

自动化

专业本科系列教材

Mohu Kongzhi Jishu

模糊控制技术

0101001000100001

韩峻峰 李玉惠 等编著

重庆大学出版社

计算机软件技术基础

微机原理与接口技术

电路原理

模拟电子技术基础

数字电子技术基础

电机与拖动基础

电力电子技术

电力拖动控制

自动控制原理

现代控制理论

信号与系统分析基础

检测技术

过程控制系统与装置

可编程控制器及其系统

电子综合设计与制作实践

控制系统计算机辅助设计

单片机原理及应用 (8096系列)

单片机原理及应用 (51系列)

C语言程序设计

现代低压电器及其控制技术

智能仪器仪表

最优控制

系统集成技术

模糊控制技术

微机控制及接口技术

自动化专业英语

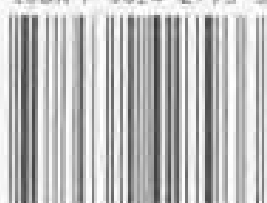
系统辨识与自适应控制

智能控制概论

计算机监控原理与技术

网络控制系统

ISBN 7-5624-2793-3



9 787562 427933 >

ISBN 7-5624-2793-3/TP·411

定价: 12.00元

模糊控制技术

韩峻峰 李玉惠 等编著

重庆大学出版社

内 容 提 要

本书是一本系统介绍最新模糊控制技术的教材。全书共分为7章,首先介绍了模糊控制的起源、特点、研究与应用的现状及其发展趋势;然后从工程应用的角度介绍了模糊逻辑与模糊推理的基础内容,在此基础上论述了模糊控制技术相关基础,并详细讨论了模糊控制器的设计;对模糊控制的发展趋势——与神经网络的结合,本书也进行了介绍;从实用化的角度,对模糊控制系统的开发方法进行了介绍;最后,以典型的例子从不同的角度对模糊控制的应用进行了介绍。

本书选材恰当、体系合理、逻辑性强、通俗易懂,侧重介绍应用,适合作为高等学校自动化和电气信息类相关专业本科教材,也可供自学人员和工程技术人员参考。

图书在版编目(CIP)数据

模糊控制技术/韩峻峰等编著. —重庆:重庆大学出版社,2003.5

(自动化专业本科系列教材)

ISBN 7-5624-2793-3

I. 模... II. 韩... III. 模糊控制—高等学校—教材 IV. TP13

中国版本图书馆 CIP 数据核字(2003)第 030626 号

模糊控制技术

韩峻峰 李玉惠 等编著

责任编辑:曾显跃 版式设计:曾显跃

责任校对:蓝安梅 责任印制:秦 梅

*

重庆大学出版社出版发行

出版人:张鸽盛

社址:重庆市沙坪坝正街174号重庆大学(A区)内

邮编:400044

电话:(023) 65102378 65105781

传真:(023) 65103686 65105565

网址:<http://www.cqup.com.cn>

邮箱:fxk@cqup.com.cn (市场营销部)

全国新华书店经销

重庆铜梁正兴印务有限公司印刷

*

开本:787×1092 1/16 印张:8 字数:199千

2003年5月第1版 2003年5月第1次印刷

印数:1—4 000

ISBN 7-5624-2793-3/TP·411 定价:12.00元

本书如有印刷、装订等质量问题,本社负责调换

版权所有 翻印必究

前言

模糊控制是近代控制理论中建立在模糊集合论基础上的一种基于语言规则与模糊推理的控制理论,它是智能控制的一个重要分支。

常规控制的基本特点是控制器的设计都要建立在被控对象的精确数学模型基础上,但是,在许多情况下被控对象(或生产过程)的精确数学模型很难建立。另一方面,人们注意到,对于很多复杂的、多因素影响的生产过程,即使不知道该过程的数学模型,有经验的操作人员也能够根据长期的实践观察和操作经验进行有效的控制,而采用传统的自动控制方法效果并不理想。人的经验参与控制过程的成功,激发了人们对控制原理的进一步深入研究。这种原理以能包含人类思维的控制方案为基础,而且反映了人类经验的控制过程知识,以及可以达到控制目的,能够利用某种形式表达出来,同时还很容易被实现。这样的控制系统既避免了那种精密、反复、有错误倾向的模型建造过程,又避免了精密地估计模型方程中各种参数的过程。在多变量、非线性、时变的大系统中,人们可以采用简单灵活的控制方式,于是就产生一个问题,能否把人的操作经验总结为若干条控制规则,并设计一个装置去执行这些规则,从而对系统进行有效的控制呢?模糊控制理论与技术由此应运而生,这就是模糊控制产生的背景。

模糊控制理论和方法的提出,归功于美国加利福尼亚大学的自动控制教授 L. A. Zadeh 于 1965 年首次提出的“模糊集合”的概念,使得有人的经验参与的控制过程成为实际可能;1973 年, L. A. Zadeh 又进一步研究了模糊语言处理,给出了模糊推理的理论基础;自从 1974 年 Mamdani 制造出用于锅炉和蒸汽机的第一个模糊控制器以来,模糊控制理论和应用技术的发展历史还不到 30 年。虽然其发展历史很短,但发展速度之快、成果之多和世人重视的程度却是少有的,特别是 1987

年在日本,基于模糊控制的仙台市地铁开通以后,各种家电的模糊产品相继研制成功并进入市场,如洗衣机、照相机、摄像机、复印机、吸尘器、电冰箱、微波炉、电饭锅、空调器、电视机、淋浴器等。这些家电产品在节约资源、方便使用以及使用效果方面更富有“人情味”,更符合人的实际生活。同时,各种各样的模糊控制系统也被研制成功。例如,各种熔炉、电气炉,水泥生成炉的控制系统、核能发电供水系统、汽车控制系统、电梯升降机控制系统、机器人控制系统以及活跃于航空、宇航、通信领域里的专家系统。这些模糊控制系统的应用取得了明显的效益,并且在日本、美国、西欧、东南亚地区以及我国引起了普遍重视。

为了适应 21 世纪科学技术、经济、社会发展趋势的要求,把体现当代控制理论学科前沿发展的新成果反映到教学之中,结合高等工科院校人才培养的特点,我们特尝试编写了此书。

全书共分 7 章:第 1 章为绪论,主要阐述了模糊控制的起源及其特点、模糊控制研究、应用现状与展望;第 2 章为模糊逻辑与模糊推理,重点介绍了模糊集合及其隶属函数、模糊矩阵与模糊关系、模糊语言与模糊逻辑、模糊推理;第 3 章为模糊控制技术基础,阐明了模糊控制基本原理、模糊化方法、解模糊方法、模糊控制规则及控制算法;第 4 章为模糊控制器设计,主要介绍了模糊控制器的基本结构、模糊控制器的设计;第 5 章为神经网络在控制系统中的应用,阐述了神经网络、模糊神经网络、基于神经网络的模糊控制器;第 6 章为模糊控制系统的开发,详细介绍了开发方法——恒温箱温度的模糊控制、数字单片机实现模糊控制、模糊单片机实现模糊控制;第 7 章为模糊控制应用实例,主要介绍了模糊控制全自动洗衣机、智能手机充电器及地铁机车模糊控制器三个应用案例。

本书结合作者的研究工作,力求反映模糊控制领域的最新成果,同时考虑到模糊控制的前沿性特征,在尽可能全面系统地阐述模糊控制技术的基础上,试图尽量弱化烦琐的数学推理,而侧重介绍应用,适合作为自动化和电气信息类相关专业本科教材,也可供自学人员和工程技术人员参考。

本书由韩峻峰、李玉惠担任主编。其中,第 1、3、4 章由韩峻峰编写,第 2、5 章由李玉惠编写,第 6 章由聂诗良编写,第 7 章由苏睿编写。全书由韩峻峰、苏睿统稿。

本书在撰写过程中曾得到一些同事和朋友的支持与帮助,

并参考了较多的著作与论文,在此,谨对所有对该书编写作出贡献的人士表示衷心的感谢。由于编者水平有限,同时考虑到模糊控制技术的前沿性特征,因而书中可能出现疏漏或不妥之处,敬请广大专家学者及读者批评指正。

编 者

2003 年 1 月

目 录

第1章 绪论	1
1.1 模糊控制的起源及其特点	1
1.2 模糊控制研究与应用现状	4
1.3 模糊控制展望	6
第2章 模糊逻辑与模糊推理	8
2.1 模糊集合及其隶属函数	8
2.1.1 模糊集合及其表示	8
2.1.2 模糊集合的基本运算及其法则	10
2.1.3 模糊集合与普通集合的关系	10
2.1.4 模糊集合的隶属函数	11
2.2 模糊矩阵与模糊关系	13
2.2.1 模糊矩阵	13
2.2.2 模糊关系	14
2.2.3 模糊映射	16
2.3 模糊语言与模糊逻辑	18
2.3.1 模糊语言	18
2.3.2 模糊逻辑	21
2.4 模糊推理	23
2.4.1 模糊推理方法	23
2.4.2 模糊条件推理	24
第3章 模糊控制技术基础	29
3.1 模糊控制基本原理	29
3.1.1 模糊控制的引入及实现	29
3.1.2 模糊控制过程	30
3.2 模糊化方法(Fuzzification)	33
3.3 解模糊方法(Defuzzification)	35
3.3.1 重心法	35
3.3.2 最大隶属度法	36
3.3.3 系数加权平均法	36
3.3.4 隶属度限幅元素平均法	36
3.3.5 中位数法	36
3.4 模糊控制规则及控制算法	37

3.4.1	模糊控制规则的表示及特性	37
3.4.2	模糊控制规则的生成	39
3.4.3	模糊控制规则的优化	42
3.4.4	模糊控制算法	46
第4章	模糊控制器设计	49
4.1	模糊控制器的基本结构及主要类型	49
4.1.1	模糊控制器的基本结构	49
4.1.2	常用的几种模糊控制器	52
4.1.3	模糊控制器的设计	54
4.2	模糊控制器的结构设计	55
4.2.1	模糊控制器的结构设计	55
4.2.2	一般设计方法	56
第5章	神经网络在模糊控制系统中的应用	66
5.1	神经网络	66
5.1.1	人工神经网络的起源和发展	66
5.1.2	神经元和神经网络	67
5.1.3	人工神经网络的特点及类型	70
5.1.4	典型的人工神经网络	71
5.2	模糊神经网络	74
5.2.1	神经网络与模糊逻辑	74
5.2.2	模糊神经网络	75
5.3	基于神经网络的模糊控制器	79
5.3.1	复合型神经模糊控制器	79
5.3.2	融合型神经模糊控制器	82
5.3.3	模糊神经网络在倒立摆控制中的应用	82
第6章	模糊控制系统的开发	84
6.1	开发方法——恒温箱温度的模糊控制	84
6.1.1	控制对象简述	84
6.1.2	模糊控制器的结构设计	84
6.1.3	模糊变量的论域及其隶属函数	85
6.1.4	根据实际操作经验给出以下控制规则	85
6.1.5	控制输出	87
6.2	数字单片机实现模糊控制	88
6.3	模糊单片机实现模糊控制	90
6.3.1	NLX230 的引脚及功能	90
6.3.2	NLX230 的工作原理和功能特点	91
6.3.3	NLX230 的内部寄存器及功能	92
6.3.4	NLX230 的操作及接口技术	93

6.3.5 NLX230 的应用实例	94
第 7 章 模糊控制应用实例	99
7.1 模糊控制全自动洗衣机	99
7.1.1 洗衣条件	100
7.1.2 模糊控制洗衣机结构	100
7.1.3 控制电路设计	104
7.1.4 模糊控制实现方法	106
7.2 智能手机充电器	108
7.2.1 智能充电原理	108
7.2.2 手机充电器模糊控制的系统原理框图	109
7.2.3 充电控制原理	109
7.2.4 控制软件	112
7.3 地铁机车模糊控制器	113
7.3.1 对评价指标的定义	113
7.3.2 对机车的运动特性模拟实验	114
7.3.3 模糊控制规则的制定	114
7.3.4 模糊控制的实现	116
参考文献	117

第 1 章

绪 论

1.1 模糊控制的起源及其特点

模糊控制是近代控制理论中建立在模糊集合论基础上的一种基于语言规则与模糊推理的控制理论,它是智能控制的一个重要分支。

20 世纪中叶以来,在科学技术与工业生产的发展过程中,自动控制理论与技术的发展发挥了巨大的作用,并取得了令人满意的控制效果,是现代高新技术的重要手段之一。

常规控制的基本特点是:对于控制器的设计,都要建立在被控对象的精确数学模型基础上,但是,在许多情况下,被控对象(或生产过程)的精确数学模型很难建立。例如,有些对象难以用一般的物理和化学方面的规律来描述;有的影响因素很多,而且相互之间又有交叉耦合,使其模型十分复杂。在这些模型方程中,含有众多的参数需要估计,求解这些参数却往往缺少足够的信息量与信息特征;简化后的数学模型不能准确地说明原来的系统,以至于没有实用价值;还有一些生产过程缺乏适当的测试手段,或者测试装置不能进入被测试区域,致使无法建立过程的数学模型。而且,随着科学技术的迅猛发展,目前研究的控制系统更多地涉及多变量、非线性、时变的大系统,建立数学模型是非常困难的,或者是根本不可能的,系统的复杂性与控制技术的精确性形成了尖锐的矛盾。于是,传统的控制理论和技术面临着新的控制要求的挑战。正如 L. A. Zadeh 指出的:当系统日益复杂,人们对它的精密而有意义的描述的能力将相应地降低,以至达到精密与有意义几乎相互排斥的地步。要想精确地描述复杂现象和系统的任何现实的物理状态,事实上是办不到的。虽然常规自适应控制技术可以解决一些问题,但范围依然有限。上述情况迫使人们在控制系统的精确性与有意义之间寻求某种平衡和折中,而使问题的描述具有实际意义。

另一方面,人们注意到,对于很多复杂的、多因素影响的生产过程,即使不知道该过程的数学模型,有经验的操作人员也能够根据长期的实践观察和操作经验进行有效的控制,而采用传统的自动控制方法效果并不理想。人的经验参与控制过程的成功,激发了人们对控制原理的深入研究。这种原理是以能包含人类思维的控制方案为基础,而且反映人类经验的控制过程

的知识,以及可以达到的控制目的能够利用某种形式表达出来,同时还很容易被实现。这样的控制系统既避免了那种精密、反复、有错误倾向的模型建造过程,又避免了精密地估计模型方程中各种参数的过程。在多变量、非线性、时变的大系统中,人们可以采用简单灵活的控制方式,于是就产生了一个问题:能否把人的操作经验总结为若干条控制规则,并设计一个装置去执行这些规则,从而对系统进行有效的控制?模糊控制理论与技术由此应运而生,这就是模糊控制产生的背景。

模糊控制最重要的特征是反映人们的经验以及人们的常识推理规则,而这些经验与常识推理规则是通过语言来表达的。比如说“温度太高,温度上升的速度也很快,则大幅度降温”。对于用语言表达的这种经验,必须给出一种描述的方式,而且这种经验是多种多样的。比如,还可以有经验规则“温度稍低,升温的速度很快,则稍微降温控制”。模糊控制规则综合考虑众多的控制策略,是一种常识推理规则。

当然,由于对系统缺乏了解,一开始控制效果可能并不好,但经过若干次探索后终归能实现预期的理想控制。这说明传统控制理论必须向前发展,而人工智能、模糊控制就是在这种背景下产生并发展起来的。也就是说,控制问题在经历了人工控制、经典控制理论和现代控制理论阶段之后,由于它们面临着一系列无法解决的问题,又要重新研究人工控制行为的特点,以便从人工控制中得到新的启发。

经典控制理论主要解决线性系统的控制问题,现代控制理论可以解决多输入与多输出的问题,系统既可以是线性的、定常的,也可以是非线性的、时变的;而对于那些数学方程很难提出但人们都有丰富控制经验的实际课题,模糊控制技术发挥了奇特的优势。特别是近几年来,模糊控制技术取得了迅速发展。可以预料,在传统控制的难题中,有一批难题可以应用模糊控制技术或者用传统控制技术与模糊控制技术相结合的方法来加以解决。

与传统的 PID 控制比较,模糊控制有其明显的优越性。由于模糊控制实质上是用计算机去执行操作人员的控制策略,因而可以避开对象复杂的数学模型,力图对人们关于某个控制问题的成功与失败的经验进行加工,总结出知识,从中提炼出控制规则,实现复杂系统的控制。1974—1979 年间,英国、联邦德国、日本等国曾先后做过对蒸汽机锅炉、炼钢炉及汽轮发电机组等模糊控制的计算机仿真,把模糊控制算法与经典的传递函数算法的计算机仿真进行比较,其中英国 Mamdani 对蒸汽发动机和锅炉的控制部分仿真的对比结果如图 1.1 所示。

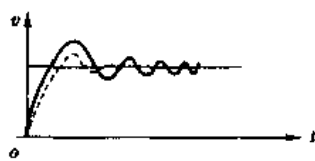


图 1.1 仿真对比

图 1.1 中的实线是经典的 PID 控制在阶跃干扰下的响应曲线,虚线是模糊控制的响应曲线。实验结果表明:在两种系统均调到最佳状态下,当工作条件大幅度变化时,经典的 PID 控制是难以调整的,超调量大并带有振荡;与此相反,模糊控制对过程参数改变不灵敏,在所有工作点上都能做到较稳定的控制,这说明模糊控制本质上是非线性的,并且对于对象的参数变化适应性强,即所谓的“鲁棒性”(Robust)较好。以上仿真结果表明,模糊控制比传统的经典控制稳定,而且效果好。

模糊控制理论和方法的提出,归功于美国加利福尼亚大学的自动控制教授 L. A. Zadeh 于 1965 年首次提出的“模糊集合”的概念,使得有人的经验参与的控制过程成为实际可能。1973 年, L. A. Zadeh 又进一步研究了模糊语言处理,给出了模糊推理的理论基础。自从 1974 年 Mamdani 制造出用于锅炉和蒸汽机的第一个模糊控制器以来,模糊控制理论和应用技术的发展历史还不到 30 年,虽然其发展历史很短,但发展速度之快、成果之多和世人重视之程度却是

少有的,特别是1987年在日本,基于模糊控制的仙台地铁开通以后,各种家电的模糊产品相继研制成功并进入市场,如洗衣机、照相机、摄像机、复印机、吸尘器、电冰箱、微波炉、电饭锅、空调器、电视机、淋浴器等,这些家电产品在节约资源、方便使用以及使用效果方面更富有“人情味”,更符合人的实际生活。同时,各种各样的模糊控制系统也被研制成功。例如,各种熔炉、电气炉、水泥生成炉的控制系统、核能发电供水系统、汽车控制系统、电梯升降机控制系统、机器人控制系统,以及活跃于航空、宇宙、通信领域里的专家系统。这些模糊控制系统的应用取得了明显的效益,并且受到了人们普遍的重视。

模糊理论与应用的研究以及模糊产品的开发像一股强劲的风浪席卷世界各地。1989年,模糊理论的创始人 L. A. Zadeh 指出:模糊理论是对“彻底排除不明确事物只以明确事物为对象”的科学界传统所做的挑战。这种理论对于如何处理与对待不明确事物,所依据的思路与过去的科学实质上完全不同。他认为模糊理论今后将在两个领域取得较大进展:一是熟练技术者替代系统,这种系统将人无意识进行的操作由机器替代,如日本仙台市地铁的自动驾驶系统;二是替代专家的专家系统。为使专家头脑中所进行的思考与决策能实现自动化,模糊理论将起重要的作用。当然,模糊理论并不能解决所有可能性问题,但是,只要不回避现实中的不确定事物,并加以认真对待,就有可能大大地提高在不确定(模糊)环境中进行智慧思考与决策的人及机器的能力。

L. A. Zadeh 教授提出的模糊集合论,其核心是对复杂的系统或过程建立一种语言分析的数学模式,使自然语言能直接转化为计算机所能接受的算法语言。模糊集合理论的诞生为处理客观世界中存在的一类模糊性问题提供了有力的工具,同时,也适应了自适应科学发展的迫切需要。

以模仿人类人工控制特点而提出的模糊控制虽然带有一定的主观性和模糊性,但往往是简单易行,而且是行之有效的。模糊控制的任务正是要用计算机来模拟这种人的思维和决策方式,对这些复杂的生产过程进行控制和操作。

从以上背景可以看出,模糊控制有以下的特点:

①模糊工程的计算方法虽然是运用模糊集理论进行的模糊算法,但最后得到的控制规律是确定性的、定量的条件语句。

②不需要根据机理与分析建立被控对象的数学模型,对于某些系统,要建立数学模型是很困难的,甚至是不可能的。

③与传统的控制方法相比,模糊控制系统依赖于行为规则库,由于是用自然语言表达的规则,更接近于人的思维方法和推理习惯,因此,便于现场操作人员的理解和使用,便于人机对话,以得到更有效的控制规律。

④模糊控制与计算机密切相关。从控制角度看,它实际上是一个由很多条件语句组成的软件控制器。目前,模糊控制还是应用二值逻辑的计算机来实现,模糊规律经过运算,最后还是进行确定性的控制。模糊推理硬件的研制与模糊计算机的开发,使得计算机将像人脑那样随心所欲地处理模棱两可的信息,协助人们决策和进行信息处理。

1.2 模糊控制研究与应用现状

模糊集合和模糊控制的概念是由美国加州大学著名教授 L. A. Zadeh 在其《Fuzzy Sets》、《Fuzzy Algorithm》和《A Relationale for Fuzzy Control》等著名论著中首先提出的。主要用于研究现实世界中一些模糊不清的问题,并使之清晰化。模糊集合的引入可将人的判断、思维过程用比较简单的数学形式直接表达出来,从而使对复杂系统做出符合实际和人类思维方式的处理成为可能,为模糊控制的形成奠定了基础。

模糊控制是根据对控制对象的粗略知识以及人们的生产技能等知识,导出自然语言的控制规则,利用模糊理论进行控制的一种控制方法。

模糊逻辑和模糊控制的研究与应用发展是很迅速的,研究成果主要集中在以下几个国家和地区:

(1) 日本模糊控制技术的应用

当代的一些高新技术的发展似乎有这样一个趋向,即欧洲以理论研究为主导,美国将技术研究作为热点,而日本则从事应用技术的开发工作。在模糊控制技术的应用开发上,日本起步较早并率先推出商品,而且,逐渐成为这项技术的主导国家,在模糊技术的应用研究方面日本已走在世界的前列。

从 20 世纪 70 年代初日本就开展模糊控制的研究,1972 年日本关东地区成立了模糊系统研究会,1980 年关西地区成立了模糊科学研究会,接着在 1989 年创建了国际模糊工程研究所 (IIFE),并且在这一年有关模糊技术的产品年产值就达 10 亿日元,其中真正以模糊技术为核心的产品约占 1 亿日元。日本在应用方面最典型的成果有:仙台地铁的模糊控制系统;三菱、日立和富士分别开发的电梯模糊控制系统;理光研究的声音模糊识别系统;日本电机硝子开发的玻璃熔化炉模糊控制系统;大日本涂料开发的彩色印刷图像评价系统;NTT 和法政大学合作研制的模糊识别 CAI 系统。在信息技术方面,日本已推出光模拟计算机、模拟模糊计算机、数字模糊计算机、高速模糊推论用多路板、模糊信息与文件检索系统、模糊数字网络系统、神经模糊推理系统等。日本将模糊控制技术作为跨世纪的热点技术,几乎每天都有新的模糊控制产品问世。

(2) 美国与欧洲模糊控制技术的研究状况

美国的模糊工程是从美国宇航管理局 (NASA) 开始的。NASA 早在 1966 年就给 Zadeh 提供经费,这是有意义的合作开始。目前,美国在模糊控制研究方面仅次于日本。在美国,从 20 世纪 80 年代中期就开始研制模糊控制器。最早应用模糊逻辑控制的是水泥生产的控制,Lafarge 公司在世界各地的水泥窑炉中有 25 个窑炉采用了包含模糊逻辑的 G2 实时专家系统。1993 年 IEEE 神经网络协会的刊物《IEEE TRANSACTIONS ON FUZZY SYSTEM》在美国创刊,这是模糊系统理论已作为一个独立学科发展的标志。美国政府机构(主要是美国国防部)也开始致力于神经网络和模糊逻辑的研究,除了休斯顿的约翰逊宇航中心把模糊逻辑应用于航天飞行器的对接和姿态控制外,位于华盛顿特区的 NASA 格达特宇航飞行中心也把模糊逻辑应用于机器人控制和图像处理,加州的 NASA 艾姆兹研究中心也在进行信息处理的模糊应用研究。美国专门从事模糊控制开发的机构是 Togai Infralogic 公司,主要从事模糊加速板和软

件开发工具的研究。此外, Apronix 公司开发了模糊控制器、模糊推理板、三级倒单摆等产品。

欧洲对模糊逻辑的研究主要集中在模糊控制上。英国伦敦大学的 Mamdani 教授于 1974 年首先把模糊理论用于锅炉和蒸汽机的控制, 开创了模糊控制的先河; 1976 年英国的 R. M. Tong 发表了第一篇有关模糊控制的论文; 德国政府对模糊控制技术的开发应用极为重视, 并已投入 8 亿美元进行模糊逻辑及相关技术的研究, Inform 公司是其主要研究单位之一, 第三代模糊微处理器 Fuzzy—166 芯片就是由 Inform 公司和 Siemens 公司联合研制的。

Fuzzy—166 是一个具有推理命令、模糊化命令和解模糊命令的模糊芯片, 除了可使用 ANSI—C 或 RISC—C 编辑器外, 还可使用扩展了的软件 Fuzzy TecH。该软件提供了一个完整的图形开发环境, 而且, 还可在转换到目标硬件前对模糊系统进行在线模拟和优化。特别是应用神经网络技术可自动生成模糊规则。因此, 德国人自豪地称他们的这一代模糊微处理器为模糊瑰宝。由于德国本身所拥有的先进工业技术和科研条件, 因而就像生产著名的“莱卡”照相机一样, 在模糊控制技术的硬件方面也毫不费力地走在了世界的前列。

德国是生产和使用小轿车的大国, 因此, 对于汽车驾驶的模糊控制特别感兴趣。1991 年春, 在汉诺威工业博览会上展出的模糊逻辑操纵的模型汽车, 尽管其重量只有 3.5kg, 发动机功率仅 1 马力 (1 马力 = 735.499 瓦), 然而它的速度可达 80km/h, 这辆模型汽车是为用模糊逻辑研究复杂动态问题的控制作用而制作的, 预计在汽车上应用模糊控制技术德国将会有更多的发展。

(3) 我国模糊控制技术的研究与发展状况

我国在模糊理论和应用方面的研究起步较晚, 但发展很快。1981 年成立了中国模糊系统和模糊数学学会, 并创办了世界上第二种模糊专业学术杂志——《模糊数学》。随着模糊理论及其技术的发展, 该刊在 1987 年改名为《模糊系统与数学》。目前我国已成为模糊数学研究的四大力量之一 (美国、西欧、日本及中国), 也是世界上模糊控制技术研究领域的领先者之一。在模糊数学理论方面的研究成果已引起国际模糊界的特别关注和重视, 模糊技术的应用研究也在稳步地发展。1979 年, 李宝绥、刘志俊等人用连续数字仿真方法研究了典型模糊控制器的性能; 1988 年 3 月由北京师范大学汪培庄教授及其博士生张洪敏等研制成功的“模糊推理机分离元件样机”, 成功地实现了控制倒摆实验; 1984 年李太航和沈祖良推出了语义推理的自学习算法; 1987 年, 张洪敏和张志明研制成功我国第一台模糊推理机; 1993 年汪培庄的博士生刘增良先后参加国家自然科学基金重大项目“模糊信息处理与机器智能”等, 并提出“因素神经网络理论 (FNN)”, 在此基础上开发完成的“模糊控制计算机系统”和“基于 FNN 的学习型模糊推理控制机”等一系列成果都达到了世界先进水平。玻璃拉管线模糊控制器、电冰箱模糊控制器、可编程模糊控制器 BFEC—898 系列产品、水泥厂矿石破碎机的模糊控制系统都已形成产品并投入运行。近年来, 我国也推出了电烤箱、电烤柜模糊控制器、模糊控制燃气快速热水器、模糊控制自动恒温器、灯光恒照度模糊控制器、模糊全自动洗衣机和模糊电饭锅等产品, 标志着我国模糊技术的应用研究也有了长足的进步。毫无疑问, 十几年来, 我国在模糊逻辑控制的研究和应用开发上已达到了一定的规模, 形成了高新技术领域的研究热点技术。

1.3 模糊控制展望

模糊控制理论是控制理论领域中很有发展前途的一个分支,鉴于它所具有的一系列传统控制无法比拟的优点,国外专家预言,这将是下一代工厂自动化的基础,并称之为21世纪的技术。工厂自动化设备的主要生产厂家美国奥姆伦电子公司的副总经理弗兰克·纽伯恩说:“就工厂而言,模糊逻辑可以帮助那些用常规技术无法实现自动化的任务实现自动化,而且使机器和操作人员之间形成一种新型伙伴关系。”

迄今为止,模糊控制技术已对我国的工业、信息产业、交通等众多领域产生了重要影响,特别是模糊家电产品的出现更是普及了大众对模糊控制技术的认识。但应该指出:目前模糊控制技术尚处于发展的初步阶段,还存在大量有待解决的问题。至今它还没有完善的系统分析手段,而且决定模糊控制系统品质优劣的因素——模糊控制规则的获取以及输入模糊变量论域和隶属函数的确定还都是实验分析和操作人员经验的汇总,受人为主观因素影响较大。

目前模糊控制下一步的发展趋势主要可归结为:

(1) 硬件方面——模糊计算机的研制

模糊计算机的研制包括:模糊计算机体系结构的硬件、操作系统与语言理解等项目。处理模糊信息的处理器——模糊逻辑电路已于1984年由日本研制成功,这意味着模糊控制理论开始深入到硬件技术之中。目前国内已生产出能够设计出几十条控制规则的通用模糊微型机,这种硬件系统的实现将带来模糊控制系统的新飞跃,即由目前结构:“检测—比较—计算—执行”发展到新的结构:“识别—推理—决策—执行”。控制系统的输入、输出除了数据信息外,还可以包括文字符号、图像、语言等符号信息。

(2) 模糊控制理论体系的突破

模糊集理论打开了研究“不确定性”的大门,它不仅包括描述客观存在的现象,而且把人的主观认识也包含在定量的研究方法之中,这正是模糊集理论受重视的原因。同时,它对人的主观认识描述得不够有力与不够深入,也是使得模糊集理论一直被怀疑的原因。学术界有力的争论将进一步推动以模糊集为代表的“不确定性定量方法”的研究与发展。

1) 系统性、奠基性的内容,有待进一步加强研究

在模糊控制基础理论方面,非模糊的精确量该如何处理,目前的处理办法是否最好?在由模糊控制器的输出模糊集确定一个相应的精确控制量,即在解模糊判决问题上,至今尚有争议;在进行模糊化时,输入输出模糊变量的基本论域应分成多少档为最佳?控制规则的条数、采样间隔以及系统增益如何最优化?如何衡量一个模糊系统的功能?稳定性问题、最优化问题应如何评价?模糊控制的一般化设计准则如何?在模糊运算中,似乎丢失了大量的信息,却又能获得优于常规控制的良好控制效果,起控制作用的因素是什么?模糊运算中的信息损失是否应设法修正或补偿?以上关于模糊控制基础理论方面的研究课题都还有待于进一步研究。

2) 立足于高度智能化的模糊控制理论与应用的研究将是今后一段时间的重点课题

高度智能化的模糊系统的理论与模糊计算机的发展,需要模糊控制理论研究进一步深入。特别是模糊控制理论与神经网络、遗传算法相结合,形成了人工智能中的“软计算”科学。

模糊逻辑推理着眼于用模糊语言变量来表达人脑善于处理模糊概念的能力,通过模糊规则作出判断,属于宏观的功能体现。神经网络是从人脑的微观结构出发去模拟人类思维的信息通道和信息处理过程,突出人脑自学习、自适应的功能。显而易见,这两种模仿人脑智力的方法都有各自的优点,也都存在着不足。如果将两者有机地结合起来,取长补短,就可以产生一种最有效的人工智能研究方法,这样的智能模糊控制系统可以根据样本数据学习生成各种模糊变量的论域及模糊控制规则,可以根据相邻规则的情况自动填补空缺的规则,可以根据环境的变化自适应调整控制规则等。近几年来,自动控制界的学者和工程技术人员正在探索将神经网络技术与模糊控制技术相融合的方法,以求提高模糊逻辑推理的智能性。从1991年下半年起,日本的一些公司就开始推出 Neuro-Fuzzy(神经模糊模式的家电产品),如三菱电机的风扇式煤油加热器、三洋的微波炉、东芝的电磁灶、日立的电饭锅、日本电气的空调器等。日前,日本产的 Neuro-Fuzzy 电饭锅已经问世。人们已开始不再满足于有“Fuzzy”标志的产品,而追求更高档、更为舒适的“Neuro-Fuzzy”产品。目前已有的研究成果表明:神经网络技术与模糊控制技术相结合,是模糊控制的一个重要发展趋势。

