



“十三五”普通高等教育本科规划教材

MATLAB

在电气工程中的应用实例

(第二版)

李维波 编著



中国电力出版社
CHINA ELECTRIC POWER PRESS



—59
高等教育本科规划教材

MATLAB

在电气工程中的应用实例

(第二版)

编著 李维波

主审 吴旭升



中国电力出版社
CHINA ELECTRIC POWER PRESS

内 容 提 要

本书为“十三五”普通高等教育本科规划教材。

全书分为两篇共9章,主要介绍 MATLAB 在数值分析、绘图、信号与系统、电子电路、自动控制、电机及其控制、电力电子装置和电力系统中的应用实例及有关 MATLAB 的建模方法和分析技巧。本书既介绍了一些典型模块库中的功能模块的使用技巧、构建方法和设计技术,也讲解了电气工程中所涉及的一些重要电路的计算机仿真技术,使初学者能够快速完成各个单元电路的分析、仿真和筛选,包括电气参数的优选和整个功能电路的设计、配合以及全部电路的连接与调试。

本书可作为高等学校电气信息类及相关专业的教材,也可作为从事电气信息、计算机仿真方面的工程技术人员和科研人员的参考用书。

图书在版编目(CIP)数据

MATLAB 在电气工程中的应用实例/李维波编著. —2 版. —北京:中国电力出版社,2016.6

“十三五”普通高等教育本科规划教材

ISBN 978-7-5123-9307-3

I. ①M… II. ①李… III. ①电气工程-Matlab 软件-高等学校-教材 IV. ①TM-39

中国版本图书馆 CIP 数据核字(2016)第 096553 号

中国电力出版社出版、发行

(北京市东城区北京站西街 19 号 100005 <http://www.cepp.sgcc.com.cn>)

北京市同江印刷厂印刷

各地新华书店经售

*

2009 年 6 月第一版

2016 年 6 月第二版 2016 年 6 月北京第四次印刷

787 毫米×1092 毫米 16 开本 20.75 印张 508 千字

定价: 42.00 元

敬告读者

本书封底贴有防伪标签,刮开涂层可查询真伪
本书如有印装质量问题,我社发行部负责退换

版权专有 翻印必究

前 言

MATLAB是由美国 Mathworks 公司于 1984 年正式推出的，从那时到现在已升级到 MATLAB 8.5 版本。随着版本的升级，内容不断扩充，功能更加强大。特别是在美国和其他发达国家大学教学和科学研究中，它已经成为线性代数、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、电路与系统、电力电子技术等重要课程的最常用且必不可少的工具，也被越来越多的学生所接受，将其作为自己有力的学习工具。

本书是配合《MATLAB 在电气工程中的应用（第二版）》（以下简称《应用》）而编写的实例教材，随着 MATLAB 的升级而重新修订，即以 8.5 版本为对象进行介绍。在编写本书过程中，坚持“它是理论联系实际的实用教程”的视角定位，坚持以有效培养读者的分析问题、解决问题的能力为宗旨，特制订了“它既是《应用》的学习基础与拓展，又是《应用》的补充与实例应用”的编写总纲，在撰写过程中，其实例选取原则与分析方法方面与《应用》教材相同，也始终贯彻“继承性强、实践性强、针对性强”的要求，自始至终坚持精心准备、悉心组织、细心安排的写作态度。

本书以 MATLAB 在数值分析、绘图、信号与系统、电子电路、自动控制、电机及其控制、电力电子装置和电力系统等方面的应用实例为主线，因此在实例选编方面，既遵循循序渐进原则，但又强调各个章节所用实例的完整性和典型性；在内容的设置方面，充分体现了电气工程专业的特色，所举实例具有针对性强、种类丰富、解析详细等特点；在实例来源方面，既强调实例的代表性，也注重实例的工程实践性，既突出实例的实用性和可借鉴性，又凸显对读者分析问题和解决问题能力的培养。

本书由李维波编写，其内容架构介绍如下：

(1) 第 1 篇，基础学习篇，它包括以下四个章节：

- 1) 第 1 章 MATLAB 的简单编程实例；
- 2) 第 2 章 MATLAB 在数值分析方面的应用实例；
- 3) 第 3 章 MATLAB 在绘图方面的应用实例；
- 4) 第 4 章 MATLAB 在信号与系统中的应用实例。

