

IYU MATLAB DE YUNDONGXUE
DONGLIXUE GUOCHENG
FENXI YU MONI

基于MATLAB的 运动学、动力学过程分析与模拟

敖文刚 杜力勤 著
黄勇刚 李勤



科学出版社

基于 MATLAB 的 运动学、动力学过程分析与模拟

敖文刚 杜力 黄勇刚 李勤 著



科学出版社

北 京

内 容 简 介

本书系统地介绍了运用MATLAB对运动学、动力学问题进行分析与模拟的基本理论、方法和过程,主要包括点的运动过程分析与模拟、点合成运动过程分析与模拟、刚体平面运动过程分析与模拟、运动学综合运用过程分析与模拟、动力学三大定理进行过程分析与模拟、运用拉格朗日方程进行过程分析与模拟。本书首先基于图形用户界面的形式对力学问题进行了分析,从传统的瞬态分析转变到建立过程分析模型(运动微分方程组),然后对运动变量方程组进行数值求解,最后将求解结果的数据进行可视化、条件查询等。

本书适合力学、土木、机械等工科专业的本科生、研究生,以及对运动学、动力学过程分析感兴趣的读者阅读。

图书在版编目(CIP)数据

基于 MATLAB 的运动学、动力学过程分析与模拟/敖文刚等著.—北京:科学出版社,2013.7

ISBN 978-7-03-038025-8

I. ①基… II. ②敖… III. ①动力学分析—Matlab 软件 IV. ①0655.9

中国版本图书馆 CIP 数据核字 (2013) 第 136000 号

责任编辑:荣洁莉 莫永国/封面设计:墨创文化

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>

成都创新包装印刷厂印制

科学出版社发行 各地新华书店经销

*

2013年6月第 一 版 开本:B5(720×1000)

2013年6月第一次印刷 印张:13.25

字数:200千字

定价:42.00元

前 言

运动学、动力学的很多问题都是以微分方程形式给出的,运用传统理论力学求解方法,一般只能对一些特殊位置作瞬态分析,而对整个运动过程中的各运动变量、约束力、所需动力或动力矩的变化规律的分析是不够理想的。本书通过 MATLAB 图形界面,利用 MATLAB 强大的数值计算能力和图形处理技术,对运动学、动力学问题进行过程分析,并将结果以曲线、动画等形式直观表现出来,加强读者对力学问题的感性认识。运用图形用户界面分析力学问题有以下优点:①图形界面通过多个按钮的回调函数完成相应功能,将整个程序分解为几个部分,使其程序编写、调试更容易;②容易实现人机交互,通过观察不同运动参数和初始条件下研究对象的运动变化,分析运动参数和初始条件对运动的影响;③可得到某时段内每一时刻的运动变量,而不仅仅是某一瞬态的结果,实现对整个运动的过程分析;④实现了结果数据的可视化,通过曲线、动画等形式,使我们对较为复杂的运动有更为直观、清晰的认识。

本书研究力学问题的思路主要有两个方面:①从传统的瞬态分析转变到建立过程分析模型(运动微分方程组)。这个过程就是要先将位移(转角)、速度(角速度)等设定为运动变量,并确定这些运动变量的微分方程组和初值,最后求出所有时刻位移(转角)、速度(角速度)、加速度(角加速度)和约束力、所需动力或动力矩的取值。②在 MATLAB 的图形用户界面上进行可视化编程,通过 ODE 函数对运动变量方程组进行数值求解,运用 MATLAB 的作图功能将数值求解的结果以曲线、动画等可视化的形式表现,也可以将数据保存为 Excel 文件或对某时刻位移(转角)、速度(角速度)、加速度(角加速度)和约束力、所需动力或动力矩的结果进行查询。第一方面是核心,定义求解微分方程组的函数文件与传统力学原理、方法内容结合紧密,实现了从瞬态分析到过程分析的自然过渡。在运用传统力学原理、方法建立过程分析模型时要注意两点:①将位移(转角)、速度(角速度)、加速度(角加速度)和约束力、所需动力或动力矩设为变量;②规定这些变量的正方向,并注意在力学定理、公式中正确运用。

