1.4　三维图形

1.4.1　三维曲线、面填色命令

命令 1　**comet3**

功能　三维空间中的彗星图。彗星图为一个三维的动画图像，彗星头（一个小圆圈）沿着数据指定的轨道前进，彗星体为跟在彗星头后面的一段痕迹，彗星轨道为整个函数所画的实曲线。注意一点的是，该彗星轨迹的显示模式 EraseMode 为 none，所以用户不能打印出彗星轨迹（只能得到一个小圆圈），且若用户调整窗口大小，则彗星会消失。

用法　comet3(z) 用向量 z 中的数据显示一个三维彗星

comet3(x,y,z) 显示一个彗星通过数据 x，y，z 确定的三维曲线。

comet3(x,y,z,p) 指定彗星体的长度为：p*length（y）。

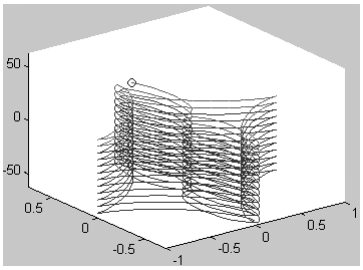

图 1-24

例 1-24

```
>>t = -20*pi:pi/50:20*pi;

>>comet3((cos(2*t).^2).*sin(t),(sin(2*t).^2).*cos(t),t);
```

图形的结果为图 1-24。

命令 2　**fill3**

功能　用指定的颜色填充三维多边形。阴影类型为平面型和 Gouraud 型。

用法　fill3(X,Y,Z,C) 填充由参数 x，y 和 z 确定多边形。若 x，y 或 z 为矩阵，fill3 生成 n 个多边形，其中 n 为矩阵的列数。在必要的时候，fill3 会自动连接最后一个节点和第一个节点。以便能形成封闭的多边形。参数 c 指定颜色，这儿 c 为引用当前色图的下标向量或矩阵。若 c 为行向量，则 c 的维数必须等于 x 的列数和 y 的列数，若 c 为列向量，则 c 的维数必须等于矩阵 x 的行数和 y 的行数。

fill3(X,Y,Z,ColorSpec) 用指定的颜色 ColorSpec 填充由 x，y 和 z 确定的多边形。

fill3(X1,Y1,Z1,C1,X2,Y2,Z2,C2,…) 对多边形的不同区域用不同的颜色进行填充。

fill3(…,'PropertyName',PropertyValue) 允许用户对特定的 patch 属性进行设置。

h = fill3(…) 返回 patch 图形对象的句柄向量，每一块（patch）对应一个句柄。

运算规则：

1．若 X，Y，Z 为同型的矩阵，fill3 生成 X，Y，Z 中相同位置的元素确定的顶点，每一列生成一个多边形。

2．若只有 X，Y 或 Z 为矩阵，则 fill3 由列向量参数生成可用的同型矩阵。

3．若用户对填充的颜色指定为 ColorSpec，则 fill3 生成阴影类型为 flat-shaded 的多边形，且设置块（patch）的属性 FaceColor 为 RGB 颜色形式的矩阵。

4．若用户用矩阵 C 指定颜色，命令 fill3 通过坐标轴属性 Clim 来调整 C 中的元素，在引用当前色图之前，用于指定颜色坐标轴的参数比例。

5．若参数 C 为一行向量，命令 fill3 生成带平面阴影（flat-shaded）的多边形，同时设置补片对象的面颜色（FaceColor）属性为 flat。向量 c 中的每一元素成为每一补片对象的颜

色数据（CData）属性的值。

6．若参数 C 为一矩阵，命令 fill3 生成带内插颜色的多边形，同时设置多边形补片对象的 FaceColor 属性为 interp。命令 fill3 采用对多边形顶点色图的下标指定的颜色采用线性内插算法，同时多边形的颜色采用对顶点颜色用内插算法得到的颜色。矩阵 C 的每一列元素变成对应补片对象的 Cdata 属性值。

7．若参数 C 为一列向量，命令 fill3 先复制 C 的元素，使之成为所需维数的矩阵，再按上面的方法 6 进行计算。

例 1-25

```
>>X = 10*rand(4);Y=10*rand(4);Z=10*rand(4);

>>C = rand(4);

>>fill3(X,Y,Z,C)
```

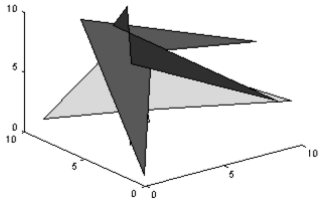

图 1-25

图形结果可能为图 1-25。

1.4.2 三维图形等高线

命令 1 **contour**

功能 曲面的等高线图

用法 contour(z) 把矩阵 z 中的值作为一个二维函数的值，等高曲线是一个平面的曲线，平面的高度 v 是 Matlab 自动取的；

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。效果同上；

contour(z,n) 画出 n 条等高线；

contour(x,y,z,n) 画出 n 条等高线；

contour(z,v) 在指定的高度 v 上画出等高线；

contour(x,y,z,v) 同上；

[c,h] = contour(…) 返回如同 contourc 命令描述的等高矩阵 c 和线句柄或块句柄列向量 h，这些可作为 clabel 命令的输入参量，每条线对应一个句柄，句柄中的 userdata 属性包含每条等高线的高度值；

contour(…,'linespec') 因为等高线是以当前的色图中的颜色画的，且是作为块对象处理的，即等高线是一般的线条，我们可象画普通线条一样，可以指定等高线的颜色或者线形。

