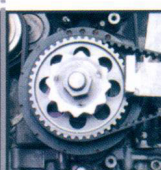
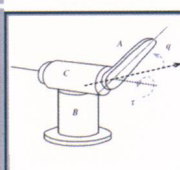
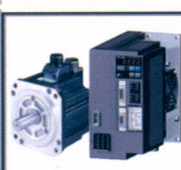


不确定机械问题的 建模与控制补偿

MODELING AND CONTROL COMPENSATION OF
UNCERTAIN MECHANICAL PROBLEMS

王永富 著



不确定机械问题的建模与控制补偿

王永富 著



机械工业出版社

非线性摩擦是一种复杂的、非线性的、具有不确定性的自然现象。为了提高系统性能,必须减小或消除摩擦对系统性能的不利影响,解决摩擦问题对提高机电产品性能具有重要意义。从这一实际出发,本书以摩擦学和非线性控制理论为基础,以非线性摩擦的自适应模糊建模与控制补偿为研究内容,以减小或消除摩擦对机械系统性能的影响为目标,以理论研究和实验相结合为手段,以目前在工程中广泛应用的机械手和医疗设备为应用对象,研究摩擦的智能建模与控制补偿的特性,逐步建立摩擦的智能建模与控制补偿的理论体系,并将研究得出的理论应用于工程实践,力求最大可能降低摩擦所带来的危害及提高设备的控制精度,进而为提高相关企业产品的技术含量做出努力。

本书可作为机电行业的从业人员参考使用,也可用于大专院校相关专业师生的辅助教材。

图书在版编目(CIP)数据

不确定机械问题的建模与控制补偿/王永富著. —北京:机械工业出版社, 2013. 7

ISBN 978-7-111-43243-2

I. ①不… II. ①王… III. ①非线性—摩擦—研究 IV. ①O313.5

中国版本图书馆CIP数据核字(2013)第156760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:赵海青 责任编辑:赵海青 刘 焯

版式设计:常天培 责任校对:张 征

封面设计:张 静 责任印制:杨 曦

北京圣夫亚美印刷有限公司印刷

2013年9月第1版第1次印刷

184mm×260mm·11.25印张·260千字

0 001—2000 册

标准书号:ISBN 978-7-111-43243-2

定价:40.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心:(010)88361066

教材网:<http://www.cmpedu.com>

销售一部:(010)68326294

机工官网:<http://www.cmpbook.com>

销售二部:(010)88379649

机工官博:<http://weibo.com/cmp1952>

读者购书热线:(010)88379203

封面无防伪标均为盗版

前 言

复杂机械系统常常具有强的非线性、不确定性、多变量、强耦合，以及运动变化频繁等特点，动态特性难于用精确的数学模型描述。自适应模糊建模与控制的基本出发点是模仿人的智能以实现对复杂不确定性系统进行有效控制，它具有自学习和适应环境的能力，自动进行信息处理以减少其不确定性，能规划、产生，并能安全、可靠地执行控制作用。因此，自适应模糊建模与控制是解决复杂机械系统不确定性问题的有效途径之一。

非线性摩擦是一种复杂的、非线性的、具有不确定性的自然现象，普遍存在于自然界和机械工程领域。人类目前对这一物理过程的了解尚处在进一步探索阶段，还无法通过数学方法给出完全真实的精确描述。现实生活中摩擦几乎无处不在，在有些情况下，摩擦是人们所期望的，如汽车的制动系统利用了摩擦进行制动。但对于大多数机械系统而言，摩擦却成为提高系统性能的障碍，使系统响应出现爬行、振荡或稳态误差以及其他不可预料的现象。因此，为提高机械系统的性能就必须减小或消除摩擦对系统性能的不利影响，解决非线性摩擦问题对提高机电产品性能具有重要的理论和实际意义。另外，近年来随着数字计算机技术的发展和应用，基于嵌入式操作系统的实时控制在国外得到了快速发展并广泛地应用在通信、军事、航空航天等高精尖技术及实时性要求极高的领域中，如卫星通信、军事演习、导弹制导、飞机导航和火星探测器等。为了推动我国在这些领域的发展与应用，结合工程实际，本书也介绍了嵌入式的实时控制、控制补偿及其在医疗设备开发方面的应用。

本书的写作和出版过程得到了闻邦椿院士、柴天佑院士、东软集团刘积仁 and 赵宏老师、澳大利亚拉筹伯大学王殿辉老师、东北大学张义民教授和其他老师的热情鼓励和支持，对诸位学者的切实帮助，谨表衷心的感谢。同时，本书得到国家自然科学基金（51275085）、流程工业综合自动化国家重点实验室开放课题、中央高校基本科研业务费人才基金（N110503001）和沈阳市科技基金（F12-175-9-00）的资助，在此一并表示深切的感谢。

本书敬献给东北大学 90 周年校庆

作者

2013 年 3 月于东北大学

目 录

前言

第 1 章 非线性摩擦的基础知识	1
1.1 机械动力学中的非线性问题	1
1.2 非线性摩擦问题	1
1.2.1 摩擦的特性	1
1.2.2 摩擦的危害	2
1.2.3 如何消除摩擦影响	3
1.3 非线性摩擦的建模	3
1.3.1 摩擦的静态模型	3
1.3.2 摩擦的动态模型	7
1.3.3 基于智能系统的摩擦模型	13
1.3.4 基于数据驱动的摩擦模型	13
1.4 非线性摩擦的补偿方法	15
1.4.1 基于传统摩擦模型的补偿方法	15
1.4.2 非模型的补偿方法	17
1.4.3 智能系统的摩擦补偿方法	19
1.5 本章小结	20
参考文献	21
第 2 章 非线性系统的模糊建模方法	27
2.1 常用的建模方法	27
2.2 模糊建模的基本理论基础	32
2.2.1 模糊集合及其表示法	33
2.2.2 隶属函数	35
2.2.3 模糊规则和模糊推理	36
2.2.4 常见的几种模糊系统	37
2.2.5 模糊系统的通用逼近性	39
2.3 WM 算法及需改进的问题	39
2.3.1 WM 算法的主要步骤	40
2.3.2 MW 算法需要改进的地方	43
2.4 iWM 算法及需改进的问题	44
2.4.1 iWM 算法的主要思想	44
2.4.2 iWM 算法需改进的问题	45

2.5 一种新的 DM 算法建立模糊模型	45
2.5.1 提取完备的模糊规则库	45
2.5.2 建立鲁棒模糊模型	48
2.5.3 WM 和 DM 算法仿真对比	48
2.5.4 iWM 和 DM 算法仿真对比	52
2.6 本章小结	54
参考文献	54
第 3 章 摩擦模糊模型的规则提取与自适应控制补偿	57
3.1 现有摩擦模型的不足	57
3.2 数据挖掘方法建立摩擦的静态模糊模型	58
3.2.1 摩擦实验数据的采集	58
3.2.2 摩擦模糊规则库的提取	59
3.2.3 建立摩擦的静态模糊模型	61
3.3 摩擦模糊模型的自适应机制	62
3.4 仿真实验	64
3.4.1 系统参数选择	65
3.4.2 摩擦建模的模糊规则提取	65
3.4.3 摩擦模糊模型的自适应控制补偿	66
3.4.4 仿真结果与分析	67
3.5 本章小结	69
参考文献	69
第 4 章 基于两种状态估计的摩擦控制补偿对比	71
4.1 问题描述	71
4.2 机器人模型	71
4.3 PD 输出反馈控制	72
4.3.1 基于高增益观测器的 PD 输出反馈控制	73
4.3.2 基于数字差分的 PD 输出反馈控制	73
4.3.3 两种方法的仿真对比	73
4.4 本章小结	76
参考文献	76
第 5 章 基于模糊观测器的多摩擦环节建模与控制补偿	77
5.1 问题描述	77
5.2 模糊观测器现状	77
5.3 数学模型	79
5.4 多摩擦环节的模糊建模与状态估计	81
5.4.1 多摩擦环节的模糊建模	81
5.4.2 状态估计与误差系统	82

5.4.3	自适应模糊估计器	83
5.4.4	有界性分析	84
5.4.5	状态估计的仿真	86
5.5	鲁棒自适应控制器设计	90
5.5.1	控制器设计	90
5.5.2	控制器仿真	91
5.6	本章小结	93
	参考文献	93
第6章	非线性摩擦的双调节模糊建模与鲁棒自适应控制	95
6.1	引言	95
6.2	不确定机械动力系统	95
6.2.1	问题描述	95
6.2.2	摩擦模型与补偿	96
6.2.3	摩擦模型的辨识与自适应特性	97
6.3	摩擦的模糊建模与鲁棒控制	98
6.3.1	摩擦的自适应模糊建模方法	98
6.3.2	鲁棒控制器设计	99
6.4	鲁棒性能指标分析	100
6.5	数值结果	102
6.5.1	传统摩擦模型的自适应特性	103
6.5.2	单参数调节摩擦模糊模型的自适应特性	103
6.5.3	双参数调节摩擦模糊模型的自适应特性	108
6.6	本章小结	111
	参考文献	111
第7章	非线性摩擦诱发的自激振动与主动控制	113
7.1	自激振动	113
7.2	摩擦诱发的自激振动	113
7.2.1	一个典型的数学模型	113
7.2.2	摩擦自激振动解析解的争议	114
7.2.3	摩擦自激振动的数值分析	116
7.3	摩擦自激振动的主动控制	118
7.3.1	模糊神经网络建立摩擦模型	118
7.3.2	摩擦自激振动的主动控制理论	121
7.4	数值结果	123
7.4.1	系统参数设置	123
7.4.2	模糊神经网络补偿器的初始化设置	123
7.4.3	模糊神经网络补偿器的自适应参数	125

7.4.4 Coulomb 摩擦诱发振动的主动控制	125
7.4.5 Stribeck 摩擦诱发振动的主动控制	126
7.5 本章小结	127
参考文献	127
第 8 章 机器人实验系统与模糊控制补偿设计实例	129
8.1 引言	129
8.2 机器人的控制系统结构	129
8.2.1 机械子系统的构造	129
8.2.2 上位机子系统	130
8.2.3 下位机子系统	132
8.2.4 电动机驱动子系统	138
8.2.5 编码器和电动机子系统	139
8.3 机器人的控制方法	140
8.3.1 PD 基础控制	140
8.3.2 模糊补偿控制	141
8.3.3 模糊 PD 控制的实验	142
8.4 本章小结	143
参考文献	143
第 9 章 多任务实时系统设计与控制补偿及其应用	144
9.1 嵌入式实时系统	144
9.1.1 嵌入式系统	144
9.1.2 嵌入式实时系统概念	145
9.1.3 嵌入式实时系统的特点	145
9.1.4 嵌入式实时系统的分类与调度	146
9.1.5 常见的嵌入式实时操作系统	147
9.2 实时多任务 Linux 操作系统	148
9.2.1 Linux 简介	148
9.2.2 嵌入式 Linux 简介	149
9.2.3 嵌入式实时系统 RTLinux 简介	150
9.2.4 嵌入式实时系统 RTAIXLinux 简介	151
9.3 系统的硬件设计	153
9.3.1 PET/CT 简介	153
9.3.2 PET 机电控制系统	154
9.4 系统的软件设计	156
9.4.1 系统整体结构	156
9.4.2 系统软件设计方法	159
9.5 信息传递过程实例	159

- 9.5.1 网络传递信息 160
- 9.5.2 信号量传递信息 163
- 9.5.3 邮箱传递信息 165
- 9.6 不确定问题的校正与补偿处理 167
 - 9.6.1 间隙补偿与打滑校正 167
 - 9.6.2 不确定振动的补偿处理 168
- 9.7 本章小结 170
- 参考文献..... 170

第 1 章 非线性摩擦的基础知识

1.1 机械动力学中的非线性问题

真实的机械动力系统中常常包含各种各样的不确定因素，诸如机械系统中的间隙、非线性摩擦，结构系统中的材料弹塑性影响、构件变形，控制系统中的元器件饱和等特性，使得机械动力系统呈现复杂的不确定性和非线性。实践中，人们经常试图用线性确定模型来替代实际的非线性不确定模型，以求方便地获得其行为的某种逼近。然而，被忽略的非线性不确定因素常常会在分析和计算中引起无法接受的误差，使得线性逼近无法实现。特别对于系统的长时间历程动力学问题，有时即使略去很微弱的非线性因素，也会在分析和计算时出现本质性的错误^[1]。

近年来，伴随着计算机技术、数值模拟和智能控制理论等的进步，非线性动力学在理论和方法上得到了快速发展。在工程科学界，以往研究人员对于非线性问题绕道而行的现象正在发生变化。人们不仅力求深入分析非线性对系统动力学的影响，使系统和产品的动态设计、加工、运行与控制满足日益提高的运行速度和精度要求，而且开始探索利用这些非线性现象造福人类。鉴于动力系统中的不确定问题种类繁多，本书把工作重点放在了非线性摩擦的建模与补偿控制的系统理论和方法的研究上，探讨了非线性摩擦的智能建模和自适应补偿控制等方法以及在工程实际中的应用。

1.2 非线性摩擦问题

1.2.1 摩擦的特性

在机械系统中，具有相对运动或相对运动趋势的两个接触面上会产生摩擦。经过长期的研究，人们发现摩擦是一种比较复杂的现象，有很多类型，其特性存在很大的差异。就机械系统中相互接触的滑动表面而言，在润滑状态下，其摩擦力是接触面间相对运动速度的函数。从静止开始加速，摩擦力的变化经历了以下四个阶段：静摩擦阶段、边界润滑阶段、部分流体润滑阶段、全流体润滑阶段，图 1-1 是这四个阶段的示意图^[2]。

第一个阶段是静摩擦阶段，静摩擦力不依赖于速度，它实际上可以认为是由弹性变形所产生的，从控制的角度看，正是这种弹性变形导致了增加的静摩擦力。这种弹性变形称之为滑前位移，尽管滑前位移较小，对于一般工程材料而言仅为 $2 \sim 5\text{nm}$ ，但对于某些精度要求极高的伺服定位系统而言，它仍然非常重要。从本质上说，静摩擦力并不是真正意义上的摩擦力，而是一种约束力，因为它既不耗能也不是滑动的结果。

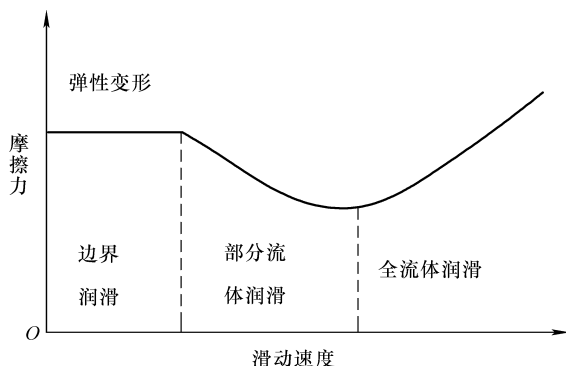


图 1-1 滑动速度和摩擦力的关系

第二阶段是边界润滑阶段，接触表面间的相对运动速度极低以至无法在其表面间建立液体薄膜，摩擦力实际上是由固体间的剪切作用引起的。虽然人们普遍认为这一阶段摩擦力大于后面两个阶段的摩擦力，但这也不是绝对的，因为对于某些物质而言，剪切力未必大于黏滞摩擦力。另外，研究表明边界润滑对 Stribeck 曲线的形状影响非常明显。此外，边界润滑与系统的低速爬行现象也存在着密切的关系。

第三阶段是部分流体润滑阶段，相对运动使接触表面间形成液体薄膜，然而由于法向压力的作用，又使部分润滑液被挤出接触表面，因此仍有部分区域为固体接触。这是最难建模的一个阶段，研究成果表明，在这一阶段“摩擦记忆”现象较为明显。

第四阶段是全流体润滑阶段，液体薄膜完全形成，不再有固体接触的区域，因此摩擦力减小，但随着相对运动速度的提高，黏滞摩擦的作用却越来越明显。

上述曲线描述了摩擦力与速度之间的稳态对应关系，它实际上是摩擦力的静态特性。摩擦力的动态特性非常复杂且具有不确定性，至今尚在研究之中。随着摩擦学的不断发展，人们除了发现上述曲线外，摩擦力还具有时间依赖性，即增加的静摩擦力和“摩擦记忆”的特性。所谓“摩擦记忆”就是接触表面间相对运动速度发生改变时，摩擦力滞后一段时间才会改变的现象。而增加的静摩擦力说明在一定的环境及润滑条件下，最大静摩擦力并非固定值，而是随接触表面间停滞时间的增加而增加，图 1-2 说明了这两种现象。摩擦力还与接触面间的压力有关，当负载增大时，压力增大，静、动摩擦力也随之增大。摩擦特性具有不确定性，它还受接触面的清洁程度、环境温度、湿度等因素的影响。

1.2.2 摩擦的危害

长期以来，机械系统中的摩擦一直困扰着众多的机械学、控制学学者。摩擦的存在降低了系统的性能，特别是对某些系统，如机械手、数控机床、转台以及坐标测量机而言，系统中的摩擦对系统性能的提高构成了严重的阻碍。经过多年理论与实践的研究发现，摩擦对系统性能主要产生如下不利影响：使伺服定位产生稳态误差，使双向运行的伺服系统产生不连续运动，从而引起振动，使高速运行的伺服系统产生较大的跟随误差、降低了跟踪精度，使单向、低速运行产生爬行现象，近年研究发现，摩擦也可以导致系统进入混沌状态。因此为

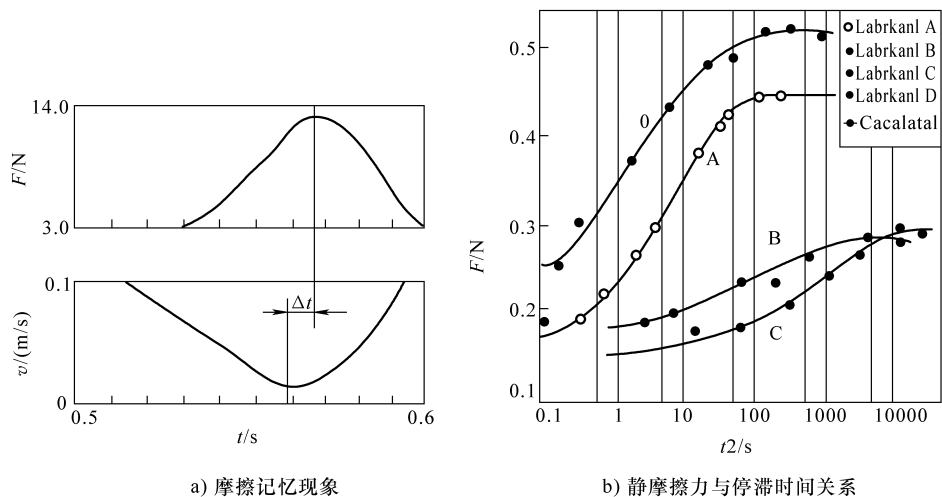


图 1-2 “摩擦记忆”现象与时间依赖性

了提高系统的性能就必须减小或消除摩擦对系统性能的不利影响。

1.2.3 如何消除摩擦影响

为了克服摩擦给系统带来的危害，最直接的办法就是尽量减小系统中存在的摩擦力。例如提高有关机械零件的加工精度、改进润滑条件以及采用高性能轴承，如：气浮轴承、磁悬浮轴承等。然而这些纯机械的方法往往造价比较昂贵，甚至是不可能做到的。因此，有必要寻找其他更经济的方法来减小或消除摩擦对系统性能的不利影响，这些方法一般称为摩擦补偿方法。近年来，摩擦补偿已成为研究的热点，除 IEEE、ASME、AUTOMATICA 等杂志上发表了很多有关摩擦补偿的论文外，美国控制会议（American Control Conference）还常设了摩擦补偿的专题，每年都发表若干最新研究成果。

1.3 非线性摩擦的建模

对非线性摩擦建立准确的数学模型，无论是从认识摩擦现象，还是从对其进行控制补偿来说都是十分重要的。因此，有关摩擦建模的研究一直非常活跃，到目前为止，已提出的摩擦模型有几十种，主要分为静态摩擦模型、动态摩擦模型、基于数据驱动的摩擦模型和基于智能系统的摩擦模型。已有很多文献对各类摩擦模型和补偿方法分别进行了综述，例如文献 [3-6]。为了便于读者很好地掌握这些摩擦模型，基于上面的文献，下面将控制领域中常用的摩擦模型和补偿控制方法加以分类概述和归纳。

1.3.1 摩擦的静态模型

早期人们根据观测和试验，对摩擦的机理进行了数学描述，建立了摩擦的静态模型。摩擦静态模型是将摩擦力描述为相对速度的函数。

1. 库仑模型

16 世纪早期, 达·芬奇在试验观察的基础上, 得出了“摩擦力正比于法向载荷, 与运行方向相反且不依赖于接触面积”的结论, 后来库仑 (Coulomb) 在达·芬奇研究基础上, 将其发展成了库仑模型, 摩擦力可表示为下式的形式, 如图 1-3a 所示。

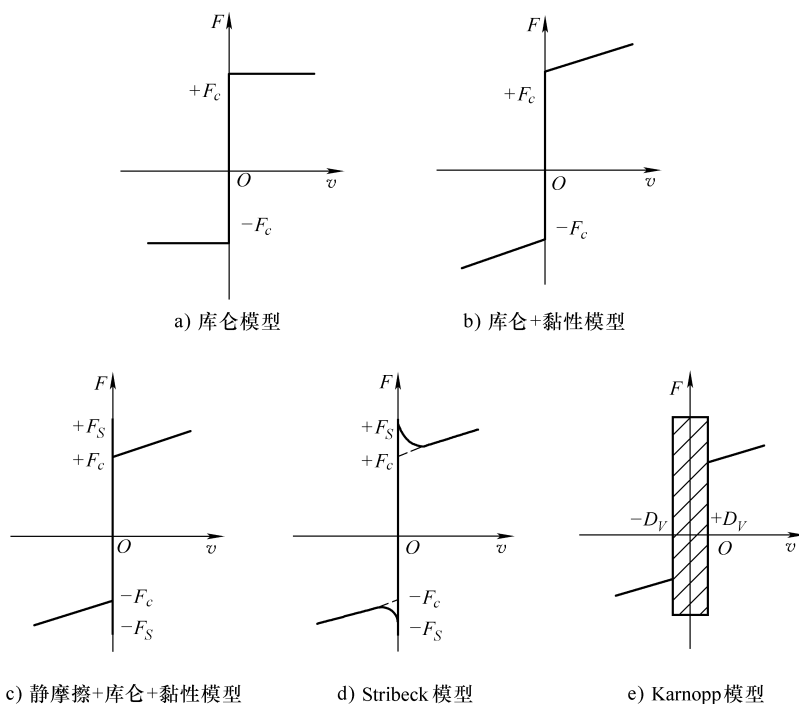


图 1-3 几类摩擦模型的示意图

$$F_f(v) = F_c \operatorname{sgn}(v) \quad (1-1)$$

式中 F_f ——摩擦力;

v ——相对滑动速度;

F_c ——库仑摩擦力, $F_c = \mu |F_n|$;

μ ——摩擦系数;

F_n ——法向力;

$\operatorname{sgn}(v)$ ——符号函数。

这是最早的关于摩擦的定义和模型, 摩擦力只是速度的函数。库仑摩擦模型局限于非零速下的摩擦, 不能够描述速度为零时的摩擦力情况。在速度为零时, 摩擦力可以为零或者介于 F_c 与 $-F_c$ 之间的任何值。而且该模型涉及符号函数, 使得系统的动力学方程变成了分段函数, 从而引入非线性使求解变得非常复杂。

2. 库仑 + 黏性模型

19 世纪, 随着流体力学的发展, 人们发现液体存在黏性, 从而导致了线性黏性摩擦模型的出现, 描述为

$$F_f(v) = F_v v \quad (1-2)$$

式中 F_v ——黏性摩擦系数;
 v ——相对滑动速度。

在某些情况下,为了更好地与试验数据相拟合,也可以建立一种与相对滑动速度成非线性关系的黏性摩擦模型,摩擦力可表示为下式的形式

$$F_f(v) = F_v |v|^{\delta_v} \text{sgn}(v) \quad (1-3)$$

式中 δ_v ——取值依赖于应用表面的几何形状。

线性黏性摩擦模型通常与库仑摩擦模型组合使用,进而发展成为另一种简单的库仑+黏性摩擦模型,摩擦力可表示为下式的形式,如图 1-3b 所示。

$$F_f(v) = F_v v + F_c \text{sgn}(v) \quad (1-4)$$

3. 静摩擦 + 库仑 + 黏性模型

Morin 于 1833 年又引入了静摩擦力的概念,也就是静止时静摩擦力与外力相互作用的思想。试验发现使系统从零速到达一个稳态速度的力要比保持这个稳态速度所需的力大,即静摩擦力水平要高于库仑摩擦力。所以,静态摩擦力 F_{static} 与外力 F_e 有关,有如下函数关系

$$F_{static} = \begin{cases} F_e, & v=0, \quad |F_e| \leq F_s \\ F_s \text{sgn}(F_e), & v=0, \quad |F_e| > F_s \end{cases} \quad (1-5)$$

式中 F_s ——最大静摩擦力。

当 $v=0$ 时,摩擦力是外力的函数而不是速度的函数,所以采用传统方式以速度为输入、力为输出来描述摩擦力并不是完全正确的。

随着静摩擦概念的引入,形成了至今还广泛使用的经典传统(静摩擦+库仑+黏性)摩擦模型,摩擦力可表示为下式的形式,如图 1-3c 所示。

$$F_f(v) = F_{static} + F_v v + F_c \text{sgn}(v) \quad (1-6)$$

4. Stribeck 摩擦模型

在上面介绍的摩擦模型均可称为经典摩擦模型,模型中滑动摩擦力都是速度的线性函数,并且静摩擦和动摩擦之间的转换是离散的。但是, Stribeck 在 1902 年试验观察到:摩擦力并不像图 1-3c 描述的那样,当克服摩擦力后不连续地下降,而是在低速下随着速度的增加而减小,呈现为速度的连续函数。这一现象也称为负斜率摩擦现象。Bo 和 Pavelescu 在 1982 年提出了一个指数模型来描述 Stribeck 现象^[7],表达式如下

$$F_f(v) = F_c + (F_s - F_c) e^{-(v/v_s)^\delta} \quad (1-7)$$

式中 v_s ——Stribeck 速度;

v_s 和 δ ——经验常数。

该模型后经 Armstrong 完善,添加了黏性摩擦项,最后的表达式如下

$$F_f(v) = F_c + (F_s - F_c) e^{-(v/v_s)^\delta} + F_v v \quad (1-8)$$

不同学者对 δ 的取值有不同看法, Bo 和 Pavelescu 将 δ 的取值范围定为 0.5~1 之间,而其他学者常取 $\delta=1$ 或 $\delta=2$, 当 $\delta=1$ 时就得到 Tustin 模型^[8], 当 $\delta=2$ 时就得到 Gauss 指数模型^[9]。Gauss 指数模型与 Lorentzian 模型^[10]近似等效, Lorentzian 模型的形式如下

$$F_f(v) = F_c \operatorname{sgn}(v) + (F_s - F_c) \frac{1}{1 + (v/v_s)^2} + F_v v \quad (1-9)$$

将式 (1-5) 和式 (1-8) 综合起来, 可以形成一种比经典摩擦模型更为一般的摩擦模型^[3], 表达式如下

$$F_f = \begin{cases} F_f(v) & v \neq 0 \\ F_e & v = 0 \text{ 且 } |F_e| < F_s \\ F_s \operatorname{sgn}(F_e) & \text{其他} \end{cases} \quad (1-10)$$

式中 $F_f(v)$ ——由式 (1-8) 给出。

该模型很好地描述了低速下的摩擦力行为。用一个衰减指数项体现了负斜率摩擦现象。试验已表明指数模型能以 90% 的精度近似拟合该区域的真实摩擦力。图 1-3d 绘制的曲线称为 Stribeck 曲线。该曲线体现了摩擦力与稳态速度之间的对应关系, 也体现了摩擦力的静态特性。该模型表达出了在非常低的速度下, 由于 Stribeck 效应的存在, 摩擦力产生不稳定效应。基于该模型可以研究由摩擦引起的黏滑运动和极限环等现象。

5. Karnopp 摩擦模型

由于前述几个摩擦模型都需要根据相对滑动速度而进行几个方程之间的切换来描述摩擦力, 因此在仿真或控制中需要判断切换点及检查零速, 这是比较困难的。Karnopp 在文献 [11] 提出 Karnopp 摩擦模型, 弥补了该项不足。Karnopp 摩擦模型定义了一个零速区间 $|v| \leq D_V$, 如图 1-3e。 D_V 是零附近非常小的速度值, 根据不同的工作条件而确定。在 $\pm D_V$ 区域外, 摩擦是速度的函数, 而在 $\pm D_V$ 区域之内, 速度则被强迫认为是零, 此时的摩擦力由系统所受的其他外力决定, 其大小等于外力的大小, 但要小于最大静摩擦力, 方向与外力相反, 表达式如下

$$F_f = \begin{cases} F_{\text{slip}} = \begin{cases} F_c + F_v v, & v \geq D_V \\ -F_c + F_v v, & v \leq -D_V \end{cases} \\ F_{\text{stick}} = \begin{cases} \min(F_e, F_s), & |v| < D_V, F_e \geq 0 \\ \max(F_e, -F_s), & |v| < D_V, F_e < 0 \end{cases} \end{cases} \quad (1-11)$$

该模型的优点是避免了零速度检测问题以及黏滞和滑动摩擦状态方程间切换问题^[12], 但是 D_V 的确定还没有明确方法, 虽然文献 [13] 在 Karnopp 摩擦模型参数的辨识中, 通过图解法估计了 D_V 值, 但该方法受噪声影响较大, 难以实现。该模型的另一个局限性是零速区间的概念和真实摩擦力不相符。该模型目前一般只用于稳态下摩擦力的描述, 不考虑速度变化的情况。

6. 七参数摩擦模型

为了捕捉摩擦的预滑动位移, 可变的静摩擦力以及黏滑等行为, Armstrong 和 Dupont 提出了七参数摩擦模型^[3]。该模型采用 3 个独立的方程分别描述不同阶段的摩擦, 七参数摩擦模型如下:

(1) 黏滞阶段 (预滑动位移)

$$F_f(x) = -K_t x \quad (1-12)$$

(2) 滑动阶段 (库仑 + 黏性 + 有记忆效应的 Stribeck 摩擦)

$$F_f(v, t) = -(F_c + F_v |v| + F_s(\gamma, t_2) \frac{1}{1 + (v(t - \tau_L)/v_s)^2}) \operatorname{sgn}(v) \quad (1-13)$$

其中可变的静摩擦力为

$$F_s(\gamma, t_2) = F_{s,a} + (F_{s,\infty} - F_{s,a}) \frac{t_2}{t_2 + \gamma} \quad (1-14)$$

式中 F_s ——Stribeck 摩擦的幅值;

$F_{s,a}$ ——Stribeck 摩擦前一个滑动阶段末端的幅值;

$F_{s,\infty}$ ——Stribeck 摩擦停止阶段的幅值;

K_t ——静态接触时的切向刚度;

v_s ——Stribeck 速度;

τ_L ——摩擦记忆的时间常数;

γ ——可变静摩擦参数;

t_2 ——零速区的停滞时间。

该模型是一个离散的动态摩擦模型, 7 个参数分别为 F_c 、 F_v 、 $F_{s,\infty}$ 、 K_t 、 v_s 、 τ_L 、 γ 。

1992 年, Plycarpou 和 Soom 报道了其在精确的动态摩擦测量试验中观测到的各种摩擦现象^[14], 七参数摩擦模型均能够定性地预测出这些摩擦现象, 包括预滑动、库仑摩擦、黏性摩擦、Stribeck 效应、摩擦记忆以及可变静摩擦力等。尽管如此, 由于该模型包含 7 个参数, 参数辨识非常困难, 目前还没有一套成型的参数辨识方法。模型中涉及两个状态空间的切换问题, 从物理学角度, 这样的切换是不合理的。因此, 七参数摩擦模型实际应用受到很大限制。

1.3.2 摩擦的动态模型

伴随着对摩擦行为的认识, 人们开发了摩擦的动态模型。动态摩擦模型是不仅仅把摩擦力描述为相对速度的函数, 还包括其他时变参数, 如摩擦力是速度和位移的函数。

1. Dahl 摩擦模型

著名的 Dahl 摩擦模型^[15-16]是为了更好地描述带有摩擦力矩的控制系统发展起来的, 是对动态摩擦特性的最简单描述。通过引入一个内部中间状态变量 z , Dahl 模型定义的微分方程形式为

$$\frac{dF_f}{dx} = \sigma_0 \left(1 - \frac{F_f}{F_c} \operatorname{sgn}(v) \right)^\alpha \quad (1-15)$$

式中 x ——位移;

σ_0 ——刚度系数;

α ——拟合参数, 通常情况下取为 1。

这个模型也可以用下面时域形式来描述

$$\frac{dF_f}{dt} = \frac{dF_f}{dx} \frac{dx}{dt} = \frac{dF_f}{dx} v = \sigma_0 \left(1 - \frac{F_f}{F_c} \operatorname{sgn}(v) \right)^\alpha v \quad (1-16)$$

当 $\alpha = 1$ 时, Dahl 模型成为

$$\frac{dF_f}{dt} = \sigma_0 v - \frac{F_f}{F_c} |v| \sigma_0 \quad (1-17)$$

若令

$$F_f = \sigma_0 z \quad (1-18)$$

则式 (1-18) 变为

$$\dot{z} = -\frac{\sigma_0}{F_c} |v| z + v \quad (1-19)$$

当时间 t 趋于 ∞ 时, 由式 (1-18) 和式 (1-19) 给出的稳态摩擦力为

$$F_f = \sigma_0 z = F_c \operatorname{sgn}(v) \quad (1-20)$$

此时, Dahl 摩擦模型退化为库仑摩擦模型。

Dahl 模型是最简单的动态摩擦模型, 理论上容易理解, 是建立其他动态模型的重要基础。该模型是连续模型, 利用切向柔顺行概念将预滑动位移引入摩擦模型, 避免了静态模型中状态切换的不连续问题。Dahl 模型描述了静摩擦下接触峰的弹簧行为, 并且开创性地引入了平均变形的概念, 即用状态变量 z 来描述无数个接触峰的平均变形。该模型的数学特性在文献 [17] 中已被研究, 包括解的存在性和唯一性, 以及迟滞作用等。虽然 Dahl 模型描述了预滑动位移, 也可以预测摩擦滞后, 但是它既没有描述静摩擦力, 也没有捕捉到 Stribeck 效应^[18]。

2. Bliman – Sorine 模型

Bliman 和 Sorine 受 Dahl 模型的启发, 试图将 Stribeck 摩擦加入该模型, 在 Rabinowicz 试验^[19]的基础上提出了 Bliman – Sorine 模型^[20–21]。该模型是一个 2 阶 Dahl 模型, 可以看成两个 1 阶 Dahl 模型的关联。假设摩擦力依赖于速度符号 $\operatorname{sgn}(v)$ 和空间变量 s

$$s = \int_0^t |v(\tau)| d\tau \quad (1-21)$$

Bliman – Sorine 模型还可以通过微分方程来表示

$$\begin{cases} \frac{dx_s}{ds} = Ax_s + Bv_s \\ F_f = Cx_s \end{cases} \quad (1-22)$$

式中 $v_s = \operatorname{sgn}(v)$ 。

1 阶摩擦模型为

$$A = -\frac{1}{\varepsilon_f}, B = \frac{F_1}{\varepsilon_f}, C = 1 \quad (1-23)$$

式 (1-22) 可写为

$$\frac{dF_f}{dt} = \frac{dF_f ds}{ds dt} = \frac{dF_f}{ds} v = \frac{F_1}{\varepsilon_f} \left(v - \frac{F_f}{F_c} |v| \right)^\alpha \quad (1-24)$$

当 $F_1 = F_c$, $F_1/\varepsilon_f = \sigma_0$, $\alpha = 1$ 时, 其形式与 Dahl 模型是一致的。

1 阶模型式 (1-22) 未能描述静摩擦, 因此需引入 2 阶模型

$$\mathbf{A} = \begin{pmatrix} -1/(\eta\varepsilon_f) & 0 \\ 0 & -1/\varepsilon_f \end{pmatrix}, \mathbf{B} = \begin{pmatrix} F_1/(\eta\varepsilon_f) \\ -F_2/\varepsilon_f \end{pmatrix}, \mathbf{C} = (1 \quad 1) \quad (1-25)$$

式中 η 、 ε_f 、 F_1 和 F_2 ——模型相关参数。

Bliman - Sorine 证明了 1 阶和 2 阶模型都是耗散系统^[21]，当 ε_f 趋向于零时，1 阶模型近似于经典的库仑摩擦模型，而 2 阶模型近似于静摩擦 + 库仑摩擦模型。Bliman - Sorine 模型只能给出运动时瞬时速度的 Stribeck 效应，而当速度和摩擦力处于稳态关系时，则无法描述 Stribeck 效应，而且该模型不能描述摩擦记忆和可变静摩擦力等现象，因此限制了其应用。

3. 鬃毛模型

为了从微观角度来描述两个接触表面间接触点的随机行为，Haessing 和 Friedland 提出了鬃毛模型^[22]，将接触表面看成是大量的弹性鬃毛的接触。鬃毛模型的简化原理如图 1-4 所示

当接触面产生相对运动时，滑动体上的易弯鬃毛相对于固体上的刚性鬃毛之间将像弹簧一样相互作用从而产生摩擦力，该摩擦力定义为

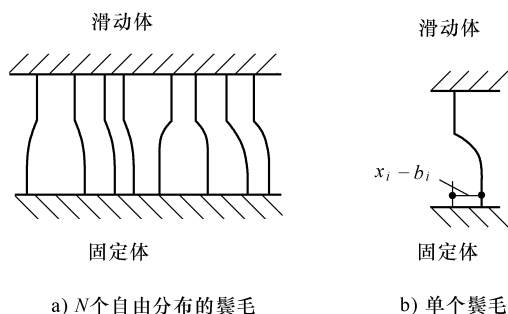


图 1-4 鬃毛模型示意图

$$F_f = \sum_{i=1}^N \sigma_0 (x_i - b_i) \quad (1-26)$$

式中 N ——鬃毛的总数；

σ_0 ——易弯鬃毛的刚度；

x_i ——易弯鬃毛的相对位置；

b_i ——刚性鬃毛位置。

界面间的摩擦力是每一对结合的鬃毛间产生的力的总和。当易弯鬃毛的挠曲超过 Δ 时，结合点会断开，而在瞬间新的结合点的随机位置又会形成，这个位置定义为

$$b_{i+1} = b_i + \text{uniform}(\Delta) \text{sgn}(x_i - b_i) \quad (1-27)$$

函数 uniform 是随机分布的鬃毛的位置，分布距离范围为 $0 \sim \Delta$ 。

随着鬃毛数量 N 的增加，模型的复杂性将增加，当采用 20 ~ 25 根鬃毛时，得到的效果较好。鬃毛模型的优点是较为准确地捕捉了微观摩擦的随机特性，但是它没有考虑到计算耗时的问题。由于鬃毛间的空间非常细小以及鬃毛突然断开的非连续性，使得该模型在仿真中的积分算法需要极短的时间步长，计算耗时量非常大，因而在数值计算上该模型的效果不是很好，一般不用于仿真。

4. 复位积分模型

在文献 [22] 中，Haessing 和 Friedland 还提出了复位积分模型，该模型是为了减少鬃毛模型的计算量而设计的，同时它保留了鬃毛模型描述微观黏滑摩擦现象的能力。该模型中引入了一个附加状态变量 z 来描述鬃毛结合部位的应变

$$\frac{dz}{dt} = \begin{cases} 0, & v > 0, z \geq z_0 \text{ 或 } v < 0, z \leq -z_0 \\ v, & \text{其他} \end{cases} \quad (1-28)$$

摩擦力表示为

$$F_f = (1 + \alpha(z)) \sigma_0(v) z + \sigma_1 \frac{dz}{dt} \quad (1-29)$$

式中 $\sigma_0(v)$ ——刚度，速度的函数；

$\sigma_1 \frac{dz}{dt}$ ——阻尼项，其只有在黏滞情况下才有效。

静态摩擦力根据函数 $\alpha(z)$ 来获得

$$\alpha(z) = \begin{cases} \alpha, & |z| < z_0 \\ 0, & \text{其他} \end{cases} \quad (1-30)$$

如果 $|z| < z_0$ ，该模型描述的黏滞阶段摩擦，摩擦力是 z 的函数。当鬃毛变形达到它的最大值 z_0 时，变量 z 保持为常数，摩擦力值下降，因为 $\alpha(z) = 0$ 。在滑动状态下摩擦力为速度的任意函数，由 $\sigma_0(v)$ 项决定。当复位积分器的弹簧应变率和阻尼系数达到无穷大时，复位积分模型可以接近 Karnopp 摩擦模型。该模型在仿真方面比鬃毛模型有效，但是变量 z 是不连续的，而且还涉及 $|z| < z_0$ 的检测问题。

5. LuGre 摩擦模型

在瑞典兰德工程技术学院和法国格勒诺布尔实验室的共同努力下，法国学者 Canudas de Wit 在 Dahl 摩擦模型的基础上提出了 LuGre 模型^[23]。该模型是 Dahl 模型的扩展，同时采纳了鬃毛模型的思想，即在微观下接触表面可以看成是大量的具有随机行为的弹性鬃毛。不同的是，鬃毛模型描述的是摩擦的随机行为，而 LuGre 模型是基于鬃毛的平均变形来建模。

鬃毛的平均变形用状态变量 z 来表示，建模为

$$\dot{z} = v - \frac{\sigma_0 |v|}{g(v)} z \quad (1-31)$$

摩擦力由鬃毛的挠曲产生，可以描述为

$$F_f = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v \quad (1-32)$$

式中 σ_0 ——鬃毛的刚度；

σ_1 ——微观阻尼系数；

σ_2 ——黏性摩擦系数。

函数 $g(v)$ 描述了 Stribeck 效应，方程形式为

$$g(v) = F_c + (F_s - F_c) e^{-\left|\frac{v}{v_s}\right|^p} \quad (1-33)$$

式 (1-32) 中阻尼系数 σ_1 一般为常数，称为 LuGre 模型的标准参数化形式。当 $g(v) = F_c$ ， $\sigma_1 = \sigma_2 = 0$ 时，LuGre 模型可以简化成 Dahl 模型。

若假设鬃毛的平均变形处于稳态运动，即 $\dot{z} = 0$ ，并且忽略黏性摩擦项 $\sigma_2 v$ ，则稳态情况下的摩擦力就成为 Stribeck 摩擦，转变成下面形式

$$F_f = \sigma_0 z = g(v) \operatorname{sgn}(v) \quad (1-34)$$

LuGre 模型中共有 6 个参数, 其中 σ_0 和 σ_1 是动态参数, F_c 、 F_s 、 v_s 和 σ_2 是静态参数。静态参数的辨识相对比较简单, 而动态参数的辨识由于引入不可测量的状态变量 z 比较复杂。该模型用一个 1 阶微分方程描述了诸多摩擦现象, 包括库仑摩擦、黏性摩擦、预滑动、可变静摩擦力、Stribeck 摩擦和摩擦滞后等, 较其他模型更能体现真实摩擦现象。文献 [24] 从数学角度讨论了 LuGre 模型具有耗散性的充分和必要条件。

和离散的 Armstrong 七参数摩擦模型相比, LuGre 模型属于连续模型, 不同摩擦状态之间能够平滑地过渡, 更容易实施。Bliman – Sorine 模型和 LuGre 模型是 Dahl 模型的两个不同扩展形式, 二者之间的详细关系见文献 [25, 26]。

LuGre 模型也可以考虑为摩擦系数, 如文献 [27] 中对回转关节处的摩擦建立了如下模型

$$M_f = rF_N\mu(z, \dot{z}, v) \quad (1-35)$$

式中 r ——摩擦力作用半径;

F_N ——正压力。

摩擦系数 μ 的形式就是 LuGre 模型

$$\mu = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v \quad (1-36)$$

LuGre 模型在具体应用时, 也可做适当改动, 如文献 [28] 针对两连杆平面机械臂, 在原始 LuGre 摩擦模型的基础上, 对两个关节建立了变化的 LuGre 摩擦模型

$$\frac{dz_i}{dt} = v_i - \frac{|v_i|}{g_i(v_i)}\sigma_{0,i}z_i \quad (1-37)$$

$$\tau_{f,i} = \sigma_{0,i}z_i + \sigma_{1,i}\dot{z}_i + f_i(v_i) \quad (1-38)$$

为了更好地拟合该文中的数据, 公式中函数 $g_i(v_i)$ 和 $f_i(v_i)$ 均采用了不同于原始 LuGre 摩擦模型的形式

$$g_i(v_i) = \alpha_{0,i} + \alpha_{1,i} \exp \left[-\frac{v_i}{v_{s1,i}} \operatorname{sgn}(v_i) \right] + \alpha_{2,i} \left(1 - \exp \left[-\frac{v_i}{v_{s2,i}} \operatorname{sgn}(v_i) \right] \right) \quad (1-39)$$

$$f_i(v_i) = \alpha_{3,i}v_i + \alpha_{4,i}v_i^2 \quad (1-40)$$

式中 $v_{s1,i}$ 、 $v_{s2,i}$ ——模型参数。

6. Leuven 摩擦模型

尽管 LuGre 摩擦模型是个比较完善且容易实施的摩擦模型, 但仍有学者指出了不足之处。Swevers^[29]指出 LuGre 模型的缺点主要表现在对迟滞现象描述不充分: 没有解释试验中观测到的非局部记忆效应, 不能调整任意的“位移 – 力”过渡曲线, 而且在预滑动阶段过于耗散。

因此, Swevers 基于 LuGre 模型提出了一种更加精细的 Leuven 摩擦模型。该模型采用状态变量 z 来描述鬃毛的平均变形, 由两个方程组成: 非线性状态方程和摩擦力方程, 其形式为

$$\frac{dz}{dt} = v \left(-\operatorname{sgn} \left(\frac{F_d(z)}{s(v) - F_b} \right) \left| \frac{F_d(z)}{s(v) - F_b} \right|^n \right) \quad (1-41)$$

$$F_f = F_h(z) + \sigma_1 \dot{z} + \sigma_2 v \quad (1-42)$$

式中 n ——描述过渡曲线的系数；

$s(v)$ ——描述常速下的摩擦行为，其表达式为

$$s(v) = \operatorname{sgn}(v) (F_c + (F_s - F_c) e^{-(|v|/v_s)^\delta}) \quad (1-43)$$

式中 $F_h(z)$ ——迟滞力，是具有非局部记忆的静态非线性力，由两部分组成

$$F_h(z) = F_b + F_d(z) \quad (1-44)$$

式中 F_b ——在速度反向时过渡曲线的起始位置；

$F_d(z)$ ——一定时间内产生的过渡曲线。

Leuven 模型不仅包含了 LuGre 模型所能描述的摩擦特性，还采用了具有非局部记忆的迟滞函数对预滑动区间建立了更为精确的模型。二者的主要区别点在于 LuGre 模型是用一个线性函数 $\sigma_0 z$ 来描述迟滞摩擦力，而 Leuven 模型则用了一个迟滞函数 $F_h(z)$ 。虽然 Leuven 模型更加准确，但是它比较复杂，需要 6 个参数和 3 项机制（即速度反向、内环闭合和迟滞模型复位）来描述摩擦行为，因此辨识参数的难度大于 LuGre 模型。

7. 广义 Maxwell 摩擦模型

文献 [32] 在 Maxwell 摩擦模型^[30-31]的基础上提出了广义 Maxwell 模型，采用无质量的块弹簧取代了原模型中的弹簧。这些块弹簧在静摩擦阶段表现为弹簧，在滑动阶段会展示摩擦的滑动特性，如 Stribeck 效应。广义 Maxwell 模型的主要思想是：采用一定数目并联的弹塑性弹簧来模拟两个表面间微观接触的凸起，每一个弹簧都有自己固有的迟滞特性，

以第 i 个块弹簧单元为研究对象，在稳态滑动阶段，其状态方程为

$$\dot{F}_i(z) = \operatorname{sgn}(v) C \left(1 - \frac{F_i}{\alpha_i g(v)} \right) \quad (1-45)$$

式中 C, α_i ——常数；

$g(v)$ ——描述 Stribeck 摩擦。

该单元在速度不为零时会一直处于滑动状态。在黏滞阶段，其状态方程为

$$\dot{F}_i(z) = k_i v \quad (1-46)$$

式中 k_i ——刚度，是每一个单元都有的独立的参数。

当 $F_i(z) > \alpha_i g(v)$ 时，这个单元就会开始滑动。通过把所有单元的状态模型加起来再加上黏性摩擦项，就得到了总体摩擦力的表达式

$$F_f = \sum_{i=1}^M F_i(t) + \sigma_2 v \quad (1-47)$$

广义 Maxwell 摩擦模型能够精确描述预滑动区域、滑动区域，也能够捕捉最大静摩擦力、黏滑现象和摩擦滞后等。它与 LuGre 模型和 Leuven 模型的状态方程主要不同之处，就是状态方程 (1-45) 没有将速度 v 作为一个单独算子。

上述介绍了几种较为重要的动态摩擦模型，还有许多其他模型，如弹塑性摩擦模型、单

状态弹塑性摩擦模型、Valanis 摩擦模型、通用摩擦模型、时间滞后模型以及单状态和多状态积分摩擦模型等，关于这些模型的详细信息请参阅相关文献。总的来看，动态摩擦模型比静态摩擦模型具有更好的连续特性，能更好描述摩擦的非线性行为。但是，对摩擦行为描述的越全面，其参数辨识难度越大。Dahl 模型由于其结构的简单性，已经被用于自适应摩擦补偿研究。LuGre 模型是一个比较完善的摩擦模型，能够准确的预测摩擦的各种重要特性，由于其对摩擦环节动态补偿效果较好，因此，该模型以成为工程应用和控制领域研究的一个热点。其他动态摩擦模型由于其自身参数辨识以及计算耗时等方面的局限性，致使它们在实际工程中的应用较少。

1.3.3 基于智能系统的摩擦模型

智能系统（如神经网络和模糊系统）自 20 世纪 80 年代以来，在理论上取得了许多突破性的成果，同时也在众多工业领域获得了成功应用。智能系统具有以任意精度逼近连续函数的性质，为此人们采用 RBF 神经网络、CMAC 神经网络、T-S 模糊系统和模糊基函数模糊系统等作为摩擦辨识器，采用补偿算法将跟踪误差限制在某一界限内。智能系统方法为解决伺服系统中的摩擦问题开辟了新的途径，但各种基于智能控制的摩擦补偿方法各有其优缺点。如：神经网络的训练时间较长、算法实时性差、系统的暂态响应难以保证；模糊规则的获取难度大等。关于用智能系统建立摩擦模型将在后面的补偿控制中一并论述。

1.3.4 基于数据驱动的摩擦模型

摩擦的动静态模型需要辨识很多参数，并且实时计算也比较耗时，这给实际应用带来了困难。最近，人们发展了基于数据驱动的摩擦模型，这是因为数据驱动的摩擦模型比传统的动静态摩擦模型所要辨识的参数少、运行速度快。如 Choudhury 模型、Kano 模型、He 模型和 Stenman 模型等（图 1-5 ~ 图 1-7），这些数据驱动的摩擦模型在文献 [4] 进行了详细描述。国内对这方面介绍和研究比较少，数据驱动的摩擦模型更便于实际应用。

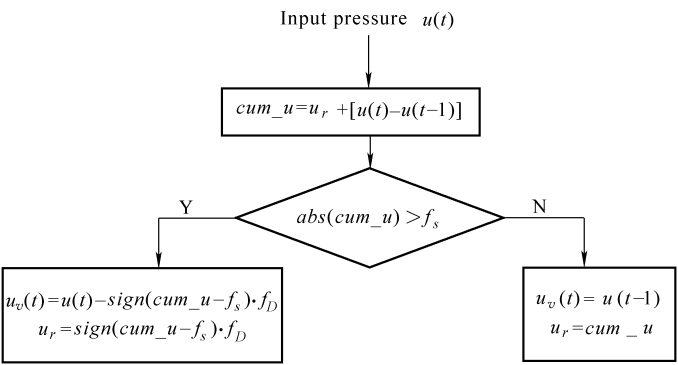


图 1-5 数据驱动的 He 模型

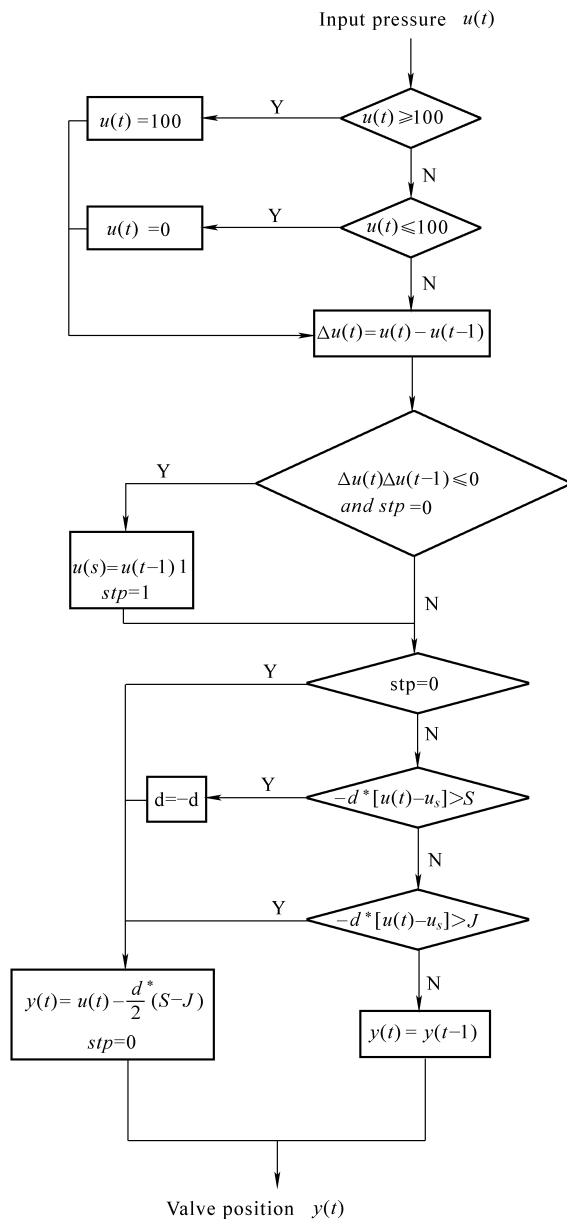


图 1-6 数据驱动的 Kano 模型

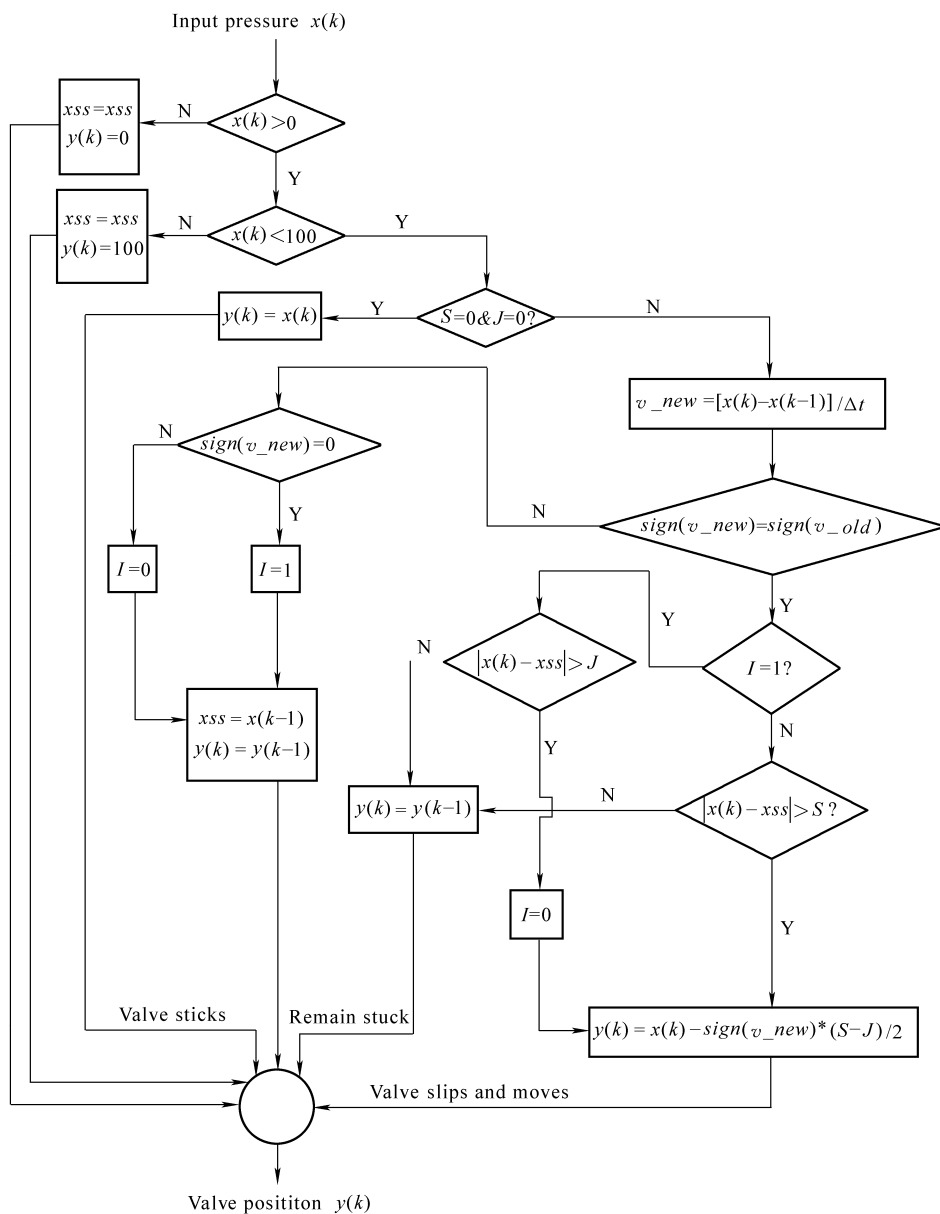


图 1-7 数据驱动的 Choudhury 模型

1.4 非线性摩擦的补偿方法

1.4.1 基于传统摩擦模型的补偿方法

基于传统摩擦模型的补偿方法的实质是前馈补偿，即首先对系统中的摩擦环节建立或选择适当的数学模型，由此模型和系统的状态变量信息，对摩擦力的值进行估计，然后在控制

力中加上摩擦力的估计值,从而消除摩擦环节对系统的影响。基于摩擦模型的补偿分为固定模型补偿和自适应补偿。对于前者,摩擦模型的参数是通过离线辨识来获得的,在控制过程中保持不变;对于后者,参数在控制过程中是可变的。

1. 基于库仑摩擦模型的补偿方法

采用库仑摩擦模型进行补偿的优点在于模型简单,易于实现,但由于该摩擦模型是静态摩擦模型,无法描述零速时摩擦的非线性特性,使其控制效果受到限制。为此,很多学者提出了相应的改进措施。例如:B. Friedland 等^[33]将库仑摩擦表示成时间的函数,提出了基于非线性观测器的自适应补偿方法,通过提高观测器增益,来弥补采用固定库仑摩擦模型进行补偿的缺陷。A. Yazdizadeh 等^[34]将此方法进一步推广,将观测器增益表示成速度的函数,通过 Lyapunov 函数推导出增益函数应满足的约束条件。目前,此方法已有一些成功的应用,如:摩擦实验仪^[35]、二坐标运动系统 (X—Ytable)^[36]、液压伺服系统^[37]等。

2. 基于静摩擦 + 库仑摩擦模型的补偿方法

基于静摩擦 + 库仑摩擦模型的补偿方法中,由于在摩擦模型中加入静摩擦项,可以预测速度为零时出现的多值非线性,因此同基于库仑摩擦模型的补偿方法相比,它能更好地改善系统在零速附近的动态响应,其缺点是对速度信号的品质要求很高。例如:S. M. Phillips 等^[38]针对机械关节控制,提出了基于静摩擦 + 库仑摩擦 + 黏性摩擦的固定模型的补偿方法;杨元凯等^[39]针对三轴转台系统,提出了基于静摩擦 + 库仑摩擦模型的一维线性观测器的设计方法;Y. P. Yang 等^[40]针对静摩擦 + 库仑摩擦模型,提出了一种基于伪线性回归原理的模型参数离线辨识方法和一种自适应摩擦补偿方法;L. W. Ping 等^[41]则由 Lyapunov 函数出发,推导出基于静摩擦 + 库仑摩擦模型的全局稳定的自适应补偿方法。

3. 基于指数摩擦模型的补偿方法

指数模型虽然是静态摩擦模型,但它在描述摩擦现象时,考虑了 Stribeck 效应,这使得在超低速段 (Stribeck 段),指数模型对摩擦现象的描述更为精确,因此基于指数模型的摩擦补偿控制对于提高系统的超低速性能和抑制稳态极限环振荡有明显的效果^[42]。但由于指数模型的参数空间是非线性的,使得参数的在线辨识较为困难,这也是基于指数模型的自适应摩擦补偿的难点所在。张伟英等^[43]对 Stribeck 曲线中的负斜率段采用直线来逼近,提出了一种基于非线性观测器的低速摩擦补偿方法;C. Canudas de Wit 等^[44]首先对指数模型在 $v=0$ 处线性化,得到线性参数模型,然后对模型参数采用最小方差辨识,构成自适应摩擦补偿;M. Feemster 等^[45]通过构造 Lyapunov 函数推导出一种鲁棒自适应摩擦补偿位置控制器,以消除摩擦参数的不确定性对系统性能的影响;K. Menon 等^[46]将其应用于车床进给系统,以改善低速性能。

4. 基于 Karnopp 摩擦模型的补偿方法

采用基于 Karnopp 模型的摩擦补偿方法的突出优点在于对速度信号的测量精度要求不高,且能较好地改善系统在零速时的动态响应,因此在实践中深受控制系统工程师们喜爱。K. C. Cheok 等^[47]将 Karnopp 摩擦模型的参数辨识归结为一个非线性优化问题,采用非线性编程技术作为在线优化方法,对摩擦参数进行辨识;C. T. Johnson 等^[47-48]提出了一种新颖的方法来离线辨识 Karnopp 模型参数,即将跟踪误差经过信号处理,逐步辨识出摩擦参数;

S. W. Lee 等^[49]提出一种基于 Karnopp 摩擦模型的摩擦补偿方案,对摩擦参数采用进化策略来离线辨识,通过自适应调整控制器的滑动输入量,在建模误差和外扰动存在的情况下,使跟踪误差收敛到预定义的区域;J. Y. Yen 等^[50]将 Karnopp 模型加以改进,使其可以描述与位置有关的摩擦特性,并在此基础上提出了鲁棒控制器。

5. 基于 LuGre 摩擦模型的补偿方法

20 世纪 90 年代以前,基于动态摩擦模型的控制补偿方法的研究较少,只有 G. D. Walrath 提出了一种基于 Dahl 模型的自适应摩擦补偿控制^[51]。1995 年, C. Canudas de Wit 等人提出了 LuGre 模型^[23]。这是一个较为完善的动态摩擦模型,它能精确地描述摩擦的各种动态和静态特性。目前,基于 LuGre 模型的摩擦补偿控制已成为理论和应用研究的一个热点。该方法的优点在于对摩擦环节的动态特性的补偿效果好,其难点是参数辨识很困难。C. Canudas de Wit 等提出一种基于 LuGre 模型的摩擦补偿算法,并证明了误差的渐进收敛性;文献 [52, 53] 中,对这种方法进行了改进,在摩擦观测器中引入比例系数,以适应在工作过程中摩擦参数的不确定性。同时,还研究了 LuGre 摩擦模型的参数辨识问题,提出一种采用非线性优化的两步辨识方法;R. M. Hirschorn 等^[54]通过构造 Lyapunov 函数,提出一种基于 LuGre 模型的动态补偿算法;P. Vedagarbha 等^[55]提出 3 种不同的观测器设计方法,对 LuGre 摩擦模型中状态变量 z 进行估计;K. M. Mis2ovec 等^[56]提出了基于 LuGre 模型的鲁棒自适应摩擦补偿方法,在参考输入为持续激励的情况下,模型中所有参数的估计都渐进收敛到真值。采用 LuGre 摩擦模型的应用研究也取得了很大进展,见文献 [57 - 59]。

1.4.2 非模型的补偿方法

不依赖于摩擦模型的传统补偿方法具有悠久的历史,方法种类繁多,主要思想是将摩擦视为外干扰,通过改变控制结构或控制参数来提高系统抑制干扰的能力,从而抑制摩擦。

1. PID 控制方法

高增益 PID 控制是人们最早使用的抑制摩擦非线性的控制器。PID 控制中的微分项能增大系统阻尼,由于摩擦记忆特性,采用 PID 控制可以在一定程度上改善低速跟踪性能,抑制爬行现象^[60],但对实际系统,增益过高会导致不稳定,使该方法的应用受到限制。文献 [61] 中研究了非线性 PID 控制方法,即根据系统的状态来调整 PID 各系数,更好地对摩擦进行抑制。

2. 信号抖动方法

通过在指令上叠加抖动信号,可以对摩擦产生补偿作用。抖动信号具有较高频率,加入系统后,能够在一定程度上平滑摩擦在低速时的不连续性。对于液压伺服系统,抖动信号的应用较为成功。S. Lee 等^[62]研究了抖动信号同系统各环节传递函数之间的关系,由此得出最优抖动信号的参数确定方法。L. Horowitz 等^[63]研究了抖动信号的频率对补偿效果的影响。

3. 脉冲控制方法

脉冲信号是具有大幅值、短周期的信号,在控制力矩中施加脉冲信号可以产生微小的位移,从而摆脱静摩擦的束缚。该方法直观、简单,但控制效果一般。脉冲控制与抖动信号控制不同,脉冲序列直接驱动执行元件运动到给定位置,而抖动信号是一种叠加在输入信号上

的高频信号。S. Yang 等^[64]提出一种控制方法,在低速时采用脉冲控制,并自适应调整脉冲宽度。B. Armstrong 等^[65]将脉冲控制应用到机器人系统中。

4. 力矩反馈方法

力矩反馈控制是一种基于力矩传感器的控制技术,通过在连接轴上安装力矩传感器对输出净力矩进行测量,形成力矩反馈回路来稳定净力矩。传感器安装在负载端,这样就能将摩擦环节包括在力矩闭环内,如果力矩闭环有足够的带宽,就能很好地抑制摩擦力矩和其他干扰力矩的影响^[66]。虽然这种方法具有不依赖于模型、控制效果好的优点,但由于传感器价格高,安装困难,且安装后增加了系统柔性,所以其应用并不广泛。G. Morel 等^[67]在机器人关节控制中,提出将力矩传感器安装在机器人基部,然后由基部测量的力矩信号,计算出各关节上的净力矩,构成力矩反馈回路。

5. 基于干扰观测器的鲁棒控制

基于干扰观测器的摩擦补偿是一种不依赖摩擦模型的补偿方法,它是目前理论研究的一个热点。干扰观测器的设计方法属于鲁棒控制的范畴,其原理是通过建立控制对象的名义模型,由实际对象和名义模型之间的输出误差,得到包括摩擦在内的各种干扰力矩的等效力矩,然后对其进行补偿。由于该方法是一种线性补偿方法,它对摩擦非线性的补偿程度取决于 Q 滤波器的带宽,而该带宽的提高又受到实际系统中机械谐振等因素的限制,这正是采用该补偿方法所存在的问题。A. Tesfaye 等^[68]采用数值优化方法,使系统灵敏度函数和选定的目标灵敏度函数达到匹配,降低了对名义模型精确程度的要求;目前,基于干扰观测器的控制方法已经在不同类型的伺服系统控制中取得了成功应用,见文献 [69-72]。

