

计算机工程设计与应用开发丛书

MATLAB 6.0 基础及应用

清源计算机工作室 编著

计算机工程设计与应用开发丛书

MATLAB 6.0 基础及应用

清源计算机工作室 编著



机械工业出版社

本书根据作者使用 MATLAB 的经验并结合大量的例子，介绍了 MATLAB 6.0 的主要功能、函数以及使用方法。全书共分为 12 章，系统地介绍了 MATLAB 6.0 的基础知识，包括数值与矩阵计算、图形处理功能、MATLAB 编程基础知识以及 MATLAB 类和对象等。

本书可作为各工程领域的教师、学生和科研人员的参考书，同时对初学者来说，也是一本理想的入门教材。

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

责任编辑：边 萌 封面设计：康 健

责任印制：路 琳

中国建筑工业出版社密云印刷厂印刷·新华书店北京发行所发行

2001 年 5 月第 1 版第 1 次印刷

787mm×1092mm 1/16 · 19.75 印张·487 千字

0001—5000 册

定价：35.00 元（1CD，含配套书）

ISBN7-900066-35-7/TP·33

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

本社购书热线电话（010）68993821、68326677-2527

前 言

随着计算机技术的发展, 计算机软件在工程设计领域的应用越来越广。在机械、电子、建筑等行业, 应用计算机软件进行产品设计的 CAD 软件也非常丰富, 使产品设计人员能够高效率地进行各自领域的产品分析、设计等工作。这些应用于工程设计领域的 CAD 软件有 AutoCAD、Protel、MATLAB 等。这些软件极大地提高了机械、电子等行业的产品设计的质量与效率, 是目前 CAD 领域应用最为广泛的软件, 是工程设计领域中最有用的辅助设计软件。为了帮助工程设计人员学习这些软件, 快速掌握这些软件的使用与开发技术, 我们特编写这套“计算机工程设计与应用开发丛书”。AutoCAD 2000i 主要应用于机械产品的设计和开发, 以及 AutoCAD 2000i 的二次开发; Protel 99 SE 主要应用于电子原理图的设计、电路板的设计和绘制, 以及电子逻辑分析和仿真等; MATLAB 6.0 主要应用于工程方面的数学计算、自动控制系统的分析, 以及图形与图像处理等。

本套书主要面向工程设计人员, 涉及的知识不但包括软件应用知识, 还包括专业基础知识, 以及软件在相关领域的二次开发技术知识, 是机械、电子领域技术人员的最佳参考书。本套书在介绍软件的使用过程中, 结合丰富的实例进行讲解, 使读者能快速掌握相关的知识。本套书不但讲述软件的基础应用知识, 还讲述了软件的中、高级应用知识, 是一套面向中、高级读者的全面而系统的参考书。本套书包括《AutoCAD 2000i 图形绘制与应用》和《AutoCAD 2000i 应用开发与实例》; 《Protel 99 SE 原理图和 PCB 设计》和《Protel 99 SE 电路设计与仿真》; 《MATLAB 6.0 基础及应用》和《MATLAB 6.0 高级应用——图形图像处理》, 基本覆盖了三种软件在机械领域和电子领域的应用。全套书均附有光盘, 以实例为主, 将软件的实际应用很生动地展现在读者面前。

本套书由清华大学从事该领域工作多年的博士生和硕士生编写, 具有贴近读者的特点。书中列举了大量的实例, 是使用这三种软件的工程设计人员不可多得的参考书。

清源计算机工作室

编 者 的 话

MATLAB 6.0 是 Math Works 公司的产品。Math Works 公司在 1982 年发行了 MATLAB 的 DOS 版本,不断完善至今形成了三种主要的版本,分别是 Windows 95 版、Windows NT 版和 Macintosh 版。MATLAB 是一种交互式、面向对象的程序设计语言,广泛应用于工业界与学术界,主要用于矩阵运算,同时在数值分析、自动控制模拟、数字信号处理、动态分析和绘图等方面也具有强大的功能。

MATLAB 6.0 程序设计语言结构完整,且具有优良的移植性,它的基本数据元素是不需要定义的数组。它可以高效率地解决工业计算问题,特别是关于矩阵和矢量的计算。MATLAB 6.0 与 C 语言和 Fortran 语言相比更容易被掌握。寥寥几行 MATLAB 程序就可以实现你意想不到的功能。

另外, MATLAB 6.0 提供了一种特殊的工具:工具箱 (ToolBoxes)。这些工具箱主要包括:信号处理 (Signal Processing)、控制系统 (Control Systems)、神经网络 (Neural Networks)、模糊逻辑 (Fuzzy Logic)、小波 (Wavelets) 和模拟 (Simulation) 等等。Math Works 公司和各个领域的专家还在积极地完善现有工具箱的功能,并不断推出新的工具箱。不同领域、不同层次的用户通过对相应工具的学习和应用,可以方便地进行计算、分析及设计工作。

