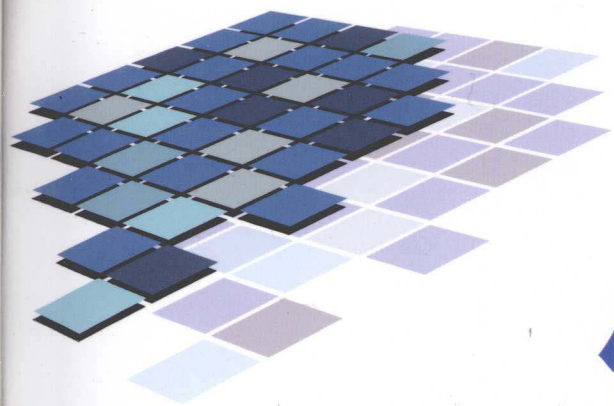
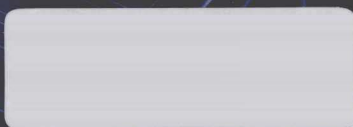


 极光



信道编码 与MATLAB仿真

刘东华 向良军 编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

信道编码与 MATLAB 仿真

刘东华 向良军 编著

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

信道编码作为一种基础技术,广泛应用于各种通信系统和存储系统,是实现可靠通信和数据存储的主要手段。MATLAB 作为一种常用的工程语言,具有强大的科学计算和可视化功能,简单易用,且提供了针对通信系统设计的工具箱(算法功能模块),是进行信道编码研究、仿真和应用开发的基本工具和首选平台。

本书首先简要介绍 MATLAB 语言和程序设计方法,然后分章节介绍常用的线性分组码、卷积码、Turbo 码、TPC 码和 LDPC 码的编/译码原理和算法,给出编码、译码和联合仿真的 MATLAB 代码和注释,并对信道编码在深空通信中的应用进行介绍,给出仿真流程和仿真结果,最后介绍 MATLAB 提供的各种信道仿真模型和误比特率计算和分析工具。

本书的主要读者对象是高校通信和电子专业本科生和研究生以及通信系统设计工程师,也可以作为高校信息论与编码等课程的实验参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

信道编码与 MATLAB 仿真 / 刘永华, 向良军编著. — 北京: 电子工业出版社, 2014.2
ISBN 978-7-121-22377-8

I. ①信… II. ①刘… ②向… III. ①信道编码—计算机仿真—Matlab 软件 IV. ①TN911.22

中国版本图书馆 CIP 数据核字(2014)第 010982 号



责任编辑: 田宏峰 特约编辑: 牛雪峰

印 刷: 北京天宇星印刷厂

装 订: 三河市皇庄路通装订厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 28.25 字数: 720 千字

印 次: 2014 年 2 月第 1 次印刷

印 数: 3 000 册 定价: 79.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

随着通信技术的不断发展,现代通信系统的复杂化以及通信业务的多样化要求系统能够实现高速、实时和可靠数据传输,而且用户对通信质量的要求也在不断提高,使得对高数据率数字通信等领域所采用的信道编码技术的要求也越来越高。

在通信系统设计的过程中,信道编码是必须要考虑的主要组成部分。如何选择合适的信道编码方法,在不超出系统复杂性要求的前提下实现目标通信的可靠性,是信道编码实用化的主要研究内容。通常,计算机仿真是选择和设计信道编码的主要手段之一。MATLAB 软件作为一种工程化语言,对信道编码的仿真有很好的支持。本书结合信道编码基本原理的介绍和具体的 MATLAB 仿真方法,为读者提供方便快捷地理解信道编码以及通过仿真进行信道编码设计和选择的方法。

全书共分 9 章,各章内容编排如下。

第 1 章首先介绍 MATLAB 集成开发环境和 Simulink 软件的使用方法,然后简要说明信道编码的作用和分类,以及几类常用的信道编码。

第 2 章主要介绍 MATLAB 的基本编程组件、数据类型、矩阵和数组、基本数学运算、数据导入导出和 MATLAB 绘图等在信道编码仿真时用到的 MATLAB 基础知识。

第 3 章重点描述 MATLAB 程序设计方法,包括 M 文件编程、MATLAB 函数方法、程序调试和出错处理、内存使用和优化、程序计划、仿真性能分析和优化等程序设计思想,最后简要介绍 MATLAB 提供的通信工具箱的内容。

