第8讲 符 号 运 算

8.1 符号对象
8.2 数值与符号变量的相互转换
8.3 符号矩阵与运算
8.4 符号表达式的化简
8.5 符号微积分
8.6 符号方程与求解
8.7 符号函数图形绘制

8.1 符 号 对 象

8.1.1 符号运算的特点

对于一般的程序设计软件，如C、C++、Fortran等语言可以顺利运行数值计算，但是在实现符号计算方面是非常困难的。而MATLAB自带有符号工具箱“Symbolic Math Toolbox”，提供的函数命令是专门研究符号运算功能和用来解算符号对象问题的，具有强大的符号运算功能。

1．符号对象

符号对象是符号工具箱中定义的另一种数据类型。符号对象是用来存储代表符号的字符串，在符号工具箱中符号对象用于表示**符号变量、符号矩阵、符号表达式和符号方程**。

在数学计算中有数值计算与符号计算之分，数值计算的表达式、矩阵变量中不允许有未定义的自由变量，例如：

```
>> A=[a,b;c,d]
??? Undefined function or variable 'a'.
```

而符号计算可以含有未定义的符号变量，例如：

```
>> A=sym('[a,b;c,d]')
A =
[ a, b]
[ c, d]
```


2．符号对象与普通数据对象的差别

数学计算有数值计算与符号计算之分。这两者的根本区别是：

- 数值计算的表达式、矩阵变量中不允许有未定义的自由变量。
- 符号计算可以含有未定义的符号变量，在符号计算的整个过程中，所运算的是符号变量。需要注意的是，在符号计算中所出现的数字也都是当作符号处理的。

下例说明了符号对象和普通的数据对象之间的差别。

8.1.2 建立符号表达式和求值

(1) 用sym()和syms()函数建立符号表达式。

用sym()函数建立符号表达式，其语法为

sym (符号变量)

例如，用sym命令创建：

```
>> x=sym('x')
>> f=sin(x)
f = sin(x)
```

或用syms()命令创建，例如：

```
>>syms x
>>f=sin(x)+cos(x)
f = sin(x)+cos(x)
```


(2) 使用已经定义的符号变量组成符号表达式。例如：

```
>> syms a b c x
>> f = a*x^2 + b*x + c
f = a*x^2 + b*x + c
```

8.2 数值与符号变量的相互转换

8.2.1 符号转换为数值

符号变量表示的值都是精确的，而数值变量表示的值可能是不精确的，有时符号运算的目的是得到精确的数值解，这就要对得到的解析解进行数值转换。

我们在符号表达式转换为数值变量时要考虑到转换精度的问题。例如：f=sym('1/3'); 将f定义为1/3，如果要转换为数值，那么我们应该转为0.3还是 0.33333呢? 计算机存储总是有限制的，我们只能存储到有限个3。

使用**digits()**、**vpa()**和**double()**函数进行符号与数值的转换。一般情况下，我们把这三个函数作为组合拳，先使用digits()设定精确程度，再使用vpa()作近似运算，最后才是double()转换为数值变量。

北京郵電大學

1．digits()函数

digits(n)：设置有效数字位数。该函数的作用是指定精确到多少(n)位有效数字，默认是32位。

2．vpa()函数

vpa(f)将符号表达式f的结果精确到digits()所设定的有效数字的位数。值得注意的是，**vpa()返回的还是符号表达式**。命令形式如下：

(1) R = vpa(A)：根据当前指定的精度，使用可变精度算术VPA函数，计算A的每一个元素。结果中的每一个元素都是符号表达式。

(2) R = vpa(A, d)：使用d代替当前的精度，计算A的每一个元素，即求A在d精度下的数值解。d的值必须是正整数，而且位于1～2^{29}+1之间。

北京邮电大学

3．double()函数

double()是将符号表达式转换为浮点数数值变量类型的函数。double()命令的形式如下：

x = double(s)：转换s为双精度型数值变量x，s可以是符号变量也可以是字符串变量。

(1) 当s是符号变量时，s必须是全为数字的符号，返回数值变量x。例如：

```
>> s1=sym('20.3');
>> x1 = double(s1) % 把符号变量 s1 转化为数值变量 x1
x1 = 20.3000
```

8.2.2 数值转换为符号

数值转换为符号的方式有以下几种：

(1) sym()函数用于生成符号变量，也可以将数值转化为符号变量。

例如，命令形式：x = sym(s)。功能是将数值s转换为符号变量x，s 不可以是表达式。

(2) sym()的另一个重要作用是将数值矩阵转化为符号矩阵。

8.2.3 poly2sym()函数与多项式的符号表达式

poly2sym()函数可以把多项式用符号表达式表示出来。用法如下：

r = poly2sym(c)：返回多项式的符号表达式，多项式的系数是数字向量c。默认符号表达式的变量是x，变量v可以指定作为第二个参数。

$$r = c_1x^{n-1} + c_2x^{n-2} + \cdots + c_n \tag{8.2.1}$$

例如：

```
>> y=[ 1  -12  44  -48  0]
>> ya=poly2sym(y)
ya = x^4 -12*x^3 + 44*x^2-48*x
```

