

# MATLAB语言在 运筹学中的应用

◎黄雍检 赖明勇 著

湖南大学出版社

教育部博士点基金项目(NO. 20010532009)

# MATLAB 语言在运筹学中的应用

黄雍检 赖明勇 著

湖南大学出版社

2005 年·长沙

## 内 容 提 要

全书共 13 章,系统而简要地介绍运筹学的基本内容和运算方法,并运用 MATLAB 语言给出了运筹学问题许多新算法.书中许多自编指令,只要输入有关参数,就得到有关问题的解,简洁快捷.书中有大量实用性例题,如需这些程序,购书者可在湖南大学经贸学院网上下载.

本书可作高等院校管理类本科生、研究生学习运筹学的教材或教学参考书,也可供从事管理科学研究和实践的相关人员参考使用.

### 图书在版编目(CIP)数据

MATLAB 语言在运筹学中的应用/黄雍检,赖明勇著. —长沙:

湖南大学出版社,2005.5

ISBN 7-81053-904-3

I. M... II. ①黄... ②赖... III. 运筹学—计算机

辅助计算—软件包, MATLAB IV. 002-39

中国版本图书馆 CIP 数据核字(2005)第 042173 号

## MATLAB 语言在运筹学中的应用

MATLAB Yuyan zai Yunchouxuezhong de Yingyong

著 者:黄雍检 赖明勇

责任编辑:厉 亚

责任校对:张建平

封面设计:张 毅

出版发行:湖南大学出版社

社 址:湖南·长沙·岳麓山 邮 编:410082

电 话:0731-8821691(发行部),8821315(编辑室),8821006(出版部)

传 真:0731-8649312(发行部),8822264(总编室)

电子邮箱:press@hnu.net.cn

网 址: <http://press.hnu.net.cn>

总 经 销:湖南省新华书店

印 装:湖南大学印刷厂

开本: 787×1092 16 开

印张: 11.5

字数: 266 千

版次: 2005 年 5 月第 1 版

印次: 2005 年 5 月第 1 次印刷

印数: 1~3 000 册

书号: ISBN 7-81053-904-3/F·85

定价: 24.00 元

版权所有,盗版必究

湖南大学版图书凡有印装差错,请与发行部联系

# 前 言

运筹学是多学科的综合性学科,它是经济中定量分析的一个重要工具. 运筹学的计算难度大,原有的电算程序可读性差,这是众所周知的. MATLAB 语言是美国 MathWord 公司推出的一种计算机语言,它始于 20 世纪 80 年代,后经版本更新,内容日趋丰富,本书采用的是 2002 年 6 月推出的 6.5 版. MATLAB 语言具有编程效率高、计算功能强、使用简便、易于扩充的特点,其程序可读性好. 基于 MATLAB 的上述特点,本书采用 MATLAB 语言来求解运筹学的问题.

全书共分 13 章,详尽介绍了 MATLAB 语言在运筹学中的应用. 第 1 章介绍 MATLAB 中的微积分运算. 第 2 章介绍 MATLAB 中矩阵运算和拼接. 第 3 章介绍线性规划指令 `linprog()` 及其在经济管理中的应用. 第 4 章介绍利用指令 `(linprog)` 解运输问题,并给出了解运输问题的自编指令. 第 5 章介绍整数规划中分枝定界法的 MATLAB 表述法,并给出了自编指令,以简化计算程序. 第 6 章介绍指派问题的 MATLAB 算法,并给出了求解指派问题的自编指令. 第 7 章给出了网络最大流和网络最小费用最大流的定解问题,并用指令 `linprog()` 解网络流问题,且给出自编指令解网络流问题. 第 8 章根据 Floyd 算法,给出自编指令求得网络距离矩阵,并由另一自编指令求得网络最短路. 第 9 章首先给出赋正权网络的最长距离矩阵,然后利用最长距离矩阵,求出事项的最早开工时间  $es$  和事项的最迟完工时间  $lf$ ,从而得到工程的关键线路;同时还给出自编指令求关键线路. 第 10 章介绍了多目标规划指令 `fgoalattain` 及其在经济管理中的应用. 第 11 章介绍了多个非线性规划指令及其应用. 第 12 章介绍两人有限零和对策,即矩阵对策问题;根据矩阵对策的线性规划解法,给出自编指令求解矩阵对策问题. 第 13 章简单介绍了动态规划的 MATLAB 算法. 书中有 2 个附录,附录 1 对 MATLAB 语言作了简介;附录 2 给出了自编指令,它是针对运筹学问题的开发,也是一种扩充,显示出 MATLAB 语言的特有功能.

