

21世纪高等院校教材

数学实验

张智丰 编著



科学出版社
www.sciencep.com

内 容 要 点

数 学 实 验

张智丰 编著

科 学 出 版 社

北 京

内 容 简 介

本书主要介绍利用 MATLAB 软件和 Mathematica 软件解决一系列数学问题的计算方法,全书共 39 个实验.将一些常用的软件命令分散在各个实验中介绍,使得每个实验难度适中,易于学生接受;为了让学有余力的学生能够有进一步学习的内容,在每个实验的习题中安排了选做习题.这类题目一般采用给出一个软件的函数名,让学生通过自学的方法掌握该函数的方法,通过一系列的自学,逐步掌握软件的使用方法.

本书可作为理工经管类本科生数学实验课程的教材,也可供相关研究人员和学生参考使用.

图书在版编目(CIP)数据

数学实验/张智丰编著. —北京:科学出版社,2008
ISBN 978-7-03-022696-9

I. 数… II. 张… III. 计算机辅助计算-软件包-MATLAB, Mathematica IV. TP391.75

中国版本图书馆 CIP 数据核字(2008)第 119108 号

责任编辑:李晓鹏 盛 杉 / 责任校对:陈玉凤
责任印制:张克忠 / 封面设计:耕者设计工作室

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>

明辉印刷有限公司 印刷

科学出版社发行 各地新华书店经销

*

2008 年 9 月第 一 版 开本:B5(720×1000)

2008 年 9 月第一次印刷 印张:11

印数:1—3 500 字数:208 000

定价:20.00 元

(如有印装质量问题,我社负责调换〈明辉〉)

前 言

随着数学实验课程在各类院校的开设,数学实验的教材也越来越多,数学实验可作为学习数学的一种方法和手段,也可作为利用数学软件将数学应用于工程实际的一种工具,这些观点正在逐步得到科学界的认同.

目前的数学实验书籍中,有很大一部分讨论的问题比较深入,常常是一个实验需要十几页甚至几十页的篇幅,虽然问题讨论得非常详尽,但是不适合低年级本科生学习.这更像是对数学模型的讨论.笔者根据近几年开设本科生数学实验课程的经验,认为在当今大众化教育的形势下,对于普通高校的学生,更实际的做法是通过简单的小实验来学习利用数学软件解决数学问题的方法,从而掌握解决工程问题的能力.

本书主要介绍 MATLAB 和 Mathematica 两个软件的应用,以介绍 MATLAB 为主.全书分为 39 个实验,每个实验计划 2 个课时,部分实验也可以由学生课外完成,全书的计划总课时为 64~72 课时.

本书适合学过微积分和线性代数的学生使用,部分内容需要概率统计的知识.书中部分内容是对高年级课程的简单介绍,在实验的编写中已经对所需知识作了必要的介绍,这样做的目的,一方面是丰富数学实验的内容,另一方面让学生了解数学在各个学科中的应用.同时,这种介绍方式也为学生参加数学建模竞赛提供了方便,不少学生在此基础上修读数学建模课程会有很大受益.

本书在编写过程中得到许多同仁的帮助,在此表示衷心的感谢!

由于编者水平所限,书中所涉及的学科领域又比较多,如有疏漏之处,敬请读者批评指正.

张智丰

2008 年 5 月 16 日

于杭州电子科技大学下沙校区

目 录

绪论	1
0.1 什么是数学实验	1
0.2 怎样做好数学实验	1
0.3 MATLAB 简介	1
0.4 Mathematica 简介	11
0.5 本书的使用	15
实验 1 矩阵的基本运算(一)	16
实验 2 矩阵的基本运算(二)	19
实验 3 MATLAB 中的极限和微分运算	21
实验 4 MATLAB 中的各种积分运算	24
实验 5 MATLAB 的图形功能	27
实验 6 MATLAB 的程序结构	32
实验 7 分形初探	36
实验 8 数字图像处理初探	40
实验 9 数字图像的边界提取	44
实验 10 图像压缩的 MATLAB 实现	48
实验 11 Bézier 曲线的绘制	51
实验 12 实验数据的插值	56
实验 13 实验数据的拟合	61
实验 14 MATLAB 在级数中的应用	64
实验 15 MATLAB 与傅里叶级数	67
实验 16 方程和方程组的求解	70
实验 17 微分方程和微分方程组的求解	74
实验 18 MATLAB 在复变函数中的应用	78
实验 19 MATLAB 中的概率统计函数	82
实验 20 方差分析的 MATLAB 实现	86
实验 21 回归分析的 MATLAB 实现(一)	90
实验 22 回归分析的 MATLAB 实现(二)	95
实验 23 统计量计算的 MATLAB 实现	99
实验 24 主成分分析的 MATLAB 编程实现	102

