



郑智琴 编著

Simulink

电子通信仿真与应用

工程师工具软件应用系列

国防工业出版社

# Simulink 电子通信仿真与应用

郑智琴 编著

国防工业出版社

·北京·

## 内 容 简 介

这是一本深入地介绍 MATLAB 的 Simulink 仿真应用的参考书。它以最新的 MATLAB 6.0 版为依据,同时也兼顾了 MATLAB 以前版本。它在介绍计算机仿真的基本概念和基本方法的基础上,概括地介绍了 MATLAB 中 Simulink 仿真的原理和技巧,并通过大量的例子,着重讲解了 Simulink 工具的使用和建模技巧。

本书适于使用 Simulink 仿真的读者阅读,更适于欲使用 Simulink 对动态系统建模的读者阅读。

### 图书在版编目(CIP)数据

Simulink 电子通信仿真与应用/郑智琴编著. —北京:国防工业出版社,2002.8

(工程师工具软件应用系列)

ISBN 7-118-02846-0

I. S... II. 郑... III. 计算机辅助计算 - 软件工具, Simulink - 应用 - 模拟信号 - 信号处理 IV. TN911.71

中国版本图书馆 CIP 数据核字(2002)第 025926 号

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

新艺印刷厂印刷

新华书店经售

\*

开本 787 × 1092 1/16 印张 21 $\frac{3}{4}$  504 千字

2002 年 8 月第 1 版 2002 年 8 月北京第 1 次印刷

印数:1 - 3000 册 定价:29.00 元

(本书如有印装错误,我社负责调换)

# 前 言

MATLAB 是 Mathworks 公司推出的一套高效率的数值计算和可视化软件,它集数值分析、矩阵运算、信号处理和图形显示于一体,构成了一个方便的、界面友好的用户环境。MATLAB 的推出得到了各个领域专家学者的广泛关注,其强大的扩展功能更为各个工程领域提供了分析和设计的基础。随着 MATLAB 在各个工程领域应用的日益广泛,专家学者们相继推出了控制系统工具箱(control systems toolbox),信号处理工具箱(signal processing toolbox),系统辨识工具箱(system identification toolbox)、最优化控制箱(optimisation)、小波分析工具箱(wavelet)等简单实用的库函数,这些工具箱给各个领域的研究和工程应用提供了强有力的工具,而且这些工具箱还在不断增加。借助于日渐完善的 MATLAB 软件,各个层次的研究人员可以直观、方便地进行分析、计算和设计工作,大大节省了时间,提高了工作效率。

然而面对国际上 MATLAB 在各个工程领域中应用的迅猛发展的形势,国内系统地介绍 MATLAB 仿真工程,尤其在电信系统的 Simulink 电子仿真方面的书籍资料还很缺乏,而本书就是在这种背景下应运而生的。

MATLAB 是一个高度集成的系统,它提供的 Simulink 是一个用来对动态系统进行建模、仿真和分析的软件包,支持线性和非线性系统,能够在连续时间域、离散时间域或者两者的混合时间域里进行建模,它同样支持具有多种采样速率的系统。在过去几年里,Simulink 已经成为教学和工业应用中对动态系统进行建模时使用得最广泛的软件包。

阅读本书的读者应该对 MATLAB 已经有一定的掌握,本书没有从 MATLAB 最基本的使用讲起,而是对 MATLAB 的基本操作在第二、三章以例子的形式进行了简单的说明。接着就着重对 MATLAB 中的 Simulink 的使用进行了详细的介绍。

本书的最大特点就是与工程实际的紧密结合,书中所列举的例子基本上都来源于工程实际,甚至有些就是现实当中的工程设计。这样就使本书能够很好地做到理论联系实际,使读者对各种 Simulink 仿真应用有一个感性的认识,使得书中的内容更加易学易用。本书的另一大特色就是注重仿真工程应用的系统化,书中严格地按照各种理论系统进行仿真过程的设计,使得所有的工程内容都可以找到理论根源,从而方便了读者对各种理论背景的查询。同时本书根据各种理论设计一套严整的仿真函数,这些函数都是 MATLAB 库函数中所没有而工程实践中又必须用到的,因而工程设计人员可以直接调用书中提供的一些函数来进行电子仿真,从而大大方便了系统分析设计人员的仿真应用。

本书由郑智琴主编,董欣、李世魁、郑亮、杜鹏、王颖、刘建军、丁建中、王国平、刘辉、李丽妍、林林、石玉等同志参与了本书的部分编写工作,李帅辉、王军、杜林、马雨平、宋涛、张浩、李申等同志调试了所需程序。当然,限于作者的水平,本书的不足之处在所难免,欢迎广大读者和用户批评指正。