书中配有大量例题,每个例题都给出了相应的 MATLAB 语言程序. 从给出的程序中,我们不难看到,它的简洁和可读性,是原有的计算程序所无法比较的. 这些程序可以在命令窗口直接运行,如要运行 `ch43`,只需在命令窗口键入 `ch43`,再按 `enter` 键,屏幕上会显示计算结果,非常方便.

限于作者水平,书中不妥和错误之处在所难免,欢迎大家批评指正.

著 者

2005 年 1 月 16 日

# 目 次

## 第 1 章 函数的运算和作图

|                   |   |
|-------------------|---|
| 1.1 函数的构成 .....   | 1 |
| 1.2 函数作图 .....    | 1 |
| 1.3 多项式运算 .....   | 3 |
| 1.4 微积分符号运算 ..... | 4 |
| 1.5 定积分近似计算 ..... | 5 |
| 1.6 例题计算程序 .....  | 6 |

## 第 2 章 矩 阵

|                    |    |
|--------------------|----|
| 2.1 矩阵的运算和生成 ..... | 9  |
| 2.2 矩阵的删改和拼接 ..... | 12 |
| 2.3 计算程序 .....     | 14 |

## 第 3 章 线性规划

|                                            |    |
|--------------------------------------------|----|
| 3.1 线性规划指令 <code>linprog()</code> 介绍 ..... | 16 |
| 3.2 线性规划的应用 .....                          | 17 |
| 3.2.1 设备配置 .....                           | 17 |
| 3.2.2 下料问题 .....                           | 18 |
| 3.2.3 生产计划安排 .....                         | 19 |
| 3.2.4 配料问题 .....                           | 20 |
| 3.2.5 投资问题 .....                           | 21 |
| 3.2.6 农业生产计划 .....                         | 23 |
| 3.3 检查线性约束和函数值计算的自编指令 .....                | 24 |
| 3.4 例题计算程序 .....                           | 25 |

## 第 4 章 运输问题

|                     |    |
|---------------------|----|
| 4.1 产销平衡运输问题 .....  | 28 |
| 4.2 销大于产运输问题 .....  | 29 |
| 4.3 产大于销运输问题 .....  | 30 |
| 4.4 障碍运输问题 .....    | 32 |
| 4.5 运输问题的自编指令 ..... | 33 |
| 4.6 例题计算程序 .....    | 34 |

|                                          |    |
|------------------------------------------|----|
| <b>第 5 章 整数规划</b>                        |    |
| 5.1 分枝定界法的 MATLAB 表述                     | 37 |
| 5.2 整数规划的自编指令                            | 40 |
| 5.3 例题计算程序                               | 41 |
| <b>第 6 章 指派问题</b>                        |    |
| 6.1 总论                                   | 47 |
| 6.2 指派问题的 MATLAB 算法                      | 49 |
| 6.2.1 平衡指派问题                             | 49 |
| 6.2.2 不平衡指派问题                            | 50 |
| 6.2.3 障碍指派问题                             | 51 |
| 6.2.4 最大指派问题                             | 51 |
| 6.3 用自编指令 assign(n,e)解指派问题               | 53 |
| 6.4 用自编指令 assign1(n,xx0,mx,i,j)调整指派方阵    | 54 |
| 6.5 例题计算程序                               | 56 |
| <b>第 7 章 网络最大流问题</b>                     |    |
| 7.1 网络最大流                                | 60 |
| 7.2 用自编指令 mxfl(m,n,aeq,ub)求网络最大流         | 66 |
| 7.3 最小费用最大流                              | 67 |
| 7.4 用自编指令 mixfl(m,n,f,fl,aeq,ub)求最小费用最大流 | 69 |
| 7.5 例题计算程序                               | 70 |
| <b>第 8 章 网络最短路求法</b>                     |    |
| 8.1 网络弧集和权矩阵                             | 75 |
| 8.2 网络最短距离的 Floyd 算法                     | 76 |
| 8.3 用自编指令 fld(n,w)求距离矩阵                  | 76 |
| 8.4 两点间的最短路                              | 78 |
| 8.5 用自编指令 stlin(n,i,j,d)求最短路             | 78 |
| 8.6 例题计算程序                               | 84 |
| <b>第 9 章 关键线路法</b>                       |    |
| 9.1 关键线路法                                | 87 |
| 9.2 关键线路的 MATLAB 算法                      | 89 |
| 9.3 用自编指令 kylin(n,d)求关键线路                | 90 |
| 9.4 例题计算程序                               | 94 |

