

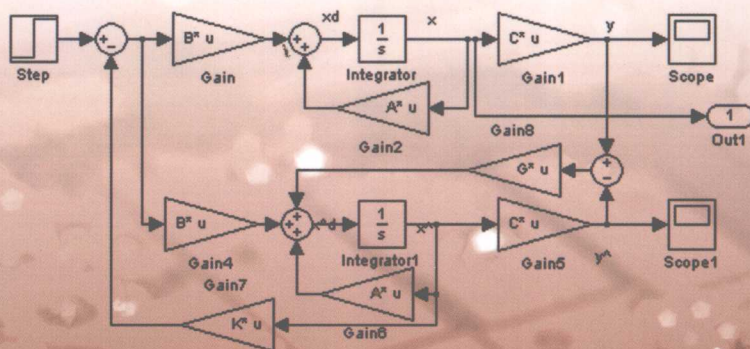


DIANQI
XINXILEI

普通高等教育“十一五”电气信息类规划教材

自动控制系统 计算机仿真

■ 张晓江 黄云志 编著



机械工业出版社
CHINA MACHINE PRESS



免费
电子课件

普通高等教育“十一五”电气信息类规划教材

自动控制系统计算机仿真

合肥工业大学 张晓江 黄云志 编 著
 张崇巍 主 审



机械工业出版社

MATLAB 及其模块化仿真部件 Simulink 是优秀的数值计算和系统仿真软件。本书在充分考虑自动化专业课程设置的情况下,以 MATLAB 语言 (MATLAB 7.0/Simulink 6.0) 为主要工具,较为全面地介绍了自动控制系统的建模、分析、仿真与设计的基本原理和方法。全书共分 8 章,内容包括自动控制系统仿真概述、控制系统计算机数字仿真基础、MATLAB 语言的基础知识、控制系统数学模型及其转换、Simulink 在系统仿真中的应用、自动控制系统计算机辅助分析、自动控制系统计算机辅助设计、电力系统工具箱及其应用实例。

本书是“普通高等教育‘十一五’电气信息类规划教材”。本书的显著特点是注重介绍仿真的应用和实例,在阐述控制系统仿真原理的同时,通过大量的有代表性的例题来讲解相应内容,使学生感觉生动有趣,不枯燥,便于学生掌握所学内容。本书配有电子课件,欢迎选用本书作教材的老师登录 www.cmpedu.com 注册下载或发邮件到 wbj@cmpbook.com 索取。

本书可作为大学本科自动化专业以及电气信息类其他专业的专业课教材,也可供相关领域的工程技术和研究人员参考。

图书在版编目 (CIP) 数据

自动控制系统计算机仿真/张晓江,黄云志编著. —北京:机械工业出版社,2009.4

普通高等教育“十一五”电气信息类规划教材

ISBN 978-7-111-26648-8

I. 自… II. ①张…②黄… III. 自动控制系统-计算机仿真-高等学校-教材 IV. TP273

中国版本图书馆 CIP 数据核字 (2009) 第 042827 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑:王保家 责任编辑:王保家 谷玉春 版式设计:霍永明

责任校对:申春香 封面设计:张静 责任印制:邓博

北京中兴印刷有限公司印刷

2009 年 5 月第 1 版第 1 次印刷

184mm×260mm·11 印张·265 千字

标准书号:ISBN 978-7-111-26648-8

定价:21.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

销售服务热线电话:(010) 68326294

购书热线电话:(010) 88379639 88379641 88379643

编辑热线电话:(010) 88379727

封面无防伪标均为盗版

前 言

自动控制系统仿真是近 20 多年来发展起来的一门新兴技术。随着计算机科学与技术的高速发展,计算机的运行速度越来越快,功能日益强大,价格日趋大众化,目前计算机已经十分普及。在此基础之上,控制系统计算机仿真已经成为对自动控制系统进行分析、设计和综合研究中的一种常规手段。随着控制系统的日益复杂,控制功能和任务多样化,传统的控制系统分析方法已经无法胜任。使用计算机进行自动控制系统的分析计算和仿真研究,已经成为从事自动控制以及相关专业的研究人员和工程技术人员所必须掌握的一门技术。

在工业、农业、交通运输、国防军事、科学研究等领域,都离不开自动控制系统与装置。控制理论与控制工程已经成为现代科学技术中不可缺少的重要组成部分,而自动控制系统计算机仿真技术则是自动控制系统建模、分析和设计过程中极其重要的工具。计算机的发展(包括计算机仿真技术的发展)不仅在技术层面,而且在理论层面,都深刻地影响着控制理论与控制工程学科的发展。相信在未来,计算机技术必将在更大程度上改变控制系统分析和设计的理论与方法。

MATLAB 语言是美国 Math Works 公司的产品,是优秀的控制系统仿真软件。历经 20 多年的发展,几乎每年都有升级的版本推出,对其不断充实和改进。本书以 MATLAB 7.0/Simulink 6.0 为背景,较为全面地介绍了 MATLAB/Simulink 的基础知识及其常用的工具箱在控制系统仿真中的应用。