第 4 章首先针对信道编码研究和开发涉及的有限域及其运算进行具体解释和说明,然后重点介绍线性分组码、循环码、BCH 码和 RS 码等分组码,给出每种分组码的原理、编码方法和译码算法以及 MATLAB 仿真程序代码,并详细介绍 MATLAB 提供的相关函数。同时本章还给出了在深空通信中 RS 码的应用和仿真结果。

第 5 章介绍卷积码的原理、编码方法和 Viterbi 译码算法,重点分析 MATLAB 提供的卷积码编译码函数和 Simulink 仿真模块,最后给出在深空通信中卷积码以及 RS 码和卷积码级联组合码的结构和仿真结果。

第 6 章针对 Turbo 码这一性能优异的级联码进行分析,在介绍并行级联编码和迭代译码原理和算法的基础上,给出了 MATLAB 仿真代码,详细解释 MATLAB 提供的各种类型的交织器函数,最后介绍深空通信中的标准 Turbo 码及仿真结果。

第 7 章介绍结构简单但性能较好的 TPC 码,描述编码方法和基于迭代的改进 Chase 译码算法,最后给出具体的 MATLAB 实现代码和仿真结果。

第 8 章重点介绍 LDPC 码的原理和基于消息传递的译码算法,给出基于迭代的译码仿真代码,结合深空通信标准的 LDPC 码介绍编码方法并给出仿真结果。

第 9 章主要介绍通信系统和信道编码仿真中常用的信道模型,包括 BSC、AWGN 信道和衰落信道模型,详细介绍各种信道模型的 MATLAB 仿真函数和 Simulink 仿真模块的参数设置和使用方法,最后简要介绍 MATLAB 提供的通信误比特率和误符号率统计函数及误比特率分析工具。

本书详尽介绍现有常用信道编码及其 MATLAB 仿真实现方法,编写过程中参考了经典的信道编码专著和部分学位论文,在此对其相关作者表示衷心的感谢。

参与本书编写工作的还有梁光明、胡耀华、刘丽华、刘祎斌、吕文龙、刘军发、敖宇、王月磊、蔺吉顺、马新军、张绘国等。

感谢电子工业出版社和本书的责任编辑田宏峰老师，在田老师的大力支持和帮助下本书才得以与广大读者见面。

由于作者水平有限，错误遗漏之处在所难免，恳请各位读者批评指正。同时也欢迎大家就信道编码及MATLAB仿真相关技术与作者交流，联系邮箱：ldh_1976@163.com。

作者

2013年12月

目 录

第 1 章 概述	(1)
1.1 MATLAB	(1)
1.1.1 MATLAB 的组成	(1)
1.1.2 开发环境	(2)
1.2 Simulink	(6)
1.2.1 启动 Simulink	(6)
1.2.2 创建 Simulink 模型	(8)
1.2.3 模块参数设置	(10)
1.2.4 仿真 Simulink 模型	(10)
1.3 信道码	(11)
1.3.1 信道码在数字通信系统中的地位和作用	(11)
1.3.2 信道码的类型	(12)
1.3.3 常用信道编译码方法	(13)
1.3.4 信道码的应用	(14)
1.4 基于 MATLAB 的信道码仿真	(14)
第 2 章 MATLAB 编程基础	(15)
2.1 基本编程组件	(15)
2.1.1 MATLAB 命令	(15)
2.1.2 变量	(16)
2.1.3 关键字	(16)
2.1.4 操作符	(17)
2.1.5 MATLAB 表达式	(18)
2.1.6 程序控制状态	(18)
2.1.7 常用符号	(20)
2.2 数据类型	(22)
2.2.1 数值	(22)
2.2.2 逻辑类型	(28)
2.2.3 字符和字符串类型	(29)
2.2.4 日期和时间	(30)
2.2.5 结构体	(31)
2.2.6 元胞数组	(31)
2.2.7 函数句柄	(32)
2.2.8 MATLAB 类和 Java 类	(32)
2.3 矩阵	(32)
2.3.1 矩阵构造	(33)
2.3.2 矩阵访问	(34)