|                                     |     |
|-------------------------------------|-----|
| <b>第 10 章 多目标规划</b>                 |     |
| 10.1 介绍指令 fgoalattain .....         | 97  |
| 10.2 多目标规划问题 .....                  | 97  |
| 10.3 例题计算程序 .....                   | 102 |
| <b>第 11 章 非线性规划</b>                 |     |
| 11.1 非线性一元函数的最小值 .....              | 105 |
| 11.2 无约束非线性多元函数的最小值 .....           | 105 |
| 11.3 有约束非线性多元函数的最小值 .....           | 106 |
| 11.4 例题计算程序 .....                   | 109 |
| <b>第 12 章 矩阵对策</b>                  |     |
| 12.1 对策的三个基本要素 .....                | 112 |
| 12.2 矩阵对策 .....                     | 113 |
| 12.3 赢得矩阵的最大最小值和最小最大值 .....         | 114 |
| 12.4 矩阵对策的最优纯策略 .....               | 115 |
| 12.5 矩阵对策中的混合策略 .....               | 118 |
| 12.6 矩阵对策的线性规划求解法 .....             | 119 |
| 12.7 用自编指令 matp(m,n,a)求矩阵对策的解 ..... | 122 |
| 12.8 本章小结 .....                     | 124 |
| 12.9 例题计算程序 .....                   | 125 |
| <b>第 13 章 动态规划</b>                  |     |
| 13.1 动态规划简介 .....                   | 128 |
| 13.2 动态规划的应用 .....                  | 131 |
| 13.3 例题计算程序 .....                   | 147 |
| <b>附录 1 MATLAB 简介</b> .....         | 158 |
| <b>附录 2 自编指令</b> .....              | 165 |
| <b>参考文献</b> .....                   | 171 |
| <b>后 记</b> .....                    | 173 |

# 第 1 章 函数的运算和作图

## 1.1 函数的构成

在 MATLAB 语言中,通常用下述方法构成函数.

例如 函数  $f = x^2 - 2\sin x$  用语句

$f = 'x^2 - 2 * \sin(x)';$  构成.

如要计算函数在  $x = \pi/2$  的值,则用语句

$x = \pi/2;$

$f0 = \text{eval}(f)$

$f0 = 0.4647$ ,  $f0$  为函数在  $x = \pi/2$  的值,  $\pi$  为圆周率,  $\text{eval}$  为计算函数值的指令.

例如 函数  $f = 2x_1x_2 + x_2^2 - 3$ , 这是一个二元函数, 用语句

$\text{fun} = '2 * x(1) * x(2) + x(2)^2 - 3';$  构成

如要计算  $x_1 = 1, x_2 = 3$  的函数值, 则用语句

$x = [1, 3];$

$\text{fu1} = \text{eval}(\text{fun});$

$\text{fu1} = 12$   $\text{fu1}$  为函数在  $x = [1, 3]$  的值.

例如 函数组  $f_1 = x_1x_2 + x_2^2, f_2 = \exp(x_1 + x_2^3)$ , 这是一个二元函数组, 用语句

$g = '[x(1) * x(2) + x(2)^2, \exp(x(1) + x(2)^3)]';$  构成.