例 1-26

```
>>contour(peaks(40))
```

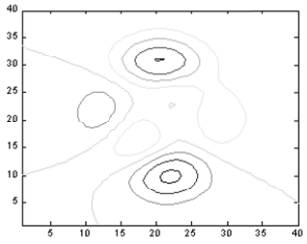

图 1-26

图形结果为图 1-26。

命令 2 **clabel**

功能 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。

用法 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。用户用鼠标左键或空格键在最接近指定的位置上放置标签，用键盘上的回车键结束该操作。当然会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。此时标签的放置的位置是随机的。

clabel(C,v) 在给定的位置 v 上显示标签

clabel(C,'manual') 允许用户通过鼠标来给等高线贴标签

例 1-27

```
>>[x,y] = meshgrid(-2:.2:2);

>>z = x.*y.*exp(-x.^2-y.^2);

>>[C,h] = contour(x,y,z);

>>clabel(C,h);
```

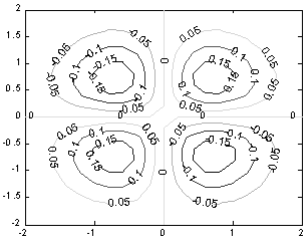

图 1-27

图形结果为图 1-27。

命令 3 **contourc**

功能 低级等高线图形计算命令。该命令计算等高线矩阵 c，该矩阵可用于命令 contour，contour3 和 contourf 等。矩阵 z 中的数值确定平面上的等高线高度值，等高线的计算结果用由矩阵 z 维数决定的间隔的宽度。

用法 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。

C = contourc(Z,n) 在矩阵 z 中计算出 n 个高度的等高线。

C = contourc(Z,v) 在矩阵 z 中计算出给定高度向量 v 上计算等高线，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 中，参量 x 与 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。

命令 4 **contour3**

功能 三维空间等高线图。该命令生成一个定义在矩形格栅上曲面的三维等高线图。

用法 contour3(Z) 画出三维空间角度观看矩阵 z 的等高线图，其中 z 的元素被认为是距离 xy 平面的高度，矩阵 z 至少为 2*2 阶的。等高线的条数与高度是自动选择的。若[m，n]=size（z），则 x 轴的范围为[1：n]，y 轴的范围为[1：m]。

contour3(Z,n) 画出由矩阵 z 确定的 n 条等高线的三维图。

contour3(Z,v) 在参量 v 指定的高度上画出三维等高线，当然等高线条数与向量 v 的维数相同；若想只画一条高度为 h 的等高线，输入：contour3(Z,[h,h])

contour3(X,Y,Z)、contour3(X,Y,Z,n)、contour3(X,Y,Z,v) 用 X 与 Y 定义 x-轴与 y-轴的范围。若 X 为矩阵，则 X(1,:)定义 x-轴的范围；若 Y 为矩阵，则 Y(:,1)定义 y-轴的范围；若 X 与 Y 同时为矩阵，则它们必须同型。不论为哪种使

用形式，所起的作用与命令 surf 相同。若 X 或 Y 有不规则的间距，contour3 还是使用规则的间距计算等高线，然后将数据转变给 X 或 Y。

contour3(…,LineSpec) 用参量 LineSpec 指定的线型与颜色画等高线。

[C,h] = contour3(…) 画出图形，同时返回与命令contourc中相同的等高线矩阵C，包含所有图形对象的句柄向量h；除非没有指定LineSpec参数，contour3 将生成patch图形对象，且当前的colormap属性与caxis属性将控制颜色的显示。不论使用何种形式，该命令都生成line图形对象。

例 1-28

```
>>[X,Y] = meshgrid([-2:.25:2]);

>>Z = X.*exp(-X.^2-Y.^2);

>>contour3(X,Y,Z,30)
```

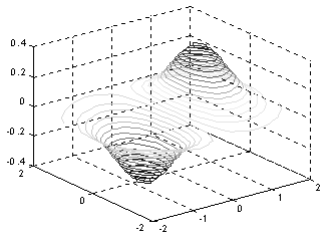

图 1-28

图形结果为图 1-28。

命令 5 **contourf**

功能 填充二维等高线图。即先画出不同等高线，然后相邻的等高线之间用同一颜色进行填充。填充用的颜色决定于当前的色图颜色。

用法 contourf(Z) 矩阵 z 的等高线图，其中 z 理解成距平面的高度。Z 至少为 2*2 阶的。等高线的条数与高度是自动选择的。

contourf(Z,n) 画出矩阵 z 的 n 条高度不同的等高线。

contourf(Z,v) 画出矩阵 z 的、由 v 指定的高度的等高线图。

contourf(X,Y,Z)、contourf(X,Y,Z,n)、contourf(X,Y,Z,v) 画出矩阵 z 的等高线图，其中 X 与 Y 用于指定 x-轴与 y-轴的范围。若 X 与 Y 为矩阵，则必须与 Z 同型。若 X 或 Y 有不规则的间距，contour3 还是使用规则的间距计算等高线，然后将数据转变给 X 或 Y。

[C,h,CF] = contourf(…)画出图形，同时返回与命令 contourc 中相同的等高线矩阵 C，C 也可被命令 clabel 使用；返回包含 patch 图形对象的句柄向量 h；返回一用于填充用的矩阵 CF。

例 1-29

```
>>contourf(peaks(30),20);

>>colormap gray
```

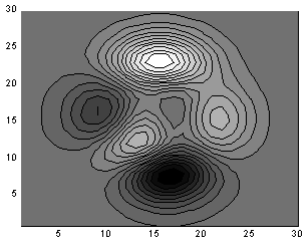

图 1-29

图形结果为图 1-29。

命令 6 **pie3**

功能 三维饼形图

用法 pie3(X) 用 x 中的数据画一个三维饼形图。X 中的每一个元素代表三维饼形图中的一部分。

pie3(X,explode) x 中的某一部分可以从三维饼形图中分离出来。explode 是一个与 x 同型的向量或矩阵，explode 中非零的元素对应 x 中从饼形图中分离出来