6. 变结构摩擦补偿

变结构控制是一种非线性控制,将变结构控制应用于伺服系统中的研究很多^[73-74]。在稳态摩擦补偿问题上,变结构控制的应用非常成功,文献 [75] 中对连续伺服系统提出了变结构摩擦补偿方法,当系统进入稳态后,采用变结构控制使稳态误差为零;M. S. Kang^[76]提出一种基于变结构控制的离散系统稳态摩擦补偿方法。对于动态摩擦补偿,有关变结构补偿方法的研究处于起步阶段,K. K. Young 等^[77]考虑了速度过零时摩擦的不连续性,提出一种基于变结构的摩擦补偿方法;S. Sivakumar 等^[78]提出了考虑摩擦模型共性情况下的伺服系统滑模控制的一般设计方法。P. Korondi 等^[79]采用基于滑动模态的干扰观测器,对系统中的不连续干扰进行观测。迄今为止,有关传统补偿方法的研究仍是摩擦补偿领域的主流,其优点是控制算法相对简单、实时性好。其中,基于摩擦模型的补偿方法的不足之处在于摩擦模型的选择、模型参数的确定过程较为烦琐,同时,由于摩擦力矩是速度的函数,控制效果依赖于速度信号的品质。不基于模型的传统补偿方法虽然原理上简单,但对零速时摩擦非线性的补偿能力有限,补偿能力提高涉及伺服系统中的其他问题,如机械谐振、参数时变等。鉴于这些问题,许多学者开始尝试用智能控制来实现摩擦补偿,目前,基于智能控制的摩擦补偿研究已成为解决摩擦问题的一个研究方向。

7. 基于扩展 Kalman 滤波器的摩擦补偿

Kalman 滤波是 Kalman 于 1960 年提出的用递归方法解决离散数据线性滤波问题的一种滤波算法,Kalman 滤波是一种高效率的递归滤波器(自回归滤波器),它能够从一系列的不

完全包含噪声的测量中,估计动态系统的状态。扩展 Kalman 滤波器 (Extended Kalman Filter, EKF) 是在 Kalman 滤波器的基础上发展而来的,能对非线性系统的状态量进行估计。Opert 等^[80]即采用 EXF 来估计系统摩擦力矩,同时也实现了滚子丝杠系统的摩擦补偿。

1.4.3 智能系统的摩擦补偿方法

智能控制理论在 20 世纪 90 年代取得了很大的发展,同传统的控制方法相比,智能控制方法不需要对象的数学模型。基于智能控制的摩擦补偿研究包括如下几种方法。

1. 重复控制、学习补偿控制方法

重复控制是基于内模原理的控制方法,对于周期信号,重复控制通过迭代调整控制器的输出来补偿上一个周期的跟踪误差,最终使稳态误差为零^[81]。该方法控制效果很理想,其缺点在于学习过程较长,且只适用于周期信号。E. D. Tung 等^[82]采用重复控制的方法来补偿低速运动时的摩擦,在二坐标运动系统上的实验结果表明不超过 12 个学习周期,就可以达到完全跟踪;S. I. Cho 等^[83]提出了一种新的学习控制方法,当系统对周期信号的响应达到稳态以后,才进行迭代学习,直到跟踪误差为零;陈永生等^[84-85]将重复控制方法应用于三轴转台系统,通过在重复控制延时回路中串入低通滤波环节来增强鲁棒性;胡恒章等^[86]在三轴转台系统控制中,采用异步自学方法来补偿由于静摩擦死区所造成的正弦运动失真。

2. 神经网络补偿控制方法

神经网络控制自 20 世纪 80 年代以来,在理论上取得了许多突破性的成果,同时也在众多工业领域获得了成功应用。神经网络具有以任意精度逼近连续函数的性质,但对于分段连续函数,其逼近能力有限。而采用神经网络对摩擦进行补偿,需要神经网络具有逼近分段连续函数的能力,这正是应用神经网络进行摩擦补偿的难点所在^[87]。为此,R. S. Rastko 等^[88]提出了一种新的神经网络结构,具有额外的“跳变”神经元,以实现分段连续函数的逼近,并提出一种基于此神经网络的死区补偿方法。目前,采用 CMAC (小脑模型关联控制器) 网络来补偿摩擦环节的研究很活跃,同自适应补偿方法相比,在有噪声和非线性较为严重的情况下,CMAC 网络控制具有明显的优点,同时,CMAC 网络学习速度快,易于硬件实现。文献 [89-91] 中分别提出了基于 CMAC 网络的摩擦补偿控制,通过对 CMAC 网络权值的迭代调整来辨识摩擦,实现高精度控制。H. L. Du 等^[92]采用 RBF 网络作为摩擦辨识器,采用补偿算法将跟踪误差限制在某一界限内。Y. H. Kim 等^[93]采用神经网络来辨识系统中的摩擦和干扰力矩,提出了基于增强学习的摩擦补偿方法。我们最近在文献 [94] 提出了用模糊神经网络对摩擦进行在线建模并用数据挖掘技术获取初始权值,在控制器中用该网络作为补偿主动控制单元,系统取得了良好的控制性能。还有其他的神经网络摩擦补偿方法,如文献 [95-96]。

3. 模糊补偿控制方法

自 1965 年 L. A. Zadeh 提出模糊集的概念以来,到现在模糊控制的发展已日益成熟。一些学者对模糊控制在伺服系统中的应用进行了研究^[97],但针对摩擦补偿方面的研究较少,且控制效果不明显。J. T. Teeter 等^[98]提出一种简单的模糊摩擦补偿方法,仅采用一条模糊

规则调整 PI 控制的参数; A. Tzes 等^[99]对于摩擦非线性环节, 采用模糊聚簇技术进行建模, 从大量数据中, 提取出摩擦环节的模糊模型, 并进行补偿; 黄进等^[100]提出采用自调整量化因子模糊控制器的摩擦补偿方法, 可根据系统参数变化在线调整量化因子, 增强补偿效果; Z. Kovacic 等^[101]针对伺服系统中的非线性, 提出一种自校正模糊逻辑控制器。但早期这些研究在以下几个方面需要进一步改进和提高:

1) 无论是摩擦建模还是摩擦补偿控制器设计都缺乏自适应特性。因为实际摩擦力是随着机械系统的磨合等情况在变化, 所以无论摩擦模型还是摩擦补偿控制都应该具有随着环境变化而进行动态调整的能力, 也就是说实现自适应特性是十分重要的, 自适应特性对保证摩擦整个动态历程的模型和控制精度时十分必要的。

2) 系统设计缺乏稳定性分析。一个闭环控制系统设计包括控制器设计、稳定性分析和仿真与实验等, 但早期的这些研究都不具备稳定性分析。

3) 早期这些研究摩擦模糊规则的获取一般是基于传统方法获取, 在自动获取摩擦模糊规则方面显得比较薄弱。

近年来, 我们系统地研究摩擦的自适应模糊建模技术与补偿控制, 主要包括: 基于经验的摩擦模糊建模与自适应补偿技术、基于数据挖掘的摩擦模与自适应补偿技术、基于观测器的摩擦模糊建模与自适应补偿技术以及双调节的摩擦模糊建模与自适应补偿技术^[102-105], 等等。

4. 遗传算法

遗传算法是目前一种较为热门的最优化算法, 其主要优点是稳定性好, 在不需要额外领域知识的情况下就可以进行最优化运算, 遗传算法透过编码技术将可行解转换成染色体, 同时采用大量可行解直接在建空间中进行搜索, 在搜索过程中交换信息, 能够避免在优化过程中陷入局部最优解的困境, 向整体最优解收敛。在摩擦补偿的领域中, 多采用遗传算法来辨识摩擦模型参数。L. C. LIH^[106-107]基于李亚普诺夫稳定性理论, 对含传动柔性和摩擦的驱动轴控制运用遗传算法预先辨识摩擦模型参数; D. C. LIAW^[108]基于摩擦模型提出了利用在线遗传算法来学习模型摩擦系数, 在摩擦系数已知的情况下, 实现了用摩擦补偿控制的输入量来抵消未知摩擦的影响, 仿真证明在测量噪声和系统不确定性存在的条件下, 在线遗传算法能取得很好的摩擦补偿效果。

1.5 本章小结

随着科技的不断发展, 人类对运动控制系统的定位和跟踪精度的要求越来越高, 使得摩擦补偿已成为高精度伺服控制系统设计中的关键技术, 研究摩擦补偿问题具有很重要的理论价值和广泛的应用前景。虽然有关摩擦建模、摩擦补偿的研究已经引起控制界的广泛关注, 并取得了一些成果, 但对这一问题的解决程度还远不能令人满意。所存在的几个问题如下:

1) 由于摩擦的内在机理还在探索与研究中, 人们提出的多种摩擦模型没有形成统一的描述形式, 在实际应用中很难确定哪种摩擦模型更合适。

2) 现有的摩擦模型只是对真实摩擦过程的一种近似描述, 故在控制补偿时很难保证控

制精度。

3) 现有这些摩擦模型从形式上看比较复杂, 有关摩擦模型参数辨识方面的研究也较少, 很难进行控制系统的稳定性分析。

目前, 这一领域有如下几个发展方向:

1) 新摩擦模型的研究, 摩擦特性的研究, 摩擦现象的共性及其数学上的描述。

2) 将传统补偿方法同智能补偿方法相结合, 以弥补各自的不足。

3) 摩擦补偿的控制策略较多, 进一步研究采用多种控制策略来实现更为有效的摩擦补偿控制。

4) 基于数据驱动的摩擦智能建模与控制补偿将是另一重要研究方向, 该技术将有利于实际工程的可实现性。

参考文献

- [1] 杨绍普, 曹庆杰, 张伟. 非线性动力学与控制的若干理论及应用 [M]. 北京: 科学出版社, 2011.
- [2] 黄进, 叶尚辉. 含摩擦环节伺服系统的分析及控制补偿研究 [D]. 西安: 西安电子科技大学, 1989.
- [3] Armstrong B, Dupont P, Canudas de Wit C. A survey of models, analysis tools and compensation methods for the control of machines with friction [J]. Automatica, 1994, 30 (7): 1083 - 1138.
- [4] Claudio G. Comparison of friction models applied to a control valve [J]. Control Engineering Practice, 2008, 16: 1231 - 1243.
- [5] 刘丽兰, 刘宏昭, 吴子英, 等. 机械系统中摩擦模型的研究进展 [J]. 力学进展, 2008, 38 (2): 201 - 213.
- [6] 刘强, 尔联洁, 刘金琨. 摩擦非线性环节的特性、建模与控制补偿综述 [J]. 系统工程与电子技术, 2002, 24 (11): 45 - 52.
- [7] Bo L C, Pavelescu D. The friction - speed relation and its influence on the critical velocity of the stick - slip motion [J]. Wear, 1982, 82 (3): 277 - 289.
- [8] Tustin A. The effects of backlash and of speed dependent friction on the stability of the closed - cycle control systems [J]. J Institution of Electrical Engineering, 1947, 94 (2): 143 - 151.
- [9] Armstrong B. Stick - slip arising from stribeck friction [C]. In: Proceedings of IEEE Conference on Robotics and Automation, Cinxinati, 1994, 1377 - 1382.
- [10] Hess D P, Soom A. Friction at a lubricated line contact operating at oscillating sliding velocity [J]. Journal of Tribology, 1990, 112 (1): 147 - 152.
- [11] Karnopp D. Computer simulation of stick slip friction in the mechanical dynamics systems [J]. ASME Journal of Dynamics Systems, Measurement, and Control, 1985, 107: 100 - 103.
- [12] Iurian C. Identification of a system with dry friction [D]. Catalunya: University Politcnica de Catalunya, 2005.
- [13] Ravanbod - Shirazi L, Besancon - Voda A. Friction identification using the Karnopp model, applied to an electropneumatic actuator [J]. Journal of Systems and Control Engineering, 2003, 217: 123 - 138.
- [14] Polycarpou A, Soom A. Transitions between sticking and slipping at lubricated line contacts, friction - induced vibration, chatter, squeal, and chaos [C]. In: Proceedings of ASME Winter Annual Meeting, Anaheim, 1992, 139 - 148.

- [15] Dahl P R. Solid friction damping of spacecraft vibrations [C]. In: Proceedings of AIAA Guidance and control conference, Boston, 1975.
- [16] Dahl P R. Solid friction damping of mechanical vibrations [C]. AIAA, 1976, 1675 – 1682.
- [17] Bliman P J. Mathematical study of the Dahl' s friction model [J]. Europe J Mech A/Solids, 1992, 11 (6): 835 – 848.
- [18] Bohlin T. A case study of grey box identification [J]. Automatica, 1994, 30 (2): 307 – 318.
- [19] Rabinowicz E. The nature of static and kinetic coefficients of friction [J]. J App Phys, 1995, 22 (11): 1373 – 1379.
- [20] Bliman P A, Sorine M. Friction modeling by hysteresis operators. Application to Dahl, sticktion, and stribek effects [C]. In: Proceedings of Models of hysteresis, Italy, 1991, 1108 – 1112.
- [21] Bliman P A, Sorine M. Ease – to – use realistic dry friction models for automatic control [C]. In: Proceedings of the 3rd European control conference, Italy, 1995, 3788 – 3794.
- [22] Haessing D A, Friedland B. On the modeling and simulation of friction [J]. ASME Journal of Dynamics Systems, Measurement, and Control, 1991, 113: 354 – 362.
- [23] Canudas D W C, OLLSON H, ASTROM K J, et al. A new model for control of systems with friction [J]. IEEE Trans. on Automatic Control, 1995, 40 (3): 419 – 425.
- [24] Barabanov N, Ortega R. Necessary and sufficient conditions for passivity of the LuGre model [J]. IEEE Trans. on Automatic Control, 2000, 45 (4): 830 – 832.
- [25] Gafvert M. Comparision two friction models [D]. France; Laboratoire d' Automatique de Grenoble, 1996, 19 – 65.
- [26] Olsson H, Astrom K J. Friction model and friction compensation [J]. European Journal of Control, 1998, 4 (3): 176 – 195.
- [27] Nitsche R, Gaul L. Smart friction driven systems [J]. Smart Mater Struct, 2005, 14: 231 – 236.
- [28] Bona B, Indri M, Smaldone N. Nonlinear friction estimation for digital control of direct – drive manipulators [J]. ASME Journal of Applied mechanics, 2003, 150 (2): 171 – 176.
- [29] Swevers J, AL – Blender F, Chris G. An integrated friction model structure with improved presiding behavior for accurae friction compensation [J]. IEEE Trans. on Automatic Control, 2000, 45 (4): 675 – 686.
- [30] Iwan W D. A distributed element model for hysteresis and its steady – state dynamic response [J]. ASME Journal of Applied mechanics, 1966, 33 (4): 893 – 900.
- [31] Iwan W D. On a class of models for yielding behavior of continuous and composite systems [J]. ASME Journal of Applied mechanics, 1967, 34 (3): 612 – 617.
- [32] Lampact V, AL – Blender F, Swevers J. A generalized Maxwell – slip friction model appropriate for control purposes [C]. In: Proceedings of IEEE Conference on Robotics and Automation, Washington, 2003, 1170 – 1177.
- [33] Friedland B, Park YJ. On adaptive friction compensation [J]. IEEE Trans on Automatic Control, 1992, 37 (10): 1609 – 1612.
- [34] Yazdizadeh A, Khorasani K. Adaptive friction compensation based on the Lyapunov scheme [C]. In Proceedings of the 1996 IEEE International Conference on Control Applications, 1996: 1060 – 1065.
- [35] Amin J, Friedland B, Harnoy A. Implementation of a friction estimation and compensation technique [C]. IEEE Control Systems, 1997, 71 – 76.

- [36] Huang P Y, Chen Y Y, Chen MS. Position – dependent friction compensation for ballscrew tables [C]. In Proceedings of the 1998 IEEE International Conference on Control Applications, 1998: 863 – 867.
- [37] Tafazoli S, De Silva C W, Lawrence P D. Tracking control of an electrohydraulic manipulator in the presence of friction [J]. IEEE Trans. on Control Systems Technology, 1998, 6: 401 – 411.
- [38] Phillips S M, Ballou K R. Friction modeling and compensation for an industrial robot [J]. Journal of Robotic Systems, 1993, 10 (7): 947 – 971.
- [39] 扬元凯, 郎需英. 精密转台中摩擦力矩的动态补偿 [J]. 自动化学报, 1983, 9 (4): 248 – 252.
- [40] Yang Y P, Chu J S. Adaptive velocity control of DC motors with coulomb friction identification [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1993, 115: 95 – 102.
- [41] Weiping L, Cheng X. Adaptive High – precision control of positioning tables – theory and experiments [J]. IEEE Trans. on Control Systems Technology, 1992, 2: 265 – 270.
- [42] Canudas De Wit C. Robust control for servo – mechanisms under in exact friction compensation [J]. Automatica, 1993, 29 (30): 757 – 761.
- [43] 张伟英, 张友安. 非线性观测器用于高精度甚低速系统的动态补偿 [J]. 控制与决策, 1989, (1): 35 – 37.
- [44] Canudas de Wit C, Noel P, Brogliato B. Adaptive friction compensation in robot manipulators: low – velocities [J]. International Journal of Robotics Research, 1993, 10 (3): 189 – 199.
- [45] Feemster M, Vedagarbha P, Dawson D M, Haste D. Adaptive control techniques for friction compensation [J]. Mechatronics, 1999, 9: 125 – 145.
- [46] Menon K, Krishnamurthy K. Control of low velocity friction and gear backlash in a machine tool feed drive system [J]. Mechatronics, 1999, 9: 33 – 51.
- [47] Cheok KC, Hu H, Loh N K. Modeling and identification of a class of servomechanism systems with Stick – Slip friction [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1988, 110: 324 – 328.
- [48] Craig T J, Lorenz R D. Experimental identification of friction and its compensation in precise, position controlled mechanisms [J]. IEEE Trans. on Industry Applications, 1992, 28 (6): 1392 – 1398.
- [49] Lee S W, Kim J H. Friction identification using evolution strategies and robust control of positioning tables [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1999, 121: 619 – 624.
- [50] Yen J Y, Huang S J, Lu S S. A new compensation for servo systems with position dependent friction [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1999, 121: 612 – 618.
- [51] Walrath C D. Adaptive bearing friction compensation based on recent knowledge of dynamic friction [J]. Automatica, 1984, 20 (6): 717 – 727.
- [52] Canudas De Wit C, Lischinsky P. Adaptive friction compensation with partially known dynamic friction model [J]. International Journal of Adaptive Control and Signal Processing, 1997, 11: 65 – 80.
- [53] Lischinsky P, Canudas de Wit C, Morel G. Friction compensation for an industrial hydraulic Robot [J]. IEEE Control Systems. 1999, 2: 25 – 232.
- [54] Hirschorn R M, Miller G. Control of nonlinear systems with friction [J]. IEEE Trans. on Control Systems Technology, 1999, 7 (5): 588 – 595.
- [55] Vedagarbha P, Dawson D M, Feemster M. Tracking control of mechanical systems in the presence of nonlinear dynamic friction effects [J]. IEEE Trans. on Control Systems Technology, 1999, 7 (4): 446 – 456.
- [56] Misovec KM, Annaswamy A M. Friction compensation using adaptive nonlinear control with persistent excita-

- tion [J]. International Journal of Control, 1999, 72 (5): 457 – 479.
- [57] Canudas de Wit C, Ge S S. Adaptive friction compensation for systems with generalized velocity/ position friction dependency [C]. In Proceedings of the 36th Conference on Decision & Control, 1997: 2465 – 2470.
- [58] Canudas de Wit C, Horowitz R. Observers for tire/ road contact Friction using only wheel angular velocity information [C]. In Proceedings of the 38th Conference on Decision & Control, 1999: 3932 – 3937.
- [59] Canudas de Wit C. Control design for ultrasonic motors with dynamic friction interface [C]. In the 14th World Congress of IFAC, Beijing, 1999: 487 – 491.
- [60] Dupont P E. Avoiding stick – slip through PD control [J]. IEEE Trans. on Automatic Control, 1994, 39 (5): 1094 – 1097.
- [61] Armstrong B, Neevel D, Kusik T. New result in NPID control: tracking, integral control, friction compensation and experimental result [C]. In Proceedings of the 1999 IEEE International Conference on Robotics & Automation, 1999: 837 – 842.
- [62] Lee S, Meerkov S M. Generalized dither [J]. IEEE Trans. on Information Theory, 1991, 37 (1): 50 – 56.
- [63] Horowitz L, Oidak S, Shapiro A. Extensions of dithered feedback systems [J]. International Journal of Control, 1991, 54 (1): 83 – 109.
- [64] Yang S, Tomizuka M. Adaptive pulse width control for precise positioning under influence of sticktion and coulomb Friction [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1988, 110 (3): 221 – 227.
- [65] Armstrong B. Control of machines with friction [M]. Norwell, Kluwer Academic Publishers, 1991.
- [66] Luh J Y S, Fisher WD, Paul R P C. Joint torque control by a direct feedback for industrial robots [J]. IEEE Trans. on Automatic Control, 1983, 28 (2): 153 – 161.
- [67] Morel G, Iagnemma K, Dubowsky S. The precise control of manipulators with high joint – friction using base force/ torque sensing [J]. Automatica, 2000, 36: 931 – 941.
- [68] Tesfaye A, Lee H S, Tomizuka M. A sensitivity optimization approach to design of an observer in digital motion control systems [J]. IEEE/ASME Trans. on Mechatronics, 2000, 5 (1): 322 – 38.
- [69] White M T, Tomizuka M, Smith C. Improved track following in magnetic disk drives using a disturbance observer [J]. IEEE/ ASME Transactions on Mechatronics, 2000, 5 (1): 3 – 11.
- [70] Ro P I, Shim W, Jeong S. Robust friction compensation for submicrometer positioning and tracking for a ball – screw – driven slide system [J]. Precision Engineering, 2000, 24: 160 – 173.
- [71] Lee H S, Tomizuka M. Robust motion controller design for high – accuracy positioning systems [J]. IEEE Trans. on Control Systems Technology, 1996, 43 (1): 48 – 55.
- [72] Taghirad H D, Belanger P R. Robust friction compensation for harmonic drive transmission [C]. In Proceedings of the 1998 IEEE International Conference on Control Applications, 1998: 547 – 551.
- [73] Seungrohk O, Hassan K K. Nonlinear output – feedback tracking using high – gain observer and variable structure control [J]. Automatica, 1999, 33 (10): 1845 – 1856.
- [74] 陈义武. 离散滑动模态控制方法在转台中的应用 [D]. 北京: 北京航空航天大学, 1998.
- [75] Southward S C, Radcliffe CJ, MacCluer C R. Robust nonlinear stick – slip friction compensation [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1991, 113: 639 – 645.
- [76] Kang M S. Robust digital friction compensation [J]. Control Engineering Practice, 1998, 6 (3):

359 – 367.

- [77] Young KD. A variable structure control approach to friction force compensation [C]. In Proceedings of the American Control Conference, 1998: 3138 – 3142.
- [78] Sivakumar S, Farshad K. Model independent friction compensation [C]. In Proceedings of the American Control Conference, 1998: 463 – 466.
- [79] Korondi P, Young D, Hashimoto H. Sliding mode based disturbance observer for motion control [C]. In Proceedings of the 37th IEEE Conference on Decision & Control, 1998: 1926 – 1928.
- [80] Opat G. Compensation and estimation of friction by using extended Kalman filter, SICE – ICASE International Joint Conference, 2006, 5032 – 5035.
- [81] Hara S, Yamamoto Y, Omata T, Nakano M. Repetitive control system: a new type servo system for periodic exogenous signals [J]. IEEE Trans. on Automatic Control, 1988, 33 (7): 659 – 668.
- [82] Tung E D, Anwar G, Tomizuka M. Low velocity friction compensation and feedforward solution based on repetitive control [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1993, 115: 279 – 284.
- [83] Cho S I, Ha I J. A learning approach to tracking in mechanical systems with friction [J]. IEEE Trans. on Automatic Control, 2000, 45 (1): 111 – 116.
- [84] 陈永生. 重复控制方法在伺服系统中的应用研究 [D]. 北京: 北京航空航天大学, 1999.
- [85] 贾亮. 具有重复控制的高精度系统跟踪控制系统的研究和应用 [D]. 北京: 北京航空航天大学, 1997.
- [86] 胡恒章, 冯幼田, 邓志东. 自学习控制在转台控制中的应用 [J]. 宇航学报, 1992 (4): 73277.
- [87] Selmic R, Lewis F L. Neural network approximation of piecewise continuous functions: application to friction compensation [C]. In Proceedings of the 12th IEEE International Symposium on Intelligent Control, 1997: 227 – 232.
- [88] Rastko R S, Lewis F L. Deadzone compensation in motion control systems using neural networks [J]. IEEE Trans. on Automatic Control, 2000, 45 (4): 602 – 613.
- [89] Larsen GA, Cetinkunt S, Donmez A. CMAC neural network control for high precision motion control in the presence of large friction [J]. ASME Journal of Dynamic Systems, Measurement, and Control, 1995, 117: 415 – 420.
- [90] Lee S I, Kim J H. CMAC network – based robust controller for systems with friction [C]. In Proceedings of the 34th Conference on Decision & Control, 1995: 2938 – 2939.
- [91] 黄进, 叶尚辉. 基于 CMAC 网络的摩擦补偿研究 [J]. 中国机械工程, 1999, 10 (3): 269 – 272.
- [92] Du H L, Nair S S. Modeling and compensation of low – velocity friction with bounds [J]. IEEE Trans. on Control Systems Technology, 1999, 7 (1): 110 – 121.
- [93] Kim Y H, Lewis F L. Reinforcement adaptive learning neural network based friction compensation for high speed and precision [C]. In Proceedings of the 37th IEEE Conference on Decision & Control, 1998: 1064 – 1069.
- [94] Yongfu Wang, Dianhui Wang, Tianyou Chai. Active control of friction self – excited vibration using neuro – fuzzy and data mining techniques [J]. Expert Systems with Applications, 2013, 40 (4): 975 – 983.
- [95] Sidi E YO, Sicard P, Massicotte D, et al. Adaptive high precision position control for a flexible joint with friction and parameter uncertainties using neural networks [J]. In IEEE on Electrical and Computer Engi-

- neering, 1998: 325 – 328.
- [96] Seidl D R, Lam S L, Putman J A, Lorenz R D. Neural network compensation of gear backlash hysteresis in position – controlled mechanisms [J]. IEEE Trans. on Industry Applications, 1995, 31 (6): 1475 – 1483.
- [97] Bose B M. Expert system, fuzzy Logic, and neural network applications in power electronics and motion control [J]. In Proceedings of IEEE, 1994, 82 (8): 1303 – 1323.
- [98] Teeter J T, Chow M Y, Jame J. A novel fuzzy friction compensation approach to improve the performance of a DC motor control system [J]. IEEE Trans. on Industrial Electronics, 1996, 43 (1): 113 – 120.
- [99] Tzes A, Peng P Y, Guthy J. Genetic – based fuzzy clustering for DC – Motor friction identification and compensation [J]. IEEE Trans. on Control Systems Technology, 1998, 6 (1): 462 – 472.
- [100] 黄进, 叶尚辉, 陈其昌. 基于自调整量化因子模糊控制器的摩擦补偿研究 [J]. 机械设计与研究, 1999 (1): 27 – 29.
- [101] Kovacic Z, Bogdan S, Balenovic M. A model reference & sensitivity model – based self – learning fuzzy Logic controller as a solution for control of nonlinear servo systems [J]. IEEE Trans. on Energy Conversion, 1999, 14 (4): 1479 – 1484.
- [102] Yongfu Wang, Dianhui Wang, Tianyou Chai. State estimate – based robust adaptive control of a linked mass – spring system with nonlinear friction [J]. ASME Journal of Dynamic Systems, Measurement and Control, 2013, 135 (2): 251 – 259.
- [103] Yongfu Wang, Dianhui Wang, Tianyou Chai. Extraction and adaptation of fuzzy rules for friction modeling and control compensation [J]. IEEE Trans. On Fuzzy Systems, 2011, 180 (2): 5029 – 5040.
- [104] Yongfu Wang, Dianhui Wang, Tianyou Chai. Modeling and control compensation of nonlinear friction using adaptive fuzzy systems [J]. Mechanical System and Signal Processing, 2009, 23 (8): 2445 – 2457.
- [105] 王永富, 张义民, 柴天佑. 一类不确定机械系统的双调节自适应模糊建模与鲁棒控制研究 [J]. 机械工程学报, 2013, 49 (9): 61 – 68.
- [106] LIN C L, LEE G Y. Hierarchical fuzzy control for C – aAxis of CNC turning centers using genetic algorithms, Journal of Intelligent and Robotic Systems, 1999, 25 (3): 255 – 275.
- [107] Lib C L, LIN J Y. Fuzzy – enhanced adaptive control for flexible drive system with friction using genetic algorithms, 1998, 23 (2 – 4): 379 – 405.
- [108] Liaw C D, Huang T J. Contact friction compensation for robots using genetic learning algorithms, Journal of Intelligent and Robotic Systems, 1998, 23 (2 – 4): 331 – 349.

第2章 非线性系统的模糊建模方法

2.1 常用的建模方法

建模方法主要有机理建模、辨识建模和混合建模三种。采用的建模技术有基于过程机理分析、回归分析、人工神经网络、案例推理、状态估计、模式识别、模糊数学、过程层析成像、相关分析和现代非线性信息处理等。相对而言,前几种建模技术的研究较为深入,在过程控制和检测中已有许多成功的应用,后三种建模技术限于技术发展水平,在过程控制中目前还应用较少^[1-3]。

1. 基于工艺机理分析的建模

基于工艺机理分析的建模主要运用化学反应动力学、物料平衡、能量平衡等原理,通过对生产过程的机理分析,建立不可测主导变量与可测辅助变量之间的机理模型。

基于工艺机理的建模方法是工程中常用的方法,其特点是简单,工程背景清晰,便于实际应用,但应用效果依赖于对工艺机理了解程度,因为这种建模方法是建立在对工艺过程机理深刻认识的基础上,建模的难度较大。对于工艺机理较为清楚的工艺过程,该方法能建立出性能较好的模型。但对于机理研究不充分、尚不完全清楚的复杂工业过程,难以建立合适的机理模型。此时该方法就需要与其他方法相结合建立混合模型。

2. 基于回归分析的建模

回归分析方法是一种经典的建模方法,不需建立复杂的数学模型,只要收集大量过程参数和质量分析数据,运用统计方法将这些数据中隐含的对象信息进行浓缩和提取,从而建立主导变量和辅助变量之间的数学模型。

根据采用的数学方法的不同,可以将回归分析方法分为线性回归和非线性回归。线性回归方法的实质是对象函数关系在操作点附近的一阶泰勒展开式,算法简单,物理意义明确,但对测量误差比较敏感。当对象的非线性特性比较显著或操作点变动范围较大时,泰勒展开式中的高阶项就不能忽略了,此时用线性回归方法建立的模型精度就会明显下降,应考虑选用非线性回归方法。线性回归中最经典的方法是最小二乘法,在此基础上又发展了主元回归法(Principal Component Regression, PCR)和部分最小二乘法(Partial Least Squares Regression, PLSR),特别适合辅助变量多且变量间相关性强的系统。从应用情况看,对于线性系统,采用PCR和PLSR的效果差不多,对于非线性系统则采用PLSR的效果较好。基于回归分析的建模,其特点是简单实用,但需要大量的样本数据,对测量误差较为敏感。

尽管用机理分析确定模型结构、用历史测量数据估计模型参数的混合方法得到了广泛的应用,但回归分析方法仍存在几个缺陷难以解决:模型准确性受样本真实性的影响,适用范

围受样本容量的制约。因此,只有当长期的生产过程可以提供大量正确的样本时,该方法才能得到正确可靠的实施。

3. 基于状态估计的建模

该方法将状态估计、参数估计和自适应控制原理相结合用于建立生产过程关键参数模型。设计方法可分为基于状态空间模型和基于过程输入输出模型两种。

(1) 基于状态估计的方法

假定已知对象的状态空间模型为

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u + \mathbf{E}v \\ y = \mathbf{C}\mathbf{x} \\ \theta = \mathbf{C}_\theta\mathbf{x} + w \end{cases} \quad (2-1)$$

式中 $\mathbf{x}$ ——过程的状态变量;

v, w ——分别表示白噪声;

y, θ ——分别表示过程的主导变量和辅助变量。

Kalman 滤波器的解为

$$\tilde{y} = \mathbf{C} (\mathbf{I} - \mathbf{A} + k_f \mathbf{C}_\theta)^{-1} (k_f \theta + \mathbf{B}u) \quad (2-2)$$

式中 y 和 θ ——分别表示主导变量和辅助变量;

k_f ——Kalman 滤波器的增益。

$$k_f = \mathbf{X} \mathbf{C}_\theta^T w^{-1}$$

基于状态估计的建模就是在建立以上生产过程的动态数学模型(包括广义动态数学模型)时,将不可测主导变量 y 看做状态变量,把可测的辅助变量 θ 看做输出变量,这样对主导变量的建模就转化为控制理论中典型的状态观测或状态估计命题,进而可采用 Luenberger 观测器和 Kalman 滤波器或自适应 Kalman 滤波器(AKF)、扩展 Kalman 滤波器(EKF)方法得到状态的估计值。Kalman 滤波器虽然是处理含噪声过程数据的有效工具,但它需要事先知道过程噪声和测量噪声的协方差矩阵,这对于许多实际工业过程来说难以实现。而且采用这种方法,基于系统工作在操作点附近时可用线性模型近似表示的假设,对于非线性严重的对象会产生很大的估计偏差,甚至出现算法发散的现象。

(2) 基于输入输出估计

假定已知对象输入输出关系如下

$$\begin{cases} y = G_u u + G_d d \\ \theta = F_u u + F_d d \end{cases} \quad (2-3)$$

通过 Brosilow 估计器来构造

$$\tilde{y} = K_B \theta + (G_u - K_B F_u) u \quad (2-4)$$

式中 $K_B = G_d F_d^*$, F_d^* , F_d 伪逆。

当可测扰动 d 为零时,该问题简化为投影估计器。对于动态模型,可以转化为基于 AR-MAX 模型的递推估计问题进行求解或采用自适应输入输出估计方法在线辨识过程参数。Ronald 介绍了非线性输入输出模型的 Hammerstein 和 Wiener 结构以及 NARMAX 的辨识过程。

这种方法同样需要一个准确的对象模型，而我们得到的控制模型通常是简化的数学模型，过程噪声与理想自噪声相差甚远，因此这种方法在工程实际中的应用实例并不多。

4. 基于神经网络的建模

人工神经网络（ANN，Artificial Neural Networks）是利用计算机模拟人脑的结构和功能的一门新学科，它具备优良的信息处理特性：应用 ANN 无需具备对象的先验知识，而根据对象的输入、输出数据直接建模；独特的非传统的表达方式和固有的学习能力，使之在解决高度非线性和严重不确定性系统控制方面具有巨大的潜力。目前，ANN 已成功地用于复杂工业过程的动态建模、系统辨识和控制、数据分析、故障诊断等方面，显现出强大的生命力。ANN 是建立模型的一种十分便利和有效的方法，将过程易测量的辅助变量作为神经网络的输入，将建模对象作为 ANN 的输出，通过网络的学习能力完成对象的估计。

基于人工神经网络（Artificial Neural Network，简记 ANN）的建模是近年来研究最多、发展很快和应用范围非常广泛的一种建模技术。由于人工神经网络建模可在不具备对象先验知识的条件下，根据对象的输入输出数据直接建模（将辅助变量作为人工神经网络的输入，而主导变量则作为网络的输出，通过网络的学习来解决不可测变量的建模问题），模型的在线校正能力强，并能实用于高度非线性和严重不确定性系统，因此它为解决复杂系统过程参数的建模问题提供了一条有效途径。

采用人工神经网络进行建模有两种形式：一种是利用人工神经网络直接建模，用网络来代替常规数学模型描述辅助变量和主导变量间的关系，完成由可测信息空间到主导变量的映射，如图 2-1a 所示；另一种是与常规模型（如机理模型等）相结合，用人工神经网络来估计常规模型的模型参数，进而实现建模，如图 2-1b 所示。

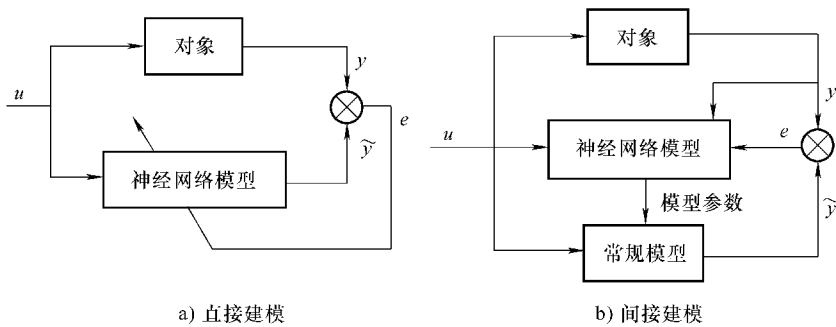


图 2-1 基于神经网络的建模

在建模中常用的 ANN 结构有反向传播网络（BPN，Backpropagation Neural Networks）、径向基函数神经网络（RBFN，Radial Basis Function Neural Networks）、回流网络（RNN，Recurrent Neural Networks）等。但 ANN 并不是完美无缺的，在结构和算法方面还存在许多问题，如 ANN 拓扑结构只能凭经验加试算进行选择，训练样本的空间分布和训练方法对 ANN 的性能有极大的影响，ANN 学习和预测算法有待改进，ANN 训练数据集的正交性和完备性问题；ANN 的知识推理等问题。在实际应用中，网络学习训练样本的数量和质量、学习算

法、网络的拓扑结构和类型等的选择对所构成的建模的性能都有重大影响。

5. 基于模糊数学的建模

模糊数学模仿人脑逻辑思维特点，是处理复杂系统的一种有效手段，在过程建模中也得到了大量应用。基于模糊数学的建模所建立的模型是一种知识性模型。该种建模方法特别适合应用于复杂工业过程中被测对象呈现亦此亦彼的不确定性，难以用常规数学定量描述的场合。实际应用中常将模糊技术和其他人工智能技术相结合，例如模糊数学和人工神经网络相结合构成模糊神经网络，将模糊数学和模式识别相结合构成模糊模式识别，这样可互相取长补短以提高建模的效能。

6. 基于案例推理的建模

基于案例推理 (Case Based Reasoning, CBR) 的核心思想是在问题求解时，可以借助于以往对该类问题的求解经验，即案例，来进行推理。基于案例的推理与基于规则的推理 (RBR) 方法不同，它是一种类比推理方法，其推理过程中蕴含了学习和归纳的功能，这使得知识获取更为容易，因而在诊断、决策、分析等依赖经验知识较多的领域都有强大的应用背景。传统的人工智能方法或者单独依赖于问题领域中的一般性知识，或是在问题描述与结论之间建立一般性的联系，案例推理与其不同之处在于，案例推理能利用过去经验中的特定知识即具体案例来解决新问题，它通过寻找与之相似的历史案例，把它重新应用到新问题的环境中来；另一点明显的区别在于，案例推理的实现是一个持续的、不断改善的学习过程，一旦解决了一个新问题，就获得了新的经验，可以用来解决将来的问题。

案例推理周期的示意图如图 2-2 所示，其中案例表示，案例检索和案例调整是案例推理研究的核心问题。在案例推理中，一个新问题的求解是通过在案例库中检索一个或多个相关案例，应用匹配案例提供的解答并检验其应用结果是否成功，否则这个解答方案将被重新修正（除非被检索的案例很匹配），以获得一个能保存的成功案例。并没有普遍性的案例推理方法可以适用于解决所有领域中的问题，其难点就在于对于不同的领域和应用环境一般需要不同的方法。一般认为案例推理研究主要包括下面五个方面的问题：

1) 案例表示：以一定的结构在案例中存储有关的信息。

2) 案例检索：在案例库中搜索与所给问题相似的案例。

3) 案例重用：重用检索到的案例的信息与知识以解决该问题。

4) 案例修正：在重用的解失败或不满意时对其进行修改。

5) 案例保存：将此解决方案中可能用于将来问题求解的部分存入库中。

其中案例检索、案例重用、案例修正和案例保存构成了一个案例推理周期。

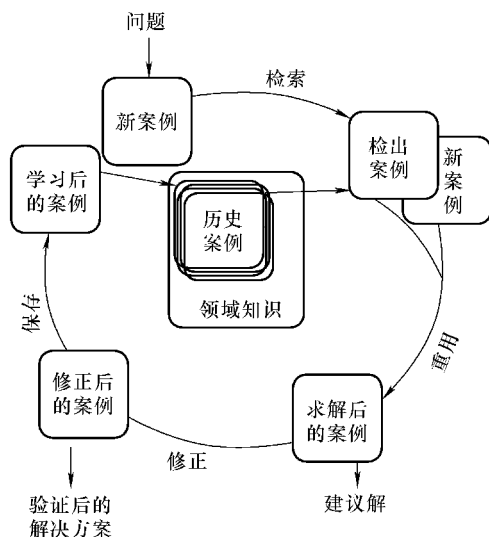


图 2-2 案例推理周期示意图

CBR 作为一种在缺乏系统模型而具有丰富经验场合下的问题求解方法, 在系统建模与过程控制领域崭露头角。

7. 基于模式识别的建模

这种建模方法是采用模式识别的方法对工业过程的操作数据进行处理, 从中提取系统的特征, 构成以模式描述分类为基础的模式识别模型。

基于模式识别方法建立的模型与传统的数学模型不同, 它是一种以系统的输入、输出数据为基础, 通过对系统特征提取而构成的模式描述式模型。该方法的优势在于它适用于缺乏系统先验知识的场合, 可利用日常操作数据来实现建模。在实际应用中, 该种建模方法常常和人工神经网络以及模糊技术结合在一起。

8. 基于过程层析成像的建模

基于过程层析成像 (Process Tomography, PT) 的建模与其他建模技术不同的是, 它是一种以医学层析成像 (Computerized Tomography, CT) 技术为基础的, 可在线获取过程参数二维或三维的实时分布信息的先进检测技术, 即一般软建模技术所获取的大多是关于某一变量的宏观信息, 而采用该技术可获取关于该变量微观的时空分布信息。

国内外对过程层析成像的研究始于 20 世纪 80 年代中后期, 目前在解决两相流和多相流系统参数 (例如气液或气固两相管流的流型判别、分相流量的测量, 流化床反应器的空隙率及其分布的检测等) 测量这一检测难题上已取得了不少进展, 是现代过程检测技术领域中的一个重要的研究方向。由于技术发展水平的制约, 该种建模技术目前离工业实用化还有一定距离, 在过程控制中的直接应用还不多。

9. 基于相关分析的建模

基于相关分析的建模技术是以随机过程中的相关分析理论为基础, 利用两个或多个可测随机信号间的相关特性来实现某一参数的在线测量。

该种建模方法采用的具体实现方法大多是互相关分析方法, 即利用各辅助变量 (随机信号) 间的互相关函数特性来进行建模。目前, 这种方法主要应用于难测流体 (即采用常规测量仪表难以进行有效测量的流体) 流速或流量的在线测量和故障诊断 (例如, 流体输送管道泄漏的检测和定位) 等。

10. 基于现代非线性信息处理技术的建模

基于现代非线性处理技术的建模是利用易测过程信息 (辅助变量, 它通常是一种随机信号), 采用先进的信息处理技术, 通过对所获信息的分析处理提取信号特征量, 从而实现某一参数的在线检测或过程的状态识别。

这种建模技术的基本思想与基于相关分析的建模技术一致, 都是通过信号处理来解决建模问题, 所不同的是具体信息处理方法不同。该种建模技术的信息处理方法大多是各种先进的非线性信息处理技术, 例如小波分析、混沌和分形技术等, 因此能适用于常规的信号处理手段难以适应的复杂工业系统。相对而言, 基于现代非线性信息处理技术的建模的发展较晚, 研究也还比较分散。该种建模技术目前一般主要应用于系统的故障诊断、状态检测和过失误差侦破等, 并常常和人工神经网络或模糊数学等人工智能技术相结合。

在建模过程中, 通常需要从实际生产过程中采集建模样本数据。一般来说用于建模的数

据分为训练用数据（学习样本）和测试用数据（测试样本），建模过程是指用训练数据不断训练模型结构和参数，使之能更好地拟合训练数据的过程。在建模完成后，用训练数据测试所建的模型的过程称为回想（recall），其误差称为拟合误差；测试样本不参与学习，只用于模型的测试，用测试样本检验模型的过程称为泛化（generalization），其误差称为泛化误差。为判断建模效果的优劣，我们引入计算校验数据集的泛化均方根误差（RMSE）和最大泛化绝对误差（MAXE）。即

$$\begin{cases} \text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\rho_i - \bar{\rho}_i)^2} \\ \text{MAXE} = \max_{i=1}^N (|\rho_i - \bar{\rho}_i|) \end{cases} \quad (2-5)$$

式中 ρ_i ——组分含量的化验室测量值；

$\bar{\rho}_i$ ——组分含量的建模估计值；

N ——校验数据的个数。

针对具体工业生产过程，根据工艺要求给定建模的泛化均方根误差（RMSE）和最大泛化绝对误差（MAXE）指标。如建立的模型校验指标小于给定指标要求，表明所建立的模型具有较高精度和较强的泛化能力。

上述方法是在建模领域中的主流方法，当然也可采用几种方法互相结合进行建模。上述方法涉及众多领域，无法详细展开，下面主要就模糊建模进行论述。

2.2 模糊建模的基本理论基础

“模糊”是与“精确”相对应的一个概念，模糊性普遍存在于人类思维和语言交流中，并且人类对于外部事物的认识的本质就是模糊性。在人类科学的发展史上，以追求事物的精确性描述为目的的研究已产生了大量的结果，并建立了庞大的数学和逻辑学体系。但是随着信息化的发展，人们不可避免地要处理越来越多的模糊现象，而传统的数学方法却无法很好地处理这些模糊信息，这就导致了精确性和模糊性的对立。精确性与模糊性的对立，目前已成为当今科学发展所面临的一个十分突出的矛盾，一方面各门学科迫切要求数学化、定量化，但是，另一方面科学的深化意味着研究对象的复杂化，复杂的东西又难于精确化。

1965年 L. A. Zadeh 教授发表了一篇名为“Fuzzy Sets”的开创性论文。首次提出了模糊集合的概念^[4]。模糊集合论是对传统集合论的一种推广，对于模糊集合来说，每一个元素都是以一定的程度隶属于某个集合的，也可以同时以不同的程度隶属于几个集合。1973年，L. A. Zadeh 教授又进一步研究了模糊语言处理，建立了模糊推理的理论基础。1974年，英国科学家 Mamdani 将基于模糊推理规则的模糊系统应用到蒸汽发电机的压力和速度控制中，并取得了成功^[5]。从此之后模糊系统在理论和应用中都得到了较快的发展，在几十年的时间里取得了丰富的成果，尤其是在自动控制、模式识别、聚类分析、故障诊断和专家系统等领域的应用取得了巨大成功^[6]。

2.2.1 模糊集合及其表示法

19 世纪末德国数学家 George Contor 创立的集合论已经成为现代数学的基础, 每个数学分支都可以看作研究某类对象的集合, 因此集合的理论统一了许多似乎没有联系的概念。

定义 1 (集合): 集合一般是指具有某种属性的、确定的、彼此间可以区别的事物的全体。将组成集合的事物称为集合的元素。通常用大写字母 A, B, C 等表示集合, 而用小写字母 a, b, c 表示集合内的元素。元素和集合之间的关系是属于或不属于的关系, 当元素 a 属于集合 A 时, 用 $a \in A$ 表示; 反之用 $a \notin A$ 来表示。

经典集合中元素与集合的关系可以通过特征函数表示法来刻画, 如某一集合 A 的特征函数定义如下

$$\chi_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases} \quad (2-6)$$

从中可以看出, 一个元素和集合的关系只有两种, 即或者是属于 A , 或者是不属于 A 。

用上述经典集合的定义所表达的概念内涵和外延都是明确的, 但在人们的思维和语言表达中, 有许多没有明确外延的概念, 即模糊概念, 如“高个子”、“青年”等。模糊概念无法用经典集合加以描述。为解决这一问题, Zadeh 提出了模糊集合 (Fuzzy Sets) 的定义。

定义 2 (模糊集合): 论域 U 上的一个模糊集合 A 是指对于论域 U 中的任一元素 $u \in U$, 都指定了 $[0, 1]$ 闭区间的一个数 $\mu_A(u) \in [0, 1]$ 与之对应, 它叫做 u 对 A 的隶属度。这意味着定义了一个映射 μ_A

$$\begin{aligned} \mu_A: U &\rightarrow [0, 1] \\ u &\rightarrow \mu_A(u) \end{aligned} \quad (2-7)$$

这个映射称为模糊集合 A 的隶属函数。

模糊集合的表达方式有以下几种。

1. U 为有限集

当 U 为有限集 $\{u_1, u_2, \dots, u_n\}$ 时, 通常有以下 3 种方式

(1) Zadeh 表示法

$$A = \frac{A(u_1)}{u_1} + \frac{A(u_2)}{u_2} + \dots + \frac{A(u_n)}{u_n} \quad (2-8)$$

在上式中, $\frac{A(u_i)}{u_i}$ 并不表示分数, 而是表示论域中的元素 u_i 与其隶属度 $A(u_i)$ 之间的对应关系; “+” 也不表示求和, 而是表示模糊集合在论域 U 的整体。

例 2.1 在由整数 1, 2, $\dots$, 5 组成的论域中, 既 $U = \{1, 2, 3, 4, 5\}$, 讨论“非常小”这一模糊概念。根据经验, 可以定量地给出他们的隶属函数, 模糊集合“非常小”可表示为

$$A = \frac{1}{1} + \frac{0.64}{2} + \frac{0.36}{3} + \frac{0.16}{4} + \frac{0.04}{5} \quad (2-9)$$

上式表明: “1” 的隶属度为 1, 说明 “1” 为 “非常小” 的可能性最大; “5” 的隶属度最

小,说明“5”为“非常小”的可能性也最小。

(2) 向量表示法

$$A = (A(u_1), A(u_2), \dots, A(u_n)) \quad (2-10)$$

采用向量表示法,上述例子 2.1 中的 A 可表示为

$$A = (1, 0.64, 0.36, 0.16, 0.04) \quad (2-11)$$

在向量表示法中,隶属度为 0 的项不能省略。有时也将上述 2 种方法结合起来表示为

$$A = \left(\frac{A(u_1)}{u_1}, \frac{A(u_2)}{u_2}, \dots, \frac{A(u_n)}{u_n} \right) \quad (2-12)$$

则上述例子 2.1 可表示为 $A = \left(\frac{1}{1}, \frac{0.64}{2}, \frac{0.36}{3}, \frac{0.16}{4}, \frac{0.04}{5} \right)$ 。向量表示法对于模糊集合的运算十分方便。

(3) 序偶表示法

将论域中的元素 u_i 与其隶属度 $A(u_i)$ 构成序偶来表示 A , 则

$$A = \{(u_1, A(u_1)), (u_2, A(u_2)), \dots, (u_n, A(u_n))\} \quad (2-13)$$

采用序偶表示法,例子 2.1 中的 A 可表示为

$$A = \{(1, 1), (2, 0.64), (3, 0.35), (4, 0.16), (5, 0.04)\} \quad (2-14)$$

此种方法隶属度为 0 的项可不写入。

2. U 为有限连续域

当 U 为有限连续域时, Zadeh 给出如下记法

$$A = \int_U \frac{\mu_A(u)}{u} \quad (2-15)$$

同样, $\frac{\mu_A(u)}{u}$ 并不表示“分数”,而表示论域上的元素 u 与其隶属度 $A(u)$ 之间的对应关系;

$\int$ 既不表示积分也不表示求和记号,而是表示论域 U 上的元素 u 与其隶属度 $A(u)$ 对应关系的一个总括。

例 2.2 以年龄作论域,取 $U = [1, 100]$, Zadeh 给出的“年老” Q 与“年轻” Y 两个模糊集合的隶属函数为

$$\mu_Q(u) = \begin{cases} 0 & 0 \leq u \leq 50 \\ \left[1 + \left(\frac{u-50}{5} \right)^{-2} \right]^{-1} & 50 < u \leq 100 \end{cases} \quad (2-16)$$

$$\mu_Y(u) = \begin{cases} 1 & 0 \leq u \leq 25 \\ \left[1 + \left(\frac{u-25}{5} \right)^2 \right]^{-1} & 25 < u \leq 100 \end{cases} \quad (2-17)$$

采用 Zadeh 表示法,“年老” Q 与“年轻” Y 两个模糊集合可写为

$$Q = \int_{0 \leq u \leq 50} \frac{0}{u} + \int_{50 < u \leq 100} \frac{\left[1 + \left(\frac{u-50}{5} \right)^{-2} \right]^{-1}}{u} = \int_{0 < u \leq 100} \frac{\left[1 + \left(\frac{u-50}{5} \right)^{-2} \right]^{-1}}{u} \quad (2-18)$$

$$Y = \int_{0 \leq u \leq 25} \frac{1}{u} + \int_{25 < u \leq 100} \frac{[1 + (\frac{u-25}{5})^2]^{-1}}{u} \quad (2-19)$$

2.2.2 隶属函数

1965年, Zadeh 在他的论文《模糊集合论》中首次提出隶属函数这个概念, 并利用这个概念来描述客观事物的不分明性, 即模糊性, 成功地运用了数学方法来描述模糊概念。因此, 正确地确定隶属函数, 是运用模糊集合理论解决实际问题的基础。隶属函数的确定过程, 本质上说应该是客观的, 但每个人对于同一个模糊概念的认识理解又有差异, 因此, 隶属函数的确定又带有主观性。一般是根据经验或统计进行确定, 也可由专家、权威人士给出。对于同一个模糊概念, 确定隶属函数的方法是多种多样的, 没有统一的模式。但是, 尽管形式不完全相同, 只要能反映同一模糊概念, 在解决和处理实际模糊问题中仍然殊途同归。事实上, 它是依赖于主观来描述客观事物的概念外延的模糊性。

这里介绍几种常用的隶属函数的确定方法, 不同的方法结果也会不同, 判别所确定的隶属函数的好与不好的标准看它是否符合实际要求。

1. 模糊统计法

在有些情况下, 隶属函数可以通过模糊统计试验的方法来确定, 这是确定隶属函数的主要方法。其思想是通过对足够多人的调查统计, 对要确定的模糊概念在讨论的论域中逐一写出定量范围, 在进行统计处理, 以确定能被大多数人认可的隶属函数。这实际上是对人群主观看法的统计处理。已有不少国内外学者进行了这方面的研究, 确定在一定条件下随着调查人数的增加, 各个元素隶属度都趋于一个稳定值。

2. 例证法

例证法是 Zadeh 在 1972 年提出的, 主要思想是从已知有限个 μ_A 的值, 来估计论域 U 上模糊集合 A 的隶属函数。例如论域 U 是全体人类, A 是“高个子的人”, 显然 A 是模糊集合。为了确定 μ_A , 可先给出一个高度 h 值, 然后选定几个语言真值 (即一句话真的程度) 中的一个, 来回答某人高度是否算“高”。如语言真值分为“真的”、“大致真的”、“似真又似假”、“大致假的”和“假的”。然后, 把这些语言真值分别用数字表示为 1, 0.75, 0.5, 0.25 和 0。将 n 个不同高度 $h_1, h_2, \dots, h_n$ 作为样本进行询问, 就可以得到 A 的隶属函数 μ_A 的离散表示法。

3. 专家经验法

根据专家的实际经验, 确定隶属函数的方法称为专家经验法。即专家的实际经验和必要的数学处理相结合来确定隶属函数。例如在各种体育比赛中, 裁判员的评分就是他们大量的丰富的实际经验的综合结果。在许多情况下, 经常是初步确定粗略的隶属函数, 然后再通过学习和实践检验逐步修改和完善, 而实际效果正是检验和调整隶属函数的依据。

隶属函数形式有多种, 可根据实际要求来确定。在实际应用中为方便起见, 常采用三角形、正态形和梯形。以下是这三种隶属函数的解析式。

(1) 三角形隶属函数

解析式为

$$\mu_F(x) = \begin{cases} \frac{x-b}{a-b} & b \leq x \leq a \\ \frac{c-x}{c-a} & a < x \leq c \\ 0 & x < b \text{ 或 } x > c \end{cases} \quad (2-20)$$

(2) 正态形隶属函数

解析式为

$$\mu_F(x) = e^{-\left(\frac{x-a}{b}\right)^2} \quad b > 0 \quad (2-21)$$

(3) 梯形函数

解析式为

$$\mu_F(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x < b \\ 1 & b \leq x \leq c \\ \frac{d-x}{d-c} & c < x \leq d \\ 0 & x > d \end{cases} \quad (2-22)$$

2.2.3 模糊规则和模糊推理

1. 语言变量

在日常生活中, 变量常用词语来描述, 例如我们说起一个人的年纪时总是说“很年轻”或“很老”等。但是当—个变量取语言值时, 经典数学理论却无法对其进行描述, 因此 Zadeh 提出了语言变量的概念以提供正式的数学体系, 使自然语言的模糊描述成为精确的数学描述。

语言变量用一个有五个元素的集合 $(N, T(N), U, G, M)$ 来表征, 其中 N 是语言变量的名称, 如年龄、数的大小等; $T(N)$ 为语言变量 N 的语言术语集合; U 为语言变量 N 的论域; G 是产生 $T(N)$ 中术语的句法规则; M 是赋给任一语言值 $A \in T(N)$ 含义的语法规则, $M(A)$ 表示 U 上的一个模糊子集。

2. 模糊 If – Then 规则

模糊 If – Then 规则, 也称为模糊规则或模糊条件语句, 其表示形式为

$$\text{If } x \text{ 是 } A \text{ Then } y \text{ 是 } B \quad (2-23)$$

式中 A 和 B ——分别是论域 x 和 y 上的模糊集合定义的语言值。

一般将“ x 是 A ”称作规则前件, 将“ y 是 B ”称作规则后件或结论。

模糊 If – Then 规则可以缩写成 $(A \rightarrow B)$ 形式, 在使用模糊规则对系统进行建模前必须将表达式 $(A \rightarrow B)$ 的意义形式化。模糊规则描述了语言变量 x, y 之间的关系, 因此可以将其定义为乘积空间 $X \times Y$ 上的二元模糊关系 R 。

3. 模糊推理

1975 年, Zadeh 利用模糊变换关系, 在广义前向推理法的基础上, 提出了模糊逻辑推理的合成规则, 建立了统一的数学模型, 用于对各种模糊推理作统一处理。其推理规则为

前提: 如果 x 是 A , 则 y 是 B

事实: x 是 A'

(2-24)

结论: 那么 y 是 $B' = A' \circ (A \rightarrow B)$

即结论 B' 可用 A' 与由 A 到 B 的推理关系进行合成而得到, 其中的算子 “ $\circ$ ” 表示模糊关系的合成运算, $(A \rightarrow B)$ 表示由 A 到 B 进行推理的关系或者条件, 即 “如果 x 是 A , 那么 y 是 B ” 的简化表示方法。上述定义中 $(A \rightarrow B)$ 的运算存在各种不同的方法, 合成运算一般采用 T 范式算子和 S 范式算子, 应用最广泛的是极大、极小算子。

2.2.4 常见的几种模糊系统

模糊系统是指那些与模糊概念和模糊逻辑有直接关系的系统。从广义上讲常见的模糊系统分为三大类: 纯模糊逻辑系统 (前提变量是语言变量), 高木 - 关野 (T-S) 模糊逻辑系统, 以及具有模糊产生器和模糊消除器的模糊逻辑系统。因模糊化处理, 模糊推理, 非模糊化处理各自有各种不同的选取方法, 因此构成了很多种模糊系统。下面介绍几种常见的模糊系统:

1. 模糊基函数模糊系统及辨识方法

Wang 和 Mendel^[7] 定义了具有模糊产生器和模糊消除器的模糊逻辑系统, 系统采用单值模糊化、乘运算、加权平均反模糊化可得系统输出为,

$$y(\underline{x} | \underline{\theta}) = \frac{\sum_{j=1}^k \theta_j \left(\prod_{i=1}^n \mu_{A_i}^j(x_i) \right)}{\sum_{j=1}^k \left(\prod_{i=1}^n \mu_{A_i}^j(x_i) \right)} \quad (2-25)$$

Wang 对模糊基函数模糊系统的辨识首先给出了基于 BP 神经网络原理提出了反向传播学习辨识算法, 该算法执行的是一个非线性寻优过程, 因此可能造成收敛较慢或陷入局部极小点。为了克服这一缺点, 又提出了正交最小二乘法, 具体来讲就是先取定模糊逻辑系统的某些参数, 以便将系统表示成模糊基函数的线性组合, 然后在利用正交最小二乘法算出主要的模糊基函数及相应的最优系数。对复杂问题而言以上介绍的辨识方法计算量较大, 接着又给出了表格查寻学习算法。需要注意的是, 上面的算法均不能保证学习后的模糊逻辑系统在任意精度上拟合样本集合中每一个输入输出数据对, 随后提出一种基于最近邻聚类原则的最优模糊逻辑系统的辨识方法^[8]。

2. 高木 - 关野 (T-S) 模糊系统及辨识方法

Takagi 和 Sugeno^[9] 提出的多输入、单输出的 MISO 模糊逻辑系统, 系统采用单值模糊化、乘运算、加权平均可得系统输出为

$$y(\underline{x} | \underline{\theta}) = \frac{\sum_{j=1}^k f^j(x_1, x_2, \dots, x_n) \left(\prod_{i=1}^n \mu_{A_i}^j(x_i) \right)}{\sum_{j=1}^k \left(\prod_{i=1}^n \mu_{A_i}^j(x_i) \right)} \quad (2-26)$$

T-S 模型的辨识首先由 Takagi 和 Sugeno 提出的, 通常包括前提结构辨识、前提参数辨识和结论参数辨识方法。这种辨识方法的突出优点是辨识精度高, 但是, 由于计算量偏大, 对于快速的动态过程难以在线实施。时隔不久, Sugeno 和 Kang^[10] 又提出了前提结构和结论结构的辨识方法, 而后, Sugeno 和 Tanaka^[11] 又提出了 T-S 模型的连续辨识方法, 该方法将辨识分为两级, 一是监控级, 另一是调整级。监控级用模糊调整规则集来确定参数调整策略。以上介绍的几种辨识方法都是将 T-S 模型的结构辨识和参数辨识分开, 也就是在柯西线性模糊模型 (QLFM) 意义上简化了辨识问题, 但却不能保证所得模型的结构是最佳的。R. R. Yager 和 D. P. Filer^[12] 引入采样可能性分布概念同时辨识结构和参数, 并用置信度概念简化复杂的 QLFM 和 QNFM (柯西非线性模糊模型)。该算法将模型结构的辨识转化为输入空间的分配问题, 使得结构辨识和参数辨识得以统一。T-S 模糊系统和基函数模糊系统在模糊建模与自适应控制中得到了广泛的应用。

3. 标准可加性 (SAM) 模糊系统及辨识方法

Kosko^[13] 通过对一般模糊系统的映射关系分析, 提出了可加性模糊逻辑系统, 而标准可加性模糊逻辑系统是可加性模糊逻辑系统中最重要的一部分。从映射角度, 论域空间与输出空间局部是一种蛋形或椭圆体映射关系, 而全局上采用加权平均的方式形成一种模糊系统, 它是从映射的角度去研究模糊系统, 同时它的应用范围也要更广泛。模糊基函数 (FBF) 模糊逻辑系统和高木-关野 (T-S) 模糊逻辑系统实际上是标准可加性系统特殊情况下的模型。采用单值模糊化、乘运算、加权平均反模糊化可得标准可加性模糊系统输出为

$$y(\underline{x} | \underline{w}) = \frac{\sum_{j=1}^k w_j a_j(x) V_j c_j}{\sum_{j=1}^k w_j a_j(x) V_j} = \sum_{j=1}^k p_j(x) c_j \quad (2-27)$$

在可加性模糊系统中可采用两种神经网络来改变椭圆规则^[14-15], 第一种神经网络采用无监督学习来发现突触量化矢量的协方差矩阵, 量化矢量在稀疏的或有噪声的数据域内跳跃, 然后出现一较大的协方差椭圆, 提供一确定性较小的规则, 量化矢量在密集或有噪声较小的数据域内跳跃较小, 提供一精确的规则。另一种神经网络采用有监督学习, 通过随机梯度下降的方式调整规则, 采用了混合神经网络联合无监督和有监督学习以发现和调整椭圆规则。

4. 其他模糊逻辑系统

国内外主要文献上模糊建模与控制几乎都采用以上几种模糊逻辑系统, 但是在个别文献上也有采用其他模糊逻辑系统, 如文献 [16] 中采用的模糊动态逻辑系统用于模糊建模与控制, 东北大学张化光教授在文献 [17] 中提出的模糊双曲正切模糊逻辑系统用于模糊建模与控制。

2.2.5 模糊系统的通用逼近性

万能逼近定理：对于任何定义在致密集 $U \in R^n$ 上的连续函数 g ，及任意的 $\varepsilon > 0$ ，存在一模糊逻辑系统 f ，使得： $\sup_{x \in U} |f(x) - g(x)| < \varepsilon$ 成立。

所谓模糊系统的通用逼近性是指模糊系统是否能以任意精度逼近紧致集上的任意连续实函数，模糊系统的理论研究和实际应用都是建立在模糊系统通用逼近性的基础之上的。这是因为从数学上看，模糊系统是从输入论域到输出论域的一个函数映射。当模糊系统用作系统建模和辨识时，通用逼近性决定了它是否能够逼近任意连续的非线性动态模型，当模糊系统用作控制时，通用逼近性决定了它是否能够对任意非线性动态对象跟踪任意连续的非线性时间函数，并实现所要求的闭环系统动态品质。可见对模糊系统通用逼近性的研究，无论在理论上还是应用上都有极为重要的意义。

各种类型的模糊系统都被证明是万能函数逼近器，它们能一致逼近定义在定义域上的任意连续函数到任意高的逼近精度。Kosko^[18]考虑了可加性模糊系统，主要应用了有限覆盖定理。Wang 讨论了基于乘积模糊推理、用重心法解模糊和 Gauss 型隶属函数的模糊系统模型，主要应用了 Stone Weistrass 定理。Buckley^[19-21]对线性解模糊的模糊控制器模型上进行了讨论，以及其他一些模糊系统都被证明是万能函数逼近器，它们能一致逼近在致密上的任意给定的连续函数，这些理论都是模糊系统作为万能逼近器的存在性理论。

给定一个非线性函数，人们感兴趣的是如何构造一个模糊系统来逼近这个函数到所期望的逼近精度。为此，一些学者提出了从输入-输出数据对来构造模糊逼近器的不同方法，一些学者给出了构造一个模糊系统的充分与必要条件。模糊系统的通用逼近性同自适应模糊控制一样，是模糊控制另一个重要的研究方向，国内外众多学者在通用逼近性方面进行了深入的研究并已取得一些重要成果。关于这方面的内容参见文献 [22]。

2.3 WM 算法及需改进的问题

复杂工业过程常常具有强非线性、不确定性、多变量、强耦合以及工况变化频繁等特点，动态特性难以用精确的数学模型描述。实际应用中，人们发现利用 IF-THEN 规则建立系统的模糊描述模型是一种行之有效的方法。众所周知，模糊规则的获取是模糊建模的关键，通常模糊规则的产生有以下三种方法：

- 1) 基于专家经验和知识产生规则。
- 2) 从样本数据中自动产生规则。
- 3) 样本数据结合专家知识产生混合规则。

近年来，从样本数据中自动产生规则引起了人们的极大关注，其主要原因在于简单系统可以基于专家知识产生规则，但是当变量增多及变量的模糊集分割过多时，单纯依靠专家知识获取模糊规则是十分困难的。因此，遗传算法、模糊聚类和神经网络等方法相继被提出用于从样本数据中自动产生规则^[23-27]。然而，这些方法要求不断地反复学习或是产生规则的机理过于复杂使得工程上难以实施。文献 [28] 提出从样本数据中获取模糊规则的 WM 算