如要求函数组在  $x_1 = 1, x_2 = 2$  的值, 则用语句

$x = [1, 2];$

$g1 = \text{eval}(g)$

$g1 = 1.0 * e + 003 * [0.0060, 8.1031]$   $g1$  为函数组在  $x = [1, 2]$  的值, 此处  $g1$  实为  $10^{-3} * [0.0060, 8.1031] = [6, 8103.1]$ .

在本书给出的程序中, 大多用上述方法构成函数. 上述计算放在程序 ch11 中.

## 1.2 函数作图

$\text{fplot}('2 * \exp(x) * \sin(x)', [0, 2 * \pi])$

$\text{title}('y = 2 * \exp(x) * \sin(x)');$  作  $y = 2e^x \sin x$  在  $[0, 2\pi]$  上的图形. (图 1.1)

$x = 0:0.2:5;$

$\text{plot}(x, \cos(x) * \cosh(x) - 1);$  作  $y = \cos x \cosh x - 1$  在  $[0, 5]$  上的图形. (图 1.2)

$\text{title}('y = \cos(x) * \cosh(x) - 1')$  图上加标题.

$x = -9:0.1:11;$



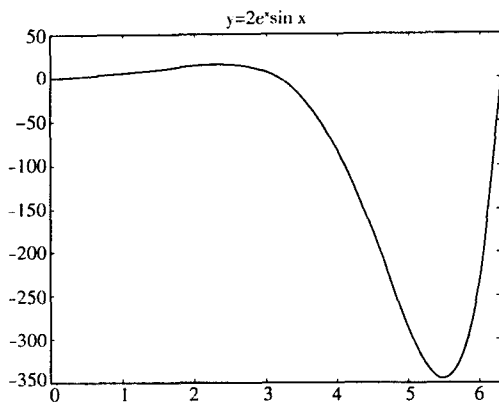


图 1.1

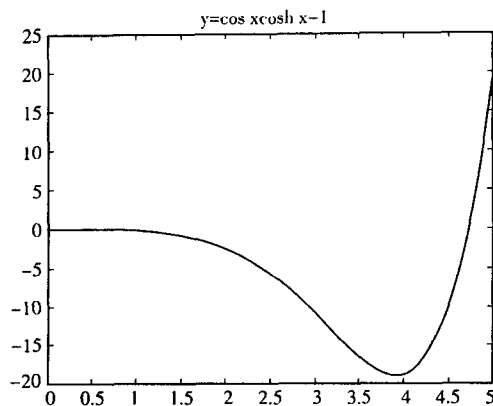


图 1.2

`plot(x, x.^3-1);` 作  $y=x^3-1$  在  $[-9,11]$  上的图形. (图 1.3)

`title('y=x^3-1')` 图上加标题.

`f='2 * exp(-x) * sin(x)';`

`fplot(f,[0,8]);` 作  $y=2e^{-x}\sin x$  在  $[0,8]$  上的图形. (图 1.4)

`title('y=2exp(-x) * sin(x)'), xlabel('x');` 图上加标题.

`f1='x.^3+10 * x.^2-2 * sin(x)-50';`

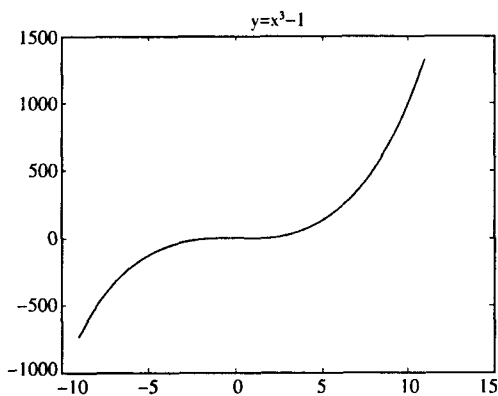


图 1.3

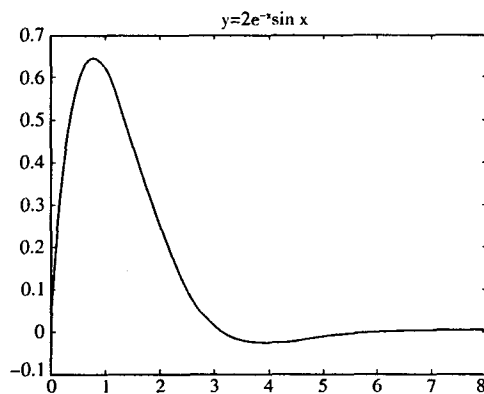


图 1.4

`fplot(f1,[-12,5]);` 作  $y=x^3+10x^2-2\sin x-50$  在  $[-12,5]$  上的图形. (图 1.5)

`title('y=x^3+10x^2-2sin(x)-50')` 图上加标题.