编 者

2002 年 6 月

# 目 录

<b>第 1 章 概论</b> .....	1
1.1 MATLAB 的发展历程和影响 .....	1
1.2 MATLAB 6.0 的基本组成和特点 .....	2
1.2.1 MATLAB 的语言部分 .....	2
1.2.2 MATLAB 的工作环境 .....	4
1.2.3 MATLAB 的视图系统 .....	5
1.2.4 MATLAB 的数学函数库 .....	7
1.2.5 MATLAB 与外部程序的交互 .....	8
1.3 与 MATLAB 6.0 配用的 Simulink 3.0 .....	8
1.3.1 Simulink 的传统优点 .....	8
1.3.2 Simulink 3.0 的特点 .....	9
<b>第 2 章 M 文件和面向对象编程</b> .....	10
2.1 入门 .....	11
2.1.1 编写和运行 .....	11
2.1.2 规则和属性 .....	12
2.2 MATLAB 控制流 .....	18
2.2.1 for 循环结构 .....	18
2.2.2 while 循环结构 .....	19
2.2.3 if - else - end 分支结构 .....	19
2.2.4 switch - case 结构 .....	20
2.2.5 try - catch 结构 .....	21
2.3 脚本文件和函数文件 .....	21
2.3.1 M 文件的一般结构 .....	21
2.3.2 “变长度”输入输出宗量 .....	22
2.4 跨空间变量传递 .....	24
2.4.1 跨空间计算串表达式的值 .....	24
2.4.2 跨空间赋值 .....	25
2.5 串演算函数 .....	25
2.5.1 eval .....	25
2.5.2 feval .....	26
2.6 内联函数创建和应用示例 .....	26
2.7 调试器应用示例 .....	28

2.8	M 文件性能分析 .....	31
2.8.1	分析器 .....	31
2.9	面向对象编程 .....	32
2.9.1	面向对象编程应用示例 .....	32
2.10	继承性创建子类的示例 .....	37
<b>第 3 章</b>	<b>图形用户界面的制作 .....</b>	<b>40</b>
3.1	入门 .....	40
3.2	界面菜单 .....	43
3.2.1	图形窗的标准菜单 .....	43
3.2.2	自制的用户菜单 .....	44
3.2.3	用户菜单的属性 .....	44
3.2.4	现场菜单的制作 .....	50
3.3	用户控件 .....	50
3.3.1	双位按键、无线电按键、控件区域框示例 .....	50
3.3.2	静态文本框、滑动键、检录框示例 .....	52
3.3.3	可编辑框、弹出框、列表框、按键示例 .....	54
3.4	由 M 函数文件产生用户菜单和控件 .....	56
3.4.1	利用全局变量编写用户界面函数文件 .....	56
3.4.2	利用 'UserData' 属性编写用户界面函数文件 .....	57
3.4.3	利用递归法编写用户界面函数文件 .....	57
3.5	图形用户界面设计工具 .....	59
3.5.1	交互式用户界面设计工具应用示例 .....	60
3.5.2	为读者提供的配套文件和数据 .....	63
<b>第 4 章</b>	<b>Simulink 入门 .....</b>	<b>68</b>
4.1	Simulink 概述 .....	68
4.1.1	什么是 Simulink .....	68
4.1.2	Simulink 模型的特点 .....	70
4.2	Simulink 入门 .....	74
4.3	熟悉 Simulink 模型窗口 .....	78
4.4	键盘和鼠标的操作 .....	84
4.5	模块库简介 .....	86
<b>第 5 章</b>	<b>Simulink 详解 .....</b>	<b>91</b>
5.1	Simulink 的模块和模块库 .....	91
5.1.1	Simulink 里的模块 .....	91
5.1.2	Simulink 的模块库 .....	96
5.2	模拟方程 .....	100
5.3	Simulink 里的数据类型 .....	103
5.3.1	Simulink 支持的数据类型 .....	103
5.3.2	数据类型的传递 .....	105

5.3.3 在模型里使用复数信号 .....	106
5.4 建立子系统 .....	107
5.4.1 建立子系统 .....	107
5.4.2 用子系统来定义库 .....	109
5.5 封装子系统 .....	111
5.5.1 子系统封装示例 .....	111
5.5.2 initialization 页 .....	114
5.5.3 icon 页(图标页) .....	120
5.5.4 documentation 页 .....	124
5.5.5 为封装的模块建立动态对话框 .....	124
5.6 建立条件子系统 .....	126
5.6.1 使能子系统 .....	126
5.6.2 触发子系统 .....	128
5.6.3 触发使能子系统 .....	130
<b>第6章 Simulink 调试器</b> .....	132
6.1 使用调试器 .....	132
6.2 增量运行模型 .....	135
6.3 设置断点 .....	138
6.3.1 非条件中断 .....	138
6.3.2 条件中断 .....	140
6.4 显示仿真有关的信息 .....	142
6.4.1 显示模块的输入和输出(I/O) .....	142
6.4.2 显示代数环信息 .....	145
6.4.3 显示系统 .....	145
6.4.4 显示积分信息 .....	146
6.5 显示模型的信息 .....	146
6.6 Simulink 4.0 的图形调试工具 .....	148
6.7 调试命令使用详解 .....	151
<b>第7章 仿真运行和结果分析</b> .....	159
7.1 使用菜单命令运行仿真 .....	159
7.2 仿真参数对话框 .....	161
7.2.1 Solver 页 .....	161
7.2.2 Workspace I/O 页 .....	165
7.2.3 Diagnostics 页 .....	171
7.2.4 Advanced 页 .....	173
7.3 改善仿真的性能和精确度 .....	175
7.3.1 加速仿真 .....	175
7.3.2 改善仿真的精度 .....	176
7.4 从命令运行仿真 .....	176