2.3.3	矩阵操作	(37)
2.3.4	对角矩阵	(39)
2.4	数学运算	(40)
2.4.1	线性运算	(40)
2.4.2	稀疏矩阵	(41)
2.4.3	多项式	(44)
2.5	数据导入/导出	(44)
2.5.1	支持的文件格式	(44)
2.5.2	导入向导的使用	(46)
2.5.3	导入/导出 MAT 文件	(48)
2.5.4	导入/导出文本数据	(49)
2.5.5	底层文件 I/O 函数	(51)
2.6	MATLAB 绘图	(52)
2.6.1	基本绘图命令	(53)
2.6.2	绘制特殊图	(60)
2.6.3	图形编辑工具	(67)
第 3 章	MATLAB 程序设计	(74)
3.1	M 文件编程	(74)
3.1.1	M 文件设计	(74)
3.1.2	M 文件结构	(75)
3.1.3	M 文件脚本和函数	(76)
3.1.4	函数调用	(77)
3.2	函数类型	(79)
3.2.1	匿名函数	(79)
3.2.2	主函数	(79)
3.2.3	嵌套函数	(80)
3.2.4	子函数	(81)
3.2.5	私有函数	(82)
3.2.6	重载函数	(82)
3.3	程序调试和出错处理	(82)
3.3.1	调试程序	(83)
3.3.2	出错处理	(87)
3.4	内存使用和优化	(91)
3.4.1	内存分配	(91)
3.4.2	内存管理函数	(92)
3.4.3	提高内存使用效率	(93)
3.5	程序计划	(96)
3.5.1	创建和设置定时对象	(97)
3.5.2	启动和停止定时器	(98)
3.5.3	创建和执行回调函数	(99)

3.5.4	定时对象执行模式	(100)
3.6	性能分析和优化	(101)
3.6.1	程序性能分析	(102)
3.6.2	MATLAB 程序优化	(106)
3.7	MATLAB 通信工具箱	(109)
3.7.1	信道编译码函数	(109)
3.7.2	交织器函数	(110)
3.7.3	信道模型函数	(111)
3.7.4	Galois 域函数	(112)
3.7.5	其他相关函数	(113)
第 4 章	线性分组码	(114)
4.1	有限域及其运算	(114)
4.1.1	基本概念	(114)
4.1.2	有限域上的运算	(115)
4.1.3	有限域多项式	(116)
4.1.4	有限域相关函数	(118)
4.2	线性分组码	(124)
4.2.1	基本原理	(124)
4.2.2	编码方法	(129)
4.2.3	译码算法	(130)
4.2.4	MATLAB 仿真实现	(132)
4.3	循环码	(150)
4.3.1	基本原理	(150)
4.3.2	编码方法	(156)
4.3.3	译码算法	(158)
4.3.4	MATLAB 仿真实现	(164)
4.4	BCH 码和 RS 码	(173)
4.4.1	基本原理	(173)
4.4.2	编码方法	(177)
4.4.3	译码算法	(178)
4.4.4	MATLAB 仿真实现	(189)
4.4.5	CCSDS 标准 RS 码	(216)
第 5 章	卷积码	(223)
5.1	基本原理	(223)
5.1.1	卷积码的定义	(223)
5.1.2	生成矩阵和校验矩阵	(224)
5.1.3	状态图和网格图描述	(231)
5.1.4	距离特性和重量分布	(233)
5.1.5	特殊卷积码	(235)

5.2	编码方法	(236)
5.3	译码算法	(236)
5.3.1	硬判决 Viterbi 译码算法	(236)
5.3.2	软判决 Viterbi 算法	(240)
5.4	MATLAB 仿真实现	(242)
5.5	CCSDS 标准卷积码及仿真	(261)
5.5.1	CCSDS 标准建议的 (7,1/2) 卷积编码器	(261)
5.5.2	CCSDS 标准建议的删余卷积编码器	(262)
5.5.3	仿真实现流程	(263)
5.5.4	仿真结果	(265)
5.6	CCSDS 标准级联码	(267)
5.6.1	级联码的编码和译码	(268)
5.6.2	仿真实现流程	(268)
5.6.3	仿真结果	(270)
第 6 章	Turbo 码	(272)
6.1	基本原理	(272)
6.1.1	Turbo 码的定义	(272)
6.1.2	分量码设计	(272)
6.1.3	交织器设计	(274)
6.2	编码方法	(276)
6.3	译码方法和算法	(279)
6.3.1	迭代译码	(279)
6.3.2	MAP 类算法	(281)
6.3.3	SOVA	(287)
6.4	MATLAB 仿真实现	(294)
6.4.1	编码	(294)
6.4.2	译码	(298)
6.4.3	编译码联合仿真	(304)
6.4.4	MATLAB 交织函数的使用	(305)
6.5	CCSDS 标准 Turbo 码	(322)
6.5.1	编码	(322)
6.5.2	译码	(323)
6.5.3	仿真实现流程	(324)
6.5.4	仿真结果	(327)
第 7 章	TPC 码	(329)
7.1	基本原理	(329)
7.1.1	TPC 码的定义	(329)
7.1.2	分量码的设计	(330)
7.2	编码方法	(332)
7.3	译码算法	(334)