法（也叫查表法），此算法的特点是简单实用，现已成为该领域的经典方法并被广泛引用。但 WM 算法的主要缺点是产生的模糊规则库缺乏良好的完备性和鲁棒性。文献 [29] 针对 WM 算法的缺陷，采用数据挖掘技术做了进一步改进工作。但我们认为这两种算法产生的模糊规则库仍然存在可改进、可完善的空间^[30-31]。下面我们首先分析 WM 算法产生模糊规则库缺乏良好完备性和鲁棒性的根源，然后进一步分析 WM 改进算法（iWM 算法）存在的问题，接下来给出一种新的数据挖掘技术产生模糊规则库的算法（DM 算法），最后将这三种算法进行仿真对比研究。

2.3.1 WM 算法的主要步骤

设给定一组期望的输入 - 输出数据对

$$(x_1^{(1)}, x_2^{(1)}; y^{(1)}), (x_1^{(2)}, x_2^{(2)}; y^{(2)}), \dots \dots \dots \quad (2-28)$$

式中 x_1, x_2 ——输入；

y ——输出。

现在的任务是：从式（2-28）给出的输入 - 输出数据对中产生一组模糊“如果 - 则”规则，并用这些规则确定出所需要的模糊逻辑系统 $f: (x_1, x_2) \rightarrow y$ 。

这种方法分如下 5 步组成。

步骤 1：把输入空间和输出空间划分为模糊区间

为了方便描述和理解，考虑一个两输入单输出的系统。设 x_1, x_2 和 y 的取值范围分别为 $[x_1^-, x_1^+]$, $[x_2^-, x_2^+]$ 和 $[y^-, y^+]$ ，这里，变量的取值范围是指该变量很可能处于该范围以内。将每一个变量的取值范围再划分成 $2N+1$ 个区间，将这 $2N+1$ 个区间分别表示为 $SN, \dots, S1, CE$ （中点）， $B1, \dots, BN$ （从小到大），且每个区间都对应一个模糊隶属函数。在图 2-3 的例子中， x_1 的取值范围分为 5 个区间（ $N=2$ ）， x_2 的取值范围分为 7 个区间（ $N=3$ ）， y 的取值范围分为 5 个区间（ $N=2$ ）。其隶属函数的形状均是三角形的；区间的中点对应于三角形的纵向顶点，且在该点上隶属函数取值为单位值 1。三角形的另两个横向顶点分别位于两个相邻区间的中点上，且隶属函数在这两个顶点上的值等于零。当然，区间的划分和隶属函数的选择也存在其他的选择。

步骤 2：由已知的输入 - 输出数据对产生模糊规则

首先，求出不同区间上已知数据 $x_1^{(i)}, x_2^{(i)}$ 和 $y^{(i)}$ 对应的隶属度。例如图 2-3 中， $x_1^{(1)}$ 在 $B1$ 区间上的隶属度为 0.8，在 $B2$ 上为 0.2，而在其他区间上为 0。同样， $x_2^{(2)}$ 在 CE 上的隶属度为 1，而在其他区间上均为 0。其次，将已知数据 $x_1^{(i)}, x_2^{(i)}$ 和 $y^{(i)}$ 分别定位于最大隶属度对应的区间上。例如，在图 2-3 中， $x_1^{(1)}$ 被认为在 $B1$ 区间上， $x_2^{(2)}$ 被认为在 CE 区间上。

最后，从每一对较满意的输入 - 输出数据对中产生一条规则。例如：

$(x_1^{(1)}, x_2^{(1)}; y^{(1)}) \Rightarrow [x_1^{(1)} (B1 \text{ 上隶属度最大, } 0.8), x_2^{(1)} (S1 \text{ 上隶属度最大, } 0.7), y^{(1)} (CE \text{ 上隶属度最大, } 0.9)] \rightarrow \text{规则 1: 如果 } x_1 \text{ 为 } B1 \text{ 且 } x_2 \text{ 为 } S1, \text{ 则 } y \text{ 为 } CE。$
 $(x_1^{(2)}, x_2^{(2)}; y^{(2)}) \Rightarrow [x_1^{(2)} (B1 \text{ 上隶属度最大, } 0.6), x_2^{(2)} (CE \text{ 上隶属度最大, } 1), y^{(2)} (B1 \text{ 上隶属度最大, } 0.7)] \rightarrow \text{规则 2: 如果 } x_1 \text{ 为 } B1 \text{ 且 } x_2 \text{ 为 } CE, \text{ 则 } y \text{ 为 } B1。$

用上述规则产生的是“逻辑与”规则，即在规则中只有当“如果”部分的条件都同时满足，“则”部分的结果才会发生。就现在的问题而言，即如何从数据信息中得出模糊规则来，只有“逻辑与”型规则满足要求。因为前提条件是某个输入向量的不同分量需要同时满足。

步骤 3: 为每一条规则赋予一个置信度

由于有许多对数据信息，而每一对产生一条规则，这样就很可能出现自相矛盾的规则，即出现“如果”部分相同而“则”部分不相同的规则。解决这个矛盾的一种方法就是为每一个数据对产生的规则赋予一个置信度，最后在自相矛盾的规则中选用具有最高置信度的那条规则。这样就不仅解决了矛盾，而且也使规则的数目大大减少。

用如下的乘积策略为每一条规则赋予一个置信度：对于“如果 x_1 为 A 且 x_2 为 B ，则 y 为 C ”这样一条规则，其置信度 D (规则) 可定义为

$$D(\text{规则}) = \mu_A(x_1) \mu_B(x_1) \mu_C(y) \quad (2-29)$$

举例说明，规则 1 的置信度为 (见图 2-3)

$$\begin{aligned} D(\text{规则 1}) &= \mu_{B1}(x_1) \mu_{S1}(x_2) \mu_{CE}(y) \\ &= 0.8 \times 0.7 \times 0.9 = 0.504 \end{aligned} \quad (2-30)$$

规则 2 的置信度为

$$\begin{aligned} D(\text{规则 2}) &= \mu_{B1}(x_1) \mu_{CE}(x_2) \mu_{B1}(y) \\ &= 0.6 \times 1 \times 0.7 = 0.42 \end{aligned} \quad (2-31)$$

在实际应用中，常常还会得到这些数据对的某些先验信息。比如，让一个专家来检查所得到的数据对，这个专家就很可能看出哪些数据有用，哪些数据是关键性数据，哪些数据不太合理或由量测误差产生。因此，可以为每一对数据再附加一个信任度以表示专家对该数据可靠性的相信程度。

设数据 $(x_1^{(1)}, x_2^{(1)}; y^{(1)})$ 的附加信任度为 $\mu^{(1)}$ ，则再将规则 1 的置信度重新定义为

$$D(\text{规则 1}) = \mu_{B1}(x_1) \mu_{S1}(x_2) \mu_{CE}(y) \mu^{(1)} \quad (2-32)$$

即一条规则的置信度定义为规则各分量的隶属度与产生该规则的数据信任度之乘积。这一点在实际应用中非常重要，因为在真实数据中，每个数据的可靠性不同，例如，某些数据可能非常不合理（既“反常数据”）。对合理的数据给一个较高的信任度，而对不合理的数据设一个很低的信任度。这样，就把专家对数据的认可程度作为另一种信息，用同一种形式包含进去了。如果强调数据的客观性而不想加进专家对数据信息的主观认同，则只需将所有数据

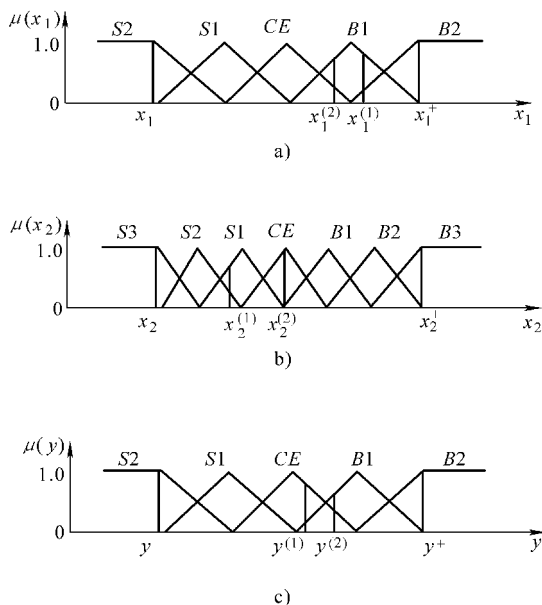


图 2-3 变量的隶属度函数

的专家信任度全部取单位值就可以了。

步骤4：组合模糊规则库的产生

图2-4给出的查寻表格就是一个模糊规则库，用以下的准则把模糊规则填入表中的空格：组合模糊规则库中的规则即可以来源于数据产生的规则，也可来源于语言性规则，如果模糊规则库中某一空格对应的规则不止一个，则选用具有最大置信度的那一条规则。这样，数据和语言两类信息就用一种统一的方式——组合模糊规则库统一起来了。如果语言规则为“逻辑或”（即规则中“如果”部分的任一条件满足，就有“则”部分发生），则它填满“如果”部分所对应区间的某一行或某一列所有的空格。例如有这样一条语言规则：“如果 x_1 为 $S1$ 或 x_2 为 CE ，则 y 为 $B2$ ”，则对图2-4所表示的模糊规则库来讲，就需将 $B2$ 填满 $S1$ 列的七个空格和 CE 行的五个空格。所有这些空格中 $B2$ 的置信度都等于这条“逻辑或”规则的置信度。

$B3$					
$B2$					
$B1$					
CE					
$S1$					
$S2$					
$S3$					
	$S2$	$S1$	CE	$B1$	$B2$

图2-4 WM方法的查寻表格

步骤5：根据组合模糊规则确定出映射关系

用下面的模糊消除策略来从已知的输入 (x_1, x_2) 中求出输出控制量 y ：首先，对于已知的输入 (x_1, x_2) ，需用乘积运算来综合第 i 条规则的所有前提条件，以求出 (x_1, x_2) 相对应的输入控制的置信度 $\mu_{O^i}^i$ ，即

$$\mu_{O^i}^i = \mu_{I_1^i}(x_1)\mu_{I_2^i}(x_2) \quad (2-33)$$

式中 O^i ——规则 i 的输出区间；

I_j^i ——规则 i 中第 j 个分量输入区间。

对规则1有

$$\mu_{CE}^1 = \mu_{B1}(x_1)\mu_{S1}(x_2) \quad (2-34)$$

则用下列的中心平均模糊消除公式就可以求出输出 y

$$y = \frac{\sum_{i=1}^M \mu_{O^i}^i \bar{y}^i}{\sum_{i=1}^M \mu_{O^i}^i} \quad (2-35)$$

式中 $\bar{y}^i$ ——区间 O^i 上中心点的取值（模糊区间中心点的定义是：在该区间上隶属函数取得单位值1的所有点中，（横轴）绝对值最小的点为中心点）；

M ——组合模糊规则库中模糊规则的数目。

从步骤1到步骤5可以看到，这个方法的整个过程既简单又直接，因为它是一个一步生成过程，不需要费时的反复学习；因此，这个方法同样具有模糊系统优于神经网络的一大优点，即构造起来简单又迅速。

上述五步过程可以很容易地推广到一般性多输入—多输出的情况。步骤1到步骤4均与输入和输出个数无关。在步骤5中，只需将方程（2-33）中的 $\mu_{O^i}^i$ 代之以 $\mu_{O_j^i}^i$ 即可，这里 j 表

示输入向量的第 j 个分量 (O_j^i 为规则 i 的第 j 个分量所对应的区间; $\mu_{O_j^i}$ 对所有的 j 都取相同的值), 同时将方程 (2-35) 变换为下式

$$y = \frac{\sum_{i=1}^M \mu_{O_j^i} \bar{y}_j^i}{\sum_{i=1}^M \mu_{O_j^i}} \quad (2-36)$$

式中 $\bar{y}_j^i$ —— O_j^i 的中心点。

2.3.2 WM 算法需要改进的地方

模糊系统已被证明能以任意精度逼近紧集上的任意函数, 下面简要回顾用查表法设计模糊系统其主要步骤及需要改进的地方。

步骤 1. 把输入与输出空间划分为模糊空间。

步骤 2. 由一个输入 - 输出数据对产生一条模糊规则。

步骤 3. 对步骤 2 中的每条规则赋予一个强度。

步骤 4. 创建模糊规则库。

步骤 5. 基于模糊规则库构造模糊逻辑系统。

定义 1 (完备性): 输入变量定义的模糊集所有可能组合的模糊子空间上都有规则分布, 称这样的规则库是完备的。

由查表法产生的模糊规则库是不完备的。在步骤 1 把第 i 个输入变量 x_i 划分 k_i 个模糊集, 其中 $i=1, 2, \dots, n$ 。那么, 查表法产生的规则数量受两个数约束: m (输入 - 输出数据对的数量) 和 $\prod_{i=1}^n (k_i)$ (由输入变量定义的模糊集的所有可能组合的数量)。如果 $m < \prod_{i=1}^n (k_i)$ 将使规则库产生第一类不完备性, 既样本数据过少造成的不完备。此外, 如果 $m \geq \prod_{i=1}^n (k_i)$ 也有可能使规则库产生另一类的不完备性, 即某些样本数据分布在同一模糊子空间 (对应于同一个格子) 从而产生了矛盾规则, 由步骤 3 可知, 强度最大的规则被选择。

我们考虑有两个输入变量 x_1, x_2 , 分别被划分成五个和七个模糊集的例子, 如图 2-3 和图 2-4 所示。假定样本集 $S = \{(x_1^1, x_2^1; y^1), \dots, (x_1^{10}, x_2^{10}; y^{10})\}$ 中包括 10 对理想输入 - 输出数据对, 从图 2-3 中可以看, $x_1^{(1)}$ 在 $B1$ 模糊集上的隶属度值最大, $x_2^{(1)}$ 在 $S1$ 模糊集上的隶属度值最大, $y^{(1)}$ 在 CE 模糊集上的隶属度值最大。由查表法可以得到下面的 IF - THEN 模糊规则

$$R^1: \text{If } x_1 \text{ is } B1 \text{ and } x_2 \text{ is } S1 \text{ Then } y \text{ is } CE \quad (2-37)$$

也就是在图 2-4 模糊规则库的模糊子空间填上 CE 。以此类推, 其他数据对产生的规则也相应填充到各自的模糊子空间。但是, 规则库的模糊子空间最多有 10 个被填充, 其他模糊子空间为空, 所以, 对小样本集来说使用查表法产生的规则库是不完备的。当然, 对大样本集来说, 如果数据对不是均匀分布在由输入变量定义的模糊集所有可能组合的模糊子空间上, 也会产生规则库的不完备性。

定义 2 (鲁棒性): 当样本集中混入少量坏的数据对干扰时, 由该样本集生成的规则库所确定的模糊模型仍能保证良好的性能, 称这样的规则库是鲁棒的。

由查表法产生的模糊规则库不具有良好的鲁棒性。在实际中所得到的样本数据常常混有坏的数据对, 如果坏的数据对产生的规则强度大于好的数据对产生的规则强度, 那么在规则库中将包含坏规则, 从而影响模糊模型的性能。

对于图 2-3 中的例子来说, 假定在原有十对理想输入-输出数据对中增加一个坏的数据对 $(x_1^{(11)}, x_2^{(11)}, y^{(11)})$, 并假定 $x_1^{(1)} = x_1^{(11)}, x_2^{(1)} = x_2^{(11)}, y^{(1)} \neq y^{(11)}$, 由于 $y^{(11)}$ (虚线) 在 $B1$ 模糊集上的隶属度值最大, 由后增加的数据对产生下面的 IF-THEN 模糊规则

$$R^2: \text{If } x_1 \text{ is } B1 \text{ and } x_2 \text{ is } S1 \text{ Then } y \text{ is } B1 \quad (2-38)$$

在相同输入点的情况下, 由于 $y^{(1)}$ 和 $y^{(11)}$ 的不同从而导致了两条矛盾规则。按照查表法的步骤 3 强度定义, 后一条规则被采纳, 从而导致了坏数据产生坏规则的现象。所以说, 仅由一个数据对就产生一条模糊规则是不可靠的, 产生这种现象的原因在于产生规则的支持度不够。

查表法也考虑了样本数据中可能有不可靠数据问题, 该文解决的办法是给每对数据一个可靠度, 但是该方法很难实施。首先, 人为地判断数据的好坏是很难的; 其次, 如果样本集过大, 对每个样本数据都人为地进行判断也是不现实的。

2.4 iWM 算法及需改进的问题

2.4.1 iWM 算法的主要思想

文献 [29] 针对 WM 所存在的问题进行了改进与完善。该文提出了三类问题, 并给出相应的解决方法。从文中可以看出, 解决问题的核心思想是相同的。下面以该文的第一类问题和解决方法为例进行分析。

对于输入输出数据对 $(x^{(p)}; y^{(p)})$, 文献 [29] 首先用下式计算了权值

$$w^{(p)} = \prod_{j=1}^m \mu_{A_{ij}}(x_{ij}^{(p)}) \quad (2-39)$$

在上式中, 若 $\sum_{p=1}^N w^{(p)} = 0$, 那么将无规则产生; 否则将 $w^{(p)}$ 作为 $y^{(p)}$ 的权值, 然后用下式计算输出的加权平均值

$$av = \frac{\sum_{p=1}^N y^{(p)} w^{(p)}}{\sum_{p=1}^N w^{(p)}} \quad (2-40)$$

接下来, 可以用下面的两种形式之一选择输出变量的模糊集。第一种形式, 设输出变量在输出空间有 K 个模糊集, 也就是 $B = \{B^1, \dots, B^K\}$, 那么输出变量由下式决定最终选择哪个模糊集

$$\mu_{B^*}(av) \geq \mu_{B^*}(av) \quad (2-41)$$

第二种形式, 首先计算加权的方差

$$\sigma = \frac{\sum_{p=1}^N |y^{(p)} - av| w^{(p)}}{\sum_{p=1}^N w^{(p)}} \quad (2-42)$$

然后输出变量的模糊集 B 可用下式来表示

$$\mu_B(y) \geq \mu(y; av, \sigma) \quad (2-43)$$

2.4.2 iWM 算法需改进的问题

从上式可以看出, iWM 算法为了提取某条规则使用了所有样本数据都参与选举的原理, 比早期的 WM 算法仅用一个样本数据选择一条规则有了改进, 这种遍历保证了模糊规则库空间的完备性。但从鲁棒性角度看, 式 (2-40) 和式 (2-41) 改成下面的形式更合理

$$av_{B^*} = \frac{\sum_{p=1}^N \mu_{B^*}(y^{(p)}) w^{(p)}}{\sum_{p=1}^N w^{(p)}} \quad (2-44)$$

可用下式决定结论变量选择的最优模糊集

$$av_{B^*} \geq av_{B^*} \quad (2-45)$$

实际上, 式 (2-44) 和式 (2-45) 与后面提到的 DM 算法以及遍历所有的输出模糊集空间是完全一样的。

式 (2-40)、式 (2-41) 和式 (2-44)、式 (2-45) 都完成了结论变量最优模糊集的选择问题, 但后者比前者具有更好的鲁棒性。原因在于式 (2-44) 是对输出变量 $y^{(p)}$ 的隶属函数值进行了加权平均, 而式 (2-40) 是直接对输出变量 $y^{(p)}$ 进行了加权平均。下面结合实际数据说明这种改进的原因, 假设输出数据 $y^{(p)}$ 在理想的情况下是在 $\{-3, 3\}$ 之间, 由于某种原因使得 $y^{(p)}$ 样本集中混入一个大的干扰数据 (如 30), 那么用式 (2-40) 很难加权平均掉这个大的干扰数据。但用式 (2-44) 却能做得很好, 因为各个数据都被隶属化到 $[0, 1]$ 之间等同地位, 即使样本集中混入少量干扰数据, 也能被其他好数据有效地屏蔽。

2.5 一种新的 DM 算法建立模糊模型

2.5.1 提取完备的模糊规则库

近年来, 随着计算机和数据库技术的发展, 数据挖掘已成为一个逐渐兴起的研究领域。它是一种基于数据库的自动信息提取技术。大型关系数据库中的数据挖掘是数据库研究的一个新课题, 作为数据挖掘的一部分, 关联规则的挖掘近年来受到许多人的重视。下面将针对在文献 [28-29] 中使用的一类多输入单输出模糊逻辑系统, 采用数据挖掘方法产生模糊

规则。这种方法与 WM 算法及 iWM 算法相比,产生的规则库具有良好的完备性和鲁棒性,从而提高了模糊模型的精度。

问题描述:假定通过实验或其他手段得到与文献 [29] 相同的输入和输出数据对集合

$$(X^{(p)}; y^{(p)}), p = 1, 2, \dots, N \quad (2-46)$$

式中 $X = (x_1, \dots, x_n)^T \in R^n$;

$y \in R$ 。

我们的问题是如何从这些输入输出数据对中提取模糊规则并用该规则库建立模糊模型。下面给出用数据挖掘方法产生模糊规则库的主要步骤。

步骤 1. 输入和输出变量的模糊区间划分

首先,设前提输入变量 x_i 与结论输出变量 y 的域值区间分别为 $[x_i^-, x_i^+]$ 和 $[y^-, y^+]$, $i = 1, \dots, n$ 。然后,在每一个变量区间上定义 c_i 个模糊集,用 $A(x_i) = \{A_1^{x_i}, \dots, A_{c_i}^{x_i}\}$ 和 $B(y) = \{B_1^y, \dots, B_{c_0}^y\}$ 表示对应的模糊集合。最后,给每个模糊集合分配隶属度函数。

步骤 2. 普通记录转化成模糊记录

将步骤 1. 的输入输出数据对存放到普通型关系数据库表中。简单地说,输入输出数据对在关系数据库中以表的形式被保存,其中行代表一条记录,列代表一条记录的一个属性。如果用 $L = \{x_1, \dots, x_n, y\}$ 表示属性集合,任一记录(行) t_p 表示属性集合 L 对应的记录值。用下面的 T_L 定义记录集合

$$T_L = \{t_1, \dots, t_N\} \quad (2-47)$$

如果用 $A(x_i) = \{A_1^{x_i}, \dots, A_{c_i}^{x_i}\}$ 表示前提属性 x_i 的模糊集合, $B(y) = \{B_1^y, \dots, B_{c_0}^y\}$ 表示结论属性 y 的模糊集合,可得到集合 $A = \{\cup_{i=1, \dots, n} A(x_i)\}$ 和 $B = \{B(y)\}$ 。对于属性 $F = A \cup B$, 用下面的 T_F 定义记录在该属性上的集合

$$T_F = \{\mu_F(t_1), \dots, \mu_F(t_N)\} \quad (2-48)$$

通过上面的定义,我们可以将模糊记录保存在另一关系表中。列表示属性 F , 行表示取值在 $[0; 1]$ 之间的记录。至此完成了普通记录转化为模糊记录的工作。

步骤 3. 计算模糊规则的支持度

从数据挖掘的角度看,如果一条模糊规则有实际意义,它必须具有足够的支持度。支持度反映了样本记录对该条规则的支持程度,现定义支持度如下

$$\text{Sup}(X \Rightarrow y) = \frac{\sum_{p=1}^N \mu_{(B_{l_0}^y)^p}(y) \prod_{i=1}^n \mu_{(A_{l_i}^{x_i})^p}(x_i)}{\sum_{p=1}^N \prod_{i=1}^n \mu_{(A_{l_i}^{x_i})^p}(x_i)} \quad (2-49)$$

式中 $\mu_{(A_{l_i}^{x_i})^p}(x_i)$ 和 $\mu_{(B_{l_0}^y)^p}(y)$ ——别表示记录 p 的隶属度函数值;

N ——在 T_F 中的记录总数;

$l_i = 1, \dots, c_i$ 。

为方便计算,也可采用下面的形式定义支持度

$$\text{Sup}(X \Rightarrow y) = \frac{1}{N} \sum_{p=1}^N \mu_{(B_{l_0}^y)^p}(y) \prod_{i=1}^n \mu_{(A_{l_i}^{x_i})^p}(x_i) \quad (2-50)$$

步骤 4. 创建完备的模糊规则库

算法具体描述如下：首先，要保证由前提变量构成的每个模糊子空间都能被遍历；其次，结论变量在模糊子空间选用的最佳模糊集，由规则的最大支持度决定。该算法的实施是容易、有规律的，只需多个循环语句就可以完成。

输入. 1) 样本数据；2) 用户指定 c_i ；3) n 和 N

输出. 产生一个具有完备性的模糊规则库

步骤 4.1. 初始化

定义全局变量 j , Max - sup 和 Max - num

步骤 4.2. 基于最大支持度获取模糊规则

For ($j = 1; c_1 \times \cdots \times c_n; j++$) {

 转到步骤 4.3;

 基于最大 Max - num 得到 Rule[j] :

}

End

步骤 4.3. 遍历 X 的所有模糊集并计算隶属度值

For ($l_1 = 1; c_1; l_1++$) {

 计算 $\mu_{A_{l_1}^x}(x_1)$;

$\vdots$

For ($l_1 = 1; c_1; l_1++$) {

 计算 $\mu_{A_{l_1}^x}(x_n)$;

 转到步骤 4.4;

}

$\vdots$

}

Return

步骤 4.4. 遍历 y 的所有模糊集并计算支持度

For ($l_0 = 1; c_0; l_0++$) {

 计算 $\mu_{B_{l_0}^y}(y)$;

 用方程 (4) 或 (5) 进行计算:

$E[l_0] = \text{Sup}(X \Rightarrow y)$;

IF ($E[l_0] > E[0]$) {

 Max - sup = $E[l_0]$;

 Max - num = l_0 ;

$E[0] = E[l_0]$;

 }

}

Return

步骤 5. 计算模糊规则的信任度

一个模糊系统就是一个从输入映射到输出 If - Then 规则的集合, 一般规则越多逼近效果越好。但随着输入变量个数和变量的模糊集分割数增多, 模糊规则的数目也呈指数增长, 那么无疑会产生规则爆炸。上面挖掘方法产生的规则是完备的, 当前提变量的个数和变量的模糊集分割数不是过多时, 完全可以采用这些完备规则。但当前提变量的个数和变量的模糊集分割数过多时, 可以通过下面的规则信任度来约简规则

$$\text{Conf}(X \Rightarrow y) = \frac{\text{Sup}(X \Rightarrow y)}{\frac{1}{N} \sum_{p=1}^N \prod_{i=1}^n \mu_{(A_{l_i}^y)^p}(x_i)} \quad (2-51)$$

这种约简所产生的规则库不完备性是在规则库完备性的基础上, 为了便于系统实施人为约简那些信任度较低的规则, 它与 WM 算法所产生规则库的不完备性具有本质的不同。

步骤 6. 增加前提和结论变量的模糊集合数

对于前提和结论变量的模糊集合数的选择, 可以采用试探法, 给定初始值, 然后通过下面的误差值判定是否满足期望的精度要求

$$PI = \frac{1}{N} \sqrt{\sum_{p=1}^N \|y_j - \hat{y}_j\|^2} \quad (2-52)$$

2.5.2 建立鲁棒模糊模型

由上面最大支持度决定的模糊规则子空间构成如下模糊语句

$$R^{(j)}: \text{if } x_1 \text{ is } A_{l_1}^{x_1}, \dots, x_n \text{ is } A_{l_n}^{x_n}, \text{ Then } y \text{ is } B_{l_0}^y \quad (2-53)$$

其中, $X = (x_1, \dots, x_n)^T \in \mathbf{R}^n$ 和 $y \in \mathbf{R}$ 是模糊系统的输入与输出, $A_{l_i}^{x_i}$ 与 $B_{l_0}^y$ 分别表示前提变量 x_i 与结论变量 y 的模糊集, $l_i = 1, \dots, c_i, j = 1, \dots, M$ 并且 $M = c_1 \times \dots \times c_n$ 。

在建立了上面的模糊规则后, 采用单值模糊化、乘积运算、加权平均反模糊化可得到下面的模糊模型

$$y(X) = \frac{\sum_{j=1}^M \bar{y}^j \prod_{i=1}^n \mu_{(A_{l_i}^{x_i})^j}(x_i)}{\sum_{j=1}^M \prod_{i=1}^n \mu_{(A_{l_i}^{x_i})^j}(x_i)} \quad (2-54)$$

其中, $\bar{y}^j$ 是第 j 条规则在模糊隶属度函数 $\mu_{B_{l_0}^y}(y)$ 取得最大值时所对应的点, 也就是 $\mu_{B_{l_0}^y}(y) = 1$ 时所对应的点。

2.5.3 WM 和 DM 算法仿真对比

利用 WM 和 DM 算法产生的规则及方程 (2-55) 所形成的模糊系统逼近下列非线性函数的对比

$$y(x) = 0.6 \sin(\pi x) + 0.3 \sin(3\pi x) + 0.1 \sin(5\pi x) \quad (2-55)$$

设定输入 x 的范围为 $[-1, 1]$, 并将它模糊分割成 9 个模糊集, 用 $\{S4, \dots, S1, CE, B1,$

$\cdots, B4\}$ 表示; y 分割成 7 个模糊集, 用 $\{S3, \cdots, S1, CE, B1, \cdots, B3\}$ 表示, 模糊隶属度函数采用高斯函数。仿真分三种情况:

第一种情况: 样本数据用 $T = \{t_1, t_2, \cdots, t_{11}\}$ 时 (见表 2-1a), 用 WM 和 DM 算法产生的规则库是相同的 (见表 2-1b)。两种方法所产生的模糊逼近模型也是相同的, 仿真结果如图 2-5 所示。

表 2-1a 第一种情况的样本数据 (理想情况)

Case1	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}
x	-1	-0.8	-0.6	-0.4	-0.2	0	0.2	0.4	0.6	0.8	1
y	0	-0.64	-0.39	-0.39	-0.64	0	0.64	0.39	0.39	0.64	0

表 2-1b 第一种情况产生的规则库

WM	CE	S2	S1	S2	CE	B2	B1	B2	CE
DM	CE	S2	S1	S2	CE	B2	B1	B2	CE
	S4	S3	S2	S1	CE	B1	B2	B3	B4
x									

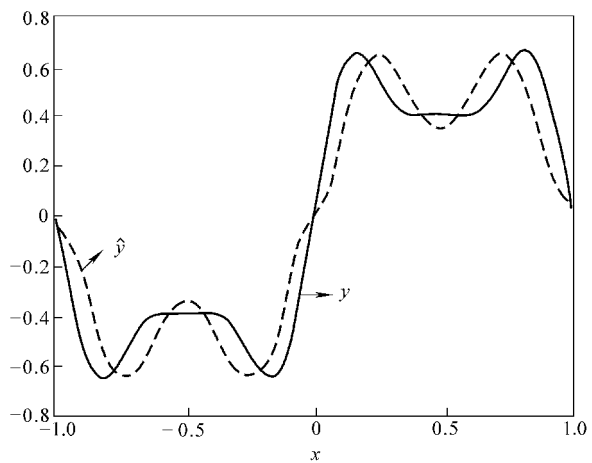


图 2-5 估计模型和实际模型

第二种情况: 将样本集里的数据记录 t_2 去掉, 采用 $T = \{t_1, t_3, \cdots, t_{11}\}$ 时 (见表 2-2a), 用 WM 和 DM 算法产生的规则库是不相同的 (见表 2-2b)。用 WM 产生的规则库是不完备的, 而 DM 算挖掘方法产生的规则库仍是完备的, 仿真结果如图 2-6 所示。图 2-6a 的逼近精度明显低于图 2-6b 的逼近精度, 这种情况说明 DM 算法在完备性方面优于 WM 算法。

表 2-2a 第二种情况的样本数据 (非理想情况)

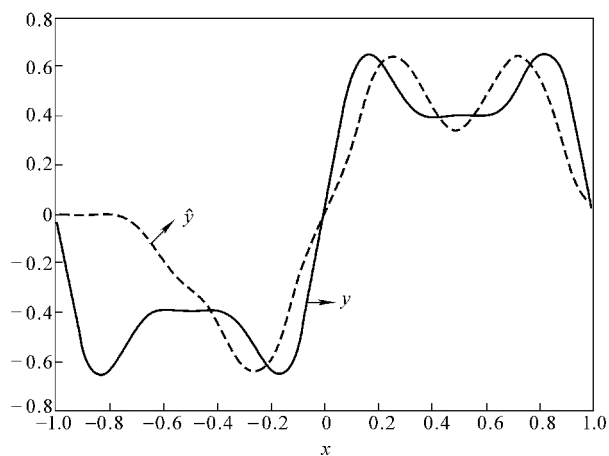
Case2	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}
x	-1		-0.6	-0.4	-0.2	0	0.2	0.4	0.6	0.8	1
y	0		-0.39	-0.39	-0.64	0	0.64	0.39	0.39	0.64	0

表 2-2b 第二种情况产生的规则库（非理想情况）

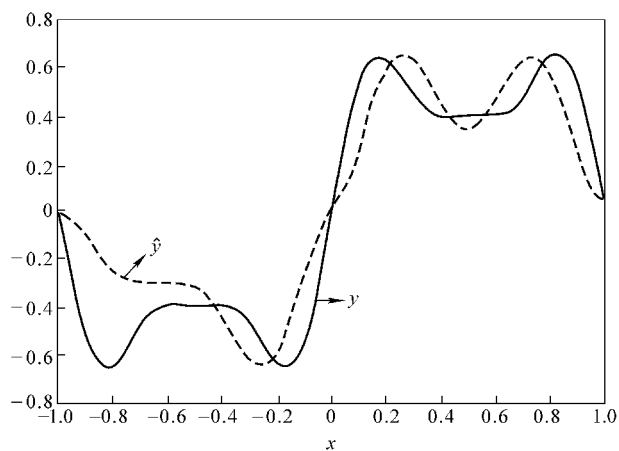
WM	CE	*	S1	S2	CE	B2	B1	B2	CE
DM	CE	S1 *	S1	S2	CE	B2	B1	B2	CE
	S4	S3	S2	S1	CE	B1	B2	B3	B4
x									

表 2-3a 第三种情况的样本数据（非理想情况）

Case3	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}
x	-1	-0.8	-0.6	-0.4	-0.2	0	0.2	0.4	0.6	0.8	1	-0.2
y	0	-0.64	-0.39	-0.39	-0.64	0	0.64	0.39	0.39	0.64	0	0.3



a) WM算法



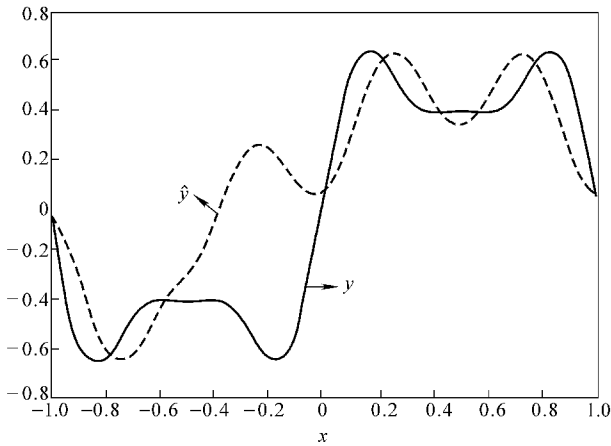
b) DM算法

图 2-6 估计模型和实际模型

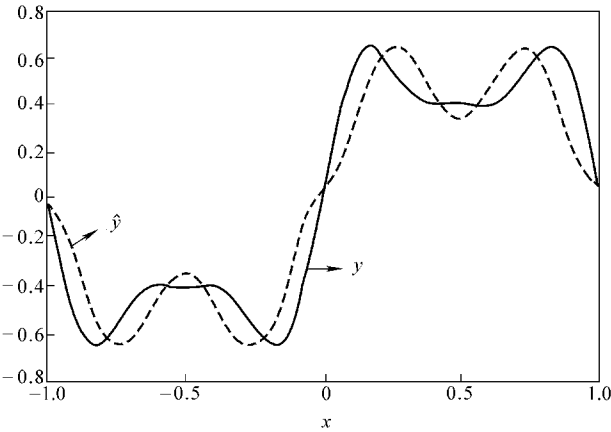
第三种情况：在样本集里增加一个坏的数据记录 t_{12} ，采用 $T = \{t_1, t_2, \cdots, t_{11}, t_{12}\}$ 时（见表 2-3a），用 WM 和 DM 算法产生的规则库是相同的见表 2-3b）。在坏的数据干扰情况下，WM 方法出现了坏数据选择坏规则的现象，而挖掘方法产生的规则库避免了这种现象，仿真结果如图 2-7 所示。图 2-7a 的逼近精度明显低于图 2-7b 的逼近精度，这种情况说明 DM 算法在鲁棒性方面优于 WM 算法。

表 2-3b 第三种情况产生的规则库

WM	CE	S2	S1	B1*	CE	B2	B1	B2	CE
DM	CE	S2	S1	S2*	CE	B2	B1	B2	CE
	S4	S3	S2	S1	CE	B1	B2	B3	B4
x									



a) WM算法



b) DM算法

图 2-7 估计模型和实际模型

注：上面模糊模型的逼近精度不高，是因为前提变量的模糊集分割数和样本数太少造成的。在这个仿真例子中，我们更注重两种方法的对比，故没有注重逼近精度。

2.5.4 iWM 和 DM 算法仿真对比

下面用一个例子来说明本文提出的 DM 算法比 iWM 算法具有更好的鲁棒性。首先分别用 iWM 算法和 DM 算法产生各自的模糊规则库,然后用各自的模糊规则库形成相应的模糊系统逼近下列非线性函数

$$y(x) = 3\sin\left(\frac{\pi}{2}x\right) \quad (2-56)$$

设输入 x 的范围为 $[-1, 1]$, 隶属度函数采用高斯函数; 输出 y 的隶属度函数采用三角函数。

在实际工程中, 得到全部理想数据是十分困难的。我们得到的数据常常带有一些干扰数据, 因此算法具有一定的抗干扰性 (鲁棒性) 是十分重要的。下面通过一个实验对比两种方法的鲁棒性。设理想样本数据为 $T = \{t_1, t_2, \dots, t_{11}\}$ 。为考察鲁棒性, 在样本集里增加一个坏的数据记录 t_{12} , 也就是样本数据为 $T = \{t_1, t_2, \dots, t_{11}, t_{12}\}$ 。

表 2-4 第三种情况的样本数据 (非理想情况)

Case	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}
x	-1	-0.8	-0.6	-0.4	-0.2	0	0.2	0.4	0.6	0.8	1	0
y	-3.00	-2.85	-2.43	-1.76	-0.93	0	0.93	1.76	2.43	2.85	3.00	3

第一种情况: 将输出 x 和输出 y 都分割成 5 个模糊集, 分别用 $\{A_1, \dots, A_5\}$ 和 $\{B_1, \dots, B_5\}$ 表示。在坏的数据干扰情况下, 用 iWM 和 DM 算法计算的平均值和支持度如表 2-5a 和表 2-5b 所示, 由此表计算的模糊模型如图 2-8 所示。其中, 图 2-8a 表示用 iWM 算法获得的模糊模型的逼近仿真结果, 曲线误差为: 0.21899; 图 2-8b 表示用 DM 算法获得的模糊模型的逼近仿真结果, 曲线误差为: 0.13225。图 2-8a 的逼近精度明显低于图 2-8b 的逼近精度, 这种情况说明 DM 算法在完备性方面优于 iWM 算法。

表 2-5a 计算平均值 (情况一)

av	B_1^y	B_2^y	B_3^y	B_4^y	B_5^y
A_1^x	0.9213	0.0787	0.0000	0.0000	0.0000
A_2^x	0.2601	0.7399	0.0000	0.0000	0.0000
A_3^x	0.0000	0.0000	0.4537	0.5463	0.0000
A_4^x	0.0000	0.0000	0.0000	0.6480	0.3520
A_5^x	0.0000	0.0000	0.0000	0.0786	0.9214

表 2-5b 计算支持度 (情况一)

sup	B_1^y	B_2^y	B_3^y	B_4^y	B_5^y
A_1^x	0.1406	0.0119	0.0000	0.0000	0.0000
A_2^x	0.0921	0.1120	0.0171	0.0002	0.0052
A_3^x	0.0035	0.0453	0.1242	0.0453	0.0868
A_4^x	0.0000	0.0002	0.0171	0.1120	0.0973
A_5^x	0.0000	0.0000	0.0000	0.0119	0.1406

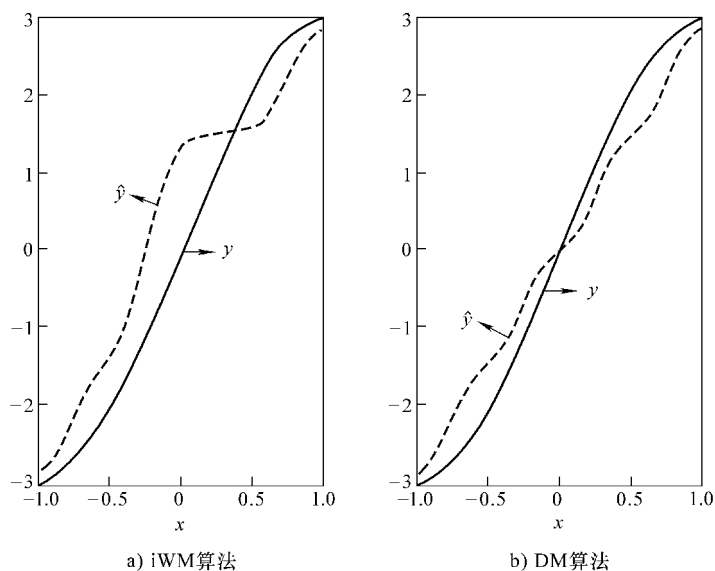


图 2-8 估计模型和实际模型

第二种情况：将输出 x 和输出 y 都分割成 7 个模糊集，分别用 $\{A_1, \dots, A_7\}$ 和 $\{B_1, \dots, B_7\}$ 表示。在坏的数据干扰情况下，用 iWM 和 DM 算法计算的平均值和支持度如表 2-6a 和表 2-6b 所示，由此表计算的模糊模型如图 2-9 所示。其中，图 2-9a 表示用 iWM 算法获得的模糊仿真结果，曲线误差为：0.14296；图 2-9b 表示用 DM 算法获得的模糊仿真结果，曲线误差为：0.0797。图 2-9a 的逼近精度明显低于图 2-9b 的逼近精度，这种情况说明 DM 算法在完备性方面优于 iWM 算法。

表 2-6a 计算平均值 (情况二)

[illegible]

表 2-6b 计算支持度 (情况二)

[illegible]

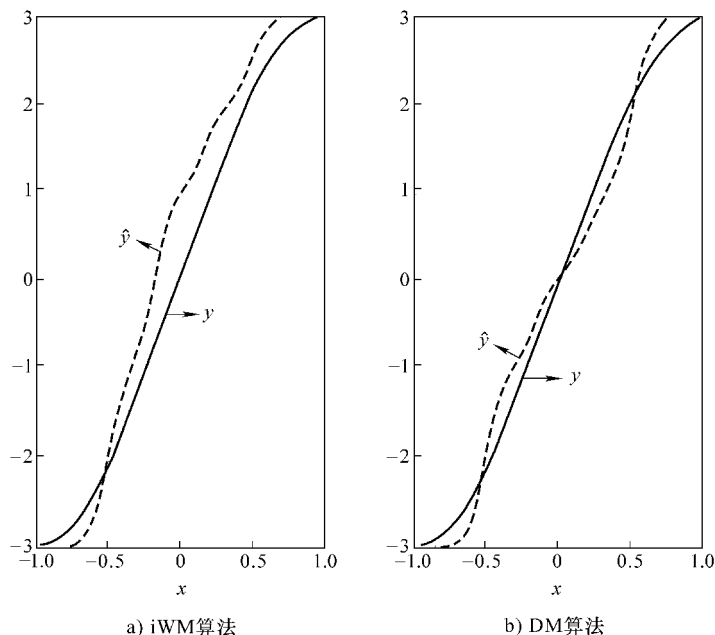


图 2-9 估计模型和实际模型

2.6 本章小结

工业生产中对象的复杂特性使得非线性建模问题具有相当大的挑战性。因此,如何利用现有技术、方法进行有效建模,对软测量和控制等问题具有重要意义。本章主要介绍了非线性系统的建模方法,特别是如何用模糊理论和技术进行智能建模。在模糊建模时,模糊规则的选取是影响模糊建模精度的一个重要因素之一。针对传统的经典方法(iWM和DM算法)产生的模糊规则库缺乏良好的完备性和鲁棒性的弊端,从而影响了模糊建模精度问题。本章提出了用数据挖掘方法产生模糊规则库,用数据挖掘方法产生的模糊规则库具有良好的完备性和鲁棒性。通过仿真对比实验验证了用数据挖掘方法所产生的模糊规则库建立的模糊模型精度比经典方法(iWM和DM算法)更高。最后,对这三种方法给出以下总结:

- 1) WM 算法运用一个采样数据选举一条模糊规则的原理,使得系统缺乏良好的完备性和鲁棒性。
- 2) iWM 算法运用所有采样数据选举一条模糊规则的原理,但由于数据没有正则化,使得系统缺乏良好的鲁棒性。
- 3) DM 算法运用所有采样数据选举一条模糊规则的原理且数据被正则化,使得系统具有良好的完备性和鲁棒性。

参考文献

- [1] 李海青,黄志尧.特种检测技术及应用[M].杭州:浙江大学出版社,2000.
- [2] 李海清,黄志尧.软测量技术原理及应用[M].北京:化学工业出版社,2000.

- [3] 王永富. 非线性系统的模糊建模与自适应控制及其应用 [D]. 沈阳: 东北大学, 2005.
- [4] Zadeh L A. Fuzzy Set [J]. Information and Control, 1965, 8 (2): 338 - 358.
- [5] Mamdani E H. Application of fuzzy algorithms for control of simple dynamic plant [C]. Proc IEEE, 1974, 121: 1585 - 1588.
- [6] 王永富, 柴天佑, 岳恒. 自适应模糊控制现状与展望 [J]. 控制工程, 2006, 17 (2): 1 - 5.
- [7] Wang L X, Mendel J M. Fuzzy basis functions, universal approximation, and orthogonal least squares learning [J]. IEEE Trans. on Neural Networks, 1992, 3 (5): 807 - 814.
- [8] Wang L X. Adaptive fuzzy systems and control: design and stability analysis [M]. Englewood Cliffs, NJ: Prentice Hall, 1994.
- [9] Takagi T, Sugeno M. Fuzzy identification of systems and its application to modeling and control [J]. IEEE Trans. on Syst., Man, Cybern., 1985, SMC - 15 (1): 116 - 132.
- [10] Sugeno M, Kang G T. Structure identification of fuzzy model [J]. Fuzzy Sets and Systems, 1988, 28 (1): 15 - 33.
- [11] Sugeno M, Tanaka K. Successive identification of a fuzzy model and its application to prediction of a complex systems [J]. Fuzzy Sets and Systems, 1991, 42 (3): 315 - 334.
- [12] Yager R R, Filev D P. Unified structure and parameter identification of fuzzy model [J]. IEEE Trans. on Systems, Man and Cybernetics, 1993, 23 (4): 1198 - 1205.
- [13] Kosko, B. Neural networks and fuzzy systems: a dynamical systems approach to machine intelligent [M]. Englewood Cliffs, NJ: Prentice Hall, 1991.
- [14] Dickerson J A, Kosko B. Neural networks and fuzzy function learning with covariance ellipsoids [C]. Proceeding of the IEEE International Conference on Neural Networks, 1993, 2 (9): 9 - 17.
- [15] Dickerson J A, Kosko B. Fuzzy function approximation with ellipsoids rules [J]. IEEE Trans. on Systems, Man and Cybernetics, 1996, 21 (1): 9 - 17.
- [16] Cao S G, Ree N W, Feng G. Analysis and design for a class of complex control systems part II: fuzzy controller design [J]. Automatica, 1997, 33 (3): 1029 - 1039.
- [17] 张化光. 模糊双曲正切模型的建模方法与控制器设计 [J]. 自动化学报, 2000, 26 (6): 730 - 735.
- [18] Kosko B. Fuzzy systems as universal approximators [J]. IEEE Trans. on Computer, 1994, 43 (11): 1329 - 1333.
- [19] Buckley J J. Universal fuzzy controller [J]. Automatica, 1992, 28 (1): 16 - 24.
- [20] Buckley J J. Sugeno type controllers are universal controllers [J]. Fuzzy Sets and Systems, 1993, 53 (2): 299 - 303.
- [21] Buckley J J, Hayashi Y. Fuzzy input - output controllers are universal approximators [J]. Fuzzy Sets and Systems, 1993, 58 (3): 273 - 278.
- [22] 丁永生, 应浩. 模糊系统逼近理论: 现状与展望 [J]. 信息与控制, 2000, 29 (2): 157 - 163.
- [23] 廖龙涛, 李少远, 黄广斌. 规则可生长与修剪的非线性系统 T - S 模糊模型辨识 [J]. 自动化学报, 2007, 33 (10): 1097 - 1100.
- [24] 李佳宁, 易建强, 赵冬斌, 西广成. 一种新的基于神经模糊推理网络的复杂系统模糊辨识方法 [J]. 自动化学报, 2006, 32 (5): 695 - 703.
- [25] Wang X Z. Fuzzy rules generations from data for process operational decision support [J]. Computers and Chemical Engineering, 1997, 21 (6): 661 - 666.
- [26] 熊熊, 汪德馨, 宋铁民. 利用模糊神经网络进行数据挖掘的一种算法 [J]. 系统工程学报, 2000, 15

(1): 32 – 37.

- [27] Guillaume S. Designing fuzzy inference systems from data: an interpreability oriented review [J]. IEEE Transactions on Fuzzy Systems, 2001, 9 (3): 426 – 443.
- [28] Wang L X, Mendel J M. Generating fuzzy rules by learning from examples [J]. IEEE Transactions on Systems, Man, and Cybernetics, 1992, 22 (6): 1414 – 1427.
- [29] Wang L X. The WM method completed: a flexible fuzzy system approach to data mining [J]. IEEE Transactions on Fuzzy Systems, 2003, 11 (6): 768 – 782.
- [30] 王永富, 柴天佑. 从数据中挖掘模糊规则及其系统实现 [J]. 系统工程学报, 2005, 20 (5): 497 – 503.
- [31] 王永富, 王殿辉, 柴天佑. 一个具有完备性和鲁棒性的模糊规则提取算法 [J]. 自动化学报, 2010, 36 (9): 1337 – 1342.

第3章 摩擦模糊模型的规则提取与自适应控制补偿

3.1 现有摩擦模型的不足

摩擦是一种复杂的、非线性的、具有不确定性的自然现象。摩擦的存在降低了系统的性能，特别是对某些系统，如机械手、数控机床、转台以及坐标测量机而言，摩擦对这些系统性能的提高构成了严重的障碍。在工程实际中为了克服摩擦给机械系统带来的危害，人们从控制角度提出了一些补偿方法^[1-8]。总的来说，分为以下两大类：如果已知摩擦模型，便可以在控制系统中施加一个控制作用，使之抵消每一瞬时的摩擦力，这样就可以消除摩擦对系统性能的影响，这就是基于摩擦模型的补偿。不依赖于模型的补偿主要是基于高增益 PD 控制器补偿方法。

基于摩擦模型的补偿是人们对摩擦的物理过程给出一种近似的数学模型，通过摩擦实验数据辨识其参数。这种基于摩擦模型的补偿技术主要问题是^[9]：

- 1) 由于摩擦的内在机理还在探索与研究中，人们提出的多种摩擦模型没有形成统一的描述形式，在实际应用中很难确定哪种摩擦模型更合适。
- 2) 现有的摩擦模型只是对真实摩擦过程的一种近似描述，故在控制补偿时很难保证控制精度。
- 3) 现有这些摩擦模型从形式上看比较复杂，很难进行控制系统的稳定性分析。

另外，不依赖于模型的高增益 PD 补偿的主要问题是只能在一定程度上减小摩擦的影响，且鲁棒性与稳定性较差。

非线性系统的自适应模糊建模与控制的基本出发点是仿人的智能以实现对复杂不确定性系统进行有效的建模与控制，它具有从环境自学习、适应环境的能力^[10-14]。从智能建模与控制角度出发，研究摩擦的智能建模与控制补偿技术应是提高含摩擦环节的机电系统性能的一条值得探索的途径。我们知道，摩擦实验可以产生大量的摩擦实验数据，而这些实验数据为用智能建模（神经网络或模糊系统）提供了数据基础。本章将采用如下思路研究摩擦的智能建模与控制补偿问题：

- 1) 基于数据挖掘技术对摩擦实验数据进行模糊规则的挖掘。
- 2) 基于此模糊规则库建立摩擦的基础模糊模型（静态模型）。
- 3) 结合具体的对象与控制问题，从 Lyapunov 稳定性导出自适应参数，建立摩擦的自适应模糊模型（动态模型）并作为摩擦的控制补偿器。
- 4) 进行仿真实验对比分析。

3.2 数据挖掘方法建立摩擦的静态模糊模型

在工业过程建模领域中,对于那些非线性复杂系统,建立精确的数学模型非常困难。实际应用中,人们发现利用专家经验建立系统的模糊描述模型是一行之有效的方法,但模糊规则的获取是模糊建模的成功关键。模糊规则的获取一般有三种方法:

- 1) 专家语言产生规则。
- 2) 从样本数据中自动产生规则。
- 3) 样本数据结合专家语言产生混合规则。

近年来,从样本数据中自动产生模糊规则引起了人们的极大关注,其主要原因在于简单系统可以采用专家语言产生规则,但是当变量增多及变量的模糊集分割过多时,单纯依靠专家语言知识获取模糊规则是十分困难的。结果,梯度下降法、最小二乘法、遗传算法、模糊聚类 and 神经网络等方法相继被提出用于从样本数据中自动产生模糊规则^[15-16]。不幸的是,这些方法要求不断地反复学习或是产生规则的机理过于复杂,使得工程难以实施。

Wang 和 Mendel 提出了从数值数据中提取模糊规则的经典算法(简称为 WM 和 iWM 算法)。从第二章我们的分析知道 WM^[17]和 iWM^[18]算法缺乏良好的完备性和鲁棒性,我们提出了 DM 算法,采用 DM 方法产生的模糊规则库具有更好的完备性和鲁棒性。在这章我们将 DM 算法用于摩擦模糊规则库的提取,基于此库建立摩擦的静态模糊模型^[19]。

3.2.1 摩擦实验数据的采集

正电子发射型计算机断层显像(Positron emission computed tomography, PET)技术采用特殊的体外测量仪器探测这些正电子核素在人体全身各脏器的分布情况^[20],其中做衰减扫描时棒源进给装置如图 3-1 所示。

该装置在进给时存在摩擦力以及棒源的颤抖都将影响图像的质量。我们将采集该装置的摩擦力数据进行建模与控制补偿研究,该部分的系统采集与控制的简图如图 3-2 所示。



图 3-1 棒源进给装置

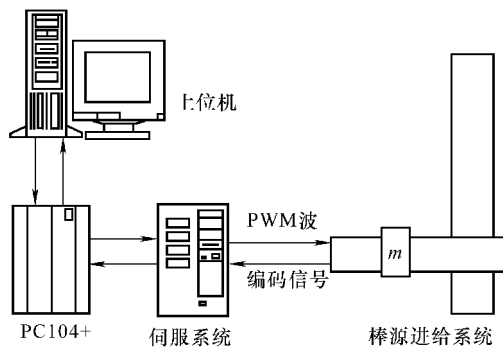


图 3-2 系统采集与控制的简图

在上面控制系统结构中:主机和 PC104+ 通过远程过程调用的 Socket 接口进行通信连接;PC104+ 向伺服系统传送控制指令;通过编码器反馈所需要的信号。在该实验中通过文

献 [1] 使用的 Breakaway 方法采集摩擦实验数据, 通过几次采集的结果如图 3-3 所示。

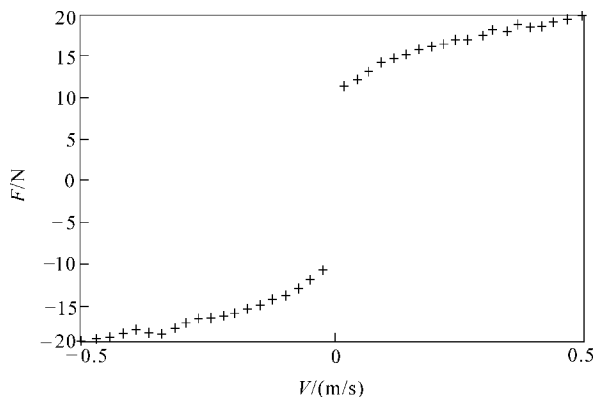


图 3-3 检测的摩擦力数据

3.2.2 摩擦模糊规则库的提取

下面给出用数据挖掘方法产生摩擦模糊规则库的主要步骤, 这种方法的优点保证了无论是用小样本数据还是大样本数据产生的摩擦模糊规则库都是完备的。另外, 这种方法保证了即使在样本数据中含少量坏数据, 产生的规则库仍能保证模糊模型的性能, 即规则库具有良好的鲁棒性。

对于给定的速度和摩擦力采样数据对

$$(\dot{x}^{(p)}, F^{(p)}), p = 1, 2, \dots, N \quad (3-1)$$

式中 $\dot{x} \in R$, 表示速度;

$F \in R$, 表示摩擦力。

设计任务是如何从采样数据对式 (3-1) 中提取模糊规则, 从而确定一个映射函数 $f: \dot{x} \rightarrow F$ 。设计的方法由下面几个步骤组成:

步骤 1. 输入输出变量的模糊区间划分

由前面的摩擦模型可知摩擦力是速度的函数, 即摩擦力随速度的变化而变化。故把速度作为模糊系统的前提变量, 而把摩擦力作为结论变量。设输入变量 $\dot{x}$ 取值区间取 $[\dot{x}^-, \dot{x}^+]$, 在此取值区间对 $\dot{x}$ 取模糊集合 $A(\dot{x}) = \{A_1^{\dot{x}}, \dots, A_{c_1}^{\dot{x}}\}$, 隶属度函数为高斯函数。另设输出变量 F 取值区间取 $[F^-, F^+]$, 在此取值区间对 F 取模糊集合 $B(F) = \{B_1^F, \dots, B_{c_0}^F\}$, 隶属度函数为三角型函数。

步骤 2. 普通行关系数据库转换为模糊型关系数据库

首先, 把 $\dot{x}$ 和 F 的采集数据存放到普通型关系数据库表中。然后把 $\dot{x}$ 和 F 的数据转换成隶属度函数值存放到模糊型关系数据库表中。在速度摩擦力数据对的普通型关系数据库中, 设 $L_1 = \{\dot{x}, F\}$ 表示属性集, 那么 t_p 表示下面 T_{L_1} 记录集合的第 p 个元组, 也就是

$$T_{L_1} = \{t_1, \dots, t_p, \dots, t_N\} \quad (3-2)$$

在速度摩擦力数据对的模糊型关系数据库中, 设 $L_2 = \{A(\dot{x}) \cup B(F)\}$ 表示属性集, $\mu(t_p)$ 表示 t_p 模糊化后的隶属度值, 那么 $\mu(t_p)$ 表示下面 T_{L_2} 记录集合的第 p 个元组, 也就是

$$T_{L_2} = \{\mu(t_1), \dots, \mu(t_p), \dots, \mu(t_N)\} \quad (3-3)$$

至此, 将 $T_{L_1} = \{t_1, \dots, t_p, \dots, t_N\}$ 一个普通型关系数据库转化为 $T_{L_1} = \{\mu(t_1), \dots, \mu(t_p), \dots, \mu(t_N)\}$ 模糊型关系数据库。为了便于理解上述过程, 一个简单的例子如表 3-1。

表 3-1 普通记录集合与模糊记录集合

T_{L_1}	$\dot{x}$	F	T_{L_2}	$A_1^{\dot{x}}$	$A_1^{\dot{x}}$	B_1^F	B_2^F
t_1	0.3	8	$\mu(t_1)$	0.3	0.6	1.0	0.1
t_2	0.1	2	$\mu(t_2)$	0.4	0.2	0.5	0.3
t_3	0.5	5	$\mu(t_3)$	0.3	0.1	0.7	0.1
t_4	0.2	15	$\mu(t_4)$	0.5	0.7	0.3	0.5
t_5	0.4	6	$\mu(t_5)$	0.6	0.9	0.0	0.2

步骤 3. 计算摩擦模糊规则的支持度

从数据挖掘的角度看, 如果一条模糊规则有实际意义, 它必须具有足够的支持度, 它反映了样本元组对该条规则的支持程度。现定义支持度如下

$$Supp(\dot{x} \Rightarrow F) = \frac{\sum_{p=1}^N \mu_{(B_0^F)^p}(F) \mu_{(A_1^{\dot{x}})^p}(\dot{x})}{\sum_{p=1}^N \mu_{(A_1^{\dot{x}})^p}(\dot{x})} \quad (3-4)$$

式中 $\mu_{(B_0^F)^p}(F)$ 和 $\mu_{(A_1^{\dot{x}})^p}(\dot{x})$ ——分别表示第 p 条记录上 F 和 $\dot{x}$ 所对应的隶属函数值;

N ——在 T_{L_2} 数据库的总记录数;

$l_0 \in \{1, \dots, c_0\}$, $l_1 \in \{1, \dots, c_1\}$ 。

为了简化计算, 也可以用下式计算支持度

$$Supp(\dot{x} \Rightarrow F) = \frac{1}{N} \sum_{p=1}^N \mu_{(B_0^F)^p}(F) \mu_{(A_1^{\dot{x}})^p}(\dot{x}) \quad (3-5)$$

由上面的支持度定义可以看出, 数据挖掘法所产生的某条模糊规则是所有空间样本数据 $T_{L_1} = \{t_1, \dots, t_p, \dots, t_N\}$ 选举出的结果。有理由相信数据挖掘方法所选举出的规则比单纯用空间样本部分数据 (如 WM 方法) 所选举出的规则更可靠、更可信。即使在样本数据中有个别坏数据, 也会被众多好数据进行有效的屏蔽, 由个别坏数据选举出坏规则的现象将在一定程度上得到抑制。

步骤 4. 创建完备的摩擦模糊规则库

首先, 算法要保证由前提变量构成的每个模糊子空间都能被遍历; 其次, 结论变量在模糊子空间选用哪个模糊集, 由规则的最大支持度决定。算法的主要步骤如第二章所示, 通过上面的遍历产生了完备的模糊规则库。如图 3-4 所示, 我们用一个简单例子说明此过程。设前提变量 $\dot{x}$ 和结论变量 F 的模糊集合分别为 $A = \{A_1^{\dot{x}}, \dots, A_5^{\dot{x}}\}$ 和 $B = \{B_1^F, \dots, B_5^F\}$ 。对于 $\{A_1^{\dot{x}}\}$, 我们首先分别计算 $\{A_1^{\dot{x}}, B_1^F\}$, $\{A_1^{\dot{x}}, B_2^F\}$, $\{A_1^{\dot{x}}, B_3^F\}$, $\{A_1^{\dot{x}}, B_4^F\}$ 和 $\{A_1^{\dot{x}}, B_5^F\}$ 的支持度, 然后结论变量 F 最终选择的模糊集由最大支持度决定。对于 $\{A_2^{\dot{x}}\}$, $\{A_3^{\dot{x}}\}$, $\{A_4^{\dot{x}}\}$ 和 $\{A_5^{\dot{x}}\}$ 重复上面的过程。由这些规则构成了所有模糊规则的一个子空间, 该子空间是最后选择的模糊规则库 (由图 3-4 中星号构成的空间)。

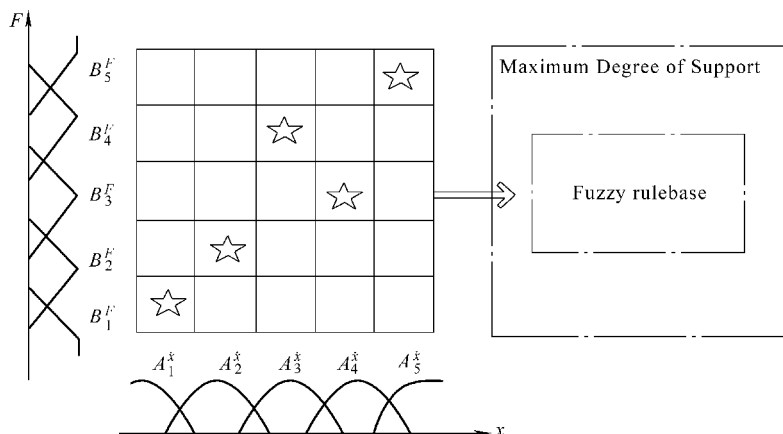


图 3-4 基于最大支持度决定的模糊规则子空间

3.2.3 建立摩擦的静态模糊模型

由上面最大支持度决定的摩擦模糊规则子空间

$$R^{(j)}: \text{If } \dot{x} \text{ is } A_{l_1}^x \text{ Then } F \text{ is } B_{l_0}^F \quad (3-6)$$

式中 $\dot{x} \in R$ 和 $F \in R$ ——分别表示模糊系统的输入和输出；

$A_{l_1}^x$ 和 $B_{l_0}^F$ ——分别是定义在输入和输出变量区间的模糊子集；

$l_0 \in \{1, \dots, c_0\}$ ；

$l_1 \in \{1, \dots, c_1\}$ ；

$j=1, \dots, M$ 。

在建立了上面的摩擦模糊语句后，采用单值模糊化、乘积运算、加权平均反模糊化可得到下面的摩擦静态模糊模型

$$\hat{F}(\dot{x}) = \frac{\sum_{j=1}^M \bar{F}^j \mu_{(A_{l_1}^x)^j}(\dot{x})}{\sum_{j=1}^M \mu_{(A_{l_1}^x)^j}(\dot{x})} \quad (3-7)$$

式中 $\bar{F}^j$ 是模糊隶属度函数取得最大值的点，也就是时 $\mu_{(B_{l_0}^F)^j}(F) = 1$ 时的点。

若定义下面的模糊基函数

$$\xi_j(\dot{x}) = \frac{\mu_{(A_{l_1}^x)^j}(\dot{x})}{\sum_{j=1}^M \mu_{(A_{l_1}^x)^j}(\dot{x})} \quad (3-8)$$

方程 (3-7) 可以改写成如下的形式

$$\hat{F}(\dot{x}) = \underline{\theta}^T \underline{\xi}(\dot{x}) \quad (3-9)$$

式中 $\underline{\theta} = (\bar{F}^1, \bar{F}^2, \dots, \bar{F}^M)^T$ ，表示模糊系统的参数向量；

$\underline{\xi}(\dot{x}) = (\xi_1(\dot{x}), \dots, \xi_M(\dot{x}))^T$ ，表示模糊基函数向量。

如果单纯从摩擦建模角度来看，模糊系统的参数向量 $\underline{\theta} = (\bar{F}^1, \bar{F}^2, \dots, \bar{F}^M)^T$ 一般取模糊隶属度函数 $\mu_{(B_{l_0}^F)^j}(F) = 1$ 最大值的点。但此时的参数向量不是自适应的，此时称为摩

擦的静态模糊模型。下面将结合 Lyapunov 稳定性进一步导出参数向量的自适应特性,也就是摩擦的动态模糊模型,从而用该自适应模糊模型在线逼近摩擦力。本章采用的基于数据挖掘和稳定性技术提取模糊规则与自适应调整的结构如图 3-5 所示。

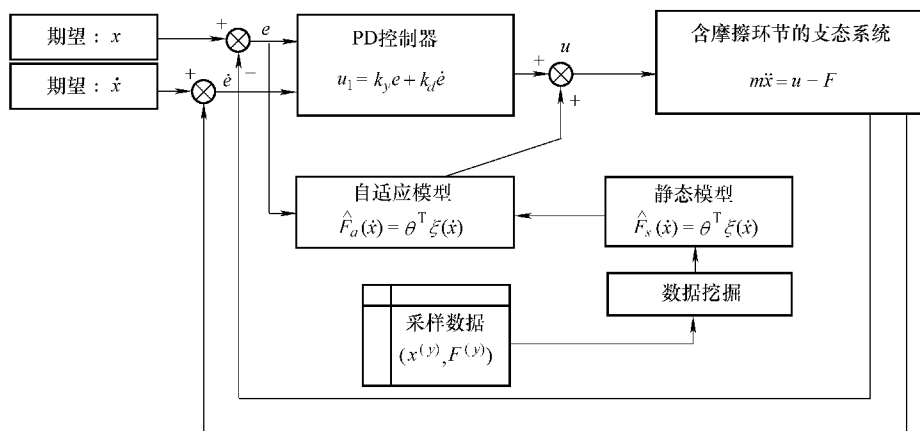


图 3-5 摩擦模糊规则提取与自适应机制结构图

3.3 摩擦模糊模型的自适应机制

为了便于问题的描述,考虑文献 [2] 所用的一维运动系统,系统运动方程如下

$$m\ddot{x} = u - F \quad (3-10)$$

式中 m ——运动体的质量;
 x ——运动体的输出位移;
 F ——摩擦力;
 u ——控制力。

若式 (3-10) 中的摩擦力已知,可采用基于摩擦模型补偿的 PD 经典控制器,也就是如下方程

$$u = k_p e + k_d \dot{e} + F \quad (3-11)$$

式中 $e = x_d - x$;
 $\dot{e} = \dot{x}_d - \dot{x}$;
 x_d ——已知的系统期望输出位移;
 k_p 和 k_d ——分别表示比例和微分系数。

