

MATLAB

MATLAB

王济 胡晓 编著

在振动信号处理中的应用

中国水利水电出版社
www.waterpub.com.cn
知识产权出版社
www.cnipr.com



MATLAB 王济 胡晓 编著
在振动信号处理中的应用

中国水利水电出版社
www.waterpub.com.cn

知识产权出版社
www.cnipr.com



内容提要

本书主要讲述借助简单、高效、功能强大的 MATLAB 系统进行振动信号的处理分析。书中介绍了 MATLAB 的基本用法和一些相关的常用命令及函数,同时还叙述了振动信号分析的基本概念、常用处理方法以及 MATLAB 语言的编程技术。本书给出了大量的振动信号处理编程实例源代码,以帮助读者快速学习和掌握振动信号处理的方法和 MATLAB 的编程技术,以应用于实际的工作之中。

本书可作为工作中涉及振动测试和振动信号处理的各行业科研人员和工程技术人员,以及水利电力工程、土木建筑工程、航天、航空、航海、交通运输、机械、核电工业以及国防工业等相关专业的研究生或大学高年级学生的参考资料。

选题策划: 阳 淼 张宝林 E-mail: yangsanshui@vip.sina.com; z_baolin@263.net

责任编辑: 阳 淼 张宝林

文字编辑: 彭天放

图书在版编目 (CIP) 数据

MATLAB 在振动信号处理中的应用/王济, 胡晓编著.

北京: 中国水利水电出版社; 知识产权出版社, 2006

ISBN 7-5084-2680-0

I. M... II. ①王... ②胡... III. 振动—信号处理
—计算机辅助计算—软件包, MATLAB IV. TN911.7

中国版本图书馆 CIP 数据核字 (2005) 第 144727 号

MATLAB 在振动信号处理中的应用

王 济 胡 晓 编著

中国水利水电出版社 出版 发行 (北京市西城区三里河路 6 号; 电话: 010-68331835 68357319)
知识产权出版社 (北京市海淀区马甸南村 1 号; 传真、电话: 010-82000893)

全国各地新华书店和相关出版物销售网点经销

北京市兴怀印刷厂印刷

787mm×1092mm 16 开 16.5 印张 391 千字

2006 年 1 月第 1 版 2006 年 1 月第 1 次印刷

印数: 0001—3000 册

定价: 32.00 元

ISBN 7-5084-2680-0

版权所有 侵权必究

如有印装质量问题, 可寄中国水利水电出版社营销中心调换

(邮政编码 100044, 电子邮件: sales@waterpub.com.cn)

前 言

振动是一种普遍存在的自然现象。虽然人们经常利用振动来进行运输、加工和打桩，做一些有益的工作，但更多的是振动往往起着影响机器设备性能和寿命，破坏建筑物的作用。目前，解决振动问题主要有两种途径：一是通过计算进行理论分析；二是用试验手段进行测试和分析。对一些复杂的大型结构，单靠现有的振动理论及计算方法来进行分析是不够的。因为一般的理论计算是建立在一定的力学模型及数学模型基础之上的，而对于这些复杂结构，其力学模型及边界条件的简化往往建立在许多假设基础之上，只能是近似地与实际情况相符合，这是理论计算不可避免的一个缺点。因而用试验的方法来解决复杂结构振动问题是不可或缺的。

振动信号处理是对振动试验和振动测试所获得的数据进行的加工。在很多情况下，只有通过振动信号的加工处理，我们才能够从振动信号中得到我们需要掌握的信息，以便采取措施，解决振动问题或对振动问题做出正确的评价。

近 30 多年来，振动信号处理经历了从模拟信号的人工分析到数字信号的计算机处理的演变过程，振动信号的处理分析无论在理论基础还是在技术上都发展得很快。特别是在计算机技术迅猛发展的今天，价格低廉的普通个人电脑加一台作为电脑外设的数据采集仪不但完全取代了过去繁重的模拟信号人工测试分析工作，而且基本淘汰了价格昂贵的专用振动信号处理设备。

用普通个人电脑进行振动信号处理是通过软件来实现的。目前，市场上振动信号处理软件的种类不是很多，其中也包括一些进口软件。这些软件的价格都十分昂贵，尤其是进口软件的价格更是让人不敢问津。虽然这些软件的处理功能非常丰富，但它们都存在一个最大的通病，就是使用起来不够灵活方便，特别是需要进行大批量数据处理或测点通道排列不规则需要进行重排处理时显得尤为突出。通过自己编写程序来处理振动信号实际上是解决以上矛盾的最佳方案，而 MATLAB 正是非软件专业的科研人员和工程技术人员编程的最好的系统平台。

