MATLAB及其应用

MATLAB and Its Application

武汉大学电子信息学院光电信息工程系 whuchj@163.com 027-63444539

学习目的:

- 掌握用计算机建模、计算,以及用图表形象表达分析结果从而促进逻辑思维的系统方法。
- 缩短研究周期,加快探索步伐,激发创作灵感。
- 不希望学生在编程上化太多时间,课程目的不是学习编程。
- 紧密结合科学的前沿。
- 论文=Matlab+word。

MATLAB 的特点:

- 编写程序犹如在演算纸上排列出公式与求解问题。
- 语法非常简单、编程效率高、易学易懂、方便实用。没有编程基础的学生也可以很快地编写出程序。
- 可以直接在 Word 中被调用,使用相当方便实用。
- 强大的科学计算能力、卓越的可视化功能、开放式可扩展环境。不同的应用学科 几十种应用工具箱
 - a. 研究人员可以直观、方便地进行分析、计算和设计工作,
 - b. 应用工具箱可以更进一步简化编程,甚至给很复杂的问题提供"**傻瓜照** 相式"的解决办法,极大地提高了工作效率。
- 可实现 MATLAB与 Visual C++等通用编程平台的混合编程。取长补短, 以提高程序的运行效率,丰富程序开发的手段。

MATLAB 的普及:

- 经历了**实践的检验、市场的筛选和时间的凝练**,日益融入到数学、物理、化学、信息工程、生物医学工程、金融、管理和社会科学等许多学科专业。
- 在大学、科研机构和各大公司中,MATLAB 得到了广泛的应用,已经成为 计算机辅助设计和分析、科学计算和应用开发的基本工具。
- 在当今欧美大学的许多专业中,MATLAB 已成为大学生必须掌握的科学计算软件。

课程主要内容:

基础准备及入门 符号计算 数值数组及向量化运算 数值计算 数据和函数的可视化 M文件 Notebook

.

教学特点:

算例的计算结果(包括数据和图形等)**现场**实时产生感受计算过程和计算结果

教 材:

《MATLAB 教程》R2008a(2009年修订) 2009年第7(或第8)次印刷 张志涌等编著,北京航空航天大学出版社,2006年第1版

参考文献:

注:请按照不同专业需要来选择。

- [1]《MATLAB及其在理工课程中的应用指南》(第二版),陈怀琛编著,西安电子科技大学出版社,2004
- [2]《MATLAB 在化学中的应用》,许国根、许萍萍、谭宪林编著,西安交通大学出版社,2005
- [3]《MATLAB 金融计算与金融数据处理》, 张树德编著, 北京航空航天大学出版社, 2008
- [4]《MATLAB 在动态经济学中的应用》,王翼、王歆明编著,机械工业出版社, 2006
- [5]《管理运筹学和 MATLAB 软件应用》, 刘顺忠编著, 武汉大学出版社, 2007
- [6]《基于 MATLAB 6.x 的系统分析与设计 神经网络》(第二版),许东、吴 铮编著,西安电子科技大学出版社,2002
- [7]《MATLAB基础与应用实例集粹》,王琦、徐式蕴、赵睿涛、高军锋、常春藤编著,人民邮电出版社,2007

网上资源:

国内中文站点:

1. MATLAB 教学网 http://matlab.net.cn

国外站点:

- 1. MathWorks 公司主页 http://www.mathworks.com
- 2. MAThTools 站点 http://www.mathtools.net/MATLAB/index.html

第 1 章 基础准备及入门

- 一 MATLAB 正常运行所必须具备的基础条件;
- 二 MATLAB及其操作桌面的基本使用方法;
- 三 MATLAB 的帮助系统。

1.1 MATLAB 的安装

推荐使用 MATLAB 7 版本

例如: MATLAB 7.0, ..., MATLAB 7.3 (MATLAB R2006b), MATLAB 7.4 (R2007a), ..., R2008a, R2008b,

MathWorks公司每年进行两次产品发布,时间分别在每年的3月和9月。

MATLAB 只有在适当的外部环境中才能正常运行。

如: MATLAB R2008a (即旧编号 MATLAB7.6) 版要求 Win2000 或 WinXP 平台。

在安装 MATLAB.R2008a 时,会出现一个界面,该界面上有两个选项: Typical 和 Custom。假如你不熟悉 MATLAB,或计算机硬盘的自由空间远大于 3G,或假如你需要用到光盘上 MATLAB 的所有功能及工具包,那么你就点选"Typical"。否则,点选"Custom"。

在点选"Custom"后,会引出如图 1.1-1 的界面。

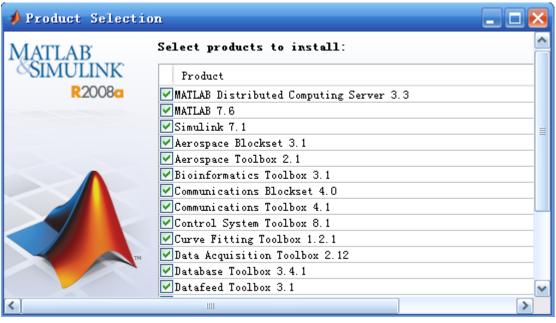


图 1.1-1 MATLAB 组件选择界面

表 1.1-1 MATLAB 各组件的功用

X III-I WATERD (LEITHOWN)			
组件名称	功用		
1. 必须选择的本原性组件			
MATLAB	这是最核心的部分。没有它就没有 MATLAB 环境。		
	有了它,就可以对各类数据(除符号类数据以外)		
	进行操作、运算和可视化。		
2. 最常选的通用性工具包组件			
Symbolic Math Toolbox	符号类数据的操作和计算。		
Extended Symbolic Math Toolbox	更丰富的符号计算函数和指令		
3. 其他通用性工具包组件			
Simulink	不用编写程序,利用方块图实现建模和仿真;主要		
	用于研究微分和差分方程描写的非线性动态系统。		
Optimization Toolbox	包含求函数零点、极值、规划等优化程序。		
Matlab Compiler	把 MATLAB 的 M 文件编译成独立应用程序。		
Matlab builder for Excel	与 MATLAB Compiler 配合使用,生成 Excel 插件。		
4. 常用专业性工具包组件			
Control System Toolbox	MATLAB 涉及控制的工具包有 10 多个,而本栏工具		
	包是最基本的,是解决控制问题所必选的。至于其		
	他控制工具包则是用户根据需要选择的。		
Signal Processing Toolbox	是 MATLAB 信号处理中的基本工具包。		
Spline Toolbox	内含样条和插值函数。		
Statistics Toolbox	包含进行复杂统计分析所需的程序。		
5. 其他专业性工具包组件(举例)	_		
Stateflow	与 Simulink 配合使用,主要用于较大型、复杂的(离		
	散事件) 动态系统的建模、分析、仿真。		
System Identification Toolbox	动态系统辨识。		

1.2 Desktop操作桌面的启动

1.2.1 MATLAB 的启动

点击图标

1.2.2 操作桌面简介

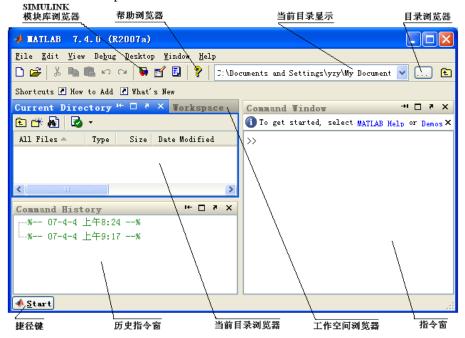
桌面的上层铺放着的界面:

指令窗(Command Window)、

当前目录(Current Directory)浏览器、

历史指令(Command History)窗。

工作空间 (Workspace) 浏览器。



● 指令窗

指令、函数、表达式;

显示除图形外的所有运算结果;

给出相关的出错提示。

● 历史指令窗

记录已经运作过的指令、函数、表达式

运行的日期、时间

所有指令、文字都允许复制、重运行及用于产生M文件。

● 当前目录浏览器

展示着子目录、M文件、MAT文件和 MDL文件等。

可进行 Windows 平台的各种标准操作。

● 工作空间浏览器

罗列出 MATLAB 工作空间中所有的变量名、大小、字节数;

可对变量进行观察、图示、编辑、提取和保存。

● 捷径 (Start) 键

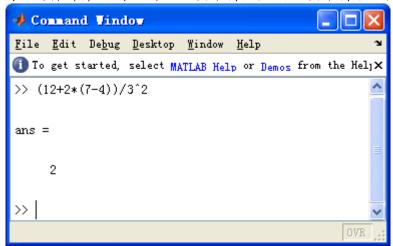
引出通往 MATLAB 所包含的各种组件、模块库、图形用户界面、帮助 分类目录、演示算例等的捷径,以及向用户提供自建快捷操作的环境。

1.3 Command Window 运行入门

直观感受 MATLAB 指令窗的使用方法

1.3.1 Command Window指令窗简介

MATLAB 指令窗默认地位于 MATLAB 桌面的右方。 希望得到脱离操作桌面的几何独立指令窗,点击该指令窗右上角的 型键。



● 让独立指令窗嵌放回桌面,点击 Command Window 右上角的 ¥按钮。

1.3.2 最简单的计算器使用法

【例 1.3-1】求 [12+2×(7-4)]÷3² 的算术运算结果。 本例演示: 最初步的指令输入形式和必需的操作步骤。

 $(12+2*(7-4))/3^2$

〖说明〗

- 一条指令输入结束后,必须按 [Enter] 键,那指令才被执行。
- 由于本例输入指令是"不含赋值号的表达式", 所以计算结果被赋给 MATLAB 的一个默

【例 1.3-2】"续行输入"法

本例演示: 由于指令太长,或出于某种需要,输入指令行必须多行书写时,该如何处理?

S=1-1/2+1/3-1/4+ ... 1/5-1/6+1/7-1/8

[说明]

● 指令执行后,变量 S被保存在 MATLAB 的工作空间(Workspace)中,以备后用。

1.3.3 数值、变量和表达式

一 数值的记述

MATLAB 的数值采用习惯的十进制表示,可以带小数点或负号。以下记述都合法。 3 -99 0.001 9.456 1.3e-3 4.5e33

二 变量命名规则

● 变量名、函数名是对字母大小写敏感的。

如:变量 myvar 和 MyVar 表示两个不同的变量。 sin 是 MATLAB 定义的正弦函数名,但 SIN,Sin 等都不是。

● 变量名的第一个字符必须是英文字母,最多可包含 63 个字符(英文、数字和下连符)。

如: myvar201 是合法的变量名。

● 变量名中不得包含空格、标点、运算符,但可以包含下连符。 如变量名 my_var_201 是合法的,且读起来更方便。 而 my,var201 由于逗号的分隔,表示的就不是一个变量名。

三 MATLAB 默认的预定义变量 (Predefined Variable)

表 1.3-1 MATLAB 中最常用的预定义变量

预定义变量	含义	预定义变量	含义
ans	计算结果的默认变量名	NaN 或 nan	不是一个数(Not a Number),
eps	机器零阈值		如 0/0, ∞/∞
Inf 或 inf	无穷大,如 1/0	nargin	函数输入宗量数目
i 或 j	虚单元 $i = j = \sqrt{-1}$	nargout	函数输出宗量数目
pi	圆周率 π	realmax	最大正实数
		realmin	最小正实数

【例】

format long

рi

1/0

0/0

realmax

realmin

eps % 在采用IEEE浮点算法的计算机上,数值通常采用"占用 64 位内存的双精度"表示。 其相对精度是 eps。

说明:

- 建议:用户在编写指令和程序时,应尽可能不对表 1.3-1 所列预定义变量名重新赋值,以免产生混淆。
- 被 0 除是允许的。它不会导致程序执行的中断,只是在给出警告信息的同时,用一个 特殊名称(如 Inf, NaN)记述。

四 运算符和表达式

(1) 经典教科书上的算术运算符在 MATLAB 中的表达方式

表 1.3-2 MATLAB 表达式的基本运算符

	数学表达式	矩阵运算符	数组运算符
加	a+b	a + b	a + b
减	a-b	a - b	a - b
乘	$a \times b$	a * b	a .* b
除	$a \div b$	a/b 或b∖a	a./b 或b.\a
幂	a^b	a ^ b	a .^ b
圆括号	()	()	()

[说明]

- 因为 MATLAB 面向矩阵/数组设计, 标量被看作 (1×1) 的矩阵/数组。
- 数组运算的"乘、除、幂"规则与相应矩阵运算根本不同。数组运算符比矩阵运算符多一个"小黑点"。
- MATLAB用左斜杠或右斜杠分别表示"左除"或"右除"运算。 对标量而言,"左除"和"右除"的作用结果相同。 但对矩阵来说,"左除"和"右除"将产生不同的结果。
- (2) MATLAB 书写表达式的规则与"手写算式"几乎完全相同。

- 表达式由变量名、运算符和函数名组成。
- 表达式将按与常规相同的优先级自左至右执行运算。
- 优先级的规定是:指数运算级别最高,乘除运算次之,加减运算级别最低。
- 括号可以改变运算的次序。
- 书写表达式时,赋值符"="和运算符两侧允许有空格,以增加可读性。

五 面向复数设计的运算——MATLAB特点之一

为描述复数,虚数单位用预定义变量i或i表示。

● MATLAB 的所有运算都是定义在复数域上的。

好处: 在进行运算时,不必像其他程序语言那样把实部、虚部分开处理。

复数 $z = a + bi = re^{i\theta}$ 直角坐标表示和极坐标表示之间转换:

real(z) 给出复数 z 的实部 $a = r \cos \theta$ 。

imag(z) 给出复数 z 的虚部 $b = r \sin \theta$.

abs(z) 给出复数 z 的模 $\sqrt{a^2+b^2}$ 。

angle(z) 以弧度为单位给出复数 z 的幅角 $arctan \frac{b}{a}$

复数与平面上的点?

【例 1.3-4】复数 $z_1 = 4 + 3i$, $z_2 = 1 + 2i$, $z_3 = 2e^{\frac{\pi}{6}i}$ 表达,及计算 $z = \frac{z_1 z_2}{z_3}$ 。

本例演示: 正确的复数输入法; 涉及复数表示方式的基本指令。

(1) 经典教科书的直角坐标表示法

z1=4+3i %合法, 但建议少用或不用

[说明]

- 在这种书写格式中, 3i是一个完整的虚数, 在3和i之间不许"空格"存在。
- (2) 采用运算符构成的直角坐标表示法和极坐标表示法

 $z^2 = 1 + 2 * i$

%运算符构成的直角坐标表示法

z3=2*exp(i*pi/6)

%运算符构成的极坐标表示法

z=z1*z2/z3

(3) 复数的实虚部、模和幅角计算

real z=real(z)

image z=imag(z)

magnitude_z=abs(z)

angle z radian=angle(z)

%弧度单位

angle_z_degree=angle(z)*180/pi

%度数单位

【例 1.3-5】图示复数 $z_1 = 4 + 3i, z_2 = 1 + 2i$ 的和。

本例演示:

MATLAB 的运算在复数域上进行;

指令后"分号"的作用;

复数加法的几何意义;

展示 MATLAB 的可视化能力(仅感受,但不要求理解)。

z1=4+3*i;z2=1+2*i;

%在一个物理行中,允许输入多条指令。 %但各指令间要用"分号"或"逗号"分开。 %指令后采用"分号",使运算结果不显示。

z12=z1+z2;

%以下用于绘图

clf,hold on

%clf清空图形窗。逗号用来分隔两个指令。

plot([0,z1,z12],'-b','LineWidth',3)

plot([0,z12],'-r','LineWidth',3)

plot([z1,z12],'ob','MarkerSize',8)

hold off,grid on,

axis equal

axis([0,6,0,6])

text(3.5,2.3,'z1')

text(5,4.5,'z2')

text(2.5,3.5,'z12')

xlabel('real')

ylabel('image')

shg

class 1 end

【例 1.3-6】用 MATLAB 计算 $\sqrt[3]{-8}$ 能得到 -2 吗?

本例演示: MATLAB 的方根运算规则;

更复杂指令的表示方式;

展现 MATLAB 的图形表现力。

(对于本例指令,读者能有体验就可,不必理解。)

(1) 直接计算时,得到处于第一象限的方根。

a=-8:

r_a=a^(1/3) %求3次根

(2) $\sqrt[3]{-8}$ 的全部方根计算如下

% 先构造一个多项式 $p(r) = r^3 - a$

p=[1,0,0,-a]; % p是多项式 p(r) 的系数向量

%指令末尾的"英文状态分号"使该指令运行后,不显示结果。

R=roots(p) %求多项式的根

(3)图形表示

MR=abs(R(1)); %计算复根的模

t=0:pi/20:2*pi; %产生参变量在 0 到 2*pi间的一组采样点

x=MR*sin(t);
y=MR*cos(t);

plot(x,y,'b:'),grid on % 画一个半径为R的圆

hold on

plot(R(2),'.','MarkerSize',30,'Color','r') % 画第一象限的方根

plot(R([1,3]),'o','MarkerSize',15,'Color','b') %画另两个方根

axis([-3,3,-3,3]),axis square %保证屏幕显示呈真圆

hold off

六 面向数组设计的运算——MATLAB特点之二

标量数据被看作(1×1)的数组(Array)数据。

所有的数据都被存放在适当大小的数组中。

为加快计算速度(运算的向量化处理), MATLAB对以数组形式存储的数据设计了两种基本运算:

- 一 数组运算
- 二 矩阵运算

【例 1.3-7】实数数组
$$\mathbf{AR} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$$
 的"一行"输入法。

本例演示: 二维数组的最基本、最常用输入法;

二维数组输入的三大要素。

(1) 在键盘上输入下列内容

AR= [1,3;2,4]

(2) 按 [Enter] 键,指令被执行。

[说明]

- 在MATLAB中,不必事先对数组维数及大小做任何说明,内存将自动配置。
- 二维数组输入的三大要素:

数组标识符"[]";

元素分隔符为: 空格或逗号",";

数组行间分隔符 分号";"或"回车键"。

注意: 所有标点符号都是"英文状态的符号"。

● MATLAB 对字母大小写是敏感的。

比如本例中的数组赋给了变量 AR, 而不是 Ar, aR, 或 ar。

● 在全部键入一个指令行内容后,必须按下 [Enter] 键,该指令才会被执行。

【例 1.3-8】实数数组
$$\mathbf{AI} = \begin{bmatrix} 5 & 7 \\ 6 & 8 \end{bmatrix}$$
的"分行"输入法。

AI=[5,7 6,8]

〖说明〗

更符合视觉习惯。

● []内"回车键"用来分隔数组中的行

【例 1.3-9】对复数数组
$$\mathbf{A} = \begin{bmatrix} 1-5i & 3-7i \\ 2-6i & 4-8i \end{bmatrix}$$
进行求实部、虚部、模和幅角的运算。

本例演示: 复数数组的生成;

MATLAB 指令对数组元素"并行操作"的实质。

(1) 创建复数数组

AR=[1,3;2,4];AI=[5,7;6,8];

A=AR-AI*i

%形成复数矩阵

(2) 求复数数组的实部和虚部

A real=real(A)

 $A_{image=imag}(A)$

(3) 求复数数组中各元素的模和幅角——循环法(笨拙!)

for m=1:2

for n=1:2

Am1(m,n)=abs(A(m,n));

Aa1(m,n)=angle(A(m,n))*180/pi; %以度为单位计算幅角

end

end

Am1,Aa1

(4) 求复数数组中各元素的模和幅角——直接法

Am2=abs(A)

Aa2=angle(A)*180/pi

〖说明〗

- 函数 real, imag, abs, angle 是同时、并行地作用于数组的每个元素。对 4 个元素运算所需的时间大致与对单个元素所需时间相同。这有利于运算速度的提高。这是"<mark>向量化"</mark>运算的一种形式。
- 本例给出了循环法求各元素模和幅角的指令。这是低效的计算方法。对于 MATLAB 以外的许多编程语言来说,可能不得不采用"循环"处理方式来解本例。

记住:对于 MATLAB 来说,应该尽量摒弃"循环"处理,而采用"向量化"处理方式。

【例 1.3-10】画出衰减振荡曲线 $y = e^{-\frac{t}{3}} \sin 3t$, t 的取值范围是[0,4 π]。

本例演示:展示数组运算的优点;展示 MATLAB 的可视化能力。

t=0:pi/50:4*pi; %定义自变量t的取值数组

y=exp(-t/3).*sin(3*t); %计算与自变量相应的y数组。注意: 乘法符前面的小黑点。

plot(t,y,'-r','LineWidth',2) %绘制曲线

axis([0,4*pi,-1,1])
xlabel('t'),ylabel('y')

〖说明〗

- 本例第二条指令中的".*"符号表示乘法是在**两个数组相同位置上的元素间**进行的。 这种乘法称为"数组乘"。
 - 数组乘的引入,不但使得程序简洁自然,而且避免了耗费机时的"循环计算"。
- 本例第二条指令是典型的"向量化"处理形式。

【例 1.3-11】复数矩阵 A·B 矩阵乘积

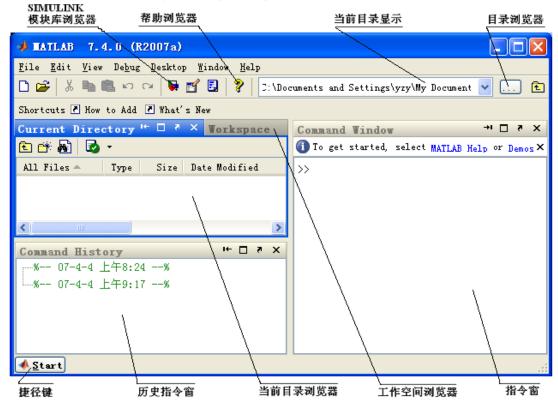
$$\mathbf{A} = \begin{bmatrix} 1 - 5i & 3 - 7i \\ 2 - 6i & 4 - 8i \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 3 + 2i & 2 + 6i \\ 5 + 3i & 4 - 2i \end{bmatrix}$$

本例演示: MATLAB 矩阵运算指令的简捷性。

A=[1-5*i,3-7*i;2-6*i,4-8*i]
B=[3+2*i,2+6*i;5+3*i,4-2*i] %复数数组的又一种输入方式 %注意标点符号的作用
C=A*B %矩阵乘法

[说明]

- 当数组被赋予"变换"属性时,数组就被称为矩阵。 只有当两个矩阵的"<mark>内维大小相等</mark>"时,矩阵乘法才能进行。 本例中,矩阵 A 的列数与矩阵 B 的行数相等,所以可以进行 A 乘 B。
- 从表达方式看,"矩阵相乘"的指令格式与"标量相乘"指令格式一样。 矩阵运算如同标准"线性代数"那样简洁易读、自然流畅。
- 在其他许多编程语言中,**矩阵乘法不得不依赖"循环"**进行。



● MATLAB 所拥有的丰富资源和友善灵活的环境特别适于用来验证一些思想,思考一些问题,和帮助进行创造性思维。用户可以在 MATLAB 环境中,边想边做,做做想想,对随时蹦出的思想"火花"可即刻通过计算加以验证。历史指令窗(Command History)就是为这种应用方式设计的。

- 当待解决问题所需的指令较多和所用指令结构较复杂时,或当一组指令通过 改变少量参数就可以被反复使用去解决不同问题时,直接在指令窗中输入指 令的方法就显得烦琐和笨拙。如何解决这个矛盾?
- 希望运用 MATLAB 资源,快捷、可靠、有效地独立解决自己面临的各种问题

但面对不断变化的实际问题,常常产生两类困惑:

- 1. 知道具体指令,但不知道该怎么用;
- 2. 想解某个具体问题,不知道 MATLAB 有哪些指令可用,如何用。

1.4 Command Window操作要旨

前一节 直观的感受。

本 节 对控制指令窗的指令和操作进行较系统的归纳

1.4.1 指令窗的显示方式

一 默认的输入显示方式

从 MATLAB7.0 起,指令窗中的字符、数值等采用更为醒目的分类显示: if, for, end 等控制数据流的关键词 蓝色字体 非控制指令、数码 黑色字体 字符串 紫色字体

二 运算结果的显示

在指令窗中显示的输出:

指令执行后,数值结果 黑色字体 运行过程中的警告信息和出错信息 红色字体

表 1.4-1	数据显示格式的控制指令

指令	含义	举例说明
	通常保证小数点后四位有效,最多不超	
format short	过7位;对于大于1000的实数,用5位	3141.59被显示为 3.1416 e+003
	有效数字的科学记数形式显示。	
format long	小数点后 15 位数字表示	3.141592653589793
format short e	5 位科学记数表示	3.1416e+00
format long e	15 位科学记数表示	3.14159265358979e+00

format short g	从 format short 和 format short e 中自动选	3.1416
	择最佳记数方式	
format long g	从 format long 和 format long e 中自动选	3.14159265358979
	择最佳记数方式	
format rat	近似有理数表示	355/113
format hex	十六进制表示	400921fb54442d18
format +	显示大矩阵用。正数、负数、零分别用+,	+
	-,空格表示。	
format bank	(金融) 元、角、分表示	3.14
format compact	显示变量之间没有空行	
format loose	在显示变量之间有空行	
7 2 H T T		_

〖说明〗

format short 显示格式是默认的显示格式。

- MATLAB 的数值数据通常占用 64位(Bit)内存,以 16位有效数字的"双精度"进行运算。
- 指令窗中直接输入相应的指令,可获得所需的数值计算结果显示格式。但仅对当前的 MATLAB 指令窗起作用,一旦 MATLAB 关闭,这种设置也就随之失效。

三 显示方式的永久设置

{File: Preferences}

选择字体风格、大小、颜色、数值计算结果显示格式等

[OK] 键完成设置

注意:该设置立即生效,并且这种设置将被永久保留,即这种设置不因 MATLAB 关闭和开启而改变,除非用户进行重新设置。

1.4.2 指令行中的标点符号

强调: 标点在 MATLAB 中的地位极其重要

表 1.4-2 MATLAB 常用标点的功能

名称	标点	作用
空格		(为机器辨认)用作输入量与输入量之间的分隔符;
工作		数组元素分隔符。
		用作要显示计算结果的指令与其后指令之间的分隔;
逗号	,	用作输入量与输入量之间的分隔符;
		用作数组元素分隔符号。
₩ F		数值表示中,用作小数点;
黑点	•	用于运算符号前,构成"数组"运算符。
		用于指令的"结尾",抑制计算结果的显示;
分号	;	用作不显示计算结果指令与其后指令的分隔;
		用作数组的行间分隔符。

		用以生成一维数值数组;
冒号	:	用做单下标援引时,表示全部元素构成的长列;
		用做多下标援引时,表示那维上的全部元素。
注释号	%	由它"启首"的所有物理行部分被看作非执行的注释
单引号对	, ,	字符串记述符
		改变运算次序;
圆括号	()	在数组援引时用;
		函数指令输入宗量列表时用。
方括号	r 1	输入数组时用;
力和力	[]	函数指令输出宗量列表时用。
花括号	{}	胞元数组记述符;
化石石	{ }	图形中被控特殊字符括号。
下连符		(为使人易读)用作一个变量、函数或文件名中的连字符;
下迁初	-	图形中被控下脚标前导符。
续行号		由三个以上连续黑点构成。它把其下的物理行看作该行的
※11	•••	"逻辑"继续,以构成一个"较长"的完整指令。
		放在函数名前,形成函数句柄;
"At"号	@	匿名函数前导符;
		放在目录名前,形成"用户对象"类目录。
DW and Vat	カルムー	the vike offers

〖说明〗为确保指令正确执行,以上符号一定要在英文状态下输入。因为 MATLAB 不能识别含有中文标点的指令。

1.4.3 指令窗的常用控制指令

表 1.4-3 常见的通用操作指令

指令	含义	指令	含义
cd	设置当前工作目录。	exit	关闭/退出 MATLAB
clf	清除图形窗	quit	关闭/退出 MATLAB
clc	清除指令窗中显示内容	more	使其后的显示内容分页进行
clear	清除 MATLAB 工作空间中保存		返回到上层调用程序; 结束键
	的变量	return	盘模式
dir	列出指定目录下的文件和子目	type	显示指定 M 文件的内容
	录清单		
edit	打开 M 文件编辑器	which	指出其后文件所在的目录

● 表 1.4-3 所列的指令是基本的,它们对 MATLAB 各版都适用。

1.4.4 指令窗中指令行的编辑

MATLAB 指令窗中实施指令行编辑的常用操作键

键名	作用	键名	作用
1	前寻式调回已输入过的指令行	Home	使光标移到当前行的首端
\	后寻式调回已输入过的指令行	End	使光标移到当前行的尾端
<	在当前行中左移光标	Delete	删去光标右边的字符
→	在当前行中右移光标	Backspace	删去光标左边的字符
PageUp	前寻式翻阅当前窗中的内容	Esc	清除当前行的全部内容
PageDown	后寻式翻阅当前窗中的内容		

[说明]

- 表 1.4-4 所列的操作对 MATLAB 各版均适用。
- 指令窗中输入过的所有指令都被显示在"**历史指令浏览器**"交互界面中,以供 随时观察和调用。

【例 1.4-1】指令行操作过程示例。

(1) 计算
$$y_1 = \frac{2\sin(0.3\pi)}{1+\sqrt{5}}$$
 的值,键入

y1=2*sin(0.3*pi)/(1+sqrt(5))

- (2) 按 [Enter] 键
- (3) 通过反复按键盘的箭头键,可实现指令回调和编辑,进行新的计算。

若又想计算
$$y_2 = \frac{2\cos(0.3\pi)}{1+\sqrt{5}}$$
,

先用 [↑] 键调回已输入过的指令 v1=2*sin(0.3*pi)/(1+sqrt(5));

然后移动光标,把yl改成y2;把sin改成cos;

再按[Enter]键,就可得到结果。即

$$y2=2*cos(0.3*pi)/(1+sqrt(5))$$

1.5 Command History历史指令窗

MATLAB所拥有的丰富资源和友善灵活的环境特别适于用来验证一些思想,思考一些问题,和帮助进行创造性思维。用户可以在 MATLAB 环境中,边想边做,做做想想,对随时蹦出的思想"火花"可即刻通过计算加以验证。历史指令窗(Command History)就是为这种应用方式设计的

1.5.1 Command History历史指令窗简介

历史指令窗记录着:每次开启 MATLAB 的时间, 开启 MATLAB 后在指令窗中运行过的所有指令行

表 1.5-1 历史指令窗主要应用功能的操作方法

应用功能	操作方法	简捷操作方法
单行或多行指令	点亮单行或多行指令;按鼠标右键引出现场菜单;选	
的复制	中{Copy}菜单项,即可用复合键 [Ctrl + V] 把它"粘贴"	
	到任何地方(包括指令窗)。	
单行指令的运行	点亮单行指令;按鼠标右键引出现场菜单;选中	鼠标左键双击
	{Evaluate Selection} 菜单项,即可在指令窗中运行,并	单行指令。
	见到相应结果。	
多行指令的运行	点亮多行指令;按鼠标右键引出现场菜单;选中	
	{Evaluate Selection} 菜单项,即可在指令窗中运行,并	
	见到相应结果。(详见例 1.5-1)	
把多行指令写成	点亮多行指令;按鼠标右键引出现场菜单;选中 {Create	
M文件	M-File} 菜单项,就引出书写着这些指令的 M 文件编辑	

1.5.2 历史指令的再运行

历史指令的重新调用 方便、直观

【例 1.5-1】演示如何再运行算例 1.3-10 中的全部绘图指令。

具体操作过程:

[Ctrl + 鼠标左键] 点亮指令

点击 鼠标右键, 引出现场菜单

选中现场菜单项{Evaluate Selection}

[说明]

● 单行历史指令的再运行操作更简单,用鼠标左键双击所需的那行指令即可。

1.6 Current Directory、路径设置器和文件管理

当在指令窗中运行一条指令时,

MATLAB 是怎样从庞大的函数和数据库中,找到所需的函数和数据的呢? 用户怎样才能保证自己所创建的文件能得到 MATLAB 的良好管理, 又怎样能与 MATLAB 原有环境融为一体呢?

1.6.1 Current Directory当前目录浏览器简介

当前目录浏览器界面上, 自上而下分别是:

当前目录名,工具条,文件、文件夹列表及文件描述区等。

当前目录窗设计了一个专门的操作菜单,处理 M 文件、MAT 文件数据等。

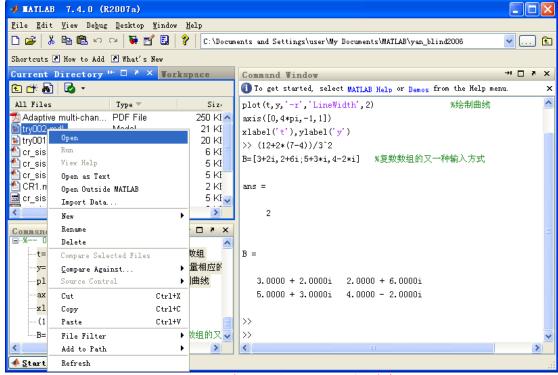


图 1.6-1 当前目录浏览器和适配的弹出菜单

〖说明〗

- MATLAB 启动后的默认当前目录通常是: C:\Documents and Settings\user\My Documents\MATLAB
- notebook 文档启动 MATLAB, 当前目录将是 MATLAB 所在的根目录。
- 提醒读者: 千万不要把 MATLAB 所在根目录设成当前目录。 用户应该通过重新设置,把当前目录设置在适当的目录上。

表 1.6-1 当前目录适配菜单的应用

应用功能	操作方法	简捷操作方法
运行M文件	点亮待运行文件;按鼠标右键引出现场菜单;选中	
	{Run}菜单项,即可使该 M 文件运行。	
编辑 M 文件	点亮待运行文件;按鼠标右键引出现场菜单;选中	鼠标左键双击
	{Open} 菜单项,此 M 文件就出现在编辑/调试器中。	M文件。
把MAT文件全部	点亮待装数据文件;按鼠标右键引出现场菜单;选中	鼠标左键双击
数据输入内存	{Open} 菜单项,此文件的数据就全部装入工作内存。	MAT文件。
把MAT文件部分	点亮待装载数据文件;按鼠标右键引出现场菜单;选	
数据输入内存	中{Import Data} 菜单项,引出数据预览选择对话框	
	"Import Wizard";在此框中"勾选"待装数据变量名,点	
	击 [Finish] 键,就完成操作。	

1.6.2 用户目录和当前目录设置

(1) 用户目录

MATLAB R2007a 在安装过程中,会自动生成一个目录

C:\Documents and Settings\user\My Documents\MATLAB.