(2) 第 2 篇，应用提高篇，它包括以下五个章节：

- 1) 第 5 章 MATLAB 在电子电路中的应用实例；
- 2) 第 6 章 MATLAB 在自动控制中的应用实例；
- 3) 第 7 章 MATLAB 在电机及其控制中的应用实例；
- 4) 第 8 章 MATLAB 在电力电子装置中的应用实例；
- 5) 第 9 章 MATLAB 在电力系统中的应用实例。

本书由海军工程大学电气学院副院长吴旭升教授审阅，并提出许多宝贵的修改意见，作者对此表示诚挚的谢意。

在重新编写本书过程中，对给予作者帮助的其他专家教授、所引参考书目的作者和中国电力出版社，致以最真诚的谢意和深深的敬意！

限于作者水平，加之条件有限，书中难免会有不妥或者错误之处，恳请读者和同行批评指正！

编 者

2016.4

目 录

前言	
第 1 篇 基础学习篇	1
第 1 章 MATLAB 的简单编程实例	1
1.1 简单实例分析	1
1.2 可视化命令函数的使用方法	15
1.3 MATLAB 中 Format 介绍	25
第 2 章 MATLAB 在数值分析方面的应用实例	27
2.1 常用命令函数	27
2.2 微积分的命令函数	28
2.3 求解方程式(组)的命令函数	28
2.4 用于统计的命令函数	34
2.5 矩阵分解运算的命令函数	34
2.6 符号运算的命令函数	38
2.7 乘方运算的命令函数	44
2.8 函数极值的命令函数	46
2.9 实例分析	50
第 3 章 MATLAB 在绘图方面的应用实例	53
3.1 符号函数绘图命令函数	53
3.2 绘制网格图的命令函数	54
3.3 绘制表面图的命令函数	55
3.4 图形后续处理的典型方法	56
3.5 综合应用实例分析	59
第 4 章 MATLAB 在信号与系统中的应用实例	77
4.1 基本命令函数	77
4.2 基本信号表示方法	82
4.3 信号基本运算方法	89
4.4 系统时频域的分析与建模	92
4.5 滤波器的分析与建模	101
第 2 篇 应用提高篇	109
第 5 章 MATLAB 在电子电路中的应用实例	109
5.1 电路的分析与建模	109
5.2 数字电路的分析与建模	113
5.3 直流稳态电路的分析与建模	118

5.4	动态电路的分析与建模	120
5.5	交流稳态电路的分析与建模	126
第6章 MATLAB在自动控制中的应用实例		136
6.1	系统时频域特性的分析与建模	136
6.2	系统复频域特性的分析与建模	141
6.3	离散系统Z域特性的分析与建模	145
6.4	系统状态变量的分析与建模	148
6.5	典型实例的分析与建模	153
第7章 MATLAB在电机及其控制中的应用实例		176
7.1	触发器的分析与建模	176
7.2	变压器的分析与建模	180
7.3	逆变器的分析与建模	186
7.4	电机的分析与建模	196
7.5	数据处理的分析与建模	208
7.6	直流电动机的分析与建模	211
7.7	电机模块库介绍	220
第8章 MATLAB在电力电子装置中的应用实例		224
8.1	SimPowerSystems 模块库介绍	224
8.2	简单实例的分析与建模	234
8.3	三相桥式全控整流电路的分析与建模	239
8.4	ZVS PWM DC/DC 变换器的分析与建模	251
8.5	AC/DC 变换器的分析与建模	257
8.6	变频控制下电压品质的分析与建模	266
第9章 MATLAB在电力系统中的应用实例		274
9.1	电流互感器的分析与建模	274
9.2	电流互感器饱和特性建模分析	281
9.3	电力变压器试验装置的分析与建模	284
9.4	传输线影响特性的分析与建模	293
9.5	小电流接地系统的分析与建模	307
9.6	同步发电机突然对地短路的分析与建模	319
参考文献		324

第1篇 基础学习篇

本书以 MATLAB8.5 (R2015a) 版本为讲述对象, 讲解它的基本应用方法和操作技巧, 包括一些典型命令函数的语法特点、使用环境和应用规律, 使初学者能够借助本篇的学习, 深入理解后续章节的内容, 并奠定必要的知识与方法。