的分量。

h = pie3(…) 返回一个分量为 patch，surface 和 text 图形句柄对象的向量。即每一块对应一个句柄。

注意：命令 pie3 将 x 的每一个元素在所有元素的总和中所占的比例表达出来。若 x 中的分量和小于 1（则所有元素小于 1），则认为 x 中的值指明三维饼形图的每一部分的大小。

例 1-30

```
>>x = [1 3 0.5 2.5 2]

>>ex = [0 1 0 0 0]

>>pie3(x,ex)
```

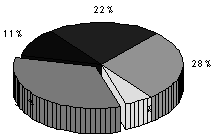

图 1-30

图形结果为图 1-30。

1.4.3 曲面与网格图命令

命令 1 mesh

功能 生成由 X，Y 和 Z 指定的网线面，由 C 指定的颜色的三维网格图。网格图是作为视点由 view（3）设定的 surface 图形对象。曲面的颜色与背景颜色相同（当要动画显示不透明曲面时，这时可用命令 hidden 控制），或者当画一个标准的可透视的网线图时，曲面的颜色就没有（命令 shading 控制渲染模式）。当前的色图决定线的颜色。

用法 mesh(X,Y,Z) 画出颜色由 c 指定的三维网格图，所以和曲面的高度相匹配，

1．若 X 与 Y 均为向量，length（X）=n，length（Y）=m，而[m，n]=size（Z），空间中的点 (X(j),Y(I),Z(I,j)) 为所画曲面网线的交点，分别地，X 对应于 z 的列，Y 对应于 z 的行。

2．若 X 与 Y 均为矩阵，则空间中的点 (X(I,j),Y(I,j),Z(I,j))为所画曲面的网线的交点。

mesh(Z) 由[n，m] = size（Z）得，X =1：n 与 Y=1：m，其中 z 为定义在矩形划分区域上的单值函数。

mesh(…,C) 用由矩阵 c 指定的颜色画网线网格图。Matlab 对矩阵 c 中的数据进行线性处理，以便从当前色图中获得有用的颜色。

mesh(…,PropertyName’,PropertyValue, …) 对指定的属性 PropertyName 设置属性值 PropertyValue,可以在同一语句中对多个属性进行设置。

h = mesh(…) 返回 surface 图形对象句柄。

运算规则：

1．数据 X，Y 和 z 的范围，或者是对当前轴的 XLimMode，YLimMode 和 ZLimMode 属性的设置决定坐标轴的范围。命令 aXis 可对这些属性进行设置。

2．参量 c 的范围，或者是对当前轴的 Clim 和 ClimMode 属性的设置（可用命令 caxis 进行设置），决定颜色的刻度化程度。刻度化颜色值作为引用当前色图的下标。

3．网格图显示命令生成由于把 z 的数据值用当前色图表现出来的颜色值。Matlab 会自动用最大值与最小值计算颜色的范围（可用命令 caxis auto 进行设置），最小值用色图中的第一个颜色表现，最大值用色图中的最后一个颜色表现。Matlab 会对数据的中间值执

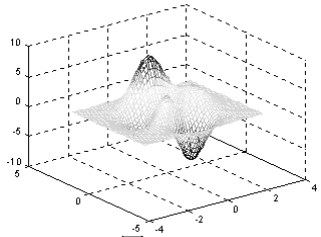

图 1-31

行一个线性变换，使数据能在当前的范围内显示出来。

例 1-31

```
>>[X,Y] = meshgrid(-3:.125:3);

>>Z = peaks(X,Y);

>>mesh(X,Y,Z);
```

图形结果为图 1-31。

命令 2　**surf**

功能　在矩形区域内显示三维带阴影曲面图。

用法　surf(Z) 生成一个由矩阵 z 确定的三维带阴影的曲面图，其中 [m，n] = size（Z），而 X = 1：n，Y = 1：m。高度 z 为定义在一个几何矩形区域内的单值函数，z 同时指定曲面高度数据的颜色，所以颜色对于曲面高度是恰当的。

surf(X,Y,Z) 数据 z 同时为曲面高度，也是颜色数据。X 和 Y 为定义 X 坐标轴和 Y 坐标轴的曲面数据。若 X 与 Y 均为向量，length（X）=n，length（Y）=m，而[m,n]=size（Z），在这种情况下，空间曲面上的节点为（X(I),Y(j),Z(I,j)）。

surf(X,Y,Z,C) 用指定的颜色 c 画出三维网格图。Matlab 会自动对矩阵 c 中的数据进行线性变换，以获得当前色图中可用的颜色。

surf(…,'PropertyName',PropertyValue) 对指定的属性 PropertyName 设置为属性值 PropertyValue

h = surf(…) 返回一个 surface 图形对象句柄给变量 h。

运算规则：

1．严格地讲，一个参数曲面是由两个独立的变量 I、j 来定义的，它们在一个矩形区域上连续变化。例如，a<=I<=b,c<=j<=d，三个变量 X，Y，Z 确定了曲面。曲面颜色由第四参数矩阵 C 确定。

2．矩形定义域上的点有如下关系：

A(I-1,j)

|

B(I,j-1) ---- C(I,j) ---- D(I,j+1)

|

E(I+1,j)

这个矩形坐标方格对应于曲面上的有四条边的块，在空间的点的坐标为[X(☺,Y(☺,Z)，每个矩形内部的点根据矩形的下标和相邻的四个点连接；曲面上的点只有相邻的三个点，曲面上四个角上的点只有两个相邻点，上面这些定义了一个四边形的网格图。

3. 曲面颜色可以有两种方法来指定：指定每个节点的颜色或者是每一块的中心点颜色。在这种一般的设置中，曲面不一定为变量 X 和 Y 的单值函数，进一步而言，有四边的曲面