本书通过大量的简单例子,一步一步地带领读者进入 MATLAB 6.0 的世界。本书系统地介绍了 MATLAB 6.0 的基础知识,包括数值与矩阵计算、图形处理功能、MATLAB 编程基础知识以及 MATLAB 类和对象等。通过对本书的学习,读者将领会到 MATLAB 6.0 的强大功能,并能利用它来帮助自己进行工程设计与计算。

读者在阅读本书时应该注意:以 “>>” 开头的内容为输入到命令窗口的命令执行过程,而函数程序段则没有 “>>” 开头。

由于作者水平有限,时间仓促,书中缺点和不足在所难免,敬请广大读者批评指正。

编 者

目 录

前言

编者的话

第 1 章 MATLAB 6.0 概述	1	2.1.6 矢量及矩阵的范数	31
1.1 MATLAB 简介	1	2.2 解线性方程	32
1.2 MATLAB 的安装	2	2.2.1 方阵系统	33
1.3 MATLAB 的工作环境	4	2.2.2 过限定系统	34
1.3.1 命令窗口	4	2.2.3 限定不足系统	35
1.3.2 启动平台	7	2.3 逆矩阵及行列式	37
1.3.3 工作空间(Workspace)	8	2.3.1 方阵的行列式与逆矩阵	37
1.3.4 命令历史纪录和当前路径窗口	8	2.3.2 伪逆矩阵	39
1.4 M 文件的编辑调试环境	8	2.4 矩阵分解	41
1.5 编辑器/调试器参数的设置	11	2.4.1 平方根分解法	42
1.5.1 编辑器/调试器字体和颜色的设置	12	2.4.2 高斯消去法	43
1.5.2 编辑器/调试器显示的设置	13	2.4.3 正交分解法	45
1.5.3 键盘和缩进的设置	14	2.5 矩阵幂与指数	48
1.5.4 打印参数的设置	15	2.6 特征值与特征矢量	51
1.6 MATLAB 的工作空间	15	2.7 奇异值分解	54
1.6.1 查看与清除工作空间中的变量	15	第 3 章 多项式和插值	57
1.6.2 保存工作空间	16	3.1 多项式	57
1.6.3 用特定的格式保存工作空间	17	3.1.1 多项式的表示及其运算	57
1.6.4 加载工作空间	17	3.1.2 多项式的曲线拟合	60
1.6.5 加载 ASCII 数据文件	18	3.1.3 部分分式	61
1.6.6 工作空间浏览器与数组编辑	18	3.2 插值	63
1.6.7 保存在字符串变量中的文件名	19	3.2.1 一维插值	63
1.6.8 使用通配符	20	3.2.2 二维插值	66
1.7 MATLAB 的路径搜索	21	3.2.3 插值方法的比较	67
1.8 命令窗口的设置	22	3.3 多维数据的插值	71
1.9 在线帮助	23	3.3.1 三维数据的插值	71
第 2 章 矩阵与线性代数	26	3.3.2 多维数据的插值	72
2.1 矩阵与线性代数的基本概念	26	3.3.3 多维数据的网格	73
2.1.1 矩阵的加法与减法运算	27	3.4 三角形测量与离散数据插值	73
2.1.2 矢量积与矩阵转置	27	3.4.1 Delaunay 三角分解	74
2.1.3 矩阵乘法	29	3.4.2 最近点搜索	75
2.1.4 单位矩阵	30	3.4.3 Voronoi 图表	76
2.1.5 Kronecker Tensor 乘积	30	第 4 章 功能函数	79

4.1 函数的表示	79	6.3.2 饼图	140
4.2 函数图像	79	6.3.3 柱形图	142
4.3 求函数的最小值与零点	83	6.3.4 绘制离散性数据的图形	144
4.3.1 求一元函数的最小值	83	6.3.5 绘制速度与方向的图形	148
4.3.2 求多元函数的最小值	84	6.3.6 等高线	152
4.3.3 设置求最小值的参数	84	第 7 章 稀疏矩阵	155
4.3.4 求一元函数的零点	85	7.1 稀疏矩阵的概念	155
4.4 积分	87	7.1.1 稀疏矩阵的保存	155
4.4.1 求一元函数的定积分	87	7.1.2 创建稀疏矩阵	155
4.4.2 计算曲线长度	88	7.2 查看矩阵元素	159
4.4.3 二重积分	89	7.2.1 查看非零元素的信息	160
第 5 章 数据分析与统计	90	7.2.2 用图形方式查看稀疏矩阵的信息	160
5.1 列向数据	90	7.2.3 查找矩阵中的非零元素	162
5.2 基本分析函数	91	7.3 邻接矩阵及图形	162
5.3 协方差矩阵与相关系数矩阵	94	7.3.1 使用邻接矩阵绘图	163
5.4 微分与梯度	95	7.3.2 Bucky Ball	163
5.5 数据预处理	96	7.3.3 图形与稀疏矩阵的特征	165
5.6 曲线拟合	98	7.4 稀疏矩阵的操作	167
5.6.1 曲线拟合实例分析	100	7.4.1 标准数学函数处理稀疏矩阵	167
5.6.2 误差范围	107	7.4.2 稀疏矩阵的交换与重新排序	168
5.7 微分方程数据过滤器	107	7.4.3 稀疏矩阵的分解	172
5.8 傅里叶变换	109	7.4.4 稀疏矩阵的解线性方程求解	175
5.8.1 傅里叶变换应用实例	110	7.4.5 稀疏矩阵的特征值与奇异值	181
5.8.2 傅里叶变换的幅度与相位	112	第 8 章 字符数组	184
5.8.3 快速傅里叶变换的长度与速度的 关系	113	8.1 字符数组	185
第 6 章 图形绘制	114	8.1.1 字符与数值的相互转换	185
6.1 基本绘图函数	114	8.1.2 创建二维字符数组	185
6.1.1 用 plot 函数绘图	114	8.2 字符串的比较	186
6.1.2 双 Y 轴图形	120	8.2.1 比较两个字符串	186
6.1.3 绘制三维线型图形	122	8.2.2 用运算符比较字符串	187
6.1.4 设置坐标轴的参数	124	8.2.3 字符串中字符的分类	188
6.1.5 图形窗口	129	8.3 字符串的查找与替换	189
6.2 图形标注	132	8.4 字符串与数值的转换	190
6.2.1 坐标轴标注	132	第 9 章 多维数组	193
6.2.2 在图形中添加文本标注和图例	133	9.1 多维数组的概念	193
6.3 专业图形	134	9.2 多维数组的创建	194
6.3.1 条形图与区域图	134	9.2.1 扩展二维数组创建多维数组	194
		9.2.2 用函数创建多维数组	195