MATLAB 是美国 MathWorks 公司开发的大型科学计算应用软件系统，它提供了强大的数值计算、矩阵处理和绘图功能。MATLAB 系统简单易用，可信度高，灵活性好，目前已经成为国际市场上科学研究和工程应用方面的

主导软件。MATLAB 提供给用户的是一种最简洁、最直观的程序开发环境。用 MATLAB 编写程序犹如在演算纸上排列出公式与求解问题，所以 MATLAB 的编程语言又被称为演算纸式的科学算法语言。

MATLAB 系统提供的基本函数库包含大量的数值分析函数。借助这些基本函数，许多复杂问题的求解计算，只需要几条简单的调用语句就能完成求解计算程序的编制。另外，再增加一两条绘图命令，计算结果的图形也随之显示或绘制出来，真可谓是立竿见影。MATLAB 系统还提供了包括信号处理在内的一系列被称作工具箱的专业求解工具。通过这些针对不同学科、不同专业所开发的专用函数包，各个领域的数值计算问题都可以得到解决。这无疑是为振动信号处理程序的编制带来极大的方便。

本书主要介绍一些振动信号处理的方法和在 MATLAB 系统中的应用，结合所讲述的振动信号处理方法给出一些相应的 MATLAB 语言编程实例的源代码。这些源代码融入了作者本人的心得体会、编程经验和技巧，希望读者能从中受益。本书作者多年来一直从事振动试验和振动信号处理的工作，具有丰富的实践经验，并取得了一定的研究成果。由于早年对程序开发的兴趣爱好，多年来养成通过编制程序解决和处理工作事务以及实现自己构思的习惯。作者曾编制过许多诸如管理、图形处理、数据处理等多方面以及跟自己本职工作有关的应用程序，另外还参加过振动信号处理商用软件的开发工作，可以说在编程技术和技巧方面有一定的造诣。几年前，本书作者开始接触 MATLAB 系统，很快就被 MATLAB 强大的功能和简单的编程方式所吸引。随着对 MATLAB 系统了解的深入，越发感到要实现高效率的编程，MATLAB 是最佳选择，特别是对于编程不太熟悉的科技人员尤其如此。

本书的写作宗旨是希望通过阅读这本书使工作中涉及振动测试和振动信号处理的广大科研人员和工程技术人员以及有关专业的大学生、研究生能在较短的时间内掌握振动信号处理的基本方法和 MATLAB 语言的编程技术，实现自己动手编程，方便、灵活、高效地完成振动试验或振动测试数据的处理分析工作。熟练掌握振动信号处理 MATLAB 语言编程技术的最有效的途径是，一要勤于动手编程，只有通过动手编程才能掌握 MATLAB 语言编程技术；二要善于从 MATLAB 系统的数值分析函数库（包括各种工具箱）中找出解决问题最合适的函数或几个函数的组合，这往往会起到事半功倍的效果。

本书共分为 10 章。第 1 章介绍 MATLAB 的基本知识、语法规则、常用命令和函数以及信号处理工具箱的一些函数。第 2 章讲述振动的概念和一些基本理论以及振动信号处理的基本内容。第 3 章简单介绍振动测试，包括振动测试仪器、数据采集系统、激振设备、振动测试方法等。第 4 章介绍一些振动信

号预处理方法。第5章讲述振动信号时域处理方法。第6章讲述振动信号频域处理方法。第7章讲述包括人工模拟地震波在内的一些振动试验常用激振信号生成的方法。第8章介绍几种常用模态参数的频域识别方法,其中包括导纳圆拟合法、最小二乘迭代法、加权最小二乘迭代法、有理分式多项式法和正交多项式法。第9章介绍一些常用模态参数的时域识别方法和预处理方法,包括随机减量法、NE_xT法、ITD法、STD法、复指数法和ARMA模型时序分析法。第10章简单介绍几种单输入多输出的整体模态参数识别方法,包括整体正交多项式法、整体复指数法和整体ARMA模型时序法。