本书是按照教育部本科自动化专业教学大纲编写而成的。教学参考学时约为 36 学时。本书的内容是作者多年来从事控制系统计算机仿真课程教学和研究的经验总结。在选材上,力争做到内容全面充实,重点突出,兼顾基本理论方法和实际应用。全书共分 8 章:第 1 章为自动控制系统仿真概述;第 2 章为控制系统计算机数字仿真基础;第 3 章为 MATLAB 语言的基础知识;第 4 章为控制系统数学模型及其转换;第 5 章为 Simulink 在系统仿真中的应用;第 6 章为自动控制系统计算机辅助分析;第 7 章为自动控制系统计算机辅助设计;第 8 章为电力系统工具箱及其应用实例。带星号(*)的章节为选学内容。其中,张晓江副教授编写第 1、5、6、7、8 章,并且负责全书统稿;黄云志副教授编写第 2、3、4 章。全书由张崇巍教授主审,他对本书提出了许多宝贵意见,在此表示衷心感谢。在本书的编写过程中,还得到了合肥工业大学电气与自动化工程学院的支持,在此也表示感谢。

本书为“普通高等教育‘十一五’电气信息类规划教材”。为了配合课堂教学,本书配有电子课件,由张晓江副教授和黄云志副教授分别制作。欢迎选用本书作教材的老师登录 www.cmpedu.com 注册下载或发邮件到 wbj@cmpbook.com 索取。

由于编者水平有限,疏漏及谬误之处在所难免,希望同行及读者不吝赐教。

编 者

目 录

前言

第1章 自动控制系统仿真概述 1

- 1.1 自动控制系统简介 1
 - 1.1.1 系统与自动控制系统 1
 - 1.1.2 自动控制系统建模 1
- 1.2 自动控制系统仿真的基本概念 2
 - 1.2.1 仿真的定义 2
 - 1.2.2 自动控制系统仿真的分类 2
 - 1.2.3 自动控制系统仿真的过程 3
- 1.3 仿真技术在控制系统设计中的应用及其重要意义 3
 - 1.3.1 自动控制理论简介 3
 - 1.3.2 仿真技术与 CAD 在自动控制系统设计中的重要意义 4
 - 1.3.3 仿真技术在自动控制系统设计中的应用现状和发展趋势 4
- 1.4 MATLAB 语言及其在控制系统设计中的应用 4
- 小结 5
- 习题 5

第2章 控制系统计算机数字仿真基础 6

- 2.1 连续系统数值积分方法 6
 - 2.1.1 欧拉法 6
 - 2.1.2 龙格-库塔法 7
 - 2.1.3 数值积分法的稳定性 10
 - 2.1.4 数值积分法的选择 11
- 2.2 控制系统的结构及其描述 12
 - 2.2.1 控制系统中的典型结构 12
 - 2.2.2 控制系统的典型环节 13
 - 2.2.3 控制系统的连接矩阵 14
- 2.3 控制系统的建模 16
- 小结 20
- 习题 20

第3章 MATLAB 语言的基础知识 21

- 3.1 MATLAB 的安装和启动 21
 - 3.1.1 MATLAB 的安装 21

- 3.1.2 MATLAB 7.x 的启动 21
- 3.2 MATLAB 7.x 的系统界面 22
 - 3.2.1 MATLAB 7.x 的系统界面窗口 22
 - 3.2.2 MATLAB 7.x 的菜单项和工具栏 24
 - 3.2.3 MATLAB 7.x 的帮助系统 24
- 3.3 MATLAB 基础知识 25
 - 3.3.1 矩阵的生成 25
 - 3.3.2 变量、常量和语句 28
 - 3.3.3 数值显示格式 29
 - 3.3.4 字符串 30
- 3.4 矩阵的运算 30
 - 3.4.1 矩阵的数学运算 30
 - 3.4.2 矩阵的数组运算 31
 - 3.4.3 矩阵操作 34
 - 3.4.4 矩阵元素的数据变换 35
- 3.5 流程控制结构 36
 - 3.5.1 for 语句 36
 - 3.5.2 while 语句 37
 - 3.5.3 if-else-end 语句 38
 - 3.5.4 switch-case 语句 39
- 3.6 m 文件 39
 - 3.6.1 脚本文件 39
 - 3.6.2 函数文件 40
- 3.7 MATLAB 的绘图功能 41
 - 3.7.1 二维图形的绘制 41
 - 3.7.2 三维图形的绘制 44
 - 3.7.3 图形的输出 45
- 3.8 MATLAB 的应用 45
 - 3.8.1 矩阵的分解 45
 - 3.8.2 多项式处理 46
 - 3.8.3 曲线拟合与插值 46
 - 3.8.4 常微分方程求解 48
- 小结 49
- 习题 50

