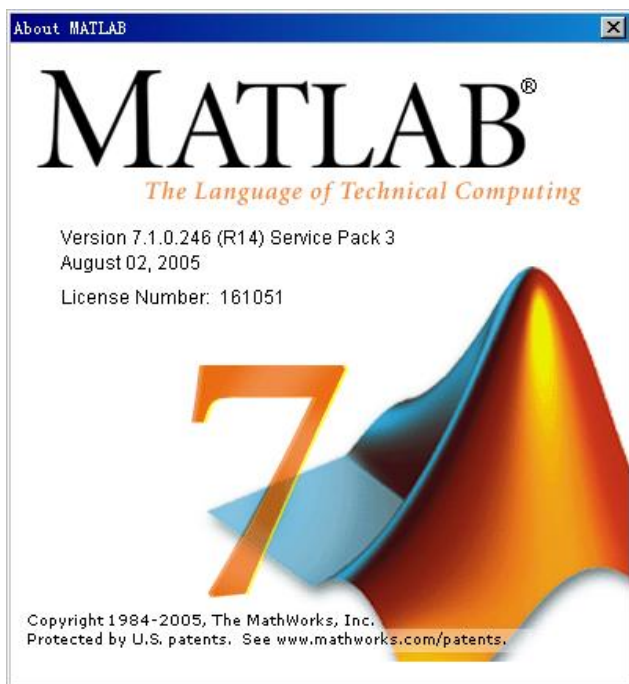


MATLAB 语言及其应用



授课教师: 刘晨晨

单位: 信息工程系

email: lcczxm@126.com

Application of Matlab Language

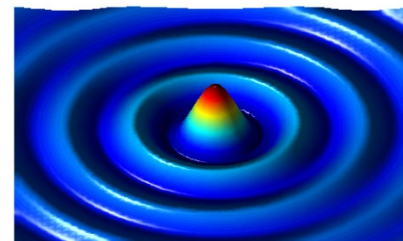


- 课程的作用

- 课程的目的

- 课程的特点

- 课程安排





课程的作用

- 在欧美各高等学校，Matlab成为线性代数、自动控制理论、数字信号处理、时间序列分析、动态系统仿真、图像处理等诸多课程的基本教学工具，成为本科生、硕士生和博士生的必须掌握的基本技能。
- 在设计研究单位和工业部门，Matlab已被广泛地用于研究和解决各种具体的工程问题。
- 可以预见，Matlab将在我国科学研究和工程应用中发挥越来越大的作用。



本课程的目的 (Objectives of This Course)

讲授MATLAB语言基础入门知识，介绍MATLAB产品的体系、MATLAB桌面工具的使用方法，重点介绍MATLAB的数据可视化、数值计算的基本步骤以及如何使用MATLAB语言编写整洁、高效、规范的程序。并涉及到一些具体的专业应用工具箱的简单应用（如：信号处理工具箱、图像处理工具箱等）。

通过本课程的学习，了解、熟悉、掌握 MATLAB的基本编程方法，并具有初步的利用计算机处理、解决实际问题的能力，为进一步学习后续的专业课程做好准备。

本课程的特点 (Features of This Course)

- 交叉性课程，是计算机技术、数学理论知识以及工程理论知识的综合。
- 实践性课程。
- 内容多，课时少，要求同学上课认真听讲，要充分利用上机实践消化、理解、掌握课上讲解内容。



课程安排

- 课堂教学：共16学时；（1-8周）
- 上机试验：共16学时。（5-8周）

□ 学习成绩：

- 1) 实验成绩占20%；
- 2) 考勤 20%；
- 3) 考试60%（另有安排）。

主要参考书

- 《精通MATLAB 6.5》 张志涌 等编著，北航出版，2003年
- 《高等应用数学问题的Matlab求解》 薛定宇等著，清华大学出版社，2004年
- 《Matlab程序设计与应用》 刘卫国主编，高等教育出版社

