

21世纪高等院校创新教材

SHUZHIFENXIJISHIYAN

数值分析 及实验

杜廷松 沈艳军 覃太贵 主编

21世纪高等院校创新教材

SHUZHIFENXIJISHIYAN

数值分析及实验



科学出版社

www.sciencep.com

·21 世纪高等院校创新教材·

数值分析及实验

杜廷松 沈艳军 覃太贵 主编

科学出版社

北 京

内 容 简 介

本书结合 Matlab 的使用全面介绍了常用的数值计算方法与技术. 内容包括线性代数方程组的数值解法、方程(组)求根的迭代法、插值法、曲线拟合和函数逼近初步、数值微积分、矩阵特征值与特征向量的计算等, 每部分均有代表性的例题和习题. 本书最明显的特点是对数值分析理论部分着重阐明构造算法的基本思想与原理, 既注重理论的严谨性, 又注重方法的实用性.

本书的读者对象, 一是本科信息与计算科学、数学与应用数学专业和其他理工科的高年级学生, 二是研究生中的工学硕士、工程硕士.

图书在版编目(CIP)数据

数值分析及实验/杜廷松, 沈艳军, 覃太贵主编. - 北京: 科学出版社, 2006
(21 世纪高等院校创新教材)

ISBN 7-03-016864-X

I. 数… II. ①杜…②沈…③覃… III. 数值计算 - 高等学校 - 教材
IV. O241

中国版本图书馆 CIP 数据核字 (2006) 第 008408 号

责任编辑: 江 兰 / 责任校对: 王望容

责任印制: 高 嵘 / 封面设计: 曹 刚 覃杜东

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

武汉大学出版社印刷总厂印刷

科学出版社发行 各地新华书店经销

*

2006 年 2 月第 一 版 开本: 787×1092 1/16

2006 年 2 月第一次印刷 印张: 15

印数: 1~4 000

字数: 368 000

定价: 22.80 元

(如有印装质量问题, 我社负责调换)

前 言

随着科学技术的不断发展,越来越多的问题靠数学解析解已远远不能够满足要求,数值解在解决实际问题中的地位显得日益重要,因此,数值分析这门课程就得到了更多的重视.不仅数学类专业开设数值分析课,很多工科专业也开设了数值分析课.长期以来,在数值分析教学中一个很大的遗憾是:将数值分析作为一门纯粹的理论课程来讲授,不重视数值分析的上机实验环节,不重视如何将编程语言与教材中给出的算法结合起来,如何去解决一个实际问题.这样学生就很难对数值分析的一些重要特征(如计算速度和数值稳定性问题等)有深刻的理解、体会.虽然已有的少数参考教材附有一些算法程序,但大多是用C语言或Fortran语言给出的,我们发现除了少数的学生外,其他多数学生还必须在真正学透C语言或Fortran语言后,才能具有独立编程的能力.因此,只讲授理论不重视实践教学的传统教学方式让学生感觉不出学习数值分析这门应用非常广泛的学科的乐趣了.究其原因,在于很多数值算法涉及复杂的程序结构,使用C语言或Fortran语言编程效率比较低,很难在有限的教学课时中包含一些复杂而又非常重要的数值计算问题,加上教学软硬件的限制,造成传统教学中一般都只讲算法原理和误差分析,而不涉及上机实验.

从20世纪90年代开始,国内出现了一些基于数学软件的数值分析教材,但遗憾的是这类教材大多只是用一个附录介绍软件,并没有将数学软件融入到教材中,本教材的目的是努力将数值分析理论学习与利用Matlab编程上机实验紧密结合起来,提供一本真正基于Matlab的、真正将Matlab融合到教材中去实现数值分析的数值算法的教材.

本教材的适用对象,一是本科信息与计算科学专业、数学与应用数学专业和其他理工科的高年级学生,二是研究生中的工学硕士、工程硕士.无疑,这是一个以“使用”数值算法为自己专业服务的群体.当然,也更鼓励他们数值算法的“创造”做出自己的贡献.不管怎样,通过本课程的数值分析与实验的学习,我们既可以看到计算机求解数学问题的机理过程,也可感受到数值计算如何为数学问题的求解开辟了另一条康庄大道.