该目录专供存放用户自己的各类 MATLAB 文件。

假若用户想另建一个目录,采用 Windows 规范操作就可实现。

(2) 应把用户目录设置成当前目录

在 MATLAB 环境中,如果**不特别**指明存放数据和文件的目录,那么 MATLAB 总默认 地将它们存放在<mark>当前目录</mark>上。因此,出于 MATLAB 运行可靠和用户方便的考虑, 建议:在 MATLAB 开始工作的时候,就应把用户自己的"用户目录"设置成当前目录。

(3) 把用户目录设置成当前目录的方法

方法一: 交互界面设置法

在 MATLAB 操作桌面右上方,或当前目录浏览器左上方,都有一个当前目录设置区。它包括:"目录设置栏"和"浏览键"。用户或在"设置栏"中直接填写待设置的目录名,或借助"浏览键"和鼠标选择待设置目录。

方法二: 指令设置法

假设待设置的用户目录是 D:\Backup\我的文档\MATLAB 指令是 cd D:\Backup\我的文档\MATLAB 一旦 MATLAB 重新启动,以上设置操作必须重新进行。

1.6.3 MATLAB的搜索路径

MATLAB的所有(M、MAT、MEX)文件都被存放在一组**结构严整的目录树**上。 MATLAB把这些目录按优先次序设计为"搜索路径"上的各个节点。

MATLAB工作时,就沿着此搜索路径,从各目录上寻找所需的文件、函数、数据。 当用户从指令窗送入一个名为 cont 的指令后,MATLAB 的基本搜索过程大抵如下。

- 检查 MATLAB 内存,看 cont 是不是变量;假如不是变量,则进行下一步。
- 检查 cont 是不是内建函数 (Built-in Function); 假如不是, 再往下执行。
- 在当前目录上,检查是否有名为 cont 的 M 文件存在;假如不是,再往下执行。
- 在MATLAB搜索路径的其他目录中,检查是否有名为 cont 的 M 文件存在。

实际搜索过程远比前面描述的基本过程复杂。 但又有一点可以肯定, **凡不在搜索路径上的内容,不可能被搜索**。

1.6.4 MATLAB搜索路径的扩展

一 何时需要修改搜索路径

假如用户有(多个)目录需要同时与 MATLAB 交换信息,那么就应把(这些)目录放置在 MATLAB 的搜索路径,使得这些目录上的文件或数据能被调用。

又假如其中某个目录需要用来存放<mark>运行中</mark>产生的文件和数据,那么还应该把这个目录 设置为<mark>当前目录</mark>。

二 利用设置路径对话框修改搜索路径

选择 {File:Set Path} 下拉菜单项。

[说明]

- 该对话框设置搜索路径有两种修改状态:
 - **当前有效**修改 假如在路径设置过程中,仅使用了该对话框的左侧按键。 **永久有效**修改 假如在设置后,点击了对话框下方 [Save] 按键。
- 所谓永久有效修改是指:所进行的修改不因 MATLAB 的关闭而消失。

三 利用指令path 设置路径

利用 path 指令设置路径的方法对任何版本的 MATLAB 都适用。 假设待纳入搜索路径的目录为 c:\my dir, 那么以下任何一条指令均能实现: path(path,'c:\my_dir') 把 c:\my_dir 设置在搜索路径的尾端 path('c:\my_dir',path) 把 c:\my_dir 设置在搜索路径的首端

[说明]

- 用 path 指令扩展的搜索路径仅在当前 MATLAB 环境下有效。 也就是说:若用户退出当前 MATLAB后,再重新启动 MATLAB,那么在前一环境下 用 path 所定义的扩展搜索路径无效。
- 用 path 指令扩展的搜索路径的方法可以编写在程序中。

1.7 工作空间浏览器和数组编辑器

1.7.1 Workspace工作空间浏览器简介

表 1.7-1 工作空间浏览器主要功能及其操作方法

功能	操作方法	简捷操作法
创建	点击 图标,在工作空间中生成一个"unnamed"的新变量;双	
新变量	击该新变量图标,引出 Array Editor 数组编辑器 (如图 1.7-3);	
	在数组编辑器中,向各元素输入数据;最后,对该变量进行重	
	命名。	
显示	点亮变量;或点击图标∰,或选中弹出菜单中的 {Open	用鼠标左键
变量内容	Selection} 项,则变量内含的数据就显示在"Array Editor"数组	双击变量。
	编辑器中。(数组编辑器详见第 1.7-3 节)	
图示变量	点亮变量; 或选中图标 <mark>₩</mark> 下拉菜单项, 或选中弹出菜单中的	
	绘图项,就可以适当地绘出选中变量的图形。(详见例 1.7-1)	
用文件	选择待保存到文件的(一个或多个)变量,或点击图标┅,或	
保存变量	选中弹出菜单中的 {Save Workspace As}项, 便可把那些变量保	
	存到 MAT 数据文件。(详见第 1.7-4 节之一)	
从文件向	点击图标 [49]; 选择 MAT 数据文件; 再单击那文件, 引出"Import	
内存装载	Wizard"界面,它展示文件所包含的变量列表;再从列表中,	
变量	选择待装载变量,便可。	

【例 1.7-1】通过"工作空间浏览器"的运作,采用图形显示内存变量 y。 (注意: 本例是在例 1.3-10运行后进行的,因此内存中保存着由例 1.3-10产生的全部变量。)

【例 1.3-10】画出衰减振荡曲线 $y = e^{-\frac{1}{3}} \sin 3t$, t 的取值范围是 $[0,4\pi]$ 。

本例演示:展示数组运算的优点;展示MATLAB的可视化能力。

t=0:pi/50:4*pi;

%定义自变量t的取值数组

y=exp(-t/3).*sin(3*t); %计算与自变量相应的y数组。注意: 乘法符前面的小黑点。

plot(t,y,'-r','LineWidth',2)

%绘制曲线

axis([0,4*pi,-1,1])

xlabel('t'),ylabel('y')

- (1) 借助"图标菜单"图示变量
- (2)借助"现场弹出菜单"图示变量

[说明]

● 例 1.3-10 的图 1.3-4 虽与本例图 1.7-2 形状相同,但含义却不同。 这两张图的纵轴刻度相同,但横轴刻度不同。 前者表现的是函数关系 y = f(t); 而后者表现的是数组 v(k), 在此 k 是数组元素的序号。

● 注意:

借助图形表现数据是十分常用的手段。这是由于图形具有表现数据内在联系和宏观特征的卓越能力。

1.7.2 工作空间的管理指令

管理工作空间的三个常用指令: who, clear, pack。

一 查询指令who及whos

【例 1.7-2】在指令窗中运用 who, whos 查阅 MATLAB 内存变量。 who, whos 在指令窗中运行后的显示结果如下: who

whos

〖说明〗

● who, whos 指令操作对 MATLAB 的所有版本都适用。

二 从工作空间中删除变量和函数的指令clear

最常用的几种格式:

clear 清除工作空间中的所有变量

clear var1 var2 清除工作空间中的 var1 和 var2 变量

clear all 清除工作空间中所有的变量、全局变量、编译过的 M 函数和 MEX 链接。

clear fun1 fun2 清除工作空间中名为 fun1 和 fun2 的函数

注意: 在第 2、4 调用格式中, clear 后面的变量名和函数名之间一定要采用"空格"分隔, 而不能采用其他符号。

三 整理工作空间内存碎片的指令pack

在 MATLAB 运行期间,它会自动地为产生的变量分配内存,也会为使用到的 M 函数分配内存。有时对于容量较大的变量,会出现"Out of memory"的错误。此时,可能使用 clear 指令清除若干内存中的变量也无济于事。

产生这种问题的一个原因是: MATLAB 存放一个变量时,必须使用"连成一片"的内存空间。对于那些被碎片分割得"支离破碎"的内存空间,即便它们的总容量超过待生成变量,也无法使用。在这种情况下,借助 pack 指令也许能解决问题。

1.7.3 Array Editor数组编辑器

双击工作空间浏览器中的变量图标,将引出数组编辑器 Array Editor。 该编辑器可用来查看、编辑数组元素:对数组中指定的行或列进行图示。

1.7.4 数据文件和变量的存取

一 借助工作空间浏览器产生保存变量的MAT文件

选中弹出菜单中的 {Save As} 项;就可把那些变量保存到(由用户自己命名的)mydata.mat的数据文件。该数据文件默认地存放在"当前目录"上。

二 借助输入向导Import Wizard向工作空间装载变量

三 存取数据的操作指令save 和load

利用 save, load 指令实现数据文件存取是 MATLAB 各版都采用的基本操作方法。它的具体使用格式如下。

把全部内存变量保存为 FileName.mat 文件 save FileName save FileName v1 v2 把变量 v1, v2 保存为 FileName.mat 文件 save FileName v1 v2 -append 把变量 v1, v2 添加到 FileName.mat 文件中 save FileName v1 v2 -ascii 把变量 v1, v2 保存为 FileName 8位 ASCII 文件 把变量 v1, v2 保存为 FileName 16 位 ASCII 文件 save FileName v1 v2 -ascii -double load FileName 把 FileName.mat 文件中的全部变量装入内存 把 FileName.mat 文件中的 v1, v2 变量装入内存 load FileName v1 v2 load FileName v1 v2 -ascii 把 FileName ASCII 文件中的 v1, v2 变量装入内存

[说明]

- FileName 文件名可以带路径,也可以带扩展名。
- 如果一组数据是经过长时间的复杂计算后获得的,那么为避免再次重复计算,常使用 save 加以保存。此后,每当需要,都可通过 load 重新获取这组数据。这种处理模式常 在实际中被采用。

1.8 Editor/Debugger和脚本编写初步

- 比较简单的问题,一次性问题 一组指令去求解
- 所需的指令较多和所用指令结构较复杂

或当一组指令通过改变少量参数就可以被反复使用去解决不同问题 解决办法: M 脚本文件

1.8.1 Editor/Debugger M文件编辑/调试器简介

M文件编辑器的启动

1.8.2 M脚本文件编写初步

所谓 M 脚本文件是指:

- (1) 该文件中的指令形式和前后位置,与解决同一个问题时在指令窗中输入的 那组指令没有任何区别。
- (2) MATLAB 在运行这个脚本时,只是简单地从文件中读取那一条条指令,送到 MATLAB 中去执行。
- (3)与在指令窗中直接运行指令一样,脚本文件运行产生的变量都是驻留在MATLAB基本工作空间中。
- (4) 文件扩展名是".m"。

【例 1.8-1】编写解算例 1.3-10 题目的 M 脚本文件,并运行之。

t=0:pi/50:4*pi; %定义自变量t的取值数组 y=exp(-t/3).*sin(3*t); %计算与自变量相应的y数组。注意: 乘法符前面的小黑点。 plot(t,y,'-r','LineWidth',2) %绘制曲线 axis([0,4*pi,-1,1]) xlabel('t'),ylabel('y')

〖说明〗

当使用 M 文件编辑调试器保存文件时,或当在 MATLAB 指令窗中运行 M 文件时, 不必写出文件的扩展名。

在 M 文件编辑调试器中, 可以采用汉字注释, 并总可获得正确显示。

1.9 帮助系统及其使用

面对不断变化的实际问题,常常产生两类困惑:

- 1. 知道具体指令,但不知道该怎么用;
- 2. 想解某个具体问题,不知道 MATLAB 有哪些指令可用,如何用?

MATLAB 帮助系统促进我们:

1.9.1 六大子系统构成的帮助体系

表 1.9-1 MATLAB 的帮助体系

子系统 名称	特点	资 源
指令窗帮 助子系统	文本形式;最可信、最原始; 不适于系统阅读	所有 M 文件的帮助注释内容;包含在 M 文件之中
帮助 导航系统	HTML 形式; 系统叙述 MATLAB规则,一般用法; 适于系统阅读和交叉查阅	位于 matlab\help 目录下; HTML和 XML 文件,物理上独立于 M 文件
典型算例 演示系统	HTML和GUI交互界面; 以算例为载体分类介绍 MATLAB使用方法	位于 matlab\toolbox 下各分类子目录的 demos 子目录; 专门编写 HTML 和演示 M 文件
视频 演示系统	Flash 形式;视听兼备,直观 形象;内容限于版本新特点	通过帮助系统直接链接到 mathworks 公司网站上
PDF 文件 帮助系统	系统的标准打印文件;便于 长时间系统阅读	下载网站 http://www.mathworks.com/access/helpdesk/help/helpdesk.html
Web 网上 帮助系统	交互式讨论具体问题	制造商网站 http://www.mathworks.com 新闻组 comp.soft-sys.matlab 讨论站 www.mit.edu/~pwb/cssm/matlab-faq.html

HTML: 超文本标记语言 XML: 可扩展标记语言

1.9.2 常用帮助指令

一 函数搜索指令

知道具体函数指令名称,但不知道该函数如何使用

函数搜索指令的调用:

help 列出所有函数分组名(Topic Name) help TopicName 列出指定名称函数组中的所有函数 help FunName 给出指定名称函数的使用方法

helpwin 列出所有函数分组名(Topic Name) helpwin TopicName 列出指定名称函数组中的所有函数 helpwin FunName 给出指定名称函数的使用方法

doc ToolboxName 列出指定名称工具包中的所有函数名 doc FunName 给出指定名称函数的使用方法

- help 搜索的资源是 M 文件帮助注释区的内容: 函数的调用格式和输入输出量含义
- helpwin 搜索的资源还是 M 文件帮助注释区的内容。"比较方便的超文本"
- doc 搜索是在 HTML 文件构成的帮助子系统中进行的。内容比 M 文件帮助注释详细。

二 词条搜索指令

"想解某具体问题,但不知道有哪些函数指令可以使用"的场合

lookfor KeyWord对 M 文件 H1 行进行单词条检索docsearch('KeyWord1 @ KeyWord2 @ KeyWordn')对 HTML 子系统进行多词条检索

〖说明〗

- KeyWord, KeyWord1, KeyWord2, KeyWord3 分别是待检索的词条 @ 表示逻辑运算符。代表 OR, AND, NOT 中的任意一个 待索词条、逻辑符之间用"空格"分隔
- lookfor 搜索的资源是 M 文件帮助注释区中的第一行(简称 H1 行)。
- docsearch 搜索在 HTML 文件构成的帮助子系统中进行 搜索到的内容较详细

1.9.3 Help 帮助浏览器

帮助导航/浏览器搜索的资源:专门的 HTML 帮助子系统。 内容来源:所有 M 文件,但更详细。 界面友善、方便 用户寻求帮助的最主要资源。 引出方法:

- (1) 在指令窗中运行 helpbrowser 或 helpdesk。
- (2) 在 MATLAB 的默认操作桌面,及各独立出现的交互窗口中 点击工具条的 图标; 或选中下拉菜单项 {Help: MATLAB Help}。

一 帮助浏览器简介

左半侧:帮助导航器 Help Navigator 右半侧:帮助浏览器 Help Browser

帮助导航器的四个"导航窗":内容分类目录 Contents,

指令检索 Index, 词条检索 Search, 演示实例 Demos。

二 Contents 目录 窗

列出"节点可展开的目录树"。

点击目录条,即可在浏览器中显示出相应标题的 HTML 帮助文件。

● 提供**全方位**系统帮助的"**向导图**"。

向导图的构造贯穿四个原则:

"深浅层次"、"功能划分"、"帮助方式"和"英文字母次序"

表 1.9-2 按深浅层次设置的条目

层次类型	功用	条目举例	阅读建议
概括	简介特点、内容、方法	Begin Here	粗读
版本特点	专门介绍版本升级变化	Release Notes	老用户必读
入门和使用初步	以简单操作、说明和算例引导用户	Getting Started	新用户必读
			老用户浏览
详细指南	全面、系统、按功能叙述	Using MATLAB	或系统读
			或选读
综合算例	强调指令间配合和使用要领演示	Examples	选读

三 Index检索窗

MATLAB 有一个事先制作的指令、函数列表文件

用户在待查栏(Eeter index term)中填入某词汇时, 其下白板上就列出与之最匹配的词汇列表。 与函数搜索指令 doc 功能一致

四 Search搜索窗

利用关键词查找全文中与之匹配章节条目的交互界面与 Index 搜索有两点主要区别

- Index 搜索只在<mark>专用指令表</mark>中查找 Search 搜索是在整个 HTML 文件中进行,覆盖面更宽。
- Index 搜索只能进行**单指令名**搜索 Search 搜索采用**多词条的逻辑组合搜索**,功能更强,搜索效率更高。 与词条搜索指令 docsearch 功能一致

1.9.4 **DEMO**演示系统

MATLAB 的 Demos 演示系统相当丰富。它以算例为切入点,视算例的实质不同,或用 M 码文件、Simulink 块图文件演示,或用 GUI 图形用户界面、视频影像表现。这些演示文件(除有些视频文件外),分布在 MATLAB 的相应子目录中。

进入图 1.9-4 所示 DEMO 演示系统的方式主要有三种:

- (1) 在指令窗的"综合信息条"中(参见图 1.2-1),点击 Demos 超链接。
- (2) 在指令窗中,运行 demos 指令。
- (3) 在帮助导航器 Help Navigator 中(参见图 1.9-4),点击"Demos"导航窗。该界面左侧采用标准的 Windows 目录树结构清晰地展示分类演示内容。界面右侧显示四种演示形式: M-file 演示形式,M-GUI 演示形式,Model 演示形式,和 Video 演示形式。

这些演示形式中的内容,有的是"入门引导"型的,有的是"学科专业"型的,有的是"MATLAB编程技巧"型的。无论是对MATLAB新用户来说还是对老用户来说,这些演示都是十分有益而宝贵的资源。该演示程序的示范作用是独特的,是包括MATLAB用户指南在内的有关书籍所不能代替的。用户若想学习和掌握MATLAB,不可不看这组演示程序。但对初学者来说,不必急于求成去读那些太复杂的程序。

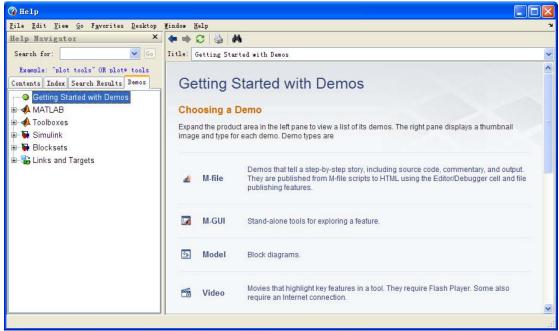


图 1.9-4 帮助导航其中的 Demos 窗

一 M-file演示形式

M 码和 M 文件是 MATLAB 最基本的指令形式和最基本的文件类型。M 文件是为实现某一特定目标而有机配合的一组 MATLAB 指令。

在图 1.9-4 所示界面左侧的目录树上,选中并展开 MATLAB 节点,就可导出如图 1.9-5 的界面。点击内容显示窗中某个 M-file 形式的文件缩略图或说明文字(如 Basic Matrix Operations),就引出图 1.9-6 所示的界面。

图 1.9-6 界面右侧窗中,采用 HTML 格式显示所选那 M 文件的内容。这种形式比较接近"书本"习惯,看起来比较舒畅。

阅读演示 M 文件的另一种方式是:点击图 1.9-6 左上方的 "Open intro.m in the Editor" 超链接,采用 M 文件编辑器显示。读者从编辑器中看到的是:直接面向 MATLAB 的 M 文件。它有助于读者习惯和理解 M 文件的编写。当然这些文件也都可以在 MATLAB 中直接运行。

如果想在指令窗中观察那 M 文件的运行情况,可直接点击图 1.9-6 右上方的 "Run in the

Command Window"超链接,就会出现如引出图 1.9-7 所示的 Command Window 界面。再通过操作该界面左侧的"Next"超链接,可一步一步地观察指令、运算结果和可视化图形之间的关系。这种方式可让读者更直接地感受 MATALB 指令的书写和作用。

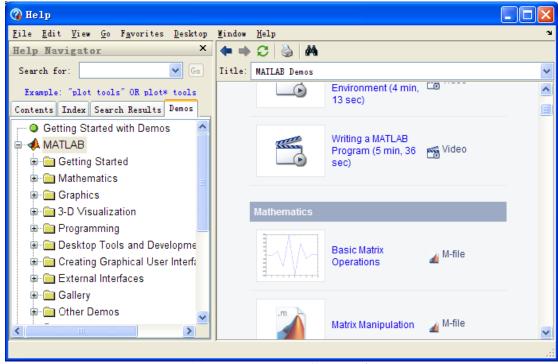


图 1.9-5 子目录展开后的 Demos 窗

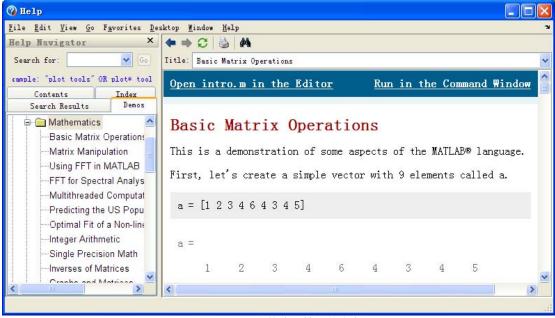


图 1.9-6 M 文件演示算例的内容

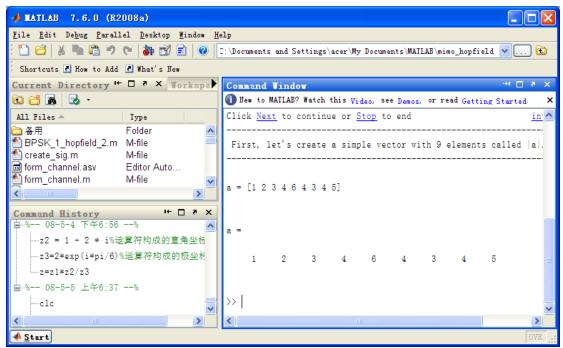


图 1.9-7 M 文件演示算例在指令窗中的逐步执行

二 M-GUI演示形式

M-GUI 是指"用 M 文件创建的图形用户界面"。

M-GUI 主要用于:采用鼠标点击等简单操作,现场改变参数或模式,直观地表演或研究某特定的函数关系、实验和任务。

创建 M-GUI 有两种方式: 借助 GUIDE 工具创建和直接编写专门的 M 文件创建。不管采用哪种创建方式,都要求读者对 MATLAB 有较好的理解。

在 Demos 系统中,罗列着许多 M-GUI 演示算例。如若,点击 Demos 导航窗 MATLAB\Mathematics分类下的 Eig. & Singular Value Show,或其左边的缩略图(见图 1.9-8),就会引出一个图形用户界面。在这个界面上,通过鼠标拖拉坐标系中的 x 向量,可以看到相应的 Ax 向量随之变动。该界面上还包含:选择 eig 还是 svd 的切换"按键",填写或选择 (2×2)矩阵的"可编辑文本框"及"弹出框"等。通过这些"控件"的操作,就可以几何地显示"特征值分解(eig 指令)"或"奇异值分解(svd 指令)"本质。

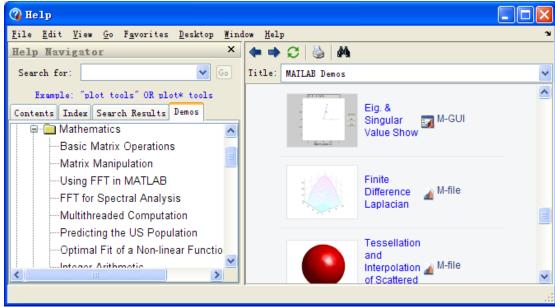


图 1.9-8 Demos 系统中表列的 M-GUI 演示算例之一

三 Model演示形式

Demos 演示系统中的 Model 指的是:用 Simulink 块图所构成的程序模型。这种模型所描述的系统结构清晰,各子系统(模块)之间的关系明了。

在 Demos 导航窗的 Simulink 和 Blocksets 目录中存放着许多 Model 演示算例(参见图 1.9-9)。用户若点击 Model 的缩略图或其旁的文字标识,就会显示出关于该模型的说明,再进一步操作,便可打开相应的模型。

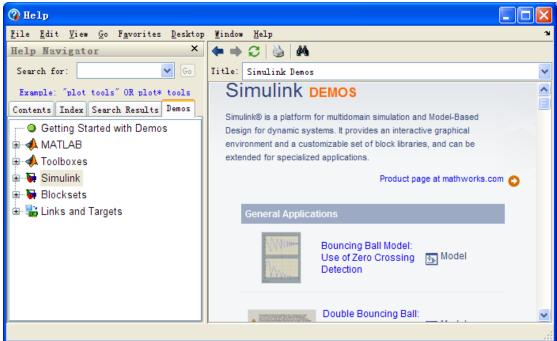


图 1.9-9 Simulink Demos 中表列的 Model 演示算例之一

四 Video演示形式

在 MATLAB 的 Demos 的 MATLAB 目录下的 GettingStarted, Graphics, Programming, Desktop Tools and Development Environment,以及 New Features in Version等子目录里还提供视频帮助资料(参见图 1.9-10)。Video 演示需要 Flash Player 和 Internet 的支持。这种演示形式模拟"教师的教学方式",使读者通过眼看界面上的鼠标操作和图像变化,耳听相配的英语解释,很容易明白各种操作之间的关系和 MATLAB 各种新功能的使用。

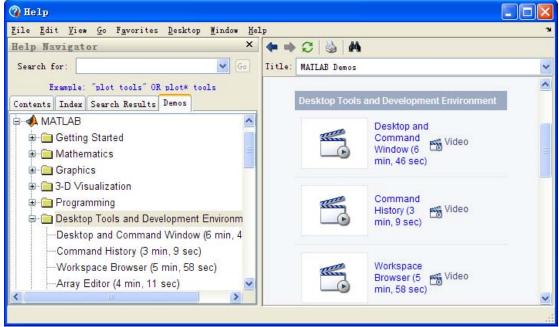


图 1.9-10 Demos 系统中的视频演示算例

习题1

- 1. 在安装 MATLAB 软件时,哪个组件是必须选择的?假如不"勾选"这个组件,那么就不可能建立 MATLAB 工作环境。
- 2. 数字 1.5e2, 1.5e3 中的哪个与 1500 相同?
- 3. 请指出如下 5 个变量名中,哪些是合法的? abcd-2 xyz_3 3chan a 变量 ABCDefgh
- 4. 在MATLAB环境中,比1大的最小数是多少?
- 5. 设 a = -8, 运行以下三条指令,问运行结果相同吗?为什么? w1=a^(2/3) w2-(a^2)^(1/3)

 $w2=(a^2)^(1/3)$ $w3=(a^(1/3))^2$

 $w3=(a^{(1/3)})^{2}$

- 6. 指令 clear, clf, clc 各有什么用处?
- 7. 以下两种说法对吗? (1) "MATLAB 的数值表达精度与其指令窗中的数据显示精度相同。" (2) MATLAB 指令窗中显示的数值有效位数不超过 7 位。"

8. 想要在 MATLAB 中产生二维数组
$$S = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$
 ,下面哪些指令能实现目的?

S=[1,2,3;4,5,6;7,8;9] S=[1 2 3;4 5 6;7 8 9]

S=[1, 2, 3; 4, 5, 6; 7, 8, 9] %整个指令在中文状态下输入

9. 试为例 1.3-5 编写一个解题用的 M 脚本文件?

第2章 符号计算

符号计算:

解算数学表达式、方程<mark>不是</mark>在离散化的<mark>数值点</mark>上进行,<mark>而是</mark>凭借一系列恒等式,数学定理,通过推理和演绎,获得**解析**结果。

符号计算建立在**数值完全准确表达**和**推演严格解析**的基础之上,所得结果完全推确。

特点:

- 一,相对于 MATLAB 的数值计算"引擎"和"函数库"而言,符号计算的"引擎"和"函数库"是独立的。
- 二,在相当一些场合,符号计算解算问题的指令和过程,显得比数值计算更自然、更简明。 三,大多数理工科的本科学生在学过高等数学和其他专业基础课以后,比较习惯符号计算的 解题理念和模式。

运算引擎 MuPAD

MuPAD是极佳数学及符号数值运算绘图软件,同时也作为MATLAB7.8的符号计算工具箱,是一具有人工智能的数学软件,非常适合科学家及工程师使用.更适合每一个人使用,使用的方法非常简单,只要输入方程式就立刻得到答案,可以求Symbolic符号解,多项式之根,求非线性方程式之根,矩阵及向量Vector and Matrices运算,代数Algebra运算,求积分之值,求微分之值 Calculus 微积分等。方程式可以处理复数计算. 完美的绘图功能,图型输入,输出,轻松无比的绘图,可以输入多个2-D函数或极坐标函数或3-D函数,选择所要绘图参数,就可以完成图形,以及图形的动画制作也是非常方便。数值计算结果并不是MATLAB命令行窗口所得的类似代码形式,而是规范数学格式。并拥有一内建的程序语言,帮助文档以及文本操作,文本操作在一定程度上可以取代word.是一个超级的工程数学计算器.

MathWorks 自从 2008年 10 开始,在 Matlab 的新版本(Matlab 2008a,即 7.6之后)中使用 MuPAD 内核替换原来的 Maple 符号计算内核!

2.1 符号对象和符号表达式

MATLAB 依靠基本符号对象(包括数字、参数、变量)、运算符及一些预定义函数来构造和衍生符号表达式和符号方程。

2.1.1 符号对象的创建和衍生

一 生成符号对象的基本规则

- 任何基本符号对象都必须借助专门的符号函数指令 sym 或 syms 定义。
- 任何包含符号对象的表达式或方程,将继承符号对象的属性。

二 符号数字

符号(类)数字的定义:

sym('Num') 创建一个符号数字 Num

sc=sym('Num') 创建一个符号常数 sc, 该常数值准确等于 Num

说明: Num 代表一个具体的数字

Num 必须处于(英文状态下的)单引号内,构成字符串(关于字符串参见附录 A.1)。

【例 2.1-1】符号(类)数字与数值(类)数字之间的差异。

a=pi+sqrt(5)

% 创建方式

sa=sym('pi+sqrt(5)')

Ca=class(a)
Csa=class(sa)

% 类别判断

vpa(sa-a)

digits

digits(8)

三 符号参数

表达式e-axsinbx中的a,b称为参数。

定义格式:

syms Para 定义符号参数 Para

Para=sym(' Para')

syms Para Flag 定义具有 Flag 指定属性的符号参数 Para

Para=sym(' Para', 'Flag')

syms Para1 Para2 ParaN 定义 Para1 Para2 ParaN 为符号参数

syms Para1 Para2 ParaN Flag 定义 Para1 Para2 ParaN 为具有 Flag 指定属性的符号参数

- 符号参数名不要用处于"字母表中小写字母 x 及其两侧的英文字母"开头。
- Flag 表示参数属性,可具体取以下词条:

positive 表示那些符号参数取正实数;

real 表示那些符号参数限定为实时;

unreal /clear 表示那些符号参数为不限定的复数。

四 符号变量

e^{-ax}sinbx中的x称为<mark>变量</mark>,符号变量的定义同符号参数。确定自由符号变量的规则:

- 在专门指定变量名的符号运算中,解题一定围绕指定变量名进行。
- 自动识别符号变量时,字母的优先次序为 x,v,w,z,v 等。

自动识别表达式中自由、独立的符号变量的指令:

findsym(EXPR) 确认表达式 EXPR 中所有自由符号变量

findsym(EXPR, N) 确认表达式 EXPR 中距离 x 最近的 N 个自由符号变量

```
【例 2.1-2】用符号计算研究方程 uz^2 + vz + w = 0 的解。
(1) 不指定变量情况
                     % 定义符号参数/变量
syms u v w z
Eq=u*z^2+v*z+w;
result_1=solve(Eq)
                                          %
findsym(Eq.1)
(2) 指定变量情况
result_2=solve(Eq,z)
【例 2.1-3】对独立自由符号变量的自动辨认。
(1)
                                      % 定义符号参数/变量
syms a b x X Y
k=sym('3');
                                      % 符号常数
                                         % 直接定义符号表达式
z=sym('c*sqrt(delta)+y*sin(theta1)');
                                     % 构成衍生符号表达式
EXPR=a*z*X+(b*x^2+k)*Y;
(2)
findsym(EXPR)
(3)
findsym(EXPR,1)
(4)
findsym(EXPR,2),findsym(EXPR,9)
【例 2.1-4】findsym 确定自由变量是对整个矩阵进行的。
syms a b t u v x y
A=[a+b*x,sin(t)+u;x*exp(-t),log(y)+v]
```

2.1.2 符号计算中的算符

findsym(A,5)

[a + b*x, u + sin(t)] [x/exp(t), v + log(y)]

A =

ans =
x,y,v,u,t

- 与数值计算中的算符在形状、名称和使用方法上几乎完全相同。
- 仅注意:在符号对象的关系运算符中,只有算符 "==", "~=" 比较结果为 "真"时,用1表示;否则用0表示。

2.1.3 符号计算中的函数指令

表 2.1-1 MATLAB 中可调用的符号计算函数指令

类 别	情况描述	与数值计算对应关系
基本函数	三角函数、双曲函数及反函数;除 atan2 外	名称和使用方法相同
	指数、对数函数(如 exp, expm)	名称和使用方法相同
	复数函数(注意:没有幅角函数 angle)	名称和使用方法相同
	矩阵分解函数(如 eig, svd)	名称和使用方法相同
	方程求解函数 solve	不同
	微积分函数(如 diff, int)	不完全相同
	积分变换和反变换函数(如 laplace, ilaplace)	只有离散 Fourier 变换
	绘图函数(如 ezplot,ezsurf)	数值绘图指令更丰富
经典特殊函数	如误差函数 erf、贝塞尔函数 besselj、第一类完全	部分
	椭圆积分 EllipticK 等;通过 mfunlist 可以看到所	
	有经典函数名	
	mfun functions perform numerical, not symbolic,	
	calculations. The input parameters should be scalars,	
	vectors, or matrices of type double, or complex	
	doubles, not symbolic variables.	
Maple 库函数	Maple 库函数在符号计算的扩展目录上;可通过	
	mhelp index 看到各子函数库的名称;函数的数量	
	很大;使用库函数,需要具备 Maple 语言知识	

注意: 使用函数注意数据类型。就数字而言,有双精度和符号类数字之分。

2.1.4 符号对象的识别

为了函数指令与数据对象的适配,MATLAB 提供了用于识别数据对象属性的指令:

class(var) 给出变量 var 的数据类别(如 double, sym 等) isa(var, 'Obj') 若变量 var 是 Obj 代表的类型,给出 1,表示"真"

whos 给出所有 MATLAB 内存变量的属性

【例 2.1-5】数据对象及其识别指令的使用。

(1)

clear

a=1;b=2;c=3;d=4;

Mn=[a,b;c,d]

Mc='[a,b;c,d]'

Ms=sym(Mc)

- %产生4个数值变量
- % 利用已赋值变量构成数值矩阵
- % 字符串中的a,b,c,d与前面输入的数值变量无关
- % Ms是一个符号矩阵,它与前面各变量无关

(2)

SizeMn=size(Mn)

SizeMc=size(Mc)

SizeMs=size(Ms)

(3)

CMn=class(Mn)

CMc=class(Mc)

CMs=class(Ms)

```
(4)
isa(Mn,'double')
isa(Mc,'char')
isa(Ms,'sym')

(5)
whos Mn Mc Ms
```

2.2 符号数字及表达式的操作

2.2.1 数值数字与符号数字之间的转换

一数值数字向符号数字的转换

在符号运算中,"数值类数字"会自动地转换为符号数字。

亦可借助 sym 函数:

```
      sym(Num,'r')或 sym(Num)
      数值类数字 Num 的广义有理表达

      sym(Num,'d')
      数值类数字 Num 的"十进制浮点"近似表达

      sym(Num,'e')
      数值类数字 Num 的带 eps 误差的理性近似表达

      sym(Num,'f')
      数值类数字 Num 的"十六进制浮点"近似表达
```

sym('Num')和 sym(Num)区别问题

IEEE-754 标准的浮点数格式

二 符号数字向双精度数字转换

double(Num_sym) 把符号数字 Num_sym 转换为双精度数字

2.2.2 符号数字的任意精度计算

digits 显示当前环境下符号数字"十进制浮点"表示的有效数字位数 digits(n) 设定符号数字"十进制浮点"表示的有效数字位数(默认 32 位) xs=vpa(x) 据表达式 x 得到 digits 指定精度下的符号数字 xs xs=vpa(x,n) 据表达式 x 得到 n 位有效数字的符号数字 xs

【例 2.2-1】digits, vpa 指令的使用。

```
digits
p0=sym('(1+sqrt(5))/2')
pr=sym((1+sqrt(5))/2)
%
pd=sym((1+sqrt(5))/2,'d')
%
e32r=vpa(abs(p0-pr))
e16=vpa(abs(p0-pd),16) %计算误差
e32d=vpa(abs(p0-pd))
```

```
Digits = 32

p0 =
  (1+sqrt(5))/2
pr =
  7286977268806824*2^(-52)
pd =
  1.6180339887498949025257388711907
e32r =
  .543211520368251e-16
e16 =
  0.
e32d =
  .543211520368251e-16
```

2.2.3 符号表达式的基本操作

collect(合并同类项) factor(进行因式分解) numden(提取公因式)等

最常用:

simple(EXPR) 把 EXPR 转换成最简形式

【例 2.2-2】简化
$$f = \sqrt[3]{\frac{1}{x^3} + \frac{6}{x^2} + \frac{12}{x} + 8}$$
。

syms x
 $f = (1/x^3 + 6/x^2 + 12/x + 8)^(1/3);$
 $g1 = simple(f)$
 $g2 = simple(g1)$

syms x
 $f = (1/x^3 + 6/x^2 + 12/x + 8)^(1/3);$
 $g1 = simple(f)$
 $g2 = simple(g1)$
 $g1 = (2*x+1)/x$
 $g2 = 2+1/x$

2.2.4 表达式中的置换操作

一 子表达式置换操作

[RS,ssub]=subexpr(S,ssub) 运用符号变量 ssub 置换子表达式,并重写 S 为 RS

```
【例 2.2-3】对符号矩阵 \begin{bmatrix} a & b \\ c & d \end{bmatrix}进行特征向量分解。
clear all
syms a b c d W
                            % v: 特征向量阵
                                          D: 特征值阵
[V,D]=eig([a b;c d])
                            %对矩阵元素中的公共子表达式进行置换表达
[RVD,W]=subexpr([V;D],W)
     通用置换指令
RES=subs(ES,old,new) 用 new 置换 ES 中的 old 后产生 RES
              用 new 置换 ES 中的自由变量后产生 RES
RES=subs(ES,new)
【例 2.2-4】用简单算例演示 subs 的置换规则。
(1) 产生符号函数
syms a x; f=a*sin(x)+5
a*sin(x) + 5
(2) 符号表达式置换
f1=subs(f,'sin(x)',sym('y'))
                                                            %<2>
class(f1)
(3) 符号常数置换
f2=subs(f,{a,x},{2,sym('pi/3')}) % <3> a被双精度数字置换,x被符号数字置换
class(f2)
(4) 双精度数值置换
f3=subs(f,{a,x},{2,pi/3})
                                                            %<4>
class(f3)
(5) 数值数组置换之一
f4=subs(subs(f,a,2),x,0:pi/6:pi)
                                                            %<5>
class(f4)
(6) 数值数组置换之二
f5=subs(f,{a,x},{0:6,0:pi/6:pi})
```

2.3 符号微积分

class(f5)

%<6>

2.3.1 极限和导数的符号计算

大学本科高等数学中的大多数微积分问题,都能用符号计算解决,手工笔算演绎的烦劳 都可以由计算机完成。

limit(f,v,a) 求极限 $\lim_{v \to a} f(v)$ 【例 2.3-1】试求 $\lim_{x \to \infty} \left(1 - \frac{1}{x}\right)^{kx}$ 。

syms x k
Lim_f=limit((1-1/x)^(k*x),x,inf)

【例 2.3-3】求
$$\mathbf{f}(x_1, x_2) = \begin{bmatrix} x_1 e^{x_2} \\ x_2 \\ \cos(x_1)\sin(x_2) \end{bmatrix}$$
的 Jacobian(雅可比)矩阵
$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} \end{bmatrix} .$$

syms x1 x2;f=[x1*exp(x2);x2;cos(x1)*sin(x2)];
v=[x1 x2];fjac=jacobian(f,v)

2.3.2 序列/级数的符号求和

symsum(f,v,a,b) 求 f 在变量 v 取遍[a, b]中所有整数时的和。a,b 缺省时默认求和区间[0, v-1]。

【例 2.3-8】求
$$\sum_{t=0}^{t-1} [t,k^3]$$
, $\sum_{k=1}^{\infty} \left[\frac{1}{(2k-1)^2}, \frac{(-1)^k}{k} \right]$ 。

syms k t;f1=[t k^3];f2=[1/(2*k-1)^2,(-1)^k/k];
s1=simple(symsum(f1)) % f1 的自变量被确认为t
s2=simple(symsum(f2,1,inf)) % f1 的自变量被确认为k
syms x y;f1=[y x^3];f2=[1/(2*y-1)^2,(-1)^y/y];
s1=simple(symsum(f1)) % f1 的自变量被确认为x
s2=simple(symsum(f2,1,inf)) % f1 的自变量被确认为y

2.3.3 符号积分

int(f,v) 求 f 对变量 v 的不定积分 **int(f,v,a,b)** 求 f 对变量 v 的定积分

【例 2.3-9】求
$$\int \frac{1}{x} \sqrt{\frac{1+x}{x}} dx$$
 。 clear syms x f=sqrt((1+x)/x)/x s=int(f,x)
s=simple(simple(s)) f = ((x + 1)/x)^{(1/2)/x} s = -2*(1/x + 1)^{(1/2)} - 2*atan((1/x + 1)^{(1/2)*i)*i} s = -2*(1/x + 1)^{(1/2)} - 2*atan((1/x + 1)^{(1/2)*i)*i}

【例 2.3-10】求 $\int \frac{ax}{1} \frac{bx^2}{x} dx$ 。 syms a b x;f=[a*x,b*x^2;1/x,sin(x)]; disp('The integral of f is');pretty(int(f))

【例 2.3-12】求阿基米德(Archimedes)螺线 $r = a \cdot \theta$, (a > 0) 在 $\theta = 0$ 到 φ 间的曲线长度 函数,并求出 $a = 1, \varphi = 2\pi$ 时的曲线长度。

F2=int(int(int(x^2+y^2+z^2,z,sqrt(x*y),x^2*y),y,sqrt(x),x^2),x,1,2)

%积分结果用 32 位数字表示

```
syms a r theta1 phil positive
x=r*cos(theta1);x=subs(x,r,a*theta1);
y=r*sin(theta1);y=subs(y,r,a*theta1);
dLdth=sqrt(diff(x,theta1)^2+diff(y,theta1)^2);
L=simple(int(dLdth,theta1,0,phil))
```

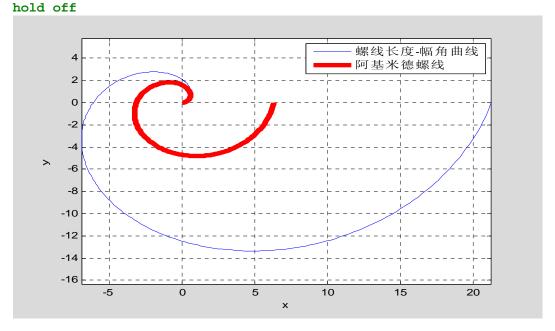
(2)

syms x y z

VF2=vpa(F2)

```
L_2pi=subs(L,[a,phi1],sym('[1,2*pi]'))
L_2pi_vpa=vpa(L_2pi)

(3)
L1=subs(L,a,sym('1'));
ezplot(L1*cos(phi1),L1*sin(phi1),[0,2*pi])
grid on
hold on
x1=subs(x,a,sym('1'));
y1=subs(y,a,sym('1'));
h1=ezplot(x1,y1,[0,2*pi]);
set(h1,'Color','r','LineWidth',5)
title('')
legend('螺线长度-幅角曲线','阿基米德螺线')
```



1. 说出以下三条指令产生的结果各属于哪种数据类型,是"双精度"对象,还是"符号"对象?