日前,混沌理论已成为21世纪初最新型智能计算机的三大支柱之一。因而在当今世界上,混沌理论已引起各国学术界和企业界的兴趣。1993年7月于汉城召开的第五届国际模糊系统协会世界大会上,除了有相当数量的与神经网络相结合的模糊神经网络的论文外,还首次出现了与混沌理论相关的研究论文共7篇,其中有在模糊控制中应用混沌理论的,也有在模糊专家系统引入混沌理论的。如韩国金星公司已推出了模糊神经混沌控制的洗衣机,日本的三洋电机公司也开发了混沌型风扇煤油取暖器。

“模糊”、“神经网络”、“混沌”这当代智能技术的新三部曲正在紧密结合,并已形成了产品。研究自然语言的描述,知识的获取与知识的处理,综合使用人类的知识,模拟人的思维过程的合情推理与发现思维,将会大大地促进模糊控制理论的发展与提高。

当然,模糊控制无论在理论上和实用上都是一门“年轻”的科学,正处于不断发展和完善的进程之中,不像经典控制理论和现代控制理论皆已形成了较完善的理论体系。同时,也许也正因为它的不完善和正在发展,显示了它有很大的发展潜力和前途。

第 2 章

模糊逻辑与模糊推理

美国加州大学控制专家 L. A. Zadeh 教授于 1965 年创立了模糊集合理论。模糊理论是在模糊集合理论的基础上发展起来的,主要包括模糊集合理论、模糊逻辑、模糊推理和模糊控制等方面的内容。本章在模糊集合理论的基础上,介绍模糊逻辑与模糊推理。

2.1 模糊集合及其隶属函数

2.1.1 模糊集合及其表示

(1) 模糊集合的概念

集合可以表达概念。符合某概念的对象全体就构成此概念的外延,一个概念所包含的那些区别于其他概念的全体本质属性就是这概念的内涵。用集合论的观点来看,内涵是集合的定义,外延就是组成集合的所有元素。一个概念的外延就是一个集合。

集合中的个体称为元素,通常用小写字母 u, v 表示;集合的全体又称为论域,通常用大写字母 U, V 表示; $u \in U$, 表示元素 u 在集合论域 U 内。一个集合如果是由有限个元素组成,则称为有限集合,不是有限集合的集合称为无限集合。集合可以是连续的,也可以是离散的。

在普通集合中,任何一个元素或个体与任何一个集合之间的关系只有“属于”和“不属于”两种情况,两者必居其一,而且只居其一,绝对不允许模棱两可。例如,“不大于 100 的自然数”是一个清晰的概念,该概念的内涵和外延均是明确的。

在现实世界中,有很多事物的分类边界是不分明的,或者说是难以明确划分的。比如,将一群人划分为“高”和“不高”两类,就不好硬性规定一个划分的标准。如果硬性规定,1.80m 以上的人算“高个子”,否则不算,则有可能出现两个本来身高“基本一样”的人却被认为一个“高”,一个“不高”,这就有悖于常理,因为这两个人在任何人看来都是“差不多高”。这种概念外延的不确定性称它为模糊性。

由此可见,普通集合在表达概念方面有它的局限性。普通集合只能表达“非此即彼”的概念,而不能表达“亦此亦彼”的现象。为此,1965 年美国加州大学控制专家 L. A. Zadeh 教授创立了模糊集合论,提出用模糊集合来刻画模糊概念。

定义 论域 U 中的模糊子集 A , 是以隶属函数 μ_A 表征的集合。即由映射

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

确定论域 U 的一个模糊子集 A 。 μ_A 称为模糊子集 A 的隶属函数, $\mu_A(u)$ 称为 u 对 A 的隶属度, 它表示论域中的元素 u 属于其模糊子集 A 的程度。它在 $[0, 1]$ 闭区间内可连续取值。

$\mu_A(u) = 1$, 表示 u 完全属于 A ; $\mu_A(u) = 0$, 表示 u 完全不属于 A ; $0 < \mu_A(u) < 1$; 表示 u 隶属于 A 的程度。

上述定义表明:

①论域 U 中的元素是分明的, 即 U 本身是普通集合, 只是 U 的子集是模糊集合, 故称 A 为 U 的模糊子集, 简称模糊集。

②隶属函数 $\mu_A(u)$ 是用来说明 u 隶属于 A 的程度的, $\mu_A(u)$ 的值越接近于 1, 表示 u 隶属于 A 的程度越高; 当 $\mu_A(u)$ 的值域变为 $\{0, 1\}$ 时, 隶属函数 $\mu_A(u)$ 蜕化为普通集合的特征函数, 模糊集合也就蜕化为普通集合。

③模糊集合完全由其隶属函数来刻画。隶属函数是模糊数学的最基本概念, 借助于它才能对模糊集合进行量化。

对于模糊概念, 不能仿照清晰概念用“属于”和“不属于”来表达, 故模糊集合也不能像普通集合那样用特征函数值 0 和 1 来描述, 而应该用某元素属于某集合的程度来描述。

为了对“属于”和“不属于”之间存在的无穷多层次、逐渐变化的灰度进行描述, 扎德把普通集合扩展为模糊集合。这就是说, 有些东西可以同时部分属于和部分不属于。这样, 人的自然语言和概念就可以用这种新的数学工具来描述和处理了。

图 2.1 和图 2.2 分别表示两种集合对温度定义的区别。

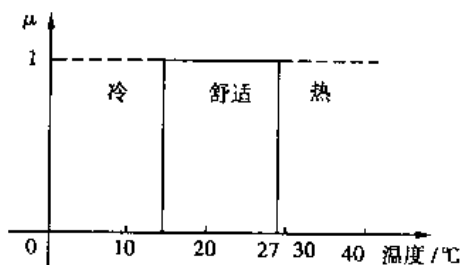


图 2.1 普通集合对温度的定义

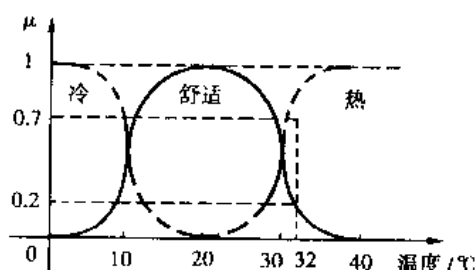


图 2.2 模糊集合对温度的定义

(2) 模糊集合的表示方法

1) Zadeh 表示方法

①当 U 为离散有限论域 $U = \{u_1, u_2, \dots, u_n\}$ 时, 模糊集合 A 表示为:

$$A = \mu_A(u_1)/u_1 + \mu_A(u_2)/u_2 + \dots + \mu_A(u_n)/u_n \quad (2.1)$$

式中, $\mu_A(u_i)/u_i$ 不代表分式, 表示论域 U 中元素 u_i 及其隶属度 $\mu_A(u_i)$ 之间的对应关系。符号“+”也不表示“加法”运算, 而是表示模糊集合在论域 U 上的整体。这是一种列举表示法。

②当 U 为连续无限论域时, 模糊集合 A 表示为:

$$A = \int_u \frac{\mu_A(u)}{u} \quad (2.2)$$

式中, 符号“ \int ”不代表普通积分, 而是表示无限多个元素与其隶属度对应关系的一个总括。

2) 向量表示法

当模糊集合 A 的论域由有限个元素构成时, 模糊集合 A 表示成向量形式:

$$A = [\mu_A(u_1), \mu_A(u_2), \dots, \mu_A(u_n)]$$

注意: 应用向量表示时, 隶属度等于零的项不能舍弃, 必须依次列入。

3) 序偶表示法

若将论域 U 中的元素 u_i 与其对应的隶属度 $\mu_A(u_i)$ 组成序偶 $(u_i, \mu_A(u_i))$, A 可表示为:

$$A = \{(u_1, \mu_A(u_1)), (u_2, \mu_A(u_2)), \dots, (u_n, \mu_A(u_n))\}$$

4) 隶属函数法

用隶属函数的解析表达式表示出相应的模糊集合。

例如, Zadeh 曾用下述表达式来表达模糊集合“老年人”, 其年龄论域 $U = [0, 100]$, u 为在论域上取值的年龄变量。

$$\mu_{\text{老年人}} = \begin{cases} 0, & 0 \leq u \leq 50 \\ 1 / \left[1 + \left(\frac{u-50}{5} \right)^2 \right], & 50 < u \leq 100 \end{cases} \quad (2.3)$$

2.1.2 模糊集合的基本运算及其法则

定义 论域 U 中模糊子集的全体, 称为 U 中的模糊幂集, 记作 $F(U)$, 即

$$F(U) = \{A \mid \mu_A: U \rightarrow [0, 1]\}$$

对于任一 $u \in U$, 若 $\mu_A = 0$, 则称 A 为空集 \emptyset ; 若 $\mu_A = 1$, 则称 $A = U$ 为全集, 通常全集记为 E 。

定义 设 A, B 是论域 U 上的两个模糊集合, 即 $A, B \in F(U)$, 若对任一 $u \in U$, 都有 $\mu_B(u) \leq \mu_A(u)$, 则称 B 包含于 A , 或称 A 包含 B , 记作 $B \subseteq A$; 若对任一 $u \in U$, 都有 $\mu_B(u) = \mu_A(u)$, 则称 B 等于 A , 记作 $B = A$ 。

设 A, B 是论域 U 上的两个模糊集合, 隶属函数分别为 μ_A 和 μ_B , 常用的运算有:

1) “并”运算 $A \cup B$

$$\mu_{A \cup B}(u) = \max[\mu_A(u), \mu_B(u)] = \mu_A(u) \vee \mu_B(u)$$

2) “交”运算 $A \cap B$

$$\mu_{A \cap B}(u) = \min[\mu_A(u), \mu_B(u)] = \mu_A(u) \wedge \mu_B(u)$$

3) “补”运算 \bar{A}

$$\mu_{\bar{A}}(u) = 1 - \mu_A(u)$$

这里, 符号“ \vee ”、“ \wedge ”称为 Zadeh 算子, 为模糊逻辑中的运算符号, 在无限集合中, 它们分别表示 \sup 和 \inf , 在有限元素之间, 则表示 \max 和 \min , 即取最大值和最小值。

2.1.3 模糊集合与普通集合的关系

普通集合表达的是内涵和外延均为明确的清晰概念。普通集合只能表达“非此即彼”的概念, 而不能表达“亦此亦彼”的现象。在普通集合中, 任何一个元素或个体与任何一个集合之间的关系只有“属于”和“不属于”的两种情况, 两者必居其一, 而且只居其一。因此, 描述普通集合的特征函数只能取 0 和 1 两个值, 即特征函数与 $\{0, 1\}$ 两个值相对应。

模糊集合表达的是一类内涵明确而外延不分明模糊概念。这种概念反映了人的认识的主观性, 因此, 模糊集合能够表达“亦此亦彼”的现象。在模糊集合中, 为描述客观事物的中介

过渡性,其特征函数的取值范围在 $[0,1]$ 区间连续取值。为了把两者区分开,就把模糊集合的特征函数称做隶属函数。若隶属函数的取值只取0和1两个值,那么模糊集合就缩简成普通集合。从这个意义上说,模糊集合的隶属函数是普通集合特征函数的扩展和一般化。

2.1.4 模糊集合的隶属函数

(1) 确定隶属函数的原则

隶属函数的确定实质上是人们对客观事物中介过渡的定性描述,这种描述本质上是客观的。由于模糊集理论研究的对象具有“模糊性”和经验性,每个人对同一模糊概念的认识和理解存在差异,因此,隶属函数的确定又含有一定的主观因素。

尽管确定隶属函数的方法带有主观因素,但主观的反映和客观的存在是有一定联系的,是受到客观制约的。因此,隶属函数的确定应遵守一些基本原则。

1) 表示隶属函数的模糊集合必须是凸模糊集合

以主观性最强的专家经验法为例来确定“舒适”温度的隶属函数。

某专家根据他本身的经验对“舒适”温度的隶属函数定义如下:

$$\text{舒适温度} = 0/0^{\circ}\text{C} + 0.5/10^{\circ}\text{C} + 1/20^{\circ}\text{C} + 0.5/30^{\circ}\text{C} + 0/40^{\circ}\text{C}$$

这里隶属度为1的温度点定的是 20°C ,即在 20°C 左右是“舒适”的温度,越是偏离这个温度,其隶属越小,即舒适的程度越小,这与大多数人的经验是吻合的。至于 30°C 的隶属度是0.5而不是0.45,也许只能说这是经验。但是,这种经验并不意味着可以任意确定,因为可以称得上专家的经验,那肯定不是一种具有任意性的经验,通常都是指具有相当成功把握和代表性的经验。通常,某一模糊概念的隶属函数的确定应首先从最适合这一模糊概念的点下手,也即确定该模糊概念的最大隶属函数中心点或区域,然后向两边延伸。从连接各点后经过平滑处理的隶属函数曲线图2.3来看,从最大隶属函数中心点出发向两边延伸时,其隶属函数的值必须是单调递减的,而不允许有波浪形,否则会产生明显不合逻辑的状态。

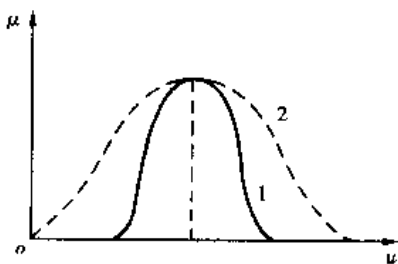


图 2.3 隶属函数向最大值两边延伸的差别

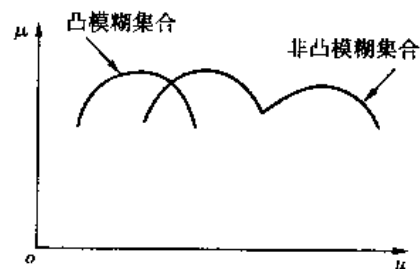


图 2.4 非凸模糊集合隶属函数

2) 变量所取隶属函数通常是对称和平衡的

一般情况下,描述变量的语言值安排得越多,模糊控制系统的分辨率就越高,其系统响应的结果就越平滑;但模糊规则会明显增多,计算时间增加,设计困难加大。如果描述变量的语言值安排得太少,则其系统的响应可能会太不敏感,并可能无法及时提供输出控制跟随小的输入变化,以使系统的输出会在期望值附近振荡。实践表明,一般取3~9个为宜,并且通常取奇数个,在“零”、“适中”或者“正常”集合的两边语言值通常对称取。

3) 隶属函数要符合人们的语义顺序,避免不恰当的重叠

在相同论域上使用的具有语义顺序关系的若干语言值的模糊集合,例如,“冷”、“凉”、“适

中”、“暖”、“热”等模糊子集其中心值位置必须按这一次序排列,不能违背常识和经验。此外,隶属函数由中心值向两边模糊延伸的范围也有一定的限制,间隔的两个模糊集合的隶属函数尽量不相交重叠,如图 2.5 所示。

除以上三条基本原则外,模糊控制系统隶属函数的选择通常应遵循:

①论域中的每个点应该至少属于一个隶属函数的区域,同时,它一般应该属于至多不超过两个隶属函数的区域;

②对同一个点没有两个隶属函数会同时有最大隶属度;

③当两个隶属函数重叠时,重叠部分的任何点的隶属函数的和应该小于等于 1。

为了定量研究隶属函数之间的重叠,Motorola 公司的 Marsh 提出重叠率和重叠鲁棒性的概念,并用这两个指数来描述隶属函数的重叠关系,如图 2.6 所示。定义如下:

$$\text{重叠率} = \frac{\text{重叠范围}}{\text{附近模糊隶属函数的范围}}$$

$$\text{重叠鲁棒性} = \frac{\text{总的重叠面积}}{\text{总的重叠最大面积}} = \frac{\int_L^U (\mu_{A_1} + \mu_{A_2}) dx}{2(U - L)}$$

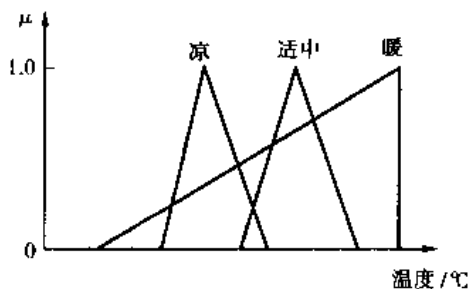


图 2.5 交叉越界的隶属度函数示意图

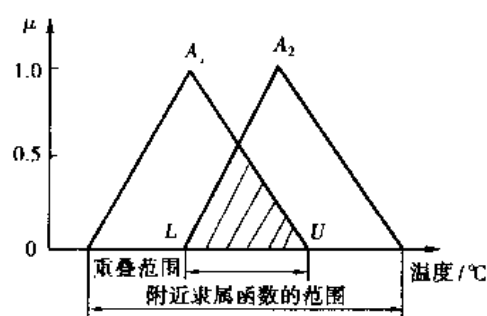


图 2.6 重叠指数定义

对于重叠指数的选择,一般取重叠率为 0.2 ~ 0.6 为宜,重叠鲁棒性的值通常比重叠率稍大一点,一般在 0.3 ~ 0.7。重叠率和重叠鲁棒性越大,模糊控制模块就更具有模糊性,而低重叠指数适用于有较大明确相关性的输入输出系统。为了使模糊控制模块更平滑地操作,应该选择一个成熟的重叠率和重叠鲁棒性。例如,重叠率可取 0.33,重叠鲁棒性可取 0.5。

(2) 确定隶属函数的方法

隶属函数是模糊集合应用于实际问题的基础,正确构造隶属函数是能否用好模糊集合的关键。然而,目前确定隶属函数还没有一种成熟有效的方法,大多数系统的隶属函数仍然还停留在依靠经验确定,然后再通过实验、试验或者计算机模拟得到的反馈信息进行修正。

这里介绍几种常用确定隶属函数的方法。

1) 模糊统计法

模糊统计是对模糊性事物的可能性程度进行统计,其统计结果即为隶属度。其基本思想是:对论域 U 上的一个确定元素 u_0 ,考虑 n 个有模糊集合 A 属性的普通集合 A^* 以及元素 u_0 对 A^* 的归属次数。 u_0 对 A^* 的归属次数和 n 的比值就是元素 u_0 对模糊集合 A 的隶属度:

$$\mu_A(u_0) = \lim_{n \rightarrow \infty} \frac{u_0 \in A^* \text{ 的次数}}{n} \quad (2.4)$$

2) 专家经验法

由专家的实际经验给出模糊信息的处理算式或相应权系数来确定隶属函数的方法。

3) 二元排序法

这是一种较实用的确定隶属函数的方法。它通过对多个事物之间两两对比来确定某种特征下的顺序,由此来决定这些事物对该特征的隶属函数的大致形状。

4) 典型函数法

根据问题的性质,应用一定的分析与推理,选用某些典型函数作为隶属函数。如三角形函数、梯形函数等。

(3) 常用隶属函数的图形

基本的隶属函数图形可分成三类:左大右小的偏小型下降函数(称做Z函数)、对称型凸函数(称作H函数)和右大左小的偏大型上升函数(称做S函数),如图2.7所示。

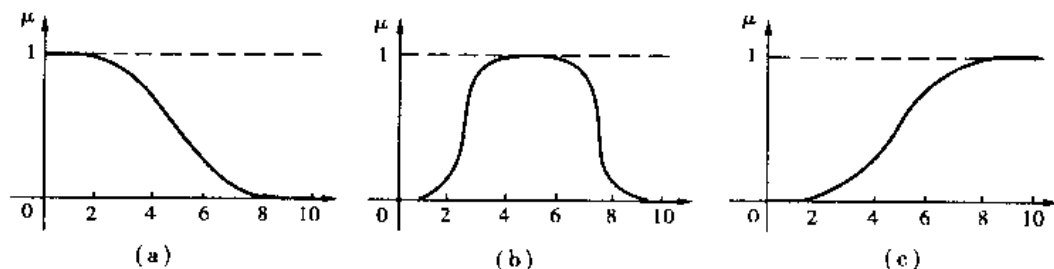


图2.7 基本隶属函数图形

(a)Z函数;(b)H函数;(c)S函数

最简单的隶属函数还是取直线型的三角形和梯形,因而常被人们使用,如图2.8所示。

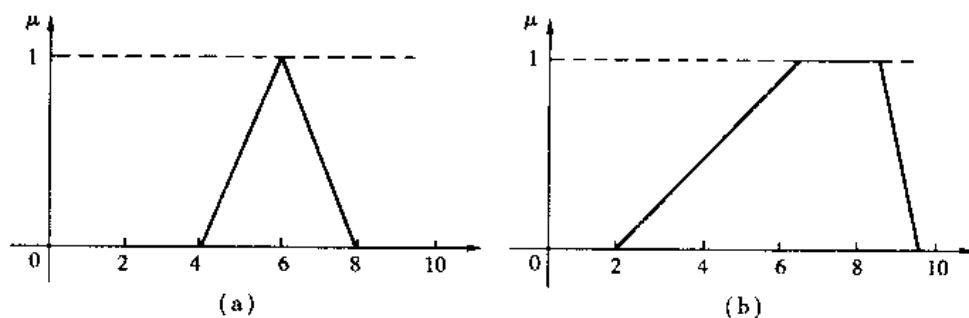


图2.8 直线型隶属函数

(a)三角形函数;(b)梯形函数

2.2 模糊矩阵与模糊关系

2.2.1 模糊矩阵

(1) 模糊矩阵的概念及其运算

定义 矩阵 $R = (r_{ij})_{n \times m}$ 称做模糊矩阵,如果对任意的 $i \leq n$ 及 $j \leq m$, 都有 $r_{ij} \in [0, 1]$ 。

定义 对于任意的模糊矩阵 $R = (r_{ij})_{n \times m}$, $S = (s_{ij})_{n \times m}$, $R \cup S = (r_{ij} \vee s_{ij})_{n \times m}$ 称做模糊矩阵

R 和 S 的并; $R \cap S = (r_{ij} \wedge s_{ij})_{n \times m}$ 称做模糊矩阵 R 和 S 的交; $\bar{R} = (1 - r_{ij})_{n \times m}$ 称做模糊矩阵 R 的余矩阵; 如果 $r_{ij} \leq s_{ij}$, ($i = 1, 2, \dots, n; j = 1, 2, \dots, m$) 则称模糊矩阵 R 被模糊矩阵 S 包含, 记作 $R \subseteq S$ 。

(2) 模糊矩阵的合成

定义 一个 n 行 m 列的模糊矩阵 $R = (r_{ij})_{n \times m}$ 对一个 m 行 l 列的模糊矩阵 $S = (s_{jk})_{m \times l}$ 的合成 $R \circ S$ 指的是一个 n 行 l 列的模糊矩阵 T , T 的第 i 行第 k 列元素 t_{ik} 等于 R 的第 i 行的元素与 S 的第 k 列的对应元素两两先取较小者, 然后在所得的结果中取较大者, 即

$$t_{ik} = \bigvee_{j=1}^m (r_{ij} \wedge s_{jk}) \quad (1 \leq i \leq n, 1 \leq k \leq l)$$

2.2.2 模糊关系

(1) 普通关系

关系是客观世界存在的普遍现象, 它描述了事物之间存在的某种联系。例如, 人与人之间有父子、亲戚、同事关系; 数与数之间有大于、等于、小于等关系; 元素与集合之间有属于、不属于等关系。两个客体之间的关系称为二元关系, 表示三个以上客体之间的关系称为多元关系。普通关系只表示元素之间是否关联。

1) 集合的直积

由两个集合 U 和 V 的各自元素 u 与 v 组成的序偶 (u, v) 的全体集合, 称为 U 与 V 的直积, 记为 $U \times V$, 即

$$U \times V = \{ (u, v) \mid u \in U, v \in V \}$$

一般情况下, $U \times V \neq V \times U$ 。

2) 普通二元关系

如果对集合 U, V 的元素之间的搭配 $[(x, y), x \in U, y \in V]$ 施加某种限制, 这时构成的集合是直积 $U \times V$ 的一个子集合。该子集具有某种特定性质, 其性质的内容包含于搭配的限制之中, 它反映 U, V 元素之间的某种特定关系。

定义 设 U 与 V 是两个非空集合。集合 U, V 的直积 $U \times V$ 的一个子集 R 称为 U 到 V 上的一个二元关系, 简称关系。

对于直积 $U \times V$ 的序偶 (u, v) , 要么 (u, v) 具有关系 R , 记为 $(u, v) \in R$; 要么 (u, v) 不具有关系 R , 记为 $(u, v) \notin R$ 。因此, 关系 R 的特征函数为:

$$C_R(u, v) = \begin{cases} 1, & (u, v) \in R \\ 0, & (u, v) \notin R \end{cases} \quad (2.5)$$

若 $U = V$, 则直积 $U \times V$ 的子集 R 称为 U 上的二元关系, 或称 U 上的关系。

3) 关系矩阵

关系 R 可以用矩阵来表示, 称为关系矩阵, 其中元素 r_{ij} 基于特征函数 $C_R(u, v)$ 的定义, 即

$$r_{ij} = C_R(u_i, v_j) = \begin{cases} 1, & (u_i, v_j) \in R \\ 0, & (u_i, v_j) \notin R \end{cases} \quad (2.6)$$

与序偶 $(u_i, v_j) \in R$ 对应者记为 1, 与序偶 $(u_i, v_j) \notin R$ 对应者记为 0。

(2) 模糊关系

关系是描述客观事物之间联系的重要概念。普通关系 R 描述了事物之间“有”与“无”的

肯定关系,但有些事物不能简单地用肯定或否定的词汇去明确表达它们之间的关系。如“ A 与 B 很相似”、“ X 比 Y 大很多”、“他比较能干”等,这些语句是日常生活中人们常常会遇到的,它们表达了客观事物之间另一种不明确、不确定的关系,称为模糊关系。模糊关系是普通关系的拓广和发展。它比普通关系的含义更丰富、更符合客观实际的多数情况。

定义 集合 U 和 V 的直积

$$U \times V = \{(u, v) \mid u \in U, v \in V\} \quad (2.7)$$

中的模糊子集 R 被称为 U 到 V 的模糊关系,又称为二元模糊关系,其特性用隶属函数描述如下:

$$\mu_R: U \times V \rightarrow [0, 1]$$

当论域 U 和 V 相同时, R 称为 U 上的模糊关系。当论域为 n 个集合 $U_i (i=1, 2, \dots, n)$ 的子集 $U_1 \times U_2 \times \dots \times U_n$ 时,它们所对应的模糊关系 R 称为 n 元模糊关系。

对于 $(u, v) \in U \times V$, $\mu_R(u, v)$ 表达 u 对 v 有关系 R 的程度或 u 对 v 的关系 R 的相关程度。

(3) 模糊关系的表示

模糊关系也是模糊集合,所以模糊关系也可用模糊集合的表示方法。

1) 模糊集合表示法

$$R = \int_{U \times V} \mu_R(u, v) / (u, v)$$

或

$$R = \int_{U \times V} ((u, v), \mu_R(u, v)) \quad (u, v) \in U \times V$$

例 2.1 设集合 $U = \{1, 2, 3\}$, $V = \{1, 2, 3, 4, 5\}$, 从 U 到 V 的一个模糊关系 R 可表示为:

$$R = 0.5/(1, 3) + 0.8/(1, 4) + 1/(1, 5) + 0.5/(2, 4) + 0.8/(2, 5) + 0.5/(3, 5)$$

2) 模糊关系表表示法

模糊关系 R 可用模糊关系表来表示。

例 2.2 例 2.1 中模糊关系 R 的模糊关系表为:

表 2.1 U 与 V 的模糊关系表

		V				
		1	2	3	4	5
U	R					
	1	0	0	0.5	0.8	1
	2	0	0	0	0.5	0.8
	3	0	0	0	0	0.5

3) 模糊矩阵表示法

当 U, V 是有限集合时,定义在 $U \times V$ 上的模糊关系 R 可用模糊矩阵来表示。

例 2.3 例 2.1 中模糊关系 R 的矩阵表示为:

$$R = \begin{bmatrix} 0 & 0 & 0.5 & 0.8 & 1 \\ 0 & 0 & 0 & 0.5 & 0.8 \\ 0 & 0 & 0 & 0 & 0.5 \end{bmatrix}$$

4) 模糊关系图表示

用图直观表示模糊关系时,则将 u_i, v_j 作为节点,在 u_i 到 v_j 的连线上标上 $\mu_R(u_i, v_j)$ 的值,这样的图便称为模糊关系图。

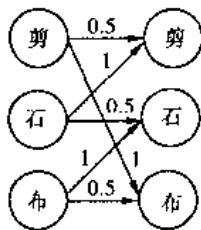


图 2.9 模糊关系图

例 2.4 二人博弈,具有相同的策略集: $U = V = \{\text{剪刀, 石头, 布}\}$,“甲胜”定为 1;“平局”定为 0.5;“甲负”定为 0。则二人胜负关系可用模糊关系图表示,如图 2.9 所示。

(4) 模糊关系的合成

定义 设 U, V, W 是论域, R 是 U 到 V 的一个模糊关系, S 是 V 到 W 的一个模糊关系,则 R 对 S 的合成 $R \circ S$ 指的是 U 到 W 的一个模糊关系 T ,它具有隶属函数:

$$\mu_{R \circ S}(u, w) = \bigvee_{v \in V} (\mu_R(u, v) \wedge \mu_S(v, w)) \quad (2.8)$$

当 U, V, W 为有限时,模糊关系的合成可用模糊矩阵的合成来表示。设

$$R = (r_{ij})_{n \times m}, S = (s_{jk})_{m \times l}, T = (t_{ik})_{n \times l}$$

则

$$t_{ik} = \bigvee_{j=1}^m (r_{ij} \wedge s_{jk})$$

定义 设 R 是 U 上的一个模糊关系。

- ①如果对于任意的 $u \in U$, 都有 $\mu_R(u, u) \equiv 1$, 则称 R 为自反模糊关系。
- ②如果对于任意的 $u, v \in U$, 都有 $\mu_R(u, v) \equiv \mu_R(v, u)$, 则称 R 为对称模糊关系。
- ③如果对于任意的 $\lambda \in [0, 1]$, R_λ 都是具有传递性的普通关系, 则称 R 为传递模糊关系。

定义 模糊关系 R 和 S , 如果总是存在 $\mu_R(x, y) = \mu_S(x, y)$, 则称模糊关系 R 与模糊关系 S 等价, 记为 $R = S$ 。

$$R = S \leftrightarrow \mu_R(x, y) = \mu_S(x, y), \forall x \in X, y \in Y$$

2.2.3 模糊映射

(1) 普通映射的扩展

映射是一种特殊的关系, 设有从 X 到 Y 的关系 R , 对于任意的 $x \in X$, 都存在惟一确定的 $y \in Y$, 使 x 与 y 有关系 R (简记为 xRy), 具有这种性质的关系, 就称做从 X 到 Y 的一个映射, 记为:

$$\begin{aligned} f: X &\rightarrow Y \\ x &\propto y \end{aligned}$$

其中, $X \rightarrow Y$ 指明 f 是从 X 到 Y 的映射, $x \propto y$ 则指明具体的对应法则: 对于任意的 $x \in X$ 对应于它的是 $y = f(x)$ 。例如:

$$\begin{aligned} f: R &\rightarrow R \\ x &\propto x^2 \end{aligned}$$

表示 f 是从实数域 R 到自身的映射, 对于任意一个实数 x , 对应于它的是实数 x^2 , 也即

$$f(x) = x^2 \quad (\forall x \in R)$$

给定了普通映射

$$f: X \rightarrow Y$$

总可以把它扩展成为这样的映射 (仍记作 f)

$$f: F(X) \rightarrow F(Y)$$

$$A \propto f(A) = \{y | y \in Y, \exists x \in A, y = f(x)\}$$

这个新的映射 f 是从 X 的幂集 $F(X) = \{A | A \subset X\}$ 到 Y 的幂集 $F(Y) = \{B | B \subset Y\}$ 的一个普通映射。