8.3 符号矩阵与运算

8.3.1 符号矩阵的索引和修改

例 符号矩阵的索引。

解 程序如下：

```
>> a=[2/3,sqrt(2),0.222;1.4,1/0.23,log(3)]
a =
   0.6667    1.4142    0.2220
   1.4000    4.3478    1.0986
>> b=sym(a)
b =
   [ 2/3,  2^(1/2),                          111/500]
   [ 7/5,  100/23,  2473854946935173/2251799813685248]
```

1．符号矩阵的索引

直接对符号矩阵的元素索引。

>> b(1,3)　　%矩阵的索引

ans = 111/500

2．符号矩阵的修改

(1) 直接修改。可用矩阵元素下标，直接修改。

>> b(2,3)='2/5' %矩阵的修改

b =

[2/3, 2^(1/2), 111/500]

[7/5, 100/23, 2/5]

(2) 使用subs() 指令修改。

A1=subs(A, 'new ')

A1=subs(A, 'new', 'old')

例 使用subs()指令修改。

解 程序如下：

```
>>syms a b
>>A =[  a, 2*b;3*a,   0]
>> A(2,2) = 4*b
>>A1 = [  a, 2*b；3*a, 4*b]
>> A2 = subs(A1, b, c)
A2 = [  a, 2*c]
        [3*a, 4*c]
```

8.3.2 符号矩阵的四则运算

1．基本运算

符号矩阵的基本运算符与数值矩阵的运算符是统一的(+ - * / \)。符号矩阵的数值运算中，与数值矩阵一样，是对应元素的运算。所有矩阵运算操作指令都比较直观、简单，如a=b+c; a=a*b；A=2*a^2+3*a-5等。

符号矩阵的行列式运算、逆运算、求秩、幂运算、数组指数运算、矩阵指数运算使用：det(a)、inv(b)、rank(a)、a^2、exp(b)、expm(b)。

例 8-3-6 符号矩阵的基本运算。

(1) 已知 $a=\begin{bmatrix}\frac{1}{x} & \frac{1}{x+1}\\ \frac{1}{x+2} & \frac{1}{x+3}\end{bmatrix}$、$b=\begin{bmatrix}x & 1\\ x+2 & 0\end{bmatrix}$，求 b−a 和 a\b 的值。

(2) 已知 f= 2*x^2+3*x−5、g= x^2+x−7，求 f+g 的值。

(3) 已知 f=cos(x)、g= sin(2*x)，求 f/g+f*g 的值。

解 (1)

```
>>x = sym('x’)
>>a = [1/x,1/(x+1); 1/(x+2), 1/(x+3)]
>>b = [x,1;x+2,0]
>>b-a
 ans =
[      x-1/x,   1-1/(x+1)]
[ x+2-1/(x+2),    -1/(x+3)]
>>a\b
 ans =
    [    -6*x-2*x^3-7*x^2,    3/2*x^2+x+1/2*x^3]
    [  6+2*x^3+10*x^2+14*x, -1/2*x^3-2*x^2-3/2*x]
```

(2) >> syms x

>> f=2*x^2+3*x-5; g= x^2+x-7;

>> h=f+g

h = 3*x^2+4*x-12

(3) >> syms x

>> f=cos(x);g=sin(2*x);

>> f/g+f*g

ans =cos(x)/sin(x)+cos(x)*sin(x)

8.4 符号表达式的化简

8.4.1 合并多项式

collect()函数用于合并多项式中的同类项，具体调用格式如下：

(1) R = collect(S)，该命令将S中的每个元素，按默认变量x的阶数进行同类项系数合并，其中S可以是数组，数组的每个元素为符号表达式。

例 合并多项式。

解 程序如下：

```
>> syms x t;
>> f=(1+x)*t+x*t;
>> collect(f)
ans = 2*t*x + t
```

(2) R = collect(S,v)，对指定的变量v进行合并，如果不指定，则默认为对x进行合并，或者由findvar()函数返回的结果进行合并。

8.4.2 展开多项式

expand()函数用于符号表达式的展开。调用格式如下：

expand(S)：对符号表达式S中每个因式的乘积进行展开计算。该命令通常用于计算多项式函数、三角函数、指数函数和对数函数等表达式的展开。

例 符号表达式的展开。

解 程序如下：

```
>> syms x
>> f=x*(x*(x-1)+3)+2;
>> y=expand(f)
y = x^3 - x^2 + 3*x + 2
```

8.4.3 转换多项式

与expand()函数相反，horner()函数把多项式转换为Horner形式，这种形式的特点是乘法嵌套，有着较好的数值计算性质，嵌套格式在多项式求值中可以降低计算时的复杂度。该函数的调用格式如下：

R = horner(P)

其中，P为由符号表达式组成的矩阵，该命令将 P 中的所有元素转化为相应的嵌套形式。

例 转换多项式。

解 程序如下：

```
>> syms x;
>> y = x^3 - x^2 + 3*x + 2
>> horner(y)
ans = x*(x*(x-1)+3)+2
```