`grid;` 图上加划网格.

这里介绍了两个绘图指令 `fplot` 和 `plot`. 指令 `title` 是绘图标. 指令 `grid` 是在图中增划网格. 上述绘图程序放在 `ch12` 中.

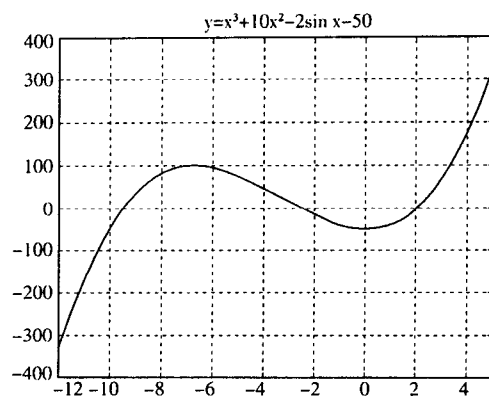


图 1.5

### 1.3 多项式运算

```

q1=[1,3,5];q2=[2,4,6]; % 此处[1,3,5]=x^2+3x+5,[2,4,6]=2x^2+4x+6.
q1p=poly2sym(q1) % q1p 为 q1 表述的多项式.
q2p=poly2sym(q2) % q2p 为 q2 表述的多项式.
q3=conv(q1,q2); % q3=(x^2+3x+5)*(2x^2+4x+6).
q3p=poly2sym(q3) % q3p 为 q3 表述的多项式.
q4=deconv(q3,q1); % q4=q3/q1.
q4p=poly2sym(q4) % q4p 为 q4 表述的多项式.
q5=[2, -6, 3, 0, 7];
q5p=poly2sym(q5) % q5p 为 q5 表述的多项式.
q6=polyder(q5); % q6 为 q5 的导数.
q6p=poly2sym(q6) % q6p 为 q6 表述的多项式.
q7=roots(q5) % q7 为 q5=0 的根.
q8=polyval(q5,1) % q8=q5(1). 多项式求值.
a=[1,2;3,4];
q9=polyvalm(q5,a) % q9=q5(a). 多项式求值,结果为矩阵.
计算结果显示:
q1p = x^2 + 3 * x + 5
q2p = 2 * x^2 + 4 * x + 6
q3p = 2 * x^4 + 10 * x^3 + 28 * x^2 + 38 * x + 30 % q3p=q1p * q2p.
q4p = 2 * x^2 + 4 * x + 6 % q4p=q3p/q1p.
q5p = 2 * x^4 - 6 * x^3 + 3 * x^2 + 7
q6p = 8 * x^3 - 18 * x^2 + 6 * x % q6p=diff(q5p).
q7 = 1.9322 + 0.4714i

```

```

1.9322 - 0.4714i
-0.4322 + 0.8355i
-0.4322 - 0.8355i
q8 = 6
q9 = 204 286
      429 633

```

上述计算程序放在 ch13 中.