定义 给定映射 $f: X \rightarrow Y$, 可以把它扩展成为映射 (仍记作 f)

$$f: F(X) \rightarrow F(Y)$$

$$A \propto f(A) = \int_Y \mu_A(x) / f(x)$$

对任意的 $x \in X$, 元素 $f(x) \in Y$ 所对应的隶属函数是 $\mu_A(x)$; 如果 x 有多个值 $x_i (i \in T)$ 都有同一个像 y , 即

$$f(x_i) = y \quad (i \in T)$$

则 y 对 $f(A)$ 的隶属度为:

$$\mu_{f(A)}(y) = \bigvee_{x: f(x)=y} \mu_A(x)$$

扩展原则采用 Zadeh 记法, 有一种很直观的解释: 给定 $f: X \rightarrow Y$, 对于 X 的任意一个模糊子集 A , 要问它在 f 之下的像是什么? 只要遵循这样一条原则: x “携带隶属度 $\mu_A(x)$ ” 到 $f(x)$ 上去, $f(x)$ 对 $f(A)$ 的隶属度完全由 x 对 A 的隶属度所确定。其直观图像见图 2.10。

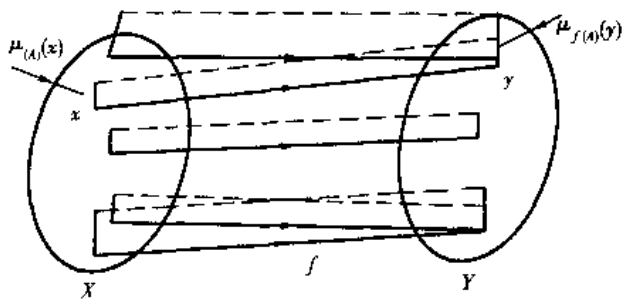


图 2.10 普通映射扩展直观图

(2) 模糊映射和模糊变换

定义 设集合 X, Y 是非空的, 如果存在一个法则 f , 通过它, 对于 X 中的任意元素 x , 都有 Y 中的惟一确定的子集 B 和 x 对应, 则称 f 为从 X 到 Y 的点—集映射, 记作

$$f: X \rightarrow F(Y)$$

$$x \propto f(x) = B$$

当一个从 X 到 Y 的点—集映射全部像都是单点集时, 这个点—集映射便自然地对应于一个从 X 到 Y 的普通映射。因此, 点—集映射是普通映射的推广。

定义 设集合 X, Y 是非空的, 如果存在一个法则 T , 通过它, 对于 X 中的任意一个子集 A , 都有 Y 中的惟一确定的子集 B 和 A 对应, 则称 T 为从 X 到 Y 的集合变换, 记作

$$T: F(X) \rightarrow F(Y)$$

$$A \propto T(A) = B$$

由定义可知, 从 X 到 Y 的集合变换 T 也是从 $F(X)$ 到 $F(Y)$ 的映射。图 2.11 是映射、点—集映射和变换的示意图。

定义 设集合 X, Y 是非空的, 如果存在一个法则 f , 通过它, 对于 X 中的任意元素 x , 有 Y 中的惟一确定的模糊子集 B 和它对应, 则称 f 为从 X 到 Y 的模糊映射, 记作

$$f: X \rightarrow F(Y)$$

$$x \propto f(x) = B$$

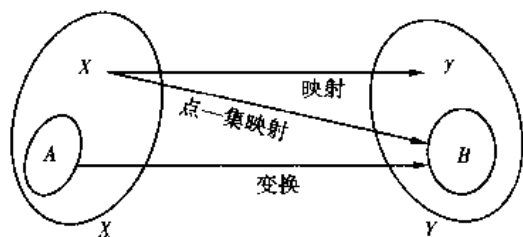


图 2.11 映射、点-集映射和变换示意图

由定义可知,从 X 到 Y 的模糊变换可以看做是从 $F(X)$ 到 $F(Y)$ 的映射

(3) 模糊映射、模糊关系和模糊变换之间的关系

如果给定了一个从 X 到 Y 的模糊映射 f ,对于 X 中的任意元素 x ,通过 f 可以得到 Y 上的一个模糊子集 $f(x)$,于是,对于 Y 的任意一个元素 y 关于 $f(x)$ 的隶属函数就惟一确定了。即

$$\mu_{f(x)}(y) \in [0,1]$$

如果让 (x,y) 与 $\mu_{f(x)}(y)$ 建立一种对应关系,这样就惟一确定一个从 X 到 Y 的模糊关系 R_f (下标 f 表示这个模糊关系是由 f 确定的),称 R_f 是由 f 诱导出的模糊关系。于是,得到这样一个结论:

任给一个从 X 到 Y 的模糊映射

$$f: X \rightarrow F(Y)$$

都惟一确定一个模糊关系 $R_f \in F(X \times Y)$,满足

$$R_f(x,y) = \mu_{f(x)}(y) \quad \forall (x,y) \in X \times Y$$

类似地,任给一个模糊关系 $R_f \in F(X \times Y)$,可以确定一个从 X 到 Y 的模糊变换 T_f ,定义如下:

$$T_f: F(X) \rightarrow F(Y)$$

$$A \propto T_f(A) = A \circ R_f \in F(Y) \quad \forall A \in F(X)$$

其中, R_f 是由 f 诱导出的从 X 到 Y 的模糊关系,“ \circ ”表示合成运算符。

2.3 模糊语言与模糊逻辑

2.3.1 模糊语言

语言是一种符号系统,是人们表达思维的一种形式。目前有两种大的语言类型——自然语言和人工语言。自然语言是指人们在日常生活和工作中所使用的语言,是以字、词为符号的一种符号系统。人工语言是人类在认识自然世界和改造自然世界的过程中人工创造的语言,如对物理世界进行抽象化的数学语言以及机器所用的各种形式语言。

自然语言的突出特点是具有模糊性,例如,“今天是个好天气”,“他跑得快”等。人们能够识别和理解具有模糊性的自然语言,是因为人们对语言的理解也是具有模糊性的。

具有模糊性的语言称为模糊语言。模糊语言虽然不够精确和严格,但它并不影响人们表达自己的思想和相互间的信息交流;恰恰相反,正是这种模糊性才使得自然语言更加生动和富有表现力,也才显示出了人们判断和处理模糊现象的能力。

(1) 语言变量

语言变量是自然语言中的字、词或句作为名称,并且以自然语言中的单词或词组作为值的变量,它不同于一般数学中以数为值的数值变量。因此,语言变量实际上是一种模糊变量,是用模糊语言表示的模糊集合。例如,若“年龄”看成是一个模糊语言变量,则它的取值不是具体岁数,而是诸如“年幼”、“年轻”、“年老”等用模糊语言表示的模糊集合。

语言变量用一个有5个元素的集合 $(N, T(N), U, G, M)$ 来表征,其中

- ① N 是语言变量的名称,如年龄、颜色、速度、体积等;
- ② U 是 N 的论域;
- ③ $T(N)$ 是语言变量值 X 的集合,每个语言值 X 都是定义在论域 U 上的一个模糊集合;
- ④ G 是语法规则,用以产生语言变量 N 的语言值 X 的名称;
- ⑤ M 是语义规则,是与语言变量相联系的算法规则,用以产生模糊子集 X 的隶属函数。

语言变量通过模糊等级规则,可以给它赋予不同的语言值,以区别不同的程度。

以语言变量名称 $N = \text{“年龄”}$ 为例,则 $T(\text{年龄})$ 可以选取为: $T(\text{年龄}) = (\text{很年轻, 年轻, 中年, 老, 很老})$, 上述每个模糊语言值如老、中、轻等是定义在论域 U 上的一个模糊集合, 设论域 $U = [0, 120]$ 。语言变量的五元体之间的相互关系可以用图 2.12 来表示。

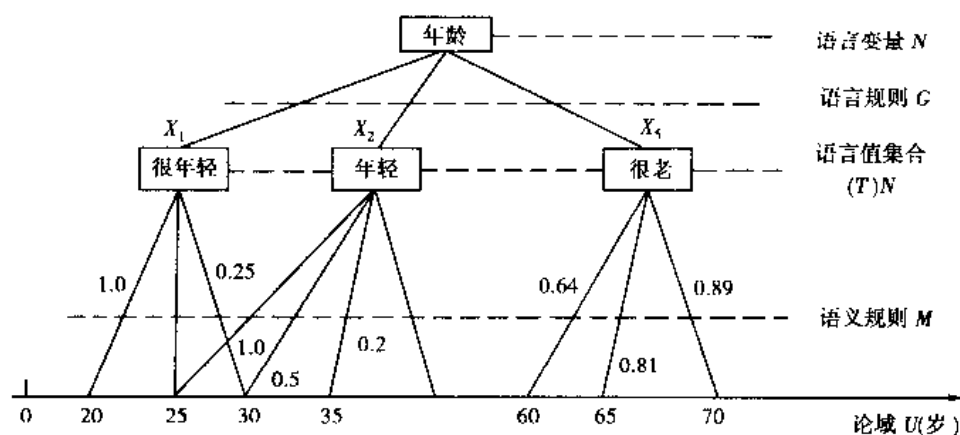


图 2.12 语言变量体系结构

(2) 语言算子

自然语言具有模糊性和灵活性。语言算子是指语言系统中的一类修饰字词的前缀词或模糊量词,通常加在单词或词组的前面,用来调整单词或词组的含义。例如,“新书”不同于“书”,“太凉”有别于“凉”,词义有所变化。语言算子有“较”、“很”、“稍微”、“相当”、“大约”、“近似”、“倾向于”等,语言算子也称为模糊算子。根据语言算子的功能不同,通常又分为语气算子、模糊化算子、判定化算子三种。

1) 语气算子

语气算子用于表达语言中对某个单词或词组的确定性程度。

设有论域 U , 若存在单词 A , 有隶属函数 $\mu_A(u) = \lambda$, 则

① 集中化算子

在单词 A 前面加上模糊量词 S 后有:

$$\mu_{SA}(u) < \lambda$$

则称 S 为集中化算子。

集中化算子是起强化语气作用的语气算子(如“极”、“很”、“相当”等),可以使模糊语言值的隶属度分布向中央集中,如图 2.13(a)所示。

②散漫化算子

在单词 A 前面加上模糊量词 Q 后有:

$$\mu_{QA}(u) > \lambda$$

则称 Q 为散漫化算子。

散漫化算子是起弱化语气作用的语气算子(如“较”、“略微”、“稍微”等),可使模糊语言值的隶属度分布由中央向两边弥散,如图 2.13(b)所示。

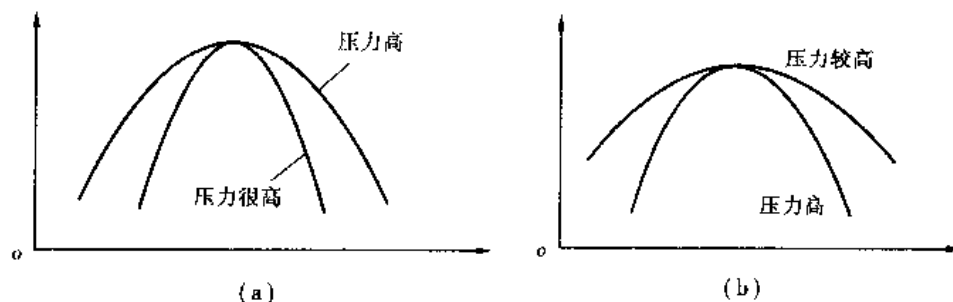


图 2.13 语气算子的作用

(a) 强化作用; (b) 弱化作用

2) 模糊化算子

模糊化算子用来使语言中某些具有清晰概念的单词或词组转化为模糊词义,或者使原来就是模糊概念的词更加模糊化。模糊化算子有“大约”、“近似”、“大概”等。

模糊化算子如果对数字进行作用,就把精确数转化为模糊数。例如,1.7m 是精确数,“近似 1.7m”就是模糊数。模糊化算子如果对模糊值进行作用,就使模糊值更模糊。例如,“年轻”是个模糊值,“大约年轻”就更模糊。

3) 判定化算子

判定化算子的作用是把模糊值进行倾向判断,对模糊值做出肯定化处理。诸如,“倾向于”、“偏向于”等。常把隶属度为 0.5 作为分界来倾向判断。

例如,年老的隶属函数为:

$$\mu_{\text{年老}}(x) = \begin{cases} 0, & 0 < x < 50 \\ \frac{1}{1 + [(x - 50)/5]^{-2}}, & x \geq 50 \end{cases}$$

则“偏老”可用 $\mu_{\text{偏老}}(x) = 0.5$ 所对应的年龄 x 为“偏老”的界线:

$$\frac{1}{1 + [(x - 50)/5]^{-2}} = 0.5$$

求出 $x = 55$ 。得到“偏老”的明确界线:

$$\text{“偏老”}(x) = P_{0.5}(\mu_{\text{年老}}(x)) = \begin{cases} 0, & x < 55 \\ 1, & x \geq 55 \end{cases}$$

正是由于语言变量适于表达因复杂而无法获得确定信息的概念和现象,它为这些通常无法进行量化的“量”提供了一种近似处理方法,把人的直觉经验进行量化,转化成计算机可以操作的数值运算,使人们有可能把专家的控制经验转化成控制算法,并实现模糊控制。

2.3.2 模糊逻辑

(1) 模糊逻辑概念

1) 二值逻辑、多值逻辑和模糊逻辑

在经典集合与二值逻辑中,认为所有的分类都有明确的边界,任一被讨论的对象要么属于这一类,要么就不属于这一类;一个命题不是真就是伪,不存在亦真亦伪或者非真非伪的情况。经典集合与二值逻辑有两个不可证明的公理:矛盾律和排中律。

矛盾律指明“同一件事不能同时既属于又不属于某个对象或者同一种方式”,排中律则更加明确“任何命题要么是肯定,要么是否定”。

由此经典集合与二值逻辑遇到了一些不能解决的问题。例如,古希腊的垛堆佯谬问题:从一堆沙子中取一粒沙,仍然还是一堆;再取一粒,还是一堆;一直取下去,最后还剩下一粒沙子,它还是一堆吗?再取走这一粒就什么也没有了,那是一堆吗?如果这不能算是一堆了,那么什么时候停止取时留下的才算是一堆呢?这种问题在经典集合论和二值逻辑中都是进退两难的问题。实际上,所有在时间上连续变化的事物和现象都存在这种矛盾。

首先突破二值逻辑的先行者是波兰的逻辑学家和哲学家 J. 卢卡斯维兹(Jan Lukasiewicz)(1878—1955),1920年他在二值逻辑的基础上,扩展成一个三值逻辑世界。他用1表示真,0表示假,另外用 $1/2$ 表示可能性,这看起来好像仅仅是插入一个值,然而却是一个突破,它导致了某事物的反面与其本身等效的“谬论”。

经典逻辑这样表达命题:“明天将下雪是真”;其反面则是:“明天将不会下雪是真”。

J. 卢卡斯维兹加上另外一种表述:“明天将下雪是可能的”,这种表述的逻辑值是 $1/2$;其反面是:“明天将不会下雪是可能的”,这种表述的逻辑值也是 $1/2$,当然,“ $1/2 = 1/2$ ”,这就是说“状态 = 反状态”。

在二值逻辑中插入的第三个逻辑值就像一个楔的作用,一旦这个口子被打开,就没有理由只能在其中插入一个值,那就可以插入任意多的值,这就构成了多值逻辑,这实际上是模糊逻辑的亚结构。用多值逻辑就可以表述一个命题的真的程度,这就为人们能更细致、更精确、更准确地进行逻辑判断提供了基础和基本条件。

模糊逻辑是在 J. 卢卡斯维兹多值逻辑基础上发展起来的,它承认从0到1之间有无穷多个相互重叠渗透的中介,或者说没有任何明确的中介具有典型的亦此亦彼性。

用模糊逻辑结构就可以解决那些在二值逻辑中感到棘手而尴尬的问题。例如,模糊逻辑就可很容易地解决“垛堆佯谬”问题。随着取走每一粒沙,沙堆在堆的集合中的隶属度就越来减小,它从1开始,慢慢减到0.8、0.6、0.2,最后到0。

模糊逻辑是通过模仿人的思维方式来表示与分析不确定、不精确信息的方法和工具。在模糊控制中的每一个特定的输入都对应着一个实际的输出,并且,这个输出值是完全可以预测的。所以,模糊逻辑并不是“模糊”的逻辑,而是用来对“模糊”进行处理,以达到消除模糊的逻辑。模糊逻辑是一种精确地解决不精确、不完全信息的一种方法,其最大特点就是用它可以比较自然地处理人的概念。

2) 模糊命题

在逻辑学中,命题是一个基本概念。普通命题就是一个意义明确、可以确定真假的陈述句,在推理上表现为二值逻辑。例如,“太阳从东边升起”是真命题。“月亮会自己发光”是假

命题。然而,有些陈述句含有模糊概念,无法直接用真假来判断。例如,“她很高兴”、“这里空气很清新”。含有模糊概念或具有模糊性的陈述句称为模糊命题。“高兴”、“清新”这些概念的界线是不分明的,因此,模糊命题不能用真假来明确判断。

这里用模糊集合来表示一个模糊命题中的模糊概念。模糊命题在推理上表现为模糊逻辑。模糊命题的真值是介于 $[0,1]$ 之间的值,即命题的真假即是命题对绝对真的隶属度。故模糊命题是一种连续逻辑,也是普通命题的推广。

(2) 模糊逻辑基本运算

常用的模糊逻辑运算定义如下:

- ①逻辑或: $P \vee Q = \max(P, Q)$
- ②逻辑与: $P \wedge Q = \min(P, Q)$
- ③逻辑非: $\bar{P} = 1 - P$
- ④蕴含: $P \rightarrow Q = ((1 - P) \vee Q) \wedge 1$
- ⑤等价: $P \leftrightarrow Q = (P \rightarrow Q) \wedge (Q \rightarrow P)$
- ⑥界限差: $P \ominus Q = (P - Q) \vee 0$
- ⑦界限和: $P \oplus Q = (P + Q) \wedge 1 = \min(P + Q, 1)$
- ⑧界限积: $P \otimes Q = (P + Q - 1) \vee 0 = \max(P + Q - 1, 0)$

(3) 模糊逻辑公式

模糊逻辑变量和运算符号或“ \vee ”、与“ \wedge ”、非“ $-$ ”及括号构成的表达式称为模糊逻辑公式,又称为模糊逻辑函数。模糊逻辑基本公式可推导如下:

- ①幂等律: $P \vee P = P$
 $P \wedge P = P$
- ②交换律: $P \vee Q = Q \vee P$
 $P \wedge Q = Q \wedge P$
- ③结合律: $P \vee (Q \vee R) = (P \vee Q) \vee R$
 $P \wedge (Q \wedge R) = (P \wedge Q) \wedge R$
- ④吸收律: $P \vee (P \wedge Q) = P$
 $P \wedge (P \vee Q) = P$
- ⑤分配律: $P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$
 $P \wedge (Q \vee R) = (P \wedge Q) \vee (P \wedge R)$
- ⑥还原律: $\bar{\bar{P}} = P$
- ⑦德·摩根律: $\overline{P \vee Q} = \bar{P} \wedge \bar{Q}$ $\overline{P \wedge Q} = \bar{P} \vee \bar{Q}$
- ⑧常数运算法则: $1 \vee P = P$ $0 \vee P = P$
 $0 \wedge P = 0$ $1 \wedge P = P$

注意:在模糊逻辑中,没有互补律。

2.4 模糊推理

2.4.1 模糊推理方法

(1) 模糊推理概念

在人类社会发展过程中,二值逻辑起了不少作用。用二值逻辑进行推理,只要大前提或者推理规则是正确的,小前提是肯定的,那么就一定会得到确定的结论。二值逻辑三段论推理结构为:

大前提:若 A 则 B

小前提:如今 A

结 论: B

例如:大前提: A 住院则 A 生病

小前提:如今 A 住院

结 论: A 生病

然而,在客观现实中,人们常常获得的信息是不精确、不完全的,但又必须对具有模糊性的命题或信息进行判断和决策。显然,这不同于二值逻辑推理方法。人们在日常生活中,对具有模糊性的命题或信息是采用近似推理的方法进行推理的,应用模糊集合理论,可以对近似推理进行定量的讨论。

模糊推理是不确定性推理方法的一种,其基础是模糊逻辑,它是一种以模糊判断为前提,运用模糊语言规则,推出一个新的近似的模糊判断结论的方法。虽然它的理论问题一直有争论,但在应用实践中证明是有效的,并且用这种推理方法得到的结论与人的思维一致或相近。

例 2.5 大前提:芒果黄了就成熟了

小前提:芒果黄了

结 论:芒果成熟了

这里“黄了”和“成熟了”都是模糊概念,但是大前提中的前件和后件是明确等价的,所以可以直接替换,这是一种直接推理,推理过程并无模糊性,仍然是精确推理。

例 2.6 大前提: A 健康则长寿

小前提: A 很健康

结 论: A 近乎会很长寿

这里小前提中的模糊判断和大前提的前件不是严格相同,而是相近,它们有程度上的差别,这就不能得到与大前提中后件的明确结论;其结论也应该是与大前提中后件相近的模糊判断。这种结论不是从前提中严格地推出来,而是近似逻辑地推出结论的方法,通常就称为假言推理或似然推理。

从以上分析可知,决定是不是模糊逻辑推理,并不是看前提和结论中是否使用模糊概念,而是看推理过程是否具有模糊性,具体表现在推理规则是不是模糊的。

(2) 模糊推理合成规则

模糊推理是一种近似推理。近似推理的提法基本是下面两种形式:

广义前向推理:给定一个模糊蕴含关系“若 A 则 B ”, $A \in U, B \in V$; 已知某一个 $A_1 \in U$, 求从蕴含关系能推断出什么样的结论 B_1 。

广义后向推理:给定一个模糊蕴含关系“若 A 则 B ”, $A \in U, B \in V$; 已知某一个 $B_1 \in V$, 求从蕴含关系能推断出什么样的结论 A_1 。

模糊控制中,应用的是与前向数据驱动推理有密切关系的前向推理,后向推理则在专家系统中经常用到。模糊蕴含关系“若 A 则 B ”一般是经过大量的实验或经验得到的,它是统计的结果。因此,这些实验或经验的量越大越多,则所得的蕴含关系就越准确。不过,在近似推理中,总是认为所得的蕴含关系是正确和可靠的,因为它是推理的依据和出发点。

对于模糊命题“若 A 则 B ”, L. A. Zadeh 在 1973 年利用模糊变换关系提出了下述近似推理的方法,称为模糊推理合成规则。

设 U 和 V 是两个各自具有基础变量 u, v 的论域, R 是在 $U \times V$ 论域上描述模糊蕴含“若 A 则 B ”的模糊关系,对于给定的 $A_1 \in U$, 可推得由 A_1 引出的结论 $B_1 \in V$ 为:

$$B_1 = A_1 \circ R$$

式中,“ \circ ”表示合成运算。

近似推理情况下的假言推理具有如下结构:

大前提:若 A 则 B

小前提:如今 A_1

结 论: $B_1 = A_1 \circ R$

这里 A 和 A_1, B 和 B_1 并不一致,如果一致的话,近似推理就退化为确定性推理。

在模糊推理合成规则中,有两个很重要的步骤:一个是求模糊蕴含关系 R ,另一个是模糊关系的合成运算。这里介绍比较常用的 Zadeh 和 Mamdani 模糊关系定义方法。

1) Zadeh 定义方法

模糊蕴含关系:

$$R = (A \times B) \cup (\bar{A} \times E) \quad (2.9)$$

式中, E 为全称矩阵。隶属函数为:

$$\mu_R(u, v) = (\mu_A(u)) \wedge (\mu_B(v)) \vee (1 - \mu_A(u)) \quad (2.10)$$

2) Mamdani 定义方法

模糊蕴含关系:

$$R = A \times B \quad (2.11)$$

隶属函数为:

$$\mu_R(u, v) = (\mu_A(u)) \wedge (\mu_B(v)) \quad (2.12)$$

模糊蕴含关系运算方法不同,其模糊推理有差异,但判断结论大体一致。

2.4.2 模糊条件推理

基于模糊推理合成规则,以下介绍三类模糊条件推理:

(1) 简单模糊条件推理

设 A 是论域 U 上的模糊集合, B 及 C 是论域 V 上的模糊集合, 则“ $\text{If } A \text{ Then } B \text{ Else } C$ ”在论域 $U \times V$ 上的模糊关系 R 为:

$$R = (A \times B) \cup (\bar{A} \times C) \quad (2.13)$$

根据推理合成规则, 可求得与已知模糊集合 A_1 对应的模糊集合 B_1 为:

$$B_1 = A_1 \circ R \quad (2.14)$$

所得模糊集合 B_1 便是在 $A = A_1$ 及“ $\text{If } A \text{ Then } B \text{ Else } C$ ”前提下得到的模糊条件推理结论。

在模糊控制中, 应用该模糊条件推理可计算出输入为 A_1 时具有“ $\text{If } A \text{ Then } B \text{ Else } C$ ”型控制规则的模糊控制器的输出 B_1 。

例 2.7 有论域 $X = \{a_1, a_2, a_3, a_4, a_5\}$ 及 $Y = \{b_1, b_2, b_3, b_4, b_5\}$, 并定义 $A = \text{轻} = 1/a_1 + 0.8/a_2 + 0.6/a_3 + 0.4/a_4 + 0.2/a_5$, $B = \text{重} = 0.2/b_1 + 0.4/b_2 + 0.6/b_3 + 0.8/b_4 + 1/b_5$, 试确定模糊条件语句“ $\text{If } x \text{ 轻 Then } y \text{ 重 Else } y \text{ 不非常重}$ ”所决定的模糊关系 R , 以及模糊语句“ x 非常轻”所对应的 y 。

解 ①根据语气算子和“补”运算, 求取下列模糊集合:

$$\text{非常重} = \overline{\text{重}} = 0.04/b_1 + 0.16/b_2 + 0.36/b_3 + 0.64/b_4 + 1/b_5$$

$$\text{不非常重} = \overline{\text{非常重}} = 0.96/b_1 + 0.84/b_2 + 0.64/b_3 + 0.36/b_4 + 0/b_5$$

$$\text{不轻} = \overline{\text{轻}} = 0/a_1 + 0.2/a_2 + 0.4/a_3 + 0.6/a_4 + 0.8/a_5$$

$$\text{非常轻} = \overline{\text{轻}}^2 = 1/a_1 + 0.64/a_2 + 0.36/a_3 + 0.16/a_4 + 0.04/a_5$$

②确定“ $\text{If } x \text{ 轻 Then } y \text{ 重 Else } y \text{ 不非常重}$ ”所决定的模糊关系 R

$$R = (A \times B) \cup (\bar{A} \times C) =$$

$$(\text{轻} \times \text{重}) \cup (\text{不轻} \times \text{不非常重}) =$$

$$\begin{bmatrix} 1 \\ 0.8 \\ 0.6 \\ 0.4 \\ 0.2 \end{bmatrix} \begin{bmatrix} 0.2 & 0.4 & 0.6 & 0.8 & 1 \end{bmatrix} \cup \begin{bmatrix} 0 \\ 0.2 \\ 0.4 \\ 0.6 \\ 0.8 \end{bmatrix} \begin{bmatrix} 0.96 & 0.84 & 0.64 & 0.36 & 0 \end{bmatrix} =$$

$$\begin{bmatrix} 0.2 & 0.4 & 0.6 & 0.8 & 1 \\ 0.2 & 0.4 & 0.6 & 0.8 & 0.8 \\ 0.4 & 0.4 & 0.6 & 0.6 & 0.6 \\ 0.6 & 0.6 & 0.6 & 0.4 & 0.4 \\ 0.8 & 0.8 & 0.64 & 0.36 & 0.2 \end{bmatrix}$$

③计算“ x 非常轻”所对应的模糊集, 即

$$A_1 = \text{非常轻} = 1/a_1 + 0.64/a_2 + 0.36/a_3 + 0.16/a_4 + 0.04/a_5$$

$$B_1 = A_1 \circ R =$$

$$\begin{bmatrix} 1 & 0.64 & 0.36 & 0.16 & 0.04 \end{bmatrix} \circ \begin{bmatrix} 0.2 & 0.4 & 0.6 & 0.8 & 1 \\ 0.2 & 0.4 & 0.6 & 0.8 & 0.8 \\ 0.4 & 0.4 & 0.6 & 0.6 & 0.6 \\ 0.6 & 0.6 & 0.6 & 0.4 & 0.4 \\ 0.8 & 0.8 & 0.64 & 0.36 & 0.2 \end{bmatrix} =$$

$$[0.36 \quad 0.4 \quad 0.6 \quad 0.8 \quad 1]$$

用 Zadeh 表示 $B_1 = [0.36 \quad 0.4 \quad 0.6 \quad 0.8 \quad 1]$, 将其与模糊集合“重”比较, 可得输出近似于“较重”的结论。

例 2.8 模糊条件语句为“*If x 轻 Then y 重 Else y 不非常重*”试问:

①若 x 是重时, y 如何?

②若 x 是极重时, y 又如何?

解 与上例同理, 利用上例的模糊关系矩阵 R , 可以推出:

①若 x 是重时

$$A_1 = \text{重} = 0.2/a_1 + 0.4/a_2 + 0.6/a_3 + 0.8/a_4 + 1/a_5$$

$$\text{则 } B_1 = A_1 \circ R = [0.2 \quad 0.4 \quad 0.6 \quad 0.8 \quad 1] \circ R = [0.8 \quad 0.8 \quad 0.64 \quad 0.6 \quad 0.6]$$

这表示此时 y 的模糊子集为: $B_1 = 0.8/b_1 + 0.8/b_2 + 0.64/b_3 + 0.6/b_4 + 0.6/b_5$, 即输出 y 近似于“不很重”。

②若 x 是极重时

$$A_1 = \text{极重} = \text{重}^4 = 0.0016/a_1 + 0.0256/a_2 + 0.1296/a_3 + 0.4096/a_4 + 1/a_5$$

$$\text{则 } B_1 = A_1 \circ R = [0.8 \quad 0.8 \quad 0.64 \quad 0.36 \quad 0.2]$$

此时, y 的模糊子集为: $B_1 = 0.8/b_1 + 0.8/b_2 + 0.64/b_3 + 0.36/b_4 + 0.2/b_5$, 即输出 y 近似于“较轻”。

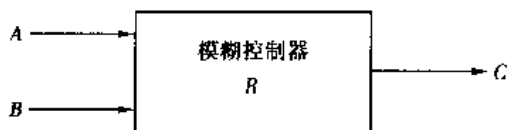


图 2.14 模糊控制器框图

(2) 多输入模糊条件推理

在模糊控制中, 常用到的是“*If A And B Then C*”条件语句, 如图 2.14 所示。这是因为在模糊控制中, 不但要考虑给定值与实际值之间的偏差, 还要考虑偏差的变化率, 下

面介绍两输入情况的模糊条件推理。

设 A, B, C 分别是论域 U, V, W 上的模糊集合, A, B 是模糊控制器的输入模糊集合, C 是输出模糊集合, 则“*If A And B Then C*”在论域 $U \times V \times W$ 上所决定的三元模糊关系 R 为:

$$R = (A \times B)^T \times C$$

式中, $(A \times B)^T$ 为由模糊关系矩阵 $(A \times B)_{n \times m}$ 构成的 $n \times m$ 维列向量, n 和 m 分别为模糊集合 A 与 B 的论域元素数。

根据推理合成规则, 可求得与已知模糊集合 $(A_1 \text{ And } B_1)$ 对应的模糊集合 C_1 为:

$$C_1 = (A_1 \times B_1)^T \circ R$$

这里, $(A_1 \times B_1)^T$ 为由模糊关系矩阵 $(A_1 \times B_1)_{n \times m}$ 构成的 $n \times m$ 维行向量。

所得模糊集合 C_1 便是在 $(A = A_1 \text{ And } B = B_1)$ 及“*If A And B Then C*”前提下, 得到的模糊条件推理结论。

例 2.9 设有论域: $U = \{a_1, a_2, a_3\}$, $V = \{b_1, b_2, b_3\}$, $W = \{c_1, c_2\}$

已知模糊集合: $A = 0.5/a_1 + 1/a_2 + 0.1/a_3$, $A \in U$

$B = 0.1/b_1 + 1/b_2 + 0.6/b_3$, $B \in V$

$C = 0.4/c_1 + 1/c_2$, $C \in W$

试确定模糊条件语句“*If A And B Then C*”所决定的模糊关系 R , 并计算由给定的输入模糊集合:

$$A_1 = 1/a_1 + 0.5/a_2 + 0.1/a_{32} + 1/a_3,$$

$$B_1 = 0.1/b_1 + 0.5/b_2 + 1/b_3$$

决定的输出模糊集合 C_1 .