8.4.4 简化多项式

1．化简函数simplify()

simplify()函数通过数学运算实现符号表达式的化简。例如：

```
>> syms  x ;
>> f=sin(x)^2+cos(x)^2;
>> simplify(f)
ans =1
```

8.4.5 因式分解与factor()函数

factor()函数实现因式分解功能，如果输入的参数为正整数，则返回此数的素数因数。如果无法在有理数的范围内作分解，那么返回的结果还是输入值。语法格式如下：

factor(X)：参量X可以是正整数、符号表达式矩阵或符号整数矩阵。若X为一正整数，则factor(X)返回X的质数分解式。若X为多项式或整数矩阵，则factor(X)分解矩阵的每一元素。若整数矩阵中有一元素位数超过16位，用户必须用sym命令生成该元素。

例 将f = x3-6x2 + 11x -6进行因式分解。

解 程序如下：

```
>> syms x;
>> f=x^3-6*x^2+11*x-6;
>> factor(f)
ans =[ x - 3, x - 1, x - 2]
```

8.4.6 分式通分

numden()函数用于求解符号表达式的分子和分母，可用于分式通分。语法格式如下：

[N,D]=numden(A)：把 A 的各元素转换为分子和分母都是整系数的最佳分式。A 是一个符号或数值矩阵，N 是符号矩阵的分子，D 是符号矩阵的分母。

例 8-4-6 已知多项式为 $\frac{x}{y}+\frac{y}{x}$，求其通分结果。

解 程序如下：

```
>> syms x y;
>> [n,d] = numden(x/y + y/x)
```

返回结果为

```
n = x^2 + y^2
d =x*y
```

即该式的通分结果为 $\frac{x}{y}+\frac{y}{x}=\frac{x^2+y^2}{xy}$。

8.4.7 符号替换

1．subs() 函数

subs()函数可以在一符号表达式或矩阵中进行符号替换，用指定符号替换表达式中的某一特定符号，如将符号表达式中的符号变量用数值代替。

在对多变量符号表达式使用subs()函数时，如果不指定变量，则系统选择默认变量进行计算。默认变量的选择规则为：选择在字母表中离x 近的变量字母作为默认变量，**如果有两个变量并且和 x 之间的距离相同，则选择字母表靠后面的变量作为默认变量**。

subs()函数的调用格式如下：

(1) R = subs(S)。对于S中出现的全部符号变量，如果在调用函数或工作空间中存在相应值，则将值代入，同时自动进行化简计算；若是数值表达式，则计算出结果。如果没有相应值，则对应的变量保持不变。

(2) R = subs(S, new)。用新的符号变量new替换S中的默认变量，即有findsym()函数返回的变量。

(3) R = subs(S, old, new)。用新值new替换表达式S中的旧值old，参量old是一符号变量或代表一变量名的字符串，new是一符号、数值变量或表达式。

若old与new是具有相同大小的阵列，则用new中相应的元素替换old中的元素；

若S与old为标量，而new为阵列或单元阵列，则标量S与old将扩展为与new同型的阵列；将S中的所有old替换为new，并将S中的常数项扩充为与new维数相同的常数矩阵。

若new为数值矩阵的单元阵列，则替换按元素的方向执行。

若new是数字形式的符号，则数值代替原来的符号计算表达式的值，所得结果仍是字符串形式。

例如，求解常微分方程dy = - a * y，其程序如下：

```
>>y = dsolve('Dy = -a*y')
```

积分结果为

```
y =
C1/exp(a*t)
```

如果工作空间存在a、C1值，或输入a、C1值：

```
>> a = 980,C1=3;
>> subs(y)
ans =
3/exp(980*t)
```

用a、C1值替换表达式中的变量值。

2．subexpr() 函数

subexpr()函数通过计算机自动寻找，将表达式中多次重复出现的因式或字符串用简短的符号或变量替换，返回的结果中包含替换之后的表达式，以及被替换的因式。该函数的调用格式如下：

(1) [Y,SIGMA] = subexpr(X,SIGMA)。指定用符号变量SIGMA来代替符号表达式X中(可以是矩阵)重复出现的字符串。替换后的结果由Y返回，被替换的字符串由SIGMA返回。例如：

```
>> syms x  a;
f=solve(x^2+a*x-1);
r=subexpr(f);
```

```
>> r
sigma =
   (a^2 + 4)^(1/2)
r =
 - a/2 - sigma/2
 sigma/2 - a/2
```

8.5 符号微积分

8.5.1 符号表达式求极限

极限是微积分的基础，微分和积分都是“无穷逼近”时的结果。在MATLAB中，limit()函数用于求表达式的极限。该函数的调用格式如下：

(1) limit(F,x,a)：当x趋近于a时表达式F的极限；

(2) limit(F,a)：当F中的自变量趋近于a时F的极限，自变量由findsym()函数确定。

(3) limit(F)：当F中的自变量趋近于0时F的极限，自变量由findsym()函数确定。

(4) limit(F,x,a,'right')：当x从右侧趋近于a时F的极限。

(5) limit(F,x,a,'left')：当x从左侧趋近于a时F的极限。

例如：

```
>> syms h n x
>> L=limit((log(x+h)-log(x))/h,h,0)
```

求当h趋近于0时表达式L的极限，式中单引号可省略掉，结果为

```
L = 1/x
>> M=limit((1-x/n)^n,n,inf)
```

求当n趋近于无穷大时表达式M的极限，结果为

M = exp(-x)或M = 1/exp(x)

