

MATLAB 的应用



前 言

随着计算机的发展与普及，继试验方法、理论方法后，科学计算已经成为科学实践的第三种手段。它在物理、力学、化学、生命科学、天文学、经济科学及社会科学等领域中得到了广泛的应用，成为不可缺少的重要工具。因此，适用于计算机教学的计算机语言已逐渐成为综合性本科院校学生的必修或者选修课程。本书是参照教育部关于计算机方面教学的基本要求来编写的，包含了 MATLAB 语言的介绍、在物理系统中分析的应用及在物理教学中的应用等内容。

根据教育部提出的“突出重概念、重方法、重应用、重能力的培养”的要求，本书不仅介绍了 MATLAB 语言的特点与功能，而且着重于解决实际物理问题的基本思想与控制原理，既注重理论的严谨性，又注重实际的应用性。另外，书中有大量的实例，以帮助读者巩固和加深理解相关内容。

科学技术的发展，使得各个学科之间不再是孤立存在的，而是各个领域间相互渗透。物理学涉及了很多控制理论的问题，而近年来控制理论发展迅猛，已较为成熟。因此，对于当代的大学生，学习和理解解决问题的方法，对于其自身科学素质的提高是大有裨益的。

在编写本书的过程中，得到了吴文良教授、徐楠老师和付在琦教授的大力帮助，在此表示感谢。

限于编者的水平，书中难免有不当之处，恳请读者批评指正。

编 者

2013 年 10 月

目 录

第 1 章 MATLAB 简介	(1)
1.1 MATLAB 概述	(1)
1.1.1 MATLAB 的发展	(1)
1.1.2 MATLAB 的系统结构	(2)
1.1.3 MATLAB 的主要功能	(5)
1.1.4 MATLAB 功能演示	(6)
1.2 MATLAB 环境的准备	(8)
1.2.1 MATLAB 的安装	(8)
1.2.2 MATLAB 的启动与退出	(8)
1.2.3 MATLAB 用户界面	(9)
1.2.4 MATLAB 主菜单及其功能	(12)
1.3 MATLAB 搜索路径	(14)
1.3.1 当前目录窗口	(14)
1.3.2 设置 MATLAB 搜索路径的方法	(14)
1.4 MATLAB 帮助系统	(16)
1.4.1 帮助命令	(16)
1.4.2 帮助窗口	(16)
1.4.3 联机演示系统	(17)
第 2 章 MATLAB 基本函数命令及绘图功能	(18)
2.1 MATLAB 数据	(18)
2.1.1 MATLAB 数据特点	(18)
2.1.2 变量和常量	(18)
2.1.3 数据显示格式	(20)
2.2 MATLAB 数据运算	(21)
2.2.1 基本的算术运算	(21)
2.2.2 标点符号的使用	(22)
2.2.3 MATLAB 基本函数	(23)
2.2.4 关系运算	(24)
2.2.5 逻辑运算	(25)
2.3 向量与矩阵运算	(26)
2.3.1 向量及其运算	(26)
2.3.2 矩阵及其运算	(28)

2.3.3	矩阵的算术运算	(34)
2.3.4	矩阵的分解	(37)
2.3.5	矩阵的旋转	(39)
2.3.6	向量和矩阵的范数	(39)
2.3.7	矩阵的条件数	(41)
2.3.8	稀疏矩阵	(41)
2.4	字符串、单元数组和结构数组	(46)
2.4.1	字符串数据	(47)
2.4.2	单元数组	(48)
2.4.3	结构数组	(51)
2.5	MATLAB 数值计算	(52)
2.5.1	多项式的表示和创建	(52)
2.5.2	多项式的运算	(52)
2.5.3	曲线插值和拟合	(57)
2.6	MATLAB 符号计算	(62)
2.6.1	符号变量的创建	(62)
2.6.2	符号函数的运算	(63)
2.7	二维平面绘图	(68)
2.8	三维立体绘图	(80)
第 3 章	MATLAB 程序设计	(90)
3.1	M 文件	(90)
3.1.1	M 文件的概述	(90)
3.1.2	M 文件的建立与编辑	(90)
3.1.3	命令文件	(91)
3.1.4	函数文件	(91)
3.2	程序流程控制语句	(92)
3.2.1	分支控制语句	(92)
3.2.2	循环控制语句	(95)
第 4 章	自动控制系统传递函数的 MATLAB 分析	(98)
4.1	控制系统的数学模型	(98)
4.2	传递函数	(98)
4.3	线性定常时不变对象	(99)
4.4	控制系统数学模型的 MATLAB 命令	(101)
4.4.1	传递函数模型的 MATLAB 命令	(101)
4.4.2	零极点增益模型的 MATLAB 命令	(102)
4.4.3	状态空间模型的 MATLAB 命令	(103)
4.4.4	三种数学模型之间的转换与实例	(104)
4.5	系统的传递函数方框图及其 MATLAB 命令	(111)
4.5.1	串联的传递函数化简的 MATLAB 命令	(112)
4.5.2	并联的传递函数化简的 MATLAB 命令	(113)