7.3.1	硬判决译码	(334)
7.3.2	软判决译码	(335)
7.3.3	改进的 Chase 算法	(335)
7.3.4	迭代译码	(340)
7.4	MATLAB 仿真实现	(341)
7.4.1	编码	(341)
7.4.2	译码	(342)
7.4.3	编译码联合仿真	(345)
7.5	仿真及性能分析	(347)
第 8 章	LDPC 码	(349)
8.1	基本原理	(349)
8.1.1	LDPC 码的定义	(349)
8.1.2	LDPC 码的因子图表示	(350)
8.2	LDPC 码的构造	(354)
8.2.1	Gallager LDPC 码	(354)
8.2.2	准循环 LDPC 码	(355)
8.2.3	基于 Q 矩阵构造 LDPC 码	(356)
8.3	译码算法	(358)
8.3.1	和积算法的基本概念	(358)
8.3.2	概率测度下 LDPC 码的和积译码算法	(360)
8.3.3	不同测度下的消息更新规则	(362)
8.4	MATLAB 仿真实现	(364)
8.4.1	编码	(364)
8.4.2	译码	(364)
8.4.3	编译码联合仿真	(366)
8.4.4	MATLAB 函数的使用	(367)
8.5	LDPC 码在深空通信中的应用	(378)
8.5.1	编码	(382)
8.5.2	译码	(385)
8.5.3	仿真实现流程	(386)
8.5.4	仿真结果	(388)
第 9 章	信道仿真和误码率分析	(391)
9.1	二元对称信道	(391)
9.1.1	数学模型	(391)
9.1.2	MATLAB 仿真	(392)
9.1.3	Simulink 仿真	(393)
9.2	高斯加性白噪声信道	(394)
9.2.1	数学模型	(394)
9.2.2	MATLAB 仿真	(395)
9.2.3	Simulink 仿真	(397)

9.3	衰落信道	(399)
9.3.1	数学模型	(400)
9.3.2	MATLAB 仿真	(401)
9.3.3	Simulink 仿真	(423)
9.4	误码率分析	(426)
9.4.1	通用误码率计算函数	(426)
9.4.2	不同信道条件下误码率的计算	(431)
9.4.3	误码率分析工具	(435)
参考文献		(441)

第 1 章

概 述

1.1 MATLAB

MATLAB 是 MathWorks 公司开发的一种集成了数值计算、可视化和编程功能的高性能科学计算语言，可以用于解决很多科学计算问题。开发环境简单易用，是主流的仿真计算工具之一。MATLAB 软件的典型应用有：数学计算，算法开发，数据获取，系统建模、仿真和原型系统建立，数据分析、研究和可视化，科学和工程制图，应用开发（包括建立 GUI）等。

本节首先简要介绍 MATLAB 软件的组成，然后以图形化的方式介绍 MATLAB 开发环境，最后对 MATLAB 软件的使用进行简要说明。

1.1.1 MATLAB 的组成

MATLAB 软件由以下部分组成。

(1) 桌面工具和开发环境：MATLAB 桌面工具和开发环境是一系列工具的集合，许多工具都支持图形用户接口 (GUI)，用户可以方便地使用 MATLAB 函数和文件进行编程。系统包括 MATLAB 桌面和命令窗、命令历史记录、编辑器、调试器、代码分析器和报告、查看帮助的浏览器、工作空间、文件和搜索路径等。