解 ①确定模糊条件语句“If A And B Then C”所决定的模糊关系 R :

$$A \times B = \begin{bmatrix} 0.5 \\ 1 \\ 0.1 \end{bmatrix} [0.1 \quad 1 \quad 0.6] = \begin{bmatrix} 0.5 \wedge 0.1 & 0.5 \wedge 1 & 0.5 \wedge 0.6 \\ 1 \wedge 0.1 & 1 \wedge 1 & 1 \wedge 0.6 \\ 0.1 \wedge 0.1 & 0.1 \wedge 1 & 0.1 \wedge 0.6 \end{bmatrix} = \begin{bmatrix} 0.1 & 0.5 & 0.5 \\ 0.1 & 1 & 0.6 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}$$

$$(A \times B)^T = \begin{bmatrix} 0.1 \\ 0.5 \\ 0.5 \\ 0.1 \\ 1 \\ 0.6 \\ 0.1 \\ 0.1 \\ 0.1 \end{bmatrix}$$

$$R = (A \times B)^T \times C = \begin{bmatrix} 0.1 \\ 0.5 \\ 0.5 \\ 0.1 \\ 1 \\ 0.6 \\ 0.1 \\ 0.1 \\ 0.1 \end{bmatrix} [0.4 \quad 1] = \begin{bmatrix} 0.1 \wedge 0.4 & 0.1 \wedge 1 \\ 0.5 \wedge 0.4 & 0.5 \wedge 1 \\ 0.5 \wedge 0.4 & 0.5 \wedge 1 \\ 0.1 \wedge 0.4 & 0.1 \wedge 1 \\ 1 \wedge 0.4 & 1 \wedge 1 \\ 0.6 \wedge 0.4 & 0.6 \wedge 1 \\ 0.1 \wedge 0.4 & 0.1 \wedge 1 \\ 0.1 \wedge 0.4 & 0.1 \wedge 1 \\ 0.1 \wedge 0.4 & 0.1 \wedge 1 \end{bmatrix} = \begin{bmatrix} 0.1 & 0.1 \\ 0.4 & 0.5 \\ 0.4 & 0.5 \\ 0.1 & 0.1 \\ 0.4 & 1 \\ 0.4 & 0.6 \\ 0.1 & 0.1 \\ 0.1 & 0.1 \\ 0.1 & 0.1 \end{bmatrix}$$

②根据推理合成规则,求取与输入模糊集合 A_1, B_1 对应的输出模糊集合 C_1 :

$$A_1 \times B_1 = \begin{bmatrix} 1 \\ 0.5 \\ 0.1 \end{bmatrix} [0.1 \quad 0.5 \quad 1] = \begin{bmatrix} 0.1 & 0.5 & 1 \\ 0.1 & 0.5 & 0.1 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}$$

$$(A_1 \times B_1)^T = [0.1 \quad 0.5 \quad 1 \quad 0.1 \quad 0.5 \quad 0.5 \quad 0.1 \quad 0.1 \quad 0.1]$$

$$C_1 = (A_1 \times B_1)^T \circ R =$$

$$[0.1 \quad 0.5 \quad 1 \quad 0.1 \quad 0.5 \quad 0.5 \quad 0.1 \quad 0.1 \quad 0.1] \begin{bmatrix} 0.1 & 0.1 \\ 0.4 & 0.5 \\ 0.4 & 0.5 \\ 0.1 & 0.1 \\ 0.4 & 1 \\ 0.4 & 0.6 \\ 0.1 & 0.1 \\ 0.1 & 0.1 \\ 0.1 & 0.1 \end{bmatrix} = [0.4 \quad 0.5]$$

注意:当模糊集合 A 和 B 的论域元素相同时,还可用下列关系式计算:

$$R = (A \times C) \cap (B \times C)$$

$$C_1 = (A_1 \circ R) \cap (B_1 \circ R)$$

两输入的情况很容易推广到多输入的情况,请参阅相关的文献。

(3) 多输入多规则模糊推理

以两输入多规则情况介绍。若有 n 条规则,其一般形式为:

If A_1 And B_1 Then C_1

Else If A_2 And B_2 Then C_2

Else If A_3 And B_3 Then C_3

.....

Else If A_n And B_n Then C_n

每一条规则 i 都对应了 $U \times V \times W$ 论域上的一个模糊关系 R_i ,这 n 条规则是“或”的关系,总的规则对应的模糊关系 R 就是 n 条规则对应的模糊关系 R_i 的“并”。

$$R = \bigcup_{i=1}^n R_i$$

在“ A' And B' ”输入下,推理结果为:

$$C' = (A' \times B')^{T2} \circ R$$

还有一些更复杂的模糊条件推理,在此略。

第 3 章

模糊控制技术基础

3.1 模糊控制基本原理

3.1.1 模糊控制的引入及实现

人类在认识世界和改造世界的过程中,在控制领域内经历了从手动控制到自动控制的过程。手动控制首先是通过观测被控对象的输出,并根据观测结果经大脑思考后做出决策,然后手动调节输入,操作人员就是通过不断地“观测——决策——调整”,实现对生产过程的手动控制。随着科学技术的进步和发展,人们应用自动化仪表或控制装置代替人工自动地对机器设备或生产过程进行控制,并经过长期研究和实践形成了经典控制理论。

经典控制理论对于解决线性定常系统的控制问题是很有效的,然而,对于非线性时变系统却难以奏效。随着计算机的应用和发展,自动控制理论和技术取得了飞跃的发展,基于状态变量描述的现代控制理论对于解决线性或非线性、定常或时变的多输入与多输出系统的控制问题,获得了广泛和成功的应用。但是,无论采用经典控制理论还是现代控制理论设计的控制系统,都需要事先知道被控对象(或过程)精确的数学模型,然后根据数学模型以及给定的性能指标,选择适当的控制规律,进行控制系统设计。然而,在许多情况下,被控对象的精确数学模型很难建立,这样一来,对于这类对象或过程就很难进行自动控制。

与此相反,对于上述难以自动控制的一些生产过程,由有经验的操作人员进行手动控制,却往往可以收到令人满意的控制效果。实际上,在生产过程中,由于种种原因,对某些设备或生产环节至今仍必须由人的手动控制来完成。在这样的事实面前,人们又重新研究和考虑人的手动控制行为的特点,探讨对于无法构造数学模型的对象能否让计算机模拟人的思维方式,模仿人的手动控制策略进行控制决策。

人的手动控制策略是通过操作者的长期实践经验积累而形成的,控制论的创始人维纳在研究人与外界相互作用的关系时曾指出:“人通过感觉器官感知周围世界,在脑和神经系统中调整获得的信息,经过适当的存储、校正、归纳和选择等过程而进入效应器官反作用于外部世

界,同时也通过像运动传感器末梢这类传感器再作用于中枢神经系统,将新接受的信息与原储存的信息结合在一起,影响并指挥将来的行动”。总结人的控制行为,正是遵循反馈及反馈控制的思想,其控制策略可以通过人的自然语言来描述,将它们总结成一系列条件语句,即控制规则。因此,这种控制属于一种语言控制,运用微机来实现这些控制规则,微机就起到了控制器的作用。然而,人的自然语言又具有模糊性,而传统自动控制恰恰无法处理这种模糊性。模糊集合和模糊逻辑的出现实时地解决了描述控制规则的条件语句中如“较大”、“稍小”、“偏高”等具有一定模糊性的词语,用模糊集合来描述这些模糊条件语句,即组成了所谓的模糊控制器。1974年 Mamdani 首先设计了模糊控制器,并用于锅炉和蒸汽机的控制,取得了巨大的成功,模糊控制就此开始诞生了。

模糊控制与经典控制的根本区别在于它并不需要建立被控对象(或过程)的精确数学模型,而是完全凭人的知识和经验,应用了人的思维和逻辑推理方法来“直观”地进行控制。与一般 PID 控制在原理与方法上完全不同,模糊控制通过模糊逻辑和近似推理方法,把人的知识和经验形式化、模型化,变成计算机可以接受的控制模型,从而让计算机代替人来进行有效的实时控制。

3.1.2 模糊控制过程

(1) 模糊控制基本思想

以人类自然语言变量的概念作为描述手动控制策略的基础来实现模糊控制,必须解决三个基本问题:第一,先通过传感器把要监测的物理量变成电量,再通过模数转换器把它转换成精确的数字量,精确输入量输入至模糊控制器后,要把这精确的输入量转换成模糊集合的隶属函数,这一步称为精确量的模糊化,其目的是把传感器的输入转换成知识库可以理解和操作的变量格式;第二,根据有经验的操作者或者专家的经验制定出模糊控制规则,并进行模糊逻辑推理,其目的是得到一个模糊输出集合,即一个新的模糊集合隶属函数,这一步称为模糊控制规则形成和推理,其目的是用模糊输入值去适配控制规则,为每个控制规则确定其适配的程度,并且通过加权计算合并那些规则的输出,得到输出控制模糊集合;第三,根据模糊逻辑推理得到的输出模糊集合隶属函数,用恰当的方法找一个具有代表性的精确值作为控制量,这一步称为模糊输出量的解模糊判决,其目的是把分布范围概括合并成单点的输出值,加到执行器上实现控制。

模糊控制的基本原理可由图 3.1 表示,它的核心部分为模糊控制器,如图中的虚线框中部分所示。模糊控制器的控制规律由计算机的程序实现,模糊控制的基本思想是:微机经中断采样获取被控制量的精确值,然后将此量与给定值比较得到偏差信号 e 。一般选偏差信号 e 作为模糊控制器的一个输入量,把偏差信号 e 的精确量进行模糊化变成模糊量,偏差 e 的模糊量可以用相应的模糊语言表示,得到了偏差 e 的模糊语言集合的一个子集。再由模糊子集和模糊控制规则(模糊关系)根据模糊推理的合成规则进行模糊决策,得到模糊控制量为:

$$\underline{u} = \underline{e} \circ \underline{R} \quad (3.1)$$

式中, \underline{u} 为一个模糊量。

为了对被控对象施加精确的控制,还需要将模糊量 \underline{u} 转换为精确量,这一步骤在图 3.1 框图中称为解模糊(也称清晰化)。得到了精确的数字控制量后,经数模转换变为精确的模拟量

送给执行机构,对被控对象进行一步控制。然后,中断等待第二次采样,进行第二步控制,这样循环下去,就实现了被控对象的模糊控制。

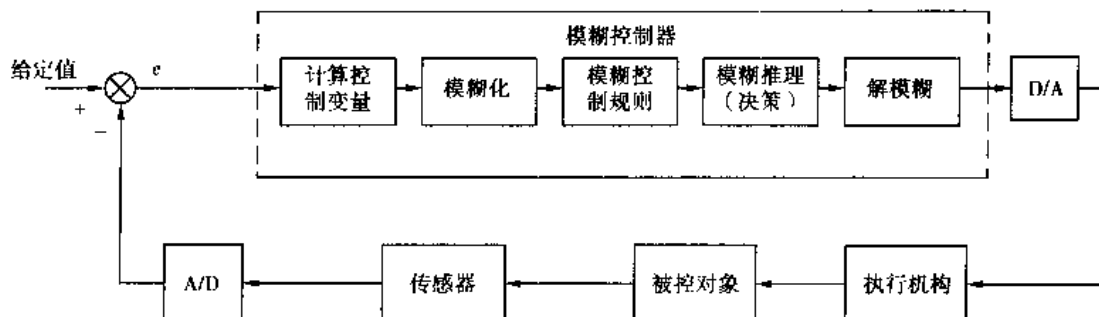


图 3.1 模糊控制原理框图

综上所述,模糊控制过程可概括为下述4个步骤:

- ①根据本次采样得到的系统的输出值,计算所选择系统的输入变量;
- ②将输入变量的精确值变为模糊量;
- ③根据输入变量(模糊量)及模糊控制规则,按照模糊推理合成规则推理计算输出控制量(模糊量);
- ④由上述得到的控制量(模糊量)计算精确的输出控制量,并作用于执行机构。

(2) 模糊控制系统的工作原理

为了说明模糊控制系统的工作原理,以下介绍一个简单的单输入单输出温控系统:

例如,某温度控制系统采用电热炉进行温度控制,按该生产工艺要求需保持炉温 800°C 恒定不变。

设电热炉的供电电压是经可控硅整流电源提供的,它的电压连续可调。当调整可控硅触发电路中的偏置电压,即改变可控硅导通角 α 时,可控硅整流电源的电压相应得到调整,即供电电压可根据需要连续可调。

当用人工操作方式进行温度控制时,根据对炉温的观测,手动调节电位器旋钮即可调节电热炉供电电压,达到升温或降温的目的。

该温度控制系统基本控制规则可以用语言描述如下:

- ①若炉温低于 800°C ,则升压,低得越多,升压越高;
- ②若炉温高于 800°C ,则降压,高得越多,降压越低;
- ③若炉温等于 800°C ,则保持电压不变。

对于上述温度控制系统,采用模糊控制方法进行炉温控制时,控制系统的工作原理及工作过程可简述如下:

1) 确定模糊控制器的输入变量和输出变量

在此将炉温 800°C 作为给定值 t_0 ,测量得到的炉温记为 $t(K)$,则误差

$$e(K) = t_0 - t(K) \quad (3.2)$$

作为模糊控制器的输入变量。

模糊控制器的输出变量是触发电压 U 的变化,该电压直接控制电热炉供电电压的高低,所以,又称输出变量为控制量。

2) 输入变量及输出变量的模糊语言描述

设描述输入变量及输出变量的语言值的模糊子集为:

{负大,负小,零,正小,正大}

采用如下简记形式:

NB = 负大, NS = 负小, Z = 零, PS = 正小, PB = 正大。

设误差 e 的论域为 X , 并将误差大小量化为 7 个等级, 分别表示为 $-3, -2, -1, 0, +1, +2, +3$, 则有:

$X = \{-3, -2, -1, 0, +1, +2, +3\}$

设执行控制量 u 的论域为 Y , 并同 X 一样也把控制量的大小划分为 7 个等级(也可以多于 7 个), 即:

$Y = \{-3, -2, -1, 0, +1, +2, +3\}$

3) 模糊控制规则的语言描述

根据手动控制策略, 模糊控制规则可归纳如下:

- ① if $e = \text{NB}$ then $u = \text{PB}$
- ② if $e = \text{NS}$ then $u = \text{PS}$
- ③ if $e = \text{Z}$ then $u = \text{Z}$
- ④ if $e = \text{PS}$ then $u = \text{NS}$
- ⑤ if $e = \text{PB}$ then $u = \text{NB}$

为简化起见, 也可以用表格形式描述控制规则, 表 3.1 即为上述的控制规则的表格化, 也称为控制规则表。

表 3.1 控制规则表

e	NB	NS	Z	PS	PB
u	PB	PS	Z	NS	NB

4) 模糊控制规则的矩阵形式

模糊控制规则实际上是一组多重条件语句, 它可以表示为从误差论域 X 到控制量论域 Y 的模糊关系 R 。因为当论域是有限时, 模糊关系可以用矩阵来表示, 而论域 X 及 Y 均是有限的(由于将精确量离散化时, 将其分成有限的几档, 如在此为七档, 每一档对应一个模糊集, 这样可使问题处理简化), 所以模糊关系可以用矩阵表示, 以便于推理和运算。

5) 模糊决策

模糊控制器的控制作用取决于控制量, 而控制量通过式(3.1)进行计算, 即

$$\underline{u} = \underline{e} \circ \underline{R}$$

控制量 \underline{u} 实际上等于误差的模糊向量 \underline{e} 和模糊关系 \underline{R} 的合成。

6) 控制量的模糊量转化为精确量

上面求得的控制量 \underline{u} 为一模糊向量, 设为:

$$\underline{u} = (0.3/-3) + (0.5/-2) + (1/-1) + (0.5/0) + (0.4/1) + (0/2) + (0/3)$$

对于控制量的模糊子集按照最大隶属度原则, 应选取控制量为“-1”级。即当误差 $e = \text{PS}$ 时, 控制量 $\underline{u} = \text{NS}$, 为“-1”级, 具体地说, 当炉温偏高时, 应降低一点电压。

实际控制时, “NS(-1)”级电压要变为精确量。“-1”这个等级控制电压的精确值根据

事先确定的范围是很容易计算得出的。通过这个精确量去控制电热炉的电压,使得炉温朝着减小误差的方向变化。

上述温度控制系统所采用的模糊控制器,是选用偏差作为一个输入变量的单输入模糊控制器,对于这样的模糊控制器,它的控制性能显然还不能令人满意。之所以举这样的例子,目的在于便于从一个最简单的模糊控制器来说明模糊控制系统的基本工作原理,为深入研究更复杂、更高级的模糊控制器奠定基础。

3.2 模糊化方法(Fuzzification)

在模糊控制应用中,被观测量通常是确定的量,即测量得到的该确定量是在一定精度范围内的精确的数值量,由于在模糊控制中的操作是基于模糊集合理论的,因此,首先必须进行模糊化。

模糊化的基本思想是定义一个模糊语言映射作为从数值域至语言域(符号域)的模糊关系,从而在数值测量的基础上,将数值域中的数值信号映射到语言域上,为实现模糊推理奠定基础。

模糊化与自然语言的含糊和不精确相联系,这是一种主观评价。把测量值(数值量)转换为主观量值(模糊量)的过程称为模糊化,也即把物理量的精确值转换成语言变量值。由此,它可以定义为在确定的输入论域中将所观测的输入空间转换为模糊集的映射,以便实现模糊控制算法。

模糊概念从本质上来说就是语言变量的语言值,而语言值多用模糊集合描述,模糊集合一般由论域和隶属函数构成。因此,模糊化的实质就是求取相应概念对应数值域的模糊集合隶属函数。

为了便于工程实现,通常把输入变量范围人为地定义成离散的若干级,所定义级数的多少取决于所需输入量的分辨率。定义输入量的隶属函数可采用吊钟型、梯形和三角形,理论上吊钟型最为理想,但是计算复杂;实践证明:用三角形和梯形函数其性能并没有十分明显的差别。为了简化计算,现在最常用的是三角形,其次是梯形隶属函数。

为了实现模糊控制器的标准化设计,目前在实际中常用的处理方法是 Mamdani 提出的方法,这就是把偏差 e 的变化范围设定为 $[-6, +6]$ 区间内的连续变化量,并使之离散化,构成含 13 个整数元素的离散集合:

$$\{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6\}$$

实际上,如果是非对称型的,也可用 1~13 取代 $-6 \sim +6$ 。

如果精确量 x 的实际变化范围为 $[a, b]$, 将 $[a, b]$ 区间的精确变量转换为 $[-6, +6]$ 区间变化的变量 y , 采用如下公式:

$$y = \frac{12}{b-a} \left[x - \frac{a+b}{2} \right] \quad (3.3)$$

将在 $[a, b]$ 之间变化的变量 x 转换为在 $[-6, +6]$ 之间变化的变量 y , 由上式计算的 y 值如果不是整数,可以把它归入最接近于 y 的整数。例如: $-4.8 \rightarrow -5$, $2.7 \rightarrow 3$, $-0.4 \rightarrow 0$ 。

模糊化是实现模糊控制的一个重要环节,但对其方法的研究目前仍很不完善,还有待于进

一步研究,目前常用的方法主要有以下几种:

(1) 线性划分法

这是最为简单的一种方法,根据研究对象的具体情况,选定相应的自然语言描述符号后,将研究对象的论域均匀划分。以温度测量为例,测温范围为 $-200 \sim 1\,000^{\circ}\text{C}$,以对称的三角函数作为模糊集合的隶属函数,模糊化结果如图 3.2 所示。

这种方法对于在模糊控制系统中人类在一定范围内不能直接感知的某些被测的量,如高炉炼钢的温度测量,研究超导现象的低温测量等应用具有简单、实用的特点。当然,隶属函数选择三角形也是出于简单考虑,此外,钟形、梯形等也是常用的隶属函数。

(2) 非线性划分法

这种方法主要应用于采用了非线性敏感元件(如热敏电阻等)的模糊控制系统的模糊化,如图 3.3 所示。

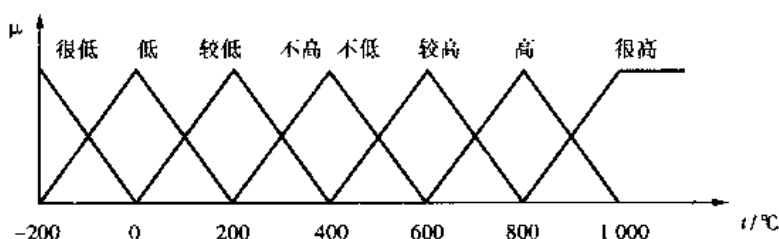


图 3.2 模糊化的线性划分法

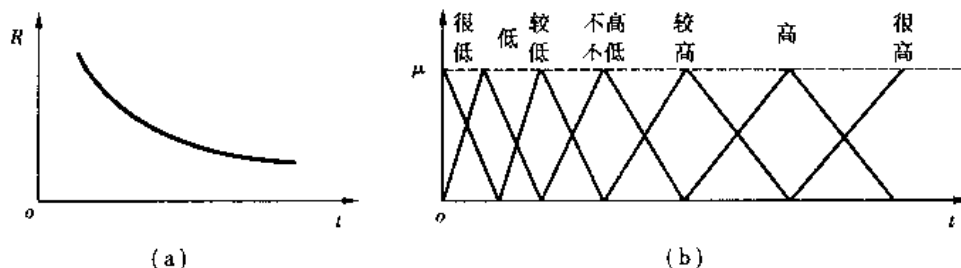


图 3.3 模糊化的非线性划分法

由于许多敏感元件(如热敏电阻等)具有较大的非线性(图 3.3(a)),因而在模糊化的同时,通过采用非线性隶属函数的划分(图 3.3(b)),可以同时进行测量的非线性校正。

(3) 语义关系生成法

模糊量的语义值(如“热”与“很热”之间的语义关系)是根据相应的数值关系来确定的。为此,E. Benoit 等人提出了基于语义关系的模糊化方法,其基本思想是:首先定义一基础概念(Genic concept)及其相应的隶属函数,然后通过语义算子的作用,产生具有相关语义的新概念及其隶属函数,常用的语义算子主要有:

$CI = \text{very}(C)$	$\mu_{CI}(x) = \mu_C^2(x)$
$CI = \text{not}(C)$	$\mu_{CI}(x) = 1 - \mu_C(x)$
$CI = \text{more or less}(C)$	$\mu_{CI}(x) = \mu_C^{\frac{1}{2}}(x)$
$CI = \text{more than}(C)$	$\mu_{CI}(x) = \mu_C(x) * \sigma(x - \Delta)$
$CI = \text{less than}(C)$	$\mu_{CI}(x) = \mu_C(x) * \sigma(x + \Delta)$
$CI = \text{above}(C)$	$\mu_{CI}(x) = \begin{cases} 0, & x < m \\ 1 - \mu_C(x), & x \geq m \end{cases}$

$$CI = \text{below}(C) \quad \mu_{ci}(x) = \begin{cases} 1 - \mu_c(x), & x < m \\ 0, & x \geq m \end{cases}$$

式中, C 为基础概念, CI 为施加了一定语义关系作用后的新概念, 只要知道基础概念“ C ”的隶属函数, 通过上述语义算子的作用, 可以自动生成相关语义概念的隶属函数。

(4) 训练法

由于人类自身对事物概念的形成和产生方法主要来源于学习, 为此, 人们提出了模糊化的训练算法。训练算法的出发点是: 根据不同的测量结构和应用场合, 由专家选定具体的论域上表述概念的个数, 然后通过经验算法自动生成相应概念的隶属函数特性曲线, 最后, 专家依据经验对其中的一个或几个概念进行训练, 以适合不同的测量要求。其基本思想是: 对于训练样本(包括论域内若干个测量点上的状态数据以及相应隶属于人类经验的被测量, 用自然语言符号描述的状态符号), 在当前概念模式下, 根据最大隶属度准则判定, 若数据状态与概念状态相一致, 则训练结束; 若不相符, 则将相应概念隶属函数曲线的修正率加以改变, 以实现符合专家经验的被测量数据状态与符号状态的一致。

训练算法能简单有效地实现从被测量到自然语言符号描述的模糊映射, 只需给定论域的范围, 就可在论域上自动生成任意有限个语言概念的隶属函数。经训练后, 可将该领域专家经验和知识集成, 作为后验知识提供给模糊控制器, 实现对被测量的自然语言符号描述。这对于实现人类对被测信息的宏观理解和感知具有较强的实用价值, 是较为实用的一种模糊化方法。

总之, 尽管上面介绍的各种模糊化方法都还是比较粗略的, 同时受人为主观因素影响也较大, 但是人脑在进行这一转化过程时同样也是不精确的, 这从某种意义上来说, 也正是模糊控制之所以能模仿专家经验的特点所在。

3.3 解模糊方法(Defuzzification)

通过模糊推理得到的结果是一个模糊集合, 也可以说是一个模糊量; 但实际模糊控制系统所需要的控制信号必须是精确量, 要用一个确定的值才能去控制执行机构。因此, 模糊控制器经过推理后的模糊输出量必须经过精确化处理, 才能去控制被控对象。

在模糊推理得到的模糊量中, 求取一个相对最能代表这个模糊集合的单值的过程称为解模糊。解模糊的方法很多, 常用的主要有如下几种:

3.3.1 重心法

重心法, 也称力矩法。它取推理结论模糊集合隶属函数曲线与横坐标轴所围成面积的重心作为代表点, 即

$$u = \frac{\int x \mu_v(x) dx}{\int \mu_v(x) dx} \quad (3.4)$$

当输出变量的隶属函数为离散单点集时, 则为:

$$u = \frac{\sum x_i \cdot \mu_v(x_i)}{\sum \mu_v(x_i)} \quad (3.5)$$

重心法的实质为加权平均法,权值为推理结论模糊集合中各元素的隶属度。

3.3.2 最大隶属度法

最大隶属度法是指在推理结论的模糊集合中选取隶属度最大的元素作为精确控制量的方法。如果论域上多个元素同时出现最大隶属度值,则取它们的平均值作为解模糊判决结果。设存在模糊集 C ,所选择的隶属度最大的元素 u^* 应满足:

$$\mu_C(u^*) \geq \mu_C(u), u \in U \quad (3.6)$$

如果该模糊集合隶属函数曲线是梯形平顶的,那么具有最大隶属度的元素就可能不止一个,这时就要对所有取最大隶属度的元素求取平均值。

在最大隶属度法中,有时还要采用一些特殊的规则,即左边最大隶属度法,或者右边最大隶属度法。

所谓左边最大隶属度法,实质是把几个最大隶属度中的最小元素作为解模糊的精确值;而所谓右边最大隶属度法,实质是把几个最大隶属度中的最大元素作为解模糊后的精确值。

3.3.3 系数加权平均法

系数加权平均法是指输出量模糊集合中各元素进行加权平均后的输出值作为输出执行量,其值为:

$$u = \frac{\int xk(x)}{\int k(x)} \quad (3.7)$$

当输出变量为离散单点集时,则为:

$$u = \frac{\sum k_i \cdot x_i}{\sum k_i} \quad (3.8)$$

这里权系数 $k(x)$ 、 k_i 的选择要根据实际情况确定,不同的权系数决定有不同的响应特性。当该系数选择 $k(x) = \mu_v(x)$ 或 $k_i = \mu_v(x_i)$ (即取其隶属度)时,就是前边所说的重心法。

3.3.4 隶属度限幅元素平均法

用所确定的隶属度值 $\alpha, \alpha \in [0,1]$,对推理结论模糊集合隶属函数曲线进行切割,再对切割后等于该隶属度的所有元素进行平均,用这个平均值作为输出执行量,称为隶属度限幅元素平均法。例如,当 α 取最大隶属度 1 时,表示“完全隶属”关系;当 α 取 0.5 时,表示“大概隶属”关系。此时,相应解模糊的结果分别为“完全隶属”关系下的元素平均值和“大概隶属”关系下元素的平均值。

3.3.5 中位数法

中位数法是全面考虑推理结论模糊集合各部分信息作用的一种方法,即把隶属函数曲线与横坐标所围成的面积分成两部分,在两部分相等的条件下,将两部分分界点所对应的论域元素作为判决结果。这种方法可以充分地利用输出模糊集合所包含的信息。

设模糊推理的输出为模糊量 C ,如果存在 u^* ,并且使:

$$\sum_{u_{\min}}^{u^*} \mu_{\bar{C}}(u) = \sum_{u^*}^{u_{\max}} \mu_{\bar{C}}(u) \quad (3.9)$$

则取 u^* 为解模糊后所得的精确值。

用上述各种解模糊判决方法进行解模糊判决,对于同一个推理结论的模糊集合而言,显然会得到不同的结果。

不同解模糊判决方法的性能可以比较如下:

①重心法不仅有公式可循,而且在理论上比较合理,它涵盖和利用了模糊集合的所有信息,并根据隶属度的不同而有所侧重,但计算复杂,主要用于理论推导和实时性要求不高的场合。

②最大隶属度法的明显优点是简单易行,使用方便,算法实时性好;但是,它的一个明显的缺点是:仅仅利用了最大隶属度的信息,忽略了较小隶属度元素的影响和作用,输出信息量利用的太少,代表性不好。这种方法常用于简单控制系统。

③系数加权平均法可以通过选择和调整权系数大小来改善系统的响应特性。这种方法灵活性较大,但权系数选择需要根据实际操作经验和实验观测来反复进行调整,才能取得较好的控制效果。

④隶属度限幅元素平均法则是介于上述几种方法的一种折中手段。

⑤中位数法虽然比较充分地利用了模糊集合提供的信息量,考虑了所有信息的作用;但是,它的计算过程较为麻烦,而且缺乏对隶属度较大元素提供主导信息的充分重视,这是这种方法的不足之处。因此,中位数法虽然是比较全面的解模糊方法,但在实际的控制系统中应用并不普遍。

总之,在上面提到的各种不同的解模糊判决方法中,如果考虑要充分地利利用模糊推理结果中模糊集合提供的有用信息量,就会导致计算繁琐,否则就会或多或少丢掉一些有用信息。因此,要根据实际系统的具体情况,如系统的复杂程度以及控制精度等,适当地确定模糊推理结论中模糊量的解模糊方法。

3.4 模糊控制规则及控制算法

模糊控制是建立在一系列模糊控制规则的基础上的,这些控制规则是人们对被控对象进行控制时的经验总结。因此,这些控制规则是一些逻辑推理规则,其形式表现为模糊条件语句。在实际控制中,把有关控制规则加以处理,产生相应的控制算法,模糊控制器就以相应的算法去控制被控对象工作。

3.4.1 模糊控制规则表示及特性

模糊控制规则是由自然语言变量所表达的模糊条件语句组成,它表征了控制目标和该领域专家的控制策略,是根据被控对象的行为特性和专家的控制经验编写的,即是吸取了该专家的知识 and 经验,以“如果…,那么…”形式所表达的模糊推理语句。

在模糊控制系统中,控制规则一般表示为“if…then…”的形式。设被控对象的状态或输出可用 X_1, X_2 两个变量描述,变量的个数可以推广到任意多个。 Y 为控制器的输出,则下式是表

示控制规则的一个典型例子:

if X_1 is 大 and X_2 is 小 then Y is 中

仿照蕴涵式的称谓“ X_1 is 大 and X_2 is 小”称为控制规则的前件部,“ Y is 中”称为控制规则的后件部。“大”、“小”、“中”等均是对某一物理量的模糊化的自然语言描述,即它们均被描述为一个模糊集合。一般而言,模糊集合并不要求必须有一个自然语言的称谓,但模糊控制是一种基于人的思维模式的控制,因此,在模糊控制规则中出现的模糊集合,往往具有可以用自然语言描述的意义。

用于描述人们控制经验的基本语句结构有三种形式,它们分别反映了三种基本的推理。这一种基本结构和形式如下:

① $(A \rightarrow B)$ 结构

这种推理是一种最简单的蕴涵关系,在语言表达时表示为“如果 A , 那么 B ”, 即有:

if A then B

② $(A \rightarrow B) \vee (A' \rightarrow C)$ 结构

这种推理较之前一种复杂,这种蕴涵关系在用语言表达时叙述为“如果 A , 那么 B ; 否则 C ”, 即有:

if A then B , else C

③ $(A \wedge B \rightarrow C)$ 结构

这种推理的前件有两个,这种蕴涵关系在用语言表达时为“如果 A 而且 B 那么 C ”, 即有:

if A and B , then C

模糊控制规则应具备如下特性:

(1) 完备性

通过设计经验和工程知识,使模糊控制规则具有完备性。所谓完备性,是指对于任意给定的输入,均有相应的控制作用。要求控制规则的完备性是保证系统能被控制的必要条件之一。通过它能为每个过程状态推理,得到一个合适的控制作用。

模糊控制算法与数据和规则有关,控制规则直接影响到完备性。控制规则的完备性好,则控制算法的完整性就好,如果完备性不足,就可能产生控制“盲区”。因此,在模糊控制规则设计时,必须进行完备性检验。

(2) 一致性

控制规则的一致性是指控制规则中不存在相互矛盾的规则。如果两条规则的条件部分相同,但结论部分相差很大,则称两条规则相互矛盾。在实际中,应避免相互矛盾的规则出现。

模糊控制规则的生成基于的是专家控制经验,而控制规则要服从不同的系统性能指标。控制规则的一致性水平反映了控制规则的目标准则是否相同或接近,如果控制规则的一致性不足,则往往会产生相互矛盾的控制作用,从而影响控制质量,通过对控制规则的聚类分析,可提高控制规则的一致性。

(3) 交互作用性

如果控制器的输出值总由数条控制规则来决定,说明控制规则之间是相互联系、相互影响的,这就是控制规则之间的交互性,可以产生复杂的控制曲面,得到更好的控制性能。

规则之间可交互作用,它是通过选择模糊蕴涵和 Sup-Star 合成来控制的。模糊控制的交

互问题比较复杂,并且目前还不十分清楚,现仅知道选择恰当的模糊蕴涵关系算法和合成算法可以控制交互性。

3.4.2 模糊控制规则的生成

模糊控制规则是对系统控制经验的总结,是设计模糊控制器的重要依据,它直接影响着控制系统的质量。因此,准确产生一套合理、科学的模糊控制规则至关重要。在设计模糊控制规则时,必须考虑控制规则的完备性、交互性和一致性。

模糊控制规则的生成目前主要有四种方法:

(1) 经验归纳法

模糊控制规则是基于手动控制策略而建立的,而手动控制策略又是人们通过学习、实验以及长期经验积累而逐渐形成的,存储在操作者或专家中的一种技术知识集合。手动控制过程一般是通过对被控对象或过程的观测,操作者再根据已有的经验和技术知识,进行综合分析并做出控制决策,调整对被控对象的控制作用,从而使系统达到预期目标。

手动控制的作用与自动控制中控制器的作用是基本相同的,所不同的是手动控制决策是基于操作者经验和技术知识,而控制器的控制决策是基于某种控制算法的数值运算。

经验归纳法是总结、归纳专家的控制经验,经过进一步加工、整理、提炼,去粗取精后产生模糊控制规则的一种方法。这是一种最直接和方便的生成控制规则的方法。由于专家对一个被控对象有丰富的经验,因而可以用语言直接给出一套控制规则,只要把这些语句用条件语句表示即可。

在由专家经验生成控制规则时,只需完成以下有关工作:

- ①确定语言值的论域;
- ②确定语言值;
- ③根据每条专家经验产生对应的控制规则。

专家经验是用语言和直觉推理描述一个控制过程的。例如,对于炉温加热系统,一个专家可以提供以下由语言表述的操作经验:

若炉温过低,则加热电压调到最大挡;
 若炉温恰当,则加热电压保持不变;
 若炉温稍高,则加热电压调到中挡;
 若炉温过高,则加热电压调到最小挡;
 若炉温过高,且炉温快速上升,则停止加热;

这种自然语言描述的控制经验可以用“if...then...”这种模糊条件语句来表示。

模糊控制器最常用的结构为二维模糊控制器,它的输入语言变量一般取偏差 e 和偏差变化 Δe ,输出语言变量则为控制量的变化 u 。对于这种结构的模糊控制器,常采用 Mamdani 控制规则。其中,偏差语言变量 e 、偏差变化语言变量 Δe 以及控制量变化语言变量 u 的模糊论域要分成若干等级,例如, $-3, -2, -1, 0, 1, 2, 3$ 等;这些语言变量的语言值又必须分成若干档,如 $\{NB, NS, Z, PS, PB\}$, 这样,上述模糊条件语句可以写成:

if $e = PB$ and $\Delta e = NB$ then $u = Z$
 if $e = PB$ and $\Delta e = NS$ then $u = NM$

.....