3/7+0.1, sym(3/7+0.1), vpa(sym(3/7+0.1))

〖答案〗

```
c1 =
    0.5286
c2 =
37/70
c3 =
.52857142857142857142857142857143
```

2. 在不加专门指定的情况下,以下符号表达式中的哪一个变量被认为是独立自由变量。 $sym('sin(w^*t)')$, $sym('a^*exp(-X)')$, $sym('z^*exp(j^*theta)')$

〖答案〗 ans = w ans = a

3. 求符号矩阵 $A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$ 的行列式值和逆,所得结果应采用"子表达式置换"

〖答案〗

简洁化。

[a31, a32, a33]

DA =

a11*a22*a33-a11*a23*a32-a21*a12*a33+a21*a13*a32+a31*a12*a23-a31*a
13*a22

IAs =

[(a22*a33-a23*a32)/d, -(a12*a33-a13*a32)/d, -(-a12*a23+a13*a22)/d]
[-(a21*a33-a23*a31)/d, (a11*a33-a13*a31)/d, -(a11*a23-a13*a21)/d]
[(-a21*a32+a22*a31)/d, -(a11*a32-a12*a31)/d, (a11*a22-a12*a21)/d]

d =

a11*a22*a33-a11*a23*a32-a21*a12*a33+a21*a13*a32+a31*a12*a23-a31*a 13*a22

4. 对函数 $f(k) = \begin{cases} a^k & k \ge 0 \\ 0 & k < 0 \end{cases}$, 当 a 为正实数时,求 $\sum_{k=0}^{\infty} f(k)z^{-k}$ 。 (实际上,这就是根据定义求 Z 变换问题。)

〖答案〗

$$Z1 = -z/(-z+a)$$

5. 对于
$$x > 0$$
, 求 $\sum_{k=0}^{\infty} \frac{2}{2k+1} \left(\frac{x-1}{x+1} \right)^{2k+1}$ 。(提示: 理论结果为 $\ln x$)

〖答案〗

 $s_s = log(x)$

6. (1)通过符号计算求 $y(t) = \left| \sin t \right|$ 的导数 $\frac{dy}{dt}$ 。(2)然后根据此结果,求 $\frac{dy}{dt} \bigg|_{t=0^-}$ 和

$$\frac{dy}{dt}\bigg|_{t=\frac{\pi}{2}}$$
°

〖答案〗

7. 求出 $\int_{-5\pi}^{1.7\pi} e^{-|x|} |\sin x| dx$ 的具有 64 位有效数字的积分值。

〖答案〗

matlab 2008a

si =

1.087849499412904913166671875948174520895458535212845987519414167

8. 计算二重积分 $\int_{1}^{2} \int_{1}^{x^{2}} (x^{2} + y^{2}) dy dx$ 。

〖答案〗

r =

1006/105

9. 在 $[0,2\pi]$ 区间,画出 $y(x) = \int_0^x \frac{\sin t}{t} dt$ 曲线,并计算 y(4.5) 。

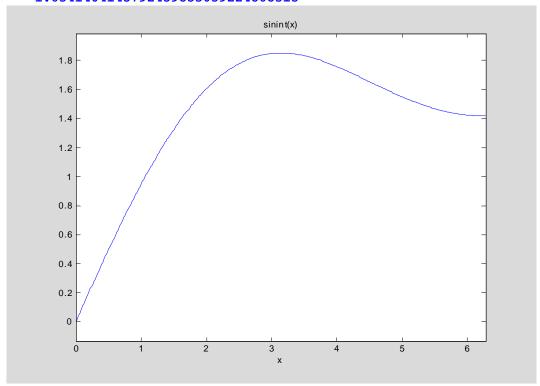
〖答案〗

v =

sinint(x)

y5 =

1.6541404143792439835039224868515



10. 求 $y(n) = \int_0^{\frac{\pi}{2}} \sin^n x dx$ 的一般积分表达式, 并计算 $y(\frac{1}{3})$ 的 32 位有效数字表达。

〖答案〗

yn =

 $1/2*pi^{(1/2)*gamma(1/2+1/2*n)/gamma(1+1/2*n)}$

11. 有序列 $x(k) = a^k$, $h(k) = b^k$, (在此 $k \ge 0$, $a \ne b$) , 求这两个序列的卷积 $y(k) = \sum_{k=0}^{k} h(n)x(k-n) .$

〖答案〗

v1 =

 $(-(b/a)^{(k+1)+1})*a^k*a/(-b+a)$

或

v2 =

 $(-b*b^k+a^k*a)/(-b+a)$

12. 设系统的冲激响应为 $h(t) = e^{-3t}$,求该系统在输入 $u(t) = \cos t$, $t \ge 0$ 作用下的输出。(提示:运用卷积进行计算)

〖答案〗

hut =

 $-3/10/\exp(t)^3+3/10\cos(t)+1/10\sin(t)$

13. 求 $f(t) = Ae^{-\alpha|t|}, \alpha > 0$ 的 Fourier 变换。

〖答案〗

F =

2*A*a/(a^2+w^2)

14. 求 $f(t) = \begin{cases} A\left(1 - \frac{|t|}{\tau}\right) & |t| \le \tau \\ 0 & |t| > \tau \end{cases}$ 的 Fourier 变换,并画出 $A = 2, \tau = 2$ 时的幅频谱。

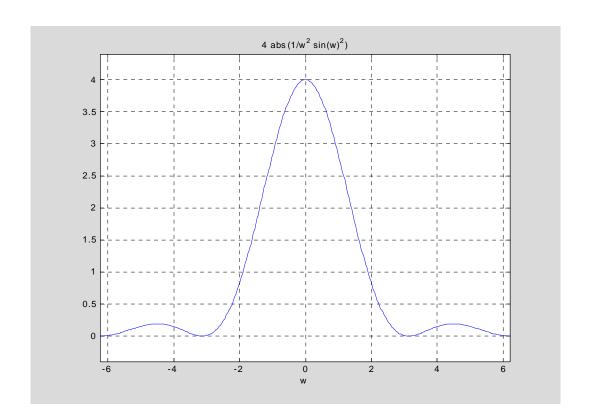
〖答案〗

Fws =

4*A/tao/w^2*sin(1/2*tao*w)^2

Fw2 =

4/w^2*sin(w)^2



15. 求
$$F(s) = \frac{s+3}{s^3 + 3s^2 + 6s + 4}$$
 的 Laplace 反变换。

〖答案〗

f =

$$1/3*\exp(-t)*(-2*\cos(3^{(1/2)*t})+3^{(1/2)}*\sin(3^{(1/2)*t})+2)$$

16. 利用符号运算证明 Laplace 变换的时域求导性质: $L\left[\frac{df(t)}{dt}\right] = s \cdot L[f(t)] - f(0)$ 。

〖答案〗

Ldy =

s*laplace(f(t),t,s)-f(0)

17. 求 $f(k) = ke^{-\lambda kT}$ 的 Z 变换表达式。

〖答案〗

 $F_z =$

 $z*exp(-lambda*T)/(z-exp(-lambda*T))^2$

18. 求方程 $x^2 + y^2 = 1$, xy = 2 的解。

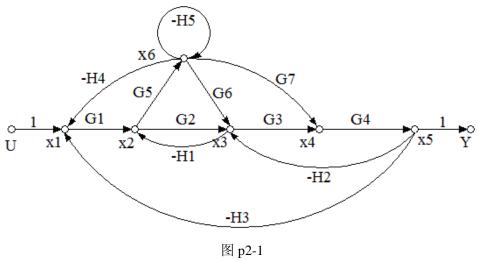
〖答案〗

x =

 $-1/2*(1/2*5^{(1/2)}+1/2*i*3^{(1/2)})^3+1/4*5^{(1/2)}+1/4*i*3^{(1/2)}$

```
-1/2*(1/2*5^{(1/2)}-1/2*i*3^{(1/2)})^3+1/4*5^{(1/2)}-1/4*i*3^{(1/2)}
-1/2*(-1/2*5^{(1/2)}+1/2*i*3^{(1/2)})^3-1/4*5^{(1/2)}+1/4*i*3^{(1/2)}
-1/2*(-1/2*5^{(1/2)}-1/2*i*3^{(1/2)})^3-1/4*5^{(1/2)}-1/4*i*3^{(1/2)}
y =
1/2*5^{(1/2)}+1/2*i*3^{(1/2)}
1/2*5^{(1/2)}-1/2*i*3^{(1/2)}
-1/2*5^{(1/2)}+1/2*i*3^{(1/2)}
-1/2*5^{(1/2)}-1/2*i*3^{(1/2)}
-1/2*5^{(1/2)}-1/2*i*3^{(1/2)}
```

19. 求图 p2-1 所示信号流图的系统传递函数,并对照胡寿松主编"自动控制原理"中的例 2-21 结果,进行局部性验证。



[答案]

传递函数 Y2U 为

G1 G4 (G3 G2 + G3 G2 H5 + G3 G6 G5 + G7 G5)/(G1 G2 G4 H3 G3 H5 + G1 G4 H2 G3 H4 G5 + G1 G4 H3 G3 G6 G5 + G1 G2 G4 H3 G3 + G1 G4 H3 G7 G5 - G4 H2 H1 G7 G5 + G4 H2 G3 H5 + G1 H4 G5 + G2 H1 H5 + G4 H2 G3 + H1 G6 G5 + G2 H1 + H5 + 1)

局部性验证用的传递函数

20. 采用代数状态方程法求图 p2-2 所示结构框图的传递函数 $\dfrac{Y}{U}$ 和 $\dfrac{Y}{W}$ 。

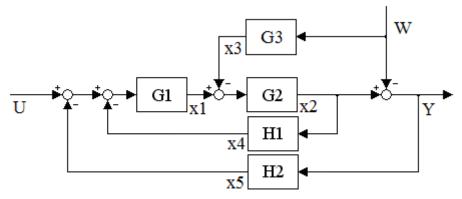


图 p2-2

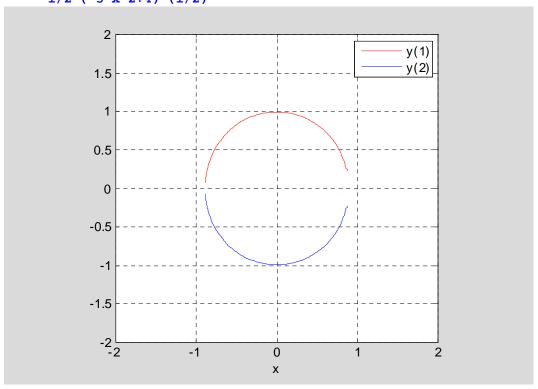
〖答案〗

传递函数 Y2U 为

传递函数 Y2W 为

21. 求微分方程 $\frac{yy'}{5} + \frac{x}{4} = 0$ 的通解,并绘制任意常数为 1 时解的图形。

〖答案〗



22. 求一阶微分方程 $x' = at^2 + bt$), x(0) = 2的解。

〖答案〗

x =

1/3*a*t^3+1/2*b*t^2+2

23. 求边值问题 $\frac{df}{dx} = 3f + 4g$, $\frac{dg}{dx} = -4f + 3g$, f(0) = 0, g(0) = 1 的解。

```
【答案】
f =
exp(3*t)*sin(4*t)
g =
exp(3*t)*cos(4*t)
```

第3章 数值数组及向量化运算

MATLAB 的核心内容: 数值数组和数组运算

- 二维数值数组的创建和寻访
- 数组运算和向量化编程
- 实现数组运算的基本函数
- 常用标准数组生成函数和数组构作技法
- 非数 NaN、"空"数组概念和应用
- 关系和逻辑操作

3.1 数值计算的特点和地位

符号计算的局限性:有很多问题 1) 无法解, 2) 求解时间过长

数值计算:适用范围广,能处理各种复杂的函数关系,计算速度快,容量大。

```
【例 3.1-1】已知 f(t) = t^2 \cos t,求 s(x) = \int_0^x f(t) dt。
(1) 符号计算解法
syms t x
ft=t^2*cos(t)
sx=int(ft,t,0,x)
ezplot(sx,0,5)
hold on
dt=0.05;
t=0:dt:5;
Ft=t.^2.*cos(t);
Sx=dt*cumtrapz(Ft); % 小梯形面积的累加求Ft曲线下的面积,由一个个宽度为dt的小梯
形面积累加得到的
t(end-4:end)
                   % end指示最后一个元素的位置
%Sx(end-4:end)
%plot(t,Sx,'.k','MarkerSize',12)
```

(2) 数值计算解法

dt=0.05; t=0:dt:5; Ft=t.^2.*cos(t);

Sx=dt*cumtrapz(Ft); % 小梯形面积的累加求Ft曲线下的面积,由一个个宽度为dt的小梯形面积累加得到的

t(end-4:end) % end指示最后一个元素的位置

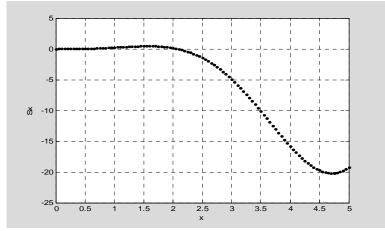


图 3.1-1 在区间[0,5]采样点上算得的定积分值

【例 3.1-2】已知
$$f(t) = e^{-\sin(t)}$$
, 求 $s(x) = \int_0^4 f(t) dt$.

(1) 符号计算解法

syms t x
ft=exp(-sin(t))
sx=int(ft,t,0,4)

(2) 数值计算解法

dt=0.05; t=0:dt:4; Ft=exp(-sin(t)); Sx=dt*cumtrapz(Ft); Sx(end) plot(t,Ft,'*r','MarkerSize',4) hold on plot(t,Sx,'.k','MarkerSize',15) hold off xlabel('x') legend('Ft','Sx')

3.2 数值数组的创建和寻访

3.2.1 一维数组的创建

x=[1,3,5,7,9] 逐个元素输入法

x=a:inc:b 冒号生成法,inc <mark>缺省</mark>时步长为 1

x=linspace(a,b,n) 线性定点法

x=logspace(a,b,n) 对数定点法 运用 diag, eye 等标准数组生成函数。

【例 3.2-1】一维数组的常用创建方法举例。 al=1:6 a2=0:pi/4:pi a3=1:-0.1:0

b1=linspace(0,pi,4)
b2=logspace(0,3,4) %创建数组[10° 10¹ 10² 10³]
c1=[2 pi/2 sqrt(3) 3+5i]
rand('state',0) %利用标准数组生成函数产生均匀分布随机数组c2=rand(1,5)

3.2.2 二维数组的创建

一 小规模数组的直接输入法

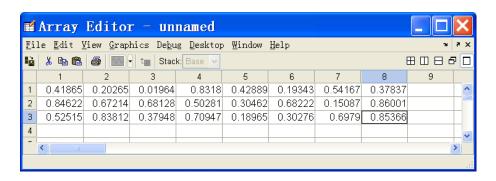
【例 3.2-2】在 MATLAB 环境下,用下面三条指令创建二维数组 C。a=2.7358; b=33/79; C=[1,2*a+i*b,b*sqrt(a);sin(pi/4),a+5*b,3.5+i]

三个要素:整个输入数组 "[]"

行与行间 ";"或"Enter" 同行中元素间 ","或"空格"

二 中规模数组的数组编辑器创建法

【例 3.2-3】根据现有数据创建一个(3×8)的数组。



三 中规模数组的M文件创建法

【例 3.2-4】创建和保存数组 AM 的 MyMatrix.m 文件。

- (1) 打开文件编辑调试器,并在空白填写框中输入所需数组(见图 3.2-2)。
- (2) 最好,在文件的首行,编写文件名和简短说明,以便查阅(见图 3.2-2)。
- (3) 保存此文件,并且文件起名为 MyMatrix.m。
- (4) 以后只要在 MATLAB 指令窗中,运行 MyMatrix.m 文件,数组 AM 就会自动生成于 MATLAB 内存中。

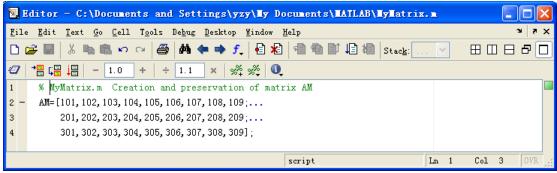


图 3.2-2 利用 M 文件创建数组

四 利用MATLAB函数创建数组

【例 3.2-5】利用最常用标准数组生成函数产生标准数组的演示。

```
ones(2,4) %产生(2×4)全1数组
ans =
1 1 1 1 1
1 1 1
```

```
randn('state',0) %把正态随机数发生器置 0
randn(2,3) %产生正态随机阵
ans =
    -0.4326    0.1253    -1.1465
    -1.6656    0.2877    1.1909
```

diag(D) %取D阵的对角元

diag(diag(D)) %外diag利用一维数组生成对角阵

randsrc(3,20,[-3,-1,1,3],1) %在[-3,-1,1,3]上产生 3×20 均布随机数组,随机发生器的状态设置为 1

3.2.3 二维数组元素的标识和寻访

【例 3.2-6】本例演示:数组元素及子数组的各种标识和寻访格式;冒号的使用; end 的作用。A=zeros(2,6)

A(2,4) % 全下标法:指定行、指定列 **A(8)** % 单下标法:单下标寻访

A(:,[1,3]) % 全下标法:全部行、指定列

A([1,2,5,6]') % 单下标法:生成指定的一维行(或列)数组

A(:,4:end) % 全下标法:全部行、指定列,end表示最后一列。

A(2,1:2:5)=[-1,-3,-5] % 全下标法: 指定行、指定列

3.2.4 数组操作技法综合

【例 3.2-7】**数组操作函数** reshape, diag, repmat 的用法; **空阵**[] 删除子数组的用法。 **a=1:8**

A=reshape(a,4,2)

A=reshape(A,2,4) %改变行数和列数

D1=repmat(B,2,4) %排列B模块repmat(A,m,n) creates a large matrix B consisting of an m-by-n tiling of copies of A.

D1([1,3],:)=[] %删除指定行

【例 3.2-8】函数 flipud, fliplr, rot90 对数组的操作体现着"矩阵变换"。

A=reshape(1:9,3,3)

B=flipud(A) %上下对称交换

C=fliplr(A) %左右对称交换

D=rot90(A,2) %逆时针旋转90度,2次

3.3 数组运算

MATLAB 面向数组/矩阵编程和运算:

- ▶ 用"数组或矩阵运算"模式去处理那些"借助循环而反复执行的标量运算"
- 显著提高程序执行速度
- 书写简洁、便于阅读

数组/矩阵运算符:见MATLAB帮助

(在 Index 窗中输入+)

服从数组运算规则的函数及其它算符

		举例
服从数组	三角、反三角	sin, cos, tan, cot, sec, csc, asin, acos, atan, acot,
运算规则		asec, acsc
的函数	双曲、反双曲	sinh, cosh,, asinh, acosh,
	指数、对数	exp, sqrt, pow2, log, log10, log2
	圆整、求余	ceil, floor, fix, round, mod, rem
	模、角、虚实部	abs, angle, real, imag, conj
	符号函数	sign
	关系运算符	==, ~=, >, <, >=, <=
	逻辑运算符	&, , ~

数组运算和向量化编程

尽可能用"数组或矩阵运算"指令

【例 3.3-1】欧姆定律: $r=\frac{u}{i}$,其中 r,u,i 分别是电阻(欧姆)、电压(伏特)、电流(安培)。验证实验: 据电阻两端施加的电压,测量电阻中流过的电流,然后据测得的电压、电流计算平均电阻值。(测得的电压电流具体数据见下列程序)。

```
(1) 非向量化程序
vr=[0.89, 1.20, 3.09, 4.27, 3.62, 7.71, 8.99, 7.92, 9.70, 10.41];
ir=[0.028, 0.040, 0.100, 0.145, 0.118, 0.258, 0.299, 0.257, 0.308,
0.345];
% -----
L=length(vr);
for k=1:L
  r(k)=vr(k)/ir(k);
end
% -----
sr=0;
for k=1:L
  sr=sr+r(k);
end
rm=sr/L
rm =
  30.5247
(2) 向量化程序
clear
vr=[0.89, 1.20, 3.09, 4.27, 3.62, 7.71, 8.99, 7.92, 9.70, 10.41];
ir=[0.028, 0.040, 0.100, 0.145, 0.118, 0.258, 0.299, 0.257, 0.308,
0.345];
r=vr./ir
                         % 注意: 运算发生在两数组相同位置元素间
                         % MATLAB现成的求平均函数
rm=mean(r)
  31.7857 30.0000 30.9000 29.4483 30.6780 29.8837 30.0669
30.8171 31.4935 30.1739
rm =
  30.5247
```

【例 3.3-2】用间距为 0.1 的水平线和垂直线均匀分割 $x \in [-5,5]$, $y \in [-2.5,2.5]$ 的矩形域,在所有水平线和垂直线交点上计算函数 $z = \sin|xy|$ 的值,并图示。

(1) 非向量化编程

```
clear
x=-5:0.1:5;
y=(-2.5:0.1:2.5)';
N=length(x);
M=length(y);
for ii=1:M
   for jj=1:N
      X0(ii,jj)=x(jj); %所有格点的x坐标
      YO(ii,jj)=y(ii); %所有格点的y坐标
      Z0(ii,jj)=sin(abs(x(jj)*y(ii))); %所有格点的函数值
   end
end
(2) 向量化编程
x=-5:0.01:5;
y=(-2.5:0.01:2.5);
[X,Y]=meshgrid(x,y); % 指定矩形域内所有格点的(x,y)坐标
                 % 数组运算计算矩形域所有格点坐标(x,y)对应的函数值
Z=sin(abs(X.*Y));
                                           %注意:函数f(•)对数
```

(3) 比较二维双精度数是否相等

norm(Z-Z0) %范数接近eps,认为相等。

```
(4) 绘图 surf(X,Y,Z) xlabel('x') ylabel('y') shading interp view([190,70])
```

组的逐个元素起作用。

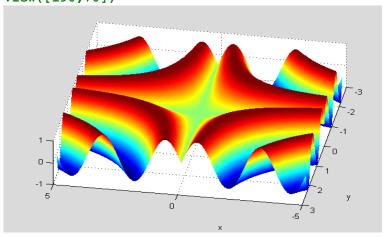


图 3.3-1 指定域上的二元函数图形

3.4 "非数"和"空"数组

◆ MATLAB 中特有的两个概念和"预定义变量"

3.4.1 非数NaN (或记为nan)

由
$$\frac{0}{0}$$
, $\frac{\infty}{\infty}$, $0\times\infty$ 等运算产生。

NaN 的性质:

- NaN 参与运算所得的结果也是 NaN, 即具有传递性;
- NaN 没有"大小"概念,不能比较两个 NaN 的大小。

NaN 的功用:

- 真实记述 $\frac{0}{0}$, $\frac{\infty}{\infty}$, $0 \times \infty$ 等运算的后果;
- 避免可能因 $\frac{0}{0}$, $\frac{\infty}{\infty}$, $0\times\infty$ 等运算而造成程序执行的中断;
- 在测量数据处理中,可以用来标识"野点(非正常点)";
- 在数据可视化中,可以裁剪图形。

【例 3.4-1】非数的产生和性质演示。

(1) 非数的产生

a=0/0,b=0*log(0),c=inf-inf

a =

```
NaN
  NaN
C =
  NaN
(2) 非数的传递性
0*a,sin(a)
ans =
 NaN
ans =
  NaN
(3) 非数的属性判断
class(a)
           % 唯一判断非数的指令
isnan(a)
ans =
double
ans =
    1
【例 3.4-2】非数元素的寻访。
rand('state',0) % 将随机发生器置 0
R=rand(2,5);R(1,5)=NaN;R(2,3)=NaN
   0.9501
         0.6068 0.8913 0.4565
                                      NaN
   0.2311
          0.4860
                      NaN
                            0.0185
                                    0.4447
                  % 对数组元素是否非数进行判断
LR=isnan(R)
LR =
           0 0
        0
    0
    0
        0
             1
                   0
                        0
                          % 确定非零数的"单下标"标识Find indices and
si=find(LR)
values of nonzero elements
                          % 转换成"全下标"标识
[ri,ci]=ind2sub(size(R),si)
[rj,cj]=find(LR)
                          % 直接确定非数的全下标
disp('非数在二维数组R中的位置')
disp(['单下标时的第',int2str(si(1)),'和第',int2str(si(2)),'个元素'])
si =
    9
ri =
ci =
rj =
    1
cj =
    3
非数在二维数组R中的位置
```

单下标时的第6和第9个元素

3.4.2 "空"数组

- ▶ 定义:数组的某维长度为0或若干维长度均为0。
- MATLAB 为操作和表述需要专门设计的一种数组。

```
【例 3.4-3】关于"空"数组的算例。
(1) 创建"空"数组的几种方法
a=[] % 二维"空"(行、列长度均为 0) 数组
b=ones(2,0)
c=zeros(2,0) % "空"数组不同于全零数组
d=eye(2,0)
f=rand(2,3,0,4)
a =
        []
b =
        Empty matrix: 2-by-0
c =
        Empty matrix: 2-by-0
f =
        Empty matrix: 2-by-0
f =
        Empty array: 2-by-3-by-0-by-4
```

(2) "空"数组的属性 % 数据类型 class(a) % 是数值数组吗? isnumeric(a) % 唯一可正确判断数组是否"空"的指令 isempty(a) ans = double ans = 1 ans = 1 % a是什么 which a % 维数 ndims(a) 注意: "空"数组并非"虚无",它确实存在。 % 数组大小 size(a) a is a variable. ans = ans = 0 0

(3) "空"数组用于子数组的删除和大数组的大小收缩

3.5 关系操作和逻辑操作

MATLAB 约定:

- 关系(或逻辑)表达式中,作为输入的任何非 0 数都被看作是"逻辑真",而只有 0 才被认为是"逻辑假"。
- 计算结果,即输出,是由 0 和 1 组成的"逻辑数组(Logical Array)"。在此数组中的 1 表示"真", 0 表示"假"。
- 逻辑数组是一种特殊的数值数组。与"数值类"有关的操作和函数对它同样适用。它表示着对事物的判断结论"真"与"假"。它有自身的特殊用途,如数组寻访等。

3.5.1 关系操作

关系操作符

指令	含义
<	小于
<=	小于等于
>	大于
>=	大于等于
==	等于
~	不等于

【例 3.5-1】关系运算示例。

A=1:9,B=10-A

r0=(A<4) % 比较在数组每个元素与标量之间进行,比较结果将与参与比较的数组同维。 r1=(A==B) % 进行比较的两数组必须维数相同,比较在两数组相同位置上的元素间进行。

A =									
	1	2	3	4	5	6	7	8	9
B =									
	9	8	7	6	5	4	3	2	1
r0 =									
	1	1	1	0	0	0	0	0	0
r1 =									
	0	0	0	0	1	0	0	0	0

【例 3.5-2】关系运算应用。

t=-3*pi:pi/10:3*pi; % 自变量数组中,存在 0 值 y=sin(t)./t; % 在t=0 处,将产生非数。

tt=t+(t==0)*eps; % 逻辑数组参与运算, 使 0 元素被一个"机器零"小数代替

yy=sin(tt)./tt; % 用sin(eps)/eps近似替代sin(0)/0 极限

subplot(1,2,1),plot(t,y),axis([-9,9,-0.5,1.2]), % 非数在绘图中的剪裁作用

xlabel('t'),ylabel('y'),title('残缺图形')

subplot(1,2,2),plot(tt,yy),axis([-9,9,-0.5,1.2])

xlabel('tt'),ylabel('yy'),title('正确图形')

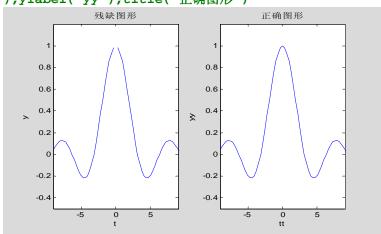


图 3.5-1 采用近似极限处理前后的图形对照

3.5.2 逻辑操作

逻辑操作符

指令	含义
&	与
	或
~	非
xor	异或

逻辑操作的引入, 使复杂关系运算成为可能。

- 标量可以与任何维数组进行逻辑运算。运算比较在标量与数组每个元素间进行,运算结果与参与运算的数组同维。
- 当逻辑运算中没有标量时,参与运算的数组必须维数相同。逻辑运算 在两数组相同位置上的元素间进行

【例 3.5-3】逻辑操作和关系操作。本例演示:逻辑、关系操作的组合; xor的作用。

(1) 逻辑、关系操作的组合

A=[-2,-1,0,0,1,2,3]

L1=~(A>1) % 哪些元素不大于 1?

L2=(A>0)&(A<2) % 哪些元素大于 0 且小于 2?

A =							
	-2	-1	0	0	1	2	3
L1	=						
	1	1	1	1	1	0	0
L2	=						
	0	0	0	0	1	0	0

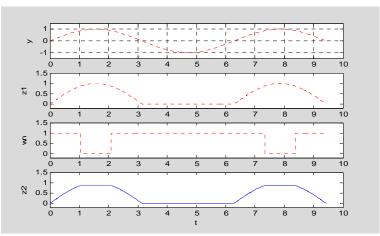
(2) xor 的作用

A,B=[0,-1,1,0,1,-2,-3]

C=xor(A,B) %当A, B数组中,两个对应元素中仅一个为 0 时,给出 1。否则为 0。

【例 3.5-4】试绘制如图 3.5-2 最下那幅子图所示的"**正弦波** $\sin t$ **的削顶半波整流波形**",削顶发生在每个周期的[60° , 120°]之间。本例演示:逐段解析函数的计算和表现。clear,t=linspace(0,3*pi,500);y= $\sin(t)$;

```
z1=((t<pi)|(t>2*pi)).*y; % 获得整流半波
w=(t>pi/3&t<2*pi/3)+(t>7*pi/3&t<8*pi/3); % 关系逻辑运算和数值运算
wn=~w;
z2=w*sin(pi/3)+wn.*z1; % 获得削顶整流半波
subplot(4,1,1),plot(t,y,':r'),axis([0,10,-1.5,1.5])
ylabel('y'),grid on
subplot(4,1,2),plot(t,z1,':r'),axis([0,10,-0.2,1.5]),ylabel('z1')
subplot(4,1,3),plot(t,wn,':r'),axis([0,10,-0.2,1.5]),ylabel('wn')
subplot(4,1,4),plot(t,z2,'-b'),axis([0,10,-0.2,1.5]),ylabel('z2')
xlabel('t')
```



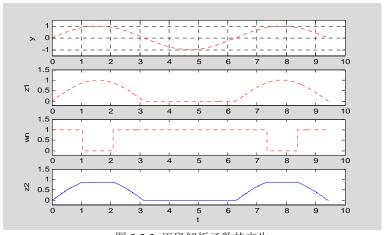


图 3.5-2 逐段解析函数的产生

3.5.3 常用逻辑函数

MATLAB中能给出"逻辑数组"类型计算结果的逻辑函数很多,常用的如下表:

表 3.5-3 常用逻辑函数

V 76 H T H T T T T T T T T T T T T T T T T						
分类	具体描述					
含0数组判断	all	数组 A 不含 0 元素,返回 1				
百0数组列则	any	数组 A 不是全 0 元素,返回 1				
	false	按指定大小, 创建全0逻辑数组				
生成逻辑数组	true	按指定大小, 创建全1逻辑数组				
	logical	创建逻辑数组; 1对应输入数组中的非0元素,其余都为0				
	isempty	是否空阵	isprime	是否质数		
数据对象判断	isfinite	是否有限数	isreal	是否实数		
数据对参判图	isinf	是否无穷大	isletter	是否字母(用于字符串)		
	isnan	是否非数	isspace	是否空格 (用于字符串)		
数据类型判断	isa	是否指定类别	ishandle	是否图柄		
	ischar	是否字符串	islogical	是否逻辑类型		
	isglobal	是否全局变量	isnumeric	是否数值类型		

可通过 MATLAB 帮助系统获得上述常用逻辑函数的具体用法。

- 1. 要求在闭区间 $[0,2\pi]$ 上产生具有 10 个等距采样点的一维数组。试用两种不同的指令实现。
- 2. 由指令 rand('state',0),A=rand(3,5)生成二维数组 A,试求该数组中所有大于 0.5 的元素的位置,分别求出它们的"全下标"和"单下标"。 《答案》

大于 0.5 的元素的全下标

行号 1 3 2 3 3 2 3 1 2 列号 1 1 2 2 3 4 4 5 5

大于 0.5 的元素的单下标

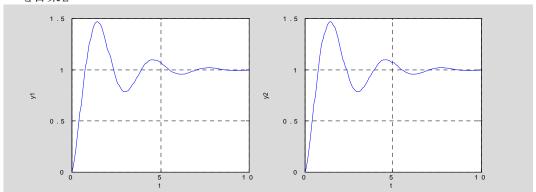
1 3 5 6 9 11 12 13 14

3. 在使用 123 作为 rand 随机数发生器的初始化状态的情况下,写出产生长度为 1000 的"等概率双位(即取-1,+1)取值的随机码"程序指令,并给出-1码的数目。 〖答案〗

Na = 490

- 4. 已知矩阵 $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$,运行指令 $B1=A.^{(0.5)}$, $B2=A^{(0.5)}$,可以观察到不同运算方法所得结果不同。(1)请分别写出根据 B1,B2 恢复原矩阵 A 的程序。(2)用指令检验所得的两个恢复矩阵是否相等。
- 5. 在时间区间 [0,10]中,绘制 $y = 1 e^{-0.5t}\cos 2t$ 曲线。要求分别采取"标量循环运算法"和"数组运算法"编写两段程序绘图。

〖答案〗

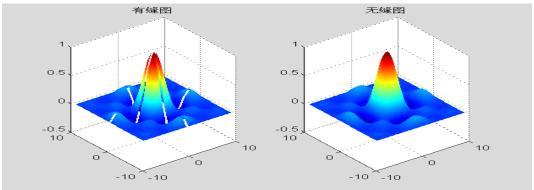


- 6. 先运行 clear,format long,rand('state',1),A=rand(3,3),然后根据 A 写出两个矩阵: 一个对角阵 B,其相应元素由 A 的对角元素构成; 另一个矩阵 C,其对角元素全为 0,而其余元素与对应的 A 阵元素相同。
- 7. 先 运 行 指 令 x=-3*pi:pi/15:3*pi; y=x; [X,Y]=meshgrid(x,y); warning off; Z=sin(X).*sin(Y)./X./Y; 产生矩阵 Z。(1)请问矩阵 Z 中有多少个"非数"数据?(2)用指令 surf(X,Y,Z); shading interp 观察所绘的图形。(3)请写出绘制相应的"无裂缝"

1

〖答案〗

NumOfNaN =



8. 下面有一段程序,企图用来解决如下计算任务:有矩阵
$$\mathbf{A}_k = \begin{bmatrix} 1 & k+1 & \cdots & 9k+1 \\ 2 & k+2 & \cdots & 9k+2 \\ \vdots & \vdots & \vdots & \vdots \\ k & 2k & \cdots & 10k \end{bmatrix}$$

当k 依次取 10,9,8,7,6,5,4,3,2,1 时, 计算矩阵 \mathbf{A}_k "各列元素的和", 并把此求和结

果存放为矩阵 Sa 的第
$$k$$
 行。例如 $k=3$ 时, A 阵为 $\begin{bmatrix} 1 & 4 & \cdots & 28 \\ 2 & 5 & \cdots & 29 \\ 3 & 6 & \cdots & 30 \end{bmatrix}$,此时它各列元素

的和是一个 (1×10) 行数组 $[6\ 15\ \cdots\ 87]$,并把它保存为 Sa 的第 3 行。问题:该段程序的计算结果对吗?假如计算结果不正确,请指出错误发生的根源,并改正之。

for k=10:-1:1
 A=reshape(1:10*k,k,10);

Sa(k,:)=sum(A);

end

Sa Sa =

第4章 数值计算

- 与符号计算相比,**数值计算**在科研和工程中的应用更为广泛。
 - ▶ 随着科研领域、工程实践的数字化进程的深入,具有数字化本质的数值 计算就显得愈益重要。
- 和十年、二十年前相比较,在当今计算机软硬件的支持下,人们所能拥有的 **计算能力**已经得到了**巨大的提升**。这就**自然地激发**人们从新的计算能力出发 去学习、理解概念,**鼓舞**人们用新计算能力试探解决实际问题的雄心。
- MATLAB 正是凭借其**卓越的数值计算能力**而称雄世界。

4.1 数值微积分

4.1.1 近似数值极限及导数

MATLAB 数值计算中,没有专门的求极限和导数的指令。 原因:数值精度有限,不能表示无穷小量,不能准确描述一个数的邻域。

```
【例 4.1-1】设 f_1(x)=\frac{1-\cos 2x}{x\sin x}, f_2(x)=\frac{\sin x}{x},试用机器零阈值 eps 替代理论 0 计算极限 L_1(0)=\lim_{x\to 0}f_1(x), L_2(0)=\lim_{x\to 0}f_2(x)。
```

理论分析表明:
$$f_1(x) = 2f_2(x)$$
, $L_2(0) = \lim_{x \to 0} f_2(x) = \lim_{x \to 0} \frac{\sin x}{x} = 1$.