第4章 控制系统数学模型及其转换 51

- 4.1 控制系统类型 51

4.2 控制系统常用的数学模型	52	分析	97
4.2.1 连续系统数学模型	52	6.1 自动控制系统的稳定性分析	97
4.2.2 离散系统数学模型	54	6.1.1 求取特征方程的根	97
4.2.3 系统模型参数的获取	55	6.1.2 控制系统的能控性和能观性 分析	99
4.3 系统数学模型的转换	55	6.1.3 利用传递函数的极点判别系统 稳定性	100
4.3.1 系统模型向状态方程形式转换	56	6.1.4 利用李亚普诺夫第二法判别 系统稳定性	101
4.3.2 系统模型向传递函数形式转换	56	6.2 控制系统时域分析	102
4.3.3 系统模型向零极点形式转换	57	6.2.1 时域分析的一般方法	102
4.3.4 传递函数形式与部分分式形式的 转换	58	6.2.2 常用时域分析函数	103
4.3.5 连续系统和离散系统之间的 转换	59	6.2.3 时域分析应用实例	104
4.4 控制系统模型的连接	60	6.3 控制系统频域分析	107
4.4.1 模型串联	60	6.3.1 频域分析的一般方法	108
4.4.2 模型并联	60	6.3.2 频域分析应用实例	108
4.4.3 反馈连接	61	6.4 根轨迹分析方法	111
4.5 系统模型的实现	62	6.4.1 幅值条件和相角条件	112
4.5.1 能控标准型	62	6.4.2 绘制根轨迹的常用函数及其 应用实例	113
4.5.2 能观标准型	65	6.5* 基于计算机仿真的非线性定常 控制系统新型稳定性判据	116
4.5.3 对角线标准型	65	6.5.1 问题的提出	116
4.5.4 标准型的实现	65	6.5.2 新型稳定性判据	117
小结	66	6.5.3 在单级倒立摆模糊控制系统 中的应用	119
习题	66	6.5.4 结论和展望	120
第5章 Simulink 在系统仿真中的 应用	68	小结	121
5.1 Simulink 建模的基础知识	68	习题	121
5.1.1 Simulink 6.0 常用模块简介	69	第7章 自动控制系统计算机辅助 设计	122
5.1.2 Simulink 其他工具箱、模块集	79	7.1 概述	122
5.2 Simulink 建模与仿真	79	7.2 超前校正、滞后校正以及滞后-超前 校正的伯德图设计	122
5.2.1 Simulink 建模方法简介	79	7.2.1 超前校正器的伯德图设计	123
5.2.2 仿真算法与控制参数选择	80	7.2.2 滞后校正器的伯德图设计	126
5.2.3 Simulink 在控制系统仿真研究中的 应用举例	83	7.2.3 滞后-超前校正器的伯德图 设计	128
5.3 子系统与模块封装技术	85	7.3 PID 控制器设计	132
5.3.1 子系统概念及构成方法	85	7.3.1 PID 控制器的传递函数	132
5.3.2 模块封装方法	86	7.3.2 PID 控制器各参数对控制 性能的影响	133
5.3.3 模块库构造	90		
5.4 S 函数及其应用	90		
5.4.1 S 函数的基本结构	91		
5.4.2 用 MATLAB 编写 S 函数举例	92		
小结	96		
习题	96		
第6章 自动控制系统计算机辅助			

7.3.3 使用 Ziegler - Nichols 经验整定公 式进行 PID 控制器设计	135
7.4 基于状态空间模型的控制器设计 方法	137
7.4.1 状态空间表达式的基本概念以 及状态方程的解	137
7.4.2 状态反馈极点配置控制器设计 ...	139
7.4.3 状态观测器设计	140
7.4.4 基于状态观测器状态反馈 控制系统	141
7.5* 线性二次型指标最优控制系统 设计	146
7.5.1 线性二次型指标与黎卡提方程 ...	146
7.5.2 设计线性二次型最优控制的 MATLAB 函数	147
7.5.3 最优控制系统设计实例	147
小结	148
习题	148
第 8 章 电力系统工具箱及其应用 实例	149
8.1 SimPowerSystems (电力系统) 模块集简介	149
8.1.1 Electrical Sources (电源) 模块子集	149
8.1.2 Elements (电路元件) 模块 子集	150
8.1.3 Machines (电机) 模块子集	151
8.1.4 Measurements (测量) 模块 子集	152
8.1.5 Power Electronics (电力电子) 模块子集	152
8.1.6 Phasor Elements (相量元件) 模块子集	152
8.1.7 Extra Library (其他模块子集) ...	153
8.2 使用电力系统工具箱进行仿真的 实例	153
小结	166
习题	166
参考文献	167

第 1 章 自动控制系统仿真概述

自动控制系统仿真是一门新兴的技术学科。它已经成为对自动控制系统进行分析、设计与综合研究的一种重要手段。经过数十年的高速发展，今天的计算机无论是硬件配置，还是软件功能都已经达到了一个相当高的水平。在自动控制系统的分析和设计过程中，利用计算机进行仿真实验和研究，已经成为从事控制领域以及相关行业的工程技术和科研人员所必须掌握的一门技术。而且，随着计算机技术的进一步发展，仿真软件的功能将更加强大，计算机仿真对于从事自动控制领域的科技人员将更加重要，成为他们必不可少的工具。

在自动控制系统的设计与分析过程中，有大量繁琐的计算和曲线绘制工作。即使有了计算机这样先进的计算工具，在仿真软件出现之前，人们不得不与计算机程序打交道，而使用通常的计算机语言（如 Basic、C、FORTRAN 等）编写自动控制系统仿真程序并非易事。随着 MATLAB 的出现，它的主程序和它附带的各种工具箱以及 Simulink 仿真工具，为控制系统的设计、计算、图形绘制以及仿真提供了方便快捷的、功能强大的工具，使自动控制系统的设计与仿真方法出现了革命性的变化。目前，MATLAB 已经成为全世界自动控制领域流行的设计与仿真软件。

1.1 自动控制系统简介

1.1.1 系统与自动控制系统

在不同的学科领域中，“系统”的定义是不同的。在控制工程中，系统的定义为：系统是由相互联系、相互作用的物体所形成的具有特定功能和运动规律的有机整体。

自动控制系统是指在没有人直接参与的情况下，利用外加的设备或装置（控制器），使机器、设备或生产过程（被控对象）的工作状态和参数（被控量）自动地按照预定的规律运行。

例如，一个电动机的转速控制系统使转速保持在设定值上而不受负载波动的影响，就是一个自动控制系统。焊接机器人也是一个自动控制系统，它可以按照预先设定的程序沿着一条曲线将两块金属板焊接在一起。