块不一定为平面的，而可以用极坐标，柱面坐标和球面坐标定义曲面。

4．命令 shading 设置阴影模式。若模式为 interp，C 必须与 X，Y，Z 同型；它指定了每个节点的颜色，曲面块内的颜色由附近几个点的颜色用双线性函数计算出来的。若模式为 facted（缺省模式）或 flat，c(I,j)指定曲面块中的颜色：

A(I,j)---------- B (I,j+1)

| C(I,j) |

C(I+1,j) --------- D(I+1,j)

在这种情形下，C 可以与 X，Y，和 Z 同型，且它的最后一行和最后一列将被忽略，换句话说，就是 C 的行数和列数可以比 X，Y，Z 少 1。

5．命令 surf 将指定图形视角为 view（3）。

6．数据 X，Y，Z 的范围或者通过对坐标轴的属性 XlimMode，YlimMode 和 ZlimMode 的当前设置（可以通过命令 axis 来设置），将决定坐标轴的标签。

7．参数 C 的范围或者通过对坐标轴的属性 Clim 和 ClimMode 的设置（可以通过命令 caxis 来设置），将决定颜色刻度化。刻度化的颜色值将作为引用当前色图的下标。

例 1-32

```
>>[X,Y,Z] = peaks(30);

>>surf(X,Y,Z)

>>colormap hsv
```

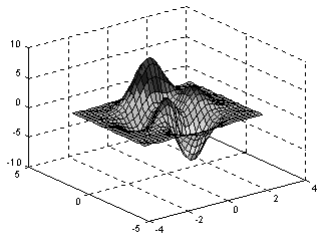

图 1-32

结果图形为图 1-32。

命令 3　surfc

功能　在矩形区域内显示三维带阴影曲面图，且在曲面下面画出等高线。

用法　surfc(Z)、surfc(X,Y,Z)、 surfc(X,Y,Z,C)、
surfc(…,'PropertyName',PropertyValue)、
surfc(…)、 h = surfc(…)

上面各个使用形式的曲面效果与命令 surf 的相同，只不过是在曲面下面增加了曲面的等高线而已。

例 1-33

```
>>[X,Y,Z] = peaks(30);

>>surfc(X,Y,Z)

>>colormap hsv
```

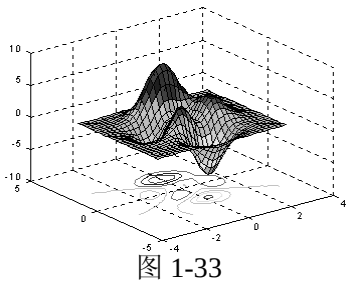

图 1-33

图形结果为图 1-33。

命令 4　surfl

功能　画带光照模式的三维曲面图。该命令显示一个带阴影的曲面，结合了周围的，散射的和镜面反射的光照模式。想获得较平滑的颜色过度，要使用有线性强度变化的色图（如：gray，copper，bone，pink 等）。参数 X，Y，Z 确定的点定义了参数曲面的“里面”和“外面”，若用户想曲面的“里面”有光照模式，只要使用：

surfl(X',Y',Z')

用法　surfl(Z) 以向量 z 的元素生成一个三维的带阴影的曲面，其中阴影模式中的光源的方位、光照系数为缺省值（见下面）。

surfl(X,Y,Z) 以矩阵 X，Y，Z 生成的一个三维的带阴影的曲面，其中阴影模式中的光源的方位、光照系数为缺省值（见下面）。

surfl(…,'light') 用一个 matlab 光照对象（light object）生成一个带颜色、带光照的曲面，这与用缺省光照模式产生的效果不同。

surfl(…,'cdata') 改变曲面颜色数据（color data），使曲面成为可反光的曲面。

surfl(…,s) 指定光源与曲面之间的方位 s，其中 s 为一个二维向量[azimuth，elevation]，或者三维向量[sx，sy，sz]。缺省光源方位为从当前视角开始，逆时针 45℉（度）。

surfl(X,Y,Z,s,k) 指定反射常系数 k，其中 k 为一个定义环境光（ambient light）系数（0<=ka<=1）、漫反射(diffuse reflection)系数（0〈=kb〈=1〉、镜面反射(specular reflection)系数（0〈=ks〈=1〉与镜面反射亮度（以相素为单位）等的四维向量[ka，kd，ks，shine]，缺省值为 k=[0.55 0.6 0.4 10]。

h = surfl(…) 返回一个曲面图形句柄向量 h。

例 1-34

```
>>[X,Y] = meshgrid(-3:1/8:3);

>>Z = peaks(X,Y);

>>surfl(X,Y,Z);

>>shading interp

>>colormap(gray);
```

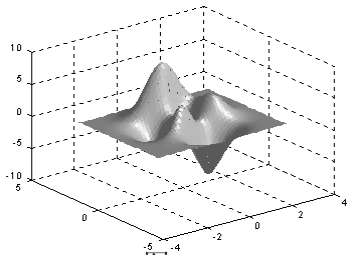

图 1-34

图形结果为图 1-34。

命令 5　waterfall

功能　瀑布图

用法　waterfall(X,Y,Z) 用所给参数 X、Y 与 Z 的数据画一“瀑布”效果图。若 X 与 Y 都是向量，则 X 与 Z 的列相对应，Y 与 Z 的行相对应，即 length(X)=Z 的列数，length(Y)=Z 的行数。参数 X 与 Y 定义了 x-轴与 y-轴，Z 定义了 z-轴的高度，Z 同时确定了颜色，所以颜色能恰当地反映曲面的高度。若想研究数据的列，可以输入：waterfall(Z')或 waterfall(X',Y',Z')