实验 25	主成分分析的 MATLAB 实现	107
实验 26	聚类分析的 MATLAB 编程实现	111
实验 27	聚类分析的 MATLAB 实现	117
实验 28	参数估计的 MATLAB 实现(一)	120
实验 29	参数估计的 MATLAB 实现(二)	125
实验 30	用 MATLAB 求解线性规划问题	129
实验 31	用 MATLAB 求解非线性优化问题	133
实验 32	有约束非线性多元变量的优化	137
实验 33	初识 Mathematica	141
实验 34	Mathematica 中的极限与微积分	145
实验 35	Mathematica 中的级数运算	149
实验 36	Mathematica 中的 2 维作图功能	151
实验 37	Mathematica 中的 3 维作图功能	156
实验 38	Mathematica 程序设计	159
实验 39	Mathematica 的输入输出	164

绪 论

0.1 什么是数学实验

所谓数学实验,就是利用计算机系统作为实验工具,以数学理论作为实验原理,以数学素材作为实验对象,以简单的对话方式或复杂的程序方式作为实验形式,以数值计算、符号演算或图形显示等作为实验内容,以实例分析、模拟仿真、归纳总结等为主要实验方法,以辅助学数学、用数学或做数学为实验目的,以实验报告为最终形式的上机实践活动。

0.2 怎样做好数学实验

由上面讨论的数学实验的内容和形式可知,要做好数学实验,需要对实验内容的数学背景有清楚的理解,对实验所使用的工具有详细的了解。只要勤于动手,多实践,多练习,就能够轻松地完成数学实验。

这里所谓的工具,是指计算机软件,本书主要使用 MATLAB。所谓有详细的了解,是指对具体的 MATLAB 函数有详细的了解,清楚函数中参数的含义。本书采用先将 MATLAB 软件作整体的介绍,然后在每一个具体的实验中,介绍本实验涉及的具体函数。因此,完成单个实验是非常简单的,但更重要的是,学生要通过逐个完成实验,逐步记忆函数的使用方法这样一个循序渐进的过程来学习和掌握 MATLAB 的使用方法,从而学会利用软件来求解数学问题的方法。

0.3 MATLAB 简介

MATLAB 是一种功能非常强大的科学计算软件。本书将利用这个软件作为实验平台。因此,在正式使用之前,对软件作一个介绍,以便使用者对软件有一个整体的认识。

一、MATLAB 的概况

MATLAB 一词源于 Matrix Laboratory,原意为矩阵实验室,该软件经过三十多年的发展,目前除具备卓越的数值计算能力外,还提供了专业水平的符号计算、

文字处理,可视化建模仿真和实时控制等功能。

MATLAB 的基本数据单位是矩阵,它的指令表达式与数学、工程中常用的形式十分相似,故用 MATLAB 来解算问题要比用 C、FORTRAN 等语言完成相同的任务简捷得多。

当前流行的 MATLAB 7.0/Simulink 7 拥有数百个内部函数的主程序和六十多种工具箱(toolbox)。工具箱又可以分为功能工具箱和学科工具箱。功能工具箱用来扩充 MATLAB 的符号计算,可视化建模仿真、文字处理及实时控制等功能。学科工具箱是专业性比较强的工具箱,控制工具箱、信号处理工具箱、通信工具箱、图像处理工具箱等都属于此类。

