

# 第 1 章 MATLAB 语言概述

## 1.1 MATLAB 语言及特点

MATLAB 是“MATrix LABoratory”的缩写（矩阵实验室），它是由美国 MathWorks 公司于 1984 年正式推出的一种科学计算软件。1988 年推出了 3.x（DOS）版本，1992 年推出了 4.x（Windows）版本，1997 年推出 5.1（Windows）版本，然后就是 6.0 版本和 7.0 版本。随着新版本的推出，MATLAB 的扩展函数越来越多，功能越来越强大。

MATLAB 语言是一种以矩阵运算为基础的交互式程序语言。它集成度高，使用方便，输入简捷，运算高效，内容丰富，并且很容易由用户自行扩展。与其他计算机语言相比，MATLAB 具有以下显著特点。

### （1）MATLAB 是一种解释性语言

MATLAB 以解释方式工作，输入算式立即得出结果，无须编译，对每条语句解释后立即执行。若有错误也立即做出反应，便于编程者马上改正。这些都大大减轻了编程和调试的工作量。

### （2）变量的“多功能性”

- 每个变量代表一个矩阵，它可以有  $n \times m$  个元素；
- 每个元素都被看作复数，这个特点在其他语言中也是不多见的；
- 矩阵的行数、列数无须定义，MATLAB 会根据用户输入的数据形式，自动决定一个矩阵的阶数，而在用其他语言编程时必须定义矩阵的阶数。

### （3）运算符的“多功能性”

所有的运算，包括加、减、乘、除、函数运算都对矩阵和复数有效。

### （4）语言规则与笔算式相似

MATLAB 的程序与科技人员的书写习惯相近，因此易写易读，易于在科技人员之间交流。

### （5）强大而简易的作图功能

- 能根据输入数据自动确定坐标绘图；
- 能规定多种坐标（极坐标、对数坐标等）绘图；
- 能绘制三维坐标中的曲线和曲面；
- 可设置不同颜色、线型、视角等。

如果数据齐全，往往只需一条命令即可给出相应的图形。

### （6）智能化程度高

- 绘图时自动选择最佳坐标，以及按输入或输出变元数自动选择算法等；
- 做数值积分时自动按精度选择步长；
- 自动检测和显示程序错误的能力强，易于调试。

### （7）功能丰富，可扩展性强

MATLAB 软件包括基本部分和专业扩展部分。基本部分包括：矩阵的运算和各种变换，代数和超越方程的求解，数据处理和傅里叶变换及数值积分等，可以满足大学理工科计算的需要。扩展部分称为工具箱（toolbox）。它实际上是用 MATLAB 的基本语句编成的各种子程序集，用于解

决某一个方面的专门问题，或某一领域的新算法。现在已经有控制系统、信号处理、图像处理、系统辨识、模糊集合、神经网络及小波分析等 20 余个工具箱，并且还在继续发展中。

MATLAB 由于其强大的功能，在欧美等国家的一些大学里，MATLAB 已经成为诸如数字信号处理、自动控制理论等课程的主要工具软件，同时也是理工科本科生、研究生必须掌握的一项基本技能。近年来，随着我国教育事业的不断发展及与国外著名高校的接轨，许多高校都开设了这门课程，MATLAB 这一功能强大的软件逐渐被越来越多的人所了解和使用。

为了帮助理工科本科生、研究生更好地学习和掌握 MATLAB，在本书中重点讲解 MATLAB 基本部分，对于工具箱的应用，则重点介绍在信号处理、自动控制、通信和电子电路仿真等四个方面的应用，涉及的课程有高等数学、信号与系统、数字信号处理、自动控制原理、数字通信等。

## 1.2 MATLAB 的工作环境

无论 MATLAB 3.x 之前的 DOS 版本，还是 MATLAB 4.x 以后的 Windows 版本，MATLAB 的一切操作都必须在 MATLAB 系统中进行；即要使用 MATLAB 语言，首先必须安装 MATLAB 系统，只有在启动 MATLAB 系统之后，方可进行操作。不同版本的 MATLAB 要安装在不同的操作系统下。MATLAB 3.x 之前的版本使用 DOS 操作系统，而 MATLAB 4.0 以后的版本都以 Windows 操作系统为基础。下面我们着重介绍在 Windows 操作系统下，MATLAB 系统的安装。

### 1.2.1 MATLAB 系统的安装

MATLAB 系统的安装非常简单，只要按照安装程序步骤和提示，根据具体需要一步一步地进行下去即可。下面以 MATLAB 6.1 为例简单地介绍一般的安装过程。

(1) 将 MATLAB 6.1 的安装盘放入光驱中，找到 setup.exe 文件，双击它开始安装（或机器自动执行安装文件）。

(2) 按照安装向导的提示进行。在【Select MATLAB Components】对话框中选择用户需要安装的选项，可选择的 MATLAB 部件包括 MATLAB、Simulink 和各种工具箱必须安装的文件，以及各部分的帮助文件（包括 HTML 和 PDF 两种格式）。