(2) 语言：MATLAB 是一种高级矩阵/数组语言，包括控制流状态、函数、数据结构、输入/输出以及面向对象编程特性等。MATLAB 采用演算纸式的编程风格，可以快速创建和编写较小的应用程序，来完成小规模算法验证、仿真和调试。使用 MATLAB 语言也可以编写规模大、算法复杂的程序，实现系统级仿真。

(3) 绘图：MATLAB 提供了图形化显示矢量和矩阵的工具，还支持图形的标记和打印，包括二维和三维数据可视化、图像处理、动画等高级函数，也包括一些基本函数。用户可以根据需要设计图形显示，在自己开发的 MATLAB 应用程序中建立完整的图形用户接口。

(4) 数学函数库：MATLAB 数学函数库由大量数学计算算法组成，从求和、正弦、余弦以及复数运算等基本函数到矩阵求逆、矩阵变换、Bessel 函数以及快速 Fourier 变换等复杂函数，涵盖了工程应用中的大部分数学运算，极大地方便了仿真程序开发。

(5) 外部接口：外部接口提供了 MATLAB 程序与 C 和 Fortran 语言混合编程的库，包括在

MATLAB 中以动态链接库的方式调用进程、将 MATLAB 作为计算引擎调用、读写 MAT 文件等。

1.1.2 开发环境

MATLAB 支持 Microsoft Windows 平台、Apple Macintosh 平台和 Open Group UNIX 平台。下面以 Microsoft Windows 平台上 MATLAB 的使用为例进行说明。

MATLAB 的启动方式有如下几种（首先安装了 MATLAB 软件）。

- 在桌面上双击 MATLAB 应用程序图标；
- 在“Start”菜单上选择 MATLAB 应用程序；
- 在 MATLAB 安装目录下的“bin”子目录中双击“MATLAB.exe”文件；
- 在命令行输入命令：（假设软件安装在 D 盘根目录下）“D:\MATLAB\R2008a\bin”。

启动后，出现 MATLAB 主界面，其中包括管理文件、变量和应用的工具（图形用户接口），如菜单、工具栏、工作空间窗口、命令窗口和历史操作窗口等。

MATLAB 的启动界面如图 1.1.1 所示。

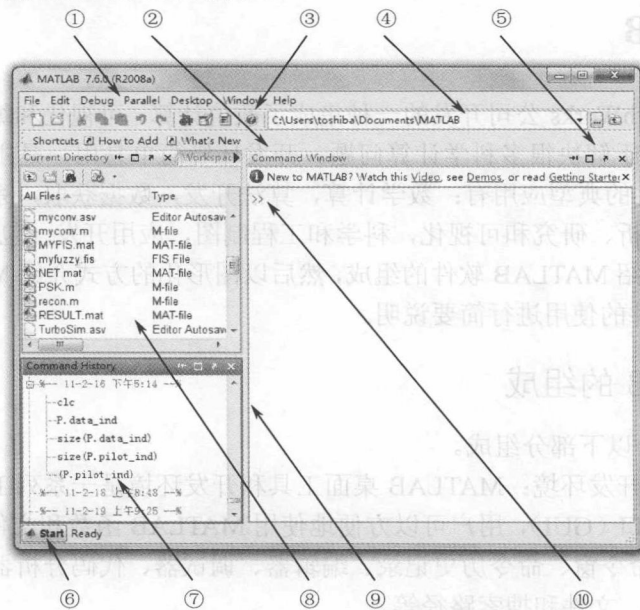


图 1.1.1 MATLAB 主界面

图中各标号的说明如下。

① 菜单和工具栏。菜单和工具栏提供 MATLAB 基本操作功能，可以根据需要调整工具栏。可以单击菜单的“File”→“Preference...”命令项或者在工具栏位置单击鼠标右键，选择“customize...”命令项，弹出如图 1.1.2 所示的配置窗口，然后选择修改工具栏显示的内容。

② 标题栏。可以选择“Command Window”、“Command History”、“Current Directory”或“Workspace”标签，使用相应的工具。