代码的开放性使 MATLAB 广受用户欢迎。除内部函数外,所有 MATLAB 主程序文件和各种工具箱都是可读可修改的文件,用户通过对源程序的修改或加入自编程序可以构造新的专用工具箱。

二、MATLAB 产生的历史背景

在 20 世纪 70 年代中期,Cleve Moler 博士和其同事在美国国家科学基金的资助下开发了调用 EISPACK 和 LINPACK 的 FORTRAN 子程序库。EISPACK 是特征值求解的 FORTRAN 程序库,LINPACK 是解线性方程的程序库。在当时,这两个程序库代表矩阵运算的最高水平。

到 70 年代后期,身为美国 New Mexico 大学计算机系系主任的 Cleve Moler,在给 学生讲授线性代数课程时,想教学生使用 EISPACK 和 LINPACK 程序库,但他发现学生用 FORTRAN 编写接口程序很费时间,于是他开始自己动手,利用业余时间为学生编写 EISPACK 和 LINPACK 的接口程序。Cleve Moler 给这个接口程序取名为 MATLAB,该名为矩阵(matrix)和实验室(laboratory)两个英文单词的前三个字母的组合。在以后的数年里,MATLAB 在 多所大学里作为教学辅助软件使用,并作为面向大众的免费软件广为流传。

1983 年春天,Cleve Moler 到 Standford 大学讲学,MATLAB 深深地吸引了工程师 John Little。John Little 敏锐地觉察到 MATLAB 在工程领域的广阔前景。同年,他和 Cleve Moler、Steve Bangert 一起,用 C 语言开发了第二代专业版。这一代的 MATLAB 语言同时具备了数值计算和数据图示化的功能。

1984 年,Cleve Moler 和 John Little 成立了 Math Works 公司,正式把 MATLAB 推向市场,并继续进行 MATLAB 的研究和开发。

在当今的数学类科技应用软件中,就软件数学处理的原始内核而言,可分为两大类:一类是数值计算型软件,如 MATLAB、Xmath、Gauss 等,这类软件擅长于数值计算,对处理大批数据效率高;另一类是数学分析型软件,Mathematica、Maple 等,这类软件以符号计算见长,能给出解析解和任意精确解,其缺点是处理大量数

据时效率较低. MathWorks 公司顺应多功能需求之潮流,在其卓越数值计算和图形显示功能的基础上,又率先在专业水平上开拓了其符号计算、文字处理、可视化建模和实时控制功能,开发了适合多学科、多部门要求的新一代科技应用软件 MATLAB. 经过多年的国际竞争, MATLAB 已经占据了数学与工程计算软件市场的主导地位.

在 MATLAB 进入市场前,国际上的许多软件包都是直接以 FORTRAN、C 等编程语言开发的. 这种软件的缺点是使用面窄、接口简陋、程序结构不开放以及没有标准的基本函数,很难适应各学科的最新发展,因而很难推广. MATLAB 为各国科学家开发学科软件提供了新的基础. 在 MATLAB 问世不久的 80 年代中期,原先控制领域里的一些软件包纷纷被淘汰或在 MATLAB 上重建.

MathWorks 公司 1993 年推出了 MATLAB 4.0 版,1995 年推出 4.2C 版(for win3. X),1997 年推出 5.0 版,1999 年推出 5.3 版. MATLAB 5. X 较 MATLAB 4. X 无论是在界面还是内容上都有长足的进展,其帮助信息采用超文本格式和 PDF 格式,分别在 Netscape 3.0 或 IE 4.0 及以上版本和 Acrobat Reader 中可以方便地浏览.

时至今日,经过 MathWorks 公司的不断完善, MATLAB 已经发展成为适合多学科、多种工作平台的功能强大的大型软件. 在国外, MATLAB 已经经受了多年考验. 在欧美高校, MATLAB 已经成为线性代数、自动控制理论、数理统计、数字信号处理、时间序列分析和动态系统仿真等高级课程的基本教学工具,也成为攻读学位的大学生、硕士生和博士生必须掌握的基本技能. 在设计研究单位和工业部门, MATLAB 被广泛用于科学研究和解决各种具体问题. 在国内,特别是工程界, MATLAB 已经盛行起来. 可以说,无论从事工程方面的哪个学科,都能在 MATLAB 里找到合适的功能.

