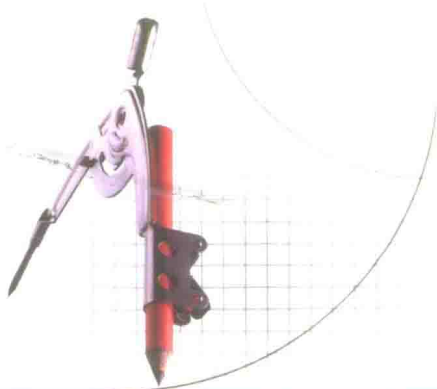




精通

MATLAB 6

- 
- 详述 MATLAB 6 的核心内容
 - 列举大量典型实例, 便于理解
 - 适合理工科的师生及技术人员使用

尹泽明 丁春利 等 编著



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



精通 MATLAB 6

尹泽明 丁春利 等 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

MATLAB 6 采用了全新的操作桌面,直接配备了 MATLAB 环境管理工具,为 MATLAB 代码、变量、数据文件、图形的存取和示例文件、在线帮助的获得提供了快捷的途径。新加入的多种交互式工具也使 MATLAB 图形的绘制和导入导出操作简单易行。同时, MATLAB 6 在数学计算能力、外部数据和代码访问能力、GUI 开发能力方面也有了大幅度的提高。

本书以 MATLAB 6 为编写基础,详细地讲述了 MATLAB 的数据类型、MATLAB 的基本数据组织单位——矩阵的应用、数值计算、数据的可视化、图形句柄的操作、交互式图形用户界面的设计、以及应用程序接口的设计、Notebook 环境的使用等内容。本书将理论讲解和应用举例结合,在更强调应用的数值计算方面,列举了大量的典型实例,有助于读者的学习和应用。

本书内容丰富,可供从事理工科学习和研究的各行各业的工程人员和研究人员使用。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名:精通MATLAB6

作 者:尹泽明 丁春利 等 编著

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编:100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑:闫红梅

印 刷 者:世界知识印刷厂

发 行 者:新华书店总店北京发行所

开 本:787×1092 1/16 印张:26.5 字数:638 千字

版 次:2002 年 6 月第 1 版 2002 年 12 月第 2 次印刷

书 号:ISBN 7-302-05461-4/TP·3218

印 数:5001~8000

定 价:38.00 元

前 言

在科学研究和工程应用中,往往需要进行大量的数学运算,其中包括矩阵运算。这些运算一般来说难以用手工精确和快捷地进行,而要借助计算机编制相应的程序做近似计算。目前流行用 BASIC、FORTRAN 和 C 语言编制计算程序,既需要对有关算法有深刻的了解,又需熟练掌握所用语言的语法和编程技巧。对多数科学工作者而言,同时具备这两方面的才能有一定困难。通常,编制程序是繁杂的,不仅耗费人力和物力,且影响工作进程和效率。

为克服上述困难,美国 MathWorks 公司于 1967 年推出了 Matrix Laboratory (缩写为 MATLAB) 软件包,并不断更新、扩充。目前最新的 6.1 (Release 12.1) 版本是一种功能强、效率高、便于进行科学和工程计算的交互式软件包,包括:一般数值分析、矩阵运算、数字信号处理、建模、系统控制和优化等应用程序,并集应用程序和图形于同一环境,便于使用。在此环境下,所解问题的 MATLAB 语言表述形式和其数学形式相同,不需要按传统的方法编程。

实践证明,用户可以在几十分钟的时间内学会 MATLAB 的基础知识,经过几个小时的使用就能初步掌握,从而能够进行高效率而富有创造性的计算。MATLAB 是高效的科研助手,自推出后即风行美国,流传世界。在 MathWorks Release 12.1 中的新产品及增强功能包括:

- New-Developer's Kit for Texas Instruments DSP 1.1 用来在 TI 处理器上开发、验证 DSP 设计;
- New-Virtual Reality Toolbox 2 在 MATLAB 和 Simulink 中创建和处理三维模型;
- Updated-Image Processing Toolbox 3.1 包括图像注册、去噪声、图像形态学和多维处理等新算法和工具;
- Updated - Real-Time Workshop Embedded Coder 2 支持生成产品级代码。