本书共分 7 章,系统地介绍了如何运用图形用户界面对各类运动学、动力学问题进行过程分析。第 1 章介绍了本书可能会涉及的相关 MATLAB 知识,并通过一个简单实例来介绍运用图形用户界面对力学问题进行过程分析的全过程;第 2~5 章对点的运动学、点的复合运动、刚体平面运动和运动学综合运用等方面的问题分别运用图形用户界面进行了过程分析;第 6、7 章对应用动力学三大定理、拉格朗日方程的动力学问题分别运用图形用户界面进行了过程分析。本书运用图

形用户界面对多个机构进行了运动学、动力学方面的过程分析，特别适合机械类专业的同学和工程技术人员参考、学习。

目 录

第 1 章	MATLAB 简介.....	1
1.1	关于 MATLAB 的介绍	1
1.2	MATLAB 中的矩阵和向量及其操作.....	6
1.3	MATLAB 中常微分方程的求解.....	11
1.4	MATLAB 图形用户界面基础	13
1.5	MATLAB 编程基础 ^[1]	18
1.6	运用图形用户界面求解运动学问题实例	24
第 2 章	点的运动学分析	33
2.1	滚动圆轮边缘点运动分析与模拟 ^[4]	33
2.2	螺线运动分析与模拟 ^[4]	44
2.3	以曲柄为原动件的曲柄摇块机构连杆上点的运动分析和模拟 ^[4]	51
第 3 章	点的合成运动分析与模拟	59
3.1	曲柄导杆机构运动分析与模拟 ^[4]	59
3.2	双滑块机构运动分析与模拟	66
3.3	牛头刨床机构运动分析与模拟 1	73
第 4 章	刚体平面运动分析	85
4.1	曲柄滑块机构运动分析与模拟 ^[4]	85
4.2	等边三角板运动分析与模拟 ^[4]	93
第 5 章	运动学综合运用分析.....	104
5.1	轻型杆式推钢机运动分析与模拟	104
5.2	放大机构运动分析与模拟	113
5.3	刨床机构运动分析与模拟	123
5.4	曲柄连杆双滑块机构运动分析与模拟	131
第 6 章	动力学三大定理综合应用	143
6.1	均质杆运动动力学分析与模拟 ^[5]	143
6.2	带可滑动质量块单摆(摆球)动力学分析与模拟 ^[4]	150
6.3	均质杆由竖直倒下的分析与模拟 ^[4]	156
6.4	双滑块机构动力学分析与模拟 ^[4]	165
6.5	弹簧木块上小球跳动的分析与模拟 ^[4]	173
第 7 章	运用拉格朗日方程分析动力学问题	182

7.1 双摆运动的分析与模拟 ^[4]	182
7.2 圆盘中心铰接均质杆动力学分析与模拟 ^[5]	191
7.3 双自由度弹簧——质量块的分析与模拟 ^[5]	198
参考文献	206

第 1 章 MATLAB 简介

1.1 关于 MATLAB 的介绍

MATLAB 是由美国 MathWorks 公司发布的，主要应用于科学计算可视化以及交互式程序设计的高科技计算环境。它将数值分析、矩阵计算、科学数据可视化以及非线性动态系统的建模和仿真等诸多强大功能集成在一个易于使用的视窗环境中，为科学研究、工程设计以及必须进行有效数值计算的众多科学领域提供了一种全面的解决方案。

总的来说，MATLAB 有如下 4 个特点。

(1) 语言简洁紧凑，使用方便灵活，库函数极其丰富。有高性能数值计算的高级算法，特别适合矩阵代数领域；MATLAB 程序书写形式自由，可以利用其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。可以说，使用 MATLAB 进行科技开发是站在专家的肩膀上进行开发。