本书结合Matlab的使用全面介绍了常用的数值计算方法与技术,内容包括Matlab简介、数值分析的若干基本概念、线性代数方程组的数值解法(直接法和迭代法)、方程(组)求根的迭代法、插值法、曲线拟合和函数逼近初步、数值微积分、常微分方程的数值解法及矩阵特征值与特征向量的计算等九章.书中各章附有精心挑选的例题讲解,所选例题思路独到,极具代表性.每章设计的习题适合手工计算,有助于培养学生的解题能力,巩固和加深学生对理论部分的基本概念、基本原理、基本方法等内容的理解.每一章的数值实验部分不仅对各有关程序的编写、调用方法做了详尽的介绍,还注重介绍有关的数值计算方法的使用条件、计算机输出结果的含义、图形可视化解的显示方法.通过给出完整的程序与实例的介绍,力图使读者全面学会如何分析问题、如何动手编程计算及如何对获得的计算结果进行分析,以提高学生上机实验动手能力和科学计算创新能力.

本教材最明显的特点是:对数值分析理论部分着重阐明构造算法的基本思想与原理;既注重理论的严谨性,又注重方法的实用性;力求写得简明,努力做到深入浅出,通俗易懂.每一章数值实验部分尽量写得详细、丰富,这种写法基于我们对这门课程的教学理念的新认识.我们

认为,只有老师“讲授”,学生“不做实验”,那是学不透多少东西的,更谈不上培养具有真实能力和创新能力的人才.简明带来可读性,可学性,我们用新的眼光对传统的教材内容精心取舍,并力求用现代风格的语言加以演绎.实验部分更是作者经过精心地加工、链接和再创作的结果,我们希望把它制作成一个培养、训练学生独立思考能力、分析处理问题能力和创造能力的“平台”,我们相信,这种写法对学生的学习是有帮助的,对教师的教学是方便的.

本书具有清晰的积木式结构,各章具有相对的独立性,因此教师容易取舍,可构成不同层次、不同要求的教学方案.

本书承蒙武汉大学数学与统计学院博士生导师费浦生教授的主审,作者对费教授提出的许多宝贵和精辟的指导意见表示衷心的感谢.作者也从多年教学所采用和参考国内多所著名大学的多本教材中受益匪浅,也无不受这些教材的启发和指引,在此,也向李庆扬、关治、陈景良、王能超、易大义等多位教授表示深深的敬意.

本书是作者在几年的数值分析教学实践的基础上编写而成的,在教学中注重由浅入深、由特殊到一般.在介绍方法和引进概念时力求揭示问题的实质,对于方法和概念之间着重阐明其联系;着重阐明算法的基本思想.在多年的实践教学中,作者作了如下一些新尝试:注重方法性和实用性;注重对实验上机环节的教学;逼近和近似思想、迭代思想、连续问题离散化的思想一直贯穿于教学的始终.

编写和出版本书得到了三峡大学理学院数学系同仁及各级领导的大力支持,同时也得到了三峡大学研究生课程建设基金的资助,对此我们表示衷心的感谢.

作为一本希望作些新尝试的教材,尽管编者作了认真的努力,但疏漏和不妥之处一定还不少,恳请老师、同学以及其他专家批评指正.