4.5.3 系统反馈的传递函数化简的 MATLAB 命令	(115)
第 5 章 自动控制系统稳定性分析的 MATLAB 计算及仿真	(117)
5.1 系统稳定的定义和条件	(117)
5.2 系统稳定性判定方法	(118)
5.2.1 Routh-Hurwitz 判据及 MATLAB 命令	(118)
5.2.2 Nyquist 根轨迹判据方法及 MATLAB 命令	(119)
5.2.3 Lyapunov 意义下稳定性判据方法及 MATLAB 命令	(119)
5.3 系统稳定性判定方法的 MATLAB 仿真	(121)
第 6 章 判定物理系统的可控性与可观性的 MATLAB 程序设计	(132)
6.1 线性系统的可控性	(132)
6.1.1 连续系统的完全可控性数学判据	(132)
6.1.2 离散系统的完全可控性数学判据	(133)
6.1.3 连续系统的完全可控标准形	(134)
6.2 线性系统的可观性	(135)
6.2.1 连续系统的完全可观性数学判据	(135)
6.2.2 离散系统的完全可观性数学判据	(136)
6.2.3 连续系统的完全可观标准形	(138)
6.3 线性系统可观性与可控性的 MATLAB 实例	(139)
第 7 章 物理系统时域分析与频域分析的 MATLAB 仿真实现	(146)
7.1 物理系统时域分析的基本概念	(146)
7.1.1 典型外作用信号	(146)
7.1.2 线性定常系统的时域响应的解	(147)
7.1.3 物理系统时域控制的基本概念	(148)
7.1.4 物理系统响应的性能评价指标	(149)
7.2 时域分析的 MATLAB 命令及实例仿真	(150)
7.2.1 时域响应 MATLAB 仿真的函数指令	(150)
7.2.2 时域分析的 MATLAB 仿真实例	(152)
7.3 物理系统频域分析的基本概念	(159)
7.3.1 频率特性的有关概念	(160)
7.3.2 频率特性的性能评价指标	(161)
7.4 频域分析的 MATLAB 命令及实例仿真	(162)
7.4.1 频域响应 MATLAB 仿真的函数指令	(162)
7.4.2 频域响应 MATLAB 仿真的实例	(165)
第 8 章 线性二次型最优控制的 MATLAB 仿真	(172)
8.1 最优控制的问题	(172)
8.1.1 最优控制的提出	(172)
8.1.2 最优控制问题的数学分类法	(173)
8.1.3 最优控制的性能指标	(173)
8.1.4 最优控制问题的求解方法	(174)
8.1.5 线性二次型的最优控制	(174)

8.2	连续系统线性二次型最优控制的 MATLAB 实现	(175)
8.2.1	连续系统线性二次型最优控制的数学描述	(175)
8.2.2	连续系统线性二次型最优控制的 MATLAB 实现	(175)
8.3	离散系统线性二次型最优控制的 MATLAB 实现	(180)
8.3.1	离散系统稳态线性二次型最优控制	(180)
8.3.2	离散系统最优控制实例	(180)
第 9 章	MATLAB 在物理教学中的综合应用	(182)
9.1	MATLAB 在电工学中的应用	(182)
9.1.1	电阻电路	(182)
9.1.2	动态电路分析	(185)
9.1.3	正弦交流电路	(190)
9.1.4	电路的频率响应	(192)
9.2	MATLAB 在光学中的应用	(194)
9.2.1	理想情况下的双缝干涉色图	(195)
9.2.2	光栅	(197)
9.2.3	光波的叠加	(198)
9.2.4	牛顿环形成	(201)
9.2.5	劈尖干涉	(202)
9.3	MATLAB 在电磁学中的应用	(204)
9.3.1	点电荷的电势分布	(204)
9.3.2	带电粒子在电场中的运动	(206)
9.3.3	带电粒子在电磁场中的运动	(208)
9.3.4	线电荷产生的电位	(210)
9.3.5	毕奥—萨伐尔定律	(211)
9.4	MATLAB 在力学中的应用	(215)
9.4.1	简谐振动	(215)
9.4.2	驻波形成—火柴杆图	(216)
9.5	MATLAB 在热学中的应用	(218)
9.5.1	固体热容量随温度变化曲线	(218)
9.5.2	熵与内能	(220)
9.5.3	受热金属块的热传导过程	(221)
9.5.4	虫口问题	(224)
9.6	MATLAB 在流体中的应用	(225)
9.6.1	流线	(225)
9.6.2	对液压消音器里流体特性的分析	(227)
参考文献		(233)