waterfall(Z) 画出一瀑布图，其中缺省地有：X=1:Z 的行数，Y=1:Z 的行数，且 Z

同时确定颜色，所以颜色能恰当地反映曲面高度。

waterfall(…,C) 用比例化的颜色值从当前色图中获得颜色，参量 C 决定颜色的比例，为此，必须与 Z 同型。系统使用一线性变换，从当前色图中获得颜色。

h = waterfall(…) 返回 patch 图形对象的句柄 h，可用于画出图形。

例 1-35

```
>>[X,Y,Z] = peaks(30);

>>waterfall(X,Y,Z)
```

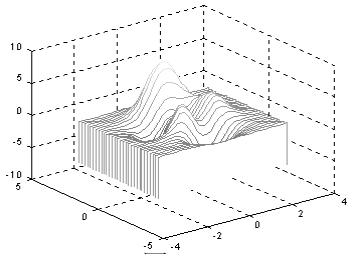

图 1-35

图形结果为图 1-35。

命令 6　cylinder

功能　生成圆柱图形。该命令生成一单位圆柱体的 x-，y-，z-轴的坐标值。用户可以用命令 surf 或命令 mesh 画出圆柱形对象，或者用没有输出参量的形式而立即画出图形。

用法　[X,Y,Z] = cylinder 返回一半径为 1、高度为 1 的圆柱体的 x-，y-，z-轴的坐标值，圆柱体的圆周有 20 个距离相同的点。

[X,Y,Z] = cylinder® 返回一半径为 r、高度为 1 的圆柱体的 x-，y-，z-轴的坐标值，圆柱体的圆周有 20 个距离相同的点。

[X,Y,Z] = cylinder(r,n) 返回一半径为 r、高度为 1 的圆柱体的 x-，y-，z-轴的坐标值，圆柱体的圆周有指定的 n 个距离相同的点。

cylinder(…) 没有任何的输出参量，直接画出圆柱体。

例 1-36

```
>>t = 0:pi/10:2*pi;

>>[X,Y,Z] = cylinder(2+(cos(t)).^2);

>>surf(X,Y,Z); axis square
```

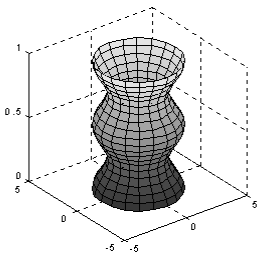

图 1-36

图形结果为图 1-36。

命令 7　sphere

功能　生成球体

用法　sphere 生成三维直角坐标系中的单位球体。该单位球体由 20*20 个面。

sphere(n) 在当前坐标系中画出有 n*n 个面的球体

[X,Y,Z] = sphere(n) 返回三个阶数为(n+1)*(n+1)的，直角坐标系中的坐标矩阵。该命令没有画图，只是返回矩阵。用户可以用命令 surf（X，Y，Z）或 mesh（X，Y，Z）画出球体。

例 1-37

```
>>[X,Y,Z]=sphere;

>>mesh(X,Y,Z)
```

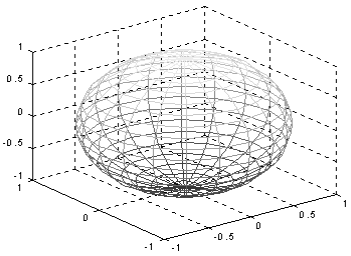

图 1-37

>>hidden off

图形结果为图 1-37。

1.4.4 三维数据的其他表现形式命令

命令 1 pcolor

功能 伪彩色图。该图为一矩形单元的、由参数 c 定义了颜色的阵列，系统通过 c 中的每相邻的四点定义的曲面补片而生成一伪彩色图。是从上面向下观看的“平面”曲面图。若用户使用命令 shading faceted 或 shading flat，则每一单元的固定颜色是与之相连的角的颜色有关的。所以，C(i,j)定义了单元的地 i 行与地 j 列的颜色。C 中的最后一行与最后一列都没有用上。若用户使用命令 shading interp，则每一单元的颜色是对它的四个顶点的颜色进行一双线性插值后的颜色，这时 c 的所有元素都参加了运算。

用法 pcolor(C) 画一伪彩色图。C 中的元素都线性地映射于当前色图下标。从 C 映射到当前的色图是由命令 colormap 和 caxis 定义的。

pcolor(X,Y,C) 在参数 x 和 y 指定的位置上画一由 C 确定的为彩色图。该图为一逻辑上为矩形、带二维格栅的、顶点在[X(i,j),Y(i,j)]的图形（若 X 和 Y 为矩阵时）。参量 X 与 Y 为指定格栅线的向量或矩阵。若 X 与 Y 为向量，则 X 对应于 C 的列，而 y 对应于 C 的行；若 X 与 Y 同为矩阵，则必须为同型矩阵。该命令等价于命令：surf(X,Y,0,C)，观察角度为：view([0,90])。

h = pcolor(…) 返回一 surface 图形对象句柄于 h

例 1-38

>>pcolor(magic(20))

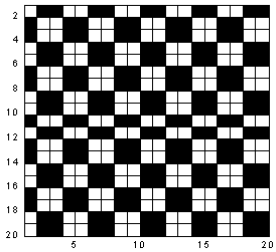

图 1-38

>>colormap(gray(2))