本书从 MATLAB 的基础知识入手,内容涵盖范围较广。在书中有大量的应用实例,读者只要按照该例子亲自动手演练,就会迅速掌握 MATLAB 的使用。鉴于本书以 MATLAB 6 为蓝本来介绍 MATLAB 的使用,因此对 MATLAB 的核心部分进行了着重的讲解,没有对各种专业应用都进行详细的介绍。

本书由北京华源中科辅龙计算机技术有限公司策划,主要部分由尹泽明、丁春利编写,另外参加编写的有:张明、霍明震、郭美山、张玉玲、邹洁、徐平、石利文、李炎、王艳燕、闫高峰、杨桂莲、李琪、刘小华、魏金兰、田丁、方柯、周毅、于彤、张凯华、靳莹、莫玉敏、董强、李泽远等同志。全书由张玉玲统稿。

由于时间仓促、作者水平有限,本书错漏之处在所难免,欢迎广大读者批评指正。

对本书内容有疑问的读者,可向华源中科辅龙公司读者服务部提出咨询。

咨询电话: 010-62565533 转 3308 E-mail: bookhelp@fulong.com.cn

网址: www.fulong.net

作 者

2002 年 1 月

目 录

第 1 章	MATLAB 简介	1
1.1	MATLAB 产生的历史背景	1
1.1.1	MATLAB 是什么	1
1.1.2	MATLAB 的发展历史	1
1.1.3	MATLAB 的语言特点	2
1.2	MATLAB 与其他数学软件的关系	9
1.2.1	Maple	9
1.2.2	Mathematica	10
1.2.3	Mathcad	10
1.3	MATLAB 集成环境的组成	11
1.4	MATLAB 6 的安装	11
第 2 章	MATLAB 基础知识	14
2.1	MATLAB 的目录结构和组件	14
2.2	MATLAB 的启动	17
2.3	MATLAB 环境	17
2.3.1	MATLAB 的工作空间	17
2.3.2	保存和检索数据	18
2.3.3	数值显示格式	19
2.3.4	关于复数	20
2.3.5	注释和标点	21
2.3.6	MATLAB 的变量	22
2.3.7	MATLAB 的命令窗口 (Command Window)	24
2.3.8	MATLAB 的程序编辑器	27
2.3.9	MATLAB 的变量浏览器	29
2.3.10	MATLAB 的路径浏览器	34
2.3.11	MATLAB 6 的帮助系统	37
2.3.12	文件管理	40
2.3.13	脚本文件	42
2.3.14	Command History 和实录命令 diary	44
第 3 章	MATLAB 的数值计算 (1)	46
3.1	数组	46
3.1.1	简单形式的数组	46

3.1.2	数组编址	47
3.1.3	构建数组	49
3.1.4	数组的方向	50
3.1.5	标量-数组运算	51
3.1.6	数组-数组运算	52
3.1.7	数组操作	56
3.1.8	子数组查找	59
3.1.9	数组大小	60
3.1.10	字符数组	62
3.1.11	元胞数组	69
3.1.12	结构体数组	74
3.1.13	MATLAB 的数据类型	79
3.2	矩阵	80
3.2.1	矩阵定义	80
3.2.2	下标引用	81
3.2.3	矩阵的加减运算	82
3.2.4	向量乘积和转置	83
3.2.5	矩阵乘法	84
3.2.6	特殊矩阵	85
3.2.7	稀疏矩阵	87
3.2.8	线性方程组	90
3.2.9	矩阵的逆和伪逆	92
3.2.10	矩阵的 LU、QR 和 Cholesky 分解	93
3.2.11	矩阵的幂和指数函数	96
3.2.12	特征值分解和奇异值分解	97
3.2.13	矩阵的 Kronecker 乘积	99
第 4 章	MATLAB 的数值计算 (2)	101
4.1	多项式	101
4.1.1	关于多项式的常用函数	101
4.1.2	卷积和解卷积	103
4.1.3	多项式曲线的拟合	104
4.1.4	多项式的展开	104
4.1.5	多项式插值	105
4.2	数值分析	111
4.2.1	函数绘图	111
4.2.2	求极小值	112
4.2.3	求零点	114
4.2.4	积分	115