编 者

2005年11月

目 录

第1章 Matlab 简介	1
§ 1.1 向量和矩阵的产生	1
§ 1.2 运算符及矩阵运算	5
§ 1.3 函数库	7
§ 1.4 Matlab 程序设计初步	9
第2章 数值分析的若干基本概念	16
§ 2.1 数值分析的研究对象	16
§ 2.2 数值计算的误差	18
§ 2.3 数值计算中误差的传播	20
§ 2.4 数值稳定性与避免误差伤害	24
§ 2.5 舍入误差与数值稳定性数值实验	27
习 题	29
实验题	30
第3章 线性代数方程组的数值解法	31
§ 3.1 引言	31
§ 3.2 Gauss 消元法	31
§ 3.3 矩阵的直接分解法	35
§ 3.4 三对角方程组的求解方法	40
§ 3.5 向量范数和矩阵范数	45
§ 3.6 解线性代数方程组的迭代法	48
§ 3.7 数值实验	56
习 题	64
实验题	67
第4章 非线性方程求根、非线性方程组数值解法初步	71
§ 4.1 问题的提出	71
§ 4.2 区间搜索法及二分法	71
§ 4.3 迭代法	73
§ 4.4 迭代加速技术	77
§ 4.5 Newton 法	80
§ 4.6 弦截法	83
§ 4.7 非线性方程组的解法	84
§ 4.8 数值实验	89
习 题	95
实验题	96

第5章 插值法	98
§ 5.1 代数插值问题	98
§ 5.2 Lagrange 插值	99
§ 5.3 差商与 Newton 插值公式	106
§ 5.4 差分与等距节点插值公式	109
§ 5.5 Hermite 插值	112
§ 5.6 分段低次插值	116
§ 5.7 三次样条插值	119
§ 5.8 多元函数插值	125
§ 5.9 数值实验	126
习 题	132
实验题	134
第6章 曲线拟合、函数逼近初步	136
§ 6.1 曲线拟合的最小二乘法	136
§ 6.2 $\ \cdot\ _1$ 和 $\ \cdot\ _\infty$ 意义下的线性拟合	142
§ 6.3 超定方程组的最小二乘解	143
§ 6.4 最佳平方逼近	145
§ 6.5 最佳一致逼近	147
§ 6.6 数值实验	149
习 题	154
实验题	155
第7章 数值微积分	156
§ 7.1 数值积分问题的提出	156
§ 7.2 插值型求积公式	157
§ 7.3 Newton-Cotes 公式	159
§ 7.4 Romberg 求积方法	163
§ 7.5 Gauss 求积公式	166
§ 7.6 数值微分	172
§ 7.7 数值实验	176
习 题	180
实验题	182
第8章 常微分方程数值解法	184
§ 8.1 Euler 法	184
§ 8.2 Runge-Kutta 方法	187
§ 8.3 线性多步法	193
§ 8.4 一阶方程组和高阶方程	196
§ 8.5 单步法的收敛性与稳定性	198
§ 8.6 数值实验	200
习 题	207
实验题	209

第9章 矩阵特征值与特征向量的计算	210
§ 9.1 问题的提出	210
§ 9.2 乘幂法和反幂法	211
§ 9.3 Jacobi 方法	215
§ 9.4 QR 算法	220
§ 9.5 数值实验	224
习 题	228
实验题	229
参考文献	230

第1章 Matlab 简介

Matlab 是 Matrix Laboratory 的英文缩写,它是由美国 Mathwork 公司于 1967 年推出的适用于不同规格计算机和各种操作系统的数学软件包,现已发展成为一种功能强大的计算机语言,特别适用于科学和工程计算.结合本课程的需要,本章仅对 Matlab 的有关内容做简要的介绍,要想更详细地了解 Matlab,可参看文献[26]~文献[29].

§ 1.1 向量和矩阵的产生

一、向量的产生

1. 冒号运算符生成向量

例 1.1 生成一个从 0 到 π 的行向量,步长为 $\pi/4$.

```
>>x=0:pi/4:pi
x=
    0    0.7854    1.5708    2.3562    3.1416
```

2. 线性等分函数等分建立向量

(1) linspace(n1,n2) 包括 n1,n2 元素,生成 100 维向量

(2) linspace(n1,n2,n) 包括 n1,n2 元素,生成 n 维向量

二、矩阵的产生

1. 简易矩阵

```
>>x=[1 2 3 4 5 6 7 8;4 5 6 7 8 9 10 11] %以“;”区隔开两行元素
x=
     1     2     3     4     5     6     7     8
     4     5     6     7     8     9    10    11
>>x(3) %提取 x 的第 3 个元素
ans=
     2
>>x(1:5); %x 的前 5 个元素
>>x(10:end); %x 的第 10 个元素后的元素
>>x(find(x>5)); %x 中大于 5 的元素
>>x(4)=100; %给 x 的第 4 个元素重新赋值
>>x(3)=[]; %删除第 3 个元素
>>x(16)=1; %加入第 16 个元素
```

