

2.4.2 文件输入输出命令与函数

目的：便于与其它软件系统进行数据交换

1. 变量保存命令 `save`

格式一： `save`

将工作空间中的所有变量保存在一个名为：“matlab.mat”的二进制格式文件中，该文件可通过load命令来重新装入工作空间。

格式二： `save` 文件名 变量名

将工作空间中指定的“变量名”保存在指定“文件名.mat”的二进制格式文件中。

格式三： `save` 文件名 选项

使用“选项”指定ASCII文件格式，将工作实间中所有变量保存到“文件名”所指定的文件中。（例8位、16位ASCII）

2. 变量调入命令load:

功能：从磁盘文件中重新调入变量内容到工作空间。

格式一：load

将保存在“matlab.mat”文件中的所有变量调入到工作空间。

格式二：load 文件名

从“文件名.mat”中调入变量，可给出全部路径。

3. 文件开关命令 fopen

功能：打开文件或获得打开文件信息。

格式一：f_id=fopen(文件名, ‘允许模式’)

以‘允许模式’指定的模式打开“文件名”所指定的文件，
返回文件标识f_id 0;

允许模式 可以是下列几个字符串之一：

‘r’： 打开文件进行读（缺省型式）；

‘w’： 删除已存在文件中的内容或生成一个新文件，打开进行写操作；

‘a’： 打开一个已存在的文件或生成并打开一个新文件，进行写操作，
在文件末尾添加数据；

格式二： [f_id , message]=fopen (文件名, ‘允许模式’ , 格式)

功能：用指定的数据“格式”打开数据文件，返回文件标识和打开文件信息两个参数。用‘格式’指定“数据格式”（如‘native’ ‘s’）。。。

如果fopen成功打开文件，则返回文件标识f_id 为0， message内容为空，如果不能成功打开，则返回f_id值为-1， message中返回一个有助于判断错误类型的字符串。

有三个值是预先定义的：

- 0：表示标准输入，一直处于打开读入状态。
- 1：表示标准输出，一直处于打开追加状态。
- 2：表示标准错误，一直处于打开追加状态。

4. 文件关闭命令 **fclose**

功能：关闭一个或多个已打开的文件。

格式一： status=fclose(f_id)

关闭指定文件，返回0表示成功，返回-1表示失败。

格式二 status=fclose(‘all’)

功能：关闭所有文件，返回0表示成功，返回-1表示失败。

5. 按二进制保存数据命令 **fread**
6. 按二进制保存数据命令 **fwrite**
7. 按照指定格式保存数据命令 **fprintf**
8. 按照指定格式读取数据命令 **fscanf**
9. 图像数据读取 **imread**

格式: **A=imread** (文件名, ‘图像文件格式’)

功能: 将文件名指定的图像文件读入A中, A为无符号8位整数。

10. 图像数据写入文件函数**imwrite**

格式: **imwrite** (A, 文件名, ‘图像文件格式’)

功能: 将变量A中的图像写入文件名指定的文件中。

。 。 。 。 。

详细命令及命令功能通过**help iofun**命令获取;

或者找

“百度”!

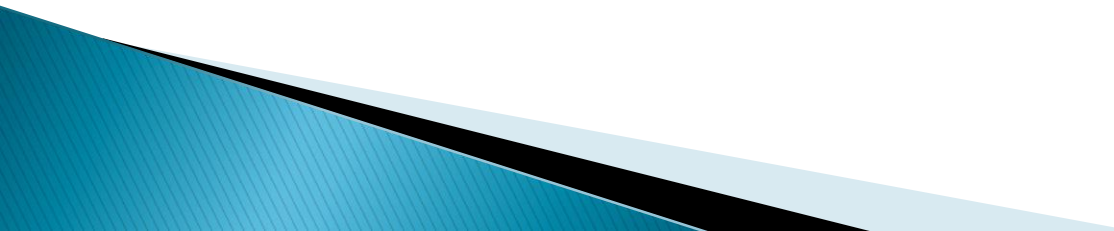
2.4 M文件及程序调试

2.5.1: 主程序文件和函数文件

MATLAB的**M文件**有两类：主程序文件和函数文件。

我们将原本要在MATLAB环境下直接输入的语句，放在一个以 .m为后缀的文件中，这一文件就称为**主程序文件**。有了主程序文件，可直接在MATLAB中输入主程序文件名(不含后缀)，这时MATLAB会打开这一M文件，并依次执行文件中的每一条语句，这与在MATLAB中直接输入语句的结果完全一致。

另一类M文件是**函数文件**，它的第一行必须是函数定义行。
函数文件一般由五部分构成：

- 函数定义行。
 - H1行。
 - 函数帮助文本。
 - 函数体。
 - 注释。
- 

例如，函数文件mean.m的内容为

```
function y=mean(x)
```

函数定义行

```
% MEAN Average or mean value.
```

H1行

```
% For vectors, MEAN(X) is the mean value of X.
```

```
% For matrices, MEAN(X) is a row vector.
```

```
% containing the mean value of each column.
```

函数帮助文本

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

```
if m==1
```

```
    m=n;    %xxx
```

注释

```
end
```

```
y=sum(x)/m;
```

函数体

1. 函数定义行

```
function y=mean(x)
```

其中，`function`为函数定义的关键字，`mean`为函数名，`y`为输出变量，`x`为输入变量。

当函数具有多个输出变量时，则以方括号括起；当函数具有多个输入变量时，则直接用圆括号括起。例如，`function [x, y, z]=sphere(theta, phi, rho)`。当函数不含输出变量时，则直接略去输出部分或采用空方括号表示，例如，`function printresults(x)`或`function []=printresults(x)`。

所有在函数中使用和生成的变量都为**局部变量**(除非利用global语句定义), 这些变量值只能通过输入和输出变量进行传递。因此, **在调用函数时应通过输入变量将参数传递给函数; 函数调用返回时也应通过输出变量将运算结果传递给函数调用者**; 在函数中产生的其它变量在返回时被全部清除。

2. H1行

在脚本和函数文件中，以%开头的行称为注释行，即%之后的字符不被MATLAB执行。

在函数文件中，其第二行一般是注释行，这一行称为H1行，实际上它是帮助文本中的第一行。H1行不仅可以由help function_name命令显示，而且lookfor命令只在H1行内进行搜索，因此这一行内容提供了这个函数的重要信息。

3. 函数帮助文本

这部分内容是以%开头的帮助文本，它用来比较详细地说明这一函数。当在MATLAB下输入`help function_name`时，可显示出H1行和函数帮助文本。这部分文本从H1行开始，到第一个非%开头的行结束。

4. 函数体

函数体是完成指定功能的语句实体，它可采用任何可用的MATLAB命令组成，它还可以包含MATLAB提供的函数和用户自己设计的M函数。

5. 注释

注释行是以%开头的行，它可出现在函数文件的任意位置，也可以加在语句行之后，以便对语句行进行注释。

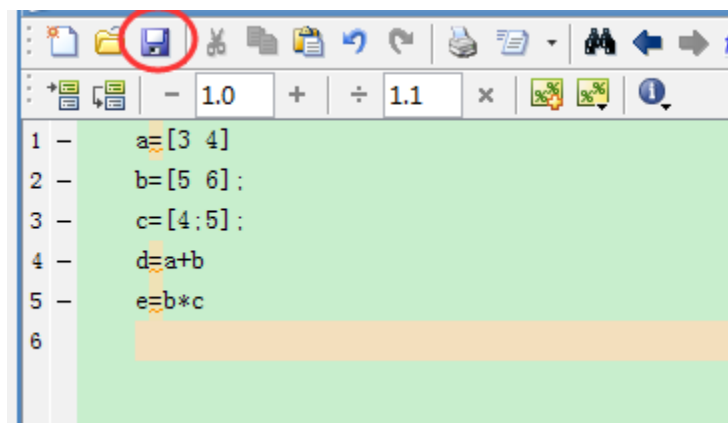
在函数文件中，除了函数定义行和函数体之外，其它部分都是可以省略的，不是必需的。但作为一个函数，为了提高函数的可用性，应加上H1行和函数帮助文本，为了提高函数的可读性，应加上适当的注释。

主程序文件

输下面的代码：

```
a=[3 4]
b=[5 6];
c=[4;5]
d=a+b
e=b*c
```

点击保存为test文件，至此我们完成了一个m文件建立。



点击run，即可运行。

运行结果如下。上面代码中末尾含分号(;)表示运行结果不显示。矩阵中的;表示换行的意思。

```
a =
     3     4

c =
     4
     5

d =
     8    10

e =
    50

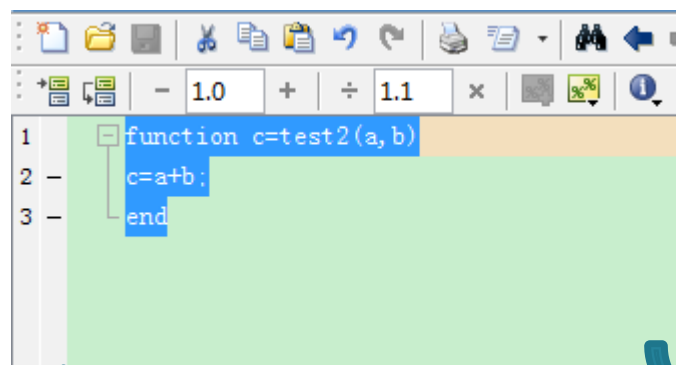
fx >>
```