对于其他经验,也可作类似的处理,可以得到如表 3.2 所示的模糊控制规则表。

表 3.2 模糊控制规则表

Δe	e				
	NB	NS	Z	PS	PB
PB	PB	NS	NB		NB
PS		Z	NS	NS	
Z		PS	Z		
NS			PS	Z	
NB		PB		PS	

用表 3.2 的控制规则进行推理,可以得出模糊控制器语言规则的输入、输出关系,这是一个关系曲面,它反映了输入和输出之间的非线性关系。当偏差较大时,控制量的变化应尽力使偏差迅速减小;当偏差较小时,除了要消除偏差外,还要考虑系统的稳定性,防止系统出现过大的超调,甚至引起系统振荡。

(2) 根据过程的模糊模型生成控制规则

如果用语言来描述一个被控过程的动态特性,那么这种语言描述可以看做过程的模糊模型。根据模糊模型可以得到模糊控制规则集,这个模糊控制规则集就形成了模糊控制器的规则。不过这种研究目前还处于起步阶段,尚未有一套完善的方法,很多问题还有待进一步改进和完善。

根据过程的模糊模型生成控制规则步骤如下:

1) 对熟练操作人员的经验进行描述

熟练操作人员在对系统控制的操作中,要不断根据控制目标和系统的状态情况进行决策,选择出认为合适的、相对最小的控制指令,同时还要有遇到突发事件如何进行处理的思想准备。这就需要对可能采用的控制指令及其控制结果进行语言描述,并作为定性的经验法则记录下来。

2) 对经验法则中描述程度的词进行模糊定量

在语言描述的经验法则中,常会用到一些如“好”、“较好”、“很大”、“较小”、“高”、“低”等描述程度的词来描述系统的控制指标和系统的状态,这时就需要把它们用模糊集合进行模糊化处理。

3) 建立被控系统的部分模型

选择有代表性的、可能实现的控制指令,根据被控对象的特性行为预测系统性能指标的变化,据此建立起系统的部分模型,这种模型可能是数学模型,也可能是定性的模糊模型。

4) 把经验法则转换成模糊控制规则

把用一般语言描述的经验法则形式化,设法将它们写成计算机语言能够表达的符号化的模糊控制规则。

5) 对模糊控制规则和状态进行调整

根据系统的实际情况,通过实验(或仿真)对原有模糊控制规则及模糊集的隶属函数进行

修改、调整和补充。

通过上述几个步骤,可以根据过程的模糊模型生成模糊控制规则。

(3) 根据对手工操作系统的观察和测量生成控制规则

在实际生产中,现场控制专家和熟练操作人员的手工操作可以很好地控制系统,但是,要把其控制经验和诀窍用逻辑形式表达出来往往却不那么容易。这是因为,这种系统既无法采用传统的控制理论进行模型的求取,也不能直观地用条件语句来表示经验。因此,操作人员是难于给出一套控制规则的。另一方面,由于操作人员的资质和学术水平问题,凭感觉和技巧可以控制好被控系统的操作人员,不一定能给出反映其知识和经验的控制规则,即所谓难以实现对“技巧”的表达。

为此,可以对系统的输入输出数据进行测量,根据系统的测量数据生成控制规则。

首先制定对系统控制的输入数据,再用这些输入数据对系统进行控制,同时测量系统的输出,这样对系统可取得 n 组输入输出数据:

$$(a_1, b_1; y_1), (a_2, b_2; y_2), \dots, (a_n, b_n; y_n)$$

其中, a_i, b_i 为输入数据, y_i 为输出数据, $i = 1, 2, \dots, n$

对这些数据进行处理求出模糊控制规则:

1) 根据数据取值范围确定模糊量

首先,确定 a_i, b_i, y_i 的取值范围,并在这 3 个物理量的取值范围中进行区间划分;然后,以每个区间对应一个模糊量,模糊量可取三角形或梯形,相邻模糊量交叠。

2) 从测量的数据对中求取每个数据对其论域模糊量的隶属度

例如,有 100 组数据组 $(a_1, b_1; y_1), \dots, (a_{100}, b_{100}; y_{100})$, 而每个物理量论域被分成 5 个模糊量,则有:对于 a_i 论域、 b_i 论域、 y_i 论域都分别有负大(NB)、负小(NS)、零(Z)、正小(PS)、正大(PB),则存在隶属度:

$$\begin{aligned} &\mu_{NB}(a_i), \mu_{NS}(a_i), \mu_Z(a_i), \mu_{PS}(a_i), \mu_{PB}(a_i) \\ &\mu_{NB}(b_i), \mu_{NS}(b_i), \mu_Z(b_i), \mu_{PS}(b_i), \mu_{PB}(b_i) \\ &\mu_{NB}(y_i), \mu_{NS}(y_i), \mu_Z(y_i), \mu_{PS}(y_i), \mu_{PB}(y_i) \end{aligned}$$

其中, $i = 1, 2, \dots, 100$ 。

3) 取有最大隶属度的数据及其模糊量

即求 $\max \mu(a_i), \max \mu(b_i), \max \mu(y_i)$

如果第 m 组数据 $(a_m, b_m; y_m)$ 有最大隶属度:

$$\max \mu(a_m) = \mu_{NB}(a_m) \quad 1 \leq m \leq 100$$

$$\max \mu(b_m) = \mu_{PS}(b_m) \quad 1 \leq m \leq 100$$

$$\max \mu(y_m) = \mu_{NS}(y_m) \quad 1 \leq m \leq 100$$

则取数据 a_m, b_m, y_m 和模糊量 NB、PS、NS,

4) 用每一组数据的最大隶属度相关的模糊量生成一条控制规则

对于第 m 组数据 $(a_m, b_m; y_m)$, 则有:

if $A = NB$ and $B = PS$ then $Y = NS$

其余同理。

5) 进行控制规则舍取

对于输入有两个语言值,每个语言值只取 NB、NS、Z、PS、PB 这 5 个模糊量的情况,它们的控制规则最多只含有 25 条;而在输入输出数据有 100 组时,会产生 100 条控制规则。显然,有的控制规则是重复或者不合理的,所以,应该进行控制规则的舍取。

首先,求每条规则的强度。设对第 m 条规则,其强度为:

$$D(R_m) = \max\mu(a_m) \times \max\mu(b_m) \times \max\mu(y_m) \quad (3.10)$$

其次,在求出每条规则的强度之后,把相互矛盾的规则中强度较小的舍去;把相同的规则合并成一条规则,从而最后得到控制规则。

在实际中,人们还对测量的数据存在有先期知识,称先验知识。例如,一个对被控对象较熟悉的操作人员或专家,可以凭自己对系统的了解而判断出测量数据的可靠程度,从而指出哪些数据对是合理的,哪些是较合理的,哪些是不太合理或不合理的。这时,人们对数据对进行评判,对每个数据对给出一个权系数,称置信度 ω ,设第 m 条规则的置信度为 ω_m ,则其强度为:

$$D(R_m) = \max\mu(a_m) \times \max\mu(b_m) \times \max\mu(y_m) \times \omega_m \quad (3.11)$$

式中, $\omega_m \in [0, 1]$ 。

上面给出的从测量数据生成控制规则的方法,有简单、直接的特点,它是一种一次性的生成方法,并且生成过程简捷,在工程应用上是一种十分有效和实用的方法。

(4) 根据学习算法生成控制规则

控制规则的生成除了根据经验,对系统测量 I/O 数据、过程模型等依据实现之外,还可以根据对系统的学习生成。

模糊控制器除了可以模仿人类的决策行为之外,还可以模仿人类的学习行为。

有时,人们得到的系统控制规则是相当粗糙的,甚至有时还会出现控制死区。用粗糙的控制规则去执行控制,往往不能取得令人满意的效果。以粗糙控制规则为基础,通过对系统的学习,生成完善和良好的控制规则,则是控制规则生成的另外一种方法。这种方法在要求较高的系统中是一种十分有效的方法。目前,这种方法还在发展中,人们也正致力于这方面的研究,以求得到各种优秀的学习方法和学习算法。

3.4.3 模糊控制规则的优化

在模糊控制系统中,模糊控制器的性能对系统的控制特性影响很大,而模糊控制器的性能在很大程度上取决于模糊控制规则。因此,模糊控制规则的优化具有重要的意义,它能使模糊控制取得更好的控制效果。

模糊控制规则的优化在本质上就是要解决控制规则的数量与质量问题,即要建立合适的规则数目和正确的规则形式,并给每条规则赋予适当的权系数,也称置信度。

(1) 模糊控制规则优化的内容

1) 模糊控制规则数量的优化

在一个模糊控制系统中,模糊控制规则的数量不能过干少。数量太少,则控制是不完全和粗糙的,可能造成未定义的盲区,使控制器没有输出;同样,控制规则的数量也不能太多,如果条数太多,则会出现功能相近的规则,造成处理上的麻烦和推理上的时间损耗。因此,模糊控制规则的数量优化是十分重要的。

2) 模糊控制规则质量的优化

控制规则的质量是指模糊控制规则的前件和后件之间的模糊推理关系是否处于合理状

态;是最合理状态,还是次合理状态?另外,不同的控制规则语句之间是否存在矛盾,如果一条规则的推理是不太合理的,则控制品质就会较差;如果两条推理规则是相互矛盾的,则可能产生控制发散,即系统会处于振荡状态。因此,模糊控制规则质量的优化是极为重要的问题。

总之,对模糊控制规则的优化,就是要寻找最恰当的规则数目,而且使控制规则有很好的-致性,同时,抛弃控制规则的交互性。

近年来,模糊控制规则的优化也越来越被人们所重视,原因就在于规则的优化直接提高了系统的控制质量。

早在1982年,龙升照和汪培庄就提出了模糊控制的自调整方法,这种方法利用修正因子对控制规则进行自动调整。后来,在1986年李东辉又进一步发展了这种方法。在这一段期间以来,也有不少人-对规则优化进行了探讨。1990年,Peng Xian—Tu提出了参数函数法;1991年,胡少华、刘少民提出了以对象规则模型为依据的自动生成控制规则方法及相应的自校正方法;1992年,Mikio Maeda和Shuta Murakami提出了实时学习校正法。这些方法各有特点,从不同的角度探讨了控制规则的优化问题,并在一定范围内在不同程度上取得了较好的控制效果。

(2) 模糊控制规则优化的方法

1) 参数函数校正法

在采用参数函数校正法进行优化时,首先要确定目标函数,一般取ITAE准则,即有:

$$J(ITAE) = \int_0^{\infty} t |e(t)| dt = \min \quad (3.12)$$

式中, $J(\cdot)$ 表示偏差加权时间之后的积分面积的大小; t 表示积分; T 为时间, A 为绝对值; E 为偏差。

目标函数确定之后,接着可以执行规则的优化:

①对参数 E 赋予一个初值,产生最初的规则。

②以该规则去对系统进行控制,并求出目标函数 $J(ITAE)$ 的值。

③修改参数的值,产生新的规则集,以新的规则去对系统进行控制,并求出目标函数新值 $J(ITAE)$ 。

④把目标函数的新值和原来的值比较,如果新值大于旧值,则反方向修改参数的值;如果新值小于旧值,则顺方向修改参数值。

⑤以新参数值产生新的规则,并以新的规则去对系统进行控制,求出目标函数新值 $J(ITAE)$,判别 $J(ITAE)$ 是否小于给定指标值 d ,如果 $J(ITAE) \geq d$,则返回④执行;如果 $J(ITAE) < d$,则优化结束。

2) 实时学习校正法

实时学习法是一种实时校正控制规则的方法,并在控制中使控制规则最适合于现行控制状态。这是Mikio Maeda和Shuta Murakami在1992年提出来的。

实时学习法首先按系统的实时响应和给定的参考模型的偏差 e 的情况,产生一个校正规则基;然后,对输入的信息执行模糊控制,并利用校正规则基去对控制规则的后件进行修正。

实时学习法的校正过程如下:

① 给出参考模型

对于一个被控系统,给出它的目标响应,用于估算控制响应对目标响应的一致程度。

参考模型是建立校正规则基的依据和出发点,考虑控制响应对目标响应的偏差情况,可以

建立一套校正规则基。

②建立校正规则基

校正规则基与控制规则基不同,两者虽然都是以条件语句组成,但它们的目标、作用是不一样的。控制规则基是模糊控制器用于控制被控系统的一个条件语句集;校正规则基则是模糊控制器用于校正控制规则的一个条件语句集。

在系统的每个采样时刻,都可以取得即时的控制响应,则校正规则基修正该时刻有关的控制规则,使它们的控制响应符合目标响应。

在控制规则中,语言变量值是单点,校正规则基是修改控制规则的后件,也即是修改控制量的语言变量值。在采样时刻,修改有关的控制规则,在本质上就是修改其后件的语言变量值。

为了得出校正规则基,考虑控制响应和目标响应之间的相互关系。在采样时刻 m 时,若偏差为负,同时有偏差变化率也为负;在采样时刻 k 时,有偏差为负,同时有偏差变化率为负。考虑采样点 k 时的校正情况:既然偏差为负,偏差变化率也为负,负偏差就还会有继续增大的趋势,这说明控制的输出信号过大,此时应该把控制校正向小的方向进行较大的改进。

③求对控制规则进行校正的校正算法

上面给出了校正规则基的形成方法以及校正规则基,这里考虑用校正规则基去进行控制规则校正的校正算法。

对于一个考虑偏差 e 、偏差变化率 Δe 、偏差变化率的变化率 $\Delta^2 e$ 的模糊控制器,它有三个输入,控制规则有如下格式:

if $e = a_e$ and $\Delta e = a_j$ and $\Delta^2 e = a_p$ then $\Delta u = b$.

对于 e 、 Δe 、 $\Delta^2 e$,其语言变量值取“负”(N)、“零”(Z)和“正”(P)三个,设它们的论域归一化到 $[-1,1]$;对于 Δu ,其语言变量值取“负大”(NB)、“负中”(NM)、“负小”(NS)、“零”(Z)、“正小”(PS)、“正中”(PM)、“正大”(PB)7个, Δu 的论域也归一化到 $[-1,1]$,模糊控制器有控制规则基如表 3.3 所示:

表 3.3 三输入的控制规则基

Δu			$\Delta^2 e$		
			N	Z	P
Δe	$e = P$	P	PM	PB	PB
		Z	PS	PM	PB
		N	Z	PS	PM
	$e = Z$	P	Z	PS	PM
		Z	NS	Z	PS
		N	NM	NS	Z
	$e = N$	P	NM	NS	Z
		Z	NB	NM	NS
		N	NB	NB	NM

从表 3.3 可以得到 27 条控制规则:

第 1 条: if $e = P$ and $\Delta e = P$ and $\Delta^2 e = P$ then $\Delta u = PB$

第 2 条: if $e = P$ and $\Delta e = P$ and $\Delta^2 e = Z$ then $\Delta u = PB$

.....

第 13 条: if $e = Z$ and $\Delta e = Z$ and $\Delta^2 e = P$ then $\Delta u = PS$

第 14 条: if $e = Z$ and $\Delta e = Z$ and $\Delta^2 e = Z$ then $\Delta u = Z$

第 15 条: if $e = Z$ and $\Delta e = Z$ and $\Delta^2 e = N$ then $\Delta u = NS$

.....

第 27 条: if $e = N$ and $\Delta e = N$ and $\Delta^2 e = N$ then $\Delta u = NB$

对于控制规则的后件语言变量值 b , 有如下校正算法:

$$b_i^k = b_i^{k-m} + (1 - FP) \Delta b \cdot \omega_i^{k-m}, \quad i = 1, 2, \dots, 13 \quad (3.13)$$

$$b_i^k = b_i^{k-m} - (1 - FP) \Delta b \cdot \omega_i^{k-m}, \quad i = 15, 16, \dots, 27 \quad (3.14)$$

式中: b_i^k ——第 i 条控制规则在第 k 时刻采样时的后件语言变量值;

b_i^{k-m} ——第 i 条控制规则在第 $k-m$ 时刻采样时的后件语言变量值;

FP——模糊性能; 其中, $FP = \min \{ \mu_{OV}(e_{OV}), \mu_{RT}(e_{RT}), \mu_{AM}(e_{AM}) \}$, $e_{OV} = OV - OV^*$, $e_{RT} = RT - RT^*$, $e_{AM} = AM - AM^*$, $\mu_{OV}(e_{OV})$ 、 $\mu_{RT}(e_{RT})$ 、 $\mu_{AM}(e_{AM})$ 分别是 e_{OV} 、 e_{RT} 、 e_{AM} 的优良度, OV 是控制响应超调, OV^* 是目标响应超调, RT 是控制响应到达时间, RT^* 是目标响应到达时间, AM 是控制响应振幅, AM^* 是目标响应振幅;

Δb ——第 k 采样时刻从校正规则所得的校正量;

ω_i^{k-m} ——第 $k-m$ 采样时刻第 i 条控制规则的适合度, 其中 $\omega_i = \mu_{ag}(e_k) \wedge \mu_{aj}(\Delta e_k) \wedge \mu_{ap}(\Delta^2 e_k)$, a_g, a_j, a_p 分别是 e 、 Δe 、 $\Delta^2 e$ 的语言变量值。

实时学习校正法是一种较有效的方法, 它的关键在于利用校正规则对控制的后件语言变量值进行调整。在控制规则中, 后件的语言变量值一般是单点; 初始的控制规则对单点的选择是不容易准确的, 实时学习法对后件单点的校正可以使控制趋于最优。

3) 修正因子校正法

修正因子校正法是利用修正因子对控制规则的后件进行校正的, 并且可用一个很简便的解析公式就可以执行优化。具有简单、方便等优点, 是一种有效的控制规则优化方法。

对一个典型的模糊控制系统, 其输入为偏差 e 、偏差变化率 Δe 和控制量 u , 这 3 个语言变量都含“正大”(PB)、“正中”(PM)、“正小”(PS)、“零”(Z)、“负小”(NS)、“负中”(NM)、“负大”(NB) 这 7 个语言变量值, 即有:

$$e = \{NB, NM, NS, Z, PS, PM, PB\}$$

$$\Delta e = \{NB, NM, NS, Z, PS, PM, PB\}$$

$$u = \{NB, NM, NS, Z, PS, PM, PB\}$$

设有初始控制规则表如表 3.4 所示。表中 e 的语言变量值为列元素, Δe 的语言变量值为行元素, e 和 Δe 相应的元素交汇点为控制量 u 的语言变量值。

为了便于对控制规则进行校正, 把控制规则数字化, 即把语言变量值定义为相应的整数对应的控制规则基, 显然, 可以用一个解析式来表示控制量 u 与偏差 e 、偏差变化率 Δe 的关系:

表 3.4 初始控制规则表

u		Δe						
		NB	NM	NS	Z	PS	PM	PB
e	NB	NB	NB	NM	NM	NS	NS	Z
	NM	NB	NM	NM	NS	NS	Z	PS
	NS	NM	NM	NS	NS	Z	PS	PS
	ZO	NM	NS	NS	Z	PS	PS	PM
	PS	NS	NS	Z	PS	PS	PM	PM
	PM	NS	Z	PS	PS	PM	PM	PB
	PB	Z	PS	PS	PM	PM	PB	PB

$$u = \langle e + \Delta e \rangle / 2 \quad (3.15)$$

其中, $\langle * \rangle$ 表示取与 $*$ 同号而其绝对值大于或等于 $|*|$ 的最小整数。

上式可以写成下面形式:

$$u = \langle 0.5 \times e + 0.5 \times \Delta e \rangle \quad (3.16)$$

或

$$u = \langle \alpha e + (1 - \alpha) \Delta e \rangle, \alpha = 0.5 \quad (3.17)$$

为了实现对控制规则的校正,把上式中的 α 取值范围扩展到 $[0, 1]$, 则有:

$$u = \langle \alpha e + (1 - \alpha) \Delta e \rangle \quad \alpha \in [0, 1] \quad (3.18)$$

从式 3.18 可知,当 α 取值不同时,则对 e 或 Δe 的权重不同,故而控制规则必定不同,控制效果必然有所不同。只要给出目标函数,在实际运行中修改 α 的值,就可以找到最优的控制规则。

3.4.4 模糊控制算法

要执行模糊控制,在计算机中就必须通过一定的算法实现,这些算法的目的就是从输入连续精确量中通过模糊推理的算法过程,求出相应的精确控制值。模糊控制算法有多种实现形式,但基于 Mamdani 推理的控制算法有三种:即关系矩阵法、查表法、解析法。

(1) 关系矩阵法

设控制系统的控制规则为:

if $e = A_i$ and $\Delta e = B_j$ then $u = C_y$,

其中:

$i = 1, 2, \dots, m; j = 1, 2, \dots, n$

e 是偏差, A_i 是偏差的语言变量值;

Δe 是偏差变化率, B_j 是偏差变化率的语言变量值;

u 是控制量, C_y 是对应于 A_i, B_j 的控制量的语言变量值。

则有模糊关系 R :

$$R = \bigcup_y A_i \times B_j \times C_y \quad (3.19)$$

式中: $i = 1, 2, \dots, m; j = 1, 2, \dots, n$

运算符“ \times ”表示对模糊量求内积,故模糊关系 R 为:

$$\mu_R(a, b, c) = \bigvee_{i=1, j=1}^{i=m, j=n} \mu_{A_i}(a) \wedge \mu_{B_j}(b) \wedge \mu_{C_{ij}}(c)$$

$$\forall a \in A, \forall b \in B, \forall c \in C$$

A, B, C 分别是偏差、偏差变化率、控制量的论域。

对于特定输入精确量 a^*, b^* , 则有输出:

$$u = (A \times B) \circ R \quad (3.20)$$

即

$$\mu_u(c) = \bigvee_{a \in A, b \in B} \mu_A(a^*) \wedge \mu_B(b^*) \wedge \mu_R(a, b, c)$$

最后再用重心法对 u 求精确值, 则可得到最终控制量 c^* 。

从上可知, 每当有一对信号 (a^*, b^*) 输入, 则必须用式(3.20)求解。 R 是一个矩阵, 故而每次都要 R 执行烦琐的合成运算, 这种关系矩阵法在实时控制系统中是不可行的。因为实时控制中要求实时性好, 不允许过长的计算时间, 所以它只是一种理论上的算法。查表法就是为了克服烦琐的实时计算而提出的。

(2) 查表法

查表法就是把所有可能的输入量都量化到语言变量论域的元素上, 并以输入量论域的元素作为输入量进行组合, 求出输入量论域元素和输出量论域元素之间关系的表格, 这个表格中元素的关系是按控制规则给出的, 称之为控制表。

在实际控制中, 模糊控制器首先把输入量量化到输入量的语言变量论域中, 再根据量化的结果元素去查表求出控制量。

查表法的关键在于制表, 生成控制表有两种方法: 一种是间接求取法, 另一种是直接求取法。间接法是首先求出模糊关系 R , 再根据输入的偏差和偏差变化率求出控制量, 最后把控制量精确化, 则可得控制表。直接法就是直接从控制规则(即推理语句)中求取控制量, 由此产生控制表。

在实际控制中, 只要执行对输入量量化和查表这两个步骤, 就可得到控制值, 可见查表法有很好的实时性, 并且方法简单方便。

(3) 解析公式法

对于控制表, 一般可以用下面的形式表示, 如表 3.5 所示:

表 3.5 典型控制表

u		Δe						
		-3	-2	-1	0	1	2	3
e	-3	-3	-3	-2	-2	-1	-1	0
	-2	-3	-2	-2	-1	-1	0	1
	-1	-2	-2	-1	-1	0	1	1
	0	-2	-1	-1	0	1	1	2
	1	-1	-1	0	1	1	2	2
	2	-1	0	1	1	2	2	3
	3	0	1	1	2	2	3	3

也可以用一个解析公式表示, 即

$$u = \left\langle \frac{e + \Delta e}{2} \right\rangle \quad (3.21)$$

其中, $\langle * \rangle$ 表示与 $*$ 同号, 而绝对值大于等于 $|*|$ 的最小整数。

显然, 可以采用下式进行控制量 u 求取:

$$u = \begin{cases} x, & x \text{ 为整数或零时} \\ x_{\text{整}} - 1, & x < 0, \text{并含小数时} \\ x_{\text{整}} + 1, & x > 0, \text{并含小数时} \end{cases} \quad (3.22)$$

其中:

$$x = \frac{e + \Delta e}{2}, x_{\text{整}} \text{ 为 } x \text{ 的整数部分。}$$

采用解析式(3.22)进行控制量求取, 无需存放控制表, 在计算机内可以节省存储空间, 而且这个解析式十分简单, 在程序处理上只是相加、移位、判别等简单操作, 因而实现十分方便。对于不同控制表也容易求出相应的解析式。

第 4 章

模糊控制器设计

模糊控制器是模糊控制系统的核心,是模糊控制系统控制品质的主要保证,因此,在模糊控制系统中,设计和调整模糊控制器的工作是很重要的。

模糊控制是以控制人员的经验为基础实施的一种智能控制,它并不需要精确的数学模型去描述系统的动态过程,因此,它的设计方法与常规控制器的设计方法有所不同。模糊控制器的设计,一般是先在经验的基础上确定各个相关参数及其控制规则,然后在运行中反复进行调整,以达到最佳控制效果。

模糊控制器的设计主要考虑以下几项主要内容:

- ①确定模糊控制器的输入变量和输出变量(即控制量);
- ②设计模糊控制器的控制规则;
- ③确立模糊化和解模糊的方法;
- ④选择模糊控制器的输入变量及输出变量的论域,并确定模糊控制器的参数(如量化因子、比例因子等);
- ⑤编制模糊控制算法的应用程序。

4.1 模糊控制器的基本结构及主要类型

4.1.1 模糊控制器的基本结构

模糊控制的基础是模糊集合理论和模糊逻辑,是用模糊逻辑来模仿人的思维对那些非线性、时变的复杂系统以及无法建立数学模型的系统实现控制的。图 4.1 是模糊控制器的基本结构,它主要由三大部分组成:模糊化接口、模糊逻辑推理机和解模糊接口。

(1) 模糊化接口(Fuzzification)

所谓模糊化,就是通过传感器把被控对象的相关物理量转换成电量,若传感器的输出量是连续的模拟量,还要事先通过 A/D 转换器转换成数字量作为计算机的输入测量值;接着再将此输入测量值进行标准化处理,即把其变化范围映射到相应内部论域中,然后将内部论域中的该输入数据转换成相应语言变量的概念,并构成模糊集合,这样就把输入的精确量转换为用模

顾名思义,知识库中存储着有关模糊控制器的所有相关知识,它们决定着模糊控制器的性能,是模糊控制器的核心。其中知识库包括应用领域的知识和相应控制目标的知识,它由数据库和语言控制规则库两部分组成。

①数据库(Data base)

数据库是用来定义模糊控制器中语言控制规则和模糊数据操作的,它虽然称做数据库,但并不是常规计算机软件中数据库的概念,它存储着有关模糊化、模糊推理、解模糊等相关知识,如前面已经介绍的模糊化中的论域变换方法、输入变量各模糊集合的隶属函数定义、模糊推理算法、解模糊算法、输出变量各模糊集合的隶属函数定义等。

②规则库(Rule base)

规则库包含若干组模糊控制规则,即以“if…then…”形式表示,对专家控制经验集成而形成的模糊条件语句。

语言控制规则库通过一系列语言控制规则来表征控制目标和该领域专家的控制策略,它是根据被控系统的行为特性和专家的控制经验总结编写而成的。

2) 模糊推理机

模糊推理机的主要功能是模仿人的思维特征,根据事先制定好的由专家知识或控制经验取得的若干组模糊条件语句构成的模糊控制规则,运用模糊数学理论对模糊控制规则进行计算推理,实际上是根据模糊控制规则对输入的一系列条件进行综合评估,以得到一个定性的用语言表示的决策输出量,这个结果给出某一个确定的输出范围,即所谓模糊输出量,完成这部分功能的模块就称做模糊推理机。

在模糊推理机中,模糊推理决策逻辑是核心,它能模仿人的模糊概念和运用模糊蕴涵运算以及模糊逻辑推理规则对模糊控制作用的推理进行决策。

(3) 解模糊接口(Defuzzification)

通过模糊推理得出的模糊输出量是不能直接去控制执行机构的,在这确定的输出范围中,还必须要确定一个最具有代表性的值作为真正的输出控制量,这就是所谓解模糊判决。完成这部分功能的模块就称做解模糊接口,它的主要功能包括:

1) 比例映射

比例映射将输出变量的量值从内部论域转化为相应的实际论域。

2) 解模糊

解模糊的主要功能是把经模糊推理所得到的模糊控制量转化为精确的控制作用,解模糊可以看做是模糊化的反过程,它从模糊推理结果中产生控制数值,作为模糊控制器的输出。

除了上述主要模块外,模糊控制器从完整和全面的角度出发,有时还包括以下几个模块:

①模糊规则修改、隶属函数修正和控制状态显示模块

增加该模块的目的是要根据模糊控制系统本身和环境的变化来对控制规则进行修正和完善,从而进一步对隶属函数做适应性修正,并随时显示当时的控制状态,以达到自组织、自适应控制的目的。