(3) 在【Select MATLAB Components】对话框中选择安装的路径。安装程序默认的路径为“C:\MATLAB”，单击【Browse...】按钮，可以设置安装路径。

(4) 单击【Next>】按钮，进行文件的解压和复制过程。

(5) 接下来安装向导会提问是否安装 MATLAB Notebook。如果用户的计算机上已经安装 Microsoft Word，那么就可以安装 MATLAB Notebook。单击【Yes】按钮确认安装，单击【No】按钮取消安装。如果安装 MATLAB Notebook，下一步可以选择 Word 的版本号，以及指定它的位置。

(6) 安装完毕。如果在安装的选项中选择了【Excel Link】，那么为了运行 MATLAB，必须重新启动计算机。用户可以选择【Yes, I want to restart my computer now】（立即重新启动计算机）或【No, I will restart my computer later】（以后启动计算机）。单击【Finish】按钮结束安装。如果系统安装成功，将在桌面上形成如图 1.1 所示的图标。否则表明安装失败，需重新安装。



图 1.1 MATLAB 应用程序图标

### 1.2.2 MATLAB 系统的启动

MATLAB 系统是一个高度集成的语言环境，使用起来非常方便；但要

使用它，首先必须启动 MATLAB 系统。启动 MATLAB 系统的方法如下：双击（或单击）桌面上（或“开始/程序/MATLAB”中）的 MATLAB 6.1 应用程序图标（如图 1.1 所示）。MATLAB 6.1 启动后，将显示如图 1.2 所示的操作界面，它表示 MATLAB 系统已建立，用户可与 MATLAB 系统进行交互操作。

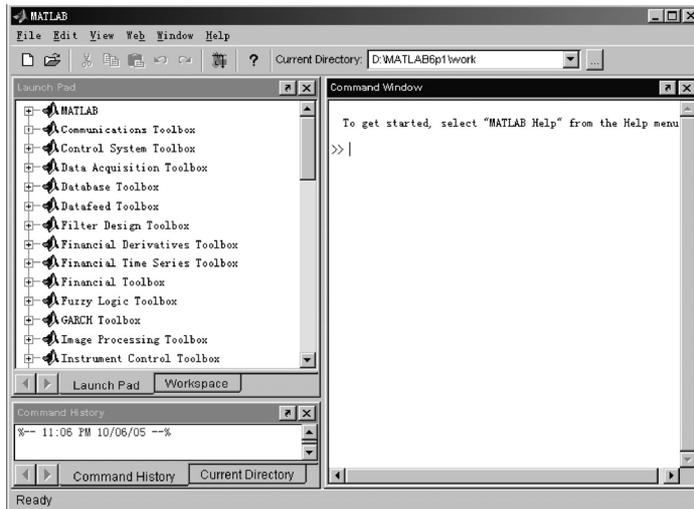


图 1.2 MATLAB 命令窗口

通常情况下，MATLAB 的工作环境主要由命令窗口（Command Window）、当前路径（Current Directory）窗口、工作空间（Workspace）浏览器窗口、命令历史（Command History）窗口、启动平台（Launch Pad）、图形（Figure）窗口和文本编辑（Editor）窗口组成。启动平台窗口是 6.x 版本的新特点，它为用户提供 MATLAB 工具箱。用户可以方便地打开工具箱中的内容，包括帮助文件、演示示例、实用工具及 Web 文档等内容。

### 1.2.3 MATLAB 的命令窗口

#### 1. 命令窗口中的菜单与功能

MATLAB 命令窗口是用户与 MATLAB 系统交互的主要窗口。在该窗口中，用户可以运行函数、执行 MATLAB 的基本操作命令，以及对 MATLAB 系统的参数设置等操作。为了灵活使用 MATLAB，下面我们将对命令窗口中的各项菜单的功能和作用进行简要介绍。