例 求极限。

(1) $\lim_{x\to 0}\left(\frac{\sin x}{x}\right)=?$ (2) $\lim_{n\to\infty}\sqrt{n+\sqrt{n}}-\sqrt{n}=?$

解 程序如下：

(1)
```
>> syms x a t h;
>> limit(sin(x)/x)
ans = 1
```

(2)
```
>> syms n;
>> limit(sqrt(n+sqrt(n))-sqrt(n),n,inf)
ans = 1/2
```

8.5.2 符号导数、微分和偏微分

MATLAB中使用diff()函数实现函数求导和求微分，可以实现一元函数求导和多元函数求偏导。当输入参数为符号表达式时，该函数实现符号微分，其调用格式如下：

(1) diff(S)：对缺省变量求微分，实现表达式S的求导，自变量由函数findvar()确定。

(2) diff(S,'v')：对指定变量v求微分，该语句还可以写为diff(S,sym('v'))。

(3) diff(S,n)：求S的n阶导数。

(4) diff(S,'v',n)：求S对v的n阶微分(导数)，该表达式还可以写为diff(S,n,'v')。

例 8-5-2　符号导数和微分。

(1) 已知 $y(x)=\sin(ax)$，求 $A=\dfrac{dy}{dx}$ 为何值，$B=y''(x)$为何值。

(2) 已知 $y(x)=\ln(1+x)$，$\left.\dfrac{d^2y}{dx^2}\right|_{x=1}$ 的值。

解　程序如下：

(1)
```
>> syms a x;    y=sin(a*x);
>> A=diff(y,x)
A = a*cos(a*x)
>> B =diff(y,x,2)
B = -a^2*sin(a*x)
```


北京邮电大学

(2)
```
>> syms x;
>> y=log(1+x);
>> d2y=diff(y,x,2)
d2y =-1/(1+x)^2
```

即 $\dfrac{d^2y}{dx^2}=-\dfrac{1}{(1+x)^2}$，然后将符号表达式转换成数值表达式：

```
>> x=1;eval(d2y)
ans = -0.2500
```

即 $\left.\dfrac{d^2y}{dx^2}\right|_{x=1}=-\left.\dfrac{1}{(1+x)^2}\right|_{x=1}=-\dfrac{1}{4}$。

北京邮电大学

例 8-5-3　已知 $z(x,y)=e^{2x}(x+y^2+2y)$，求偏微分：$a=\dfrac{\partial z}{\partial x}$ 为何值，$b=\dfrac{\partial z}{\partial y}$ 为何值，$A=\dfrac{\partial^2 z}{\partial x^2}$ 为何值，$B=\dfrac{\partial^2 z}{\partial y^2}$ 为何值，$AB=\dfrac{\partial^2 z}{\partial x\partial y}$ 为何值。

解　程序如下：

```
>> syms x y;
>> z=exp(2*x)*(x+y^2+2*y);
>> a=diff(z,x)
>> b=diff(z,y)
>> A=diff(z,x,2)
>> B=diff(z,y,2)
>> AB=diff(a,y)
```


结果如下：

a =2*exp(2*x)*(x+y^2+2*y)+exp(2*x)，即 $a=\dfrac{\partial z}{\partial x}=2e^{2x}(x+y^2+2y)+e^{2x}$

b =exp(2*x)*(2*y+2)，即 $b=\dfrac{\partial z}{\partial y}=e^{2x}(2y+2)$

A =4*exp(2*x)*(x+y^2+2*y)+4*exp(2*x)，即 $A=\dfrac{\partial^2 z}{\partial x^2}=4e^{2x}(x+y^2+2y)+4e^{2x}$

B =2*exp(2*x)，即 $B=\dfrac{\partial^2 z}{\partial y^2}=2e^{2x}$

AB =2*exp(2*x)*(2*y+2)，即 $AB=\dfrac{\partial^2 z}{\partial x\partial y}=2e^{2x}(2y+2)$

8.5.3 多元函数的导数与jacobian()函数

在MATLAB中，多元函数的导数由jacobian()函数来实现。微积分中一个非常的重要概念为Jacobian矩阵，计算函数向量的微分。MATLAB中，jacobian()函数用于计算Jacobian 矩阵。该函数的调用格式如下：

R = jacobian(f,v)。如果f是向量函数或函数，v为自变量向量，则计算f的Jacobian矩阵；如果f是标量，则计算f的梯度，如果v也是标量，则其结果与diff()函数相同。