## 1.4 微积分符号运算

在微积分运算中,如果存在准确解,符号运算将给出准确解.

```

syms x h
dc=limit((cos(x+h)-cos(x))/h,h,0)
% 求(cos(x+h)-cos(x))/h 在 h 趋于 0 的极限.
f=sym('x+2-x^2'); % f=x+2-x^2.
df=diff(f) % df 为 f 的一阶导数.
df2=diff(f,2) % df2 为 f 的二阶导数.
tf=int(f); % tf 为 f 的不定积分.
tf2=int(f,-1,2) % tf2 为 f 从 -1 到 2 的定积分.
x1=solve('sin(x)=1/2') % 方程 sinx=1/2 求根.
x2=solve('x^3-1=0') % 方程 x^3-1=0 求根.
y1=dsolve('Dy=5') % 微分方程 dy/dt=5 求解.
y2=dsolve('Dy=x','x') % 微分方程 dy/dx=x 求解. x 指定自变量.
y3=dsolve('D2y=1+Dy') % 微分方程 d2y/dt2=1+dy/dt 求解.
y4=dsolve('D2y=1+Dy','y(0)=1','Dy(0)=0')
% 求上述微分方程在条件% y(0)=1,dy(0)/dt=0 的解.
[y5,y6]=dsolve('Dx=y+x','Dy=2*x')
% 求微分方程组 dx/dt=y+x,dy/dt=2*x 的解. 此处 x=y5,y=y6.
[y7,y8]=dsolve('Dx=y+x','Dy=2*x','x(0)=0,y(0)=1')
y9=dsolve('Dy=-2*y+2*x^2+2*x','y(0)=1','x')
% y9 是 y'=-2y+2x^2+2x,y(0)=1 的解.
计算结果显示:
dc = -sin(x)
df = 1-2*x
df2 = -2
tf2 = 9/2
x1 = 1/6*pi
x2 = [ 1 ]

```

```

[ -1/2+1/2 * i * 3^(1/2)]
[ -1/2-1/2 * i * 3^(1/2)]
y1 = 5 * t + C1
y2 = 1/2 * x^2 + C1
y3 = -t + C1 + C2 * exp(t)
y4 = -t + exp(t)
y5 = 1/3 * C1 * exp(-t) + 2/3 * C1 * exp(2 * t) + 1/3 * C2 * exp(2 * t) - 1/3 * C2 *
exp(-t)
y6 = 2/3 * C1 * exp(2 * t) - 2/3 * C1 * exp(-t) + 2/3 * C2 * exp(-t) + 1/3 * C2 *
exp(2 * t)
y7 = 1/3 * exp(2 * t) - 1/3 * exp(-t)
y8 = 2/3 * exp(-t) + 1/3 * exp(2 * t)
y9 = x^2 + exp(-2 * x)
上述计算程序放在 ch14 中.

```

## 1.5 定积分近似计算

```

q6=quad('x+2-x.^2',-1,2)
quad('f',a,b)是求 f 从 a 到 b 的定积分近似值的指令,它基于低阶的辛普森法.
x=-1:0.01:2;
y=x+2-x.^2;
q7=trapz(x,y)
trapz(x,y)是求 y 从 -1 到 2 的定积分近似值的指令,它基于梯形法公式.
q8=int('x+2-x.^2',-1,2) q8 是上述定积分符号解,是准确值.
q9=quad('1./(x.^2+4*x+5)',0,1)
x=0:0.01:1;
y=1./(x.^2+4*x+5);
q10=trapz(x,y)
q11=int('1/(x.^2+4*x+5)',0,1)
q11=double(q11) 给出 q11 的近似数值.
计算结果显示:
q6 = 4.5000
q7 = 4.5000
q8 = 9/2
q9 = 0.1419
q10 = 0.1419
q11 = atan(3)-atan(2)
q11 = 0.1419

```

上述计算程序放在 ch15 中.

## 1.6 例题计算程序

**% ch11 函数构成**

```
f='x^2-2*sin(x)';  
x=pi/2;  
f0=eval(f) % 函数在 pi/2 的值.  
fun='2*x(1)*x(2)+x(2)^2-3';  
x=[1,3];  
f1=eval(fun) % 函数在 x=[1,3]的值.  
g='[x(1)*x(2)+x(2)^2,exp(x(1)+x(2)^3)]';  
x=[1,2];  
g1=eval(g) % 函数在 x=[1,2]的值.
```