在命令窗口的菜单条下，共有 6 个下拉子菜单：File, Edit, View, Web, Windows 和 Help。

##### (1) File 菜单

File 菜单所包含的选择项如图 1.3 所示，各选项的含义分述如下。

**【New 及其子菜单】**：允许用户打开一个新的文件（M 文件）、新的图形窗（Figure）、仿真模型文件（.mdl）和图形用户界面文件（GUI）。

**【Open...】**：从指定的相应路径和文件名打开一个已经存在的文件。

**【Close Command Window】**：关闭命令窗口。

**【Import Data...】**：在 MATLAB 工作空间中生成一变量，并从指定的路径和相应的文件中获取数据。

**【Save Workspace As...】**：将工作空间中的所有变量数据保存在指定的路径下的相应的文件（.mat）中。

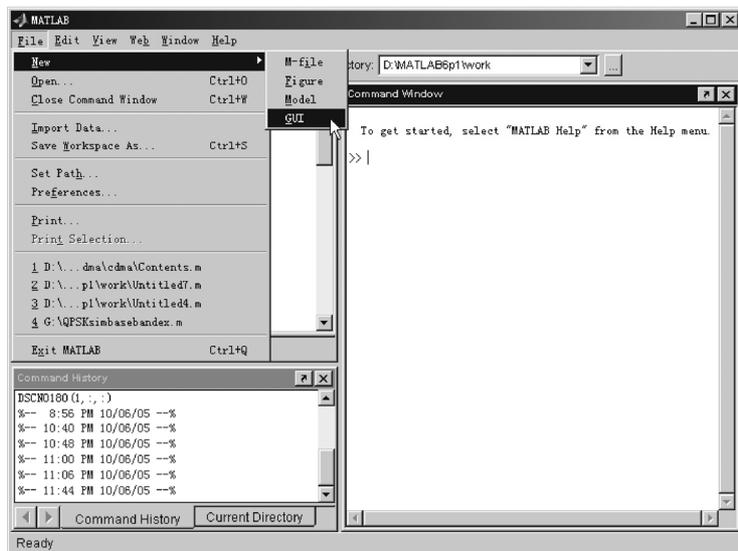


图 1.3 命令窗口下的 File 子菜单

【Set Path...】: 设置 MATLAB 的搜索路径。

【Preferences...】: 允许用户对系统的一些性能参数进行设置, 如数据格式、字体大小与颜色等。

## (2) View 菜单

View 菜单所包含的选择项如图 1.4 所示, 各选项的含义分述如下。

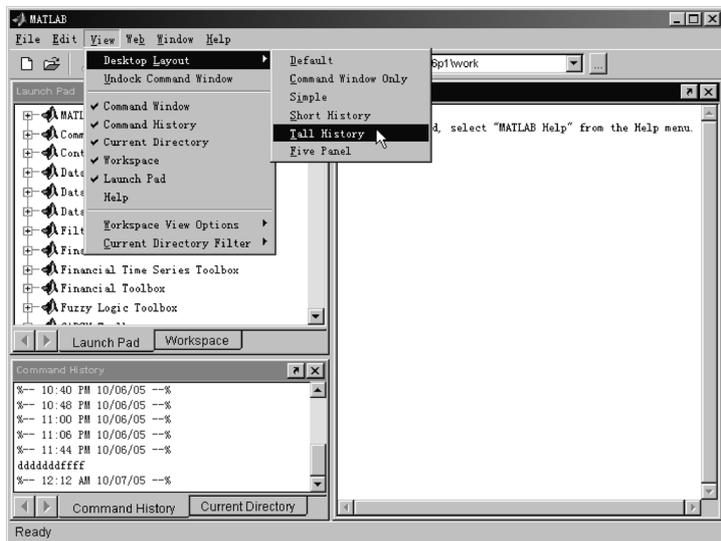


图 1.4 命令窗口下的 View 子菜单

【Desktop Layout 及其子菜单】: 允许用户在桌面上同时显示不同的窗口。

【Undock Command Window】: 单独显示命令窗口。

【Current Directory Filter 及其子菜单】: 允许用户设置当前目录浏览器中浏览的文件类型。

【Workspace View Options 及其子菜单】: 允许用户设置工作空间窗口中所显示变量的属性(大小、比特数、变量类型), 以及显示变量的方式(按变量名、大小)。

至于 Edit、Web、Windows 和 Help 菜单的用法, 由于它们与其他一些常见的应用软件用法相

同，这里就不再介绍。

MATLAB 6.1 命令窗口的工具栏如图 1.5 所示。



图 1.5 MATLAB 命令窗口的工具栏

工具栏中各按钮的含义分述如下。

-  打开一个新的.m 文件编辑器窗口。
-  在编辑器中打开一个已有的 MATLAB 相关文件。
-  剪切。
-  复制。
-  粘贴。
-  撤销上一步操作。
-  恢复上一步操作。
-  创建一个新的 Simulink 模块文件。
-  打开 MATLAB 的帮助。

## 2. 命令窗口的编辑特殊功能键与命令窗口的设置

命令窗口是 MATLAB 的主窗口。当用户使用命令窗口进行工作时，在命令窗口中可以直接输入相应的命令，系统将自动显示信息。例如在命令输入提示符“>>”后输入指令：

```
>>ty=[1,2,3;4,5,6;7,8,9];
```

按回车键（Enter）后，系统即可完成对变量 ty 的赋值。在命令输入过程中，除了可以采用常规编辑软件所定义的快捷键或功能键来完成对命令输入的编辑外，MATLAB 还提供以下特殊的功能键，为命令的输入和编辑带来方便。

↑ 调出上一个（历史）命令行

↓ 调出下一个命令行

Esc 恢复命令输入的空白状态

这些功能在程序调试时十分有用。对于已执行过的命令，如要做些修改后重新执行，可不必重新输入，用“↑”键调出原命令直接修改即可。

当输入命令的语句过长，需要两行或多行才能输入时，则使用“...”作为连接符号，按回车键转入下一行继续输入。

当用户使用命令窗口进行工作时，用户可以根据自己的习惯与要求，设置命令窗口的显示方式。

设置命令窗口时，首先要选择【File】菜单中的【Preferences】项，打开如图 1.6 所示的参数设置对话框，单击【Command Window】标签即可进入命令窗口的设置。