三、MATLAB 的语言特点

一种语言之所以能如此迅速地普及,显示出如此旺盛的生命力,是由于它有着不同于其他语言的特点,正如同 FORTRAN 和 C 等高级语言使人们摆脱了需要对计算机硬件资源进行操作一样,被称作第四代计算机语言的 MATLAB,利用其丰富的函数资源,使编程人员从繁琐的程序代码中解放出来. MATLAB 最突出的特点就是简捷. MATLAB 用更直观的,符合人们思维习惯的代码,代替了 C 和 FORTRAN 语言的冗长代码. MATLAB 给用户带来的是最直观、最简捷的程序开发环境. 以下简单介绍一下 MATLAB 的主要特点.

(1) 语言简洁紧凑,使用方便灵活,库函数极其丰富. MATLAB 程序书写形式自由,利用丰富的库函数避开繁杂的子程序编程任务,压缩了一切不必要的编程工作. 由于库函数都由本领域的专家编写,用户不必担心函数的可靠性.

具有 FORTRAN 和 C 等高级语言知识的读者可能已经注意到,如果用 FORTRAN 或 C 语言去编写程序,尤其当涉及矩阵运算和画图时,编程会很麻烦.例如,如果用户想求解一个线性代数方程,就得编写一个程序块读入数据,然后再使用一种求解线性方程的算法(如高斯消去法)编写一个程序块来求解方程,最后再输出计算结果.在求解过程中,最麻烦的要算第二部分.解线性方程的麻烦在于要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.即使有部分源代码,用户也会感到麻烦,且不能保证运算的稳定性.解线性方程的程序用 FORTRAN 和 C 这样的高级语言编写,至少需要四百多行,调试这种几百行的计算程序可以说很困难.以下用 MATLAB 编写一个解线性方程组和求矩阵特征值的程序,来看一看其简捷性.

例 利用 MATLAB 求解下列方程,并求矩阵 A 的特征值:

$$Ax = b, \text{ 其中, } A = \begin{pmatrix} 32 & 13 & 45 & 67 \\ 23 & 79 & 85 & 12 \\ 43 & 23 & 54 & 65 \\ 98 & 34 & 71 & 35 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}.$$

解 在 MATLAB 命令窗口中输入

```
>> A = [32,13,45,67; 23,79,85,12; 43,23,54,65; 98,34,71,35];
      b = [1; 2; 3; 4];
      x = A\b
      e = eig(A)
```

此时,系统返回

```
x =      193.4475
      56.6905
     -48.1919
      -1.9461

e =
    0.1809
    0.5182
   -0.5333
    0.1862
```

这里, $x=A \setminus b$ 表示 A 的逆矩阵乘 b , $\text{eig}(A)$ 是求 A 的特征值的函数.

可见, MATLAB 的程序极其简短.更为难能可贵的是, MATLAB 甚至具有一定的智能水平,如上面的解方程, MATLAB 会根据矩阵的特性选择方程的求解方法,所以用户根本不用怀疑 MATLAB 的准确性.

(2) 运算符丰富.由于 MATLAB 是用 C 语言编写的, MATLAB 提供了和 C 语言几乎一样多的运算符,灵活使用 MATLAB 的运算符将使程序变得极为简短.

(3) MATLAB 既具有结构化的控制语句(如 for 循环、while 循环、break 语句和 if 语句),又有面向对象编程的特性.