7.4.1	使用 sim 命令 .....	176
7.4.2	使用 set_param 命令 .....	177
7.5	分析仿真结果 .....	177
7.5.1	观看输出结果的轨迹 .....	177
7.5.2	线性化 .....	181
7.5.3	平衡点的分析 .....	186
<b>第 8 章</b>	<b>深入理解 Simulink</b> .....	<b>191</b>
8.1	Simulink 如何工作 .....	191
8.1.1	基本模型 .....	191
8.1.2	进行仿真 .....	192
8.1.3	过零检测 .....	193
8.2	代数环 .....	198
8.2.1	直接馈入环路(direct feedthrog)—代数环 .....	198
8.2.2	非代数直接馈入环路 .....	200
8.3	离散时间系统 .....	200
8.4	使用回调函数 .....	204
8.4.1	回调函数基本概念 .....	204
8.4.2	回调函数示例 .....	208
8.5	模型文件格式 .....	211
<b>第 9 章</b>	<b>使用 Real-Time Workshop</b> .....	<b>223</b>
9.1	Real-Time Workshop 概述 .....	223
9.1.1	Real-Time Workshop 能做什么 .....	223
9.1.2	使用前的准备工作 .....	224
9.1.3	RTW 中的基本概念 .....	225
9.2	生成普通的实时程序 .....	226
9.2.1	Simulink 模型 .....	227
9.2.2	生成实时代码 .....	228
9.2.3	代码验证 .....	233
9.3	代码生成和建立过程 .....	237
9.3.1	自动程序建立 .....	237
9.3.2	Real-Time Workshop 用户界面 .....	238
9.4	外部模式 .....	243
9.4.1	介绍 .....	243
9.4.2	使用 grt(普通实时目标)的外部模式入门 .....	244
9.4.3	外部模式 GUI .....	249
9.4.4	外部模式的 TCP/IP .....	252
9.5	RTW 代码库 .....	253
9.5.1	Custom Code Library(自定义代码库) .....	254
9.5.2	使用自定义代码模块示例 .....	256



<b>第 10 章 用 S-函数扩展 Simulink</b> .....	259
10.1 S-函数综述 .....	259
10.1.1 什么是 S-函数 .....	259
10.1.2 S-函数如何工作 .....	264
10.1.3 M 文件和 C MEX 文件 S-函数综述 .....	266
10.1.4 S-函数概念 .....	266
10.2 建立 M 文件 S-函数 .....	269
10.2.1 如何使用模板 .....	269
10.2.2 定义 S-函数的初始信息 .....	276
10.2.3 输入和输出参量说明 .....	278
10.2.4 M 文件 S-函数的几个示例 .....	279
10.3 C MEX S-函数 .....	294
10.3.1 介绍 .....	294
10.3.2 编写基本的 C MEX S-函数 .....	295
10.3.3 建立更复杂的 C MEX S-函数 .....	301
10.4 建立 C++ S-函数 .....	304
10.4.1 源文件格式 .....	304
10.4.2 建立永久 C++ 对象 .....	308
<b>附录 A 常用“关键符(词)”</b> .....	310
<b>附录 B Internet 资源</b> .....	334

# 第 1 章 概 论

## 1.1 MATLAB 的发展历程和影响

MATLAB 名字由 MATrix 和 LABoratory 两词的前三个字母组合而成。那是 20 世纪 70 年代后期的事:时任美国新墨西哥大学计算机科学系主任的 Cleve Moler 教授出于减轻学生编程负担的动机,为学生设计了一组调用 LINPACK 和 EISPACK 库程序的“通俗易懂”的接口,此即用 FORTRAN 语言编写的萌芽状态的 MATLAB。

经几年的校际流传,在 Little 的推动下,由 Little、Moler、Steve Bangert 合作,于 1984 年成立了 MathWorks 公司,并把 MATLAB 正式推向市场。从这时起, MATLAB 的内核采用 C 语言编写,而且除原有的数值计算能力外,还新增了数据视图功能。

MATLAB 以商品形式出现后,仅短短几年,就以其良好的开放性和运行的可靠性,使原先控制领域里的封闭式软件包(如英国的 UMIST,瑞典的 LUND 和 SIMNON,德国的 KEDDC)纷纷淘汰,而改以 MATLAB 为平台加以重建。在时间进入 20 世纪 90 年代的时候, MATLAB 已经成为国际控制界公认的标准计算软件。

到 90 年代初期,在国际上三十几个数学类科技应用软件中, MATLAB 在数值计算方面独占鳌头,而 Mathematica 和 Maple 则分居符号计算软件的前两名。Mathcad 因其提供计算、图形、文字处理的统一环境而深受中学生的欢迎。

