*rand(length(t), 1);
e2 = 2-2*randn(length(t),1);
% 绘图
figure
% 分裂窗口为 2*2 个子窗口
subplot(2,2,1)
plot(t,e1,'k')
title('误差1')
subplot(2,2,3)
plot(t,e2,'k')
title('误差 2')
% 坐标轴调整
subplot(2,2,2)
plot(t,e1,'k')
title('坐标轴调整后的误差1')
axis([0 .3 -4 4])
subplot(2,2,4)
plot(t,e2,'k')
title('坐标轴调整后的误差 2')
axis([0 .3 -4 4])
```

MATLAB运行结果如图 3.28 所示。



• 89 •

同时,对于一些复杂的函数,例如, *y*=cos(tan(πx)),也可以修改坐标轴刻度来更清楚 地观察曲线的局部特性。

【例 3-30】 观察曲线 y=cos(tan(πx))在 x=0.5 附近的图形曲线。MATLAB 代码如下:

```
% x 轴
x = 0:1/3000:1;
% 生成误差曲线
y = cos(tan(pi*x));
% 绘图
figure
% 分裂窗口为 2*1 个子窗口
subplot(2,1,1)
plot(x,y)
title('\itcos(tan(\pix))')
% 坐标轴调整
subplot(2,1,2)
plot(x,y)
axis([0.4 0.6 -1 1]);
title('复杂函数的局部透视')
```

MATLAB运行结果如图 3.29 所示。





```
% 横轴
t = 0:pi/20:2*pi;
% 坐标系
[x,y] = meshgrid(t);
% 高度
z = sin(x).*cos(y);
```

• 90 •

% 绘图 figure plot(t,z) % 坐标轴控制 axis([0 2\*pi -1 1]) % 取消边界 box off % 取消坐标轴 axis off title('无坐标轴和边框图形')

MATLAB运行结果如图 3.30 所示。



图 3.30 无坐标轴与边框的曲线

## 2. 坐标轴字体的设置

在 MATLAB 中,坐标轴字体可以通过对字体的属性进行设置,常用的字体属性如下。

- □ FontName: 字体的类型属性,包括常用的字体类型。
- □ FontSize: 字体的大小属性。
- □ FontUnits: 字体的单位属性。
- □ FontWeight: 字体样式属性,主要包括正常(normal)、粗体(bold)、倾斜(light)、 黑体(demi)等样式。
- □ FontAngle: 字体角度属性,主要包括斜体(italic)。

在 MATLAB 中,需要通过坐标轴句柄使用函数 set()完成坐标轴字体的设置,其中坐标轴对象句柄的获取主要通过 h=gca 语句返回当前坐标系的句柄,使用如下命令进行坐标轴的字体设置。

set(gca, propertyName, propertyValue)

【例 3-32】 设置坐标轴字体为新罗马,字号为 16 号加粗斜体。MATLAB 代码如下:

```
% 横轴
x = -2:.1:2;
% 曲线数据
```

• 91 •

y1 = x.^2; % 绘图 figure plot(x,y1, 'r-') % 标题标注 title('y=x^2 曲线') % 设置坐标轴字体 set(gca,'FontName', 'Tmie New Roma', 'FontSize', 16, 'FontWeight', 'Bold', 'FontAngle', 'italic')

MATLAB运行结果如图 3.31 所示。

#### 3. 坐标轴边框的设置

在 MATLAB 中,坐标轴边框也可以进行设置,常用的属性有颜色属性、线条属性等。 下面以 X 轴为例进行介绍。常用的坐标轴边框属性如下。

□ XDir: 控制 X 轴的方向,默认情况下是正常的,可以进行逆转(reverse)。

**口** XColor: 设置 X 轴的颜色属性。

□ LineStyleOrder: 设置坐标轴边框的线条类型属性。

□ LineWidth: 设置坐标轴边框的线条宽度属性。

【例 3-33】 设置坐标轴 X 轴逆转,坐标轴边框的颜色为红色,线宽为 5 号。MATLAB 代码如下:



MATLAB运行结果如图 3.32 所示。



• 92 •

#### 4. 坐标轴刻度的设置

在 MATLAB 中,坐标轴精度的设置是通过对坐标轴刻度属性进行设置的。对坐标轴 刻度属性设置的调用格式如下。

- □ set(gca,'XTick',[XTickmin:XTickStep:XTickmax]): 设置 X 轴的数字刻度的显示范围 和精度,与当前绘制的图形中的数据范围相对应。
- □ set(gca,'*X*TickLabel',{*X*TickLabelmin:*X*TickLabelStep:*X*TickLabelmax}): 设置 *X*坐标 轴刻度线下的数值显示,默认状态下为当前绘制图形的数据范围和刻度。
- □ set(gca,'*X*TickLabel',{str1, str2, str3,...}): 设置 *X* 坐标轴刻度线处显示的字符为 str1、 str2、str3...。
- □ set(gca, '*Y*Tick',[*Y*Tickmin: *Y*TickStep:*Y*Tickmax]): 设置 *Y*轴的数字刻度的显示范围 和精度,与当前绘制的图形中的数据范围相对应。
- □ set(gca,'*Y*TickLabel',{*Y*TickLabelmin: *Y*TickLabelStep: *Y*TickLabelmax}): 设置 *Y*坐标 轴刻度线下的数值显示,默认状态下为当前绘制图形的数据范围和刻度。
- □ set(gca,'*Y*TickLabel',{str1, str2, str3, ...}): 设置 *Y*坐标轴刻度线处显示的字符为 str1、 str2、str3...。

【例 3-34】 设置坐标轴刻度。MATLAB 代码如下:

```
% 横轴
x = 0:.1:2;
% 曲线数据
y1 = x.^2;
% 绘图
figure
plot(x,y1, 'r-')
% 标题标注
title('y=x^2 曲线')
% 设置坐标刻度
set(gca, 'XTick', [0: .5: 2])
set(gca, 'YTick', [0: .5: 4])
set(gca, 'XTickLabel', {'x=0', 'x=0.5', 'x=1', 'x=1.5', 'x=2'})
```

MATLAB运行结果如图 3.33 所示。

#### 5. 坐标轴位置的设置

在 MATLAB 中,类似图形窗口大小的设置,坐标轴的位置和大小同样也可以设置, 其设置需要调用如下格式。

set(gca, 'Position', [left bottom width height]):设置坐标轴的窗口位于所绘制的图形窗口 的位置为[left bottom],而大小为[width height]。

【例 3-35】 设置坐标轴位置。MATLAB 代码如下:

```
% 横轴
x = 0:.1:2;
% 曲线数据
y1 = x.^2;
% 绘图
figure
plot(x,y1, 'r-')
```

• 93 •

```
% 标题标注
title('y=x^2 曲线')
% 设置坐标轴位置
set(gca, 'Position', [0.1 0.2 0.3 0.4])
```

MATLAB运行结果如图 3.34 所示。



## 3.1.6 边界和网格控制

在 MATLAB 中,给坐标加网格线用 grid 命令来控制。grid on/off 命令控制是否画网格 线,不带参数的 grid 命令可以很方便地在两种状态之间进行切换。

给坐标加边框用 box 命令来控制。box on/off 命令控制是否加边框线,不带参数的 box 命令可以很方便地在两种状态之间进行切换。

【例 3-36】在同一坐标中,可以绘制 3 个同心圆, 增加网格线, 关闭坐标边框。MATLAB 代码如下:

```
% 横轴
t = 0:0.01:2*pi;
x = exp(i * t);
% 曲线数据
y = [x; 2*x; 3*x]';
% 绘图
figure
plot(y)
% 加网格线
grid on;
% 关闭坐标边框
box on;
% 坐标轴采用等刻度
axis equal
```

MATLAB运行结果如图 3.35 所示。



## 3.1.7 图形窗口的分割

在 MATLAB 中, subplot()函数用于进行图形窗口的分割,也就是说在同一个图形窗口 中可以同时显示多个坐标轴的图形,如例 3-29 所示比较直观,便于对比。subplot()函数首 先将图形窗口分成多个子窗口,然后依次在每个子窗口中进行绘图,其调用格式如下。

□ subplot(*m*, *n*, *p*): 该函数将当前图形窗口分成 *m*×*n* 个绘图区,即每行 *n* 个,共*m* 行,区号按行优先编号,且选定第 *p* 个区为当前活动区。在每一个绘图区允许以 不同的坐标系单独绘制图形。

【例 3-37】 利用 subpot()函数绘制多个子图。MATLAB 代码如下:

x=-3:1:9; % 绘图 figure subplot(2, 1, 1) plot(x, 3\*x) subplot(2, 1, 2) plot(x, cos(2\*x))

MATLAB运行结果如图 3.36 所示。



## 3.1.8 图形的可视化编辑

MATLAB 在 6.0 及以上的版本中,在图形窗口中提供了可视化的图形编辑工具,利用 图形窗口菜单栏或工具栏中的有关命令,可以完成对窗口中各种图形对象的编辑处理。

在图形窗口上有菜单栏和工具栏。菜单栏包含 File、Edit、View、Insert、Tools、Desktop、Window 和 Help 共 8 个菜单项,如图 3.37 所示。在菜单栏中,部分菜单项与 MATLAB 主 界页面的菜单项类似,这里主要介绍菜单项中图形窗口所特有的功能。如果能够熟练应用 这些图形的可视化编辑方法,将会为以后的学习和科研工作提供非常有利的帮助。下面对

• 95 •

所涉及的窗口和菜单进行一一介绍。

#### 1. File菜单

File 菜单中包括很多图形处理的子菜单,如图 3.88 所示,包括新建菜单 New、打开菜 单 Open、关闭菜单 Close、保存菜单 Save 和另存菜单 Save As 等。

Figure 1							×		Figure 1
File Edit Vie	w Insert To	ools Deskto	p Window	Help			ъ.	E	le <u>E</u> dit <u>V</u> iew Inse
1 🗃 🖬 🖉	🎍 🗛 🔍	, 🔍 👋 🕲	🐙 🔏 •	3 🛛 🗉	] = 🔲				New
									<u>O</u> pen
1.									<u>C</u> lose
									Save
0.9									Save As
0.8									Generate <u>M</u> -Fi
									Import Data
0.7									Save <u>W</u> orkspa
0.6									Preferences
0.5									Export Setup
0.5									Print Preview
0.4									Print
0.2									0.3
0.5									0.5
0.2									0.2 -
0.1									0.1
0.1									0.1
0	0	0.0	0.4	0.0	0.0	4			0
	U	0.2	0.4	0.6	0.8				

图 3.37 图像绘制的窗口



0.4

0.6

0.8

rt <u>T</u>ools <u>D</u>esktop <u>W</u>indow <u>H</u>elp

Ctrl+O Ctrl+W Ctrl+S

Ctrl+P

0.2

ce As..

ų 🖌 - 🗔 🔲 📰 💷 🛄

- -

在本书中,重点要介绍的是图形的保存和输出功能,主要包括如下几个菜单项。

- □ Save 菜单项: 对绘制的图形进行保存, 可以保存为常用的多种图片格式。
- □ Save As 菜单项: 对绘制的图形另存,可以将其另保存为常用的多种图片格式,如 图 3.39 所示。
- □ Generate M-File 菜单项: MATLAB 可以通过该菜单自动生成创建图形的 MATLAB 代码,并且所创建的代码中保存了当前创建的图形对象设置的属性,下次可以利 用代码创建相同的图形。
- □ Export Setup 菜单项: 对绘制的图形进行输出设置。选择 File→Export Setup 命令, 将打开如图 3.40 所示的 Export Setup 窗口。在此窗口中,包含输出 Properties (属 性)和 Export Style (样式)的设置。其中 Properties 设置中又包含 Size、Rendering、 Fonts、Lines 属性的设置,Size 属性用于设置图形的大小,Rendering 属性用于设 置颜色模式和分辨率,其中对有较高要的图像需要提高分辨率,Fonts 属性用于设 置图形字体类型和字号,Lines 属性用于对图形中线条的设置。设置完毕后,可以 单击 Export Setup 窗口中的 Export 按钮保存设置的图形属性,以备下次重复使用。
- □ Print Preview 菜单项:用于打印预览。
- □ Print 菜单项:用于直接打印图形。
- 2. Edit菜单

Edit 菜单主要用于对图形的编辑,常用的菜单项如下。

□ Copy Figure 菜单项:可以复制当前的图形对象,并且直接粘贴到 Word 等文件中, 然后进行保存,这样的图片清晰度较高,且易于编辑。因此,这个菜单非常重要,

• 96 •

	🔰 Figure 1					0 13	5
ſ	📣 Save As	the local division of	the state of			>	3
	保存在(I):	🌗 bin		- + 6	∎ 🔺 🖬 ▼		
1	Ca.	名称	*		修改日期		1
	最近访问的位 置	퉬 registry			2010/9/17 20	):11 ):12	3
	桌面	퉬 win32			2010/9/17 20	):12	:
	<mark>唐</mark>						
	【】 计算机						
1		•	III				•
	Mat	文件名(01):	untitled. fig		• 1	保存 (S)	
		保存类型(T):	MATLAB Figure (*.fig)		•	取消	
			MATLAS Figure (* fig) Adobe Illustrator file (* Bitmap file (* bmp) EFS file (* eps) Enhanced metafile (*.emf) JEGG image (*.jpg) Paintbrush 24-bit file (* Portable Bitmap file (*.p Portable Bocument Format Fortable Graymap file (*.p Fortable Hetwork Graphics Portable Fixmap file (*.p TIFF image (*.tif) TIFF image (*.tif)	<. ai) . pcx)	,ng)		

是进行论文撰写和报告撰写的必备技能之一,希望读者可以好好掌握。



Size Rendering Fonts Lines	Width: Height	auto • I	Units: <u>centimet</u>	ers V		Apply to Figure Restore Figure Export OK Cancel Help
Export Styles Load settings f	rom:	default		•	Load	
· · · ·		d a f a sub			<b></b>	

图 3.40 Export Setup 菜单项窗口

□ Copy Options 菜单项:图形复制时参数的设置。选择 File→Copy Options 命令,将

• 97 •

打开如图 3.41 所示的图形复制属性设置页面。在此窗口中包括 Clipboard format(复制形式设置)、Figure background color(图片背景色设置)和 Size(大小设置)等功能选项区域。

General	Figure Copy Template Copy Options Preferences
Keyboard	
Fonts	Clipboard format
Colors	Metafile (may lose information)
M-Lint	Presence information (metafile if possible)
Toolbars	Preserve information (metallie il possible)
Command Window	🔘 Bitmap
Command History	Figure background color
-Editor/Debugger	
Help	○ Use figure color
Web	Force white background
Veriable Editory	
Warlable Editor	Transparent background
	Size
Time Series Teels	DALL C.
Figure Copy Template	Ministeringure screen size
Copy Options	select this option to copy the figure as it
Report Generator	the Print Preview settings to determine its size.
SystemTest	
Database Toolbox	
Image Processing	
Instrument Control	
Simulink	
Simscape	
-Simulink 3D Animation	
-Simulink Control Design	
Video and Image Processing Blocks	
	OK Cancel Apply Help

图 3.41 Copy Options 菜单项窗口

□ Figure Properties 菜单项: 打开图形窗口属性设置对话框,可以设置图形颜色、标题、显示类型等属性。选择 Edit→Figure Properties 命令,在原图下方将打开图形窗口设置对话框,如图 3.42 所示。在界面中显示的可以设置的属性有 Figure Name (图形名称),Colormap(图形控制),Figure Color(颜色)和 Show Figure Number (是否显示图形名称)。其他的一些属性,可以通过单击 Figure Properties 对话框中的 More Properties 按钮,打开如图 3.43 所示的图形窗口属性查看器,查看并修改设置各属性。



•	2i =:			
	BeingDeleted	off		^
	BusyAction	queue	*	
	ButtonDownF		ø	
	Clipping	on	٠	
	CloseReques	dosereq	ø	Ξ
Đ	Color			
	CreateFcn		ø	
	CurrentChara	0	ø	
ŧ	CurrentPoint	[330 129]		
	DeleteFcn		ø	
	DockControls	on	٣	
	FileName		ø	
	FixedColors	[ 0.0; 1.0;	D	
	HandleVisibil	on	*	
	HitTest	on	٠	
	IntegerHandle	on	٣	
	Interruptible	on	*	
	InvertHardco	on	٣	
	KeyPressFcn	[3x1 cell.	. 0	
	KeyReleaseFcn		ø	
	MenuBar	figure	Ŧ	Ŧ

图 3.42 Figure Properties 菜单项窗口



• 98 •

□ Axes Properties 菜单项:打开图形窗口坐标轴对象的属性设置对话框,可以设置坐标范围、刻度、比例、标注等信息。选择 Edit→Axes Properties 命令,在原图下方将打开坐标轴设置对话框,如图 3.44 所示。在界面中包 x 轴、y 轴、z 轴设置页面,打开相应的选项,可对不同坐标轴进行设置。对每一坐标轴,界面中显示的可以设置的属性有坐标轴的标注、标轴的数据范围、坐标轴的尺寸度、字体等,其中对于刻度及其刻度下的显示,需要单击 Ticks 按钮,打开如图 3.45 所示的刻度设置对话框进行设置。刻度的设置可以选择 Auto(自动)、Manual(人工)、Step by(按指定步长)选项,同时可以手动直接在下方的表格中输入相应的刻度。



图 3.44 Axes Properties 菜单项窗口

图 3.45 Ticks 按钮窗口

- Current Object Properties 菜单项:打开当前对象的属性设置对话框,通过标题栏中的图标选择当前对象,再单击此菜单,将打开当前对象的属性设置对话框。此菜单的属性设置是很有用的,需要设置什么属性,直接用鼠标选中即可,所见即所得,与之前的通过属性查看器设置其他属性相比更为方便、实用。
- □ Colormap 菜单项:用于设置色图的模式,色图是指以不同的颜色对应不同的数值, 其中的对应方式即为 Colormap 菜单项设置的内容。选择 File→Colormap 命令,将 打开如图 3.46 所示的 Colormap Editor 窗口,在此窗口中,选择 Tools 菜单,可以 选择不同的色图模式,有 autumn、blue 等,颜色映射中数据的范围通过 Color data min 和 Color data max 文本框进行设置。
- 3. View菜单

View 菜单用于决定不同的工具条形和对话框的显示,如图 3.47 所示,选择该菜单项,即显示相应的工具条,View 菜单中的图形窗口显示的工具条将前有"✓"标志。

下面对每个菜单项的功能进行介绍。

- □ Figure Toolbar 菜单项: 控制图形窗口中工具栏的显示。
- □ Camera Toolbar 菜单项: 控制图形中照片操作工具栏的显示。
- □ Plot Edit Toolbar 菜单项: 控制画图编辑工具条的显示。

- □ Figure Palette 菜单项: 控制图画板的显示。
- □ Plot Browser 菜单项: 控制绘图浏览器的显示。
- □ Property Editor 菜单项: 控制属性编辑器的显示。



图 3.46 Colormap 菜单项窗口

图 3.47 View 菜单项窗口

#### 4. Insert菜单

通过 Insert 菜单可以向图片中添加不同的绘图对象,可以在图形窗口中添加的对象有 X Label (X 轴)、Y Label (Y 轴)、Z Label (Z 轴)、Title (图例)、Legend (图例)、 Colorbar (颜色条)、Line (直线)、Arrow (箭头)、Text Arrow (带箭头的文本框)、 Double Arrow (双向箭头)、TextBox (文本框)、Rectangle (矩形)、Ellipse (椭圆)、 Axes (坐标轴)和 Light (光源),如图 3.48 所示。



图 3.48 Insert 菜单项窗口

• 100 •

#### 5. Tool菜单

Tool 菜单用于提供一些图形编辑的工具, 便于更好地观察、编辑图形, 如图 3.49 所示。



图 3.49 Tool 菜单项窗口

- □ Edit 菜单项: 控制图形编辑状态,当选择该菜单,菜单前有"✓"标志,表示当前 图形窗口处于被编辑状态。
- □ Zoom In 菜单项: 控制图形放大。
- □ Zoom Out 菜单项: 控制图形缩小。
- □ Pan 菜单项: 控制手动移动图形。
- □ Rotate 3D 菜单项: 控制三维旋转图形, 以便从不同角度观察图形, 如图 3.50 所示。



图 3.50 Rotate 3D 菜单项示意图

- □ Data Cursor 菜单项:从图形中显示数据点的坐标。
- □ Reset View 菜单项: 重置编辑过的图形。
- □ Options 菜单项:用于上述菜单项的一些附加数据设置,包括缩放设置: Unconstrained zoom(无限制的缩放)、Horizonted Zoom(水平方向缩放)、Vertical Zoom(垂直方向缩放);图形移动的控制:Unconstrained Pan(无限制的移动)、 Horizonted Pan(水平方向移动)、Vertical Pan(垂直方向移动);Display Cursor as Datatip(在图形窗口内显示数据点坐标)、Display Cursor in Window(在另外的窗 口内显示数据点的坐标)。
- □ View Layout Grid 菜单项:控制图形窗口子窗口的排列布局,选择此菜单项将弹出 如图 3.51 所示的 Align Distribute Tool 窗口,可设置子窗口的对齐方式。紧接着 View Layout Grid 菜单项下面的 Align 与 Distribute 菜单项功能与其类似,但是 Align Distribute Tool 窗口在设置子菜单窗口的排列布局方式时更为直观。



图 3.51 View Layout Grid 菜单项窗口

- □ Basic Fitting 菜单项:数据曲线拟合,选择此菜单将打开如图 3.52 所示的 Basic Fitting 窗口。在该窗口中,用户可以 Select data(选择待拟合的数据源)、Center and scale X data(控制数据归一化)、Check to display fits on figure(设置拟合的数据 模型)、Show equations(设置拟合函数的显示)、Significant digits(设置数值的 有效位数)、Plot residuals(控制拟合模型残差的绘制)和 Show norm of residuals (控制最大残差模的显示)等。
- Data Statistics 菜单项:对绘图数据进行简单地统计分析,选择此菜单将打开如图 3.53 所示的 Data Statistics 窗口。在此期间,用户可以获取的统计参数有 min (最小值)、max (最大值)、mean (平均值)、median (中位数)、std (方差)及 range(极差)。同时单击 Save workspace 按钮,可以把这些统计参数保存至 MATLAB 工作空间中。

• 102 •

A Basic Fitting - 1	
Select data: data 1 💌	Numerical results
Check to display fits on figure spline interpolant shape-preserving inter linear quadratic cubic dth degree polynomial 5th degree polynomial 6th degree polynomial 7th degree polynomial 7th degree polynomial 7th degree polynomial 9 Show equations Significant digits: 2 • Plot residuals Bar plot Subplot Subplot	Fit: 4th degree polynomial Coefficients and norm of residuals y = p1*x <sup>2</sup> 4 + p2*x <sup>2</sup> 3 + p5 Coefficients: p1 = 2,3566e=016 p2 = -2,4522=015 p3 = 9,7879e=015 p4 = 1 p5 = 5,2768e=015 Borm of residuals = 1.2755e=015 Save to workspace
Help Close	$\leftarrow \rightarrow$

图 3.52 Basic Fitting 菜单项窗口

4	🔺 Data Statistics - 1						
9	Statistics for data 1						
0	Check to	plot statistics or	n f	igure:			
		Х		Y			
	min	1		1			
	max	5		5			
	mean	3		3			
	median	3		3			
	mode	1		1			
	std	1.581		1.581			
	range	4		4			
	Save to workspace Help Close						

图 3.53 Data Statistics 菜单项窗口

## 6. 数据浏览窗口的图形绘制

在 MATLAB 工作空间的数据浏览窗口提供了对存储数据的快速绘图方式,如图 3.54 所示,选中数据,单击 workspace 的绘图工具栏,即可快速绘制不同类型的图形,便于数据的观察。

Eile Edit View G	raphics Debug Parallel	Desktop Window Help		
1 6 8 9 7	🥙 🍓 🗊 🖹 🥝 Current	Directory: C:\Users\mike\Documents\MATLAB	<ul> <li>✓ (a)</li> </ul>	
Shortcuts El How t	o Add 🕅 What's New	•		
Workspace	* E *	× Command Window	• • • × Command History • • • •	×
1 🖻 🐿 🖏 🐻 [ 🖂	• Stack: Base •	fx >>	-std(Iads(:))	•
Name + Val	ue Min		-Iaa = abs(I-I2900)>15;	
			%形态学	
			-% se=strel('diamond', {	
			-% fo = Iaa;	
			-% fo=imclose(fo, se);%.	
			-% fo=imopen(fo, se);%直	
			-% se=strel('square', 8)	
			-% fo=imclose(fo, se);%	
			-% fo=imopen(fo, se);%直	
			-% figure,	
			-% imshow(fo)	
× []		P	-Ib = I.*fo;	
Current Directory	H □ 4	x	-imshow(Ib,[])	
🔹 🔹 📜 « mike 🕨 De	ocuments + MATLAB + -	0-	-Iaa = abs(I-I2900)>15;	
D Name -	Date Modified		-Ib = Iaa. *I;	
▲ 小昭存利 m	16-4-6 下午7:07		-imshow(Ib,[])	
空光服专利.asv	16-4-6下午7:03		<b>≜%</b> 16-4-11 下午11:04 -	
🖺 Untitled2.m	15-8-2 下午5:52		-plot([1 2 3 4 5])	
🖄 trdec.m	15-11-3 下午9:40	-	cle	1
#S \$11 \$ 4	10.0.7740.00	^		
A Start				T

图 3.54 图像可视化绘制窗口

3.2 其他二维图形

除了上述的常用绘图方法外,在 MATLAB 中还有一些其他的绘图手段可以用来绘制

• 103 •

日常办公和生活中产生的各种数据,以利用于对比和分析。例如:在绘制图形时,除了 plot() 函数外,还可以使用 semilogx()、semilogy()和 loglog()函数。另外,MATLAB 还提供一些特殊的二维图形的绘制方法,如饼图、条形图、直方图、冲击响应函数等特殊图形的绘制。下面进行一一介绍。

## 3.2.1 其他绘图函数

在 MATLAB 中为了更好地显示数据的对比性,或更好、更直观地描述数据的特点,除了常用的笛卡尔坐标系绘图函数 plot()函数和极坐标系绘图函数 polar()函数以外,还提供了半对数坐标图形绘制函数 semilogx()和 semilogy()以及对数坐标图形的绘制函数 loglog()函数。下面对这些函数进行介绍。

#### 1. 半对数坐标图形绘制函数

semilogx()和 semilogy()函数用来绘制半对数坐标图形,其调用格式与 plot()函数类似, 唯一不同的是在半对数坐标系中绘制图形,这样对于变化范围较大的曲线,容易显示出直 观的图形。semilogx()和 semilogy()函数的区别如下。

□ semilogx: 绘制 x 轴位对数坐标, y 轴位线性坐标的二维图形。

□ semilogy: 绘制 y 轴位对数坐标, x 轴位线性坐标的二维图形。

【例 3-38】 绘制函数 y=e<sup>x</sup> 的半对数坐标图形。MATLAB 代码如下:

```
° x轴
x=0:0.5:5;
°y轴
y = \exp(x);
8 绘图
figure
% 笛卡尔坐标系
subplot(3, 1, 1)
plot(x, y, 'r-.')
title('笛卡尔坐标系')
8 半对数坐标系
subplot(3, 1, 2)
semilogx(x, y, 'g:')
title('x 轴为对数坐标系')
subplot(3, 1, 3)
semilogy(x, y, 'b-')
title('y 轴为对数坐标系')
```

MATLAB运行结果如图 3.55 所示。

## 2. 对数坐标图形绘制函数

loglog()函数用来绘制对数坐标图形,其调用格式与 plot()函数类似,唯一不同的是在 对数坐标系中绘制图形,这样对于变化范围较大的曲线,容易显示出直观的图形。

【例 3-39】 绘制函数 y=e<sup>x</sup>的对数坐标图形。MATLAB 代码如下:

• 104 •



MATLAB运行结果如图 3.56 所示。



## 3. 符号函数的绘制

fplot()函数可以根据函数表达式自动调整自变量的范围,然后在不用给函数显示赋值

• 105 •

的情况,直接将函数绘制出来。fplot()函数常用来查看符号函数的变化规律,能够自动根据符号函数变化快慢,进行自适应采样,也就是说,对于变化快的地方,采样间隔小,而对于变化慢的地方,采样间隔大,从而在绘制图形的计算量大大降低时,仍可以精确反映图形的变化情况。函数的调研格式如下。

- □ fplot('fun', limites): 在指定的坐标范围 limits 内绘制函数 fun()的图形,其中 limites 的取值方式与坐标轴控制函数 axis()的坐标轴取值方式相同。而函数 fun()必须是一个包含 y=fun(x)的 M 函数文件,或包含变量 x 的 MATLAB 自带函数并能够用 eval() 函数计算的字符串。
- □ fplot('fun', limites, LineSpec): 在指定的坐标范围 limits 内绘制函数 fun()的图形, 其中 limites 的取值方式与坐标轴控制函数 axis()的坐标轴取值方式相同。而 LineSpec 参数设置图形绘制时采样的线型、数据点的样式和颜色等属性。
- □ fplot('fun', limites, err): 在指定的坐标范围 limites 内绘制函数 fun()的图形,其所允 许的相对误差不超过 err。