(2) 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。

(3) 功能强大的工具箱。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。其工具箱又分为两类：功能性工具箱和学科性工具箱。功能性工具箱主要用来扩充其符号计算功能、图示建模仿真功能、文字处理功能以及与硬件实时交互功能，可用于多种学科；学科性工具箱是专业性比较强的，如(control,signal processing,communnication)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，用户无需编写自己学科范围内的基础程序，而可直接进行高精尖的研究。

(4) 高级但简单的程序环境，具备与其他语言编写的程序结合和输入输出格式化数据的能力；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。

本书选用 MATLAB 来分析理论力学中运动学、动力学问题，就是在利用其强大的数值计算和数据可视化功能。

1.1.1 MATLAB 的工作环境

安装 MATLAB 7.0 系统，需运行系统自带的安装程序 setup.exe。运行后，将进入 MATLAB 安装程序的欢迎对话框，用户可以按照安装提示依次操作。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, 打开 MATLAB 的主界面。

1. 主窗口

MATLAB 主窗口是 MATLAB 的主要工作界面, 如图 1.1 所示。主窗口除了嵌入一些子窗口外, 还包括菜单栏和工具栏。在 MATLAB 7.0 主窗口的菜单栏, 包含 File, Edit, Debug, Desktop, Window 和 Help 共 6 个菜单项。MATLAB 7.0 主窗口的工具栏共提供了 12 个命令按钮和一个当前路径列表框。

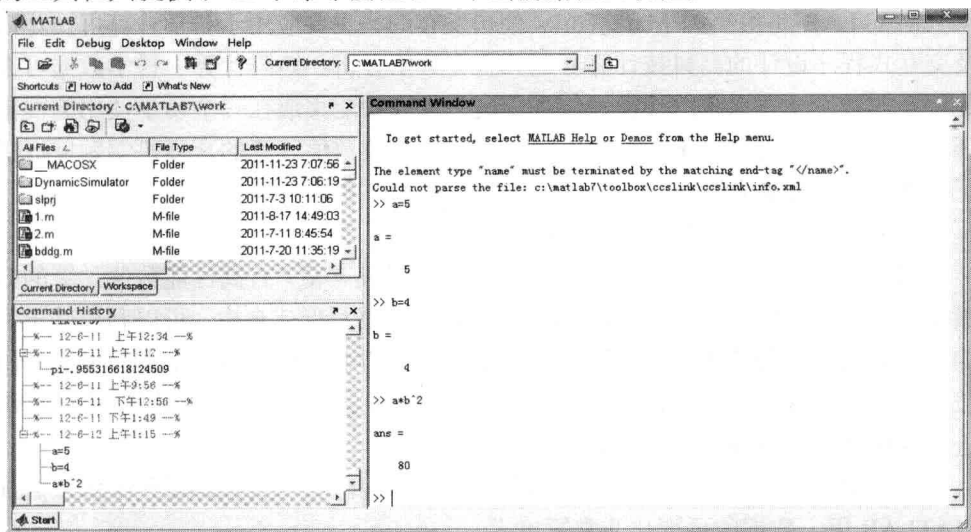


图 1.1 MATLAB 主窗口

在默认情况下, MATLAB 的操作界面包含了三个最常见的子窗口: 命令窗口 (Command Window)、工作空间浏览器 (Workspace) 和历史命令窗口 (Command History)。当前工作目录 (Current Directory) 的选项卡是隐藏的, 窗口的左下角为“开始”按钮。

2. 主界面各分部功能介绍

(1) 命令窗口: 是 MATLAB 最主要的交互窗口, 用于输入命令并显示除图形以外的所有执行结果。MATLAB 命令窗口中的“>>”为命令提示符, 表示 MATLAB 正处于准备状态。在命令提示符后键入命令并按下回车键后, MATLAB 就会解释执行所输入的命令, 并在命令后面给出计算结果。

(2) 工作空间浏览器: 是 MATLAB 用于存储各种变量和结果的内存空间。在该窗口中显示工作空间中所有变量的名称、大小、字节数和变量类型说明, 可对变量进行观察、编辑、保存和删除。