MathWorks 公司于 1993 年推出 MATLAB 4.0 版本,从此告别 DOS 版。4.x 版在继承和发展其原有的数值计算和图形可视能力的同时,出现了以下几个重要变化:(1)推出了 Simulink。这是一个交互式操作的动态系统建模、仿真、分析集成环境。它的出现使人们有可能考虑许多以前不得不做简化假设的非线性因素、随机因素,从而大大提高了人们对非线性、随机动态系统的认知能力。(2)开发了与外部进行直接数据交换的组件,打通了 MATLAB 进行实时数据分析、处理和硬件开发的道路。(3)推出了符号计算工具包。1993 年 MathWorks 公司从加拿大滑铁卢大学购得 Maple 的使用权,以 Maple 为“引擎”开发了 Symbolic Math Toolbox 1.0。MathWorks 公司此举加快结束了国际上数值计算、符号计算孰优孰劣的长期争论,促成了两种计算的互补发展新时代。(4)构造了 Notebook。MathWorks 公司瞄准应用范围最广的 Word,运用 DDE 和 OLE,实现了 MATLAB 与 Word 的无缝连接,从而为专业科技工作者创造了融科学计算、图形可视、文字处理于一体的高水准环境。

1997 年仲春, MATLAB 5.0 问世,紧接着陆续推出 5.1、5.2 和 1999 年春的 5.3 版。与 4.x 相比,现今的 MATLAB 拥有更丰富的数据类型和结构、更友善的面向对象、更加快速精良的图形可视、更广博的数学和数据分析资源、更多的应用开发工具(关于 MATLAB 5.x 的特点下节将作更详细的介绍)。

诚然,到 1999 年底,Mathematica 也已经升到 4.0 版,它特别加强了以前欠缺的大规模数据处理能力。Mathcad 也赶在 2000 年到来之前推出了 Mathcad 2000,它购买了 Maple 内核和库的部分使用权,打通了与 MATLAB 的接口,从而把其数学计算能力提高到专业层次。但是,就影响而言,至今仍然没有一个别的计算软件可与 MATLAB 匹敌。

在欧美大学里,诸如应用代数、数理统计、自动控制、数字信号处理、模拟与数字通信、时间序列分析、动态系统仿真等课程的教科书都把 MATLAB 作为内容。这几乎成了 20 世纪 90 年代教科书与旧版书籍的标志性区别。在那里,MATLAB 是攻读学位的大学生、硕士生、博士生必须掌握的基本工具。

在国际学术界,MATLAB 已经被确认为准确、可靠的科学计算标准软件。在许多国际一流的学术刊物(尤其是信息科学刊物)上,都可以看到 MATLAB 的应用。

在设计研究单位和工业部门,MATLAB 被认作进行高效研究、开发的首选软件工具。如美国 National Instruments 公司信号测量、分析软件 LabVIEW, Cadence 公司信号和通信分析设计软件 SPW 等,或者直接建筑在 MATLAB 之上,或者以 MATLAB 为主要支撑。又如 HP 公司的 VXI 硬件, TM 公司的 DSP, Gage 公司的各种硬卡、仪器等都接受 MATLAB 的支持。

## 1.2 MATLAB 6.0 的基本组成和特点

经过近 20 年实践,人们已经意识到:MATLAB 作为计算工具和科技资源,可以扩大科学研究的范围、提高工程生产的效率、缩短开发周期、加快探索步伐、激发创造活力。那么,作为当前最新版本的 MATLAB 6.0 究竟包括哪些内容? 有哪些特点呢?

### 1.2.1 MATLAB 的语言部分

MATLAB 5.0 以前版本的 MATLAB 语言比较简单。它只有双精度数值和简单字符串两种数据类型,只能处理 1 维、2 维数组。它的控制流和函数形式也都比较简单。这一方面与当时软件的整体水平有关,另一方面与 MATLAB 仅限于数值计算和图形可视应用的设计目标有关。

从 MATLAB 5.0 起,MATLAB 对其语言进行了根本性的变革,使之成为一种高级的“阵列”式语言。

#### 1. MATLAB 语言的传统优点

MATLAB 自问世起,就以数值计算称雄。MATLAB 进行数值计算的基本处理单位是复数数组(或称阵列),并且数组维数是自动按照规则确定的。这一方面使 MATLAB 程序可以被高度“向量化”,另一方面使用户易写易读。

比如已知  $t$  的采样数据是  $(n \times m)$  维数组,要计算  $y = e^{-2t} \sin(5t)$ 。对一般的计算语言来说,必须采用两层循环才能得到结果。这不但程序复杂,而且那讨厌的循环十分费时。MATLAB 处理这类问题则简洁快捷得多,它只需直截了当的一条指令  $y = \exp(-2 * t) .* \sin(5 * t)$ ,就可获得同样是  $(n \times m)$  维的  $y$  数组。这就是所谓的“数组运算”。这种运算在信号处理和图形可视中,将被频繁使用。