MATLAB 中有数千个命令函数, 而且很多命令函数有多种调用方式和语法格式, 要想全面掌握 MATLAB 非常困难, 即使只掌握 MATLAB 的基本命令函数, 也不是一件轻松的事情。尤其是对于初学者来说, 更是如此。为了学习 MATLAB 需注意以下几点:

- (1) 在学习中, 要做到学中练、练中学, 这样学习起来有兴趣、信心强、上手快;
- (2) 尽可能在英文输入状态而非中文输入状态, 进行 MATLAB 语句的录入和编程工作, 以减少调试过程中遇到的一些莫名其妙的错误;
- (3) 矩阵(数组)是 MATLAB 的基本数据结构, 在遇到问题时, 要尽量采用矩阵(数组)结构表示数学模型, 这样在 MATLAB 中使用起来方便、快捷和高效;
- (4) MATLAB 有两种帮助形式, 即一种在命令窗口使用帮助命令搜索帮助; 另一种是使用 MATLAB 的帮助系统, 从菜单 help 进入, 单击 MATLAB help 即可出现该系统, 这对于快速学习和掌握 MATLAB 至关重要。

第1章 MATLAB 的简单编程实例

MATLAB 在数值计算、数据分析和图形处理等方面, 功能强大, 通过简单程序的设计与调试, 即可实现。为此希望读者朋友快速阅读本章内容, 以求达到以下两点:

- (1) 掌握 MATLAB 编程操作的基本方法;
- (2) 熟悉 MATLAB 程序设计的基本方法。

1.1 简单实例分析

1.1.1 基本认识

本书是以 MATLAB 8.5 版本为例, 进行示例分析的。MATLAB 的操作界面如图 1-1 所示。MATLAB 编程中需注意以下几点:

- (1) 在 MATLAB 中, 百分比符号 (%) 之后的文字, 被视为注解 (Comments) 内容, 程序不会执行;
- (2) 在 MATLAB 的程序编制过程中, 各个标点符号, 如逗号 (,)、单引号 (' ') 和分号 (;), 均需要在英文输入状态中录入, 否则会出现一些奇奇怪怪的“错误问题”;
- (3) 认识 MATLAB 中的一般符号, 如加 (+)、减 (-)、乘 (*)、除 (/) 的数学运