1.1.2 自动控制系统建模

自动控制系统的模型是对该控制系统的特征与变化规律的一种定量抽象表示，是人们为了认识事物所采用的一种手段。通常有以下几种模型：

1) 物理模型：根据相似原理，把真实系统按比例放大或缩小制成的模型。

例如，在研制新型飞机时，要将飞机的实物模型放在风洞中进行试验研究，以确定其空气动力学性能。

使用物理模型做仿真实验研究的优点是效果逼真、精度高；缺点是造价高昂。

2) 数学模型：用数学方程、结构图来描述系统特性的模型。随着计算机技术的发展，

人们越来越多地采用数学模型，在计算机上进行仿真试验研究。

3) 数学模型和物理模型相结合的模型(半实物模型):这种模型结合了数学模型和物理模型的优点,对某些数学模型不确切的关键部件采用物理模型,其他部分则由计算机根据其数学模型来构成。

自动控制系统建模就是以相关的理论为依据,把系统的运动规律概括为数学方程关系或函数关系。通常包括以下内容:

- 1) 确定控制系统模型的结构,建立系统的约束条件,确定系统的属性与运动。
- 2) 测取模型数据。
- 3) 运用相关领域的理论建立系统的数学描述。
- 4) 检验所建立的数学模型的准确性。

由于自动控制系统的数字仿真以该系统的数学模型为基础的,因此仿真结果的可信度在很大程度上取决于系统建模的准确程度。可见,系统建模至关重要,它在很大程度上决定了数字仿真实验的成败。目前,在国际上流行的仿真软件 MATLAB/Simulink 中,集成了各个领域的专家学者对常见的被控对象所建的模型以及编写的程序模块(如电力系统模块集中的各种电机模块、电力电子器件模块等)。在 MATLAB/Simulink 环境下,通常建模的过程会变得十分方便快捷和真实准确,因而仿真结果也更加可信。

1.2 自动控制系统仿真的基本概念

1.2.1 仿真的定义

自动控制系统的计算机仿真是指以数字计算机为主要工具,编写并运行反映真实系统运行状况的程序。对计算机输出的信息进行分析和研究,从而对实际系统运行状态和演化规律进行综合评估与预测。它是非常重要的设计自动控制系统或者评价系统性能和功能的一种技术手段。

仿真的依据是相似原理,即真实系统与它的模型在某种意义上是相似的。

例如,我们在设计一个化工厂反应釜压力控制系统时,要确定采用何种控制方案才能够达到所要求的性能指标。可以先对该系统采用不同方案进行控制的效果进行仿真,经过分析比较之后,确定最佳控制方案。

1.2.2 自动控制系统的分类

1. 按照仿真模型的属性分类

(1) 物理仿真

按照实际系统的物理性质构造系统的物理模型,并且在该物理模型上进行实验研究,这种方法称为物理仿真。

例如,为了获得新型飞机的空气动力学特性的数据,要用按比例缩小的飞机模型在风洞实验室中进行实验。

(2) 数学仿真

按照实际系统的运动规律构造系统的数学模型,并且在数字计算机上进行实验研究。

数学仿真具有经济、方便、使用灵活等特点,已经得到越来越广泛的应用。本书介绍的计算机仿真,就是指数学仿真。

(3) 数学—物理仿真

将系统部分用数学模型和部分用物理模型有机地组合起来构成仿真模型,进行实验研究,称为数学—物理仿真,也称为半实物仿真。

这种方法结合了数学仿真和物理仿真的特点,常常用在一些特定的场合,例如,培训飞行员时使用的飞行驾驶模拟舱。

2. 按系统状态的时间连续性分类

(1) 连续系统仿真

系统的各状态变量,以及输入、输出信号均为时间的连续函数,可以用微分方程、状态空间表达式、传递函数等具有连续特性的数学模型来描述。

(2) 离散事件系统仿真

系统的状态只能在离散时刻观测与控制的系统称为离散事件系统。

1.2.3 自动控制系统仿真的过程

自动控制系统的计算机仿真的过程包括以下几个步骤:

1. 建立控制系统的数学模型

根据系统的实际结构与系统各变量之间所遵循的物理、化学基本定律,例如,牛顿运动定律、基尔霍夫定律、电机学基本原理来列写变量间的数学表达式以建立系统的数学模型。

对于一些复杂的系统,需要通过实验的方法,利用系统辨识技术,忽略一些次要因素。使模型既能够准确地反映系统的动态本质,又能够简化分析计算的工作。

2. 建立自动控制系统的仿真模型

为自动控制系统所建立的数学模型,通常是用微分方程、差分方程、传递函数、状态方程等形式来描述的,还不能直接用来对系统进行仿真,应该将其转换成能够在计算机上对系统进行仿真的模型。

利用 MATLAB 及其模块化图形界面仿真部件 Simulink,以及附带的各种工具箱作为仿真工具,可以方便地构建自动控制系统仿真模型,用来研究和分析控制系统。

3. 在计算机上进行仿真实验并输出仿真结果

首先,将编制好的仿真程序(或者 Simulink 模型)设置初始参数;然后进行仿真实验;对仿真程序和仿真模型作必要的调整修改,再将最后的仿真结果以数据、曲线、图形、动画等方式输出;最后,进行仿真总结,提交系统仿真报告。

1.3 仿真技术在控制系统设计中的应用及其重要意义

1.3.1 自动控制理论简介

自动控制理论的发展大致经历了经典控制理论和现代控制理论两个阶段。其中,经典控制理论主要研究单输入单输出(SISO)系统,所涉及的系统大多是线性定常系统。控制系统设计的常用方法包括频率特性法和根轨迹法等。经典控制理论是与生产过程的局部自动化