把式 (3-11) 代入式 (3-10) 可得如下动态误差方程

$$m\ddot{e} + k_d \dot{e} + k_p e = m\ddot{x}_d \quad (3-12)$$

上面的控制器在运动系统中得到了广泛应用,它在定位调节控制中取得了十分满意的性能;在跟踪控制中能保证跟踪误差的有界性。

式 (3-12) 的结果是在假设摩擦力 F 已知情况下得到的。对摩擦项 F 来说,由于建立精确摩擦模型的高难性,得到精确控制算法 (3-11) 是不可能的。为此,拟用前文的模糊系统替代方程 (3-11) 中的摩擦项,可得如下方程

$$u = k_p e + k_d \dot{e} + \hat{F} \quad (3-13)$$

式中 $\hat{F}$ ——前文提到的模糊系统。

把式 (3-13) 代入式 (3-10) 可得如下动态误差方程

$$m\ddot{e} + k_d \dot{e} + k_p e = m\ddot{x}_d + F - \hat{F} \quad (3-14)$$

因 x_d 是有界的、已知的参考输入, 故可以把 $m\ddot{x}_d + F$ 看成是摩擦的广义函数, 式 (3-14) 等价如下方程

$$\ddot{e} + k_2 \dot{e} + k_1 e = m^{-1} [F_g - \hat{F}] \quad (3-15)$$

式中 $k_2 = m^{-1} k_d$;

$k_1 = m^{-1} k_p$;

$F_g = m\ddot{x}_d + F$ 。

下面用模糊系统对广义摩擦项进行逼近, 定义最优参数向量为

$$\underline{\theta}^* = \arg \min_{\underline{\theta} \in \Omega_\theta} (\sup_{\dot{x} \in \Omega_{\dot{x}}} \|\hat{F} - F_g\|) \quad (3-16)$$

式中 Ω_θ —— θ 的界。

那么最小逼近误差可表示如下方程

$$\varepsilon = F_g - \underline{\theta}^{*T} \underline{\xi}(\dot{x}) \quad (3-17)$$

由式 (3-9) 和式 (3-17) 知, 式 (3-15) 可写成

$$\ddot{e} + k_2 \dot{e} + k_1 e = m^{-1} [\tilde{\theta}^T \underline{\xi}(\dot{x}) + \varepsilon] \quad (3-18)$$

式中 $\tilde{\theta} = \underline{\theta}^* - \underline{\theta}$ 。

式 (3-18) 也等价如下形式

$$\dot{\underline{e}} = \underline{A} \underline{e} + \underline{B} \tilde{u} \quad (3-19)$$

式中 $\underline{A} = \begin{pmatrix} 0 & 1 \\ -k_1 & -k_2 \end{pmatrix}$;

$\underline{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$;

$\underline{e} = [e, \dot{e}]^T$;

$\tilde{u} = m^{-1} [\tilde{\theta}^T \underline{\xi}(\dot{x}) + \varepsilon]$ 。

定理 1: 如果采用式 (3-20) 和式 (3-21) 的主动控制器, 自适应参数由式 (3-22) 在线调节

$$u = k_p e + k_d \dot{e} + \hat{F} \quad (3-20)$$

$$\hat{F}(\dot{x}) = \underline{\theta}^T \underline{\xi}(\dot{x}) \quad (3-21)$$

$$\dot{\underline{\theta}} = \begin{cases} \gamma \underline{\xi}(\dot{x}) \underline{B}^T \underline{P} \underline{e}, & \text{if } (\|\underline{\theta}\| < M) \text{ or } (\|\underline{\theta}\| = M \text{ and } \underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\dot{x}) \underline{\theta} \leq 0) \\ \underline{P}[\cdot], & \text{if } (\|\underline{\theta}\| = M \text{ and } \underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\dot{x}) \underline{\theta} > 0) \end{cases} \quad (3-22)$$

式中 $\underline{P}[\cdot] = \gamma \underline{\xi}(\dot{x}) \underline{B}^T \underline{P} \underline{e} - \gamma \frac{\underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\dot{x}) \underline{\theta}}{\|\underline{\theta}\|^2} \underline{\theta}$, 是投影算子。

正定解 $P > 0$ 满足下面的李雅普诺夫方程 (3-23)

$$A^T P + P A = -Q \quad (3-23)$$

那么, 跟踪误差收敛到某一紧集。

证明: 取 Lyapunov 函数为

$$V = \frac{1}{2} \underline{e}^T P \underline{e} + \frac{m^{-1}}{2\gamma} (\tilde{\theta}^T \tilde{\theta}) \quad \gamma > 0 \quad (3-24)$$

求 V 的微分, 可推出

$$\dot{V} = \frac{1}{2} \underline{e}^T P \dot{\underline{e}} + \frac{1}{2} \dot{\underline{e}}^T P \underline{e} + \frac{m^{-1}}{\gamma} (\tilde{\theta}^T \dot{\tilde{\theta}}) \quad (3-25)$$

因为 $\dot{\tilde{\theta}} = -\dot{\theta}$, 并结合 (3-23) 可得

$$\dot{V} = \frac{1}{2} [\underline{e}^T (P A + A^T P) \underline{e}] + m^{-1} (\tilde{\theta}^T \xi(\dot{x}) + \varepsilon) B^T P \underline{e} - \frac{m^{-1}}{\gamma} (\tilde{\theta}^T \dot{\tilde{\theta}}) \quad (3-26)$$

由方程 (3-22) 和 (3-23), 上面方程对消一部分后可得到

$$\dot{V} = -\frac{1}{2} \underline{e}^T Q \underline{e} + m^{-1} \varepsilon B^T P \underline{e} \quad (3-27)$$

因此, 可以得到

$$\dot{V} \leq -\frac{1}{2} \lambda_{\min}(Q) \|\underline{e}\|^2 - 2m^{-1} \|\varepsilon_0\| \lambda_{\max}(P) \|\underline{e}\| \quad (3-28)$$

式中 ε_0 —— ε 的界。

从式 (3-28) 可以看出其是变号函数, 但只要方括号中的项大于零就可得 $\dot{V} < 0$, 即满足下式

$$\|\underline{e}\| > \frac{2 \|\varepsilon_0\| \lambda_{\max}(P)}{m \lambda_{\min}(Q)} \quad (3-29)$$

由标准 Lyapunov 理论可知, 只要误差项大于式 (3-29) 的右边项, 因 $\dot{V} < 0$, 那么跟踪误差就会减小。这也说明跟踪误差的有界性, 即

$$\|\underline{e}\| \leq \frac{2 \|\varepsilon_0\| \lambda_{\max}(P)}{m \lambda_{\min}(Q)} \quad (3-30)$$

式 (3-30) 的有界性说明闭环系统的跟踪误差最终进入某一个紧集。由式 (3-30) 还可以看出, 跟踪误差的界依赖于建模误差的界 ε_0 , 如果建模误差为零, 那么将取得零跟踪误差。所以说模糊系统对摩擦的逼近精度对系统的精度提高是至关重要的, 故本方案首先基于数据挖掘技术建立了摩擦的静态模型, 然后根据 Lyapunov 稳定性建立了摩擦的动态模型, 从而提高对摩擦的逼近精度。

3.4 仿真实验

把文中所采用的摩擦建模与控制补偿用于如下的运动系统进行仿真实验

$$m\ddot{x} = u - F \quad (3-31)$$

式中 m ——运动体的质量;
 $\dot{x}$ ——运动体的输出位移;
 F ——摩擦力;
 u ——控制力。

3.4.1 系统参数选择

$m=4/3\text{kg}$, $x(0)=(0.0, 0.0)^T$; 参考输入信号 $x_d=\sin(t)$ 和 $\dot{x}_d=\cos(t)$; $k_p=200$, $k_d=20$, $k_1=150$, $k_2=15$, $-40 < u < 40$ 。选择 $Q=\begin{pmatrix} 6000 & 1200 \\ 1200 & 260 \end{pmatrix}$, 解 Lyapunov 方程 (3-23) 可以得到 $P=\begin{pmatrix} 600 & 20 \\ 20 & 10 \end{pmatrix}$ 。

3.4.2 摩擦建模的模糊规则提取

步骤 1. 定义隶属度函数

对于输入变量 $\dot{x}$ 取如下 5 个高斯型函数: $\mu_{A_1\dot{x}}=1/(1+\exp(2.5(\dot{x}+0.37)))$, $\mu_{A_2\dot{x}}=\exp(-5.0(\dot{x}+0.25)^2)$, $\mu_{A_3\dot{x}}=\exp(-5.0(\dot{x})^2)$, $\mu_{A_4\dot{x}}=\exp(-5.0(\dot{x}-0.25)^2)$, $\mu_{A_5\dot{x}}=1/(1+\exp(-2.5(\dot{x}-0.37)))$ 。对于摩擦输出变量 F 取三角型隶属函数 (见图 3-4), 其参数为: $[-20 \quad -10]$, $[-20 \quad -10 \quad 0]$, $[-10 \quad 0 \quad 10]$, $[0 \quad 10 \quad 20]$, $[10 \quad 20]$ 。

步骤 2. 建立速度摩擦数据对的关系表

首先, 建立 $\dot{x}$ 和 F 的 10 个样本数据采集点 $t_1 \rightarrow t_{10}$ 的普通型关系表 (见表 3-2)。然后, 将 $\dot{x}$ 和 F 的 10 个样本数据采集点的普通型关系表转换成模糊型关系表 (见表 3-3)。

表 3-2 普通型关系表的样本数据

T_{L_1}	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
$\dot{x}$	-0.5	-0.4	-0.3	-0.2	-0.1	+0.1	+0.2	+0.3	+0.4	+0.5
F	-20	-18	-17	-16	-14	+14	+16	+17	+18	+20

注: 表 3-2 为节省篇幅, 实际应行列互换或转置。

表 3-3 模糊型关系表的样本数据

T_{L_2}	$A_1^{\dot{x}}$	$A_2^{\dot{x}}$	$A_3^{\dot{x}}$	$A_4^{\dot{x}}$	$A_5^{\dot{x}}$	B_1^F	B_2^F	B_3^F	B_4^F	B_5^F
$\mu(t_1)$	1.00	0.04	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.30
$\mu(t_2)$	0.61	0.32	0.00	0.00	0.00	0.80	0.20	0.00	0.00	0.00
$\mu(t_3)$	0.14	0.88	0.01	0.00	0.00	0.70	0.30	0.00	0.00	0.00
$\mu(t_4)$	0.01	0.88	0.14	0.00	0.00	0.60	0.40	0.00	0.00	0.00
$\mu(t_5)$	0.00	0.32	0.61	0.00	0.00	0.40	0.60	0.00	0.00	0.00
$\mu(t_6)$	0.00	0.00	0.61	0.32	0.00	0.00	0.00	0.00	0.60	0.40
$\mu(t_7)$	0.00	0.00	0.14	0.88	0.01	0.00	0.00	0.00	0.40	0.60
$\mu(t_8)$	0.00	0.00	0.01	0.88	0.14	0.00	0.00	0.00	0.30	0.70
$\mu(t_9)$	0.00	0.00	0.00	0.32	0.61	0.00	0.00	0.00	0.20	0.80
$\mu(t_{10})$	0.00	0.00	0.00	0.04	1.00	0.30	0.00	0.00	0.00	1.00

步骤 3. 摩擦模糊规则库的提取

按照前面的算法，基于最大支持度原则提取模糊规则库。由表 3-4 的计算的支持度结果可得如下模糊规则库

- ① If $\dot{x}$ is $A_1^{\dot{x}}$ Then F is B_1^F
- ② If $\dot{x}$ is $A_2^{\dot{x}}$ Then F is B_1^F
- ③ If $\dot{x}$ is $A_3^{\dot{x}}$ Then F is B_2^F or B_4^F
- ④ If $\dot{x}$ is $A_4^{\dot{x}}$ Then F is B_5^F
- ⑤ If $\dot{x}$ is $A_5^{\dot{x}}$ Then F is B_5^F

表 3-4 计算支持度

Supp	B_1^F	B_2^F	B_3^F	B_4^F	B_5^F
$A_1^{\dot{x}}$	0.16	0.02	0.00	0.00	0.03
$A_2^{\dot{x}}$	0.16	0.09	0.00	0.00	0.00
$A_3^{\dot{x}}$	0.03	0.04	0.00	0.04	0.03
$A_4^{\dot{x}}$	0.00	0.00	0.00	0.09	0.16
$A_5^{\dot{x}}$	0.03	0.00	0.00	0.02	0.16

注 1: 在上面的数据挖掘过程中，第三条规则出现了两个最大支持度相等的情况，我们采用下面的原则来处理：

- 1) 取两个模糊集合最大值平均值所对应的模糊集合。
- 2) 取距离两个模糊集合最大值平均值距离最近所对应的模糊集合。

在这里我们选择 1) 的方式，所以第三条模糊规则是：If $\dot{x}$ is $A_3^{\dot{x}}$ Then F is B_3^F 。另外，通过上面的数据挖掘过程得到参数向量取值为 $\underline{\theta} = (-20, -20, 0, 20, 20)^T$ 。

3.4.3 摩擦模糊模型的自适应控制补偿

在上面过程得到了参数 $\underline{\theta}$ 的初始值，但不具有自适应功能。在下面的仿真里，来动态调整自适应参数。取 $\gamma = 500$, $M = 25\sqrt{5}$, $S = \sum_{j=1}^5 \mu_{A_j^{\dot{x}}}(\dot{x})$, $\underline{\xi}(\dot{x}) = \frac{1}{S}(\mu_{A_1^{\dot{x}}}(\dot{x}), \dots, \mu_{A_5^{\dot{x}}}(\dot{x}))^T$ 。可得到下面的摩擦力模型和自适应机制

$$\hat{F}(\dot{x}) = \underline{\theta}^T \underline{\xi}(\dot{x}) \quad (3-32)$$

$$\dot{\underline{\theta}} = \begin{cases} \gamma \underline{\xi}(\dot{x})(20e + 10\dot{e}), & \text{if } (\|\underline{\theta}\| < M) \text{ or } (\|\underline{\theta}\| = M \text{ and } e^T \underline{P} \underline{B} \underline{\xi}^T(\dot{x}) \underline{\theta} \leq 0) \\ P[\cdot], & \text{if } (\|\underline{\theta}\| = M \text{ and } e^T \underline{P} \underline{B} \underline{\xi}^T(\dot{x}) \underline{\theta} > 0) \end{cases} \quad (3-33)$$

式中 $P[\cdot] = \gamma \underline{\xi}(\dot{x})(20e + 10\dot{e}) - \gamma(20e + 10\dot{e}) \frac{\underline{\xi}^T(\dot{x}) \underline{\theta}}{\|\underline{\theta}\|^2} \underline{\theta}$ ，是投影算子。

注 2: 从现有文献看，自适应参数的界都是在 $\|\underline{\theta}\| \leq M$ 情况下给定的，这对于连续系统是正确。但计算机数值计算 (ODE45) 完全有可能使 $\|\underline{\theta}\| > M$ 的情况发生，故在数值计算时应考虑这种情况。方程 (3-33) 中的 $\|\underline{\theta}\| = M$ 情况应改成 $\|\underline{\theta}\| \geq M$ ，其他保持不

变。在本文 M 的界取为 $25\sqrt{5}$ ，仿真时把自适应参数拓展为状态空间的增广。

3.4.4 仿真结果与分析

图 3-6a 和图 3-6b 分别是假设系统中无摩擦力和有摩擦力情况下获得的，在这组仿真里，控制器都取 $u = k_p e + k_d \dot{e}$ ，其中： $k_p = 200$ ， $k_d = 20$ 。这组仿真说明，系统存在摩擦和不存在摩擦对控制效果的影响是不同的，所以在控制器设计时应充分考虑摩擦的影响，否则控制精度变低。

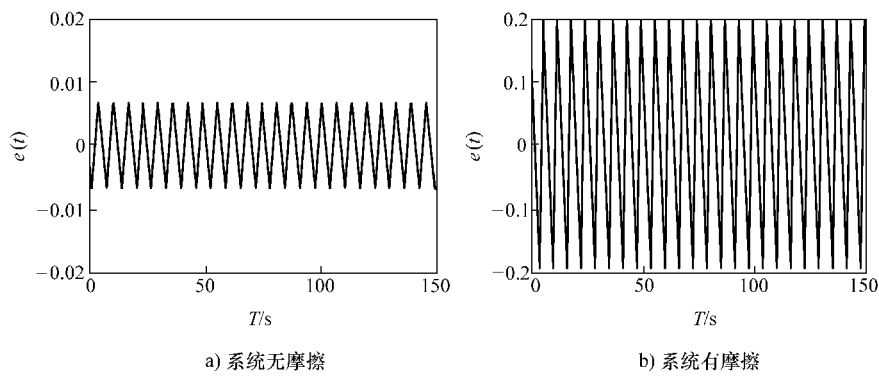


图 3-6 基于相同控制器的误差 $e(t)$ 的跟踪曲线

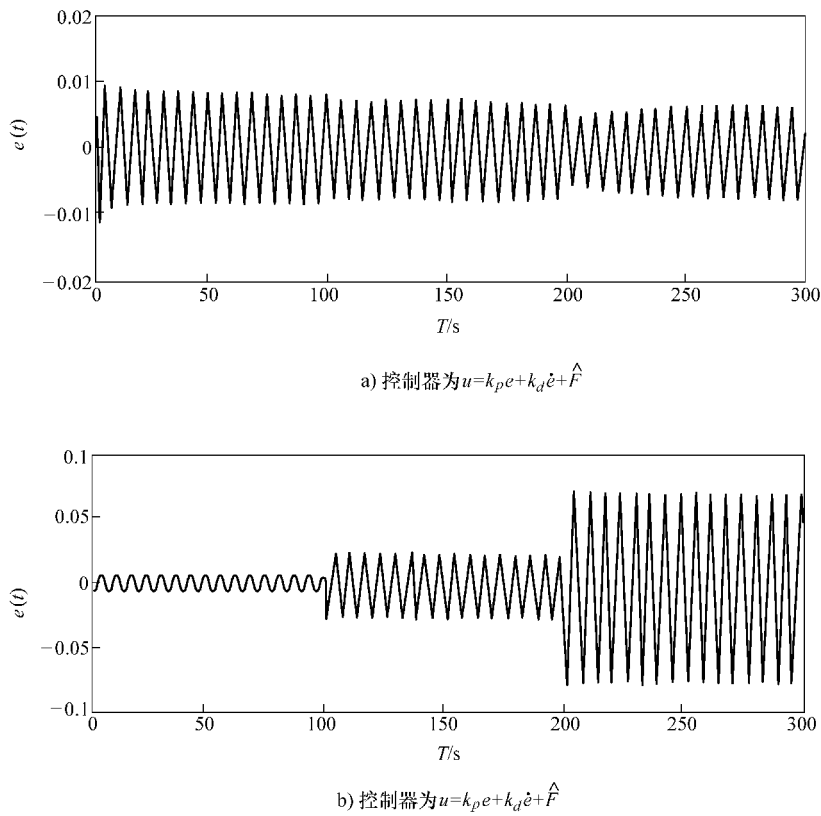


图 3-7 基于不同控制器的误差 $e(t)$ 的跟踪曲线

在接下来的仿真里，我们验证基于数据挖掘的摩擦模糊建模的自适应特性。设方程(3-31)的摩擦力是随速度和环境在变化的，并假设机械系统的初始摩擦因数较大，随系统的磨合摩擦因数逐渐变小。这一物理现象在仿真过程如下：设方程(3-31)中的初始摩擦力为 $F = 40\dot{x}$ ， $0 \leq t < 100$ ；经过 100s 后变成 $F = 35\dot{x}$ ， $100 \leq t < 200$ ；最后阶段取 $F = 25\dot{x}$ ， $200 \leq t \leq 300$ 。

图 3-7a 是系统跟踪误差 e 的时间响应曲线，该结果是控制取 $u = k_p e + k_d \dot{e} + \hat{F}$ 时获得的。而图 3-7b 是跟踪误差 e 的时间响应曲线，该结果是控制取 $u = k_p e + k_d \dot{e} + F$ 时获得的，其中： $F = 40\dot{x}$ ，在 $0 \leq t < 300$ 时间段内。图 3-7b 说明如果用传统的摩擦力模型进行控制补偿，控制器中的摩擦力如果不随实际摩擦力的改变而改变，该控制器很难取得良好的跟踪效果。图 3-7a 表明在用基于数据挖掘的摩擦模糊建模的自适应控制补偿时，比传统的摩擦力模型进行控制补偿取得了更好的效果，因为该摩擦模糊模型进行了自适应调整。类似，图 3-8 是跟踪误差 $\dot{e}$ 对应的的时间响应曲线。图 3-9 对应的是五个自适应参数随时间变化的响应曲线。从这个仿真实验可以看出，采用数据挖掘、模糊建模和自适应技术对不确定摩擦进行建模与控制补偿的有效性，明显比传统的摩擦模型控制补偿效果要好。

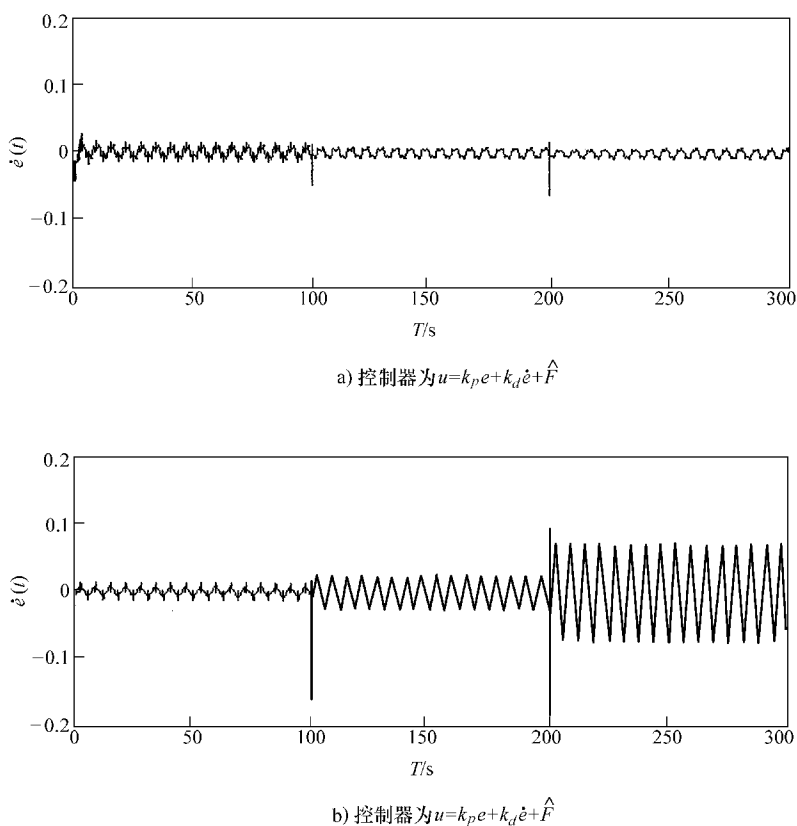


图 3-8 基于不同控制器的误差 $\dot{e}(t)$ 的跟踪曲线

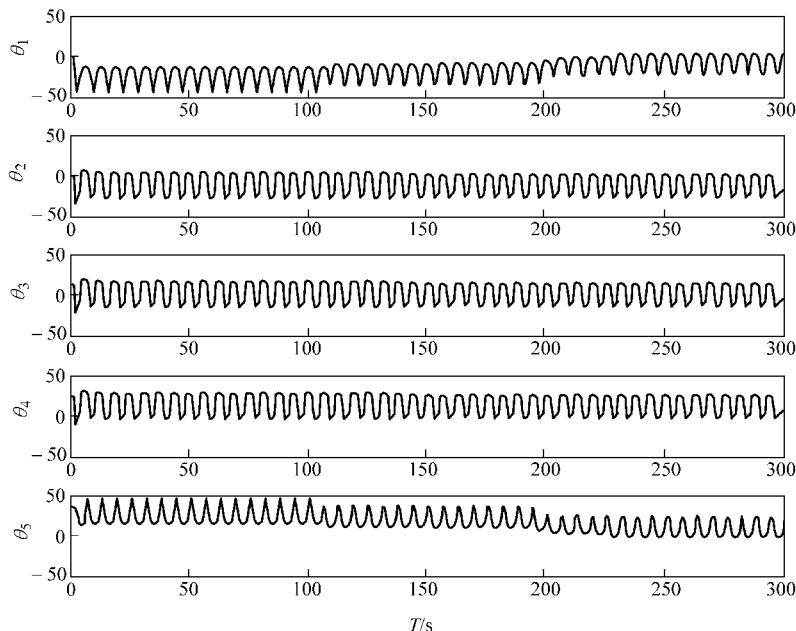


图 3-9 五个自适应参数随时间响应的曲线

3.5 本章小结

本章对机械系统的不确定摩擦力采用了模糊系统进行在线逼近，并形成了基于模糊系统前馈补偿摩擦的 PD 控制器。其中该模糊系统的辨识分为两个步骤：首先采用数据挖掘技术辨识出模糊系统的静态模型，然后基于 Lyapunov 稳定性进一步辨识出模糊系统的动态模型。理论分析和仿真实验说明采用该控制系统不但可以获得较好的跟踪性能，而且具有良好的稳定性。也充分说明了自适应模糊建模与控制技术对不确定性机械系统进行有效的建模与控制补偿作用，它具有从环境自学习、自适应环境的能力。

参 考 文 献

- [1] B. Armstrong and P. C. D. Canudas DeWit. A survey of models, analysis tools and compensation methods for control of machines with friction [J]. Automatica, 1994, 30 (7): 1083 – 1138.
- [2] P. C. D. Canudas De Wit, H. Ollson, K. J. Astrom, and P. Lischinsky. A new model for control of systems with friction [J]. IEEE Trans. Autom. Control, 1995, 40 (3): 419 – 425.
- [3] J. Swevers, F. Al – Bender, C. G. Ganseman, and T. Prajogo. An integrated friction model structure with improved presliding behaviour for accurate friction compensation [J]. IEEE Trans. Autom. Control, 2000, 45 (4): 675 – 686.
- [4] H. G. Kwatny Teolis. Variable structure control of systems with nonlinear friction [J]. Automatica, 2002, 38 (7): 1251 – 1256.
- [5] S. J. Kim and I. J. Ha. An efficient identification method for friction in single – DOF motion control systems [J]. IEEE Trans. Control Syst. Technol., 2004, 12 (4): 555 – 563.

- [6] P. Dupont, V. Hayward, B. Armstrong, and F. Altpeter. Single state elasto – plastic friction models [J]. IEEE Trans. Autom. Control, 2002, 47 (5): 787 – 792.
- [7] Y. Guo and Z. H. Qu. Control of frictional dynamics of a one dimensional particle array [J]. Automatica, 2008, 44 (10): 2560 – 2569.
- [8] D. D. Rigos and S. D. Fassois. Friction identification based upon the LuGre and Maxwell slip models [J]. IEEE Trans. Control Syst. Technol. , 2009, 17 (1): 153 – 160.
- [9] Y. F. Wang, D. H. Wang, and T. Y. Chai. Modeling and control compensation of nonlinear friction using adaptive fuzzy systems [J]. Mech. Syst. Signal Process. , 2009, 103 (8): 2445 – 2457.
- [10] L. X. Wang. Stable adaptive fuzzy control of nonlinear systems [J]. IEEE Trans. Fuzzy Syst. , 1993, 1 (1): 146 – 155.
- [11] C. Y. Sue and Y. Stepanenko. Adaptive control of a class of nonlinear systems with fuzzy logic [J]. IEEE Trans. Fuzzy Syst. , 1994, 2 (1): 285 – 294.
- [12] B. S. Chen, C. H. Lee, and Y. C. Chang. H_{∞} tracking design of uncertain nonlinear SISO systems: Adaptive fuzzy approach [J]. IEEE Trans. Fuzzy Syst. , 1996, 4 (1): 32 – 43.
- [13] T. Y. Chai and S. C. Tong. Fuzzy control for a class of nonlinear systems [J]. Fuzzy Sets Syst. , 1999, 103 (3): 379 – 387.
- [14] S. C. Tong and H. X. Li. Fuzzy adaptive sliding – mode control for MIMO nonlinear systems [J]. IEEE Trans. Fuzzy Syst. , 2003, 11 (3): 354 – 360.
- [15] Serge G. Designing fuzzy inference systems from data: an interpretability oriented review [J]. IEEE Trans. Fuzzy Syst. 2001, 9 (3): 426 – 443.
- [16] 熊熊, 汪德馨, 1083 – 1138 宋轶民. 利用模糊神经网络进行数据挖掘的一种算法 [J]. 系统工程学报, 2000, 15 (1): 32 – 37.
- [17] L. X. Wang and J. M. Mendel. Generating fuzzy rules by learning from examples [J]. IEEE Trans. Syst. , Man, Cybern. , 1992, 22 (6): 1414 – 1427.
- [18] L. X. Wang. The WM method completed: A flexible fuzzy system approach to data mining [J]. IEEE Trans. Fuzzy Syst. , 2003, 11 (6): 768 – 782.
- [19] Yongfu Wang, Dianhui Wang, Tianyou Chai. Extraction and adaptation of fuzzy rules for friction modeling and control compensation [J]. IEEE Trans. On Fuzzy Systems, 2011, 19 (4): 682 – 693.
- [20] 王永富, 李甲递, 赵宏, 等. 基于实时 Linux 的 PET 控制软件设计与实现 [J]. 东北大学学报, 2008, 29 (4): 497 – 500.

第 4 章 基于两种状态估计的摩擦控制补偿对比

4.1 问题描述

在过去的 10 多年里,关于机器人手臂的运动控制已取得很多成果,人们提出了各种各样的控制方法,包括不依赖机器人动力学模型的 PD 控制、模糊控制,以及依赖机器人动力学模型的自适应控制、鲁棒控制等^[1-4],其中 PD 控制的简单与实用性使其在实际中得到了广泛的应用。但是,在现有这些方法中机器人控制器设计一般都是基于完全状态反馈,即它假设机器人的角位置和角速度都是可测的。

机器人实际过程控制中,机器人的角位置可以通过编码器精确得到,但对机器人角速度的处理一般有三种方法:

- 1) 通过硬件测速仪得到,但是这种方法增加了成本,同时被检测的信号易被噪声污染,故很少采用。
- 2) 通过检测到的角位置采用数字差分方法求出角速度。
- 3) 设计状态观测器估计角速度。

但采用数字差分和观测器哪种方法使角速度估计和控制精度更理想,从理论、仿真和实验等角度进行对比研究具有一定意义,关于这方面的对比在国内外文献很少见到。

在国内外文献上机器人的角速度估计采用数字差分方法主要包括 LPP、RT、TSE、BDE、LSF 等估计器。而机器人观测器设计方法基本上有两类:基于机器人动力学模型^[5-7]和不依赖机器人动力学模型^[7-9]。基于机器人动力学模型方法是假设机器人动力学模型是完全或部分知道,如变结构观测器,自适应观测器等。不依赖机器人动力学模型的观测器是不要求机器人模型是已知的,而只是通过角位置和控制量构造观测器,其中比较流行的是高增益观测器。下面将采用 LPP 差分和高增益观测器方法分别估计角速度,控制器采用了基于摩擦和重力补偿的 PD 算法,并从仿真角度对角速度估计误差和稳态误差进行了对比研究^[10]。

4.2 机器人模型

为了分析 LPP 差分和高增益观测器方法分别估计角速度的精度问题,和验证基于摩擦和重力补偿的 PD 算法的有效性,在本章考虑下面简单的单输入单输出系统(见图 4-1)。

$$J \frac{d\dot{q}}{dt} + \tau_f + \tau_g = \tau \quad (4-1)$$

式中 $\dot{q} \in R$, 机械手关节的角位置、角速度和角加速度;

$\tau \in R$, 是执行力矩;

J ——机械手的转动惯量;

τ_g ——重力项;

τ_f ——摩擦力。

对于系统只有机械手的角位置 θ 是可测的。

设计任务: 对比两种软测量方法所形成的 PD 输出反馈控制在机器人定位调节问题上的控制性能, 以及角速度的估计性能。

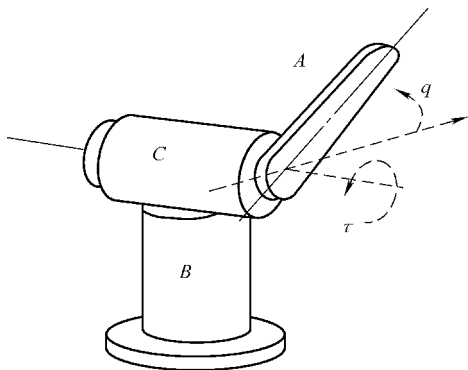


图 4-1 机器人的模型图

4.3 PD 输出反馈控制

在对方程 (4-1) 设计输出反馈控制器之前, 先给出方程 (4-1) 的如下状态空间表示形式

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(x_1, x_2) + g(x_1)u \\ y = x_1 \end{cases} \quad (4-2)$$

式中 $x_1 = q \in R$, 机器人的角位置;

$x_2 = \dot{q} \in R$, 机器人的角速度;

$y \in R$, 机器人可测的角位置向量;

$f(x_1, x_2) = -J^{-1}(\tau_f + \tau_g)$;

$g(x_1) = J^{-1}$;

$\tau = u$ 。

我们首先假设摩擦力项是已知的, 只是速度信号是未知的。首先判断基于差分方法的 PD 控制和基于高增益观测器的 PD 控制的性能对比问题, 为第 5 章的基于自适应模糊观测器的输出反馈奠定基础。两种方法的 PD 输出反馈控制结构图如图 4-2 所示。

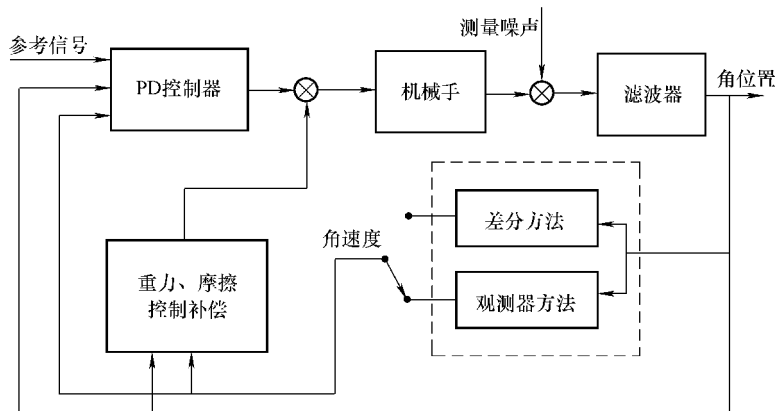


图 4-2 基于状态估计的机器人控制模型图

4.3.1 基于高增益观测器的 PD 输出反馈控制

对于非线性机械手系统 (4-1) 或 (4-2) 的高增益状态观测器为

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + \frac{1}{\varepsilon} H_p (y - \hat{x}_1) \\ \dot{\hat{x}}_2 = \frac{1}{\varepsilon^2} H_v (y - \hat{x}_1) \end{cases} \quad (4-3)$$

式中 $\hat{x}_1 \in R$, $\hat{x}_2 \in R$ 分别表示为 x_1 , x_2 的估计;

ε ——给定的较小的正常数;

H_p , H_v ——正定增益系数。

对于方程 (4-2) 采用如下带补偿的 PD 输出反馈控制器, 其中角速度估计采用方程 (4-3) 的高增益观测器

$$\begin{cases} \mathbf{u} = \boldsymbol{\tau}_f + \boldsymbol{\tau}_g + k_p (\mathbf{x}_1^d - \mathbf{x}_1) + k_d (\mathbf{x}_2^d - \hat{\mathbf{x}}_2) \\ \boldsymbol{\tau}_f = f\hat{\mathbf{x}}_2 \end{cases} \quad (4-4)$$

式中 $\mathbf{x}_1^d \in R$, 参考角位置向量;

$\mathbf{x}_2^d \in R$, 参考角速度向量;

k_p , $k_d \in R$, 比例和微分系数, 假定参考角位置向量及其导数是已知且有界的。

$\boldsymbol{\tau}_f + \boldsymbol{\tau}_g$ 是考虑重力和摩擦项的补偿控制项。在机器人实际过程控制中, 如果机器人模型已知可直接采用该补偿控制项; 如果机器人模型未知可采用第3章的自适应模糊系统逼近该项。我们在本章的主要任务是从仿真角度考察两种不同速度估计方法对机器人性能的不同影响, 故假设机器人模型已知而直接采用该控制补偿项。

4.3.2 基于数字差分的 PD 输出反馈控制

对于非线性机械手系统式 (4-1) 或式 (4-2) 的角速度估计也可采用最简单的差分方法 LPP 作为估计器

$$\hat{x}_2(k) = \frac{x_1(k) - x_1(k-1)}{T} = \frac{\Delta x_1(k)}{T} \quad (4-5)$$

式中 T ——采样周期。

对于方程 (4-2) 采用如下带补偿的 PD 输出反馈控制器, 其中角速度估计采用方程 (4-5) 的 LPP 估计器

$$\begin{cases} \mathbf{u} = \boldsymbol{\tau}_f + \boldsymbol{\tau}_g + k_p (\mathbf{x}_1^d - \mathbf{x}_1) + k_d (\mathbf{x}_2^d - \hat{\mathbf{x}}_2) \\ \boldsymbol{\tau}_f = f\hat{\mathbf{x}}_2 \end{cases} \quad (4-6)$$

式中 $\mathbf{x}_1^d \in R$, 参考角位置向量;

$\mathbf{x}_2^d \in R$, 参考角速度向量;

k_p , $k_d \in R$, 比例和微分系数, 假定参考角位置向量及其导数是已知且有界的。

4.3.3 两种方法的仿真对比

把本文所采用的角速度估计方法及 PD 控制器用于单自由度机械手系统的仿真, 该机械

手（见图 4-1）的动态模型为

$$\begin{cases} M\ddot{q} + f\dot{q} + \frac{1}{2}mgl\sin(q) = u \\ y = q \end{cases} \quad (4-7)$$

式中 $q, \dot{q}, \ddot{q} \in R$ ，分别为机械手臂的角位置，角速度和角加速度；

l ——连杆长度；

g ——重力加速度；

m ——杆的质量；

f ——摩擦系数；

M ——运动惯量；

u ——驱动力矩。

设 $x_1 = q, x_2 = \dot{q}$ ，那么上式能被改写成如下状态方程

$$\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \left(\frac{u - fx_2 - 0.5mgl\sin x_1}{M} \right) \\ y = x_1 \end{cases} \quad (4-8)$$

系统参数选择： $m=1, l=1, M=2, f=0.6, x(0)=[0 \ 0]^T, \hat{x}(0)=[0 \ 0]^T, x_1^d=1,$

$x_2^d=0, H_p=0.03, H_v=0.03, H_v=0.06, k_p=160, k_d=10, T=0.001, \varepsilon=0.001$ ，控制器的输出限幅为 $[-10 \ +10]$ 。

用高增益状态观测器式（4-3）和控制器式（4-4）构成了系统的基于高增益状态观测器的输出反馈控制；而差分方法式（4-5）和控制器式（4-6）构成系统的基于差分方法的输出反馈控制。下面将从仿真角度对比这两种输出反馈控制的效果。

从图 4-3 可以看出，在系统动态调节过程，基于高增益观测器方法形成的定位调节控制问题要好于数字差分方法。也就是用高增益观测器方法形成的定位调节（图 4-3a）经过短暂的动态过程达到了期望的位置，而差分方法的动态过程明显要长些。而图 4-4 是角速度的响应曲线，从该图也可以看出基于高增益观测器方法形成的定位调节控制问题要好于数字差

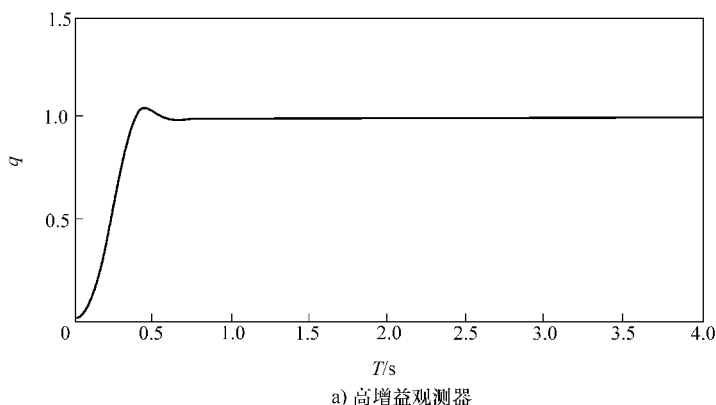
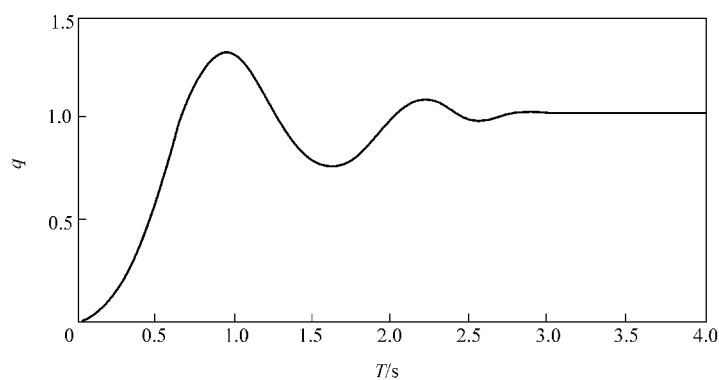


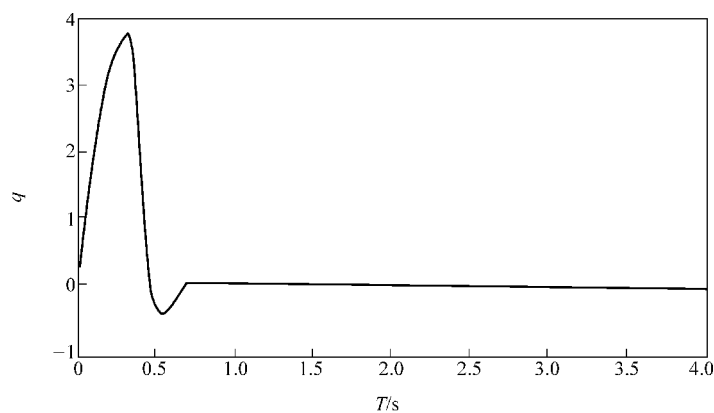
图 4-3 基于状态估计的机器人角位置 q 的输出曲线



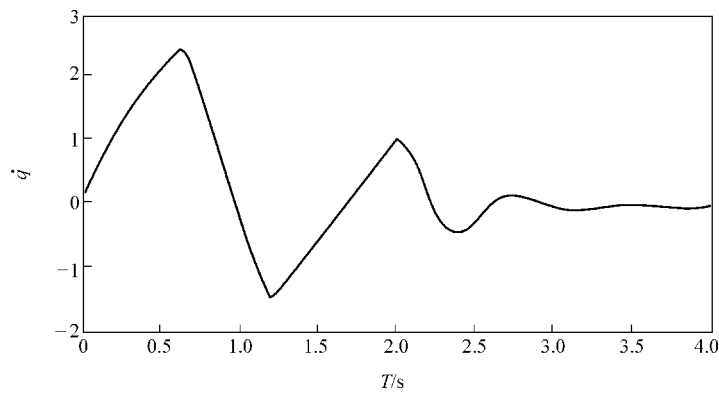
b) 差分方法

图 4-3 基于状态估计的机器人角位置 q 的输出曲线 (续)

分方法。通过上面的仿真，可知基于观测器方法的摩擦补偿模型与输出反馈控制取得了良好的控制效果。



a) 高增益观测器



b) 差分方法

图 4-4 基于状态估计的机器人角位置 q 的输出曲线

4.4 本章小结

在机器人的控制器和摩擦模型中要用到速度信号, 实际中速度信号常常是不可检测的。本章采用数字差分和高增益观测器方法分别对机械人的角速度估计及形成的定位调节问题进行了对比。从仿真可以看出基于高增益观测器的方法明显优于基于差分方法的摩擦建模与控制器设计, 所以设计良好的观测器对系统状态不完全可测的摩擦建模与控制补偿具有重要意义。本章的进一步工作是从理论分析上给出基于观测器方法优于基于差分方法的摩擦建模与控制器设计的理论证明。

参 考 文 献

- [1] 申铁龙. 机器人鲁棒控制基础 [M]. 北京: 清华大学出版社, 2000.
- [2] 顾海军, 张天平, 曹永忠. 基于观测器的机器人鲁棒跟踪控制 [J]. 控制工程, 2003, 10 (6): 568 - 570.
- [3] 周云龙, 项龙江, 徐心和. 双足机器人的滑模控制 [J]. 东北大学学报 (自然科学版), 2003, 24 (9): 892 - 895.
- [4] Jose A R, Rafael K, Ilse C. Semi2global stability of saturated linear PID control for robot manipulators [J]. Automatica, 2003, 39 (5): 990 - 995.
- [5] Canudas C W, Slotine J E. Sliding observers for robot manipulator [J]. Automatica, 1991, 27 (5): 859 - 864.
- [6] Berghuis H, Nijmeijer H. A passivity approach to controller - observer design for robots [J]. IEEE Tran on Robot Automat, 1993, 9 (3): 740 - 754.
- [7] Nicosia S, Tornambe A. High - gain observers in the state and parameter estimation of robots having elastic joints [J]. System Control Letter, 1989, 13 (2): 331 - 337.
- [8] Jose A, Wen Y. A high - gain observer - based PD control for robots manipulator [A]. Proceedings of the American Control Conference [C]. Chicago, 2000, 2518 - 2522.
- [9] Kim Y H, Lewis F L, Abdallah C T. Dynamic recurrent neural network2based adaptive observer for a class of nonlinear systems [J]. Automatica, 1997, 33 (8): 1539 - 1543.
- [10] 王永富, 柴天佑. 机器人角速度两种软测量方法的 PD 控制对比 [J]. 东北大学学报 (自然科学版), 2004, 25 (6): 515 - 518.

第 5 章 基于模糊观测器的多摩擦环节建模与控制补偿

5.1 问题描述

目前,无论是用传统方法还是智能方法,摩擦建模与控制补偿几乎都是针对含单摩擦环节的机械系统进行的,或者说是针对单输入单输出系统。但实际机械系统,一般具有多个输入量或输出量的系统。同单变量系统相比,多变量系统的控制复杂得多。在多变量控制系统中,被控对象、测量元件、控制器和执行元件都可能具有一个以上的输入变量或一个以上的输出变量。本章将探讨含多摩擦环节的多输入多输出机械系统的建模与控制补偿问题。

实际研究表明摩擦力是速度与位置的函数,因此为了获取摩擦力大小,必须同时获得速度和位置的值。然而,在实际工程应用中,一般机械系统的位置通过编码器可以直接检测到,由于成本和噪声等原因使得速度很少直接检测。因此,研究有效的状态估计技术对摩擦建模与反馈控制具有一定的理论和实际意义。在第 4 章已验证基于高增益观测器技术设计摩擦估计模型和控制补偿器的有效性问题。本章将针对速度不可测的含多摩擦环节的多输入多输出机械系统,给出基于模糊观测器的摩擦模糊建模与自适应控制器设计。本章提出的方法适用于多自由度机械系统的建模与控制,对于含多摩擦环节的不完全可测机械系统具有普遍意义。

5.2 模糊观测器现状

非线性系统的状态观测器是指基于非线性系统的输入、输出观测数据估计其不可测的内部状态。非线性状态观测器在科学研究和工程应用领域具有重要价值,如工业过程中的状态反馈控制、航空制导系统、飞行轨迹重构、飞行目标跟踪、环境监控、故障诊断、生化反应的状态提取等。从对象模型已知和未知角度来分,观测器设计方法基本上有两类:已知对象模型^[1-3]和未知对象模型^[4-6]。已知对象模型是对象模型的结构和参数是完全或部分知道,观测器的构造依赖于对象模型和输入、输出观测数据,如经典的线性 Luenberger 观测器以及 Marino 等人提出的非线性广义 Luenberger 自适应观测器等。未知对象模型方法设计观测器是不需要对象模型是已知的,而只是通过输入、输出观测数据构造观测器,其中比较流行的是高增益观测器。非线性系统的广义 Luenberger 自适应观测器要求被观测对象模型结构已知的条件,我们知道在实际系统中的对象模型结构很难确定,故非线性系统的广义 Luenberger 自适应观测器应用到实际中有一定难度。

近年来,用模糊系统研究结构和参数均未知的非线性系统的自适应控制得到了重视,以

Lyapunov 稳定性理论为基础并结合监督控制、 H_∞ 控制和滑模控制等方法得到了广泛的研究与应用^[7-13]。上述成果中, 自适应模糊控制算法都利用了系统状态是完全可测的条件。而在很多实际系统中的状态变量并不完全可测, 如机器人系统中机器人的角位置可以通过编码器精确得到, 但对机器人角速度的处理一般很少用硬件测速仪直接检测, 因为这种方法增加了成本同时被检测的信号易被噪声污染。对机器人角速度的处理一般采用估计方法:

1) 通过检测到的角位置采用数字差分方法求出角速度。

2) 设计状态观测器估计角速度。在对机器人位置定位调节及跟踪要求较精确的情况下, 采用差分方法所得到的角速度及跟踪效果并不十分理想, 近些年采用观测器方法得到了重视^[14]。

近几年来, 国内外一些学者研究了基于观测器的自适应模糊控制问题, 我们知道应用最多的模糊系统是基于 T-S 模型的模糊系统和基于基函数模型的模糊系统。无论是基于 T-S 模型的模糊观测器还是基于基函数模型的模糊观测器, 它们都不需要被控对象模型的结构和参数已知的条件, 而只是通过输入、输出观测数据构造模糊观测器, 所以它们本质上都是不依赖被控对象模型的。基于基函数模型构造模糊观测器的自适应输出反馈控制只是在近几年得到了关注, 相对来说它还不成熟、问题比较多。

基于基函数模型构造模糊观测器的自适应输出反馈控制在近几年的一些文献中得到了研究, 下面简要分析这些文献中存在的问题及需要改进的地方。

在文献 [15] 中, 观测器方程为下面的式 (5-1)、控制器方程中的 u_s 监督补偿控制项及自适应参数 $\hat{\theta}$ 中都是用误差 $\tilde{e}$ 来调节的, 也就是下列方程

$$\begin{cases} \dot{\hat{e}} = A \hat{e} - B K_c^T \hat{e} + K_o (e_1 - \hat{e}_1) \\ \hat{e}_1 = C^T \hat{e} \end{cases} \quad (5-1)$$

$$u = u_D(\hat{x} | \hat{\theta}) + u_S(\hat{x}) \quad (5-2)$$

$$u_D(\hat{x} | \hat{\theta}) = \hat{\theta}^T \xi(\hat{x}) \quad (5-3)$$

$$u_S(\hat{x}) = I^* \operatorname{sgn}(\tilde{e}^T P B) \frac{1}{g_L(\hat{x})} \left\{ |g^U(\hat{x}) u_D| + \frac{g^U(\hat{x})}{g_L(\hat{x})} [f^U(\hat{x}) + |y_r^{(n)}| + |K_c^T|] + d_m \right\} \quad (5-4)$$

$$\dot{\hat{\theta}} = \gamma \tilde{e}^T P B \xi(\hat{x}) \quad (5-5)$$

式中 $\tilde{e} = e - \hat{e}$;

$$e = y_r - x = [e, \dot{e}, \dots, e^{(n-1)}]。$$

可见在控制器又用到了全部状态, 而实际上 x_1 可用 (因为 $y = x_1$), 而 x_2 等是不可用的, 所以说此控制器还是基于完全状态反馈的, 关于该文更详细的评论参见文献 [16]。

文献 [17] 针对单变量系统给出了模糊观测器和自适应控制器, 但该文还存在以下几个问题需要改进, 该文在应用严格正实得到方程

$$\begin{cases} A_c^T P + P A_c = -Q \\ P B_c = C_c \end{cases} \quad (5-6)$$

要求 $\mathbf{P} = \mathbf{P}^T > 0$ 和 $\mathbf{B}_c = [0 \ \cdots \ b_m]^T$ 条件, 上面方程是无解的。

该文在仿真中只对角位置 x_1 进行了状态估计和跟踪, 而对角速度 x_2 并没有进行状态估计和跟踪。而实际上应主要是对 x_2 进行状态估计与跟踪, 因为 x_1 的信息是可测的。该文是针对跟踪误差进行观测, 而不是直接对状态进行观测, 也就是用如下跟踪误差方程来观测状态

$$\begin{cases} \dot{\hat{\mathbf{e}}} = \mathbf{A} \hat{\mathbf{e}} - \mathbf{B} \mathbf{K}_c^T \hat{\mathbf{e}} + \mathbf{B} u_c - \mathbf{B} v + \mathbf{K}_o (e_1 - \hat{e}_1) \\ \hat{e}_1 = \mathbf{C}^T \hat{\mathbf{e}} \end{cases} \quad (5-7)$$

$$\hat{\mathbf{e}} = \mathbf{y}_r - \hat{\mathbf{x}} \quad (5-8)$$

由 $\hat{\mathbf{e}} = \mathbf{y}_r - \hat{\mathbf{x}}$ 可以进一步推导出状态 $\mathbf{x}$ 的观测方程 $\hat{\mathbf{x}}$ 。

文献 [15, 17] 这种做法如果只是研究基于观测器的控制问题是可取的, 但在实际中观测器并不是只用到控制问题, 也可用到故障诊断等其他问题中。故我们认为文献 [15, 17] 设计的间接模糊观测器并不具有普遍意义。虽然可以用 $\hat{\mathbf{e}} = \mathbf{y}_r - \hat{\mathbf{x}}$ 可以推导出状态 $\mathbf{x}$ 的观测方程 $\hat{\mathbf{x}}$, 但要求的条件是知道参考输入 $\mathbf{y}_r$ 。

最近, A. Boulkroune 教授于 2008 年在国际杂志《Fuzzy set and systems》发表的论文^[18]系统地总结了该方面的主要论文有 16 篇, 其中对 14 篇论文进行了评论, 认为这些论文存在严重错误或不正确等问题。而我们早年在《American Control Conference》发表的文献^[19]是仅有的两篇之一没有被评论的, 认为是合理的。在此会议论文基础上, 我们系统地分析了国内外自适应模糊观测器所存在的问题, 我们设计了单输入单输出系统的自适应模糊观测器和输出控制器, 该方法具有更好的通用性。这方面的成果发表在国际期刊^[20], 我们在该文所设计的模糊观测器和其他论文的主要不同在于:

1) 采用类似经典的线性 Luenberger 观测器, 以及 Marino 等人提出的广义 Luenberger 自适应非线性观测器的方法, 即直接对状态进行观测并给出了观测误差收敛性证明, 而不是对跟踪误差进行观测。或者说, 我们设计了基于模糊系统的非线性广义 Luenberger 自适应模糊观测器, 是对非线性广义 Luenberger 自适应观测器的一种扩展。

2) 针对前文中的不可实现和无解问题进行了改进, 把观测器设计、控制器设计, 稳定性分析和系统实现进行了有效的整合。

3) 在仿真中对观测误差和跟踪误差都进行了详细的仿真实验验证。

4) 为了验证模糊观测器的有效性, 在相同控制器和参数的情况下与常用的传统高增益观测器进行了对比仿真研究。

最近, 将我们提出的单输入单输出系统的自适应模糊观测器技术扩展到含摩擦环节的多输入多输出机械系统的摩擦建模与控制补偿中^[21]。

5.3 数学模型

对于图 5-1 由质量、弹簧和阻尼构成的多自由度系统, 在仅考虑摩擦阻尼的情况下, 按

照牛顿第二定律可以得到下面的微分方程

$$\begin{cases} m \ddot{y}_1 = -F_{f1} - k_1 y_1 - k_2 (y_1 - y_2) + u_1 \\ \vdots \\ m \ddot{y}_p = -F_{fp} + k_p (y_{(p-1)} - y_p) + u_p \end{cases} \quad (5-9)$$

式中 u_i ——控制输入;

F_{fi} ——不确定摩擦力;

k_i ——刚性系数;

u ——控制力;

$i = 1, \dots, p$ 。

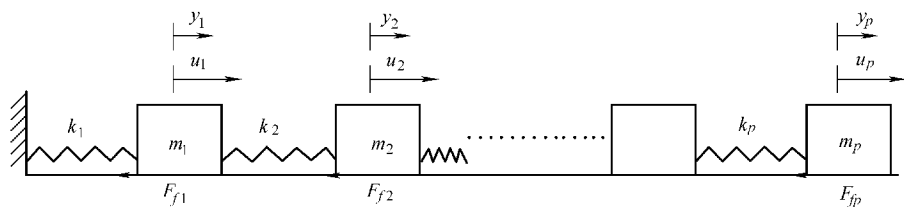


图 5-1 多自由度系统的示意图

设 $\mathbf{x} = (x_1, \dots, x_{2p})^T = (y_1, \dot{y}_1, \dots, y_p, \dot{y}_p)^T$ 表示该系统的状态向量, 只有系统的输出向量 $\mathbf{y} = (y_1, \dots, y_p)^T$ 是可检测的。方程 (5-9) 可改写为如下形式

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}(\mathbf{F}(\mathbf{x}) + \mathbf{H}(\mathbf{y}) + \mathbf{G}\mathbf{u}) \\ \mathbf{y} = \mathbf{C}\mathbf{x} \end{cases} \quad (5-10)$$

式中 $\mathbf{A} = \text{diag}[\mathbf{A}_1, \dots, \mathbf{A}_p]$;

$\mathbf{B} = \text{diag}[\mathbf{B}_1, \dots, \mathbf{B}_p]$;

$\mathbf{C}^T = \text{diag}[\mathbf{C}_1, \dots, \mathbf{C}_p]$;

$\mathbf{G} = \text{diag}[1/m_1, \dots, 1/m_p]$ 。

$\mathbf{u} = (u_1, \dots, u_p)^T$;

$\mathbf{A}_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$;

$\mathbf{B}_i = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$;

$\mathbf{C}_i = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$;

$$F(x) = \begin{bmatrix} \frac{F_{f_1}}{m_1} \\ \vdots \\ \frac{F_{f_p}}{m_p} \end{bmatrix};$$

$$H(y) = \begin{bmatrix} \frac{-k_1 y_1 - k_2 (y_1 - y_2)}{m_1} \\ \vdots \\ \frac{k_p (y_{(p-1)} - y_p)}{m_p} \end{bmatrix}。$$

对于给定的系统参考输入信号 $y_r = (y_{r_1}, \dots, y_{r_p})^T$, 设系统变量 $x = (x_1, \dot{x}_1, \dots, x_p, \dot{x}_p)^T$, 系统状态变量估计为 $\hat{x} = (\hat{x}_1, \dot{\hat{x}}_1, \dots, \hat{x}_p, \dot{\hat{x}}_p)^T$ 以及 $Y_r = (y_{r_1}, \dot{y}_{r_1}, \dots, y_{r_p}, \dot{y}_{r_p})^T$, 那么系统跟踪误差为 $e = Y_r - x = (e_1, \dot{e}_1, \dots, e_p, \dot{e}_p)^T$, 状态估计误差为 $\tilde{x} = x - \hat{x} = (\tilde{x}_1, \dot{\tilde{x}}_1, \dots, \tilde{x}_p, \dot{\tilde{x}}_p)^T$ 。

设计任务: 针对多自由度系统式 (5-10), 设计基于状态估计的摩擦模糊模型和鲁棒自适应控制, 以便满足下面的要求:

- 1) 模糊状态估计器能保证状态估计性能并且状态估计误差 $\tilde{x}$ 是一致最终有界的 (Ultimately uniformly bounded, UUB)。
- 2) 鲁棒自适应控制器能保证跟踪性能并且跟踪误差 e 也是一致最终有界的。

5.4 多摩擦环节的模糊建模与状态估计

5.4.1 多摩擦环节的模糊建模

由式 (5-9) 或式 (5-10) 可以看出, 含有多个摩擦环节的摩擦力向量是一种不确定量, 并且建立该模型的状态也是不完全可测的。所以, 构建下面基于状态估计的向量模糊系统逼近该摩擦力向量。用自适应模糊逻辑系统 $\hat{F}(\hat{x}, \hat{\Theta})$ 逼近式 (5-10) 中的摩擦力项 $F(x)$, 其中, $\hat{\Theta}$ 是可调参数向量。设 $\hat{F}_i(\hat{x}_i, \hat{\Theta}_i)$ 表示 $\hat{F}(\hat{x}, \hat{\Theta})$ 的第 i 个分量, 采用单值模糊化、乘积运算、加权平均反模糊化可得 I 型模糊逻辑系统为

$$\hat{F}_i(\hat{x}_i, \hat{\Theta}_i) = \xi_i^T(\hat{x}_i) \hat{\Theta}_i, i = 1, \dots, p \quad (5-11)$$

式中 $\hat{\Theta}_i = (\hat{\theta}_{i1}, \dots, \hat{\theta}_{im_i})^T \in \mathbf{R}^{m_i}$, 可调参数向量;

$\xi_i(\hat{x}_i) = (\xi_{i1}(\hat{x}_i), \dots, \xi_{im_i}(\hat{x}_i))^T \in \mathbf{R}^{m_i}$, 模糊基函数向量, 其中: $\hat{x}_i = (\hat{x}_{2i-1}, \hat{x}_{2i})^T$, 且有 $m_i > 0$ 。

模糊基函数 $\xi_{il}(\hat{x}_i)$ 定义如下

$$\xi_{il}(\hat{x}_i) = \frac{\mu_{F_{2i-1}^l}(\hat{x}_{2i-1})\mu_{F_{2i}^l}(\hat{x}_{2i})}{\sum_{l=1}^{m_i} \mu_{F_{2i-1}^l}(\hat{x}_{2i-1})\mu_{F_{2i}^l}(\hat{x}_{2i})}, l = 1, \dots, m_i \quad (5-12)$$

式中 $\mu_{F_{2i-1}^l}(\hat{x}_{2i-1})$ 和 $\mu_{F_{2i}^l}(\hat{x}_{2i})$ ——隶属函数。

最后模糊逻辑系统 $\hat{F}(\hat{x}, \hat{\Theta})$ 可表示为

$$\hat{F}(\hat{x}, \hat{\Theta}) = Y(\hat{x}) \hat{\Theta} \quad (5-13)$$

式中 $Y(\hat{x}) = \text{diag}[\xi_1^T(\hat{x}_1), \dots, \xi_p^T(\hat{x}_p)]$, 模糊基函数阵;

可调参数 $\hat{\Theta} = (\hat{\Theta}_1^T, \dots, \hat{\Theta}_p^T)^T \in R^m$, 且有 $m = \sum_{i=1}^p m_i$ 。

采用自适应模糊方法对状态不完全可观测的摩擦系统进行建模具有如下优点:

1) 众所周知, 摩擦是一种比较复杂的物理现象, 其动态特性有很大的不确定性, 人们至今也未能完全洞悉其机理。如果采用自适应观测器和 Luenberger 观测器等传统方法对系统不可测变量进行估计, 被观测对象的结构通常需要已知, 因此, 这些方法被限制了在工程上的应用。而采用自适应模糊方法不需要被观测对象精确的模型结构, 其系统模型可以使用模糊系统进行充分的逼近。所以, 自适应模糊方法对解决不完全可观测的不确定摩擦系统的状态估计和自适应控制具有重要意义。

2) 传统的摩擦机理模型认为摩擦力只是速度的函数, 但文献 [22] 研究认为摩擦力与位置和速度都有关系, 也就是说摩擦力应是位置和速度的函数。针对摩擦力与输入变量之间的不确定性, 在这里采用模糊处理方法, 通过自适应调整隶属度函数的参数来决定摩擦力与位置和速度的关系。

3) 采用自适应模糊方法, 专家的经验知识便于以规则的形式嵌入到摩擦模糊模型中。

5.4.2 状态估计与误差系统

由于摩擦力估计式 (5-13) 中的状态变量是不完全可测的, 为此构建下面的状态估计器

$$\begin{cases} \dot{\hat{x}} = A \hat{x} + B(\hat{F}(\hat{x}, \hat{\Theta}) + H(y) + Gu - v) + K_o(y - C \hat{x}) \\ \hat{y} = C \hat{x} \end{cases} \quad (5-14)$$

式中 $\hat{x}$ —— x 的估计;

$K_o = \text{diag}[K_1, \dots, K_p]$ 为估计器的增益, 系数的选择使 $A - K_o C$ 为稳定的, 其中: $K_i = (k_{i1}, k_{i2})^T (i = 1, \dots, p)$ 。

$\hat{F}(\hat{x}, \hat{\Theta})$ —— $F(x)$ 的估计量, 鲁棒监督补偿控制项 $v = (v_1, \dots, v_p)^T$ 是用来补偿逼近误差和估计误差。

由于 $\tilde{x} = x - \hat{x}$ 和 $\tilde{y} = y - \hat{y}$, 下面的状态估计误差可以得到

$$\begin{cases} \dot{\tilde{x}} = (A - K_o C) \tilde{x} + B(\tilde{F}(x, \hat{x}, \hat{\Theta}) + v) \\ \tilde{y} = C \tilde{x} \end{cases} \quad (5-15)$$

在上式中, $\tilde{F}(x, \hat{x}, \hat{\Theta})$ 可表示为如下形式

$$\tilde{F}(x, \hat{x}, \hat{\Theta}) = F(x) - \hat{F}(\hat{x}, \hat{\Theta}) \quad (5-16)$$

式 (5-15) 可改为如下形式

$$\tilde{y} = H(s)(\tilde{F}(x, \hat{x}, \hat{\Theta}) + v) \quad (5-17)$$

式中 s ——微分算子 d/dt ;

$H(s)$ ——稳定的传递函数阵, 其状态空间实现为 $(A - K_o C, B, C)$ 。

5.4.3 自适应模糊估计器

根据自适应模糊系统的万能逼近性质, 式 (5-10) 中的摩擦力项可用下面的模糊系统进行最优逼近

$$F(x) = Y(x)\Theta^* + \varepsilon \quad (5-18)$$

式中 Θ^* ——最优参数向量;

$\varepsilon = (\varepsilon_1, \dots, \varepsilon_p)^T$ ——最优逼近误差向量。

由式 (5-13) 可知, 摩擦力 $F(x)$ 的自适应模糊估计为 $\hat{F}(\hat{x}, \hat{\Theta})$ 。式 (5-16) 中 $\tilde{F}(x, \hat{x}, \hat{\Theta})$ 可表示为如下形式

$$\tilde{F}(x, \hat{x}, \hat{\Theta}) = Y(x)\Theta^* + \varepsilon - Y(\hat{x})\hat{\Theta} \quad (5-19)$$

在式 (5-19) 中分别增加和减去一项 $Y(\hat{x})\Theta^*$, 经过几步直接运算, 式 (5-19) 可改写为

$$\tilde{F}(x, \hat{x}, \hat{\Theta}) = Y(\hat{x})\tilde{\Theta} + \varepsilon + w \quad (5-20)$$