函数文件

m函数文件的写法，同样新建一个空白文件，然后输入下面的代码

```
function c=test2(a,b)
c=a+b;
end
```

其中function是关键，表示这个文件是m函数文件，a,b是输入参数，c为输出参数



点击保存为test2.m，注意这里文件名必须的和函数名相同。可以看出matlab中m文件和函数文件的图标是有区别的，带fx的是函数文件。

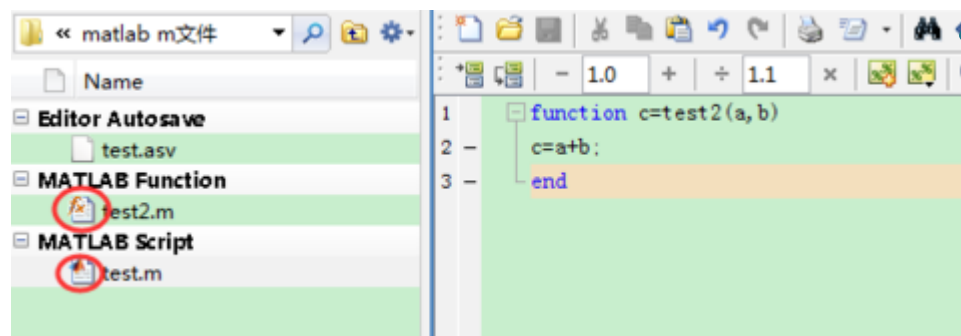
然后在命令窗口输入

```
test2(3,4)
```

就可以调用函数了。注意函数文件一般不能直接点运行按钮运行，除非函数没有输入参数。

```
>> test2(3,4)
ans =
     7

fx >>
```



脚本文件（主程序文件）和函数文件比较

	脚 本 文 件	函 数 文 件
定义行	无需定义行	必须有定义行
输入/输出变量	无	有
数据传送	直接访问基本工作空间中的所有变量	通过输入变量获得输入数据，通过输出变量提交结果
编程方法	直接选取 MATLAB 语句	精心设计完成指定功能
用途	重复操作	MATLAB 功能扩展

将函数文件去掉其第一行的定义行就转变成了主程序文件，但这样一来，原先在函数内部使用的局部变量也就变成了基本工作空间中的变量，这会带来以下几个问题：

- 基本工作空间与函数文件中同名的变量会引起冲突。
- 使基本工作空间中变量数急剧增加，造成内存紧张。
- 编程时要细心考虑各个函数文件所用到的变量。

这些问题在函数文件中不复存在，MATLAB通过实参与形参一一对应的方式来实现函数的调用，这极大地方便了程序设计。

例如，编写出求取平均值的脚本文件stat1.m，再编写出求取标准差的函数文件stat2.m。程序如下：

主程序文件stat1.m

```
% 主程序文件
% 求阵列x的平均值和标准差
[m,n]=size(x);
if m==1
    m=n;
end
s1=sum(x); s2=sum(x.^2);
mean1=s1/m;
stdev=sqrt(s2/m-mean1.^2);
```

函数文件stat2.m

```
function [mean1,stdev]=stat2(x)
% 求阵列x的平均值和标准差
% 调用格式为
% [mean,stdev]=stat2(x)
[m,n]=size(x);
if m==1
    m=n;
end
s1=sum(x); s2=sum(x.^2);
mean1=s1/m;
stdev=sqrt(s2/m-mean1.^2);
```


然后，在MATLAB下执行这两个文件，从而对主程序文件和函数文件有一个基本的了解。在MATLAB中输入

```
>> clear all % 清除内存
```

```
>> x=rand(4,4)+2;
```

```
>> stat1
```

执行后检查基本工作空间中的变量情况:

>>whos

Name	Size	Bytes	Class
m	1x1	8	double array
mean1	1x4	32	double array
n	1x1	8	double array
s1	1x4	32	double array
s2	1x4	32	double array
stdev	1x4	32	double array
x	4x4	128	double array

Grand total is 34 elements using 272 bytes

这说明，在主程序文件中产生的所有变量都保存在基本工作空间。检查执行结果

```
>> disp([mean1; stdev])
```

```
2.5685  2.5321  2.6684  2.5605
```

```
0.2587  0.3359  0.1513  0.2888
```

另一方面，通过函数文件来进行同样的操作，这时输入

```
>> clear m n s1 s2 mean1 stdev
```

```
[m1,st]=stat2(x);
```

执行后，同样检查基本工作空间的变量情况：

```
>> whos
```

Name	Size	Bytes	Class
m1	1x4	32	double array
st	1x4	32	double array
x	4x4	128	double array

Grand total is 24 elements using 192 bytes

这说明，在基本工作空间中，除了原本产生的x矩阵，调用函数stat2.m后，只增加了由函数返回的结果。通过disp([m1;st])可得到与stat1.m相同的结果。

2.5.2 子函数

在函数文件中可包含多个函数，其中第一个函数称为主函数，其函数名和文件名相同，它可由其它M文件或基本工作空间引用。在M函数文件中的其它函数称为**子函数**，它只能由这一个M函数文件中的主函数或其它子函数引用。

每个子函数也由函数定义行开始，紧跟其后的语句为函数体。各种子函数的次序任意，但主函数必须是第一个函数。

例如，我们编写了一个求均值和中值的函数mmval.m，它包含了两个子函数：

```
function [avg,med] = mmval(u)      %主函数  
  
%NEWSTATS Find mean and median with internal functions.  
  
n = length(u);  
  
avg = mean(u,n);  
  
med = median(u,n);
```

```
function a = mean(v,n)            %子函数  
  
%Calculate average.  
  
a = sum(v)/n;
```

```
function m = median(v,n)           %子函数
```

```
%Calculate median.
```

```
w = sort(v);
```

```
if rem(n,2) == 1
```

```
m = w((n+1)/2);
```

```
else
```

```
m = (w(n/2)+w(n/2+1))/2;
```

```
end
```



在同一个M文件内的子函数，也只能访问由输入/输出变量传递的变量、声明为全局变量的变量和函数内部的局部变量，这一点与主函数相同。

当在M文件中调用另一个函数时，MATLAB首先在该文件内检查是否为子函数，然后检查是否为私人函数。最后在搜索路经上检查标准的M文件。由于首先检查的是子函数，因此可使用与已有M文件同名的子函数。如上例中的mean，虽然存在mean.m函数，但在调用mmval函数时，不会调用mean.m文件，而只执行mmval.m中的子函数mean。注意，在同一个M文件中，函数名应该惟一。

2.6 MATLAB绘图

2.6.1 基本图形和图形操作

1. plot

功能：绘制二维图形(曲线)。

格式：

plot(y)

plot(x1,y1,...)

plot(...,'PropertyName', PropertyValue,...)

plot(x1, y1, LineSpec,...) h=plot(...)

说明：

当 y 为实向量时，`plot (y)`以 y 的序号作为 x 轴，以向量 y 的值作为 y 轴绘制出二维曲线；

当 y 为复向量时，则`plot (y)`相当于`plot(real(x), imag(y))`，即 y 的实部为 x 轴，虚部为 y 轴。在后面几种格式中，虚部均被忽略。

`plot(x1, y1, ...)`可按 $(x1, y1)$ ， $(x2, y2)$ ，...成对绘制出曲线，而且在同一张图上以不同颜色显示。

如果 x_n 或 y_n 之一为矩阵，则取矩阵的行或列与另一个向量构成数据对绘制出曲线。

◆ **plot(x1, y1, LineSpec, ...)**可绘制出所有由三元组 $(x_n, y_n, \text{LineSpec})$ 指定的曲线，其中LineSpec用于指定线型、标记和线颜色。有关LineSpec的内容可参见下面的注释1。

◆ **h=plot(...)**可在绘制出图形的同时，得到图形(曲线)的句柄向量，每条曲线对应于一个句柄值。

表2.20 LineSpec指定的线型、颜色和标记

符号	线型或颜色	符号	颜色	符号	标记	符号	标记
-	实线	c	青色	+	加号	^	向上尖三角
--	虚线	r	红色	o	圆圈	v	向下尖三角
:	点线	g	绿色	*	星号	<	向左尖三角
-.	点划线	b	蓝色	.	黑点	>	向右尖三角
		w	白色	x	叉号	pentagram	五角星
y	黄色	k	黑色	square	正方形	hexagram	六角星
m	洋红色			diamond	菱形		

例2.6 绘制出显函数方程在区间内的曲线。

- ▶ 在命令窗口输入如下语句
- ▶ `x = -pi:pi/10:pi; %以pi/10为步长`
- ▶ `y = tan(sin(x)) - sin(tan(x)); %求出各点的函数值`
- ▶ `plot(x,y,'--rs','LineWidth',2,...`
- ▶ `'MarkerEdgeColor','k',...`
- ▶ `'MarkerFaceColor','g',...`
- ▶ `'MarkerSize',10)`
- ▶ 得到的图形如图2.6所示，并标上了线型属性。

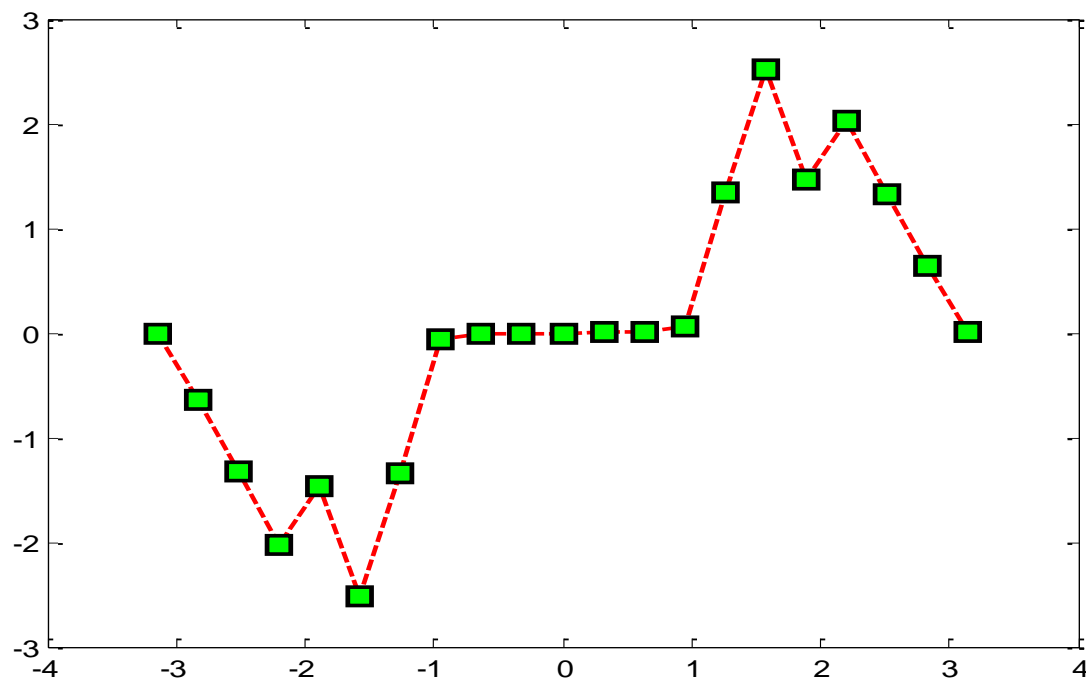


图2.6 以 $\pi/10$ 为步长的曲线 $\tan(\sin(x)) - \sin(\tan(x))$

- ▶ 由于plot（）函数只将给定点用直线连接起来，因此步长选得过大，曲线将看似一折线，因此将上述绘图语句步长改为0.05，并在及两个子区间内加密自变量选择点，即将上述语句修改为
- ▶ `x=[-pi:0.05:-1.8,-1.801:0.001:-1.2,-1.2:0.05:1.2,1.2:0.001:1.8,1.81:0.05:pi];`
- ▶ `%以步长方式构造自变量`
- ▶ `y = tan(sin(x)) - sin(tan(x));`
- ▶ `plot(x,y)`
- ▶ 得到曲线如图3.2所示。