又如对于求解  $Ax = b$  代数方程问题。教科书的基本叙述:当  $A$  是标量时,  $x = \frac{b}{A}$ ; 当  $A$  是非奇异矩阵时,  $x = A^{-1}b$ ; 当  $A$  是行数大于列数的满秩阵时,  $x = (A^T A)^{-1} A^T b$ ; 当  $A$  的列数大于行数时,  $x$  有无数解。一般程序就必须按以上不同情况进行编程。然而对 MATLAB 来说,它只需一条指令:  $x = A \setminus b$ 。指令是简单的,但其内涵却远远超出了普通教科书的范围,其计算的快速性、准确性和稳定性都是普通程序所远不及的。

## 2. MATLAB 6.0 语言新特点

### (1) 数据类型和面向对象编程技术

MATLAB 6.0 与旧版最显著的不同在于数据类型的变化。MATLAB 6.0 现有六种基本数据类型:双精度数组、字符串数组、元胞数组、构架数组、稀疏矩阵和 unit8 数据。

数据类型的变革,面向对象编程技术的采用,所产生的影响之一是广泛而深层的。这种影响首先表现在 MATLAB 的自身。从 5.0 版起, MATLAB 就用新数据类型逐步地对其自身的函数指令加以改造。这个过程一直延续到 5.3 版才基本完成。比如 5.3 版就推出了一组名称全新(求取极小值等)的泛函指令,它们优化参数的设置是采用构架数组进行的。再如 5.x 版提供的常微分方程解算指令 ODE Solver 的参数设置也全是靠新数据类型进行的。

新数据类型和面向对象技术的影响之二:若干通用工具包的相应升级。以符号计算为例,在 MATLAB 4.2c 中, Symbolic Math Toolbox 1.0 处理符号计算的指令形式与数值计算指令形式很不协调,显得十分生硬。比如,符号矩阵的“四则”运算的旧版指令分别是 `symadd`, `symsub`, `symmul` 和 `inverse`。但在 5.x 版中,符号工具包已升级为 2.0 版,新的“四则”符号运算指令形式上与数值计算完全相同,它们分别是 `+`, `-`, `*`, `/`。新的符号计算形式已被改造得与“MATLAB 风格数值计算形式”浑然统一。

新数据类型和面向对象技术影响之三:一系列的应用工具包相继升级。这不仅使应用工具包表现更为友善,而且功能大大加强。以控制工具包为例,新版利用构架数组和重载技术,把线性时不变系统(Linear Time - Invariant system)设计为“LTI 对象”。这样处理后,不管 LTI 是由传递函数产生、由零极点增益方式产生,还是由状态方程形式创建,只要是 LTI 对象,它们之间就可方便地进行各种数学运算。比如在控制教科书上,由前向控制环节  $G_1$ 、对象  $G_0$ 、反馈环节  $G_2$  组成的负反馈闭环系统传递函数  $G = \frac{G_0 G_1}{1 + G_0 G_1 G_2}$ 。假如  $G_0$ 、 $G_1$ 、 $G_2$  都是用传递函数,那么读者“手算”时,必须进行多项式展开、乘法、合并、简化等运算。如果利用“旧版 MATLAB”求  $G$ ,那末必须先使用 `series` 指令把  $G_0$ 、 $G_1$  串接成“中间”函数  $G_{01}$ ,然后再用 `feedback` 指令由  $G_{01}$ 、 $G_2$  算得  $G$ 。然而“新版 MATLAB”只需直截了当地用一条指令  $G = G_0 * G_1 / (1 + G_0 * G_1 * G_2)$ ,并且式中的  $G_0$ ,  $G_1$ ,  $G_2$  的表达形式可以各不相同,任取传递函数、零极点增益、状态方程等形式。当然 LTI 的优点远不止于此,比如它还可直接对多输入多输出系统进行统一运算,而无须分解成若干个子系统进行。

### (2) 控制流和函数类型

新版 MATLAB 的控制流新增了多分支结构 `switch - case`、`try - catch` 结构和警告提示指令 `error`、`warning`,进一步提高了程序的可读性和运行可靠性。

新版的函数类型很丰富,适应编制和管理复杂程度不同的程序。例如内联函数比较简练,适用于各类比较简单数学模型。而子函数、私用函数的增添,使得复杂函数比较容易组织,既提高了软件的“重用度”,又避免了众多内存变量名的冲突和庞大工具库的函数名冲突。

为函数设计了新的变长度输入输出宗量 `varargin`、`varargout`。采用了这种变长度宗量, MATLAB 自身的新版指令被进一步“柔性化”。一个指令可以接受任意多个输入宗量,可以产生任意多个输出宗量,以适应不同场合的需要。所有这些措施使得 MATLAB 能更加便捷地编制复杂的大型程序。当然,用户也可以借助这种变长度宗量来编制灵活多变的应用程序。

## 1.2.2 MATLAB 的工作环境

所谓工作环境是指:帮助系统、工作内存管理、指令和函数管理、搜索路径管理、操作系统、程序调试和性能剖析工具等。

### 1. 传统工作环境

与同时期其他数学类软件相比,旧版 MATLAB 的工作环境虽属比较友善之列,但其工作环境确实比较“单调”。它的帮助系统是“纯文本”形式的;内存管理、路径管理、调试工具是单纯指令操纵形式的;文件类型也形式单一,仅有 M 文件和 MAT 文件。4.2c 版情况开始变化,但那只是过渡形式。