式中 $\tilde{\Theta} = \Theta^* - \hat{\Theta}$, 是自适应参数误差。

扰动项 w 由下式给出

$$w = \tilde{Y}(x, \hat{x})\Theta^* \quad (5-21)$$

式中 $\tilde{Y}(x, \hat{x}) = Y(x) - Y(\hat{x})$, 是模糊基函数误差阵;

$w = (w_1, \dots, w_p)^T$ 。

那么, 上面动态方程 (5-14) 和方程 (5-15) 可改写为

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + B(Y(\hat{x})\hat{\Theta} + H(y) + Gu - v) + K_o(y - C\hat{x}) \\ \hat{y} = C\hat{x} \end{cases} \quad (5-22)$$

$$\begin{cases} \dot{\tilde{x}} = (A - K_o C)\tilde{x} + B(Y(\hat{x})\tilde{\Theta} + v + d) \\ \tilde{y} = C\tilde{x} \end{cases} \quad (5-23)$$

式中 $d = \varepsilon + w$, 是总的不确定量, 由状态估计误差和逼近误差组成。

在式 (5-23) 中的输出估计误差 $\tilde{y}$ 可表示为

$$\tilde{\mathbf{y}} = H(s) (Y(\hat{\mathbf{x}}) \tilde{\Theta} + v + d) \quad (5-24)$$

式中 $H(s)$ ——稳定的传递函数阵，可以用 $(A - K_o C, B, C)$ 来实现。

下面将采用类似文献 [6] 的严格正实 Lyapunov 设计方法分析状态估计误差的一致最终有界性。

5.4.4 有界性分析

为了应用严格正实理论，输出估计误差 (5-24) 可以写成下面的形式

$$\tilde{\mathbf{y}} = H(s) L(s) (Y^l(\hat{\mathbf{x}}) \tilde{\Theta} + v^l + d^l) \quad (5-25)$$

式中 $Y^l(\hat{\mathbf{x}}) = L^{-1}(s) Y(\hat{\mathbf{x}})$;

$$v^l = L^{-1} v;$$

$$d^l = L^{-1} d。$$

选择一稳定的传递函数 $L(s)$ ，使 $H(s)L(s)$ 是严格正实的。

根据文献 [5] 用到的 Kalman - Yakub - Popov 定理，由于 $H(s)L(s)$ 是严格正实的，存在一对称正定矩阵 P 使得

$$\begin{cases} A_c^T P + P A_c = -Q \\ P B_c = C_c^T \end{cases} \quad (5-26)$$

式中 Q ——对称正定矩阵；

$$A_c = A - K_o C;$$

$$B_c = \text{diag} [B_{c_1}, \dots, B_{c_p}], \text{ 且有 } B_{c_i} = (b_{i1}, b_{i2})^T;$$

$$C_c = C。$$

那么，式 (5-24) 的状态空间实现表示如下

$$\begin{cases} \dot{\tilde{\mathbf{z}}} = A_c \tilde{\mathbf{z}} + B_c (Y^l(\hat{\mathbf{x}}) \tilde{\Theta} + v^l + d^l) \\ \tilde{\mathbf{y}} = C \tilde{\mathbf{z}} \end{cases} \quad (5-27)$$

假设 1 假设 d_i 满足条件 $\|d_i\| \leq d_{iH}$ ，其中， d_{iH} 是正的常数 ($i=1, \dots, p$)。总的确定项 d_i 是由逼近误差和状态估计误差组成，由万能逼近定理可知这种假设是合理的。

定理 1 对于系统 (5-10) 满足上面的假设 1，相应的状态估计器为式 (5-22)， $\hat{\Theta}$ 的自适应律为式 (5-28)，在上式中， v 为式 (5-29) 的鲁棒控制补偿项

$$\dot{\hat{\Theta}} = \gamma \text{Proj}(\hat{\Theta}, \hat{\Phi}) \quad (5-28)$$

$$v = -Sp, \text{sign}(\tilde{y}_i) = \begin{cases} +1, \tilde{y}_i \geq 0 \\ -1, \tilde{y}_i < 0 \end{cases} \quad (5-29)$$

式中 $\gamma > 0$;

$$S = \text{diag}[\text{sign}(\tilde{y}_1), \dots, \text{sign}(\tilde{y}_p)];$$

$\boldsymbol{\rho} = (\rho_1, \dots, \rho_p)^T$, 且有 $\rho_i > d_{iH}$;

$\hat{\Phi} = Y^l(\hat{\mathbf{x}}) \tilde{\mathbf{y}}$;

$$\text{Proj}(\hat{\Theta}, \hat{\Phi}) = \begin{cases} \hat{\Phi} - \frac{\hat{\Phi}^T \hat{\Theta}}{\|\hat{\Theta}\|^2} \hat{\Theta} \\ \hat{\Phi}, \text{其他} \end{cases}。$$

如果 $\|\hat{\Theta}\| = M_{\hat{\Theta}}$ 且 $\dot{\Phi}^T \hat{\Theta} > 0$, 那么状态估计误差 $\tilde{\mathbf{x}}$ 是一致最终有界的。

证明: 取 Lyapunov 函数为

$$V = \frac{1}{2} \tilde{\mathbf{z}}^T \mathbf{P} \tilde{\mathbf{z}} + \frac{1}{2\gamma} \tilde{\Theta}^T \tilde{\Theta} \quad (5-30)$$

式中 $\mathbf{P} = \mathbf{P}^T > 0$ 。

微分式 (5-30) 可推出

$$\dot{V} = \frac{1}{2} \dot{\tilde{\mathbf{z}}}^T \mathbf{P} \tilde{\mathbf{z}} + \frac{1}{2} \tilde{\mathbf{z}}^T \mathbf{P} \dot{\tilde{\mathbf{z}}} + \frac{1}{\gamma} \tilde{\Theta}^T \dot{\tilde{\Theta}} \quad (5-31)$$

基于式 (5-27), 可以得到

$$\dot{V} = \frac{1}{2} \tilde{\mathbf{z}}^T (\mathbf{A}_c^T \mathbf{P} + \mathbf{P} \mathbf{A}_c) \tilde{\mathbf{z}} + \tilde{\mathbf{z}}^T \mathbf{P} \mathbf{B}_c Y^l(\hat{\mathbf{x}}) \tilde{\Theta} + \tilde{\mathbf{z}}^T \mathbf{P} \mathbf{B}_c v^l + \tilde{\mathbf{z}}^T \mathbf{P} \mathbf{B}_c d^l + \frac{1}{\gamma} \tilde{\Theta}^T \dot{\tilde{\Theta}} \quad (5-32)$$

因为 $\tilde{\mathbf{z}}^T \mathbf{P} \mathbf{B}_c = \tilde{\mathbf{z}}^T \mathbf{C}_c^T = \tilde{\mathbf{y}}^T$, $\dot{\tilde{\Theta}} = -\dot{\hat{\Theta}}$, 上式可以写成

$$\dot{V} = -\frac{1}{2} \tilde{\mathbf{z}}^T \mathbf{Q} \tilde{\mathbf{z}} + \tilde{\mathbf{y}}^T (Y^l(\hat{\mathbf{x}}) \tilde{\Theta} + v^l + d^l) - \frac{1}{\gamma} \tilde{\Theta}^T \dot{\hat{\Theta}} \quad (5-33)$$

因为

$$\tilde{\mathbf{y}}^T (\mathbf{v} + \mathbf{d}) = \sum_{i=1}^p \tilde{y}_i (-\rho_i \text{sign}(\tilde{y}_i) + d_{iH})$$

对于 $\tilde{y}_i \geq 0$, 由于 $\rho_i > d_{iH}$, 我们可以得到 $(-\rho_i \text{sign}(\tilde{y}_i) + d_{iH}) < 0$; 对于 $\tilde{y}_i < 0$, 由于 $\rho_i > d_{iH}$, 可以得到 $(-\rho_i \text{sign}(\tilde{y}_i) + d_{iH}) > 0$, 因此 $\tilde{\mathbf{y}}^T (\mathbf{v} + \mathbf{d}) < 0$ 。另外, $\mathbf{L}^{-1}(s)$ 可取稳定的传递函数和 $H(s)L(s)$ 是严格正实的, 可得到 $\tilde{\mathbf{y}}^T (v^l + d^l)$ 。所以方程 (5-33) 可写成下面不等式

$$\begin{aligned} \dot{V} &\leq -\frac{1}{2} \lambda_{\min} \|\tilde{\mathbf{z}}\|^2 + \tilde{\mathbf{y}}^T Y^l(\hat{\mathbf{x}}) \tilde{\Theta} - \frac{1}{\gamma} \tilde{\Theta}^T \dot{\hat{\Theta}} \\ &= -\frac{1}{2} \lambda_{\min} \|\tilde{\mathbf{z}}\|^2 + \left(\tilde{\mathbf{y}}^T Y^l(\hat{\mathbf{x}}) - \frac{1}{\gamma} \dot{\hat{\Theta}}^T \right) \tilde{\Theta} \end{aligned} \quad (5-34)$$

根据自适应律可以得到

$$\dot{V} \leq -\frac{1}{2} \lambda_{\min} \|\tilde{\mathbf{z}}\|^2 \quad (5-35)$$

按照标准 Lyapunov 理论, 说明状态观测误差 $\tilde{\mathbf{z}}$ 和自适应参数误差 $\tilde{\Theta}$ 是一致最终有界的。为了说明状态估计误差 $\tilde{\mathbf{x}}(t)$ 的有界性, 考虑方程 (5-23) 的估计误差, 轨迹可表示为如下

形式

$$\tilde{\mathbf{x}}(t) = \Phi(t, 0)\tilde{\mathbf{x}}(0) + \int_0^t \Phi(t, \tau) \mathbf{B}U_1(\tau) d\tau \quad (5-36)$$

式中 $U_1 = \mathbf{Y}(\hat{\mathbf{x}})\tilde{\Theta} + v + d$ 。

由范数不等式及文献 [5] 中的引理 2.2 可得

$$\|\tilde{\mathbf{x}}(t)\| \leq C_1 + (C_2 + C_3 \|\tilde{\Theta}\|_2^\alpha) \frac{1}{\sqrt{\alpha}} \quad (5-37)$$

式中 C_1 ——按指数衰减到零；

C_2 和 C_3 ——正数。

因为 $\|\tilde{\Theta}\|$ 是一致最终有界并结合上式可知 $\|\tilde{\mathbf{x}}(t)\|$ 是一致最终有界的，即设计任务达到。

5.4.5 状态估计的仿真

把本文所提出的摩擦状态估计器用于图 5-2 系统的仿真，该系统的动态模型为

$$\begin{cases} m\ddot{y}_1 = -F_{f1} - k_1 y_1 - k_2(y_1 - y_2) + u_1 \\ m\ddot{y}_2 = -F_{f2} + k_2(y_1 - y_2) + u_2 \end{cases} \quad (5-38)$$

式中 k_1 和 k_2 ——刚性弹簧系数；

F_{f1} 和 F_{f2} ——摩擦力；

u_1 和 u_2 ——控制输入；

y_1 和 y_2 ——系统输出。

设 $x_1 = y_1$, $x_2 = \dot{y}_1$, $x_3 = y_2$, $x_4 = \dot{y}_2$ ，那么上面的方程 (5-38) 能改写为如下形式

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}(\mathbf{F}(\mathbf{x}) + \mathbf{H}(\mathbf{y}) + \mathbf{G}\mathbf{u}) \\ \mathbf{y} = \mathbf{C}\mathbf{x} \end{cases} \quad (5-39)$$

式中 $\mathbf{x} = (x_1, x_2, x_3, x_4)^T$ ；

$\mathbf{A} = \text{diag}[A_1, A_2]$ ；

$\mathbf{B} = \text{diag}[B_1, B_2]$ ；

$\mathbf{C}^T = \text{diag}[C_1, C_2]$ ；

$\mathbf{u} = (u_1, u_2)^T$ ，

$A_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ ；

$B_i = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ ；

$C_i = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ ；

$i = 1, 2$ ；

$\mathbf{G} = \begin{bmatrix} 1/m_1 & 0 \\ 0 & 1/m_2 \end{bmatrix}$ ，

$$F(x) = [F_{f_1}/m_1, F_{f_2}/m_2]^T;$$

$$H(y) = \left[\frac{-k_1 y_1 - k_2 (y_1 - y_2)}{m_1}, \frac{k_2 (y_1 - y_2)}{m_2} \right]^T。$$

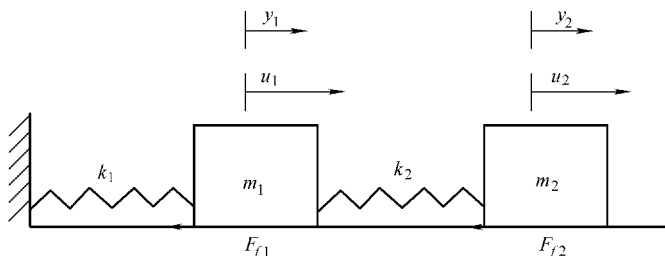


图 5-2 二自由度系统的示意图

仿真过程的摩擦力 $F_{f_i} = \mu F_N$ ，其摩擦因数 μ 的三种模型：① $\mu = \mu_c \operatorname{sgn}(\dot{x}_1)$ 。② $\mu = \mu_c \operatorname{sgn}(\dot{x}_3) + \mu_v \dot{x}_3$ 。③ $\mu = \mu_c \operatorname{sgn}(\dot{x}_1) + (\mu_s - \mu_c) e^{-1 \dot{x}_1^2} + \mu_v \dot{x}_1$ 。

在第一个仿真里，图 5-2 中的 F_{f_1} 和 F_{f_2} 摩擦模型分别取 Couloumb 模型和 Couloumb + Viscous 摩擦模型。在另一个仿真里，将图 5-2 中 F_{f_1} 的 Couloumb 模型换成一个较复杂的摩擦模型，也就是 Stribeck 模型，进一步考察系统的观测性能。表 5-1 是三种摩擦模型的参数定义及给定值。

表 5-1 摩擦模型参数与结构

参数	Couloumb	Couloumb + Viscous	Stribeck
μ_c	0.20	0.25	0.25
μ_v	0.00	0.04	0.04
μ_s	0.00	0.00	0.04

步骤 1. 变量 $\hat{x}_i$ 的隶属度函数选择如下：

$\mu_{F_1^1} = 1/(1 + \exp(5(\hat{x}_i + 2)))$, $\mu_{F_1^2} = \exp(-(\hat{x}_i + 1.5)^2)$, $\mu_{F_1^3} = \exp(-(\hat{x}_i + 0.5)^2)$, $\mu_{F_1^4} = \exp(-(\hat{x}_i - 0.5)^2)$, $\mu_{F_1^5} = \exp(-(\hat{x}_i - 1.5)^2)$, $\mu_{F_1^6} = 1/(1 + \exp(-5(\hat{x}_i - 2)))$ ($i = 1, 2, 3, 4$)。

建立如下的模糊规则库：

1) $R_1^{(j)}$: If $\hat{x}_1$ is F_1^j and $\hat{x}_2$ is F_2^j Then F_{f1} is G_1^j ($j = 1, \dots, 6$)。

2) $R_2^{(j)}$: If $\hat{x}_3$ is F_3^j and $\hat{x}_4$ is F_4^j Then F_{f2} is G_2^j ($j = 1, \dots, 6$)。

设

$$\begin{cases} \xi_{1j}(\hat{\mathbf{x}}_1) = \frac{\mu_{F_1^j}(\hat{x}_1)\mu_{F_2^j}(\hat{x}_2)}{\sum_{j=1}^6 \mu_{F_1^j}(\hat{x}_1)\mu_{F_2^j}(\hat{x}_2)} \\ \xi_{2j}(\hat{\mathbf{x}}_2) = \frac{\mu_{F_3^j}(\hat{x}_3)\mu_{F_4^j}(\hat{x}_4)}{\sum_{j=1}^6 \mu_{F_3^j}(\hat{x}_3)\mu_{F_4^j}(\hat{x}_4)} \end{cases} \quad (5-40)$$

那么得到下面的模糊基函数:

$$\begin{cases} \xi_1(\hat{x}_1) = (\xi_{11}(\hat{x}_1), \dots, \xi_{16}(\hat{x}_1))^T \\ \xi_2(\hat{x}_2) = (\xi_{21}(\hat{x}_2), \dots, \xi_{26}(\hat{x}_2))^T \end{cases} \quad (5-41)$$

步骤 2. 系统仿真参数选择:

$\gamma = 100$, $L^{-1} = 1/(S + 3)$, $\rho = 15$, $x(0) = (0.5, 0.0, -0.5, 0.0)^T$, $\hat{x}(0) = (0.0, 0.0, 0.0, 0.0)^T$, $K_o = [89 \ 184 \ 0 \ 0; 0 \ 0 \ 89 \ 184]^T$, $\hat{\Theta}_1(0) = \hat{\Theta}_2(0) = (-8, -6, -1, 3, 8, 1)^T$ 。

步骤 3. 求解方程 (5-22), 可得摩擦建模状态估计器

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + 89(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_2 = \hat{\Theta}_1^T \xi_1(\hat{x}_1) + \frac{-k_1 y_1 - k_2(y_1 - y_2)}{m_1} + \frac{u_1}{m_1} - v_1 + 184(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_3 = \hat{x}_4 + 89(x_3 - \hat{x}_3) \\ \dot{\hat{x}}_4 = \hat{\Theta}_2^T \xi_2(\hat{x}_2) + \frac{k_2(y_1 - y_2)}{m_2} + \frac{u_2}{m_2} - v_2 + 184(x_3 - \hat{x}_3) \\ u_1 = u_2 = \sin(2t) + \cos(20t) \end{cases} \quad (5-42)$$

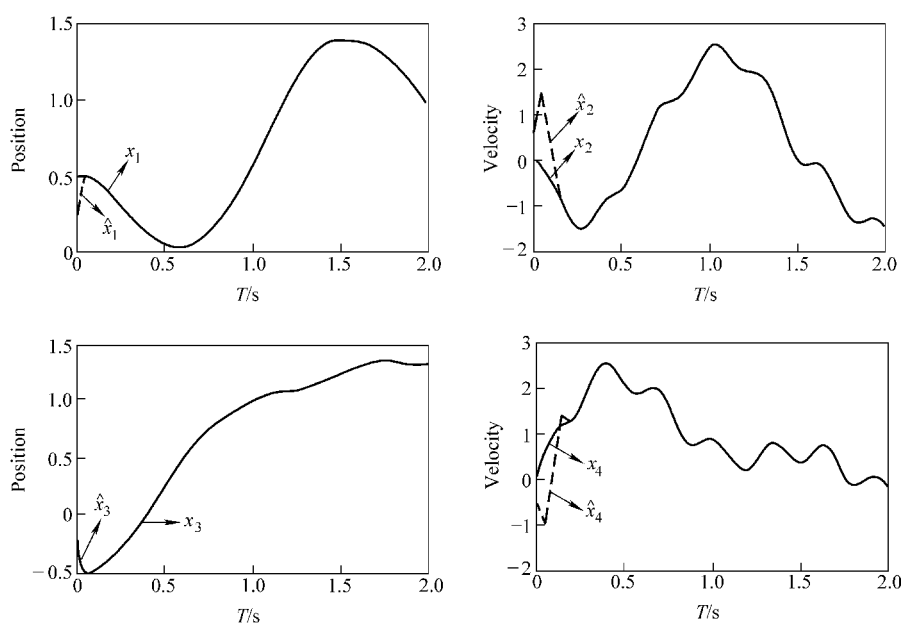
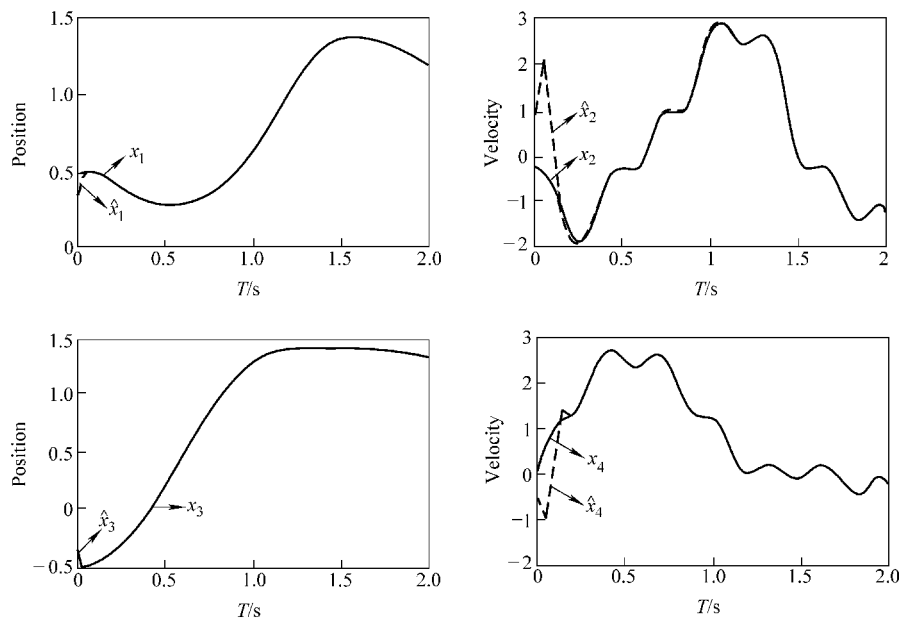
步骤 4. 选择下面的参数自适应律

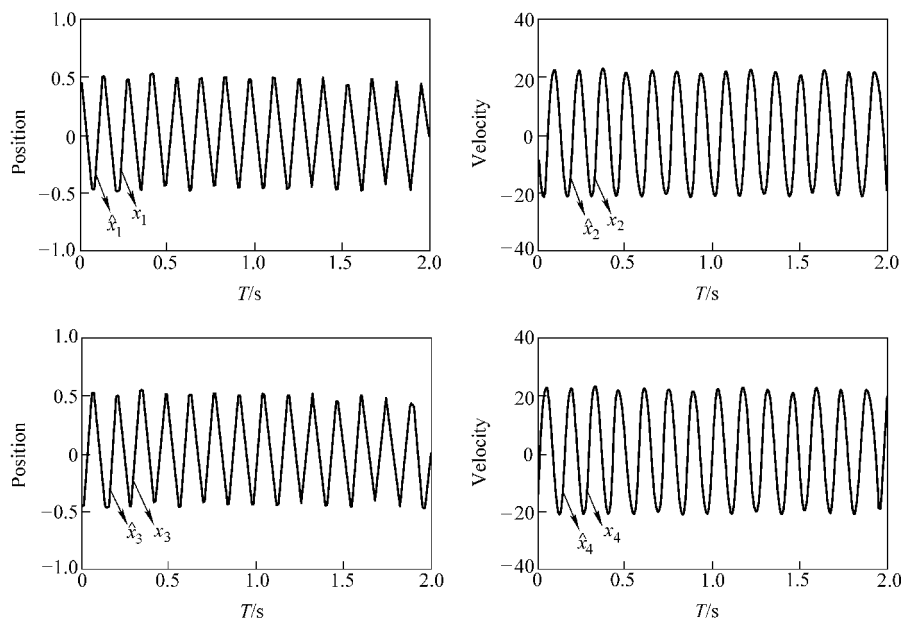
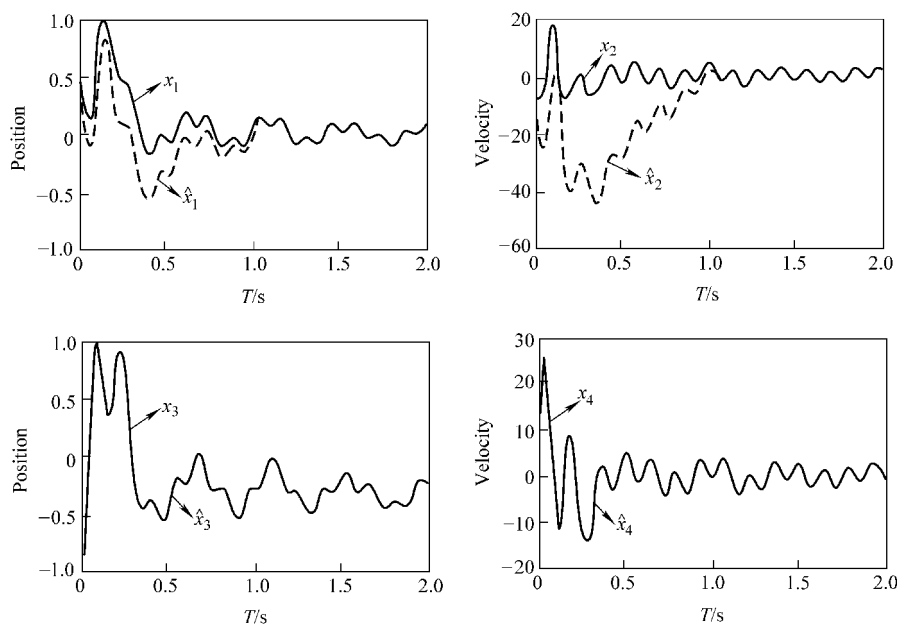
$$\begin{cases} \dot{\hat{\Theta}}_1^T = \gamma \xi_1^l(\hat{x}_1)(x_1 - \hat{x}_1) \\ \dot{\hat{\Theta}}_2^T = \gamma \xi_2^l(\hat{x}_2)(x_3 - \hat{x}_3) \end{cases} \quad (5-43)$$

图 5-3 是图 5-2 中的 F_{f1} 和 F_{f2} 分别取 Couloumb 和 Couloumb + Viscous 摩擦时, 状态估计器所估计的位置与速度曲线和实际曲线。而图 5-4 是图 5-2 中的 F_{f1} 和 F_{f2} 分别取 Stribeck 和 Couloumb + Viscous 摩擦时, 状态估计器所估计的位置与速度曲线和实际曲线。在前面两个仿真里, 鲁棒补偿控制项为 $\mathbf{v} = (-10\text{sgn}(x_1 - \hat{x}_1), -10\text{sgn}(x_3 - \hat{x}_3))^T$, $m_1 = m_2 = 1.00\text{kg}$, $k_1 = 1.0\text{N/m}$, $k_2 = 2.0\text{N/m}$ 。

图 5-5 是图 5-2 中的 F_{f1} 和 F_{f2} 分别取 Couloumb 和 Couloumb + Viscous 摩擦时, 状态估计器所估计的位置与速度曲线和实际曲线。而图 5-6 是图 5-2 中的 F_{f1} 和 F_{f2} 分别取 Stribeck 和 Couloumb + Viscous 摩擦时, 状态估计器所估计的位置与速度曲线和实际曲线。在后面两个仿真里, 鲁棒补偿控制项为 $\mathbf{v} = (-10\text{sgn}(x_1 - \hat{x}_1), -10\text{sgn}(x_3 - \hat{x}_3))^T$, $m_1 = 0.20\text{kg}$, $m_2 = 0.25\text{kg}$, $k_1 = 100.0\text{N/m}$, $k_2 = 200.0\text{N/m}$ 。

从这四个仿真结果可以看出, 无论摩擦力取 Couloumb 摩擦模型、Couloumb + Viscous 摩擦模型、Stribeck 摩擦模型, 还是系统的质量和弹簧刚性参数在变化, 我们所设计的摩擦状态估计器具有良好的状态估计效果, 具有良好的自适应特性, 完全可以用该估计状态对摩擦进行建模与控制补偿。

图 5-3 实际状态 x 和估计状态 $\hat{x}$ 图 5-4 实际状态 x 和估计状态 $\hat{x}$

图 5-5 实际状态 $\mathbf{x}$ 和估计状态 $\hat{\mathbf{x}}$ 图 5-6 实际状态 $\mathbf{x}$ 和估计状态 $\hat{\mathbf{x}}$

5.5 鲁棒自适应控制器设计

5.5.1 控制器设计

对于系统 (5-10)，如果系统状态可测且 $\mathbf{F}(\mathbf{x})$ 已知，则理想控制律为

$$\mathbf{u} = \mathbf{G}^{-1}(-\mathbf{F}(\mathbf{x}) - \mathbf{H}(\mathbf{y}) + \mathbf{y}_r^{(2)} + \mathbf{K}_c \mathbf{e}) \quad (5-44)$$

把式 (5-44) 代入式 (5-10) 得

$$\mathbf{E} + \mathbf{K}_c \mathbf{e} = 0 \quad (5-45)$$

式中, $\mathbf{E} = (e_1^{(2)}, \dots, e_p^{(2)})^T$ 。

显然如果取 $\mathbf{K}_c = \text{diag}[\mathbf{K}_1, \dots, \mathbf{K}_p]$, 且有 $\mathbf{K}_l = (k_{l1}, k_{l2})$ 使系统为稳定的, 则有 $\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| = 0$ 。然而在系统状态不完全可测和 $\mathbf{F}(\mathbf{x})$ 未知情况下。获得理想控制器 (5-44) 是不可能的。在这种情况下, 采用前文所构造的模糊逻辑系统 $\hat{\mathbf{F}}(\hat{\mathbf{x}}, \hat{\boldsymbol{\Theta}})$ 来逼近摩擦力项。取控制律为

$$\mathbf{u} = \mathbf{G}^{-1}(-\hat{\mathbf{F}}(\hat{\mathbf{x}}, \hat{\boldsymbol{\Theta}}) - \mathbf{H}(\mathbf{y}) + \mathbf{y}_r^{(2)} + \mathbf{K}_c \hat{\mathbf{e}} - u_\alpha) \quad (5-46)$$

式中 u_α ——鲁棒补偿控制。

应用式 (5-46) 到式 (5-10) 经过几步直接的运算后可得如下的跟踪误差方程

$$\dot{\mathbf{e}} = \mathbf{A}\mathbf{e} + \mathbf{B}(\hat{\mathbf{F}}(\hat{\mathbf{x}}, \hat{\boldsymbol{\Theta}}) - \mathbf{F}(\mathbf{x})) - \mathbf{B}\mathbf{K}_c \hat{\mathbf{e}} + \mathbf{B}u_\alpha \quad (5-47)$$

基于式 (5-20), 上面的方程可以写为

$$\dot{\mathbf{e}} = \mathbf{A}\mathbf{e} - \mathbf{B}\mathbf{K}_c \hat{\mathbf{e}} - \mathbf{B}\mathbf{Y}(\hat{\mathbf{x}}) \tilde{\boldsymbol{\Theta}} + \mathbf{B}u_\alpha - \mathbf{B}d \quad (5-48)$$

定理 2. 对于系统 (5-10), 如果采用控制方案 (5-49), 系统状态估计用式 (5-22), 自适应参数调节采用式 (5-28),

$$\begin{cases} \mathbf{u} = \mathbf{G}^{-1}(-\hat{\mathbf{F}}(\hat{\mathbf{x}}, \hat{\boldsymbol{\Theta}}) - \mathbf{H}(\mathbf{y}) + \mathbf{y}_r^{(2)} + \mathbf{K}_c \hat{\mathbf{e}} - u_\alpha) \\ u_\alpha = -v \end{cases} \quad (5-49)$$

那么跟踪误差是一致最终有界的。

证明: 考虑下面基本定义

$$\tilde{\mathbf{e}} = \mathbf{e} - \hat{\mathbf{e}} = (\mathbf{Y}_r - \mathbf{x}) - (\mathbf{Y}_r - \hat{\mathbf{x}}) = -\tilde{\mathbf{x}} \quad (5-50)$$

把式 (5-50) 代入式 (5-47), 经简单运算后可得

$$\dot{\mathbf{e}} = (\mathbf{A} - \mathbf{B}\mathbf{K}_c)\mathbf{e} - \mathbf{B}(\mathbf{Y}(\hat{\mathbf{x}}) \tilde{\boldsymbol{\Theta}} - \mathbf{K}_c \tilde{\mathbf{e}}) + \mathbf{B}(u_\alpha - d) \quad (5-51)$$

方程 (5-48) 的轨迹可表示为

$$\mathbf{e}(t) = \boldsymbol{\Phi}(t, 0)\mathbf{e}(0) + \int_0^t \boldsymbol{\Phi}(t, \tau)\mathbf{B}U_2(\tau) d\tau \quad (5-52)$$

式中 $U_2 = \mathbf{Y}(\hat{\mathbf{x}}) \tilde{\boldsymbol{\Theta}} - \mathbf{K}_c \tilde{\mathbf{e}} - u_\alpha + d$ 。

由范数不等式及文献[5]中的引理 2.2 可得

$$\|\mathbf{e}(t)\| \leq D_1 + (D_2 + D_3 \|\tilde{\boldsymbol{\Theta}}\|_2^\alpha + D_4 \|\tilde{\mathbf{e}}\|_2^\alpha) \frac{1}{\sqrt{\alpha}} \quad (5-53)$$

式中 D_1 ——按指数衰减到零; D_2 、 D_3 和 D_4 ——正数。

从前文已证 $\|\tilde{\boldsymbol{\Theta}}\|$ 和 $\|\tilde{\mathbf{e}}\|$ 是一致最终有界, 并结合上式可知 $\|\mathbf{e}(t)\|$ 是一致最终有界的, 即设计任务达到。

5.5.2 控制器仿真

对于前面图 5-2 的机械系统, 用下面的鲁棒自适应控制器验证其跟踪效果

$$u = G^{-1}(-\hat{F}(\hat{x}, \hat{\Theta}) - H(y) + y_r^{(2)} + K_e \hat{e} - u_\alpha) \quad (5-54)$$

式中 $K_e = [4 \ 4 \ 0 \ 0; 0 \ 0 \ 4 \ 4]^T$;

$$u_\alpha = -v_o.$$

图 5-2 中的 F_{f1} 和 F_{f2} 分别取 Coulomb 和 Coulomb + Viscous 摩擦模型, 鲁棒自适应模糊输出反馈控制器所形成的跟踪曲线和参考曲线如图 5-7 所示, 该仿真结果是在参考输入信号取 $y_r = (0.5\sin(t), 0.5\cos(t))^T$ 时获得的。图 5-8 是系统摩擦力 F_{f1} 和 F_{f2} 分别取 Stribeck

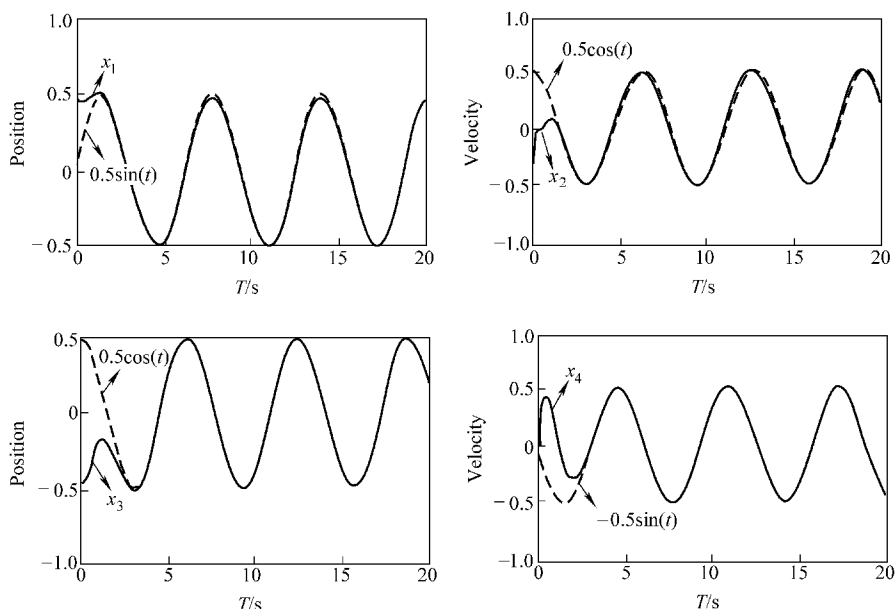


图 5-7 实际状态 x 和参考输入 y_r 的曲线

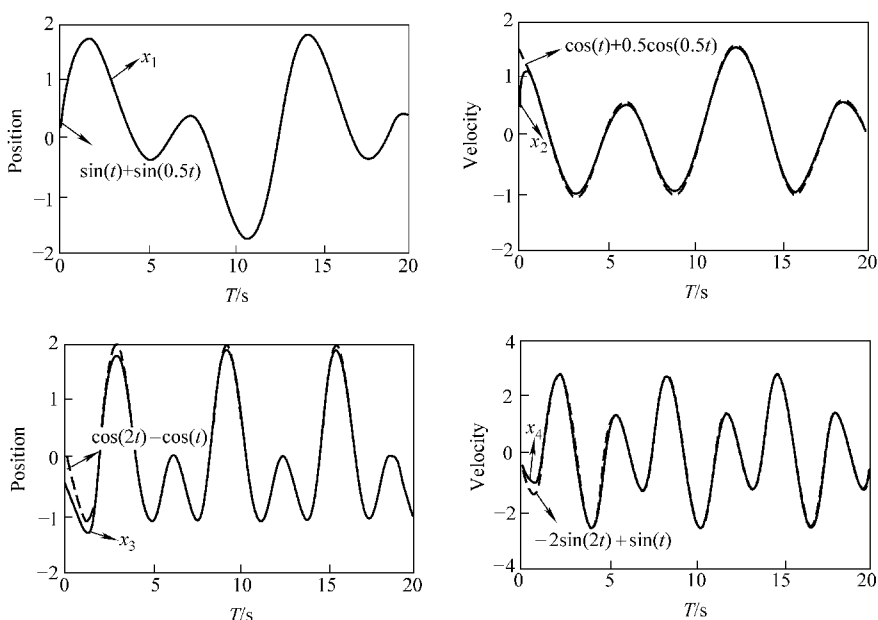


图 5-8 实际状态 x 和参考输入 y_r 的曲线

和 Coulomb + Viscous 摩擦模型, 该控制器所形成的跟踪曲线和参考曲线, 该仿真参考输入信号取 $y_r = (\sin(t) + \sin(0.5t), \cos(2t) - \cos(t))^T$ 。从仿真结果可以看出, 采用基于模糊观测器的鲁棒自适应输出反馈控制获得了良好的跟踪效果。综上, 基于模糊观测器的多摩擦建模与补偿控制获得了满意效果。

5.6 本章小结

本章针对机械系统中普遍存在的摩擦问题进行了研究, 探讨了多自由度系统在摩擦力是未知的, 且状态不完全可测的情况下, 如何用基于状态估计的多个自适应模糊系统在线逼近非线性摩擦力项。为了建立基于状态估计的摩擦模糊模型, 对摩擦力中的状态变量进行了状态估计, 并用严格正实和 Lyapunov 理论保证了状态估计误差的一致有界性。接下来, 对闭环系统误差的有界性进行了理论分析, 从仿真中可以看出采用该控制器可以获得良好的跟踪性能。本章的方法虽然是以质量弹簧阻尼对象展开研究的, 但该方法对其他机械系统 (如机器人等) 也是适用的。

参考文献

- [1] Marino R, Tomei P. Global adaptive observer for nonlinear systems via filtered transformations [J]. IEEE Trans. Automatic Control, 1992, 37 (8): 1239 - 1245.
- [2] Marino R, Tomei P. Adaptive observer with arbitrary exponential rate of convergence for nonlinear systems [J]. IEEE Trans. Automatic Control, 1995, 40 (7): 1300 - 1304.
- [3] Marino R, Tomei P. Robust adaptive observer for nonlinear systems with bounded disturbances [J]. IEEE Trans. Automatic Control, 2001, 46 (6): 967 - 972.
- [4] Jose A, Wen Y. A high - gain observer - based PD control for robots manipulator [C]. Proceedings of the American Control Conference, 2000, 2518 - 2522.
- [5] Kim Y H, Frank L L. Dynamic recurrent neural network - based adaptive observer for a class of nonlinear systems [J]. Automatica, 1997, 33 (8): 1539 - 1543.
- [6] Wang Y F, Chai T Y, Tong S C. Adaptive observer for a class of nonlinear systems based on fuzzy basis functions [C]. Proceedings of the Second International Conference On Machine Learning and Cybernetics. Xian China, 2003, 1015 - 1020.
- [7] Wang L X. Stable adaptive fuzzy control of nonlinear systems [J]. IEEE Trans. on Fuzzy Systems, 1993, 1 (1): 146 - 155.
- [8] 张乃尧, 金晖. 对稳定的模糊自适应控制方案的研究与改进 [J]. 自动化学报, 1997, 23 (2): 160 - 166.
- [9] Chen B S, Lee C H, Chang Y C. Tracking design of uncertain nonlinear SISO systems: adaptive fuzzy approach [J]. IEEE Trans. on Fuzzy Systems, 1996, 4 (1): 32 - 43.
- [10] 佟绍成, 柴天佑. 关于多变量非线性系统的自适应模糊控制 [J]. 自动化学报, 1998, 24 (6): 793 - 797.
- [11] Chang Y C. Robust tracking control of nonlinear MIMO systems via fuzzy approaches [J]. Automatica, 2000, 36 (3): 1535 - 1545.

- [12] C. Y. Sue, Y. Stepanenko. Adaptive control of a class of nonlinear systems with fuzzy logic [J]. IEEE Trans. on Fuzzy Systems, 1994, 2 (2): 285 – 294.
- [13] 张天平, 冯纯伯. 一类非线性系统的自适应模糊滑模控制 [J]. 自动化学报, 1997, 23 (2): 361 – 369.
- [14] Sun F C, Sun Z Q. Neural network – based adaptive controller design of robotics manipulators with an observer [J]. IEEE Trans. on Neural Networks, 2001, 12 (1): 54 – 67.
- [15] Wang C H, Liu H L, Lin T C. Direct fuzzy – neural control with observer and supervisory control for unknown nonlinear dynamical systems [J]. IEEE Trans. on Fuzzy Systems, 2002, 10 (1): 39 – 49.
- [16] Tong S C, Li H X, Wang W. Comment on “Direct fuzzy – neural control with Observer and supervisory control for unknown nonlinear dynamical systems” [J]. IEEE Trans. on Fuzzy Systems, 2003, 11 (5): 703 – 705.
- [17] Leu Y G. Observer – based adaptive fuzzy neural control for unknown nonlinear dynamical systems [J]. IEEE Trans. on Systems, Man, Cybernetics, 1999, 29 (3): 583 – 591.
- [18] A. Boulkroune, M. Tadjine, M. Saad, M. Farza, How to design a fuzzy adaptive controller based on observers for uncertain affine nonlinear systems [J]. Fuzzy Sets and Systems, 2008, 159 (8): 926 – 948.
- [19] Yongfu Wang, Tianyou Chai. Output – feedback control of uncertain nonlinear systems using adaptive fuzzy observer [C]. American Control Conference, 2005, 2613 – 2618.
- [20] Yongfu Wang, Tianyou Chai, Yimin Zhang. State observer – based adaptive fuzzy output – feedback control for a class of uncertain nonlinear systems [J]. Information Science, 2010, 180 (2): 5029 – 5040.
- [21] Yongfu Wang, Dianhui Wang, Tianyou Chai. State estimate – based robust adaptive control of a linked mass – spring system with nonlinear friction [J]. ASME Journal of Dynamic Systems, Measurement and Control, 2013, 135 (2): 251 – 259.
- [22] Selmic R R, Lewis F L. Neural – network approximation of piecewise continuous functions; application to friction compensation [J]. IEEE Transactions on Neural Networks, 2002, 13 (3): 745 – 751.

第 6 章 非线性摩擦的双调节模糊建模与鲁棒自适应控制

6.1 引言

在前面章节已对摩擦进行了静态和动态模糊建模与控制补偿的研究，为用模糊逻辑系统研究结构和参数均未知的非线性摩擦的控制补偿问题开辟了新途径。但前面章节中，摩擦模型的模糊自适应参数仅考虑由系统跟踪误差进行调节，而没有考虑摩擦建模误差的影响。本章将探索摩擦模型的自适应参数由跟踪误差和摩擦建模误差共同调节的方案，并考虑系统存在不确定摩擦建模误差情况下如何设计鲁棒补偿控制器。采用此方案的意义在于：

- 1) 由于自适应参数中引入了摩擦建模误差，加快了系统跟踪误差的收敛速度。
- 2) 摩擦建模误差调节自适应参数的本质问题就是自适应参数收敛到真值问题，使得自适应模糊系统能更好地逼近非线性摩擦力。
- 3) 自适应参数由两种误差调节更具有双重意义，它为我们如何构造具有实际意义的自适应模糊系统提供了一个方向性指导，也就是既要保证控制系统的跟踪性能，也要保证建模系统的逼近性能。

6.2 不确定机械动力系统

6.2.1 问题描述

为了便于问题的描述，考虑下面所用的一维运动系统^[1]，系统运动方程如下

$$m \frac{d^2 x}{dt^2} = u - F_f \quad (6-1)$$

式中 m ——运动体的质量；
 x ——运动体的位移；
 u ——控制力；
 F_f ——不确定摩擦力。

对于给定的有界参考输入信号 $\underline{y}_r = (y_r, \dot{y}_r)^T$ 假设均有界可测，定义系统输出跟踪误差为 $e = y_r - y$ ，跟踪误差向量为 $\underline{e} = (e, \dot{e})^T$ 。

目标：设计控制器 $u = u(\underline{e} | \underline{\theta})$ 及自适应参数向量 $\underline{\theta}$ 的自适应律满足以下所要证明指标：

- 1) 对于给定的减弱水平 $\rho > 0$ ，输出跟踪误差取得如下 H_∞ 性能指标

$$\int_0^T \underline{e}^T Q \underline{e} dt \leq \underline{e}^T(0) P \underline{e}(0) + \frac{1}{\eta} \tilde{\theta}^T(0) \tilde{\theta}(0) + \rho^2 \int_0^T \varpi^T \varpi dt$$

$$\forall T \in [0, \infty) \quad \varpi \in L_2[0, T] \quad (6-2)$$

式中 加权矩阵 $Q = Q^T \geq 0$;

$$P = P^T \geq 0;$$

$$\eta > 0.$$

2) 如果 $\varpi \in L_\infty [0, \infty)$, 那么跟踪误差是一致最终有界 (UUB)。

① ϖ 表示系统的干扰信号, 在这里表示摩擦建模的最小逼近误差。对于 $\forall T \in [0, \infty)$, $\varpi \in L_2[0, T]$ 空间, 也就是满足如下 L_2 范数定义

$$\|\varpi(t)\|_2 = \left\{ \int_0^\infty \varpi^T(t) \varpi(t) dt \right\}^{1/2} \quad (6-3)$$

对于 $\forall T \in [0, \infty)$, $\varpi \in L_\infty [0, \infty)$ 空间, 也就是满足如下 L_∞ 范数定义

$$\|\varpi(t)\|_\infty = \sup_{t \in [0, \infty)} |\varpi(t)| \quad (6-4)$$

② 如果系统初始点 $\underline{e}(0) = 0$, $\tilde{\theta}(0) = 0$, 那么方程 (6-2) 的 H_∞ 性能指标可以写出如下形式

$$J = \sup_{\varpi \in L_2[0, T]} \frac{\|\underline{e}\|_Q}{\|\varpi\|_2} \leq \rho \quad (6-5)$$

显然, J 描述了干扰信号 ϖ 对系统的破坏最严重时的跟踪误差信号与干扰信号的范数比值。 J 反映了系统对干扰信号 ϖ 的抑制能力, 或称为 L_2 增益。

③ 如果 $\rho = \infty$, 表示系统没有干扰信号的干扰, 系统获得最小跟踪误差。

④ 设 $\underline{e}(t)$ 是动态系统的跟踪误差, U 是 R^n 中的一个紧集, 若对于所有的 $\underline{e}(t_0) = \underline{e}_0 \in U$, 存在一个 $\delta > 0$ 和 $T(\delta, x_0)$ 使 $\|\underline{e}(t)\| < \delta$ 对所有的 $t \geq t_0 + T$ 都成立, 则称跟踪误差是一致最终有界 (UUB)。

6.2.2 摩擦模型与补偿

对于系统 (6-1), 设 $\hat{F}_f$ 是真实摩擦力 F_f 的摩擦估计值, 那么基于摩擦模型补偿的控制律可用下面形式

$$u = m(\ddot{y}_r + k_1 \dot{e} + k_2 e) + \hat{F}_f \quad (6-6)$$

式中 $e = y_r - x$;

$$\dot{e} = \dot{y}_r - \dot{x};$$

y_r ——期望参考输入;

k_1 和 k_2 ——分别表示微分和比例系数。

把式 (6-6) 代入式 (6-1) 得

$$m\ddot{x} = m(\ddot{y}_r + k_1 \dot{e} + k_2 e) + \hat{F}_f - F_f \quad (6-7)$$

经简单整理, 可得到下面误差动态方程为

$$m(\ddot{e} + k_1 \dot{e} + k_2 e) = \tilde{F}_f \quad (6-8)$$

式中 $\tilde{F}_f = F_f - \hat{F}_f$, $\ddot{e} = \ddot{y}_r - \ddot{x}$ 。

如果 $F_f = \hat{F}_f$, 则下面的动态跟踪误差方程可以得到

$$\ddot{e} + k_1 \dot{e} + k_2 e = 0 \quad (6-9)$$

显然, 如果取向量 $K = (k_1, k_2)^T$ 使 $H(s) = s^2 + k_1 s + k_2$ 为 Hurwitz 的, 则有 $\lim_{t \rightarrow \infty} |e(t)| = 0$ 。

从以上分析可以看出, 摩擦力的精确建模对于系统的控制精度是十分重要的。因此, 有关摩擦建模的研究一直非常活跃, 到目前为止, 已提出的摩擦模型有几十种, 主要分为静态摩擦模型和动态摩擦模型两大类。

6.2.3 摩擦模型的辨识与自适应特性

为克服摩擦对伺服系统性能的不利影响, 在摩擦建模及控制补偿技术方面的研究已有近百年的历史, 先后出现了多种摩擦模型。如经典的库仑摩擦 + 黏性摩擦模型、Stribeck 模型、Karnopp 模型、Dahl 模型、LuGre 模型、Leuven 模型、Maxwell - slip 模型等^[2]。其中, LuGre 模型能准确描述摩擦过程中复杂的动静态特性, 如预滑动位移、摩擦记忆、变静摩擦、爬行及静态 Stribeck 曲线。但由于 LuGre 模型的辨识比较困难, 国内对 LuGre 模型的研究还未普遍在实际工业控制中得到应用。

经典的静态摩擦模型由于较为简洁和易于补偿, 应用非常广泛。Stribeck 摩擦模型是一种能较准确地反映摩擦非线性特性的指数摩擦模型, 该模型在不同的摩擦阶段, 摩擦力与速度之间的对应关系, 即为 Stribeck 曲线。Stribeck 摩擦模型的公式表述如下

$$F_f(v) = F_c + (F_s - F_c)e^{-(v/v_s)^\delta} + F_v v \quad (6-10)$$

式中 v ——直线速度;

F_s ——最大静摩擦;

F_v ——库仑摩擦;

v_s ——Stribeck 速度;

δ ——经验参数。

法国格勒诺布尔实验室和瑞典兰德工程技术学院的共同努力, Canudas 等人在 Dahl 模型的基础上提出了 LuGre 模型^[1]。该模型认为刚体表面是通过有弹性的鬃毛接触, 下表面材料的刚度大于上表面。当施加外力时, 由于切向力作用, 鬃毛产生变形, 从而产生摩擦力, 当切向力足够大时, 鬃毛进一步变形, 以致产生滑动。目前, 基于 LuGre 模型的摩擦补偿控制已成为理论和应用研究的一个热点, 设状态量 z 代表接触面鬃毛的平均变形, 则摩擦力可由下面的 LuGre 模型描述, 其表达式如下

$$\begin{cases} F_f = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v \\ \dot{z} = v - \frac{|v|}{g(v)} z \\ \sigma_0 g(v) = F_c + (F_s - F_c)e^{-(v/v_s)^2} \end{cases} \quad (6-11)$$

式中 z ——鬃毛的平均变形;

σ_0 和 σ_1 ——动态摩擦参数;

σ_2 、 F_c 、 F_s 和 v_s ——静态摩擦参数, 其中 F_c 为库仑摩擦;

F_s ——静摩擦;

σ_2 ——黏性摩擦因数;

v_s ——切换速度。

LuGre 动态摩擦模型在稳态时能反映了 Stribeck 现象。

LuGre 摩擦模型的静态参数值与 Stribeck 摩擦模型的摩擦参数一致, 求解静态参数的过程其实就是描绘 Stribeck 曲线的过程。向红标博士在文献 [3] 对摩擦模型的辨识进行了较为详细的描述。其中静态参数的辨识方法是通过拟合这些速度与摩擦力矩之间的曲线来实现的, 一般有描图法、最小二乘法 and 遗传算法等, 这些方法各有其优缺点。描图法最为简捷, 但辨识精度低; 最小二乘法辨识速度快, 精度高, 但信噪比较低时容易造成局部极小; 遗传算法能避免局部极小, 但计算时间长, 要进行大量的无效迭代。

LuGre 摩擦模型的动态参数的辨识较为困难, 因为 LuGre 摩擦模型存在一个无法测量的状态, 即鬃毛的平均变形 z , 目前存在的辨识方法有近似计算、最小二乘和遗传算法等。但理论近似求解的结果偏差较大, 而合理选择输入信号是最小二乘或遗传算法辨识的关键, 若实验数据不够充分, 其辨识结果相应也会不够精确。

LuGre 摩擦模型的优点在于对摩擦环节的动态特性补偿效果较好。其缺点是:

1) 由公式 (6-6) 和文献 [3] 可知模型参数多, 准确辨识其参数十分困难。

2) 实际摩擦力是随着机械系统的磨合等情况在变化, 但 LuGre 模型参数很难实现随着环境的变化而进行动态调整, 也就是说很难实现自适应特性, 故很难保证整个动态历程的模型精度。

3) 用 LuGre 模型形式比较复杂, 很难进行控制系统的设计与稳定性分析。

6.3 摩擦的模糊建模与鲁棒控制

6.3.1 摩擦的自适应模糊建模方法

由上面的 LuGre 摩擦机理模型可知, 建立精确的摩擦模型非常困难, 而且即使得到了较精确的模型, 也会因其复杂的表达形式使得系统分析和基于模型的补偿难于实施。因此, 为了达到控制目的, 我们探讨了用自适应模糊系统对摩擦进行在线建模, 其中自适应模糊系统仅是速度的函数^[4]。但一些专家提出了动态摩擦力可能是位置和速度的函数^[5]。在这里我们进一步扩展该设计, 如果系统式 (6-1) 的状态完全可测, 构造下面的模糊语句:

$$R^{(j)}: \text{If}(x_1 \text{ is } A_1^j) \text{ and } (x_2 \text{ is } A_2^j) \text{ Then } \hat{F}_f \text{ is } B^j \quad (6-12)$$

式中 $\underline{x} = (x_1, x_2)^T = (x, \dot{x})^T$;

A_1^j , A_2^j 和 B^j ——表示模糊集合;

$\hat{F}_f$ ——真实摩擦力的估计值。

在建立了上面的摩擦模糊语句后,采用单值模糊化、乘积运算、加权平均反模糊化可得到下面的摩擦模糊模型

$$\hat{F}_f(\underline{x}) = \frac{\sum_{j=1}^M \bar{F}^j (\prod_{i=1}^2 \mu_{A_i^j}(x_i))}{\sum_{j=1}^M (\prod_{i=1}^2 \mu_{A_i^j}(x_i))} \quad (6-13)$$

式中 $\bar{F}^j$ ——模糊隶属度函数取得最大值的点,也就是 $\mu_{B^j}(F^j) = 1$ 时的点。

若定义下面的模糊基函数

$$\xi_j = \frac{\prod_{i=1}^2 \mu_{A_i^j}(x_i)}{\sum_{j=1}^M (\prod_{i=1}^2 \mu_{A_i^j}(x_i))} \quad (6-14)$$

方程 (6-13) 可以改写成如下的形式

$$\hat{F}_f(\underline{x}) = \underline{\theta}^T \underline{\xi}(\underline{x}) \quad (6-15)$$

式中 $\underline{\theta} = (\bar{F}^1, \bar{F}^2, \dots, \bar{F}^M)^T$, 表示模糊系统的参数向量;

$\underline{\xi}(\underline{x}) = (\xi_1(\underline{x}), \dots, \xi_M(\underline{x}))^T$, 表示模糊基函数向量。

6.3.2 鲁棒控制器设计

由前面的式 (6-12) 至式 (6-15) 的分析过程可以看出,如果摩擦估计 $F_f = \hat{F}_f$, 则可以获得理想的跟踪效果,但实际中获得理想的逼近 $\hat{F}_f = F_f$ 是几乎不可能的。考虑到摩擦建模误差的存在,式 (6-6) 改取如下反馈控制律

$$u = m(\ddot{y}_r + k_1 \dot{e} + k_2 e) + \hat{F}_f - u_c \quad (6-16)$$

式中 u_c ——考虑摩擦建模误差存在的 H_∞ 补偿控制。

把式 (6-16) 代入式 (6-1), 经简单整理后可得如下动态误差方程

$$\dot{\underline{e}} = \underline{A} \underline{e} + \underline{B}(F_f - \hat{F}_f) + \underline{B}u_c \quad (6-17)$$

式中 $\underline{A} = \begin{pmatrix} 0 & 1 \\ -k_1 & -k_2 \end{pmatrix}$;

$$\underline{B} = \begin{pmatrix} 0 \\ 1/m \end{pmatrix}, \quad \underline{e} = \begin{pmatrix} e \\ \dot{e} \end{pmatrix}。$$

定义如下最优参数向量为

$$\underline{\theta}^* = \arg \min_{\underline{\theta} \in \Omega_{\underline{\theta}}} (\sup_{\underline{x} \in \Omega_{\underline{x}}} \|F_f(\underline{x}) - \hat{F}_f(\underline{x}|\underline{\theta})\|) \quad (6-18)$$

定义摩擦建模的最小逼近误差为

$$\varpi = F_f(\underline{x}) - \hat{F}_f(\underline{x}|\underline{\theta}^*) \quad (6-19)$$

那么跟踪误差方程 (6-17) 能改写成如下形式

$$\dot{\underline{e}} = \underline{A} \underline{e} + \underline{B}(\hat{F}_f(\underline{x}|\underline{\theta}^*) - \hat{F}_f(\underline{x}|\underline{\theta})) + \underline{B}u_c + \underline{B} \varpi \quad (6-20)$$

联合式 (6-15), 式 (6-20) 能被改写如下形式

$$\dot{\underline{e}} = \underline{A} \underline{e} + \underline{B} \underline{\xi}^T(\underline{x}) \tilde{\underline{\theta}} + \underline{B}u_c + \underline{B} \varpi \quad (6-21)$$

式中 $\tilde{\theta} = \theta^* - \theta$ 。

定义摩擦逼近误差为

$$\varepsilon = m \ddot{x} - m \ddot{\hat{x}} = (u - \hat{F}_f) - (u - F_f) = F_f - \hat{F}_f \quad (6-22)$$

联合式 (6-15) 和式 (6-19), 可以得到如下逼近误差方程

$$\varepsilon = \varpi + \xi^T(\underline{x}) \tilde{\theta} \quad (6-23)$$

对于反馈控制律 (6-16) 的关键问题是如何设计自适应模糊系统的参数向量和鲁棒补偿控制项 u_c , 下面将用式 (6-23) 的摩擦逼近误差 ε 和跟踪误差 $\underline{e}$ 来调节自适应参数向量^[6], 现将反馈控制律、鲁棒补偿控制项 u_c 以及自适应参数的选取总结如下定理 1。

定理 1. 对于不确定机械系统式 (6-1), 如果采用控制方案如式 (6-24) ~ 式 (6-27), 其中 P 为下列 Riccati 方程 (6-28) 的正定解, 那么系统可获得: ①如果 $\varpi \in L_2[0, \infty)$, 获得鲁棒 H_∞ 跟踪性能指标; ②如果 $\varpi \in L_\infty[0, \infty)$, 那么跟踪误差是一致最终有界。

$$u = m(\ddot{y}_r + k_1 \dot{e} + k_2 e) + \hat{F}_f - u_c \quad (6-24)$$

$$u_c = -\frac{1}{r} B^T P \underline{e} \quad (6-25)$$

$$\hat{F}_f(\underline{x}) = \theta^T \xi(\underline{x}) \quad (6-26)$$

$$\dot{\theta} = \begin{cases} \gamma \xi(\underline{x}) (B^T P \underline{e} + \eta \varepsilon), & \|\theta\| < M_f \\ \gamma \xi(\underline{x}) (B^T P \underline{e} + \eta \varepsilon), & \|\theta\| = M_f (\underline{e}^T B P + \eta \varepsilon) \xi^T(\underline{x}) \theta \geq 0 \\ P[\cdot], & \|\theta\| = M_f (\underline{e}^T B P + \eta \varepsilon) \xi^T(\underline{x}) \theta < 0 \end{cases} \quad (6-27)$$

$$PA + A^T P + Q - \frac{2}{r} PB B^T P + \frac{1}{\rho^2} P B B^T P = 0 \quad (6-28)$$

式中 γ 和 η ——指定的常数, 投影算子为

$$P[\cdot] = \gamma \xi(\underline{x}) (B^T P \underline{e} + \eta \varepsilon) - \gamma \frac{(\underline{e}^T B P + \eta \varepsilon) \xi^T(\underline{x}) \theta}{\|\theta\|^2} \theta \quad (6-29)$$

在实现上述控制器时, 需要注意下面事项:

- 1) 如果自适应方程 (6-27) 中的 $\eta \neq 0$, 意味着摩擦建模误差参与自适应参数的调节。
- 2) 在设计实际控制器上时, 由于 F_f 是未知的, 为了得到摩擦建模误差 ε 信号需要高阶信号 $\ddot{x}$ 是已知条件。但在仿真时, 可以用 $\varepsilon = F_f - \hat{F}_f$ 获得建模误差 ε 信号。
- 3) 实际应用中如果考虑到传感器硬件成本等因素, 可以考虑我们提出的模糊观测器法^[7]或差分法估计高阶信号 $\ddot{x}$ 。

6.4 鲁棒性能指标分析

证明: 取 Lyapunov 函数为

$$V = \frac{1}{2} \underline{e}^T P \underline{e} + \frac{1}{2\gamma} \tilde{\theta}^T \tilde{\theta} \quad (6-30)$$

求 V 的微分, 可推出

$$\dot{V} = \frac{1}{2} \dot{\underline{e}}^T \underline{P} \underline{e} + \frac{1}{2} \underline{e}^T \underline{P} \dot{\underline{e}} + \frac{1}{\gamma} \dot{\tilde{\theta}}^T \tilde{\theta} \quad (6-31)$$

基于式 (6-21), 可以得到

$$\begin{aligned} \dot{V} = & \frac{1}{2} (\underline{e}^T \underline{A}^T \underline{P} \underline{e} - \frac{1}{r} \underline{e}^T \underline{P}^T \underline{B} \underline{B}^T \underline{P} \underline{e} + \tilde{\theta}^T \underline{\xi}(\underline{x}) \underline{B}^T \underline{P} \underline{e} + \varpi^T \underline{B}^T \underline{P} \underline{e} \\ & + \underline{e}^T \underline{P} \underline{A} \underline{e} - \frac{1}{r} \underline{e}^T \underline{P}^T \underline{B} \underline{B}^T \underline{P} \underline{e} + \underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\underline{x}) \tilde{\theta} + \underline{e}^T \underline{P} \underline{B} \varpi) + \frac{1}{\gamma} \dot{\tilde{\theta}}^T \tilde{\theta} \end{aligned} \quad (6-32)$$

上式可进一步写成

$$\begin{aligned} \dot{V} = & \frac{1}{2} \underline{e}^T (\underline{A}^T \underline{P} + \underline{P} \underline{A} - \frac{2}{r} \underline{P} \underline{B} \underline{B}^T \underline{P}) \underline{e} + \underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\underline{x}) \tilde{\theta} \\ & + \frac{1}{2} (\varpi^T \underline{B}^T \underline{P} \underline{e} + \underline{e}^T \underline{P} \underline{B} \varpi) + \frac{1}{\gamma} \dot{\tilde{\theta}}^T \tilde{\theta} \end{aligned} \quad (6-33)$$

由式 (6-28), 上式可改写成

$$\begin{aligned} \dot{V} = & -\frac{1}{2} \underline{e}^T \underline{Q} \underline{e} - \frac{1}{2\rho^2} \underline{e}^T \underline{P} \underline{B} \underline{B}^T \underline{P} \underline{e} + \underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\underline{x}) \tilde{\theta} \\ & + \frac{1}{2} (\varpi^T \underline{B}^T \underline{P} \underline{e} + \underline{e}^T \underline{P} \underline{B} \varpi) + \frac{1}{\gamma} \dot{\tilde{\theta}}^T \tilde{\theta} \end{aligned} \quad (6-34)$$

上式可进一步写成

$$\begin{aligned} \dot{V} = & -\frac{1}{2} \underline{e}^T \underline{Q} \underline{e} - \frac{1}{2} \left(\frac{1}{\rho} \underline{B}^T \underline{P} \underline{e} - \rho \varpi \right)^T \left(\frac{1}{\rho} \underline{B}^T \underline{P} \underline{e} - \rho \varpi \right) + \\ & \frac{1}{2} \rho^2 \varpi^T \varpi + \underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\underline{x}) \tilde{\theta} + \frac{1}{\gamma} \dot{\tilde{\theta}}^T \tilde{\theta} \end{aligned} \quad (6-35)$$

方程 (6-35) 可以改写成下面不等式

$$\dot{V} \leq -\frac{1}{2} \underline{e}^T \underline{Q} \underline{e} + \frac{1}{2} \rho^2 \varpi^T \varpi + \underline{e}^T \underline{P} \underline{B} \underline{\xi}^T(\underline{x}) \tilde{\theta} + \frac{1}{\gamma} \dot{\tilde{\theta}}^T \tilde{\theta} \quad (6-36)$$

由于 $\dot{\tilde{\theta}} = -\dot{\theta}$, 如果自适应方程 (6-27) 中 $\eta=0$, 也就是自适应参数仅由跟踪误差调节, 不等式 (6-36) 可写成下面形式

$$\dot{V} \leq -\frac{1}{2} \underline{e}^T \underline{Q} \underline{e} + \frac{1}{2} \rho^2 \varpi^T \varpi \quad (6-37)$$

如果自适应方程 (6-27) 中 $\eta \neq 0$, 也就是自适应参数由跟踪误差和摩擦建模误差共同调节, 不等式 (6-36) 可写成下面形式

$$\dot{V} \leq -\frac{1}{2} \underline{e}^T \underline{Q} \underline{e} + \frac{1}{2} \rho^2 \varpi^T \varpi + \eta \varepsilon \varpi \quad (6-38)$$

由于最小逼近误差 ϖ 是一很小量, 上面不等式可忽略后面的微小项 $\eta \varepsilon \varpi$, 近似不等式 (6-37)。

从 $t=0$ 到 $t=T$ 对式 (31) 积分可推得

$$V(T) - V(0) \leq -\frac{1}{2} \int_0^T \underline{e}^T \underline{Q} \underline{e} dt + \frac{1}{2} \rho^2 \int_0^T \varpi^T \varpi dt \quad (6-39)$$

因为 $V(T) \geq 0$, 上式可改写如下形式

$$\frac{1}{2} \int_0^T \underline{e}^T \underline{Q} \underline{e} dt \leq \frac{1}{2} \underline{e}^T(0) \underline{P} \underline{e}(0) + \frac{1}{2\gamma} \hat{\underline{\theta}}^T(0) \hat{\underline{\theta}}(0) + \frac{1}{2} \rho^2 \int_0^T \underline{\omega}^T \underline{\omega} dt \quad (6-40)$$

上式就是 H_∞ 性能指标, 即控制目标①达到。

如果 $\underline{\omega} \in L_\infty$, 也就是, $\|\underline{\omega}\| \leq \Xi_\omega$, 其中 $\Xi_\omega > 0$ 。由不等式 (6-37) 可推得

$$\dot{V} \leq -\frac{1}{2} \underline{e}^T \underline{Q} \underline{e} + \frac{1}{2} \rho^2 \underline{\omega}^T \underline{\omega} \leq -\frac{1}{2} \lambda_q \|\underline{e}\|^2 + \frac{1}{2} \rho^2 \Xi_\omega^2 \quad (6-41)$$