>>axis ij;axis square

图形结果为图 1-38。

命令 2 quiver

功能 矢量图或速度图

用法 quiver(U,V) 在范围为 x =1:n 和 y =1:m 的坐标系中显示由 U 和 V 定义的向量，而[m,n]=size(U)=size(V)，这种形式是在一个几何矩形中画出 U 和 V 的，quiver 命令本身会自动地画出这些向量，使之不会重叠。

quiver(X,Y,U,V) 由向量 X 和 Y 中的分量的任意组合而成的向量与。若 X 与 Y 都是向量 length（X）=n，而 length（Y）=m，而[m，n]=size（U）=size（V），向量 X 对应于矩阵 U、V 的列向量，而向量 Y 对应于矩阵 U、V 的行向量。

quiver(…,scale) 自动对向量的长度进行处理。使之不会重叠，当然可以对 scale 进行取值，若 scale=2，则向量长度伸长 2 倍，若 scale=0，则如实画出向量图。

quiver(…,LineSpec) 可以指定画矢量图用的线型，符号，颜色，quiver 命令会在原来的向量图上画出记号。

quiver(…,LineSpec,'filled') 对用 LineSpec 指定的记号进行填充

h = quiver(…) 返回每个向量图的句柄

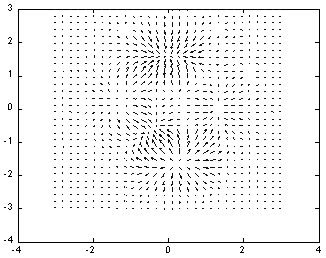

图 1-39

例 1-39

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

>>[Dx,Dy]=gradient(z,0.1,0.1);

>>quiver(x,y,Dx,Dy)
```

图形结果为图 1-39。

命令 3 **slice**

功能 立体切片图。该命令显示通过立体图形的矩形切片图。

用法 slice(X,Y,Z,V,sx,sy,sz) 显示三元函数 V=V(X,Y,Z)确定的超立体形在 x-轴、y-轴与 z-轴方向上的若干点（对应若干平面。即若函数 V=V(X,Y,Z)中有一变量如 X 取一定值 X0，则函数 V=V(X0,Y,Z)变成一立体曲面（只不过是将该曲面通过颜色表示高度 V,从而显示于一平面而已。）的切片图，各点的坐标由参量向量 sx、sy 与 sz 指定。参量 X、参量 Y 与参量 Z 为三维数组，用于指定立方体 V 的坐标。参量 X、Y 与 Z 必须有单调的、正交的间隔（如同用命令 meshgrid 生成的一样）。在每一点上的颜色由对超立体 V 的三维内插值确定。

slice(V,sx,sy,sz) 显示三元函数 V=V(X,Y,Z)确定的超立体形在 x-轴、y-轴与 z-轴方向上的若干点（对应若干平面）的切片图，各点的坐标由数量向量 sx、sy 与 sz 指定。其中 V 为三维数组（阶数为 m*n*p），缺省地有：X = 1:m、Y = 1:n、Z = 1:p。

slice(V,XI,YI,ZI) 显示参量矩阵 XI、YI 与 ZI 确定的、超立体图形的切面图。参量 XI、YI 与 ZI 定义了一曲面，同时会在曲面的点上计算超立体 V 的值。参量 XI、YI 与 ZI 必须为同型矩阵。

slice(X,Y,Z,V,XI,YI,ZI) 沿着由矩阵 XI、YI 与 ZI 定义的曲面画穿过超立体图形 V 的切片。

slice(…,'method') 指定内插值的方法。‘method’为如下方法之一：‘linear’、‘cubic’、‘nearest’：

‘linear’——指定使用三次线性内插值法（该状态为缺省的）；

‘cubic’—— 指定使用三次立方内插值法；

‘nearest’——指定使用最近点内插值法。

h = slice(…) 返回一曲面图形对象的句柄向量 h。

命令 4 **axis**

功能 坐标轴的刻度与外在显示

用法 axis([xmin xmax ymin ymax]) 设置当前坐标轴的 x-轴与 y-轴的范围。

axis([xmin xmax ymin ymax zmin zmax cmin cmax]) 设置当前坐标轴的 x-轴、y-轴与 z-轴的范围，当前颜色刻度范围。该命令也同时设置当前坐标轴的属性 Xlim、Ylim 与 Zlim 为所给参数列表中的最大值和最小值。另外，坐标轴属性 XlimMode、YlimMode 与 ZlimMode 设置为‘manual’。

v = axis 返回一包含 x-轴、y-轴与 z-轴的刻度因子的行向量，其中 v 为一四维或

六维向量，这取决于当前坐标为二维还是三维的。返回的值包含当前坐标轴的 XLim、Ylim 与 Zlim 属性值。

axis auto 设置系统到它的缺省动作——自动计算当前轴的范围，这取决于输入参量 x，y 与 z 的数据中的最大值与最小值。同时将当前坐标轴的属性 XlimMode、YlimMode 与 ZlimMode 设置为‘auto’用户可以指定对某一坐标轴进行自动操作。例如：

axis 'auto x' 将自动计算 x-轴的范围；

axis 'auto yz' 将自动计算 y-轴与 z-轴的范围。

axis manual 、axis(axis) 把坐标固定在当前的范围，这样，若保持状态(hold)为 on，后面的图形仍用相同界限。该命令设置了属性 XLimMode、属性 YLimMode 与属性 ZlimMode 为 manual。

axis tight 把坐标轴的范围定为数据的范围，即坐标轴中没有多余的部分。

axis fill 该命令用于将坐标轴的取值范围分别设置为绘图所用数据在相应方向上的最大、最小值。

axis ij 使用矩阵坐标系：坐标原点在左上角、横坐标（j-轴）的值从左到右增加，纵坐标（i-轴）的值从上到下增加。

axis xy 使用笛卡儿坐标系（缺省）：坐标原点在左下角、横坐标（x-轴）的值从左到右增加，纵坐标（y-轴）的值从下到上增加。

axis equal 设置坐标轴的纵横比，使在每个方向的数据单位都相同。其中 x-轴、y-轴与 z-轴将根据所给数据在各个方向的数据单位自动调整其纵横比。

axis image 效果与命令 axis equal 相同，只是图形区域刚好紧紧包围图象数据。

axis square 设置当前图形为正方形（或立方体形），系统将调整 x-轴、y-轴与 z-轴，使它们有相同的长度，同时相应地自动调整数据单位之间的增加量。

axis normal 自动调整坐标轴的纵横比，还有用于填充图形区域的、显示于坐标轴上的数据单位的纵横比。

表 1-7 显示由上面三个命令设置的坐标轴属性。

表 1-7

命令 / 坐标轴属性	axis equal	axis normal	axis square	axis tightequal
DataAspectRatioMode	[1 1 1]	没有设置	没有设置	[1 1 1]
PlotBoxAspectRatio	manual	auto	auto	Manual
PlotBoxAspectRatioMode	[3 4 4]	没有设置	[1 1 1]	Auto
Stretch-to-fill	禁止	可行	禁止	禁止

axis vis3d 该命令将冻结坐标系此时的状态，以便进行旋转。

axis off 关闭所用坐标轴上的标记、格栅和单位标记。但保留由 text 和 gtext 设置的对象。

axis on 显示坐标轴上的标记、单位和格栅。

[mode,visibility,direction] = axis('state') 返回表明当前坐标轴的设置属性的三个字符串，见表 1-8。

表 1-8

输出参量	返回字符串	说明
Mode	‘’auto’或 ’manual’	若 XLimMode、YlimMode 与 ZlimMode 都设置为 auto，则 mode 为 auto；若 XLimMode、YlimMode 或者 ZlimMode 都设置为 manual，则 mode 为 manual
Visibility	‘’on’或’off’	
Direction	‘’xy’或’ij’	

例 1-40