(4) 程序限制不严格,程序设计自由度大.例如,在 MATLAB 里,用户无需对

矩阵预定义就可使用。

(5) 程序的可移植性很好,基本上不作修改就可以在各种型号的计算机和操作系统上运行。

(6) MATLAB 的图形功能强大.在 FORTRAN 和 C 语言里,绘图都很不容易,但在 MATLAB 里,数据的可视化非常简单. MATLAB 还具有较强的编辑图形界面的能力。

(7) MATLAB 的缺点是,它和其他高级程序相比,程序的执行速度较慢.由于 MATLAB 的程序不用编译等预处理,也不生成可执行文件,程序为解释执行,所以速度较慢。

(8) 功能强大的工具箱是 MATLAB 的另一特色.如前所述,目前 MATLAB 包含两个部分:核心部分和各种可选的工具箱.核心部分中有数百个核心内部函数.而工具箱又分为两类:功能性工具箱和学科性工具箱.其中的功能性工具箱可用于多种学科,而学科性工具箱是专业性比较强的,如 control toolbox、signal processing toolbox、communication toolbox 等.这些工具箱都是由该领域内学术水平很高的专家编写的,所以用户无需编写自己学科范围内的基础程序,就可以直接进行高、精、尖的研究。

(9) 源程序的开放性.开放性也许是 MATLAB 最受人们欢迎的特点.除内部函数以外,所有 MATLAB 的核心文件和工具箱文件都是可读可改的源文件,用户可通过对源文件的修改以及加入自己的文件构成新的工具箱。

四、MATLAB 的一般操作

MATLAB 的安装和启动与大多数基于 Windows 操作系统的应用软件一样,这里不再叙述,系统启动后的界面如图 0.1 所示。

主界面上有 5 个窗口:主窗口、命令窗口、当前目录窗口、工作空间窗口和命令历史窗口,主要使用主窗口和命令窗口,其他窗口为辅助性窗口.命令窗口是和系统交互的场所。

在 MATLAB 软件中进行基本数学运算,只需将需要运算的表达式直接输入提示符(>>)之后,并按 Enter 键即可.例如,要计算 $(5 \times 2 + 1.3 - 0.8) \times 10 \div 25$,只要做如下的操作:

```
>>(5 * 2 + 1.3 - 0.8) * 10 / 25
```

按 Enter 键,即得

```
ans = 4.2000
```

MATLAB 会将运算结果直接存入一变量 ans,代表 MATLAB 运算后的答案(answer)并显示其数值于屏幕上。

也可将运算式的结果存放在一个指定的变量 x 中,

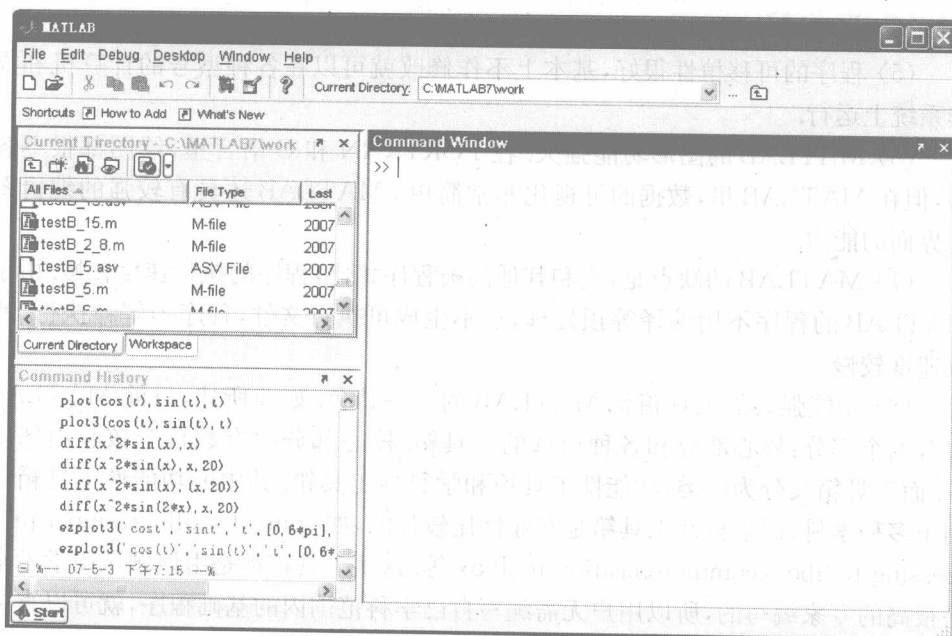


图 0.1 MATLAB7.0 系统主界面

```
>>x = (5 * 2 + 1.3 - 0.8) * 10^-2/25
```

按 Enter 键, 即得

```
x = 42
```

此时 MATLAB 会直接显示 x 的值. 由上例可知, MATLAB 认识所有一般常用的加(+)、减(-)、乘(*)、除(/)、幂次运算(^)等的数学运算符.