和单机自动化相适应的,主要依赖于图解法,采用手工进行分析和综合。这个特点是与 20 世纪 50 年代前后科学技术发展水平密切相关的。

现代控制理论可以用来解决多输入多输出(MIMO)系统的问题,系统可以是线性的或非线性的,定常的或时变的。其主要的研究方法是状态空间法,它的分析不仅限于单纯的闭环,而且可以扩展为自适应环、学习环等。现代控制理论的出现是从 20 世纪 60 年代开始的人类探索宇宙空间的需要,也是计算机飞速发展和普及的结果。

1.3.2 仿真技术与 CAD 在自动控制系统设计中的重要意义

从事控制系统分析和设计的技术人员在进行系统分析和设计时常常会面临巨大的、繁琐的计算工作量。随着计算机科学与技术的飞速发展,计算机越来越普及。自动控制系统仿真与 CAD 作为一门以计算机为工具进行设计与分析的技术得到广泛应用,极大地提高了工作效率和分析计算的精确度。

正如目前世界上已经很少有机电工程师只用纸和笔来手工绘制机械零件图那样,现在,也不可想象世界上会有控制工程师在设计和分析实际控制系统的性能时只用纸和笔而不用计算机。掌握自动控制系统仿真与 CAD 技术是当今控制系统工程师必须具有的基本技能。否则,就会被时代所淘汰。

1.3.3 仿真技术在自动控制系统设计中的应用现状和发展趋势

由于计算机技术的飞速发展,带动了仿真实论和方法的快速进步。控制系统仿真技术有以下发展趋势:

- 向更加广阔的时空发展。对大型复杂系统、分布系统、综合系统进行实时仿真。由于信息量大,需要对信息进行快速高效的处理和传输。多计算机并行的数字仿真系统将会有更多发展。以现代复杂军事系统为例,它涉及战略、战术决策系统;作战指挥、通信、作战人员运输、武器装备及其运载系统;战区地理环境、战时气象环境、地面与空中的各军兵种协同作战等。这种系统对实时性与实用性等的要求都很高。
- 向模型更加准确方向发展。通常为了方便快捷地建立系统模型,往往在系统建模过程中进行一些简化,系统模型和实际系统之间存在一定偏差。因此,需要有规范化的模型校核、验证和确认过程,来评价模型的正确性和可信度。并且通过反复多次地修正系统模型,提高其正确性和可信度。
- 向虚拟现实技术,以及智能化、一体化方向发展。虚拟现实是将真实环境、模型化物理环境、用户融为一体,为用户提供视觉、听觉、嗅觉和触觉感官以逼真感觉信息的仿真系统。虚拟现实技术可以使人产生如同身临其境的感受。需要将计算机仿真技术、传感器、各种类型的驱动器融为一体。

1.4 MATLAB 语言及其在控制系统设计中的应用

MATLAB 是由美国的 Math Works 公司推出的一个科技应用软件。它是由 Matrix Laboratory (矩阵实验室)两个单词,分别取前 3 个字母组合而成的。MATLAB 最显著的特点就是:功

能强大、易学易用。它通常被称为演算纸式的科学工程计算语言。

初版的 MATLAB 软件是由 Cleve Moler 教授开发的。他曾经在美国密西根大学、斯坦福大学和新墨西哥大学任数学与计算机科学教授。1980 年前后, 时任新墨西哥大学计算机系主任的 Moler 教授在讲授线性代数课程时, 发现使用其他高级语言 (如 Basic、FORTRAN、C 等) 来编写矩阵运算程序, 极为不方便。于是他构思并且开发了便于使用的、交互式的 MATLAB 软件。

1984 年, Math Works 公司推出了第一款 MATLAB 的商业版本, 其核心是用 C 语言编写的。1990 年推出了 MATLAB 3.5 版, 它是可以运行于 Microsoft Windows 下的版本, 它又增加了丰富多彩的图形图像处理、多媒体、符号运算等功能, 还增加了与其他流行软件的接口功能, 使得 MATLAB 的功能越来越强大。

Math Works 公司于 1992 年推出了具有划时代意义的 MATLAB 4.0 版, 于 1994 年又推出了 4.2 版, 扩充了 4.0 版的功能, 尤其在图形界面设计方面提供了新的方法。1997 年推出的 MATLAB 5.0 版支持更多的数据结构, 使其成为一种更加方便的编程语言。1999 年推出的 MATLAB 5.3 版在很多方面又进一步改进了 MATLAB 的功能。2000 年 10 月, 该公司推出了 MATLAB 6.0 版本, 在操作界面上有了很大的改观, 为用户提供了很大方便。2001 年 6 月, MATLAB 6.1 版以及 Simulink 4.0 版问世, 功能进一步加强。2002 年 6 月, Math Works 公司推出了 MATLAB 6.5/Simulink 5.0 (即 MATLAB Release 13), 其功能在原有基础上有了进一步改善。2004 年, Math Works 公司推出了 MATLAB 7.0/Simulink 6.0 (即 MATLAB Release 14), 其中, 主要包括 12 个新产品模块, 同时升级了 28 个产品模块。MATLAB/Simulink 产品仍然在不断的升级之中, 2006 年推出了 MATLAB 2006a 和 MATLAB 2006b, 扩充了一些模块。新的版本对计算机配置的要求也越来越高。如果计算机配置较低, 新版本 MATLAB 的运行速度会大受影响。本书主要介绍 MATLAB 7.0/Simulink 6.0。

目前, MATLAB 已经成为国际上优秀的科学与工程计算软件之一。以其模块化的计算方法、可视化与智能化的人机交互功能、丰富的矩阵运算、图形绘制和数据处理函数, 以及它所附带的模块化图形组态的动态系统仿真工具 Simulink, MATLAB 已经成为控制系统设计和仿真领域最受欢迎的软件。

小 结

本章简要地介绍了控制系统仿真的基本概念和意义, 阐述了控制系统仿真的几种主要的方法。并且着重介绍了当前在世界上流行的控制系统仿真软件 MATLAB 的发展历史、特点及功能。

习 题

- 1-1 什么是仿真? 它的主要优点是什么? 它所遵循的基本原则是什么?
- 1-2 你认为计算机仿真的发展方向是什么?
- 1-3 计算机数字仿真包括哪些要素? 它们的关系如何?