```
【例 3-40】 利用符号函数绘制函数 y=e<sup>x</sup> 的图形。MATLAB 代码如下:
```

```
% 绘图
figure
fplot('exp(x)',[0, 5] , 'k-.')
set(findallcgcf,'type','linewidth',3)
title('符号函数绘制图形')
```

MATLAB运行结果如图 3.57 所示。



## 3.2.2 饼图

饼图可以方便且清晰地描述各项数据的大小和所占总数据和的比例,从而在生产成本 控制等系列问题中经常使用。饼图是指一个将总数据显示成一个圆形,然后将这个圆剖分 成多个扇区,每个扇区都代表一个数据项,描述各个数据项占数据总和的比例。在 MATLAB 中可以利于 pie()函数描绘平面饼图。调用格式如下。

• 106 •

- □ pie(*x*): 绘制数据 *x* 的饼图, *x* 可以是向量或者矩阵, *x* 中的每一个元素将代表饼图的一个扇区,同时饼图中显示各元素总和的比例。
- □ pie(*x*, explode): 绘制数据 *x* 的饼图,其中参数 explode 可以用来设置饼图中某个重要的扇区进行抽取式重点显示,这里需要注意的是, explode 向量的长度与 *x* 中的元素个数相等,并与 *x* 中的元素意义对应, explode 元素为非零值,对应的元素扇区将从饼图中分离显示,通常非零值都设置为 1。
- □ pie(*x*, labels): 绘制数据 *x* 的饼图,其中参数 labels 可以用来设置饼图中各个扇区 的显示标注,注意参数 labels 应该为字符串或者数字利用向量 *X* 中的数据描绘饼图。

【例 3-41】 尝试画出 X = [1, 1, 2, 2, 3, 4, 5]的饼图。MATLAB 代码如下:

```
% 准备数据
X = [1, 1, 2, 2, 3, 4, 5];
% 绘图
pie(X)
```

MATLAB运行结果如图 3.58 所示。

注意, *X*中的数据被看做频数, 饼图中比率的获得: *X*中的元素 *x*[i]/sum(*X*)。当 *X*中 所有元素的和 sum(*X*)<1.0 时, 图形不是整一个圆。例如令 *X* = [0.1,0.2,0.3], 再次运行例 3-41 中的 MATLAB 程序可以得到如图 3.59 所示的结果,可以看到所绘制的饼图的图形不再是 一个整圆。







图 3.59 所有数据和小于 1 时饼图的绘制图形

【例 3-42】 有一位研究生,在一年中平均每月的费用为生活费 190 元,资料费 33 元, 电话费 45 元,购买衣服 42 元,其他费用 45 元。请以饼图表示出他每月的消费比例,并在 饼图中分离出使用最多的费用和使用最少的费用的切片。MATLAB 代码如下:

```
% 数据准备
x=[190 33 45 42 45];
% 分离显示设置
explode=[1 1 0 0 0];
% 绘图
figure()
colormap hsv
pie(x,explode,{'生活费','资料费','电话费','购买衣服','其他费用'})
title('饼图')
```

MATLAB运行结果如图 3.60 所示。

## 3.2.3 条形图

条形图是用单位长度表示一定的数量,各数据 变量按照数量的多少画成长短不同的条形,以便于 实验分析。MATLAB 中绘制条形图的基本函数为 bar()和 barh(),它们的调用格式如下。

 □ bar(Y): 当 Y 为矢量时, Y 中的每个元素绘出一个条形; 当 Y 为矩阵时,函数 bar 先把 Y 矩阵分解为行向量,再分别对每行元素绘制条形,设 Y 为 m×n 的矩阵,那么绘制条 形图时先将 Y 分成 m 组,再分别对每组中的 n 个元素绘出图形。



- □ bar(*x*,*Y*): 将 *x* 作为坐标轴, 绘制数据 *Y* 的条形图。注: 要求 *x* 向量必须单调递增。
- □ bar(...,width): 控制相邻条形的宽度和组内条形的分离,默认值为 0.8,当指定其值 为 1 时,组内的条形挨在一起。
- □ bar(...,'style'): 设置条形的类型, style 有两种类型, 即 stacked 和 group。stacked 参数为在矩阵 Y 中每一行绘制一个条形, 条形的高度由行元素和控制, 每个条形 都用多种颜色表示,不同颜色表示不同种类元素及每行元素所占总和的比例; group 绘制 *n* 条形图组(*n* 为矩阵 Y 的行数),每一个条形图中有 *m* 个垂直条形(*m* 为矩阵 Y 的列数),group 为 style 的默认值。
- □ bar(...,LineSpec): LineSpec 控制绘制条形的颜色。
- □ h=bar(...): 返回所绘制图形句柄。
- □ barh(...): 绘制水平条形图。
- □ h=barh(...): 返回所绘制水平图形句柄。

【例 3-43】 随机产生 5×3 的数组,设定条形的宽度为 1.5,画出堆型二维垂直、水平 条形图。MATLAB 代码如下:

```
%随机函数产生 5*3 的数组,对产生的数据取整
Y = round(rand(5,3)*10);
8 绘图
subplot(2,2,1)
bar(Y, 'group')
title 'Group'
%堆型二维垂直条形图
subplot(2,2,2)
bar(Y,'stack')
title('Stack')
%堆型二维水平条形图
subplot(2,2,3)
barh(Y, 'stack')
title('Stack')
%设定条形的宽度为1.5
subplot(2,2,4)
```

• 108 •

```
bar(Y,1.5)
title('Width = 1.5')
```

MATLAB运行结果如图 3.61 所示。



图 3.61 条形图运行结果

【例 3-44】 有一位研究生,在一年中平均每月的费用为生活费 190 元,资料费 33 元, 电话费 45 元,购买衣服 42 元,其他费用 45 元。请以柱状图表示出他每月的消费比例。 MATLAB 代码如下:

```
% 数据准备
y=[190 33 45 42 45];
x=1:5
% 绘图
figure
bar(x,y)
title('柱状图');
set(gca,'xTicklabel',{'生活费','资料费','电话费','购买衣服','其他费用'})
```

MATLAB运行结果如图 3.62 所示。

## 3.2.4 直方图

直方图是根据数据的分布情况,对数据进 行分组,以组距为底边、以频数为高度的长方 形矩形图。绘制直方图经常用 hist()函数

> □ *h*=hist(*y*):将向量*y*中的函数放到十个 柱的直方图中,返回值 *h* 为包含每个柱 的元素个数组成的向量。如果 *y* 是矩 阵,则按照矩阵的列来画图。



- □ *h*=hist(*y*, *m*): *m* 为标量, 指定柱的个数。
- □ h=hist(y, x): x 是向量,将参数 y 中的元素放到 length(x)个由 x 中元素指定的位置

• 109 •

为中心的直方图中。

与直方图相比,条形图是处理分类变量的,如男、女,一年级、二年级、三年级,这 类变量(如男和女)中间没有其他选项。但直方图是连续变量,这种变量是任何数都能取 的,如收入,3000~4000元中间还有可能是3500元,3000元和3500元中间还有3200元。

【例 3-45】 绘制简单的直方图。MATLAB 代码如下:

hist(x)

MATLAB运行结果如图 3.63 所示。

## 3.2.5 面积图

面积图将数据点显示为一组由线连接的点,并填充线下方的所有区域。MATLAB 中绘制面积图的函数为 area()函数,其调用格式如下。

□ area(y): 当 y 为向量时,则以 y 的下标为横坐标, y 中各数据点连接成线,在默认 情况下将 x 轴作为基准线,填充 x 轴与连接线间的区域部分。当 y 为矩阵时,则对 矩阵中的各列进行操作,绘出多条曲线,此时每列数据都会以前列数据绘制的曲 线为基线进行绘制,并填充相应的颜色。

□ area(*x*, *y*): 制定绘制的面积图的横坐标。

【例 3-46】 绘制简单的面积图。MATLAB 代码如下:

```
x=[1,2,3,4];
area (x)
```

MATLAB运行结果如图 3.64 所示。



# 3.2.6 散点图

散点图是将数据序列显示为一组点的图形。它能够反映因变量随自变量变化而变化的趋势,常用于回归分析。在 MATLAB 中使用 scatter()函数绘制散点图,该函数调用格式如下。

- □ scatter(*x*,*y*): 绘制二维散点图, *x*、*y*用于指定散点位置。
- □ scatter(*x*,*y*,*s*,*c*):可以采用着色散点表示数据的位置,向量*x*、*y*用于指定散点位置, *s*表示散点尺寸,*c*表示散点颜色,注意两者可以单独使用。

• 110 •

【例 3-47】 绘制简单的散点图。MATLAB 代码如下:

t = 0:0.5:pi; y = cos(t); scatter (t, y,'filled')

MATLAB运行结果如图 3.65 所示。

3.2.7 排列图

排列图又称累托(Pareto)图,由一 个横坐标、两个纵坐标、多个按高低顺序 排列的条形和一条折线组成。其中,横坐 标表示各因素,左纵坐标表示频数,右纵 坐标表示频率,折线表示累积的频率。该 图能较好地分析各因素的重要性,可用于 寻找主要问题或主要原因。在 MATLAB 中



pareto()函数用于绘制排列图,其调用格式如下。

- **\Box** pareto(*y*): 绘制数据 *y* 的排列图。*y* 值的大小用排列图条形的高度表示。
- □ pareto(*y*,*x*): 绘制数据 *y* 的排列图。当 *x* 为数值时,用于指定数值型的横坐标。当 *x* 为字符串时,用于指定字符串型的横坐标。

【例 3-48】 绘制简单的排列图。MATLAB 代码如下:

Y=[100 98 97 90 90]; names={'第1名' '第2名' '第3名' '第4名' '第5名'}; pareto(Y,names)

MATLAB运行结果如图 3.66 所示。



## 3.2.8 罗盘图

MATLAB 中提供了 compass()函数绘制罗盘图,罗盘图绘制于一个圆盘中,从原点出发的箭头,箭头在圆盘中的角度用于表示数据的角度,箭头的长短用于表示数据的大小。 其调用格式如下。

• 111 •

- □ compass(u,v): 输入参数u, v分别指定数据在罗盘图中的x分量和y分量。
- □ compass(*z*): 绘制仅一个输入参数的罗盘图, *z* 为复数矩阵, 复数的实部代表罗盘 图的 *x* 分量, 虚部代表 *y* 分量。
- □ compass(..., LineSpec): 绘制罗盘图,设置罗盘图的线型。
- □ h = compass(...)函数:返回 line 对象的句柄给 h。

【例 3-49】 绘制简单的罗盘图。MATLAB 代码如下:

```
x = 1:100;
y = rand(1,100);
compass(x,y)
```

MATLAB运行结果如图 3.67 所示。

## 3.2.9 羽毛图

羽毛图是以箭头的形式绘制矢量数据,在 MATLAB 中,绘制羽毛图的函数为 feather() 函数,该函数可以用于进行光流变换的描述等,其调用格式如下。

- □ feather(u,v): 输入参数u、v分别指定数据在羽毛图中对应的x分量和y分量。
- □ feather(z): 输入参数 z 为复数矩阵, 复数的实部代表羽毛图的 x 分量, 虚部代表 y 分量。

【例 3-50】 绘制简单的羽毛图。MATLAB 代码如下:

```
x = 1:100;
y = rand(1,100);
feather(x,y)
```

MATLAB运行结果如图 3.68 所示。



## 3.2.10 矢量图

在 MATLAB 中,函数 quiver()用于在二维平面上绘制矢量图,矢量图通常和其他图形 一起使用,用于显示数据的方向,如绘制电磁场或者力场等,其调用格式如下。 quiver(*x*, *y*, *u*, *v*):输入参数 *x*、*y*用于指定绘制矢量的位置,*u*、*v*用于指定绘制的矢量

• 112 •

在水平和竖直方向的大小。

【例 3-51】 绘制简单的矢量图。MATLAB 代码如下:

x = 1:4; y = rand(1,4); u = [1 0 -1 0]; v = [0 1 0 -1]; quiver(x, y, u, v)

MATLAB运行结果如图 3.69 所示。

## 3.2.11 杆型图

杆型图主要用来表示离散数据的变化 规律,以离散的圆点表示每个数据点,并 用线段把数据点和坐标轴连接起来,形如 杆型。而 plot()函数默认把离散的数据点间 用线段连接起来。MATLAB 提供 stem()函 数用于绘制杆型图,其调用格式如下。

□ stem(y):绘制离散数据y的杆型图, 横坐标为默认,如果数据y为向量,



即最后绘制的杆型图为数据 y 在 x 轴上等间距排列的杆型,横坐标为 1: length(y); 如果 y 为矩阵,则横坐标为矩阵的行,同一行中的元素绘制在相同的横坐标下,表现为两条杆线。

- □ stem(*x*, *y*): 指定横坐标 *x*, 绘制数据 *y* 的杆型图。
- □ stem(..., 'fill', LineSpec): 'fill'表示杆型图中表示数据的点设置为填充, LineSpec 设置杆型图杆型的线条,两者可以单独使用。

【例 3-52】 绘制简单的杆型图。MATLAB 代码如下:

```
x = 0:0.5:pi;
y = sin(x);
stem(y, 'fill', 'r-.')
```

MATLAB运行结果如图 3.70 所示。





## 3.2.12 阶梯图

函数 stairs()用于绘制阶梯图,其调用格式如下。

- □ stairs(*y*): 绘制数据 *y* 的阶梯图。如果 *y* 为向量,则绘制数据 *y* 中每个元素的阶梯 变化图,横坐标 *x* 为 1 到 length(*y*),如果 *y* 为矩阵,则对数据 *y* 的每一行画一阶梯 图,横坐标 *x* 为从 1 到 *y* 的列数.
- □ stairs(x, y):指定横坐标 x 绘制阶梯图。

□ stairs(..., LineSpec):参数 LineSpec 设置阶梯图中线条的线型、标记符号和颜色等。 【例 3-53】 绘制简单的阶梯图。MATLAB 代码如下:

```
x = 0:0.5:pi;
y = sin(x);
stairs(y, 'k-.')
```

MATLAB运行结果如图 3.71 所示。



3.3 三维图形

在 3.2 节中主要介绍了二维图形的绘制,但在有些信息量比较丰富的情况下,往往要 绘制更为复杂的三维图形,三维图形信息丰富,能更好地反映数据之间的相互规律。但一 般情况下三维图形的绘制比较复杂,而 MATLAB 为我们提供了一些函数可以直接绘制漂 亮的三维图形,同时还提供了完整的三维图形编辑的功能。与一般的统计绘图软件相比, 在三维图形绘制和编辑方面,MATLAB 软件更有优势。本节将重点介绍 MATLAB 在三维 图形绘制中的基础知识,包括三维图形绘制和编辑的相关知识。

## 3.3.1 三维曲线

#### 1. 用plot3()函数画三维曲线

在 MATLAB 中提供了 plot3()函数用于绘制三维曲线,其函数的调用法与二维曲线绘

• 114 •

制函数 plot()类似,其调用格式如下。

- □ plot3(*x*1, *y*1, *z*1, LineSpec, ...): 在三维空间中绘制以 *x*、 *y*、 *z* 为坐标轴的曲线,曲线 由 *x*1、 *y*1、 *z*1 中的元素确定, *x*1、 *y*1、 *z*1 为向量或矩阵,输入参数 LineSpec 用于 指定绘制的三维曲线的线型、标记符、颜色等。
- □ plot3(...,'PropertyName',PropertyValue,...): 设置三维绘图函数绘制的三维曲线的各种属性的属性值。

【例 3-54】 绘制简单的三维曲线图。MATLAB 代码如下:

```
% 数据准备
t=0:pi/100:20*pi;
x=sin(t);
y=cos(t);
z=t.*sin(t).*cos(t);
% 绘图
figure
plot3(x,y,z)
grid
title('三维绘图');
xlabel('X')
ylabel('Y')
zlabel('Z')
```

MATLAB运行结果如图 3.72 所示。



#### 2. 三维网格图的绘制

在 MATLAB 中,进行三维图形绘制时,常常需要首先创建三维网格,也就是先创建 平面图的坐标系。在 MATLAB 中,常用 meshgrid()函数生成网格数据,其调用格式如下。

- □ [*X*,*Y*]=meshgrid(*x*,*y*):用于生成向量*x*和*y*的网格数据,即变换为矩阵数据*X*和*Y*, 矩阵*X*中的行向量为向量*x*,矩阵*Y*的列向量为向量*y*。
- □ [X,Y]=meshgrid(x): 生成向量x的网格数据,函数等同于[X,Y]=meshgrid(x,x)。
- □ [*X*,*Y*,*Z*]=meshgrid(*x*,*y*,*z*): 生成向量 *x*、*y*、*z* 的三维网格数据, 生成的数据 *X* 和 *Y* 可分别表示三维绘图中的 *x* 和 *y* 坐标。

三维网格图形是指在三维空间内连接相邻的数据点,形成网格。在 MATLAB 中绘制

• 115 •

三维网格图的函数主要有 mesh()函数、meshc()函数和 meshz()函数。其中, mesh()函数最常用, 其调用格式如下。

□ mesh(*x*,*y*,*z*): 绘制三维网格图, *x*、*y*、*z*分别表示三维网格图形在*x*轴、*y*轴和*z*轴的坐标,图形的颜色由矩阵*z*决定。

□ mesh(*Z*): 绘制三维网格图,分别以矩阵 *Z* 的列下标、行下标作为三维网格图的 *x* 轴、*v* 轴的坐标,图形的颜色由矩阵 *Z* 决定。

□ mesh(...,*C*): 输入参数 *C* 用于控制绘制的三维网格图的颜色。

□ mesh(...,'PropertyName',PropertyValue,...): 设置三维网格图的指定属性的属性值。

函数 meshc()可绘制带有等值线的三维网格图,其调用格式与函数 mesh()基本相同,但函数 meshc()不支持对图形网格线或等高线指定属性的设置。

函数 meshz()可绘制带有图形底边的三维网格图,其调用格式与函数 mesh()基本相同,但函数 meshz()不支持对图形网格线指定属性的设置。

另外,函数 ezmesh()、ezmeshc()和 ezmeshz()可根据函数表达式直接绘制相应的三维网格图。

由于网格线是不透明的,绘制的三维网格图有时只能显示前面的图形部分,而后面的部分可能被网格线遮住了,没有显示出来。 MATLAB 中提供了命令 hidden 用于观察图形 后面隐藏的网格, hidden 命令的调用格式如下。

□ hidden on:设置网格隐藏部分不可见,默认情况下为此状态。

□ hidden off: 设置网格的隐藏部分可见。

□ hidden: 该命令用于切换网格的隐藏部分是否可见。

【例 3-55】 绘制简单的三维网格图。MATLAB 代码如下:

```
% 数据准备
t=0:pi/10:pi;
x=sin(t);
y=cos(t);
[X,Y]=meshgrid(x,y);
z =X + Y;
% 绘图
figure
mesh (z,'FaceColor','W','EdgeColor','K')grid
title('三维网格图');
```

#### MATLAB运行结果如图 3.73 所示。



图 3.73 简单的三维网格图

• 116 •

#### 3. 三维空间等高线图

在 MATLAB 中,采用 contour3()函数进行三维空间等高线图的绘制。该函数生成一个 定义在矩形格栅上曲面的三维等高线图,其调用格式如下。

- □ contour3(*Z*): 绘制矩阵 *Z* 的三维等高线图,矩阵 *Z* 中的元素表示距离 *xy* 平面的高度(矩阵 *Z* 至少是 2×2 阶的)。等高线的条数和高度由 MATLAB 自动选取。当形式为[*m*, *n*]=size(*z*)时,指定了 *x* 轴范围为[1: *n*], *y* 轴范围为[1: *m*]。
- □ contour3(*Z*, *n*): 绘制出矩阵 *Z* 的 *n* 条三维等高线图。
- □ contour3(*Z*, *v*): 指定等高线的高度,向量 *v* 的维数要求与等高线条数相同; contour3(*Z*,[*h*,*h*])表示只画一条高度为*h*的等高线。
- □ contour3(*X*,*Y*,*Z*)、contour3(*X*,*Y*,*Z*,*n*)、contour3(*X*,*Y*,*Z*,*v*): *X*和*Y*分别表示 *x*-轴和*y*-轴的范围。当*X*为矩阵时,用*X*(1,:)表示 *x*-轴的范围;当*Y*为矩阵时,用*Y*(1,:)表 示*y*-轴的范围;当*X*和*Y*均为矩阵时,二者必须为同型矩阵。使用形式虽然会有 不同,但所起作用相同,都与命令 surf 相同。当*X*或*Y*的间距不规则时,contour3 仍会按照规则间距计算等高线,再把数据转变给*X*或*Y*。
- □ contour3(...,LineSpec): LineSpec 用于指定等高线的线型和颜色。
- □ [*C*,*h*]=contour3(...): 绘制等高线图并返回参量 *C* 和 *h*。*C* 表示与命令 contourc 中相 同的等高线矩阵, *h* 表示所有图形对象的句柄向量; 除非没有指定 LineSpec 参数, contour3 会生成 patch 图形对象, 且当前的 colormap 属性与 caxis 属性将控制颜色 的显示。不管使用哪种形式,此命令均能生成 line 图形对象。

【例 3-56】 简单举例说明 contour3 用法。MATLAB 代码如下:

```
% 数据准备
[X,Y] = meshgrid([-2:.25:2]);
Z = X.*exp(-X.^2-Y.^2);
% 绘图
figure
contour3(X, Y, Z, 30, 'K')
```

MATLAB运行结果如图 3.74 所示。



图 3.74 简单的三维等高线图

## 3.3.2 三维表面图的绘制

三维表面图也可以用来表示三维空间内数据的变化规律,与之前讲述的三维网络图的 不同之处在于对网格的区域填充了不同的色彩。在 MATLAB 中绘制三维表面图的函数为 surf()函数,其调用格式如下。

- □ surf(Z): 绘制数据 Z 的三维表面图,分别以矩阵 Z 的列下标、行下标作为三维网 格图的 x 轴、y 轴的坐标,图形的颜色由矩阵 Z 决定。
- □ surf(X, Y, Z): 绘制三维表面图, X、Y、Z分别表示三维网格图形在 x 轴、y 轴和 z 轴的坐标,图形的颜色由矩阵 Z 决定。
- □ surf(*X*, *Y*, *Z*, *C*): 绘制三维表面图, 输入参数 *C*用于控制绘制的三维表面图的颜色。

□ surf(..., 'PropertyName', PropertyValue): 绘制三维表面图,设置相应属性的属性值。

函数 surfc()用于绘制带等值线的三维表面图,其调用格式同函数 surf()基本相同,函数 surfl()可用于绘制带光照模式的三维表面图,与函数 surf()和 surfc()不同的调用格式如下。

- □ surfl(...,'light'): 以光照对象 light 生成一个带颜色、带光照的曲面。
- □ surfl(....'cdata'): 输入参数 cdata 设置曲面颜色数据, 使曲面成为可反光的曲面。

□ surfl(...,s): 输入参数 s 为一个二维向量[azimuth,elevation], 或者三维向量[x,y,z], 用于指定光源方向,默认情况下光源方位从当前视角开始,逆时针 45°。

【例 3-57】 简单对 surf()函数进行举例。MATLAB 代码如下:

```
%数据准备
xi=-10:0.5:10;
yi=-10:0.5:10;
[x,y]=meshgrid(xi,yi);
z=sin(sqrt(x.^2+y.^2))./sqrt(x.^2+y.^2);
% 绘图
surf(x,y,z)
```

MATLAB运行结果如图 3.75 所示。



图 3.75 简单的三维表面图

• 118 •

## 3.3.3 三维切片图的绘制

在 MATLAB 中 slice()函数用于绘制三维切片图。三维切片图可形象地称为"四维图",可以在三维空间内表达第四维的信息,用颜色来标识第四维数据的大小。slice()函数的调用格式如下。

- □ slice(*v*, *sx*, *sy*, *sz*): 输入参数 *v* 为三维矩阵(阶数为 *m x n x p*), *x*、 *y*、 *z* 轴默认状态下分别为 1: *m*、 1: *n*、 1: *p*,数据 *v* 用于指定第四维的大小,在切片图上显示为不同的颜色,输入参数 *sx*、 *sy*、 *sz* 分别用于指定切片图在 *x*、 *y*、 *z* 轴所切的位置。
- □ slice(x, y, z, v, sx, sy, sz): 输入参数x, y, z用于指定绘制的三维切片图的x, y, z轴。
- □ slice(..., 'method'): 输入参数 method 用于指定切片图绘制时的内插值法, 'method' 可以设置的参数有: 'linear'(三次线性内插值法, 默认)、'cubic'(三次立方内插 值法)、'nearest'(最近点内插值法)。

【例 3-58】 观察函数在 $-2 \le x \le 2$ 、 $-2 \le y \le 2$ 、 $-2 \le z \le 2$ 上的体积情况。MATLAB 代码如下:

```
% 数据准备
xi=-10:0.5:10;
yi=-10:0.5:10;
[x,y]=meshgrid(xi,yi);
z=sin(sqrt(x.^2+y.^2))./sqrt(x.^2+y.^2);
[x,y,z] = meshgrid(-2:.2:2, -2:.25:2, -2:.16:2);
v = x.*exp(-x.^2-y.^2-z.^2);
xslice = [-1.2,.8,2]; yslice = 2; zslice = [-2,0];
% 绘图
slice(x,y,z,v,xslice,yslice,zslice)
```

MATLAB运行结果如图 3.76 所示。



图 3.76 简单的三维切片图

## 3.3.4 常用三维图形

由于一些特殊的需要,有时可能需要绘制一些具有一定形状的三维图,例如,瀑布图、 柱面图、球体等,下面简单演示这些图形的绘制。

• 119 •

## 1. 瀑布图

在 MATLAB 中, waterfall()函数可以绘制在 x 轴或者 y 轴方向具有流水效果的瀑布图, 其调用格式如下所示。

waterfall(x,y,z): 输入参数 x, y = z 分别用于指定瀑布图的 x 轴、y 轴和 z 轴,同时数 据 z 标识瀑布图的颜色。

【例 3-59】 绘制简单的瀑布图。MATLAB 代码如下:

```
% 数据准备
xi=-10:0.5:10;
yi=-10:0.5:10;
[x,y]=meshgrid(xi,yi);
% 绘图
zx.*x+y.*y
waterfall(x,y, z)
colormap gray
```

MATLAB运行结果如图 3.77 所示。



图 3.77 简单的瀑布图

## 2. 柱面图

在 MATLAB 中, cylinder()函数可生成关于 z 轴旋转对称的柱面体,结合函数 surf()或 mesh()可生成柱面体的三维曲面图,其调用格式如下。

- □ [*x*,*y*,*z*]=cylinder(r): 返回半径为*r*, 高度为1的圆柱体的*x*、*y*、*z*轴的坐标值, 圆柱 一周的分点数为20。
- □ [*x*,*y*,*z*]=cylinder(*r*, *n*): 返回半径为*r*,高度为1的圆柱体的*x*、*y*、*z*轴的坐标值,圆柱一周的分点数为*n*。
- 【例 3-60】 绘制简单的柱面图。MATLAB 代码如下:

```
cylinder
axis square
h = findobj('Type','surface');
set(h,'CData',rand(size(get(h,'CData'))))
```

• 120 •

title('简单柱面图')

MATLAB运行结果如图 3.78 所示。 【例 3-61】 绘制半径变化的柱面图。MATLAB 代码如下:

t = 0:pi/10:2\*pi; figure(1) [X,Y,Z] = cylinder(2+cos(t)); surf(X,Y,Z) axis square title('半径变化的柱面图')

MATLAB运行结果如图 3.79 所示。



#### 3. 球形图

在 MATLAB 中, sphere()函数用于在直角坐标系内绘制球形图, 其调用格式如下。

- □ sphere(*n*): 绘制单位球形图, 球体包含  $n \times n$  个球面。
- □ [x,y,z]=sphere(*n*): 返回含 *n*×*n* 个球面的球体的坐标,结合函数 surf()或 mesh()绘制三维球体曲面图。

【例 3-62】 绘制简单的球形图。MATLAB 代码如下:

```
sphere(50)
axis equal
title('球')
```

MATLAB运行结果如图 3.80 所示。

### 4. 椭球形图

在 MATLAB 中, ellipsoid()函数可生成绘制椭球形图的坐标数据,结合函数 surf()或者 mesh(),即可绘制三维椭球体图,其函数的调用格式如下。

[*x*,*y*,*z*]=ellipsoid(*xc*, *yc*, *zc*, *xr*, *yr*, *zr*, *n*): 输入参数中 *xc*、*yc*、*zc* 为椭球体的球心坐标, *xr*、*yr*、*zr* 为椭球体 3 个半轴的长度, *n* 为椭球体的分点数, 没有参数 *n* 的情况下, 椭球体的分点数默认为 20。

【例 3-63】 绘制简单的椭球形图。MATLAB 代码如下:

• 121 •



## 5. peaks()函数

MATLAB 还有一个 peaks()函数,称为多峰函数,常用于三维曲面的演示。 【例 3-64】 绘制多峰函数三维曲面图形。MATLAB 代码如下:

```
[x,y,z]=peaks(30);
surf(x,y,z);
```

MATLAB运行结果如图 3.82 所示。



图 3.82 多峰函数的三维曲面图形

## 3.3.5 其他三维图形

在介绍二维图形时,曾提到条形图、杆图、饼图和填充图等特殊图形,它们还可以以 三维形式出现,使用的函数分别是 bar3()、stem3()、pie3()和 fill3(),调用格式基本上与其

• 122 •

二维形式相同,这里就不再赘述。bar3()函数绘制三维条形图,stem()函数绘制三维杆状图,pie3()函数绘制三维饼图,fill3()函数等效于三维函数 fill(),可在三维空间内绘制出填充过的多边形。下面简单举几个有趣的例子来绘画三维其他图形。

【例 3-65】 尝试利用 stem3()函数绘制三维火柴杆图。MATLAB 代码如下:

```
% 数据准备
x = rand(1,10);
y = rand(1,10);
z = x.*y;
k = x+y;
% 绘图
figure
subplot(2,1,1)
stem3(x, y, z)
subplot(2,1,2)
%填充
stem3(x, y, k, 'filled')
```

MATLAB运行结果如图 3.83 所示。



图 3.83 三维杆状图

【例 3-66】 绘制如下的三维图形。

- □ 绘制魔方阵的三维条形图。
- □ 以三维杆图形式绘制曲线 y=2sin(x)。
- □ 已知 x=[2347,1827,2043,3025], 绘制饼图。
- □ 用随机的顶点坐标值画出5个黄色三角形。

MATLAB 代码如下:

```
subplot(2,2,1);
% 绘制魔方阵的三维条形图
bar3(magic(4))
subplot(2,2,2);
% 以三维杆图形式绘制曲线 y=2sin(x)
y=2*sin(0:pi/10:2*pi);
stem3(y);
subplot(2,2,3);
%绘制饼图
```

pie3([2347,1827,2043,3025]);
subplot(2,2,4);
%用随机的顶点坐标值画出 5 个黄色三角形
fill3(rand(3,5),rand(3,5),rand(3,5),'y')

MATLAB运行结果如图 3.84 所示。

【例 3-67】 绘制多峰函数的瀑布图和等高线图。MATLAB 代码如下:

```
subplot(1,2,1);
[X,Y,Z]=peaks(30);
waterfall(X,Y,Z)
colormapgray
xlabel('X-axis')
ylabel('Y-axis')
zlabel('Z-axis')
subplot(1,2,2);
contour3(X,Y,Z,12,'k');
```

MATLAB运行结果如图 3.85 所示。



图 3.84 综合绘图示例

图 3.85 多峰函数的瀑布图和等高线图

# 3.4 本章小结

在第 3 章中主要学习的是画基本的二维、三维图形。此章学习后,大家应该对用 MATLAB语言画图有一个基本的认识,并且能够自己画出各种单根二维曲线、多根二维曲 线,会对曲线进行样式的设定,熟练掌握对坐标轴的控制。熟练使用各种常用画图函数。

## 3.5 习题

1. 在 0≤t≤2pi 区间内, 绘制曲线 y=3sint+1。

• 124 •

2. 在同一坐标系里绘制 0≤t≤2pi 区间内的正弦函数、余弦函数、正切函数和余切 函数。

3. 利用 polar()函数绘制轮胎图和笛卡尔心形图。

4. 用不同标度在同一坐标内绘制曲线 y1=0.2e-0.5xcos(4πx) 和 y2=2e-0.5xcos(πx)。

5. 绘制 peak()函数的等高线。

6. 在同一坐标内,分别用不同线型和颜色绘制曲线 y1=0.2e-0.5xcos(4 π x) 和 y2=2e-0.5xcos(πx)。

7. 绘制正弦曲线,设置图形的标题为 y=sin(x),设置 x 轴的标注为 x,设置 y 轴的标 注为 sin(x),并对曲线进行文本标注。

8. 绘制抛物线和三次幂曲线,设置图形的标题、x 轴和 y 轴的标注、设置曲线文本标 准、设置图例标注。

9. 绘制两条随机误差曲线,同时绘制出利用 axis()函数修改成相同比例后的误差曲线。 10. 绘制函数 *y*=e<sup>x</sup>的半对数坐标图形,并绘制函数 *y*=e<sup>x</sup>的对数坐标图形。

11. 有一位研究生,在一年中平均每月的费用为生活费 590 元,资料费 130 元,电话费 45 元,购买衣服 200 元,其他费用 55 元。请以饼图表示出他每月的消费比例,并在饼 图中分离出使用最多的费用和使用最少的费用的切片。

12. 有一位研究生,在一年中平均每月的费用为生活费 590 元,资料费 130 元,电话费 45 元,购买衣服 200 元,其他费用 55 元。请以柱状图表示出他每月的消费比例。

13. 绘制 peak()函数的三维等高线图和三维表面图。

14. 绘制简单的柱面图和球形图。