● 命令窗口（Command Window）：可以输入 MATLAB 命令并显示执行结果。

● 命令历史记录（Command History）：记录在 MATLAB 命令窗输入过的命令，可以在该窗口浏览输入的命令并单击重新执行命令。

● 当前路径（Current Directory）：显示当前 MATLAB 应用执行的路径。

● 工作空间 (Workspace): MATLAB 运行的主窗口。

③ 帮助按钮。单击可以获得联机帮助, 如图 1.1.3 所示。

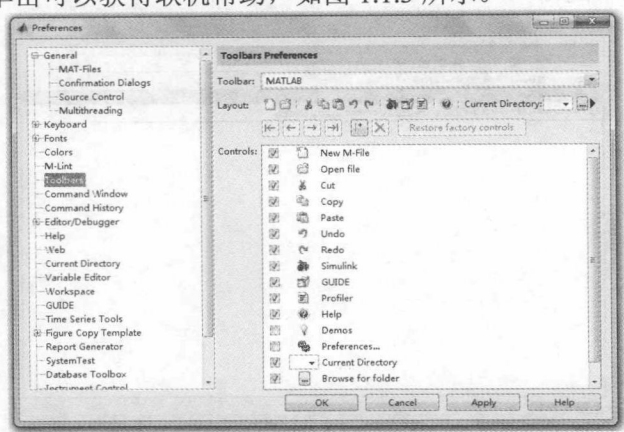


图 1.1.2 桌面工具配置窗口

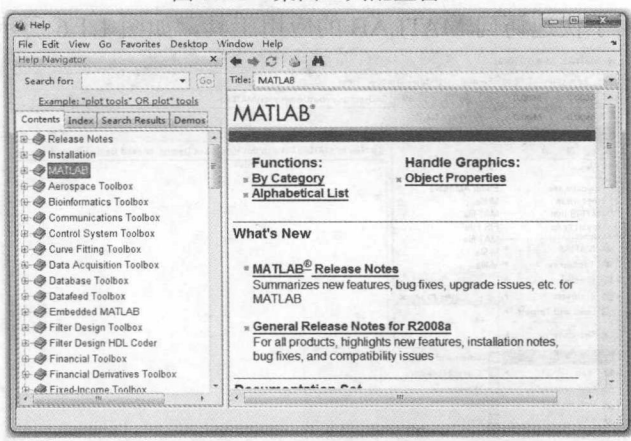


图 1.1.3 帮助主窗口

④ 当前路径。查看、设置和修改当前工作路径。另外, 可以单击菜单的“File”→“Set Path...”命令项, 在弹出的如图 1.1.4 所示窗口中添加、删除工作路径。