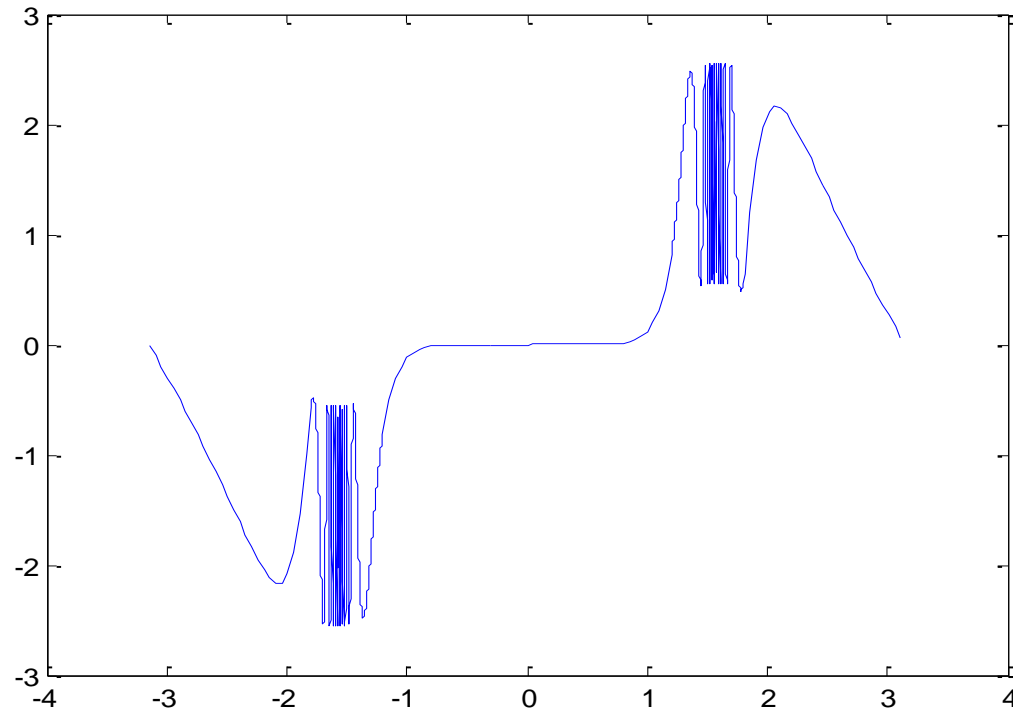


图3.2 细化步长后的曲线 $\tan(\sin(x)) - \sin(\tan(x))$

2.6.2 设置轴的形式与刻度设置

在绘制图形时，用户可以使用函数axis和set对坐标轴的刻度范围进行重新设定，其调用格式如下：

(1) 函数axis

`axis([xmin xmax ymin ymax zmin zmax])`

axis函数功能丰富，常用的用法还有：

`axis equal` 纵、横坐标轴采用等长刻度

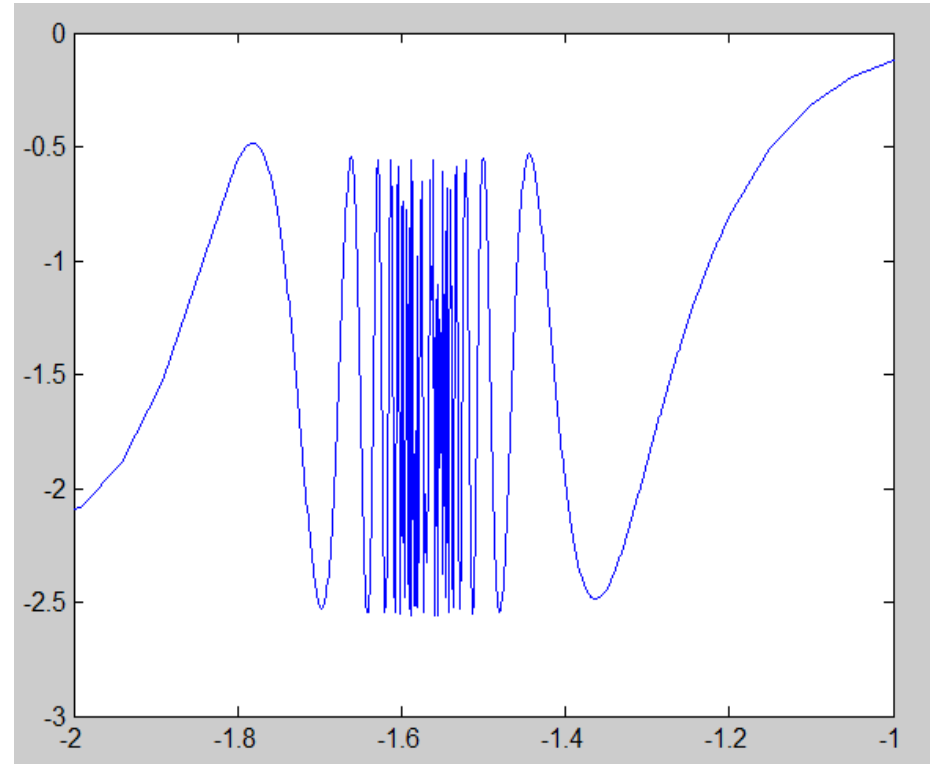
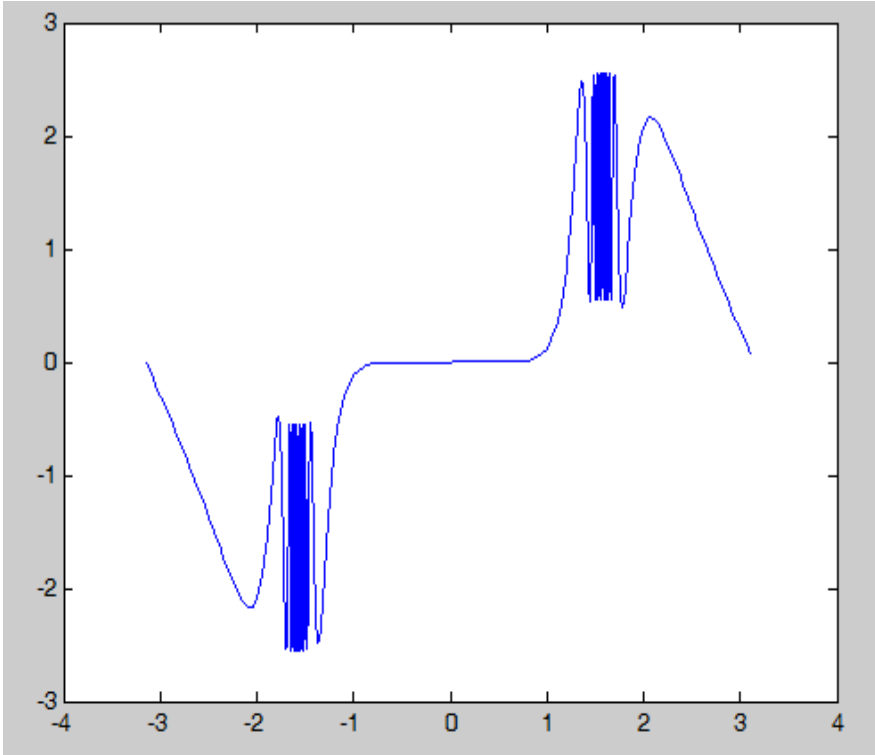
`axis square` 产生正方形坐标系(缺省为矩形)

`axis auto` 使用缺省设置

`axis off` 取消坐标轴

`axis on` 显示坐标轴

- ▶ `x=[-pi:0.05:-1.8,-1.801:0.001:-1.2,-
1.2:0.05:1.2,1.2:0.001:1.8,1.81:0.05:pi];`
- ▶ `% 以步长方式构造自变量`
- ▶ `y = tan(sin(x)) - sin(tan(x));`
- ▶ `plot(x,y)`
- ▶ `axis([-2,-1,-3,0])`



(2) 函数set

- ▶ `set(gca,'xtick',标示向量)` , `set(gca,'ytick',标示向量)` , 按照标示向量设置x, y轴的刻度标示;
- ▶ `set(gca,'xticklabel','字符串|字符串...')` , `set(gca,'yticklabel','字符串|字符串...')` , 按照字符串设置x, y轴的刻度标志

▶ 例2.4 给正弦曲线设置刻度标示。

- ▶ MATLAB程序如下:
- ▶ `t=0:0.05:7;`
`plot(t,sin(t))`
- ▶ `set(gca,'xtick',[0 1.4 2.64 5 6.28])`
- ▶ `set(gca,'xticklabel','0|1.4|half|5|one')`