②PID 控制模块

如果用模糊推理来调整 PID 控制算法中的参数,那么经模糊推理得到的结果就不是直接作为系统的输出,而是用该结果来决定 PID 控制器参数,再根据 PID 算法来决定实际系统的控制输出。这就构成一种称为“支持型”的间接模糊控制器,如图 4.2 所示。

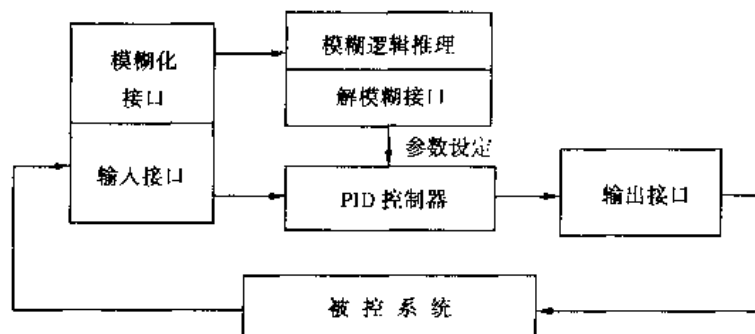


图 4.2 支持型的间接模糊控制器

③模拟模块

模拟模块是用纯软件的方法来模拟整个模糊控制的过程,并利用图形将运行状态在屏幕上直观地显示出来,还可以把输入和输出的关系用控制曲面方式显示出来,用以观察整个控制过程的平滑性。在设计控制系统前,可以先行用不同的数据作模拟试验,以得到可作为设计参考的某些有用的结果;在调试过程中,可以通过改变隶属函数及其相关参数来对各种不同的结果做比较,据此对控制系统做进一步的改进和优化。

4.1.2 常用的几种模糊控制器

(1)简单模糊控制器及其特性

简单模糊控制器就是目前广泛采用的二维模糊控制器,它的输入变量为误差 e 及误差变化率 Δe ,输出变量为控制量 u ,其结构如图 4.3 所示。

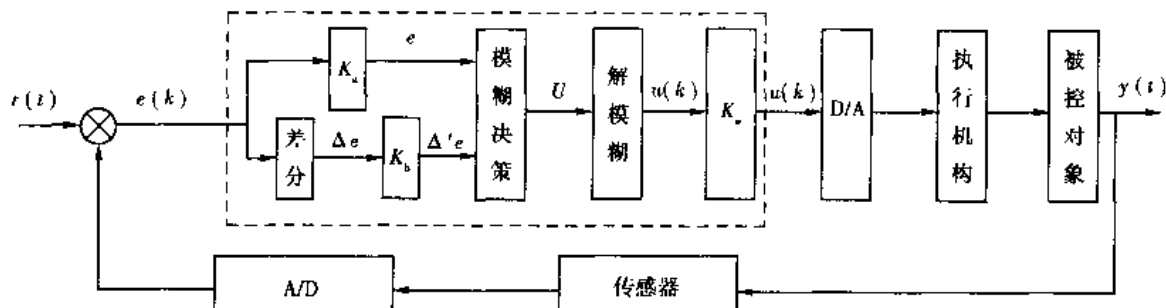


图 4.3 简单模糊控制器的结构

这种模糊控制器由模糊集合及论域根据实际情况确定输入变量和输出变量的隶属函数与模糊控制规则表,再通过一定的算法求出模糊控制查询表。在实际应用时,将采样所得的输入变量量化到其相应论域中,再根据量化的结果查找模糊控制规则表,得到的控制量,即作为模糊控制器的输出对被控对象施加控制作用。二维模糊控制器的输入语言变量虽然是两个输入量,但却是同一种物理量类型。由于偏差和偏差变化能够较全面严格地反映被控过程的动态特性,因此,其控制效果比一维模糊控制器好得多。

该模糊控制器具有设计简单、控制性能较好、对被控对象参数变化适应能力强等特点。但是,它有如缺点:因模糊控制表由输入输出变量及其论域和模糊变量的隶属函数所决定,一旦模糊控制表确定之后,这种模糊控制器的控制规则也就固定不变了。对于不同的被控对象,采用控制规则不变的简单模糊控制器难以获得预期的控制效果。

(2) 模糊自调整控制器

模糊控制器性能的好坏直接影响到模糊控制系统的控制特性,而模糊控制器的性能又取决于控制规则的完善与否。因此,如果在简单模糊控制器的输入输出关系式中加入修正因子,便能对控制规则进行自动调整,从而可对不同的被控对象都能获得相对满意的控制效果。

在简单模糊控制器中,如果将误差 e 、误差变化率 Δe 及控制量 u 的关系描述为:

$$u = \alpha e + (1 - \alpha) \Delta e$$

通过调整 α 的大小,可以改变对误差和误差变化的不同加权程度,从而可对不同的控制系统进行灵活方便的调整,这正是模仿了人工操作手动控制时的思维特点。当被控对象是低阶控制系统时,对误差的加权值应该大于对误差变化的加权值,即 α 应取大于 0.5 的值;当被控对象是高阶控制系统时,对误差的加权值应该小于对误差变化的加权值,即 α 应取小于 0.5 的值。

(3) 变结构模糊控制器

变结构模糊控制器实质上是多个模糊控制器的软组合,即在一个模糊控制器内部集成有多个简单的模糊控制器软件,每个简单模糊控制器的参数和控制规则各不相同,系统内部的软件开关根据系统的偏差情况分别接通不同的模糊控制器,而每个简单的模糊控制器都是针对系统不同状态下的不同要求而设计的,它能在相应状态下发挥良好的控制效果。变结构模糊控制器是将控制系统各个阶段的控制作用综合在一起,因此,这种模糊控制器有着良好的控制品质。

(4) 模糊 PID 控制器

目前,工业过程中广泛采用的 PID 控制,对具有非线性、大时滞、强耦合等特性的被控对象控制效果并不理想,虽然有最优 PID、非线性 PID 及自适应 PID 等改进形式的 PID 控制,但从根本上说,由于对 PID 参数的寻优是对比例、积分、微分三种控制作用的折中以及对于干扰抑制整定和目标值跟踪整定的折中,因而其整定的参数并不是最优的。也就是说, PID 控制器对不同的对象要用不同的 PID 参数,而且调整不方便,抗干扰能力差,超调量大;而模糊控制是一种语言控制,不依赖于被控对象的数学模型,设计算法简单、易于实现,能够直接从操作者的经验归纳、优化而得到,且适应能力强、抗干扰能力强、鲁棒性好。然而,模糊控制也有它的局限性和不足,这就是它的控制作用只能按挡处理,是一种非线性控制,控制精度不高,存在着静态余差,一般在语言变量偏差趋于零时有振荡;而传统的 PID 控制却能使控制的精度大大提高,消除稳态误差。因此,人们提出将传统的控制方法与模糊控制技术相结合,综合其优点,克服彼此的不足,组成一种复合控制器,即模糊 PID 控制器。

目前常见的模糊 PID 控制器有参数模糊自整定 PID 控制器和 P—FUZZY—PI 多模多段控制器。前者利用模糊控制器对 PID 控制器进行 PID 参数的在线自整定,其实现过程是:先找出 PID 三参数和误差与误差变化率之间的模糊关系,在运行中通过不断检测误差和误差变化,再根据模糊控制原理对上述三个参数进行在线修改,以满足不同误差和误差变化时对控制器参数的不同要求。后者是根据不同的条件和要求分段用不同的模态进行控制,即当误差大于某一个阈值时,用比例控制,以提高系统的响应速度,加快响应过程;当误差小于某一个阈值时,切换转入模糊控制,以提高系统的阻尼特性,减小响应过程的超调;当误差达到平衡点附近时,采用 PI 控制,利用其积分作用的特点最终消除误差。

(5) 模糊自组织控制器

有时人们对某一过程的认识比较贫乏,或者总结不出完整的经验来,这时提供的模糊控制规则就可能显得很粗糙,这势必影响模糊控制效果。另外,虽然模糊控制器具有一定的鲁棒性,但是对于非线性阶数较高、大时滞或随机干扰的过程,仅靠对操作者实践经验的总结或模糊信息的归纳,很难得出完整的模糊规则。另一方面,即使控制规则比较完善,由于过程是不断变化的,绝对时不变系统是不存在的,总是按原来的控制规则进行决策,所得的结果可能与实际情况相差甚远。这就促使人们去研究这样的模糊控制器,它在运行过程中能实时地自动修改、完善和调整控制规则,使系统的性能不断完善,以适应不断变化的情况,直到获得预期的效果,这就是模糊自组织控制器。

(6) 模糊自适应控制器

模糊自适应控制器可以根据系统的运行状态获取过程状态的连续信息,通过在线辨识和修正过程的模糊模型,从中获取所需的控制规则,实现在线模糊控制规则的自学习,自动调整模糊控制器的参数,以便适应环境条件或过程参数的变化以及由于人的经验获得的模糊控制规则的主观性和局限性,使系统大大地提高适应能力,获得较强的鲁棒性,维持控制系统所要求的性能准则。模糊自适应控制系统的结构如图 4.4 所示。

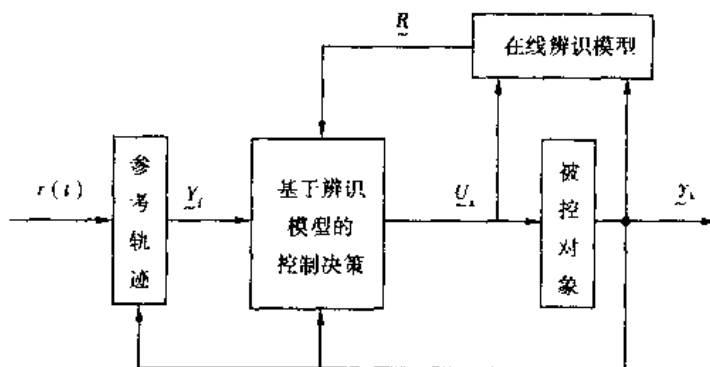


图 4.4 自适应模糊控制结构图

对于模糊自适应控制,实质上是在辨识对象模糊模型的同时,寻找一组模糊控制量,改变模糊控制器的参数,以满足最佳性能指标。

4.1.3 模糊控制器的设计

任何控制器的设计目标都要通过对给定的输入量进行运算推理后产生所期望的输出控制作用。当确定采用模糊逻辑控制机制后,首先就要决定是用硬件实现还是用软件实现。通常模糊控制器有两种实现方式:一种是由模糊逻辑芯片组成的硬件专用模糊控制器,它是直接用硬件芯片来实现模糊控制算法,其特点是推理速度快、控制精度高,但价格昂贵,输入和输出及模糊控制规则都十分有限,而且灵活性较差,因此,在实际控制系统中较少采用;另一种是用通用单片机组成硬件系统,因为大部分单片机的结构都集成了 CPU、存储器、I/O 接口甚至 A/D 和 D/A 转换器等,为实现模糊控制应用提供了较为理想的平台,加上最近几年国外不断推出的种种开发软件工具,使模糊控制设计和实现变得相对更加容易,所以目前大部分模糊控制应用都是通过通用单片机上运行模糊控制算法软件来实现的,这种模糊控制器的特点是资源开销小、灵活性高、通用性强、应用范围广。只有那些非常复杂和时间上有苛刻要求的应用才

可能需要用到模糊逻辑专用硬件芯片。

在模糊控制中,模糊控制器是整个控制系统的核心,它的硬件结构与普通数字控制器相同,因此,模糊控制器的设计实质上是设计模糊控制算法。当设计一个模糊控制器时,并不需要像设计数字控制器时那样,必须知道被控对象的精确的数学模型,而是根据模糊控制原理,按以下步骤来设计模糊控制器:

1) 确定模糊控制器的结构

即根据具体的系统确定其输入、输出变量。

2) 输入、输出变量的模糊化

即把输入、输出的精确量转化为对应语言变量的模糊集合。

3) 模糊推理决策算法的设计

即根据模糊控制规则进行模糊推理,并决策出输出模糊量。

4) 对输出模糊量进行解模糊判决

即通过各种解模糊方法完成由模糊量到精确量的转化,实现对被控对象的控制。

基于以上的设计步骤,在实际应用中,往往是根据实际系统的动态和静态特性的要求,尽量地简化算法,努力地提高模糊控制器的实时性及自适应性能。

4.2 模糊控制器的结构设计

4.2.1 模糊控制器的结构设计

模糊控制器的结构设计实质上是模糊控制器输入语言变量及输出语言变量的选取和模糊控制器的不同组合与扩展问题。模糊控制器的结构选择是否合理,不仅直接影响模糊控制器的性能,而且对于那些复杂的多输入多输出耦合系统来说,是至关重要的。模糊控制器的结构设计是模糊控制设计成功的第一步。究竟选择哪些变量作为模糊控制器的信息量,必须深入研究在手动控制过程中人如何获取和输出信息,因为模糊控制器的控制规则归根到底还是要模拟人脑的思维决策方式。

通常将模糊控制器输入变量的个数称为模糊控制的维数,如图4.5所示。一般情况下,一维模糊控制器用于一阶被控对象,由于这种控制器输入变量只选误差一个,它的动态控制性能不佳,所以,目前被广泛采用的均为二维模糊控制器,这种控制器以误差和误差的变化为输入变量,以控制量的变化为输出变量。

从理论上讲,模糊控制器的维数越高,控制越精细;但是,维数过高时,模糊控制规则就会变得过于复杂,控制算法的实现相当困难。因此,目前人们广泛地设计和应用的都是二维模糊控制器。

在设计模糊控制器时,首先是根据被控对象的具体情况和对系统的性能指标要求作为结构选型参考的依据。由于模糊控制器的控制规则是根据操作人员的控制经验提出的,而一般操作人员只能观察到被控对象的输出变量和输出变量的变化,或者观察到输出变量和输出变量的总和这两个状态,因此,在模糊控制器中,总是选取误差和误差的变化率或误差及误差的和作为它的输入变量,而把控制量的变化作为模糊控制器的输出变量,这样就确定了模糊控制

器的结构。

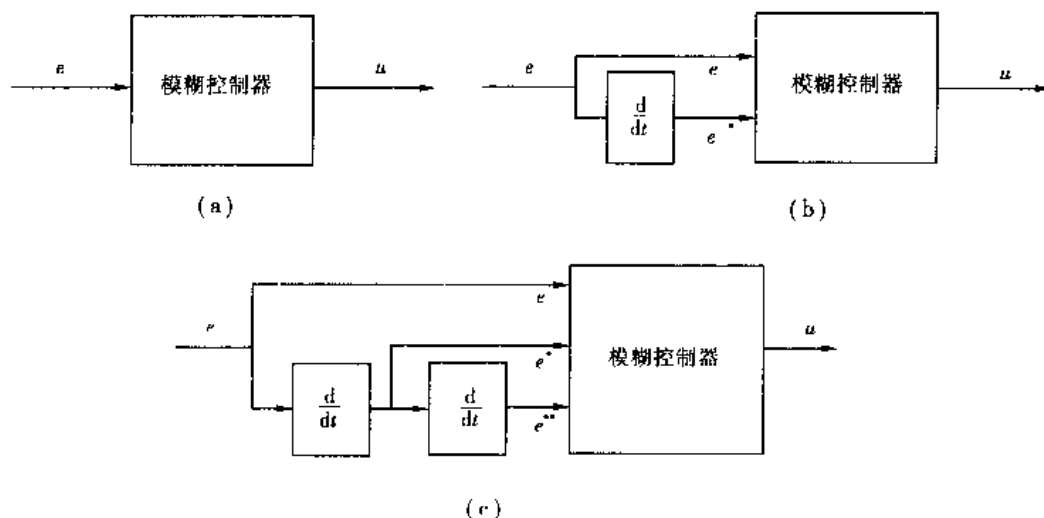


图 4.5 常见模糊控制器的结构类型

(a) 一维模糊控制器; (b) 二维模糊控制器; (c) 三维模糊控制器

一旦确定好模糊控制器的结构,模糊控制器的输入语言变量和输出语言变量也就相应地确定了。一般用得最多的是将偏差 e 和偏差变化 Δe 作为输入语言变量,模糊化后分别用符号 e 、 Δe 表示,这时的模糊控制器就类似于一个 PD 控制器,从而有利于保证系统的稳定性,减少响应过程的超调量并减弱其振荡。

4.2.2 一般设计方法

(1) 模糊控制器的设计原则

模糊控制器是一种利用人的直觉和经验设计的专家控制系统,设计时不是用数学解析模型来描述被控系统的特性,一般性的原则性设计步骤如下:

1) 模糊语言变量的语言值分档和模糊论域分级的选取

① 模糊语言变量语言值分档的选取

模糊条件语句中描述输入、输出语言变量状态的词汇(也称为语言值,如“正大”,“正中”,“负小”,“负大”等)的集合,称为这些模糊语言变量的词集(也可以称为模糊语言变量的模糊状态),它是根据模糊语言的定义,由语法规则生成的语言值的集合。而每一个语言值本身又是一个模糊集合,因此,要确定一个模糊语言变量,必须首先确定其语言值,才能构造模糊控制规则。那么,如何选取模糊语言变量的语言值(词集),也就是如何对语言变量值分档呢?人类在对某一个事物用语言进行描述时,总是习惯于用如“高”、“中”、“低”,“大”、“中”、“小”,“快”、“中”、“慢”等 3 个词汇来描述模糊控制器的输入输出变量的状态。另外,由于人类的行为在正、反两个方向的判断基本上是对称的,所以可以进一步将模糊语言变量加上正负两个方向并考虑“零”状态,例如,“负大”(NB)、“负中”(NM)、“负小”(NS)、“零”(Z)、“正小”(PS)、“正中”(PM)、“正大”(PB)等,对于“零”,有时候还可以细分为“负零”(NZ)、“正零”(PZ),然后根据需要,还可以对模糊语言变量的取值加上不同的程度副词(语气算子),如“很”、“非常”、“稍微”等,这样就使得对事物的描述显得更生动和丰富多彩。

选择较多的词汇描述输入、输出变量,可以使制定控制规则方便,但控制规则相应变得复

杂;选择词汇过少,使得描述变量变得粗糙,导致控制器的性能变坏。一般来说,一个语言变量 m 选用 2~10 个语言值较适宜,通常都选择“负大”、“负中”、“负小”、“零”、“正小”、“正中”、“正大”等 7 个词汇,但也可以根据实际系统需要选择 3 个或 5 个语言变量。

在定义输入和输出变量时,要考虑到软件实现的限制,一般当输入变量小于 10 个时,软件推理还能应付,但当输入变量的数目再增加时,就要考虑采用专用模糊逻辑控制推理集成芯片,这在常用的控制系统中并不多见。

需要说明的是,在采用不同的模糊推理方法时,语言变量值的分档覆盖范围是有区别的。在采用合成推理(CRI)法推理时,需要把变量的基本论域(也称实际论域或物理论域)转换成模糊变量的模糊论域(整数论域),然后在模糊论域中去对语言变量值分档;在非 CRI 推理时,则可以直接对变量的物理论域进行语言变量值的分档。

②模糊语言变量模糊论域分级的选取

根据被控系统的实际情况,确定输入变量的测量范围和输出变量的控制作用范围,以进一步地确定每个变量的论域,然后再安排每个变量的语言概念及其相对应的隶属函数。

在采用合成推理算法时,为了在实时控制中避免模糊关系矩阵合成运算所浪费的计算机时间,降低其工作效率,总是采取在脱机状态下将全部输入、输出之间的关系计算出来,制成一张控制表(响应表)存入计算机中,实际工作时,在某一采样时刻根据实测的输入变量直接去查控制表就可以得到输出响应。控制表是以整数表示输入量和控制量的。为了能生成控制表,在 CRI 推理方法中,要求将变量的基本论域转换成有限个数的模糊论域,也就是把连续变化的论域离散化,使其成为离散论域。

对于输入偏差变化 Δe 、输入偏差变化的变化 $\Delta^2 e$ 以及输出变量 u 也可以做类似的划分。

由于语言变量值的档数不能过多也不能过少(一般 m 取 2~10),基于同样的道理,模糊论域中元素也不能太多或太少。一般而言,模糊论域的元素个数为 $2n+1$,它和语言变量的分档数 m 的关系为:

$$2n+1 = km \quad (4.1)$$

式中, k 取 2~3。

2)量化因子和模糊语言变量值分档范围的确定

①量化因子及其确定方法

设偏差的基本论域为 $e = [-e_{\max}, e_{\max}]$,其模糊论域为:

$$E = \{a\} = \{-n_a, -n_a+1, \dots, -1, 0, 1, \dots, n_a-1, n_a\}$$

又设偏差变化的基本论域为:

$$\Delta e = [-\Delta e_{\max}, \Delta e_{\max}]$$

其模糊论域为:

$$E_c = \{b\} = \{-n_b, -n_b+1, \dots, -1, 0, 1, \dots, n_b-1, n_b\}$$

定义:

$$K_a = \frac{n_a}{e_{\max}} \quad (4.2)$$

$$K_b = \frac{n_b}{\Delta e_{\max}} \quad (4.3)$$

式中, K_a 、 K_b 分别称为偏差语言变量的量化因子和偏差变化语言变量的量化因子。

如果在采样时刻得到物理量基本论域中的一个精确值(例如,偏差的精确值 e_i),则可以找到模糊论域 E 中的一个元素 a_i 与之对应,这就是所谓的量化过程。即

$$a_i = K_e e_i \quad (4.4)$$

如果所得到的结果含有小数,则可以采用四舍五入的方法对 a_i 取整。

同理,可对偏差变化精确值 Δe_i 进行量化。

量化因子的确定是很关键的,它不仅影响系统的动态品质,也会影响到系统的稳定性。概括地说, K_e 越大,系统的超调量也就越大,过渡过程时间相应就越长; K_e 越小,则系统变化越慢,但稳态精度降低; K_d 越大,则系统输出变化就越慢; K_d 越小,则系统反应越快,但超调量会相应增大。

②模糊语言变量值分档范围的确定

对于CRI推理,设偏差或偏差变化的模糊论域为 $\{-4, -3, -2, -1, 0, 1, 2, 3, 4\}$, 偏差或偏差变化率语言变量值分 m 档,取 $k \approx 1.5$,根据式(4.1)有:

$$m = \frac{2n+1}{k} = \frac{2 \times 4 + 1}{1.5} = 6$$

这时,可以取偏差或偏差变化语言变量值的6档分布如下:

NB——隶属度峰值取在 $-4, -3$ 附近;

NS——隶属度峰值取在 $-2, -1$ 附近;

NZ——隶属度峰值取在 -0 附近;

PZ——隶属度峰值取在 $+0$ 附近;

PS——隶属度峰值取在 $1, 2$ 附近;

PB——隶属度峰值取在 $3, 4$ 附近。

反之,若已知控制器模糊语言变量值的分档 m ,根据式(4.1)有:

$$n = \frac{km-1}{2}$$

就可以求得模糊语言变量模糊论域的级数 n ,然后再加以分配。

3)模糊论域上模糊集合隶属函数的确定及其表示方法

①模糊语言变量各语言值模糊集合隶属函数的确定

模糊语言变量的每一个语言值实际上是一个在模糊论域上的模糊集合。由模糊集合的定义可知,模糊集合最终总是通过隶属函数来描述的,定义一个模糊集合,实际上就是要确定模糊集合隶属函数曲线的形状。隶属函数可以通过总结操作者的控制经验,采用多种方法来确定。将确定的隶属函数曲线离散化,就得到了有限个点上的隶属度,构成了一个相应的模糊变量的模糊集合。语言值隶属函数又称为语言值的语义规则,它有时以连续函数的形式出现,也可以离散的量化等级的形式表达,连续的隶属函数描述比较准确,而离散化的量化等级简洁、直观。所以,模糊论域上的一个连续或离散的隶属函数就代表着一个模糊语言变量的一个语言值。

②语言变量值隶属函数的表示方法

一个语言变量是由它的一系列语言值的集合来描述的,每一个语言值又是一个模糊量(模糊集合),而模糊集合终归是用隶属函数描述的。这些语言值的模糊集合可以用图形、表格、公式及坐标等隶属函数的表示方法来进行描述。

③语言值隶属函数对模糊控制性能的影响

模糊语言变量在确定其语言值模糊集合的个数及每个语言值所覆盖的模糊论域大小时,应该考虑以下几方面情况:

A. 隶属函数的幅宽大小对控制性能的影响

实验研究结果表明,语言值模糊集合隶属函数的形状是三角形、梯形或是正态分布,对控制效果的影响不是很大,但各个模糊集合其隶属函数的幅宽大小(也就是每个模糊集合对整数论域覆盖范围的大小)对性能的影响较大,所以,一般选用三角形、梯形隶属函数作为描述模糊集合的隶属函数,这是因为它们的数学表达和运算较简单,占用内存空间小,在输入值变化时,比正态分布或钟形分布隶属函数具有更大的灵敏性,当存在一个偏差时,能很快反应产生一个相应的控制量输出。

由于三角形的隶属函数它的形状仅与直线斜率有关,因此,适合于有隶属函数在线调整的自适应模糊控制;而梯形可视为三角形的截头,所以其适用范围与三角形隶属函数相同。

高斯型隶属函数是描述模糊集合的一种比较合理的形式,这种隶属函数的特点是:连续且处处可微,适合自适应、自学习模糊控制隶属函数的修正。

隶属函数曲线形状较尖的模糊集合其分辨率较高,控制灵敏度也较高;相反,隶属函数曲线形状较缓,控制特性也较平缓,系统稳定性较好。因此,在选择模糊变量的模糊集合的隶属函数时,在误差较大的区域,应采用低分辨率的模糊集合;在误差较小的区域,应采用较高分辨率的模糊集合;当误差接近于零时,宜选用高分辨率的模糊集合。

B. 隶属函数元素个数对控制性能的影响

以上仅就某一模糊语言变量(如偏差)的某一个模糊语言值子集的隶属函数进行了分析,而在定义同一个模糊语言变量的各个模糊集合时,则要使它们全体在模糊论域上的分布合理,即应该较好地覆盖整个论域。因此,在确定某一语言变量模糊集合的个数时,应使论域中任何一个元素对这些模糊集合的隶属度的最大值都不能太小,否则,会在这些点附近出现不灵敏区的“空档”,甚至造成失控,使模糊控制系统性能变坏。

为此,应适当地增加各模糊变量的模糊集合论域中的元素个数,如一般论域中的元素个数的选择均不低于13个,而模糊集合总数通常选7个。当论域中元素总数为模糊集合总数的2~3倍时,模糊集合对论域的覆盖程度较好。

C. 模糊集合隶属度之间相互关系对控制性能的影响

如图4.6所示,各个模糊集合隶属度之间相互关系对控制性能的影响,可以用模糊语言变量所取的所有语言值模糊集合中任何两个模糊集合的交集的最大隶属度中的最大值 α 来描述。 α_1 及 α_2 分别为两种情况下两个模糊集合A和B的交集的最大隶属度,显然, α_1 小于 α_2 。当 α 值较小时,控制灵敏度较高;而当 α 值较大时,模糊控制器对于被控对象参数变化适应性强,鲁棒性较好,即控制器具有较好的适应对象特性参数变化的能力。 α 值取得过小或过大都是不利的,一般选取 α 值为0.4~0.8。 α 值过大时,造成两个模糊集合难以区分,使控制的灵敏度显著降低。在实际工作中,无论模糊集合采用什么样的隶属函数,相邻模糊集合之间要存在交集,同时,不应该有3个隶属函数相交的状态,掌握这一条原则是非常必要的。

D. 隶属函数的位置分布对控制性能的影响

设计一个模糊控制器,原则上应首先从简单开始,语言值隶属函数档数可以取少些(如3档),在进一步优化时,再根据情况考虑增加。另外,隶属函数在整个模糊论域平均分布,控制

效果并不好,作为优化的第一步,将三角形模糊集合“零”固定在“工作点”上,而其他模糊集合则向“零”集靠拢。一个模糊控制器的非线性性能与隶属函数总体的位置分布有密切关系,通过隶属函数总体位置的非线性分布设计,可以在一定程度上解决控制器的非线性性能。

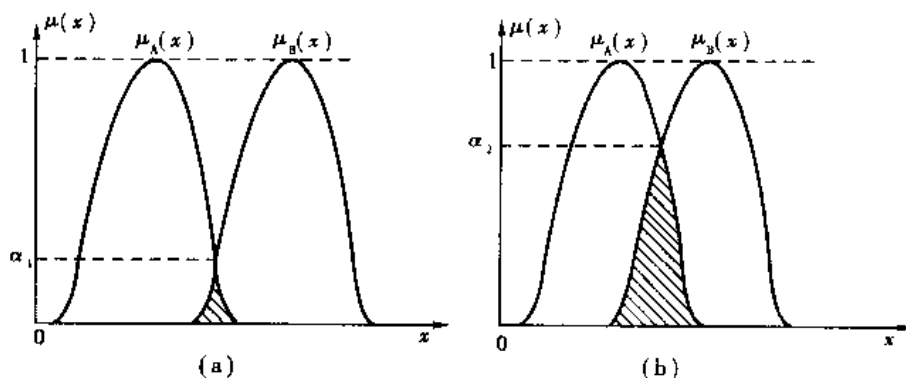


图 4.6 两个隶属函数曲线的重叠(相交)程度

4) 模糊控制规则及算法结构的确定

模糊控制器的控制规则是由输入输出模糊语言变量的不同语言值排列组合而构成的一组模糊条件语句,它们反映了人工手动控制的某种思维方式,而手动控制策略又是人们通过学习、试验以及长期经验积累而逐渐形成并存储在操作者头脑中的一种技术知识集合。

手动控制的作用与自动控制系统中的控制器的作用是基本相同的,所不同的是手动控制决策是基于人脑操作的经验和知识,而控制器的控制决策是基于某种控制算法的数值运算。

利用语言归纳手动控制策略的过程,实际上就是建立模糊控制器的控制规则的过程,这是一个把专家知识和熟练操作者的经验转换为用语言表达的模糊控制规则的过程。手动控制策略一般都可以用条件语句加以描述。

在模糊控制中,控制策略的选择是非常关键的一步。只有选择恰当的控制规则,才能很好地体现有经验的操作者的控制策略,以获得较好的控制效果,既能保证响应的快速性,又能保证系统的稳定性。

算法结构就是体现模糊控制规则的模糊关系 R ,它相当于一般控制器的传递函数,但这种算法结构不是建立在对象数学模型基础上综合出来的,而是根据控制系统输入输出关系的数据观测,并采用模糊集合理论处理而得到的。

① 模糊控制规则的形式

由于模糊控制规则是一系列模糊条件语句的集合,因此,讨论模糊控制规则的形式就必然涉及模糊条件语句的形式。

A. 一维模糊控制器

一维模糊控制器的结构如图 4.5(a)所示,其中,输入模糊语言变量 e 可以取许多语言值,如 NB, NS, ..., PS, PB 等,记为 \underline{e}_i ,即 $\underline{e} = \underline{e}_i$, \underline{e}_i 属于模糊论域 E ; 输出模糊语言变量 u , 所取语言值记为 \underline{u}_i ,即 $\underline{u} = \underline{u}_i$, 因此,其模糊控制规则是一系列手动操作经验的归纳总结。也就是说,只要有一个输入语言值 \underline{e}_i , 就有一个输出语言值 \underline{u}_i 与其对应。模糊控制规则是由一系列模糊条件语句组成的,即

if $e = e_i$ then $u = u_i$

也可以说,模糊控制规则是输入语言变量和输出语言变量之间的模糊关系 R 的集合,这种模糊控制规则反映的是一种比例(P)关系。

B. 二维模糊控制器

二维模糊控制器的结构如图 4.5(b)所示,其中,输入模糊语言变量 e 可以认为是系统的偏差,它可以取若干个语言值 e_i ,如 NB,NS,⋯,PS,PB 等,这些语言值是系统输入偏差精确量 e 的模糊化结果,属于整数论域;输入模糊语言变量 Δe 是系统偏差变化率,它可以取若干个语言值 Δe_i ,这些语言值是系统偏差变化率精确量 Δe 模糊化的结果,输出模糊语言变量 u 是模糊控制器的输出量变化,取若干个语言值 u_i ,这些语言值经过解模糊后成为精确量 u_i ,这类模糊控制规则是下面一系列模糊条件语句,即

if $e = e_i$ and $\Delta e = \Delta e_i$ then $u = u_i$

这是模糊控制系统中用得最多的一种模糊控制规则,它反映了一种比例加微分(PD)控制规律。

C. 多输入单输出模糊控制器

这类模糊控制器最典型的是三维模糊控制器,这类模糊控制器的结构可用图 4.5(c)来说明,其中多个输入模糊语言变量 $e, \Delta e, \Delta^2 e, \dots$ 各自可以取若干个语言值 $e_i, \Delta e_i, \Delta^2 e_i, \dots$,而输出模糊语言变量取语言值 u_i ,这种模糊控制规则可以用下面一系列模糊条件语句表示。

if $e = e_i$ and $\Delta e = \Delta e_i$ and $\Delta^2 e = \Delta^2 e_i$ and \dots then $u = u_i$