(3) 当前工作目录: 指 MATLAB 运行文件时的工作目录, 只有在当前目录或搜索路径下的文件、函数可以被运行或调用。在当前目录窗口中可以显示或改变

当前目录，还可以显示当前目录下的文件并提供搜索功能。将用户目录设置成当前目录也可使用 `cd` 命令。例如，将用户目录 `c:\mydir` 设置为当前目录，可在命令窗口输入命令：`cd c:\mydir`。

(4)历史命令窗口：在默认设置下，历史记录窗口中会自动保留自安装起所有用过的命令的历史记录，并且还标明了使用时间，从而方便用户查询。而且，通过双击命令可进行历史命令的再运行。如果要清除这些历史记录，可以选择 Edit 菜单中的 Clear Command History 命令。

(5)“开始”按钮：MATLAB 7.0 的主窗口左下角有一个 Start 按钮，单击该按钮会弹出一个菜单，选择其中的命令可以执行 MATLAB 产品的各种工具，并且可以查阅 MATLAB 包含的各种资源。

要退出 MATLAB 系统，也有三种常见方法：在 MATLAB 主窗口 File 菜单中选择 Exit MATLAB 命令；在 MATLAB 命令窗口输入 Exit 或 Quit 命令；单击 MATLAB 主窗口的“关闭”按钮。

1.1.2 MATLAB 的帮助系统

MATLAB 有数千条命令，初学者不可能也不必等到学完了之后才开始使用 MATLAB。通常只需对 MATLAB 略有了解便可使用它，边学边用，由浅入深，逐渐熟练。此阶段最容易遇到的问题是，不了解 MATLAB 有哪些命令，也找不到自己想要的命令，当然更不会使用这些命令。随着 MATLAB 的推广应用，MATLAB 的版本更新也不断加快，功能不断增加，但各版本的命令可能会有不同，随时掌握新版本的命令可以借助 MATLAB 的帮助系统实现。

1. 帮助命令

MATLAB 帮助命令包括 `help` 命令和 `lookfor` 命令。

(1) `help` 命令。在 MATLAB 命令窗口中直接输入 `help` 命令将会显示当前帮助系统中包含的所有项目，即搜索路径中所有的目录名称。同样，可以通过 `help` 加函数名来显示该函数的帮助说明。

(2) `lookfor` 命令。`help` 命令只搜索出那些与关键字完全匹配的结果，`lookfor` 命令对搜索范围内的 M 文件进行关键字搜索，条件比较宽松。`lookfor` 命令只对 M 文件的第一行进行关键字搜索。若在 `lookfor` 命令中加上 `-all` 选项，则可对 M 文件进行全文搜索。

2. 帮助窗口

进入帮助窗口可以通过以下三种方法：单击 MATLAB 主窗口工具栏中的 Help 按钮；在命令窗口中输入 `helpwin`、`helpdesk` 或 `doc`；选择 Help 菜单中的 MATLAB Help 选项。如图 1.2 所示，MATLAB 帮助窗口左窗格有 Content、Index、Search 和 Demos 4 个选项卡。