可以说,这是一本比较实用的书,侧重于振动信号处理方法的应用,书中没有太多的理论描述和公式推导,所讲述的振动信号处理方法也是目前应用广泛,较为经典的方法。书中的编程实例的源代码大多以命令M文件(脚本文件)的形式给出,程序中添加了足够的注释,在MATLAB环境下可直接运行。本书的程序全部在计算机上调试通过,同时还采用了一些实测振动数据作为考核程序的例题,并给出处理结果图形和数据。这些编程实例多数具有非常实用的振动信号处理功能。

由于本书内容众多,涉及领域广泛,加之作者学识水平有限,疏漏和不当之处在所难免,敬请广大读者批评指正。

王 济

2005年9月

目 录

前 言

第 1 章 MATLAB 快速入门	1
1.1 MATLAB 简介	1
1.2 MATLAB 通用命令	2
1.3 MATLAB 的数据类型	3
1.3.1 数据结构特点	3
1.3.2 矩阵（数组）	3
1.3.3 字符数组	4
1.3.4 单元数组	6
1.3.5 结构型数组	6
1.4 MATLAB 的基本运算操作	6
1.4.1 变量	6
1.4.2 算术运算符	7
1.4.3 操作符	7
1.4.4 关系运算符	8
1.4.5 逻辑运算符	9
1.4.6 常用数学函数	9
1.5 M 文件的程序设计	10
1.5.1 M 文件的特点及功能	11
1.5.2 命令 M 文件	12
1.5.3 函数 M 文件	12
1.5.4 M 文件中常用的动态交互命令	13
1.6 程序流程的控制	13
1.6.1 循环语句	14
1.6.2 选择语句	14
1.6.3 分支语句	15
1.6.4 其他控制语句	16
1.7 MATLAB 文件的输入和输出	16
1.7.1 数据文件打开与关闭	17
1.7.2 二进制文件的读写	17
1.7.3 格式化文件的读写	18
1.7.4 格式化字符串的读写	19

1.7.5	文件位置指针控制	19
1.7.6	文件操作出错信息查询	19
1.8	MATLAB 绘图功能	20
1.8.1	图形窗口设置	20
1.8.2	二维图形绘制	21
1.8.3	三维图形绘制	21
1.8.4	图形坐标设置	22
1.8.5	图形标注	22
1.8.6	图形控制	23
1.9	MATLAB 信号处理工具箱介绍	23
1.9.1	信号变换	23
1.9.2	IIR 数字滤波器设计	24
1.9.3	FIR 数字滤波器设计	25
1.9.4	窗函数	26
1.9.5	随机信号时域处理	27
1.9.6	随机信号频域处理	27
1.9.7	信号的重采样	28
1.9.8	波形生成	29
第 2 章	振动信号处理的基本概念	30
2.1	振动的描述	30
2.1.1	周期振动	31
2.1.2	非周期振动	33
2.1.3	随机振动	34
2.2	单自由度系统的振动力学模型和特征参数	35
2.2.1	单自由度系统的振动力学模型	35
2.2.2	单位脉冲响应函数与杜哈美尔积分	37
2.2.3	单自由度系统的传递函数和频响分析	39
2.3	多自由度系统的传递函数和频响函数	42
2.3.1	比例阻尼系统 (实模态理论)	43
2.3.2	一般阻尼系统 (复模态理论)	44
2.3.3	多自由度系统的脉冲响应函数与频响函数的关系	48
2.4	振动信号处理的基本内容	48
2.4.1	振动信号的预处理	49
2.4.2	振动信号的时域处理	49
2.4.3	振动信号的频域处理	49
2.4.4	试验模态参数识别	49
2.5	傅里叶变换	50
2.5.1	傅里叶级数的展开	50

2.5.2	连续傅里叶变换	51
2.5.3	傅里叶变换的性质	51
2.5.4	离散傅里叶变换	52
2.5.5	泄漏和加窗	52
2.5.6	快速傅里叶变换	53
第3章	振动测试	55
3.1	振动测试内容简介	55
3.2	振动测试系统的组成	56
3.2.1	振动信号测量	56
3.2.2	振动数据采集	56
3.3	振动测量仪器	57
3.4	数据采集系统	58
3.4.1	采集系统的工作原理	58
3.4.2	数据采集的操作过程	60
3.4.3	采样定理	61
3.5	激振设备	62
3.5.1	振动台	62
3.5.2	激振器	63
3.5.3	起振机	64
3.5.4	力锤	65
3.6	振动测试的基本内容及激振方式	65
3.6.1	振动强度测试	66
3.6.2	动力特性测试	67
第4章	振动信号预处理方法	69
4.1	采样数据的标定变换	69
4.2	消除多项式趋势项	69
4.3	采样数据的平滑处理	73
4.3.1	平均法	73
4.3.2	五点三次平滑法	78
第5章	振动信号时域处理方法	83
5.1	数字滤波	83
5.1.1	数字滤波的频域方法	84
5.1.2	数字滤波的时域方法简介	89
5.1.3	IIR 数字滤波器	90
5.1.4	FIR 数字滤波器	97
5.2	振动信号的积分和微分变换	104
5.2.1	时域积分	105
5.2.2	时域微分	105