第2章 控制系统计算机数字仿真基础

2.1 连续系统数值积分方法

连续系统的主要特征是系统的状态变化在时间上是连续的,通常用微分方程或差分方程来描述系统的模型,如过程控制系统、调速系统、随动系统等。在数字计算机上对连续系统进行仿真时,首先遇到的问题是,数字计算机的数值及时间均具有离散性,而被仿真系统的数值及时间均具有连续性,后者如何用前者实现。从根本意义上讲,连续系统数字仿真要从时间和数值两方面对原系统进行离散化,并选择合适的数值计算方法来近似积分运算。连续系统数字仿真的离散化方法有两类,即数值积分法和离散相似法。数值积分法就是利用数值积分的方法对常微分方程(组)建立离散化形式的数学模型——差分方程,并求其数值解,也称为数值解法。基于离散相似法的连续系统仿真和数值积分法不同,它首先将连续系统模型离散化,再借用离散系统仿真算法。本节主要讨论数值积分法。

设一阶常微分方程

$$\begin{cases} \frac{dy}{dt} = f(t, y) \\ y(t_0) = y_0 \end{cases} \quad (2-1)$$

式(2-1)的解 $y(t)$ 在区间 $[a, b]$ 上是连续变化的。将区间分成若干个子区间,时间间隔为 h ,在 $[t_k, t_{k+1}]$ 区间积分,得

$$y_{k+1} = y_k + \int_{t_k}^{t_{k+1}} f(t, y) dt \quad k = 0, 1, 2, \dots, N \quad (2-2)$$

这样在区间 $[a, b]$ 上每一个离散点 t_k ,均可求出对应的 y_k ,并将这些 $y_k (k=1, 2, \dots, N)$ 作为解 $y(t)$ 的近似值。数值积分法的主要问题就是如何求式(2-2)中定积分的近似解。为此,首先要将连续变量问题用数值积分方法转化成离散的差分方程的初值问题,然后根据已知的初始条件,逐步地递推计算后续时刻的数值解。采用不同的递推算法,就出现各种不同的数值积分方法。

2.1.1 欧拉法

欧拉(Euler)法是最简单的一种数值积分方法。我们用它来说明数值积分法的基本思想。

已知一阶常微分方程如式(2-1)所示,由微分的定义知

$$\frac{dy}{dt} = \lim_{h \rightarrow 0} \frac{y(t+h) - y(t)}{h} \quad (2-3)$$

在 $t = t_k$ 时刻,当 $h = \Delta t = t_{k+1} - t_k$ 足够小时,可以用差分的形式近似代替微分,即

$$\frac{dy}{dt} = f(t, y) \approx \frac{\Delta y}{\Delta t} = \frac{y_{k+1} - y_k}{h} \quad (2-4)$$

式中, $y_{k+1} = y(t + \Delta t)$; $y_k = y(t)$ 。将式 (2-4) 改写, 得

$$y_{k+1} = y_k + hf(t_k, y_k) \quad k=0, 1, 2, \dots, N \quad (2-5)$$

比较式 (2-5) 和式 (2-2), $hf(t_k, y_k)$ 部分近似代替了积分部分, 其几何意义是把 $f(t, y)$ 在 $[t_k, t_{k+1}]$ 区间内的曲边面积用矩形面积近似代替, 如图 2-1 所示。当 h 很小时, 可以认为产生的误差在允许范围内。这样式 (2-5) 可以从 t_0 开始, 逐点递推求得 t_1 时的 y_1 , t_2 时的 y_2 ……直到 t_N 时的 y_N , 称为欧拉递推公式, 利用递推公式进行数值求解的过程如图 2-2 所示。从图中可以看出, 当 h 很小时, 是利用 $y(t)$ 在 t_k 处的切线方程获得 t_{k+1} 处 $y(t)$ 的近似值 y_{k+1} , 因此, 欧拉法也称为折线法。

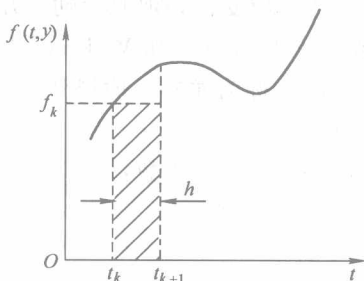


图 2-1 欧拉法的几何意义

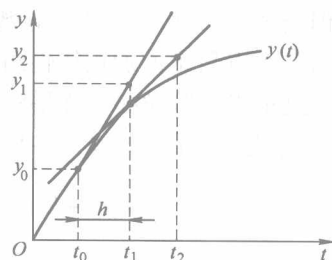


图 2-2 欧拉法数值积分过程

欧拉法方法简单, 计算量小, 只要给定初始条件 y_0 和步长 h , 就可以进行递推运算, 由前一点值 y_k 仅一步递推就可以求出后一点值 y_{k+1} , 为单步显式法, 可以自启动。但是欧拉法的精度较差, 欧拉法与泰勒展开式

$$y(t + \Delta t) = y(t) + \dot{y}(t)\Delta t + \frac{1}{2!}\ddot{y}(t)(\Delta t)^2 + \dots \quad (2-6)$$

