

第 6 章 Maple 中作图

6.1 二维函数作图命令 plot

6.1.1 二维函数作图

用 plot 命令可以画出一元函数在指定区间上的二维函数图形。其用法有

```
plot (函数, 变量名)
```

```
plot (函数, 范围, 选项)
```

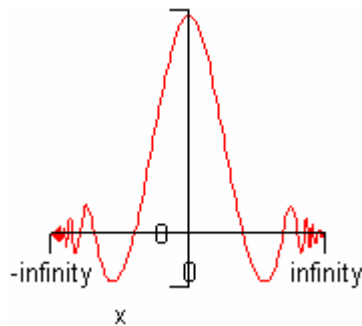
范围和选项均可省略, 缺省时系统自动选取最佳设置。最简单的 plot 语句为

```
plot(f(x), x=a..b)
```

画出 $f(x)$ 在区间 $[a, b]$ 上的图像, 其中 f 可为过程或表达式。

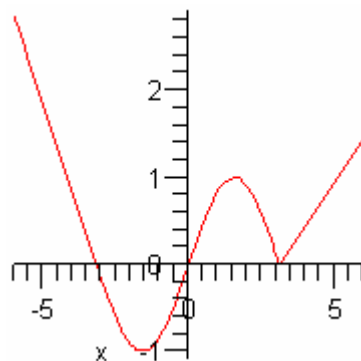
例: 画出函数 $f(x) = \frac{1}{x} \sin x$ 在区间 $(-\infty, \infty)$ 上的图形。

```
> plot(sin(x)/x, x=-infinity..infinity);
```



例: 画出分段函数 $f(x) = \begin{cases} -x - \pi & x < -\pi \\ \sin x & -\pi \leq x \leq \pi \\ \frac{1}{2}(x - \pi) & x > \pi \end{cases}$ 在区间 $[-6, 6]$ 上的图像。

```
> f:=x->piecewise(x<-Pi,-x-Pi,x<=Pi and x>=-Pi,sin(x),x>Pi,(x-Pi)/2):  
plot(f(x), x=-6..6);
```



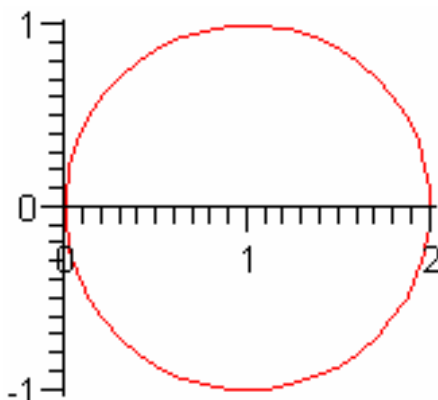
6.1.2 plot 选项

6.1.3 参数方程作图

用 `plot` 函数画参数曲线的一般形式为 `plot ([x(t),y(t),t=a..b], 选项)` 或 `plot ([[x(t),y(t),t=a..b],[u(t),v(t),t=c..d]], 选项)` 在一个坐标系中同时画两条参数曲线。

例：画参数曲线 $\begin{cases} x = 1 + \cos t \\ y = \sin t \end{cases}, t \in [0, 2\pi]$ 。

```
> plot([1+cos(t),sin(t),t=0..2*Pi]);
```



6.1.4 特殊坐标系下作图

`plot` 通常画的是直角坐标下的函数图像，通过设置 `coords` 选项，`plot` 也可以画出特殊坐标下的函数图像。例如，画出极坐标下函数 $r = r(t)$, $a \leq t \leq b$ 的图形可用命令

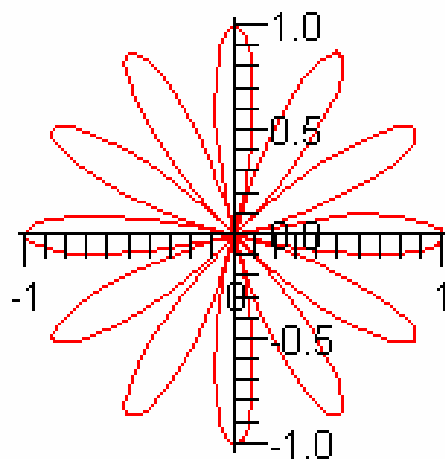
```
plot(r(t),t=a..b,coords=polar)
或 plot([r(t),t,t=a..b],coords=polar)
```