例 求 $\begin{cases} x^2+y^2=4 \\ x^2-y^2=1 \end{cases}$ 函数的 jacobi 矩阵。

解 程序如下：

```
>> x=sym(['x']);y=sym(['y']);z=sym(['z']);
>> jacobian([x^2+y^2; x^2-y^2],[x,y])
ans =
     [ 2*x,   2*y]
     [ 2*x, -2*y]
```

例 计算不定积分、定积分。

(1) $f_1(x)=\int \frac{x^2+1}{(x^2-2x+2)^2}dx$；

(2) $f_2(x)=\int_0^{\pi/2} \frac{\cos(x)}{\sin(x)+\cos(x)}dx$；

(3) $f_3(x)=\int_0^{+\infty} e^{-x^2}dx$。

解 程序如下：

```
>> syms x
   y1=(x^2+1)/(x^2-2*x+2)^2;
   y2=cos(x)/(sin(x)+cos(x));
   y3=exp(-x^2);
>> f1=int(y1)
>> f2=int(y2,0,pi/2)
>> f3=int(y3,0,inf)
```

结果为

```
f1 = (3*atan(x - 1))/2 + (x/2 - 3/2)/(x^2 - 2*x + 2)
f2 = pi/4
f3 = pi^(1/2)/2
```

例 计算二重不定积分。

(1) $\iint xe^{-xy}\,dxdy$；

(2) $\iint \frac{y^2}{x^2}dxdy$ ($\frac{1}{2}\leqslant x\leqslant 2$，$1\leqslant y\leqslant 2$)。

解 程序如下：

(1)
```
>> syms x y
>>f=x*exp(-x*y)
F=int(int(f,x),y)
 F = 1/(y*exp(x*y))
```

(2)
```
>> syms x y;
f=y^2/x^2;
int(int(f,x,1/2,2),y,1,2)
 ans =7/2
```


8.5.4 计算不定积分、定积分

与微分对应的是积分，在MATLAB中，int()函数用于实现符号积分运算。该函数的调用格式如下：

(1) R = int(S)。求表达式S的不定积分，自变量由findsym()函数确定；

(2) R = int(S,v)。求表达式S对自变量v的不定积分；

(3) R = int(S,a,b)。求表达式S在(a,b)区间上的定积分，自变量由findsym()函数确定；a为积分下限、b为积分上限，上限、下限缺省时为不定积分。

(4) R = int(S,v,a,b)。求表达式S在(a,b)区间上的定积分，自变量为 v。

例 举例说明符号对象和普通数据对象之间的差别。

解 在命令窗口中输入如下命令：

```
>> sqrt(2)
ans = 1.4142
>> x=sqrt(sym(2))
x = 2^(1/2)
```

由上例可以看出，当采用符号运算时，并不计算出表达式的结果，而是给出符号表达式。如果需要查看符号 x 所表示的值，在窗口中输入：

```
>> double(x)
ans = 1.4142
```

8.6 符号方程与求解

8.6.1 符号代数方程求解

代数方程包括线性方程、非线性方程和超越方程等。

1．解代数方程

在MATLAB中，solve()函数用于求解代数方程和方程组，其调用格式如下：

(1) g = solve(eq)。对默认自变量x求解，求方程符号eq的解，输入参数eq可以是符号表达式或字符串，即solve(eq)对方程eq中的默认变量x，求解方程eq=0。若输出参量g为单一变量，则对于有多重解的非线性方程，g为一行向量。

(2) solve(eq,var,vlue)。求解方程 eq 的解，对指定自变量var求解。例如g=solve(x^2-1) 或g=solve(x^2-1,x,0) 或g=solve(x^2-1,0) 。

对于单个方程的情况，返回结果为一个符号表达式，或是一个符号表达式组成的数组。

例 求一元二次方程$ax^2 + bx + c = 0$的根。

解 程序如下：

```
>>syms a b c x
>>f=a*x^2+b*x+c
>> solve(f)
ans =
 -(b + (b^2-4*a*c)^(1/2))/(2*a)
 - (b-(b^2-4*a*c)^(1/2))/(2*a)
```

即方程$ax^2 + bx + c = 0$的根为

$$x_1 = \frac{(-b + \sqrt{b^2 - 4ac})}{2a}$$

$$x_2 = \frac{(-b - \sqrt{b^2 - 4ac})}{2a}$$

即结果为 $x = \dfrac{-b \pm \sqrt{b^2 - 4ac}}{2a}$。

如果把 a 作为自变量求解，则结果如下：

```
>> solve(f,'a')
ans= – (b*x+c)/x^2
```

2．代数方程组的符号解法

代数方程组同样使用solve()函数进行求解。

对于方程组的情况，返回结果为一个结构体，结构体的元素为每个变量对应的表达式，各个变量按照字母顺序排列，其格式如下：

(1) solve(eq1,eq2,...,eqn)。求由方程eq1、eq2、…、eqn等组成的系统，自变量为默认自变量；

(2) solve(eq1,eq2,...,eqn,var1,var2,...,varn)。求由方程eq1、eq2、…、eqn等组成的系统，自变量为指定的自变量：var1、var2、…、varn。