②算法结构

模糊推理算法与模糊控制规则直接相关,它的复杂性依赖于模糊规则语句中的模糊集合的隶属函数的确定。Mamdani 方法是最早最常用的一种方法,也是一种比较简便的方法,选择一些简单的、又能反映模糊推理结果的隶属函数,可以大大地简化模糊推理的计算过程。目前,在大多数的控制过程中多数选用三角形隶属函数。尽管如此,实现模糊控制的计算量仍是很大的,为提高控制系统的实时性,应尽量地优化算法结构。

基于人工控制策略所得的每一条模糊条件语句只是代表了一种情况下的一个控制策略,工程实践中,由于操作者在进行控制的过程中要碰到各种可能出现的情况,因此,反映手动控制过程完整的控制规则要由若干条结构相同而语言变量取值不同的模糊条件语句组成。显而易见,各条模糊条件语句决定的控制策略之间是一种并列关系,即“或”的关系。输出控制量应取上述各相关控制策略的“和”。

③选择模糊控制规则应注意的问题

模糊规则可以通过领域的专家给出,也可以通过大量实验数据得到。要得到模糊规则,首先要解决一个模糊规则生成的问题。特别是在给出一组观测数据的情况下,要有一种方法能够通过观测数据得到模糊规则,这也就是精确数据模糊化的问题。

不管是用领域专家给出的模糊规则,还是用精确数据模糊化的方法得到的模糊规则,这些模糊规则都是近似的和并不完善的规则,有时甚至会存在一些相互矛盾的规则,这时,必须要解决这些模糊规则的相互谐调问题。也就是说,要建立一种方法来删掉一些相互矛盾的规则,使剩余的模糊规则是基本谐调的,即必须进行模糊控制规则的优化。

A. 建立模糊控制规则表的基本思想

在建立模糊控制规则表时,首先考虑误差为负的情况,例如,当误差为负大时,若当误差变化为负,这时误差有增大的趋势,为尽快地消除已有的负大误差并抑制误差变大,所以,控制量的变化取正大。

当误差为负而误差变化为正时,系统本身已有减少误差的趋势,所以,为尽快地消除误差且又不超调,应取较小的控制量;当误差为负大且误差变化为正小时,控制量的变化取为正中;若误差变化正大或正中时,控制量不宜增加,否则,造成超调会产生正误差,因此,这时控制量变化取为“零”等级。

当误差为负中时,控制量的变化应该使误差尽快地消除,基于这种原则,控制量的变化选取同误差为负大时相同。

当误差为负小时,系统接近稳态,若误差变化为负时,选取控制量变化为正中,以抑制误差向负方向变化;若误差变化为正时,系统本身有消除负小的误差的趋势,选取控制量变化为正小即可。

上述选取控制量变化的原则是:当误差大或较大时,选择控制量以尽快地消除误差为主;而当误差较小时,选择控制量要注意防止超调,以系统的稳定性为主要出发点。

B. 模糊控制规则数的确定

首先要决定被控系统有哪些输入的操作控制状态必须被监测和哪些输出的控制作用是必须的,然后根据输入和输出变量的个数,就可以求出所需要规则的最大数目:

$$N = n_{out} \cdot (n_{level})^{n_{in}} \quad (4.5)$$

这里, n_{in} 是输入变量的个数, n_{out} 是输出变量的个数, n_{level} 是输入模糊划分的数目。

然而,实际上有的组合状态不会出现,所以真正用到的规则数没有这么多。根据实验结果建议用以下经验公式来计算:

$$N = n_{out} \cdot (n_{in} \cdot (n_{level} - 1) + 1) \quad (4.6)$$

在对模糊控制器的语言变量值分档时,如果分档过多过细,由于人们没有足够的控制规则知识,会产生规则数量太多而导致规则质量下降的问题。但是,控制规则太少也是不利的。例如,对于二维输入单输出的模糊控制器来说,若将每个输入语言变量的语言值分成5挡,按矩阵排列应该有25条模糊控制规则。但是,如果只取位于横轴 $e = Z$ 和位于纵轴 $\Delta e = Z$ 上十字交叉形的规则库作为控制规则,显然,只有9条模糊控制规则。若按这9条控制规则进行模糊推理的话,就会出现“未定义值”的盲区,这样的控制效果是很差的,这显然是由于9条控制规则太少所造成的,解决的办法可以将控制规则数适当地增加。

C. 模糊控制规则的置信度

模糊控制的效果除了与控制规则条数多少以及正确的规则形式有关之外,还与每条规则的置信度(权数)有关。当某一规则不受限制地适用时,它的置信度等于1;而当它仅在一定的过程状态下适用时,它的权数因子应取小于1的数。所谓某条规则的置信度,就是在其结论部分的语言值上乘以0~1的系数。置信度一般凭经验给出或通过仿真试验来确定,而目前正在兴起的神经网络技术则是建立规则和确定置信度的最佳方法之一。

图4.7列出了一个只有十字形交叉规则库和经过优化后的规则库对同一对象进行模糊控制效果的比较。由图可见,在语言变量语言值档相同的情况下,规则的质量对于控制品质的优劣起着关键性作用。因此,语言值档数多少并不是评价控制品质的惟一标准,即使控制规则数

量不多,但其质量较好的情况下,仍然能达到较理想的控制效果。

5) 实时精确量的量化及模糊化

在实时采样过程中,得到的输入偏差精确量 e_i 或偏差变化 Δe_i 必须经过量化模糊化,只有将精确量变成模糊量后才能实施模糊控制。

模糊决策的作用是从已知的连续输入精确量中,通过各种不同的模糊推理算法,求出模糊控制器相应的输出量,这个输出量是一个模糊集合。通常被控对象只能接受一个精确的控制量。因此,必须从决策值模糊集合中判决出一个确切的精确量,这就需要通过解模糊判决来实现。解模糊判决也叫做反模糊化。判决即是从模糊集合到普通集合的一个映射,是从一个模糊量变成清晰量的过程,判决的结果是输出模糊论域中的一个元素,即等级数,这是一个清晰值。解模糊判决是在模糊决策之后进行的一个控制环节。

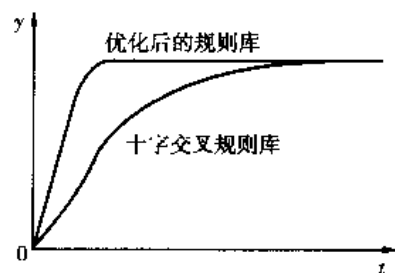


图 4.7 模糊控制器的控制效果比较

(2) 模糊控制器的设计途径

目前,设计模糊控制器的途径可以从三个方面来考虑:

- 根据专家的知识 and 经验
- 通过建立熟练操作者的控制模型
- 通过建立被控制对象的模糊模型

1) 以专家的知识 and 经验作为依据的设计方法

如前所述,模糊控制器实际上是应用于控制的专家系统,其设计依据就是专家的知识 and 经验。这里通过一个简单例子说明这种设计思路。

如果要实现一个模糊控制的沿直线道路单行道自行驾驶的汽车,为简单起见,假设车速是固定的,汽车上装有超声波探测器,用以探测方向和位置。其探测方法为:在沿行车路线的一侧建有回音壁,汽车上的超声波发生器不断发射脉冲,车上的超声波接收器根据回音壁的超声反射信号来判断汽车的位置,并通过动态调节接收器的方向,以使接收到的信号最大,并由此确定汽车运动方向与道路方向之间的夹角,这样就可用车的位置和方向这两个输入参数作为控制汽车方向盘转动角度的依据。

设道路中线到汽车中心的距离为 d ,汽车前进方向与道路中线的夹角为 θ ,方向盘的转动角度是 φ ,则有:

$$d, \theta \rightarrow \varphi$$

约定,当汽车中心在道路中线左边时,其距离为正值,在右边时为负值,其变量分别用“左”和“右”来表示。

根据专家知识和经验有:

“如果汽车在道路中线的左边,方向与道路方向一致,那么应该将方向盘向右转,以接近道路中线。”

“如果汽车在道路中线的右边,方向与道路方向一致,那么应该将方向盘向左转,以接近道路中线。”

“如果汽车方向偏左,那么应该将方向盘向右转,以对准道路方向。”

“如果汽车方向偏右,那么应该将方向盘向左转,以对准道路方向。”

把这些知识和经验转换成相应的四条模糊规则就是：

R1:如果 $d = \text{“左”}$, 且 $\theta = \text{“中”}$, 那么 $\varphi = \text{“右”}$;

R2:如果 $d = \text{“右”}$, 且 $\theta = \text{“中”}$, 那么 $\varphi = \text{“左”}$;

R3:如果 $\theta = \text{“左”}$, 那么 $\varphi = \text{“右”}$;

R4:如果 $\theta = \text{“右”}$, 那么 $\varphi = \text{“左”}$ 。

根据上述控制规则,如果在行进中,汽车位于道路中线左边,方向偏右,这时会激活规则一和规则四两条规则,根据规则一的要求,方向盘应该转向右;但是,根据规则四的要求,方向盘又应该转向左,这时就要比较两者的隶属度,若规则一的满足程度大于规则四的满足程度,则将方向盘稍微转向右边;但是,当继续转向右边时,规则四的满足程度会不断增加,同时,规则一的满足程度会不断下降,当规则四的满足程度大于规则一时,就执行规则四的结果,修正汽车方向;其余类推。同样,如果汽车在道路中线的左边,方向偏左时,就激活规则一和规则二,因为这两条规则的结果都是要求方向盘转向右,所以方向盘可以较大的角度向右转,修正汽车的方向。由此可见,即使在汽车偏离了道路中心线的情况下,经过几次调整修正汽车就会自动回到目标线上,得以实现良好的控制。

2) 通过建立熟练操作者控制模型的设计方法

现场控制专家和熟练操作者可以巧妙地根据其经验实现对复杂系统的控制,但是,要把专家或者操作者控制的经验和诀窍用逻辑形式表达出来就不那么容易,而且不同的专家所拥有的经验不尽相同,虽然殊途同归,但毕竟其控制效果也各有差异。专家可以控制复杂系统,但他未必能用语言来准确描述整个控制的过程和方法步骤。还是以驾驶汽车为例,如果在驾驶汽车转弯时,方向盘的控制要根据车速和转弯半径的大小对刹车时刻和轻重的掌握,全凭当时驾驶员的感觉和手握方向盘与脚踩刹车的巧妙配合。这些连续的控制和配合几乎无法用语言准确地表达出来。熟练驾驶员所能提供的只能是一些原则性的知识描述,学习驾驶的人要完成这一系列的动作,只有这一些原则性的知识描述还远远不够,还需要通过训练去掌握一系列协调配合动作的技能。大脑中存储的知识可以用语言来表达,但是,用控制动作才能表达的“技巧”就难于用纯语言准确地表达出来。目前人工智能领域通常意义下的咨询专家系统只考虑知识的表达,但是,在控制专家系统中不仅需要知识的表达,而且必不可少地需要这种“技巧”的表达,对于仅考虑了知识表达的模糊控制规则,为了真正能达到模仿熟练操作者控制的能力,就必须考虑系统能通过训练获得所需要的技巧,具有不断改善和自学习的功能。

3) 建立被控对象模糊模型的设计方法

以上两种方法都是通过建立专家的模型,并以此模型作为推理规则来实现模糊控制。显然,这类模糊控制器的性能不可能超越所依赖的专家的水平。而有的控制对象根本无法找到该领域有经验的控制专家,对这样的被控对象进行模糊控制时,目前通常采用的方法是通过建立被控对象的模糊模型来实现。

所谓建立被控对象的模糊模型,就是用像建立模糊控制规则一样的“如果……那么……”这种形式来描述被控对象的动态特性。一条“如果……那么……”表达式就是一条控制规则,因此,被控对象的模型是由多条控制规则组成的,这样,通过该模型就可从输入推理得到输出。

如前所述,在经典控制理论中,系统模型是用传递函数来描述的,在现代控制理论中,则用微分方程来描述;而在模糊控制理论中则用“如果……那么……”这种形式来描述。但是,前两者的一个共同特点是用对实际系统简化后的“精确的”数学函数关系来描述整个系统,而模

模糊控制理论则是用未经简化但具有覆盖一定区域的“不精确的”语言来描述整个系统。要建立被控制对象的模糊模型,就首先要研究模糊控制的特征。

模糊模型之所以具有“如果……那么……”的形式,主要是为了控制器与被控对象能融为一体、和谐的整体。

模糊控制的推理有三种方法:直接推理方法、间接推理方法和后件是线性函数的推理方法。相应模糊模型描述形式也应该有三种,但最适用于控制对象模型化的推理方法是第三种推理方法,因为这种方法适于推理后件部分用线性表达式来描述。如果用第一种推理方法,被控对象的控制规则数就要多得多。一般在已知被控对象模型的情况下,有两种设计控制器的方法:第一种方法是先设定所要控制对象的行为,据此找到能满足这种要求的控制算法。在模糊控制器中,被控对象的响应可以看成由被控对象的规律加上控制规则产生的;第二种方法是用评估函数来定量评估被控对象的响应特征,以找出能使其变化量最小的控制算法,这种方法可以把控制规律看成由被控对象的规律加上对被控对象响应的评估而产生的。

第 5 章

神经网络在模糊控制系统中的应用

神经元是以生物神经系统的神经细胞为基础的生物模型,在人们对生物神经系统进行研究和探讨人工智能的机制时,把神经元数学化,从而产生了神经元数学模型。神经网络模型是一个高度非线性动力学系统,具有并行分布处理、存储及学习能力、高度鲁棒性和容错能力,能充分逼近复杂的非线性关系。因此,用神经网络可以表达物理世界的各种复杂现象。

5.1 神经网络

5.1.1 人工神经网络的起源和发展

随着科学技术的发展,人们对自身的研究从先期的医学和心理学研究到后来信息学的介入,生物医学家企图从解剖学的角度来说明认知行为和人脑在这种行为中的作用;心理学家希望从人的记忆、思维、联想心理活动的特征上来解释人脑的思维活动机理;信息学家则试图从人脑的拓扑结构、信息的传递、信息的存储方式、信息的加工方式来阐述人脑的功能,并模拟人脑的工作功能。不同学科的科学家不断地对人脑的研究和互相影响,渐渐地形成了神经网络学科。神经网络的研究已有 50 多年的历史,这期间经历了曲折不平的发展过程。

1943 年,心理学家 McCulloch 和数学家 Pitts 合作,提出了第一个神经网络 MP 模型。1949 年,心理学家 Hebb 通过对大脑神经细胞学习和条件反射的观察研究,提出了改变神经元连接强度的 Hebb 规则。1957 年, Rosenblatt 提出了感知器 Perceptron 模型,试图模拟人脑的感知和学习能力,并提出了引入隐层处理单元的三层感知器的概念。1960 年, Widrow 和 Hoff 提出了自适应线性元件 Adaline 模型及一种有效的学习方法 Widrow-Hoff 规则,从而在 20 世纪 60 年代,掀起了神经网络研究的第一次热潮。

神经网络是从仿生学角度模拟人脑神经系统的运作方式,使机器具有人脑那样的感知、学习和推理能力。这种神经网络不同于当时以顺序离散符号推理为基本特征的人工智能工作方式,因而引起了学术上的研究兴趣和争议。随着研究的发展,人们在应用和实现方面遇到了一时难以解决的难题。人工智能的创始人之一 Minsky 和 Papert 潜心数学,于 1969 年出版了颇具影响的《Perceptron》一书,得出了悲观的结论。由于 Minsky 在学术上的地位,这些观点使许

多神经网络研究者丧失了信心。另一方面,当时数字计算机的发展正处于鼎盛时期,基于数字计算机的人工智能得到了迅速的发展并取得了显著的成就,从而掩盖了发展新型模拟计算机和人工智能技术的必要性和迫切性,使神经网络的研究走向低潮。

这期间仍有不少学者在极其困难的情况下致力于神经网络的研究。1969年, Grossberg 等提出了 ART 自适应共振理论; 1972 年 Kohonen 提出了自组织映射理论; 与此同时, 心理学家 Anderson 提出了 BSB 模型; 日本学者 Fukushima 提出了 Neocognitron 认知机理论; Werbos 提出了 BP 误差反向传播理论; Widrow 发展了 Adaline 模型; Amari 则致力于有关神经网络的数学理论的研究。这些开创性的研究为神经网络的进一步发展奠定了理论基础。

1982 年, 美国加州工学院的物理学家 Hopfield 教授提出了 HNN 模型, 使神经网络的研究有了突破性的进展。它通过引入“能量函数”的概念, 给出了网络的稳定性判据。此外, HNN 的电子电路实现为神经计算机研究奠定了基础, 同时, 开拓了神经网络用于联想记忆和优化计算的新途径, 掀起了神经网络研究的又一热潮。随后, Feldman 和 Ballard 提出了连接网络模型, 指出了传统的人工智能“计算”与生物“计算”的不同点, 给出了并行分布计算的处理原则; Hinton 和 Sejnowskii 借助统计物理学的概念和方法, 提出了 Boltzmann 机神经网络模型, 并采用模拟退火技术进行网络的训练, 保证了整个网络系统处于全局稳定点; Rumelhart 和 McClelland 以及他们领导的 PDP 小组致力于认知微观结构的探索, 提出了 PDP 理论, 同时, 发展了多层网络的 BP 学习算法, 将神经网络的研究推向了高潮。

这一时期, 随着大量开拓性研究工作的深入, 研究者们提出了数十种网络结构和学习算法, 硬件实现的研究工作也在积极开展。神经网络理论的应用研究已经渗透到各个领域, 并取得了令人可喜的进展。

1987 年 6 月, IEEE 在 San Diego 召开了第一届神经网络国际会议, 国际神经网络学会随之成立; 1988 年《神经网络》杂志创刊; 1990 年 IEEE 神经网络会刊问世; 各种神经网络学术刊物不断涌现, 研究成果层出不穷。

我国的神经网络研究起步较晚, 始于 20 世纪 80 年代后期, 主要应用领域开展了一些基础性工作。1989 年召开了全国第一届神经网络—信号处理会议; 1990 年召开了第一届神经网络学术大会。1992 年国际神经网络学会和 IEEE 联合学术会议在北京召开。

20 世纪 90 年代以来, 神经网络的研究进入了一个空前高涨时期。多数研究集中在网络结构、学习算法和实际应用三个方面, 已经广泛地应用于控制领域, 尤其是非线性系统领域。神经网络在系统中的应用主要用做神经网络建模和神经网络控制。

神经网络的研究目前还处于起步阶段, 随着科学技术的进步及相关学科的发展, 神经网络及其应用的研究将取得更大的进步。

5.1.2 神经元和神经网络

(1) 神经元模型

1) 生物神经元模型

人脑由大量的神经细胞相互连接组合而成。神经元是以生物神经系统的神经细胞为基础的生物模型, 是生物神经系统的最基本单元, 其生物学解剖结构可以用图 5.1 表示。

从图中看出: 脑神经元由细胞体、树突和轴突三部分组成。细胞体是神经元的中心, 细胞体的伸延部分产生的分枝称为树突。树突是神经元的信息入口, 其作用是接受从其他神经元

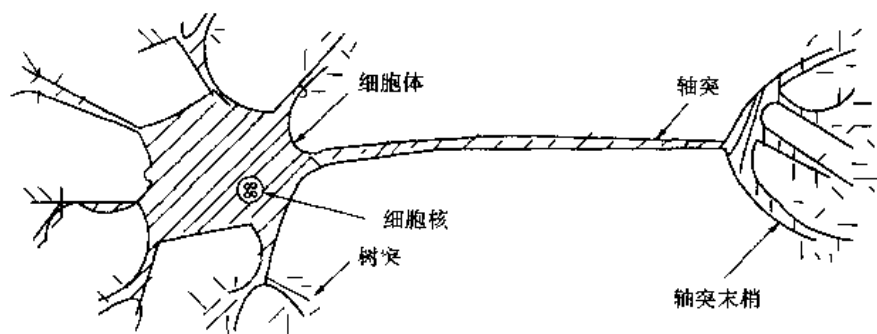


图 5.1 生物神经元模型

传入的信息。细胞体突出的最长的外伸管状纤维称为轴突。轴突是把神经元兴奋的信息传出到其他神经元的出口。

2) 人工神经元模型

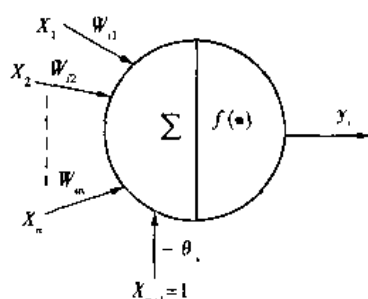


图 5.2 人工神经元模型

神经元模型是模拟生物神经元的结构和功能的数学化模型，一般是一个多输入单输出非线性的信息处理单元。典型的人工神经元模型如图 5.2 所示。

图中， x_j 是神经元的输入， $j=1, 2, \dots, n$ ， θ_i 是 i 神经元的阈值； w_{ij} 分别是 i 神经元对 x_j 的连接权系数； y_i 是 i 神经元的输出， $f(\cdot)$ 是激发函数，它决定 i 神经元受到输入 x_j 的共同刺激达到阈值时以何种方式输出。该神经元的数学模型表达式为：

$$\text{状态: } u_j = \sum_{i=1}^n \omega_{ij} x_j - \theta_i$$

输出: $y_i = f(u_i)$

对于激发函数 $f(\cdot)$ 有多种形式，其中，常用的有如下 3 种：

① 阶跃型函数: $f(u_i) = \begin{cases} 1, & u_i > 0 \\ 0, & u_i \leq 0 \end{cases}$

② 线性型函数: $f(u_i) = ku_i$

③ Sigmoid 型函数: $f(u_i) = \frac{1}{1 + \exp(-u_i)}$

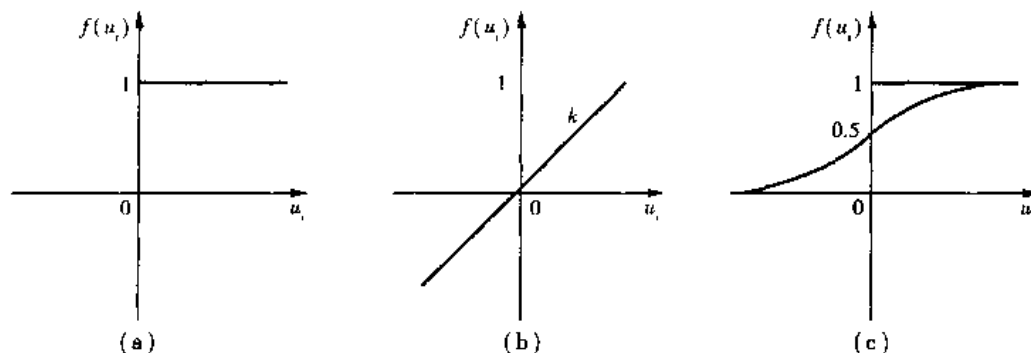


图 5.3 常用激发函数

(2) 人工神经网络

人工神经网络是生物神经网络的一种模拟近似,本书主要介绍几种典型的人工神经网络模型及在模糊控制方面的应用。

1) 神经网络模型

神经网络在目前已近 40 种不同的模型,人们从不同的角度对神经网络进行分类,通常可按以下 5 个原则进行神经网络的归类:

- 按照网络的结构区分,则有前向网络和反馈网络。
- 按照学习方式区分,则有有教师学习和无教师学习网络。
- 按照网络性能区分,则有连续型和离散型网络,随机型和确定型网络。
- 按照突触性质区分,则有一阶线性关联网络和高阶非线性关联网络。
- 按对生物神经系统的层次模拟区分,则有神经元层次模型、组合式模型、网络层次模型、神经系统层次模型和智能型模型。

神经网络的强大功能是通过神经元的互连而实现的。常用的神经网络有如图 5.4 所示的 4 种形式:

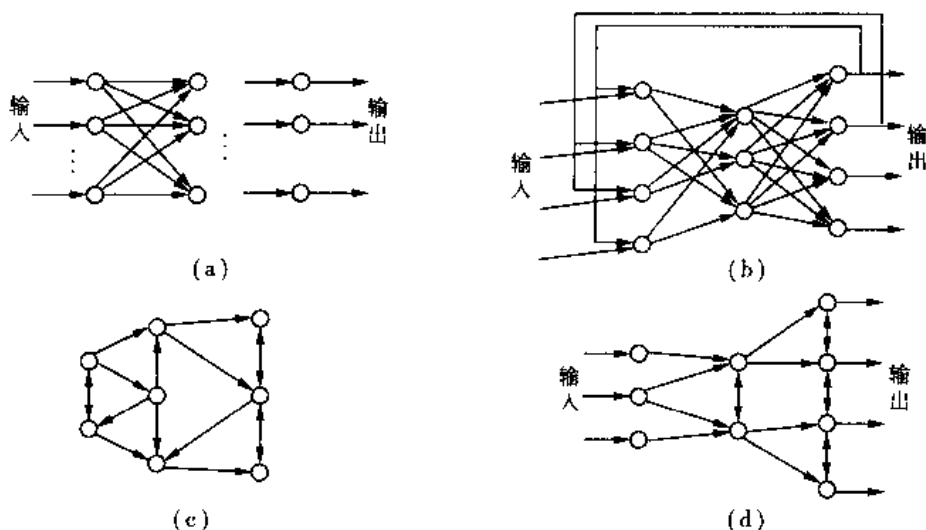


图 5.4 神经网络结构示意图

①前向网络

前向网络如图 5.4(a)所示:神经元分层排列,组成输入层、隐含层和输出层。每一层的神经元只接收前一层神经元的输入。输入模式经过各层的顺序变换后,在输出层输出。各神经元之间不存在反馈。感知器和误差反向传播算法中使用的网络都属于这种类型。

②反馈网络

在前向网络的基础上,存在输出层到输入层神经元之间的反馈,如图 5.4(b)所示。这种模式可用来存储某种模式序列,如神经认知机或 Hopfield 网络。也可以用于动态时间序列过程的神经网络建模。

③相互结合型网络

相互结合型网络属于网状结构,如图 5.4(c)所示。这种神经网络在任意两个神经元之间都可能存在连接。HNN 或 Boltzman 机网络都属于这一类型。信号将在神经元之间反复往返传递,网络处于一种不断改变的动态之中。从某初态开始,经过若干次变化,才会到达某种平衡

状态,还可能进入周期振荡或其他(如混沌等)平衡状态。

④混合型网络

它是层次型网络和网状型网络的一种结合,如图 5.4(d)所示。通过层内神经元的相互结合,可实现同一层内神经元之间的横向抑制或兴奋作用,这样可以限制每层内能同时动作的神经元数,或者把层内的神经元分为若干组,让每组作为一个整体来动作。

2) 神经网络学习算法

学习的过程实质上是针对一组给定输入,使网络产生相应的期望输出的过程。神经网络学习算法分为两大类:有教师学习和无教师学习。

有教师学习是指在神经网络训练过程中,始终存在一个期望的网络输出,期望输出与实际输出之间的距离作为误差度量并用于调整网络连接权系数。无教师学习指的是网络不存在一个期望的输出值,因而没有直接的误差信息,因此,为实现对神经网络的训练,需建立一个间接的评价函数,以对网络的某种行为趋向做出评价。

5.1.3 人工神经网络的特点及类型

(1) 人工神经网络的特点

1) 非线性

人脑的思维是非线性的,故人工神经网络模拟人的思维也应是非线性的。

2) 非局域性

这是人类神经网络的一个特性,人的整体行为是非局域性的最明显体现。神经网络以大量的神经元连接模拟人脑的非局域性,它的分布存储是非局域性的一种体现

3) 非定常性

神经网络是模拟人脑思维运动的动力学系统,它应按不同时刻的外界刺激对自己的功能进行修改,因而它是一个时变的系统。

4) 非凸性

神经网络的非凸性是指它有多个极值,也即系统具有不只一个的平衡状态。这会使系统的变化多样化,神经网络的全局优化算法就反映了这一点,例如,模拟退火法。

5) 具有泛化功能

它能够处理那些未经训练过的数据,而获得相应于这些数据的合适的解答。同样,它能够处理那些有噪声或不完全的数据,从而显示了很好的容错能力。对于许多实际问题来说,泛化能力是非常有用的,因为现实世界所获得的数据常常受到噪声的污染或残缺不全。

6) 具有自适应功能

根据所提供的数据,通过学习和训练,找出和输出之间的内在联系,从而求得问题的解答,而不是依靠对问题的先验知识和规则,因而它具有很好的适应性。

7) 高度并行处理

神经网络的处理是高度并行的,因此,用硬件实现的神经网络的处理速度,可远远高于通常计算机的处理速度。

(2) 人工神经网络的类型

在人们提出的几十种神经网络模型中,应用较多的是 Hopfield 网络、BP 网络、Blotzman 机和 ART 自适应共振理论网络。

Hopfield 网络是最典型的反馈网络模型,它是目前人们研究得最多的模型之一。Hopfield 网络是由相同的神经元构成的,并且不具学习功能的自联想网络,它需要对称连接。这个网络可以完成制约优化和联想记忆等功能。

BP 网络是反向传播(Back Propagation)网络。它是一种多层前向网络,采用最小均方差学习方式。这是一种应用最广泛的网络。它可用于语言综合、识别和自适应控制等用途。BP 网络需有教师训练。

Blotzman 机是典型的随机网络模型。它的单元具有二值状态,神经元的连接是双向的,并且有概率状态转移机制。Blotzman 机允许含隐单元来获取学习中的高阶规则。它用噪声过程顶替代价函数的全局极小值。它主要用于模式识别。Blotzman 机训练时间长,并且有噪声。

ART 网络是一种自组织网络模型。这是一种无教师学习网络。它能够较好地协调适应性、稳定性和复杂性的要求。在 ART 网络中,通常需要两个功能互补的子系统相互作用,这两个子系统称注意子系统和新生子系统。ART 网络主要用于模式识别,它不足之处在于对转换失真和规模变化敏感。

5.1.4 典型的人工神经网络

Hopfield 网络和 BP 网络在自动控制中应用较广,模糊神经网络常用 BP 网络来实现。因此,这里主要介绍 BP 网络。

(1) Hopfield 网络

Hopfield 神经网络属于反馈型神经网络,如图 5.4 所示。从系统观点看,它是一种动力系统,网络的运动是多自由度的,网络学习的目的是使之趋于一个能量函数的局部极小值,并获得一个渐进稳定解。Hopfield 网络有离散型和连续型两种。

1) 离散型 Hopfield 网络

典型的离散型 Hopfield 网络只包含输入层和输出层,并且输入节点与输出节点数是一样的,每个输出层神经元的激发函数均为符号函数,网络动态行为有异步和同步两种工作方式。异步方式中任一时刻,只有一个神经元发生变化,其余神经元状态都不变。同步方式中同一时刻,有多个神经元发生变化,或所有神经元都发生变化。

2) 连续型 Hopfield 网络

若把离散型 Hopfield 网络中的激发函数改为连续函数,网络行为用微分方程表示,就构成了连续型 Hopfield 网络。连续型 Hopfield 网络被广泛地用于约束优化问题。

(2) BP 多层神经网络

如图 5.5 所示,BP 多层神经网络是用 BP 算法所训练的多层前向网络。在人工神经网络的实际应用中,最常见到的是 BP 网络或它的变化形式,BP 网络主要用于以下几方面:

①函数逼近 用输入和相应的输出矢量训练一个网络逼近一个函数。

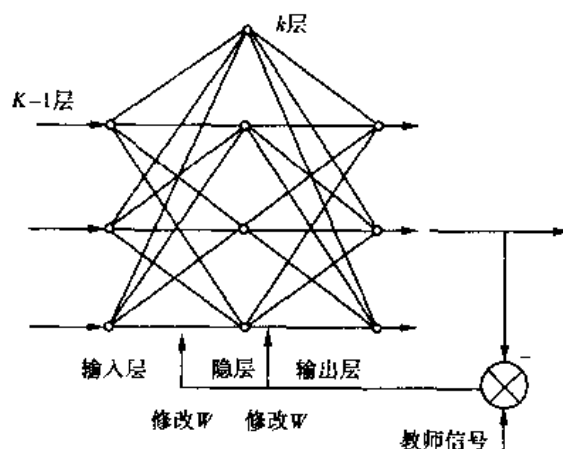


图 5.5 多层网络及 BP 算法

②模式识别 用一个特定的输出矢量将它与输入矢量联系起来。

③分类 把输入矢量以所定义的合适方式进行分类。

④数据压缩 减少输出矢量维数,以便于传输或存储。

1985年,Rumhart和McClelland等提出了反向传播学习方法,即BP(back propagation)算法。这种算法可以对网络中各层的连接权系数进行修正,故适用于多层前向网络的学习。BP算法是目前使用最广泛的神经网络学习算法之一,是在自动控制中常用的学习算法。