第 1 章 MATLAB 简介

随着计算机软件和硬件的不断发展，20 世纪 80 年代以来，出现了科学计算语言，也称数学软件，目前比较流行的有 MATLAB、Maple、Spss、C 语言等。它们具有功能强、效率高、简单易学等特点，在许多领域得到了广泛的应用，目前流行最广的当属 MATLAB 语言。

本章首先介绍 MATLAB 的发展、系统结构和主要功能，然后介绍 MATLAB 软件的使用环境。

1.1 MATLAB 概述

1.1.1 MATLAB 的发展

MATLAB 是 Matrix Laboratory (矩阵实验室) 的缩写。20 世纪 70 年代，美国新墨西哥大学计算机科学系主任 Cleve Moler 教授为了减轻学生编程的负担，为学生编写了方便使用 Linpack 和 Eispack 的接口程序，并命名为 MATLAB，这就是 MATLAB 的雏形。

早期的 MATLAB 是用 FORTRAN 语言编写的，功能十分简单，但作为免费软件，还是吸引了大批的使用者。经过几年的时间，于 1984 年由 John Little、Cleve Moler、Steve Bangert 合作成立了 Math Works 公司，正式把 MATLAB 推向市场。从这时起，MATLAB 的核心采用 C 语言编写，功能越来越强大。到 20 世纪 90 年代，MATLAB 已成为国际控制界的标准计算软件。Math Works 公司于 1992 年推出划时代意义的 MATLAB 4.0 版本，从此告别了 DOS 版本。1994 年推出的 MATLAB 4.2 版本扩充了 MATLAB 4.0 版本的功能，尤其是在图形界面设计方面提供了新的方法。1997 年，MATLAB 5.0 版本问世，它支持更多的数据结构，如单元数据、结构数据、多维数组、对象与类等。紧接着是 MATLAB 5.1、MATLAB 5.2 版本。1999 年推出了 MATLAB 5.3 版本，在这个版本中包含了全新版本的最优化工具箱和 Simulink 3.0 版，至此 MATLAB 达到了很高的水平。2000 年，MATLAB 6.0 版本面世，在操作界面上有了很大的改观，计算的速度变得更快，图形界面设计更趋于合理，与外部 C 语言接口及转换的兼容性更强，与之推出的 Simulink 4.0 版功能也十分强大。2002 年，又推出了 MATLAB 6.5 版本及 Simulink 5.0 版。2004 年 7 月，Math Works 公司推出了最新的 MATLAB 7.0 版本，其中集成了最新的 MATLAB 7.0 编译器、Simulink 6.0 版仿真软件以及很多工具箱。这一版本增加了很多新的功能和特性，内容十分丰富。随后几乎每年 MATLAB 都会推出新的版本。目前最新的版本为 2013 年发布的 MATLAB 2013a。本书以 MATLAB 7.0 为基础，全面介绍它的各种功能和使用。

1.1.2 MATLAB 的系统结构

MATLAB 系统结构分为三个层次，其核心的部分是软件主包 MATLAB。整个系统结构如图 1.1 所示。

1.1.2.1 MATLAB 系统的基础层

MATLAB 系统的基础层分为三个部分，即软件主包 MATLAB、编译器 Compiler 和工具箱 Toolboxes。该层次的核心内容是 MATLAB 部分，也是整个系统的最重要的部分。

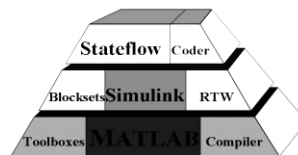


图 1.1 MATLAB 的系统结构

软件主包 MATLAB 构成的主要内容如下：

- (1) MATLAB 语言。
- (2) MATLAB 的数学函数与数值(符号) 计算。
- (3) MATLAB 的数据可视化。
- (4) MATLAB 的图形处理系统。
- (5) MATLAB 的应用程序接口。
- (6) MATLAB 的工作环境管理。

MATLAB Compiler 可以将 M 文件编译成 C/C++ 源代码和独立应用，即将 MATLAB 语言编写的 M 文件自动转换为 C/C++ 源代码文件，然后编译成为可执行文件或动态链接库，使程序的执行效率大大高于 M 文件。另外，可以利用 Math Works 提供的 C/C++ 数学库或图形库，将 M 文件编译成 EXE 文件，脱离 MATLAB 环境执行。

围绕 MATLAB 这个计算核心，有许多针对不同应用领域开发的应用程序软件包，称为 MATLAB 系统的工具箱。Math Works 公司所提供的应用工具箱大约有 40 多个，另外还有一些其他公司提供的专门应用领域的工具箱也合并在该系统中。目前可用的工具箱几乎有 100 个，并还在不断增加。可用的工具箱所涉及的应用领域大致如下：