9.3 多维数组的引用	197	11.4 数据类型	233
9.4 改变多维数组的形状	197	11.5 运算符	234
9.4.1 删除多维数组中“大小为 1 的维”	199	11.5.1 算术运算符	234
9.4.2 多维数组的转置	200	11.5.2 关系运算符	235
9.5 多维数组的计算	200	11.5.3 逻辑运算符	235
9.5.1 处理矢量的函数	200	11.5.4 运算符的优先级	237
9.5.2 逐个处理数组元素的函数	201	11.6 流程控制	238
9.5.3 处理位面或矩阵的函数	203	11.6.1 if、else 与 elseif 语句	239
9.6 多维数据的组织	204	11.6.2 switch 语句	240
第 10 章 结构体与单元数组	206	11.6.3 while 语句	242
10.1 结构体	206	11.6.4 for 语句	242
10.1.1 用赋值方式创建结构体	206	11.6.5 break 语句	243
10.1.2 用函数 struct 创建结构体	207	11.6.6 try ... catch 语句	243
10.1.3 访问结构体中的数据	208	11.6.7 return 语句	244
10.1.4 添加或删除结构体中的字段	210	11.7 子函数	245
10.1.5 处理结构体的数据	210	11.8 私有函数	246
10.1.6 组织结构体的数据	211	11.9 数组的下标引用	247
10.1.7 结构体的嵌套	213	11.9.1 删除矩阵中的元素	248
10.2 单元数组	214	11.9.2 高级下标引用	249
10.2.1 创建单元数组	215	11.10 字符串求值	250
10.2.2 获取单元数组的数据	217	11.11 错误与警告	251
10.2.3 删除单元数组中的单元	218	11.11.1 用函数 eval 和 lasterr 处理错误 ...	251
10.2.4 改变单元数组的形状	219	11.11.2 显示错误和警告信息	253
10.2.5 用单元数组作为变量	219	11.12 时间与日期	253
10.2.6 嵌套单元数组	221	11.12.1 日期格式	254
10.2.7 含有结构体的单元数组	222	11.12.2 日期字符串输出格式	256
10.2.8 单元数组与数值数组之间的 转换	223	11.12.3 当前日期与时间	257
第 11 章 MATLAB 编程	224	11.12.4 时间与日期工具函数	257
11.1 脚本程序	225	11.12.5 计时器	258
11.2 函数程序	226	11.13 获取用户输入	260
11.2.1 函数程序的结构	226	11.14 优化程序代码	261
11.2.2 函数命名	228	11.14.1 矢量化程序代码	261
11.2.3 函数的工作过程	228	11.14.2 为数组预分配存储空间	262
11.2.4 检查函数的参数个数	229	11.14.3 内存管理	263
11.2.5 参数中变量传递的个数	230	第 12 章 MATLAB 的类与对象	264
11.3 局部变量与全局变量	232	12.1 类与对象的概念	264
		12.1.1 面向对象编程的特性	264
		12.1.2 创建对象	264