5.2.3	频域积分	105
5.2.4	频域微分	108
5.3	随机振动信号时域处理方法	112
5.3.1	随机振动信号的特性	112
5.3.2	概率分布函数和概率密度函数	112
5.3.3	均值、均方值及方差	113
5.3.4	相关函数	114
第6章	振动信号频域处理方法	118
6.1	随机振动信号频域处理方法	118
6.1.1	平均周期图方法	119
6.1.2	自功率谱密度函数	119
6.1.3	互功率谱密度函数	120
6.1.4	频率响应函数(频响函数)	120
6.1.5	相干函数(凝聚函数)	120
6.2	窗函数	124
6.2.1	矩形窗	125
6.2.2	汉宁窗	125
6.2.3	海明窗	125
6.2.4	布莱克曼窗	125
6.2.5	三角窗	125
6.2.6	余弦坡度窗	126
6.2.7	帕曾窗	126
6.2.8	指数窗	126
6.2.9	高斯窗	126
6.2.10	窗函数的选择	127
6.3	ZOOM-FFT	130
6.4	三分之一倍频程谱	134
6.5	倒频谱变换	137
6.5.1	实倒谱	138
6.5.2	复倒谱	138
6.6	反应谱	141
第7章	数字信号的生成	145
7.1	频率扫描信号	145
7.2	拍波信号	147
7.3	白噪声随机波信号	149
7.4	人工模拟地震波信号	152
第8章	试验模态参数的频域识别方法	160
8.1	模态参数识别的概念及频域识别方法的特点	160

8.2	导纳圆拟合法	162
8.3	最小二乘迭代法	168
8.4	加权最小二乘迭代法	179
8.5	有理分式多项式法	187
8.6	正交多项式法	193
第9章	试验模态参数的时域识别方法	202
9.1	时域识别方法的特点	202
9.2	随机减量法	203
9.3	NExT 法	207
9.4	ITD 法	210
9.5	STD 法	219
9.6	复指数法	226
9.7	ARMA 模型时间序列分析法	235
第10章	试验模态参数的整体识别方法	243
10.1	模态参数整体识别方法简介	243
10.2	整体正交多项式法	244
10.3	整体复指数法	245
10.4	整体 ARMA 时序分析法	247
	参考文献	250

第 1 章 MATLAB 快速入门

1.1 MATLAB 简介

MATLAB 是美国 MathWorks 公司开发的大型数学计算应用软件系统，它提供了强大的矩阵处理和绘图功能，简单易用，可信度高，灵活性好，因而在世界范围内被科学工作者、工程师以及大学生和研究生广泛使用，目前已经成为国际市场上科学研究和工程应用方面的主导软件。掌握 MATLAB 并借助它解决理论与应用问题已经成为每一个从事科学研究和工程技术人员应该具备的技能。

MATLAB 是 MATrix LABoratory（矩阵实验室）的缩写。MATLAB 自 1984 年推向市场以来，在许多优秀程序设计和编制人员的不断努力和卓越贡献下，经过十几年的完善和扩充，使它从一个简单的矩阵分析软件逐渐发展成为一个通用性高、带有规模大和覆盖面广的工具箱、有强大可视化功能的科学计算操作平台。因此 MATLAB 被誉为巨人肩上的工具。