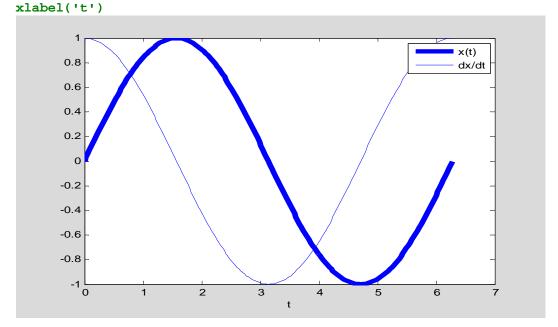
- 借助符号计算所求的极限与理论值一致;
- 用数值法近似计算的极限与理论不一致。

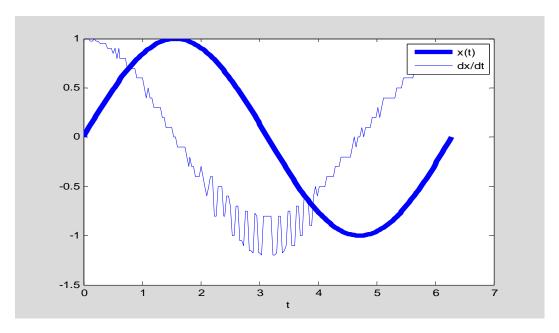
提醒:除非数值近似法求的极限经过理论验证,否则绝不要借助数值法求取极限。 函数 y = f(x) 在点 x_0 处的 导数

$$y'\Big|_{x=x_0} = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x} = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}$$

【例 4.1-2】已知 $x = \sin(t)$, 求该函数在区间 $[0, 2\pi]$ 中的近似导函数。

d=pi/100; t=0:d:2*pi; x=sin(t); dt=1000*eps; x_eps=sin(t+dt);% 自变量的增量为 5eps。dxdt_eps=(x_eps-x)/dt;% 近似数值导数 plot(t,x,'LineWidth',5) hold on plot(t,dxdt_eps) hold off legend('x(t)','dx/dt')





4.1-1 增量过小引起有效数字严重丢失后的毛刺曲线

```
x_d=sin(t+d);
dxdt_d=(x_d-x)/d; %以d=pi/100 为增量算得的数值导数
plot(t,x,'LineWidth',5)
hold on
plot(t,dxdt_d)
hold off
legend('x(t)','dx/dt')
xlabel('t')
```

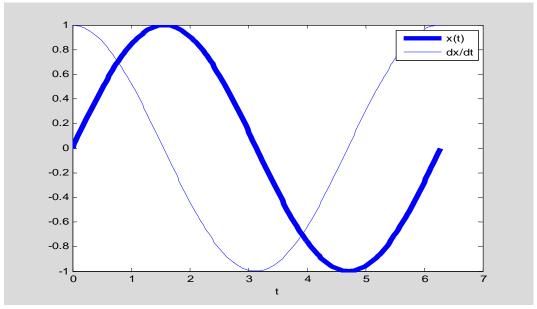


图 4.1-2 增量适当所得导函数比较光滑

结论: 自变量增量的选取一定要避免太小。

```
当 X 是矩阵时, dx = X(2:n,:) - X(1:n-1,:);
         dx 的长度比 X 的"长度"短一个元素(或一行)。
x = [1 \ 2 \ 3 \ 4 \ 5]:
y = diff(x)
v =
                1
                      1
     1
          1
diff(X,2) is the same as diff(diff(X))
z = diff(x, 2)
          0
     0
                 0
        FX = gradient(F)
                           求梯度
        [FX,FY] = gradient(F)
         当 F 是向量时, FX(1) = F(2) - F(1),
                     FX(end) = F(end) - F(end-1),
                     FX(2:end-1) = (F(3:end) - F(1:end-2))/2,
        FX的"长度"与F相同。
         当F是矩阵时, FX和FY是与F同样大小的矩阵。
                     FX 的每行给出 F 相应行元素间的"梯度";
                      FY 的每列给出 F相应列元素间的"梯度"。
【例 4.1-3】已知 x = \sin(t),采用 diff 和 gradient 计算该函数在区间 [0, 2\pi] 中的近似导函
数。
clf
d=pi/100;
t=0:d:2*pi;
x=sin(t);
dxdt_diff=diff(x)/d;
dxdt_grad=gradient(x)/d;
                   % 宏观上看,diff和gradient所求得近似导数大体相同
subplot(1,2,1)
plot(t,x,'b')
hold on
plot(t,dxdt_grad,'m','LineWidth',8)
plot(t(1:end-1),dxdt_diff,'.k','MarkerSize',8)
axis([0,2*pi,-1.1,1.1])
title('[0, 2\pi]')
legend('x(t)','dxdt_{grad}','dxdt_{diff}','Location','North')
xlabel('t'),box off
hold off
subplot(1,2,2)
kk=(length(t)-10):length(t); % 最后 11 个数据的下标
hold on
plot(t(kk),dxdt_grad(kk),'om','MarkerSize',8)
                                                  % 微观上看,不仅数值上有差异,
plot(t(kk-1),dxdt_diff(kk-1),'.k','MarkerSize',8)% 而且diff没有给出最后一点的导数。
title('[end-10, end]')
legend('dxdt_{grad}','dxdt_{diff}','Location','SouthEast')
xlabel('t'),box off
hold off
```

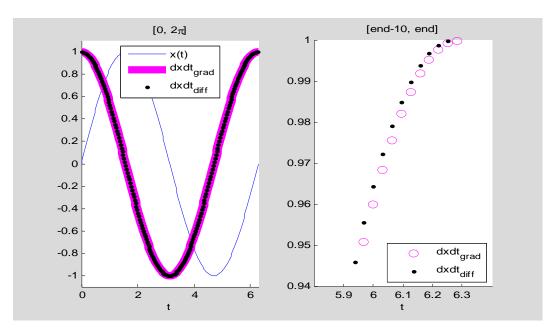


图 4.1-3 diff 和 gradient 求数值近似导数的异同比较

4.1.2 数值求和与近似数值积分

Sx=sum(X) 沿列方向求和 $Sx(k) = \sum_{i=1}^{m} X_{m \times n}(i,k)$

Scs=cumsum(X) 沿列方向求累计和

St=trapz(x,y) 采用梯形法沿列方向求函数 y 关于自变量 x 的积分 Sct=cumtrapz(x,y) 采用梯形法沿列方向求函数 y 关于自变量 x 的累计积分 采样点数愈多,精度越高,但无法定量控制积分精度。

【例 4.1-4】求积分 $s(x) = \int_0^{\pi/2} y(t) dt$, 其中 $y = 0.2 + \sin(t)$ 。

```
clear
d=pi/8;
t=0:d:pi/2;
y=0.2+sin(t);
s=sum(y);
s_sa=d*s;
s_ta=d*trapz(y);
                      % 可用s_ta=trapz(t,y)替换,此通用格式适用于"非等间隔采样"的情况。
disp(['sum求得积分',blanks(3),'trapz求得积分'])
disp([s_sa, s_ta])
t2=[t,t(end)+d];
y2=[y,nan];
stairs(t2,y2,':k')
hold on
plot(t,y,'r','LineWidth',3)
h=stem(t,y,'LineWidth',2);
set(h(1),'MarkerSize',10)
axis([0,pi/2+d,0,1.5])
                           %由图可见: 阶梯虚线所占的自变量区间比积分区间多一个采样子区间。
hold off
shg %show graph window
```

sum 求得积分 trapz 求得积分

1.5762 1.3013

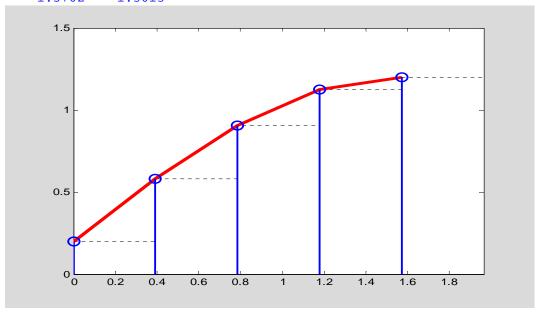


图 4.1-4 sum 和 trapz 求积模式示意

hlines = stem('v6',...) returns the handles of line objects instead of stemseries objects for compatibility with MATLAB 6.5 and earlier.

hlines contains the handles to three line graphics objects:

 $\mbox{hlines(1)}$ — The marker symbol at the top of each stem

hlines(2) — The stem line

hlines(3) — The baseline handle

采样点数愈多,精度越高。

存在的问题:无法预先设置欲求积分的精度。

4.1.3 计算精度可控的数值积分

数值积分:

闭型算法: 需要计算积分区间端点处的函数值 开型算法: 不需要计算积分区间端点处的函数值

闭型数值积分指令:

S1=quad(fun, a, b, tol) 采用递推自适应 Simpson 法计算积分 S1=quadl(fun, a, b, tol) 采用递推自适应 Lobatto 法计算积分

S2=dblquad(fun, xmin, xmax, ymin, ymax, tol)
二重数值积分

S3=triplequad(fun, xmin, xmax, ymin, ymax, zmin, zmax, tol) 三重数值积分

被积函数 fun:字符串、匿名函数、M函数文件的函数句柄和内联对象被积函数的自变量一般采用字母 x。

tol: 用来控制绝对误差,默认积分绝对精度为 10⁻⁶。

【例 4.1-6】求
$$s = \int_{1}^{2} \int_{0}^{1} x^{y} dx dy$$
。

syms x y
s=vpa(int(int(x^y,x,0,1),y,1,2))
s = .40546510810816438197801311546432
% 符号计算结果

format long
s_n=dblquad('x.^y',0,1,1,2) %被积函数一定要写成"数组运算"格式
s_n = 0.405466267243508
% 数值积分结果

4.1.4 函数极值的数值求解

[x,fval,exitflag,output]=fminbnd(fun, x1, x2, options) 求一元函数在区间(x1, x2)中极小值

[x,fval,exitflag,output]=fminsearch(fun, x0, options) 求多元函数极值点

fun: 字符串、匿名函数、M函数文件的函数句柄和内联对象

options:用于配置优化参数,一般不必改动。

x: 极值点

fval: 目标函数极值

```
【例 4.1-7】已知 y = (x + \pi) \cdot e^{|\sin(x + \pi)|}, 在-\pi/2 \le x \le \pi/2区间,求函数的极小值。
(1) 符号法
syms x
y=(x+pi)*exp(abs(sin(x+pi)));
                             % 求导函数
yd=diff(y,x);
                             % 求导函数为 0 的自变量值
xs0=solve(yd)
yd_xs0=vpa(subs(yd,x,xs0),6) % 导函数在xs0 处是否为 0?
                            % 和边界点函数值比较
y_xs0=vpa(subs(y,x,xs0),6)
y_m_{pi=vpa(subs(y,x,-pi/2),6)}
y_p_pi=vpa(subs(y,x,pi/2),6)
Warning: Warning, solutions may have been lost
xs0 =
-1.0676598444985783372948670854801
yd_xs0 =
.1e-4
y_xs0 =
4.98043
y_m_pi =
4.26987
y_p_pi =
12.8096
(2) 采用优化算法
x1=-pi/2;x2=pi/2;
yx=@(x)(x+pi)*exp(abs(sin(x+pi))); % 采用匿名函数(6.3.3)形式定义函数
[xn0,fval,exitflag,output]=fminbnd(yx,x1,x2)
xn0 =
 -1.2999e-005
fval =
    3.1416
exitflag =
                  % 给出大于 0 的数,表明成功搜索到极值点。
    1
output =
   iterations: 21 % 给出迭代次数和优化算法等信息
    funcCount: 22
    algorithm: [1x46 char]
      message: [1x112 char]
(3) 图形法
xx=-pi/2:pi/200:pi/2;
yxx=(xx+pi).*exp(abs(sin(xx+pi)));
plot(xx,yxx)
                   % 从函数变化趋势可知: (1)导函数不连续,函数的极小值在导函数不连续点;
xlabel('x'),grid on % (2) x=-1 附近,存在导数为 0 的点,有极大值存在。回顾解法(1)、(2)。
```

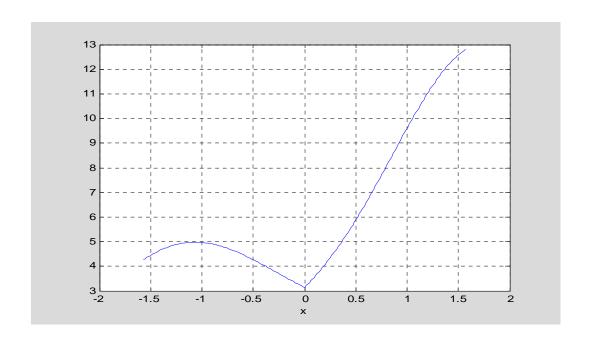


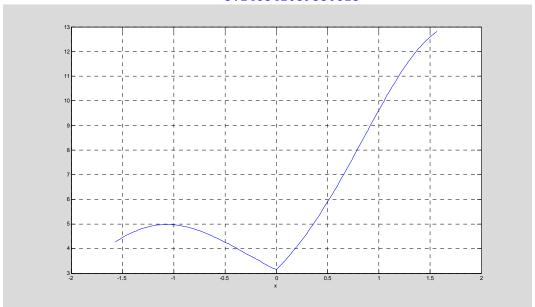
图 4.1-5 在[-pi/2,pi/2]区间中的函数曲线

[xx,yy]=ginput(1)

% 用鼠标从图形上获取点的坐标(x,y)(参见 5.2.4)

xx = -0.002016129032258

yy = 3.148342059336823



4.1-6 函数极值点附近的局部放大和交互式取值

数值计算涉及到的问题:

数值微积分

矩阵和代数方程

概率分布和统计分析

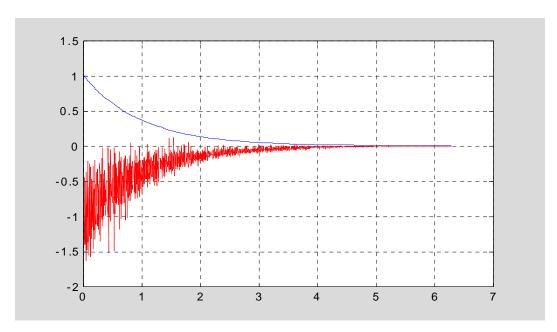
多项式运算和卷积

Fourier 分析

等等

1. 根据题给的模拟实际测量数据的一组t和 y(t)试用数值差分 diff 或数值梯度 gradient 指令计算 y'(t),然后把 y(t)和 y'(t)曲线绘制在同一张图上,观察数值求导的后果。(模拟数据从 prob_data401.mat 获得)

〖答案〗

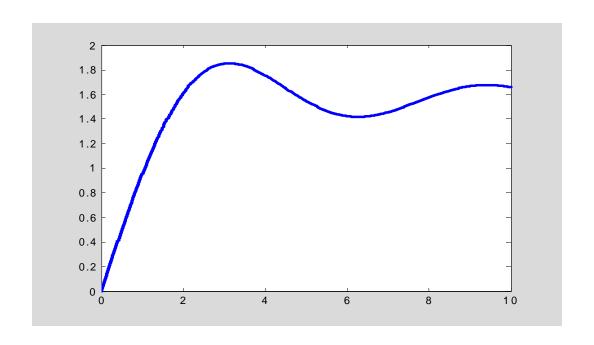


2. 采用数值计算方法,画出 $y(x) = \int_0^x \frac{\sin t}{t} dt$ 在[0,10] 区间曲线,并计算 y(4.5) 。

〖答案〗

s45 =

1.6541



3. 求函数 $f(x) = e^{\sin^3 x}$ 的数值积分 $s = \int_0^\pi f(x) dx$, 并请采用符号计算尝试复算。

〖答案〗

3 =

5.1370

Warning: Explicit integral could not be found.
> In sym.int at 58
ss =
int(exp(sin(x)^3),x = 0 .. pi)

4. 用 quad 求取 $\int_{-5\pi}^{1.7\pi} e^{-|x|} |\sin x| dx$ 的数值积分,并保证积分的绝对精度为 10^{-9} 。

〖答案〗

sa =

1.08784993815498

5. 求函数 $f(t) = (\sin 5t)^2 e^{0.06t^2} - 1.5t \cos 2t + 1.8 | t + 0.5 |$ 在区间 [-5,5] 中的最小值点。

〖答案〗

最小值点是 -1.28498111480531 相应目标值是 -0.18604801006545

6. 设 $\frac{d^2y(t)}{dt^2} - 3\frac{dy(t)}{dt} + 2y(t) = 1$, y(0) = 1, $\frac{dy(0)}{dt} = 0$, 用数值法和符号法求 $y(t)|_{t=0.5}$ 。

〖答案〗

数值解

 $y_05 =$

0.78958020790127

符号解 ys = 1/2-1/2*exp(2*t)+exp(t) ys_05 = .78958035647060552916850705213780

7. 已知矩阵 A=magic(8), (1)求该矩阵的"值空间基阵"B; (2)写出"A 的任何列可用基向量线性表出"的验证程序(提示:利用 rref 检验)。

```
〖答案〗
三组不同的基
B1 =
   64
        2
             3
            54
        55
    9
            46
   17
        47
            27
   40
       26
   32
       34
            35
   41
        23
            22
       15
   49
            14
       58 59
    8
B2 =
                   0.3536
  -0.3536 0.5401
  -0.3536 -0.3858 -0.3536
  -0.3536 -0.2315 -0.3536
  -0.3536
           0.0772
                   0.3536
                   0.3536
  -0.3536 -0.0772
          0.2315 -0.3536
0.3858 -0.3536
  -0.3536
  -0.3536
  -0.3536 -0.5401
                   0.3536
B3 =
   0.3536
          0.6270 0.3913
   0.3536 -0.4815
                   -0.2458
   0.3536
         -0.3361
                   -0.1004
           0.1906
   0.3536
                   -0.0451
   0.3536
           0.0451 -0.1906
   0.3536
           0.1004
                   0.3361
                   0.4815
          0.2458
   0.3536
   0.3536 -0.3913 -0.6270
```

第5章 数据和函数的可视化

视觉是人们感受世界、认识自然的最重要途径。

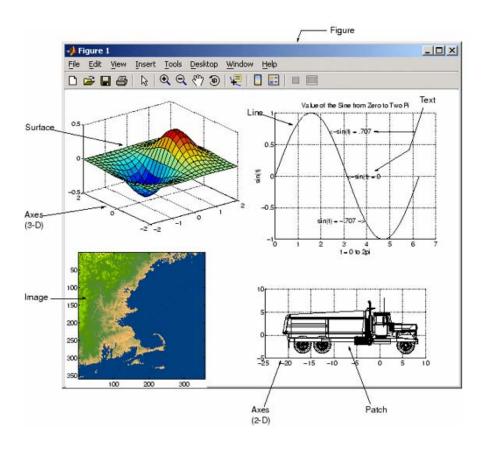
可视化的目的:

通过图形,观察数据间的内在关系,

感受由图形所传递的内在本质。

Core Graphics Objects

Function	Purpose
axes	Axes objects define the coordinate system for displaying graphs. Axes are always contained within a figure.
<u>image</u>	2-D representation of a matrix where numeric values are mapped to colors. Images can also be 3-D arrays of RGB values.
light	Directional light source located within the axes. Lights affect patches and surfaces, but cannot themselves be seen.
line	A line is drawn by connecting the data points that define it.
<u>patch</u>	Filled polygons with separate edge properties. A single patch can contain multiple faces, each colored independently with solid or interpolated colors.
rectangle	2-D object that has settable edge and face color, and variable curvature (can draw ellipses)
<u>surface</u>	3-D grid of quadrilaterals created by plotting the value of each element in a matrix as a height above the x-y plane
text	Character strings positioned in the coordinate system defined by the axes



5.1 引导

离散数据和离散函数的可视化 5.1.1

- 一对**实数标量**(x,y)可表示为平面上的一个点;
- 一对**实数"向量"(x,y)**可表现为平面上的一组点。
- 离散函数可视化的步骤:

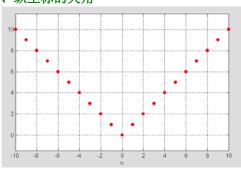
先根据离散函数**特征**选定一组自变量 $\mathbf{x} = [x_1, x_2, \dots, x_N]^T$; 再根据所给离散函数 $y_n = f(x_n)$ 算得相应的 $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$, 然后在平面上几何地表现这组向量对(x,y)。

【例 5.1-1】图形表示离散函数 y = |n|。

n=(-10:10)';%适当选取自变量:通过局部的、非完整的图形最大限度地表现函数的特征 %数组算法计算相应点的函数值 y=abs(n);plot(n,y,'r.','MarkerSize',20) axis equal %为恰当显示和横、纵坐标的夹角

grid on

xlabel('n')



5.1.2 连续函数的可视化

连续函数可视化包含三个重要环节:

- 一. 从连续函数获得一组采样数据, 即选定一组自变量采样点(包括采样的起点、终点和采样步长), 并计算相应的函数值;
- 二. 离散数据的可视化;
- 三. 图形上离散点的连续化:
- (1)增加离散点数量去获得"连续感";
- (2) 线性插值:实质是顺次连接各离散点。MATLAB 在绘制连续曲线时自动完成。 自变量采样点必须**足够多**,且**单调**排列。

【例 5.1-2】用图形表示连续调制波形 $y = \sin(t)\sin(9t)$ 。(图 5.1-2)

```
t1=(0:11)/11*pi;
                               % 12 个采样点
                               % 401 个采样点
t2=(0:400)/400*pi;
t3=(0:50)/50*pi;
                               % 51 个采样点
y1=sin(t1).*sin(9*t1);
y2=sin(t2).*sin(9*t2);
y3=sin(t3).*sin(9*t3);
subplot(2,2,1),plot(t1,y1,'r.')
                                           %<7>
axis([0,pi,-1,1]),title('(1)点过少的离散图形')
subplot(2,2,2),plot(t1,y1,t1,y1,'r.')
                                           %<9>
axis([0,pi,-1,1]),title('(2)点过少的连续图形')
subplot(2,2,3),plot(t2,y2,'r.')
                                           %<11>
axis([0,pi,-1,1]),title('(3)点密集的离散图形')
                                        %<13> MATLAB具有自动"线性
subplot(2,2,4),plot(t3,y3)
axis([0,pi,-1,1]),title('(4)点足够的连续图形') % 插值" 绘制连续曲线的能力。
```

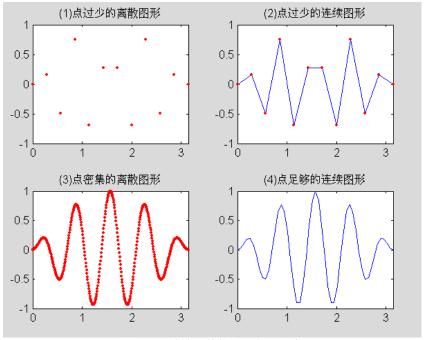


图 5.1-2 连续函数的图形表现方法

【例 5.1-3】绘制奇数正多边形。

% 多边形的边数 N=9;% 0,2pi为同一点 t=0:2*pi/N:2*pi; % 参数方程 x=sin(t);y=cos(t); % 把行向量重排成"二维数组" tt=reshape(t,2,(N+1)/2);% 上下两行互换 tt=flipud(tt); tt=tt(:); xx=sin(tt);yy=cos(tt); subplot(1,2,1),plot(x,y)title('(1) 正常排序图形'),axis equal off,shg subplot(1,2,2),plot(xx,yy)

%自变量必须单调排列

title('(2) 非正常排序图形'),axis equal off,shg

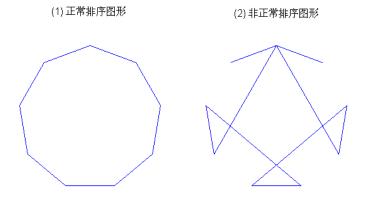


图 5.1-3 自变量排列次序对连续曲线图形的影响

二维曲线和图形 5.2

MATLAB 提供了多种二维图形的绘制指令

表 5.2-1 MATLAB 提供的二维图形绘制指令

指令名	含义	指令名	含义
area	面域图; 主用于表现比例、成份	plot	基本二维曲线图形指令
bar	直方图,主用于 统计 数据	polar	以极坐标绘制曲线
compass	射线图,主用于方向和速度	quiver	二维箭头图,主用于场强、流向
feather	羽毛图,主用于速度	rose	频数扇形图,主用于 统计
hist	频数直方图,主用于 统计	stairs	阶梯图, 主用于采样数据
pie	二维饼图; 统计 数据极坐标形式	stem	二维杆图,主用于离散数据

可通过 MATLAB 帮助系统详细了解各绘图指令。

二维曲线绘制的基本指令plot 5.2.1

一 基本调用格式plot(x,y,'s') x,y是长度相同的一维数组

(x,y,'s'): 平面绘线三元组, 's'缺省时为"蓝色细实线"。

表 5.2-2 离散数据点形允许设置值

符号	含义	符号	含义	符号	含义
d	菱形符	X	叉字符	<	朝左三角符
h	六角星符	•	实心黑点	^	朝右三角符
0	空心圆圈	+	十字符	v	朝下三角符
p	五角星符	*	米字符		
S	方块符	٨	朝上三角符		

表 5.2-3 连续线形允许设置值

符号	含义	符号	含义
-	细实线 (默认)		点划线
:	虚点线		虚划线

表 5.2-4 点线色彩允许设置值

符号	b	g	r	с	m	y	k	w	
----	---	---	---	---	---	---	---	---	--

含义	蓝 (默认)	绿	红	青	品红	黄	黑	白
----	--------	---	---	---	----	---	---	---

二 plot的衍生调用格式

(1) 单色或多色绘制多条曲线

plot(X,Y,'s') 用 s 指定的点形线型色彩绘制多条曲线 plot(X,Y) 采用默认的色彩次序用细实线绘制多条曲线

(2) 多<mark>三元组</mark>绘制多条曲线 plot(X1,Y1,'s1',X2,Y2,'s2', ... ,Xn,Yn,'sn')

(3) 单输入量绘线 **plot(Y)**

三 plot的属性可控调用格式

plot(x, y, 's', 'PropertyName', PropertyValue, ...)

表 5.2-5 线对象的常用属性名(PropertyName)和属性值(PropertyValue)

含义	属性名	属性值	说明
点、线颜色	Color	$[\nu_r, \nu_g, \nu_b]$,RGB三元组中每个元素可在 $[0,1]$ 取任意值。	最常用的色彩可通过表5.2-4中的字母表示常用色彩可通过's'设置,蓝色为默认色彩。
线型	LineStyle	4 种线型参见表 5.2-3	可通过's'设置,细实 线为默认线型
线宽	LineWidth	正实数	默认线宽为 0.5
数据点形	Marker	14 种点型参见表 5.2-2	可通过's'设置
点的大小	MarkerSize	正实数	默认大小为 6.0
点边界颜色	MarkerEdgeColor	[<i>v</i> _r , <i>v</i> _g , <i>v</i> _b],RGB三元组中每 个元素可在[0,1]取任意值。	
点域色彩	MarkerFaceColor	$[v_r, v_g, v_b]$,RGB三元组中每个元素可在 $[0,1]$ 取任意值。	

```
【例 5.2-1】二维曲线绘图指令演示之一。(图 5.2-1)clf
t=(0:pi/50:2*pi)'; % (101*1)
k=0.4:0.1:1; % (1*7)
Y=cos(t)*k; % (101*7)
subplot(1,2,1),plot(t,Y,'LineWidth',1.5)
title('By plot(t,Y)'),xlabel('t')
subplot(1,2,2),plot(Y,'LineWidth',1.5)
title('By plot(Y)'),xlabel('row subscript of Y')
```

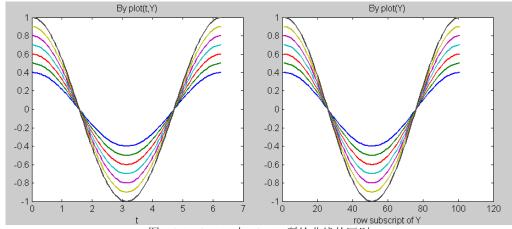


图 5.2-1 plot(t,Y)与 plot(Y)所绘曲线的区别

【例 5.2-2】用图形表示连续调制波形 $y = \sin(t)\sin(9t)$ 及其包络线。

```
t=(0:pi/100:pi)';
                             % (101,1)
                                                    <1>
y1=sin(t)*[1,-1];
                             % (101,2)
                                        包络线函数值
                                                    <2>
y2=sin(t).*sin(9*t);
                             %(101, 1) 调制波列向量
                                                    <3>
t3=pi*(0:9)/9;
                             % (1,10)
                                                    <4>
y3=sin(t3).*sin(9*t3);
                               (1,10)
                                                    <5>
                                                 <6>
plot(t,y1,'r:',t,y2,'-bo')%
hold on
                             % 使后面所绘图形发生在已有的同一张图上 <7>
plot(t3,y3,'s','MarkerSize',10,'MarkerEdgeColor',[0,1,0],'MarkerFaceC
olor',[1,0.8,0])
                                                    <8>
                             %
axis([0,pi,-1,1])
                             %
                                                     <9>
hold off
                             %
                                                    <10>
```

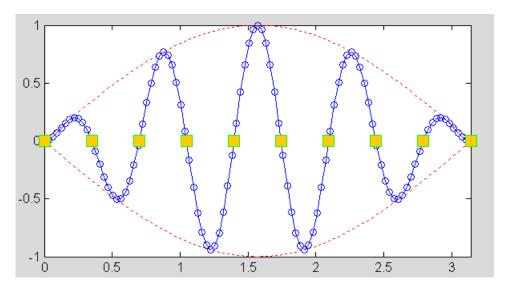


图 5.2-2 属性控制下所绘曲线

5.2.2 坐标控制和图形标识

MATLAB 对图形风格的控制比较完备友善。

一方面,在最通用的层面上,它采用了一系列考虑周全的默认设置,因此在绘制图形时,无需人工干预,它就能根据所给数据自动地确定坐标取向、范围、刻度、高宽比,并给出相当令人满意的画面。

另一方面,在适应用户的层面上,它又给出了一系列便于使用的指令,可让用户根据需要和喜好改变默认设置。

一 坐标轴的控制

在 MATLAB 帮助系统的 Index 窗中输入 axis,点击出现的 axis[1][2][3],再点击右边浏览器中第一行出现的 axis,得到**坐标轴的各种控制指令**。

【例 5.2-3】观察各种轴控制指令的影响。演示采用长轴为 3.25,短轴为 1.15 的椭圆。注意:采用多子图(图 5.2-3)表现时,图形形状不仅受"控制指令"影响,而且受整个图面"宽高比"及"子图数目"的影响。t=0:2*pi/99:2*pi;

x=1.15*cos(t);y=3.25*sin(t); % x为短轴,y为长轴。

subplot(2,3,1),plot(x,y),axis normal,grid on, %默认: 矩形坐标系title('Normal and Grid on')

subplot(2,3,2),plot(x,y),axis equal,grid on,title('Equal') %纵横等长刻度
subplot(2,3,3),plot(x,y),axis square,grid on,title('Square')%正方形坐标
%系

subplot(2,3,4),plot(x,y),axis image,box off,title('Image and Box off') % 纵横轴等长刻度,且坐标框紧贴数据范围

subplot(2,3,5),plot(x,y),axis image fill,box off % 坐标充满整个绘图区title('Image and Fill')

subplot(2,3,6),plot(x,y),axis tight,box off,title('Tight')

% 把数据范围直接设为坐标范围

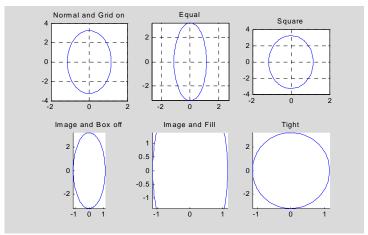


图 5.2-3 各种轴控制指令的不同影响

试用 plot 单独绘出各图,准确体会控制指令的影响。

二 分格线和坐标框

grid 是否画分格线的双向切换指令(使当前分格线状态翻转)

grid on 画出分格线

grid off 不画分格线(默认设置)

box 坐标形式在封闭式和开启式之间切换指令

box on 使当前坐标呈封闭形式(默认设置)

box off 使当前坐标呈开启形式

三 图形标识指令

图形标识包括: 图名(Title)、坐标轴名(Label)、图形注释(Text)和图例(Legend)。标识指令的最简捷使用格式如下:

title(S)书写图名xlabel(S)横坐标轴名ylabel(S)纵坐标轴名

legend(S1,S2,...) 绘制曲线所用线型、色彩或数据点形图例 text(xt,yt,S) 在图面(xt, yt)坐标处书写字符注释

● S, S1, S2 为字符串。

四 标识指令中字符的精细控制

表 5.2-7 图形标识用的希腊字母

指 令	字符	指 令	字名	于	指 令	字符	指令	>	字符
\alpha	α	\eta	η	\1	Vu	ν	\upsilon	ì	υ
							\Upsilon	n	Y
\beta	β	\theta	θ	\ x	i	ξ	\phi		φ
		\Theta	Θ	\3	Ki	Е	\Phi		Φ
∖gamma	γ	\iota	l	/p	i	π	\chi		χ
\Gamma	Γ			\ F	Pi	П			
\delta	δ	\kappa	К	\r	ho	ρ	\psi		Ψ
\Delta	Δ						\Psi		Ψ
\epsilon	\mathcal{E}	\lambda	λ	\s	igma	σ	\omega		ω
		\Lambd	Λ	\S	Sigma	Σ	\Omega	1	Ω
		a							
\zeta	5	\mu	μ	\t:	au	τ			
	使用示例								
指令	效果	指	令	效果	(果 指 令 対		汝果		
'sin\beta'	$\sin \beta$	'\zeta\om	iega'	ζω	\	in}R^{m\tiɪ	mesn}'	$A \in$	$R^{m \times n}$

表 5.2-8 图形标识用的其他特殊字符

指令	符	指 令	符	指令	符	指 令	符	指 令	符
\approx	æ	\propto	∞	\exists	3	\cap	\bigcirc	\downarrow	\downarrow
\cong	211	\sim	~	\forall	\forall	\cup	\supset	\leftarrow	\leftarrow
\div	÷	\times	×	\in	\in	\subset		\leftrightarrow	\leftrightarrow
\equiv	=	\oplus	\oplus	\infty	8	\subseteq	\subseteq	\rightarrow	\rightarrow
\geq	\geq	\oslash	Ø	\perp	\perp	\supset	U	\uparrow	↑
\leq	S	\otimes	\otimes	\prime	,	\supseteq	\supseteq	\circ	0
\neq	≠	\int	ſ	\cdot	•	\Im	3	\bullet	•
\pm	±	\partial	∂	\ldots	•••	\Re	\Re	\copyright	©

表 5.2-9 上下标的控制指令

	指令	arg 取值	举例	
			示例指令	效 果
上标	^{arg}	任何合法字符	'\ite^{-t}sint'	$e^{-t}\sin t$
下标	_{arg}	任何合法字符	$'x\sim{\langle chi }_{\langle alpha }^{2}(3)'$	$x \sim \chi_{\alpha}^2(3)$

表 5.2-10 字体式样设置规则

		• • •	11 - (11)(11)		
字体	指 令	arg	举例		
		取 值	示例指令	效果	
名称	\fontname{arg}	arial; courier;	'\fontname{courier}Example 1'	Example 1	
		roman ; 宋体;	'\fontname{隶书}范例 2'	范例2	
		隶书;黑体			

风格	\arg	bf	(黑体)	'\bfExample 3'	Example 3
		it	(斜体一)	'\itExample 4'	Example 4
		sl	(斜体二)		
		rm	(正体)		
大小	\fontsize{arg}	正整	数。	'\fontsize{14}Example 5'	Example 5
		默讠	人值为 10	'\fontsize{6}Example 6'	Example 6
		(Po	oints磅)。		

```
【例 5.2-4】
clf;t=0:pi/50:2*pi;
y=sin(t);
plot(t,y)
axis([0,2*pi,-1.2,1.2])
text(pi/2,1,'\fontsize{16}\leftarrow\itsin(t)\fontname{隶书}极大值')
%<5>
title('y=sin(t)')
xlabel('t')
ylabel('y')
```

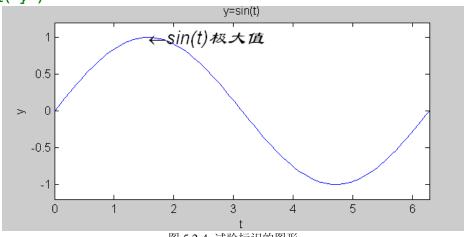


图 5.2-4 试验标识的图形

试着改变第<5>条指令,观察标识的变化。

5.2.3 多次叠绘、双纵坐标和多子图

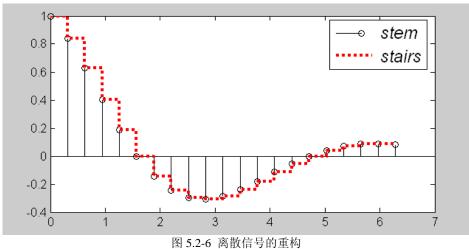
一 多次叠绘

hold on 使当前轴及图形保持而不被刷新,准备接受此后将绘制的新曲线。

hold off 使当前轴及图形不再具备不被刷新的性质。 hold 当前图形是否具备刷新性质的双向切换开关。

【例 5.2-6】利用 hold 绘制离散信号通过零阶保持器后产生的波形。 t=2*pi*(0:20)/20; y=cos(t).*exp(-0.4*t); stem(t,y,'g','Color','k'); hold on stairs(t,y,':r','LineWidth',3) hold off

$\label{legend('\fontsize{14}\it stem','\fontsize{14}\it stairs')} box \ on$



二 双纵坐标图

plotyy(X1,Y1,X2,Y2)

s=cumtrapz(y)*dx;

以左、右不同纵轴绘制 X1-Y1、X2-Y2 两条曲线。

plotyy(X1,Y1,X2,Y2,'FUN')

以左、右不同纵轴把 X1-Y1、X2-Y2 绘制成 FUN 指定形式的两条曲线。

plotyy(X1,Y1,X2,Y2,'FUN1','FUN2')

以左、右不同纵轴把 X1-Y1、X2-Y2 绘制成 FUN1、FUN2 指定的不同形式的两条曲线。

%<2>

%<3>

%<4>

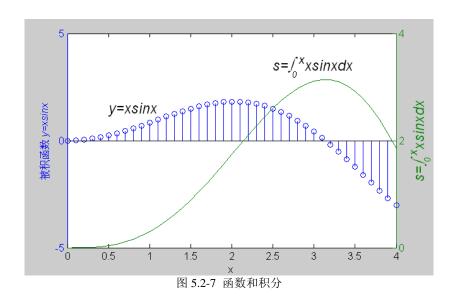
%<5>

%<6>

%<7>

```
【例 5.2-7】画出函数 y = x \sin x 和积分 s = \int_0^x (x \sin x) dx 在区间[0,4]上的曲线(图 5.2-7)。 clf;dx=0.1;x=0:dx:4;y=x.*sin(x);
```

```
a=plotyy(x,y,x,s,'stem','plot');
text(0.5,1.5,'\fontsize{14}\ity=xsinx')
sint='{\fontsize{16}\int_{\fontsize{8}0}^{ x}}';
ss=['\fontsize{14}\its=',sint,'\fontsize{14}\itxsinxdx'];
text(2.5,3.5,ss)
set(get(a(1),'Ylabel'),'String','被积函数 \ity=xsinx')%<8> 套用
set(get(a(2),'Ylabel'),'String',ss) %<9>
xlabel('x')
```

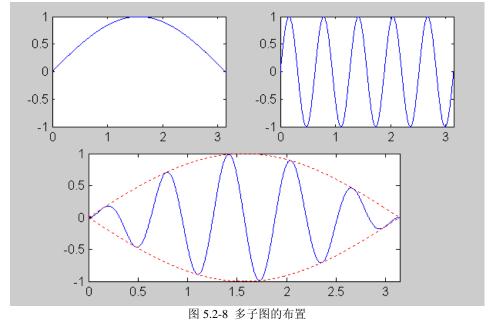


三 多子图

subplot(m,n,k)
subplot('position',[left bottom width height])