```
>>x = 0:.025:pi/2;

>>plot(x,exp(x).*sin(2*x),'-m<')

>>axis([0    pi/2    0    5])
```

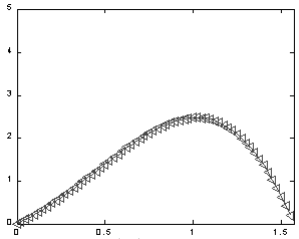

图 1-40

图形结果为图 1-40。

命令 5　**hidden**

功能　在一网格图中显示隐含线条。隐含线条的显示，实际上是显示那些从观察角度观看没有被其他物体遮住的线条。

用法　hidden on 对当前图形打开隐含线条的显示状态，使网格图后面的线条被前面的线条遮住。设置曲面图形对象的属性 FaceColor 为坐标轴背景颜色。这是系统的缺省操作。

hidden off 对当前图形关闭隐含线条的显示

hidden 在两种状态 on 与 off 之间切换

例 1-41

```
>>mesh(peaks)

>>hidden off
```

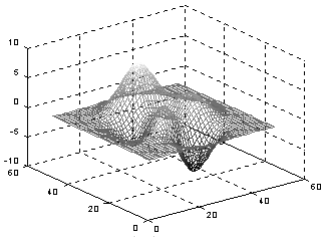

图 1-41

图形结果为图 1-41。

命令 6　**shading**

功能　设置颜色色调属性。该命令控制曲面与补片等的图形对象的颜色色调。同时设置当前坐标轴中的所有曲面与补片图形对象的属性 EdgeColor 与 FaceColor。命令 shading 设置恰当的属性值，这取决于曲面或补片对象是表现网格图或实曲面。

用法　shading flat 使网格图上的每一线段与每一小面有一相同颜色，该颜色由线段的末端的端点颜色确定；或由小面的、有小型的下标或索引的四个角的颜色确定。

shading faceted 带重叠的黑色网格线的平面色调模式。这是缺省的色调模式。

shading interp 在每一线段与曲面上显示不同的颜色，该颜色为通过在每一线段两边的、或者为不同小曲面之间的色图的索引或真颜色进行内插值得到的颜色。

例 1-42

```
>>sphere(16)

>>axis square

>>shading flat

>>title('Flat Shading')
```

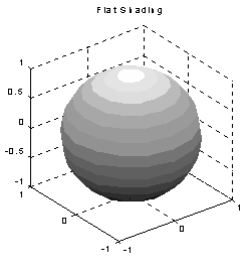

图 1-42

图形结果为图 1-42。

命令 7　caxis

功能　颜色坐标轴刻度。命令 caxis 控制着对应色图的数据值的映射图。它影响下面对象之一的、用带索引的颜色数据（CData）与颜色数据映射（CDataMapping）控制的刻度的图形对象 surface、patches 与 images；它没有影响带用颜色数据（CData）或颜色数据映射（CDataMapping）直接设置的颜色的图形对象 surface、images 或 patches。该命令还改变坐标轴图形对象的属性 Clim 与 ClimMode。

用法　caxis([cmin cmax]) 用指定的最大值与最小值设置颜色范围。数据值中小于 cmin 或大于 cmax 的，将分别地映射于 cmin 与 cmax；处于 cmin 与 cmax 之间的数据将线性地映射于当前色图。

caxis auto 让系统自动地计算数据的最大值与最小值对应的颜色范围。这是系统的缺省动作。数据中的正无穷大（Inf）对应于最大颜色值；负无穷大（-Inf）对应于最小颜色值；带颜色值设置为 NaN 的面或者边界将不显示。

caxis manual、caxis(caxis) 冻结当前颜色坐标轴的刻度范围。这样，当 hold 设置为 on 时，可使后面的图形命令使用相同的颜色范围。

v = caxis 返回一包含当前正在使用的颜色范围的二维向量 v=[cmin cmax]。

caxis(axes_handle,…) 使由参量 axis_handle 指定的坐标轴，而非当前坐标轴。

颜色坐标轴刻度工作原理：

使用带索引的颜色数据(Cdata)与颜色数据映射(CdataMapping)的图形对象 surface、patch 与 image 将设置成刻度化的，在每次图形渲染时，将映射颜色数据值为当前图形的颜色。当颜色数据值等于或小于 cmin 时，将它映射为当前色图中的第一个颜色；当颜色数据值等于或大于 cmax 时，将它映射为当前色图中的最后一个颜色；对于处于 cmin 与 cmax 之间的颜色数据（例如 c），系统将执行下列线性转换，以获得对应当前色图（它的长度为 m）中的颜色的索引（当前色图的行指标 index）：

index = fix((C-min)/(cmax-cmin)*m)+1

例 1-43