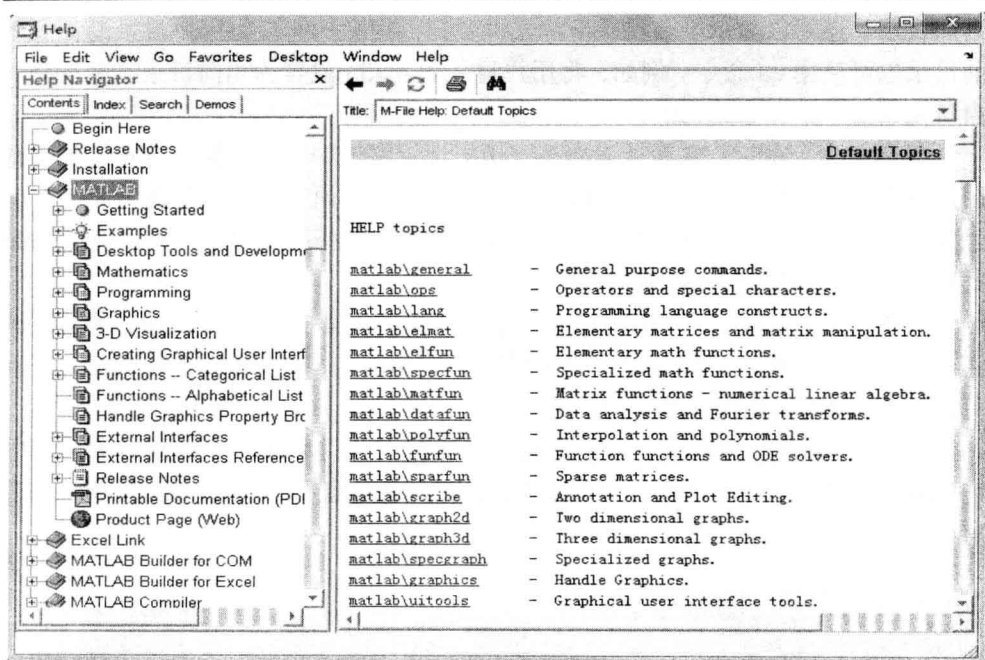


图 1.2 MATLAB 帮助窗口

3. 演示系统

在 MATLAB 中, 各个工具包都有设计好的演示程序, 这组演示程序在交互界面中运行, 操作非常简便。因此, 运行这组演示程序, 然后研究演示程序的相关 M 文件, 对 MATLAB 用户而言是十分有益的。这种演示功能对提高用户的 MATLAB 运用能力有着重要的作用。特别对于那些初学者而言, 不需要了解复杂的程序就可以直观地查看程序结果, 可以加强用户对 MATLAB 的掌握能力。在帮助窗口中选择 Demos 选项卡, 然后在其中选择相应的演示模块, 或者在命令窗口输入 Demos, 或者选择主窗口 Help 菜单中的 Demos 子菜单, 打开演示系统, 如图 1.3 所示。可以在对话框的左窗格中选择演示的内容, 例如选择 Graphics 选项, 在右窗格中会出现该项目下的各种类别的演示程序。单击对话框中的 3-D Surface Plots 选项, MATLAB 中会演示关于 3-D Surface Plots 演示程序的介绍, 然后单击对话框中的 Run this demo 选项, MATLAB 会打开 3-D Plots in Handle Graphics 对话框, 这就是演示 demo 的交互界面, 如图 1.4 所示。演示该 demo 的是一个交互界面, 可以调整该界面中下拉框的选项, 就可改变命令、显示参数, 这些修改会体现在 Command Window 框中, 而其效果在做图区显示。