(1) 应用数学类工具箱。

① 统计工具箱 (Statistics Toolbox)。

- * 概率分布和随机数生成
- * 多变量分析
- * 回归分析
- * 主元分析
- * 假设检验

② 优化工具箱 (Optimization Toolbox)。

- * 线性规划和二次规划
- * 求函数的最大值和最小值
- * 多目标优化
- * 约束条件下的优化
- * 非线性方程求解

③ 样条工具箱 (Spline Toolbox)。

- * 分段多项式和 B 样条
- * 样条的构造

- * 曲线拟合及平滑
- * 函数微分、积分
- ④ 偏微分方程工具箱 (Partial Differential Equation Toolbox)。
 - * 二维偏微分方程的图形处理
 - * 几何表示
 - * 自适应曲面绘制
 - * 有限元方法
- ⑤ 符号数学工具箱 (Symbolic Math Toolbox)。
 - * 符号表达式和符号矩阵的创建
 - * 符号微积分、线性代数、方程求解
 - * 因式分解、展开和简化
 - * 符号函数的二维图形
 - * 图形化函数计算器
- (2) 电子技术类工具箱。
 - ① 信号处理工具箱 (Signal Processing Toolbox)
 - * 数字和模拟滤波器设计、应用及仿真
 - * 谱分析和估计
 - * FFT、DCT 等变换
 - * 参数化模型
 - ② 通信工具箱 (Communication Toolbox)。
 - * 信号编码
 - * 调制解调
 - * 滤波器和均衡器设计
 - * 通道模型
 - * 同步
 - * 错误控制编码
 - ③ 小波工具箱 (Wavelet Toolbox)。
 - * 基于小波的分析 and 综合
 - * 图形界面和命令行接口
 - * 连续和离散小波变换及小波包
 - * 一维、二维小波
 - * 自适应去噪和压缩
 - (3) 图像处理工具箱 (Image Processing Toolbox)。
 - * 二维滤波器设计和滤波
 - * 图像恢复增强
 - * 色彩、集合及形态操作
 - * 二维变换
 - * 图像分析和统计
 - (4) 财经与金融工具箱 (Financial Toolbox)。
 - * 成本、利润分析, 市场灵敏度分析

- * 业务量分析及优化

- * 偏差分析

- * 资金流量估算

- * 财务报表

(5) 神经网络工具箱 (Neural Network Toolbox)。

- * BP、Hopfield、Kohonen、自组织、径向基函数等网络

- * 竞争、线性、Sigmoidal 等传递函数

- * 前馈、递归等网络结构

- * 性能分析及应用

(6) 自动控制类工具箱。

① 线性矩阵不等式控制工具箱 (LMI Control Toolbox)。

- * LMI 的基本用途

- * 基于 GUI 的 LMI 编辑器

- * LMI 问题的有效解法

- * LMI 问题的解决方案

② 控制系统工具箱 (Control System Toolbox)。

- * 状态空间和传递函数

- * 模型转换

- * 频域响应: Bode 图、Nyquist 图、Nichols 图

- * 时域响应: 冲击响应、阶跃响应、斜坡响应等

- * 根轨迹、极点配置、LQG

③ 频域系统辨识工具箱 (FD System Identification Toolbox)。

- * 辨识具有未知延时的离散和连续系统

- * 计算幅值/相位、零点/极点的位置区间

- * 设计周期激励信号、最小峰值、最优能量

在 MATLAB 产品家族中, MATLAB 工具箱是整个体系的基座, 它是一个语言编程型 (M 语言) 开发平台, 提供了体系中其他工具所需要的集成环境 (比如 M 语言的解释器)。同时, 由于 MATLAB 对矩阵和线性代数的支持, 使得工具箱本身也具有强大的数学计算能力。可以预测, 由于 MATLAB 具有丰富的 Toolbox 资源的支持, 使得用户可以方便地进行具有开创性的建模与算法开发工作, 并通过 MATLAB 强大的图形和可视化能力反映算法的性能和指标。所得到的算法可以在 Simulink (仿真) 环境中以模块化的方式实现, 通过全系统建模, 进行全系统的动态仿真, 以得到算法在系统中的动态验证。

1.1.2.2 Simulink 系统建模仿真层

MATLAB 产品体系的演化历程中最重要的一個体系变更是引入了 Simulink, 用来对动态系统建模仿真。其框图化的设计方式和良好的交互性, 对工程人员本身的计算机操作与编程的熟练程度的要求降到了最低, 工程人员可以把更多的精力放到理论和技术的创新上去。Simulink 是建立在 MATLAB 之上的系统建模仿真环境, 它提供了一个图形化的编辑界面。Simulink 的基本环境如图 1.2 所示。