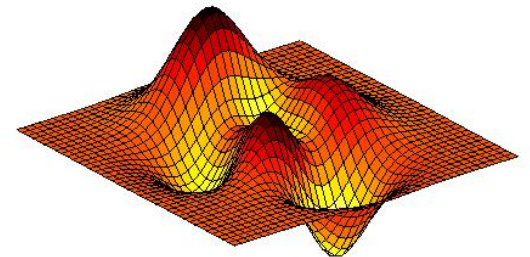


授课宗旨

- 讲授MATLAB的通用功能。
- 寓教于例，由浅入深。
- 关于科学计算，着重强调理论概念、算法和实际计算三者之间的关系。

第一讲 Matlab概述

- 前言
- Matlab软件概述
- Matlab的桌面环境及入门知识





1 Matlab概述

■ 内容

- Matlab发展历史
- Matlab产品家族(Matlab family of products)体系
- Matlab 语言的特点。

■ 目的

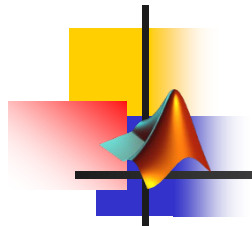
- 全面了解 Matlab软件包
- 激发对Matlab软件的学习兴趣。

1.1 MATLAB的历史及影响

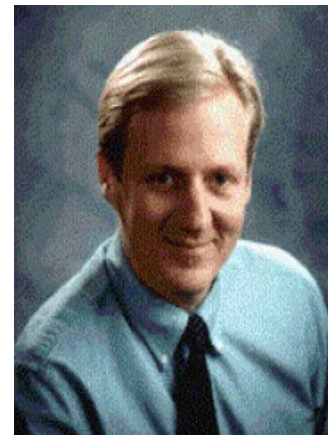
- 70年代中期，Cleve Moler博士及其同事在美国国家基金会的帮助下，开发了LINPACK和EISPACK的FORTRAN语言子程序库，这两个程序库代表了当时矩阵运算的最高水平。
- 到了70年代后期，身为美国新墨西哥州大学计算机系系主任的Cleve Moler，在给学生上线性代数课时，为了让学生能使用这两个子程序库，同时又不用在编程上花费过多的时间，开始着手用FORTRAN语言为学生编写使用LINPACK和EISPACK的接口程序，他将这个程序取名为MATLAB，其名称是由MATrix和LABoratory（矩阵实验室）两个单词的前三个字母所合成。
- 在1978年，Malab就面世了。这个程序获得了很大的成功，受到了学生的广泛欢迎。在以后的几年里，Matlab在多所大学里作为教学辅助软件使用，并作为面向大众的免费软件广为流传。



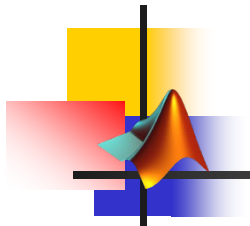
Cleve Moler



- 将MATLAB商品化的不是Cleve Moler,而是一个名叫Jack Little的人。当免费的MATLAB软件到Stanford大学, Jack Little正在该校主修控制, 便接触到了当时MATLAB, 直觉告诉他, 这是一个具有巨大发展潜力的软件。因此他在毕业没多久, 就开始用C语言重新编写了MATLAB的核心。在Moler的协助下, 于1984年成立MathWorks公司, 首次推出MATLAB商用版。在其商用版推出的初期, MATLAB就以其优秀的品质(高效的数据计算能力和开放的体系结构) 占据了大部分数学计算软件的市场, 原来应用于控制领域里的一些封闭式数学计算软件包(如英国的UMIST、瑞典的LUND和SIMNON、德国的KEDDC) 就纷纷被淘汰或在MATLAB上重建。



Jack Little



MATLAB经过了30多年的专门打造、20多年的千锤百炼，它以高性能的**数组运算（包括矩阵运算）**为基础，不仅实现了大多数数学算法的**高效运行函数和数据可视化**，而且提供了非常高效的**计算机高级编程语言**，在用户可参与的情况下，**各种专业领域的工具箱**不断开发和完善，MATLAB取得了巨大的成功，已广泛应用于**科学研究、工程应用，用于数值计算分析、系统建模与仿真**。

早在**20世纪90年代初**，欧美等发达国家的大学就将MATLAB列为一种必须掌握的编程语言。近几年来，国内的很多大学也将MATLAB列为了本科生必修课程。

与Maple、Mathematica数学计算软件相比，MATLAB以数值计算见长，而Maple等以符号运算见长，能给出解析解和任意精度解，而处理大量数据的能力远不如MATLAB。

MATLAB软件功能之强大、应用之广泛，已成为**21世纪最为重要的科学计算语言**。可见学习掌握这一工具的重要性。

1.2 MATLAB产品的体系结构

实际上MATLAB
习使用MATLAB呢？

MATLAB产

➤ MATLAB

➤ MATLAB Real-Time

➤ MATLAB

➤ Simulink

➤ Simulink E

➤ Real-Time W

➤ Stateflow

➤ Stateflow Coder

MATLAB是MATLAB产品家族的计算核心与基础，是集高性能数值计算与数据可视化于一体的高效编程语言。

Simulink是MATLAB产品家族的专用工具箱，如Communication Blockset、DSP Blockset、SimPowerSystem Blockset、Signal Processing Blockset等，详见MATLAB在线帮助文档。