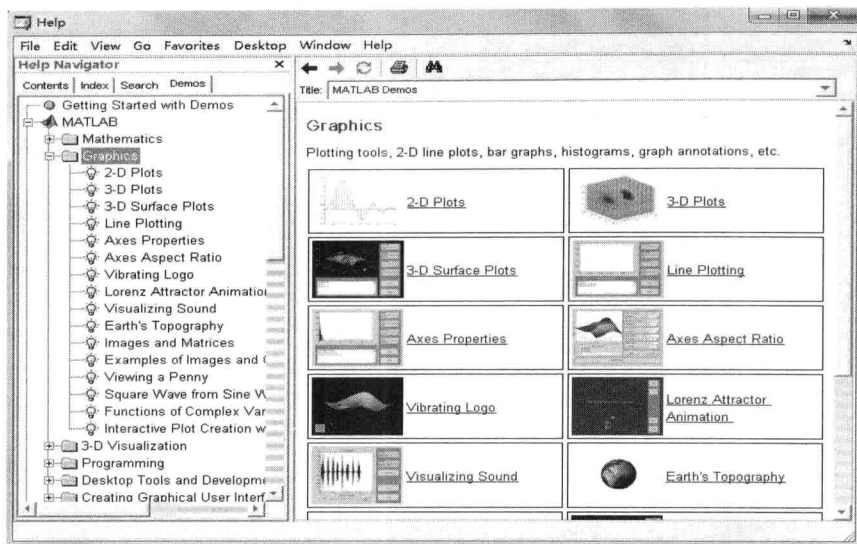


图 1.3 MATLAB 的 demo 帮助

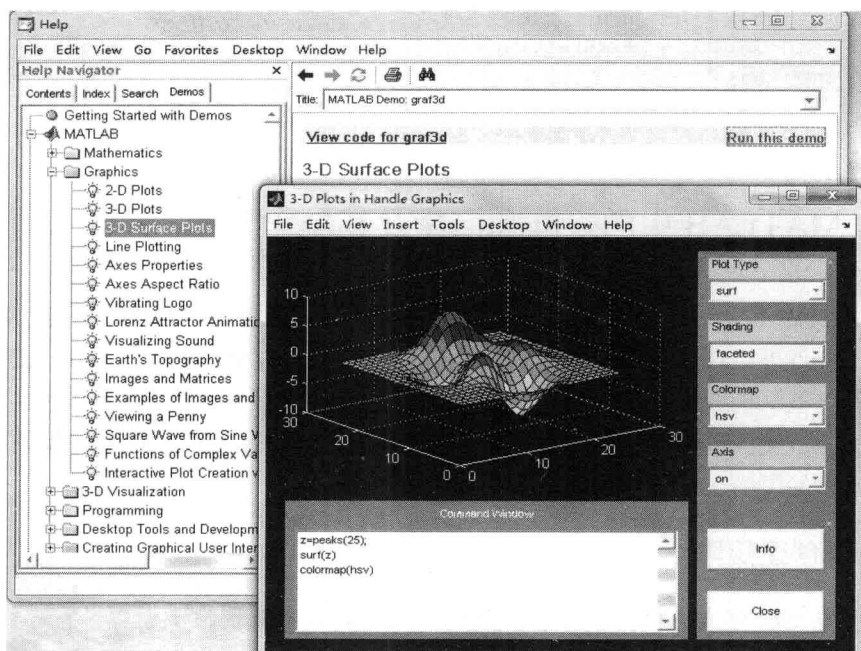


图 1.4 demo 交互界面

除了可以在打开的动态界面中演示 demo 外,还可以查看该 demo 的程序代码,单击 View code for graf3d 选项,查看该 GUI 界面的 MATLAB 程序代码,如图 1.5 所示。