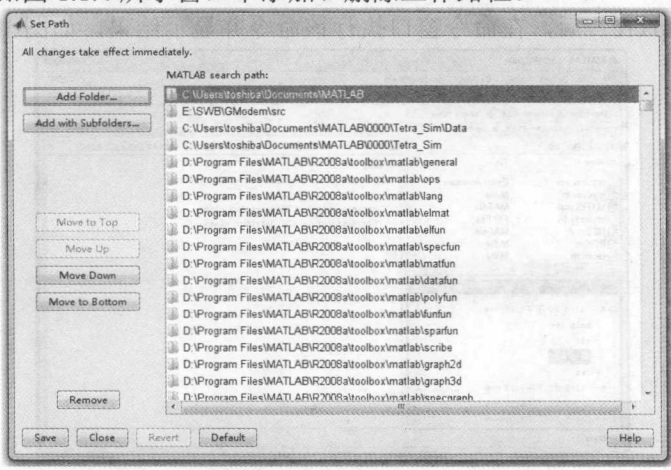


图 1.1.4 工作路径设置窗口



⑤ 窗口显示状态按钮。实现窗口的移动、最大化、最小化或者关闭。例如，最大化命令窗口，结果如图 1.1.5 所示。

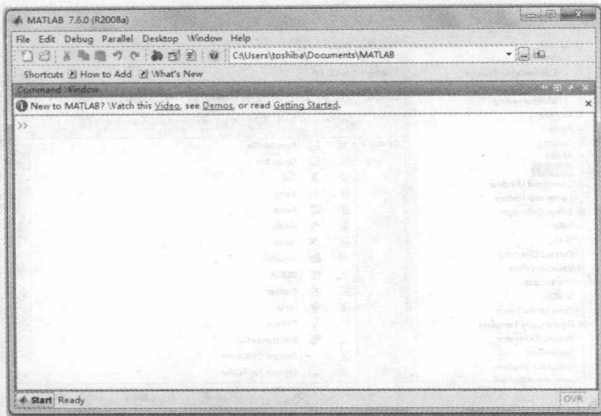


图 1.1.5 最大化命令窗口

⑥ 启动按钮。用于快速访问 MATLAB 的应用工具，如图 1.1.6 所示。

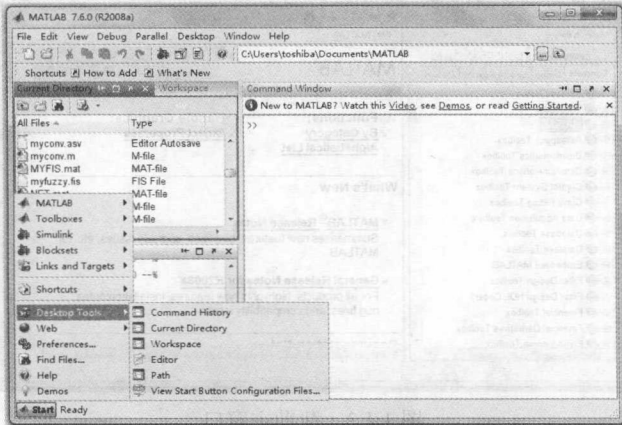


图 1.1.6 启动按钮弹出菜单

⑦ 历史操作窗口。用于查看和执行以前执行过的命令和操作。例如，在历史操作窗口单击以前的操作“help crc”后，在命令窗口执行该命令并显示结果，如图 1.1.7 所示。