Stateflow是有限状态机理论针对复杂的事件驱动系统进行建模、仿真的工具。

Stateflow Coder是基于Stateflow状态图生成高效、优化的程序代码。

围绕着MATLAB这个计算核心，形成了诸多针对不同应用领域的专用工具箱

的列表以及每个工具箱的使用文档。MATLAB本身所提供的包，其他公司或研究单位的总数已有100多个，如果你有特别的应用有相关的工具箱，

据代码生

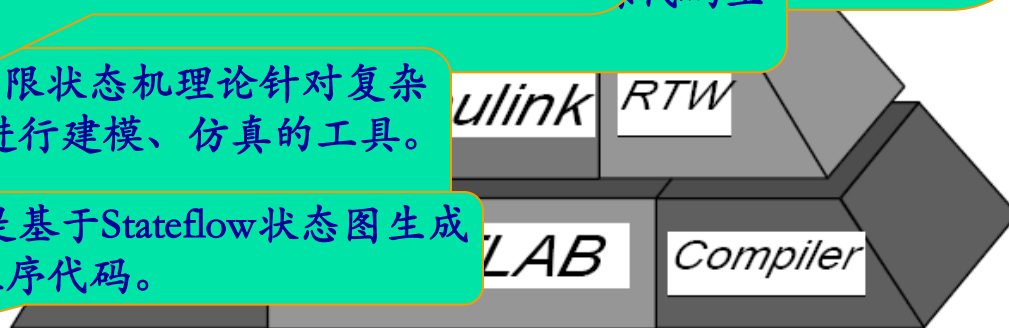


图1.1 MATLAB产品体系结构

由这些模块产品之间的关系可以图1.1表示。



1.3 MATLAB编程语言的特点

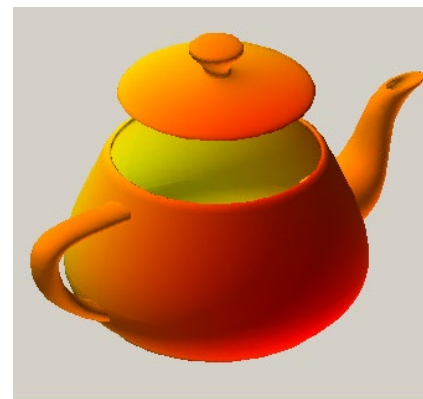
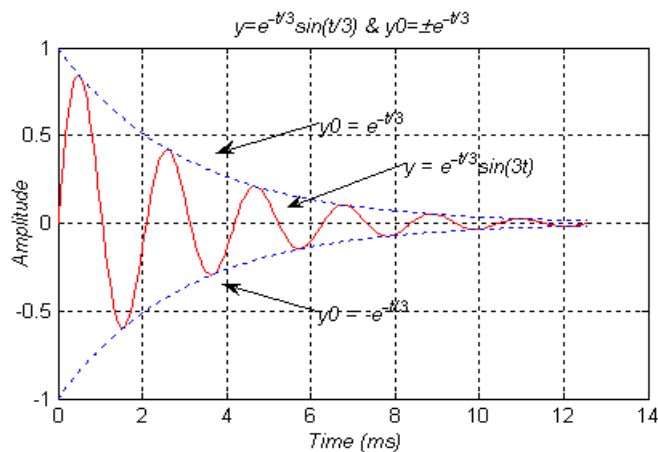
MATLAB语言主要有以下几个特点：

- **语法规则简单**。尤其内定的编程规则，与其他编程语言（如C、Fortran等）相比更接近于常规数学表示。对于数组变量的使用，不需类型声明，无需事先申请内存空间。
- **MATLAB基本的语言环境提供了数以千计的计算函数**，极大的提高了用户的编程效率。如，一个fft函数即可完成对指定数据的快速傅里叶变换，这一任务如果用C语言来编程实现的话，至少要用几十条C语言才能完成。
- **MATLAB是一种脚本式（scripted）的解释型语言**，无论是命令、函数或变量，只要在命令窗口的提示符下键入，并“回车（Enter）”，MATLAB都予以解释执行。
- **平台无关性（可移植性）**。MATLAB软件可以运行在很多不同的计算机系统平台上，如Windows Me/NT/2000/XP、很多不同版本的UNIX以及Linux。无论你在哪一个平台上编写的程序都可以运行在其它平台上，对于MATLAB数据文件也一样，是平台无关的。极大保护了用户的劳动、方便了用户。其绘图功能也是平台无关的。无论任何系统平台，只要MATLAB能够运行，其图形功能命令就能正常运行。