MATLAB 将所有变量均存成 double 的形式, 所以不需经过变量说明(variable declaration), MATLAB 同时也会自动进行记忆体的使用和回收, 而不必像 C 语言, 必须由使用者一一指定并一一释放. 这些功能使得 MATLAB 易学易用, 使用者可专心致力于撰写程序, 而不会被软件的枝节问题所干扰.

由于 MATLAB 的基本数据单位是矩阵, 因此, 在 MATLAB 中, 变量可以用来存放向量或矩阵, 并进行各种运算, 如下例的行向量(row vector)运算:

```
>> x = [1 3 5 2];
```

```
>> y = 2 * x + 1
```

```
y = 3 7 11 5
```

若不想让 MATLAB 每次都显示运算结果, 只需在运算式最后加上分号(;)即可, 如下例:

```
>> y = sin(10) * exp(-0.3 * 4^2);
```

系统将不显示任何结果, 此时, 若要显示变量 y 的值, 直接键入 y 即可

```
>> y
      y = -0.0045
```

在上例中, $\sin(x)$ 是正弦函数, $\exp(x)$ 是指数函数, 这些都是 MATLAB 经常用到的数学函数。

在 MATLAB 中, 变量命名须遵循如下的规则:

- (1) 第一个字符必须是英文字母;
- (2) 字符间不可留空格;
- (3) 最多只能有 19 个字符, MATLAB 会忽略多余字符。

系统根据需要允许更改、增加或删除向量的元素, 具体的操作方法为

```
>> y(3) = 2      % 更改向量 y 第 3 个元素
      y = 3 7 2 5
```

```
>> y(6) = 10    % 加入向量 y 的第 6 个元素
```

此时, 由于原来向量 y 只有 4 个元素, 因此, 系统自动将 y 的第 5 个元素赋值为 0。

```
      y = 3 7 2 5 0 10
>> y(4) = []    % 删除 y 的第 4 个元素
      y = 3 7 2 0 10
```

在上例中, MATLAB 会忽略所有在百分号 (%) 之后的文字, 因此“%”之后的文字均视为程序的注解 (comments)。

MATLAB 亦可取出向量的一个元素或一部分来作运算。

```
>> x(2) * 3 + y(4) % 取出 x 的第 2 个元素和 y 的第 4 个元素来作
      运算
```

```
      ans = 9
```

```
>> y(2:4) - 1     % 取出 y 的第 2 至第 4 个元素来作运算
      ans = 6 1 -1
```

在上例中, 2:4 代表一个由 2, 3, 4 组成的向量。

若对 MATLAB 函数的用法有疑问, 可随时使用 help 来寻求帮助, 在提示符“>>”后输入 help linspace, 即可得到

```
Linspace Linearly spaced vector.
Linspace(X1,X2) generates a row vector of 100 linearly
equally spaced points between X1 and X2.
Linspace(X1,X2,N) generates N points between X1 and X2.
For N<2, Linspace returns X2.
```

若要输入矩阵, 则必须在每一行结尾加上分号 (;), 或者按 Enter 键, 如下例:

```
>> A = [1 2 3 4; 5 6 7 8; 9 10 11 12];
```

```
A =
     1     2     3     4
     5     6     7     8
     9    10    11    12
```

若要检查现存于工作空间(workspace)的变量,可键入 who.

```
>> who
```

```
Your variables are:
```

```
A b e x
```

这些是由使用者定义的变量. 若要知道这些变量的详细资料,可键入 whos, 可得

Name	Size	Bytes	Class
A	4x4	128	double array
b	4x1	32	double array
e	4x1	32	double array
x	4x1	32	double array

Grand total is 28 elements using 224 bytes

使用 clear 可以删除工作空间的变量.

```
clear A
```

```
A
```

```
??? Undefined function or variable 'A'.
```