使 $(m \times n)$ 幅子图中的第k幅成为当前图在指定位置上开辟子图,并成为当前图。

【例 5.2-8】演示 subplot 指令对图形窗的分割(图 5.2-8)。clf;t=(pi*(0:1000)/1000)';y1=sin(t);y2=sin(10*t);y12=sin(t).*sin(10*t);subplot(2,2,1),plot(t,y1);axis([0,pi,-1,1])subplot(2,2,2),plot(t,y2);axis([0,pi,-1,1])subplot('position',[0.2,0.1,0.6,0.40])plot(t,y12,'b-',t,[y1,-y1],'r:')axis([0,pi,-1,1])



5.2.4 获取二维图形数据的指令ginput

[x,y]=ginput(n) 用鼠标从二维图形上获取 n 个点的数据坐标 (x,y)。

```
【例 5.2-9】采用图解法求(x+2)^x = 2的解。
```

```
(1)
clf
x=-1:0.01:5;
y=(x+2).^x-2;
plot(x,y)
grid on
```

(2)

[x,y]=ginput(1); %缩小自变量范围和放大局部图形有利于获取点的坐标。(参见 5.5 节)

(3)
format long
x,y
x =
 0.69828692903537
y =
 -5.884401711382421e-010

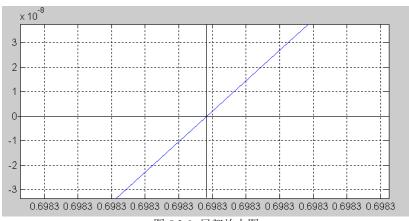


图 5.2-9 局部放大图

5.3 三维曲线和曲面

5.3.1 三维线图指令plot3

plot3(X,Y,Z,'s') 用 s 指定的点形线型色彩绘制曲线 **plot3(X1,Y1,Z1,'s1',X2,Y2,Z2,'s2', ...**) 用 s1, s2 指定的点形线型色彩绘制多类曲线

【例 5.3-1】三维曲线绘图。本例演示: 三维曲线的参数方程; 线型、点形和图例。(图 5.3-1) t=(0:0.02:2)*pi; x=sin(t);y=cos(t);z=cos(2*t); plot3(x,y,z,'bb',x,y,z,'bd') view([-82,58]),box on xlabel('x'),ylabel('y'),zlabel('z') legend('链','宝石')

图 5.3-1 宝石项链

5.3.2 三维曲面/网线图

一 三维曲面/网线可视化的数据准备

画函数 z = f(x, y) 所代表的三维空间曲面,需要做以下数据准备:

二 绘制曲面/网线图的基本指令

surf(Z) 以 Z 矩阵列、行下标为 x, y 轴自变量,画曲面图。

surf(X,Y,Z) 最常用的曲面图调用格式

surf(X,Y,Z,C) 最完整调用格式,画由 C 指定用色的曲面图。 mesh(Z) 以 Z 矩阵列、行下标为 x,y 轴自变量,画网线图。

mesh(X,Y,Z) 最常用的网线图调用格式

mesh(X,Y,Z,C) 最完整调用格式,画由C指定用色的网线图。

【例 5.3-2】用曲面图表现函数 $z = x^2 + y^2$ 。(图 5.3-2)。

```
clf
x=-4:4;y=x;
[X,Y]=meshgrid(x,y);
Z=X.^2+Y.^2;
surf(X,Y,Z);
colormap(hot)
hold on
stem3(X,Y,Z,'bo')
hold off
xlabel('x'),ylabel('y'),zlabel('z')
axis([-5,5,-5,5,0,inf])
view([-82,21])
```

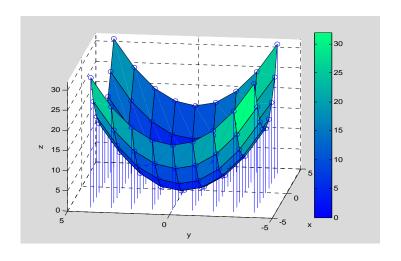


图 5.3-2 曲面图和格点

5.3.3 曲面/网线图的精细修饰

一 视角控制view

view([az,el])
view([vx,vy,vz])

通过方位角、俯视角设置视点 通过直角坐标设置视点

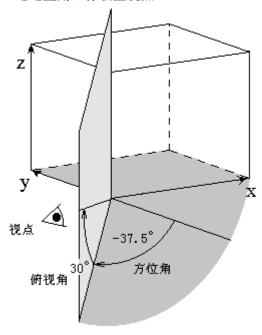


图 5.3-3 视点设置参数示意

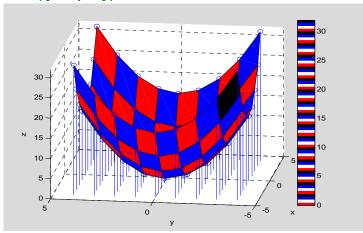
二 色图colormap

colormap(CM) 设置当前图形窗的着色色图为 CM

表 5.3-1 MATLAB 的预定义色图矩阵 CM

X 5.5 T MITTELD HIXXEX ENDING TOWN			
CM	含 义	CM	含义
autumn	红、黄浓淡色	jet	蓝头红尾饱和值色
bone	蓝色调浓淡色	lines	采用 plot 绘线色
colorcube	三浓淡多彩交错色	pink	淡粉红色图
cool	青、品红浓淡色	prism	光谱交错色
copper	纯铜色调线性浓淡色	spring	青、黄浓淡色
flag	红-白-蓝-黑交错色	summer	绿、黄浓淡色
gray	灰色调线性浓淡色	winter	蓝、绿浓淡色
hot	黑、红、黄、白浓淡色	white	全白色
hsv	两端为红的饱和值色		
注意: jet 时默认色图。			

```
clf
x=-4:4;y=x;
[X,Y]=meshgrid(x,y);
Z=X.^2+Y.^2;
surf(X,Y,Z);
colormap(flag)
colorbar
hold on
stem3(X,Y,Z,'bo')
hold off
xlabel('x'),ylabel('y'),zlabel('z')
axis([-5,5,-5,5,0,inf])
view([-82,21])
```



三 浓淡处理shading

shading options 图形对象着色的浓淡处理

```
【例 5.3-3】三种浓淡处理方式比较。(图 5.3-4)
clf
x=-4:4;y=x;
[X,Y]=meshgrid(x,y);
Z=X.^2+Y.^2;
surf(X,Y,Z)
colormap(jet)
subplot(1,3,1),surf(Z),axis off
subplot(1,3,2),surf(Z),axis off,shading flat
subplot(1,3,3),surf(Z),axis off,shading interp
set(gcf,'Color','w')

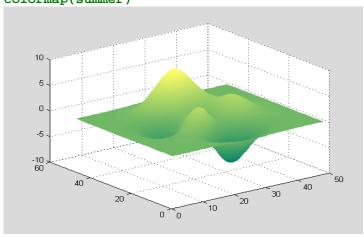
faceted flat interp
```

图 5.3-4 三种浓淡处理方式比较

四 透明度控制 alpha

alpha(v) 对面、块、象三种图形对象的透明度加以控制

【例 5.3-4】半透明的表面图。(见图 5.3-5) clf surf(peaks) shading interp alpha(0.5) colormap(summer)



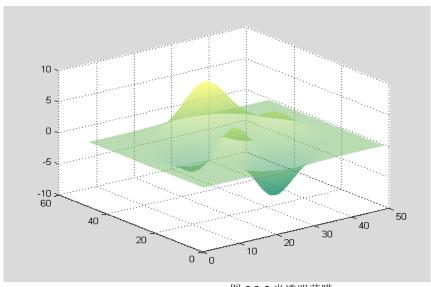


图 5.3-5 半透明薄膜

五 灯光设置light

见帮助系统

light('color',option1,'style',option2,'position',option3)

灯光设置

六 照明模式lighting

lighting options 设置照明模式

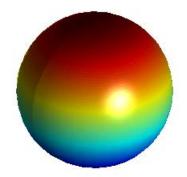
lighting flat lighting gouraud lighting phong lighting none

七 控制光反射的材质指令material

material options

使用预定义反射模式

```
【例 5.3-5】灯光、照明、材质指令所表现的图形。(图 5.3-6)
clf;
[X,Y,Z]=sphere(40);
colormap(jet)
                                                               %<3>
subplot(1,2,1),surf(X,Y,Z),axis equal off,shading interp
                                                               %<4>
light('position',[0 -10 1.5],'style','infinite')
                                                               %<5>
lighting phong
                                                               %<6>
material shiny
                                                               %<7>
subplot(1,2,2),surf(X,Y,Z,-Z),axis equal off,shading flat
                                                               %<8>
light; lighting flat
                                                               %<9>
light('position',[-1,-1,-2],'color','y')
                                                               %<10>
light('position',[-1,0.5,1],'style','local','color','w')
                                                               %<11>
set(gcf,'Color','w')
```



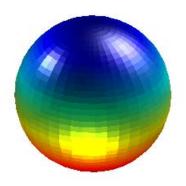


图 5.3-6 灯光、照明、材质指令所表现的图形

5.3.4 透视、镂空和裁切

一 图形的透视

hidden off 透视被叠压的图形 hidden on 消隐被叠压的图形

【例 5.3-6】透视演示(图 5.3-7)。
[X0,Y0,Z0]=sphere(30);
X=2*X0;Y=2*Y0;Z=2*Z0;
surf(X0,Y0,Z0);
shading interp
hold on,mesh(X,Y,Z),colormap(hot)
hold off
hidden off
axis equal,axis off

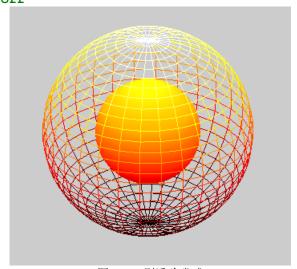


图 5.3-7 剔透玲珑球

二 图形的镂空

【例 5.3-7】演示:如何利用"非数"NaN,对图形进行镂空处理。(图 5.3-8)

```
P=peaks(30);
P(18:20,9:15)=NaN;
surfc(P);
colormap(hot)
light('position',[50,-10,5])
material([0.9,0.9,0.6,15,0.4])
grid off,box on
```

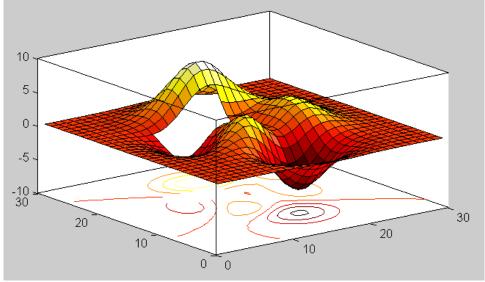


图 5.3-8 镂方孔的曲面

三 裁切

【例 5.3-8】表现切面(图 5.3-9)。

```
clf,x=[-8:0.1:8];y=x;[X,Y]=meshgrid(x,y);ZZ=X.^2-Y.^2;
ii=find(abs(X)>6|abs(Y)>6);
ZZ(ii)=zeros(size(ii));
surf(X,Y,ZZ),shading interp;colormap(copper)
light('position',[0,-15,1]);lighting phong
material([0.8,0.8,0.5,10,0.5])
```

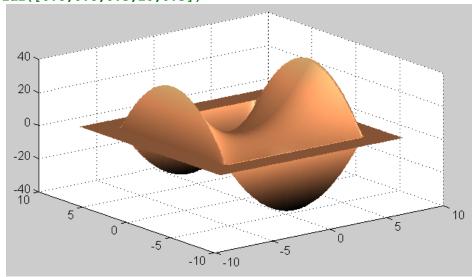


图 5.3-9 经裁切处理后的图形

5.4 高维可视化

5.4.1 二维半图指令pcolor, contour, contourf

```
【例 5.4-1】"二维半"指令 pcolor, contour, contourf 的应用。(图 5.4-1)
clf;clear;[X,Y,Z]=peaks(40);
n=6;
subplot(1,2,1),
pcolor(X,Y,Z)
shading interp
zmax=max(max(Z));zmin=min(min(Z));
caxis([zmin,zmax])
colorbar
hold on
C=contour(X,Y,Z,n,'k:');
clabel(C)
hold off
subplot(1,2,2)
[C,h]=contourf(X,Y,Z,n,'k:');
clabel(C,h)
colormap(cool)
set(gcf,'Color','w')
```

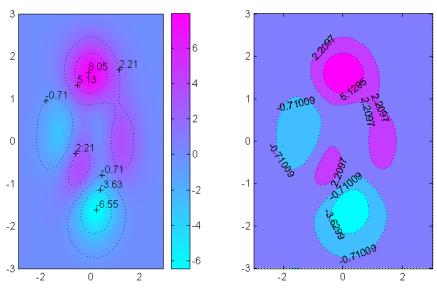


图 5.4-1 "二维半"指令的演示

5.4.2 四维表现

人对自然界的理解和思维是多维的。人的感官不仅善于接受一维、二维、三维的几何信息,而且对几何物体的运动,对颜色、声音、气味、触感等反应灵敏。从这意义上讲,MATLAB色彩控制、动画等指令为四维或更高维表现提供了手段。

一 准四维表现

【例 5.4-2】用颜色表现 z = f(x, y) 函数的其他特征(如梯度、曲率等)。(图 5.4-2)

```
clf
x=3*pi*(-1:1/15:1);y=x;[X,Y]=meshgrid(x,y);
R=sqrt(X.^2+Y.^2)+eps;Z=sin(R)./R;
```

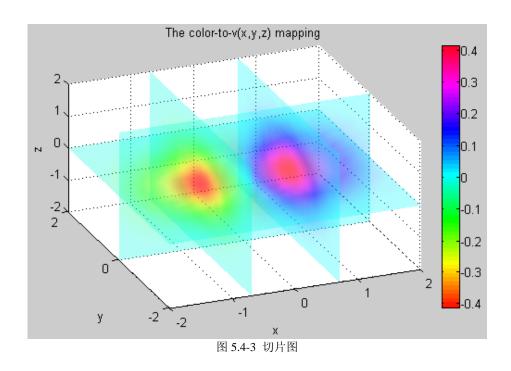
```
[dzdx,dzdy]=gradient(Z);
dzdr=sqrt(dzdx.^2+dzdy.^2);
                                         %<4>
dz2=del2(Z);
                                         %<5>
subplot(1,2,1),surf(X,Y,Z,abs(dzdr))
shading faceted;
colorbar('SouthOutside')
brighten(0.6);
colormap hsv
title('No. 1
                   surf(X,Y,Z,abs(dzdr))')
subplot(1,2,2);surf(X,Y,Z,abs(dz2))
shading faceted
colorbar('NorthOutside')
title('No. 2
                   surf(X,Y,Z,abs(dz2))')
                   surf(X,Y,Z,abs(dzdr))
                                                        0.04
                                                 0.02
                                                                0.06
                                             No. 2
                                                   surf(X,Y,Z,abs(dz2))
         0.5
          0
                                        0.5
         -0.5
                                          0
          10
                                   10
                                        -0.5
               0
                             0
                                         10
                                                                    10
                   -10 -10
                                               0
                                                             0
                                                   -10 -10
               0.05 0.1 0.15 0.2 0.25
```

二 切片图

[X,Y,Z]=meshgrid(x,y,z)由采样向量产生三维自变量"格点"数组slice(X,Y,Z,V,sx,sy,sz)三元函数切片图

图 5.4-2 色彩分别表现函数的径向导数和曲率特征

```
【例 5.4-3】图形表现 v = xe^{-x^2-y^2-z^2}。 (图 5.4-3) clf [x,y,z] = meshgrid(-2:.2:2,-2:.25:2,-2:.16:2); v =x.*exp(-x.^2-y.^2-z.^2); xs = [-0.7,0.7]; ys = 0; zs =0; slice(x,y,z,v,xs,ys,zs) colorbar shading interp colormap hsv xlabel('x'),ylabel('y'),zlabel('z') title('The color-to-v(x,y,z) mapping') view([-22,39]) alpha(0.3)
```



5.4.3 动态图形

在 MATLAB 的"上层"图形指令中的彗星轨线指令、色图变幻指令、影片动画指令,能很方便地使图形及色彩产生动态变化效果。

由于在 Notebook 和硬拷贝下,这种动态变化效果都无法表现,因此本节所有例题都不提供图形,而只给出有关指令。当读者在 MATLAB 指令窗中运作这些指令后,便可在图形窗中看到相应的动态图形。

一 彗星状轨迹图

comet(x,y,p)二维彗星轨线comet3(x,y,z,p)三维彗星轨线

```
【例 5.4-4】简单二维彗星示例。(请读者自己在指令窗中运行以下指令。) shg;n=2;t=n*pi*(0:0.000005:1);x=sin(t);y=cos(t); plot(x,y,'g');axis square hold on comet(x,y,0.0001) hold off
```

二 色图的变幻

spinmap

```
spinmap(t)
spinmap(inf)
spinmap(t,inc)

【例 5.4-5】指令 spinmap 的应用。(图 5.4-4)
ezsurf('x*y','circ'); shading flat; view([-18,28])
C=summer;
CC=[C;flipud(C)];
colormap(CC)
```

spinmap(30,4)

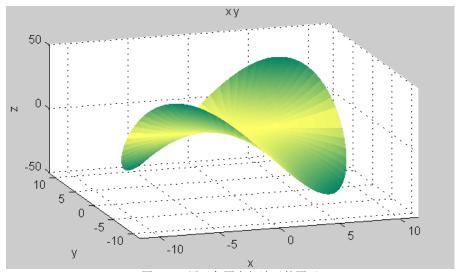


图 5.4-4 用于色图变幻演示的图形

三 影片动画

M(i)=getframe movie(M,k)

【例 5.4-6】三维图形的影片动画。(因印刷关系,无法表现本例。请读者自己在指令窗中运行以下指令。)

```
clf;shg,x=3*pi*(-1:0.05:1);y=x;[X,Y]=meshgrid(x,y);
R=sqrt(X.^2+Y.^2)+eps; Z=sin(R)./R;
h=surf(X,Y,Z);colormap(jet);axis off
n=12;mmm=moviein(n);
for i=1:n
    rotate(h,[0 0 1],25);
    mmm(:,i)=getframe;
end
movie(mmm,5,10)
```

四 实时动画

【例 5.4-7】制作红色小球沿一条带封闭路径的下旋螺线运动的实时动画(图 5.4-5)。 (1) $anim_zzy1.m$

function f=anim_zzy1(K,ki)

```
% anim_zzy1.m 演示红色小球沿一条封闭螺线运动的实时动画% 仅演示实时动画的调用格式为 anim_zzy1(K)% 既演示实时动画又拍摄照片的调用格式为 f=anim_zzy1(K,ki)% K 红球运动的循环数(不小于1)% ki 指定拍摄照片的瞬间,取1到1034间的任意整数。% f 存储拍摄的照片数据,可用image(f.cdata)观察照片。
```

%

```
t1=(0:1000)/1000*10*pi;x1=cos(t1);y1=sin(t1);z1=-t1;
t2=(0:10)/10;x2=x1(end)*(1-t2);y2=y1(end)*(1-t2);z2=z1(end)*ones(size(x2));
t3=t2;z3=(1-t3)*z1(end);x3=zeros(size(z3));y3=x3;
t4=t2;x4=t4;y4=zeros(size(x4));z4=y4;
x=[x1 \ x2 \ x3 \ x4];y=[y1 \ y2 \ y3 \ y4];z=[z1 \ z2 \ z3 \ z4];
plot3(x,y,z,'Color',[1,0.6,0.4],'LineWidth',2.5)
axis off
%
h=line('xdata',x(1),'ydata',y(1),'zdata',z(1),'Color',[1
                                                            0
                                                                     0],'Marker',
                                                                                          ١.,
'MarkerSize',40,'EraseMode','xor');
n=length(x);i=2;j=1;
while 1
    set(h,'xdata',x(i),'ydata',y(i),'zdata',z(i));
                                                % <22>
 drawnow:
 pause(0.0005)
                                                    % <23>
 i=i+1;
 if nargin==2 & nargout==1
   if(i==ki&j==1);f=getframe(gcf);end
                                                    % <26>
 end
 if i>n
   i=1;j=j+1;
   if j>K;break;end
 end
end
 (2)
f=anim_zzy1(2,450);
 (3)
image(f.cdata),axis off
```

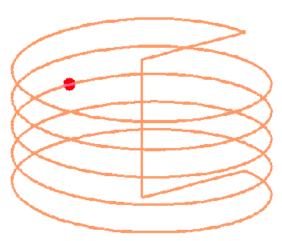


图 5.4-5 红球沿下旋螺线运动的瞬间照片

5.5 图形窗功能简介

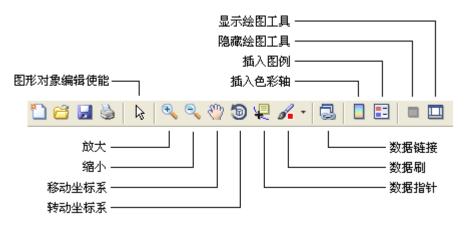
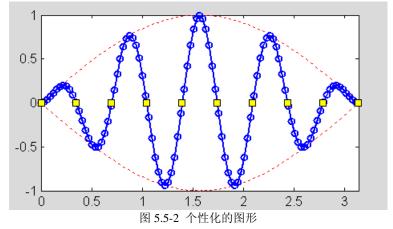


图 5.5-1 图形窗工具条专用按键

【例 5.5-1】利用图形窗的编辑功能,绘制图 5.5-2 所示的连续调制波形 $y = \sin(t)\sin(9t)$ 及其包络线。



(1) 运行以下指令绘出简图。

t=(0:pi/100:pi)'; y1=sin(t)*[1,-1]; y2=sin(t).*sin(9*t); t3=pi*(0:9)/9;
y3=zeros(size(t3));
plot(t,y1,t,y2,t3,y3)

(2) 使图形窗工作在编辑状态。

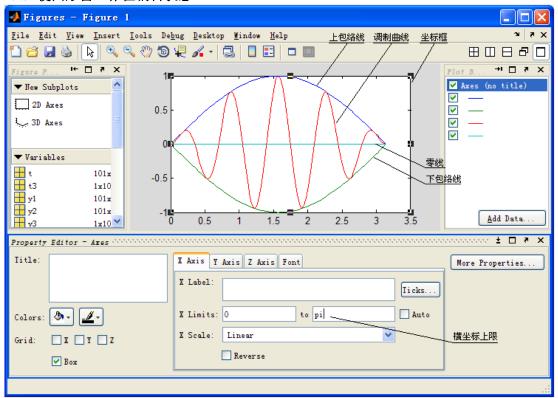
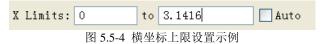


图 5.5-3 编辑工作模式下的图形窗界面

(3) 点击图形的坐标框,改变坐标范围。



(4) 点击图形的包络线,改变设置。



图 5.5-5 包络线的设置示例

(5) 点击调制曲线,改变设置。

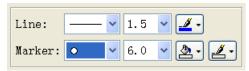


图 5.5-6 调制曲线的设置示例

(6) 点击零线,改变设置。

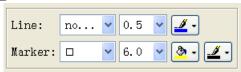


图 5.5-7 调制曲线的设置示例

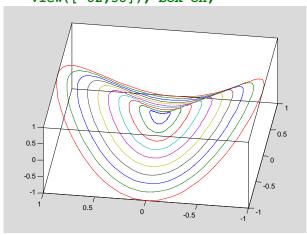
绘多条三维曲线

```
t=(0:0.02:2)'.*pi;
x=sin(t);y=cos(t);z=cos(2*t);
r=0.1:0.1:1;
x=x*r;y=y*r;z=z*r;
plot3(x,y,z);
view([-82,58]); box on;
```

8 三维坐标用参数方程表示

% t为参数,得到三个同维向量

8 画系数不同的10条曲线



6.1.1 三维曲面/网线图

一 三维曲面/网线可视化的数据准备

画函数 z = f(x, y) 所代表的三维空间曲面,需要做以下数据准备:

二 绘制曲面/网线图的基本指令

 $\operatorname{surf}(\mathbf{Z})$ 以 Z 矩阵列、行下标为 x, y 轴自变量,画曲面图。

surf(X,Y,Z) 最常用的曲面图调用格式

surf(X,Y,Z,C) 最完整调用格式,画由 C 指定用色的曲面图。 mesh(Z) 以 Z 矩阵列、行下标为 x,y 轴自变量,画网线图。

mesh(X,Y,Z) 最常用的网线图调用格式

mesh(X,Y,Z,C) 最完整调用格式,画由C指定用色的网线图。

【例 5.3-2】用曲面图表现函数 $z = x^2 + y^2$ 。(图 5.3-2)。

clf
x=-4:4;y=x;
[X,Y]=meshgrid(x,y);
Z=X.^2+Y.^2;
mesh(X,Y,Z);
colormap(hot)
colorbar
hold on
stem3(X,Y,Z,'bo')
hold off
xlabel('x'),ylabel('y'),zlabel('z')
axis([-5,5,-5,5,0,inf])

view([-82,21])

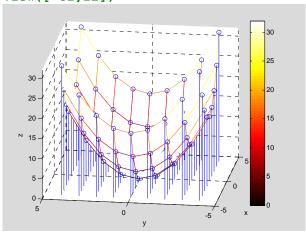


图 5.3-2 曲面图和格点

曲面/网线图的精细修饰 6.1.2

视角控制view

view([az,el])

通过方位角、俯视角设置视点

view([vx,vy,vz]) 通过直角坐标设置视点

视点与旋动

(1) 视点,即观察点,如下图所示。

Matlab 控制视点的命令为:

view([az, el]) - 通过方位角 az、俯仰角 el 来设置视点,单位是度。(二 维图缺省为 0、90 度。三维缺省为 37.5、30 度)

view([vx, vy, vz])一通过视点的直角坐标点来设定。

(2) 图形的旋动即图形的旋转, 命令为: rotate(h, direction, alpha, orgin), 其 中:

h-被旋转的对象;

direction 一方向轴, 可用球坐标 [theta, phi]或直角坐标[x, y, z]表示;

> alpha一按右手法则旋转的角度; orgin一支点。

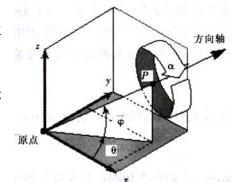


图 5.3-3 视点设置参数示意

【例】旋转指令示例

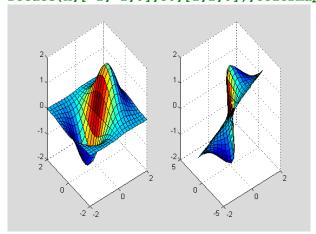
shq;clf;

[X,Y]=meshgrid([-2:.2:2]); $Z=4*X.*exp(-X.^2-Y.^2);$

G=gradient(Z);

- % 图形窗口置前,清除
- % 取格点坐标
- % 计算函数值
- % 近似梯度

subplot(1,2,1),surf(X,Y,Z,G) % 子图 1 画曲面 subplot(1,2,2),h=surf(X,Y,Z,G);% 子图 2 画曲面 rotate(h,[-2,-2,0],30,[2,2,0]),colormap(jet) % 旋转



二 色图colormap

colormap(CM) 设置当前图形窗的着色色图为 CM

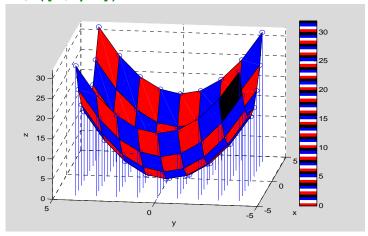
每个图形窗只有一个色图; 色图为 m*3 矩阵,可以直接赋值,也可生成;

表 5.3-1 MATLAB 的预定义色图矩阵 CM

CM	含义	CM	含义
autumn	红、黄浓淡色	jet	蓝头红尾饱和值色
bone	蓝色调浓淡色	lines	采用 plot 绘线色
colorcube	三浓淡多彩交错色	pink	淡粉红色图
cool	青、品红浓淡色	prism	光谱交错色
copper	纯铜色调线性浓淡色	spring	青、黄浓淡色
flag	红-白-蓝-黑交错色	summer	绿、黄浓淡色
gray	灰色调线性浓淡色	winter	蓝、绿浓淡色
hot	黑、红、黄、白浓淡色	white	全白色
hsv	两端为红的饱和值色		
注意: jet 时默认色图。64*3 矩阵			

clf
x=-4:4;y=x;
[X,Y]=meshgrid(x,y);
Z=X.^2+Y.^2;
surf(X,Y,Z);
colormap(flag)
colorbar
hold on
stem3(X,Y,Z,'bo')

hold off xlabel('x'),ylabel('y'),zlabel('z') axis([-5,5,-5,5,0,inf]) view([-82,21])



三 浓淡处理shading

shading options 图形对象着色的浓淡处理

shading flat 一整条线条或贴片用一种颜色 shading interp 一用线性插值成色 shading faceted 一勾画出网格线,在 flat 基础上再在贴片的四周勾画黑色网线;

【例 5.3-3】三种浓淡处理方式比较。(图 5.3-4) clf

x=-4:4;y=x;

[X,Y]=meshgrid(x,y);

Z=X.^2+Y.^2;

surf(X,Y,Z)

colormap(jet)

subplot(1,3,1),surf(Z),axis off

subplot(1,3,2),surf(Z),axis off,shading flat

subplot(1,3,3),surf(Z),axis off,shading interp

set(gcf,'Color','w')

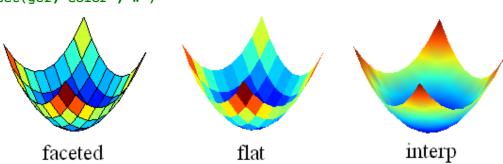


图 5.3-4 三种浓淡处理方式比较

四 透明度控制 alpha

alpha(v) 对面、块、象三种图形对象的透明度加以控制 0~1

【例 5.3-4】半透明的表面图。(见图 5.3-5) clf surf(peaks) shading interp alpha(0.5)

colormap(summer)

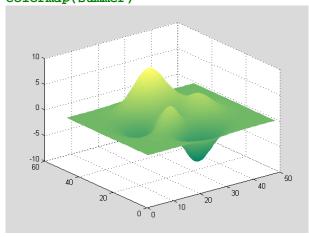


图 5.3-5 半透明薄膜

五 灯光设置light

见帮助系统

light('color',option1,'style',option2,'position',option3) 灯光设置

六 照明模式lighting

lighting options 设置照明模式

七 控制光反射的材质指令material

material options 使用预定义反射模式

Matlab 不指定光照,图形采用强度各处相等的漫射光。 (1)设置光源命令: light('color',cl,'style',sl,'position',pl)

其中三个参数:

- c1-代表光的颜色,用[r,g,b]表示,缺省[1 1 1]
- s1-无穷远光='infinite', 近光='local'
- p1-表示[x, y, z], 对于远光,表示穿过该点射向原点对于近光,指光源的位置

(2) 设置照明模式:

- lighting flat 一均匀洒落在图形对象上(缺省)
- lighting gouraud 一采用插补,用于曲面表现
- lighting phong 一对顶点处法线插值,再计算反射光,效果最好,费时

多

lighting none 一关闭光源

(3) 控制光效果的材质指令:

- material shiny 一使对象比较明亮,镜反射大
- material dull 一使对象比较暗淡,漫反射大
- material metal 一使对象带金属光泽(缺省模式)
- material default 一返回缺省模式
- material([ka kd ks n sc]) 一对反射五要素设置
 - ka一均匀背景光的强度
 - kd一漫反射的强度
 - ks一反射光的强度
 - n 一控制镜面亮点大小
 - sc一控制镜面颜色的反射系数

【例 5.3-5】灯光、照明、材质指令所表现的图形。(图 5.3-6) clf;

```
[X,Y,Z]=sphere(40);
                                                                %<3>
colormap(jet)
subplot(1,2,1),surf(X,Y,Z),axis equal off,shading interp
                                                                %<4>
light('position',[10 10 10],'style','infinite')
                                                                %<5>
lighting phong
                                                                %<6>
material shiny
                                                                %<7>
subplot(1,2,2),surf(X,Y,Z,-Z),axis equal off,shading flat
                                                                %<8>
light; lighting flat
                                                                %<9>
light('position',[-1,-1,-2],'color','y')
                                                                %<10>
light('position',[-1,0.5,1],'style','local','color','w')
                                                                %<11>
set(gcf,'Color','w')
```





图 5.3-6 灯光、照明、材质指令所表现的图形

6.1.3 透视、镂空和裁切

一 图形的透视

透视被叠压的图形 hidden off hidden on 消隐被叠压的图形

【例 5.3-6】透视演示(图 5.3-7)。

[X0,Y0,Z0]=sphere(30); % 产生单位球面的三维坐标

 X=2*X0;Y=2*Y0;Z=2*Z0;
 % 产生半径为 2 的球面坐标

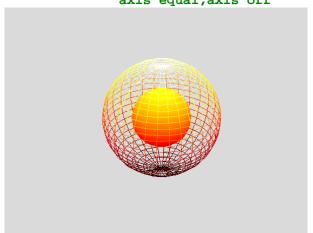
 surf(X0,Y0,Z0);
 % 画单位球面

shading interp % 对球的着色进行浓淡细化处理

hold on; % 绘图保持 mesh(X,Y,Z) % 绘大球 % 定义色表 colormap(hot); hold off % 取消绘图保持 % 产生透视效果 hidden off

axis equal, axis off

% 坐标等轴并隐藏



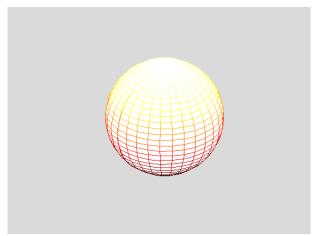


图 5.3-7 剔透玲珑球

二 图形的镂空

【例 5.3-7】演示: 如何利用"非数"NaN, 对图形进行镂空处理。(图 5.3-8) P=peaks(30); P(18:20,9:15)=NaN; surfc(P); colormap(hot) light('position',[50,-10,5]) material([0.9,0.9,0.6,15,0.4]) grid off,box on

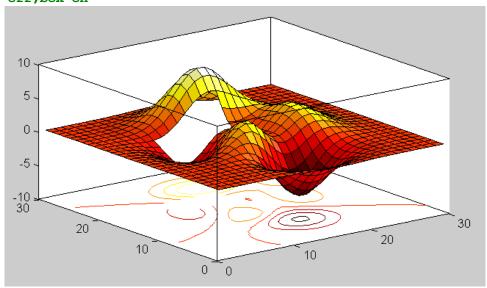


图 5.3-8 镂方孔的曲面

三 裁切

```
【例 5.3-8】表现切面(图 5.3-9)。
clf,x=[-8:0.1:8];y=x;[X,Y]=meshgrid(x,y);ZZ=X.^2-Y.^2;
ii=find(abs(X)>6|abs(Y)>6);
ZZ(ii)=zeros(size(ii));
surf(X,Y,ZZ),shading interp;colormap(copper)
light('position',[0,-15,1]);lighting phong
material([0.8,0.8,0.5,10,0.5])
```

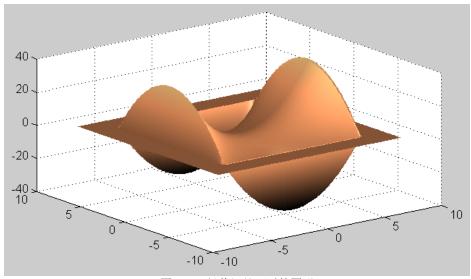


图 5.3-9 经裁切处理后的图形

6.2 高维可视化

6.2.1 二维半图指令pcolor, contour, contourf

```
【例 5.4-1】 "二维半"指令 pcolor, contour, contourf 的应用。(图 5.4-1)
clf;clear;[X,Y,Z]=peaks(40);
n=6;
subplot(1,2,1),
pcolor(X,Y,Z)
                %A pseudocolor plot is a rectangular array of cells with
colors determined by Z
shading interp
zmax=max(max(Z));zmin=min(min(Z));
                           %Color axis scaling 决定色标尺的范围
caxis([zmin,zmax])
colorbar
              %垂直色标尺
hold on
C=contour(X,Y,Z,n,'k:');
clabel(C)
hold off
subplot(1,2,2)
[C,h]=contourf(X,Y,Z,n,'k:');
clabel(C,h)
colormap(cool)
set(gcf,'Color','w')
```

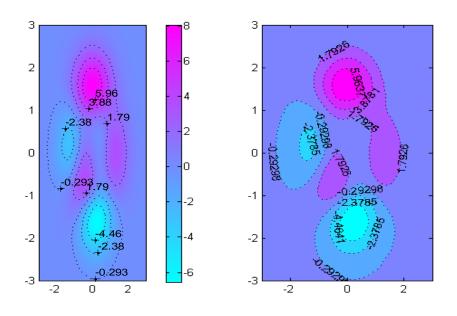


图 5.4-1 "二维半"指令的演示

6.2.2 四维表现

人对自然界的理解和思维是多维的。人的感官不仅善于接受一维、二维、三维的几何信息,而且对几何物体的运动,对颜色、声音、气味、触感等反应灵敏。从这意义上讲,MATLAB色彩控制、动画等指令为四维或更高维表现提供了手段。

一 准四维表现

【例 5.4-2】用颜色表现 z = f(x, y) 函数的其他特征(如梯度、曲率等)。(图 5.4-2)

```
clf
x=3*pi*(-1:1/15:1);y=x;[X,Y]=meshgrid(x,y);
R=sqrt(X.^2+Y.^2)+eps;Z=sin(R)./R;
[dzdx,dzdy]=gradient(Z);
dzdr=sqrt(dzdx.^2+dzdy.^2);
                                     %<4>
dz2=del2(Z);
                                     %<5>
subplot(1,2,1),surf(X,Y,Z,abs(dzdr))
shading faceted;
colorbar('SouthOutside')
brighten(0.6);
colormap hsv
title('No. 1
                 surf(X,Y,Z,abs(dzdr))')
subplot(1,2,2);surf(X,Y,Z,abs(dz2))
shading faceted
colorbar('NorthOutside')
title('No. 2
                 surf(X,Y,Z,abs(dz2))')
```

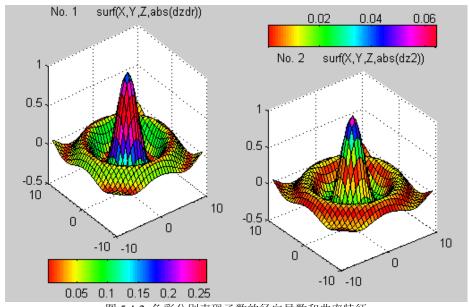
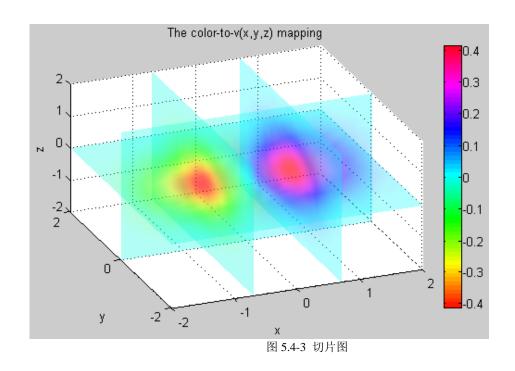


图 5.4-2 色彩分别表现函数的径向导数和曲率特征

二切片图

[X,Y,Z]=meshgrid(x,y,z)由采样向量产生三维自变量"格点"数组slice(X,Y,Z,V,sx,sy,sz)三元函数切片图

```
【例 5.4-3】图形表现 v = xe^{-x^2-y^2-z^2}。 (图 5.4-3) clf [x,y,z] = meshgrid(-2:.2:2,-2:.25:2,-2:.16:2); v =x.*exp(-x.^2-y.^2-z.^2); xs = [-0.7,0.7]; ys = 0; zs =0; slice(x,y,z,v,xs,ys,zs) colorbar shading interp colormap hsv xlabel('x'),ylabel('y'),zlabel('z') title('The color-to-v(x,y,z) mapping') view([-22,39]) alpha(0.1)
```



6.2.3 动态图形

在 MATLAB 的"上层"图形指令中的彗星轨线指令、色图变幻指令、影片动画指令,能很方便地使图形及色彩产生动态变化效果。

由于在 Notebook 和硬拷贝下,这种动态变化效果都无法表现,因此本节所有例题都不提供图形,而只给出有关指令。当读者在 MATLAB 指令窗中运作这些指令后,便可在图形窗中看到相应的动态图形。

一 彗星状轨迹图

comet(x,y,p)二维彗星轨线comet3(x,y,z,p)三维彗星轨线

【例 5.4-4】简单二维彗星示例。(请读者自己在指令窗中运行以下指令。) shg;n=2;t=n*pi*(0:0.0001:1);x=sin(t);y=cos(t);z=cos(2*t)

comet3(x,y,z,0.1)

二 色图的变幻

spinmap
spinmap(t)
spinmap(inf)
spinmap(t,inc)

The spinmap function shifts the colormap RGB values by some incremental value. For example, if the increment equals 1, color 1 becomes color 2, color 2 becomes color 3, etc.