因此，MATLAB是一个简单易用、功能强大的高效编程语言。

■ 功能强大

- 数值运算优势
- 符号运算优势(Maple)
- 强大的2D、3D数据可视化功能
- 许多具有算法自适应能力的功能函数





■ 语言简单、内涵丰富

- 语言及其书写形式非常接近于常规数学书写形式；
- 其操作和功能函数指令就是常用的计算机和数学书上的一些简单英文单词表达的，如：help、clear等；
- 完备的帮助系统，易学易用。

■ 扩充能力、可开发能力较强

- MATLAB完全成了一个开放的系统
- 用户可以开发自己的工具箱
- 可以方便地与Fortran、C等语言接口

■ 编程易、效率高

- Matlab以数组为基本计算单元
- 具有大量的算法优化的功能函数



2 MATLAB的桌面环境及入门知识

- 启动与退出MATLAB
- 命令窗口及使用
- 数值表示、变量、表达式
- 命令历史窗口
- 工作空间
- 获取在线帮助



2.1 启动与退出MATLAB

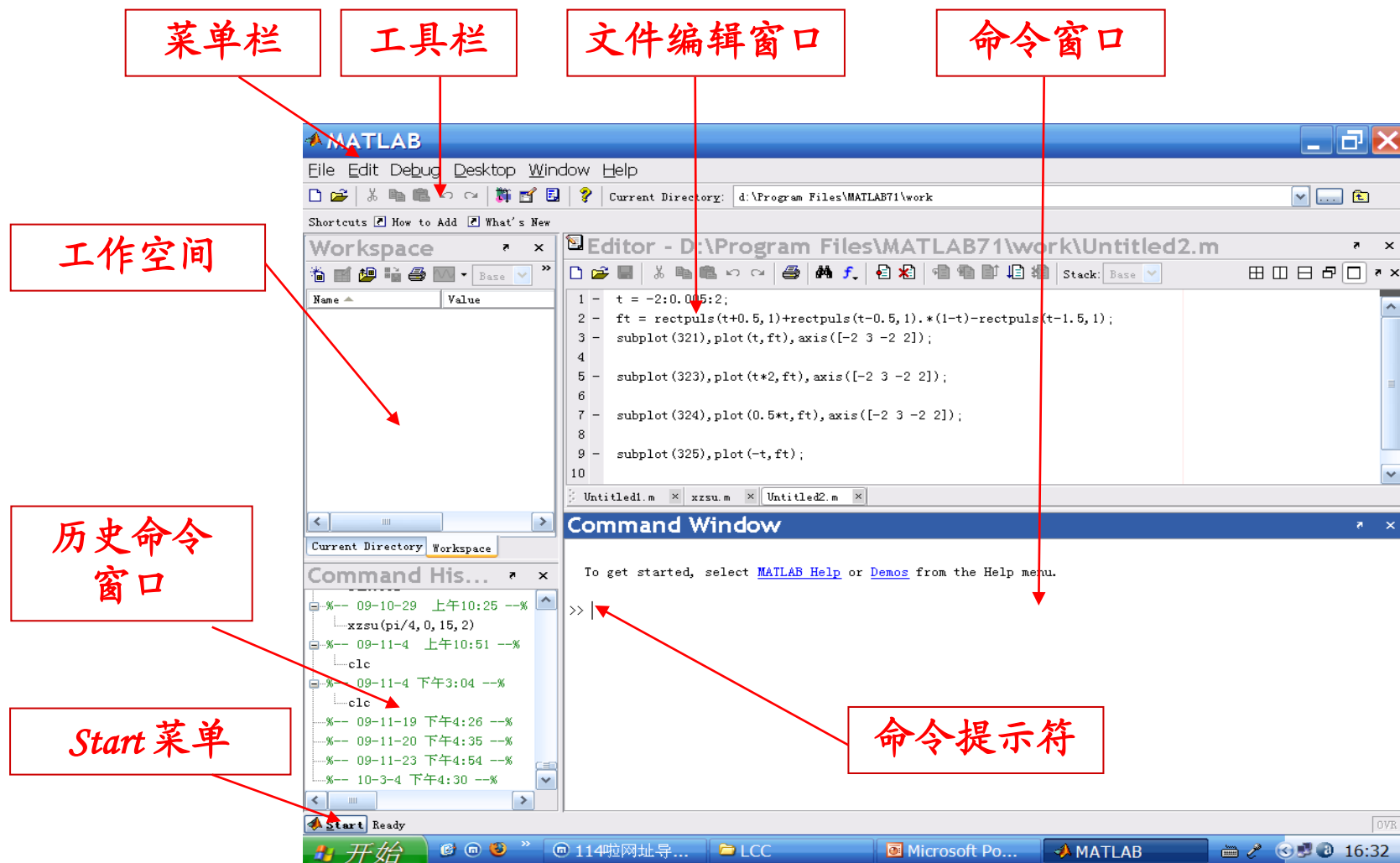
■ 启动MATLAB

- 直接用鼠标双击桌面上MATLAB7.1图标
- 或Windows桌面的“开始”——> “所有程序”——> “MATLAB7.1”——> “MATLAB7.1”。