程序运行结果如图3.5所示。

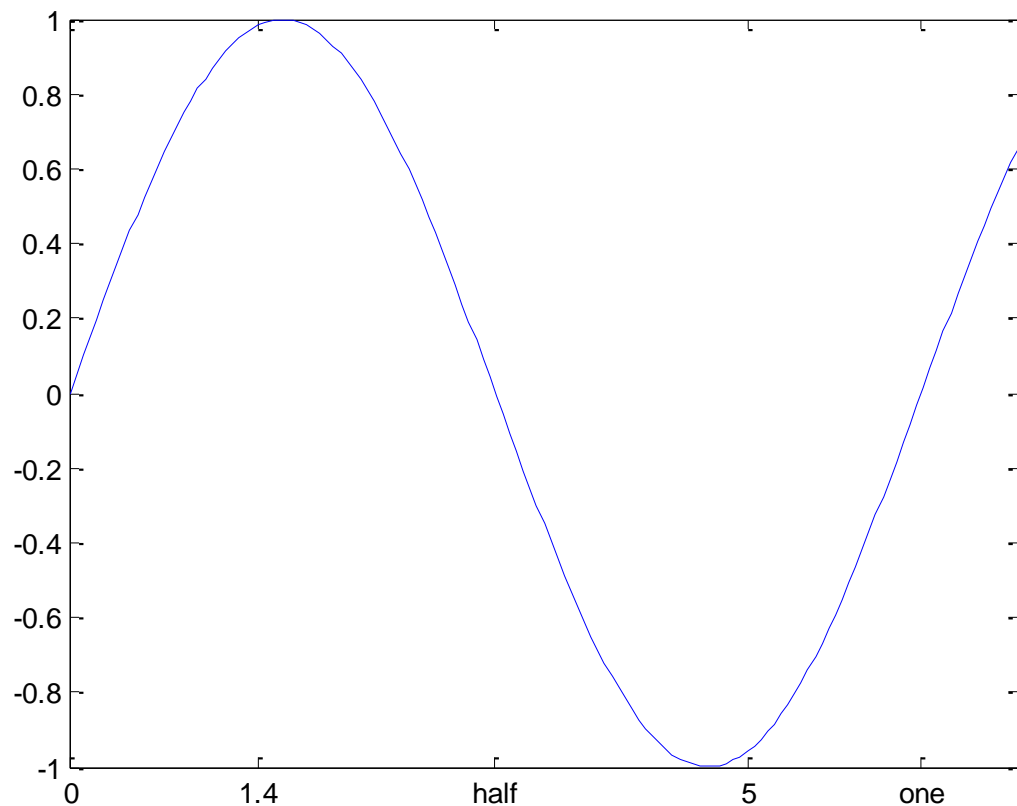


图3.5 设置刻度标示的正弦曲线

2.6.3 图形的标注、网格及图例说明

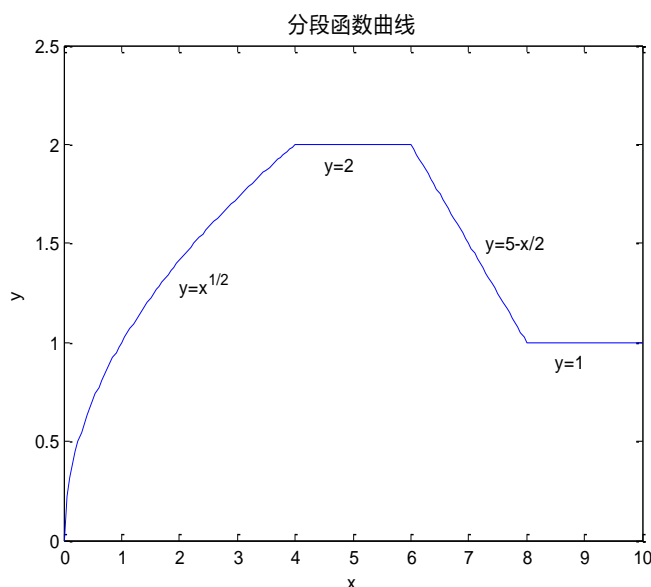
- ▶ 绘制图形时，可以对图形加上一些说明，添加网格和图例等，基本函数及其调用格式如下：
 - ▶ **(1) 添加图形标题命令title**
 - ▶ title ('string')，在当前坐标系的顶部加一个文本串string，作为图形的标题。
 - ▶ title('text','Property1',PropertyValue1,'Property2',PropertyValue2,...),设置标题名属性。
 - ▶ **(2) 添加坐标轴标志函数xlabel、ylabel、zlabel**
 - ▶ xlabel('string'), ylabel ('string'), zlabel ('string'),给当前X轴或Y轴或Z轴标注文本文注。

- ▶ xlabel ('text','Property1',
PropertyValue1, 'Property2',
PropertyValue2,...)
- ▶ 或ylabel ('text','Property1',
PropertyValue1, 'Property2',
PropertyValue2,...)
- ▶ 或xlabel ('text','Property1',
PropertyValue1, 'Property2',
PropertyValue2,...)
- ▶ 对X轴、Y轴、Z轴分别进行属性设置。

- ▶ (3) 文本注释函数 `text`, `gtext`
- ▶ `text(x,y,'string')`, 在二维图形 (x,y) 位置处标注文本注释 'string'
- ▶ `text(x,y,z,'string')`, 在三维图形 (x,y,z) 位置处标注文本注释 'string'
- ▶ `gtext('string')` 一用鼠标拖动来确定标注文字 'string' 的位置。

例3.5 绘制分段函数曲线并添加图形标注。

$$f(x) = \begin{cases} \sqrt{x}, 0 \leq x < 4 \\ 2, 4 \leq x < 6 \\ 5 - x/2, 6 \leq x < 8 \\ 1, x \geq 8 \end{cases}$$



MATLAB程序如下：

```
clc
close all
clear
x=0:0.05:10;
y=zeros(1,length(x));
for n=1:length(x)
    if x(n)>=8
        y(n)=1;
    elseif x(n)>=6
        y(n)=5-x(n)/2
    elseif x(n)>=4
        y(n)=2
    else
        y(n)=sqrt(x(n))
    end
end
End
```

```
elseif x(n)>=4
    y(n)=2
else
    y(n)=sqrt(x(n))
end
end
plot(x,y)
axis([0 10 0 2.5])
title('分段函数曲线');
xlabel('x')
ylabel('y')
text(2,1.3,'y=x^{1/2}');
text(4.5,1.9,'y=2');
text(7.3,1.5,'y=5-x/2');
text(8.5,0.9,'y=1');
```

图2.6 绘制分段函数曲线并添加图形标注

(4) 创建图形窗命令figure

- ▶ figure, 打开不同的图形窗口, 绘制不同的图形
- ▶ figure('PropertyName',PropertyValue,...), 创建具有特定属性值的图形窗口
- ▶ figure(h), 创建或显示句柄h定义的图形窗口, 如果h不是整数, 则返回错误。
- ▶ h = figure(...)

- ▶ **（5）设置网格线命令grid**
- ▶ grid on (off),对当前做标图加上网格线或撤销网格线。若直接调用grid命令即可设置或撤销网格线
- ▶ **（6）保持图形窗口内容命令hold**
- ▶ hold on （ off ） 保持当前图形窗口内容命令（解除保持）。若直接调用hold命令即可保持或解除保持当前图形窗口内容。

- ▶ 例2.6分别绘制正余弦曲线并绘制标题，添加或去除栅格。
- ▶ MATLAB程序如下：
- ▶ `close all`
- ▶ `clc`
- ▶ `clear t=[0:pi/20:9*pi]; %定义时间范围`
- ▶ `figure(1) %建立图形窗口1`

- ▶ `plot(t,sin(t),'r:*)`
- ▶ `grid on` %在所画出的图形坐标中添加栅格，注意用在`plot`之后
- ▶ `text(pi,0,' \leftarrow sin(\pi)','FontSize',18)`
- ▶ `title('添加栅格的正弦曲线')`
- ▶ `xlabel('x')`
- ▶ `ylabel('sint')`

- ▶ `figure(2)`
- ▶ `plot(t,cos(t))`
- ▶ `grid on`
- ▶ `pause`
- ▶ `grid off` %删除栅格

- ▶ `text(pi,0,' \leftarrow \cos(\pi)', 'FontSize', 18)`
- ▶ `title('去除栅格的余弦曲线')`
- ▶ `xlabel('x')`
- ▶ `ylabel('cost')`
- ▶ 程序执行结果如图3.7， 3.8所示。