例 求 $\begin{cases} x + y = 1 \\ x - 11y = 5 \end{cases}$ 的解。

解 程序如下：

```
>>syms x y
>>f1 = x+y-1
>>f2 = x-11*y-5
>>S = solve(f1,f2)
```

返回结果为一个结构体S：

```
S =
   x: [1x1 sym]
   y: [1x1 sym]
>> S.x              ans =4/3
>> S.y              ans =-1/3
```


8.6.2 常微分方程的解析解

1．常微分方程的求解

常微分方程解析解的求解法，一般有以下三种。

1) 积分求解

有些微分方程可直接通过积分求解，例如通过一阶常系数常微分方程：

$$\frac{dy}{dt} = y + 1$$

化为

$$\frac{dy}{y+1} = dt$$

两边积分可得通解为

$$y = ce^{t}-1$$

其中，c为任意常数。

有些常微分方程可用一些技巧，如分离变量法、积分因子法、常数变异法、降阶法等可化为可积分的方程而求得解析解。

2) 求特解和通解

线性常微分方程的解满足叠加原理，求解可归结为求一个特解和相应齐次微分方程的通解，一阶变系数线性微分方程总可用这一思路求得显式解。

3) 求高阶线性常系数微分方程的基本解和特解

高阶线性常系数微分方程可用特征根法求得相应齐次微分方程的基本解，再用常数变异法求特解。

一阶常微分方程与高阶微分方程可以互化，已给一个n阶方程

$$y(n) = f(t, y', y'', \cdots, y^{(n-1)})$$

令

$$y_1 = y，\ y_2 = y'，\ y_3 = y''，\ y_n = y^{(n-1)}$$

则可将上式化为一阶方程组

$$\begin{cases} y_1' = y_2 \\ y_2' = y_3 \\ \quad \vdots \\ y_n' = f(t, y_1, y_2, \cdots, y_n) \end{cases}$$

反过来，在许多情况下，一阶微分方程组也可化为高阶方程，所以一阶微分方程组与高阶常微分方程的理论与方法在许多方面是相通的，一阶常系数线性微分方程组也可用特征根法求解。

MATLAB中符号常微分方程使用dsolve()函数进行求解，具体调用格式如下：

(1) r = dsolve('eq1','eq2', …,'cond1','cond2',...,'v')。其中，eq1、eq2等表示待求解的方程，v是指定的自变量，如果不指定自变量，那么默认的自变量为x，也可任意指定自变量为't'、'u'等。cond1、cond2 等表示初始值，通常表示为y(a) = b或者diff(a) = b。如果不指定初始值，或者初始值方程的个数少于因变量的个数，则最后得到的结果中会包含常数项，表示为C1、C2 等。dsolve()函数最多可接受12个输入参数。

(2) [y1,y2…]=dsolve(x1,x2,…xn)。返回微分方程的解。

例 求微分方程：$\frac{dx}{dt}=-ax$，$\frac{d^2x}{dt^2}=-ax$。

解 语法如下：

```
syms a x(t);
eqn1 = diff(x,t) == -a*x;
s1 = dsolve(eqn1)

eqn2 = diff(x,t,2) == -a*x;
s2 = dsolve(eqn2)
```

结果为：

```
s1 =
C1*exp(-a*t)
s2 =
C1*exp((-a)^(1/2)*t) + C2*exp(-(-a)^(1/2)*t)
```


例 求微分方程 y' = x 的通解。

解 程序如下：

```
>> syms x y(x)    %定义 x，y 为符号
>> eqn1 = diff(y) == x;
>>y = dsolve(eqn1)
```

结果为：

```
y =
x^2/2 + C1
```

例 求微分方程 $\begin{cases} y'' = x + y' \\ y(0) = 1,\ \ y'(0) = 0 \end{cases}$ 的特解。

解 程序如下：

```
>> syms x y(x)
>> eqn = diff(y,x,2) == x + diff(y,x);
>> Dy = diff(y);
>> cond = [y(0) == 1, Dy(0) == 0];
>> S = dsolve(eqn,cond)
```

结果是：

```
S =
exp(x) - x - x^2/2
```


2．微分方程组的求解

微分方程组通过dsolve()函数进行求解，格式为：

r = dsolve(eq1,eq2,..., cond1,cond2,..., v)

该语句求解由参数eq1、eq2等指定的方程组成的系统，初值条件为cond1、cond2等，v为自变量

例　求微分方程组 $\begin{cases} x' = x + y \\ y' = 2x \end{cases}$ 的通解。

解　程序如下：

```
>> syms x y
>> syms x(t) y(t)
>> eqn1 = diff(x,t) == y + x;
>> eqn2 = diff(y,t) == 2 * x;
>> [x y] = dsolve(eqn1,eqn2)
```

结果为

```
x = -1/2*C1*exp(-t)+C2*exp(2*t)
y =C1*exp(-t)+C2*exp(2*t)
```

例　求微分方程组：$\begin{cases} u' = v \\ v' = w \\ w' = -u \end{cases}$，u(0)=0，v(0)=0，w(0)=1的结果。