■ 退出MATLAB

- 关闭MATLAB桌面
- 在命令窗口执行quit或exit命令
- MATLAB缺省桌面（见下页）

2.1 启动与退出MATLAB (续)



2.2 命令窗口的使用

- ⊕ 激活命令窗口。
- ⊕ “>>” 与闪烁的光标一起表明系统就绪，等待输入。
- ⊕ 命令窗口脱离MATLAB桌面。

■ 简单计算

【例2.2-1】 计算 $[12 + 2 \times (7 - 4)] \div 3^2$

(1) 在MATLAB命令窗口输入
以下内容：

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

(2) 按【Enter】键，指令执行。

(3) 返回的计算结果：

`ans=`

`2`

A screenshot of the MATLAB Command Window. The title bar reads "Command Window". The input line shows the command ">> (12+2*(7-4))/3^2". Below the input, the output is displayed: "ans =" followed by "2" on the next line. The prompt ">>" is visible at the bottom of the window.

```
Command Window
>> (12+2*(7-4))/3^2

ans =

     2

>>
```



2.2 命令窗口（续）

〔说明〕

- 在命令窗口【Enter】键提交命令执行。
- Matlab所用运算符（如+、-、^等）是各种计算程序中常见的。
- 计算结果中的“ans”是英文“answer”的一种缩写，其含义就是“运算答案”。ans是Matlab的一个预定义变量。

2.2 命令窗口（续）

■ 简单计算（续）

【例2.2-2】 计算 $\sin(45^\circ)$

```
>>sin(45*pi/180)
```

```
ans=
```

```
0.7071
```

- **Matalb**中正弦函数**sin**就是常见的正弦函数。
- 它的参数值是以“弧度”为单位的。
- **pi**也是**Matalb**的预定义变量。
- **pi=3.14159...**
- **Matlab**对字母大小写是敏感的。

【例2.2-3】 计算 $\left(\sqrt{2e^{x+0.5}}+1\right)$ 的值，

```
>>sqrt(2*exp(4.92+0.5)+1)
```

```
ans=
```

```
21.2781
```

- **Matalb**中开平方—**sqrt(x)**，是英文**square root**的缩写。
- **Matalb**中指数函数**exp(x)**，常见的表达方式。

2.2 命令窗口（续）

 “clc”清除窗口显示内容的命令。

【例2.2-4】 计算 $y = \frac{2\sin(0.3\pi)}{1+\sqrt{5}}$ 的值。

```
>>y=2*sin(0.3*pi)/(1+sqrt(5))
```

```
y=  
0.5000
```

【例2.2-5】 计算 $y = \frac{2\cos(0.3\pi)}{1+\sqrt{5}}$ 的值。

```
>>y=2*cos(0.3*pi)/(1+sqrt(5))
```

```
y=  
0.3633
```

 命令行编辑

- “↑” 键调回已输入过命令。
- 修改。

2.2 命令窗口（续）

【例2.2-5】 计算半径为5.2m的圆的周长和面积。

```
>>radius=5.2; %圆的半径
```

```
>>area=pi*5.2^2, circle_len=2*pi*5.2
```

```
area =
```

```
84.9487
```

```
circle_len =
```

```
32.6726
```

- 以上两例，命令行中用到了等号“=”。
- 计算结果不再赋给“**ans**”，而是赋给用户指定的变量**y**、**area**、**circle_len**。
- 无论是预定义变量还是用户自定义变量都被存储在系统的工作空间内，即系统定义的一个存储窗口变量的内存空间。
- **Who**、**whos**命令用来显示工作空间的变量
- **clear**命令用来清除工作空间的变量。



2.2 命令窗口 (续)

>>who

Your variables are:

ans circle_len y

area radius

>>whos

Name	Size	Bytes	Class
ans	1x1	8	double array
area	1x1	8	double array
circle_len	1x1	8	double array
radius	1x1	8	double array
y	1x1	8	double array

Grand total is 5 elements using 40 bytes



2.2 命令窗口 (续)

```
>>clear y
```

```
>>who
```

Your variables are:

ans circle_len

area radius

```
>>clear ans area
```

```
>>whos
```

Your variables are:

Name	Size	Bytes	Class
circle_len	1x1	8	double array
radius	1x1	8	double array

Grand total is 2 elements using 16 bytes

2.2 命令窗口（续）

■ 数值显示格式设置

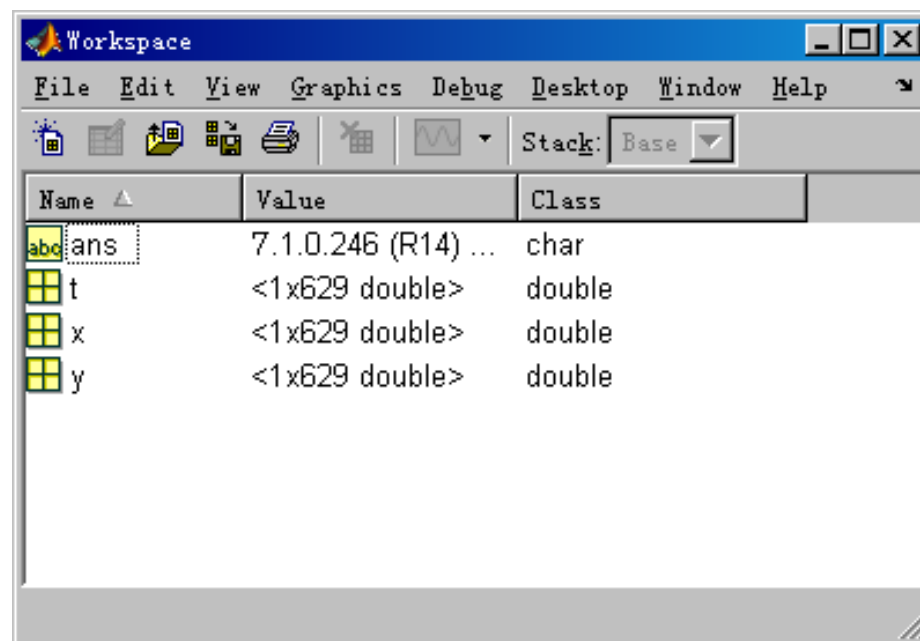
- 缺省显示格式：简洁的短（short g）格式
- 窗口命令及语法格式：format 显示格式关键字
如：format long %15位数字显示

■ 常见通用命令

命令	含义
clc	清除命令窗口的显示内容
clear	清除Matlab工作空间中保存的变量
who或whos	显示Matlab工作空间中的变量信息
dir	显示当前工作目录的文件和子目录清单
cd	显示或设置当前工作目录
type	显示指定m文件的内容
help或doc	获取在线帮助
quit或exit	关闭/退出MATLAB