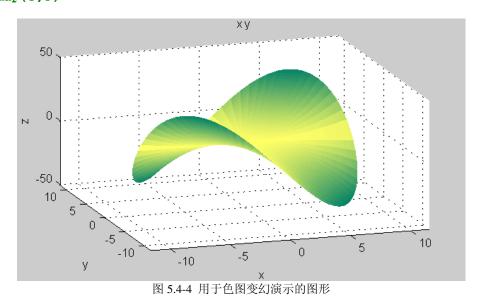
spinmap cyclically rotates the colormap for approximately five seconds using an incremental value of 2.

spinmap(t) rotates the colormap for approximately 10*t seconds. The amount of time specified by t depends on your hardware configuration (e.g., if you are running MATLAB® over a network).

spinmap(t,inc) rotates the colormap for approximately 10*t seconds and specifies an increment inc by which the colormap shifts. When inc is 1, the rotation appears smoother than the default (i.e., 2). Increments greater than 2 are less smooth than the default. A negative increment (e.g., -2) rotates the colormap in a negative direction.

spinmap('inf') rotates the colormap for an infinite amount of time. To break the loop, press Ctrl+C.

```
【例 5.4-5】指令 spinmap 的应用。(图 5.4-4)
ezsurf('x*y','circ'); shading flat; view([-18,28])
C=hot;
CC=[C;flipud(C)];
colormap(C)
spinmap(5,3)
```



三 影片动画

M(i)=getframe movie(M,k)

【例 5.4-6】三维图形的影片动画。(因印刷关系,无法表现本例。请读者自己在指令窗中运行以下指令。)

```
clf;shg,x=3*pi*(-1:0.05:1);y=x;[X,Y]=meshgrid(x,y);
R=sqrt(X.^2+Y.^2)+eps; Z=sin(R)./R;
h=surf(X,Y,Z);colormap(jet);axis off
n=12;mmm=moviein(n);
for i=1:n
    rotate(h,[0 0 1],25);
    mmm(:,i)=getframe;
end
movie(mmm,5,10)
```

四 实时动画

【例 5.4-7】制作红色小球沿一条带封闭路径的下旋螺线运动的实时动画(图 5.4-5)。
(1) $anim_zzy1.m$

function f=anim zzy1(K,ki) 演示红色小球沿一条封闭螺线运动的实时动画 % anim zzy1.m % 仅演示实时动画的调用格式为 anim zzy1(K) % 既演示实时动画又拍摄照片的调用格式为 f=anim_zzy1(K,ki) 红球运动的循环数(不小于1) % K 指定拍摄照片的瞬间,取1到1034间的任意整数。 % ki 存储拍摄的照片数据,可用image(f.cdata)观察照片。 % f t1=(0:1000)/1000*10*pi;x1=cos(t1);y1=sin(t1);z1=-t1; t2=(0:10)/10;x2=x1(end)*(1-t2);y2=y1(end)*(1-t2);z2=z1(end)*ones(size)t3=t2;z3=(1-t3)*z1(end);x3=zeros(size(z3));y3=x3; t4=t2;x4=t4;y4=zeros(size(x4));z4=y4; x=[x1 x2 x3 x4];y=[y1 y2 y3 y4];z=[z1 z2 z3 z4];plot3(x,y,z,'Color',[1,0.6,0.4],'LineWidth',2.5) axis off h=line('xdata',x(1),'ydata',y(1),'zdata',z(1),'Color',[1 0 0],'Marker', '.', 'MarkerSize',50,'EraseMode','xor'); n=length(x);i=2;j=1; while 1 set(h,'xdata',x(i),'ydata',y(i),'zdata',z(i)); % <22> pause(0.0005) % <23> i=i+1; if nargin==2 & nargout==1 if(i==ki&j==1);f=getframe(gcf);end % <26> end if i>n i=1;j=j+1; if j>K;break;end end end (2) f=anim_zzy1(2,450);

图 5.4-5 红球沿下旋螺线运动的瞬间照片

(3)

image(f.cdata),axis off

6.3 图形窗功能简介

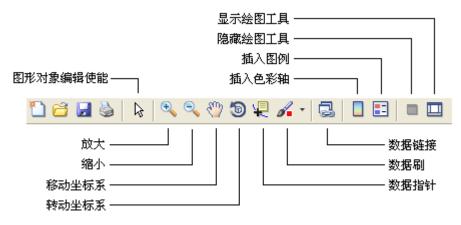
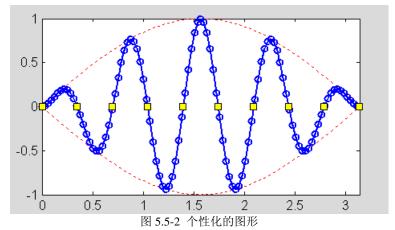


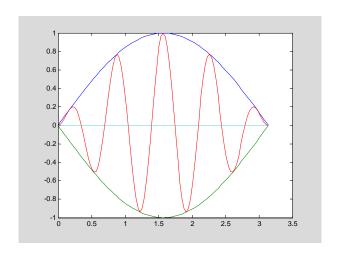
图 5.5-1 图形窗工具条专用按键

【例 5.5-1】利用图形窗的编辑功能,绘制图 5.5-2 所示的连续调制波形 $y = \sin(t)\sin(9t)$ 及其包络线。



(1) 运行以下指令绘出简图。

```
t=(0:pi/100:pi)';
y1=sin(t)*[1,-1];
y2=sin(t).*sin(9*t);
t3=pi*(0:9)/9;
y3=zeros(size(t3));
plot(t,y1,t,y2,t3,y3)
```



(2) 使图形窗工作在编辑状态。

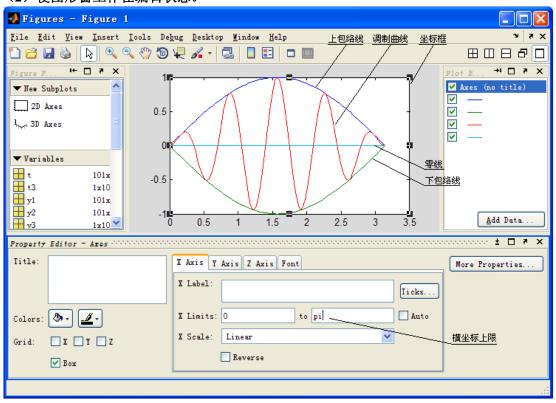
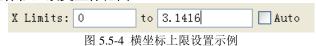


图 5.5-3 编辑工作模式下的图形窗界面

(3) 点击图形的坐标框,改变坐标范围。



(4) 点击图形的包络线,改变设置。

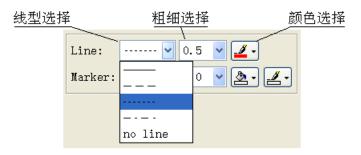


图 5.5-5 包络线的设置示例

(5) 点击调制曲线,改变设置。

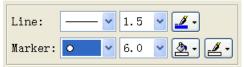


图 5.5-6 调制曲线的设置示例

(6) 点击零线,改变设置。

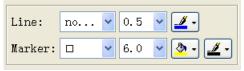


图 5.5-7 调制曲线的设置示例

第6章 M文件

- 比较简单的问题,特别是一次性简单问题,从指令窗中直接输入指令进行计算是十分轻松的事。
- 但随指令数的增加,或随控制流复杂度的增加,或重复计算要求的提出,直接从指令窗进行计算就显得烦琐。

此时就需要将程序放进 M 文件中,这种程序的扩展名为.m。

6.1 MATLAB 控制流

MATLAB 用于程序流控制的指令与其它语言相应指令的用法十分相似。

6.1.1 if-else-end条件控制

if-else-end 指令为程序流提供了一种分支控制。

表 6.1-1 if-else-end 分支结构的使用方式

单分支	双分支	多分支
if expr	if expr	if expr1
(commands)	(commands1)	(commands)
end	else	elseif expr2
	(commands2)	(commands)
	end	•••••
		else
		(commandsk)
		end
当 expr 给出"逻辑	当 expr 给出"逻辑 1"	expr1, expr2, 中, 首先给出"逻
1"时, (commands)	时,(commands1)指令组	辑 1"的那个分支的指令组被执行;
指令组才被执行。	被执行; 否则,	否则,(commandsk)被执行。
	(commands2)被执行。	该使用方法常被 switch-case 所取代。

注:

- 1. $\exp r$ 通常是关系、逻辑运算构成的表达式。也可以是一般代数表达式,此时给出的任何非零值的作用等同于"逻辑 1"。
- 2. 当 expr 给出数组时,仅当数组元素不包含任何零元素时,该 expr 控制的分支才执行。

6.1.2 switch-case控制结构

表 6.1-2 switch-case 控制结构的使用方式

指令格式	含 义
switch expr	● expr 为根据此前给定变量进行计算的表达式
case value_1	● value_1 是给定的数值、字符串标量(或胞元数组, 见附录 A.2)
(commands1)	● 若 expr 结果与 value_1(或其中的胞元元素)相等,则执行。
case value_2	
(commands2)	
case value_k	● value_k 是给定的数值、字符串标量(或胞元数组)
(commandsk)	● 若 expr 结果与 value_k(或其中的胞元元素)相等,则执行。
otherwise	● 该情况是以上的"并"的"补"
(commands)	● 若所有 case 都不发生,则执行该组命令。
end	

6.1.3 for循环和while循环

表 6.1-3 循环结构的使用方式

for 循环	while 循环
--------	----------

for ix=array	while expression	
(commands)	(commands)	
end	end	

注: for 循环的次数确定, while 循环的次数不确定。

【例 3.3-1】欧姆定律: $r = \frac{u}{i}$,其中r,u,i分别是电阻(欧姆)、电压(伏特)、电流(安培)。验证实验: 据电阻两端施加的电压,测量电阻中流过的电流,然后据测得的电压、电流计算平均电阻值。(测得的电压电流具体数据见下列程序)。

```
(1) 非向量化程序
clear
vr=[0.89, 1.20, 3.09, 4.27, 3.62, 7.71, 8.99, 7.92, 9.70, 10.41];
ir=[0.028, 0.040, 0.100, 0.145, 0.118, 0.258, 0.299, 0.257, 0.308,
L=length(vr);
for k=1:L
  r(k)=vr(k)/ir(k);
end
% -----
sr=0;
for k=1:L
 sr=sr+r(k);
end
rm=sr/L
rm =
  30.5247
(2) 向量化程序
clear
vr=[0.89, 1.20, 3.09, 4.27, 3.62, 7.71, 8.99, 7.92, 9.70, 10.41];
ir=[0.028, 0.040, 0.100, 0.145, 0.118, 0.258, 0.299, 0.257, 0.308,
0.345];
                          相同位置元素进行运算
r=vr./ir
                          % MATLAB现成的函数,查阅方法
rm=mean(r)
 Columns 1 through 7
  31.7857 30.0000 30.9000 29.4483 30.6780 29.8837 30.0669
 Columns 8 through 10
  30.8171 31.4935 30.1739
  30.5247
```

【例 6.1-4】编写计算
$$S = \sum_{n=1}^{N} \frac{1}{\sum_{k=1}^{n} k}$$
 ,其中 $N = \arg\min\left\{\frac{1}{\sum_{k=1}^{N} k} \le \mathcal{E}\right\}$, \mathcal{E} 是预先给定的控制

精度。

(1) 编写 M 函数文件 exm060104.m

```
function [S,N]=exm060104(epsilon)
% [S,N]=exm060104(epsilon)
%
           Calculate the sum of a special series S=1+1/(1+2)+...+1/(1+2+...+N)
% S
           Sum of a special series
% N
           The minimum among all numbers to have 1/sum(1:N)<epsilon
% epsilon
           Given accuracy
k=0;
s=0;
d=inf;
S=0;
while d>epsilon
 k=k+1;
 s=s+k;
 d=1/s;
 S=S+d;
end
N=k;
(2) 把文件 exm060104.m 保存在处于 MATLAB 搜索路径上的文件夹中。
(3)运行以下指令,控制精度取 0.01。
[S,N]=exm060104(0.01)
    1.8667
N =
    14
```

6.1.4 控制程序流的其它常用指令

表 6.1-4 控制程序流的其它常用指令

指令及使用格式	使 用 说 明	
v=input('message')	该指令执行时,"控制权"交给键盘;待输入结束,按下 Enter 键,	
v=input('message','s')	"控制权"交还 MATLAB。	
	message 是提示用的字符串。	
	第二种格式,不管键入什么,总以字符串形式赋给变量 v。	
keyboard	遇到 keyboard 时,将"控制权"交给键盘,用户可以从键盘输入	
	各种 MATLAB 指令。仅当用户输入 return 指令后,"控制权"才	
	交还给程序。	
break	break 指令,或导致包含该指令的 while、for 循环终止,或导致 if -	
	end, switch – case 等中断。	
continue	跳过位于它之后的循环体中其它指令,而执行循环的下一个迭代。	
pause	第一种格式使程序暂停执行,等待用户按任意键继续;	
pause(n)	第二种格式使程序暂停 n 秒后,再继续执行。	
return	结束 return 指令所在函数的执行,而把控制转至主调函数或者指令	
	窗。否则,只有待整个被调函数执行完后,才会转出。	
	结束键盘模式。	

6.2 脚本文件和函数文件

M 文件分为 M 脚本文件和 M 函数文件。

6.2.1 M脚本文件

"脚本"本身反映这样一个事实: MATLAB 只是按文件所写的指令执行。

- (1) 该文件中的指令形式和前后位置,与解决同一个问题时在指令窗中输入的指令没有任何区别。
- (2) MATLAB 在运行这个脚本时,只是简单地从文件中读取那一条条指令,送到 MATLAB 中去执行。
- (3)与在指令窗中直接运行指令一样,脚本文件运行产生的变量都是驻留在 MATLAB 基本工作空间(base workspace)中。只要用户不使用 clear 指令加以清除,或 MATLAB 指令窗不关闭,这些变量将一直保存在基本工作空间中。基本工作空间随 MATLAB 的启动而产生;只有当关闭 MATLAB 时,该基本空间才被删除。
- (4) 文件扩展名是".m"。

6.2.2 M函数文件

与脚本文件不同,函数文件(Function file)犹如一个"黑箱"。从外界只看到: 传给它的输入量和送出来的计算结果,而内部运作从外界看是藏而不见的。函数文件运行时分配临时的函数工作空间。

6.2.3 局部变量和全局变量

(1) 局部 (Local) 变量

存在于函数空间内部的中间变量,产生于该函数的运行过程中,其影响范围也仅限于该函数本身。正由于这种空间、时间上的局部性,中间变量被称为局部变量。

(2) 全局 (Global) 变量

通过 global 指令,MATLAB 也允许几个不同的函数空间以及基本工作空间共享同一个变量。这种被共享的变量称为全局变量。

全局变量损害函数的封装性, 不提倡使用。

6.2.4 M函数文件的一般结构

【例 6.2-1】

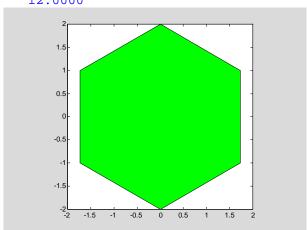
(1) 编写 M 函数文件 exm060201.m (MATLAB 以保存文件名为准,函数文件名必须以字母开头,后面可以是字母、下划线以及数字的任意组合。)

function [S,L]=exm060201(N,R,str) % 函数声明行 , 必不可少, 脚本文件无此行。 % exm060201.m The area and perimeter of a regular polygon (正多边形的面积和周长),H1行 **%** The number of sides 在线帮助文本区(包括H1行) N % R The circumradius % A line specification to determine line type/color str **%** The area of the regular polygon S % L The perimeter of the regular polygon 用蓝实线画半径为1的圆 % exm060201 用蓝实线画外接半径为1的正N边形 % exm060201(N) 用蓝实线画外接半径为R 的正N 边形 % exm060201(N,R) 用str 指定的线画外接半径为R的正N边形 % exm060201(N,R,str) 给出多边形面积 S.并画相应下多边形填色图 % S=exm060201(...) 给出多边形面积S和周长L,并画相应正多边形填色图 % [S,L]=exm060201(...)

% Zhang Zhiyong 编写于2006-1-31

编写和修改记录

```
% 函数体,必不可少。
switch nargin
 case 0
   N=100;R=1;str='-b';
                       % 正100边形 - 近似为圆
 case 1
   R=1;str='-b';
 case 2
   str='-b';
 case 3
 otherwise
   error('输入量太多。');
end;
t=0:2*pi/N:2*pi;
x=R*sin(t);y=R*cos(t);
if nargout==0
 plot(x,y,str);
elseif nargout>2
 error('输出量太多。');
 S=N*R*R*sin(2*pi/N)/2;
 L=2*N*R*sin(pi/N);
 fill(x,y,str)
end
axis equal square
box on
shg
(2) 把 exm060201.m 文件保存在 MATLAB 的搜索路径上,然后在指令窗中运行以下指令。
                               %计算外接半径为2的正六边形面积和周长,并绘图。
[S,L]=exm060201(6,2,'-g')
                                 % 允许使用比标称数目少的输入/输出量
   10.3923
L =
   12.0000
```



第7章 SIMULINK交互式 仿真集成环境

7.1 连续时间系统的建模与仿真

7.1.1 基于微分方程的SIMULINK建模

【例 7.1-1】在图 7.1-1 所示的系统中,已知质量m=1 kg,阻尼b=2 N.sec/m,弹簧系数 k=100 N/m,且质量块的初始位移 x(0)=0.05 m,其初始速度 x'(0)=0 m/sec,要求创建该系统的 SIMULINK 模型,并进行仿真运行。本例演示,据物理定理建立微分方程,并以此微分方程创建 SIMULINK 模型的完整步骤:微分方程的整理;模块的复制;信号线的构画;模块参数设置;示波器的调整;仿真参数设置。

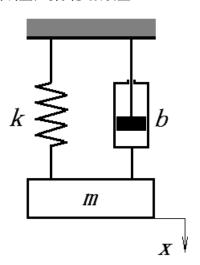


图 7.1-1 弹簧—质量—阻尼系统

- (1) 建立理论数学模型
- (2) 建模的基本思路
- (3) 打开 SIMULINK 模块库

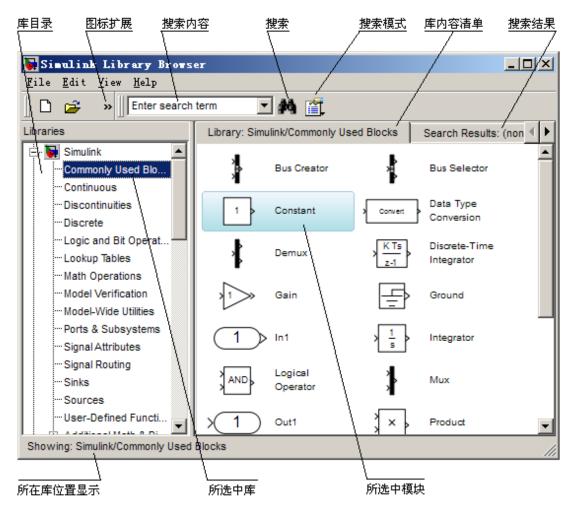
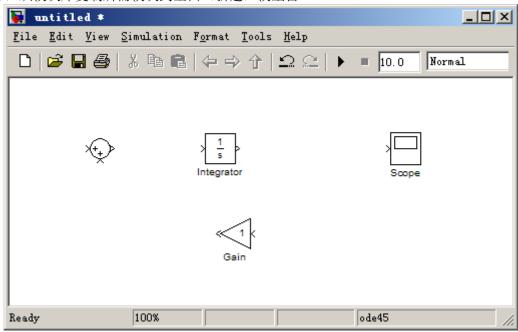


图 7.1-2 SIMULINK 模块库浏览器

- (4) 开启空白(新建)模型窗
- (5) 从模块库复制所需模块到空白(新建)模型窗



- (6) 新建模型窗中的模型再复制
- (7) 模块间信号线的连接

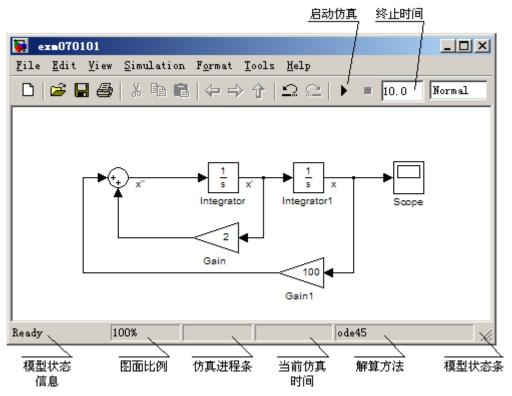


图 7.1-4 已构建完成的新模型窗

(8) 根据理论数学模型设置模块参数

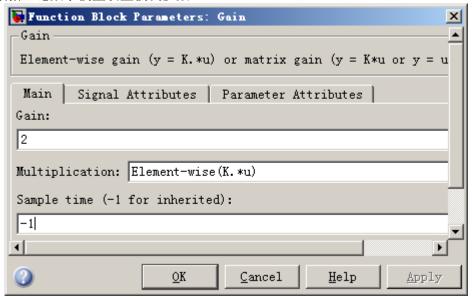


图 7.1-5 参数已经修改为 2 的<Gain>增益模块参数设置窗

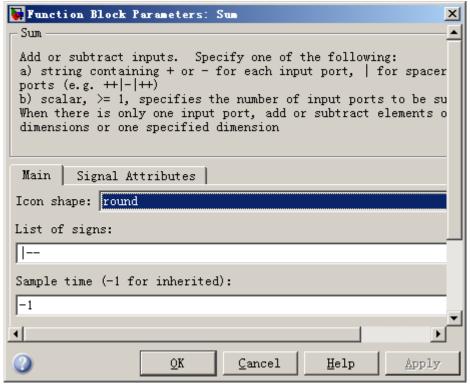


图 7.1-6 改变输入口符号的求和模块参数设置窗

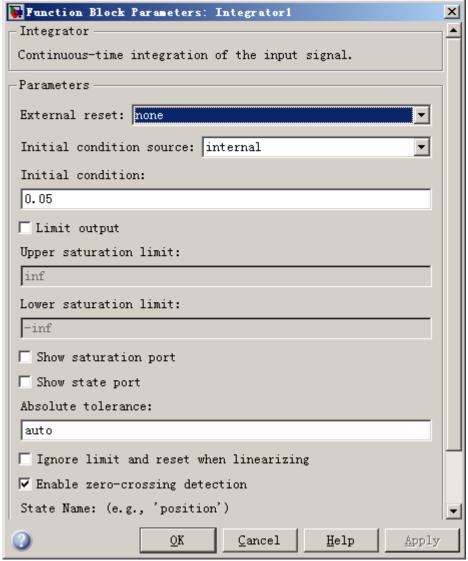


图 7.1-7 实现初始位移 0.05 设置的<Integrator1>设置窗

- (9) 仿真运行参数采用默认解算器"ode45"、默认"变步长"和默认仿真终止时间 10。
- (10) 把新建模型保存为 exm070101.mdl。
- (11) 试运行,以便发现问题加以改善。

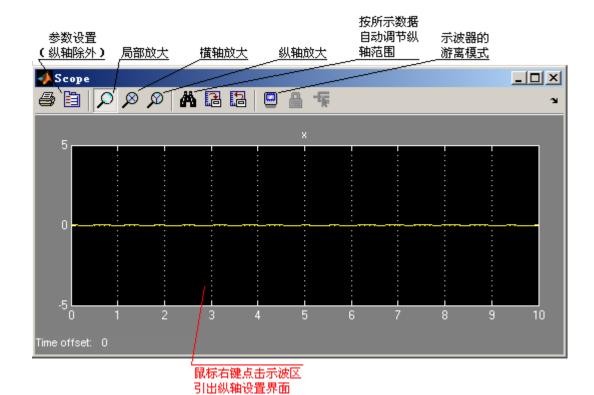


图 7.1-8 坐标范围设置不当时的信号显示

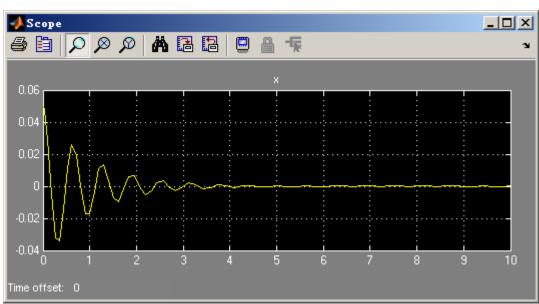


图 7.1-9 采用轴自动设置功能后的信号显示

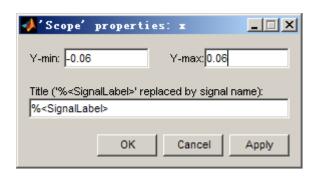


图 7.1-10 对显示屏的纵坐标范围进行设置



图 7.1-11 对示波器时间显示范围的设置

(12) 据试运行结果,进行仿真参数的再设置。

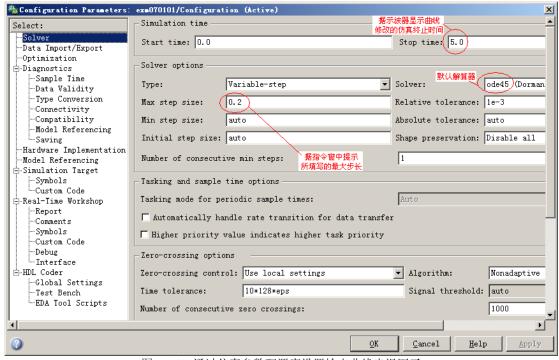


图 7.1-12 通过仿真参数配置窗设置输出曲线光滑因子

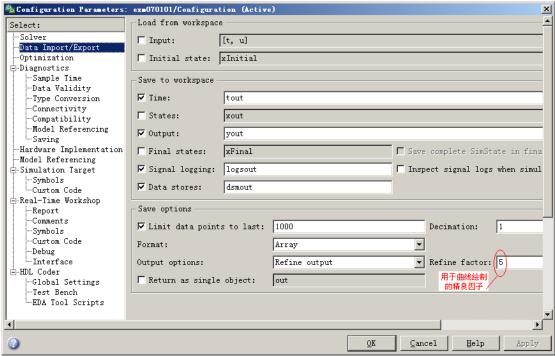


图 7.1-13 通过仿真参数配置窗设置输出曲线光滑因子

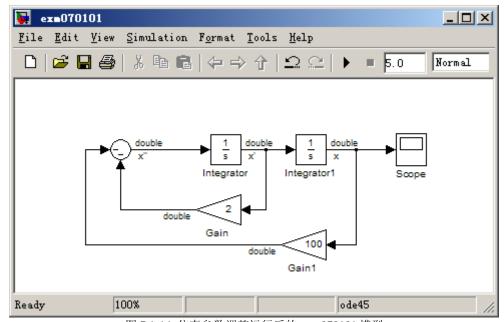


图 7.1-14 仿真参数调整运行后的 exm070101 模型

(13) 仿真结果显示

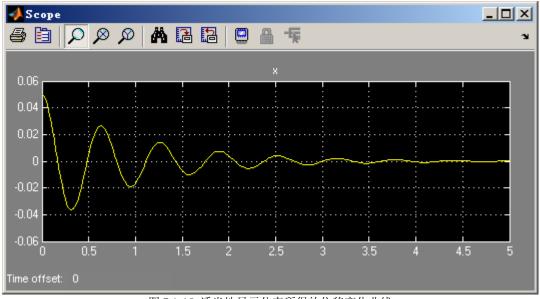


图 7.1-15 适当地显示仿真所得的位移变化曲线

7.1.2 基于传递函数的SIMULINK建模

【例 7.1-2】对于图 7.1-15 所示的多环控制系统,(1)求系统传递函数 $G(s) = \frac{Y(s)}{U(s)}$; (2)求该系统的单位阶跃响应。

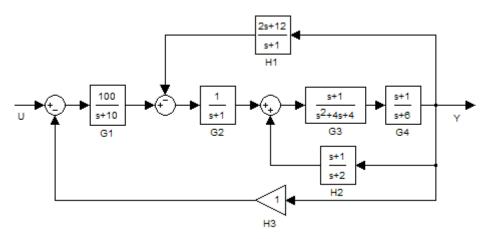


图 7.1-15 多环控制系统

- (1) 建模的基本思路
- (2) 构造"用于系统传递函数计算"的 SIMULINK 模型

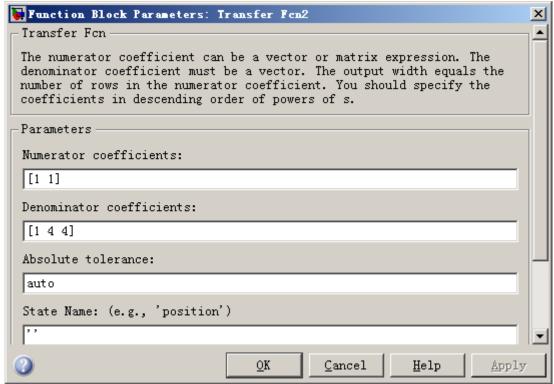


图 7.1-16 传递函数参数设置对话窗

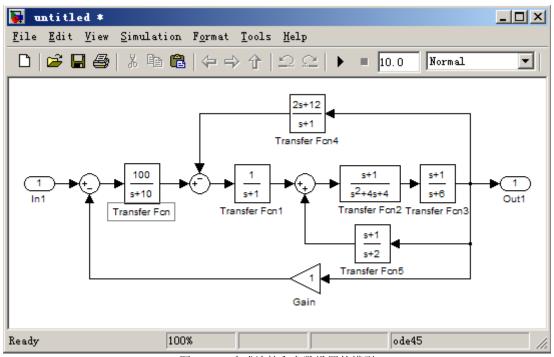


图 7.1-17 完成连接和参数设置的模型

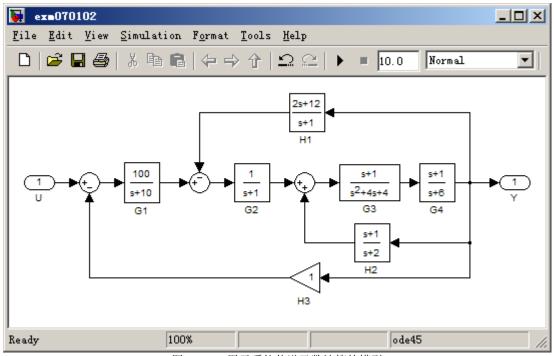


图 7.1-18 用于系统传递函数计算的模型

```
(3)
[A,B,C,D]=linmod2('exm070102');
STF=tf(minreal(ss(A,B,C,D)))
                                    %
[Num,Den]=tfdata(STF);
Num{:},Den{:}
                             %
2 states removed.
Transfer function:
            100 \text{ s}^2 + 300 \text{ s} + 200
s^5 + 21 s^4 + 157 s^3 + 663 s^2 + 1301 s + 910
ans =
                  0
                       0 100.0000 300.0000 200.0000
ans =
 1.0e+003 *
   0.0010 0.0210 0.1570 0.6630 1.3010 0.9100
(4)
t0=(0:0.1:5)';
[y,t]=step(STF,t0);
                              %
plot(t,y,'LineWidth',3)
grid on
axis([0,5,0,0.4])
xlabel('t'),ylabel('y')
```

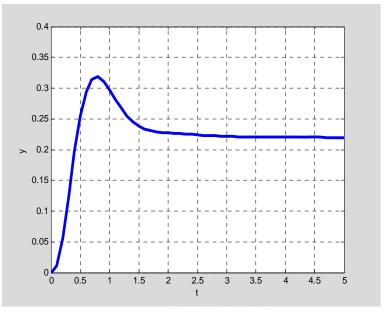


图 7.1-19 系统的单位阶跃响应

7.2 离散时间系统的建模与仿真

【例 7.2-1】设计一个数字低通滤波器 F(z),从受噪声干扰的多频率混合信号 x(t) 中获取 10Hz 的信号。

$$x(t) = \sin(2\pi \cdot 10 \cdot t) + 1.5\cos(2\pi \cdot 100 \cdot t) + n(t)$$
 (7.3)

在此, $n(t) \sim N(0,0.2^2)$, $t=k\cdot \frac{1}{f_s}=k\cdot T_s$ 。 采样频率取 $f_s=1000$ (Hz),即采样

周期 $T_s = 0.001$ (秒)。本例演示: 纯离散时间系统建模的完整过程; 离散时间仿真模型中采样周期的设定; SIMULINK 模块参数与 MATLAB 内存变量之间的数据传递; 影响模块几何结构的参数。

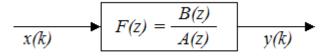


图 7.2-1 数字滤波示意图

- (1) 分析
- (2) 模型的构建

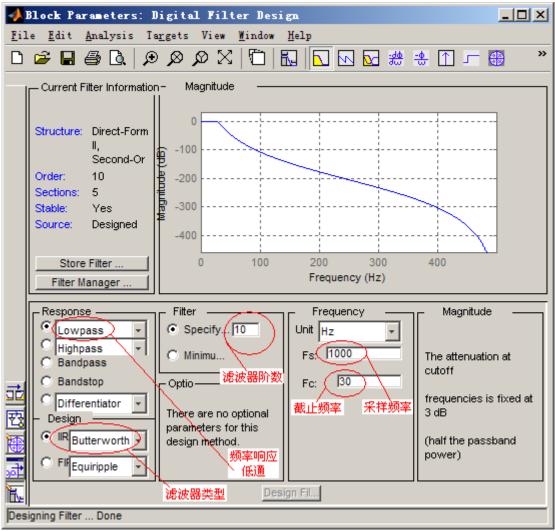


图 7.2-2 数字滤波设计模块参数设置

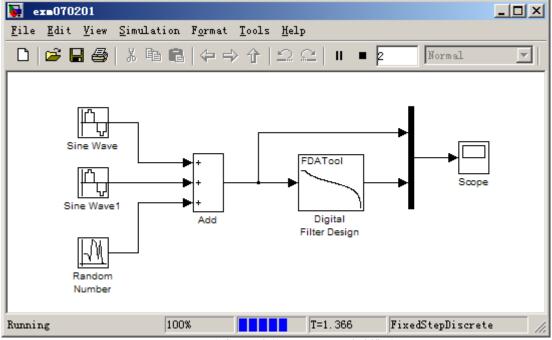


图 7.2-3 仿真进程中的 SIMULINK 滤波模型

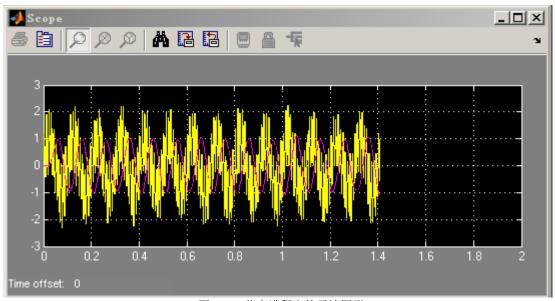


图 7.2-4 仿真进程中的示波图形

7.3 SIMULINK实现的元件级电路仿真

【例 7.3-1】在图 7.3-1 所示的电路中,已知 L=0.3(H), C=0.3(F), $R_1=2(\Omega)$, $R_2=0.01(\Omega)$, $R_3=5(\Omega)$, $V_C(0^-)=-1(V)$, $i_L(0^-)=1(A)$, $V_s=10(V)$, 开关 K 在 t=0 时闭合。试采用 SIMULINK 的 SimPowerSystems 模块库器件进行元件级仿真,求 i_L 和 V_C 。