12.1.3	调用类的方法	265	12.3.3	类 <code>polynom</code> 的 <code>display</code> 方法	277
12.1.4	私有方法	265	12.3.4	类 <code>polynom</code> 的 <code>subsref</code> 方法	277
12.1.5	调试类的方法	266	12.3.5	重载算术运算符	278
12.1.6	设置类的目录	266	12.3.6	重载函数	281
12.1.7	数据结构	267	12.4	类的继承与聚合	284
12.2	设计用户的 <code>MATLAB</code> 类	267	12.4.1	继承与聚合简述	284
12.2.1	<code>MATLAB</code> 类的标准方法	267	12.4.2	创建类 <code>asset</code> 及其子类	284
12.2.2	构造方法	268	12.4.3	设计类 <code>stock</code>	291
12.2.3	<code>display</code> 方法	269	12.4.4	创建类 <code>portfolio</code>	299
12.2.4	访问对象数据	269	12.5	对象的存取	303
12.2.5	定义对象的 <code>end</code> 索引方法	272	12.5.1	定义类 <code>portfolio</code> 的 <code>saveobj</code> 方法 ...	303
12.2.6	<code>subsindex</code> 方法	272	12.5.2	定义类 <code>portfolio</code> 的 <code>loadobj</code> 方法	304
12.3	创建一个多项式类	273	12.6	对象的优先级及方法调用的 次序	305
12.3.1	类 <code>polynom</code> 的构造方法	273			
12.3.2	类 <code>polynom</code> 的两个转换方法	274			

第 1 章 MATLAB 6.0 概述

1.1 MATLAB 简介

MATLAB 是 Math Works 公司的产品。Math Works 公司在 1982 年发行了 MATLAB 的 DOS 版本,不断完善至今形成了 3 种主要的版本,分别是 Windows 95 版、Windows NT 版和 Macintosh 版。MATLAB 6.0 是一种交互式、面向对象的程序设计语言,广泛应用于工业界与学术界,主要用于矩阵运算,同时在数值分析、自动控制模拟、数字信号处理、动态分析、绘图等方面也具有强大的功能。

MATLAB 6.0 程序设计语言结构完整,且具有优良的移植性,它的基本数据元素是不需要定义的数组。它可以高效率地解决工业计算问题,特别是关于矩阵和矢量的计算。MATLAB 与 C 语言和 Fortran 语言相比,更容易被掌握。寥寥几行 MATLAB 程序就可以实现你意想不到的功能。

另外, MATLAB 6.0 提供了一种特殊的工具:工具箱 (Toolboxes)。这种工具箱主要包括:信号处理 (Signal Processing)、控制系统 (Control Systems)、神经网络 (Neural Networks)、模糊逻辑 (Fuzzy Logic)、绘图 (Graph)、地图 (Mapping)、Image Processing (图像处理)、小波 (Wavelets) 和模拟 (Simulation) 等功能。Math Works 公司和各个领域的专家还在积极地完善现有工具箱的功能,并不断推出新的工具箱。不同领域、不同层次的用户通过对相应工具的学习和应用,可以方便地进行计算、分析及设计工作。

总的来说, MATLAB 系统包括以下 5 个主要部分:

(1) MATLAB 编程语言,对应于工具箱中的 6 个目录,分别为 ops (操作符与特殊字符)、lang (编程语言结构)、strfun (字符串操作)、iofun (文件的输入、输出)、timefun (时间和日期) 和 datatypes (数据类型和结构)。

(2) MATLAB 的工作环境,提供编程和调试的环境,管理工作空间中的变量及输入、输出的数据。工作环境的特性对应于 general 目录。

(3) MATLAB 的图形处理,包括高级的二维、三维图形可视化的处理工具,也包括低级的命令用于定义图形外观,并可为 MATLAB 应用程序,创建完整的图形用户界面 (GUI)。图形处理功能组织在 MATLAB 工具箱中的以下 5 个目录中: graph2d (二维图形功能)、graph3d (三维图形功能)、specgraph (专业图形)、graphics (图形处理) 和 uitools (图形用户界面工具)。

(4) 数学函数库,包括初等数学函数,如求和、正弦、余弦及其算法;也包括更复杂的功能,如矩阵求逆、求矩阵的特征值、快速傅里叶变换等等。数学函数库分别组织在以下 8 个 MATLAB 工具箱中: elmat (初等矩阵及其运算)、elfun (初等数学函数)、specfun (专业数学函数)、matfun (矩阵函数、线性代数)、datafun (数值分析和傅里叶变换)、polyfun (多项式和插值)、funfun (函数功能及解微分方程) 和 sparsfun (稀疏矩阵)。

(5) MATLAB 应用程序接口 (API), 提供接口程序使程序员可以编写 C/C++、Fortran 程序调用 MATLAB 作为计算引擎,并读写 MATLAB 文件;或者用 MATLAB 调用应用程序。

MATLAB的6.0版是现在的最新版本，在这个版本中增加了许多新的功能，其中包括：

- 增强了工具箱的功能及增加了工具箱的种类。
- 增加了许多功能函数。
- 增强了对多维矩阵的运算功能。
- 增强了用户界面的交互性。
- 扩充了矢量与矩阵的类型。
- 增加了微分方程的解法。
- 扩充了绘图功能。
- 在SIMULINK中可以拥有自己的编辑器与调试器。

1.2 MATLAB 的安装

MATLAB 对硬件的需求虽然不高，但要想使 MATLAB 的运行速度适中的话，至少需要下列配置：

- CUP Pentium II 250 MHz以上
- 显示卡 至少能支持256色
- 显示器 至少能支持256色，分辨率800×600 pixels
- 内存 32MB
- 操作系统 Windows 95/98/2000/NT