**% ch12 函数作图**

```
fplot('2*exp(x)*sin(x)',[0,2*pi])  
title('y=2*exp(x)*sin(x)'); % y=2e^x sin x 在[0,2pi]上的图形.  
pause; % 程序暂停,敲任意键程序继续.  
x=0:0.2:5;  
plot(x,cos(x).*cosh(x)-1); % y=cosxcoshx-1 在[0,5]上的图形.  
title('y=cos(x)*cosh(x)-1')  
pause;  
x=-9:0.1:11;  
plot(x,x.^3-1); % y=x^3-1 在[-9,11]上的图形.  
title('y=x^3-1')  
pause;  
f='2*exp(-x).*sin(x)';  
fplot(f,[0,8]); % y=2e^x sin x 在[0,8]上的图形.  
title('y=2exp(-x)*sin(x)'),xlabel('x'); pause;  
f1='x.^3+10*x.^2-2*sin(x)-50';  
fplot(f1,[-12,5]); % y=x^3+10x^2-2sinx-50 在[-12,5]上的图形.  
title('y=x^3+10x^2-2sin(x)-50')  
% 这里介绍了两个绘图指令 fplot 和 plot. 指令 title 是绘图标.  
grid; % 图中增划网格.  
% 图形显示:
```

**% ch13 多项式运算**

```
q1=[1,3,5];q2=[2,4,6];
```

```

q1p=poly2sym(q1) % q1p 为 q1 表述的多项式.
q2p=poly2sym(q2) % q2p 为 q2 表述的多项式.
q3=conv(q1,q2); % q3=(x^2+3x+5)*(2x^2+4x+6).
q3p=poly2sym(q3) % q3p 为 q3 表述的多项式.
q4=deconv(q3,q1); % q4=q3/q1.
q4p=poly2sym(q4) % q4p 为 q4 表述的多项式.
q5=[2,-6,3,0,7];
q5p=poly2sym(q5) % q5p 为 q5 表述的多项式.
q6=polyder(q5); % q6 为 q5 的导数.
q6p=poly2sym(q6) % q6p 为 q6 表述的多项式.
q7=roots(q5) % q7 为 q5=0 的根.
q8=polyval(q5,1) % q8=q5(1). 多项式求值.
a=[1,2,3,4];
q9=polyvalm(q5,a) % q9=q5(a). 多项式求值,结果为矩阵.
% 计算结果显示:

```

#### % ch14 微积分符号运算

% 在微积分运算中,如果存在准确解,符号运算将给出准确解.

```
syms x h
```

```
dc=limit((cos(x+h)-cos(x))/h,h,0)
```

% 求  $(\cos(x+h)-\cos(x))/h$  在  $h$  趋于 0 的极限.

```
f=sym('x+2-x^2'); % f=x+2-x^2.
```

```
df=diff(f) % df 为 f 的一阶导数.
```

```
df2=diff(f,2) % df2 为 f 的二阶导数.
```

```
tf=int(f); % tf 为 f 的不定积分.
```

```
tf2=int(f,-1,2) % tf2 为 f 从 (-1) 到 2 的定积分.
```

```
x1=solve('sin(x)=1/2') % 方程 sin(x)=1/2 求根.
```

```
x2=solve('x^3-1=0') % 方程 x^3-1=0 求根.
```

```
y1=dsolve('Dy=5') % 微分方程 dy/dt=5 求解.
```

```
y2=dsolve('Dy=x','x') % 微分方程 dy/dx=x 求解. x 指定自变量.
```

```
y3=dsolve('D2y=1+Dy') % 微分方程 d2y/dt2=1+dy/dt 求解.
```

```
y4=dsolve('D2y=1+Dy','y(0)=1','Dy(0)=0')
```

% 求上述微分方程在条件  $y(0)=1, dy(0)/dt=0$  的解.

```
[y5,y6]=dsolve('Dx=y+x','Dy=2*x')
```

% 求微分方程组  $dy/dt=y+x, dy/dt=2x$  的解.

```
[y7,y8]=dsolve('Dx=y+x','Dy=2*x','x(0)=0,y(0)=1')
```

```
y9=dsolve('Dy=-2*y+2*x^2+2*x','y(0)=1','x')
```

% y9 是  $y'=-2y+2x^2+2x, y(0)=1$  的解.