4.2.5	微分	117
4.2.6	微分方程	119
4.2.7	M 文件举例	121
4.3	数据分析和统计	124
4.3.1	数据分析函数	124
4.3.2	数据预处理	126
4.3.3	曲线拟合	128
4.3.4	傅里叶分析	132
4.4	双重函数	137
4.4.1	函数的文件表示	138
4.4.2	函数的绘制	138
4.5	关系和逻辑运算	140
4.5.1	关系操作符	140
4.5.2	逻辑操作符	141
4.5.3	关系与逻辑函数	142
第 5 章	MATLAB 的数值计算 (3)	144
5.1	控制流	144
5.1.1	For 循环	144
5.1.2	While 循环	146
5.1.3	if-else-end 分支结构	147
5.1.4	switch-case 结构	148
5.1.5	try-catch 结构	149
5.1.6	控制程序流的其他常用命令	150
5.2	M 文件的编写	151
5.2.1	M 文件的规则和属性	151
5.2.2	全局变量和局部变量	157
5.2.3	运算符	157
5.2.4	变量的检测、传递	158
5.2.5	子函数和私有函数	159
5.2.6	串演算函数	161
5.2.7	错误信息和警告信息	163
5.2.8	日期和时间函数	164
5.2.9	用户和程序的交互	165
5.2.10	Shell 函数	166
5.2.11	程序的优化	166
5.3	类和对象	170
5.3.1	类和对象概述	170
5.3.2	运算符重载	174

5.3.3	对象优先级.....	178
5.3.4	类和对象的继承.....	181
5.4	文件.....	184
5.4.1	文件的打开和关闭.....	185
5.4.2	格式文件.....	186
5.5	M 文件的调试.....	188
5.5.1	用调试器调试.....	189
5.5.2	在命令窗口中调试.....	193
5.5.3	文件评述.....	193
第 6 章	数据的可视化 (1).....	195
6.1	图形窗口.....	195
6.2	二维图形.....	198
6.2.1	plot 命令.....	198
6.2.2	线型和标记.....	201
6.2.3	加格线和标注.....	202
6.2.4	加图例.....	204
6.2.5	定制图形坐标轴.....	204
6.2.6	图形的保持.....	205
6.2.7	多图形窗口.....	206
6.2.8	屏幕的刷新.....	208
6.2.9	缩放命令.....	209
6.2.10	其他的二维图.....	210
6.3	三维图形.....	210
6.3.1	函数 plot3.....	211
6.3.2	改变视角.....	213
6.3.3	含有两个变量的标量函数.....	215
6.3.4	杂乱或散射数据的插值.....	217
6.3.5	网格图和曲面图.....	218
6.3.6	等高线图.....	223
6.3.7	三维数据的二维图.....	225
6.3.8	其他函数.....	226
6.3.9	动画.....	228
6.4	颜色的使用.....	231
6.4.1	颜色映像原理.....	231
6.4.2	颜色映像使用.....	233
6.4.3	颜色映像显示.....	234
6.4.4	颜色映像的建立和修改.....	236
6.4.5	照明和材质处理.....	238