解　程序如下：

```
>> syms u(t) v(t) w(t);
>> eqn = [diff(u) == v; diff(v) == w; diff(w) == -u];
>> cond = [u(0) == 0; v(0) == 0; w(0) == 1];
>> S = dsolve(eqn, cond)
```

结果为一个结构：

```
S =
   u: [1x1 sym]
   v: [1x1 sym]
   w: [1x1 sym]
```

```
>> S.u
      ans =
      1/(3*exp(t)) - (cos((3^(1/2)*t)/2)*exp(t)^(3/2))/(3*exp(t))
        + (3^(1/2)*sin((3^(1/2)*t)/2)*exp(t)^(3/2))/(3*exp(t))
```

可用查询结构成员的方法，查询结果：

8.7 符号函数图形绘制

8.7.1 符号函数二维绘图函数fplot()

1．显函数的调用

对于显函数，其调用格式如下：

(1) fplot(f)。绘制函数f在区间($-5 \le x \le 5$)内的图形，f可以是函数句柄或串。

(2) fplot(f,[min,max])。绘制函数f在指定区间[min, max]内的图形。

2．隐函数的调用

隐函数定义为fun2(x,y)，fplot()函数的调用格式如下：

(1) fplot(fun2)：绘制函数fun2(x,y)=0在区间($-5 \le x \le 5$)的图形。

(2) fplot(fun2,[TMIN TMAX])：绘制函数fun2(x,y)=0，x(t)和y(t)的取值范围由tmin < t < tmax决定。

例 MATLAB中符号计算中提供单位阶跃函数heaviside(t-a)、斜坡可以使用阶跃和直线方程构成。

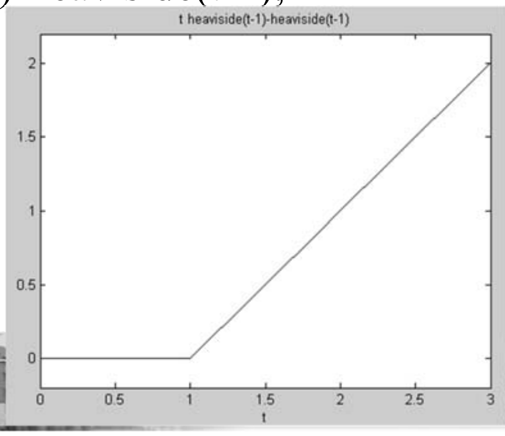
解 (1) 绘制a=4时的阶跃函数。

```
>> f=@(t)heaviside(t-4);
>> fplot(f,[0 5]) %
```

(2) 在t=1时发生转折斜率为1的斜坡可以表示为

```
>> f=@(t)t.*heaviside(t-1)-heaviside(t-1);
>> fplot(f,[0 3])
```

绘制出斜坡函数曲线。



3．参数方程的调用

对于参数方程，fplot()函数的调用格式有：

(1) fplot(x,y)：绘制参数方程x = x(t)、y = y(t)在（0 < t < 2）的曲线。

(2) fplot(x,y,[tmin,tmax])：绘制参数方程 x = x(t)、y = y(t)在tmin < t < tmax的曲线。

8.7.2 符号函数三维绘图函数fplot3()

fplot3()函数用于绘制三维参数曲线。该函数的调用格式如下：

(1) fplot3(x,y,z)。在默认区间(0<t <2p 内绘制参数方程x = x(t)、y = y(t)、z = z(t)的图像。

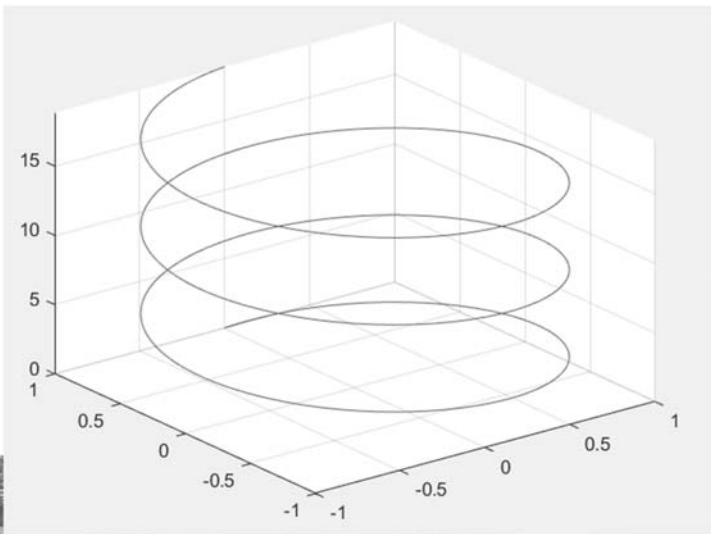
(2) fplot3(x,y,z,[tmin,tmax])。在区间tmin<t<tmax内绘制参数方程x = x(t)、y = y(t)、z = z(t) 的图像。