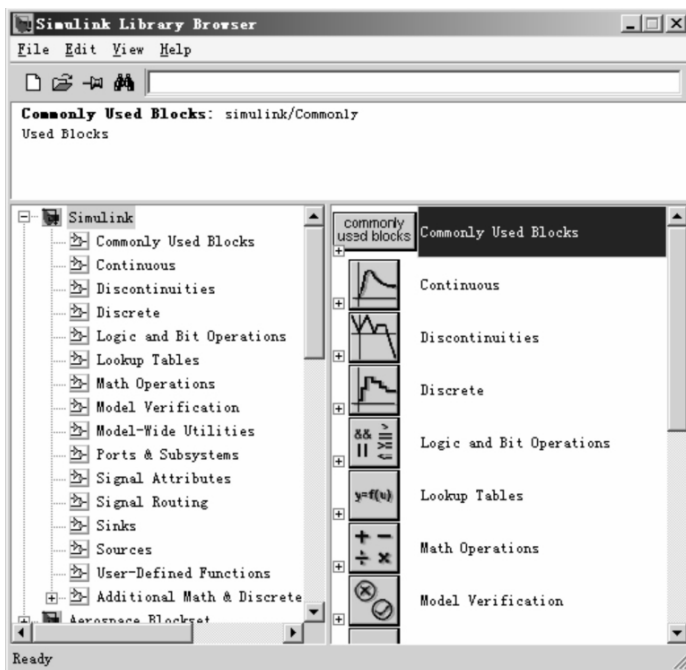


图 1.2 Simulink 的基本环境

1.1.3 MATLAB 的主要功能

MATLAB 自推出市场以来，经过几十年的发展和竞争，现已风靡全球。目前，在高等院校，MATLAB 已经成为线性代数、控制理论、数字信号处理等许多课程的基本数学教学工具，成为大学生必须掌握的基本编程语言。在科研与工程应用领域，MATLAB 已被广泛地应用于科学研究和解决各种具体的实际问题。MATLAB 具有下面一些主要功能。

1.1.3.1 友好的工作平台和编程环境

MATLAB 由一系列工具组成。这些工具方便用户使用 MATLAB 的函数和文件，其中许多工具采用的是图形用户界面，包括 MATLAB 桌面和命令窗口、历史命令窗口、编辑器和调试器、路径搜索和用于用户浏览帮助、工作空间、文件的浏览器。随着 MATLAB 的商业化以及软件本身的不断升级，MATLAB 的用户界面也越来越精致，更加接近 Windows 的标准界面，人机交互性更强，操作更简单。而且新版本的 MATLAB 提供了完整的联机查询、帮助系统，极大地方便了用户的使用。简单的编程环境提供了比较完备的调试系统，程序不必经过编译就可以直接运行，而且能够及时地报告出现的错误并进行出错原因的分析。

1.1.3.2 简单易学的程序语言

MATLAB 一个高级的矩阵/阵列语言，它包含控制语句、函数、数据结构、输入和输出和面向对象编程等。用户可以在命令窗口中将输入语句与执行命令同步，也可以先编写好一个较大的复杂的应用程序（M 文件）后再一起运行。新版本的 MATLAB 语言是基于最流行的 C++ 语言的，因此语法特征与 C++ 语言极为相似，而且更加简单，更加符合科技人员对数学表达式的书写格式，使之更利于非计算机专业的科技人员使用。并且这种语言可移植性好、可拓展性极强，这也是 MATLAB 能够深入到科学研究及工程计算各个领域的重要原因。

1.1.3.3 强大的数据处理能力

MATLAB 是一个包含大量计算算法的集合。其拥有 600 多个工程中要用到的数学运算函数，可以方便地实现用户所需的各种计算功能。在计算要求相同的情况下，使用 MATLAB 的编程工作量会大大减少。MATLAB 的这些函数集包括从最简单、最基本的函数到诸如矩阵、特征向量、快速傅里叶变换的复杂函数。函数所能解决的问题大致包括矩阵运算和线性方程组的求解、微分方程及偏微分方程组的求解、符号运算、傅里叶变换和数据的统计分析、工程中的优化问题、稀疏矩阵运算、复数的各种运算、三角函数和其他初等数学运算、多维数组操作以及建模动态仿真等。

1.1.3.4 出色的图形处理功能

利用 MATLAB 绘图十分方便，它既可以绘制各种图形，包括二维图形和三维图形，还可以对图形进行修饰和控制，以增强图形的表现效果。

1.1.3.5 应用广泛的模块集合工具箱

MATLAB 对许多专门的领域都开发了功能强大的模块集合工具箱。一般来说，它们都是由特定领域的专家开发的，用户可以直接使用工具箱学习、应用和评估不同的方法，而不需要自己编写代码。

1.1.3.6 实用的程序接口和发布平台

MATLAB 可以利用 MATLAB 编译器和 C/C++ 数学库和图形库，将自己的 MATLAB 程序自动转换为独立于 MATLAB 运行的 C 和 C++ 代码。允许用户编写可以和 MATLAB 进行交互的 C 或 C++ 语言程序。另外，MATLAB 网页服务程序还容许在 Web 应用中使用自己的 MATLAB 数学和图形程序。MATLAB 的一个重要特色就是具有一套程序扩展系统和一组称之为工具箱的特殊应用子程序。工具箱是 MATLAB 函数的子程序库，每一个工具箱都是为某一类学科专业和应用而定制的，主要包括信号处理、控制系统、神经网络、模糊逻辑、小波分析和系统仿真等方面的应用。