### （1）Text display

该选项组用来设置命令窗口中的数据格式、窗口数字显示与 Tab 制表符的字符数。【Numeric format】下拉列表框用来设置数字显示格式，MATLAB 可显示的格式如表 1.1 所示。【Numeric display】下拉列表框用来设置命令窗口的文字显示格式，选择【Compact】选项表示以文字紧缩形式显示；选择【Loose】选项表示以文字宽松形式显示。【Space per tab】文本框用来设置 Tab 制表符的宽度。

表 1.1 数字显示格式（对同一数据）

显示形式	范例(215/6)	说明
short (默认)	35.8333	两位整数, 4 位小数
long	35.83333333333334	16 位十进制数
short e	3.5833e+001	5 位十进制数加指数
long e	3.583333333333334e+001	16 位十进制数加指数
hex	4041eaaaaaaaaab	16 位十六进制数
short g	35.833	5 位十进制数
long g	35.83333333333333	15 位十进制数
bank	35.83	两位小数
+	+	正、负、零
rat	215/6	分数近似

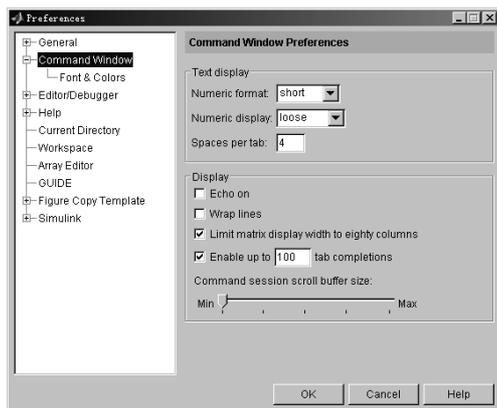


图 1.6 命令窗口参数设置对话框

## (2) Display

该选项组有以下复选框。

**【Echo on】**: 在执行 M 文件时, 如果想将执行的命令显示在命令窗口, 则可以选中该复选框。

**【Limit matrix display width to eighty columns】**: 如果想在命令窗口中显示 80 列输出, 则可以选中该复选框。

**【Enable up to 100 tab completions】**: 如果选中该复选框, 则可在命令窗口输入函数时使用 Tab 键。

**【Command session scroll buffer size】**: 该滑杆用来设置命令窗口中卷轴缓冲器的大小。

## 1.2.4 工作空间窗口

工作空间窗口 (Workspace) 是 MATLAB 6.x 版本的新特点; 以前的工作空间只是一个对话框, 可操作性差。MATLAB 6.x 版本的工作空间作为一个独立的窗口, 其操作性相当方便。它允许用户查看当前 MATLAB 工作空间的内容, 如图 1.7 所示。它的作用与命令 “whos” 相同 (“whos” 的作用是: 在命令窗口中直接输入 “whos”, 回车后即可在命令窗口中查看当前 MATLAB 工作空间的内容), 不同的是用图形化的表示方法来显示。而且, 通过它可以对工作空间中的变量进行删除、保存、修改等操作, 十分方便。



图 1.7 工作空间窗口

在工作空间中, 用鼠标双击所选变量 (也可用鼠标先对一个或多个变量完成选择后, 再单击工具条中的图标 ), 则进入数组编辑器 (Array Editor), 如图 1.8 所示。此时用户可对变量的维数、内容等进行修改。若在工作空间选择某变量后, 再单击鼠标右键即可弹出如图 1.9 所示的操作菜单, 实现对该变量的曲线、曲面等图形的绘制。

若在工作空间选择某变量后, 再单击鼠标右键即可弹出如图 1.9 所示的操作菜单, 实现对该变量的曲线、曲面等图形的绘制。

## 1.2.5 命令历史窗口与当前路径窗口

命令历史窗口 (Command History) 主要显示曾经在 Command Window 窗口执行过的命令。

当前路径窗口 (Current Directory) 主要显示当前工作在什么路径下, 包括 M 文件的打开路径等。当前路径窗口允许用户对 MATLAB 的路径进行查看和修改, 如果修改了路径会立即产生作用。通常启动 MATLAB 系统之后的默认当前路径是 “\MATLAB\work”, 如果不改变当前目录,