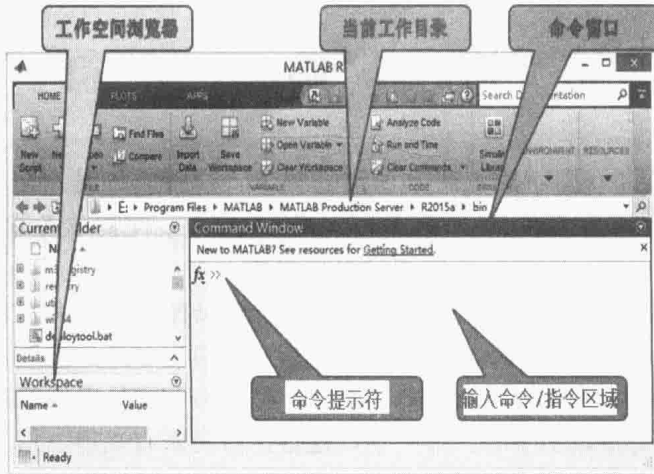


图 1-1 MATLAB 的操作界面

算符号，以及幂次运算 (^)；

(4) “>>”是 MATLAB 的提示符号 (Prompt)，又称标识符。

1.1.2 实例分析

在 MATLAB 8.5 版本中，要打开它的编程窗口的操作界面，其基本操作步骤如图 1-2 所示。

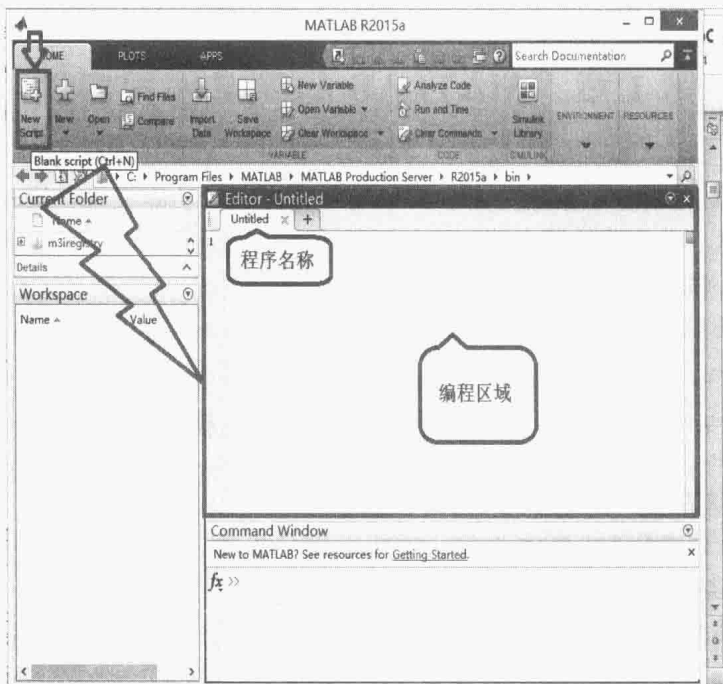


图 1-2 打开 MATLAB 的编程窗口的基本操作步骤

首先，将鼠标放置在 New Script 按钮处→弹出“Blank Script (Ctrl+N)”空编程文档

提示语;

其次, 单击 New Script 按钮→弹出 Untitled 窗口未命名的编程空文档。

[实例 1] 对于一个系统的传递函数的表达式为

$$H(s) = \frac{7s}{s^2 + 6s + 205} \quad (1-1)$$

试分析它的零点-极点图、伯德图和阶跃响应曲线。

分析: 新建一个编程空文档, 并将其命名为 exam_1.m, 其语句如下:

```
num = [7 0]; % 定义传递函数的分子部分
den = [1 6 205]; % 定义传递函数的分母部分
figure(1) % 生成图 1: figure 1
pzmap(num, den); grid on % 绘制零点-极点图
figure(2); % 生成图 2: figure 2
bode(num, den, 'bd'); grid on % 绘制阶跃响应图
figure(3); % 生成图 3: figure 3
step(num, den, 'r.*'); grid on % 绘制阶跃响应图
```