### 2. MATLAB 6.0 工作环境新特点

#### (1) 大量引入图形用户界面

MATLAB 6.0 改变了过去单调依靠“在指令窗通过纯文本形指令进行各种操作”面貌,引入了许多让使用者一目了然的图形界面,如在线帮助的交互型界面 `helpwin`,管理工作内存的 `workspace`,交互式的路径管理界面 `pathtool`,指令窗显示风格设置界面等。它们的开启方式有:工具条图标开启、选择菜单项开启,直接“文本式”指令开启。

MATLAB 6.0 更进一步把图形显示窗改造成了交互操作的可编辑图形界面。

#### (2) 引入了全方位帮助系统

“临场”在线帮助:这些帮助内容,大多嵌附在 M 文件中,即时性强,反应速度快。它对求助内容的回答最及时准确。MATLAB 旧版就一直采用这种帮助系统,并深受用户欢迎。新版保留原功能的同时,还新增一个内容与之完全对应的图形界面 `helpwin`,加强了对用户的引导。

综合型在线帮助文库 `helpdesk`:该文库以 HTML 超文本形式独立存在。整个文库按 MATLAB 的功能和核心内容编排,系统性强,且可以借助“超链接”方便地进行交叉查阅。但是,这部分内容偶而会发辏与真实 M 文件脱节的现象。

完整易读的 PDF 文档:这部分内容与 HTML 帮助文库完全对应。PDF 文档不能直接从指令窗中开启,而必须借助 Adobe Acrobat Reader 软件阅读。这种文件的版面清楚、规范,适宜有选择地系统阅读,也适宜于制作硬拷贝。

演示软件 `demo`:这是一个内容广泛的演示程序。MATLAB 一向重视演示软件的设计,因此无论 MATLAB 旧版还是新版,都随带各自的演示程序。但是,新版内容更丰富了。

### (3) M 文件编辑、调试的集成环境

新的编辑器有十分良好的文字编辑功能,它可采用色彩和制表位醒目地区分标识程序中不同功能的文字,如运算指令、控制流指令、注释等。通过编辑器的菜单选项可以对编辑器的文字、段落等风格进行类似 Word 那样的设置。

从 5.2 版起,还新增了“变量现场显示”功能,只要把鼠标放在变量名上(Mouse over),就能在现场显示该变量的内容。

在 MATLAB 6.0 中,调试器已经被图形化,它与编辑器集成为一体,只需点动交互窗上的调试图标就可完成对程序的调试。

### (4) M 文件的性能剖析

调试器只负责 M 文件中语法错误和运行错误的定位,而性能剖析指令 profile 将给出程序各环节的耗时分析报告。MATLAB 6.0 剖析指令的分析报告特别详细,它将帮助用户寻找影响程序运行速度的“瓶颈”所在,以便改进。

### (5) Notebook 新的安装方式

从 4.2c 版引入 Notebook 以来,这种集文字、计算、图形于一体的“活”环境就深受用户赞赏。但直到 5.2 版,Notebook 的安装都是与 MATLAB 的安装同步进行的。这种安装方式的不便之处是:一旦 Word 发生变动,就必须把 MATLAB 全盘重装。MATLAB 6.0 改变了这种局面,它可以在 MATLAB 指令窗中“随时”进行 Notebook 安装,省时灵活。

### (6) MATLAB 环境可运行文件的多样化

旧版中,用户可编制和运行的程序文件只有 M 脚本文件和 M 函数文件。5.x 版新增了产生伪代码 P 文件的 pcode 指令和产生二进制 MEX 文件的 mex 指令。较之 M 文件,这两种文件的运行速度要快得多,保密性也好。

## 1.2.3 MATLAB 的视图系统

### 1. 传统的图形表现力

MATLAB 的图形可视能力在所有数学软件中是首屈一指的。MATLAB 的图形系统有高层和低层两个部分组成。高层指令友善、简便;低层指令细腻、丰富、灵活。

一般说来,不管二元函数多么复杂,它的三维图形,仅需 10 条左右指令,就能得到富于感染力的表现。数据和函数的图形可视手段包括:线的勾画、色图使用、浓淡处理、视角选择、透视和裁剪。MATLAB 有比较完备的图形标识指令,它们可标注:图名、轴名、解释文字和绘画图例。

### 2. MATLAB 6.0 的视图新特点

#### (1) MATLAB 6.0 的可编辑图形窗

对一般用户来说,在使用 MATLAB 6.0 图形功能时,感受最强烈的变化是图形窗。此前的图形窗只具单纯的显示功能,而 MATLAB 6.0 则不同,它是可编辑的图形显示窗。在 5.3 版的图形窗里,只需点动工具图标或菜单选项,就可直接对显示图形的各种“对象属性”进行随心所欲的设置,可交互式地改变线条型式、粗细、颜色,可动态地变换观察视角,可在图形窗随意位置标识文字或子图。MATLAB 6.0 的图形窗是十分成功的“图柄”操作的图形用户界面。