2.3 工作空间

- 查看工作空间内存变量，可以由who、whos。
- 命名新变量。
- 修改变量名
- 删除变量
- 绘图
- 保存变量数据
- 装入数据





2.4 历史窗口

■ 历史窗口：

- 首先记录每次启动时间

- 并记录在命令窗口输入命令，此次运行期间，输入的所有命令被记录为一组，并以此次启动时间为标志。

■ 使用历史窗口：

- 可以查看命令窗口输入过的命令或语句

- 可以选择一条或多条命令执行拷贝、执行、创建M文件等。

要清除历史记录，可以选择Edit菜单中的Clear Command History 命令



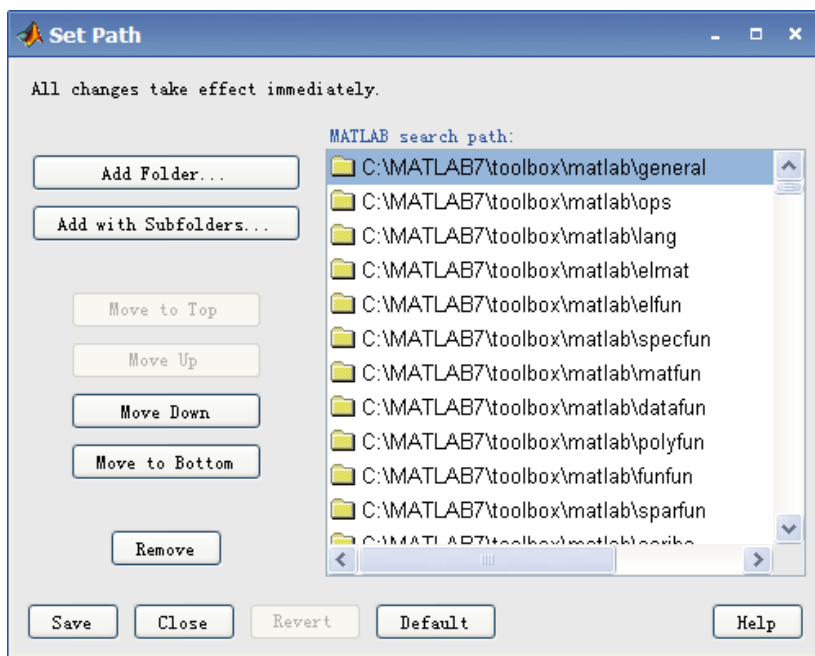
2.5 当前目录窗口和搜索路径

当前目录窗口：指Matlab运行时的工作目录。

- 只有在当前目录和搜索路径下的文件、函数才可以被运行和调用。
- 如果没有特殊指明，数据文件也将存放在当前目录下；
- 用户可以将自己的工作目录设置成当前目录，从而使所有操作都在当前目录中进行。

搜索路径：指Matlab执行过程中对变量、函数和文件进行搜索的路径。

- 在File菜单中选择Set Path命令或在命令窗口输入pathtool命令，出现搜索路径设置对话框：



！修改完搜索路径后，需要进行保存。



2.6 获取在线帮助

- MATLAB提供的帮助信息有两类
 - 简单纯文本帮助信息
 - **help**
 - **lookfor**（条件比较宽松）例：**inverse**
 - 窗口式综合帮助信息（文字、公式、图形）
 - **doc**
 - **helpwin**



【功能演示-1】

求方程 $2x^5 - 3x^3 + 71x^2 - 9x + 13 = 0$ 的全部根。

p = [2,0,-3,71,-9,13]; % 建立多项式系数向量

x = roots(p); 求根

x =

-3.4914

1.6863 + 2.6947i

1.6863 - 2.6947i

0.0594 + 0.4251i

0.0594 - 0.4251i

【功能演示-2】 求解线性方程组

$$\begin{cases} 2x + 3y - z = 2 \\ 8x + 2y + 3z = 4 \\ 45x + 3y + 9z = 23 \end{cases}$$

a = [2,3,-1;8,2,3;45,3,9]; %建立系数矩阵a

b = [2;4;23]; %建立列向量b

x = inv(a)*b

x =

0.5531

0.2051

-0.2784



符号计算

syms x y z %建立符号变量

[x,y,z]=solve(2*x+3*y-z-2,8*x+2*y+3*z-4,45*x+3*y+9*z-23)

x =

151/273

y =

8/39

z =

-76/273



【功能演示-3】 求解定积分

求解 $I = \int_0^1 x \ln(1+x) dx$

quad('x.*log(1+x)',0,1)

ans =

0.250

或

syms x

int(x*log(1+x),0,1)