1.1.3.7 应用软件开发（包括用户界面）

在开发环境中，使用户更方便地控制多个文件和图形窗口；在编程方面支持函数嵌套、有条件中断等；在图形化方面，有了更强大的图形标注和处理功能，包括连接注释等；在输入输出方面，可以直接与 Excel 进行连接。

1.1.4 MATLAB 功能演示

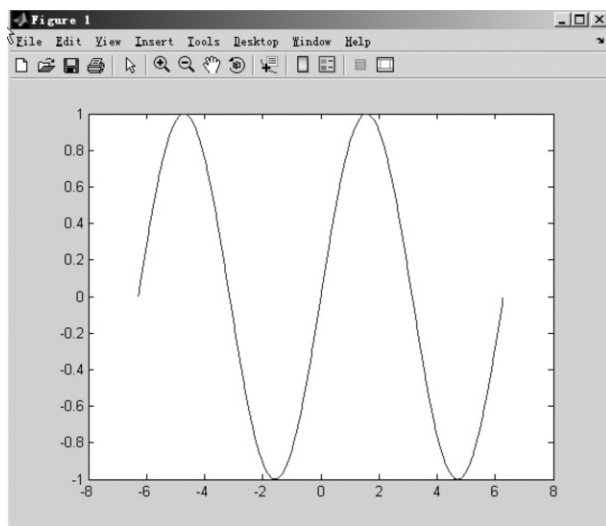
下面通过几个简单的例子来演示 MATLAB 的功能。

例 1.1 画出 $y = \sin x$ 的图形。

在 MATLAB 命令窗口输入命令：

```
x = -2*pi:pi/180:2*pi;  
plot(x, sin(x))
```

其中第一条命令是建立 x 向量， x 从 -2π 变化到 2π ，第二条命令是绘制曲线，如图 1.3 所示。

图 1.3 $y = \sin x$ 函数图形

例 1.2 求矩阵 A 的最大元素和最小元素。

$$A = \begin{bmatrix} 12 & -56 & 67 \\ 23 & 25 & 35 \\ -10 & 100 & 20 \end{bmatrix}$$

命令如下：

```
A=[12,-56,67;23,25,35;-10,100,20];
```

```
max(max(A))
```

```
ans =
```

```
100
```

```
min(min(A))
```

```
ans =
```

```
-56
```

例 1.3 解方程 $2x^4 - 3x^2 + 9x - 5 = 0$ 。

命令如下：

```
p=[2,0,-3,9,-5];
```

```
x=roots(p)
```

其中第一条命令是建立多项式的系数向量，第二条命令是使用 roots 函数求根。其结果如下：

```
x =
```

```
-2.0652
```

```
0.7035 + 1.1593i
```

```
0.7035 - 1.1593i
```

```
0.6583
```

例 1.4 求函数 $y = x \sin x$ 的二阶导数。

命令如下：

```
syms x y
```

```
y = x*sin(x);
```

```
diff(y,x,2)
```

其中第一条命令是定义变量 x 、 y ，第二条命令是输入函数 y ，第三条命令是对函数求二阶导数。其结果如下：

```
ans =
```

```
2*cos(x) - x*sin(x)
```

1.2 MATLAB 环境的准备

1.2.1 MATLAB 的安装

安装 MATLAB 7.0 系统，需运行系统自带的安装程序 setup.exe。开始安装时，将 MATLAB 7.0 系统光盘放入 CD-ROM 驱动器中，一般情况下，安装程序会自动运行。如果没有自动运行，则双击 setup.exe 文件即可。运行后，将进入 MATLAB 安装程序的欢迎 (Welcome to the Math Works Installer) 对话框，用户可以按照安装提示依次操作。

在欢迎对话框中，有“Install”和“Update license without installing anything, using a new PLP”两个可选项。选择 Install 选项可直接安装 MATLAB。后一个选项一般用于更新个人许可密码(Personal License Password, PLP)，适用于以前已经安装了 MATLAB 7.0 试用版，现在将升级为正式版的情况，在这种情况下，不需要重新安装，只需要更新 PLP 即可。

在欢迎对话框中，选择 Install 选项，单击 Next 按钮，将进入个人许可信息输入 (License Information) 对话框，这时需要输入用户个人信息及个人许可密码。

输入个人信息后，单击 Next 按钮打开许可协议 (License Agreement) 对话框，选中 Yes 表示接受协议，然后单击 Next 按钮打开产品和文件夹选择 (Product and Folder Selection) 对话框。在该对话框中，需要输入安装的路径或单击 Browse 按钮来选择安装路径，然后选择需要安装的 MATLAB 组件。对于 MATLAB 7.0 版本，其功能组件很多，用户可以自行取舍。但是对于软件运行所必需的组件必须选中，如主模块。当用户确定选择方案后，就可以单击 Next 按钮，进入定制安装 (Custom Installation) 对话框，可以选择一些安装选项。然后单击 Next 按钮，进入确认 (Confirmation) 对话框，对所选择的组件进行确认后，单击 Install 按钮，进入文件复制 (Copy File) 对话框。