## (2) MATLAB 6.0 的 Tex 特殊字符集

图形功能的另一个较大变化是标识能力的大幅增强。具体表现:一,引进 Tex 特殊字符集,可标注如  $\alpha, \beta, \Sigma$  等数学字符;二,可书写上下标;三,可对英文、中文进行字体形式和大小的设置;四,可采取多种方式进行多行文字注释。

## (3) MATLAB 6.0 的简捷绘图指令

这组指令的特点是:“指令的前两个字母是 ez”,英文含义是“Easy to”。这组指令有两个功能:一,直接表现用字符串描写的函数图形;二,与符号计算配套使用,作为符号计算结果的图形可视工具。

这种指令的使用方法极其简单。例如使用一条指令 `ezsurf('y/(1+x^2+y^2)')` 就可以绘制二元函数  $z = \frac{y}{1+x^2+y^2}$  的曲面。

这组指令与普通“数值型”绘图指令起着互为补充的作用。假若就方便易用排序,简捷指令最方便,普通“数值型”绘图指令次之,低层指令最繁;假若就绘图的细致和个性化能力排序,那末低层指令最强,简捷指令最弱。

## (4) MATLAB 6.0 增强了高层绘图指令的排版能力

新版在同一图形窗口中可设置大小不同、非等距排列的任意个子图,而旧版只能开设面积等分的子图。新版可以在同一图形中使用两套不同的坐标系统,而旧版则不能。

## (5) MATLAB 6.0 新增的其他高层绘图指令

增添了主要用于表现统计数据的面域图 `area`, 水平直方图 `barh`, 三维直方图 `bar3`, `bar3h`, 二维、三维饼图 `pie`, `pie3`, 三维杆图 `stem3` 等;新增了四维数值表现力更强的切片等位线图 `slicecontour`;改造了切片图 `slice`, 允许任意设置切面;新增了表现不规则数据点的三维网线和曲面图 `trimesh`, `trisurf`;新增了若干色图函数,如 `spring`, `summer`, `autumn`, `winter` 等;增加了表现数据点的 8 种新“点型”。

## (6) MATLAB 6.0 读写图像文件能力的加强

旧版 MATLAB 能读写的图像文件类型比较狭窄。新版能够读写的文件格式有: `bmp`, `hdf`, `jpg`, `jpeg`, `pex`, `tif`, `tiff`, `xwd` 等。这无疑为进一步开拓图像处理方面的应用程序提供了更好的条件。

## (7) MATLAB 6.0 低层指令结构的改变和能力的加强

“轴”对象上新增的“子对象”`light`。该对象的增设,再配合增强了的光照模式 `lighting` 和控制光线反射的材质指令 `material`, 使得图形表现具真实感。

“轴”新增的照相机属性和投影属性,能更好地满足人们的视觉要求。

“面”对象的面色属性可以采用纹理影射技术,从而可以在各种形状曲面上彩绘各种图像,或表现表面的凹凸不平、材料纹路。

## (8) MATLAB 6.0 的图形用户界面 GUI 制作工具

从 4.2c 版起, MATLAB 就开始向用户提供制作 GUI 的指令,但十分稚嫩。随 5.0, 5.1, 5.2 版的升级, GUI 制作工具不断改进。现在 MATLAB 6.0 中:不仅可以制作位置固定的用户菜单 `uimenu`, 而且可以制作位置不固定的“现场”菜单 (`Context menu`); 用户控件 `uicontrol` 已增加到 10 种;不管是菜单,还是控件,都可以进行“使能”和“可见性”控制。

MATLAB 向用户提供两种制作 GUI 的途径:依靠指令制作 GUI;借助交互式工具

guide 制作 GUI。这两种方法各有优缺点:前者灵活、细致;后者直观、全局观念强。用户交替运用这两种制作手段,可高效地制作 GUI,开发出各种生动活泼的应用程序。

## 1.2.4 MATLAB 的数学函数库

### 1. 世界一流水平的数值计算函数库

MATLAB 自问世起,就抱定一个宗旨:其所有数值计算算法都必须是国际公认的、最先进的、可靠算法;其程序由世界一流专家编制,并经高度优化;而执行算法的指令形式则必须简单、易读易用。MATLAB 正是仰赖这些高质量的数值计算函数赢得了声誉。

MATLAB 数值计算函数库的另一个特点是其内容的基础性和通用性。它正由于这一特点,而适应了诸如自动控制、信号处理、动力工程、电力系统等应用学科的需要,并进而开发出一系列应用工具包。

在整个 MATLAB 的发展过程中,这数值计算函数库,从内容到形式,变化最小。

### 2. MATLAB 6.0 函数库的变化

#### (1) MATLAB 6.0 新增的常微分方程解算程序

MATLAB 6.0 MATLAB 数值计算方面的最大变化是增添了一组常微分方程数值解算程序 ODE Solver。这组解算程序无论是算法还是软件结构都十分精良,它包含 ode23, ode45, odel13, ode23t, odel5s, ode23s, ode23tb 等不同解算指令,用以解算包括 Stiff 方程在内的各种微分方程。