式中 λ_q ——矩阵 $\underline{Q}$ 的最小特征值。

对于 $\delta > 0$, 如果选择 $\lambda_q > \left(\frac{\rho^2 \Xi_\omega^2}{\delta^2} \right)$, 那么存在 $\xi > 0$ 使得

$$\begin{aligned} \dot{V} &\leq -\frac{1}{2} \underline{e}^T \underline{Q} \underline{e} + \frac{1}{2} \rho^2 \underline{\omega}^T \underline{\omega} \\ &\leq -\frac{1}{2} \lambda_q \|\underline{e}\|^2 + \frac{1}{2} \rho^2 \Xi_\omega^2 \leq -\xi \|\underline{e}\|^2 < 0 \quad \forall \|\underline{e}\| > \delta \end{aligned} \quad (6-42)$$

上式说明跟踪误差是一致最终有界 (Uniformly ultimately bounded, UUB) 的, 即控制目标②达到。所以从稳定性理论上讲, 自适应参数无论是用单调调节方案还是双调节方案, 系统都能保证良好的稳定性及性能。

6.5 数值结果

把文中所采用的自适应摩擦建模与鲁棒控制方案用于如下运动系统进行试验

$$m \frac{d^2 x}{dt^2} = u - F_f \quad (6-43)$$

步骤 1: 建立自适应模糊系统

定义如下模糊集及隶属函数

$$\begin{aligned} \mu_{A_1^1}(x_i) &= \frac{1}{1 + \exp(5(x_i + 2))}, \mu_{A_1^2}(x_i) = \exp(-(x_i + 1.5)^2) \\ \mu_{A_1^3}(x_i) &= \exp(-(x_i + 0.5)^2), \mu_{A_1^4}(x_i) = \exp(-(x_i - 0.5)^2) \\ \mu_{A_1^5}(x_i) &= \exp(-(x_i - 1.5)^2), \mu_{A_1^6}(x_i) = \frac{1}{1 + \exp(-5(x_i - 2))} \end{aligned}$$

建立摩擦模糊规则库

$$R^{(j)}: \text{If } (x_1 \text{ is } A_1^i) \text{ and } (x_2 \text{ is } A_2^j) \text{ Then } \hat{F}_f(\underline{x}) \text{ is } B^j \quad (j=1, 2, \dots, 6).$$

可得到估计摩擦力 $\hat{F}_f = \underline{\theta}^T \underline{\xi}(\underline{x})$, 其实现如下

$$\begin{cases} \underline{\xi}(\underline{x}) = \frac{1}{S} (\mu_{A_1^1} \mu_{A_2^1}, \dots, \mu_{A_1^6} \mu_{A_2^6})^T \\ \underline{\theta} = (\theta_1, \theta_2, \dots, \theta_6)^T \\ S = \sum_{j=1}^6 \prod_{i=1}^2 \mu_{A_i^j}(x_i) \end{cases} \quad (6-44)$$

步骤 2: 系统参数选择

$m = 1\text{kg}$, $\gamma = 0.1$, $k_1 = 2$, $k_2 = 4$, $r = 0.08$, $\eta = 100$, $F_N = 5\text{N}$, $F_v = 0.04F_N$, $v_s = 1$, $\delta_s = 1$ 。设参考输入为 $y_r(t) = \sin(t)$, 系统状态初始值选择为 $x_1(0) = x_2(0) = 1.2$, 自适应参数初始值选择为 $\underline{\theta} = (-8, -6, -1, 3, 8, 1)$ 、为 $(-10, 10)$ 和 $(0, 10)$ 之间的随机数。

选择 $Q = \begin{pmatrix} 20 & 15 \\ 15 & 30 \end{pmatrix}$, 解 Riccati 方程 (6-28) 可得 $P = \begin{pmatrix} 15 & 5 \\ 5 & 5 \end{pmatrix}$ 。

步骤3: 控制方程及自适应律

$$\begin{cases} u = (-\sin(t) + 2e + 4\dot{e}) + \hat{F}_f - u_c \\ u_c = -\frac{5}{r}(e + \dot{e}) \\ \dot{\underline{\theta}} = \gamma \underline{\xi}(\underline{x})(5e + 5\dot{e} + \eta e) \end{cases} \quad (6-45)$$

6.5.1 传统摩擦模型的自适应特性

在这个例子中, 我们将验证传统摩擦模型作为控制补偿项的自适应特性。假设方程式 (6-43) 中的初始真实摩擦力 $F_f = F_c + (F_s - F_c)e^{-|v/v_s|^{\delta_s}} + F_v v$ 为 Stribeck 摩擦力, 并且该摩擦力的参数和结构随系统的磨合等环境因素在逐渐变化。这一物理现象如表 6-1 分两种情况: 环境 1 表示真实摩擦力 F_f 的参数在三个不同时间段的参数变化, 但模型结构没有变化。环境 2 表示真实摩擦力 F_f 不但参数变化, 模型结构逐渐演变为 Coulomb 摩擦力。

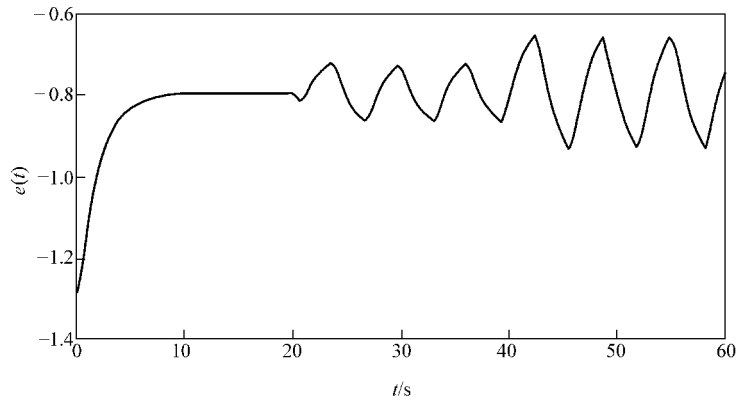
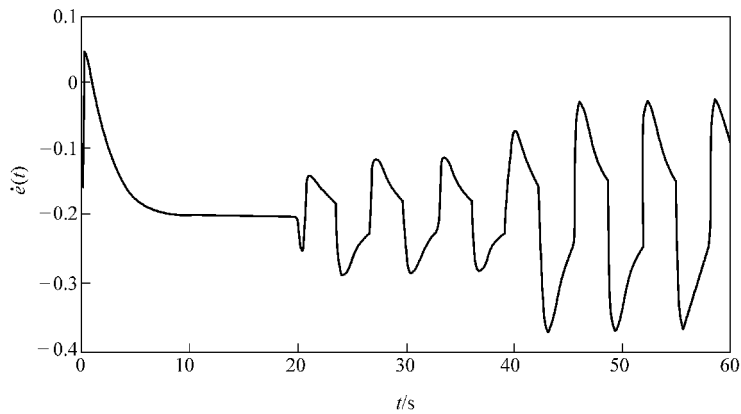
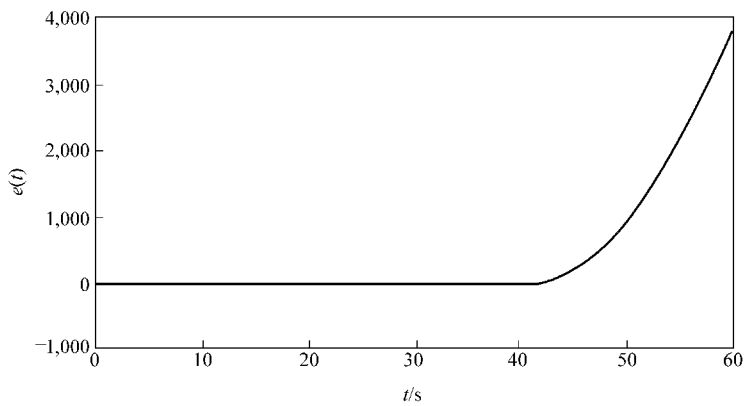
表 6-1 摩擦模型参数与结构

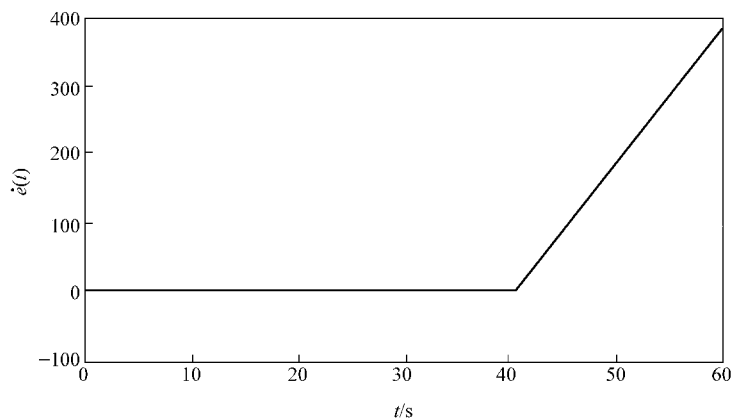
F_f	$0 < t \leq 20$	$20 < t \leq 40$	$40 < t \leq 60$
	Stribeck	Stribeck	Stribeck
环境 1	$F_c = 0.40F_N$	$F_c = 0.30F_N$	$F_c = 0.20F_N$
	$F_s = 0.60F_N$	$F_s = 0.50F_N$	$F_s = 0.40F_N$
	Stribeck	Stribeck	Coulomb
环境 2	$F_c = 0.40F_N$	$F_c = 0.30F_N$	$F_c = 0.20F_N$
	$F_s = 0.60F_N$	$F_s = 0.50F_N$	

图 6-1 和图 6-2 分别是系统跟踪误差向量 $\underline{e}$ 在摩擦环境 1 和环境 2 随时间响应的曲线。该结果是控制器取 $u = m(\ddot{y}_r + k_1\dot{e} + k_2e) + \hat{F}_f$ 时获得的, 其中控制器中的摩擦力 $\hat{F}_f$ 估计用 Stribeck 摩擦模型, 但其参数在 $0 < t \leq 60$ 过程中始终为初始值 $F_c = 0.40F_N$ 和 $F_s = 0.60F_N$ 。图 6-1 说明如果用传统的摩擦力模型进行控制补偿时, 控制器的摩擦力如果不随真实摩擦力的参数环境变化而改变, 将使系统的跟踪性能变差。图 6-2 则说明真实摩擦力从一种结构变换成另一种结构, 而控制器保持原初始补偿摩擦模型产生了不稳定现象。

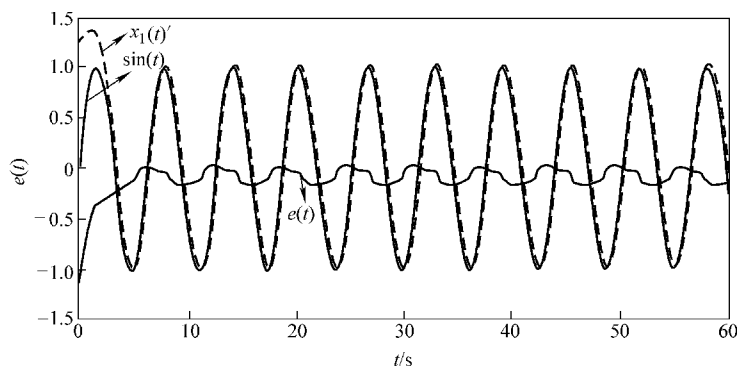
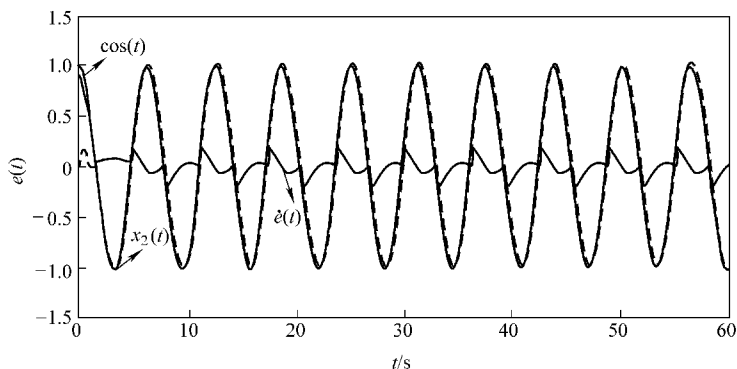
6.5.2 单参数调节摩擦模糊模型的自适应特性

在这个例子中, 我们将验证单参数自适应模糊系统作为摩擦新模型所取得的跟踪控制效果。两种仿真环境和前面叙述的相同。图 6-3 是在摩擦环境 1 情况下, 也就是真实摩擦力的

a) 误差 e 的响应曲线b) 误差 $\dot{e}$ 的响应曲线图 6-1 系统跟踪误差 e 的响应曲线 (环境 1)a) 误差 e 的响应曲线图 6-2 系统跟踪误差 e 的响应曲线 (环境 2)

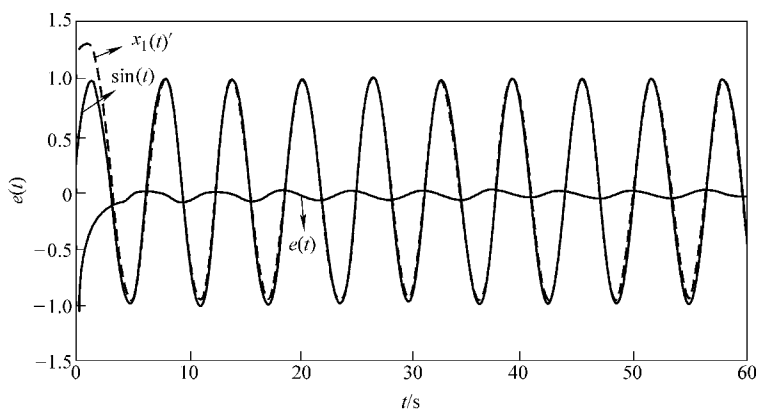
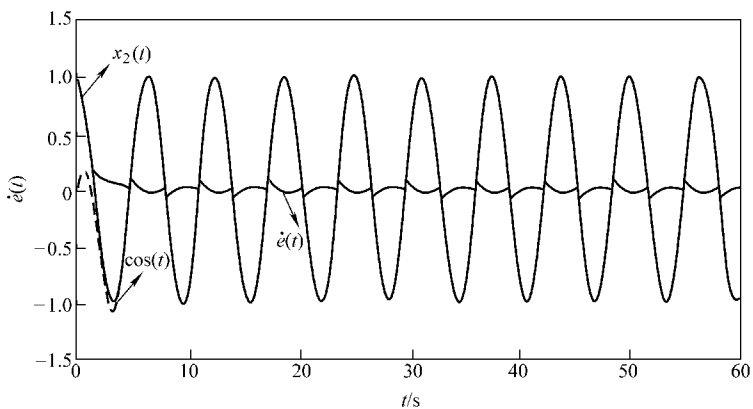
b) 误差 $\dot{e}$ 的响应曲线图 6-2 系统跟踪误差 e 的响应曲线 (环境 2) (续)

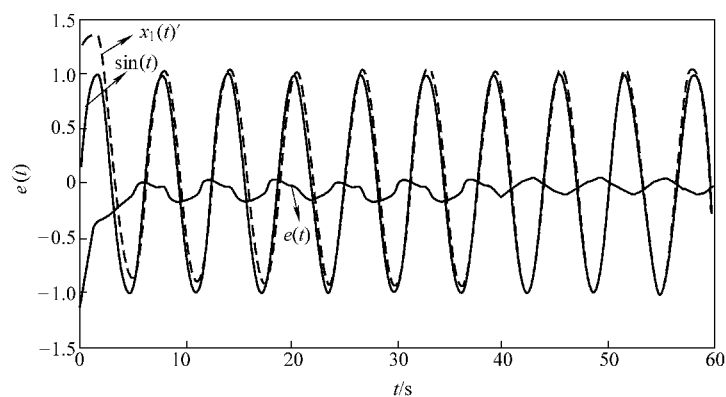
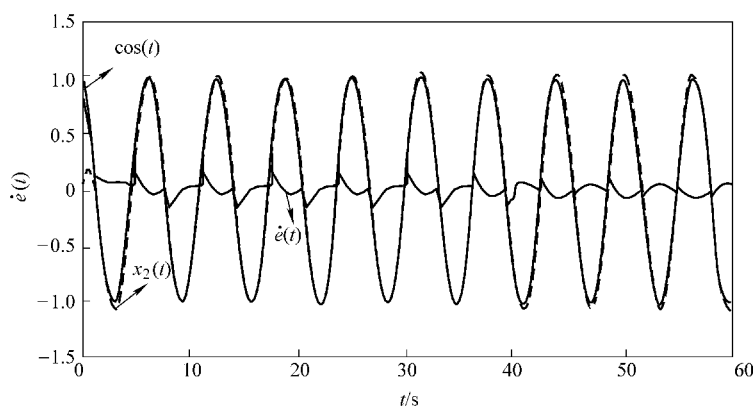
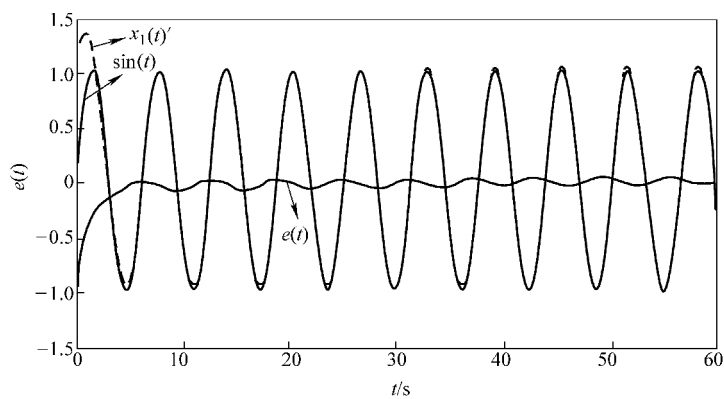
参数在变化, 系统跟踪误差随时间的响应曲线。该结果是控制器 $u = m(\ddot{y}_r + k_1 \dot{e} + k_2 e) + \hat{F}_f - u_c$ 时获得的, 其中控制器中的摩擦力 $\hat{F}_f$ 采用单参数自适应模糊系统。图 6-3 是自适应参数仅由跟踪误差 e 进行调节, 而建模误差 ε 没参与调节形成的。在图 6-3 的仿真中, 控制器的鲁棒补偿控制项的 u_c 的 $r=0.25$, 也就是对应着 Riccati 方程 (6-28) 的 ρ 相对较小的情况, 也就是干扰信号 ω 对系统的性能干扰较大。

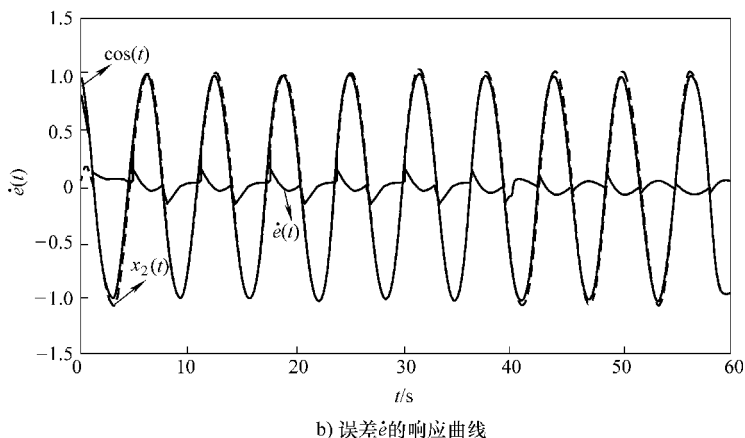
a) 误差 e 的响应曲线b) 误差 $\dot{e}$ 的响应曲线图 6-3 系统跟踪误差 e 的响应曲线 (环境 1)

在图 6-3 的这个仿真例子中，我们验证了单参数自适应模糊系统作为摩擦新模型所取得的控制效果，其中的控制器的鲁棒补偿控制项 u_c 的 $r=0.25$ 。在图 6-4 这个仿真例子中，仍然采用单参数自适应模糊系统作为摩擦新模型，但控制器的鲁棒补偿控制项 u_c 的 $r=0.08$ ，也就是对应着 Riccati 方程 (6-28) 的 ρ 为较大的情况，也就是干扰信号 ω 对系统性能干扰较小。从这两个例子可以看出，单参数自适应模糊系统作为摩擦新模型在干扰信号 ω 较小的情况下取得了好的跟踪性能，但在较大干扰信号 ω 时，不是很理想，但比传统摩擦模型要好。

在下面的图 6-5 和图 6-6 两个仿真例子里，我们将进一步验证单参数自适应模糊系统作为摩擦新模型，在真实摩擦模型的结构和参数都变化情况下，系统跟踪误差随时间的响应曲线。其控制器设计和自适应参数调整和上面图 6-3 和图 6-4 完全相同。图 6-5 和图 6-6 的例子进一步说明，即使真实摩擦力的结构发生变化，我们采用的摩擦单参数自适应模糊建模技术和鲁棒控制方案也能保证系统的跟踪性能和稳定性。

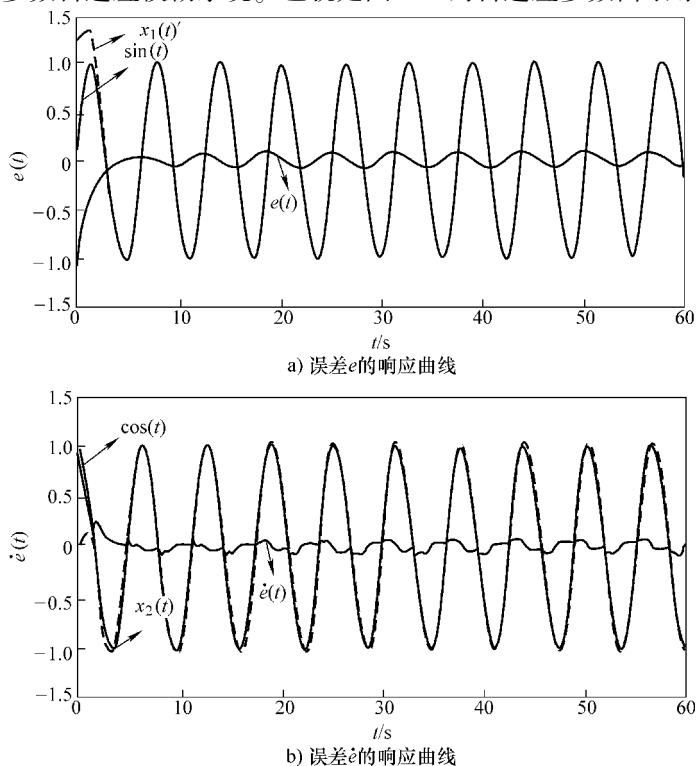
a) 误差 e 的响应曲线b) 误差 $\dot{e}$ 的响应曲线图 6-4 系统跟踪误差 e 的响应曲线 (环境 1)

a) 误差 e 的响应曲线b) 误差 $\dot{e}$ 的响应曲线图 6-5 系统跟踪误差 e 的响应曲线 (环境 2)a) 误差 e 的响应曲线图 6-6 系统跟踪误差 e 的响应曲线 (环境 2)

图 6-6 系统跟踪误差 e 的响应曲线（环境 2）（续）

6.5.3 双参数调节摩擦模糊模型的自适应特性

在接下来的仿真例子中，验证双参数自适应模糊系统作为摩擦新模型所取得的控制效果。两种仿真环境和前面叙述的相同。图 6-7 是在摩擦环境 1 情况下，系统跟踪误差随时间的响应曲线。该结果是控制器 $u = m(\ddot{y}_r + k_1 \dot{e} + k_2 e) + \hat{F}_f - u_c$ 时获得的，其中控制器中的摩擦力 $\hat{F}_f$ 采用双参数自适应模糊系统。也就是图 6-7 的自适应参数由跟踪误差 e 和建模误差

图 6-7 系统跟踪误差 e 的响应曲线（环境 1）

ε 共同调节形成的。在图 6-7 的仿真中, 控制器的鲁棒补偿控制项的 u_c 的 $r=0.25$, 也就是对应着 Riccati 方程 (6-28) 的 ρ 为较小的情况, 也就是干扰信号 ω 对系统的性能干扰较大。

在图 6-8 这个仿真例子中, 仍然采用双参数自适应模糊系统作为摩擦新模型, 但控制器的鲁棒补偿控制项 u_c 的 $r=0.08$, 也就是对应着 Riccati 方程 (6-28) 的 ρ 为较大的情况, 也就是干扰信号 ω 对系统的性能干扰较小。从这两个例子可以看出, 双参数自适应模糊系统比单参数自适应模糊系统取得了更好的跟踪性能。

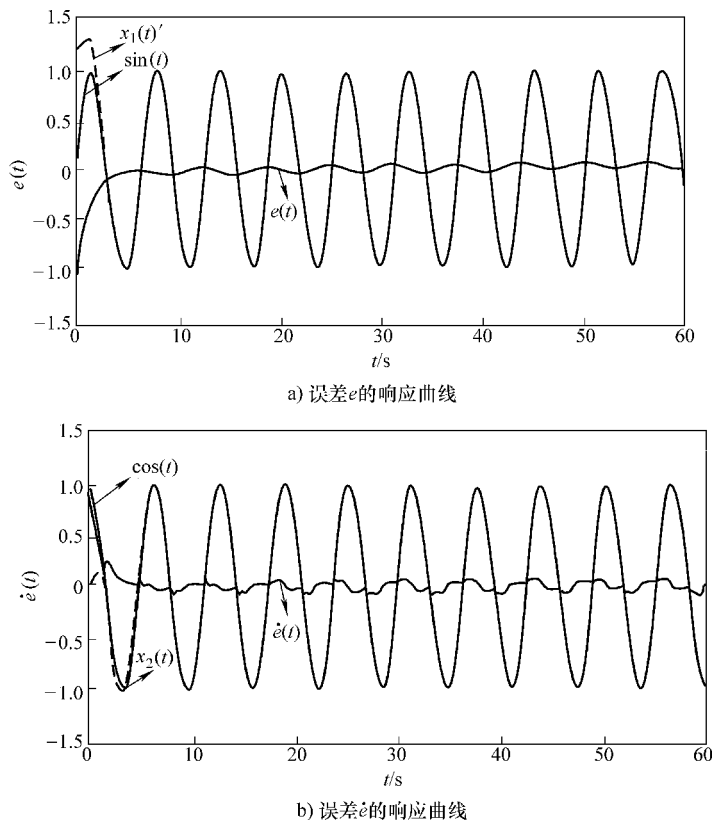
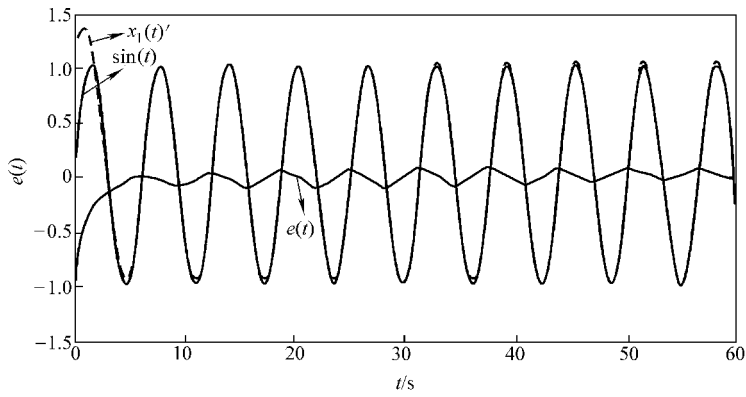
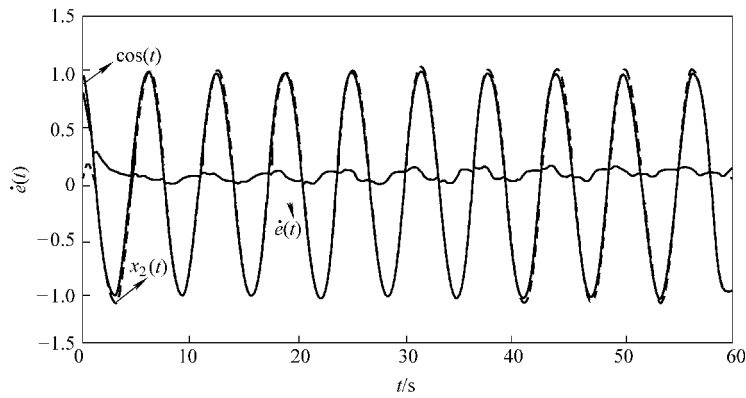
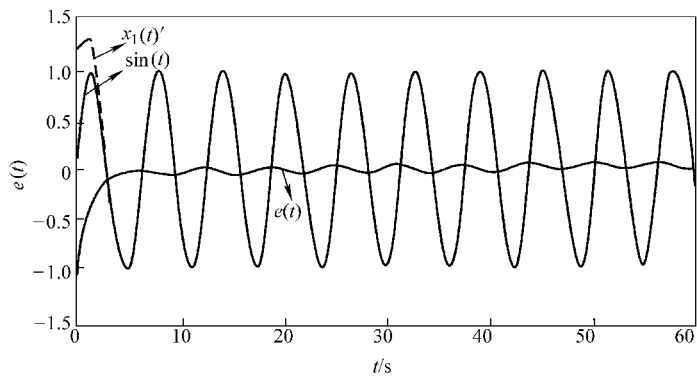
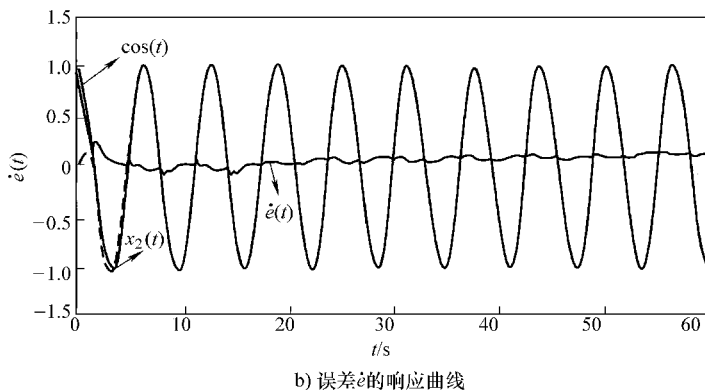


图 6-8 系统跟踪误差 e 的响应曲线 (环境 1)

在下面的图 6-9 和图 6-10 两个仿真例子里, 我们将进一步验证双参数自适应模糊系统作为摩擦新模型, 在真实摩擦模型的结构和参数都变化的情况下, 系统跟踪误差随时间响应的曲线。其控制器设计和自适应参数调整和图 6-7 和图 6-8 完全相同。图 6-9 和图 6-10 的例子进一步说明, 即使真实摩擦力的结构发生变化, 我们采用的摩擦双参数自适应模糊建模技术和鲁棒控制方案也能保证系统的跟踪性能和稳定性。图 6-3 ~ 图 6-6 和图 6-1 ~ 图 6-2 相比较说明, 采用单参数自适应模糊系统作为摩擦新模型比采用传统摩擦模型要取得更好的跟踪性能, 即使实际摩擦力的结构发生变化也能保证系统的稳定性。图 6-7 ~ 图 6-10 和图 6-3 ~ 图 6-6 相比较说明, 采用双参数自适应模糊系统作为摩擦新模型比单参数自适应模糊系统作为摩擦新模型要取得更好的跟踪性能。

a) 误差 e 的响应曲线b) 误差 $\dot{e}$ 的响应曲线图 6-9 系统跟踪误差 e 的响应曲线（环境 2）a) 误差 e 的响应曲线图 6-10 系统跟踪误差 e 的响应曲线（环境 2）

图 6-10 系统跟踪误差 $\dot{e}$ 的响应曲线（环境 2）（续）

注 1: 从现有文献看^[8-10], 自适应参数式 (6-27) 的界都是在 $\|\underline{\theta}\| \leq M_f$ 情况下给定的, 这对于连续系统是正确的。但计算机数值计算 (ODE45) 完全有可能使 $\|\underline{\theta}\| > M_f$ 的情况发生, 故在数值计算时应考虑这种情况。在仿真时, 方程 (6-27) 中的 $\|\underline{\theta}\| = M_f$ 情况应改成 $\|\underline{\theta}\| \geq M_f$, 其他保持不变。仿真时把自适应参数拓展为状态空间的增广。

6.6 本章小结

1) 我们设计了一类不确定机械系统的双调节自适应模糊建模与鲁棒控制方案。为了克服传统摩擦模型缺乏准确性和自适应特性的弊端, 采用了自适应模糊建模技术在线逼近摩擦力并将辨识结果用在鲁棒控制器补偿中, 其中的自适应参数是基于李亚普诺夫稳定性理论设计的。

2) 摩擦模糊模型的自适应参数考虑了两种调节方案, 仅由跟踪误差调节, 或者由跟踪误差与摩擦建模误差共同调节。仿真实例验证了建立自适应模糊摩擦模型并进行控制补偿的有效性, 另外, 也验证了自适应参数引入摩擦建模误差后使得系统取得更好的跟踪性能。

参考文献

- [1] CANUDAS D W C, OLLSON H, ASTROM K J, et al. A new model for control of systems with friction [J]. IEEE Trans. on Automatic Control, 1995, 40 (3): 419-425.
- [2] ARMSTRONG B, CANUDA D W C. A survey of models, analysis tools and compensation methods for control of machines with friction [J]. Automatica, 1994, 30 (7): 1083-1138.
- [3] 向红标, 裘祖荣, 李醒飞. 开放式伺服系统的摩擦建模与补偿研究 [D]. 天津: 天津大学, 2010.
- [4] WANG Yongfu, WANG Dianhui, CHAI Tianyou. Extraction and adaptation of fuzzy Rules for friction modeling and control compensation [J]. IEEE Trans. On Fuzzy Systems, 2011, 19 (4): 682-693.
- [5] SELMIC R R, LEWIS F L. Neural-network approximation of piecewise continuous functions: application to friction compensation [J]. IEEE Trans. on Neural Networks, 2002, 13 (3): 745-751.
- [6] 王永富, 张义民, 柴天佑. 一类不确定机械系统的双调节自适应模糊建模与鲁棒控制研究 [J]. 机械

工程学报, 2013, 49 (9): 61 – 68. .

- [7] 王永富, 王殿辉, 柴天佑. 基于状态估计的摩擦模糊建模与鲁棒自适应控制 [J]. 自动化学报, 2011, 37 (2): 245 – 252.
- [8] WANG Lixin. Stable adaptive fuzzy control of nonlinear systems [J]. IEEE Trans. on Fuzzy Systems, 1993, 1 (1): 146 – 155.
- [9] CHEN Borsen. H_∞ tracking design of uncertain nonlinear SISO systems: Adaptive fuzzy approach [J]. IEEE Trans. on Fuzzy Systems, 1996, 4 (1): 32 – 43.
- [10] MEHRDAD H. Hybrid adaptive fuzzy identification and control of nonlinear systems [J]. IEEE Trans. on Fuzzy Systems, 2002, 10 (2): 198 – 210.

第 7 章 非线性摩擦诱发的自激振动与主动控制

7.1 自激振动

自激振动是一种由系统本身产生的激励所维持的非线性振动。自激振动系统除具有振动元件外，还具有非振荡性的能源、调节环节和反馈环节。因此，不存在外部激励时它也能产生一种稳定的周期性振动，维持自激振动的交变力是由运动本身产生的，并且由反馈和调节环节控制。运动本身一旦停止，其交变力随之消失，自激振动也随即停止^[1]。在自然界、人们的日常生活中、物理学和工程界中，自激振动的实例多种多样，但其本质都是一样的。

自激振动系统由能源（或驱动原动件）、控制调节环和振动体三部分组成。系统不需要周期性的外部激励，并且一旦切断能源供应，系统的振动便立即消失。系统的自激振动由能源系统供给的能量来维持，但这种能源不一定是振荡性的。调节环将振动体的交变运动量变换为交变力并反馈回振动体，以使振动体为此持续的振动。

自激振动系统的振幅和频率，完全取决于系统本身的参数和调节环的调节特性。当调节环子系统的非线性不明显时，其振动的频率接近于振动体线性系统的某个固定频率，其振动波形也接近于正弦波。但随着调节特性非线性的增加，无论是频率还是波形都逐渐离开近似的线性系统。当非线性特性很明显时，振动呈现张弛振动形式。此时，自激振动的一个周期将分成完全不同的几个运动阶段，在各个阶段的边界上，振动体的位移和速度将产生某种跃变。如机床滑动部件的爬行现象，就是一种短暂停顿与突然跳跃的运动过程。

7.2 摩擦诱发的自激振动

摩擦自激振动就是与摩擦有关或由摩擦引起的自激振动。在自然界和工程界，滑动摩擦是一种极其普遍存在的现象，绝大多数自激振动也都是由滑动摩擦引起。例如，各种人造飞行器（宇航飞船与飞机）、各种机动车辆、各种机械设备、各种机械加工制造过程出现的颤振几乎都和滑动摩擦有关。因此，自激振动家族的最主要成员非摩擦自激振动莫属。摩擦自激振动产生根本原因一直是专家关注的问题，如文献 [2-3] 给出负阻尼理论解释这一现象，但文献 [1] 认为这一理论是有缺陷的，很可能和实验结果相违背。由于该问题的复杂性使得摩擦诱发振动的真实原因还处在进一步探索中。

7.2.1 一个典型的数学模型

机械系统的弹性体受到摩擦力的作用有可能产生自激振动。下面以图 7-1 的典型平动刚体为研究对象，建立系统运动微分方程^[4-5]

$$mx''(\tau) + kx(\tau) = F_f(v) + u_c \quad (7-1)$$

式中 m ——滑块质量;
 x ——滑块位移;
 v_0 ——传动带速度;
 $F_f(v)$ ——传动带给滑块的摩擦力;
 u_c ——外加主动控制力。

为便于问题的描述,用下式建立无因次归一化时间坐标

$$t = \omega_0 \tau, \quad \omega_0 = \sqrt{\frac{k}{m}} \quad (7-2)$$

将 x 对 t 的一次和二次导数记作 $\dot{x}$ 和 $\ddot{x}$, 结合方程 (7-2) 有下式成立

$$\begin{cases} \dot{x} = \frac{d(x)}{dt} = \frac{x'}{\omega_0} \\ \ddot{x} = \frac{d(\dot{x})}{dt} = \frac{x''}{\omega_0^2} \end{cases} \quad (7-3)$$

利用方程 (7-2) 和方程 (7-3) 可将运动方程 (7-1) 简化成下列形式

$$\ddot{x}(t) + x(t) = F(v) + u \quad (7-4)$$

式中 $v = v_0 - \omega_0 \dot{x}$;
 $F(v) = F_f(v)/k$;
 $u = u_c/k$ 。

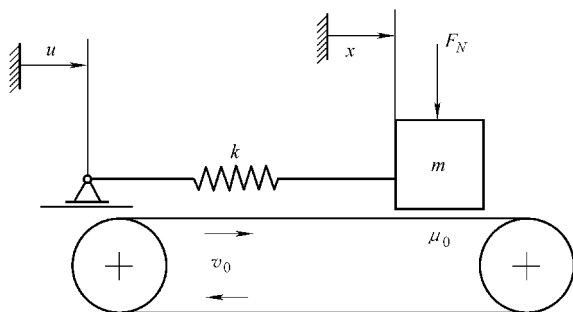


图 7-1 摩擦诱发自激振动的简化力学模型图

7.2.2 摩擦自激振动解析解的争议

如果外加主动控制力 $u=0$, 那么广义摩擦力 $F(v)$ 的模型性质将决定系统式 (7-4) 如何产生自激振动和自激振动的形式。文献 [2, 3] 提出负阻尼理论来解释摩擦自激振动成因, 在文献 [2] 中假设式 (7-4) 中的摩擦力是如下形式

$$F(v) = F_0 + \gamma \dot{x} \quad (7-5)$$

式中 F_0 ——速度为零时的摩擦力;
 γ ——摩擦力曲线的斜率。

将方程 (7-5) 代入式 (7-4), 可得下式

$$\ddot{x}(t) - \gamma \dot{x} + x(t) = F_0 \quad (7-6)$$

上面方程的解为

$$x(t) = ae^{-i\Omega t} + b \quad (7-7)$$

式中 a 和 b ——常数;

$$\Omega = \pm \sqrt{(k/m) - (1/4m^2)\gamma^2} + i\gamma/2m。$$

另外,文献[3]进一步忽略了上方程中 F_0 项,使得求解更为简化。从而用负阻尼效应解释摩擦自激振动成因及其形式。

但是,文献[1]认为这一理论存在缺陷,有可能和实验结果相违背。造成负阻尼理论和实际相矛盾的关键在于两方面:一是方程中的常数力项(如 F_0)被不合理地抛弃了;二是忽略了静摩擦力与动摩擦力之间差值的影响。

最近,文献[5]给出了另一种解释并给出了摩擦产生极限环振荡的条件,该文将摩擦力 F 在 v_0 点进行泰勒展开

$$F - F(v_0) = -F_v^{(1)}x(t) + \frac{F_v^{(2)}}{2}\dot{x}^2(t) - \frac{F_v^{(3)}}{6}\dot{x}^3(t) \quad (7-8)$$

式中 $F_v^{(i)} = \left. \frac{d^i F}{dv^i} \right|_{v=v_0}$, $i=1, 2, 3$ 。

将上面方程代入式(7-4)中,可得下式

$$\ddot{x}(t) + x(t) + F_v^{(1)}x(t) - \frac{F_v^{(2)}}{2}\dot{x}^2(t) + \frac{F_v^{(3)}}{6}\dot{x}^3(t) = 0 \quad (7-9)$$

上面方程可转化成如下方程求解

$$\ddot{y}(v) + \alpha \dot{y}(v) + y(v) - \beta \dot{y}^2(v) + \gamma \dot{y}^3(v) = 0 \quad (7-10)$$

式中 $\omega = \sqrt{\frac{k}{m}}$

$$v = \omega t$$

$$\alpha = \frac{F_v^{(1)}}{\sqrt{km}}$$

$$y(v) = x(v) / x_0$$

$$\beta = \frac{F_v^{(2)}}{2mk} F(v_0)$$

$$\gamma = \frac{F_v^{(3)}}{6(mk)^{3/2}} F^{(2)}(v_0)$$

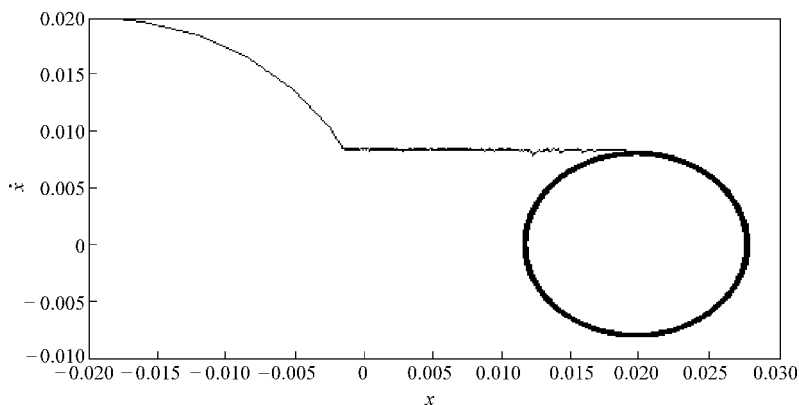
通过求解上面的微分方程可获得系统的响应,并进一步给出了极限环产生的条件。

通过上面的分析可以看出,从微分方程的解析解给出的摩擦自激振动成因是在假设摩擦力满足某特定条件下进行的。这种分析有一定的理论意义,但正如文献[1]所指出的,有可能和实验结果相违背或有些偏差,原因在于摩擦力模型的假设与简化都是为了能给出解析解,忽略了实际物理意义。

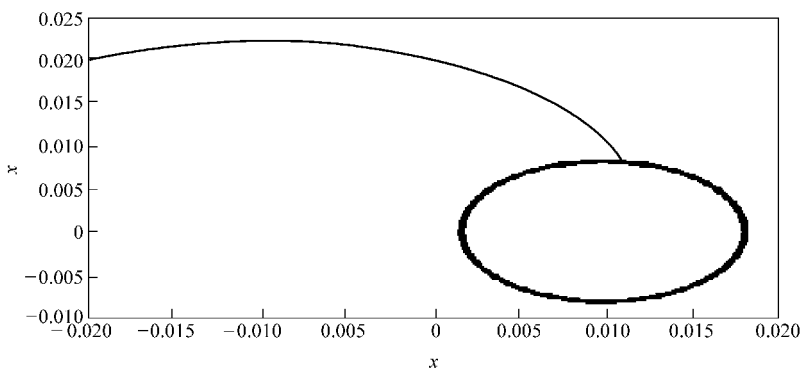
7.2.3 摩擦自激振动的数值分析

从摩擦文献可知,到目前为止已提出的摩擦模型有几十种^[6],这些摩擦模型是否都能产生自激振动,以及什么条件下产生极限环振荡一直是人们关心的问题。一种办法是按照上面的思路对这些模型进行某种假设,从而给出解析解。另一种方法是从数值解角度考虑,从而达到控制目的。下面将方程(7-4)中的广义控制力取零($u=0$),分析系统在一定时间的响应。参考文献[2],系统参数取值: $m=0.60\text{kg}$, $k=763\text{N/m}$, $F_N=15.0\text{N}$, $v_0=0.3\text{m/s}$, $\mu_0=0.25$ 。

Matlab 中的四阶五级 Fehlberg - Runge - Kutta 方法为工程技术人员求解非线性微分方程提供了强有力的工具。下面用该方法 (ODE45) 给出两类摩擦振动问题的数值分析,通过 Matlab 的 ODE45 求解,可得到图 7-2 和图 7-3 的数值图解。从图可以看出,这两种摩擦力模型都产生了极限环,一个是 Coulomb 模型



a) 参数 $F_c=15$ 的相平面曲线



b) 参数 $F_c=7.5$ 的相平面曲线

图 7-2 Coulomb 摩擦诱发的振动极限环

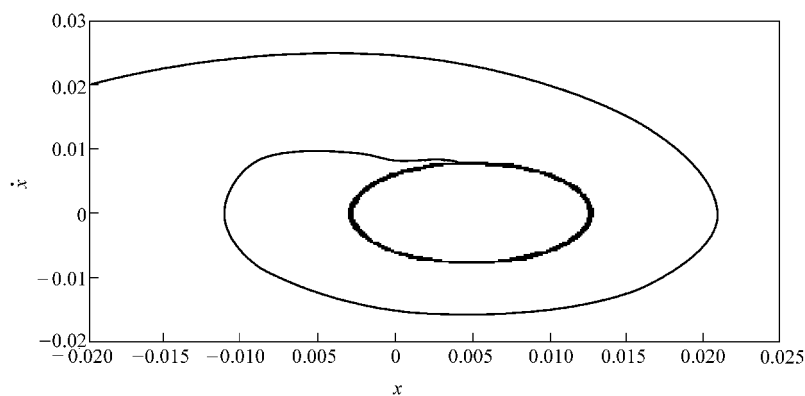
c) 参数 $F_c=3.75$ 的相平面曲线

图 7-2 Coulomb 摩擦诱发的振动极限环 (续)

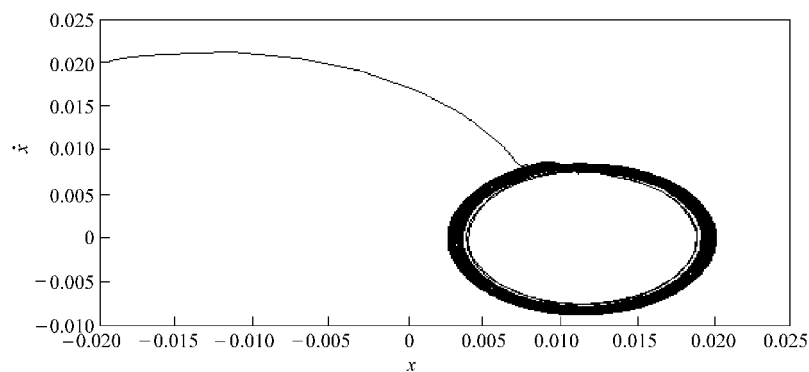
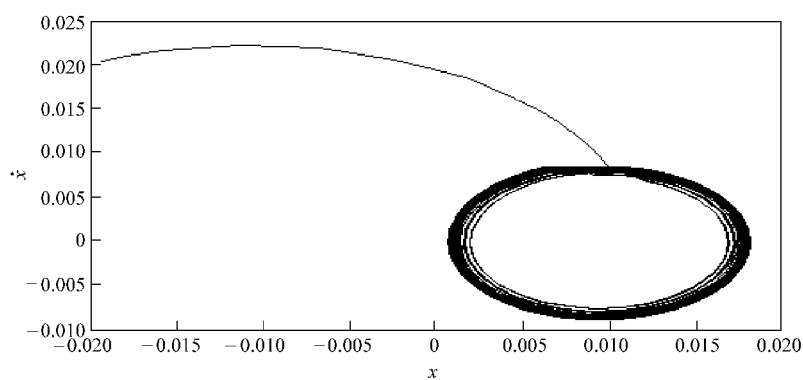
a) 参数 $F_c=7.5, F_s=9, F_v=0.6$ 的相平面曲线b) 参数 $F_c=6, F_s=7.5, F_v=0.6$ 的相平面曲线

图 7-3 Coulomb 摩擦诱发的振动极限环

$$F_f = F_c \operatorname{sgn}(v) \quad (7-11)$$

式中 F_f ——摩擦力;
 v ——相对滑动速度;
 F_c ——库仑摩擦力;

$\text{sgn}(v)$ ——符号函数。

另一个是 Stribeck 摩擦模型

$$F_f = F_c + (F_s - F_c) e^{-|v/v_s|^{\delta_s}} + F_v v \quad (7-12)$$

式中 F_s ——最小静摩擦力；

F_v ——黏性摩擦因数；

v_s ——Stribeck 速度；

v_s 、 δ_s ——经验常数。

该模型虽然是静态摩擦模型，但它考虑了 Stribeck 效应，该模型对摩擦现象描述更为精确。

7.3 摩擦自激振动的主动控制

一个系统除应满足特定功能外，还应考虑两个重要的动力学问题，即系统的稳定性和动力响应。一个正常的系统应是稳定的，且其动力响应必须限定在可接受的范围内，有时要求其振荡愈小愈好。由于被动控制的局限性，使得振动主动控制技术正越来越受到人们的重视，并已被广泛应用于精密仪表、航空航天、交通运输和机械工程等领域。主动控制是指在振动控制过程中，根据传感器所检测到的反馈信号，应用一定的主动控制策略，经过实时计算，进而驱动执行器对控制目标施加一定的影响，从而达到抑制或消除振动的目的。

近些年，一些文献对摩擦诱发振动进行了主动控制的研究^[7-8]。但基本上都采用传统的主动控制技术，很少用智能控制方法研究这一问题。近年来理论和实际工程应用表明，神经网络和自适应模糊系统等智能方法，对不确定非线性系统进行智能建模与控制具有有效性和实用性^[9-12]。它的根源在于智能建模与控制的基本出发点是仿人智能，以实现对复杂不确定性系统进行有效的建模与控制，它具有从环境自学习、自适应环境的能力。因此，智能建模与控制技术应是解决机械系统非线性问题一条新的途径。特别是对摩擦以及摩擦诱发的自激振动这类不确定性问题更具有重要的理论和实际意义。

7.3.1 模糊神经网络建立摩擦模型

神经网络具有并行计算、分布式信息存储、容错能力强，及具备自适应学习功能等一系列优点。但神经网络不适合于表达基于规则的知识，因此在对神经网络进行训练时，由于不能很好利用已有的经验知识，常常只能将初始权值取为零或随机数，从而增加了网络的训练时间或陷入非要求的局部极值。模糊逻辑也是一种处理不确定性和非线性以及其他不适应性问题的有力工具。它比较适于表示那些模糊或定性的知识，其推理方式比较类似于人的思维模式。但是一般说来它不容易实现自适应学习的功能。若将模糊逻辑与神经网络适当地结合起来，吸取两者的长处，则可组成比单纯的神经网络或模糊逻辑性能更好的系统。

模糊模型的表示方法主要有两种：一种是模糊规则后件为输出量的一个模糊集合，如 NB 、 PB 等；另一种是模糊规则后件为输入语言变量的函数，典型的情况是输入变量的线性组合。由于该模型表示是 Takagi 和 Sugeno 首先提出的，因此通常称之为模糊系统的 T-S 模

糊模型^[13]。针对摩擦具体问题,我们构造下面类似 T-S 模型的模糊语句如下

$$R^{(j)}: \text{If } \dot{x} \text{ is } A_j^{\dot{x}} \text{ Then } \hat{F} \text{ is } w_j \quad (7-13)$$

式中 $A_j^{\dot{x}}$ ——输入变量 $\dot{x}$ 的模糊集合;

w_j ——第 j 条规则对应的输出。

根据上面给出的模糊语句,可设计出如图 7-4 所示的 3 层模糊神经网络结构^[14]。

第一层(输入层):该层由单个结点构成,该结点直接与输入变量 $\dot{x}$ 相连接,它起着将输入值 $\dot{x}$ 传送到下一层的作用,即

$$\hat{h}^{(1)} = e^{(1)} = \dot{x} \quad (7-14)$$

第二层(隶属度函数层):该层由 N 个结点构成,每个结点代表输入变量 $\dot{x}$ 的一个语言变量值,如 A_1 、 A_2 等。它的作用是计算输入分量属于各语言变量值模糊集合的隶属度函数,即

$$\hat{h}_j^{(2)} = \mu_j(e_j^{(2)}) = \exp^{-\frac{(x_i - b_j)^2}{(\sigma_j)^2}} \quad (7-15)$$

式中 μ ——隶属度函数,这里采用高斯型函数。

第三层(输出层):该层由单个结点构成,该结点直接与输出变量 $\hat{F}$ 相连接,它计算系统总的输出量,即

$$\hat{F} = \hat{h}^{(3)} = \sum_{j=1}^N w_j \times e_j^{(3)} \quad (7-16)$$

式中 w_j ——加权因子。

在完成上面的模糊神经网络设计后,将用下面的数据挖掘技术设计加权因子 w_j 的初始值。假定通过实验或摩擦模型获得下面的实验数据对

$$(\dot{x}^{(p)}, F^{(p)}), p = 1, 2, \dots, P \quad (7-17)$$

式中 $\dot{x}$ ——速度;

F ——摩擦力。

步骤 1: 输入输出变量的模糊区间划分

由前面的摩擦模型可知摩擦力是速度的函数,即摩擦力随速度的变化而变化。故把速度作为模糊系统的前提变量,而把摩擦力作为结论变量。设输入变量 $\dot{x}$ 取值区间取 $[\dot{x}^-, \dot{x}^+]$, 在此取值区间取模糊集合 $A(\dot{x}) = \{A_1^{\dot{x}}, \dots, A_N^{\dot{x}}\}$, 隶属度函数取高斯函数。另设输出变量 F 取值区间取 $[F^-, F^+]$, 在此取值区间取模糊集合 $B(F) = \{B_1^F, \dots, B_M^F\}$, 隶属度函数取三角型函数。

步骤 2: 关系数据库转换为模糊型关系数据库

在摩擦的普通型关系数据库中,设 T_{L_1} 表示一个普通型关系数据库, t_p 表示 T_{L_1} 的第 p 个元组, $L_1 = \{\dot{x}, F\}$ 表示属性集, $t_p[\dot{x}]$ 和 $t_p[F]$ 分别表示属性 $\dot{x}$ 和 F 在第 p 个元组上的值

$$T_{L_1} = \{t_1, \dots, t_p, \dots, t_p\} \quad (7-18)$$

在摩擦的模糊型关系数据库中,设 T_{L_2} 表示一个模糊型关系数据库, $\mu(t_p)$ 表示 T_{L_2} 的第

p 个元组, $L_2 = \{A(\dot{x}) \cup B(F)\}$ 表示属性集, $\mu(t_p)$ 表示 t_p 模糊化后的隶属度值。至此, 将 T_{L_1} 一个普通型关系数据库转化为模糊型关系数据库 T_{L_2} , 即

$$T_{L_2} = \{\mu(t_1), \dots, \mu(t_p), \dots, \mu(t_p)\} \quad (7-19)$$

步骤3: 计算摩擦模糊规则的支持度

从数据挖掘的角度看, 如果一条模糊规则有实际意义, 它必须具有足够的支持度, 它反映了样本元组对该条规则的支持程度。现定义支持度如下

$$Sup(\dot{x} \Rightarrow F) = \frac{\sum_{p=1}^P \mu_{(B_f)^p}(F) \mu_{(A_j^{\dot{x}})^p}(\dot{x})}{\sum_{p=1}^P \mu_{(A_j^{\dot{x}})^p}(\dot{x})} \quad (7-20)$$

为了简化计算, 也可以用下式计算支持度

$$Sup(\dot{x} \Rightarrow F) = \frac{1}{P} \sum_{p=1}^P \mu_{(B_f)^p}(F) \mu_{(A_j^{\dot{x}})^p}(\dot{x}) \quad (7-21)$$

步骤4: 创建完备的摩擦模糊规则库

首先, 算法要保证由前提变量构成的每个模糊子空间都能被遍历。其次, 结论变量在模糊子空间选用哪个模糊集, 由规则的最大支持度决定。算法的主要步骤如第2章所示, 通过上面的遍历产生了完备的模糊规则库。我们用一个简单例子说明此过程。设前提变量 $\dot{x}$ 和结论变量 F 的模糊集合分别为 $A = \{A_1^{\dot{x}}, \dots, A_5^{\dot{x}}\}$ 和 $B = \{B_1^F, \dots, B_5^F\}$ 。对于 $\{A_1^{\dot{x}}\}$, 我们首先分别计算 $\{A_1^{\dot{x}}, B_1^F\}$, $\{A_1^{\dot{x}}, B_2^F\}$, $\{A_1^{\dot{x}}, B_3^F\}$, $\{A_1^{\dot{x}}, B_4^F\}$ 和 $\{A_1^{\dot{x}}, B_5^F\}$ 的支持度, 然后结论变量 F 最终选择的模糊集由最大支持度决定。对于 $\{A_2^{\dot{x}}\}$, $\{A_3^{\dot{x}}\}$, $\{A_4^{\dot{x}}\}$ 和 $\{A_5^{\dot{x}}\}$ 重复上面的过程。由这些规则构成了所有模糊规则的一个子空间, 该子空间是最后选择的模糊规则库。

步骤5: 初始化模型参数

为了得到模糊神经网络的初始权值, 基于最大支持度原则, 取下面的摩擦模糊规则进行模糊神经网络建模

$$R^{(j)}: \text{If } \dot{x} \text{ is } A_j^{\dot{x}} \text{ Then } \hat{F}_f \text{ is } B_l^*, (j=1, \dots, N; l=1, \dots, M) \quad (7-22)$$

上面的模糊规则中的 B_l^* 通过下面的准则得到, 也就是在 M 个模糊子集中 $B_1, \dots, B_M$ 中通过式 (7-23) 找到 B_l^*

$$Sup(\dot{x}_{A_j} \Rightarrow F_{B_j^*}) \geq Sup(\dot{x}_{A_j} \Rightarrow F_{B_j}) \quad (7-23)$$

通过比较式 (7-13) 和式 (7-22), 可以得到下面的初始权值

$$w_j = \bar{F}_{B_l^*} \quad (7-24)$$

式中 $\bar{F}$ ——隶属度函数 $\mu_{B_l^*}(F)$ 取得最大值的点。

定义模糊神经网络的基函数如下

$$\varphi_j(\dot{x}) = \frac{\mu_j(e_j^{(2)})}{\sum_{j=1}^N \mu_j(e_j^{(2)})} \quad (7-25)$$

模糊神经网络系统的总输出为

$$\hat{F}(\dot{x}) = w^T \varphi(\dot{x}) \quad (7-26)$$

式中 $w = (w_1, w_2, \dots, w_j)^T$, 表示模糊系统的参数向量;

$\varphi(\dot{x}) = (\varphi_1(\dot{x}), \varphi_2(\dot{x}), \dots, \varphi_j(\dot{x}))^T$, 表示模糊基函数向量。

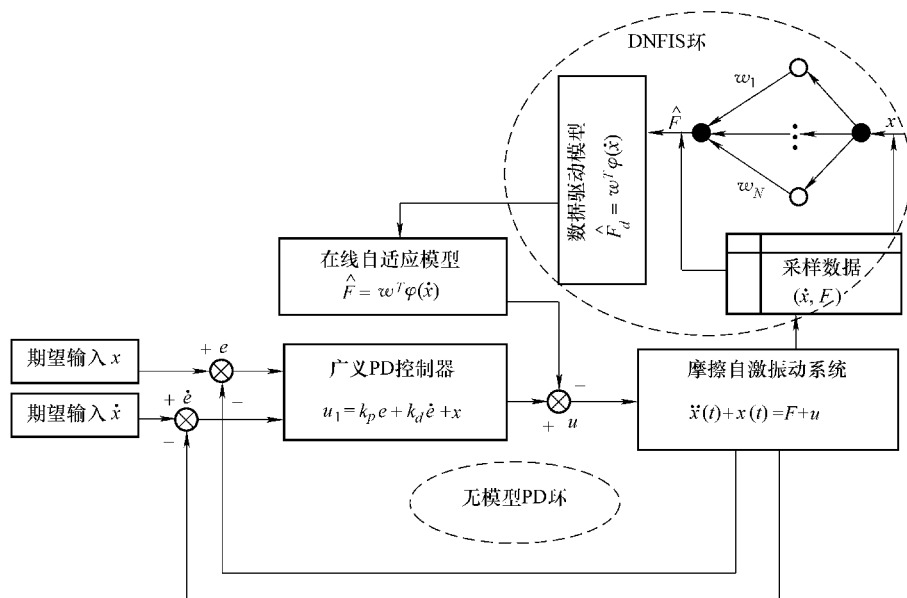


图 7-4 基于模糊神经网络补偿的主动控制图

7.3.2 摩擦自激振动的主动控制理论

若方程 (7-4) 中的摩擦力 F 和 $x(t)$ 已知, 可采用基于摩擦模型补偿的 PD 主动控制, 也就是如下方程

$$u = k_p e + k_d \dot{e} - F + x \quad (7-27)$$

式中 $e = x_m - x$;

$\dot{e} = \dot{x}_m - \dot{x}$;

x_m ——已知的系统期望输出;

k_p 和 k_d ——分别表示比例和微分系数。

把方程 (7-27) 代入方程 (7-4) 可得如下动态误差方程

$$\ddot{e} + k_d \dot{e} + k_p e = \ddot{x}_m \quad (7-28)$$

由于在定位调节中的平衡位置参考输入 $\ddot{x}_m = 0$, 方程 (7-28) 可以写成如下方程

$$\ddot{e} + k_d \dot{e} + k_p e = 0 \quad (7-29)$$

显然, 如果取向量 $\mathbf{k} = (k_d, k_p)^T$ 使 $H(s) = s^2 + k_d s + k_p$ 为 Hurwitz, 则有 $\lim_{t \rightarrow \infty} \|e\| = 0$ 。

在实际工程中, 位移 x 是可以检测到的, 但摩擦力 F 很难精确检测或建模得到。对摩擦项 F 来说, 由于建立精确动态摩擦模型的高难性, 获得精确算法方程 (7-27) 是不可能的。为此, 用前文的模糊系统替代方程 (7-27) 中的摩擦项可得

$$u = k_p e + k_d \dot{e} - \hat{F} + x \quad (7-30)$$

式中 $\hat{F}$ ——前文提到的模糊系统。

把方程 (7-30) 代入方程 (7-4) 可得如下动态误差方程

$$\ddot{e} + k_d \dot{e} + k_p e = \hat{F} - F \quad (7-31)$$

定义最优参数向量为

$$\mathbf{w}^* = \operatorname{argmin}_{\mathbf{w} \in \Omega_w} (\sup \|\hat{F} - F\|) \quad (7-32)$$

式中 Ω_w —— $\mathbf{w}$ 的界。

那么最小逼近误差可表示如下方程

$$\varepsilon = F - \varphi^T(\dot{x}) \mathbf{w}^* \quad (7-33)$$

由方程 (7-26) 和方程 (7-33), 知方程 (7-31) 可写成如下

$$\ddot{e} + k_d \dot{e} + k_p e = (\tilde{\mathbf{w}}^T \varphi(\dot{x}) - \varepsilon) \quad (7-34)$$

式中 $\tilde{\mathbf{w}} = \mathbf{w} - \mathbf{w}^*$ 。

方程 (7-34) 也等价如下形式

$$\underline{\dot{e}} = \mathbf{A} \underline{e} + \mathbf{B} \tilde{u} \quad (7-35)$$

$$\text{式中 } \mathbf{A} = \begin{pmatrix} 0 & 1 \\ -k_p & -k_d \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\underline{e} = (e, \dot{e})^T$$

$$\tilde{u} = \tilde{\mathbf{w}}^T \varphi(\dot{x}) - \varepsilon$$

定理 1. 如果采用式 (7-36) 和式 (7-37) 作为主动控制器, 其中自适应参数由式 (7-38) 在线调节

$$u = k_p e + k_d \dot{e} - \hat{F} + \dot{x} \quad (7-36)$$

$$\hat{F}(\dot{x}) = \mathbf{w}^T \varphi(\dot{x}) \quad (7-37)$$

$$\dot{\mathbf{w}} = \begin{cases} -\gamma \varphi(\dot{x}) \mathbf{B}^T \mathbf{P} \underline{e}, & \text{if } (\|\mathbf{w}\| < \Re) \text{ or } (\|\mathbf{w}\| = \Re \text{ and } \underline{e}^T \mathbf{P} \mathbf{B} \varphi^T(\dot{x}) \mathbf{w} \geq 0) \\ P[\cdot], & \text{if } (\|\mathbf{w}\| = \Re \text{ and } \underline{e}^T \mathbf{P} \mathbf{B} \varphi^T(\dot{x}) \mathbf{w} < 0) \end{cases} \quad (7-38)$$

式中 $P[\cdot] = -\gamma \varphi(\dot{x}) \mathbf{B}^T \mathbf{P} \underline{e} + \gamma \frac{\underline{e}^T \mathbf{P} \mathbf{B} \varphi^T(\dot{x}) \mathbf{w}}{\|\mathbf{w}\|^2} \mathbf{w}$, 是投影算子。

正定解 $\mathbf{P} > 0$ 满足下面的李雅普诺夫方程

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} = -\mathbf{Q} \quad (7-39)$$

那么, 可得到 $\lim_{t \rightarrow \infty} \|e(t)\| = 0$ 。

证明: 取 Lyapunov 函数为

$$V = \frac{1}{2} \underline{e}^T \mathbf{P} \underline{e} + \frac{1}{2\gamma} (\tilde{\mathbf{w}}^T \tilde{\mathbf{w}}) \quad \gamma > 0 \quad (7-40)$$

求 V 沿方程 (7-40) 的微分, 可推出

$$\dot{V} = \frac{1}{2} \underline{e}^T \mathbf{P} \dot{\underline{e}} + \frac{1}{2} \dot{\underline{e}}^T \mathbf{P} \underline{e} + \frac{1}{\gamma} (\tilde{\mathbf{w}}^T \dot{\tilde{\mathbf{w}}}) \quad (7-41)$$

因为 $\dot{\tilde{\mathbf{w}}} = \dot{\mathbf{w}}$ ，并结合式 (7-35) 可得

$$\dot{V} = \frac{1}{2} [\mathbf{e}^T (\mathbf{P}\mathbf{A} + \mathbf{A}^T\mathbf{P}) \mathbf{e}] + [\tilde{\mathbf{w}}^T \varphi(\dot{x}) - \varepsilon] \mathbf{B}^T \mathbf{P} \mathbf{e} + \frac{1}{\gamma} (\tilde{\mathbf{w}}^T \dot{\tilde{\mathbf{w}}}) \quad (7-42)$$

由方程 (7-38) 和方程 (7-39)，上面方程对消一部分后可得到

$$\dot{V} = -\frac{1}{2} \mathbf{e}^T \mathbf{Q} \mathbf{e} - \varepsilon \mathbf{B}^T \mathbf{P} \mathbf{e} \quad (7-43)$$