输入上述语句之后, MATLAB 的编程操作界面如图 1-3 所示。

实例 1 的执行结果如图 1-4 所示, 其中图 1-4 (a) 表示零点-极点图, 图 1-4 (b) 表示伯德图, 图 1-4 (c) 表示阶跃响应曲线。

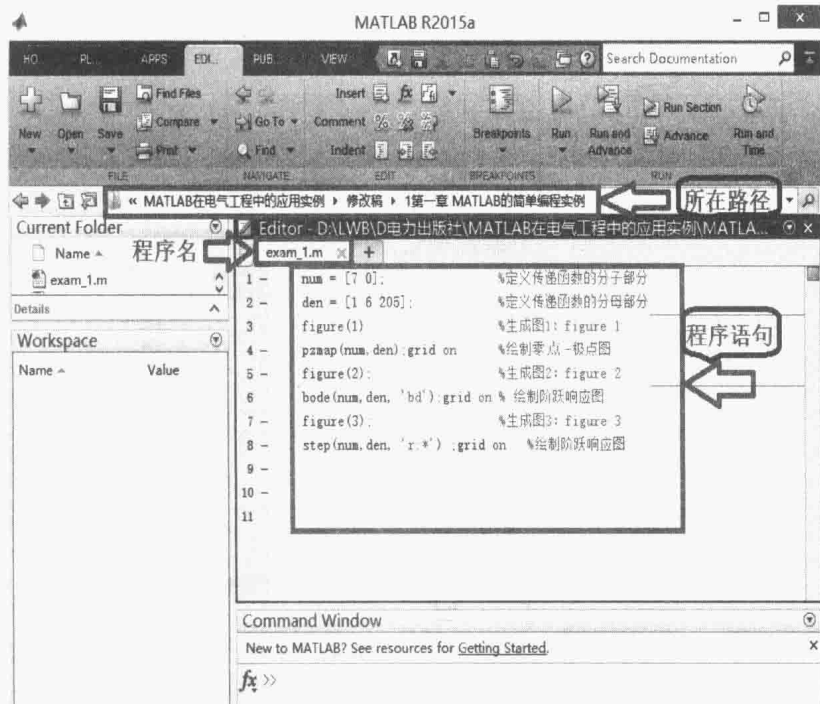


图 1-3 实例 1 的编程操作界面

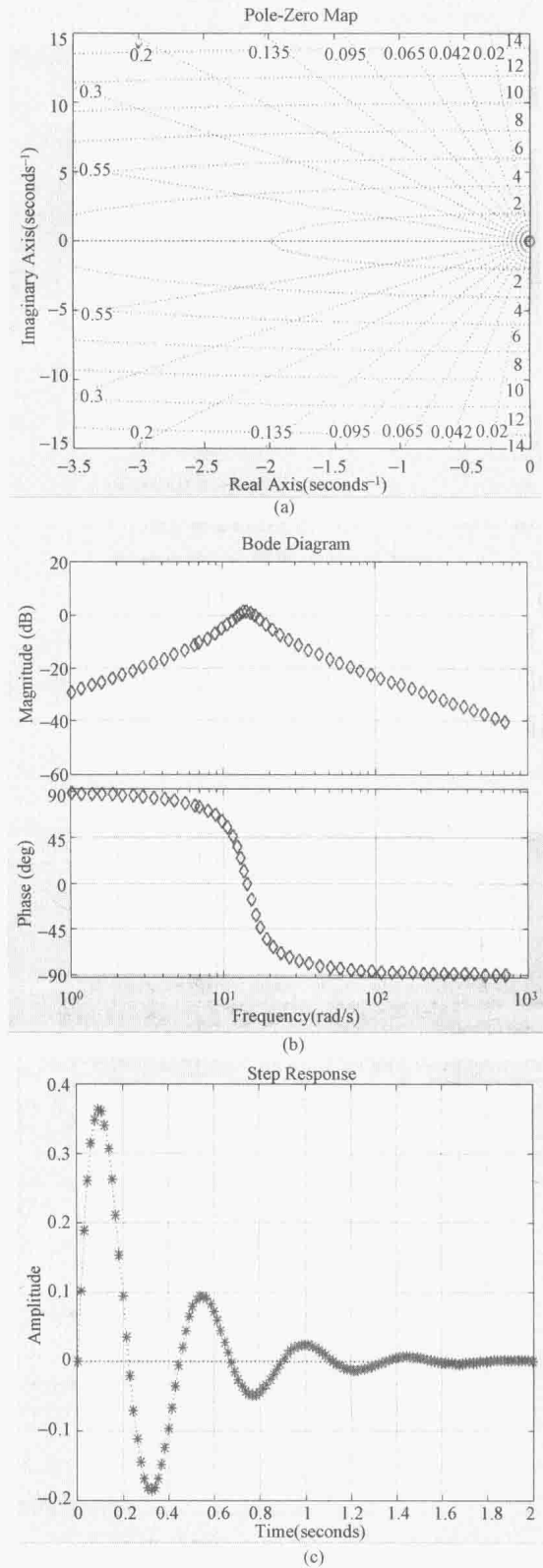


图 1-4 实例 1 的执行结果

(a) 零点-极点图; (b) 伯德图; (c) 阶跃响应图

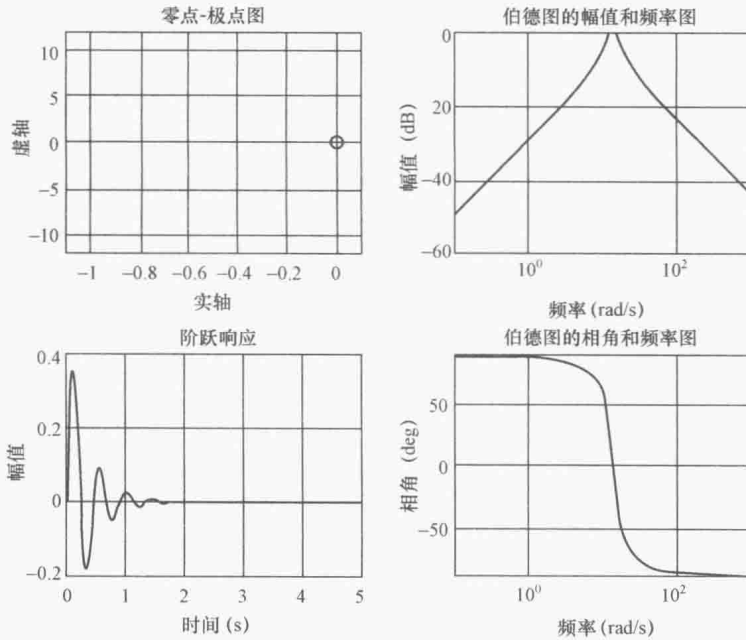


图 1-5 实例 2 的执行结果

```
while(sum <= 5000)
n = n + 1;
sum = sum + n;
end
n - 1
```

实例 3 的执行结果:

```
n = 99
```

[实例 4] 编写程序, 用 for 循环结构编写程序, 求 2 的 0 到 25 次幂的和。

分析: 新建一个编程空文档, 并将其命名为 exam_4.m, 其语句如下:

```
sum = 0;
for n = 0: 25
sum = 2^(n) + sum;
n = n + 1;
end
sum
```