在 6.3 小节中，还将给出 `plots` 程序包中画特殊坐标系下的函数图像的命令，例如

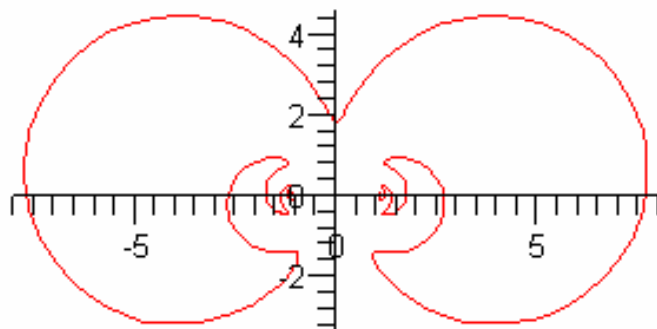
```
polarplot(r(t),t=a..b)
```

例：特殊坐标系下的函数图像。

```
> plot([cos(6*x),x,x=0..2*Pi],coords=polar);
```



```
> plot(cos(6*t), t=-Pi..Pi, coords=bipolar); # 双极坐标
```



6.2 三维函数作图 plot3d

6.2.1 三维函数作图

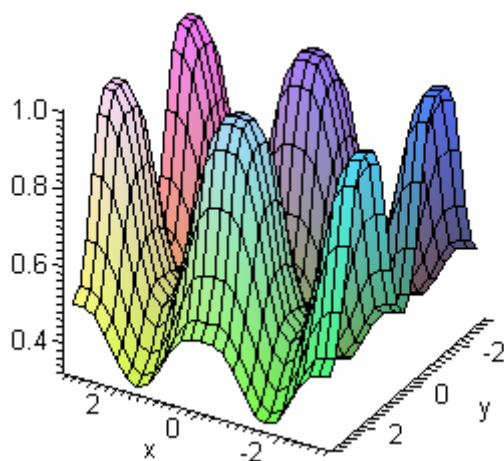
画二元函数 $f(x, y)$ 的三维图像可用 plot3d 命令

```
plot3d(f(x,y), x=a..b, y=c..d, 选项)
```

与 plot 函数相似, plot3d 的选项可有可无, 项数可多可少。plot3d 的部分选项的名称和功能的细节请看 6.2.3 小节。

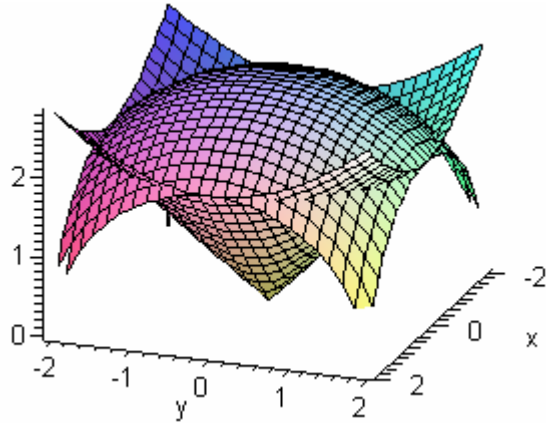
例: 画出函数 $f(x, y) = \frac{1}{1 + \sin^4 x + \cos^4 y}$, $-\pi \leq x, y \leq \pi$ 的图像。

```
> plot3d(1/(1+sin(x)^4+cos(y)^4), x=-Pi..Pi, y=-Pi..Pi);
```



例: 画出球面与锥面所围立体及其表面。

```
> plot3d([sqrt(8-x^2-y^2), sqrt(x^2+y^2)], x=-2..2, y=-2..2);
```



6.2.2 三维参数方程作图

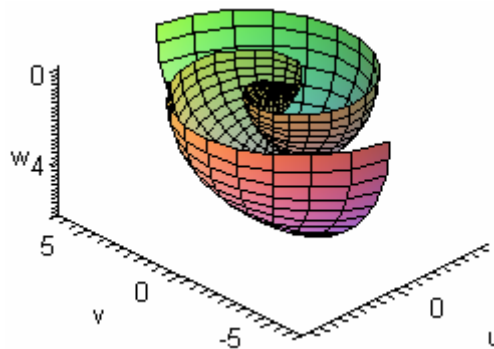
用 `plot3d` 命令画三维参数曲面 $(x(s,t), y(s,t), z(s,t))$ 的一般形式为

```
plot3d([x(s,t),y(s,t),z(s,t)],s=a..b,t=c..d)
plot3d([f,g,h],a..b,c..d)
```