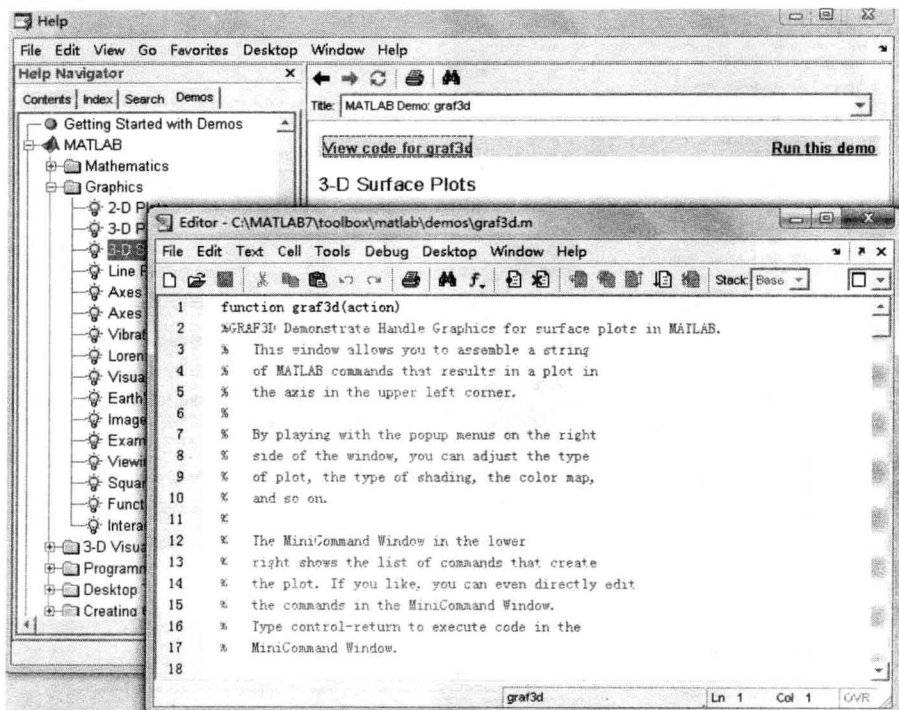


图 1.5 demo 程序代码

1.2 MATLAB 中的矩阵和向量及其操作

MATLAB 的基本数据结构为矩阵，其所有运算都是基于矩阵进行的。从形式上看，矩阵可以理解成二维的数组，矩阵可以方便地存储和访问 MATLAB 中众多数据类型，构成矩阵的元素可以是 MATLAB 中的任何数据类型。本章主要讲解矩阵和数组的基本结构及操作内容，包括矩阵的生成、矩阵的拼接、矩阵变形、矩阵元素的寻访、获取矩阵的信息。

1.2.1 创建矩阵

矩阵是所有 MATLAB 运算的基础，用户如果要想实现科学运算、程序设计、特性绘制等目标，必须要确定矩阵的类型，并建立矩阵。在 MATLAB 中创建一个矩阵有两种常用方法：一是直接输入矩阵元素，二是调用矩阵创建函数。

对于简单的矩阵，特别是元素数目不多的矩阵，逐个输入矩阵元素是最常用、最便捷的矩阵创建方法，须遵循以下三条原则：运用矩阵构造符“[]”包含所创建矩阵的所有元素；使用逗号“,”或者空格“ ”分隔矩阵的列；使用分号“;”或者回车键分隔矩阵的行。

如在 MATLAB 的命令窗口中输入下面的程序代码：

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

在命令窗口中输入变量 a,b，可以得到以下结果：

```
a =
     1     2     3
     4     5     6

b =
     7     8     9
    10    11    12
```

上面的例子演示了直接创建矩阵的方法。MATLAB 7.0 还提供了若干特殊矩阵的生成函数，在调用函数时，用户根据需要设置参数，就可以方便地得到需要的矩阵，常用的特殊矩阵函数见表 1.1。

表 1.1 常用的特殊矩阵函数

函 数	生成矩阵形式
ones	全 1 元素矩阵
zeros	全 0 元素矩阵
eye	单位矩阵，即主对角线元素为 1，其余元素全为 0
rand	均匀分布随机矩阵
randn	正态分布随机矩阵
magic	魔术矩阵
diag	对角矩阵
compan	伴随矩阵
gallery	测试矩阵
hadamard	Hadamard 矩阵
hilb	Hilbert 矩阵
invhilb	Hilbert 矩阵转置
rosser	经典对称特征值测试矩阵
toeplitz	Toeplitz 矩阵
pascal	Pascal 矩阵
vander	Vandermonde 矩阵
wilkinson	Wilkinson 特征值测试矩阵

举如下几例以说明矩阵函数的运用:

```
>> z=zeros(3,4)

z =

     0     0     0     0
     0     0     0     0
     0     0     0     0

>> x=ones(2,2)

x =

     1     1
     1     1

>> y=randn(3,3)

y =

    -0.4326    0.2877    1.1892
    -1.6656   -1.1465   -0.0376
     0.1253    1.1909    0.3273

>> a=[1,2,3];

>> A=diag(a)

A =

     1     0     0
     0     2     0
     0     0     3
```

1.2.2 拼接矩阵^[1]

矩阵的拼接是指两个或者两个以上的单个矩阵,按一定的方向进行连接,生成新的矩阵。从本质上说,矩阵的拼接就是一种创建矩阵的特殊方法,区别在于基础元素是原始矩阵,目标是新的合并矩阵。本节主要介绍矩阵拼接的两种方法:一是利用矩阵生成符“[]”,二是调用矩阵拼接函数。