MATLAB 的编程语言是一种面向科学与工程计算的高级语言，允许用数学形式的语言编写程序。MATLAB 的基本数据单元是矩阵，即使向量和标量也当作是矩阵的特殊形式。在进行程序编制时，矩阵的维数和大小无需定义，可以随时任意扩大和缩小，数据类型也不需要说明，使用起来十分方便。MATLAB 提供给用户的是一种最简洁、最直观的程序开发环境。由于 MATLAB 编程语言所使用数学表达形式和运算规则与人们进行科学计算时通常习惯的思路和表达方式完全一致，所以不像学习其他高级语言，例如 Basic、Fortran 和 C 等那样难于掌握。用 MATLAB 编写程序犹如在演算纸上排列出公式与求解问题，所以 MATLAB 语言又被称为演算纸式的科学算法语言。MATLAB 语言是一种解释执行的语言，它灵活、方便，其调试程序手段丰富，调试速度快，需要学习时间少，并且把编辑、编译、连接和执行融为一体。它能使用户在同一画面上进行灵活操作，快速排除输入程序中的错误，加快用户编写、修改和调试程序的速度。

MATLAB 系统的基本函数库具有初等函数、初等矩阵和矩阵变换、包括线性代数方程组和矩阵特征值问题等数值线性代数、多项式运算和求根、函数的插值和数据的多项式拟合、数值积分和常微分方程数值解、单变量非线性方程求根、函数求极值、数据分析和傅里叶变换以及某些特殊的矩阵函数和数学函数等众多的内容。

MATLAB 还包括了一系列被称作工具箱（TOOLBOX）的专业求解工具。工具箱实际上是 MATLAB 针对不同学科、不同专业所开发的专用函数库，用来求解各个领域的数值计算问题，包括信号处理、图像处理、小波分析、控制系统、系统辨识和神经网络等。随着 MATLAB 版本的不断升级，所含的工具箱的功能越来越丰富，规模越来越庞大，因

此, 应用范围也越来越广泛, 成为各种专业科研人员和工程技术人员得力的工具。

MATLAB 还有一个受人欢迎的特点就是程序的开放性。除内部函数外, 所有 MATLAB 基本函数文件和所有工具箱的函数文件都是可以进行修改的源文件。用户可以对源文件进行修改, 加入自己编写的内容来构成新的专用工具箱。

MATLAB 具有二维、三维曲线和三维曲面绘图功能, 使用方法十分方便。几条简单的语句, 就能将离散的数据用图形的方式显示出来, 让用户一目了然。通过对图形的线型、坐标、立面、色彩、渲染、光线和视角等的控制以及动画的绘制, 可以把数据的特征表现得淋漓尽致。另外用户可以根据需要在图形上加标题、图例标注、坐标轴标记、文本注释及坐标网格等, 还可以在一幅图上同时画出几条曲线, 在一个图形窗口上画出若干幅图形。

在 Windows 系统桌面上双击 MATLAB 的图标, 即可启动 MATLAB, 进入 MATLAB 默认设置的桌面平台。命令窗口 (Command Window) 是 MATLAB 程序运行时的主窗口, MATLAB 的人机交互和数据的输入输出, 都需要命令窗口调用其他程序来完成。在命令窗口中, “>>” 为运算提示符, 表示 MATLAB 正处在准备状态。当提示符后输入一段运算式或某个命令并按 Enter 键后, 窗口将显示计算结果或进入调用该命令后的工作状态。

1.2 MATLAB 通用命令

通用命令在 MATLAB 中执行最基本的功能和操作, 因此在开始使用 MATLAB 时, 需要先对一些通用命令作初步了解。常用的通用命令包括以下几条:

(1) cd 显示当前工作目录。cd c:\data 改变工作目录到 c:\data。

(2) dir 列出当前工作目录下的文件。

(3) type 显示文件内容。type a.m 显示文件 a.m 的内容。

(4) clear 清除内存中所有变量和函数。clear A 清除变量 A。

(5) clf 清除当前图形窗口。

(6) close 关闭当前图形窗口。close all 关闭所有图形窗口。

(7) pack 收集内存碎片, 扩大内存空间。

(8) clc 清除工作窗口所显示的内容。

(9) path 指定搜索路径。path 显示 MATLAB 的所有搜索路径; path ('newdir') 将搜索路径改变为字符串 'newdir' 指定的路径。

(10) hold 图形保持开关。hold on 开, hold off 关。

(11) disp 显示变量或文字内容。disp (A) 如果 A 是数组, 则显示数组的数值, 如果 A 是字符串, 则显示字符串的内容。

(12) help 在线帮助, 是 MATLAB 最为常用的命令。help 显示 MATLAB 当前的帮助系统所包含的所有项目。help 函数类名, 显示该类所有函数的名称及功能。help 命令或函数名, 显示命令或函数的功能和调用方式。