因此，可以得到下面不等式

$$\begin{aligned} \dot{V} &\leq -\frac{\lambda_{\min}(\mathbf{Q}) - 1}{2} \|\mathbf{e}\|^2 + \frac{1}{2} \|\mathbf{B}^T \mathbf{P} \varepsilon\|^2 - \frac{1}{2} (\|\mathbf{e}\|^2 + 2\varepsilon \mathbf{B}^T \mathbf{P} \mathbf{e} + \|\mathbf{B}^T \mathbf{P} \varepsilon\|^2) \\ &\leq -\frac{\lambda_{\min}(\mathbf{Q}) - 1}{2} \|\mathbf{e}\|^2 + \frac{1}{2} \|\mathbf{B}^T \mathbf{P} \varepsilon\|^2 \end{aligned} \quad (7-44)$$

式中 $\lambda_{\min}(\mathbf{Q})$ —— $\mathbf{Q}$ 的最小特征值。

对上式两边积分并取 $\lambda_{\min}(\mathbf{Q}) > 1$ ，可以得到

$$\int_0^t \|\mathbf{e}\|^2 dt \leq \frac{2}{\lambda_{\min}(\mathbf{Q}) - 1} (V(0) + V(t)) + \frac{1}{\lambda_{\min}(\mathbf{Q}) - 1} \|\mathbf{B}^T \mathbf{P}\|^2 \times \int_0^t \|\varepsilon\|^2 dt \quad (7-45)$$

如果 $\varepsilon \in L_2$ ，可得到 $e \in L_2$ 。由于方程 (7-35) 右边的所有变量是有界的，也就是 $\dot{e}(t) \in L_\infty$ 。由 Barbalat 引理^[15]可知 $\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| = 0$ 。

7.4 数值结果

把文中所采用的模糊神经网络系统与自适应补偿控制方案用于如下摩擦自激运动系统进行仿真试验

$$\ddot{x}(t) + x(t) = F(v) + u \quad (7-46)$$

式中 $F(v)$ ——归一化后的广义摩擦力；

u ——归一化后的广义外加主动控制力。

7.4.1 系统参数设置

在下面的仿真中，Coulomb 和 Stribeck 摩擦模型将被用到，系统参数选择如下： $m = 0.6 \text{ kg}$ ， $F_N = 15 \text{ N}$ ， $x(0) = (-0.02, 0.02)^T$ ， $x_m = (0.0, 0.0)^T$ ， $k_p = 150$ ， $k_d = 15$ ， $-40 < u < 40$ 。取 $\mathbf{P} = \begin{pmatrix} 600 & 20 \\ 20 & 10 \end{pmatrix}$ ，解方程 (7-39) 可得 $\mathbf{Q} = \begin{pmatrix} 6000 & 1200 \\ 1200 & 260 \end{pmatrix}$ 。

7.4.2 模糊神经网络补偿器的初始化设置

步骤 1. 定义隶属度函数

针对变量 $\dot{x}$ 取隶属函数： $\mu_{A_1}^{\dot{x}} = \exp(-5.0(\dot{x} + 0.50)^2)$

$$\mu_{A_2}^{\dot{x}} = \exp(-5.0(\dot{x} + 0.25)^2)$$

$$\mu_{A_3^{\dot{x}}} = \exp(-5.0(\dot{x})^2)$$

$$\mu_{A_4^{\dot{x}}} = \exp(-5.0(\dot{x} - 0.25)^2)$$

$$\mu_{A_5^{\dot{x}}} = \exp(-5.0(\dot{x} - 0.50)^2)$$

对于变量 F 取三角型隶属函数, 5 组参数分别为 $[-20, -10]$ 、 $[-20, -10, 0]$ 、 $[-10, 0, 10]$ 、 $[0, 10, 20]$ 、 $[10, 20]$ 。

步骤 2. 摩擦数据的关系表建立

首先, 建立 $\dot{x}$ 和 F 的 10 个样本数据采集点 $t_1 \rightarrow t_{10}$ 的普通型关系表 (表 7-1)。然后, 将 $\dot{x}$ 和 F 的 10 个样本数据采集点的普通型关系表转换成模糊型关系表 (表 7-2)。

表 7-1 样本数据

T_{t_1}	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
$\dot{x}$	-0.5	-0.4	-0.3	-0.2	-0.1	+0.1	+0.2	+0.3	+0.4	+0.5
F	-20	-18	-17	-16	-14	+14	+16	+17	+18	+20

表 7-2 样本数据模糊化

T_{t_2}	$A_1^{\dot{x}}$	$A_2^{\dot{x}}$	$A_3^{\dot{x}}$	$A_4^{\dot{x}}$	$A_5^{\dot{x}}$	B_1^F	B_2^F	B_3^F	B_4^F	B_5^F
$\mu(t_1)$	1.00	0.04	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.30
$\mu(t_2)$	0.61	0.32	0.00	0.00	0.00	0.80	0.20	0.00	0.00	0.00
$\mu(t_3)$	0.14	0.88	0.01	0.00	0.00	0.70	0.30	0.00	0.00	0.00
$\mu(t_4)$	0.01	0.88	0.14	0.00	0.00	0.60	0.40	0.00	0.00	0.00
$\mu(t_5)$	0.00	0.32	0.61	0.00	0.00	0.40	0.60	0.00	0.00	0.00
$\mu(t_6)$	0.00	0.00	0.61	0.32	0.00	0.00	0.00	0.00	0.60	0.40
$\mu(t_7)$	0.00	0.00	0.14	0.88	0.01	0.00	0.00	0.00	0.40	0.60
$\mu(t_8)$	0.00	0.00	0.01	0.88	0.14	0.00	0.00	0.00	0.30	0.70
$\mu(t_9)$	0.00	0.00	0.00	0.32	0.61	0.00	0.00	0.00	0.20	0.80
$\mu(t_{10})$	0.00	0.00	0.00	0.04	1.00	0.30	0.00	0.00	0.00	1.00

步骤 3. 提取模糊规则

按照前面的算法, 基于最大支持度原则提取模糊规则库。由表 7-3 的计算的支持度结果可得如下模糊规则库: ①If $\dot{x}$ is $A_1^{\dot{x}}$ Then F is B_1^F ; ②If $\dot{x}$ is $A_2^{\dot{x}}$ Then F is B_1^F ; ③If $\dot{x}$ is $A_3^{\dot{x}}$ Then F is B_2^F ; ④If $\dot{x}$ is $A_4^{\dot{x}}$ Then F is B_5^F ; ⑤If $\dot{x}$ is $A_5^{\dot{x}}$ Then F is B_5^F 。

表 7-3 计算支持度

Sup	B_1^F	B_2^F	B_3^F	B_4^F	B_5^F
$A_1^{\dot{x}}$	0.16	0.02	0.00	0.00	0.03
$A_2^{\dot{x}}$	0.16	0.09	0.00	0.00	0.00
$A_3^{\dot{x}}$	0.03	0.04	0.00	0.04	0.03
$A_4^{\dot{x}}$	0.00	0.00	0.00	0.09	0.16
$A_5^{\dot{x}}$	0.03	0.00	0.00	0.02	0.16

7.4.3 模糊神经网络补偿器的自适应参数

在这个仿真例子, 取 $\gamma = 2000$, $\Re = 25\sqrt{5}$, $S = \sum_{j=1}^5 \mu_{A_j^i}(\dot{x})$, $\varphi(\dot{x}) = \frac{1}{S}(\mu_{A_1^i}(\dot{x}), \dots, \mu_{A_5^i}(\dot{x}))^T$, 可以得到下面的模糊神经网络补偿器的自适应特性

$$\hat{F}(\dot{x}) = w^T \varphi(\dot{x}) \quad (7-47)$$

$$\dot{w} = \begin{cases} -\gamma \varphi(\dot{x}) (20e + 10\dot{e}), & \text{if } (\|w\| < \Re) \text{ or } \|w\| = \Re \text{ and } e^T PB \varphi^T(\dot{x}) w \geq 0 \\ P[\cdot], & \text{if } \|w\| = \Re \text{ and } e^T PB \varphi^T(\dot{x}) w < 0 \end{cases} \quad (7-48)$$

式中 $P[\cdot] = -\gamma \varphi(\dot{x}) (20e + 10\dot{e}) + \gamma (20e + 10\dot{e}) \frac{\varphi^T(\dot{x}) w}{\|w\|^2} w$, 是投影算子。

7.4.4 Coulomb 摩擦诱发振动的主动控制

在这个例子中, 我们验证模糊神经网络系统的建模的自适应特性。设方程 (7-46) 的摩擦力为 Coulomb 摩擦力, 并假设机械系统的初始摩擦力较大, 随系统的磨合逐渐变小。这一物理现象在仿真过程如下: 设方程 (7-11) 中的初始值为 $0 \leq t < 100$, $F_c = 0.70F_N$; 经过 100s 后变成 $100 \leq t < 200$, $F_c = 0.60F_N$; 最后阶段取 $200 \leq t \leq 300$, $F_c = 0.50F_N$ 。在这个例子中取 $k = 763 \text{ N/m}$ 。

图 7-5 是系统 x 和 $\dot{x}$ 的时间响应曲线以及相平面曲线。该结果是控制取 $u = k_p e + k_d \dot{e} - F + x$ 时获得的。其中控制器中的摩擦力始终为 $F = 0.70F_N \text{sgn}(v)$ 。这个例子说明, 如果用传统的摩擦力模型进行补偿控制时, 控制器的摩擦力如果不随实际摩擦力的改变而改变, 该控制器短时间内很难消除摩擦诱发的振动。

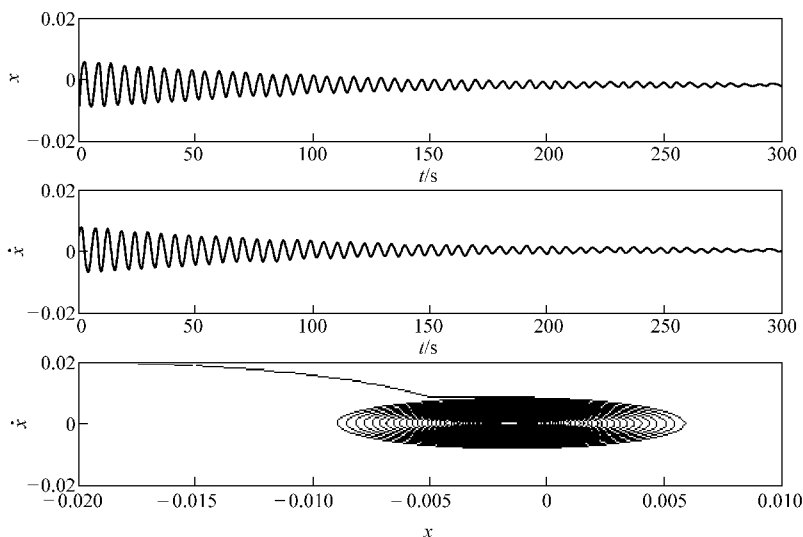


图 7-5 基于传统摩擦模型补偿的主动控制图

图 7-6 是系统 x 和 $\dot{x}$ 的时间响应曲线以及相平面曲线。该结果是控制取 $u = k_p e + k_d \dot{e} -$

$F + x$ 时获得的。其中控制器中的 $\hat{F}$ 是模糊神经网络系统式 (7-37)。这个例子显示了模糊神经网络系统的自适应特性的重要性, 它随着摩擦环境改变而进行动态调整, 该控制器短时间内就能消除摩擦诱发的振动。

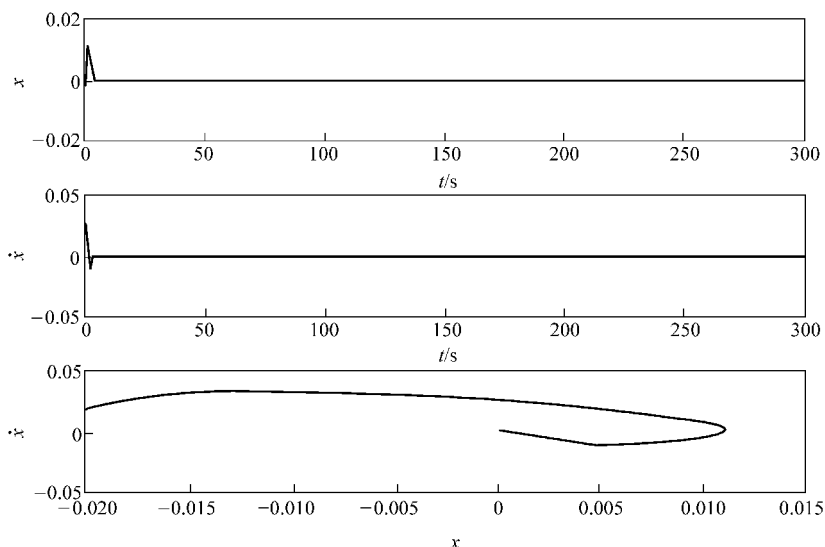


图 7-6 基于模糊神经网络补偿的主动控制图

7.4.5 Stribeck 摩擦诱发振动的主动控制

在这个例子中, 我们用另一摩擦模型进一步验证模糊神经网络系统在消除摩擦振动方面所取得的效果。设方程 (7-46) 的摩擦力为 Stribeck 摩擦力, 也就是方程 (7-46) 中的 Stribeck 摩擦力参数为 $F_c = 0.40F_N$ 、 $F_s = 0.60F_N$ 、 $F_v = 0.04F_N$ 、 $F_N = 15.0\text{N}$ 、 $v_s = 1$ 和 $\delta_s = 1$ 。图 7-7 和图 7-8 所示是 $k = 1500\text{N/m}$ 和 $k = 763\text{N/m}$ 时, 系统 x 和 $\dot{x}$ 的时间响应曲线以及相平

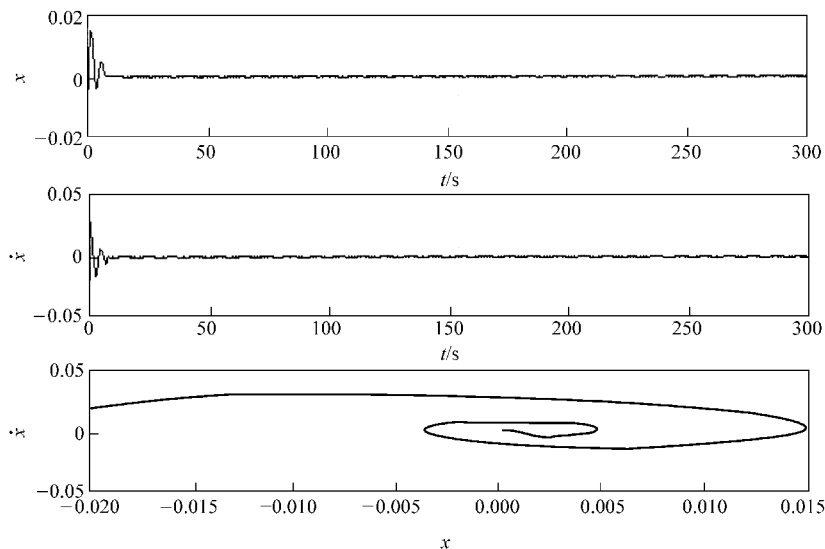


图 7-7 基于模糊神经网络补偿的主动控制图 (1)

面曲线,该结果是控制器取 $u = k_p e + k_d \dot{e} - F + x$ 时获得的。从图 7-7 和图 7-8 仿真结果可以看出基于模糊神经网络系统补偿的主动控制不但消除了极限环,并且无稳态误差。

从这两个例子可以看出,本章所提出的方法在应对摩擦环境的变化,甚至系统参数的变化都具有良好的自适应特性,系统取得了良好的主动控制效果。

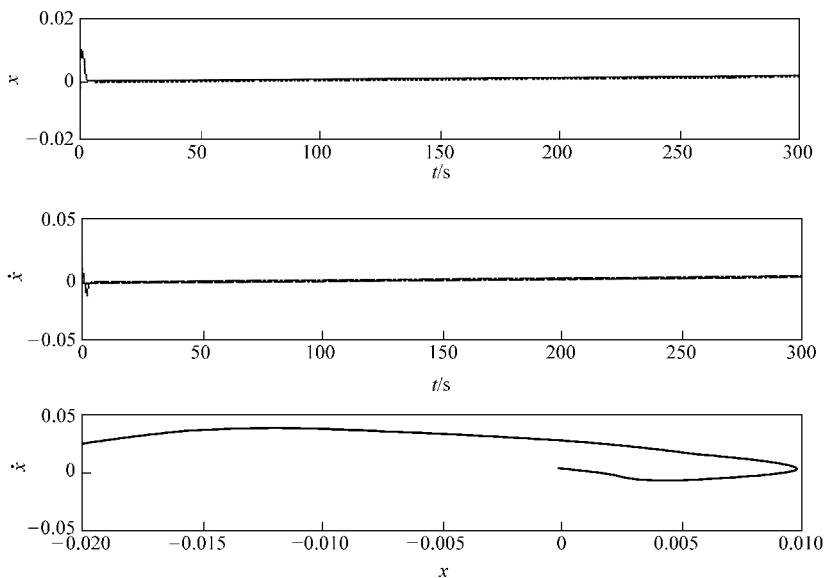


图 7-8 基于模糊神经网络补偿的主动控制图 (2)

7.5 本章小结

本章分析了 Coulomb 和 Stribeck 摩擦诱发的自激振动特征及自激振动产生的成因问题。为了对摩擦诱发振动极限环的振幅进行主动控制,采用自适应模糊神经网络建模技术在线逼近摩擦力,并将辨识结果用在振动的主动控制中,其中的自适应参数调节是基于李雅普诺夫稳定性理论设计的。基于自适应模糊神经网络补偿的 PD 主动控制技术,对抑制极限环的振幅具有良好的效果,不但可以消除极限环,而且也可以消除稳态误差。进一步的研究工作将深化探索摩擦诱发振动的解析解的精确形式和适用范围,以及其他摩擦模型诱发的振动及相关的智能主动控制技术。

参考文献

- [1] 高中庸. 涂油降噪理论与实践 [M]. 北京: 科学出版社, 2009, 49-52.
- [2] Heckl M A, Abrahams I D. Active control of friction-driven oscillations [J]. Journal of Sound and Vibration, 1996, 193 (1): 417-426.
- [3] 丁文镜. 自激振动 [M]. 北京: 清华大学出版社, 2009, 100-107.
- [4] Hinrichs N, Oestreich M, Popp K. On the modeling of friction oscillators [J]. Journal of Sound and Vibration, 1998, 216 (): 435-459.

- [5] Zjinjade P B, Mallik A K. Impact damper for controlling friction – driven oscillations [J] . Journal of Sound and Vibration, 2007, 306 (2): 238 – 251.
- [6] ARMSTRONG B, CANUDA D W C. A survey of models, analysis tools and compensation methods for control of machines with friction [J] . Automatica, 1994, 30 (7): 1083 – 1138.
- [7] Chatterjee S. Non – linear control of friction – induced self – excited vibration [J] . International Journal of Non – linear Mechanics, 2007, 42 (3): 459 – 469.
- [8] Chatterjee S. Vibration control by recursive by time – delayed acceleration feedback [J] . Journal of Sound and Vibration, 2008, 317 (1): 67 – 90.
- [9] Wang L X, Stable adaptive fuzzy control of nonlinear systems [J] . IEEE Trans. on Fuzzy Systems, 1993, 1 (1): 146 – 155.
- [10] Ding H, Wang J. Recurrent neural networks for minimum infinity – norm kinematic control of redundant manipulators [J] . IEEE Trans. on systems, man and cybernetics – part A, 1999, 29 (3): 269 – 276.
- [11] Selmic R R, Lewis F L. Neural – network approximation of piecewise continuous functions: application to friction compensation [J] . IEEE Trans. on Neural Networks, 2002, 13 (3): 745 – 750.
- [12] 王永富, 王殿辉, 柴天佑. 基于数据挖掘与系统理论建立摩擦模糊模型与控制补偿 [J] . 自动化学报, 2010, 36 (3): 412 – 420.
- [13] Takagi T, Sugeno M. Fuzzy identification of systems and its application to modeling and control [J] . IEEE Trans. on Syst. , Man, Cybern. , 1985, SMC – 15 (1): 116 – 132.
- [14] Yongfu Wang, Dianhui Wang, Tianyou Chai. Active control of friction self – excited vibration using neuro – fuzzy and data mining techniques [J] . Expert Systems with Applications, 2013, 40 (4) : 975 – 983.
- [15] Sastry S S, Bodson M. , Adaptive Control: Stability, Convergence, and Robustness [M] . Englewood Cliffs, NJ: Prentice – Hall, 1989.

第8章 机器手实验系统与模糊控制补偿设计实例

8.1 引言

机器人技术是一门以机械、电子及信息技术为基础，自动控制理论为媒介，结合的综合性工程技术。研究机器人控制技术，对促进我国产业技术的发展具有深远意义。

从控制的角度看，机器人是一个非线性系统。为了获得理想的控制品质，应该充分利用现代控制理论的最新成果，开发有效的高级控制策略。在工程实际中不可避免地存在着许多无法用数学模型精确描述的不确定性，如不确定摩擦以及摩擦诱发的振动等问题。模糊建模与自适应控制的基本出发点，是仿人的智能以实现对复杂不确定性系统进行有效的控制，它具有从环境自学习、适应环境的能力，自动进行信息处理以减少其不确定性，能规划、产生并能安全、可靠地执行控制作用，模糊建模与自适应控制正是以具有不确定性和未知性的系统为研究对象的控制技术。

8.2 机器手的控制系统结构

ZEBRA ZERO 机械手是美 IMI 公司生产的一台小型低惯量的机械手，只能在适中的最大速度、加速度下运行，并只能忍受轻度的损伤。最大承受力 100kgf，力矩 4000kgf·mm，正常工作时，力限最大为 2.0kgf，力矩 500kgf·mm。这台六自由度的机械手原系统由以下几个主要部分组成：操作机和手爪、六轴腕装力传感器、电源放大器、运动控制板和计算机。为了能验证控制算法，针对原机械手的机械部分，我们自主设计和开发了该机械手控制系统的软硬件平台。该平台具有完整的机械系统——六个自由度并带有力传感器，便于进行非线性控制理论研究，以及开展相关算法基础实验，该实验平台的成功开发对新型控制算法的验证具有重要意义。本章将模糊系统应用到机器人的建模与控制补偿实验中。

8.2.1 机械子系统的构造

机械手的所有关节运动都由电动机带动，而且它的齿轮结构是耦合的。每个电动机的传动过程如下：一号电动机转动将动力传给齿轮 B，齿轮 B 与齿轮 B' 啮合，B' 固定在底座上从而使机械手与底座发生相对转动，实现腰部的转动。二号电动机转动将动力传给齿轮 C，齿轮 C 与斜齿轮 C' 啮合，斜齿轮 C' 与轴 A 配合从而带动肩关节的转动。三号电动机转动将动力传给齿轮 D，齿轮 D 与齿轮 D1 啮合，齿轮 D1 通过轴将动力传给斜齿轮 D2，斜齿轮 D2 与斜齿轮 D3 啮合，将动力传给斜齿轮 D3，斜齿轮 D3 固定在前臂的套筒上，从而带动前臂运动。四号电动机转动将动力传给齿轮 E，齿轮 E 与齿轮 E1 啮合，齿轮 E1 通过

轴将动力传给斜齿轮 E2, 斜齿轮 E3 同时与斜齿轮 E2、E4 啮合这样动力就传到斜齿轮 E4 上, 斜齿轮 E4 通过轴将动力又传给齿轮 E5, 齿轮 E5 与齿轮 E6 啮合同时固定在中空轴 M 上, 手腕部就是与中空轴 M 相连, 中空轴 M 转动从而带动腕部转动。六号电动机转动将动力传给齿轮 F, 齿轮 F 与齿轮 F1 啮合, 齿轮 F1 通过轴将动力传给斜齿轮 F2, 斜齿轮 F3 同时与斜齿轮 F2、F4 相啮合, 这样动力传到齿轮 F4 上, 齿轮 F4 固定在 L 轴上, L 轴通过键配合与斜齿轮 F5 连接。这样五号电动机的转动会引起轴 L 的转动, L 的转动带动 F5 的转动。F6 同时与斜齿轮 F5 固定在手爪上的斜齿轮 H 齿轮啮合 L 的转动将使手爪能够摆动 180° 。五号电动机也是控制手爪的运动, 它控制的动作是使手爪能够实现自转。它的动力传动过程基本上与前三个电动机相似, 不同的是在动力传到前臂时是通过一个中空轴将动力传给斜齿轮 G6, G6 通过 G7 将动力传给 H 实现自转。但由于齿轮 F5、F6、G6、G7H 是耦合的, 实际上要实现自转五号、六号电动机应同时转动或五号电动机转动时六号电动机无负载, 否则将阻碍自转运动。要实现腕部不转动而爪部自转四号电动机也必须同时配合转动。该部分的构成如图 8-1 所示。

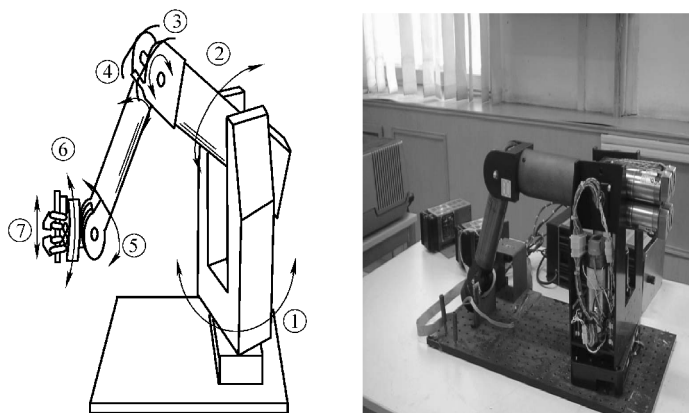


图 8-1 机械手的结构与实物图

① ~ ⑦—运动自由度

8.2.2 上位机子系统

上位机子系统采用普通微机, 操作系统用 Microsoft 的 Windows 系统, 机器人控制系统应用软件采用 Microsoft 的 Visual C++ 6.0 开发的。之所以选择 Microsoft 的 Visual C++ 6.0 作为开发工具, 是因为 Visual C++ 6.0 是一个功能强大的可视化编程环境, 它为我们提供了一种方便、快捷的 Windows 应用程序开发工具。它使用了 Microsoft Windows 图形用户界面的许多先进特性和设计思想, 采用了弹性的可重用的面向对象 C++ 程序语言。上位机子系统的主界面如图 8-2 所示。在上位机子系统中, 由四个软件子模块组成或者说上位机子系统主要完成的功能分别为: OpenGL 子模块、MSComm 串口通信控件子模块、外置控制算法组件子模块和 Matlab 子模块。功能分别如下^[1]:

1. OpenGL 子模块

OpenGL 是 SGI 公司开发的, 可独立于操作系统和硬件环境的三维图形库, 已在各种工



图 8-2 上位机子系统的主界面

作站和高档微机中运行，由于其强大的图形功能和跨平台的能力，已成为事实上的图形标准，被人们广泛应用于科学可视化、实体造型、CAD/CAM、模拟仿真等诸多领域。目前，包括 Microsoft、SGI、IBM、DEC、SUN、HP 等大公司都采用了 OpenGL 作为三维图形标准。在机器人上位机子系统，我们采用了 OpenGL 图形库构造机器人的实体模型，并把传感器码盘的位置信息实时传到图形参数中。这样就可以保证上位机的实时图形显示和机器人的实际运动是同步的。采用该项技术的好处在于：

1) 从上位机可以看到机器人的运动情况，增加了视觉效果。

2) 对于机器人在危险环境或者远距离工作，而人又无法到达现场，采用此项技术的意义是明显的，它可以保证人在上位机系统进行实时监控机器人的工作状况，以便发出新的控制指令。

2. MSComm 串口通信控件子模块

MSComm 是 Microsoft 公司提供的简化 windows 下串口通信编程的 Active 控件，它为应用程序提供了通过串行接口收发数据的简便方法。具体来说，它提供了两种处理通信问题的方法：一是事件驱动方法，另一个是查询法。

事件驱动方式是处理串行端口交互作用的一种非常有效的方法。在许多情况下，在事件发生时需要得到通知，例如，在串口接收缓冲区中有字符，或者 Carri erDetect (CD) 或 RequestToSend (RTS) 线上一个字符到达或一个事件发生时。在这些情况下，可以利用 MSComm 控件的 OnComm 事件捕获并处理这些通信事件。OnComm 时间还可以检查和处理通信错误。CommEvent 属性列出了所有通信事件和通信错误，可以通过查阅 MSDN (Microsoft Developer Network) 获得。在编程过程中，就可以在 OnComm 时间处理函数中加入自己的处理代码。这种方法的优点是程序响应及时，可靠性高。每个 MSComm 控件对应着一个串行

端口。如果应用程序需要访问多个串行端口，必须使用多个 MSComm 控件。

查询方式实质上还是属于事件驱动，但在某些情况下，这种方式显得更为便捷。在程序的每个关键功能之后，可以通过检查 CommEvent 属性的值来查询事件和错误，只要 CommEvent 属性的值发生变化，就表明一个通信时间或一个错误发生。如果应用程序较小，并且是自成一体的，这种方法可能是更可取的。在本实验系统中，我们采用了 MSComm 控件作为上位机（PC）和下位机（单片机系统）的串口通信的基础，并对该控件的一些功能进行了拓展。

3. 外置控制算法组件子模块

外置控制算法是相对于内置控制算法而言的，所谓内置控制算法就是在下位机系统中我们采用了机器人专用运动控制芯片 HCTL1100，该芯片内部固化了一个控制算法，我们只需往该芯片指定的寄存器单元写入设定值，并把传感器位置码盘信息返回到相应的寄存器单元中，那么该内置算法会根据误差自动调节输出的 PWM 波的占空比。但我们对该内置算法只能机械地操作，该算法的内部详细结构以及原代码不能深入了解与分析，该算法应是保密的。

我们屏蔽了 HCTL1100 芯片的内置控制算法，只用了 HCTL1100 芯片的写读功能，而在上位机系统开发了外置控制算法。目前，主要开发了两个外置控制算法：PD 控制和基于模糊补偿的 PD 控制算法。对外置控制算法的程序设计采用了组件化设计原则，这样便于拓展该算法以及自适应调节算法参数。

4. Matlab 接口子模块

Matlab 接口子模块的主要目的是想利用 Matlab 的强大科学计算能力，把 Matlab 的计算数据导入到 VC 中，以便进行控制。特别是 Matlab 提供的专用工具箱为一些专业问题提供了简洁的求解方法。

8.2.3 下位机子系统

下位机子系统分为测试用单片机系统及控制用单片机系统。测试用单片机主要完成发生测试用 PWM 波，切换控制模式；控制用单片机主要完成与上位机的串口通信，对 HCTL1100 运动芯片的读写操作以及对驱动电路的操作。在系统设计中测试用单片机用 89c52（51 内核）单片机，用 PLM51 语言进行编程。测试用单片机安装在驱动测试电路板上，测试用单片机独立于上位机系统，配合面板按键工作。当进行测试实验时，用该单片机发生测试用 PWM 波，进而驱动电动机，用以监测电动机工作正常与否。当进行控制算法实验时，用以切换控制模式，保证控制通信电路板发生的控制用 PWM 波正确进入驱动系统。

控制用单片机的编程采用 C51 语言在 u Vision2 环境下进行下位机开发，用 89s52（51 内核）实现。控制用单片机安装在驱动控制电路板上，控制用单片机独立于上位机系统，配合面板按键工作。当进行控制实验时，用该单片机发生控制用 PWM 波，进而驱动电动机。当进行控制算法实验时，保证控制通信电路板发生的控制用 PWM 波正确进入驱动系统。控制用单片机程序模块由 HCTL-1100 操作模块和通信模块组成。

1. HCTL-1100 操作模块

HCTL-1100 是美国 Agilent 公司生产的高性能通用运动控制系列芯片，它内部集成了数字滤波器、换向器等可编程器件，因而可以使系统用最少的元件来实现运动控制，这样不仅减轻了上位主机的控制负担，同时还具有很大的灵活性。HCTL-1100 是一个通用的电动机控制芯片，用于控制直流有刷、直流无刷以及步进电动机。HCTL-1100 接收来自主控制器的命令和增量式编码器的反馈脉冲，采用 8 位地址/数据总线接口与主机相连。HCTL-1100 内部包含了一个可编程数字控制器，64 个 8 位寄存器，其功能的实现是通过对寄存器和控制器的操作完成的。控制用单片机最主要的工作就是对 HCTL1100 进行读写操作，读写操作以严格按照时序进行。HCTL1100 的内部功能框图如图 8-3 所示^[2-3]。

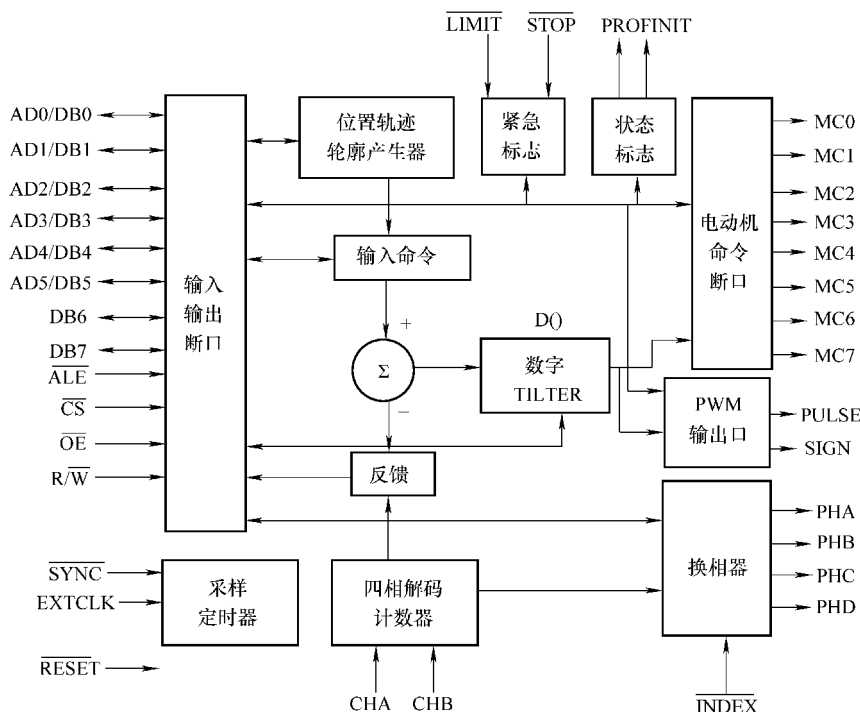


图 8-3 HCTL1100 的内部功能图

(1) HCTL-1100 的引脚功能

HCTL-1100 具有 40-PDIP 和 44-PLCC 两种封装形式，如图 8-4 分别为这两种封装形式的引脚排列图。HCTL-1100 运动控制芯片各主要引脚的功能说明如下：

同步引脚（SYNC）：该引脚只在 INIT/IDLE 模式下有效。可用来同步两个或多个芯片。将多个芯片的 SYNC 端连在一起接收上位机的命令可实现同步控制。

限位引脚（LIMIT）：用于输入紧急限位标志，低电平有效。当该脚有效时，器件将进入 INIT/IDLE 模式，清除电机额定值并切断电动机电源，同时清除三个控制模式标志。当由 INIT/IDLE 模式重新进入控制模式后，这些标志可自动恢复原值。不用时应将该脚连到 VDD，否则会引起触发而使系统进入错误的紧急状态。

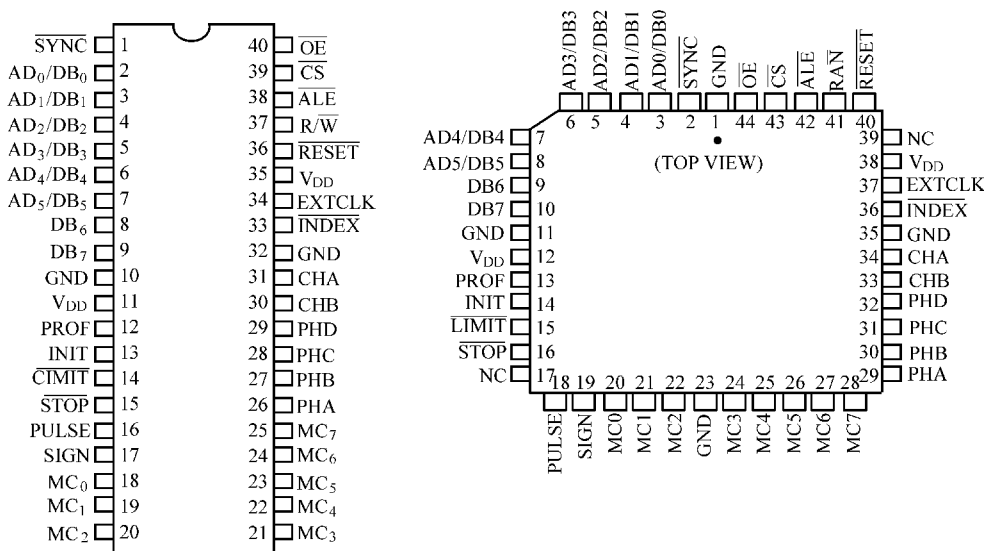


图 8-4 HCTL1100 的引脚功能

停止引脚 (STOP): 该停止标志仅适用于积分速度模式，低电平有效。有效时，系统将会减速直到停止，同时将保持零速度状态直到该位变高和新的速度命令被接收为止。不用时也应将该端连到 VDD。

编码器输入引脚 (CHA, CHB, INDEX): 这三个引脚可用于接收增量式编码器的 TTL 电平输出。其中通道 A 和 B 所接收的信号将被编码成 24 位位置计数器信号，并从换向器输出。

电动机额定值端口 (MC0 - MC7): 八位电动机额定值端口由寄存器 R08H 组成，该寄存器的数据将直接传到外部引脚 MC0 - MC7。其中 MC7 是最高有效位，R08H 可读可写，写入时通常为 INIT/IDLE 状态。

脉宽调制 (PWM) 输出端口 (PLUSE, SIGN): PWM 端口由 PULSE 和 SIGN 引脚组成。该端口输出的电动机额定值为带有正负极性的脉宽调制信号。

梯度式引脚 (PROF): 该引脚与内部状态寄存器的软件标志位 4 相连，也可由标志寄存器 (R00H) 的位 0 表示。当引脚和标志位为高时，表明系统正处于梯度式移动状态。移动结束后，控制器将对该标志位清零。应当注意标志位清零的那一刻并不表明电动机已停止工作。因此，它只是命令描述的完成而不是实际操作的完成。如果电动机在移动中发生延迟而不能跟上命令所要求的物理移动，则标志位将在移动完成前被清零。

初始化/空闲引脚 (INIT): 该引脚的状态用于表明 HCTL - 1100 正处于初始化/空闲状态，它应与内部状态寄存器的位 5 相连，也可由标志寄存器 (R00H) 的位 1 表示。

换向器引脚 (PHA ~ PHD): 这些引脚用于驱动无刷或步进电动机。使用时可以对这四个引脚进行编程，以用于驱动多相电动机的每个线圈。

(2) HCTL - 1100 的内部寄存器

HCTL - 1100 的工作过程由内部的 64 个八位寄存器控制，其中 35 个寄存器用于进行初

始化或命令字的写入等操作, 剩余的 29 个作为临时寄存器, 但这些寄存器不能直接使用。可用的 35 个寄存器可分为一般控制、输出、滤波器和换向器四类。一般控制寄存器又包括标志寄存器 (R00H)、可编程计数器 (R05H)、状态寄存器 (R07H)、采样定时寄存器 (R0FH)、读实际位置和预置实际位置寄存器等。而输出寄存器 (包括禁止极性转换, 8 位电动机额定值和脉宽调制额定值输出) 则包括电动机额定值寄存器 (R08H) 和脉宽调制额定值寄存器 (R09H) 两种。HCTL-1100 的内部寄存器的功能框图如图 8-5 所示。

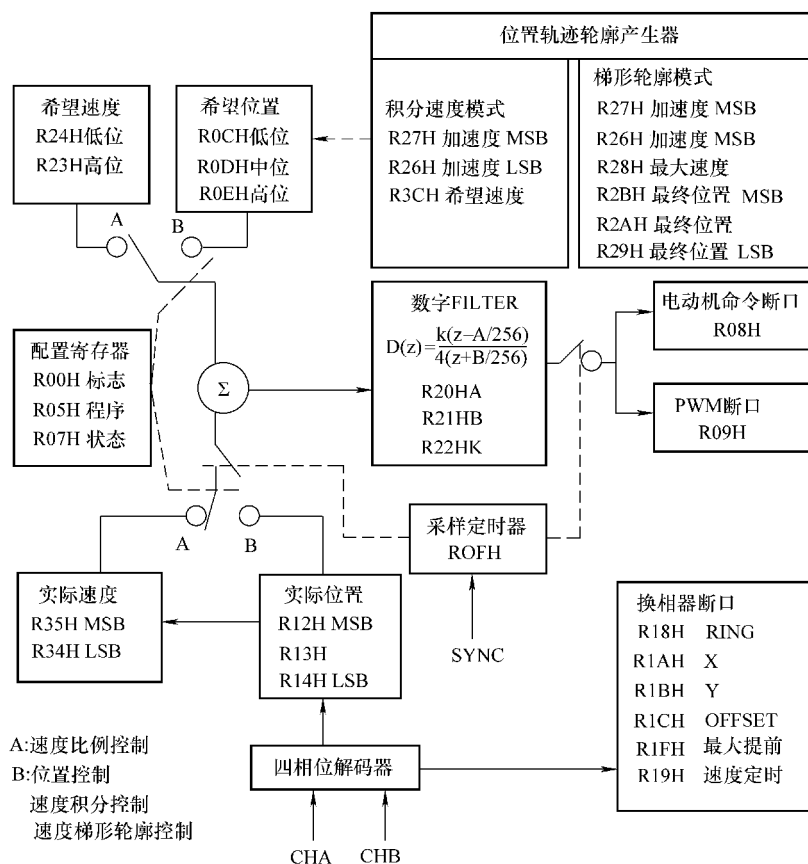


图 8-5 HCTL-1100 的内部寄存器

(3) HCTL-1100 的 I/O 操作

HCTL1100 可视为主处理器的一个 8 位寄存器或外部 RAM, 可进行读写操作。这些寄存器中的数据控制 HCTL1100 的操作。主处理器通过一根 8 位的数据总线与 HCTL1100 进行通信。4 根 I/O 控制线 ALE, CS, OE, R/W 完成数据传输。任意 I/O 操作都以 ALE 信号中断为起点, 然后从外部总线读取信号到地址锁存器。ALE 置高或 CS 置低都将停止数据采样到地址锁存。在 ALE 置高后, CS 置低将从外部总线采样数据到数据锁存器。CS 置高后, 将停止数据采样并进入内部同步进程。在进行写操作时, 数据锁存其中的数据被写入到指定的地址。在进行读操作时, 指定地址的数据写入到一个内部输出锁存器中。OE 选通后, 该所存其中的数据传送到外部总线。OE 信号和内部输出锁存使得 I/O 灵活操作并且消除了读操作

时的总线冲突。重要的在于，主机可以在 HCTL1100 的一个采样时间内完成更多的 I/O 操作。每个 I/O 操作都可以中断 HCTL1100 一个时钟周期的内部编码。尽管对每个采样时间的 I/O 操作都分配了额外的时间，但是减少该值对于输入采样时间寄存器的值可以进行简化。对 HCTL1100 的核心操作，即对 HCTL1100 内部寄存器进行读写操作，因此频繁调用读写子程序。在读写子程序的编写过程中应严格按照 I/O 时序表，对 HCTL1100 进行控制。HCTL1100 提供了三组不同的时间顺序结构，在此我们可以选用任意一组时序结构进行编程，在该实验系统的读写子程序的编制中时序采用 ALE 与 CS 交迭的方式。

(4) HCTL-1100 的控制模式

HCTL-1100 有三种设置路径和四种控制模式供用户选择。三种设置路径包括复位、初始化/空闲和排列对齐。四种控制模式分别为：位置控制、比例速度控制、梯形速度控制和积分速度控制。可以利用 HCTL-1100 提供的不同控制模式对关节电动机进行控制从而规划机械手臂的运动轨迹。本文用到的位置模式其流程如图 8-6 所示。

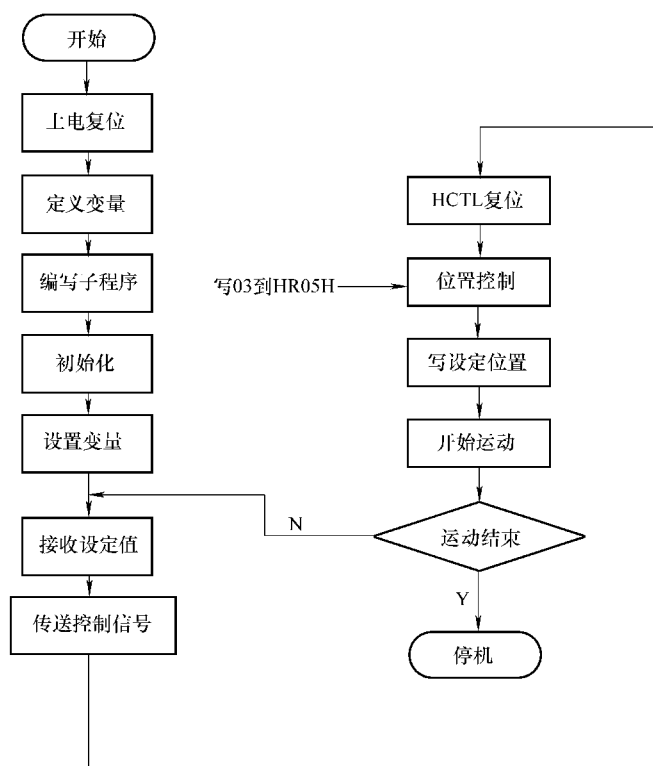


图 8-6 位置控制模式流程图

2. 与上位机串口通信模块

基于该系统的验证试验和算法实验都由控制用单片机和上位机协调控制完成。上位机发送命令通过串行通信到达控制用单片机，然后由单片机向控制芯片 HCTL1100 发送读写寄存器指令。电动机运动时运动信号由 HCTL1100 采集后进行处理，或自身完成运算再次发送指令，或经串行通信把电动机的运动信号上传给上位机，由上位机进行处理后再次发送指令，

完成操作。以下分别就内置算法实验和外置算法实验加以说明。

(1) 内置算法实验编程

内置算法实验通过上位机发送设定值，经串行通信传给单片机，然后单片机把设定值写入运动控制芯片 HCTL1100 内部的寄存器，HCTL1100 接收设定值后由 PWM 口输出命令控制电动机运动。电动机运动时其运动信号由 HCTL1100 采集后进行处理，由于运动控制芯片 HCTL1100 本身内置了四种控制模式，可以通过这些内置算法完成对电动机的控制。在此需要解决的问题主要集中在串行数据的正确传输，设定值的设定和判定以及上位机界面操作上。每次从上位机界面输入的设定值经转换后由上位机发送给串口 COM1，单片机接收该数据。每组数据由六个 8 位数组成，其定义如下：

1) 第一个数为 FF 为识别码，确保数据的正确传输。

2) 第二个数为数据判定位，用以对每组数据进行识别，8 位的具体定义为 7—5 位为通道识别位，用选择六个自由度某一通路进行操作，通道定义为：

通道 1	通道 2	通道 3	通道 4	通道 5	通道 6
000	001	010	011	100	101

4—3 位为模式选择位，用以选择 HCTL1100 内置四种控制模式的一种对电动机进行控制，对应关系为：

位置控制	比例速度	积分速度	梯形轨迹
模式 PM	控制模式	控制模式	控制模式
00	01	10	11

2 位为方向控制位，在位置控制模式下 0 定义为正向运动，1 定义为反向运动。

1—0 位为梯形轨迹控制模式的扩展位。

3) 第三、四、五个数为实际传输的数据，即设定初值，第三个数定义为高位数据 (MSB)，第五个数定义为低位数据 (LSB)。

4) 第六个数为 FF 为识别码，确保数据的正确传输。

上位机在确认数据传输正确后，加入 24V 直流电源，驱动电动机运动，完成目标设定。

(2) 外置算法实验编程

外置算法实验利用 HCTL1100 的 PWM 寄存器，在 HCTL1100 的初始化空闲模式下直接对 PWM 寄存器进行操作。电动机的运动信号由 HCTL1100 采集后，经由串行通信返回给上位机，上位机自行编制算法对返回值进行处理，然后发送命令给 PWM 寄存器对电动机进行控制。编程工作集中在上位机算法的编制以及对数据收发。以下给出整体编程的流程图：上位机图形界面输入设定，经数据转换后直接向 PWM 寄存器输入数据，给出一定占空比控制电动机运动。电动机运动后，其运动信号返回后经上位机算法处理再次向 PWM 寄存器输入数据，对 PWM 波的占空比进行调整，实现对电动机的控制。上位机每次发送的数据为 2 个 8 位数，包含通道信息和 PWM 输入值；接收数据为 2 个 8 位数，返回电动机的运动信息。发送数据的定义如下。

1) 第一个数为通道判别和数据识别, 具体定义为:

7—5 位为通道识别位, 用一般定选择六个自由度的某一通路进行操作, 通道定义为:

通道 1	通道 2	通道 3	通道 4	通道 5	通道 6
000	001	010	011	100	101

4 位不用, 3—0 位为 F 即 1111, 确保数据的正确传输。

2) 第二个数为 PWM 输入值, 每个输出值对应一定占空比, 其对应关系见 HCTL1100 芯片资料。

接收数据的定义如下: 上位机每次接收 3 个数据, 为电动机运动信号。第一个数为电动机运动的高位信息 (MSB), 第三个数为电动机运动的低位信息 (LSB), 第三个数为电动机运动的方向信息。

8.2.4 电动机驱动子系统

驱动电路是根据原系统可逆桥式电路的设计思想, 基于 PWM 波直流调速的基本原理而设计的。驱动部分为模拟电路和逻辑控制电路, 两部分相互独立。下面以 1 号电动机为例, 分析其逻辑控制功能及电动机的驱动功能的实现, 如图 8-7 所示。

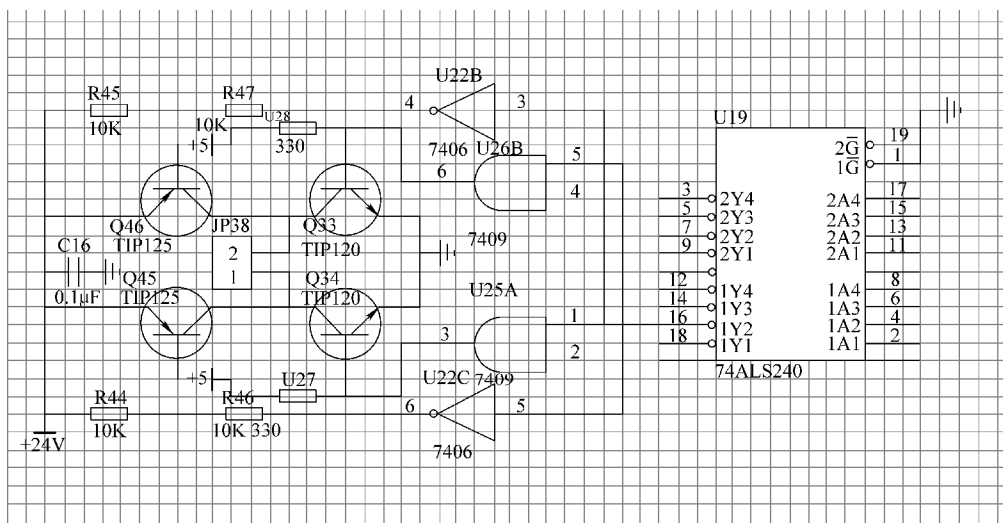


图 8-7 驱动电路图

PWM 信号与方向控制信号通过上拉电阻分别从 74LS240 的 2 号、4 号引脚接入, 通过 74LS240 的反向作用从 18 号、16 号引脚输出, 与与门 74LS09 及非门 74LS06 的输入端相连。当 PWM 信号与方向控制信号分别为 0、0 时, 与门 74LS09 及非门 74LS06 输入分别为:

74LS06 - 3: 1 74LS06 - 4: 0 74LS06 - 5: 0 74LS06 - 6: 1

74LS09 - 1: 1 74LS09 - 2: 1 74LS09 - 3: 1 74LS09 - 4: 1

74LS09 - 5: 0 74LS09 - 6: 0

通过电阻 R44 - R47 及 U27U28 电阻排的上拉作用, 到达晶体管基极的电流方向分别为:

Q33 流出, Q34 流入; Q45 流入, Q46 流出, 因此判定晶体管 Q34、Q46 导通, Q33、Q45 截止。此时电动机导通, 开始工作, 电流从电动机的 2 号引脚流向 1 号引脚。

当 PWM 波位改变而方向控制位不发生变化时, 即 74240 的 2 号引脚为 1, 4 号引脚仍保持 0 时, 其逻辑门的各引脚电平为:

74LS06 - 3: 1 74LS06 - 4: 0 74LS06 - 5: 0 74LS06 - 6: 1
74LS09 - 1: 1 74LS09 - 2: 0 74LS09 - 3: 0 74LS09 - 4: 0
74LS09 - 5: 0 74LS09 - 6: 0

晶体管基极的电流方向分别为: Q33 流出, Q34 流出; Q45 流入, Q46 流出, 因此判定晶体管 Q46 导通, Q33、Q34、Q45 截止。此时电动机无电流通过, 停止工作。

同理, 当方向控制信号改变时, 晶体管 Q33、Q45 导通, Q34、Q46 截止, 电流从电动机的 1 号引脚流向 2 号引脚。此时, 电动机反向运动。反映到电动机两端的波形为幅值为 24V, 占空比与输入占空比相反的 PWM 波。

通过改变输入端 PWM 波控制信号的高低电平, 就可以控制桥路的导通与否, 进而控制电动机的开关; 通过改变 PWM 波控制信号的占空比, 就可以控制桥路的导通时间, 进而控制电动机的开关时间。由于 PWM 信号为 20kHz 的循环方波, 切换频率极快, 从宏观上来根本无法观察到电动机的起停, 只能看到电动机的转速发生变化。至此, 通过 PWM 信号与方向控制信号的电平变化, 实现了对电动机的转速和方向进行控制。

8.2.5 编码器和电动机子系统

该机械手模型的脉冲编码器采用的是关节位置传感器。机器人关节位置传感器使机器人最早、最广泛使用的传感器, 它用于检测机械手各运动关节的旋转角度或伸缩距离。对于旋转关节, 通常采用光敏轴角传感器检测关节角度, 这种传感器分为绝对式编码传感器和增量式编码传感器两类。这两种传感器都是利用光电转换原理, 将随轴转动的码盘角度或者码盘角度的变化量转换成电脉冲, 并由此得到角度信号。

增量式光敏轴角传感器的工作原理如图 8-8 所示, 它采用的码盘沿圆周排列了一圈黑白相间的条码, 一对发光、光敏二极管安装在码盘两侧。当黑色条纹转到两个二极管之间时光线被遮挡, 光敏二极管输出信号为 0, 否则为 1。因此随着码盘的转动得到一串脉冲信号, 每个埋藏对应于一组黑白条纹, 通过累积脉冲数就可以得到转动的角度。

为了区分转动的方向, 通常采用两对发光、光敏二极管同时进行检测。这两对二极管安装位置存在一个角度差, 因此测到的埋藏信号相互之间有一个相位差, 通过相位插可以判断出转动的方向如图 8-9 所示。

另外, 为了避免计数器的积累误差, 还安装了第三对发光、光敏二极管, 专门用于检测零位。即在码盘上专门刻了一个小透光孔, 使码盘每次转到零位时有第三对二极管测到一个脉冲, 表明码盘转了一圈, 并对第一、二对二极管的输出累计脉冲清零。

绝对式光敏轴角编码器的工作原理与增量式编码器相似, 其不同之处在于码盘上的编码是由若干组环形码道组成, 每条码道代表编码中的一位, 而码道中的条纹分布根据码盘要求的 0、1 位置来决定。另外, 在绝对式编码传感器中安装了多对发光二极管来检测每条码道

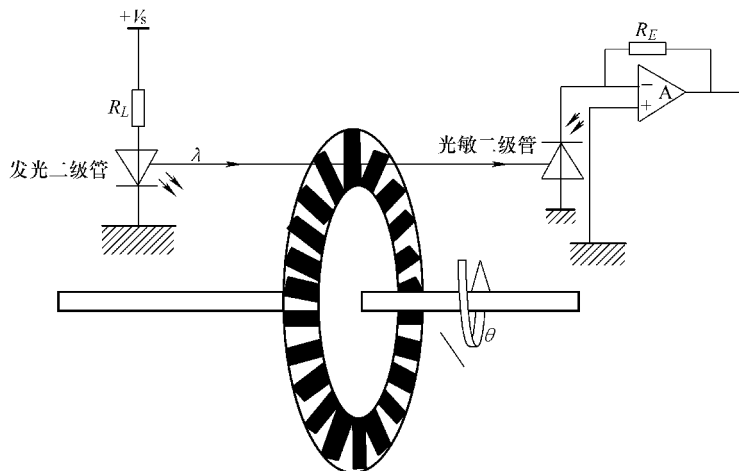


图 8-8 编码器的工作原理

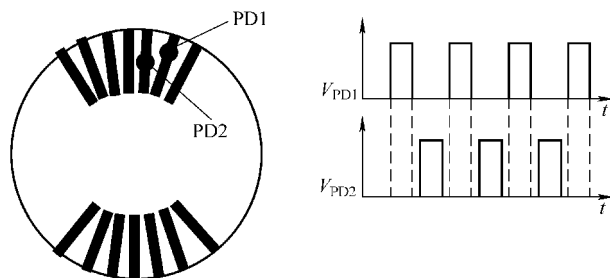


图 8-9 编码器的相位差

的 0、1 信号。码盘转动到任意位置都可以得到一组由 0、1 构成的编码，这一组码与转角之间存在着明确的对应关系，与转动过程无关，因此称为绝对编码传感器。由于绝对轴角传感器可以在任意位置开始工作，无需清零，也不存在积累误差，因此特别适合电源频繁开关和不允许机器经常进行回零校正的场合。实验中的编码器采用这种绝对式光电编码器。

8.3 机器人的控制方法

8.3.1 PD 基础控制

在工程实际中，应用最为广泛的调节器控制规律为比例、积分、微分控制，简称 PID 控制，又称 PID 调节，它实际上是一种算法。PID 控制器问世至今已有近 70 年历史，它以其结构简单、稳定性好、工作可靠、调整方便而成为工业控制的主要技术之一。当被控对象的结构和参数不能完全掌握，或得不到精确的数学模型时，控制理论的其他技术难以采用时，系统控制器的结构和参数必须依靠经验和现场调试来确定，这时应用 PID 控制技术最为方便。即当我们不完全了解一个系统和被控对象，或不能通过有效的测量手段来获得系统参数

时,最适合用PID控制技术。PID控制,实际中也有PI和PD控制。PID控制器就是根据系统的误差,利用比例、积分、微分计算出控制量进行控制的。实际应用中只使用其中的一项或者两项就可以达到控制要求,在机械手中常常使用PD控制,表达式如下

$$u(k) = k_p e(k) + k_d [e(k) - e(k-1)] \quad (8-1)$$

在PD控制算法中,存在着比例和微分两种控制作用。这两种控制作用的特点如下^[4]:

(1) 比例控制作用的特点 系统误差一旦产生,控制器立即就有控制作用,使得被PID控制的对象向着减小误差的方向变化,控制作用的强弱取决于比例系数 K_p 。缺点是对于具有自平衡能力的被控对象存在静差。加大 K_p 可减小静差,但 K_p 过大,会导致系统超调增大,使得系统的动态性能变坏。

(2) 微分控制作用的特点 通过对误差进行微分,能感觉出误差的变化趋势,增大微分控制作用可加快系统响应,使超调减小。缺点是对于干扰同样敏感,使系统对干扰的抑制能力降低。

根据被控对象的不同,适当地调整PD参数,可以获得比较满意的控制效果。因为其算法简单,参数调整方便,并且有一定的控制精度,因此它已成为当前最为普遍采用的控制算法。PD控制算法也有它的局限性和不足,由于PID算法只用在系统模型参数为非时变的情况下,才能获得理想的效果。当一个调好参数的PD控制器被应用到模型参数时变系统时,系统的性能会变差,甚至不稳定。另外,在对PD参数进行整定的过程中,PD参数的整定值是具有一定局限性的优化值,因此这种控制作用无法从根本上解决动态品质和稳态精度的矛盾。

8.3.2 模糊补偿控制

模糊控制器具有无需建立被控对象的数学模型,对被控对象的时滞、非线性和时变性具有一定的适应能力等优点,同时对噪声也具有较强的抑制能力,即鲁棒性好。但它也有一些需要进一步改进和提高的地方。模糊控制器本身能消除系统稳态误差的性能比较差,难以达到较高的控制精度。尤其是在离散有限论域设计时,更为明显。在上面的机械手补偿控制中使用下面简单的模糊语句

$$\begin{aligned} \text{If } \bar{x}(k) \text{ is } A_1 \text{ Then } F_1 \text{ is } B_1 \\ \text{If } \bar{x}(k) \text{ is } A_2 \text{ Then } F_1 \text{ is } B_2 \\ \text{If } \bar{x}(k) \text{ is } A_3 \text{ Then } F_1 \text{ is } B_3 \\ \text{If } \bar{x}(k) \text{ is } A_4 \text{ Then } F_1 \text{ is } B_4 \\ \text{If } \bar{x}(k) \text{ is } A_5 \text{ Then } F_1 \text{ is } B_5 \end{aligned} \quad (8-2)$$

式中 $\bar{x}(k) = x(k) - x(k-1)$;

F_1 ——对不确定摩擦等未建模动态的补偿。

基于上面的模糊规则,构建下面的模糊系统

$$\hat{F}_1(\bar{x}(k)) = \frac{\sum_{j=1}^M \bar{F}^j(k) \mu_{A_j}(\bar{x}(k))}{\sum_{j=1}^M \mu_{A_j}(\bar{x}(k))} \quad (8-3)$$

8.3.3 模糊 PD 控制的实验

用上面的控制器进行了如下控制实验。

实验目标：设定机器人腰部环节转角 30° ，通过两种外加控制方法进行调节，对调节误差进行分析，以便分析控制方法。

在机器人的上位机界面输入的是十进制的设定角位置，而码盘检测到的实际角位置是用十六进制表示。十进制的角位置和十六进制的角位置经测量有如下的对应关系

$$y_{16 \rightarrow 10} = \alpha x_{10} \quad (8-4)$$

在上式中，测量值经过统计回归得 $\alpha = 2.1$ 。在这个实验里，设定转角 30° ， 30° 对应码盘十六进制为 3F。

ZEBRAZERO 机械手是美国 IMI 公司生产的一台小型低惯量机械手，只能在适中的最大速度、加速度下运行，最大承受力 980N，力矩 $39.2\text{N} \cdot \text{m}$ 。正常工作时，力限最大为 19.6N ，力矩 $4.9\text{N} \cdot \text{m}$ 。在这台机械手上验证本文提出的算法，控制系统结构简图如图 8-10 所示^[5]。

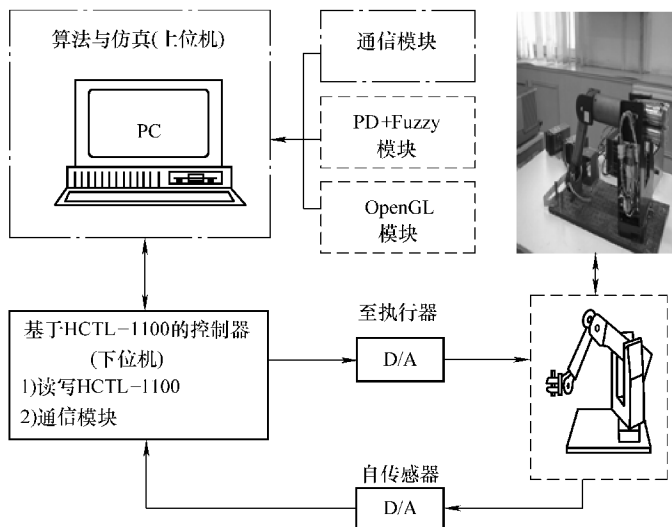


图 8-10 控制系统的结构图

核心部件 HCTL - 1100 是美国 Agilent 公司生产的高性能通用运动控制系列芯片，它内部集成了数字滤波器、换向器等可编程器件，内部具有内置控制算法，因而可以使系统用最少的元件来实现运动控制。为了验证本文的算法，我们屏蔽了 HCTL - 1100 芯片的内置控制算法，只用 HCTL - 1100 芯片的读写功能，而在上位机系统开发了外置控制算法。控制结果如表 8-1 所示。

表 8-1 算法控制结果

控制方法	期望值	编码器返回值
无补偿控制	3F	00 07 13 1E 28 2F 35 3E 41 42
		40 3E 3D 41 42 41 3E 3B
有补偿控制	3F	00 06 12 1B 26 2D 33 37 3B 40
		40 40 40 40 40 40 40 40

在上面控制系统结构中：上位机和下位机通过通信模块完成信息通信；下位机向伺服系统传送控制指令，通过编码器反馈回所需要的位置信号。为完成本文算法，速度和加速度信号通过编码器的位置信号经差分得到。通过上面的码盘返回值可以看出，用无补偿控制方法系统被调节到 30° （十六进制表示为 3F）附近，而产生了波动。用本文提出的基于自适应模糊系统的双调节补偿控制方法系统被调节到 30.5° （十六进制表示为 40），但没有产生波动。码盘的返回值用三个字节表示，在本试验中只用到中位字节，而高、低位字节没用。如果把高、低位字节考虑进去，试验结果会更理想。

8.4 本章小结

一个实际机械系统除应满足特定功能外，还应考虑两个重要的动力学问题，即系统的稳定性和动力响应。一个正常的系统应是稳定的，且其动力响应必须限定在可接受的范围内，有时要求其振荡愈小愈好。但实际机械系统中常常存在非线性摩擦力、间隙、松动、结构尺寸公差、电动机输出噪声等的不确定扰动，这些不确定因数统称为不确定动态，在动力学方程中的主要表现形式是未建模动态和参数不确定性，它们常常引起系统的稳态误差和不确定振动等问题，必须加以有效控制。本章的模糊补偿实验验证了补偿这些不确定因数的重要性。

参考文献

- [1] 王永富, 柴天佑. 非线性系统的模糊建模与自适应控制及其应用 [D]. 沈阳: 东北大学, 2005.
- [2] 石光明, 谢雪梅. 通用电机数字控制器 HCTL1100 特性与应用 [J]. 电子技术应用, 1996, 5: 28-30.
- [3] 吕英杰. 运动控制系列芯片 HCTL1100 的原理及应用 [J]. 国外电子元器件, 2001, 12: 19-21.
- [4] 张化光, 何希勤. 模糊自适应控制理论及其应用 [M]. 北京: 北京航空航天大学出版社, 2002.
- [5] 王永富, 王殿辉, 柴天佑. 基于数据挖掘与系统理论建立摩擦模糊模型与控制 [J]. 自动化学报, 2010, 36 (3): 412-420.