实例 4 的执行结果:

```
sum =
```

```
67108863
```

[实例 5] 编写程序, 用 while 循环结构编写程序, 求 2 的 0 到 25 次幂的和。

分析: 新建一个编程空文档, 并将其命名为 exam_5.m, 其语句如下:

```
n = 0;
sum = 0;
while(n < 26)
```

```
sum = sum + 2^(n);
n = n + 1;
end
sum
```

实例 5 的执行结果:

```
sum =
```

```
67108863
```

[实例 6] 编写程序, 建立矩阵 A , 然后找出大于 4 的元素的位置。其中, 已知矩阵 $A = [4, -65, -54, 0, 6; 56, 0, 67, -45, 10]$ 。

分析: 新建一个编程空文档, 并将其命名为 exam_6.m, 其语句如下:

```
clc; clear;
A = [4, -65, -54, 0, 6; 56, 0, 67, -45, 10];
A
find(A > 4)
```

实例 6 的执行结果如图 1-6 所示。通过本例的学习, 可以清楚地知道 find 命令的使用方法。

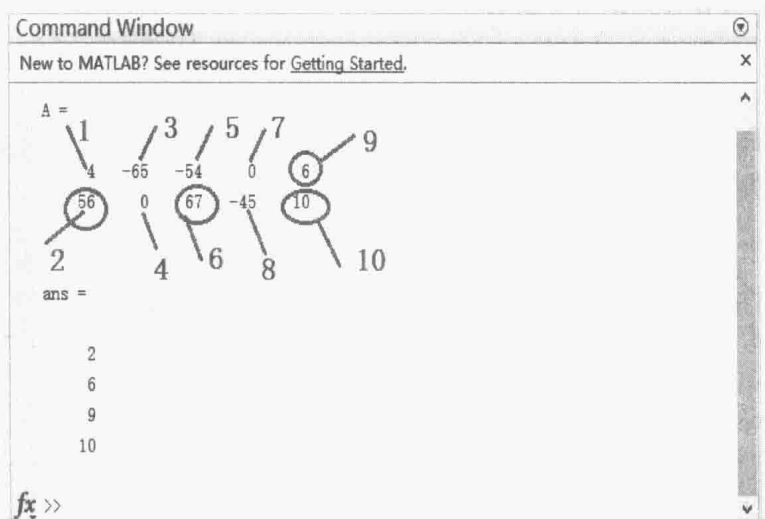


图 1-6 实例 6 的执行结果

[实例 7] 根据控制原理, 闭环系统的控制框图如图 1-7 所示, 其传递函数的表达式为

$$H(s) = \frac{Y(s)}{X(s)} = \frac{G(s)}{1 + KG(s)} \quad (1-2)$$

分析: 为获取该控制系统的根轨迹图, 可以通过编程实现。假设 $H(s)$ 的表达式同实例 1, 其程序保存为 exam_7.m, 即

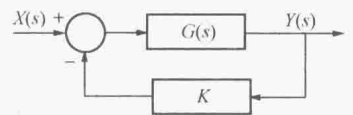


图 1-7 闭环系统框图

```
clc; clear;
num = [7 0];           % 定义分子
den = [1 6 205];      % 定义分母
```

```
rlocus(num, den);           % 绘制根轨迹图
grid on
```