在文件复制前，系统会检测硬盘空间是否满足要求，如果空间不够，系统会发出警告，让用户删除某些不需要的文件，为系统的安装腾出空间。如果空间满足要求，安装程序会自动进入文件复制阶段。

在文件复制完成后，会弹出产品配置 (Product Configuration Notes) 对话框。在该对话框中，用户可以直接单击 Next 按钮进入安装完成 (Setup Complete) 对话框。在安装完成对话框中，单击 Finish 按钮退出安装程序。至此，MATLAB 系统安装完毕。

1.2.2 MATLAB 的启动与退出

1.2.2.1 MATLAB 系统的启动

与一般的 Windows 程序一样，启动 MATLAB 系统有以下 3 种常见方法：

(1) 在 Windows 桌面，单击任务栏上的“开始”按钮，选择“程序”菜单项，然后选择“MATLAB 7.0”程序选项，就可启动 MATLAB 系统。

(2) 在 MATLAB 7.0 安装路径中找到 MATLAB 系统启动程序 matlab.exe 并运行。

(3) 利用建立快捷方式的功能，将 MATLAB 系统启动程序以快捷方式放在 Windows 桌面上，只要双击该图标即可启动 MATLAB。

启动 MATLAB 后，将进入 MATLAB 7.0 集成环境，如图 1.4 所示。MATLAB 7.0 集成环境包括多个窗口，除了 MATLAB 主窗口外，还有命令(Command)窗口、工作空间 Workspace)窗口、命令历史(Command History)窗口和当前目录(Current Directory)窗口。这些窗口都可以内嵌在 MATLAB 主窗口中，组成 MATLAB 的工作界面。此外，在 MATLAB 主窗口的左下角，还有个 Start 按钮。

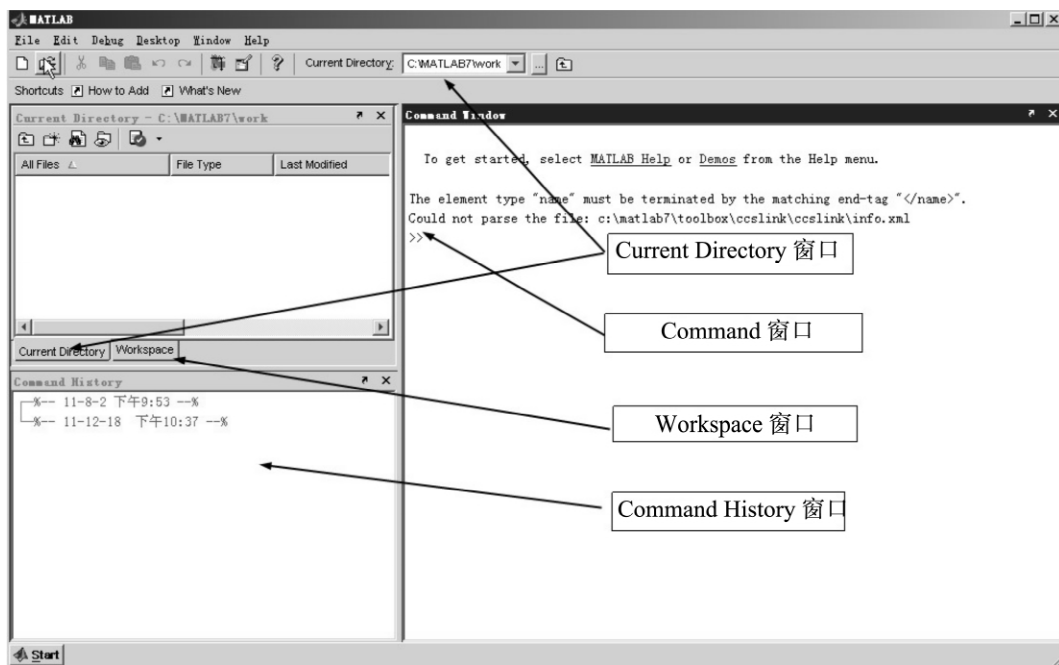


图 1.4 MATLAB 默认窗口

1.2.2.2 MATLAB 系统的退出

要退出 MATLAB 系统，也有以下 3 种常见方法：

- (1) 在 MATLAB 主窗口 File 菜单中选择 Exit MATLAB 命令。
- (2) 在 MATLAB 命令窗口输入 Exit 或 Quit 命令。
- (3) 单击 MATLAB 主窗口右上角的关闭按钮。