MATLAB 的安装过程比较简单，以在 Windows 98/2000 下安装 MATLAB 6.0 为例，其过程如下。

(1) 放入 MATLAB 的安装盘，启动 setup 文件，显示如图 1-1 所示的“欢迎安装 MATLAB”对话框。在此对话框中建议用户关闭所有其他正在运行的程序，并声明版权。

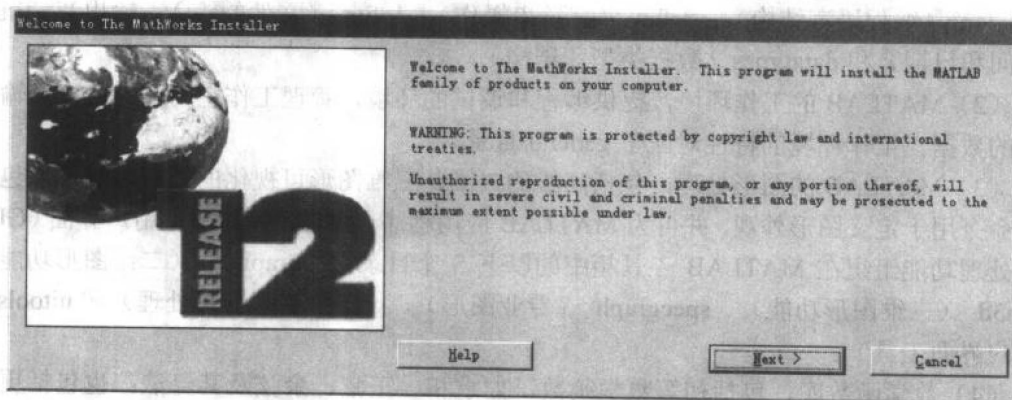


图 1-1 “欢迎安装 MATLAB”对话框

(2) 在如图 1-2 所示的“输入软件协议密码”对话框中，填写用户信息和使用许可码。单击 Next 按钮进行下一步。

(3) 单击 Next 按钮，显示如图 1-3 所示的“软件许可协议”对话框，单击 Yes 按钮接受此协议，或者单击 No 按钮不接受此协议，开始下一步。

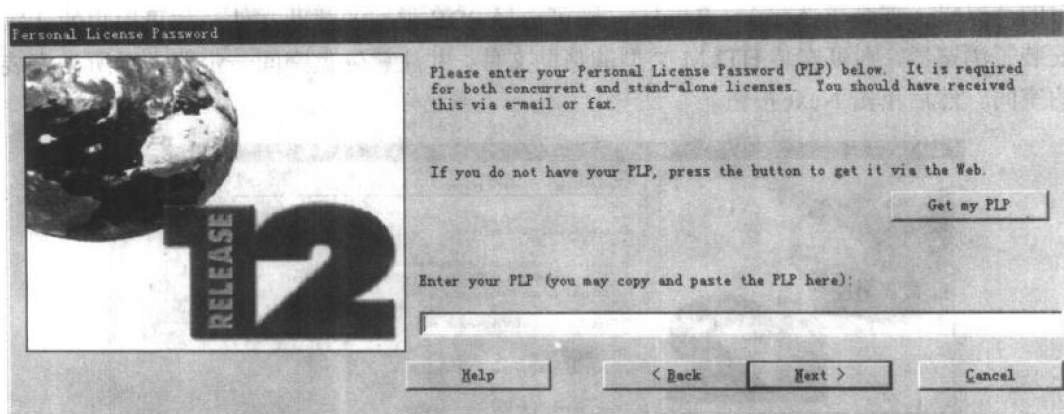


图 1-2 “输入软件协议密码”对话框

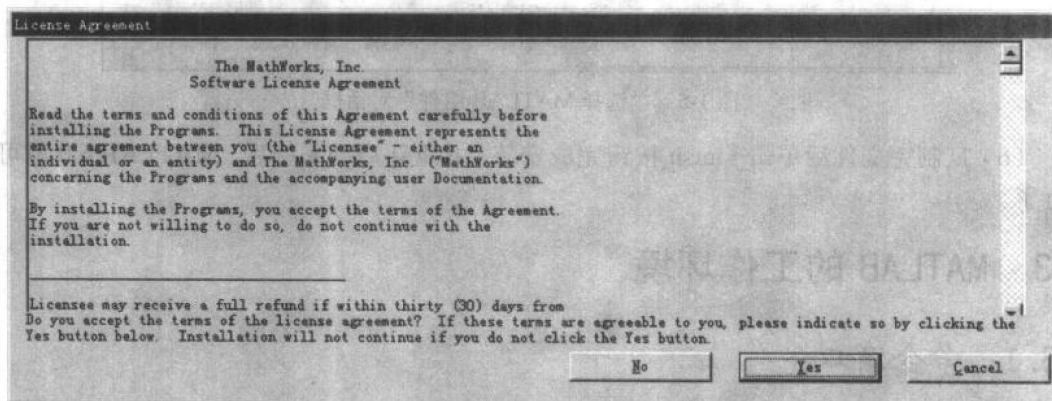


图 1-3 “软件许可协议”对话框

(4) 系统弹出如图 1-4 所示的对话框，此时用户可以填写自己的个人信息。

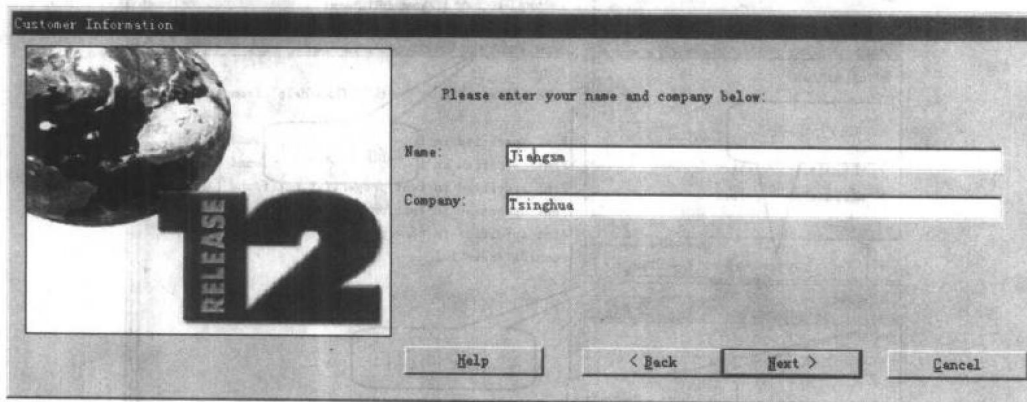


图 1-4 “输入用户信息”对话框