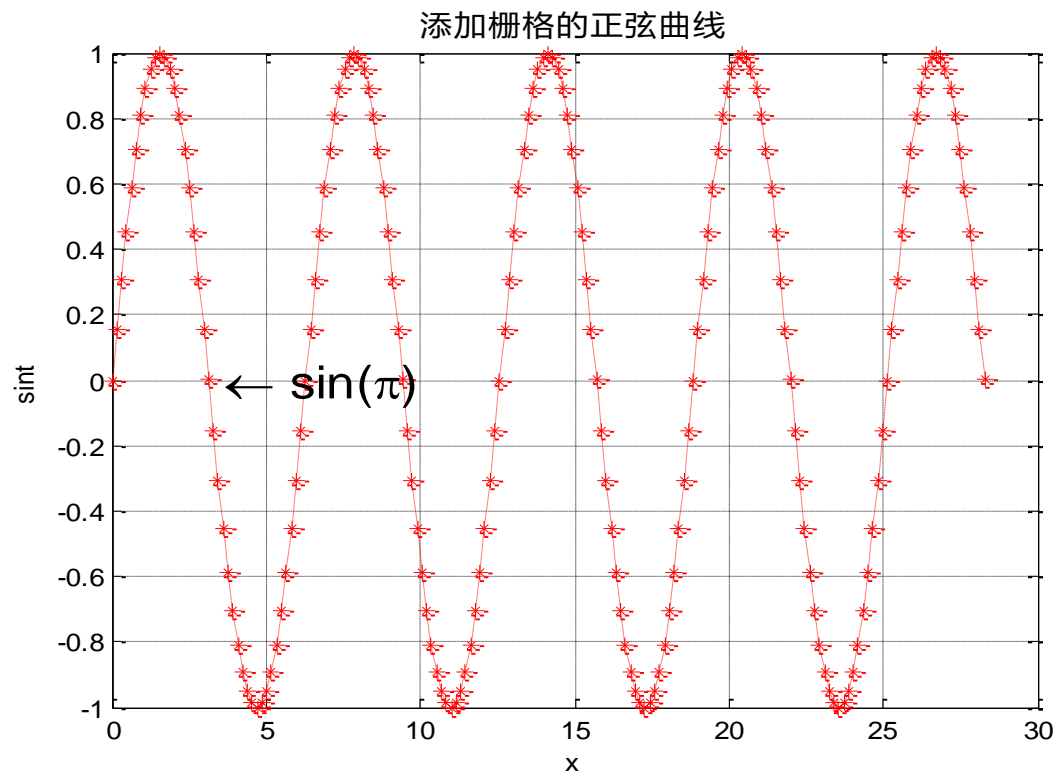


图2.7 正弦曲线

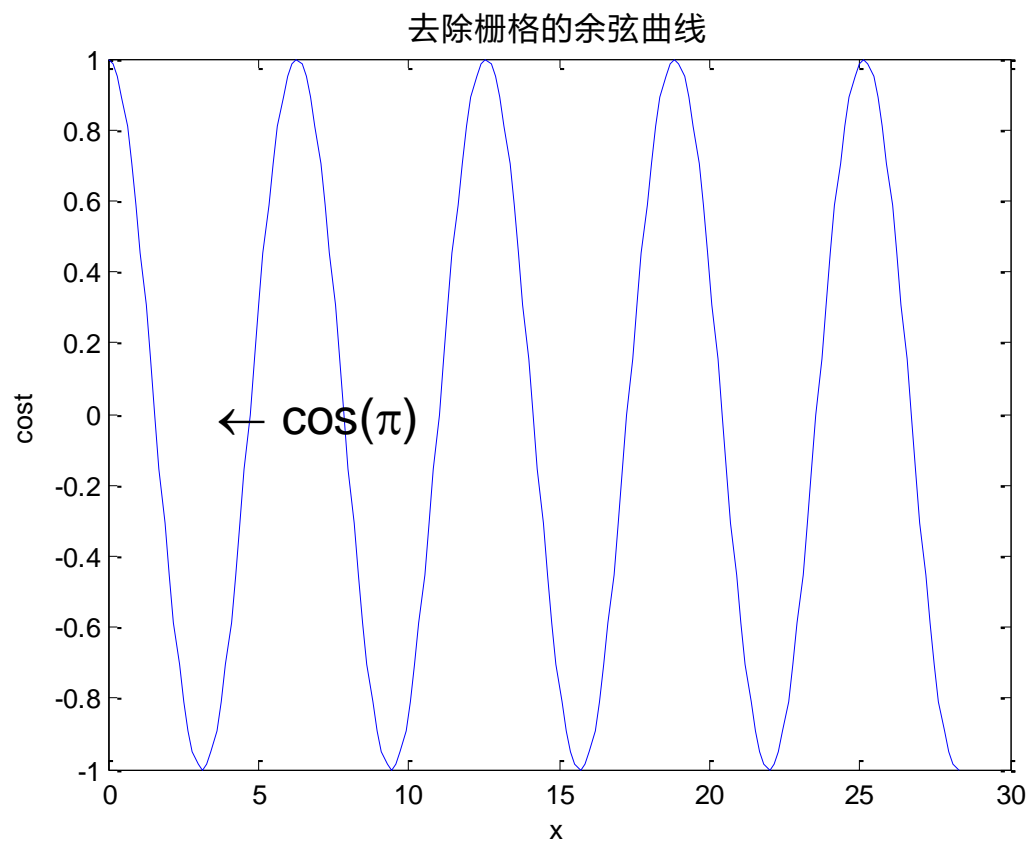


图2.8 余弦曲线

- ▶ (7) 图形标注函数legend
- ▶ legend('string1','string2',...), 在当前图中添加图例;
- ▶ legend(...,pos) 由pos 确定图例标注的位置, 可以返回给句柄: h= legend(...,pos)
- ▶ 参数字符串的含义:
- ▶ pos = -1 放置图例在轴边界的右边;
- ▶ pos = 0 放置图例在轴边界里;
- ▶ pos = 1 放置图例在轴边界里右上角 (为默认设置);
- ▶ pos = 2 放置图例在轴边界里左上角;
- ▶ pos = 3 放置图例在轴边界里左下角;
- ▶ pos = 4 放置图例在轴边界里右下角;
- ▶ Legend off , 撤销当前坐标图上的图例。

- ▶ 例3.7 给正弦余弦曲线图形添加图例
- ▶ MATLAB程序如下：
- ▶ $x = -\pi:\pi/20:\pi;$
- ▶ `plot(x,cos(x),'-ro',x,sin(x),'-.b')`
- ▶ `h = legend('cos','sin',2);`
- ▶ 程序执行结果如图2.9所示。

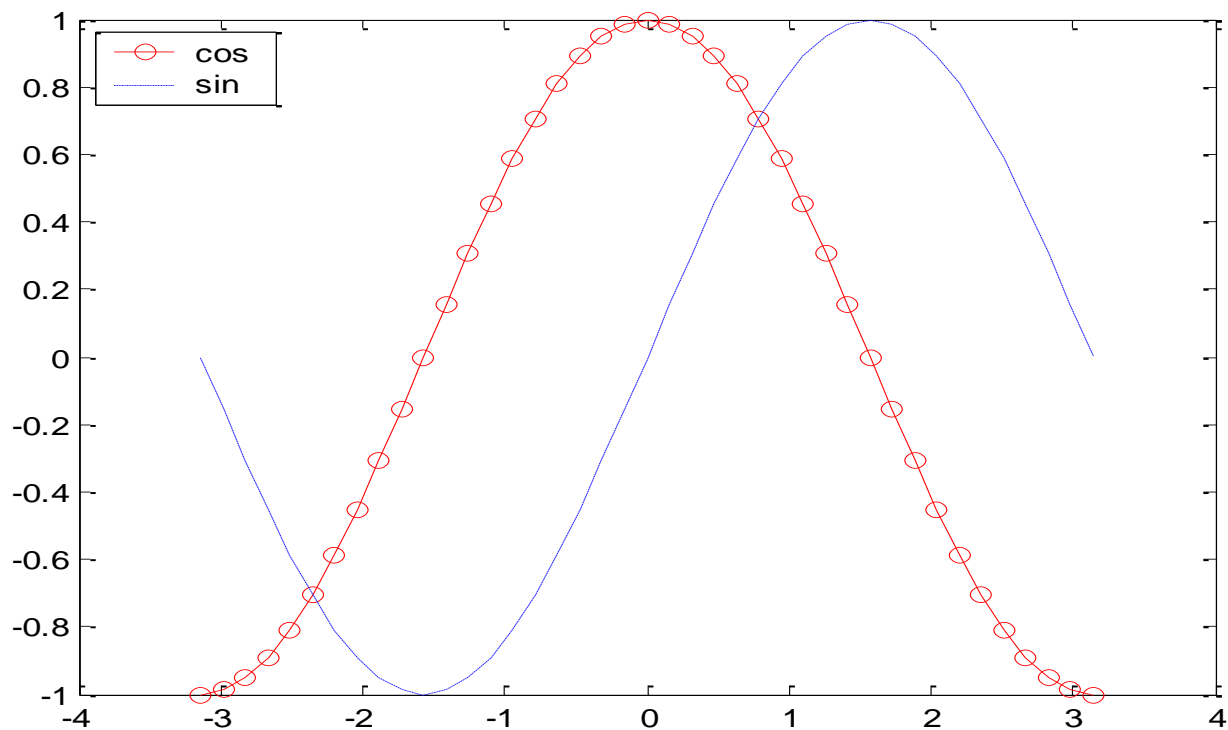


图2.9 添加图例的正余弦曲线

2.6.4 绘制二维图形的其他函数

- ▶ 1. 分割图形显示窗口命令 **subplot** (m,n,k)

subplot(m,n,p) 按平铺位置建立子图坐标系，将一个图形窗口分成 $m*n$ 个子图窗口，从左至右，从上往下第 p 个子图形窗口。

- ▶ 例2.8 将一个图形窗口分割成4个子图窗口，并且分别绘制不同函数曲线。
- ▶ MATLAB程序如下：
- ▶ `subplot(2,2,1),fplot('humps',[0 1])`
- ▶ `subplot(2,2,2)`

- ▶ `fplot('abs(exp(-j*x*(0:9))*ones(10,1))',[0 2*pi])`
- ▶ `subplot(2,2,3)`
- ▶ `fplot('[tan(x),sin(x),cos(x)]',2*pi*[-1 1 -1 1])`
- ▶ `subplot(2,2,4)`
- ▶ `fplot('sin(1./x)',[0.01 0.1],1e-3)`
- ▶ 程序执行结果如图2.60所示。