1.2.3 MATLAB 用户界面

MATLAB 具有几个显示窗口，如果不小心关闭了一个重要窗口，可通过选择菜单栏 Desktop > Desktop Layout > Default 菜单恢复到默认设置。

1.2.3.1 Command 窗口

MATLAB 有两种使用模式：一种是命令模式，如果对具体的命令需要立即得到结果，则这一种模式非常有效，它类似于一个科学计算器环境；另一种是编辑模式，在该模式下可

以逐步设计出实际的解决方案，它可以以独立窗口的形式浮动在界面上(选中命令窗口，再选择 Desktop 菜单中的 Undock Command Window 命令)，也可以内嵌在 MATLAB 的工作界面上。

MATLAB 命令窗口中的“>>”为命令提示符，表示 MATLAB 正处于准备状态。一般来说，一个命令行输入一条命令，命令行以回车结束。但一个命令行也可以输入若干条命令，各命令之间以逗号隔开，若前面带有分号，则逗号可以省略。例如：

```
x = 13, y = 10
```

```
x = 13; y = 10
```

这两条命令行都是合法的，第一条命令执行后显示 x 和 y 的值；第二条命令执行后因 x 后面带有分号，x 的值不显示，只显示 y 的值。

如果一条命令行很长，一个物理行之内写不下，可以在第一个物理行之间加 3 个小黑点并按下回车键，然后继续写命令的其他部分，所以这 3 个小黑点称为续行符。

在 MATLAB 中，有很多控制键和方向键可用于命令窗口中命令行的编辑，见表 1.1，使用这些键将大大提高操作效率。

表 1.1 命令行编辑常用的控制键及其功能

键 名	功 能
↑	向上寻找已输入过的命令
↓	向下寻找已输入过的命令
←	在当前行中左移光标
→	在当前行中右移光标
PgUp	向上翻滚一页
PgDn	向下翻滚一页
Del	删除光标右边的字符
Backspace	删除光标左边的字符
Home	光标移到当前行首
End	光标移到当前行尾
Esc	删除当前所有内容

1.2.3.2 Command History 窗口

Command History 窗口按照先后顺序记录了在 Command 窗口中执行过的命令。当退出 MATLAB 或者运行 clc(清除命令)时，Command 窗口中列出的命令将被清除，而 Command History 窗口中仍将保持已发出的命令列。可以通过在 Command History 窗口中双击一条命令或者通过选中一条命令并将其拖到 Command 窗口中。如果需要清除 Command History 窗口中的命令，可以利用菜单栏的 Edit > Clear Command History 菜单对 Command History 进行清除。

1.2.3.3 Workspace 窗口

Workspace 窗口会记住在 Command 窗口中运行命令时所定义的变量。在该窗口中显示了变量名、变量当前值以及变量数据类型三种标识，可以对变量进行观察、编辑、保存和删除，如图 1.5 所示。如果已经取消了 Workspace 窗口，仍可以通过 who 或 whos 命令找出已

经定义的变量，其中 `who` 列出变量名的名称，而 `whos` 将同时列出变量名的名称、大小及其类型，如图 1.6 和图 1.7 所示。

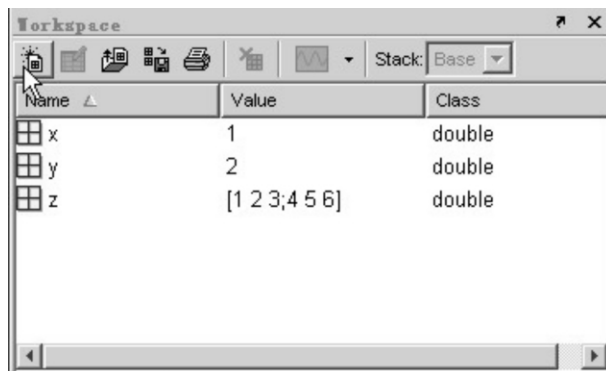


图 1.5 Workspace 窗口

```
>> who
Your variables are:
x y z
```

图 1.6 who 命令窗口

```
>> whos

Name      Size      Bytes  Class

x         1x1         8  double array
y         1x1         8  double array
z         2x3        48  double array
```

Grand total is 8 elements using 64 bytes

图 1.7 whos 命令窗口

1.2.3.4 Current Directory 窗口

Current Directory 窗口是向硬盘上存取文件或存储时用到的窗口。它可以在 Current Directory 窗口的下拉式列表中选择当前的目录位置，或者使用下拉列表右边的浏览按钮，浏览计算机上的文件夹来改变当前目录。

1.2.3.5 Document 窗口

双击 Workspace 窗口中任意变量的值将自动运行一个包含数组编辑器的 Document 窗口，如图 1.8 所示。存储变量中的值以数据表格的形式显示，可以在数组编辑器中改变变量的值，也可以添加新的值。但在命令行后加分号时，将禁止显示其中的赋值操作，因此变量的值也不会出现在 Command 窗口中出现。