2. 由多个向量生成矩阵

例 1.2 由 x, y, z 向量生成一个矩阵.

```
>>x=(0:0.2:1.0)';
```

```
>>y=exp(-x).*sin(x);
>>z=(1:size(x))';
>>[x,y,z]
ans =
         0         0    1.0000
    0.2000    0.1627    2.0000
    0.4000    0.2610    3.0000
    0.6000    0.3099    4.0000
    0.8000    0.3223    5.0000
    1.0000    0.3096    6.0000
```

3. 用函数建立矩阵

在 Matlab 中不需要预先说明矩阵和向量的维数,但经常要使用维数.对此有两个测量矩阵大小的函数经常用到:

`n=length(A)` 取矩阵 A 的行数和列数的最大值 n

`[m,n]=size(A)` 取矩阵 A 的行数 m 和列数 n

由此可见,当 x 为向量时, $n=length(x)$ 为 x 的维数.

用于建立矩阵的函数有下列几种:

(1) 函数 `eye` 产生单位矩阵.例如:

`eye(n)` 为 n 阶单位方阵, n 是正整数

`eye(m,n)` 为 $m \times n$ 阶单位矩阵

`eye(size(A))` 为与矩阵 A 同阶的单位矩阵

其中的单位矩阵理解为对角线元素是 1,其他元素是 0 的矩阵.

(2) 函数 `zeros` 和 `ones` 分别产生 0 和 1 矩阵.例如:

`zeros(n)` 为 n 阶 0 方阵

`zeros(m,n)` 为 $m \times n$ 阶 0 矩阵

`zeros(size(A))` 为与矩阵 A 同阶的 0 矩阵

函数 `ones` 与此类同.

(3) 函数 `rand(m,n)` 产生 $m \times n$ 阶随机数矩阵.

(4) 函数 `diag(A)`,`tril(A)`,`triu(A)` 分别取矩阵 A 的对角、下三角、上三角部分.其中,三角矩阵包含对角部分.

4. 下标编辑、调用矩阵

例 1.3 下标修改矩阵元素.

```
>>A=[1,2,3;4,5,6;7,8,9]
```

```
A=
```

```
    1    2    3
    4    5    6
    7    8    9
```

```
>>A(3,3)=A(1,3)+A(3,1)
```

```
A=
```

1	2	3
4	5	6
7	8	10

Matlab的一个重要特点是可以下标的方法调用矩阵的子矩阵. 例如, 设 A 是已知的 10×10 阶矩阵, 则

$A(:, 3)$	为 A 的第 3 列元素构成的列向量
$A(5, :)$	为 A 的第 5 行元素构成的行向量
$A(1:5, 3)$	为 A 的前 5 行的第 3 列元素构成的列向量
$A(1:5, 7:10)$	为 A 的前 5 行、第 7 到 10 列元素构成的子矩阵
$A([1\ 3\ 5], [2\ 4\ 6])$	为 A 的第 1, 3, 5 行、第 2, 4, 6 列元素构成的子矩阵
$A(:, 7:-1:3)$	为 A 的第 7, 6, 5, 4, 3 列元素构成的子矩阵

如此等等.