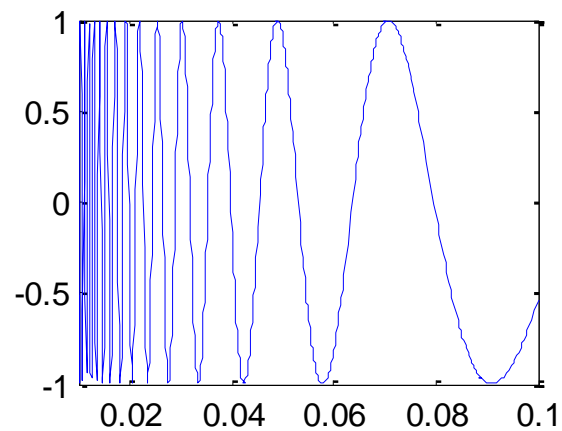
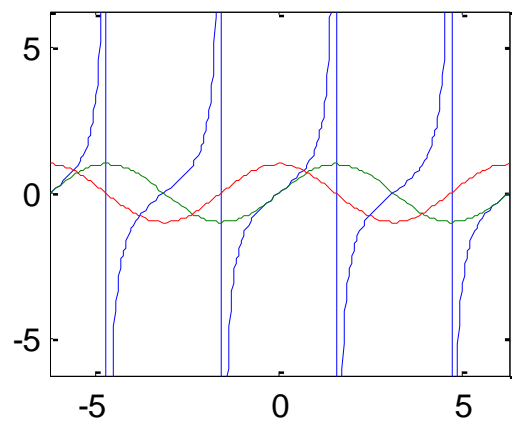
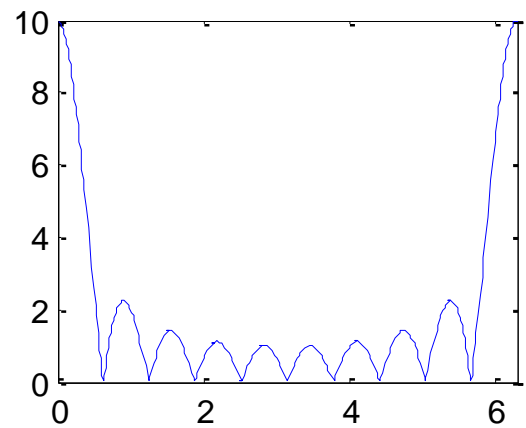
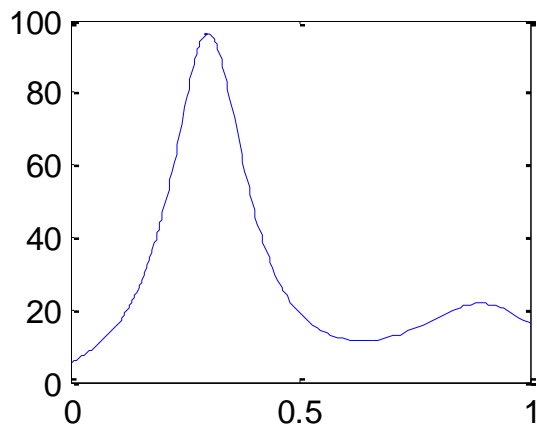


图2.60 subplot函数和fplot函数的运用

2、双纵坐标绘图命令plotyy

- ▶ `plotyy(X1,Y1,X2,Y2)`, 设有两个纵坐标Y1, Y2, 以便绘制两个y尺度不同的变量, 但x仍用同一个比例尺。

- ▶ 例3.9 在同一图形窗口按不同纵坐标绘制曲线。
- ▶ MATLAB程序如下：
- ▶ $x = 0:0.01:20;$
- ▶ $y1 = 200*\exp(-0.05*x).*\sin(x);$
- ▶ $y2 = 0.8*\exp(-0.5*x).*\sin(10*x);$
- ▶ $\text{plotyy}(x,y1,x,y2);$
- ▶ 程序执行结果如图2.61所示。从图中可以看出，左纵坐标的幅度范围为 $[-200,200]$ ，对应 $y1$ ，而右纵坐标的幅度范围为 $[-0.8, 0.8]$ ，对应 $y2$ 。

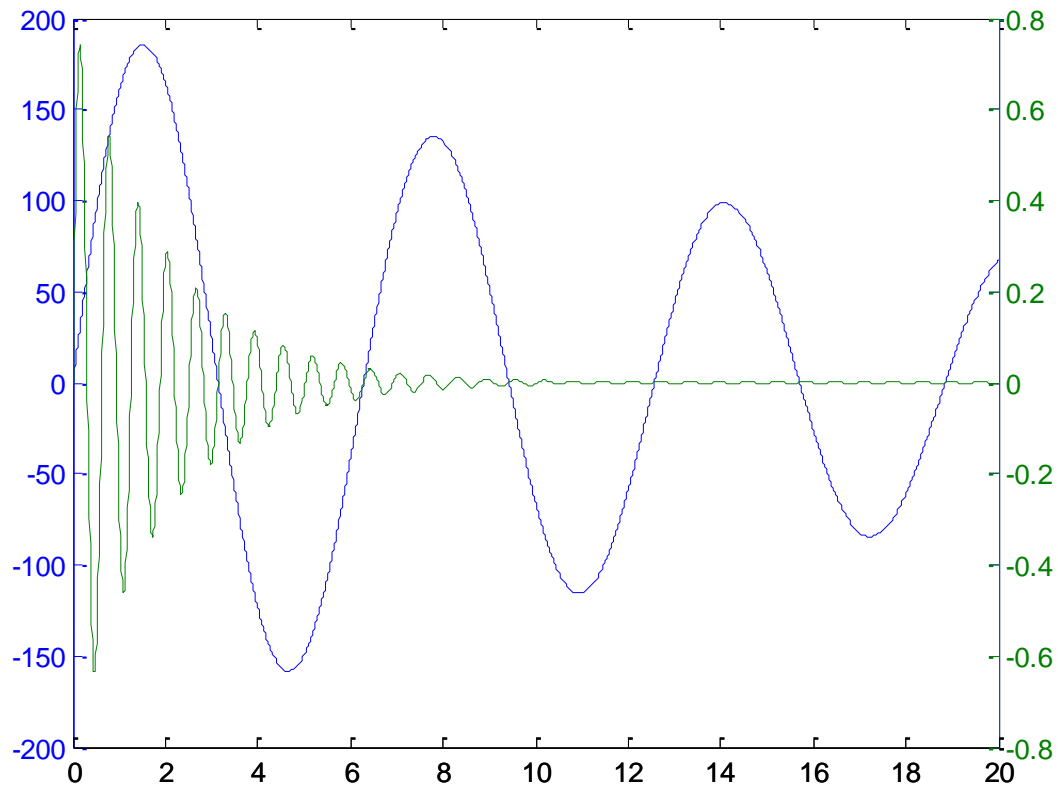


图2.61 曲线 $200 \cdot \exp(-0.05 \cdot x) \cdot \sin(x)$ 和 $200 \cdot \exp(-0.05 \cdot x) \cdot \sin(x)$;

3.其他形式的线性直角坐标图

- ▶ 在线性直角坐标系中，其他形式的图形有条形图、阶梯图、杆图和填充图等，所采用的函数分别是：
- ▶ `bar(x,y,选项)`
- ▶ `stairs(x,y,选项)`
- ▶ `stem(x,y,选项)`
- ▶ `fill(x1,y1,选项1,x2,y2,选项2,...)`

- ▶ 例2.60 分别以条形图、填充图、阶梯图和杆图形式绘制曲线 $y=2e^{-0.5x}$ 。
- ▶ MATLAB程序如下：
 - ▶ `x=0:0.35:7;`
 - ▶ `y=2*exp(-0.5*x);`
 - ▶ `subplot(2,2,1);bar(x,y,'g');`
 - ▶ `title('bar(x,y,"g")');axis([0,7,0,2]);`
 - ▶ `subplot(2,2,2);fill(x,y,'r');`
 - ▶ `title('fill(x,y,"r")');axis([0,7,0,2]);`
 - ▶ `subplot(2,2,3);stairs(x,y,'b');`
 - ▶ `title('stairs(x,y,"b")');axis([0,7,0,2]);`
 - ▶ `subplot(2,2,4);stem(x,y,'k');`
 - ▶ `title('stem(x,y,"k")');axis([0,7,0,2]);`

程序运行结果如图2.62所示。

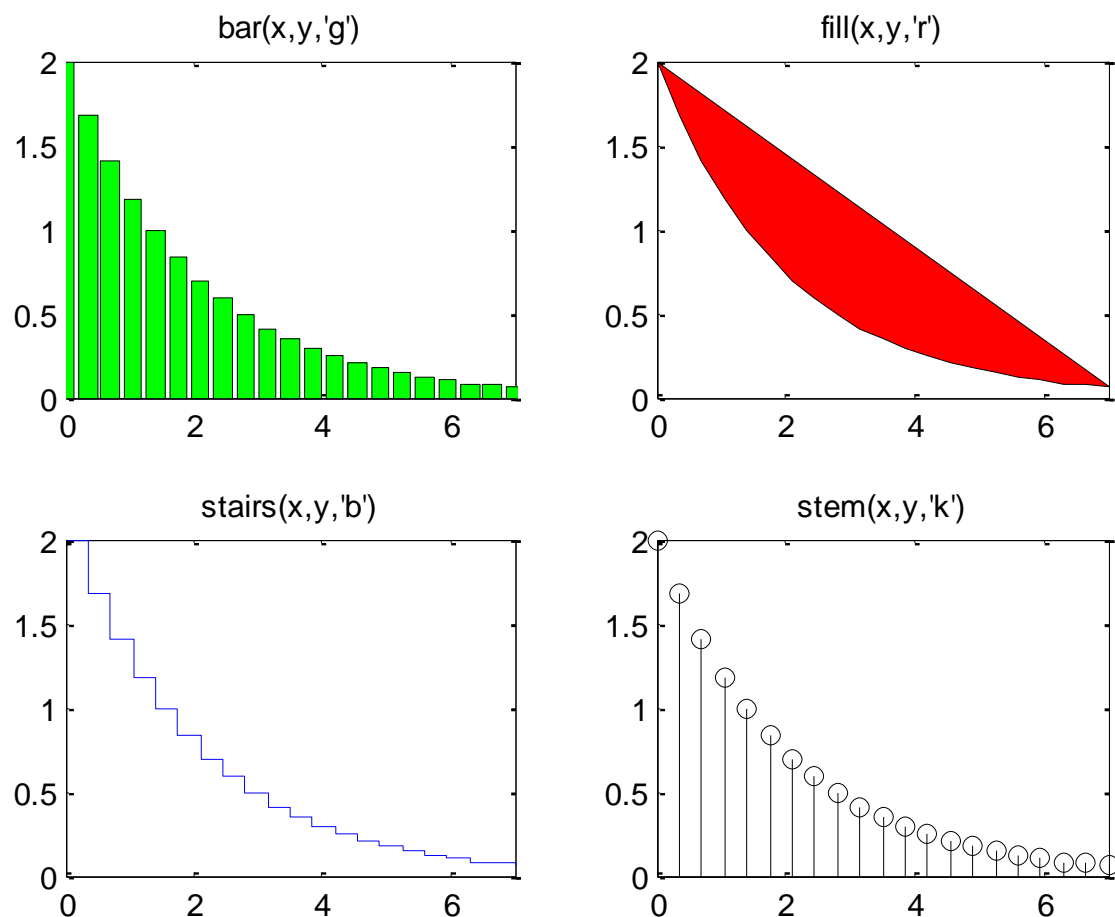


图2.62 几种不同形式的二维图形

3.2 三维图形

- ▶ 3.2.1 绘制三维曲线的基本函数
- ▶ 最基本的三维图形函数为`plot3`, 可用来绘制三维曲线, 其调用格式为:
- ▶ `plot3(x,y,z)` —— x,y,z 是长度相同的向量;
- ▶ `plot3(X,Y,Z)` —— X,Y,Z 是维数相同的矩阵, 以 X 、 Y 、 Z 对应列元素绘制三维曲线, 曲线条数等于矩阵列数;
- ▶ `plot3(x1,y1,z1,'s1', x2,y2,z2,'s2', ...)`, 选项的定义和`plot`函数相同。