另外 MATLAB 有些永久常数(permanent constants), 虽然在工作空间中看不到, 但使用者可直接取用, MATLAB 的永久常数主要包括:

i 或 j: 基本虚数单位;

eps: 系统的浮点(floating-point)精确度;

inf: 无限大, 如 1/0;

nan 或 NaN: 非数值(not a number), 如 0/0;

pi: 圆周率 π (=3.1415926...);

realmax: 系统所能表示的最大数值;

realmin: 系统所能表示的最小数值;

nargin: 函数的输入参数个数;

nargout: 函数的输出参数个数.

五、MATLAB 的 M 文件

若要一次执行大量的 MATLAB 命令, 可将这些命令存放于一个扩展名为 m 的文件中, 并在 MATLAB 提示符下键入此文件的文件名即可. 此种包含 MAT-

LAB 命令的档案都以 m 为扩展名,因此称其为 M 文件(M-files).通常将 MATLAB 的 M 文件存放在一个特定的文件夹中,如 E 盘的 myfile 中.这时,首先在 E 盘建一个名为 myfile 的文件夹,然后在 MATLAB 的主菜单中作如下的操作:

选 File → SetPath...,出现如图 0.2 的界面.

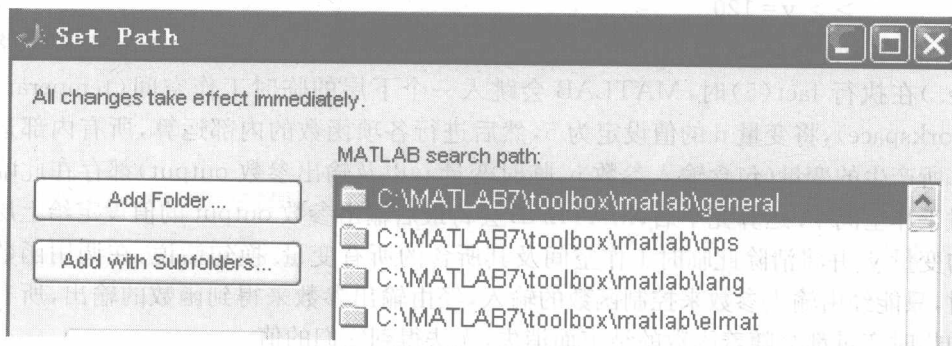


图 0.2 设置路径界面图

选左边的“Add Folder”,选择所需要的文件夹,如 E:\myfile,按“确定”即可.

经过这样的操作,系统就会到 E:\myfile 中去查找存放的文件.例如,一个存放在 E:\myfile 中名为 mytest.m 的 M 文件,包含一系列的 MATLAB 命令,经过上述路径设置以后,只要在命令窗口键入 mytest,即可执行其所包含的命令:

```
% mytest.m
A = [1 2]
B = [2 3]
C = A + B
```

严格来说,M 文件可再细分为命令集(scripts)及函数(functions).前述的 mytest.m 即为命令集,其效用和将命令逐一输入完全一样,因此在命令集可以直接使用工作空间的变量,而且在命令集中设定的变量,也都在工作空间中看得到.函数则需要用到输入参数(input arguments)和输出参数(output arguments)来传递信息,这就像是 C 语言的函数,或是 FORTRAN 语言的子程序(subroutines).举例来说,若要计算一个正整数的阶乘(factorial),可以写一个如下的 MATLAB 函数并将之存档于 fact.m:

```
function output = fact(n)
% FACT Calculate factorial of a given positive integer.
output = 1;
for i = 1:n,
output = output * i;
```

```
end
```

其中, fact 是函数名, n 是输入参数, output 是输出参数, 而 i 则是此函数用到的临时变量. 要使用此函数, 直接键入函数名及适当输入参数值即可:

```
>> y = fact(5)
```

```
>> y = 120
```

(当然, 在执行 fact 之前, 必须先进入 fact. m 所在的目录, 或用 SetPath 设置好路径.) 在执行 fact(5) 时, MATLAB 会跳入一个下层的临时工作空间 (temperary workspace), 将变量 n 的值设定为 5, 然后进行各项函数的内部运算, 所有内部运算所产生的变量 (包含输入参数 n, 临时变量 i, 以及输出参数 output) 都存在此临时工作空间中. 运算完毕后, MATLAB 会将最后输出参数 output 的值设定给上层的变量 y, 并将清除此临时工作空间及其所含的所有变量. 换句话说, 在调用函数时, 只能经由输入参数来控制函数的输入, 经由输出参数来得到函数的输出, 所有的临时变量都会随着函数的结束而消失, 无法得到它们的值.