5. 稀疏矩阵生成:sparse

例 1.4 将矩阵

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \\ 4 & 3 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

变成稀疏矩阵.

```
>>x=[0 1 0;0 0 2;0 0 0;4 3 0;0 0 0];
```

```
>>sp=sparse(x)
```

```
sp=
```

(4,1)	4
(1,2)	1
(4,2)	3
(2,3)	2

例 1.5 构造一个带状矩阵

$$\begin{bmatrix} -4 & 1 & & \\ 1 & -4 & 1 & \\ & 1 & -4 & 1 \\ & & 1 & -4 \end{bmatrix}$$

```
>>n=4;
```

```
>>sp=sparse(1:n,1:n,-4*ones(1,n),n,n,n)
```

% 计算主对角线各元素

```
sp=
```

(1,1)	-4
(2,2)	-4
(3,3)	-4
(4,4)	-4

```
>>spl=sparse(2:n,1:n-1,ones(1,n-1),n,n,n-1)
```

%计算下次对角线各元素

```

spl=
    (2,1)      1
    (3,2)      1
    (4,3)      1
>>spu=spl'      %转置求上次对角线各元素
spu=
    (1,2)      1
    (2,3)      1
    (3,4)      1
>>ss=sp+spl+spu  %计算三条对角线各元素
ss=
    (1,1)     -4
    (2,1)      1
    (1,2)      1
    (2,2)     -4
    (3,2)      1
    (2,3)      1
    (3,3)     -4
    (4,3)      1
    (3,4)      1
    (4,4)     -4
>>sf=full(ss)
sf=
    -4     1     0     0
     1    -4     1     0
     0     1    -4     1
     0     0     1    -4

```

6. 带状稀疏矩阵生成:spdiags

例 1.6 构造一个如同例 1.5 的带状矩阵.

```

>>n=4;
>>p=ones(n,1);
>>sp2=spdiags([p,-4*p,p],[-1,0,1],n,n);
%[p,-4*p,p]三条对角线元素,[-1,0,1]三条对角线元素的位置
>>sf=full(sp2)
sf =
    -4     1     0     0
     1    -4     1     0
     0     1    -4     1
     0     0     1    -4

```

§ 1.2 运算符及矩阵运算

一、运算符

1. 算术运算

算术运算符:

“+”加 “-”减 “*”乘 “^”幂 “\”左除 “/”右除

2. 关系运算

关系运算符:

“<”小于 “<=”小于等于 “>”大于
“>=”大于等于 “==”等于 “~=”不等于

它们用于比较两个元素的大小关系,结果是1表明为真,结果是0表明为假.

3. 逻辑运算

逻辑运算符:

“&”与 “|”或 “~”非

它们用于元素或0-1矩阵的逻辑运算.

二、矩阵运算

1. 矩阵转置

```
>>x=[1,2,3]'
```

```
x=
```

```
1
```

```
2
```

```
3
```

```
>>A=[1    2    3  
      4    5    6  
      7    8    9];
```

```
>>B=A'
```

```
B=
```

```
1    4    7
```

```
2    5    8
```

```
3    6    9
```

2. 矩阵相加减

同阶矩阵相加减,对应元素相加减.

任何矩阵都可以和标量(即 1×1 阶矩阵)相加减,规则是矩阵的每一个元素和标量相加减.

3. 矩阵的乘法

在Matlab中,矩阵的乘、除、乘方及各种函数运算等有两种方式:

(1) 矩阵运算;

(2) 元素运算(.运算).

矩阵乘(*): 两个内维相同的矩阵相乘.

元素乘(.*)：两个同维的矩阵相应元素相乘。

例 1.7 比较矩阵的两种乘法。

```
>>A=[1 2 3;4 5 6; 7 8 9]
```

A=

1	2	3
4	5	6
7	8	9

```
>>B=eye(3);
```

```
>>C=A*B
```

C=

1	2	3
4	5	6
7	8	9

```
>>D=A.*B
```

D=

1	0	0
0	5	0
0	0	9

4. 矩阵相除

(1) 矩阵除. 有左除和右除两种, 分别用符号“\”和“/”表示. 例如, 若 A 和 B 都是 n 阶方阵, 且 A 非奇异, 则

$$A \setminus B = A^{-1}B \quad B/A = BA^{-1}$$

(2) 元素除. 元素左除为“.\”, 元素右除为“./”: 两个同维的矩阵相应元素相除.

5. 矩阵的乘方

(1) 方阵 A 的乘方用符号“^”表示.