(13) lookfor 在线查找。lookfor 关键词, 查询并显示根据关键词所搜索到的函数

名称。

(14) who 列出工作空间中的变量名。

(15) whos 列出工作空间中变量的详细信息。

(16) length 给出向量的长度或字符串的字符数。

(17) size 给出矩阵的维数。

(18) save 保存内存变量到指定文件。save filename 将工作空间的所有变量按二进制格式存到名为 filename.mat 的文件中。save filename.dat x y z 将变量 x、y、z 按二进制格式存到名为 filename.dat 的文件中。save filename.txt -ascii 将变量 x 按文本格式 (ASCII 码) 存到名为 filename.txt 的文件中。

(19) load 从磁盘文件中重新读入变量的内容到工作空间。load filename 从名为 filename.mat 的二进制文件读入变量。load filename -ascii 从名为 filename.mat 的文本文件读入变量。

(20) quit 退出 MATLAB 系统。

1.3 MATLAB 的数据类型

MATLAB 的数据类型主要包括矩阵 (数组)、字符串、单元型数组和结构型数组。本节将介绍这些数据类型。

1.3.1 数据结构特点

MATLAB 语言的一个最重要的特点是以矩阵为基本数据单元, 所有运算和操作都是围绕着矩阵进行的, 并且矩阵的运算和操作可以像单个数的运算和操作一样简单方便。

在 MATLAB 中, 从数的角度来看, 矩阵和数组都是数的集合, 没有明显区别。但从运算的角度来看, 矩阵运算和数组运算却是截然不同的。矩阵运算从矩阵的整体出发, 按照线性代数的法则运算, 而数组运算是对应数组中每个元素进行的, 是元素与元素之间的运算。所以, 作为数据的存储形式, 矩阵和数组是一样的, 可以都看作是数组, 以便与其他编程语言在概念上保持一致, 只是在进行操作或运算时才加以区别。

所有 MATLAB 数据类型的存储方式均按列存储, 这与 C 语言按行存储的方式是不同的, 对于习惯使用 C 语言的用户, 需要注意到这一点。

1.3.2 矩阵 (数组)

复数矩阵是 MATLAB 中最常用和最基本数据类型, 是 MATLAB 语言操作与运算的主要对象。矩阵的大小可以表示为 $n \times m$, 其中 n 和 m 分别为矩阵的行数和列数。一个矩阵既可以是普通数学意义上的矩阵, 也可以是标量或向量。对于标量 (一个数) 可以将其称为 1×1 矩阵, 而向量则可以认为是 $1 \times n$ (行向量) 或 $n \times 1$ (列向量) 矩阵。另外, 一个 0×0 矩阵在 MATLAB 中也是有意义的, 称为空矩阵。矩阵是 MATLAB 的基本数据单元, 学习 MATLAB 的最好方法是从学习操作矩阵开始。通常意义下, MATLAB 将一切矩阵均视为复数矩阵, 实数矩阵是复数矩阵所有元素虚部均为零的特例。

创建矩阵是了解 MATLAB 矩阵操作的初步。在 MATLAB 中，直接输入矩阵元素比较直观，尤其是对于维数较低的矩阵，显得非常方便。直接输入一个矩阵只需将该矩阵的所有元素按行的顺序排列，用方括号将这些元素括起来，并且同行各元素用逗号或者空格分隔，矩阵行与行之间用分号隔开即可。例如，需要定义一个矩阵：

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

在 MATLAB 中，可写成 $A = [1\ 2\ 3; 4\ 5\ 6; 7\ 8\ 9]$ 。

向量生成在振动信号数据处理中时常要用到，利用冒号表达式生成向量是最常用的一种方式。冒号表达式的基本形式为 $a = a_0 : \text{step} : a_n$ ，其中 a_0 表示向量第一个元素数值， a_n 表示向量最后一个元素数值，step 表示前后元素数值的差值。step 可以是正数或负数，这取决于 $a_n - a_0$ 大于 0 或小于 0，但是 $a_n - a_0$ 必须是 step 的整数倍值。若 $\text{step} = 1$ ，可直接表示为 $a = a_0 : a_n$ 。MATLAB 提供的线性等分函数 `linspace` 也是向量生成的常用工具，调用方式为 $b = \text{linspace}(a_0, a_1, n)$ ，返回生成的 n 维行向量，使得 $b(1) = a_0$ ， $b(n) = a_1$ 。