用户自己的工作空间和文件都将保存到该目录。

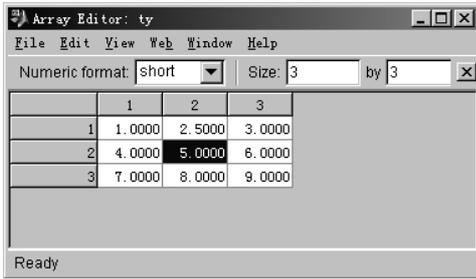


图 1.8 数组编辑器

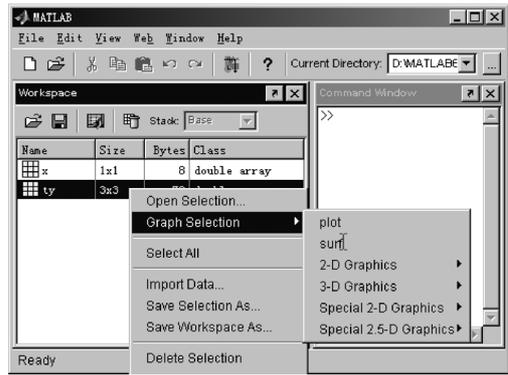


图 1.9 工作空间操作菜单

需要注意的是，在“\MATLAB\bin”路径的目录下存放着 MATLAB 的许多重要文件，如果用户操作不慎，比如误删了一些重要的系统文件，MATLAB 的运行就可能出现意想不到的问题。所以对 MATLAB 初学者来说，这一点更应引起注意。

另外，MATLAB 是采用路径搜索的方法来查找文件系统中的 M 文件的。如果在命令窗口中输入命令

```
>>test
```

(回车)

MATLAB 对这一命令的搜索顺序为：

(1) 检查“test”是否为存储在工作空间中的变量。若为工作空间中的变量，则返回该变量的内容；

(2) 检查“test”是否为 MATLAB 的内部函数。若为内部函数，则返回要求输入到该函数的参数信息。例如在命令窗口中输入命令

```
>>fft
```

则得到下面的反馈信息：

```
??? Error using ==> fft
```

(错误使用 fft 函数)

```
Not enough input arguments.
```

(没按要求格式输入)

(3) 检查当前目录中是否有 test.m、test.mex 或 test.dll 文件；

(4) 检查 MATLAB 搜索路径上是否存在 test.m、test.mex 或 test.dll 文件；

(5) 如不满足上述任何一个条件，则返回出错信息。

如果在搜索路径中存在两个或多个同名函数时，则只能发现搜索路径中的第一个函数，而其他同名函数不被执行。此搜索的顺序只是一般情况下的顺序，而实际的搜索规则则要复杂得多。

对于初学者来说，有时会出现运行自己编写的程序时，MATLAB 系统告之该程序不存在的情况，并出现如图 1.10 所示对话框。这是因为该程序不在 MATLAB 的搜索路径中。为了运行该程序，用户可以选择对话框中任何一项，确认后即可运行。各项的含义分述如下。

**【Change MATLAB current directory】：**将文件所在的路径更换为 MATLAB 的当前路径。

**【Add directory to the top of the MATLAB path】：**将文件所在的路径添加到 MATLAB 路径的前端。

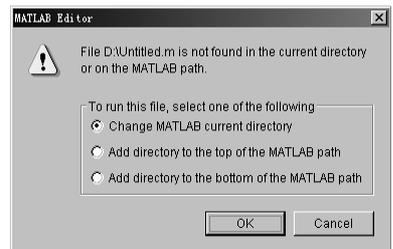


图 1.10 文件不存在对话框

【Add directory to the bottom of the MATLAB path】: 将文件所在的路径添加到 MATLAB 路径的末端。

为了运行已有程序, 用户可事先设置好 MATLAB 的搜索路径, 然后再运行程序。

方法一: 在当前路径窗口 (Current Directory) 将文件所在的路径设置为当前路径。

方法二: 单击【File】菜单中的【Set Path...】选项, 弹出路径设置对话框如图 1.11 所示。

在【Set Path】对话框中, 可以使用【Move Up】、【Move Down】、【Move to Top】、【Move to Bottom】等按钮调整搜索路径的顺序。使用【Remove】按钮可以删除选中的搜索路径。

单击【Add Folder】按钮则打开如图 1.12 所示的【浏览文件夹】对话框, 选择要添加的目录。在【Set Path】对话框中还可以单击【Add with Subfolders】按钮, 将选中的目录路径的子目录也包含在搜索路径中。

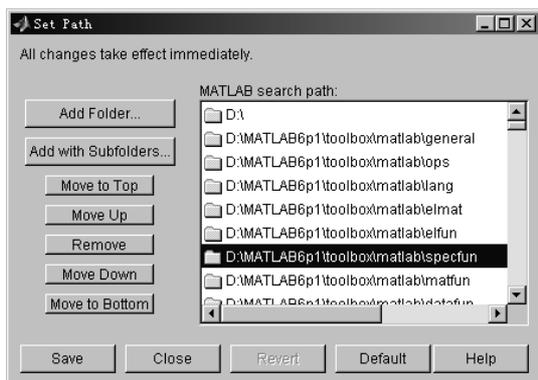


图 1.11 路径设置对话框



图 1.12 【浏览文件夹】对话框

## 1.2.6 图形窗窗口

MATLAB 图形窗窗口 (Figure) 主要用于显示用户所绘制的图形。通常, 只要执行了任意一种绘图命令, 图形窗窗口就会自动产生。绘图都在该图形窗中进行。如果要再建一个图形窗窗口, 则可输入 figure 命令, MATLAB 会新建一个图形窗窗口, 并自动给它排出序号。