- ▶ 例2.64 画一个三维的螺旋线。
- ▶ MATLAB程序如下：
- ▶ $t = 0:\pi/50:10*\pi;$
- ▶ `plot3(sin(t),cos(t),t)`
- ▶ `grid on`
- ▶ `axis square`
- ▶ 程序运行结果如图2.66所示。

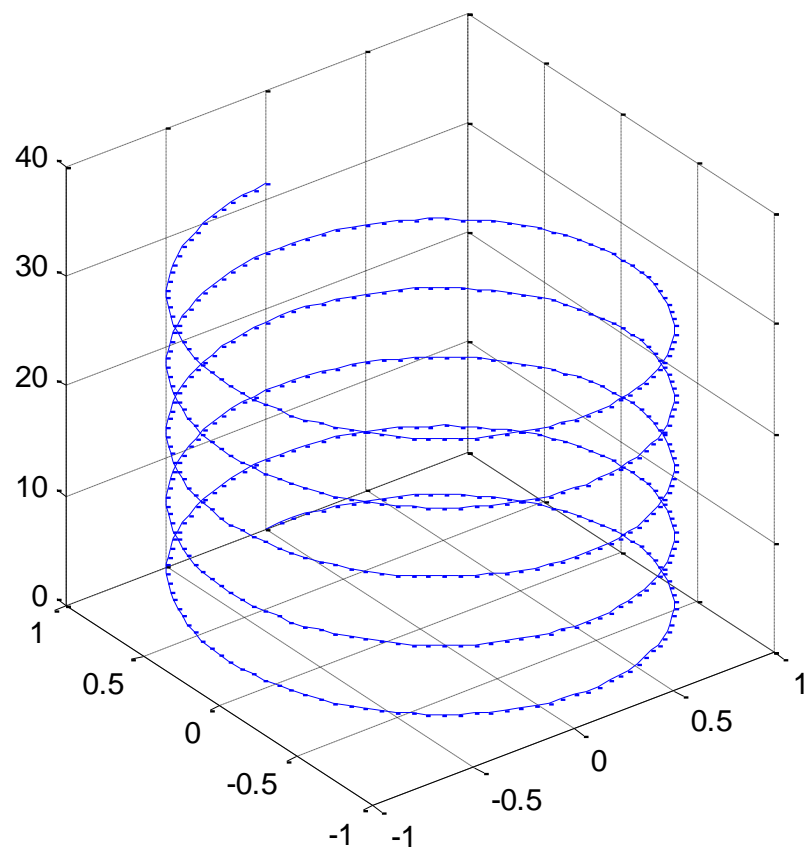


图2.66 三维的螺旋线

3.2.2 三维曲面

- ▶ 1. 平面网格坐标矩阵的生成函数 `meshgrid`
- ▶ 绘制 $z=f(x,y)$ 所代表的三维曲面图，先要在 xy 平面选定一矩形区域，假定矩形区域 $D=[a,b]*[c,d]$ ，然后将 $[a,b]$ 在 x 方向分成 m 份，将 $[c,d]$ 在 y 方向分成 n 份，由各划分点分别作平行于两坐标轴的直线，将区域 D 分成 $m*n$ 个小矩形，生成代表每一个小矩形顶点坐标的平面网格坐标矩阵。生成网格坐标矩阵的函数为 `meshgrid`，其调用格式如下：

- ▶ $[X,Y] = \text{meshgrid}(x,y)$ 转换向量 x , y 为一个特定的矩阵 X,Y , 矩阵 X 的每一行
- ▶ 向量 x , 行数等于向量 y 的元素个数, 矩阵 Y 的每一列都是向量 y , 列数等于向量 x 的元素个数。因此
- ▶ $x=a:dx:b$;
- ▶ $y=c:dy:d$;
- ▶ $[X,Y]=\text{meshgrid}(x,y)$;
- ▶ 等同于
- ▶ $x=a:dx:b$;
 $y=(c:dy:d)'; X=\text{ones}(\text{size}(y))*x; Y=y*\text{ones}(\text{size}(x))$;

- ▶ 例2.65 已知 $6 < x < 30$, $15 < y < 36$, 求不定方程 $2x + 5y = 126$ 的整数解。
- ▶ MATLAB程序如下:
- ▶ $x = 7:29$; $y = 16:35$;
- ▶ $[x,y] = \text{meshgrid}(x,y)$; %在 $[5,29] \times [14,35]$ 区域生成网格坐标
- ▶ $z = 2 * x + 5 * y$;
- ▶ $k = \text{find}(z == 126)$; %找出解的位置

▶ $x1 = x(k), y1 = y(k)$ %输出对应位置的x,y即方程的解

▶ 程序执行结果如下：

▶ $x1 =$

▶ 8

▶ 13

▶ 18

▶ 23

▶ $y1 =$

▶ 22

▶ 20

▶ 18

▶ 16

2. 绘制三维曲面的函数surf函数和mesh函数

- ▶ MATLAB提供了mesh函数和surf函数来绘制三维曲面图。Mesh函数用于绘制三维网格图，surf函数用于绘制三维曲面图，各线条之间的曲面用颜色填充，其调用格式如下：
- ▶ mesh(X,Y,Z)，根据矩阵X、Y、Z绘制彩色的三维网线图。X、Y、Z中对应的元素为三维空间上的点，点与点之间用线连接。其中网线的颜色随着网点高度的改变而改变。

- ▶ 在绘制二元函数 $z=f(x,y)$ 的三维网格图时，首先应通过 $[X,Y]=\text{meshgrid}(x,y)$ 语句，在 X - Y 平面上建立网格坐标，然后利用 X 和 Y 计算每一个网格点上的 Z 坐标的大小，该坐标就定义了曲面上的点。最后由 $\text{mesh}(Z)$ 命令完成三维网线图的绘制。
- ▶ $\text{mesh}(x,y,Z)$, n 维向量 x ， m 维向量 y 和 $m*n$ 矩阵 Z 绘制网线图。节点的坐标为 $(x(j),y(i),Z(i,j))$ ，网线的颜色随着网点高度的改变而改变。
- ▶ $\text{mesh}(Z)$ 由数值对 $(i,j,Z(i,j))$ 实现绘图。

- ▶ `meshc`和`meshz`函数除可绘制三维网线图外，同时还能分别绘制三维网线图的等高线图 and 它下面的幕帘线。其调用格式和`mesh`相同。
- ▶ `surf(X,Y,Z,C)`，绘制由四个矩阵所指定的带色参数的网状表面图。颜色范围由`C`指定。
- ▶ `surf(X,Y,Z)`，设`C`为与`Z`相等，则颜色与网的高度成正比。

- ▶ 例2.66 绘制MATLAB自带的函数peaks的具有等高线的三维网格线图和网状表面图。
- ▶ MATLAB程序如下：
- ▶ `[X,Y] = meshgrid(-3:.1 25:3);`
- ▶ `Z = peaks(X,Y);`
- ▶ `meshc(X,Y,Z);`
- ▶ `axis([-3 3 -3 3 -10 5])`
- ▶ `figure`
- ▶ `surfc(X,Y,Z);`
- ▶ 程序运行结果如图2.67，图2.68所示。