实例 7 的执行结果如图 1-8 所示。

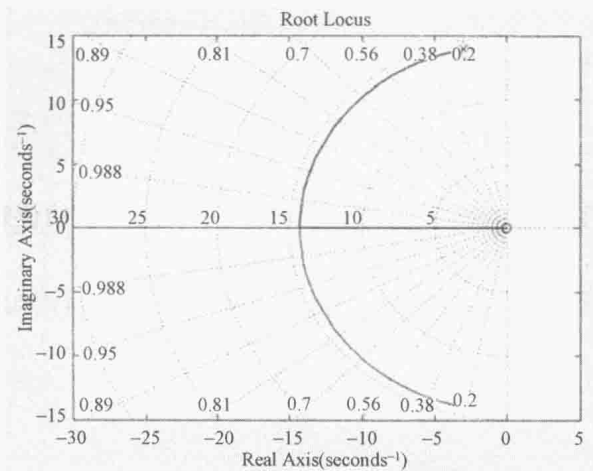


图 1-8 实例 7 的执行结果

[实例 8] 假设某系统传递函数为

$$H(s) = \frac{7s}{s^2 + 6s + 205} \quad (1-3)$$

试分析它在输入信号 $x(t) = \sin(0.5\pi t)$ 作用下的响应情况，其中 t 介于 $0 \sim 10$ 。

分析：实例 8 可以通过编程实现，该程序保存为 exam_8.m，即

```
clc; clear;
num = [7 0];           % 定义传递函数的分子部分
den = [1 6 205];      % 定义传递函数的分母部分
t = linspace(0, 10, 401); % 时间 t 介于 0~10 之间，共计 401 个矢量
u = sin(0.5 * pi * t); % 给定输入信号
[y, x] = lsim(num, den, u, t); % 计算输出响应
subplot(2, 1, 1)
plot(t, y, 'r', t, u, 'b-.', 'linewidth', 3); % 绘制响应曲线(红色)和输入曲线(蓝色)
grid on
title('x(t)作用下的响应情况');
xlabel('时间(s)'); ylabel('幅值');
legend('y(t)', 'x(t)') % 标注曲线
subplot(2, 1, 2)
plotyy(t, u, t, y); % 绘制响应曲线和输入曲线
grid on
title('x(t)作用下的响应情况');
xlabel('时间(s)'); ylabel('幅值');
legend('y(t)', 'x(t)') % 标注曲线
```

实例 8 的执行结果如图 1-9 所示，分析发现利用 plot 命令函数绘制的图如图 1-9 (a) 所示，该曲线 $y(t)$ 在 $0 \sim 1$ 有一个跳跃状态，但不是很明显。对比而言，利用 plotyy 命令函数

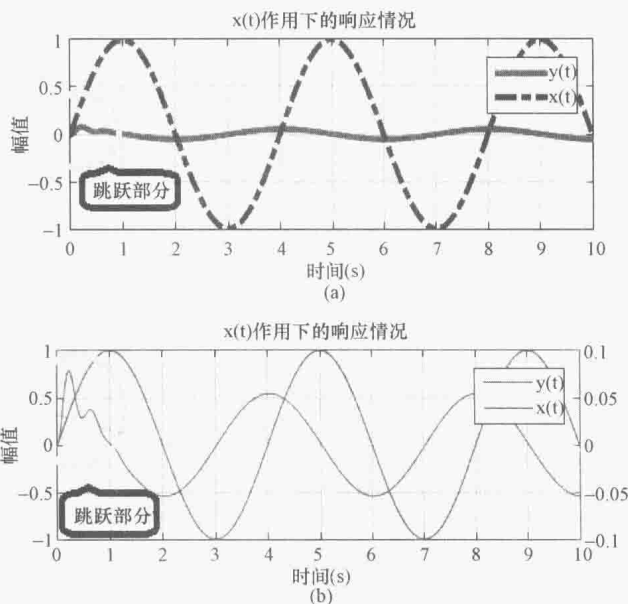


图 1-9 实例 8 的执行结果

(a) 利用 plot 命令函数绘制的图；(b) 利用 plotyy 命令函数绘制的图

绘制的图如图 1-9 (b) 所示, 该曲线 $y(t)$ 在 $0 \sim 1$ 有一个明显的跳跃状态, 且被非常清楚地显示出来。

两幅图为什么存在如此大的区别? 其原因在于, plot 所绘制的曲线中 x 和 y 坐标尺度是一样的, 但是 plotyy 所绘制的曲线中 x 和 y 坐标尺度因曲线特性不同而不同。plotyy 命令函数的使用方法为