关于图形窗口的功能说明, 将在 2.6.6 节详细介绍。

## 1.2.7 文本编辑窗窗口

### 1. 文本编辑窗窗口启动

通常, MATLAB 的命令编辑有行命令方式和文件方式两种。行命令方式就是在命令窗口中一行一行地输入命令, 计算机对每一行命令做出反应。文件方式就是将多行语句组成一个文件 (.M 文件), 然后让 MATLAB 来执行这个文件中的全部语句。因此, 行命令方式只能编辑简单的程序, 在入门时通常用这样方式完成命令编辑。文件方式可以编写较复杂的程序。

文本编辑窗的作用就是用来创建、编辑和调试 MATLAB 相关文件 (.M 文件), 它与一般的编辑调试器有相似的功能。

MATLAB 文本编辑/调试器的启动可以从命令窗口中选择【新建】或【打开】文件按钮进入, 或在命令窗口中输入: edit (回车)。其编辑窗口如图 1.13 所示。

下面简要介绍调试【Debug】菜单和断点设置【Breakpoints】菜单中相关项的功能与作用。

【Save and Run】: 保存并运行程序, 直到遇到下一个断点, 对应工具按钮为.

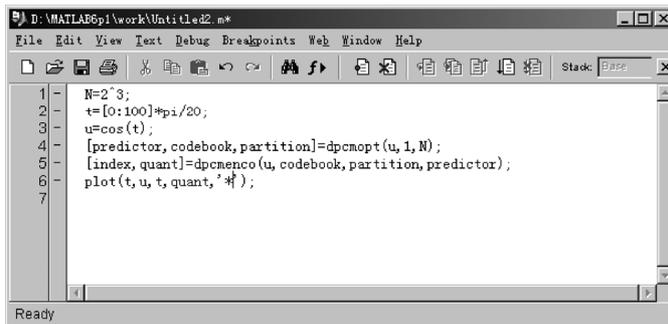


图 1.13 文本编辑窗口

【Single Step】: 单步执行，对应工具按钮为 。

【Step In】: 运行当前程序行，对应的工具按钮为 。如果当前行调用了另外一个函数，则跳转到这个函数中。

【Exit Debug Mode】: 退出调试模式，对应的工具按钮为 。

【Set/Clear Breakpoint】: 设置/清除断点，对应的工具按钮为 。

【Clear All Breakpoints】: 清除所有的断点，对应的工具按钮为 。

【Stop If Error】: 程序运行时遇到错误则停止。

【Stop If Warning】: 程序运行遇到警告则停止。

【Stop If NaN or Inf】: 程序运行时遇到不是数（Not a Number）或是无穷大（Infinite Value）则停止。

## 2. 文本编辑器的参数设置

当使用 MATLAB 编辑/调试器编辑文件时，常常需要设置一些适合自己需要的工作环境，此时，可选择【File】菜单中的【Preferences】项，打开参数设置对话框，单击【Editor/Debugger】

标签即可进入如图 1.14 所示的文本编辑器参数设置界面。参数设置共分为五大项：【Editor/Debugger】（编辑与调试器）设置、【Font & Colors】（字体与颜色）设置、【Display】（显示方式）设置、【Keyboard & Indenting】（键盘与缩进）设置和【Printing】（打印）设置，每个大项中又由若干个小的设置项组成。用鼠标单击相应的标签，即可弹出相应的参数设置对话框界面。

(1) 编辑与调试器 (Editor/Debugger) 的参数设置

【Editor】选项组：选中【Built-in editor】项表示使用 MATLAB 的内置编辑器；选中【Other】项表示可以使用其他编辑器，此时要求输入编辑器的路径及应用程序名。

【Debugger Options】选项组：选中【Automatically open files when debugging】复选框表示在调试时自动打开文件。

【Most recently used files list】选项组：用来设置最近使用的文件列表数目。

【On restart】选项组：若选中【Reopen files from previous MATLAB session】复选框，则表示

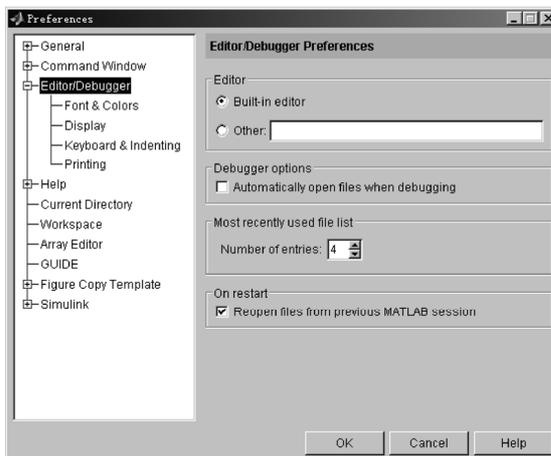


图 1.14 文本编辑器参数设置界面

下次启动 MATLAB 时，打开上一次退出 MATLAB 时正在编辑调试的文件。

### (2) 字体与颜色 (Font & Colors) 的设置

【Font】选项组：用来设置字体。选中【Use desktop font】项，则表示 Editor/Debugger 窗口中的字体采用 Windows 桌面字体；若选中【Use custom font】项，则用户可以设置自己喜欢的字体，包括字体的类型与大小。