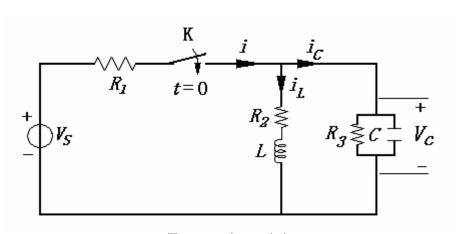


图 7.3-1 二阶 RLC 电路

- (1) 建模的基本思路
- (2) 仿真模型所用器件的来源及参数设置
- (3) 关于仿真模型元器件间连线的说明

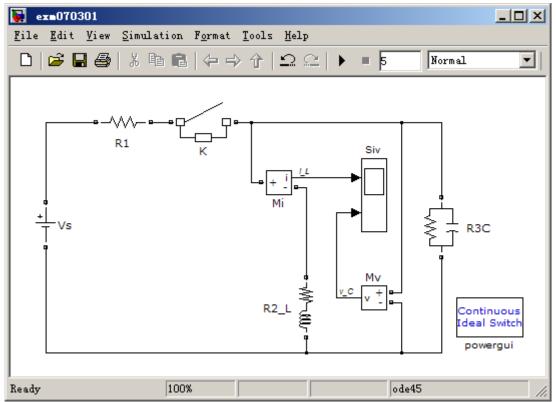


图 7.3-2 元件级仿真模型 exm070301

(4) 营造 SimPowerSystems 仿真环境的<powergui>模块



图 7.3-3 exm070301 模型的 powergui 对话窗

(5) 理想开关的设置

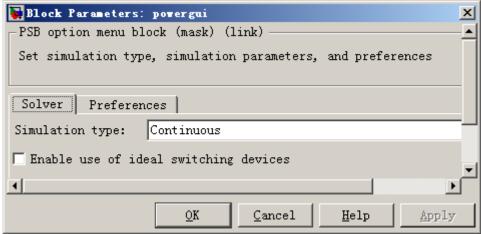


图 7.3-4 仿真类型和理想开关设置窗的默认界面

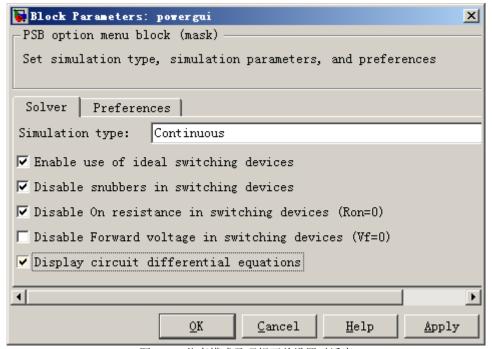


图 7.3-5 仿真模式及理想开关设置对话窗

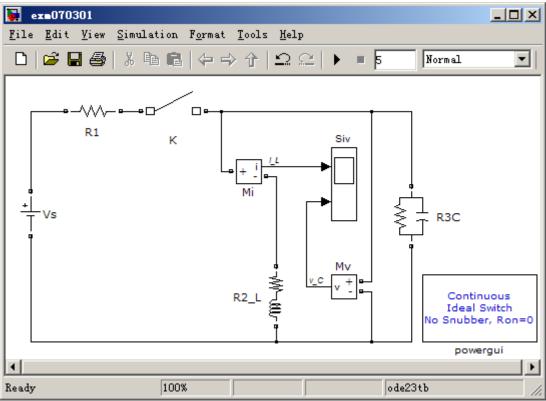


图 7.3-6 理想开关设置后的模型窗

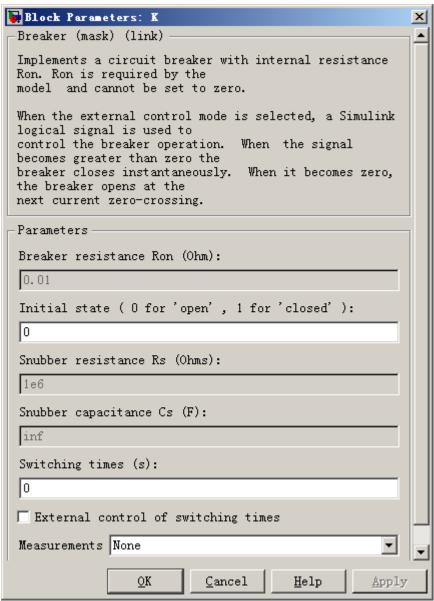


图 7.3-7 开关经失能设置后的对话窗

- (6) 仿真类型的选定
- (7) 模型初始状态的设定

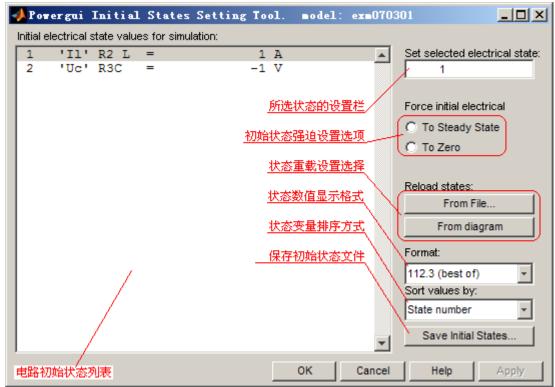


图 7.3-8 powergui 的初始值设置窗

(8) 仿真运行

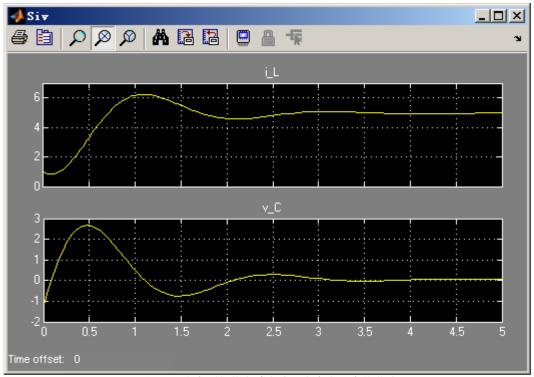


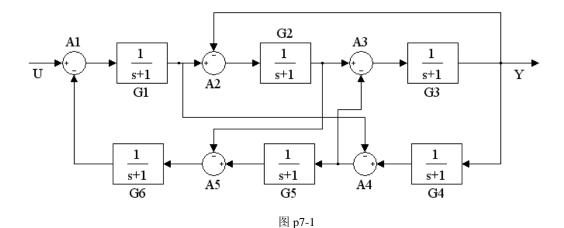
图 7.3-9 仿真所得的电感电流和电容电压变化曲线



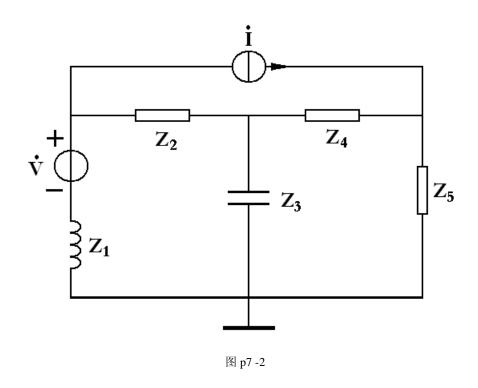
图 7.3-10 RLC 库器件原型

习题 7

- 1. 利用 SIMULINK 求解 $I(t) = \int_0^t e^{-x^2} dx$ 在区间 $t \in [0,1]$ 积分,并求出积分值 I(1) 。(提示:时间变量由 Clock 产生;注意使用 Product, Math function, Integrator, Display, Scope 等库模块;计算结果可与例 4.1-5 对照。)
- 2. 利用 SIMULINK 求解微分方程 $\frac{d^2x}{dt^2} \mu(1-x^2)\frac{dx}{dt} + x = 0$,方程的初始条件为 $x(0) = 1, \frac{dx(0)}{dt} = 0$ 。在增益模块 "Gain"取值分别为 2 和 100 的情况下(即数学表达 式中 $\mu = 2,100$)运行,给出运行结果。(提示:注意使用 Constant, Product, Add, Gain, Integrator, Scope 等库模块;注意初始状态设置;针对不同 μ ,采用不同解算器,并设置不同仿真终止时间;运算结果可与例 4.1-9 对照。)
- 3. 已知某系统的框图如图 p7-1 所示, 求该系统的传递函数。(提示:参照例 7.1-2)



- 4. 采用 SIMULINK 基本库和 Signal Processing Blockset 库的"连续"时间的模块构建的 SIMULINK 模型解决第 7 章算例 7.2-1。(提示:可使用 Sine Wave, Analog Filter Design 等库模块;注意采样时间设置)
- 5. 在如图 p7-2 所示的交流电路中,其中 $Z_1 = j1(\Omega)$, $Z_2 = Z_4 = Z_5 = 1(\Omega)$, $Z_3 = -j1(\Omega)$, $\dot{V} = 20\angle 120^\circ(V)$, $\dot{I} = 10\angle 45^\circ(A)$, f = 50 (HZ),求 Z_3 支路中的电流和 Z_5 两端的电压。(提示:采用相量分析法;要注意电压源、电流源库模块的频率、相角设置;特别要注意模块所采用的幅值是"峰幅值",它应是"有效值"的 $\sqrt{2}$ 倍;电压、电流测量模块的输出信号,可选择 Magnitude-Angle 幅相模式;powergui 模块的仿真类型 Simulation type 应选择"相量 Phasor";仿真终止时间非 0 即可;仿真可采用纯离散解算器 discrete。)



第8章 图形用户界面(GUI)

假如读者所从事的数据分析、解方程、计算结果可视工作比较单一,那么一般不会考虑图形用户界面(Graphical User Interfaces ,GUI)的制作。但是如果读者想向别人提供某种新的设计分析工具,想体现某种新的设计分析理念,想进行某种技术、方法的演示,那么图形用户界面也许是最好的选择之一。

MATLAB 本身提供了很多的图形用户界面,如 sisotool "单输入单输出控制系统设计工具"、fdatool "滤波器设计和分析工具"等。这些工具的出现不仅提高了设计和分析效率,而且改变原先的设计模式,引出了新的设计思想,改变了和正在改变着人们的设计、分析理念。正是出于这种观察,作者将图形用户界面内容列入本教程。

本章由四个精心设计的算例组成。这四个算例,一方面尽可能多地向读者展现构成 GUI 的各种控件或组件,另方面借助算例中回调函数的编写较快地将读者引向 GUI 制作的纵深。

本书作者相信:读者只要耐心地按照示例进行操作,便能在愉快感受 GUI 成功制作的同时,事半功倍地掌握 GUI 的制作技巧。

8.1 图形用户界面入门示例

【例 8.1-1】为演示归一化二阶系统 $G(s) = \frac{1}{s^2 + 2\zeta s + 1}$ 中阻尼比 ζ 对单位阶跃响应的影响,

需要制作如图 8.1-1 所示的用户界面。要求: 在界面右侧的编辑框中输入阻尼比 ζ 的具体"大于 0 的数值标量",就能在坐标轴上画出相应曲线。

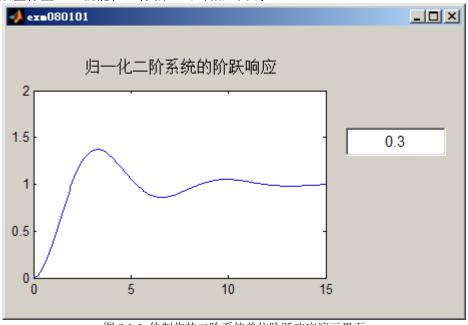


图 8.1-1 待制作的二阶系统单位阶跃响应演示界面

(1) GUIDE 开发环境的引入

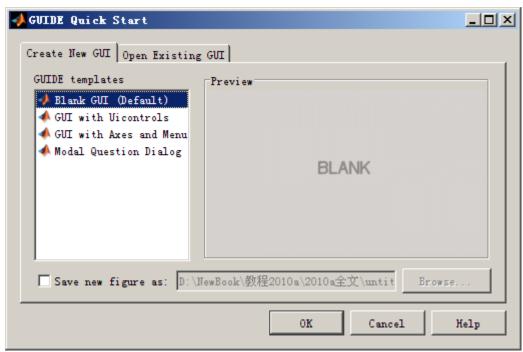


图 8.1-2 进入 GUIDE 开发环境的默认引导对话窗

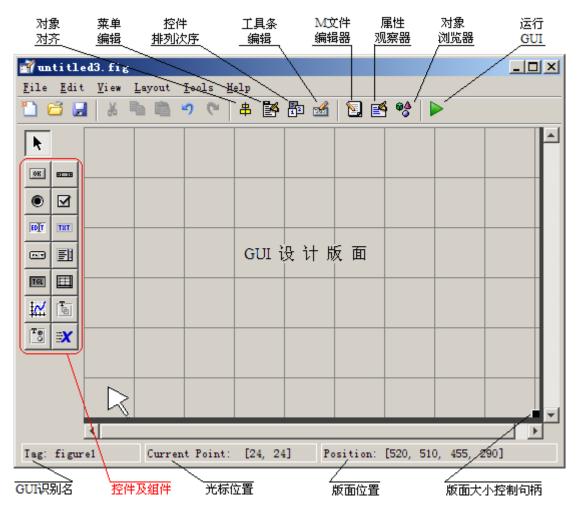


图 8.1-3 默认的空白用户界面开发环境 GUIDE

(2) 根据要求选配界面组件



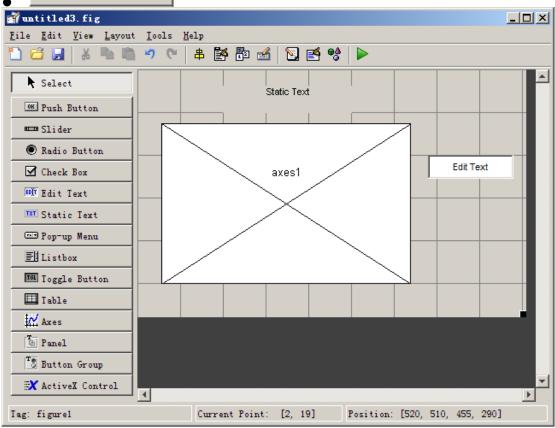


图 8.1-4 选配界面组件后的版面编辑器

(3) 界面组件的参数设置

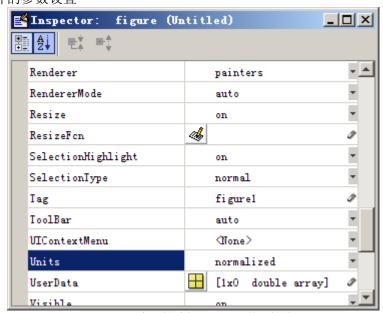


图 8.1-5 窗属性编辑器和设置的属性值

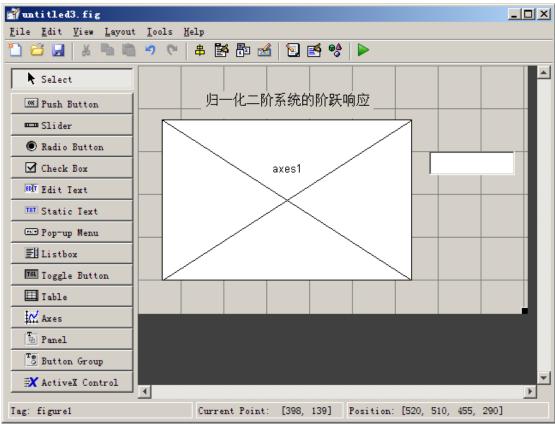


图 8.1-6 经属性设置后的设计版面

(5) 初步设计界面的保存

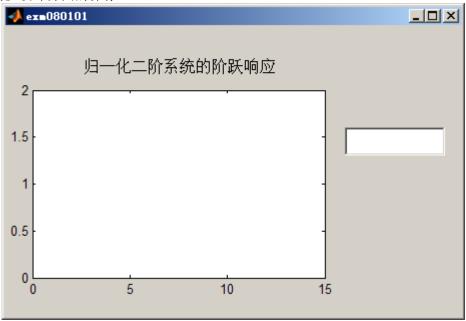


图 8.1-7 仅在外形上与题给要求一致的 GUI

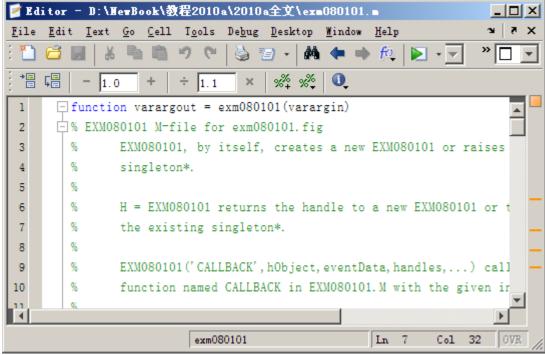


图 8.1-8 版面编辑器自动生成的设计界面 M 文件

(6) 回调程序的编写

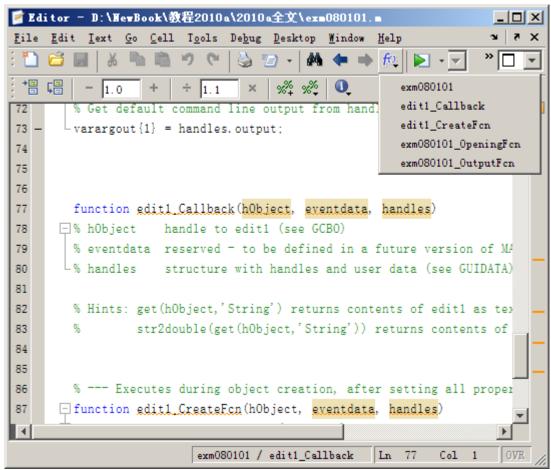


图 8.1-9 GUIDE 生成的"空白回调函数体"示图

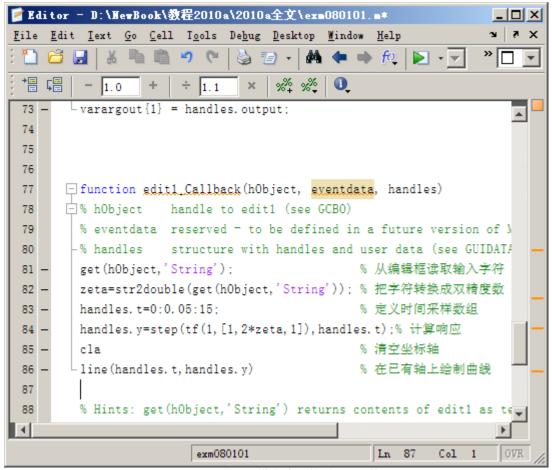


图 8.1-10 填写工作指令后的回调子函数

(7) 所建界面的使用

8.2 控件创建及应用示例

【例 8.2-1】为归一化二阶系统 $G(s) = \frac{1}{s^2 + 2\zeta s + 1}$ 单位阶跃响应制作如图 8.2-1 所示的用户

界面。要求: (1)通过编辑框和滑键都能输入阻尼比。(2)刚启动的界面初始形态如图 8.2-1 所示。(3)在刚启动的初始界面上,响应曲线用红线绘制;而一旦界面被操作,则响应曲线将用蓝线绘制。(4)在列表框中的三个选项可以任意组合。

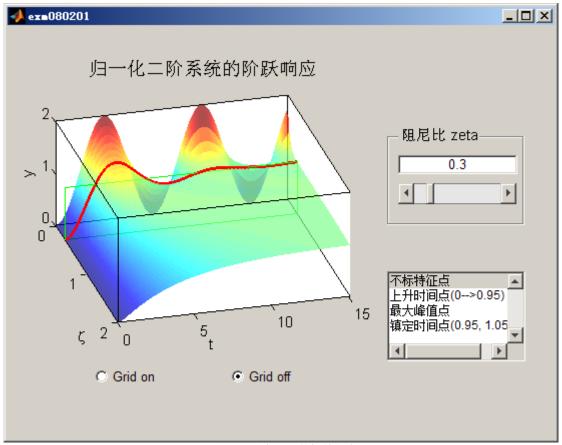


图 8.2-1 用户界面的初始状态

- (1) GUI 版面编辑器的开启
- (2) 对未来界面窗属性设置
- (3) 为版面设计区引入坐标参照系

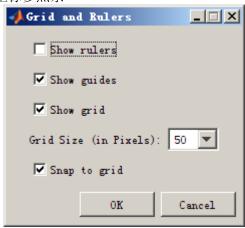


图 8.2-2 "格尺"对话框(默认状态)

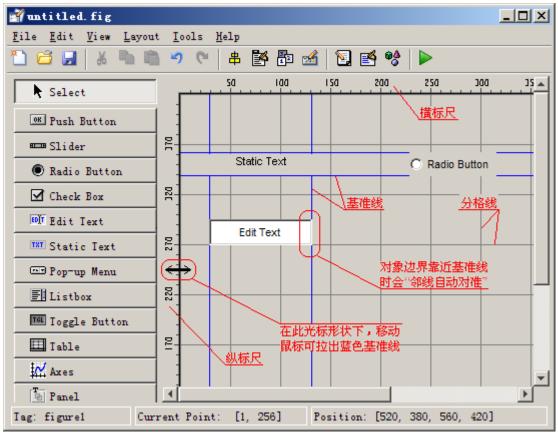


图 8.2-3 一般坐标参照工具示意

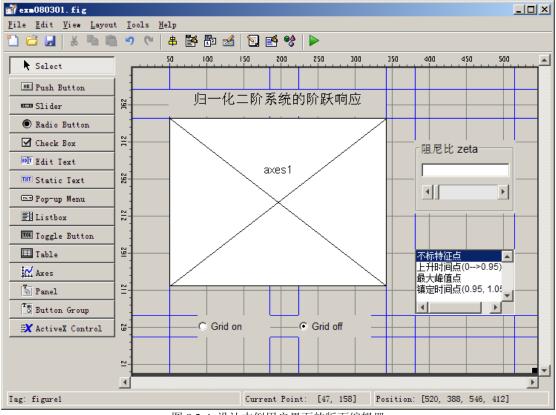
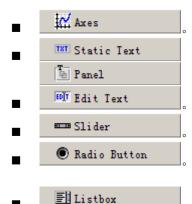


图 8.2-4 设计本例用户界面的版面编辑器

(4) 根据题目要求进行界面构建



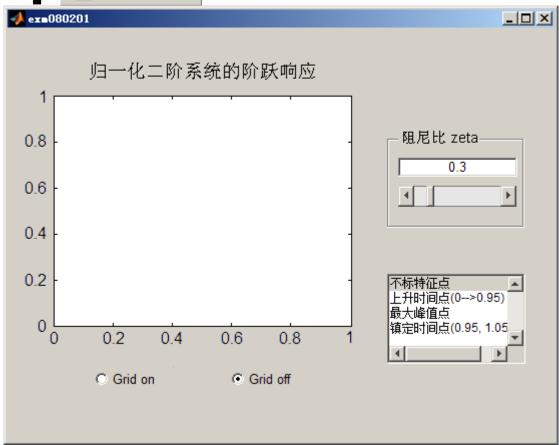


图 8.2-5 执行文件尚未填写时的生成界面

(5) 由版面编辑器自动产生的 exm080201.m 文件的结构

(6) 编写初始化程序

function exm080201_OpeningFcn(hObject, eventdata, handles, varargin)

- % This function has no output args, see OutputFcn.
- % hObject handle to figure
- % eventdata reserved to be defined in a future version of MATLAB
- % handles structure with handles and user data (see GUIDATA)
- % varargin command line arguments to exm080201 (see VARARGIN)
- %U_Start --- "%U_Stat"和 "%U_End"之间的 M 码均由作者编写-----U Start zeta=0.3;

set(handles.edit1,'String',num2str(zeta))

set(handles.slider1,'Value',zeta)

set(handles.gridon,'Value',0)

set(handles.gridoff,'Value',1)

```
set(handles.listbox1,'Value',1)
handles.t=0:0.05:15;
handles.Color='Red';
handles.zeta=zeta;
handles.flag=0;
handles=surfplot(handles);
handles.flag=1;
handles.Color='Blue':
                    -----U End
%U End -----
handles.output = hObject;
guidata(hObject, handles);
 (7) 可编辑框的回调子函数
function edit1_Callback(hObject, eventdata, handles)
% hObject
% eventdata reserved - to be defined in a future version of MATLAB
% handles
% U_Start------U_Start
sz=get(hObject,'String');
zeta=str2double(sz);
set(handles.slider1,'Value',zeta)
handles.zeta=zeta;
handles=surfplot(handles):
guidata(hObject, handles);
%U_End-----U End
 (8) 滑键回调子函数
function slider1 Callback(hObject, eventdata, handles)
%U Start-----U Start
zeta=get(hObject, 'Value');
set(handles.edit1, 'String', num2str(zeta))
handles.zeta=zeta;
handles=surfplot(handles);
guidata(hObject, handles);
%U End------U End
 (9) 无线电按键回调子函数
function gridon_Callback(hObject, eventdata, handles)
%U Start------U Start
set(handles.gridoff,'Value',0)
grid on
%U End-----U End
function gridoff_Callback(hObject, eventdata, handles)
%U Start------U Start
set(handles.gridon,'Value',0)
grid off
%U End------U End
(10) 列表框回调子函数
function listbox1 Callback(hObject, eventdata, handles)
%U Start-----U Start
listindex=get(hObject,'Value');
```

```
if any(listindex==1)
  set(handles.listbox1,'Value',1)
end
handles.flag=0;
handles=surfplot(handles);
handles.flag=1;
guidata(hObject, handles);
%U End-----
                           -----U End
 (11) 绘图子函数
function handles=surfplot(handles)
% handles=surfplot(handles)
% handles
%
zeta=handles.zeta;
t=handles.t;
listindex=get(handles.listbox1,'Value');
Nt = length(t);
if handles.flag==0
  cla
  zmin=get(handles.slider1,'Min');
  zmax=get(handles.slider1,'Max');
  zt=zmin:0.05:zmax;
  Nz = length(zt);
  [ZT,T]=meshgrid(zt,t);
  Y=zeros(Nt,Nz);
  for k=1:Nz
     Y(:,k)=step(tf(1,[1,2*zt(k),1]),t);
  end
  surface(ZT,T,Y)
  shading flat
else
  delete(handles.g1)
  delete(handles.rline)
end
xz=ones(1,Nt)*zeta;
y1=ones(1,Nt)*1;
y = step(tf(1,[1,2*zeta,1]),t);
gz=[zeta,zeta,xz,zeta,zeta,xz];
gt=[t(1),t(1),t,t(end),t(end),fliplr(t)];
gy=[0,1,y1,1,0,0*y1];
handles.g1=line(gz,gt,gy,'Color','g','LineWidth',1);
handles.rline=line(xz,t,y,'Color',handles.Color,'LineWidth',2);
K=length(get(handles.listbox1,'Value'));
for jj=1:K
  switch listindex(jj)
    case 1
    case 2
       k95 = min(find(y>0.95)); k952 = [(k95-1), k95];
       t95=interp1(y(k952),t(k952),0.95);
       line(zeta,t95,0.95,'marker','+','markeredgecolor','k','markersize',6);
    case 3
       [ym,km]=max(y);
       if km < Nt & (ym-1) > 0
         line(zeta,t(km),ym,'marker','.','markeredgecolor','k','markersize',5);
       end
```

(12) 用户界面的运行

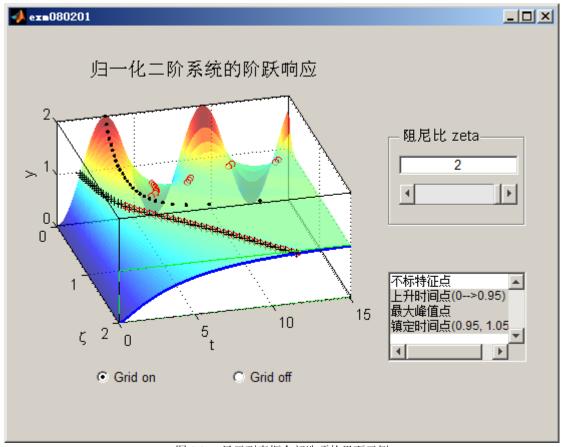


图 8.2-6 显示列表框全部选项的界面示例

〖说明〗

8.3 菜单及工具图标的设计示例

8.3.1 为界面配置标准菜单条和工具条

【例 8.3-1】在例 8.2-1 产生的界面上,配置 MATLAB 标准图形窗菜单。

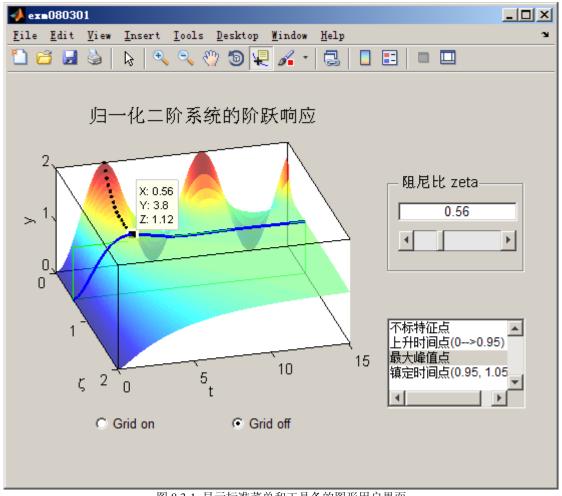


图 8.3-1 显示标准菜单和工具条的图形用户界面

- (1) 以例 8.2-1 的用户界面为基础制作新界面
- (2) 利用属性编辑器为界面配置标准菜单条和工具条
- (3) 界面操作示范

〖说明〗

8.3.2 菜单定制和标准图标选用

【例 8.3-2】制作如图 8.3-2 所示的图形用户界面。该用户界面上,不仅配置了一个定制菜单,用以控制界面上坐标框的是否封闭,而且还配置了一个"图形数据标识图标"。

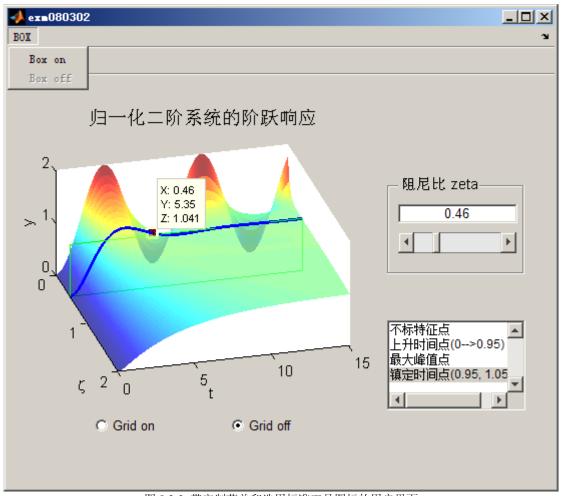


图 8.3-2 带定制菜单和选用标准工具图标的用户界面

- (1) 利用已有用户界面为基础制作新界面
- (2) 撤销原界面上的标准菜单和工具条
- (3) 定制菜单

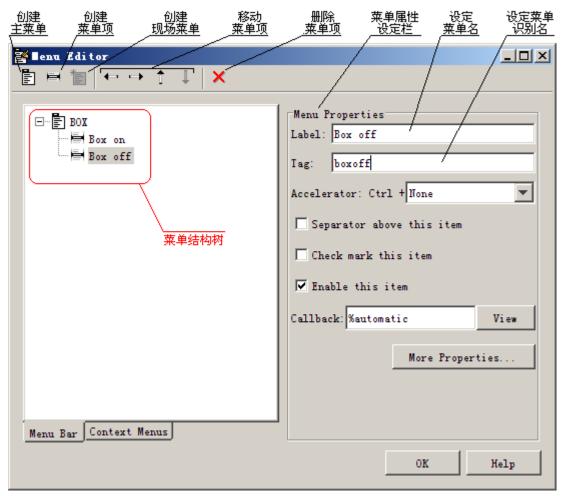


图 8.3-3 菜单编辑器的外形和功能分区

(5) 选用标准工具图标

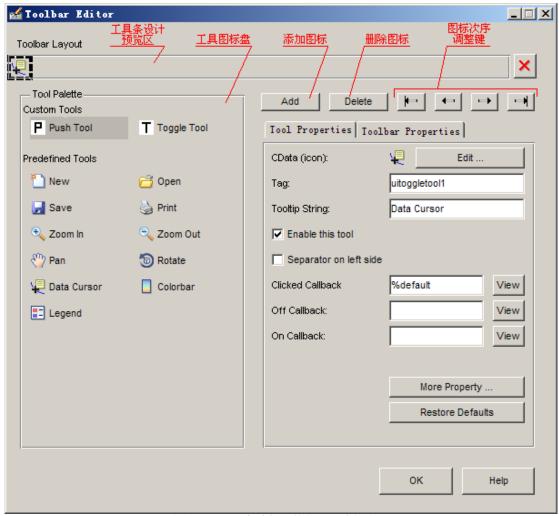


图 8.3-4 工具条编辑器的外形和功能分区

(4) 保存及生成题目所要求的用户界面

〖说明〗

习题8

1. 以算例 8.3-2 的图形用户界面文件 exm080302.m 和 exm080302.fig 为基础, 删除原界面上的两个无线电按键, 以便获得如图 p8-1 的新图形用户界面。

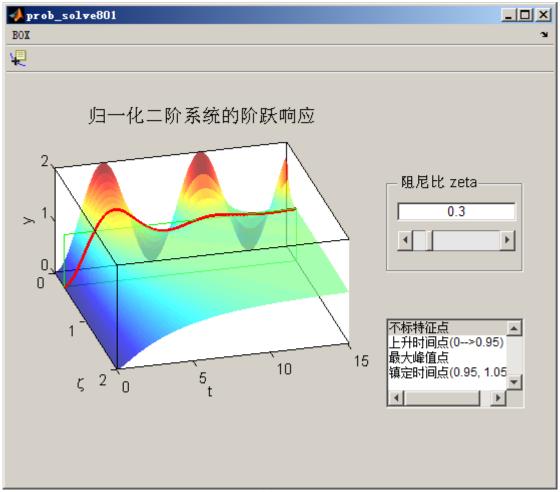


图 p8-1

2. 以算例 8.3-2 的图形用户界面文件 exm080302.m 和 exm080302.fig 为基础, 用双稳态按键 Toggle Button 替代无线电按键实现对坐标网格绘制的控制, 产生如图 p8-2 的新图形用户界面。

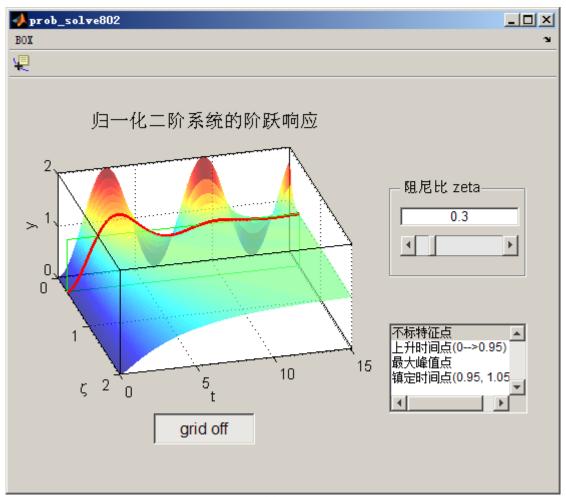


图 p8-2

第9章 Notebook

9.1 Notebook的配置和启动

9.1.1 Notebook的配置

9.1.2 Notebook的启动

1 创建新的M-book文件

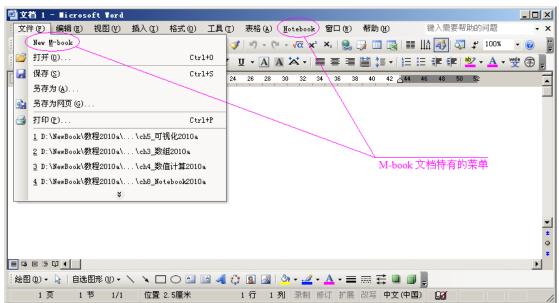


图 9.1-1 新建 M-book 文档的界面

2 打开已有的M-book文件

- (1) 在 Word 默认窗口下打开已有的 M-book 文件
- (2) 在资源管理器中打开已有的 M-book 文件
- (3) 在 MATLAB 当前目录窗中打开已有的 M-book 文件
- (4) 在 MATLAB 指令窗中开启已有的 M-book 文件

9.2 M-book模板的使用



图 9.2-1 {Notebook} 下拉菜单的常用功能项

9.2.1 输入细胞(群)的创建和运行

1 细胞(群)

2 输入细胞(群)操作示例

【例 9.2-1】

(1) 输入细胞的单纯生成法

```
xx=(1:5)/5*pi;yy=sin(xx).*exp(xx)
```

(2) 输入细胞生成、运行同时进行的操作方法

```
x=(1:4)/4*pi;y=sin(x).*exp(x)
y =
    1.5509    4.8105    7.4605    0.0000
```

```
【例 9.2-2】
```

```
t=0:0.1:10;y=1-cos(t).*exp(-t);
                                                                 %<1>
tt=[0,10,10,0];
yy=[0.95,0.95,1.05,1.05];
fill(tt,yy,'g'),axis([0,10,0,1.2]),xlabel('t'),ylabel('y')
                                                                 %<4>
                                                                 %<5>
hold on
plot(t,y,'k','LineWidth',4)
                                                                 %<6>
hold off
                                                                 %<7>
ymax=max(y)
                                                                 %<8>
ymax =
    1.0669
```

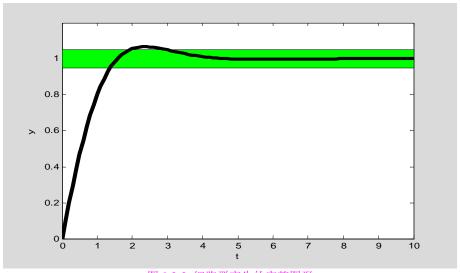


图 9.2-2 细胞群产生的完整图形

9.2.2 Notebook菜单的其他选项

- 1 自初始化细胞及其应用
- (1) 自初始化细胞
- (2) 工作内存的初始化
- 2 删去M-book 文件所有输出细胞
- 9.2.3 输出细胞的格式控制



图 9.2-3 控制输出细胞格式的对话框

- 1 输出数据的表示法
- 2 输出数据间的空行控制
- 3 图形的嵌入控制

```
【例 9.2-3】
surf(peaks) %
colormap(hot)
t=(0:50)/50*pi;y=sin(t);
plot(t,y) %
```

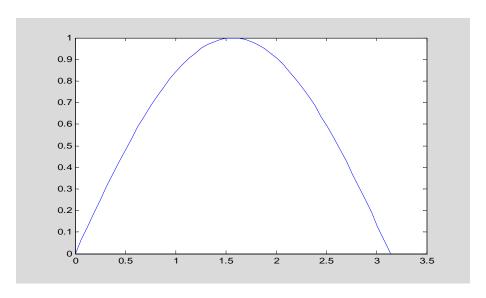


图 9.2-4 同一细胞群中最后一幅绘制的曲线图

- 4 嵌入图形大小的控制
- 5 嵌入图形的背景色问题
- 6 M-book处理活动画面的能力

【例 9.2-4】

anim_zzy1(1),shg

9.3 使用 M-book 模板的若干参考技法

附录A: 字符串、胞元及构架数组

A.1 字符串数组

```
【例 A.1-1】
clear
a=12345.6789
                  %
class(a)
                  %
a s=size(a)
a =
 1.2346e+004
ans =
double
a_s =
b='S'
class(b)
b_s=size(b)
b =
S
ans =
char
b_s =
  1
whos
 Name
                         Bytes Class Attributes
          Size
           1x1
                              8 double
 a
                             16 double
           1x2
 a_s
                             8 char
          1x4
 ans
                             2 char
 b
          1x1
          1x2
                            16 double
 b_s
【例 A.1-2】
(1)
a='This is an example.'
a =
This is an example.
(2)
size(a)
ans =
    1
        19
(3)
A='这是算例。'
A =
这是算例。
ab=[A(1:4),'A.1-2',A(5)]
ab =
```