这里有关阶乘函数只是用来说明 MATLAB 的函数观念. 若实际要计算一个正整数 n 的阶乘 (即 $n!$) 时, 可直接写成 `prod(1:n)`, 或直接调用 gamma 函数: `gamma(n+1)`.

MATLAB 的函数支持递归 (recursive) 调用, 也就是说, 一个函数可以调用它本身. 如 $n! = n \cdot (n-1)!$, 因此前面的阶乘函数可以改成递归的写法:

```
function output = fact1(n)
    % FACT Calculate factorial of a given positive integer recursively.
    if n == 1, % Terminating condition
        output = 1;
        return;
    end
    output = n * fact1(n-1);
```

在写一个递归函数时, 一定要包含结束条件 (递归出口), 否则此函数将会一直调用自己, 永远不会停止, 直到电脑的记忆体被耗尽为止. 以上例而言, $n=1$ 即满足结束条件, 此时直接将 output 设为 1, 而不再调用此函数本身.

六、搜寻路径

在前一节中, 假设 mytest. m 所在的目录是 d:\myfile. 如果不先进入这个目录, MATLAB 就找不到要执行的 M 文件. 如果希望 MATLAB 不论在何处都能执行 mytest. m, 就必须将 d:\myfile 加入 MATLAB 的搜寻路径 (search path) 中. 要了解 MATLAB 的搜寻路径, 键入 path 即可


```
path
```

此搜寻路径会依已安装工具箱(toolboxes)的不同而有所不同。

很显然 d:\myfile 并不在 MATLAB 的搜寻路径中,因此 MATLAB 找不到 mytest.m 这个 M 文件,要将 d:\myfile 加入 MATLAB 的搜寻路径,可用 path 命令

```
>> path(path,'d:\myfile');
```

此时 d:\myfile 已加入 MATLAB 搜寻路径(键入 path 试看看),因此 MATLAB 已经“看”得到 mytest.m

```
> which mytest
```

```
d:\myfile\mytest.m
```

现在就可以直接键入 mytest,而不必先进入 mytest.m 所在的目录。

这是除了刚才介绍的通过 MATLAB 的主窗口中选择 File>Set Path...->Add Folder 来完成以外的另一种方法。

M 文件还可以专门用来存放数据,使用方法和命令集文件相仿,只是其中仅仅存放数据,以便将数据和程序分开存放。

七、结束 MATLAB

有三种方法可以结束 MATLAB:

(1) 键入 exit;

(2) 键入 quit;

(3) 直接关闭 MATLAB 的命令窗口(command window)。

这些是关于 MATLAB 的一般介绍,详细的使用方法通过后面实验来逐步熟悉。

0.4 Mathematica 简介

Mathematica 是美国 Wolfram Research 公司开发的数学软件。它的主要使用者是从事理论研究的数学及其他科学工作者、从事实际工作的工程技术人员、学校的老师和学生。Mathematica 可以用于解决各种领域的涉及复杂符号计算和数值计算的问题。它可以完成许多复杂的工作,如求不定积分、作多项式的因式分解等。它代替了许多以前仅仅只能靠纸和笔解决的工作,这种思维和解题工具的革新对各种研究领域和工程领域产生了深远的影响。

Mathematica 可以做许多符号演算工作,它能做多项式计算、因式分解、展开等,作各种有理式计算、求多项式、有理式方程和超越方程的精确解和近似解,作数值的或一般代数式的向量、矩阵的各种计算,求极限、导数、积分,作幂级数展开,求