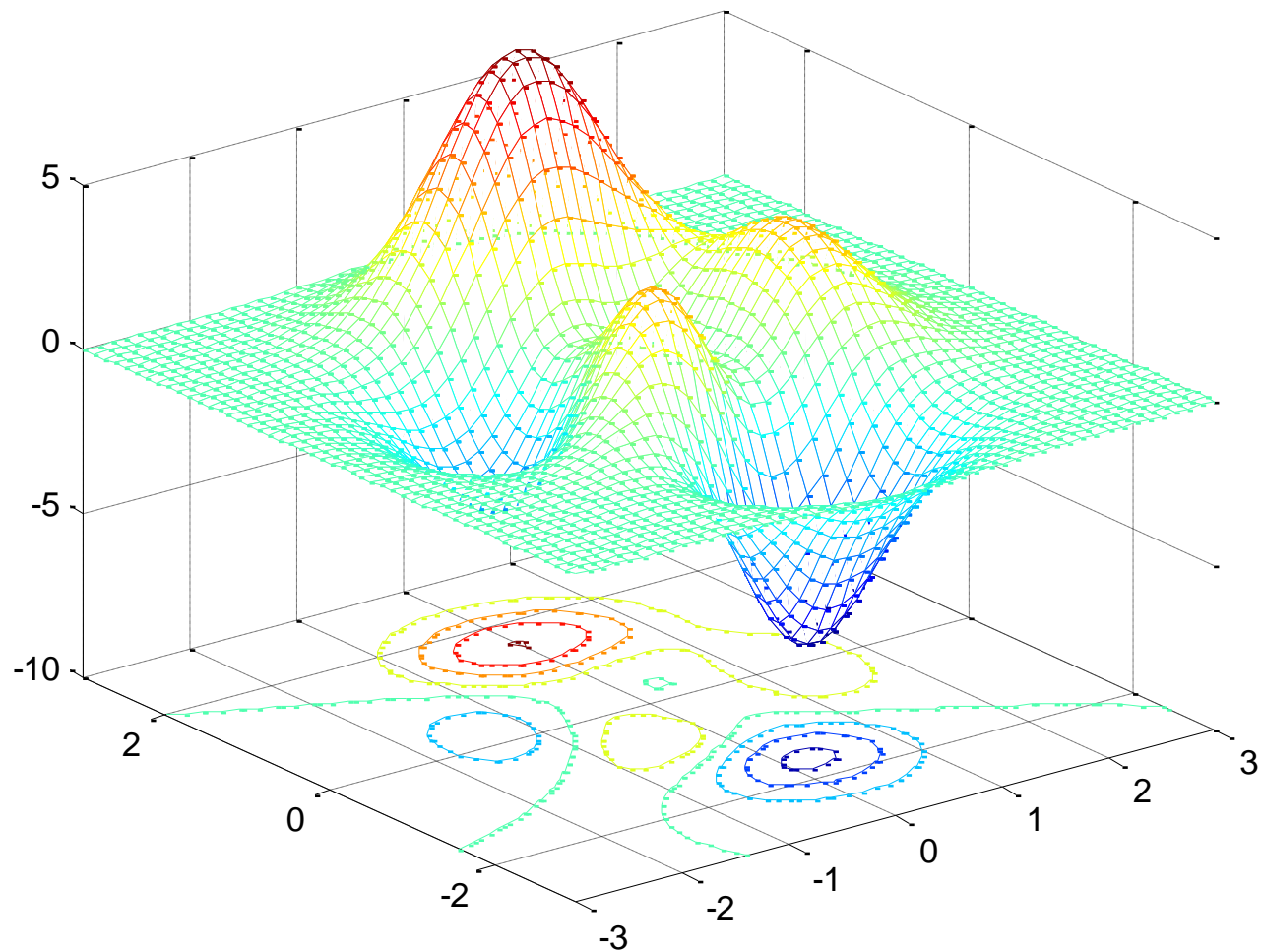


图2.67 函数peaks的具有等高线的三维网格线图

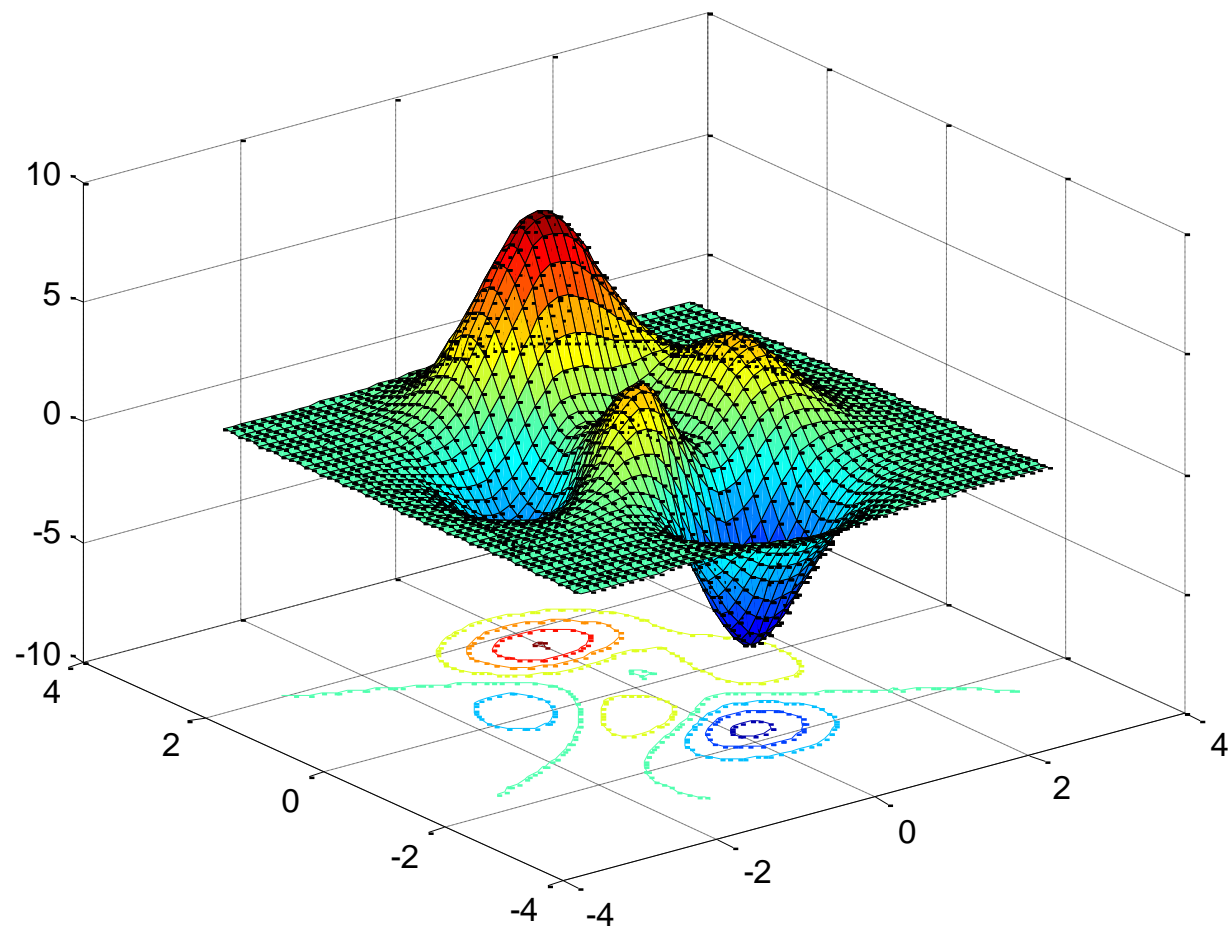


图2.68 函数peaks的具有等高线的网状表面图

- ▶ 例2.67 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。
- ▶ MATLAB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)')`
- ▶ 程序运行结果如图2.69所示。

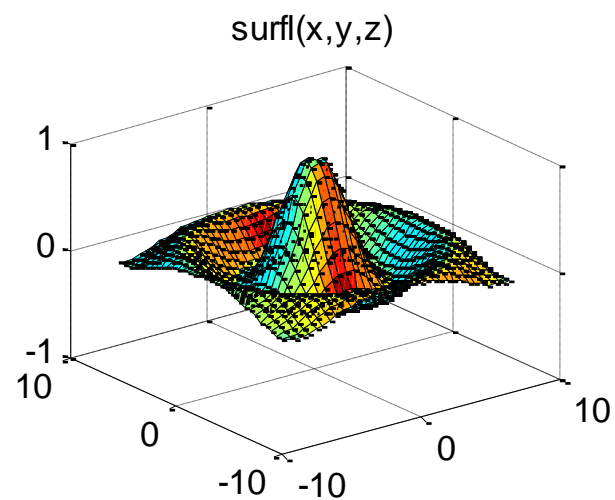
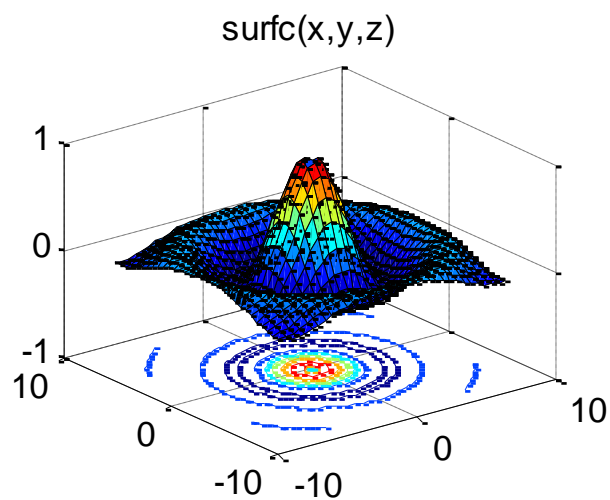
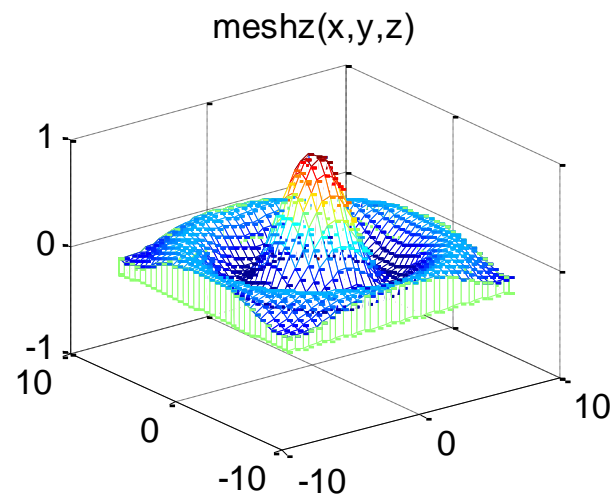
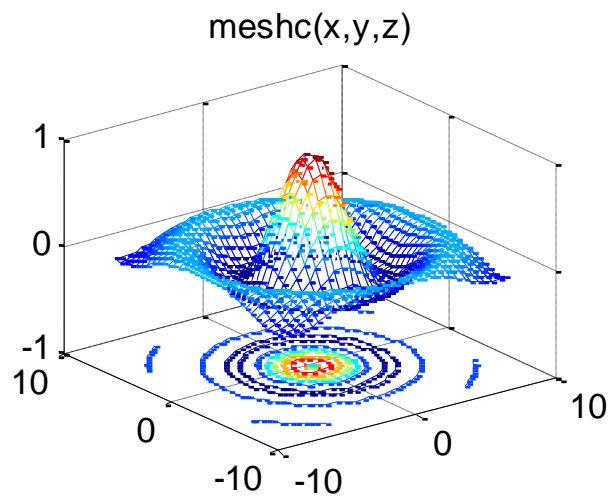


图2.69 函数 $\sin(\sqrt{x.^2+y.^2})./\sqrt{x.^2+y.^2+eps}$ 的四种三维曲面图