6.4.6	图形中使用一个以上的颜色映像	240
6.4.7	图像	241
6.4.8	用颜色描述第四维	243
第 7 章	数据的可视化 (2)	245
7.1	特殊图形	245
7.1.1	面积图和直方图	245
7.1.2	统计频率直方图	249
7.1.3	饼状图	250
7.1.4	火柴杆图	251
7.1.5	阶梯图	253
7.1.6	矢量场图	253
7.1.7	交互绘图函数 ginput	255
7.2	句柄图形	256
7.2.1	句柄对象	257
7.2.2	通用函数 get、set	263
7.2.3	用鼠标句柄来选择对象	267
7.2.4	位置和单位	268
7.2.5	默认属性	269
7.2.6	图形输出对象的设置	270
7.3	图像对象	273
7.3.1	图像的类型	274
7.3.2	单字节图像	275
7.3.3	图像对象的属性	277
7.4	块对象	280
7.4.1	基本块操作	280
7.4.2	多面块	282
7.5	轴对象	284
7.5.1	轴位框的几何属性和多轴位框	285
7.5.2	轴刻度的属性控制	285
7.5.3	坐标轴尺度、方向、位置属性	286
7.6	图对象	288
7.6.1	图对象的定位	288
7.6.2	图对象的颜色控制	289
第 8 章	MATLAB Notebook	292
8.1	Notebook 的安装	292
8.2	Notebook 的启动	293
8.2.1	从 Word 中启动 Notebook	293
8.2.2	从 MATLAB 中启动 Notebook	294

8.2.3 Notebook 菜单简介	296
8.3 M-book 的使用	297
8.3.1 元胞的定义和使用	298
8.3.2 计算区、循环运行	301
8.3.3 输出控制	302
8.3.4 元胞的形式	306
8.3.5 M-book 模板的使用总结	306
第 9 章 MATLAB 的符号计算	307
9.1 符号计算入门	307
9.1.1 符号对象和符号表达式	307
9.1.2 符号对象的操作和转换 (1)	311
9.1.3 符号对象的操作和转换 (2)	314
9.1.4 生成符号函数	317
9.1.5 实变量、复变量和抽象函数	317
9.2 微积分	318
9.2.1 微分	318
9.2.2 积分	320
9.2.3 求极限	322
9.2.4 级数求和	324
9.3 解方程	325
9.3.1 代数方程求解	325
9.3.2 微分方程求解	328
9.4 特殊数学函数	331
9.4.1 符号矩阵	331
9.4.2 代数运算	333
9.4.3 线性代数运算	334
9.4.4 其他特性	335
第 10 章 图形用户界面	339
10.1 简单用户图形示例	339
10.2 GUI 对象层次结构	341
10.3 菜单	342
10.4 中断回调的规则	349
10.5 用户自制 GUI M 文件	350
10.6 用户界面设计原则	353
10.7 界面菜单	353
10.8 上下文菜单的制作	358
10.9 用户控件的制作	359

第 11 章 MATLAB 与应用程序的接口	366
11.1 MATLAB 与 C 语言的关系	367
11.2 MATLAB 编译器	375
11.3 创建独立的应用程序	383
11.4 Matcom 的简单应用示例	391
11.5 VC++调用 MATLAB Engine	392
附 录	394
MATLAB 的标点及符号	394
MATLAB 的函数及命令 (Functions and Commands)	395
SIMULINK 的库模块	405
图形对象属性	406
参考文献	409

第 1 章 MATLAB 简介

1.1 MATLAB 产生的历史背景

1.1.1 MATLAB 是什么

MATLAB 是一种用于科学工程计算的高效率高级语言。MATLAB 原先作为矩阵实验室 (Matrix Laboratory), 是提供使用 LINPACK 和 EISPACK 矩阵软件包接口的。后来它逐渐地发展成通用科学计算、图视交互系统和程序设计语言。

与 BASIC、FORTRAN、C/C++ 等语言相比, MATLAB 的语法规则简单, 更贴近人的思维方式。用 MATLAB 编写程序, 犹如在一张演算纸上排列公式和求解问题一样效率高, 因此被称为“演算纸式的”科学工程算法语言。MATLAB 的基本数据单位是矩阵, 它的命令表达与数学、工程中常用的习惯表达形式十分相似。比如, 矩阵方程 $b=Ax$, 在 MATLAB 中被写成 $b=A*x$ 。而若要通过 A 、 b 求 x , 那么只要写出命令 $x=A\b$ 即可, 而完全不需要对矩阵的乘法和求逆进行编程。由此可见, 用 MATLAB 解计算问题要比用 BASIC、FORTRAN、C/C++ 等语言方便简洁得多。