第9章 多任务实时系统设计与控制补偿及其应用

9.1 嵌入式实时系统

9.1.1 嵌入式系统

国际电气和电子工程师协会（IEEE）对嵌入式系统的定义是：“用于控制、监视或者辅助操作机器和设备的装置”（原文为：Devices Used to Control, Monitor or Assist the Operation of Equipment, Machinery or Plants）。这主要是从应用对象上加以定义，从中可以看出嵌入式系统是软件和硬件的综合体，还可以涵盖机械等附属装置。目前，国内一个普遍被认同的定义是：以应用为中心、以计算机技术为基础、软件硬件可裁剪、适应应用系统对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。

嵌入式系统的核心是由一个或几个预先编程好以用来执行少数几项任务的微处理器或者微控制器组成。与通用计算机能够运行用户选择的软件不同，嵌入式系统上的软件通常是暂时不变的；所以经常称为“固件”。可从几方面来理解嵌入式系统：

1) 嵌入式系统是面向用户、面向产品、面向应用的，它必须与具体应用相结合才会具有生命力、才更具有优势。因此可以这样理解上述含义，即嵌入式系统是与应用紧密结合的，它具有很强的专用性，必须结合实际系统需求进行合理的裁减利用。

2) 嵌入式系统是将先进的计算机技术、半导体技术、电子技术和各个行业的具体应用相结合后的产物，这一点就决定了它必然是一个技术密集、资金密集、高度分散和不断创新的知识集成系统。所以，介入嵌入式系统行业，必须有一个正确的定位。例如 Palm 软件之所以在 PDA 领域占有 70% 以上的市场，就是因为其立足于个人电子消费品，着重发展图形界面和多任务管理。而风河公司的 Vxworks 软件之所以在火星车上得以应用，则是因为其高实时性和高可靠性。

3) 嵌入式系统必须根据应用需求对软硬件进行裁剪，满足应用系统的功能、可靠性、成本和体积等要求。所以，如果能建立相对通用的软硬件基础，然后在其上开发出适应各种需要的系统，是一个比较好的发展模式。目前的嵌入式系统的核心往往是一个只有几 K 到几十 K 微内核，需要根据实际的使用进行功能扩展或者裁减，但是由于微内核的存在，使得这种扩展能够非常顺利的进行。

实际上，嵌入式系统本身是一个外延极广的名词，凡是与产品结合在一起的具有嵌入式特点的系统都可以叫嵌入式系统，而且有时很难以给它下一个准确的定义。另外，嵌入式系统都带操作系统吗？从嵌入式系统发展的历史过程中可以看出，嵌入式系统可能有也可能没有。一个功能简单的嵌入式系统可能根本不需要操作系统；而对于一个功能复杂的嵌入

式系统,则需要操作系统。现在人们讲嵌入式系统时,某种程度上指近些年比较热的具有操作系统的嵌入式系统。

9.1.2 嵌入式实时系统概念

设备中的嵌入式系统常用于实现数据采集、信息处理和实时控制等功能。而采集、处理和控制往往是一个连续过程,必须在一定时间内完成,这就要求具有实时性,所以嵌入式系统普遍具有实时性要求。为完成实时性目的,在嵌入式实时系统中采用的操作系统称为嵌入式实时操作系统,它既是嵌入式操作系统,又是实时操作系统。作为一种嵌入式操作系统,它具有嵌入式软件共有的可裁剪、低资源占用和低功耗等特点。作为一种实时系统,它能够在指定或者确定的时间内完成系统功能和外部或内部、同步或异步时间做出响应的系统。

嵌入式实时操作系统建立在计算机机硬件系统之上,用户的一切开发工作都在其上进行,因此它可以称作一个平台。采用实时操作系统的用户不必花大量时间学习硬件,和直接开发相比起点更高,所有的硬件设置和资源访问都要通过实时操作系统的核心。硬件这样被屏蔽起来以后,用户不必清楚硬件系统的每一个细节就可以进行开发,这样就减少了开发前的学习量。实时操作系统还是一个标准化的平台,它定义了每个应用任务和内核的接口,它可以统一协调各个任务,优化 CPU 时间和系统资源的分配,使之不空闲、不拥塞。另外,这种标准化平台也促进了应用程序的标准化。应用程序标准化后便于软件的存档、交流、修改和扩展,为嵌入式软件开发的工程化创造了条件、减少开发管理工作量。

9.1.3 嵌入式实时系统的特点

1. 多任务类型

在实时系统中,不但包括周期任务、偶发任务、非周期任务,还包括非实时任务。实时任务要求满足时限,而非实时任务要求要使其响应时间尽可能的短。多种类型任务的混合,使系统的可调度性分析更加困难。

2. 多约束性

任务的约束包括时间约束、资源约束、执行顺序约束和性能约束。时间约束是任何实时系统都固有的约束。资源约束是指多个实时任务共享有限的资源时,必须按照一定的资源访问控制协议进行同步,以避免死锁和高优先级任务被低优先级任务堵塞的时间(即优先级倒置时间)不可预测。执行顺序约束是指各任务的启动和执行必须满足一定的时间和顺序约束。例如,在分布式端到端实时系统,同一任务的各子任务之间存在前驱/后驱约束关系,需要执行同步协议来管理子任务的启动和控制子任务的执行,使它们满足时间约束和系统可调度要求。性能约束是指必须满足如可靠性、可用性、可预测性和服务质量等性能指标。

3. 可预测性

可预测性是指系统能够对实时任务的执行时间进行判断,确定是否能够满足任务的时限要求。由于实时系统对时间约束要求的严格性,使可预测性成为实时系统的一项重要性能要求。除了要求硬件延迟的可预测性以外,还要求软件系统的可预测性,包括应用程序的响应时间是可预测的,即在有限的时间内完成必需的工作;以及操作系统的可预测性,即实时原

语、调度函数等运行开销应是有界的，以保证应用程序执行时间的有界性。

4. 可靠性

大多数实时系统要求有较高的可靠性。在一些重要的实时应用中，任何不可靠因素和计算机的一个微小故障，或某些特定强实时任务（又叫关键任务）超过时限，都可能引起难以预测的严重后果。为此，系统需要采用静态分析和保留资源的方法及冗余配置，使系统在最坏情况下都能正常工作或避免损失。可靠性已成为衡量实时系统性能不可缺少的重要指标。

5. 与外部环境的交互作用性

实时系统通常运行在一定的环境下，外部环境是实时系统不可缺少的一个组成部分。计算机子系统一般是控制系统，它必须在规定的时间内对外部请求做出反应。外部物理环境往往是被控子系统，两者互相作用构成完整的实时系统。大多数控制子系统必须连续运转以保证子系统的正常工作或准备对任何异常行为采取行动。

9.1.4 嵌入式实时系统的分类与调度

实时系统主要分为以下两类。

(1) 强实时系统 (Hard Real-Time) 在航空航天、军事、核工业等一些关键领域中，应用时间需求应能够得到完全满足，否则就可能造成飞机失事之类的重大安全事故，造成重大生命财产损失和生态破坏。因此，在这类系统的设计和实现过程中，应采用各种分析、模拟及形式化验证方法对系统进行严格的检验，以保证在各种情况下应用的时间需求和功能需求都能够得到满足^[1]。

(2) 弱实时系统 (Soft Real-Time) 某些应用虽然提出了时间需求，但实时任务偶尔违反这种需求对系统的运行以及环境不会造成严重影响，如视频点播系统、信息采集与检索系统就是典型的弱实时系统。在视频点播系统中，系统只需保证绝大多数情况下视频数据能够及时传输给用户即可，偶尔的数据传输延迟对用户不会造成很大影响，也不会造成像飞机失事一样严重的后果。

为了精确管理“时间”资源，已达到实时性和预测性要求，并能够满足实时系统的新要求，需用实时调度理论对任务进行调度和可调度性分析。任务调度技术包括调度策略和可调度性分析方法，两者是紧密结合的。任务调度技术研究的范围包括任务使用系统资源（包括处理机、内存、I/O、网络等资源）的策略和机制，以及提供判断系统性能是否可预测的方法和手段。例如，什么时候调度任务运行、在哪运行（当系统为多处理机系统或分布式系统时）、运行多长时间等；以及判断分析用一定参数描述的实时任务能否被系统正确调度。

给定一组实时任务和系统资源，确定每个任务何时何地执行的整个过程就是调度。在非实时系统中，调度的主要目的是缩短系统平均响应时间，提高系统资源利用率，或优化某一项指标；而实时系统中调度的目的则是要尽可能地保证每个任务满足他们的时间约束，及时对外部请求做出响应。抢占式调度通常是优先级驱动的调度，每个任务都有优先级，任何时候具有最高优先级且已启动的任务先执行。一个正在执行的任务放弃处理器的条件为：自愿

放弃处理器（等待资源或执行完毕）；有高优先级任务启动，该高优先级任务将抢占其执行。除了共享资源的临界段之外，高优先级任务一旦准备就绪，可在任何时候抢占低优先级任务的执行。抢占式调度的优点是实时性好、反应快，调度算法相对简单，可优先保证高优先级任务的时间约束，其缺点是上下文切换多。而非抢占式调度是指不允许任务在执行期间被中断，任务一旦占用处理器就必须执行完毕或自愿放弃。其优点是上、下文切换少；缺点是在一般情况下，处理器有效资源利用率低，可调度性不好。

9.1.5 常见的嵌入式实时操作系统

1. Vxworks 实时操作系统

VxWorks 操作系统是美国 WindRiver 公司于 1983 年设计开发的一种嵌入式实时操作系统 (RTOS)，是嵌入式开发环境的关键组成部分。Tornado 是 WindRiver 公司推出的一套实时操作系统开发环境，类似 Microsoft Visual C，但是提供了更丰富的调试、仿真环境和工具。VxWorks 支持几乎所有现代市场上的嵌入式 CPU，包括 x86 系列、MIPS、PowerPC、Freescale ColdFire、Intel i960、SPARC、SH-4、ARM、StrongARM 以及 xScaleCPU。VxWorks 以其良好的可靠性和卓越的实时性被广泛地应用在通信、军事、航空、航天等高精尖技术及实时性要求极高的领域中，如卫星通信、军事演习、弹道制导、飞机导航等。在美国的 F-16、F/A-18 战斗机、B-2 隐形轰炸机和“爱国者”导弹上，甚至连 1997 年 7 月在火星表面登陆的火星探测器，2008 年 5 月在火星表面上登陆的“凤凰号”火星探测器，2011 年首飞的 X-47B 无人机，2012 年 8 月在火星表面上登陆的“好奇号”火星探测器上也都使用到了该实时操作系统。Vxworks 占据了嵌入式系统的大半江山，特别是在通信设备等实时性要求较高的系统中，几乎非 Vxworks 莫属。但因 Vxworks 价格很高，所以一些小公司或小产品中往往用不起，目前很多公司都在转用免费的嵌入式 Linux 系统。

2. Windows CE 实时操作系统

Windows CE 是一个抢先式多任务并具有强大通信能力的 Win32 嵌入式操作系统，是微软专门为信息设备、移动应用、消费类电子产品、嵌入式应用等非 PC 领域而从头设计的战略性操作系统产品。Windows CE 中的 C 代表袖珍 (Compact)、消费 (Consumer)、通信能力 (Connectivity) 和伴侣 (Companion)；E 代表电子产品 (Electronics)。与 Windows95/98、Windows NT 不同的是，Windows CE 是所有源代码全部由微软自行开发的嵌入式新型操作系统，其操作界面虽来源于 Windows95/98，但 Windows CE 是基于 WIN32 API 重新开发、新型的信息设备的平台。Windows CE 具有模块化、结构化和基于 Win32 应用程序接口和与处理器无关等特点。Windows CE 不仅继承了传统的 Windows 图形界面，并且在 WindowsCE 平台上可以使用 Windows95/98 上的编程工具（如 Visual Basic、Visual C++ 等）、使用同样的函数、使用同样的界面风格，使绝大多数的应用软件只需简单的修改和移植就可以在 Windows CE 平台上继续使用。

基于 Windows CE 构建的嵌入式系统大致可以分为 4 个层次，从底层向上依次是：硬件层、OEM 层、操作系统层和应用层。不同层次是由不同厂商提供的，一般来说，硬件层和 OEM 层由硬件 OEM 厂商提供；操作系统层由微软公司提供；应用层由独立软件开发商提

供。每一层分别由不同的模块组成，每个模块又由不同的组件构成。这种层次性的结构试图将硬件和软件、操作系统和应用程序隔开，以便于实现系统的移植，便于进行硬件、软件、操作系统、应用程序等开发的人员分工合作、并行开发。硬件层是指由 CPU、存储器、I/O 端口、扩展板卡等组成的嵌入式硬件系统，是 Windows CE 操作系统必不可少的载体。一方面，操作系统为嵌入式应用提供一个运行平台；另一方面，操作系统要运行在硬件之上，直接与硬件打交道并管理硬件。值得注意的是，由于嵌入式系统是以应用为核心的，嵌入式系统中的硬件通常是根据应用需要定制的，因此，各种硬件体系结构之间的差异非常大。“更小、更快、更省钱”几乎是所有嵌入式系统硬件的设计目标。OEM 层是逻辑上位于硬件和 Windows CE 操作系统之间的一层硬件相关代码。它的主要作用是对硬件进行抽象，抽象出统一的接口，然后 Windows CE 内核就可以用这些接口与硬件进行通信。

3. Linux 实时操作系统

嵌入式 linux 是将日益流行的 Linux 操作系统进行裁剪修改，使之能在嵌入式计算机系统上运行的一种操作系统。嵌入式 linux 既继承了 Internet 上无限的开放源代码资源，又具有嵌入式操作系统的特性。嵌入式 Linux 的特点是版权费免费，而且性能优异，软件移植容易，代码开放，有许多应用软件支持，应用产品开发周期短，新产品上市迅速，实时性能，稳定性好，安全性好等特点。嵌入式 linux 主要的应用领域有信息家电、PDA、机顶盒、数字电话、网络设备、远程通信、医疗电子、交通运输、计算机外设、工业控制、航空航天等领域。该部分将在下节进行详细介绍。

9.2 实时多任务 Linux 操作系统

9.2.1 Linux 简介

Linux 的诞生可以追溯到 1991 年，当 Linus 还是芬兰赫尔辛基大学的一名学生时，他对当时为教学而设计的 Minix 操作系统提供的功能不满意，于是他决定自己写比 Minix 更强大的类 UNIX 操作系统来取代 Minix，后来这个类 UNIX 操作系统就是 Linux。Linus 从一开始就决定自由扩散 Linux，他把源代码发布在 Internet 上，随即就引起爱好者的注意，他们通过 Internet 加入了 Linux 的内核开发工作，一大批高水平程序员的加入，使得 Linux 得到迅猛发展，他们为 Linux 修复错误、增加新功能，不断尽其所能地改进它。现在，Linux 凭借优秀的设计，不凡的性能，加上 IBM、Intel、AMD、DELL、Oracle、Sybase 等国际知名企业的大力支持，市场份额逐步扩大，逐渐成为主流操作系统之一。Linux 其本质只是操作系统的核心，负责控制硬件、管理文件系统、程序进程等。Linux Kernel（内核）并不负责提供用户强大的应用程序，没有编译器、系统管理工具、网络工具、Office 套件、多媒体、绘图软件等，这样的系统也就无法发挥其强大功能，用户也无法利用这个系统工作，因此有人便提出以 Linux Kernel 为核心再集成搭配各式各样的系统程序或应用工具程序，组成一套完整的操作系统，经过如此组合的 Linux 套件即称为 Linux 发行版。

Linux 操作系统在短短的几年之内得到了非常迅猛的发展，这与 Linux 具有下面良好特

性是分不开的。

(1) 开放性 系统遵循世界标准规范,特别是遵循开放系统互连(OSI)国际标准。凡遵循国际标准所开发的硬件和软件,都能彼此兼容,可方便地实现互连。它是基于GPL(General Public License)的架构之下,任何人都可以免费的使用或者是修改其中的原始码。开放性架构对科学界来说是相当重要的,因为很多工程师由于特殊的需求,常常需要修改系统的原始码,使该系统可以符合自己的需求,而这个开放性的架构将可以满足各不同需求的工程师。

(2) 多用户 系统资源可以被不同用户各自拥有使用,即每个用户对自己的资源有特定的权限,互不影响。

(3) 多任务 多任务是现代计算机的最主要的一个特点。它是指计算机同时执行多个程序,而且各个程序的运行互相独立。Linux系统调度每一个进程,平等地访问微处理器。

(4) 良好的用户界面 Linux向用户提供了两种界面,用户界面和系统调用。Linux的传统用户界面是基于文本的shell命令行界面,shell有很强的程序设计能力,用户可方便地用它编制程序,从而为用户扩充系统功能提供了更高级的手段。

(5) 设备独立性 设备独立性是指操作系统把所有外部设备统一当成文件来看待,只要安装它们的驱动程序,任何用户都可以像使用文件一样,操纵和使用这些设备,而不必知道它们的具体存在形式。

(6) 提供了丰富的网络功能 Linux在通信和网络功能方面优于其他操作系统。其他操作系统不包含如此紧密地和内核结合在一起的连接网络的能力,也没有内置这些联网特性的灵活性。而Linux为用户提供了完善的、强大的网络功能。

(7) 可靠的系统安全 Linux采取了许多安全技术措施,包括对读、写进行权限控制、带保护的子系统、审计跟踪、核心授权等,这为网络多用户环境中的用户提供了必要的安全保障。

(8) 良好的可移植性 可移植性是指将操作系统从一个平台转移到另一个平台,使它仍然能按其自身的方式运行的能力。Linux是一种可移植的操作系统,能够在从微型计算机到大型计算机的任何环境中和平台上运行。

9.2.2 嵌入式Linux简介

嵌入式系统是指嵌入到对象体系中执行独立功能的专用计算机系统。嵌入式系统的三个基本要素是专用性、计算机系统和嵌入性,包括定时器、存储器、传感器、微控制器、微处理器等一系列器件与微电子芯片,以及嵌入在存储器中的控制应用软件、微型操作系统,共同实现诸如数据处理、实时控制、管理、监视、移动计算等各种自动化处理任务。嵌入式Linux是指对Linux小型化裁减后,能够固化在容量有限的存储器芯片中,应用于特定嵌入式场合的Linux操作系统。因为Linux自身的优点使其非常适合作为嵌入式操作系统,其优点表现为以下几个方面:

1) 开发成本小,源代码公开。大多数商业操作系统都是价格昂贵的,Linux是一个免费的性能优秀的操作系统且源代码公开,使不同领域和不同层次的用户可以根据自己的应用需要对内核进行裁减。

2) 可靠性高, 稳定性好, 容易移植。同目前流行的操作系统相比, Linux 是非常稳定和可靠的, 这一点已经得到很多用户的肯定。另外, Linux 也是一个多平台的操作系统, 可以运行在很多种处理器上, 这一点对于嵌入式的应用尤其重要。

3) 功能强大的开发工具。Linux 提供 C、C++、Java 等很多开发工具, 而且开发者可以免费获得。Linux 具备这么多的工具, 使得用户可以自行建立嵌入式系统的开发环境和交叉运行环境, 而且可以不用专门的仿真工具直接在 Linux 内核调试即可。

正是由于 Linux 具有上述的诸多优势, 使得它近几年在嵌入式应用领域发展迅速, 嵌入式实时 Linux 的开发和研究成为目前操作领域的一个热点。然而, Linux 作为嵌入式实时操作系统, 也存在着一些不足, 如 Linux 系统的实时性较差, 极大地限制了 Linux 系统在嵌入式方面的应用, 主要表现在以下方面。

1) 关中断问题。在系统调用中, 为了保护临界区资源, Linux 处于内核临界区时, 中断会被系统屏蔽, 这就意味着假如当前进程正处于临界区, 即使他的优先级较低, 也会延迟高优先级的中断请求。在实时应用中, 这是个十分严重的问题。

2) 进程调度问题。Linux 采用标准的 UNIX 技术使得内核是不可抢占的。采用基于固定时间片的可变优先级调度, 不论进程的优先级多么低, Linux 总会在某个时候分给该进程一个时间片运行, 即使同时有能够运行的高优先级进程, 它也必须等待低优先级进程的时间片用完, 这对一些需要高优先级进程立即抢占 CPU 的实时应用是不能满足需要的。

3) 时钟问题。Linux 为了提高系统的平均吞吐率, 将时钟中断的最小间隔配置为 10ms, 这对于一个周期性的实时任务, 间隔需要小于 10ms 时, 就不能满足实时任务的需要。假如要把时钟的间隔改小以满足周期性的实时任务的需要, 由于 Linux 的进程转换比较费时, 时钟中断越频繁, 而花在中断处理上的时间就越多, 系统的大部分时间是调用进程调度程式进行进程调度而不能进行正常的处理。

这些问题是 Linux 用于嵌入式环境时碰到的问题, 但是 Linux 操作系统本身的优势决定了 Linux 经过改造后仍然适合嵌入式环境, 将 Linux 经过改造, 使其成为一个能运行于嵌入式系统的操作系统是现在操作系统领域中研究的一个热点, 例如 RTLinux 和 RTAIX 就是 Linux 实时性改造的成功范例^[2-5]。

9.2.3 嵌入式实时系统 RTLinux 简介

RTLinux 是美国新墨西哥州大学计算机科学系 Victor Yodaiken 和 Micae Brannanov 开发的。它在 Linux 内核的下层实现了一个简单的实时内核, 而 Linux 本身作为这个实时内核的优先级最低的任务, 所有的实时任务的优先级都高于 Linux 系统本身的, 以及 Linux 系统下的一般任务。

RTLinux 的设计思想是: 应用硬件的实时约束将实时程序分割成短小简单的部分, 较大部分承担较复杂的任务。根据这一原则, 将应用程序分为硬件实时和程序两个部分。硬件实时部分被作为实时任务来执行, 并从外部设备拷贝数据到一个叫做实时有名管道 (RTFIFO) 的特殊 I/O 端口, 程序主要部分作为标准 Linux 进程来执行。它将从 RTFIFO 中读取数据, 然后显示并存储到文件中, 实时部分将被写入内核。设计实时有名管道是为了使实时任务在读和写数据时不被阻塞。

RTLinux 通过对标准 Linux 内核进行改造, 将 Linux 内核工作环境作了一些变化。在 Linux 进程和硬件中断之间, 本来由 Linux 内核完全控制。现在, 在 Linux 内核和硬件中断的地方加上了一个 RTLinux 内核的控制。Linux 的控制信号都要先交给 RTLinux 内核进行处理。在 RTLinux 内核中实现了一个虚拟中断机制, Linux 本身永远不能屏蔽中断, 它发出的中断屏蔽信号和打开中断信号都修改成向 RTLinux 发送一个信号。

到目前为止, RTLinux 已经成功地应用于航天飞机的空间数据采集、科学仪器测控和电影特技图像处理等广泛领域。RTLinux 开发者并没有针对实时操作系统的特性而重写 Linux 的内核, 因为这样做的工作量非常大, 而且要保证兼容性也非常困难。为此, RTLinux 提出了精巧的内核, 并把标准的 Linux 核心作为实时核心的一个进程, 同用户的实时进程一起调度。这样对 Linux 内核的改动非常小, 并且充分利用了 Linux 下现有的丰富的软件资源。

9.2.4 嵌入式实时系统 RTAI Linux 简介

RTAI (Real-Time Application Interface) 是由意大利米兰理工学院航天工程系发起开发的遵循 GPL 的开源项目。RTAI 是对 Linux 内核的硬实时扩展, 它遵循自由软件规范。它可以提供工业级的 RTOS 功能, 而且其所有的功能都可无缝的通过 GNU/Linux 环境访问。RTAI 支持的 CPU 类型包括 I386, PPC 和 ARM。

1. RTAI Linux 提供的功能

RTAI 为开发硬实时系统提供了丰富的功能。首先、提供多种对实时任务的调度方式, 它的调度器分为 UP (Uniprocessor), SMP (Symmetric Multiple Processor) 和 MUP (Multi-Uniprocessor) 三类。其中, UP 调度器用于单处理器的系统; SMP 调度器用于对称的多处理器的系统, 所有的 CPU 使用相同的时钟中断源, 也就是说, 任务在所有的 CPU 上都是以相同的模式运行; 而 MUP 调度器也是用于多处理器系统, 只不过每个 CPU 有自己独立的时钟中断源, 所以, 在不同的 CPU 上运行的任务可以以不同的时钟模式运行。在这三类调度方式中, 时钟运行的模式都有两种, 单触发模式 (one-shot mode) 和周期模式 (periodic mode)。另外, 它提供了丰富的任务间的通信方法, 主要包括邮箱机制 (Mailbox), 消息和过程调用机制 (Message and RPC), 信号量机制 (Semaphore), 管道机制 (FIFO), 共享内存机制 (Shared memory)。其中的管道机制和共享内存机制可用于 RTAI 的实时任务和 Linux 进程之间的通信。此外, RTAI 还提供了 LXRT 机制, 使得可以在 Linux 的用户空间创建软实时进程, 满足软实时应用的要求。RTAI 还在不断发展之中, 而且它在实时环境中的应用也越来越多, 它将来的主要发展方向包括实现更多硬件平台上的移植、为 RTAI 提供 C++ 开发环境、实时文件系统、对基于 Flash 的文件系统的支持、高级的实时内存管理机制等。

2. RTHAL 与 RTAI Linux

RTAI 在 Linux 上定义了一组实时硬件抽象层 RTHAL (Real Time Hardware Abstraction Layer), RTHAL 将所有需要的 Linux 内部数据和函数的指针集合到一个 RTHAL 的结构中。RTHAL 结构用于截取 Linux 硬件操作, 在双内核结构工作时, 它们可以被 RTAI 重定向, 以取代 Linux 中原有的函数; 同时 RTAI 只是用此程序界面与 Linux 进行沟通。通过这种方法

就可以把对 Linux 内核源码的改动程度降到最低,可以避免 RTLinux 方案对 Linux 内核源码改动过大的问题,便于在不同 Linux 版本之间的移植。在 RTHAL 架构下,RTAI 以独立实时内核的形式运行于硬件和 Linux 之间,它将 Linux 视为优先级最低的任务。

基于 RTHAL 的 RTAI 实时化方案的技术思想是构造一个简单的硬实时内核,应用的实时部分作为实时进程直接运行在这个硬实时内核之上;原来的 Linux 内核作为一个优先级最低的任务,也为这个实时内核所调度,应用的非实时部分作为非实时进程运行在 Linux 内核从而可以获得 Linux 内核所提供的一切服务。RTAI 当初设计的目的是为了解决 RTLinux 在不同 Linux 版本之间难于移植的问题,为此,RTAI 在 Linux 内核与硬件之间定义了一个实时硬件抽象层 RTHAL,实时任务通过这个抽象层提供的接口和 Linux 系统进行交互,这样给 Linux 内核增加实时支持时可以尽可能少地修改 Linux 内核源代码,同时减少跟踪移植到新版 Linux 的工作量。

RTAI 修改了内核中与体系结构相关的代码而形成了 RTHAL 层,RTHAL 通过接管硬件中断,建立软件模拟中断机制,将 Linux 内核与中断隔离。RTAI 引入一个实时调度器(schedule)来保证其实时性,将 Linux 内核当成一个任务模块来执行,调度那些与实时应用紧密相关的内核函数指针集中组织到 RTHAL 结构中,当系统要转入实时处理时,这些指针可以被 RTAI 动态转换。通常情况下,这些指针变量指向嵌入式 Linux 中的函数,此时系统相当于工作在嵌入式 Linux 下,RTHAL 不过增加了一个中间环节而已;当有实时要求时,即加载 RTAI 主模块后,该主模块的初始代码将这些指针变量修改为指向 RTAI 主模块中同名的函数,此时所有硬件中断都将被实时硬件抽象层接收,由相应函数来处理。当原有 Linux 试图屏蔽中断时,系统实际上并没有真正屏蔽中断,而只是设置了 RTHAL 中的某个结构,实时内核仍可处理中断。当发生中断时,RTAI 不会马上调用嵌入式 Linux 的中断处理程序,而是分两步进行处理:首先接受中断请求,判断 RTAI 中是否存在对应的中断服务程序,若存在则转至该服务程序,否则由实时内核的中断服务程序处理该中断;然后查看 RTHAL 中的软件模拟中断控制器,如果 Linux 允许中断,则将中断交给 Linux 中断服务程序,如果不允许中断则返回。

3. Adeos 与 RTAIXLinux

Adeos 的全称是 Adaptive Domain Environment for Operating System,它的目标是为操作系统提供了一个灵活的、可扩展的自适应环境,在这个环境下,多个相同或不同的操作系统可以共存,共享硬件资源;随着历史的发展,目前已经存在了不少优秀的操作系统,正是因为这些操作系统的共同存在发展,而产生了以下的问题:一方面,对于那些面向相同的用户,具有相似设计理念和功能的操作系统来说,它们之间是不兼容或不完全兼容的,例如,在 Windows 操作系统上的应用程序是不能直接在 Linux 上运行的;这就将用户(包括程序设计者和系统管理员)限制在了一个固定的软件环境下,用户在应用的选择上缺少灵活性。另一方面,由于最初的应用环境和面向的用户的不同,有些操作系统在设计理念和功能上完全不同;但是,随着计算机技术的发展,现在的一个计算机系统往往具有以前多个系统的功能,各种计算机系统之间的界限越来越模糊。例如,现在的终端电子设备对实时性的要求越来越高(如智能手机),而某些实时系统也对人机界面的交互有了更高的要求,这就会促使

这两种应用环境下的操作系统相互融合。目前,主要存在两类方法使多个操作系统运行在同一个系统上;第一类方法是模拟,例如,VMWare, Plex86 和 VirtualPC 等;它们都是在已有的操作系统上提供一个虚拟的硬件环境,在这个虚拟硬件环境下可以运行另外操作系统。这样,用户就可以充分的利用两个或多个系统所提供的功能和软件;但这种方法最大的缺点就是会极大地降低系统的性能,因为它包含三个软件层次:主机操作系统(Host Operating System) -> 虚拟硬件环境 -> 客户操作系统(Guest Operating System)。第二类方法,是在硬件上实现一个所谓的超微内核(nano-kernel),通过这个超微内核实现硬件的共享,然后再在这个内核上构建实用的操作系统,例如,SPACE、Cache kernel 和 Exokernel 等,但这些方法都没有考虑当前已经存在的操作系统和用户。

而 Adeos 是在已有的操作系统下插入一个软件层,通过向上层多个操作系统提供某些原语和机制而实现硬件共享。但是 Adeos 并不对硬件的使用强加任何的限制,上层的操作系统仍然可以自由的操作硬件,而不会因为 Adeos 的存在而有任何的约束(实际上,上层的操作系统可以完全不知道有 Adeos 的存在)。Adeos 除了可以实现操作系统对系统资源的共享之外,还可以用于新的操作系统的开发、操作系统内核的调试、跟踪等。目前, Adeos 是基于 Linux 内核实现的,主要的应用是在 Linux 的实时化方面,使基于 Linux 的系统能满足硬实时的要求。

9.3 系统的硬件设计

9.3.1 PET/CT 简介

PET 全称为正电子发射断层显像。该显像技术是将极其微量的正电子核素示踪剂注射到人体内,然后采用特殊的体外测量仪器(PET)探测这些正电子核素在人体全身各脏器的分布情况,通过计算机断层显像的方法显示人体的全身主要器官以及大脑、心脏的生理代谢功能和结构。同时,应用 CT 技术为这些核素分布情况进行精确定位,主要用于肿瘤、神经系统和心血管疾病的临床诊断。PET/CT 作为 PET 的升级产品,融合了 PET 和 CT 的优点,是目前影像诊断技术中最理想的结合,是分子影像学的重要技术代表。

从国内外生产厂家看:由于 PET/CT 是投资大、开发周期长、技术复杂的项目,该产品几乎被几大国际公司垄断。卫生部卫生经济研究所成本测算中心根据国家卫生部的要求,对我国引进 PET/CT 设备的医院使用效率、运营成本、收费等运行情况进行调研。得到的结果是:PET/CT 这类高端医疗设备市场主要集中在 GE、西门子、飞利浦国外供应商。由于国外厂家在该领域内技术的垄断,造成了我国医院购置 PET/CT 每台设备费用为 3000 万元左右、每年的维护费用平均约在 200 万元左右,设备和维护的高成本造成了病人的检查费用为一次 7000 元左右,即使按目前的收费标准仍不能回收成本^[6-8]。同国外相比,国内的 PET/CT 只是刚刚起步,与国外几家跨国公司差距的确较大。但国内的厂家和研究人员在 PET 和 PET/CT 研发上还是做了大量艰苦努力工作,如早年的中科院高能物理研究所,近年的北京亿仁赛博和沈阳东软集团。目前,沈阳东软集团第一代 PET 机经多年研制获得成功,

并取得美国 FDA 认证，现已投放到国内外市场。

9.3.2 PET 机电控制系统

PET 的机电控制部分尽管不直接参与投影数据的探测、编码、符合、采集和成像的整个流程，但同样在 PET 系统中起着举足轻重的作用。作为系统底层硬件与上层服务器之间通信以及底层硬件之间通信的交通枢纽，它负责接收来自上层服务器的命令，传给底层执行部件，命令这些部件做相应的动作，然后再将底层执行部件的完成情况和当前状态实时反馈给服务器作相应处理，同时它为前端数据采集子系统提供增益表和编码表等控制信息。可以说，机电控制部分几乎完成除机械设计和整个成像生成外的所有底层执行部件控制以及状态显示，各种模拟量、数据量的采集与控制工作，为系统成像作好各种准备工作。PET 的机电控制部的实际设备和实验设备如图 9-1 所示。

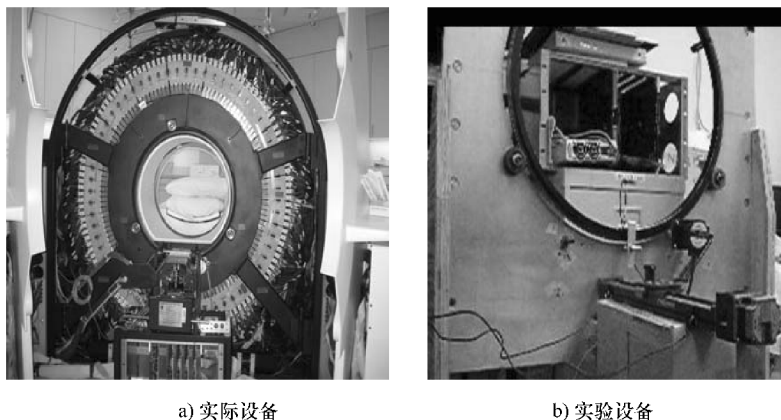


图 9-1 实际设备与实验设备

1. PET 的机电子系统

PET 的机电控制部分主要包括下面几个子系统：

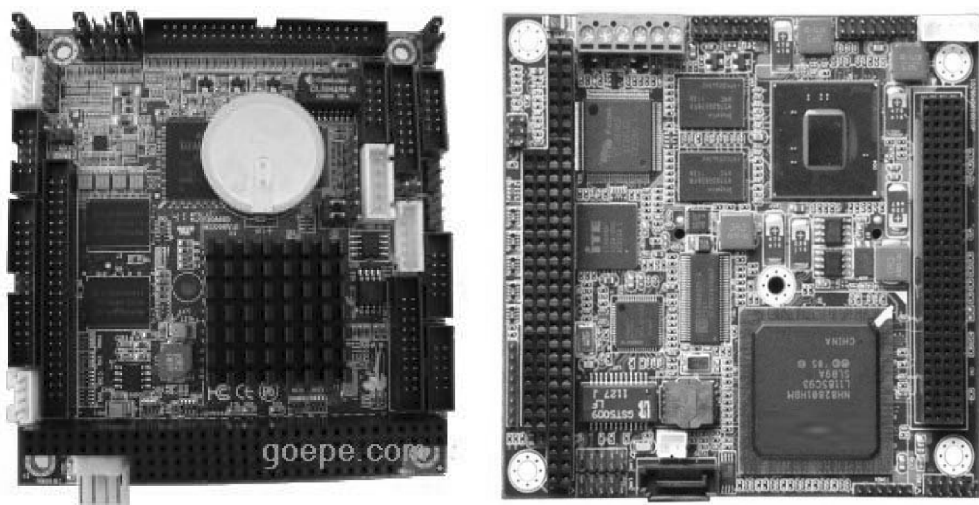
- 1) 扫描床子系统：该子系统主要完成扫描床的水平 and 升降运动，其中升降运动采用手动方式，而水平运动采用交流伺服电动机驱动形成闭环自动控制。
- 2) 棒源子系统：该子系统主要完成环和多棒源运动。棒源子系统的主要目的是完成衰减扫描，最后和发射扫描融合形成最后的 PET 图像。该子系统用步进电动机驱动形成开环自动控制。
- 3) 隔板子系统：该子系统主要完成隔板（Septa）从采集孔的移出和移进，如果隔板从采集孔移出，系统将做 3D 扫描；如果隔板移进采集孔，系统将做 2D 扫描。该子系统也用步进电动机驱动形成开环自动控制。
- 4) 摆动装置子系统：该子系统主要完成摆动装置（Wobble）的摆动，摆动装置的摆动将增加采集率和投影计数率。该子系统也用步进电动机驱动形成开环自动控制。

2. PC/104 与 PC/104 +

PC/104 是一种专门为嵌入式控制而定义的工业控制总线，近年来在国际上广泛流行，

被 IEEE 协会定义为 IEEE - P996.1。我们知道, IEEE - P996 是 PC 和 PC/AT 工业总线规范, 而从 PC/104 被定义为 IEEE - P996.1 就可以看出, PC/104 总线实质上是一种紧凑型, 小型化的 IEEE - P996, 其总线信号定义和 PC/AT 基本一致, 但电气和机械规范却完全不同, 是一种优化的、小型化、堆栈式结构的嵌入式控制系统。

PC104 有两个版本, 8 位和 16 位, 分别与 PC 和 PC/AT 相对应。在 PC104 总线的两个版本中, 8 位 PC104 共有 64 个总线管脚, 单列双排插针和插孔, P1: 64 针, P2: 40 针, 合计 104 个总线信号, PC104 因此得名。PC104+ 是专为 PCI 总线设计的, 可以连接高速外接设备。PC104+ 在硬件上通过一个 3X40 即 120 孔插座, PC104+ 包括了 PCI 规范 2.1 版要求的所有信号。为了向下兼容, PC104+ 保持了 PC104 的所有特性。我们选用 PC/104+ 作为 PET 计算机控制系统。PC104 和 PC104+ 如图 9-2 所示。



a) PC104 主板

b) PC104+ 主板

图 9-2 PC104 主板与 PC104+ 主板

3. 运动控制卡

在对 PET 机电电子系统进行运动控制时, 可以采用商用化的 PLC 控制器或运动控制卡。由于 PLC 的控制精度远低于板卡类, 我们采用了运动控制卡。运动控制卡通常是采用专业的运动控制芯片或高速 DSP 来满足一系列运动控制需求的控制单元, 其可通过 PCI、PC104 等总线接口安装到 PC 和工业 PC 上, 可与步进和伺服驱动器连接, 驱动步进和伺服电动机完成各种运动 (单轴运动、多轴联动、多轴插补等), 接收各种输入信号, 可输出控制继电器、电磁阀和气缸等元件。用户可使用 VC、VB 等开发工具, 调用运动控制卡函数库, 快速开发软件。运动控制卡是 PCI、cPCL、PXI 等总线形式的板卡, 通俗的说我们可以把它看成一个单片机, 有自己的算法, 可以通过 VC、VB、labview、BCB 等语言实现其功能。其中比较有名的是美国 PMAC 运动控制卡, 考虑成本因素, 最后我们选择了日本的 PCL6045B 运动控制卡。

4. 伺服器与伺服电动机

为了适应数字控制的发展趋势,运动控制系统中大多采用步进电动机或全数字式交流伺服电动机作为执行电动机。虽然两者在控制方式上相似(脉冲串和方向信号),但在使用性能和应用场合上存在着较大的差异。步进电动机是一种离散运动的装置,步进电动机是按照特性命名的,指它的转动是一步一步的。例如一个 360 分度的步进电动机,转一圈要走 360 步,每一步转 1° ,如果你要求它转半度,是不可能实现的。所以步进电动机是把一圈分成若干步,不累积误差,一般用做开环控制。在目前国内的数字控制系统中,步进电动机的应用十分广泛。随着全数字式交流伺服系统的出现,交流伺服电动机也越来越多地应用于数字控制系统中。伺服电动机一般是功率小,运行精确,能高速制动,惯量小,适合闭环控制,也就是能检测到实际位置和理论位置的误差,并消除误差。综上所述,交流伺服系统在许多性能方面都优于步进电动机。但在一些要求不高的场合也经常用步进电动机来做执行电动机。所以,在控制系统的设计过程中要综合考虑控制要求、成本等多方面的因素,选用适当的控制电动机。在 PET 扫描床子系统选用了交流伺服电动机,而棒源子系统、隔板子系统和摆动装置子系统选用了步进电动机。选用了日本富士的 RYC401D3 - VVT2 伺服器和日本富士伺服电动机。伺服器与伺服电动机如图 9-3 所示。



图 9-3 伺服器与伺服电动机

9.4 系统的软件设计

9.4.1 系统整体结构

如图 9-4 所示,可以把整个基于 RTAI + Linux 的 PET 实时控制系统划分成硬件平台和软件平台两大部分:硬件平台主要由 PC104 +、PCL6045B 运动控制卡、PCMUI096 I/O 卡以及控制扫描床的交流伺服电动机和控制棒源、隔板、摆动装置的步进电动机组成(虚线部分)。软件平台主要由 RTAI, Linux kernel 和 Linux 应用程序几部分组成(实线部分),下面把各个部分之间的关系归结如下^[9]:

关系 1 表示实时任务如何与 RTAI 提供的实时任务接口相互作用。实时任务都是以 Linux 内核模块方式实现的,要实现一个实时任务,在模块初始化的时候要调用 RTAI 的任务创建函数初始化实时任务相关的数据和环境,指定定时器的运行模式(单触发模式或周期模式)并初始化定时器,然后开始执行任务;当没有加载任何 RTAI 的实时任务模块的时候,RTAI 的任务调度和时钟中断都没有启动。关系 2 表示实时任务通过 RTAI 提供的管道(FIFO)和共享内存与 Linux 用户空间中的进程进行通信,通过这种方式,实时任务获取的实时数据就可以传递到用户空间让非实时进程对数据进行后续的处理。关系 3

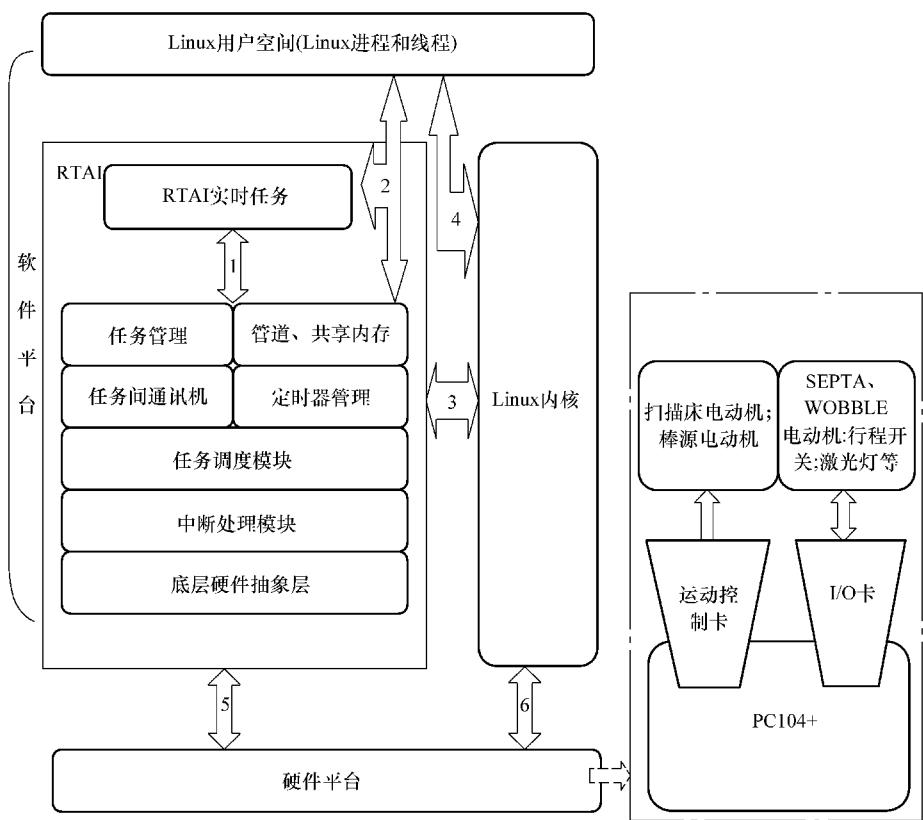


图 9-4 系统整体结构

表示 RTAI 本身的实现需要用到 Linux 内核提供的某些功能；例如，RTAI 本身（包括各种提供给实时任务的服务模块）是以 Linux 内核驱动模块的形式存在的，这就需要用到 Linux 内核的动态内核模块加载功能；另外，RTAI 目前的内存管理模块在初始化时是使用 Linux 的内存分配接口分配足够的内存。关系 4 表示传统的 Linux 内核和 Linux 用户程序的关系。关系 5 表示 RTAI 和底层硬件之间的交互，当外部事件触发了实时任务之后，实时任务在处理的过程中一般要对外部设备执行某些操作，例如控制采集卡进行数据采集、控制电动机等。关系 6 表示 Linux 和硬件平台之间的关系。PET 控制系统执行过程如图 9-5 所示。

在用户空间：

① 接收/反馈进程主要功能是通过远程过程调用的 socket 接口和主机 sun 相连接，从 sun 接命令信息并把命令信息传递到 commFIFO，同时发起另一个线程 statusTask 完成状态和检测信息的反馈。

② 显示进程是在用户空间运行的另一个进程，该进程通过串口 485 把错误信息、床位置等送到显示屏显示。

在内核空间：

① comTask 任务主要功能完成各实时任务的初始化工作，从 commFIFO 接命令信息并按

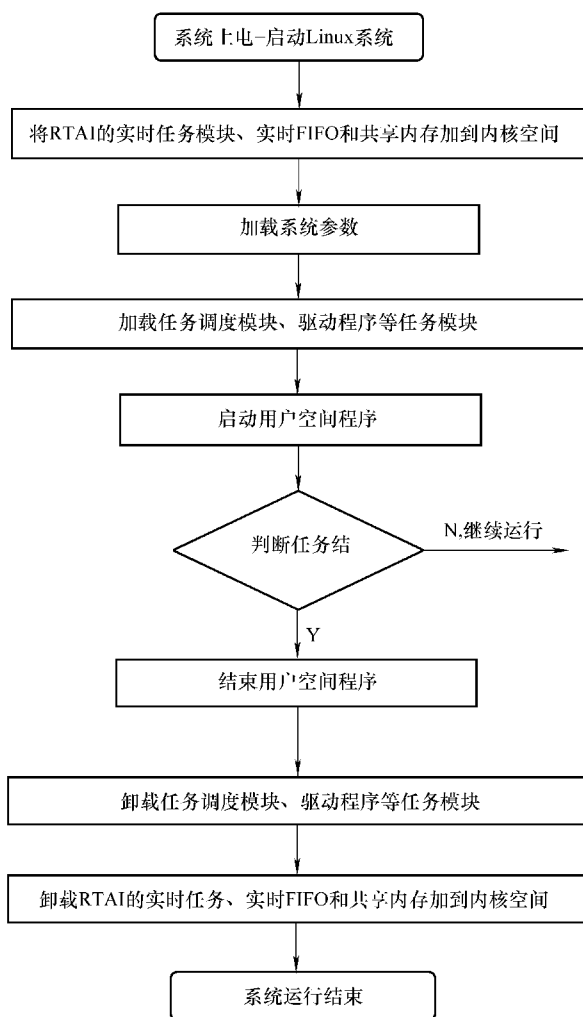


图 9-5 系统执行过程

要求发送到其他邮箱。

② bedTask 任务主要功能是从 bedMBX 邮箱接收床控命令并完成床的相应运动。

③ rapTask 任务主要功能从 rapMBX 邮箱接棒源、隔板 (Septa) 以及摆动装置控制命令并完成相应运动。

④ errTask 任务的主要功能完成从 errMBX 邮箱接错误命令信息并传给用户空间的显示进程。

⑤ monTask 任务的主要功能是完成各种状态监控。

⑥ dioTask 任务的主要功能是完成控制面板和 I/O 卡信号的监控。

主机系统和用户空间的进程的通信主要是依靠远程过程调用 (RPC) 完成, 主要调用 sock 接口函数完成。内核空间和用户空间下进程的通信主要是依靠实时扩展部件 RTAI 提供的实时 FIFO 和共享内存来完成。内核空间的实时任务之间的通信依靠邮箱完成。

9.4.2 系统软件设计方法

上位机软件主要完成文件管理和控制参数设定等功能。由于下位机软件是本章分析的重点，按照前文的论述将用户空间模块（虚线左）和内核空间模块（虚线右）之间关系汇总如图9-6所示，其中在用户空间运行两个进程，在内核空间完成六个实时任务。多任务调度算法采用了基于优先级的 FIFO 算法。

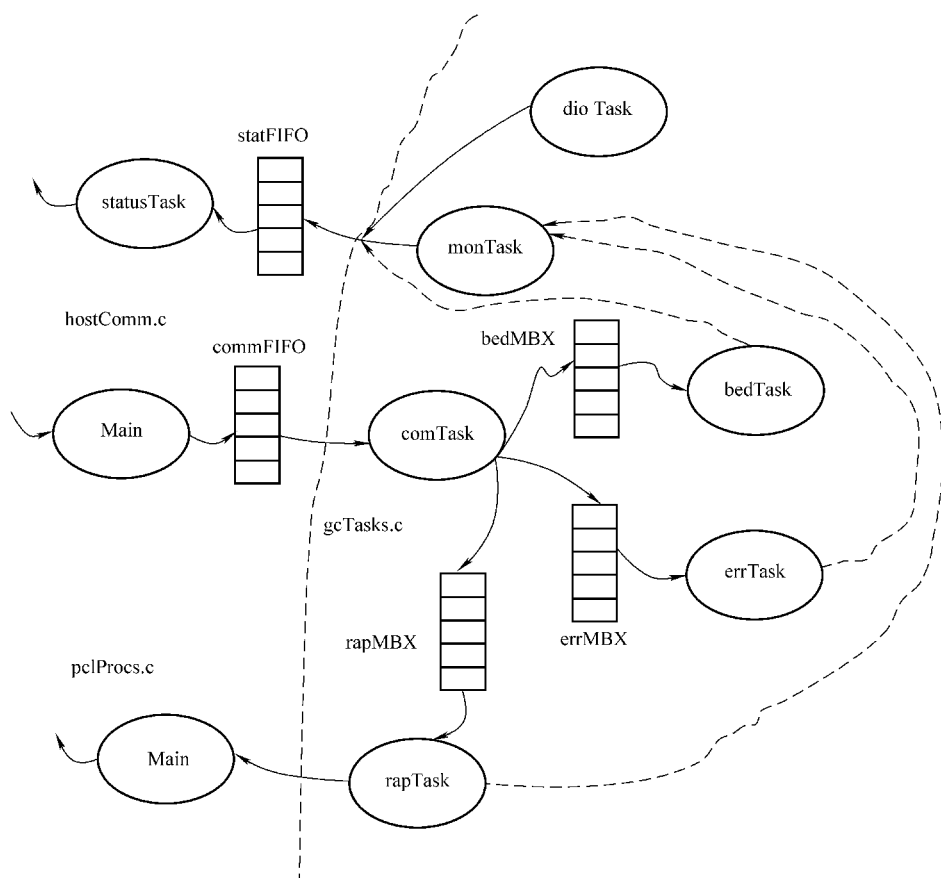


图 9-6 系统软件任务结构模块

9.5 信息传递过程实例

Linux 中的进程运行在用户空间和内核空间两种模式下：一般在 Linux 用户空间运行的是非实时任务（也就是进程和线程），而在 Linux 内核空间运行的是 RTAI 的实时任务。在实时 Linux 下为了保证 PET 控制系统的实时任务能够即时响应，所有实时任务都必须放在内核空间运行，这些任务包括：rapTask、bedTask、comTask、errTask、monTask、dioTask。而非实时任务则放在用户空间运行，它们在任何时候都不会打断实时任务的运行，非实时任务

包括接收/反馈进程和显示进程。而主机系统包括：文件管理、状态显示，协议运行，参数设置等组成，PET 控制系统软件结构如图 9-7 所示。

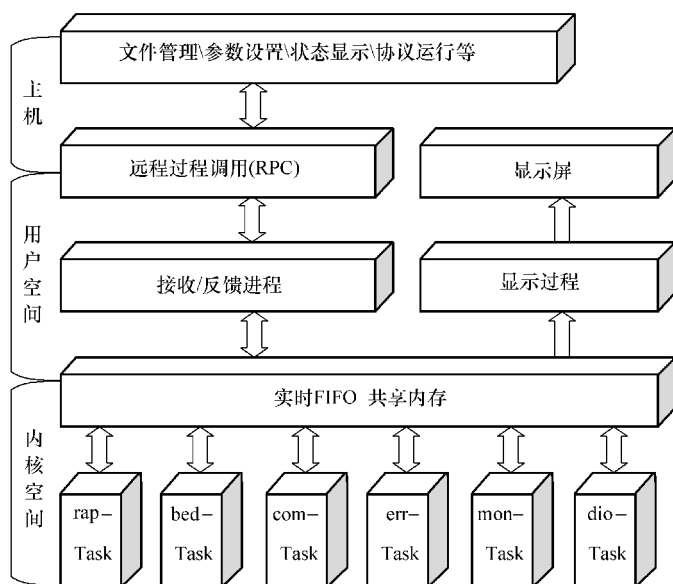


图 9-7 PET 控制系统软件结构

9.5.1 网络传递信息

在主机系统和 PC104 + 的用户空间程序之间通过网络完成信息的传递，下面的表 9-1、图 9-8 和后面的部分程序是完成该部分信息传递的实例。

表 9-1 linux 下常用的 socket 处理函数

函数	说明
socket ()	socket 创建套接字
bind ()	bind 将套接字与本地地址绑定
listen ()	listen 开始侦听
connect ()	connect 连接远端的套接字
accept ()	accept 接受来自网络的连接请求并新建通信套接字
send ()	send 发送数据
recv ()	recv 接收数据

在网络通信中，首先由服务器端创建一个侦听用套接字 (socket)，然后与服务器的本地地址绑定，接着进入侦听模式。客户端开始时，同样也需要创建一个套接字，不同的是这个套接字只用于通信，然后连接服务器端。对应到网络通信就是：服务器端用于侦听的套接字接受 (accept) 这个来自客户端的连接，然后建立新的通信用套接字，并利用新建的通信


```

    fprintf (stderr, "Couldn't bind to port %d. \n", LISTENPORT);
    close (listenFd); //关闭 Socket
    exit (1);
}

fprintf (stderr, "\nhostComm: listening on port %d. \n", LISTENPORT);
if (listen (listenFd, 1) == -1) {
    fprintf (stderr, " Listen failed \n");
    close (listenFd);
    exit (1);
}

while (1)
{
    char sCmd [100];
    int nBytes;
    // Accept the connection:
    conn.socFd = accept (listenFd, (struct sockaddr *) NULL, NULL);
    fprintf (stderr, " \nhostComm: accepted conn. socFd = %d. \n",
        conn.socFd);
    pthread_ attr_ init (&tattr);
    //pthread_ attr_ setscope (&tattr, PTHREAD_ SCOPE_ PROCESS);
    if (pthread_ create (&tSta, &tattr,
        (void * ( *) (void *)) statusTask, (void *) &conn) == 0)
    {
        printf (" hostComm: statusTask thread created \n");
    }
    else
    {
        printf (" hostComm: statusTask thread NOT created \n");
    }

    while (1)
    {
        bzero (sCmd, sizeof (sCmd));

        if ((nBytes = recv (conn.socFd, sCmd, PACKET_ LEN + 2, 0)) != PACKET_ LEN + 2)
        {

```

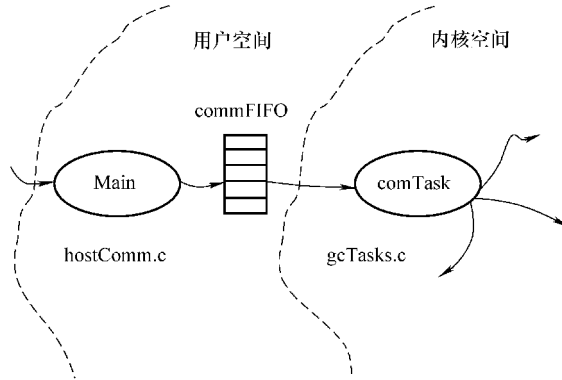



图 9-9 基于信号量的信息传递

```

while (1)
{
    rt_ printk ("comTask: waiting for commSem. . . \n");
    rt_ sem_ wait (&commSem);
    .....
}

```

```

/ * * * * *

```

```

int init_ module (void)

```

所在文件: gcTasks. c

任务说明: 完成扫描床的实时控制

```

* * * * *

```

```

int init_ module (void)

```

```

{

```

```

    .....

```

```

    rtf_ create (commFIFO, 8000);

```

```

    rtf_ create_ handler (commFIFO, commHandler);

```

```

    rt_ sem_ init (&commSem, 0);

```

```

    .....

```

```

}

```

```

/ * * * * *

```

```

static int commHandler (unsigned int fifo)

```

所在文件: gcTasks. c

任务说明: 完成扫描床的实时控制

```

* * * * *

```



```

static void comTask (int t)
{
    .....
    while (1)
    {
        .....
        /* 向 bedMBX 邮箱发送信息，任务将阻塞到所有的消息都传送到邮箱后 */
        rt_mbx_send (&bedMBX, sCmd, PACKET_LEN);
        rt_printk (" bedMBX: %c \n", sCmd [0]);
        break;
        .....
    }
}

/* * * * * *
void bedTask ()
所在文件: bedTask.c
任务说明: 完成扫描床的实时控制
* * * * *
void bedTask ()
{
    .....
    while (1)
    {
        /* 从 bedMBX 邮箱接收信息，任务将阻塞直到所有的消息都到达 */
        rt_mbx_receive (&bedMBX, sCmd, PACKET_LEN);
        rt_printk (" bedTask: mbx_rec'd sCmd %s\n", sprintFixedBuf(sCmd));
        switch (sCmd [0])
        {
            case 'H': // tmp
                rt_printk (" bedTask: bedHome \n");
                bedHome ();
                break;
            .....
        }
    }
}

```

9.6 不确定问题的校正与补偿处理

我们知道，非线性不确定摩擦是一种复杂的、非线性的、具有不确定性的自然现象。摩擦使系统响应出现爬行、振荡、打滑或稳态误差，以及其他不可预料现象。在前面章节里从理论上提出了各种补偿方法，以消除摩擦的不利影响。实际产品也考虑了这些不确定现象，并在产品中嵌入了一些补偿措施，如本章节用到的 PCL6045B 运动控制卡就有此功能^[10]。

9.6.1 间隙补偿与打滑校正

PCL6045B 运动控制卡具有间隙补偿与打滑校正功能，这些功能输出一定数量的指令脉冲，其速度由校正速度寄存器 RFA 中的速度设定值决定。间隙补偿功能在每次改变操作方向时执行，而打滑校正功能则在指令操作之前执行，与运动方向无关。校正量和校正量在 RENV6 寄存器中指定。根据 RENV3 寄存器的设置，校正操作中也可以让计数器（C1 - C4）进行计数操作。表 9-4 是间隙补偿与打滑校正的参数设置。需要注意的是：

1) 在圆弧插补运动过程中各轴运动方向会发生改变。间隙补偿与打滑校正不能在定时器和圆弧插补中使用，但可以再圆弧插补启动时使用。如果想在圆弧插补中使用间隙补偿需要将圆弧插补分段，在各段开始时使用间隙补偿。

表 9-4 间隙补偿与打滑校正的参数设置

输入插补值（RENV6 中的 BR0 ~ BR11） 间隙补偿或打滑校正量：0 ~ 4095	[RENV6]（写） 158 <table><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>n</td><td>n</td><td>n</td><td>n</td></tr></table> 70 <table><tr><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td></tr></table>	-	-	-	-	n	n	n	n	n	n	n	n	n	n	n	n
-	-	-	-	n	n	n	n										
n	n	n	n	n	n	n	n										
指定前向运动时间（RENV6 中的 ADJ0 ~ ADJ1） 00：关闭校正功能 01：间隙补偿 10：打滑校正	[RENV6]（写） 158 <table><tr><td>-</td><td>-</td><td>-</td><td>n</td><td>n</td><td>-</td><td>-</td><td>-</td></tr></table>	-	-	-	n	n	-	-	-								
-	-	-	n	n	-	-	-										
反冲/打滑校正的功能（RENV3 中的 CU1B ~ CU4B） CU1B（第 16 位）=1：允许 COUNTER1（指令位置计数器） CU2B（第 17 位）=1：允许 COUNTER2（机械位置计数器） CU3B（第 18 位）=1：允许 COUNTER3（偏差计数器） CU4B（第 19 位）=1：允许 COUNTER4（通用计数器）	[RENV3]（写） 3124 <table><tr><td>-</td><td>-</td><td>-</td><td>0</td><td>n</td><td>n</td><td>n</td><td>n</td></tr></table>	-	-	-	0	n	n	n	n								
-	-	-	0	n	n	n	n										

2) 在直线插补 1 和圆弧插补中, 必须为各轴指定速度数据。然而, 在补偿操作中 PRMG 寄存器的设置被用作速度数据。如果想在插补操作中使用间隙补偿, 应设置 PRMG 寄存器中的值与各轴速度数据相同, 也要为各轴设置 RFA 寄存器。

3) PCL 和 RFA 寄存器中提供一个匀速度间隙补偿。替代状态寄存器检测值 SFU (加速阶段)、SFD (减速阶段) 和 SFC (匀速阶段) 将变为 0。

间隙补偿软件实现如下的参考代码:

```
void Jcompensation ()
{
    P645_ vset (AXS_ AX, 100L, 1000L, 2000, 0, 0, 0, ' L ', 10);
    P645_ wreg (AXS_ AX, WRENB6, 0x0000100A);
    P645_ wreg (AXS_ AX, WRENB3, 0x000fF0000);
}
```

9.6.2 不确定振动的补偿处理

PCL6045B 运动控制卡具有减小停止时振动的功能。此功能的实现, 是在完成指令脉冲的操作后, 立即添加一个反向脉冲和一个正向脉冲。在 RENV7 寄存器中指定额外脉冲的输出时序。当反向定时 (RT) 和正向定时 (FT) 均不为零时, 即允许减振功能。设置 FT 和 RT 的单位为 32 倍的参考时钟频率。图 9-11 中的虚线表示减振功能添加的脉冲, 图中为正方向脉冲。表 9-5 是实现减振功能的参数设置, 其中 FT 和 RT 的最佳值将随机器和负载的不同而不同, 因此最好通过试验来获得这些值。

```
void Vcompensation ()
{
    P645_ wreg (AXS_ AX, WRENV7, 0x01520266);
}
```

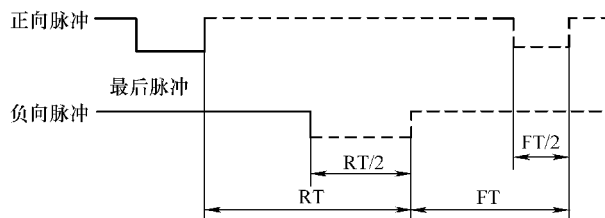
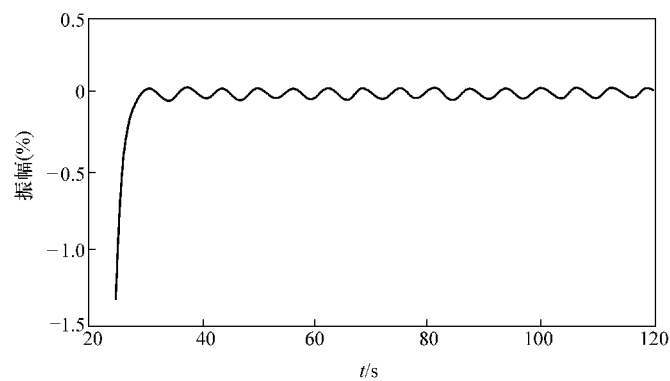


图 9-11 减振功能实现的时序

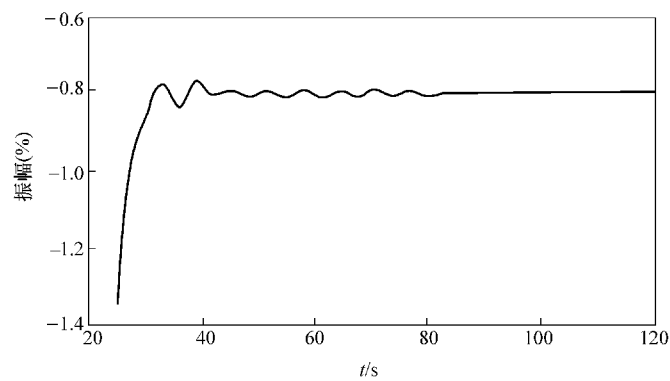
图 9-12 是使用减振功能时电动机运动的波形采样, 连接一个编码器到一个 5 相位步进电动机。给上/下计数器提供编码信号, 并且通过程序控制以一定的间隔读取计数器的值。然后计算运动量和检测电动机运动状态。通常, 在使用 S 形曲线加减速时停止过程中要比直线加减速时容易减振。

表 9-5 振动补偿的参数设置

指定反向运动时间（设定 RENV7 中的 RT0 ~ RT15） RT 范围：0 ~ 65535 单位为 32x 参考时钟频率（当 CLK = 19.6608MHz 时，大约为 1.6μs） 可设置范围：0 ~ 0.1s	[RENV7]（写）							
	15							8
	n	n	n	n	n	n	n	n
	7							0
	n	n	n	n	n	n	n	n
指定前向运动时间（设定 RENV7 中的 FT0 ~ FT15） FT 范围：0 ~ 65535 单位为 32x 参考时钟频率（当 CLK = 19.6608MHz 时，大约为 1.6μs） 可设置范围：0 ~ 0.1s	[RENV7]（写）							
	31							24
	n	n	n	n	n	n	n	n
	23							16
	n	n	n	n	n	n	n	n



a) 减振功能关闭时



b) 减振功能开启时

图 9-12 减振功能实现的效果

9.7 本章小结

本章给出了基于 RTLinux 为软件平台的 PET 控制系统的多任务实时系统设计与控制补偿实例。经过反复实验证明, PET 的扫描床、棒源、隔板以及摆动装置能够很好地完成既定控制任务, 运动控制精度达到了期望的性能指标, 并且系统运行稳定, 目前该系统已完成产品化目标, 在东软 PET 系统获得运行。

参考文献

- [1] Bernet G, Burns A. Weakly Hard Real-Time Systems. IEEE Trans Computers, 2001, 50 (4): 308-321.
- [2] 邹勇, 王青, 李明树. Linux 内核的实时支持的研究与实现 [J]. 计算机研究与发展, 2002, 39 (4): 466-472.
- [3] 朱达宇, 李彦等. 基于 RTLinux 的全软件数控系统 [J]. 计算机集成制造系统, 2004, 10 (12): 1571-1575.
- [4] 郭锐, 赵万生, 李论, 李志勇, 郑君民. 基于 Linux 的微细电火花加工数控系统的研究 [J]. 计算机集成制造系统, 2007, 13 (2): 370-375.
- [5] 李志勇, 赵万生. 基于 Linux 内核的数控系统关键技术研究 [J]. 计算机集成制造系统, 2004, 10 (5): 569-573.
- [6] 孙强. PET/CT 运行成本及效率研究 [J]. 中国卫生经济, 2005, 24 (1): 37-41.
- [7] David W T, Jonathan P J. PET/CT today and tomorrow [J]. Journal of Nuclear and Medicine, 2004, 45 (1): 1-14.
- [8] 王永富, 赵宏, 刘积仁. 国产 PET 实时控制与成像系统的研究 [D]. 沈阳: 东北大学, 2008.
- [9] 王永富, 李甲递, 赵宏, 刘积仁. 基于实时 Linux 的 PET 控制软件设计与实现 [J]. 东北大学学报, 2008, 29 (4): 497-500.
- [10] 叶佩青, 张辉. PCL6045B 运动控制与数控应用 [M]. 北京: 清华大学出版社, 2007.



不确定机械问题的 建模与控制补偿

MODELING AND CONTROL COMPENSATION OF
UNCERTAIN MECHANICAL PROBLEMS

地址:北京市百万庄大街22号
邮政编码:100037

电话服务

社服务中心:010-88361066

销售一部:010-68326294

销售二部:010-88379649

读者购书热线:010-88379203

网络服务

教材网: <http://www.cmpedu.com>

机工官网: <http://www.cmpbook.com>

机工微博: <http://weibo.com/cmp1952>

封面无防伪标均为盗版

扫一扫更多汽车技术
精品图书供你选



ISBN 978-7-111-43243-2

策划编辑◎赵海青 / 封面设计◎张静

ISBN 978-7-111-43243-2



9 787111 432432 >

定价: 40.00元