【Colors】选项组：用来设置颜色。【Text color】项完成对字体颜色的设置；【Background color】项完成对背景颜色的设置；【Syntax highlighting】项可使编辑框中的语法项高亮显示，以便与其他语句区别开来。若想设置高亮显示颜色，可以单击【Set Color】按钮进入颜色设置。

### (3) 显示方式 (Display) 的设置

【Opening files in editor】选项组：设置编辑器中文件打开方式。【Single window contains all files (tabbed style)】项表示在一个窗口中显示多个文件，各个文件以标签的形式显示在左下角；【Each file is displayed in its own window】项表示每个文件在各自独立的窗口中显示。

【Display】选项组：【Show toolbar】项表示在 Editor/Debugger 中显示工具栏；【Show line numbers】项表示在 Editor/Debugger 中显示文本的行数，这在修改与调试 M 文件时非常有用；选中【Enable datatips in edit mode】项，在编辑窗口中，用户用鼠标指针指向某个变量时，系统会自动显示该变量的内容。

### (4) 键盘与缩进 (Keyboard & Indenting) 参数设置

【Key bindings】选项组：设置用户习惯的键盘定义。【Windows】项表示使用 Windows 系统约定的键盘快捷定义，如复制和粘贴的快捷键分别为 Ctrl+C、Ctrl+V；【Emacs】项表示使用 Emacs 约定的键盘快捷定义，如复制快捷键为 Ctrl+Y。

【M-file indenting for Enter key】选项组：设置 M 文件的不同缩进格式。【No indent】项表示文本无缩进格式，【Block indent】项表示以块形式缩进格式，【Smart indent】项表示智能缩进格式。

【Indent】选项组：设置适合用户的缩进参数。【Indent size】文本框表示可输入同一标准的嵌套代码列数，【Emacs style Tab key smart indenting】项表示可以通过 Tab 键缩进当前行。

【Tab】选项组：设置适合用户的制表符参数。【Tab size】文本框表示设置两表符 Tab 间的空格数。【Tab key insert space】项表示可插入一个 Tab 字符。

## 1.3 MATLAB 的基本操作命令

MATLAB 的命令基本上可以分为五类：管理命令和函数、管理变量和工作空间的命令、控制命令窗口的命令、对文件和环境操作的命令，以及退出 MATLAB 的命令。这些基本命令放在 matlab\general 目录下，用户只需在命令窗口中输入

```
>>help matlab\general    (回车)
```

就可查看这些命令。下面简要介绍一些常用基本命令，并且只介绍它们的主要用法或调用格式。至于各种详细用法，用户可以查看帮助，方法是在命令窗口中输入：help 相应的命令（回车）。

### 1. 窗口命令

clf: 清除当前图形窗口 (Figure) 中的所有非隐藏图形对象。

close: 关闭当前的图形窗口 (Figure)。

close all: 关闭所有的图形窗口 (Figure)。

clc: 清除命令窗口中的内容，光标回到窗口的左上角。

home: 光标回到窗口的左上角。

## 2. 工作空间管理命令

who: 列出当前工作空间里的所有变量。

who('global'): 列出全局变量。

whos: 列出当前工作空间里的所有变量及大小、类型和所占的存储空间。

whos('global'): 列出全局变量及大小、类型和所占的存储空间。

clear: 从工作空间清除所有变量。

clear global: 从工作空间清除所有全局变量。

clear all: 从工作空间清除所有变量、函数和 MEX 文件。

pack: 将所有变量保存到磁盘, 然后清除内存并从磁盘恢复变量, 有利于提高内存的利用率。

save: 将工作空间里的变量保存到磁盘文件。

load: 将磁盘文件里的变量加载到工作空间。

workspace: 显示工作空间浏览器 (Workspace)。

quit: 退出 MATLAB 系统。

## 3. 显示格式设置命令

format type: 输出数据格式显示控制命令。

echo on(off): 显示 (不显示) 正在执行的 M 文件语句。

more on(off): 屏幕显示内容多少的控制 (不控制)。

## 4. 路径编辑命令

path: 显示所有的 MATLAB 路径。path(path,'newpath')表示把一个新的路径 (newpath) 附加到当前搜索路径后。path('newpath',path)表示把一个新的路径 (newpath) 附加到当前搜索路径前。