当 p 是整数时, $A \wedge p$ 是 A 的 p 次幂; $p=0$ 时, $A \wedge 0$ 为与 A 同维单位矩阵; 当 p 为负整数时, 只有 A 非奇异才有意义.

例 1.8 求矩阵的整数乘方, 并验证.

```
>>A=fix(10*rand(3))
```

A=

9	4	4
2	8	0
6	7	8

```
>>AB=A^2
```

AB =

113	96	68
34	72	8
116	136	88

>> AA=A*A %验证上面AB的结果

AA=

113	96	68
34	72	8
116	136	88

(2) 矩阵 $A_{m \times n}$ 的 b 元素乘方 ($\wedge$) 相当于矩阵 A 中每一个元素的 b 次乘方.

例 1.9 求例 1.8 中矩阵 A 的元素运算乘方.

>> ABC=A.^2 %注意结果与例 1.8 中的 $A \wedge 2$ 不同

ABC=

81	16	16
4	64	0
36	49	64

§ 1.3 函 数 库

Matlab 拥有丰富的函数库,这里只列出常用的初等函数以及与本书有关的部分数值方法函数,供引用.

一、初等函数

- (1) sin——正弦, asin——反正弦, sinh——双曲正弦, asinh——反双曲正弦.
- (2) sqrt——开平方, log——自然对数, exp——以 e 为底的指数, log10——以 10 为底的对数.
- (3) abs——绝对值或复数模.
- (4) sign——正负号函数, max——最大分量, sum——元素和, min——最小分量.
- (5) ceil——向 $+\infty$ 取整, fix——向 0 取整, floor——向 $-\infty$ 取整, mod——模除求余, rem——求余数, round——向靠近整数取整.

二、矩阵函数

在 Matlab 中,对于数值分析常用的有关矩阵函数有如下几个:

rank(A)	秩
norm(A,1)	1-范数 $\ A\ _1$
norm(A)	2-范数 $\ A\ _2$
norm(A,inf)	无穷范数 $\ A\ _\infty$
norm(A,'fro')	Frobenius 范数 $\ A\ _F$
condest(A)	1-条件数 $\text{cond}(A)_1 = \ A^{-1}\ _1 \ A\ _1$
cond(A)	2-条件数 $\text{cond}(A)_2 = \ A^{-1}\ _2 \ A\ _2$
condest(A')	无穷条件数 $\text{cond}(A)_\infty = \ A^{-1}\ _\infty \ A\ _\infty$

因为 Matlab 中没有求无穷条件数的函数,所以用转置矩阵的 1-条件数计算.

三、多项式和插值拟合函数

表 1.1 多项式和插值拟合函数及其意义

函数	意义
roots(c)	求多项式 $p_n(x)=c_nx^n+\cdots+c_1x+c_0$ 的根. 输入系数 $c=(c_n,\cdots,c_1,c_0)$, 输出 $p_n(x)$ 的根
poly(A)	求方阵 A 的特征多项式. 输入 n 阶方阵 A , 输出特征多项式的系数 $c=(c_n,\cdots,c_1,c_0)$
polyval(c,x)	求多项式 $p_n(x)$ 在点列 x 上的值. x 可以是任何矩阵. 输入多项式的系数 c 和点列 x , 输出多项式的值
polyfit(x,y,n)	对于离散数据 $(x,y)=\{x_i,y_i\}$, 根据最小二乘原理求 n 次多项式拟合. 输入离散数据 (x,y) 和 n , 输出拟合多项式 $p_n(x)$ 的系数 c
spline(x,y,xi)	对于离散数据 $(x,y)=\{x_k,y_k\}$ 进行三次样条插值. 输入离散数据 (x,y) 和插值点列 x_i , 输出点列 x_i 上的三次样条函数值. 点列 x 单调增, x_i 在 x 之中
interp1(x,y,xi,method)	对于离散数据 $(x,y)=\{x_k,y_k\}$ 进行一维插值. 输入离散数据 (x,y) 、点列 x_i 和方法选择 $method$. 输出点列 x_i 上的函数值. 方法主要有两种: 'linear'——分段线性插值(可缺省); 'spline'——三次样条插值. 点列 x 单调增, x_i 在 x 之中. 这里的三次样条插值与函数 spline 等价