通常,矩阵的拼接有水平方向和垂直方向两种。例如,对前述对角矩阵 A 进行水平方向拼接,拼接表达式和结果分别如下所示:

```
>> B=[A,A]

B =

     1     0     0     1     0     0
     0     2     0     0     2     0
     0     0     3     0     0     3
```

对前述对角矩阵 A 进行垂直方向拼接,拼接表达式和结果分别如下所示:

```
>> C=[A:A]
C =
     1     0     0
     0     2     0
     0     0     3
     1     0     0
     0     2     0
     0     0     3
```

1.2.3 向量、标量

向量是行数或列数为 1 的特殊矩阵，其一般显示为 $1 \times n$ 或 $n \times 1$ 的数列。用户在构造新矩阵，以及对矩阵进行访问、修改等操作时，常用到向量数列。MATLAB 提供了生成等差向量数列的符号——冒号，例如， $(p:q)$ 生成 $p \sim q$ ，差为 1 的递增向量数列。若要生成 $p \sim q$ ，差为 x 的递增等差数列，可用命令 $(p:x:q)$ 。例如，创建 1~10、1~19 的等差递增数列，在命令窗口分别输入代码及执行结果如下：

```
>> C=(1:10)
C =
     1     2     3     4     5     6     7     8     9    10
>> B=(1:2:19)
B =
     1     3     5     7     9    11    13    15    17    19
```

也可以用 `linspace` 函数生成向量，如在命令窗口输入代码及执行结果如下：

```
>> Y=linspace(1,19,10)
Y =
     1     3     5     7     9    11    13    15    17    19
```

标量是行列数都是 1 的特殊矩阵。任意以矩阵形式表示的单个实数或复数，称为标量。如下的实数 $x=3$ 就是一个标量，其维数为 2，视为一个一行一列的矩阵（总维数为 2）。

```
>> x=3
x =
     3
>> ndims(x)
ans =
     2
>> size(x)
```

```
ans =  
1
```

1.2.4 矩阵寻访^[1]

矩阵作为存储各种数据的基本单位，是若干相关元素的有序集合，为方便用户访问矩阵中的一个或者多个元素，MATLAB 引入了元素下标的概念。本节主要介绍不同的元素下标表示方法，以及如何利用矩阵的下标访问矩阵中的单元素和多元素。

1. 访问单元素

MATLAB 7.0 中，矩阵的下标表示与常用的数学习惯相同，使用“双下标”，即分别表示行和列，矩阵中的元素都有对应的“第几行，第几列”。这种表示方法简单直观，几何概念比较清晰。如矩阵元素 A_{ij} 表示为 $A(i,j)$ ，向量元素 B_k 表示为 $B(k)$ 。同时矩阵元素也可以用一个下标来表示，把矩阵看成是一个列向量，元素计数的顺序是第一列、第二列直到最后一列。如对前面的 3×3 矩阵 A ，矩阵元素 $A(2,2)$ 可以表示为 $A(5)$ ：

```
>> A(2,2)  
ans =  
2  
>> A(5)  
ans =  
2
```

2. 访问多元素

访问矩阵的多元素，包括访问某一行或某一列的若干元素，访问整行、整列元素，访问若干行或若干列的元素以及访问矩阵所有元素等。首先，介绍一下在访问矩阵多元素时常用的一个符号——冒号。前面章节介绍过，冒号可以表示等差步进，从而帮助生成向量矩阵；其实冒号还有另外一个重要的功能，即表示矩阵中的多个元素。例如 $A(1:k,n)$ 表示矩阵 A 中第 n 列的 $1 \sim k$ 行元素， $B(m,:)$ 表示矩阵 B 中第 m 行的所有元素， $B(:,m)$ 表示矩阵 B 中第 m 列的所有元素。在对矩阵中某部分元素进行访问和赋值时，此种表示方法显得十分方便。如对前面的 3×3 对角矩阵 A ，可使用如下方法访问多元素。

```
>> A(:,1)  
ans =  
1  
0  
0
```