addpath: 将一个新目录添加到 MATLAB 的搜索路径里。调用格式为: addpath('directory')。

rmpath: 从 MATLAB 搜索路径里清除某个目录。

## 5. 调试命令

dbclear: 清除断点。

dbcont: 重新开始运行。

dbdown: 改变局部工作空间上下文, 但在此之前必须运行过一次 dbup 命令。

dbquit: 退出调试模式。

dbstack: 显示当前运行程序行的行号和 M 文件名。

dbstatus: 列出所有断点。

dbstep: 从一个断点开始运行一行或多行程序。

dbstop: 在 M 文件中设置断点。

dbtype: 带行号显示当前执行点所在的 M 文件。

dbup: 与 dbdown 相对应。

## 6. 文件操作命令

what: 列出当前目录下 MATLAB 指定的文件, 包括 M、MAT、MEX、MDL 和 P 文件等。

**which:** 显示函数或文件的位置。  
**type:** 在命令窗口中显示文件的内容。  
**edit:** 编辑 M 文件。

## 7. 操作系统命令

**cd:** 输出当前目录名。  
**cd<目录>:** 进入目录。  
**cd...:** 回到上一级目录。  
**dir<目录名>:** 列出指定目录中的文件及其子目录。  
**delete:** 删除文件或图形对象。  
**pwd:** 显示当前工作目录的名称。  
**mkdir:** 创建目录。  
**copyfile:** 复制文件, 与 DOS 下的 copy 命令一样。  
**web:** 打开网络浏览器, 并连接到某个具体的网址或文件。  
**computer:** 显示计算机的类型。

## 8. 帮助命令

**help:** 在命令窗口中显示 MATLAB 函数或命令的帮助信息。  
**lookfor:** 在注释的第一行中按主题搜索用户的目标文件或函数。  
**doc:** 在 help 浏览器中显示某个函数的联机帮助文档, 参数为函数名。  
**helpwin:** 在 help 浏览器中显示 M 文件的联机帮助文档。联机帮助按函数的类别分类, 用户可以进入不同类别的帮助目录, 对这一类函数一览无遗。  
**helpdesk:** 显示 help 浏览器。  
**ver:** 显示 MATLAB 的版本。

MATLAB 为用户提供的这些命令或函数命令, 用户既可在命令窗口中按要求格式输入命令来实现相应的功能, 也可在文本编辑窗口中, 通过编辑形成程序文件后让 MATLAB 来执行。但值得注意的是, 虽然 MATLAB 语言是在 C 语言的基础上开发而成的, MATLAB 语言的结构与 C 语言有相似的地方, 但存在本质的不同。MATLAB 语言是解释性语言, 而 C 语言是编译性语言。因此, 用 MATLAB 语言编写的程序不能脱离 MATLAB 的工作环境而运行, 它的执行过程为: MATLAB 系统对一条命令或一组命令逐条进行翻译和处理, 并返回每条命令的运算结果。

MATLAB 语言比较好学, 其原因是有时用户需要实现的功能, 只需调用一个函数命令就可达到目的, 简化了烦琐的编程; 其二是 MATLAB 只有一种数据类型, 一种标准输入输出语句, 不用指针, 不需编译, 比其他语言少了很多内容。MATLAB 语言的难点就是函数命令较多, 仅基本部分就有 700 多个, 其中常用的有近 200 个。为了克服这一不足, MATLAB 为用户提供非常方便的在线帮助命令 (help) 和演示命令 (demo), 它们可提供各个函数的用法指南, 包括格式、参数说明、注意事项及相关函数等内容。对于初学者, 应充分学会帮助命令和演示命令的使用。通过 MATLAB 的演示和帮助命令, 可以方便地在线学习各种函数的用法及其内涵。对常用的函数命令要尽量多记少查, 以提高编程效率。

help 命令主要有以下几种格式。

格式一: help

功能: 显示 MATLAB 的所有目录项。

格式二: help 目录名

功能: 显示指定目录中的所有命令及其函数。例如

```
help matlab\general
```

格式三: help 命令名 或 help 函数名 或 help 符号

功能: 显示出有关指定命令/函数/符号的详细信息, 包括命令格式及注意事项。如查看均值函数命令 mean 的调用格式, 在命令窗口中输入:

```
>> help mean          (回车)
```

```
显示:  MEAN    Average or mean value.      ←函数名及函数功能
      For vectors, MEAN(X) is the mean value of the elements in X. For
      matrices, MEAN(X) is a row vector containing the mean value of
      each column.  For N-D arrays, MEAN(X) is the mean value of the
      elements along the first non-singleton dimension of X.
      MEAN(X,DIM) takes the mean along the dimension DIM of X.
      Example: If X = [0 1 2 3 4 5]
      then mean(X,1) is [1.5 2.5 3.5] and mean(X,2) is [1 4]
      See also MEDIAN, STD, MIN, MAX, COV.    ←相关函数
      Overloaded methods                    ←进一步帮助
      help fints/mean.m
```