随着 MATLAB 版本的不断更新, 其功能越来越强, 使它在诸如一般数值计算、数字信号处理、系统识别、自动控制、振动理论、时序分析与建模、优化设计、神经网络控制、化学统计学、动态仿真系统、特殊函数和图形领域表现出一般高级语言难以比拟的优势, 并可以方便地用于几乎所有的科学和工程计算的各个方面。可以说, MATLAB 不仅是一种编程语言, 而且在广义上是一种语言开发系统。

经过 MathWorks 公司的不断完善, 时至今日, MATLAB 已经发展成为适合多学科、多种工作平台的功能强劲的大型软件。

1.1.2 MATLAB 的发展历史

在科学研究和工程应用中, 经常需要进行大量的数学计算, 其中包括矩阵运算。这些运算一般来说难以用手工精确和快捷地进行, 需要编制相应的程序, 并借助于计算机的强大的计算能力来做计算。利用 BASIC、FORTRAN 和 C/C++ 语言编制计算程序, 不但要对相关的算法有深刻的理解和认识, 而且还要熟练地掌握所使用语言的语法和编程技巧。对于多数科学工作者来说, 同时在这两方面具备很强的能力有一定的困难。面对这样的形势, 一种具有优秀的数值计算能力的软件——MATLAB 诞生了。

20 世纪 70 年代中期, Cleve Moler 博士 (现 MathWorks 公司的首席科学家) 和他的同事在美国国家科学基金的资助下研究开发了调用 LINPACK 和 EISPACK 的 FORTRAN 子程

序库。LINPACK 是解线性方程的 FORTRAN 程序库, EISPACK 是解特征方程的 FORTRAN 程序库。这两个程序库代表着当时矩阵计算软件的水平。

到了 20 世纪 70 年代后期, 身为 New Mexico 大学计算机系主任的 Cleve Moler, 在给 学生讲授线性代数课程时, 想教学生使用 LINPACK 和 EISPACK 程序库。当他发现学生花了大量的时间在用 FORTRAN 编写接口程序而背离了他的开课目的时, 他觉得有必要编写一套接口程序供学生使用, 于是他自己亲自动手, 利用业余时间编写方便使用 LINPACK 和 EISPACK 的接口程序。Cleve Moler 给这个接口程序取名为 MATLAB (矩阵实验室), 该名为 matrix 和 laboratory 各取前三个字母组成。

有一次 Cleve Moler 去另一所大学讲学。在访问结束后, 他把 MATLAB 留在了那所大学的计算机上, 在这以后的一两年时间里, MATLAB 开始受到广泛欢迎, 并成了应用数学界的术语。在以后很多年里, MATLAB 作为教学辅助软件在很多大学里使用, 并作为面向大众的免费软件广为流传。

1983 年早春, Cleve Moler 到 Stanford 大学访问, 作为工程师的 John Little (MathWorks 公司总裁) 受到了 MATLAB 的深深吸引, 他敏锐地察觉到 MATLAB 在工程领域潜在的应用天地。同年, 他和 Cleve Moler、Steve Bangert 一起用 C 语言合作开发了第二代专业版 MATLAB。从这一代起, MATLAB 的核心开始采用 C 语言编写, 并且在具有数值计算能力的基础上, 具有了数据视图的功能。

1984 年, Cleve Moler 和 John Little 成立了 MathWorks 公司, 正式把 MATLAB 推向市场, 并且继续进行 MATLAB 的研究和开发工作。

1992 年, MathWorks 公司推出了划时代意义的 MATLAB 4.0 版本, 并于 1993 年推出了其微机版, 可以配合 Microsoft Windows 一起使用, 使之应用范围越来越广。1997 年推出了 Windows 95 下的 MATLAB 5.0 和 SIMULINK 2.0, 该版本在继承 MATLAB 4.0 和 SIMULINK 1.3 版本功能的基础上, 实现了真正的 32 位运算, 数值计算更快, 图形表现更丰富有效, 编程更简洁直观, 用户界面更友好; 其帮助消息采用超文本格式和 PDF 格式, 在 Netscape 3.0 或 IE 3.0 及以上版本、Acrobat Reader 中可以方便地浏览。