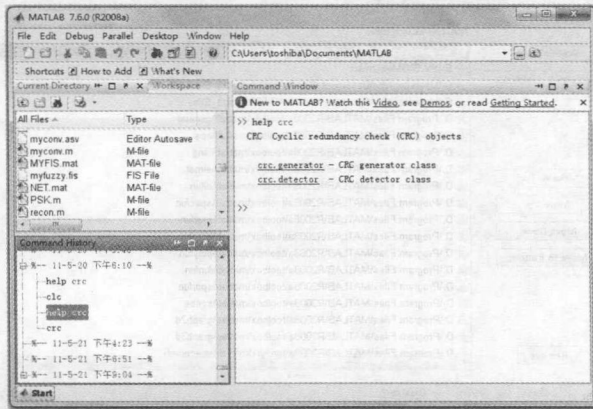


图 1.1.7 历史操作窗口的使用

⑧ 工作空间窗口。显示当前路径下的文件以及工作空间的变量，如图 1.1.8 和图 1.1.9 所示。

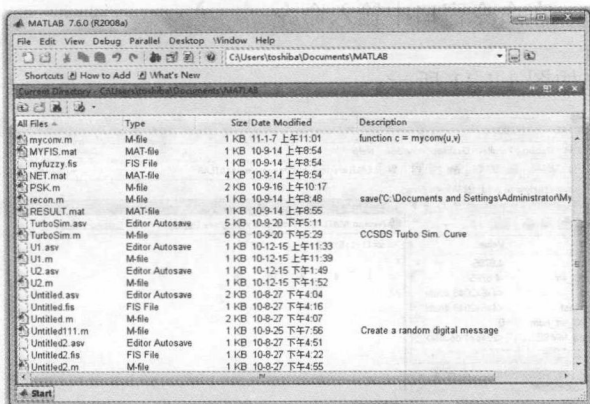


图 1.1.8 当前路径文件列表

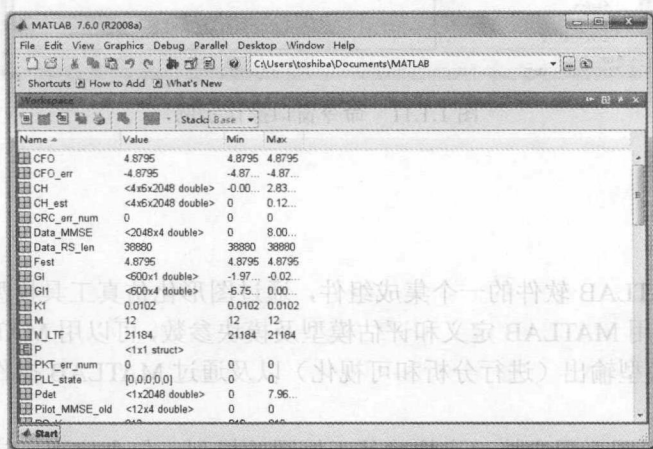


图 1.1.9 工作空间变量信息列表

⑨ 分割线。拖动可以改变窗口大小，如图 1.1.10 所示。

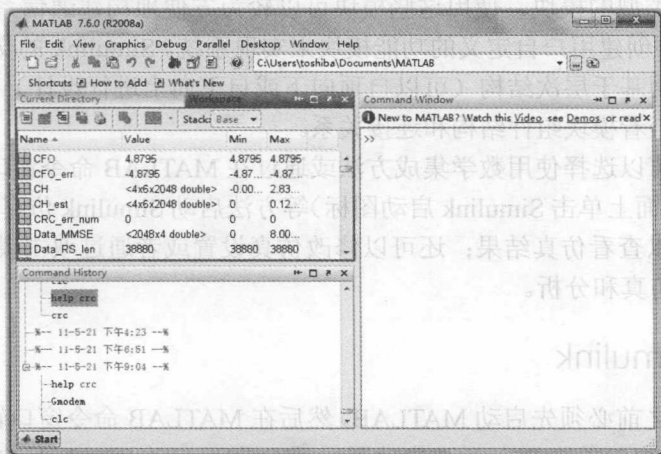


图 1.1.10 改变窗口大小

⑩ 命令窗口。在命令提示符“>>”后输入符合 MATLAB 编程语法规则的语句，执行后在窗口显示结果。例如，在命令窗口中输入命令

```
x=(1:1:5).^2
```

回车后显示输出结果，如图 1.1.11 所示。

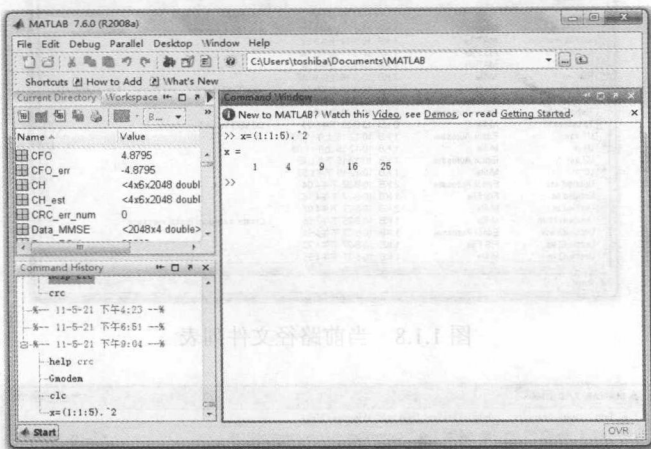


图 1.1.11 命令窗口操作示例

1.2 Simulink

Simulink 是 MATLAB 软件的一个集成组件，通过图形化仿真工具模型化、仿真和分析动态系统。Simulink 使用 MATLAB 定义和评估模型及模块参数，可以用 MATLAB 定义 Simulink 的模型输入、存储模型输出（进行分析和可视化）以及通过 MATLAB 操作符和函数在模型内实现指定功能。

Simulink 提供了图形用户接口来构建基于框图的模型，包含信源、信宿、线性和非线性组件以及连接器等多种模块库。针对通信系统设计和仿真，Simulink 提供的通信模块库中包含通信信源和信宿、信源编码、差错控制（信道编码）、交织、调制、滤波、信道、同步、均衡及辅助工具等多种类型的模块，使用这些模块可以较为方便地构建通信系统并进行仿真。另外，还可以根据需求创建用户自定义的功能模块来实现系统。Simulink 的交互式图形环境简化了模型化过程，采用基于层次结构（可以自顶向下或自下向上进行设计）的 Simulink 模型设计方法，可以逐层查看模块组件结构和连接关系。

在定义模型后可以选择使用数学集成方法或通过 MATLAB 命令窗口输入命令（或者直接在 MATLAB 主界面上单击 Simulink 启动图标）等方法启动 Simulink 仿真过程。使用 Simulink 提供的显示模块可以查看仿真结果；还可以修改仿真设置或者通过双击模块修改参数配置，按照用户要求进行仿真和分析。

1.2.1 启动 Simulink

启动 Simulink 之前必须先启动 MATLAB，然后在 MATLAB 命令窗口输入命令“simulink”或者在 MATLAB 主窗口单击 Simulink 启动图标，即可启动 Simulink 软件。

启动后的 Simulink 主界面如图 1.2.1 所示。