这是算例 A.1-2。

```
【例 A.1-3】
(1)
A=eye(2,4);
A_str1=int2str(A)
A_str1 =
1 0 0 0
  1
     0 0
(2)
rand('twister',0)
B=rand(2,4);
B3=num2str(B,3)
B3 =
                    0.424
                               0.438
0.549
          0.603
0.715
          0.545
                    0.646
                               0.892
【例 A.1-4】
clear
a=2;
                 %
w=3;
                 %
t=0:0.01:10;
                 %
y=exp(-a*t).*sin(w*t);
[y_max,i_max]=max(y);
                                  %
t_text=['t=',num2str(t(i_max))]; %
                                             <7>
                                             <8>
y_text=['y=',num2str(y_max)];
max_text=char('maximum',t_text,y_text);%
                                             <9>
tit=['y=exp(-',num2str(a),'t)*sin(',num2str(w),'t)'];
                                  %
                                             <11>
plot(t,zeros(size(t)),'k')
                                            %
hold on
                                            %
plot(t,y,'b')
                                            %
plot(t(i_max),y_max,'r.','MarkerSize',20)
text(t(i_max)+0.3,y_max+0.05,max_text) %
                                             <16>
title(tit),xlabel('t'),ylabel('y')
hold off
```

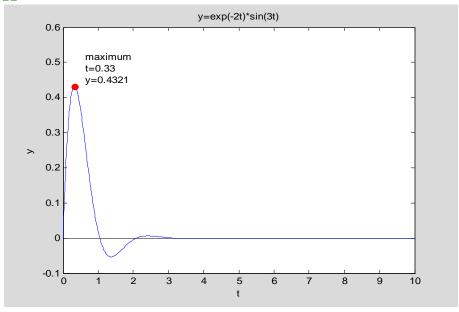


图 A.1-1 字符串运用示意图

A.2 胞元数组

```
【例 A.2-1】
clear
C_str='这是胞元数组创建算例 1';
R=reshape(1:9,3,3);
Cn=[1+2i];
S_sym=sym('sin(-3*t)*exp(-t)');
%
B{1,1}=C_str;
                                                                %<6>
B\{1,2\}=R;
B{2,1}=Cn;
B{2,2}=S_sym;
                                                                %<9>
%
a=B(1,2)
                    %
class(a)
    [3x3 double]
ans =
cell
b=B\{1,2\}
                    %
class(b)
b =
     1
                 7
     2
           5
                 8
     3
           6
                 9
ans =
double
         构架数组
A.3
(1)
clear
                           %
G.name='一号房';
                                        <1>
G.volume=2000;
                                        <2>
G.temperature=[31.2,30.4,31.6,28.7];
                                        <3>
G.humidity=[62.1,59.5,57.7,61.5;63,60,58.1,62.3];
 (2)
G(2).name='二号房';
G(2).volume=2400;
(3)
G
                                        <7>
G =
1x2 struct array with fields:
   name
    volume
```

附录B: 光盘使用说明

B.1 光盘文件的结构

在光盘上有如下两个文件夹:

● **mbook** 存放着包含本书全部算例的 M-book 形式 DOC 文档。

● mfiles 汇集了全书 90%以上算例的 M 文件、MDL 文件和 FIG 文件。

B.2 光盘对软件环境的要求

- 需要 Offiece2003, XP 和 MATLAB R2010a 支持。(书中除 Simulink 模型 MDL 文件外,其他绝大多数指令和文件在 MATLAB 其他新老版本中也都能运行。)
- 假如要完好运行 mbook 文件夹上的 DOC 文件,则需要 Notebook 环境。(关于 Notebook 环境的设置参见第8章)

B.3 光盘文件的操作准备

在运行光盘文件之前,应首先把\mfiles 文件夹设置为当前目录或设置在 MATLAB 的搜索路径上(具体方法详细参见第 1.6 节)。

B.4 mbook文件夹上DOC文件的使用

光盘 DOC 文件在 MATLAB **R2010a** 的 Notebook 环境中生成,DOC 文件各章节的编号、名称与印刷版完全一致。

(1) 光盘 DOC 文件的功用

弥补了印刷版丢失的色彩信息

在 MATLAB中,用 M 文件编辑器或 Notebook 编写的指令或文件运行的结果(尤其图形)都采用不同的色彩鲜明地表现对象特征。但目前印刷版书籍出于价格和技术原因,不得不牺牲色彩信息而采用"黑白"处理。读者借助本光盘可克服印刷版丢失色彩信息的缺憾。

● 为教师制作本教材电子讲稿提供模板

据本人十多年 MATLAB 教学经验,作者认为:在 MATLAB 的课堂教学中,电子讲稿最好使用 M-book 模板制作,而不宜采用 PowerPoint 制作。正是出于这样的考虑,本书光盘提供了各章的 DOC 文件。它们包含完整的章节结构和名称,包含所有算例的题解要求和完整的解题程序。

主讲这门课程的老师,可以根据自己经验和心得,针对具体教学对象,通过对本 光盘 DOC 文件进行适当的剪裁,增补数量不多但相当醒目的提示、警告和归纳性文 字,就可得到因地制宜的电子讲稿。

● 为学生完成电子作业提供模板

学生平时作业最好使用 M-book 模板完成。一,因为只有光盘电子版才准确描述了习题的要求(如图形色彩、光照等); 二,只有在 M-book 文件中,才能把文字解释、MATLAB 指令、计算结果及图形有机地结合在一起。

● 为习题提供答案

由于有些习题答案(如图形等)必须通过光盘电子版才能较准确地表达,所以本书相关章节的习题答案被放置在光盘中。

● 为读者提供了与印刷版对应的 Notebook 演练环境

本光盘中 DOC 文件的章节结构、算例编号与印刷版完全相同。因此在学习过程中,读者可在本光盘启动的 Notebook 环境中,或直接运行算例,观察运行结果;或改变若干指令,举一反三地观察运行结果的变化;或通过简单的复制操作,使相应指令在 MATLAB 指令窗中运行,而避免自己键入的错误。

(2) DOC 文件的开启

所有 DOC 文档都是在"Word + MATLAB R2010a"构成的 Notebook 环境中生成的。 因此,在相同环境下开启是最佳选择。此时,文档具有"活性"。假如读者的 MATLAB 与 Word 联接正确,用鼠标双击光盘上的 DOC 文件,就能直接进入 Notebook 环境。

(3) 光盘 DOC 文件的使用方法

● 作为演练环境使用

在正常打开的光盘 DOC 文件中,读者只要把光标放在绿色的输入细胞内,按组合键 [Ctrl + Enter],就可使该输入细胞重新执行计算。在演练中,读者可以通过对指令的修改、变化和重新运行,观察运算结果的变化,从而达到举一反三的效果。

● 作为样板使用

先打开光盘 DOC 文件,然后删去原光盘文件内容,再写入读者自己所需的内容,最后通过菜单项的"另存为"操作保存为自己的文件。这样获得的文件能正常地在 Notebook 环境下工作,也就是既可以输入文字、公式,又可以运行 MATLAB 指令、嵌入数值或图形结果; 既拥有 Word 的所有文字处理能力,又具备 MATLAB 的运算、表现能力。

B.5 mfiles文件夹上的M、MDL文件的使用

所有算例都以 M 文件或 MDL 文件形式刻录在光盘的 \mfiles 文件夹中。

本光盘M文件应在MATLAB **R2010a**以上版本运行,涉及符号计算的应有Symbolic Math Toobox 3.2.3 以上版本-适配,MDL文件应有SIMULINK7.1 以上版本适配。

(1) 光盘 M、MDL 文件的功用

提供可直接运作的 M 源码文件

只要有 MATLAB 环境,本光盘上的 M 文件就可以运行。它的适用条件比 \mbook 文件夹上的 DOC 文件宽松得多,也就是不管读者是否正确安装 Notebook,不管文件产生的是动画还是交互操作界面,它们都能在 MATLAB 环境中正确执行。

每个算例文件都是完整的,可在 MATLAB 环境中直接运行的,所得结果与印刷版相对应。但出于运行方式不同的考虑,有些光盘 M 文件与印刷版文件指令可能会存在少许差别,目的是为把算例特征表现得更充分。

此外,本光盘提供的 M 文件中,有许多是很通用的,读者只要稍加修改,就可为己所用。

● 弥补了印刷版没有 SIMULINK 模型文件的缺陷

由于 SIMULINK 工作特点的缘故,所以迄今为止所有涉及 SIMULINK 的印刷版书籍中都没有能直接运行的模型文件。这给读者带来许多困惑和麻烦:一,读者如想验证书中结论,那就不得不从建模做起;二,仿真模块中的参数设置常使初学者顾此失彼,从而造成仿真失败。本光盘上 MDL 模型文件都可直接在 MATLAB 中运行,进行验证。用户也可以在模型打开后,修改参数,观察变化。

(2) mfiles 文件夹上文件的放置规则

exm 为前缀的文件都是可直接运行的算例文件

前缀后的编号与算例编号对应。最左边的两位数字为"章"编号标注,其后两位数字是"节"编号,最后两位数字是节内题号。具体举例如下:

【例 1.3-6】对应的 M 文件是 exm010306.m;

【例 7.1-2】对应的 MDL 模型文件为 exm070102.mdl;

而配用 M 文件是 exm070102s.m;

● 其他非 exm 前缀文件是被调用文件

在\mfiles 文件夹上还有一些不以 exm 为前缀的文件,它们不与算例直接对应,而是算例必不可少的被调用文件,或是做习题所需的演示文件、数据文件。在印刷版上可以找到有关它们的说明。

(3) M、MDL 的使用方法

直接在 MATLAB 指令窗中,运行(不带扩展名的)算例 M 文件名,就可得到相关结果。在此要再次提醒的是:必须把\mfiles 文件夹设置在 MATLAB 的搜索路径上。

B.6 其他

在本书印刷版发行后,光盘软件的更新内容将通过光盘所带的 Readme 文件作简短的概述性说明。

附录C MATLAB指令索引

C.1 标点及特殊符号指令

+	加	(arith)	1.3.3, 2.1.2, 3.3.1, 4.2.1
-	减	(arith)	1.3.3, 2.1.2, 3.3.1, 4.2.1
*	矩阵乘	(arith)	1.3.3, 2.1.2, 3.3.1, 4.2.1
.*	数组乘	(arith)	1.3.3, 2.1.2, 3.3.1
٨	矩阵乘方	(arith)	1.3.3, 2.1.2, 3.3.1
.^	数组乘方	(arith)	1.3.3, 2.1.2, 3.3.1
/	斜杠或右除	(slash)	3.3.1, 4.2.2
\	反斜杠或左除	(slash)	3.3.1,4.2.3,2.6.1
./或.\	数组除	(slash)	1.3.3, 2.1.2, 3.3.1, 3.3.2, 6.1.3
==	等号	(relop 或 eq)	2.1.2, 3.3.1, 3.5.1, 4.4.1, 6.1.3, 6.2.4
~=	不等号	(relop 或 ne)	2.1.2, 3.3.1, 3.5.1
<	小于	(relop 或 lt)	3.2.3, 3.3.1, 3.5.1
>	大于	(relop 或 gt)	3.3.1
<=	小于或等于	(relop 或 le)	3.3.1, 3.5.1
>=	大于或等于	(relop 或 ge)	3.3.1, 3.5.1
&	逻辑与	(relop 或 and)	3.3.1, 3.5.2
	逻辑或	(relop 或 or)	3.3.1, 3.5.2
~	逻辑非	(relop 或 not)	3.3.1, 3.5.2
:	冒号	(colon)	1.4.2, 3.1, 3.2.1, 3.2.3
()	圆括号	(paren)	1.3.3, 1.4.2
[]	方括号、空数组	(paren)	1.4.2, 3.2.2, 3.2.4, 3.4.2
{ }	花括号	(paren)	1.4.2, 2.7.2, 2.7.3, 4.1.1, 5.2.2, 6.1.2, 7.1.2, 附录 A.2
@	创建函数句柄	(punct)	1.4.2, 4.1.3, 4.1.4, 6.3.2, 6.3.3, 6.4.2
	小数点、数组运	(punct)	1.4.2, 2.6.3,附录 A.3
	算标识、构架域		
	续行号	(punct)	1.3.2, 1.4.2
,	逗号	(punct)	1.3.3, 1.4.2, 3.2.2
;	分号	(punct)	1.3.3, 1.4.2, 3.2.2
%	注释号	(punct)	1.4.2, 6.2.4
=	赋值符号	(punct)	1.3.3
[,]	数组元素水平串接	(horzcat)	1.3.3
[;]	数组元素垂直串接	(vertcat)	1.3.3
1	用于形成单引号对	(punct)	1.3.3, 1.4.2, 4.1.3, 5.2.1, 5.2.2, 附录 A.1
	共轭转置号	(transpose)	2.1.2, 3.3.1, 4.2.1
.'	非共轭转置号	(transpose)	2.1.2, 3.3.1

C.2 主要函数指令

\mathbf{A}	ล
Α	a

abs	模	1.3.3, 2.3.1, 3.3.1, 3.3.2
acos	反余弦	2.1.3
acosh	反双曲余弦	2.1.3
acot	反余切	2.1.3
acoth	反双曲余切	2.1.3
acsc	反余割	2.1.3
acsch	反双曲余割	2.1.3
all	所有元素均非零则为真	3.5.3
alpha	透明控制	5.3.3, 5.4.2
angle	相角	1.3.3, 3.3.1
ans	最新表达式的运算结果	1.3.2
any	有非零元则为真	3.5.3
area	面域图	5.2
asec	反正割	2.1.3, 3.3.1
asech	反双曲正割	2.1.3, 3.3.1
asin	反正弦	2.1.3, 3.3.1
asinh	反双曲正弦	2.1.3, 3.3.1
atan	反正切	2.1.3, 3.3.1
atan2	四象限反正切	2.1.3, 3.3.1
atanh	反双曲正切	2.1.3, 3.3.1
autumn	红、黄浓淡色图阵	5.3.3
axis	轴的刻度和表现	1.3.3, 3.5.1, 5.1.2, 5.2.2, 5.3.2, 5.3.3, 6.2.4
B b) . F	
bar	直方图	5.2
binocdf	二项分布累计概率	4.3.1
binopdf	二项分布概率密度	4.3.1
binornd	产生二项分布随机数组	4.3.1
blanks	空格字符	2.4.3
bode	对数频率特性曲线	7.1.2
bone	蓝色调浓淡色图阵	5.3.3

break终止最内循环5.4.3, 6.1.3, 6.1.4brighten控制色彩的明暗5.4.2

butter ButterWorth 滤波器 7.2

坐标封闭开关

C c

box

caxis(伪)颜色轴刻度5.4.1cd设置当前工作目录1.4.3, 1.7.4

4.1.1, 5.2.2, 5.3.4

cdf2rdf	复数对角型转换到	4.2.2
	实数块对角型	
ceil	朝正无穷大方向取整	3.3.1
cell	创建胞元数组	6.1.2
char	创建或转换为字符串	2.3.1, 2.9.1,附录 A.1
charfen	Maple 函数	2.5.3
Children	图形对象的子对象	4.3.1,2.4.2
clabel	等高线标注	5.4.1
class	判别数据类别	2.1.1, 2.1.4, 2.3.1, 3.4.1, 3.4.2, 4.4.1, 6.4.1
clc	清除指令窗中显示内容	1.4.3, 6.1.3
clear	从内存中清除变量和函数	1.3.3, 1.4.3, 1.7.2, 2.1.5, 2.3.3, 6.1.3, 6.2.1
clf	清除当前图形窗图形	1.4.3, 2.4.3, 2.9.1, 5.2.1, 5.3.3
close	关闭图形窗	8.2.3
collect	合并同类项	2.2.3, 2.5.4
Color	图形对象色彩属性	4.3.1, 5.2.1, 5.4.3
colorbar	显示色条	5.4.1, 5.4.2
colorcube	三浓淡多彩交错色图阵	5.3.3
colordef	定义图形窗色彩	8.2.3
colormap	设置色图	2.9.1, 5.3.2, 5.3.3, 5.4.2, 5.4.3
colspace	矩阵列空间基	2.6.1
comet	彗星状轨迹图	5.4.3
comet3	三维彗星动态轨迹线图	5.4.3
compass	射线图;主用于方向和速度	5.2
cond	矩阵条件数	4.2.3
conj	复数共轭	3.3.1
continue	将控制转交给外层的 for	6.1.3, 6.1.4
	或 while 循环	
contour	等高线图	5.4.1
contourf	填色等高线图	5.4.1
conv	卷积和多项式相乘	4.4.1, 4.4.3
cool	青和品红浓淡色图阵	5.3.3
copper	线性变化纯铜色图阵	5.3.3
corrcoef	相关系数	4.3.2
cos	余弦	2.1.3, 3.3.1
cosh	双曲余弦	2.1.3, 3.3.1
cot	余切	2.1.3, 3.3.1
coth	双曲余切	2.1.3, 3.3.1
cov	协方差矩阵	4.3.2
csc	余割	2.1.3, 3.3.1
csch	双曲余割	2.1.3, 3.3.1
cumsum	元素累计和	4.1.2

梯形法累计积分 3.1, 4.1.2, 5.2.3

cumtrapz

•	`	1
	1	$\boldsymbol{\alpha}$
	,	u

expm

矩阵指数

D d		
dblquad	二重(闭型)数值	4.1.3
	积分指令	
deconv	解卷和多项式相除	4.4.1
del2	计算曲率	5.4.2
demos	演示函数	1.9.4
det	行列式的值	2.6.1
diag	创建对角阵,抽取对角 向量	2.6.1, 3.2.2, 3.2.4, 6.1.3
diary	把指令窗输入记录为文件	1.4.3
diff	求导数,差分和近似微分	2.1.3, 2.3.1 , 2.3.1, 2.3.3, 4.1.1
digits	控制符号数值的有效数	2.2.2
uigits	字位数	2.2.2
dir	列出目录清单	1.4.3, 1.7.4
dirac	单位冲激函数	2.5.2
disp	显示数值和字符串内容	2.3.3, 2.4.3, 2.6.3, 3.4.1
disttool	概率分布计算交互界面	4.3.1
doc	列出指定工具包中所有	1.9.2, 2.1.6
	函数名	
doc(symengin	ne)	
	引出 MuPAD 帮助浏览器	2.1.6
docsearch	进行多词条检索	1.9.2
double	转化为双精度数值	2.2.1, 2.6.1
drawnow	刷新屏幕	5.4.3
dsolve	求解符号常微分方程	2.4.2, 2.4.3, 2.9.2
E e		
edit	打开 M 文件编辑器	1.4.3, 1.8.1
Ei	指数积分	2.1.6,
eig	矩阵特征值和特征向量	2.1.3, 2.2.4, 2.6.1, 4.2.2
emlBlock	符号表达式转换为	2.9.1
	SIMULINK 模块	
end	数组的最大下标,结束	3.1, 3.2.3, 4.1.1
	for, while, if 语句	
eps	浮点相对误差	1.3.3, 3.3.2, 3.5.1, 4.1.1, 5.4.2
EraseMode	图形对象属性	5.4.3
erfc	补误差函数	2.1.6
error	显示错误信息	6.3.2
evalin	跨空间执行	2.1.3, 2.1.5, 2.1.6, 2.3.1, 2.3.2, 2.8.3, 2.9.1, 2.9.2
exit	关闭 MATLAB	1.4.3
exp	指数	2.1.3, 3.3.1
expand	对指定项展开	2.2.3, 2.7.2
expint	数值指数积分	2.9.3
	かし リカ ナビ チャ	0.1.0.0.6.1

2.1.3, 2.6.1

单位阵 eye 3.2.2, 3.4.2 画等位线简捷指令 2.8.1 ezcontour 画填色等位线简捷指令 2.8.1 ezcontourf ezmesh 画网线图简捷指令 2.8.1 画带等位线的网线图简 ezmeshc 2.8.1 捷指令 画二维曲线简捷指令 ezplot 2.1.3, 2.3.3, 2.4.3, 2.5.1, 2.8.1, 2.8.2 画三维曲线简捷指令 ezplot3 2.8.1 画极坐标曲线简捷指令 2.8.1 ezpolar ezsurf 画曲面图简捷指令 2.1.3, 2.8.1, 2.8.2, 2.8.3 画带等位线的曲面图简 ezsurfc 2.8.1, 2.8.3 捷指令 Ff factor 进行因式或因子分解 2.2.3 factorial n的阶乘 2.5.3 按指定大小创建全0逻 false 3.5.3 辑数组 feather 从 X 轴出发的复数向量图 5.2 feval 函数宏指令 2.9.1, 6.3.2, 6.3.3 fill 多边形填色图 4.3.1, 6.2.4 find 寻找满足条件的数组元 2.5.1, 3.4.1, 5.3.4 素下标 findsym 确认表达式中自由符号 2.1.1, 2.3.1, 2.3.2, 2.4.3 变量 finverse 求反函数 2.8.2 fix 朝零方向取整 3.3.1 flag 红-白-蓝-黑交错色图阵 5.3.3 矩阵的左右翻转 fliplr 3.2.4 矩阵的上下翻转 3.2.4, 5.1.2, 5.4.3 flipud floor 朝负无穷大方向取整 3.3.1 fminbnd 求非线性函数极小值点 4.1.4 单纯形法求多元函数极 4.1.4 fminsearch 小值点 按规定次数重复执行语句 for (end) 2.4.3, 3.3.2, 4.2.1, 6.1.3 format 设置数据输出格式 1.3.3, 1.4.1, 4.1.3, 4.1.4, 4.2.1, 4.4.1, 5.2.4, 6.1.3 fourier Fourier 变换 2.5.1 fsolve 解非线性方程组 4.2.4 函数文件头 4.1.5 function functions 观察函数句柄内涵 6.4.1 fzero 求单变量函数的零点 4.2.4

Gg

gallery 产生测试矩阵 3.2.2, 4.2.3

gca获得当前轴的柄5.2.2gcf获得当前图的柄5.3.3get获得图柄2.4.2, 4.3.1, 5.2.3, 8.1, 8.2

getframe获得影片动画图象的帧5.4.3get_param获取模块参数值7.3

ginput 用鼠标在图上获取数据 4.1.4, 4.2.4, 5.2.4

global 定义全局变量 6.2.3

gradient 梯度 4.1.1, 5.4.2

gray线性灰度色图阵5.3.3grid on/off画坐标网格线5.2.2,5.3.4guide引出 GUIDE 开发环境8.1, 8.2guidata保存用户界面更新数据8.2

H h

heaviside 单位阶跃函数 2.5.1, 2.5.2

help 在线帮助指令 1.9.2, 2.1.6, 6.2.4, 6.3.1, 6.3.2, 6.4.1

helpbrowser引出帮助浏览器1.9.3helpdesk引出帮助界面1.9.3helpwin帮助浏览器中显示帮助1.9.2

信息

hidden网线图消隐开关5.3.4hist统计频数直方图5.2, 4.3.2histfit带拟合曲线的统计频数4.3.2

直方图

hold on/off图形的保持2.4.3, 5.2.3horner转换成嵌套形式2.2.3hot黑-红-黄-白交错色图阵5.3.3hsv饱和色图阵5.3.3

Ιi

 i, j
 虚数单位
 1.3.3

 if-end
 条件执行语句
 3.3.2, 5.4.3

 if-else-end
 程序分支控制
 6.1.1, 6.2.4

 ifourier
 Fourier 反变换
 2.5.1

 ilaplace
 Laplace 反变换
 2.1.3, 2.5.2

 有数块模型
 1.2.2.2.2.1

 Itaplace
 Laplace 反受换
 2.1.3, 2.5.2

 imag
 复数虚部
 1.3.3, 3.3.1

 image
 显示图像
 5.4.3

 impulse
 系统冲激响应
 7.1.2

 ind2sub
 据单下标换算出全下标
 3.4.1

inf 或 Inf 无穷大 1.3.3, 3.4.1, 6.1.3

inline 创建内联函数 4.2.4

input 提示键盘输入 6.1.3, 6.1.4

int 计算积分 2.1.3, 2.3.3, 2.5.4, 2.9.3, 3.1

interp1 线性插值 8.2

intmax	可表达的最大正整数	1.3.3
intmin	可表达的最小负整数	1.3.3
int2str	整数转换为字符串	4.3.1, 6.1.3, 附录 A.1
inv	矩阵的逆	2.6.1, 4.2.3, 6.1.3
invhilb	求逆 Hilbert 阵	6.1.3
isa	判断指定变量类别	2.1.4, 3.5.3, 6.4.1
ischar	若是字符串则为真	3.5.3
isempty	若是空矩阵则为真	3.4.2, 3.5.3
isfinite	若是有限数则为真	3.5.3
isglobal	若是全局变量则为真	3.5.3
ishandle	是否图柄	3.5.3
isinf	若是无穷大则为真	3.5.3
isletter	串中是字母则为真	3.5.3
islogical	若是逻辑数则为真	3.5.3
isnan	若为非数则为真	3.4.1, 3.5.3
isnumeric	若是数值则为真	3.4.2, 3.5.3
isolate	Maple的特殊指令	2.3.1
isprime	是否质数	3.5.3
isreal	若是实数矩阵则为真	3.5.3
isspace	串中是空格则为真	3.5.3
iztrans	Z反变换	2.5.3
Јj		
jacobian	Jacobian 矩阵	2.3.1
jet	变异 HSV 色图阵	5.3.3
jordan	Jordan 分解	2.6.1, 4.2.2
K k		
keyboard	键盘获得控制权	6.1.4
kroneckerDel	lta	
	Kronecker 单位脉冲函数	2.5.3
Ll		
laplace	Laplace 变换	2.1.6, 2.5.2, 2.5.4
legend	形成图例说明	2.6.1, 3.1.2, 4.1.2, 5.2.2, 5.2.3, 5.3.1
length	确定数组长度	4.2.2, 4.4.1, 4.4.2
light	灯光控制	2.8.1, 2.8.3, 5.3.4
lighting	设置照明模式	5.3.3
limit	求极限	2.3.1, 4.1.1, 6.1.3
line	创建线对象	5.4.3, 8.1
LineStyle	图形线对象属性-线型	5.2.1
LineWidth	图形属性-线宽	4.1.1,5.2.1,5.4.3
linmod2	从 SIMULINK 模型得到	7.1.2
	系统的状态方程	

linspace	线性等分向量	3.2.1
load	从磁盘调入数据变量	1.7.4
Location	图形对象属性-位置	4.1.1
log	自然对数	3.3.1,3.4.1
log10	常用对数	3.3.1
log2	以2为底的对数	3.3.1
logical	将数值转化为逻辑值	3.5.3, 6.1.3
logspace	对数刻度向量	
lookfor	关键词检索	1.9.2, 6.2.4, 6.4.1
M m		
magic	魔方阵	3.2.2, 4.2.1, 4.2.2, 6.4.1
maple	进入Maple工作空间计算	2.1.3, 2.3.1, 2.8, 2.8.3
Marker	图形对象属性-点形	4.3.1, 5.2.1, 5.4.3
MarkerEdgeC	olor	
	图形对象属性-点边界色彩	5.2.1
MarkerFaceCo	olor	
	图形对象属性-点域色彩	5.2.1
MarkerSize	图形对象属性-点大小	4.1.1, 5.2.1, 5.4.3
material	对象材质	2.8.2, 5.3.3, 5.3.4
matlabFunctio	n	
	符号表达式转化为 M 函数	2.9.1, 2.9.2
max	最大值	4.3.2, 5.4.1, 8.2, 附录 A.1
mean	平均值	3.3.2, 4.3.2
mesh	三维网线图	5.3.2, 5.3.4, 5.4.2
meshgrid	用于三维曲面的分格线	3.3.2, 5.3.2, 5.4.2
	坐标	
mfun	对 MuPAD 中若干经典	
	特殊函数实施数值计算	2.1.3, 2.1.6, 2.9.1, 2.9.3
mfunlist	MuPAD 经典特殊函数列表	
mhelp	查阅Maple中的库函数 2.1.:	3, 2.8, 2.8.2
	及其调用方法	=
min	最小值	4.3.2, 5.4.1, 8.2
minreal	状态方程最小实现	7.1.2
mkdir	创建目录	1.7.4
mod	模数求余	3.3.1, 6.1.3
more	命令窗口分页输出的控	1.4.3
	制开关	
movie	播放影片动画	5.4.3
moviein	影片动画内存初始化	5.4.3
mtaylor	多元 Taylor 级数展开	2.3.1, 2.8.3
N n		

N n

NaN 或 nan 非数 1.3.3, 3.2.3, 3.4.1, 4.1.2, 5.3.4

	云 华·林) 艮·孙 A 华·	122542624
nargin	函数输入量的个数	1.3.3,5.4.3,6.2.4
nargout	函数输出量的个数	1.3.3, 5.4.3, 6.2.4
ndims	数组的维数	3.4.2
norm	矩阵或向量范数	3.3.2, 4.2.1, 4.2.2, 4.2.3
normcdf	正态分布累计概率	4.3.1
normpdf	正态分布概率密度	4.3.1
normrnd	产生正态分布随机数组	4.3.1
notebook	创建或打开 M-book 文件	8.1.1
null	零空间	2.6.1, 4.2.2, 4.2.3
num2str	把数值转换为字符串	4.3.1, 附录 A.1
numden	提取公因式	2.2.3
Οo		
ode45	高阶法解微分方程	4.1.5
ones	全1 数组	3.2.1, 3.2.2, 3.4.2, 4.4.1
optimset	创建/编辑泛函指令的	4.1.4
	控制参数	
orth	值空间	4.2.2
Pр	A M	
pack	合并工作内存中的碎块	1.7.2
pascal	Pascal 矩阵	4.2.1
path	控制 MATLAB 的搜索 路径	1.6.4
pathtool	修改搜索路径	1.6.4
pause	暂停	5.4.3, 6.1.4
pcolor	用颜色反映数据的伪色图	5.4.1
peaks	产生 peaks 图形数据	5.3.3, 5.3.4, 5.4.1
pi	3.1415926535897	1.3.3
piecewise	MuPAD 分段函数	2.3.2, 2.5.2, 2.5.3, 2.8.2, 2.9.3
pie	饼形统计图	5.2
pink	淡粉红色图阵	5.3.3
plot	二维直角坐标曲线图	2.9.2, 3.5.1, 4.2.4, 5.2, 5.2.1
plot3	三维直角坐标曲线图	5.3.1, 5.4.3
plotyy	双纵坐标图	4.3.1, 5.2.3
polar	极坐标曲线图	5.2
poly	特征多项式,由根创建	2.6.1, 4.4.1
	多项式	
poly2sym	将多项式转换为符号多	2.9.1, 4.4.1
	项式	
polyfit	多项式拟合	4.4.2
polyval	求多项式的值	4.4.1
polyvalm	求矩阵多项式的值	4.4.1
pow2	2 的幂	3.3.1

显示有理分式的易读形式 2.3.1, 2.3.3, 2.5.3, 2.7 pretty 光谱色图阵 prism 元素积 prod Qq quad 低阶法数值积分 4.1.3 高阶法数值积分 4.1.3 quadl 退出 MATLAB 1.4.3 quit 二维箭头图: 主用于场 5.2 quiver 强、流向 Rr rand 均匀分布随机数组 3.2.1, 3.2.2, 3.4.1, 3.4.2, 4.2.1, 4.3.2 randn 正态分布随机数组 3.2.2, 4.2.3, 4.3.2 random 产生各种分布随机数组 3.2.2 在指定字符集上产生均 randsrc 3.2.2 布数组 秩 rank 2.6.1 复数实部 1.3.3, 3.3.1, 4.2.2, 4.4.1 real realmax 最大浮点数 1.3.3 realmin 最小正浮点数 1.3.3 解递推方程 2.1.6 rec 求余数 3.3.1 rem repmat 铺放模块数组 3.2.4, 6.1.3 矩阵变维 3.2.4, 4.2.1, 6.1.3 reshape 重启 MuPAD 引擎 reset 2.1.5, 2.2.2 residue 求部分分式表达 4.4.1 2.8.1 重新启动Maple引擎 restart 1.4.3, 6.1.4, 6.2.2 return 返回 roots 求多项式的根 4.4.1 频数扇形图; 主用于统计 5.2 rose 矩阵逆时针旋转90度 3.2.4 rot90 旋转指令 5.4.3 rotate 四舍五入取整 round 3.3.1, 6.1.3 rref 转换为行阶梯形 2.6.1, 4.2.2 $\mathbf{S} \mathbf{s}$ 把内存变量存入磁盘 save 1.7.4 正割 2.1.3, 3.3.1 sec 双曲正割 sech 2.1.3, 3.3.1 设置图形对象属性 set 2.4.3, 2.5.1, 4.3.1, 5.2.2, 5.2.3, 5.3.3, 5.4.3, 8.2

2.3.1, 3.3.1

2.9.1, 5.3.3

5.1.2, 6.2.4, 6.3.2, 8.2.3

图形渲染模式

函数符号,符号函数

显示图形窗

shading

shg

sign

simple 运用各种指令化简符号 2.2.3, 2.3.2, 2.5.1, 2.5.4

表达式

simplify 恒等式简化 2.2.3 simulink 打开 SIMULINK 集成环境 7.1.1

sin 正弦 2.1.3, 2.3.1, 3.3.1, 3.4.1

sinh 双曲正弦 2.1.3, 3.3.1

size 确定数组大小 2.1.4, 3.3.2, 3.4.1, 3.4.2, 4.2.1

slice 切片图 5.4.2

solve 求解代数方程组 2.1.3, 2.2.4, 2.6.3, 4.2.4

sort 对数组或向量中元素排序 2.7

sphere 产生球面数据 5.3.3, 5.3.4

 spinmap
 颜色周期性变化操纵
 5.4.3

 spring
 青、黄浓淡色图阵
 5.3.3

sqrt平方根2.3.3, 3.3.1square轴属性为方型1.3.3, 6.2.4ss产生状态方程 LTI 对象7.1.2

 stairs
 阶梯形曲线图
 5.2, 5.2.3

 state
 用于设置 randn 随机数
 3.2.2, 4.2.3, 4.3.2

发生器状态的关键词

std 标准差 4.3.2

stem 杆图 4.4.3, 5.2, 5.2.3

三维离散杆图 5.3.2 stem3 计算阶跃响应 8.1 step str2double 把字符串转换为双精度数 8.1 str2func 创建函数句柄 6.4.1 比较字符串 6.1.2 strcmp String 图形对象属性-字符串 4.3.1

subexpr 运用符号变量置换子表 2.2.4, 2.6.1

达式

subplot创建子图2.8.1, 3.5.1, 3.5.2, 4.4.3, 5.2.2, 5.2.3subs通用置换指令2.2.4,2.3.1,2.3.3,2.5.1,2.5.4, 2.7.2, 2.8.2

sum 元素和 4.1.2,6.1.3

summer 绿、黄浓淡色图阵 5.4.3

surf 三维表面图 5.3.2,5.3.4,5.4.2

 surfc
 带等高线的三维表面图
 5.3.4

 surface
 绘制曲面的底层指令
 8.2

 svd
 奇异值分解
 2.6.1

switch-case 多个条件分支 6.1.2, 6.2.4, 6.3.2

sym产生符号对象2.1.1, 2.1.5, 2.3.1, 2.2.1, 2.2.2, 2.3.3, 2.5.2, 2.9.1symengineMuPAD 符号计算引擎2.1.5, 2.1.6, 2.2.2, 2.3.1, 2.3.2, 2.5.3, 2.8.3

syms 定义基本符号对象 2.1.1, 2.1.5, 2.3.1, 2.5.1

symsum 符号序列的求和 2.3.2, 6.1.3

sym2poly 符号多项式转换为

数值多项式系数 2.9.1

T t		
tan	正切	2.1.3, 3.3.1
tanh	双曲正切	2.1.3, 3.3.1
taylor	Taylor 级数	2.3.1
text	图形上文字标注	2.4.3, 4.3.1, 4.4.3, 5.2.2
tf	产生传递函数 LTI 对象	7.1.2, 8.1
tfdata	从对象中提取传递函数	7.1.2
	分子分母多项式系数	
tic	秒表起动	4.2.3, 6.1.3
title	图形名	2.4.3, 3.5.1, 5.2.2
toc	秒表终止和显示	4.2.3, 6.1.3
trace	迹	4.2.1
trapz	梯形数值积分	4.1.2
tril	下三角分解	2.6.1
triu	上三角分解	2.6.1
true	按指定大小创建全1逻	3.5.3
	辑数组	
triplequad	三重(闭型)数值积分	4.1.3
	指令	
twister	用于设置 rand 指令随机	3.2.1, 3.4.1, 4.2.1, 4.3.2, A.1
	数发生器状态的关键词	
type	显示文件内容	1.4.3
V v		
var	求方差	4.3.2
version	MATLAB 版本	
vectorize	字符串表达式向量化	2.9.1
view	设定 3-D 图形观测点	2.8.2, 5.3.2, 5.3.3
vpa	给出数值型符号结果	2.2.2, 2.3.3, 4.1.4
W w		
which	确定指定文件所在的目录	1.4.3, 3.4.2, 6.4.2
while end	不确定次数重复执行语句	3.3.2, 6.1.3
whitebg	图形底色控制	8.2.3
who	列出工作内存中的变量名	1.7.2, 1.7.4
whos	列出工作内存中的变量	1.7.2, 2.1.4
	细节	7 · ·
winter	蓝、绿浓淡色图阵	5.3.3
Хх		
xlabel	X轴名标注	3.5.1, 5.2.2, 5.3.1, 5.3.2
	□ n	

xor

异或

3.5.2

Y y

Ycolor 图形对象属性-纵轴颜色 4.3.1

ylabel Y 轴名标注 3.5.1, 4.3.1, 5.2.2, 5.3.1, 5.3.2

 $\mathbf{Z} \mathbf{z}$

zeros 全零矩阵 3.2.2, 3.4.2, 4.2.1, 5.3.4, 6.1.3

 zlabel
 Z 轴名标注
 5.3.1, 5.3.2

 zoom
 二维图形的变焦放大
 4.2.4

ztrans Z 变换 2.5.3

C.3 Simulink模块

Add	求和模块	7.1.1,7.1.2
Breaker	开关	7.3
Current Measurement	电流测量器	7.3
Dc Voltage Source	直流电压源	7.3
Digital Filter Design	数字滤波器设计模块	7.2
Discrete Filter	<u> 离散滤波器模块</u>	7.1.2
Gain	增益模块	7.1.1,7.1.2
In1	输入端口模块	7.1.2
Integrator	连续函数积分	7.1.1
Out1	输出端口模块	7.1.2
Parallel RLC Branch	RLC 并联支路	7.3
Powergui	营造 SimPowerSystems	7.3
	仿真环境	
Random Number	随机数模块	7.2
Random Source	随机信号模块	7.1.2
Scope	示波模块	7.1.1,7.1.2
Series RLC Branch	RLC 串联支路	7.3
Simulink	SIMULINK 基本库	7.1.1
Sine Wave	正弦波输出	7.1.2
Transfer Fcn	传递函数模块	7.1.2
Voltage Measurement	电压测量器	7.3

参考文献

- [1] 张志涌等,MATLAB与仿真(2010a),北京航空航天大学出版社,2010(即将出版)。
- [2] MathWorks, MATLAB R2010a, 2010.