其中 x 、 y 、 z 是关于 s 和 t 的表达式， f 、 g 、 h 是有两个参数的过程。

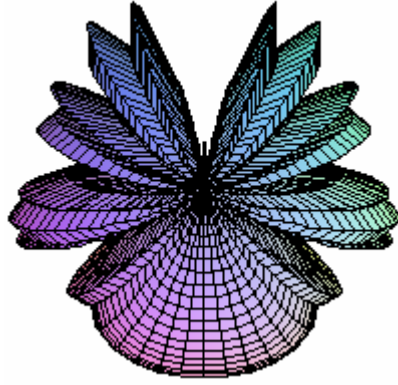
例：画出参数曲面
$$\begin{cases} u(x,y) = x \sin x \cos y & 0 \leq x \leq 2\pi \\ v(x,y) = x \cos x \cos y, & 0 \leq y \leq \pi \\ w(x,y) = x \sin y \end{cases}$$

```
> plot3d([x*sin(x)*cos(y),x*cos(x)*cos(y),x*sin(y)],
x=0..2*Pi,y=0..Pi,labels=['u','v','w']);
```



例：用球坐标作图。

```
> plot3d(cos(x^2)*sin(y),x=-2*Pi..2*Pi,y=-Pi..Pi,coords=spherical,
numpoints=10000);
```



6.3 plots 程序包

在 `plots` 包中有 56 个画图命令和选项设置命令，在本节中只能列出部分命令。动画演示命令 `animate`、`animate3d` 和 `animatecurve` 在 6.4 节中，绘制微分方程数值解的函数图像的命令 `odeplot` 在第 5 章中。

```
> with(plots);
```

```
Warning, the name changecoords has been redefined
```

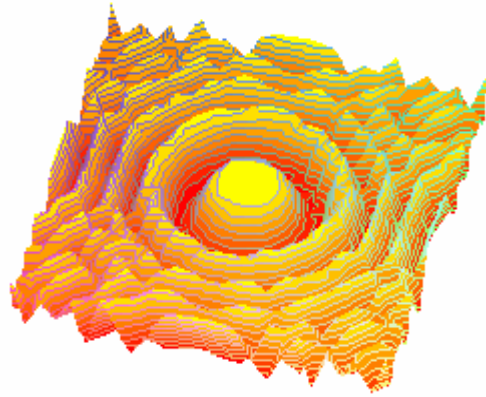
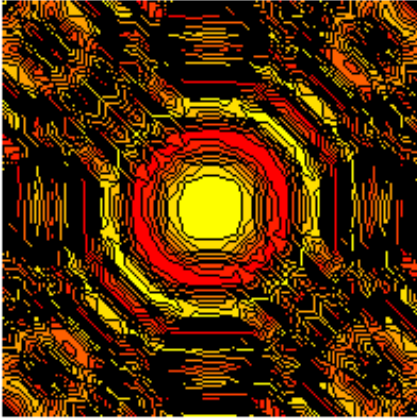
```
[animate, animate3d, animatecurve, arrow, changecoords, complexplot,  
complexplot3d, conformal, conformal3d, contourplot, contourplot3d,  
coordplot, coordplot3d, cylinderplot, densityplot, display,  
display3d, fieldplot, fieldplot3d, gradplot, gradplot3d,  
graphplot3d, implicitplot, implicitplot3d, inequal, interactive,  
interactiveparams, listcontplot, listcontplot3d, listdensityplot,  
listplot, listplot3d, loglogplot, logplot, matrixplot, multiple,  
odeplot, pareto, plotcompare, pointplot, pointplot3d, polarplot,  
polygonplot, polygonplot3d, polyhedra_supported, polyhedraplot,  
replot, rootlocus, semilogplot, setoptions, setoptions3d,  
spacecurve, sparsematrixplot, surfdata, textplot, textplot3d,  
tubepplot]
```

- 等值线图

```
contourplot (函数, 范围, 选项)  
contourplot3d (函数, 范围, 选项)
```

例：函数 $f(x) = \sin(\cos(x^2 + y^2))$ 的等值线图。

```
> contourplot(sin(cos(x^2+y^2)),x=-10..10,y=-10..10,filled=true);  
contourplot3d(sin(cos(x^2+y^2)),x=-10..10,y=-10..10,filled=true);
```