MATLAB 中提供了许多生成特殊矩阵的函数，常用基本函数有 `zeros`、`ones`、`eye`、`rand` 和 `randn`。函数 `zeros` 用来生成元素为 0 的矩阵，调用方式 $A = \text{zeros}(n, m)$ 和 $A = \text{zeros}(n)$ 分别表示生成 $n \times m$ 维和 $n \times n$ 维的元素全为 0 矩阵。函数 `ones` 用来生成元素为 1 矩阵，`eye` 生成单位矩阵，`rand` 生成 0~1 之间按均匀分布的随机数矩阵，`randn` 生成 0~1 之间按正态分布的随机数矩阵。这些函数的调用方式与 `zeros` 基本相同。表明矩阵大小的函数 `size` 在构造矩阵中是十分有用，例如， $A = \text{zeros}(\text{size}(B))$ 表示生成一个与 B 同样大小的 0 矩阵存于 A 。另外，通过读入外部文件数据来构造矩阵，在振动信号数据处理中常常需要用到，读入数据的具体方法可参见本章第 1.7 节的内容。

1.3.3 字符数组

字符数组也称为字符串，是 MATLAB 语言的另一种基本数据类型。在 MATLAB 中，所有的字符串都需要用单引号定义后才能输入或赋值，例如 $\text{str} = \text{'MATLAB'}$ 。字符串中每个字符都是字符数组的一个元素。与 C 语言不同的是 MATLAB 字符中可以包括空格符。

MATLAB 的字符串处理功能是很丰富的。对于图形用户界面设计中涉及许多图形对象的属性及其取值，需要通过字符数组的处理来完成，有时甚至需要相当的技巧。学习和掌握字符串的处理功能是不可缺少的基础知识。以下介绍一些处理字符串常用的函数：

(1) `char` 将不同长度的字符串组合成同一长度的字符矩阵。调用方式 $A = \text{char}(\text{str1}, \text{str2}, \dots, \text{strn})$ 表示将 n 个字符串组合成一个字符矩阵 A 并以最长的串为基准，其余的字符串右边补足空格。

(2) `deblank` 删除生成字符矩阵时添加的空格。调用方式 $c = \text{deblank}(A(i,:))$ 表示提取字符矩阵 A 的第 i 行字符串，并去除其中的空格，结果存于字符串 c 中。

(3) `cellstr` 把字符矩阵转换为单元数组。调用方式 `B=cellstr(A)` 表示将字符矩阵 `A` 转换为字符单元数组 `B`。单元数组 (Cell Array) 是 MATLAB 中一类特殊的数组, 它的每个元素可以是类型不同和维数不同的矩阵、向量、标量或多维数组。字符单元数组正好满足了长度不等的字符串组合成矩阵的需要。反过来, 函数 `char` 又可以将字符单元数组转换为字符矩阵。

(4) `strcmp` 比较两字符串是否相等。调用方式 `k=strcmp(str1, str2)` 表示比较字符串 `str1` 和 `str2`, 如果对应位置上的字符都相等则返回 1 (逻辑真), 否则返回 0 (逻辑假)。`F=strcmp(S, T)` 表示其中字符串 `S` 和 `T` 必有一个是字符单元数组。比较运算的结果将返回一个与 `S, T` 大小相同的数组 `F`, `F` 的元素由 0 和 1 组成, 0 表示 `S` 和 `T` 对应位置上的元素不相等, 1 表示 `S` 和 `T` 对应位置上的元素相等。`S` 和 `T` 必须大小相等, 除非其中一个为标量。

(5) `strncmp` 比较两字符串前 `n` 个字符是否相等。调用方式 `k=strcmp(str1, str2, n)` 就是在函数 `strcmp` 的两种调用方式基础上加上第三个参数 `n` 来指定比较的字符数。

(6) `isletter` 和 `isspace` 分别判别一个字符串中的每个元素是否为字母和空格。它们都返回与字符串大小相同的 0 或 1 向量, 0 表示相应的字符不是字母或空格, 1 则表示是字母或空格。

(7) `findstr` 查找字符串中的子串。调用方式 `k=findstr(str1, str2)` 表示将给出 `str1` 和 `str2` 中较短的字符串在另一个较长的字符串中出现的第一个下标, 如果不止一次出现, 则返回一个包括该字符串所有出现位置的行向量。