% 计算结果显示：

% ch15 积分近似计算

q6=quad('x+2-x^2',-1,2)

% quad('f',a,b)是求 f 从 a 到 b 的定积分近似值的指令,它基于低阶的辛普森法.

x=-1:0.01:2;

y=x+2-x.^2;

q7=trapz(x,y)

% trapz(x,y)是求 y 从 -1 到 2 的定积分近似值的指令,它基于梯形法公式.

q8=int('x+2-x^2',-1,2) % q8 是上述定积分符号解,是准确值.

q9=quad('1./(x.^2+4\*x+5)',0,1)

x=0:0.01:1;

y=1./(x.^2+4\*x+5);

q10=trapz(x,y)

q11=int('1/(x^2+4\*x+5)',0,1)

q11=double(q11) % 给出 q11 的近似数值.

% 显示计算结果：

## 第2章 矩 阵

在 MATLAB 语言中, 矩阵占有非常重要的地位, 本章将择要介绍. 对参加运算的矩阵, 其要求与线性代数相同.

### 2.1 矩阵的运算和生成

```
a=[3 4 -1 1;6 5 0 7;1 -4 7 -1;2 -4 5 -6]
b=[1 2 4 6;7 9 16 -5;8 11 20 1;10 15 28 13]
c=[1 3 5 7]';
d0=a+b % 矩阵相加.
d1=a-b % 矩阵相减.
d2=3*a % 数乘矩阵.
d3=3.*a % 数点乘矩阵, d2=d3.
d4=3+a % a 的所有元素都加 3.
d5=a*b % 矩阵相乘. 要求 a 的列数等于 b 的行数.
d6=a.*b % 矩阵点乘为对应元素相乘, 要求 a 和 b 的阶数相同.
d7=det(a) % 方阵 a 的行列式.
d8=rank(a) % 求矩阵 a 的秩.
d9=rank(b) % 求矩阵 b 的秩.
d10=inv(a) % 求 a 的逆矩阵.
d11=sum(a) % a 的每列相加, 构成一个行矩阵.
d12=sum(sum(a)) % 为 a 矩阵各元素之和.
d13=max(a) % 为 a 矩阵每列的最大值, 构成一个行矩阵.
d14=max(a') % 为 a 矩阵每行的最大值, 构成一个行矩阵.
d15=min(a) % 为 a 矩阵每列的最小值, 构成一个行矩阵.
d16=min(a') % 为 a 矩阵每行的最小值, 构成一个行矩阵.
d17=max(max(a)) 为矩阵 a 元素中的最大值.
d18=min(min(a)) 为矩阵 a 元素中的最小值.
d19=eye(6) % d19 为 6 阶单位矩阵, 指令生成.
d20=zeros(6) % d20 为 6 阶零矩阵, 指令生成.
d21=zeros(5,8); % d21 为 5 行 8 列零矩阵, 指令生成.
d22=ones(3) % d22 为 3 阶全 1 阵, 指令生成.
d23=ones(3,6) % d23 为 3 行 6 列全 1 阵, 指令生成.
d24=0:5 % d24=[0,1,2,3,4,5], 从 0 到 5, 步长为 1, 指令生成.
```



```

d25=0:0.1:1 % d25 为从 0 到 1, 步长为 0.1 的行矩阵,指令生成.
d26=b/a % a 为方阵,d26=b * inv(a).
d27=a\b % a 为方阵,d27=inv(a) * b.
d28=a.*b % a、b 为同阶矩阵,或者 b 为常数;d28=[aij*bij].
d29=a./b % a、b 为同阶矩阵,d29=[aij/bij].
d30=b.\a % a、b 为同阶矩阵,d30=[aij/bij].

```

计算结果显示:

```

a= 3    4   -1    1
    6    5    0    7
    1   -4    7   -1
    2   -4    5   -6
b=  1    2    4    6
    7    9   16   -5
    8   11   20    1
   10   15   28   13
d0=  4    6    3    7
    13   14   16    2
    9    7   27    0
   12   11   33    7
d1=  2    2   -5   -5
    -1   -4  -16   12
    -7  -15  -13   -2
    -8  -19  -23  -19
d2=  9    12   -3    3
    18   15    0   21
    3  -12   21   -3
    6  -12   15  -18
d3=  6    7    2    4
    9    8    3   10
    9   -1   10    2
    5   -1    8   -3
d4=  9    12   -3    3
    18   15    0   21
    3  -12   21   -3
    6  -12   15  -18

```