例 绘制空间曲线的动态轨迹。

解 程序如下：

```
>> x= sin(t);y= cos(t);z=t;
>> fplot3(x,y,z,[0,6*pi])
```

生成空间曲线的动态轨迹。



8.7.3 符号函数曲面网格图及表面图的绘制

1．fmesh()、fsurf()函数

fmesh()、fsurf()函数分别用于绘制三维网格图和三维表面图。这两个函数的用法相同，下面以函数ezmesh()函数为例介绍三维曲面的绘制。该函数的调用格式如下：

(1) fmesh(f)：绘制函数f(x,y)在默认区域$-5\le x\le 5$、$-5\le y\le 5$内的图像。

(2) fmesh(f,domain)：在domain指定区域绘制函数f(x,y)的图像；

(3) fmesh(x,y,z)：在默认区域$-5\le s\le 5$、$-5\le t\le 5$内，绘制三维参数方程x = x(s,t)，y = y(s,t)，and z = z(s,t)的图形。

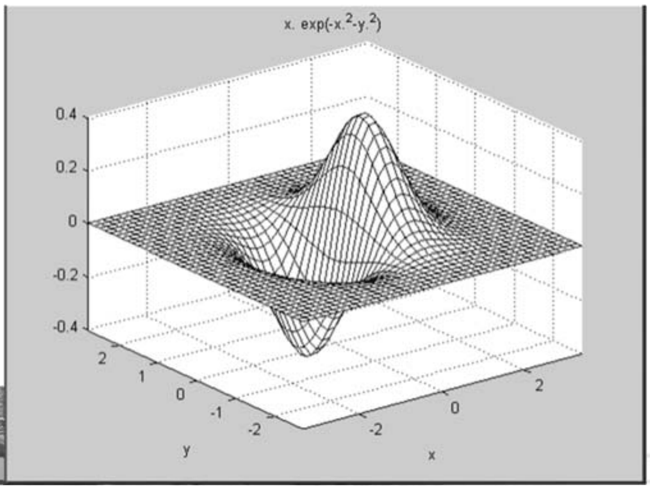
(4) fmesh(x,y,z,[smin,smax,tmin,tmax])或fmesh(x,y,z,[min,max])：在指定区域绘制三维参数方程的图像。

例 已知函数 $f(x,y)=xe^{-x^2-y^2}$，绘制三维网格图。

解 程序如下：

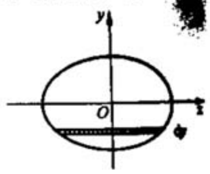
```
>> fh = @(x,y) x.*exp(-x.^2-y.^2);
>> fmesh(fh)
>> colormap([0 0 1])
```

绘制出的三维网格图。



工程实例——储油罐的油量计算

问题：一平放的椭圆柱形状的油罐，长度为L，椭圆的长半轴为a，短半轴为b，油的密度为ρ，油罐中油的高度为h。油罐的横断面如图所示。



解题思路：

（1）将横断面的方程表达式列出：$\frac{x^2}{a^2}+\frac{y^2}{b^2}=1$

（2）将椭圆对y进行微分，则图中小矩形的面积为 $2a\sqrt{1-\frac{y^2}{b^2}}dy$

（3）油罐中油横断面中高度为h时的面积为 $\int_{-b}^{-b+h} 2a\sqrt{1-\frac{y^2}{b^2}}dy$

（4）油量为 $\int_{-b}^{-b+h} 2\rho La\sqrt{1-\frac{y^2}{b^2}}dy$

（5）对油量进行积分得到 $\int_{-b}^{-b+h} 2\rho La\sqrt{1-\frac{y^2}{b^2}}dy \xrightarrow{y=b\sin\theta,-\frac{\pi}{2}\le\theta\le\frac{\pi}{2}} \int_{-\pi/2}^{-\arcsin(-1+h/b)} 2\rho Lab\cot^2 t\,dt$

$$=\rho Lab\left[\arcsin\left(-1+\frac{h}{b}\right)+\left(\frac{h}{b}-1\right)\sqrt{1-\left(\frac{h}{b}-1\right)^2}+\frac{\pi}{2}\right]$$

代码实现：

```
clear,clc,close all;
syms a b h y L r;
m = sqrt(b^2-y^2);
m1=int(m);
m2 = int(m,'-b','n');
m3 = subs(m2,'n','y');
S=2*a/b*m3;
simplify(S);
V=S*L*r;
y=h-b;
V1=subs(V,'y','h-b');
simplify(V1)
```

作 业

1、肿瘤大小分析

肿瘤大小V生长的速率与V的a次方成正比，其中a为形状参数，0<=a<=1；而比例系数K随时间减小，减小速率又与当时的K值成正比，比例系数为环境系数b。设某肿瘤参数a=1,b=0.1，K的初始值为2，V的初始值为1.（1）请分析此肿瘤生长不会超过多大？（2）画图分析肿瘤大小V和K随时间变化趋势进行分析；（3）多长时间肿瘤大小翻一倍？（4）肿瘤生长速率由递增转为递减的时间？

结题思路：首先建立肿瘤生长数学模型，然后对微分方程进行求解。

$$\begin{cases} \dfrac{dv}{dt} = kv^{a} \\ \dfrac{dk}{dt} = -bk \end{cases}$$