四、数值线性代数

表 1.2 求解数值线性代数的函数及其意义

函数	意义
det(A)	求方阵 A 的行列式的值
trace(A)	求矩阵 A 的迹, $\text{trace}(A)=\sum_i a_{ii}$
inv(A)	求非奇异方阵 A 的逆矩阵
lu(A)	矩阵的列主元 LU 三角分解
chol(A)	对称正定矩阵 A 的 Cholesky 分解 $A=R^T R$, R 是上三角阵. $R=\text{chol}(A)$ 产生上三角阵 R
qr(A)	求矩阵 A 的 QR 分解
eig(A)	求方阵 A 的特征值和特征向量

五、数值积分和常微分方程数值解

表 1.3 求解数值积分和常微分方程数值解的函数及其意义

函数	意义
quad	低阶数值积分. 用自适应 Simpson 方法进行数值积分, 相对误差 $\leq 10^{-3}$. $Q = \text{quad}('f', a, b)$ 求出 $\int_a^b f(x)dx$ 的近似值. f 是被积函数 $f(x)$ 的函数名. 一般用法: $[Q, cnt] = \text{quad}('f', a, b, tol, trace)$ 其中: a 和 b 可以是同长度的向量, 这时的结果 Q 也是同长度的向量; tol 是精度要求, $trace \neq 0$ 时作出被积函数的图形, tol 和 $trace$ 可以缺省; cnt 是区间 $[a, b]$ 的分段数.
quad8	高阶数值积分. 用自适应 9 点 Newton-Cotes 方法进行数值积分. 用法和参数说明同 quad
ode23	低阶法解常微分方程(组)初值问题: $\begin{cases} \frac{dy_i(t)}{dt} = f_i(t, y_1, \dots, y_n) & (i=1, 2, \dots, n; \quad t_0 \leq t \leq t_f) \\ y_i(t_0) = y_{i,0} \end{cases}$ 用二阶和三阶 Runge-Kutta 方法进行积分, 精度不低于 10^{-3}. 一般调用方法: $[T, Y] = \text{ode23}('F', t0, tf, Y0, tol, trace)$ 其中: F 是方程的右端函数(向量值函数); $[t0, tf]$ 是积分区间; $Y0$ 是初始值(向量); tol 是精度要求; $trace$ 是结果屏幕显示控制, 它们都可缺省; $[T, Y]$ 是积分曲线的点列, 可用作图语句 $\text{plot}(T, Y)$ 作它(或它们)的图形
ode45	高阶法解常微分方程(组)初值问题. 用 4 阶和 5 阶 Runge-Kutta 方法进行积分. 用法和参数说明同 ode23, 精度控制 tol 缺省时为 $tol = 10^{-6}$

§ 1.4 Matlab 程序设计初步

一、M 文件

M 文件有两种形式: 命令文件和函数文件.

- (1) 命令文件包含一系列 Matlab 命令, 当命令窗口调用它时, 自动执行这些命令.
- (2) 函数文件的第一行是以 function 开头的语句, 其中至少有一个输入或输出参数, 然后是一系列 Matlab 命令. 函数文件中定义的变量都是局部变量, 它们只在本函数内有效.

Matlab 语言中最基本最重要的成分是函数, 它的一般形式为

$$\text{function}[a, b, c, \dots] = \text{fun}[d, e, f, \dots]$$

其中: fun 是自定义的函数名, 只要不与库函数名相重, 并且符合字符串的书写规则即可; $d, e, f, \dots$ 是输入变量, 可以是形参, 也可以是实参; $a, b, c, \dots$ 是输出变量. 输入和输出变量可以缺省, 如没有输入变量或没有输出变量.

Matlab 的函数保存为相互分离的 M 文件, 它们等同于其他语言的子程序和函数.