(5) 在如图 1-5 所示的“选择 MATLAB 组件”对话框中，选择要安装的组件，其中有各种可选的工具箱及其帮助文件（PDF 或 HTML），根据需要选择所要安装的工具箱。如果

用户的计算机上安装了 Acrobat Reader, 则可选择 PDF 类型的帮助文件; 如果用户的计算机上安装了浏览器, 则可安装 HTML 类型的帮助文件。并注意硬盘空间一定要大于所需安装文件的空。然后单击 Next 按钮。

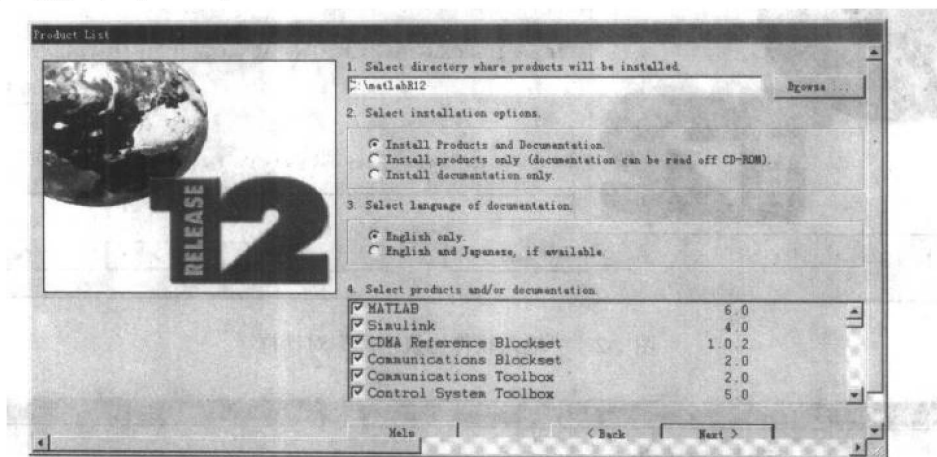


图 1-5 “选择 MATLAB 组件”对话框

(6) 复制完文件后单击 Finish 按钮完成安装, 并重新启动系统后, MATLAB 6.0 就可以运行了。

1.3 MATLAB 的工作环境

1.3.1 命令窗口

启动 MATLAB 后, 显示的窗口如图 1-6 所示。

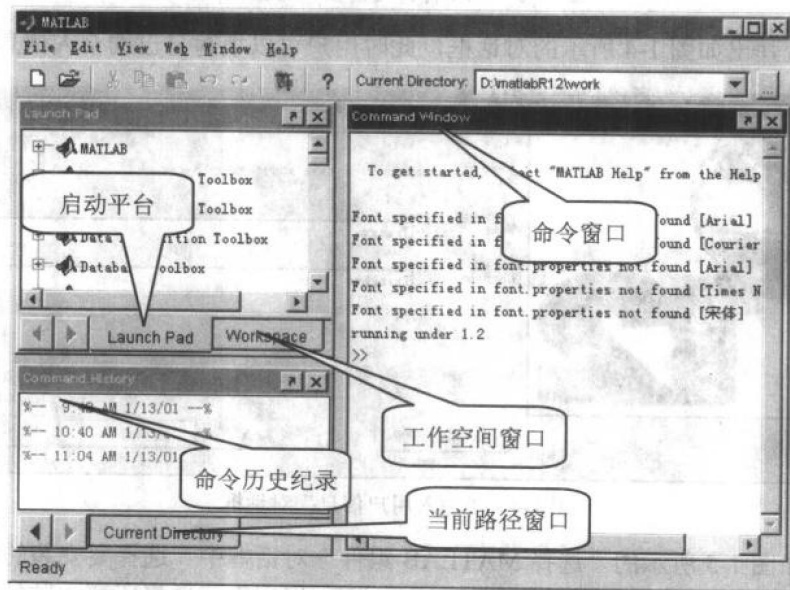


图 1-6 MATLAB 的命令窗口

Command Window (命令窗口) 是用来与 MATLAB 交互的主窗口, 用户可以将命令窗口放大, 使用鼠标点  按钮即可独立打开命令窗口, 如图 1-7 所示。在命令窗口中输入指令后, 系统自动地反馈信息。例如在命令窗口中输入指令:

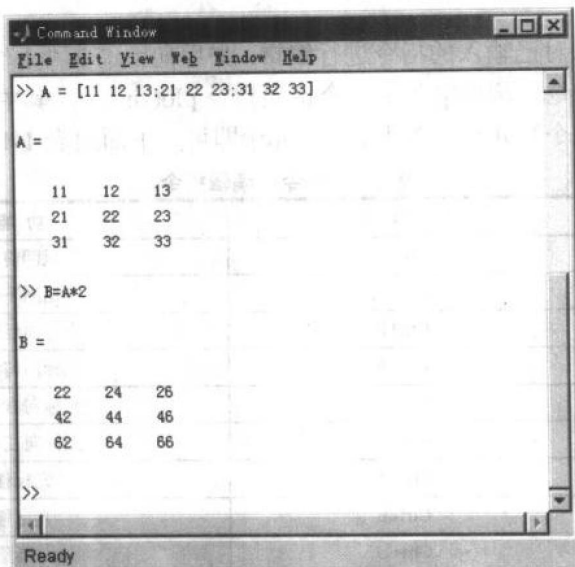


图 1-7 命令窗口

```
A = [11 12 13; 21 22 23; 31 32 33]
```

系统解释此指令为输入一个 3×3 数组的值, 并显示如下结果:

```
A =  
    11    12    13  
    21    22    23  
    31    32    33
```

如继续输入指令: $B=A*2$, 并按回车键执行, 则显示如下结果:

```
B =  
    22    24    26  
    42    44    46  
    62    64    66
```

如果要多行指令一次运行, 而不是逐行地执行指令, 可以在指令后加上 “;” 如上例输入数组 A, 并把数组 A 乘以 2, 将其值赋给数组 B。可以使用以下指令:

```
A = [11 12 13; 21 22 23; 31 32 33];
```

此时回车换行不显示数组 A 的赋值结果。

```
B=A*2
```

此行不加 “;”, 按回车键则执行这两行指令, 并显示结果。

如果因一个指令很长或其他原因, 需要跨行输入, 则使用 “...” 作连接符号, 按回车键转到下一行继续输入指令。如下面的指令:

```
1+2+...
```

3+...

5 其中“...”表示紧接下一行输入的指令。

各符号间可以加上空格,以增加指令的可读性,同时应注意一个指令最多可以包括 4096 个字符。

在命令窗口中可以对已输入的指令进行编辑,如用向上箭头“↑”可以回到上一句指令,用“↓”显示下一句指令。例如输入了一个错误指令 `plo(sin(a))`,其中 `plo` 应为 `plot`,只要用“↑”回到这个错误指令行并将其改为 `plot(sin(a))`即可。下面用表 1-1 列出命令行编辑指令。

表 1-1 命令行编辑指令

功 能 键	快 捷 键	功 能 说 明
↑	Ctrl+P	返回前一行输入
↓	Ctrl+N	返回下一行输入
←	Ctrl+B	光标向后移一个字符
→	Ctrl+F	光标向前移一个字符
Ctrl- →	Ctrl+R	光标向右移一个字符
Ctrl- ←	Ctrl+L	光标向左移一个字符
Home	Ctrl+A	光标移到行首
End	Ctrl+E	光标移到行尾
Esc	Ctrl+U	清除一行
Del	Ctrl+D	删除光标所在的字符
Backspace	Ctrl+H	删除光标前一个字符
	Ctrl+K	删除到行尾
	Ctrl+C	中断正在执行的指令

清除命令窗口显示的内容可以用命令 `clc`,但是此命令并不清除工作空间,仅清除窗口显示,用向上箭头键“↑”仍可以返回前一行输入的指令。

可以用 `format` 命令来控制数值显示的格式,或者打开 File 菜单的 Preferences 选项,选择如图 1-8 所示的 Command Window 操作标签,可以在 Numeric format 选择框选择所需的数值显示格式。下面的表 1-2 列出了数值显示的格式及其范例。

表 1-2 数值显示格式及其范例

显 示 格 式	范 例 1 (7/6)	范 例 2 (1.23334422e-5)
Short	1.1667	0.0000
Short E	1.1667e+000	1.2333e-005
Shor G	1.1667	1.2333e-005
Long	1.1666666666667	0.00001233344220
Long E	1.16666666666667e+000	1.233344220000000e-005
Long G	1.1666666666667	1.23334422e-005
Bank	1.17	0.00
Rat	7/6	3/243241
Hex	3ff2aaaaaaaab	3ee9dd7763803b29