1.1.3 MATLAB 的语言特点

MATLAB 之所以能如此迅速地普及, 显示出强大的生命力, 是由于它有着不同于其他语言的特点。被称做第四代计算机语言的 MATLAB, 利用其丰富的函数资源, 使编程人员从烦琐的程序代码中解放出来。MATLAB 最突出的特点就是简洁。MATLAB 用更直观的、更符合人们思维习惯的代码, 代替了 C/C++ 和 FORTRAN 语言的冗长代码。MATLAB 给用户提供的是最直观、最简洁的开发环境。

1. 高效方便的矩阵和数组运算

MATLAB 语言像 BASIC、FORTRAN 和 C 语言一样规定了矩阵的算术运算符、关系运算符、逻辑运算符、条件运算符以及赋值运算符, 而且这些运算符大部分可以照搬到数组的运算, 有些如算术运算符只要增加 “.” 就可以用于数组间的运算, 并且它不需要定义数

组间的维数，并给出矩阵函数、特殊矩阵专门的库函数，使之在数字信号处理、建模、系统识别、自动控制、优化等领域的问题时，显得十分简洁、高效，具有其他高级语言不可比拟的优势。

例 1: 已知矩阵

```
A=
    2    5   -8    7
    3    8   13    2
   -6   11   -2    0

B=
   22    7
   -3   16
   -9    0
    1    8
```

求 A、B 矩阵的积 C。

C 语言程序如下：

```
main()
{
int  a[3][4]={2,5,-8,7},{3,8,13,2},{-6,11,-2,0}};
int  b[4][2]={22,7},{-3,16},{-9,0},{1,8}};
int  c[3][2],i,j,k;

for(i=0;i<3;i++){
    for(j=0;j<2;j++){
        c[i][j]=0;
        for(k=0;k<4;k++){
            c[i][j]+=a[i][k]*b[k][j];
        }
    }
    for(i=0;i<3;i++){
        for(j=0;j<2;j++){
            printf("c[%d][%d]=%d,",i,j,c[i][j]);
            printf("\n");
        }
    }
}
```

在 Turbo 2.0 下调试结果为：

```
c[0][0]=108,c[0][1]=150,
c[1][0]=-73,c[1][1]=165,
c[2][0]=-147,c[2][1]=134,
```

FORTTRAN 程序如下：

```
dimension a(3,4),b(4,2),c(3,2)
data a/2.,3.,-6.,5.,8.,11.,-8.,13.,-2.,7.,2.,0./
data b/22.,-3.,-9.,1.,7.,16.,0.,8./
do 10 i=1,3
```

```

do 10 j=1,2
c(i,j)=0.
do 10 k=1,4
c(i,j)=c(i,j)+a(i,k)*b(k,j)
write(*,'(a)') 'a'
do 20 i=1,3
write(*,*)(a(i,j),j)=1,4)
write(*,'(a)') 'b'
do 30 i=1,4
write(*,*)(b(i,j),j)=1,2)
write(*,'(a)') 'c'
do 40 i=1,3
write(*,*)(c(i,j),j)=1,2)
stop
end

```

BASIC 程序如下:

```

DIM a(3,4),b(4,2),c(3,2)
FOR i=1 TO 3
FOR j=1 TO 4
READ      a(i,j)
NEXT      j
NEXT      i
FOR i=1 TO 4
FOR j=1 TO 2
READ      b(i,j)
NEXT      j
NEXT      i
DATA      2.,5.,-8.,7.,3.,8.,13.,2.,-6.,11.,-2.,0.
FOR i=1 TO 3
FOR j=1 TO 2
c(i,j)=0
FOR k=1 TO 4
c(i,j)= c(i,j)+a(i,k)*b(k,j)
NEXT      k
NEXT      j
NEXT      i
PRINT     "a"
FOR i=1 TO 3
PRINT     a(i,1),a(i,2),a(i,3),a(i,4)
NEXT      i
PRINT     "b"
FOR i=1 TO 4
PRINT     b(i,1),b(i,2)
NEXT      i
PRINT     "c"
FOR i=1 TO 3
PRINT     c(i,1),c(i,2)
NEXT      i