当 $t = t_k$, 且取 $h = \Delta t$ 时的对应式

$$y_{k+1} = y_k + h\dot{y}_k + \frac{1}{2!}h^2\ddot{y}_k + \dots \quad (2-7)$$

中的一阶近似展开式相同, 即

$$y_{k+1} = y_k + h\dot{y}_k + o(h^2) \approx y_k + h\dot{y}_k \quad (2-8)$$

其误差 $o(h^2)$ 与 h^2 同数量级, 称其具有一阶精度。尽管欧拉法的精度很差, 但欧拉法仍然很重要, 许多高精度的数值积分方法都是以它为基础推导得到的。

2.1.2 龙格-库塔法

由图 2-1 可清楚地看出, 欧拉法是用前一点的斜率值 f_k 确定下一点的值 $y(t_{k+1})$, 精度较低。为了提高精度, 可以使用两点斜率的平均值确定 $y(t_{k+1})$, 即

$$y_{k+1} = y_k + \frac{h}{2}(f(t_k, y_k) + f(t_{k+1}, y_{k+1})) \quad (2-9)$$

由式 (2-9) 可知, 求解 y_{k+1} 的算式中隐含 y_{k+1} 本身, 称为隐式算法, 不能自启动。在实际计算中, 首先用欧拉法预估 y_{k+1}^* 的值, 然后再进行校正。这就是预估-校正算法, 其几何意义是把 $f(t, y)$ 在 $[t_k, t_{k+1}]$ 区间内的曲边面积用梯形面积近似代替, 即

$$\begin{cases} \text{预估: } y_{k+1}^* = y_k + hf(t_k, y_k) \\ \text{校正: } y_{k+1} = y_k + \frac{h}{2}(f(t_k, y_k) + f(t_{k+1}, y_{k+1}^*)) \end{cases} \quad (2-10)$$

又

$$y_{k+1} = y_k + \frac{h}{2}(\dot{y}_k + (\dot{y}_{k+1}^*)) = y_k + h\dot{y}_k + \frac{h^2}{2}\ddot{y}_k + o(h^3) \quad (2-11)$$

显然具有二阶精度，平均斜率达到了提高精度的目的。

欧拉法取泰勒展开式的前两项进行近似计算求微分方程的数值解，预估-校正算法取泰勒展开式的前三项进行近似计算。可见，要想得到较高的计算精度，必须取泰勒展开式的前若干项，但公式中直接利用高阶导数，计算不方便。数学家 C. Runge 和 W. Kutta 提出，用计算区间内几个点斜率值加权线性组合的数值积分计算方法，称为龙格-库塔法（Runge-Kutta 法）。这里以二阶龙格-库塔法为例介绍其基本原理。

设 $y(t)$ 为式 (2-1) 的解，将其在 t_k 附近以 h 为变量展开为泰勒级数：

$$y(t_k + h) = y(t_k) + h\dot{y}(t_k) + \frac{h^2}{2!}\ddot{y}(t_k) + \cdots \quad (2-12)$$

其中

$$\begin{aligned} \dot{y}(t_k) &= f(t_k, y_k) = f_k \\ \ddot{y}(t_k) &= \left. \frac{df(t, y)}{dt} \right|_{\substack{t=t_k \\ y=y_k}} = \left(\frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} f \right) \bigg|_{\substack{t=t_k \\ y=y_k}} = f'_{t_k} + f'_{y_k} f_k \end{aligned}$$

并记

$$y(t_k + h) = y_{k+1}, y(t_k) = y_k$$

则

$$y_{k+1} = y_k + hf_k + \frac{h^2}{2!}(f'_{t_k} + f'_{y_k} f_k) + \cdots \quad (2-13)$$

将式 (2-13) 用斜率 k_i 的线性组合表示，则

$$y_{k+1} = y_k + h \sum_{i=1}^r w_i k_i \quad (2-14)$$

式中， r 为精度阶次； w_i 为待定系数； k_i 用式 (2-15) 表示：

$$k_i = f(t_k + a_i h, y_k + h \sum_{j=1}^{i-1} b_j k_j) \quad i = 1, 2, 3, \cdots, r \quad (2-15)$$

式中， a_i 、 b_j 为待定系数，一般取 $a_1 = 0$ 。

当 $r=1$ 时，式 (2-14) 变成欧拉数值积分公式：

$$y_{k+1} = y_k + hf(t_k, y_k)$$

当 $r=2$ 时，

$$\begin{cases} k_1 = f(t_k, y_k) \\ k_2 = f(t_k + a_2 h, y_k + h b_1 k_1) \\ y_{k+1} = y_k + w_1 h k_1 + w_2 h k_2 \end{cases} \quad (2-16)$$

将 k_2 按照二元函数展开并带入 y_{k+1} ，得

$$\begin{aligned}
 y_{k+1} &= y_k + w_1 h f_k + w_2 h (f_k + a_2 h f'_{t_k} + h b_1 k_1 f'_{y_k}) \\
 &= y_k + (w_1 + w_2) h f_k + w_2 a_2 h^2 f'_{t_k} + w_2 b_1 h^2 f_k f'_{y_k}
 \end{aligned}$$

与二阶近似公式比较, 即得到以下关系式:

$$w_1 + w_2 = 1, \quad w_2 a_2 = \frac{1}{2}, \quad w_2 b_1 = \frac{1}{2}$$

式中, 待定系数的个数超过方程数, 因此解不唯一, 有以下几种取法:

(1) $w_1 = w_2 = \frac{1}{2}$, $a_2 = 1$, $b_1 = 1$ 时, 则

$$\begin{cases}
 y_{k+1} = y_k + \frac{h}{2}(k_1 + k_2) \\
 k_1 = f(t_k, y_k) \\
 k_2 = f(t_k + h, y_k + h k_1)
 \end{cases}$$

(2) $w_1 = 0$, $w_2 = 1$, $a_2 = \frac{1}{2}$, $b_1 = \frac{1}{2}$ 时, 则

$$\begin{cases}
 y_{k+1} = y_k + h k_2 \\
 k_1 = f(t_k, y_k) \\
 k_2 = f(t_k + \frac{h}{2}, y_k + \frac{h}{2} k_1)
 \end{cases}$$

(3) $w_1 = \frac{1}{4}$, $w_2 = \frac{3}{4}$, $a_2 = \frac{2}{3}$, $b_1 = \frac{2}{3}$ 时, 则

$$\begin{cases}
 y_{k+1} = y_k + \frac{h}{4}(k_1 + 3k_2) \\
 k_1 = f(t_k, y_k) \\
 k_2 = f(t_k + \frac{2}{3}h, y_k + \frac{2}{3}h k_1)
 \end{cases}$$

以上几种形式均称为二阶龙格-库塔法公式。

当 $r=4$ 时, 四阶龙格-库塔法公式:

$$\begin{cases}
 y_{k+1} = y_k + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\
 k_1 = f(t_k, y_k) \\
 k_2 = f(t_k + \frac{h}{2}, y_k + \frac{h}{2}k_1) \\
 k_3 = f(t_k + \frac{h}{2}, y_k + \frac{h}{2}k_2) \\
 k_4 = f(t_k + h, y_k + h k_3)
 \end{cases} \quad (2-17)$$

式 (2-17) 有较高的精度, 所以在数字仿真中应用比较普遍。

所有的龙格-库塔法的公式都有以下特点:

1) 在计算 y_{k+1} 时只用到 y_k , 而不直接用 y_{k-1} 、 y_{k-2} 等项, 即前一点值一步递推就可以求出后一点值, 为单步法, 显然它不仅能使存储量减小, 而且此法可以自启动, 即已知初值后, 就能由初值逐步计算得到后续各时间点上的数值。实际上在逐步递推的过程中, 计算

y_{k+1} 之前就已经产生了 y_{k-1} 、 y_{k-2} 等项,利用这些结果计算 y_{k+1} 的方法就是多步法。多步法利用的信息量大,因而比单步法更精确,但无法自启动。本章不单独讨论多步法,有兴趣的读者可参阅相关参考文献。

2) 步长 h 在整个计算中并不要求固定,可以根据精度要求改变,但是在一步中计算若干系数 k_i (俗称龙格-库塔系数),则必须用同一个步长 h 。

3) 龙格-库塔法的精度取决于步长 h 的大小及方法的阶次。许多计算实例表明:为达到相同的精度,四阶方法的 h 可以比二阶方法的 h 大10倍,而四阶方法的每步计算量仅比二阶方法大1倍,所以总的计算量仍比二阶方法小。正是由于上述原因,一般系统进行数字仿真实常用四阶龙格-库塔公式。值得指出的是:高于四阶的方法由于每步计算量将增加较多,而精度提高不多,因此使用得也比较少。

2.1.3 数值积分法的稳定性

利用数值积分法进行仿真时常常会出现这样的情况,一个系统本来是稳定的,可是仿真结果却是发散的。这种情况通常是由积分步长选得不合适造成的。那么,为什么计算步长选得不合适会引起数值解不稳定呢?这就需要分析各种数值解法的稳定性。

首先来看一个例子,用欧拉法求一阶系统

$$\dot{y} + 10y = 0 \quad y(0) = 1$$

的数值解。

设计算步长为 h ,则欧拉递推公式为

$$y_{k+1} = y_k + h(-10y_k) = (1 - 10h)y_k = (1 - 10h)^{k+1}y_0 = (1 - 10h)^{k+1}$$

当 $h > 0.2$ 时, $|1 - 10h| > 1$, 数值解是发散的;

当 $h = 0.2$ 时, $|1 - 10h| = 1$, 数值解等幅振荡;

当 $0 < h < 0.2$ 时, $|1 - 10h| < 1$, 数值解是收敛的;

原来系统是稳定的,时间常数为0.1,解析解 $y(t) = e^{-10t}$,用欧拉法进行仿真时,当步长 $h \geq 0.2$ 时,由于步长太大引起截断误差过大,造成数值解不稳定。

微分方程(组)的数值解法,实质上就是将微分方程差分化,然后从初值开始进行迭代运算。显然,要使迭代运算正常进行,首先必须保证这一数值解法的稳定性。所谓数值解法的稳定性,是指在扰动(如初始误差、舍入误差、截断误差等)影响下,其计算过程中的累积误差不会随计算步数的增加而无限增长。不同的数值解法对应着不同的差分递推公式。一个数值法是否稳定取决于该差分方程的特征根是否满足稳定性要求。为了说明这个问题,我们讨论一个简单的微分方程:

$$\frac{dy}{dt} = f(t, y) = \lambda y \quad (2-18)$$

式中, $\lambda = \alpha + j\beta$ ($\alpha < 0$) 为方程的特征根。

仍以欧拉法为例,将式(2-18)代入欧拉递推公式,得

$$y_{k+1} = y_k + h\lambda y_k = (1 + h\lambda)y_k \quad (2-19)$$

两边进行 z 变换,得

$$zY(z) = (1 + h\lambda)Y(z)$$