MATLAB 6.0 为解 ODE 问题所设计的文件十分严整,包括解算指令 Solver、被 Solver 调用的微分方程描述文件、进行积分算法参数设置的 odeset 和 odeget、解算输出指令 odeplot, odephas2 等。

#### (2) MATLAB 6.0 新增的其他数值计算指令

MATLAB 6.0 版新增的许多计算指令与数值计算方面的最新成就直接相关。举例来说,新版及时地增添了广义奇异值计算指令 gsvd;高维快速傅里叶变换和反变换 fftn, ifftn;高维插值指令 interpn 等。

### 3. MATLAB 6.0 的符号计算工具包

关于符号计算和数值计算优劣的争论曾经历过一段时间,但那是 20 世纪 90 年代以前的事。MathWorks 公司打破门户之见,把 Maple 的内核和数学函数库引入了 MATLAB,从而使 MATLAB 具有了数值和符号双重计算能力。用户可以视具体问题而进行适当的选择。比如,对于比较复杂的“初值类”非线性微分方程,有时符号计算或无法解、或求解时间太长,而数值算法却比较有效;反之,对于边值类微分方程,数值算法的实现可能比较繁琐,而符号计算有时倒比较简便。

Maple 的函数库十分庞大,包含 2000 多个函数。它几乎囊括了一般用户所需的所有函数。与 5.x 版 MATLAB 配用的 Symbolic Math Toolbox 2.0 允许用户在三个不同层次上做符号计算。第一层次是,在进行符号对象定义后,直接利用 MATLAB 格式进行矩阵分解、微分、积分、积分变换、代数方程求解、微分方程求解等运算。第二层次是,借助 Maple 指令,把单个 Maple 格式的指令送往 Maple 引擎计算。第三层次是,借助 procread 把整段 Maple 程序送往 Maple 计算。



### 1.2.5 MATLAB 与外部程序的交互

MATLAB 的卓越性能引发了用户的新需求:希望把在 MATLAB 环境中编制的程序开发成能脱离 MATLAB 独立运行的程序;希望能在外部程序中调用 MATLAB 作为计算引擎。需求决定商品,市场也真出现了诸如 Mediva 等商品软件,能把 MATLAB 的 M 文件转变为独立于平台的 EXE 可执行文件;出现许多专用软件把 MATLAB 直接当计算引擎使用。鉴于此, MATLAB 在 4.x 版时代的后期,就尝试性地推出了自己的编译器。

#### 1. 与 MATLAB 6.0 配用的编译器 Compiler 3.0

伴随 MATLAB 5.0 向 MATLAB 6.0 升级的过程中,变化较大,更新较快的是 MATLAB 编译器。与 5.2 版配用的是 Compiler 1.2 版,而与 MATLAB 6.0 配用的则是 Compiler 3.0 版、2.0 版。

无论是 1.2 版编译器,还是 2.0 版编译器,它们都不但可以把全 M 函数文件编译成独立应用程序,而且也可以把 C 或 FORTRAN 程序与 M 文件混编成独立应用程序。这种程序的优点是:一,可以脱离 MATLAB 环境独立运行;二,运行速度快。

MATLAB 编译器无疑给用户开发计算类 EXE 可执行程序提供了快捷、高效的工具。举例来说,假如用户想编制一个求解各类线性代数方程的可独立执行的程序,用户只要先在 MATLAB 环境中编写由 30 来条指令组成的 M 函数文件,然后借助编译器把它变成独立执行的 EXE 程序。这 EXE 文件不但可以计算“恰定”方程,而且可以解算“超定”、“欠定”方程。值得指出:原 M 文件是不过 2K 字节的小小程序,而编译生成的 EXE 文件却超过 200K 字节,这可是一个不算小的程序。

与 1.2 版相比,3.0 版编译器具有处理高维数组、元胞数组、构架数组的能力,支持变长度输入输出宗量,支持多分支等控制流。

#### 2. MATLAB 6.0 的 API 应用程序接口

与 MATLAB 编译器相比, MATLAB 的 API 应用程序接口问世得更晚,也更不成熟。MATLAB API 由一系列接口指令组成。借助这些接口指令,用户就可在 C 或 FORTRAN 中,或直接读写 MATLAB 的 MAT 数据文件,或把 MATLAB 当作计算引擎使用。

## 1.3 与 MATLAB 6.0 配用的 Simulink 3.0

Simulink 是 MathWorks 公司开发的又一个产生重大影响的软件产品。它的前身 SIMULIB 问世于 20 世纪 90 年代初,以工具库的形式挂接在 MATLAB 3.5 上。以 Simulink 名称广为人知,是在 MATLAB 4.2x 时期。Simulink 不能独立运行,而只能在 MATLAB 环境中运行。现在较为流行的有:与 MATLAB 5.2 配用的 Simulink 2.2;与 MATLAB 5.3 配用的 Simulink 3.0。

### 1.3.1 Simulink 的传统优点

不管是什么版本, Simulink 总由模块库、模型构造及分析指令、演示程序等三部分组成。在 Simulink 环境中,对于由微分方程或差分方程描写的动态系统,用户无须编写文本形式的程序,而只要通过一些简单的鼠标操作就可形象地建立起被研究系统的数学模