```

RETURN

而下面的 MATLAB 程序要简洁得多。

```
>> A=[2 5 -8 7
      3 8 13 2
      -6 11 -2 0];
>> B=[22 7
      -3 16
      -9 0
      1 8];
>> C=A*B;
>> C
```

按回车键，得

```
C =
    108    150
    -73    165
   -147    134
```

程序中第一行输入矩阵 A，第二行输入矩阵 B，前两行为输入部分；第三行求 A、B 之积，“*”表示矩阵相乘的运算符，“>>”表示输入命令提示符。矩阵的乘积，赋给矩阵 C，在没有指定变量时，赋给 ans（ans 为系统指定的默认变量）。

读者已经通过自身在 BASIC、FORTRAN、C/C++ 方面的编程经验和上面所举的例子中看到，用 BASIC、FORTRAN、C/C++ 及类似的语言去编写程序，尤其在涉及到矩阵运算或是画图时，会很麻烦。例如，用户要求解一个线性方程，首先要编写一个读入数据的子程序，然后选用一种求解线性方程的算法（例如逼近法）编写一个子程序来求解方程，最后再编写输出数据的子程序。编写求解线性方程的部分会很麻烦，麻烦主要在于对矩阵的元素作循环、选择稳定的算法以及代码的调试都不容易。再如用户要用 QR 法求解矩阵的特征值，如果用 FORTRAN 语言编写，至少要三、四百行，编写及调试这种程序都是一项费时和易出错的工作，而这一步可能仅是一项系统工程中的微乎其微的一小步。而用 MATLAB 求解这些问题时，就显得轻松多了。以下为用 MATLAB 编写以上两个问题的程序方法。

用 MATLAB 求解下列方程，并求矩阵 A 的特征值。A x = b，其中：

```
A=
  3   -2    0   -4    9
 -7    5   -6    2   -3
  8   -1    7   -5    0
  4   -9   12    2  -11
 17   -8    6    0    1
b=[1 2 3 4 5]'
```

解为：x=A\b；设 A 的特征值组成的向量为 e，e=eig(A)；在 MATLAB 中解的过程如下：

```
>> A=
  3   -2    0   -4    9
 -7    5   -6    2   -3
```

```
      8   -1    7   -5    0
      4   -9   12    2  -11
     17   -8    6    0    1
>> b=
      1
      2
      3
      4
      5
>> x=A\b;
>> e=eig(A);
>> x
x =
    0.3247
   -2.1787
   -2.6617
   -3.3711
   -1.9795
>> e
e =
   19.1033
   -6.2840
   -1.4104
    3.2955+8.1895i
    3.2955-8.1895i
```

从以上程序可以看出, MATLAB 的程序极其简短。更重要的一点是, MATLAB 具有很严格的解题规范, 以上面的题为例, 它会根据矩阵的特征选择适合方程的求解算法, 所以不用担心 MATLAB 解题的准确性。

在 MATLAB 环境下, 许多复杂的数学运算, 如求矩阵的行列式值、求矩阵的逆及求特征值、求函数的微分、求函数的积分、进行多项式插值、解微分方程等, 都有现成的库函数可以调用, 并且各个库函数是根据不同的应用情况采用不同的优化算法的, 保证了结果的可靠性和求解的快速性。

2. 编程效率高

MATLAB 语言简洁紧凑, 使用灵活方便, 程序书写形式自由。库函数非常丰富, 避免了繁杂的子程序编程任务, 删除了一切不必要的工作。由于库函数都是由本领域的专家编写, 在可靠性和算法的高效性上有很好的保证。可以说, 使用 MATLAB 进行科技开发是在吸取专家经验的基础上进行的。

3. 结构化/面向对象

新的 MATLAB 版本采用 C++ 编写, 既具有结构化的控制语句 (如 for 循环、while 循环、break 和 continue 语句、if 语句), 又具有面向对象编程的特性, 并且在制作图形用户界面 (GUI) 方面增加了许多新的功能。