ans =

1/4

【功能演示-4】 多项式曲线拟合

考虑如下 x-y 一组实验数据：

$x=[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$

$y=[1.2, 3, 4, 4, 5, 4.7, 5, 5.2, 6, 7.2]$

注： $y(x) = x^3 - 2x^2 - 5$ In MATLAB $y=[1 \quad -2 \quad 0 \quad -5]$

■ 一次多项式拟合：

$p1 = \text{polyfit}(x,y,1)$

■ 三次多项式拟合：

$p3 = \text{polyfit}(x,y,3)$

■ plot 原始数据、一次拟合曲线和三次拟合曲线

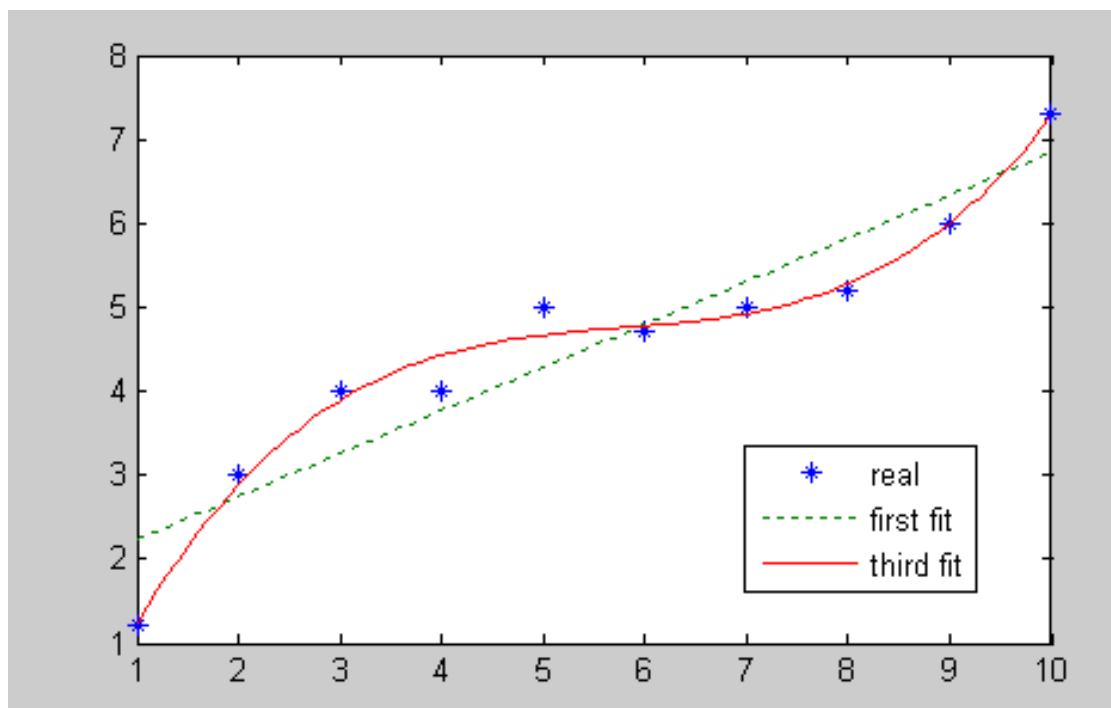
$x2=1:0.1:10;$

$y1=\text{polyval}(p1,x2)$

$y3=\text{polyval}(p3,x2)$

$\text{plot}(x, y, '*', x2, y1, ':', x2, y3)$

拟合曲线图



由图可见，三次拟合结果较好。

2.3 数值表示、变量及表达式

■ 数值的记述

Matlab的数只采用习惯的十进制表示，可以带小数点和负号;其缺省的数据类型为双精度浮点型（double）。

例如：3 -10 0.001 1.3e10 1.256e-6

■ 变量命令规则

- 变量名、函数名对字母的大小写是敏感的。如 **myVar**与**myvar**表示两个不同的变量。
- 变量名第一个字母必须是英文字母。
- 变量名可以包含英文字母、下划线和数字。
- 变量名不能包含空格、标点。
- 变量名最多可包含63个字符（6.5及以后的版本）。

2.3 数值表示、变量及表达式（续）

■ Matlab预定义的变量

变量名	意义
ans	最近的计算结果的变量名
eps	MATLAB定义的正的极小值=2.2204e-16
pi	圆周率 π
inf	∞ 值，无限大
i或j	虚数单元， sqrt(-1)
NaN	非数，0/0、 ∞/∞

[[说明]]

- 每当MATLAB启动完成，这些变量就被产生。
- MATLAB中，被0除不会引起程序中断，给出报警的同时用inf或NaN给出结果。
- 用户只能临时覆盖这些预定义变量的值，Clear或重启MATLAB可恢复其值。

2.3 数值表示、变量及表达式（续）

■ 运算符和表达式

运算	数学表达式	MATLAB运算符	MATLAB表达式
加	$a+b$	+	$a+b$
减	$a-b$	-	$a-b$
乘	$a \times b$	*	$a*b$
除	a/b 或 $a \setminus b$	/或\	a/b 或 $a \setminus b$
幂	a^b	^	a^b

【说明】

- Matlab用“\”和“/”分别表示“左除”和“右除”。对标量而言，两者没有区别。对矩阵产生不同影响。
- MATLAB表达式的书写规则与“手写方式”几乎完全相同。
- 表达式按与常规相同的优先级自左至右执行运算。
- 优先级：指数运算级别最高，乘除次之，加减最低。
- 括号改变运算的次序。

2.3 数值表示、变量及表达式（续）

■ 复数及其运算

- MATLAB中复数的表达： $z=a+bi$ ，其中a、b为实数。
- MATLAB把复数作为一个整体，象计算实数一样计算复数。

【例2.3-1】复数 $z_1=3+4i$ ， $z_2=1+2i$ ， $z_3=2e^{\frac{\pi}{6}i}$

计算 $z = \frac{z_1 z_2}{z_3}$

```
>>z1=3+4*i, z2=1+2*i, z3=2*exp(i*pi/6), z=z1*z2/z3
```

```
>>z_real=real(z), z_image=imag(z),
```

```
>>z_angle=angle(z), z_length=abs(z),
```



课堂总结

- 对MATLAB整个软件产品进行了概述
 - 历史发展
 - 软件产品家族体系构成
 - M语言的特点
- MATLAB的入门知识
 - MATLAB桌面
 - MATLAB数值的记述和数据显示格式
 - 变量及其命名规则
 - 运算符及表达式
 - 常用的MATLAB命令
 - 在线帮助的使用



思考题

1. 简述MATLAB软件的功能及特点。
2. 简述MATLAB变量的命名规则。
3. 熟悉课件中的例子。