```
>>[X,Y,Z] = sphere;

>>C = Z;surf(X,Y,Z,C)

>>caxis([-1 3])
```

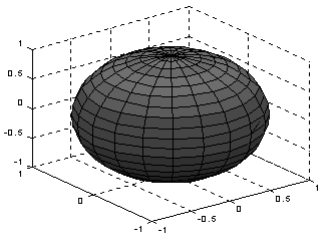

图 1-43

图形结果为图 1-43。

命令 8　view

功能　指定立体图形的观察点。观察者（观察点）的位置决定了坐标轴的方向。用户可以用方位角（azimuth）和仰角（elevation）一起，或者用空间中的一点来确定观察点的位置。

用法　view(az,el)、view([az,el]) 给三维空间图形设置观察点的方位角。方位角 az 与仰角 el 为这两个旋转角度：做一通过视点与 z-轴的平面，与 xy 平面有一交线，该交线与 y-轴的反方向的、按逆时针方向（从 z-轴的方向观察）计算的、单位为度的夹角，就是观察点的方位角 az。若角度为负值，则按顺时针方向计算；在通过视点与 z-轴的平面上，用一直线连接视点与坐标原点，该直线与 xy 平面的夹角就是观察点的仰角 el。若仰角为负值，则观察点转移到

曲面下面。

view([x,y,z]) 在笛卡儿坐标系中于点（x,y,z）设置视点。注意：输入参量只能是方括号的向量形式，而非数学中的点的形式。

view(2) 设置缺省的二维形式视点。其中 az=0，el=90，即从 z-轴上方观看。

view(3) 设置缺省的三维形式视点。其中 az=-37.5，el=30。

view(T) 根据转换矩阵 T 设置视点。其中 T 为 4*4 阶的矩阵，如同用命令 viewmtx 生成的透视转换矩阵一样。

[az,el] = view 返回当前的方位角 az 与仰角 el。

T = view 返回当前的 4*4 阶的转换矩阵 T。

例 1-44

```
>>peaks;

>>az = 0;el = 90;

>>view(az, el)
```

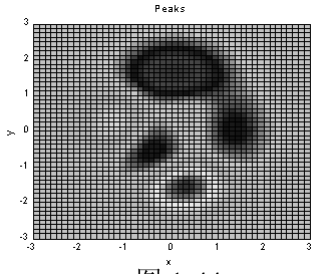

图 1-44

图形结果为图 1-44。

命令 9　viewmtx

功能　视点转换矩阵。计算一个 4*4 阶的正交的或透视的转换矩阵，该矩阵将一四维的、齐次的向量转换到一个二维的视平面上（如计算机平面上）。

用法　T = viewmtx(az,el) 返回一与视点的方位角 az 与仰角 el（单位都为度）对应的正交矩阵，并没有改变当前视点。

T = viewmtx(az,el,phi) 返回一透视的转换矩阵，其中参量 phi 是单位为度的透视角度，为标准化立方体（单位为度）的对像视角角度与透视扭曲程度。

表 1-9

Phi 的值	说明
0 度	正交投影
10 度	类似以远距离投影
25 度	类似以普通投影
60 度	类似以广角投影

用户可以通过使用返回的矩阵，用命令 view(T)改变视点的位置。该 4*4 阶的矩阵将变换四维的、同次的向量成形式为(x,y,z,w)的非标准化的向量，其中 w 不等于 1。正交化的 x-元素与 y-元素组成的向量(x/w,y/w,z/w,1)为我们所需的二维向量。（注：一四维同次向量为在对应的三维向量后面增加一个 1。例如：[x,y,z,1]为对应于三维空间中的点[x,y,z]的四维向量。）

T = viewmtx(az,el,phi,xc) 返回以在标准化的图形立方体中的点 xc 为目标点的透视矩阵（就像相机正对着点 xc 一样），目标点 xc 为视角的中心点。用户可以用一三维向量 xc=[xc,yc,zc]指定该中心点，每一分量都在区间[0，1]上。缺省值为 xc=[0 0 0]。

命令 10　surfnorm

功能　计算与显示三维曲面的法线。该命令计算用户命令 surf 中的曲面法线。

用法　surfnorm(Z)、surfnorm(X,Y,Z) 画出一曲面与它的法线图。其中矩阵 Z 用于指定曲面的高度值；X 与 Y 为向量或矩阵，用于定义曲面的 x 与 y 部分。

[Nx,Ny,Nz] = surform(…) 返回组成曲面的法线在三个坐标轴上的投影分量 Nx,Ny 与 Nz。

例 1-45

```
>>[x,y,z] = cylinder(1:10);

>>surfnorm(y,x,z)

>>axis([-12 12 -12 12 -0.1 1])
```

图形结果为图 1-45。

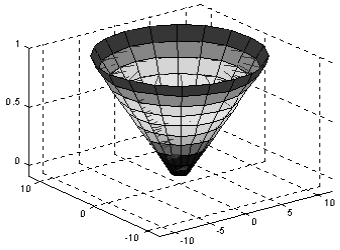

图 1-45　曲面法线图