(8) `strrep` 替换字符串的子串。调用方式 `str=strrep(str1, str2, str3)` 表示将字符串 `str1` 中的所有子串 `str2` 用字符串 `str3` 来替换, 替换后的结果返回给 `str`。`strrep(str1, str2, str3)` 表示当 `str1, str2` 和 `str3` 中有一个是字符单元数组且不指定输出参数时, `strrep` 将返回一个和三个输入参数同样大小的字符单元数组, 并用 `str3` 中的元素来替换 `str1` 中对应位置上的元素的子串, 这个子串由 `str2` 的对应位置上的元素指定。如果我们需要同时替换多个字符串中的子串, 就可以采用这一种调用方式。

(9) `int2str` 转换整数成十进制字符串。调用方式 `str=int2str(m)` 表示将 `m` 转换成十进制字符串, 其中 `m` 可以是单个整数或是整数向量和矩阵, 如果 `m` 不是整数, 则先将 `m` 取整, 然后再进行转换。

(10) `num2str` 按指定输出精度或格式把数值转换为十进制字符串。调用方式 `str=num2str(A, precision)` 表示将数值矩阵 `A` 按 `precision` 指定最大精度, 即输出字符串所包含的最多数字的位数转换为十进制字符串, 缺省时按四位精度转换。`str=num2str(A, format)` 表示按 `format` 指定的格式将矩阵 `A` 转换成十进制字符串。缺省时 `format` 的格式为 `%11.4g`。

(11) `str2num` 将字符串转换为十进制数值。调用方式 `A=str2num(str)` 表示将字符串 `str` 转换为十进制数值并存于矩阵 `A`。

对于图形的标注是非常有用的。在图形的标注中, 需要使用字符串, 而在标注的字符串中往往包含有数值, 这时我们可以用 `int2str` 和 `num2str` 把其中的数值转换成字符串。

1.3.4 单元数组

单元数组是 MATLAB 系统中非常特殊的一种数据类型，单元数组的每一个元素称为一个单元 (cell)，每一个单元可以为各种的 MATLAB 基本数据类型。单元数组的各个单元之间可以互不相同，极为灵活。创建单元矩阵总的来说有两种方式：一种方式为直接对单元阵列的单元进行赋值；另一种为首先使用 cell 函数为单元阵列分配空间，然后进行赋值。例如在 MATLAB 命令提示符下，键入如下命令：

```
A (1, 1) = { [1 2 3; 4 5 6; 7 8 9]};
A (1, 2) = { ' It is a cell. '};
A (2, 1) = { [1.2 3.4; 5.6 7.8]};
A (2, 2) = { [1 2 3 4 5 6]};
```

1.3.5 结构型数组

MATLAB 的结构型数组是另一种可以将不同类型数据组合在一起的 MATLAB 语言数据类型。与单元型数组不同的是，结构型数组是通过指针方式来传递数据的。结构型数组中一个元素类似 C 语言的一个结构变量的定义。创建结构数组可以通过直接赋值和使用 struct 函数这两种方法来实现。

直接赋值时，需要指定结构的成员名，并以指针操作符“.”来连接元素的结构名与成员名，例如：

```
C= [4 3 2 1]; D= [4 3; 2 1];
A (1) .a='abcd'; A (1) .b=1.2; A (1) .c= [1 2 3 4]; A (1) .d= [1 2; 3 4];
A (2) .a='efgh'; A (2) .b =5.4; A (2) .c=C;          A (2) .d=D;
.....
```

其中，A (1) 为结构数组的一个元素的结构名，a、b、c、d 分别为结构的成员名。

用 MATLAB 语言中提供的 struct 函数来创建结构型数组也是很方便的，对于上例 A (2) 可以调用 struct 函数来创建：

```
A (2) =struct ('a', 'efgh', 'b', 5.4, 'c', C, 'd', D);
```

对结构型数组进行赋值时，也可以只对部分元素赋值，其余的元素将被赋以空矩阵，并可以随时对该数组加以修改和添加。

1.4 MATLAB 的基本运算操作

1.4.1 变量

变量在 MATLAB 语言程序中最基本的单元，是以矩阵的形式存在的。与其他常规的编程语言不同的是，MATLAB 语言不要求用户在创建变量的时候进行声明，也不需要指定其大小和类型，也就是说用户在 MATLAB 工作空间内添加一个新的变量时，根据输入或赋值的情况，MATLAB 会自动给该变量分配适当的内存。这个变量可以是单个元素、一个向