- 密度图

`densityplot (函数, 范围, 选项)`

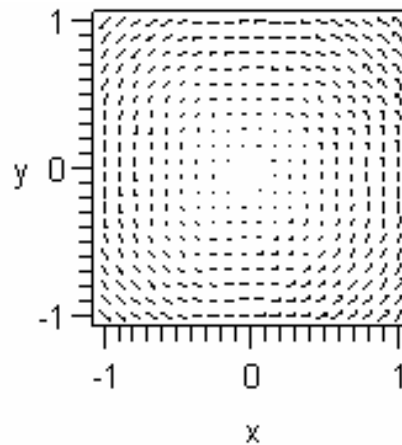
- 向量场图

`fieldplot (向量场, 范围, 选项)`

`fieldplot3d (向量场, 范围, 选项)`

例：画二维向量场

`> fieldplot([sin(2*x*y),cos(2*x-y)],x=-2..2,y=-2..2,axes=boxed);`



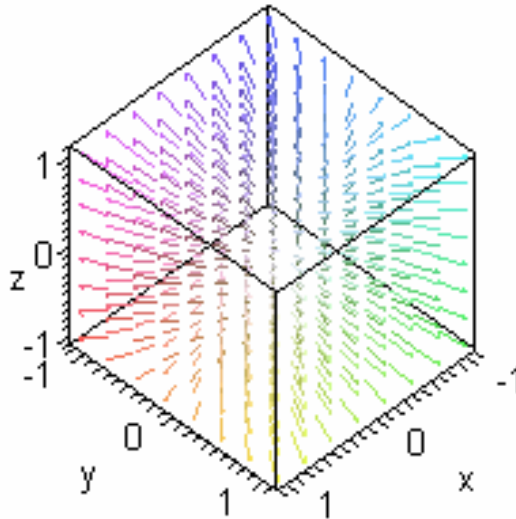
- 梯度图

`gradplot (函数, 范围, 选项)`

`gradplot3d (函数, 范围, 选项)`

例：画三维梯度图。

`> gradplot3d(x^2+y^2+z^2,x=-1..1,y=-1..1,z=-1..1,axes=boxed);`



- 隐函数作图

```
implicitplot (方程, 范围, 选项)
implicitplot3d (方程, 范围, 选项)
```

画出函数 $x^2 + y^2 = z^2$ 的图形。

```
> implicitplot3d(x^2+y^2=z^2,x=-1..1,y=-1..1,z=-1..1);
```

画出函数 $x^3 + y^3 + z^3 + 1 = (x + y + z + 1)^3$ 的图形。

```
> implicitplot3d(x^3+y^3+z^3+1=(x+y+z+1)^3, x=-2..2, y=-2..2,
  z=-2..2, style=patchnogrid, numpoints=10000);
```

- 数据点列作图

- 复函数作图

6.4 动画演示

动画命令 `animate`

```
animate(f(x,t),x=a..b,t=c..d)
```

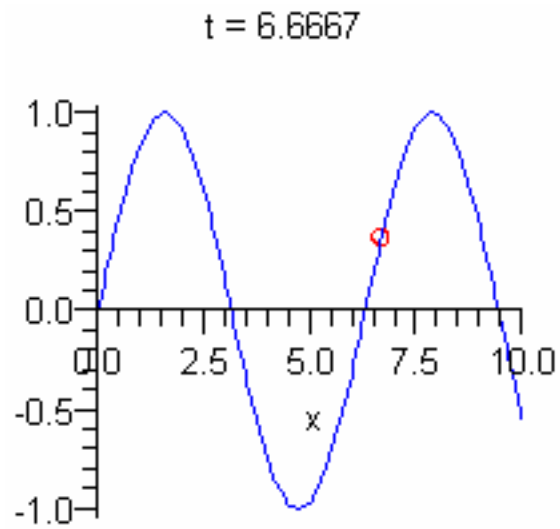
`animate` 函数在 `plots` 程序包中，使用前要先调出 `plots` 程序包。

例：动画演示 $\sin(x)$ 图像。

```
> with(plots);
> animate(sin(x-t),x=-2*Pi..2*Pi,t=0..10, scaling=constrained);
```

例：红点沿着蓝色曲线滑动。

```
> curve:=plot(sin(x),x=0..10,color=blue):
> animate(pointplot,[[t,sin(t)],symbol=circle,symbolsize=10],
  t=0..10,color=red,background=curve);
```



例：用函数图形做动画背景。

```
> fg:=animate(plot3d,[sin(x*y*t),x=-Pi..Pi,y=-Pi..Pi],t=-1..1):
  bg:=plot3d(x-y,x=-Pi..Pi,y=-Pi..Pi): display(fg,bg);
```

t = .75000