命令格式: `plotyy(x1, y1, x2, y2)`

解释说明: plotyy 命令函数是具有两个纵坐标标度的图形的命令函数, 其中 $x1$ 、 $y1$ 对应一条曲线, $x2$ 、 $y2$ 对应另一条曲线。 $x1$ 和 $x2$ 的横坐标的标度相同, 纵坐标有两个 $y1$ 和 $y2$, 左边的纵坐标用于 $x1$ 、 $y1$ 数据对, 右边的纵坐标用于 $x2$ 、 $y2$ 数据对。

[实例 9] 计算 sinc 函数。已知 sinc 函数的表达式为

$$f(t) = \text{sinc}(t) = \frac{\sin t}{t} (t = \pm 10\pi) \quad (1-4)$$

分析: 新建一个编程空文档, 并将其命名为 exam_9.m, 其语句如下:

```
clc; clear;
t=-10*pi:0.01*pi:10*pi; % 定义时间范围向量 t
f=sinc(t/pi); % 计算 sinc 函数
plot(t, f, 'r', 'linewidth', 3); % 绘制 sinc 的波形
xlabel('时间(s)'); ylabel('幅值'); % 标注曲线
title('sinc 函数曲线'); grid on
```

实例 9 的执行结果如图 1-10 所示。

[实例 10] 计算叠加信号函数, 这在信号组建、信号模拟方面应用较广。已知函数 $f(t)$ 的表达式为

$$f(t) = 6\sin 18\pi t + 20\cos 20\pi t - 3 \quad (1-5)$$

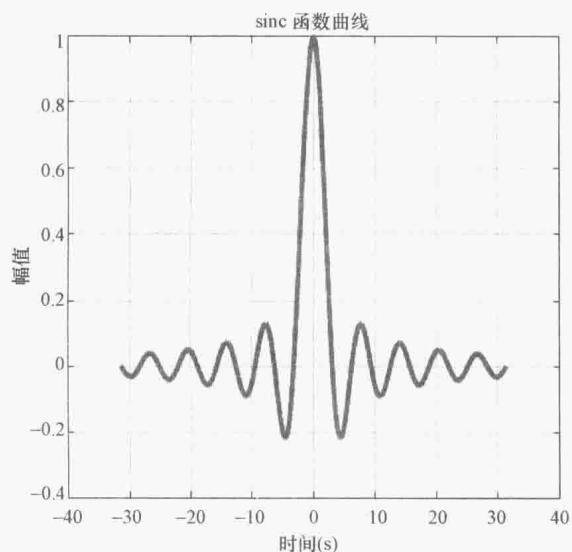


图 1-10 实例 9 的执行结果

分析：新建一个编程空文档，并将其命名为 exam_10_1.m，其语句如下：

```
clc; clear;
syms t; % 定义符号变量 t
f = 6 * sin(18 * pi * t) + 20 * cos(20 * pi * t) - 3; % 计算符号函数 f(t)
ezplot(f, [0 pi]); % 绘制 f(t) 的波形
xlabel('时间(s)'); ylabel('幅值');
grid on
```

程序 exam_10_1.m 的执行结果如图 1-11 所示。

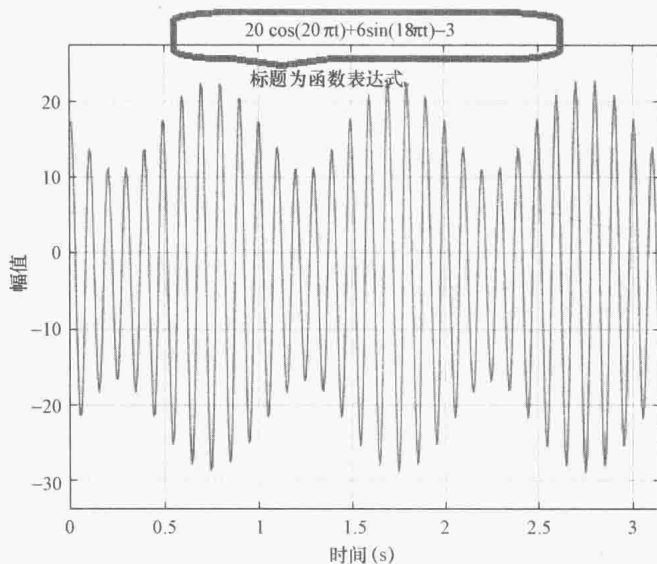


图 1-11 程序 exam_10_1.m 的执行结果

新建一个编程空文档，并将其命名为 exam_10_2.m，其语句如下：