上述显示格式都可以在命令窗口中直接输入,如 `format long e`。另外命令 `format loose` 与 `format compact` 用于控制显示的数值之间是否换行。在如图 1-7 所示的标签页中如果设置显示

格式为 `plus`, 则仅显示数值的符号, 正数显示为 “+”, 负数显示为 “-”。

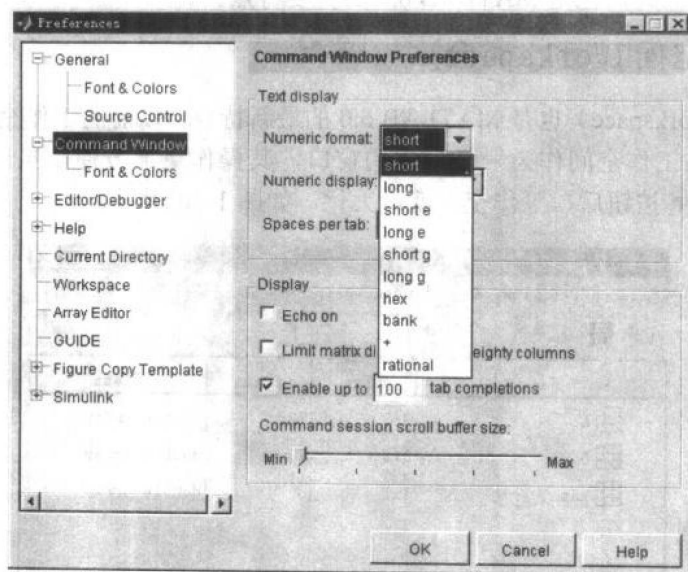


图 1-8 Preferences 对话框

1.3.2 启动平台

当用户需要进行特定的工作时, 需要启动某个工具箱的应用程序, 可以在 **Launch Pad** (启动平台) 中实现, 当用户使用鼠标单击 **Launch Pad** 窗口的  按钮后, 启动平台就最大化, 如图 1-9 所示。此时用户就可以很方便地从事自己的工作, 比如启动 **Control System Toolbox** (控制系统工具箱) 的 **Demos**, 用户可以使用鼠标双击该项目即可。



图 1-9 启动平台

通过启动平台,用户可以打开各个工具箱的帮助、Demos 和其他相关的文件或应用程序,这是一个非常好的工具,也是 MATLAB 6.0 的重要特点。

1.3.3 工作空间 (Workspace)

工作空间 (Workspace) 也是 MATLAB 6.0 的最新特点,以前的工作空间是一个对话框,可操作性差,现在工作空间作为一个独立的窗口,其操作更加方便。当用户使用鼠标单击 Workspace 窗口的  按钮后,工作空间就最大化,如图 1-10 所示。

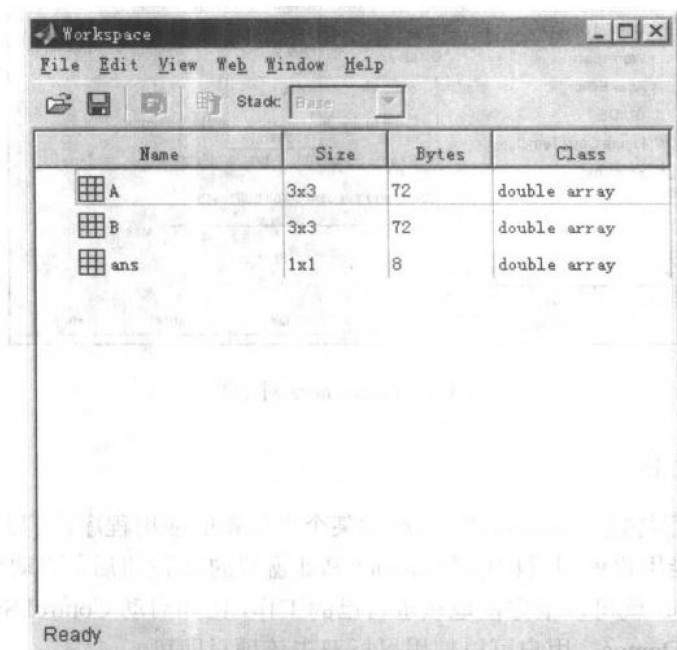


图 1-10 工作空间窗口

1.3.4 命令历史纪录和当前路径窗口

命令历史 (Command History) 纪录窗口主要显示已执行过的命令。当前路径窗口主要显示的是当前工作在什么路径下进行,包括文件的保存等均是在当前路径下实现的。用户也可以执行 File/Set path 命令设置当前路径。

1.4 M 文件的编辑调试环境

MATLAB 的程序文件和脚本文件通常保存为后缀为 “.m” 的文件,本书就称之为 M 文件。编辑 M 文件当然可以用其他的文本编辑器,要启动 MATLAB 的 M 文件编辑器和调试器,可在命令窗口中输入 Edit 命令,也可以选择 File 菜单中的 New/M-file 选项,或者单击图标按钮 。编辑器和调试器如图 1-11 所示。