1) BP 算法的原理

BP算法适用于多层前向网络。多层前向网络含有输入层、中间层和输出层。中间层有单层或多层,也称为隐层。在隐层中的神经元称隐单元。隐层的状态影响输入、输出之间的关系。也就是说,改变隐层的权系数,可以改变整个多层神经网络的性能。

设有一个 m 层的神经网络,并在输入层加有样本 X ,设第 k 层的 i 神经元的输入总和表示为 u_i^k 、输出 x_i^k ;从第 $k-1$ 层的第 j 个神经元到第 k 层的第 i 个神经元的权系数为 ω_{ij} ,各个神经元的激发函数为 f ,则各个变量的关系可用下面有关的数学式表示:

$$x_i^k = f(u_i^k) \quad (5.1)$$

$$u_i^k = \sum_j \omega_{ij} x_j^{k-1} \quad (5.2)$$

反向传播算法分两步进行,即正向传播和反向传播。

①正向传播

输入样本从输入层经过隐层单元一层一层地进行处理,通过所有的隐层之后,则传向输出层;在逐层处理的过程中,每一层神经元的状态只对下一层神经元的状态产生影响。在输出层把现行输出和期望输出进行比较,如果现行输出不等于期望输出,则进入反向传播过程。

②反向传播

反向传播时,把误差信号按原来正向传播的通路反向传回,并对每个隐层的各个神经元的权系数进行修改,以望误差信号趋向最小。

2) BP 算法的数学表达

BP算法实质是求取误差函数的最小值问题。这种算法采用非线性规划中的最速下降方法,按误差函数的负梯度方向修改权系数。

多层网络的训练方法是把一个样本加到输入层,并根据向前传播的规则:

$$u_i^k = \sum_j \omega_{ij} x_j^{k-1} \quad (5.3)$$

$$x_i^k = f(u_i^k) \quad (5.4)$$

不断、一层一层地向输出层传递,最终在输出层可以得到输出 x_i^m 。

把 x_j^m 和期望输出 y_j 进行比较,如果两者不等,则产生误差信号 e ,接着则按下面的公式反向传播修改权系数:

$$\Delta \omega_{ij} = -\eta \cdot d_i^k \cdot x_j^{k-1} \quad (5.5)$$

式中, η 为学习速率,即步长,一般取 $0 \sim 1$ 间的数;当 $k = m, d_i^m = x_i^m(1 - x_i^m)(x_i^m - y_i)$;当 $k < m, d_i^k = x_i^k(1 - x_i^k) \cdot \sum_l \omega_{il} \cdot d_l^{k+1}$ 。

通过多个样本的反复训练,同时,向误差渐渐减小的方向对权系数进行修正,以达到最终消除误差。如果网络的层数较多时,所用的计算量就相当大,导致收敛速度不快。为了加快收

敛,一般考虑上一次的权系数,并以它作为本次修正的依据之一,因而有修正公式:

$$\Delta\omega_{ij}(t+1) = -\eta \cdot d_i^k \cdot x_j^{k-1} + \alpha\Delta\omega_{ij}(t) \quad (5.6)$$

式中, η 取0.1~0.4,为学习速率; α 取0.7~0.9,为权系数修正常数。

3) BP 算法的执行步骤

在反向传播算法应用于前向多层网络,采用 Sigmoid 为激发函数时,可用下列步骤对网络的权系数 ω_{ij} 进行递归求取:

①对权系数 ω_{ij} 置初值

对各层的权系数 ω_{ij} 置一个较小的非零随机数,但其中 $\omega_{in+1} = \theta_i$ 。

②输入一个样本 $X = (x_1, x_2, \dots, x_n, 1)$ 以及对应期望输出 $Y = (y_1, y_2, \dots, y_n)$ 。

③计算各层的输出

对于第 k 层第 i 个神经元的输出 x_i^k ,有:

$$u_i^k = \sum_{j=1}^{n+1} \omega_{ij} x_j^{k-1}, \quad x_{n+1}^{k-1} = 1 \quad (5.7)$$

$$x_i^k = f(u_i^k) \quad (5.8)$$

④求各层的学习误差 d_i^k

对于输出层 $k = m$,有:

$$d_i^m = x_i^m (1 - x_i^m) (x_i^m - y_i) \quad (5.9)$$

对于其他各层,有:

$$d_i^k = x_i^k (1 - x_i^k) \sum_l \omega_{il} \cdot d_l^{k+1} \quad (5.10)$$

⑤修改权系数 ω_{ij} 和阈值 θ_i

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \cdot d_i^k \cdot x_j^{k-1} \quad (5.11)$$

或

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \cdot d_i^k \cdot x_j^{k-1} + \alpha\Delta\omega_{ij}(t) \quad (5.12)$$

⑥当求出了各层各个权系数之后,可按给定品质指标判别是否满足要求。如果满足要求,则算法结束;如果未满足要求,则返回③执行。

这个学习过程对于任一给定样本 $X_p = (x_{p1}, x_{p2}, \dots, x_{pn}, 1)$ 和期望输出 $Y_p = (y_{p1}, y_{p2}, \dots, y_{pn})$ 都需要执行,直到满足所有输入为止。

4) BP 网络的设计

设计 BP 网络时,一般应从以下几方面考虑:

①网络的层数

理论证明:具有偏差和至少一个 S 型隐含层加上一个线性输出层的网络,能够逼近任何有理函数。因此,增加层数可以进一步降低误差,但同时会使网络复杂化,增加网络的训练时间。而精度的提高实际上也可以通过增加隐层中的神经元数目来获得,其训练效果也比增加层数更容易观察和调整。所以,通常应先考虑增加隐层中的神经元数。

输入神经元数可以根据要求解的问题和数据所表示的方式来确定。如果输入的是电压波形,那么可根据电压波形的采样点数来决定输入神经元的个数,也可以用一个神经元,使输入样本为采样的时间序列。如果输入为图像,则输入可以用图像的像素,也可以为经过处理后的图像特征来确定其神经元个数。

②隐层的神经元数

对于网络训练精度的提高,可以通过增加隐层节点的方法来获得。这在结构上要比增加隐层要简单得多。隐层节点数的选取在理论上并没有明确的规定。在具体设计时,通常是通过不同的节点数进行训练对比,然后适当地加上一些余量。

③初始权值的选取

由于系统是非线性的,初始值对于学习是否达到局部最小,是否能够收敛以及训练时间的长短影响很大。如果初始权值太大,使得加权后的输入落在了S型激活函数的饱和区,导致激活函数导数 $f'(s)$ 非常小,使得权值修正 $\Delta\omega_j \rightarrow 0$,从而使得调节过程几乎停顿。所以,一般总是希望经过初始加权后的每个神经元的输出值都接近于零,这样可以保证每个神经元的权值都能够在它们的S型激活函数变化最大之处进行调节。因此,一般取初始权值在 $(-1,1)$ 之间的随机数。

④学习速率

学习速率决定每一个循环训练中所产生的权值变化量。大的学习速率可能导致系统的不稳定;小的学习速率导致较长的训练时间,可能收敛很慢,不过能保证网络的误差值不跳出误差表面的低谷而最终趋于最小误差值。所以,在一般情况下,倾向于选取较小的学习速率,以保证系统的稳定性。学习速率的选取范围在 $0.01 \sim 0.8$ 之间。

对于较复杂网络,在误差曲面的不同部位可能需要不同的学习速率。为了减少寻找学习速率的训练次数以及训练时间,比较合适的方法是用变化的自适应学习速率,使网络的训练在不同的阶段自动设置不同的学习速率。

⑤期望误差的选取

在网络的训练过程中,期望误差值也应通过对比训练后确定一个合适的值,这是相对于所需要的隐层节点数来确定的,因为较小的期望误差值是要靠增加隐含层的节点以及训练时间来获得。

5.2 模糊神经网络

5.2.1 神经网络与模糊逻辑

模糊控制利用专家经验建立模糊集、隶属函数和模糊推理规则等实现非线性、不确定复杂系统的控制。神经网络控制则利用其学习和自适应能力实现非线性的控制和优化。这两种控制方式和手段在许多难以用精确数学模型表示的系统控制中发挥了巨大的作用。

然而,神经网络无法处理语言变量,也不可能将专家的先验知识注入到神经网络控制系统的设计中,从而使得原来并不属于“黑箱”结构的系统设计问题只能用“黑箱”系统设计理论来进行。而在模糊控制系统设计中,人们容易将专家的知识化为模糊控制规则,这使模糊逻辑在人们经常碰到的不确定性系统的控制中显示出优势;但由于专家知识的局限性以及环境的可变性,任何一个专家都无法得到一个最佳的规则或最优的隶属度函数。而神经网络擅长于在海量数据中寻找特定的模式,可以用神经网络来辨识因果关系,通过在输入和输出数据中找出模式而生成模糊逻辑规则。所以,这两种技术具有互补性。

因此,利用神经网络的学习功能来优化模糊控制规则和相应的隶属函数,将一些知识预先分布到神经网络中去,是提出模糊神经网络理论的两个基本出发点。模糊控制思想与神经网络学习能力的结合,使得模糊控制规则和隶属函数可以通过对样本数据的学习自动地生成,克服了人为选择模糊控制规则主观性较大的缺陷。

5.2.2 模糊神经网络

(1) 模糊神经网络的结构

这里介绍一种用多层前向传播神经网络逼近的模糊神经网络系统,如图 5.6 所示。

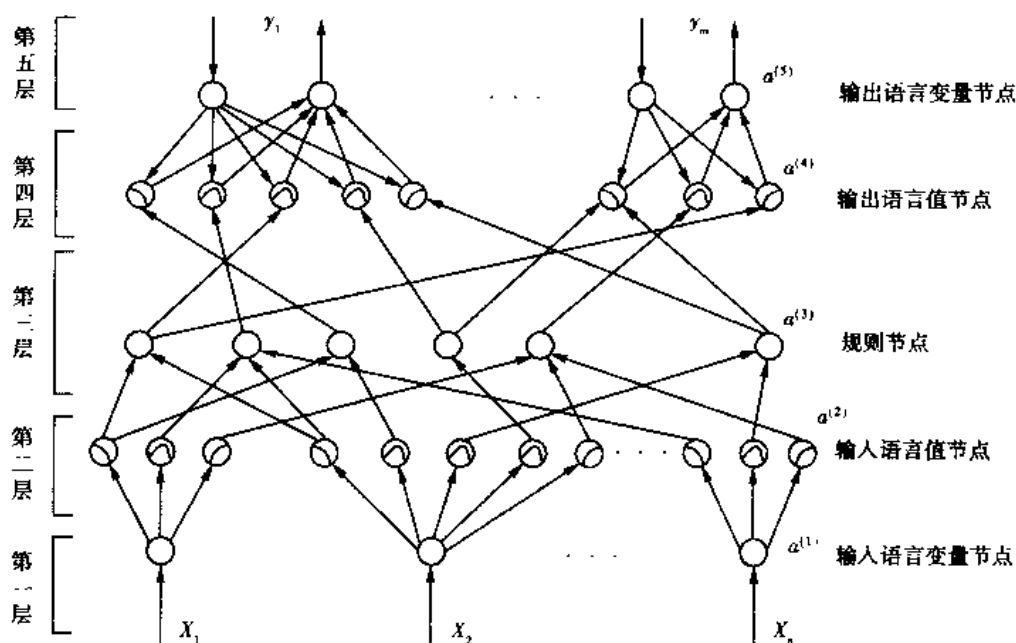


图 5.6 模糊神经网络的结构图

第一层为输入层,节点用来表示输入语言变量。

最后一层是输出层,每个输出变量有两个节点:一个用于训练神经网络需要的期望输出信号的馈入,另一个表示模糊神经网络实际推理决策的输出信号。

第二层和第四层的节点称为项节点,用来表示输入和输出语言变量语言值的隶属函数。

第三层节点称为规则节点,用来实现模糊逻辑推理。

第三层与第四层节点之间的连接模型实现了连接推理工程,从而避免了传统模糊推理逻辑的规则匹配推理方法。图中箭头方向表示系统信号的走向。从下到上的信号流向表示模糊神经网络训练完成以后的正常信号流向,而从下到上表示模糊神经网络训练时所需的期望输出的反向传播信号流向。

神经元的激发函数是神经元输入函数响应 f 的函数,即

$$\text{output} = o_i^k = a(f) \quad (5.13)$$

式中, a 表示神经元的激发函数。

为了满足模糊控制的要求,对每一层的神经元函数应有不同的定义:

第一层,这一层的节点只是将输入变量值直接传送到下一层。所以

$$f_j^{(1)} = u_j^{(1)} \quad a_j^{(1)} = f_j^{(1)}$$

且输入变量与第一层节点之间的连接系数 $\omega_{ji}^{(1)} = 1$

$$u_j^{(1)} = x_j \quad j = 1, 2, \dots, n$$

第二层,如果采用一个神经元节点来实现语言值的隶属度函数变换,则这个节点的输出就可以定义为隶属度函数的输出,如钟型函数就是一个很好的隶属度函数。

$$f_j^{(2)} = M_{x_i}(m_{\mu}^{(2)}, \sigma_{\mu}^{(2)}) = -\frac{(u_i^{(2)} - m_{\mu}^{(2)})^2}{(\sigma_{\mu}^{(2)})^2}, \quad a_j^{(2)} = e^{f_j^{(2)}}$$

式中, m_{μ} 和 σ_{μ} 分别表示第 i 个输入语言变量 x_i 的第 j 个语言变量值隶属度函数的中心值和宽度。

第三层,这一层的功能是完成模糊逻辑推理条件部的匹配工作。因此,由最大、最小推理规则可知,规则节点实现的功能是模糊“与”运算,即

$$f_j^{(3)} = \min(u_i^{(3)}, u_2^{(3)}, \dots, u_p^{(3)}), \quad a_j^{(3)} = f_j^{(3)}$$

且第二层节点与第三层节点之间的连接系数 $\omega_{jk}^{(3)} = 1$ 。

第四层,在这一层上的节点有两种操作模式:一种是实现信号从上到下的传输模式;另一种是实现信号从下到上的传输模式。在从上到下的传输模式中,此节点的功能与第二层中的节点完全相同。只是在此节点上实现的是输出变量的模糊化,而第二层的节点实现的是输入变量的模糊化。这一层的主要用途是为了使模糊神经网络的训练能够实现语言化规则的反向传播学习;而在从下到上的传输模式中,实现的是模糊逻辑推理运算。根据最大、最小推理规则,这一层上的神经元实质上是模糊“或”运算,用来集成具有同样结论的所有激活规则。

$$f_j^{(4)} = \max(u_i^{(4)}, u_2^{(4)}, \dots, u_p^{(4)}), \quad a_j^{(4)} = f_j^{(4)}$$

或

$$f_j^{(4)} = \sum_{i=1}^p u_i^{(4)}, \quad a_j^{(4)} = \min(1, f_j^{(4)})$$

且第三层节点与第四层节点之间的连接系数 $\omega_{jk}^{(4)} = 1$ 。

第五层,在这一层中有两类节点:一类节点执行从上到下的信号传输方式,实现了把训练数据反馈到神经网络中去的目的,对于这类节点,其神经元节点函数定义为:

$$f_j^{(5)} = y_j^{(5)} \quad a_j^{(5)} = f_j^{(5)}$$

第二类神经元节点执行从下到上的信号传输方式,它的最终输出就是模糊神经网络的模糊推理控制输出。在这一层上的节点主要实现模糊输出的解模糊运算。下列函数可以用来模拟重心法的解模糊运算:

$$f_j^{(5)} = \sum \omega_{\mu}^{(5)} u_i^{(5)} = \sum_i (m_{\mu}^{(5)} \sigma_{\mu}^{(5)}) u_i^{(5)}, \quad a_j^{(5)} = \frac{f_j^{(5)}}{\sum_i \sigma_{\mu}^{(5)} u_i^{(5)}}$$

(2) 模糊神经网络的学习方法

混合学习算法对于以上模糊神经网络结构是非常有效的。具体包括两大部分:自组织学习阶段和有教师指导下学习阶段。为此必须首先确定和提供:

- ①初始模糊神经网络结构;
- ②输入、输出样本训练数据;
- ③输入、输出语言变量的模糊分区(如每一输入输出变量语言值的多少等)。

混合学习算法第一阶段的主要任务是进行模糊控制规则的自组织、输入输出语言变量各语言值隶属函数参数的预辨识,以得到一个符合该被控对象的合适的模糊控制规则和初步的

隶属函数分布;第二阶段的主要任务是优化隶属函数的参数以满足更高精度的要求。

1) 自组织学习阶段

首先估计覆盖在已有训练样本数据上的隶属函数域来确定现有配制的模糊神经网络结构中各语言值的隶属度函数的中心位置和宽度。隶属度函数的中心值 m_i 估计算法采用 Kohonen 的自组织映射法,宽度值 σ_i 则与重叠参数 r 以及中心点 m_i 邻域内分布函数值相关。由于 Kohonen 神经网络能够实现自组织映射,因此,如果输入样本足够多时,则输入样本与 Kohonen 输出节点之间的连接权系数经过一段时间的学习后,其分布可以近似地看做输入随机样本的概率密度分布。如果输入的样本有几种类型,则它们会根据各自的概率分布集中到输出空间的各个不同区域内。Kohonen 自组织学习算法计算隶属度函数中心值 m_i 的公式为:

$$\|x(t) - m_{\text{closest}}(t)\| = \min_{1 \leq i \leq k} \|x(t) - m_i(t)\| \quad (5.14)$$

式中,初始的 $m_i(0)$ 为一个小的随机数; $k = |T(x)|$ 表示语言变量 x 语言值的数目。

$$m_{\text{closest}}(t+1) = m_{\text{closest}}(t) + \alpha(t)[x(t) - m_{\text{closest}}(t)] \quad (5.15)$$

$$m_i(t+1) = m_i(t) \quad m_i(t) \neq m_{\text{closest}}(t) \quad (5.16)$$

式中, $\alpha(t)$ 是一个单递减的标量学习因子。

此语言变量语言值所对应的宽度 σ_i 的计算通过下列目标函数的极小值来获取的,即

$$E = \frac{1}{2} \sum_{i=1}^v \left[\sum_{j \in N_{\text{nearest}}} \left(\frac{m_i - m_j}{\sigma_i} \right)^2 - r \right]^2 \quad (5.17)$$

式中, r 为重叠参数; N 为最近邻域法的阶数。

自组织学习法只是找到语言变量的初始分类估计值,一般采用一阶最近邻域法求取:

$$\sigma_i = \frac{|m_i - m_{\text{closest}}|}{r} \quad (5.18)$$

记 $o_i^{(3)}(t)$ 为规则节点的激励强度, $o_j^{(4)}(t)$ 为第四层输出语言值节点输出,则可以通过对样本数据的竞争学习得出其模糊推理规则。再记 ω_{ij} 为第 j 个输出语言值节点与第 i 个规则节点的连接权系数,则对于每一个样本数据权值的更新公式为:

$$\Delta\omega_{ij}(t) = o_j^{(4)}[-\omega_{ij}(t) + o_i^{(3)}] \quad (5.19)$$

在极端的情况下,如果第四层的神经元是一个阈值函数,则上述算法就退化为只有胜者才能学习的一个简单学习公式。

为了简化神经网络的结构,可以再通过规则结合的办法来减少系统总的规则数,如果:

- ①该组节点具有完全相同的结论部;
- ②在该组规则节点中某些条件部是相同的;
- ③该组规则节点的其他条件输入项包含了所有其他输入语言变量的某一语言值节点的输出。

当存在一组规则节点满足以上三个条件,则可以用具有惟一相同条件部的一个规则节点来代替这一组规则节点。图 5.7 给出了规则节点合成的一个例子。

2) 有教师指导下的学习阶段

此阶段利用训练样本数据实现输入、输出语言变量各语言值隶属函数的最佳调整,同时,它也为模糊神经网络的在线学习提供保证。有教师指导下的模糊神经网络学习问题可以这样来描述:给定的训练样本数据 $x_i(t)$, $i=1,2,\dots,n$, 期望的输出样本值 $y_i(y)$, $i=1,2,\dots,m$, 模糊分区 $|T(x)|$ 和 $|T(y)|$ 以及模糊逻辑控制规则。有教师学习过程的实质是最优地调整隶属

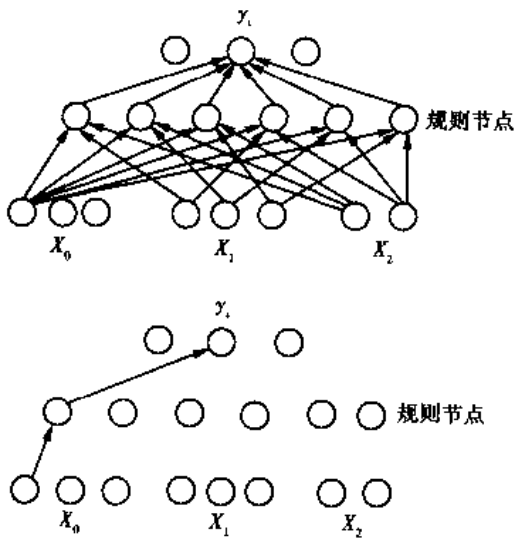


图 5.7 规则节点合并示例

度函数的参数($m_{\mu}^{(2)}, \sigma_{\mu}^{(2)}, m_{\mu}^{(5)}, \sigma_{\mu}^{(5)}$)的过程。

模糊控制规则确定以后,学习的任务就是调整隶属度函数的参数,以满足更高精度的要求。因此,有教师学习算法也可以套用传统的反向传播学习算法的思想,取学习指标函数:

$$e = \frac{1}{2} [y(t) - \hat{y}(t)]^2 = \min \quad (5.20)$$

式中, $y(t)$ 是当前时刻的期望系统输出; $\hat{y}(t)$ 是当前时刻的模糊神经网络实际输出。

对于每一个样本数据对,从输入节点开始通过前向传播计算出各节点的输出值,然后再从输出节点开始使用反向传播计算出所有隐含节点的偏导数,广义学习规则为:

$$\Delta \omega \propto - \frac{\partial e}{\partial \omega}$$

$$\omega(t+1) = \omega(t) + \eta \left(- \frac{\partial e}{\partial \omega} \right) \quad (5.21)$$

式中, η 为学习速率或步长。

$$\frac{\partial e}{\partial \omega} = \frac{\partial e}{\partial (net)} \cdot \frac{\partial (net)}{\partial \omega} = \frac{\partial e}{\partial f} \cdot \frac{\partial f}{\partial \omega} = \frac{\partial e}{\partial f} \cdot \frac{\partial f}{\partial a} \cdot \frac{\partial a}{\partial \omega} \quad (5.22)$$

下面将详细给出图 5.6 所表示模糊神经网络结构各层连接系数的反向传播算法(注:这里讨论的隶属度函数为钟型函数,其中 m_i, σ_i 为可调参数)。

中心值 $m_{\mu}^{(5)}$ 的更新公式为:

$$m_{\mu}^{(5)}(t+1) = m_{\mu}^{(5)}(t) + \eta [y(t) - \hat{y}(t)] \frac{\sigma_{\mu}^{(5)} u_i^{(5)}}{\sum_i \sigma_{\mu}^{(5)} u_i^{(5)}} \quad (5.23)$$

隶属函数宽度 $\sigma_{\mu}^{(5)}$ 的更新公式:

$$\sigma_{\mu}^{(5)}(t+1) = \sigma_{\mu}^{(5)}(t) + \eta [y(t) - \hat{y}(t)] \frac{m_{\mu}^{(5)} u_i^{(5)} (\sum_i \sigma_{\mu}^{(5)} u_i^{(5)}) - (\sum_i m_{\mu}^{(5)} \sigma_{\mu}^{(5)} u_i^{(5)}) u_i^{(5)}}{(\sum_i \sigma_{\mu}^{(5)} u_i^{(5)})^2} \quad (5.24)$$

系统输出误差反向传播到上一层的广义误差 $\delta^{(5)}$ 为:

$$\delta_j^{(5)} = - \frac{\partial e}{\partial f_j^{(5)}} = - \frac{\partial e}{\partial a_j^{(5)}} \cdot \frac{\partial a_j^{(5)}}{\partial f_j^{(5)}} = y(t) - \hat{y}(t) \quad (5.25)$$

第四层:在从下向上的传输模式中,广义误差信号 $\delta_j^{(4)}$ 为:

$$\delta_j^{(4)} = \frac{\partial e}{\partial f_j^{(4)}} = \frac{\partial e}{\partial f_j^{(5)}} \cdot \frac{\partial f_j^{(5)}}{\partial f_j^{(4)}} = \frac{\partial e}{\partial f_j^{(5)}} \cdot \frac{\partial f_j^{(5)}}{\partial u_j^{(5)}} \cdot \frac{\partial u_j^{(5)}}{\partial f_j^{(4)}} = \frac{\partial e}{\partial f_j^{(5)}} \cdot \frac{\partial}{\partial u_j^{(5)}} \left[\frac{\sum_i m_{\mu}^{(5)} \sigma_{\mu}^{(5)} u_i^{(5)}}{\sum_i \sigma_{\mu}^{(5)} u_i^{(5)}} \right] =$$

$$[y(t) - \hat{y}(t)] \frac{m_j^{(5)} \sigma_j^{(5)} (\sum_i \sigma_{ji}^{(5)} u_i^{(5)}) - (\sum_i m_j^{(5)} \sigma_{ji}^{(5)} u_i^{(5)}) \sigma_j^{(5)}}{(\sum_i \sigma_{ji}^{(5)} u_i^{(5)})^2} \quad (5.26)$$

第三层广义误差信号 $\delta_j^{(3)}$ 的计算公式为:

$$\delta_j^{(3)} = \frac{\partial e}{\partial f_j^{(3)}} = \frac{\partial e}{\partial a_j^{(3)}} \cdot \frac{\partial a_j^{(3)}}{\partial f_j^{(3)}} = \frac{\partial e}{\partial f_j^{(4)}} \cdot \frac{\partial f_j^{(4)}}{\partial a_j^{(3)}} = \delta_j^{(4)} \frac{\partial f_j^{(4)}}{\partial u_i^{(3)}} = \delta_j^{(4)} \quad (5.27)$$

如果输出语言变量有 m 个,则

$$\delta_j^{(3)} = \sum_{k=1}^m \delta_k^{(4)} \quad (5.28)$$

第二层输入语言变量各语言值隶属度函数中心值 $m_j^{(2)}$ 的学习公式为:

$$m_j^{(2)}(t+1) = m_j^{(2)}(t) - \eta \frac{\partial e}{\partial a_j^{(2)}} e^{f_j^{(2)}} \frac{2(u_i^{(2)} - m_j^{(2)})}{(\sigma_j^{(2)})^2} \quad (5.29)$$

输入语言变量各语言值隶属度函数宽度值 $\sigma_j^{(2)}$ 的学习公式为:

$$\sigma_j^{(2)}(t+1) = \sigma_j^{(2)}(t) - \eta \frac{\partial e}{\partial a_j^{(2)}} e^{f_j^{(2)}} \frac{2(u_i^{(2)} - m_j^{(2)})^2}{(\sigma_j^{(2)})^3} \quad (5.30)$$

整个混合学习过程的流程可以用图 5.8 表示。由于混合学习算法在第一阶段已经进行了大量的自组织学习训练,因此,第二阶段有教师学习的 BP 学习算法通常比常规的 BP 算法收敛要快。上面推导出来的学习算法是针对第二层中用单一神经元来实现语言值的隶属度函数,但它可以扩展到用子神经网络逼近的隶属度函数的情形。利用反向传播的思想,可以将输出误差信号反传到子网络中去,从而可以实现子网络参数的学习和调整。

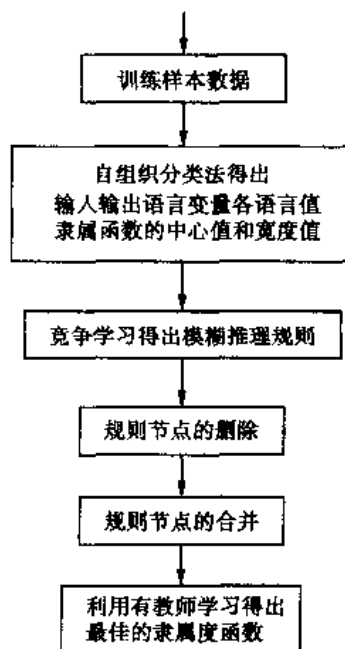


图 5.8 模糊神经网络混合学习的流程图

5.3 基于神经网络的模糊控制器

5.3.1 复合型神经模糊控制器

日本高木敏幸提出的基于神经网络的模糊控制器,其系统结构如图 5.9 所示。该系统由神经网络 1、2,判断机构 1、2 和模糊控制器等五部分组成。

(1) 神经网络 1

这是用于对控制对象的输出波形进行分类,相当于大脑对全局的判断。

1) 预处理

预处理将控制对象的输出波形分成七类,并用误差反向传播学习算法来让神经网络学习

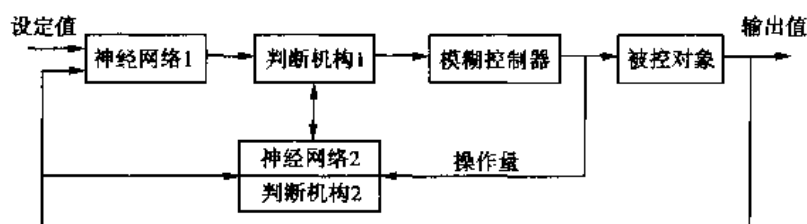


图 5.9 系统结构

这些波形。对目标只有很小的定常偏差和在允许范围内波动的所期望的输出波形称做 A 类波形；发散振荡的输出波形称做 B 类波形；对目标值成收敛振荡但收敛很慢的输出波形称做 C 类波形；对目标值成渐进收敛但收敛很慢的输出波形称做 D 类波形；对目标值超调很多并且有很大定常偏差的输出波形称做 E 类波形；对目标值欠调很多并且有很大反向定常偏差的输出波形称做 F 类波形；而其他波形都归为 G 类波形。

2) 完成的功能

神经网络 1 根据对波形的分类,向判断机构 1 发送以下不同命令:

- ①对 A 类波形:再次构造控制规则;
- ②对 B 类波形:大幅度减小控制量;
- ③对 C 类波形:减小一点控制量;
- ④对 D 类波形:增加一点控制量;
- ⑤对 E 类波形:大幅度减小控制量;
- ⑥对 F 类波形:大幅度增加控制量;
- ⑦对 G 类波形:增加波形分类。

(2) 神经网络 2

用于学习被控对象的动态特性,建立被控对象动态特性的数学模型,以对被控对象进行仿真。在这里也需要进行一些预处理,并根据预处理结果完成一些相应的功能。

1) 预处理

根据被控对象的输入波形和输出波形,用误差反向传播学习算法通过神经网络 2 来学习被控对象的动态特性。

2) 完成的功能

神经网络 2 通过学习建立被控对象动态特性的模型,以让判断机构 2 可以利用这些有关控制对象行为的知识。例如,利用神经网络 2 就可以知道,要得到控制对象某种输出波形,应该输入怎样的输入波形。如果不能用当前的模糊控制器中的控制规则直接控制对象时,可以把学习后的神经网络 2 当做控制对象,用网络来模拟控制对象。

(3) 判断机构 1

用于判断是否进行控制规则的调整或增加。

(4) 判断机构 2

用于规定控制规则后件的模糊变量和校正量。

(5) 模糊控制器控制规则的构成步骤

①用判断机构 1 确定最初的控制输入 在构成模糊控制器的最初阶段模糊控制器内没有任何控制规则,因此,为了构成最初的规则,就需要向控制对象输入某适当的控制输入值,以观

测系统的响应。

②发散波形处理 当以上决定的控制量使控制对象的输出波形发散时,根据神经网络1的波形分类,判断机构1大幅度减小控制量。

③收敛波形处理 当以上决定的控制量使控制对象的输出波形收敛时,判断机构2观察该波形的收敛时间,把1/2的收敛时间称做 t_h 。

④当控制对象输出波形的各个取样时刻 $t_i < t_h$ 时,判断机构2检测它们与目标值的偏差 E_i 及其变化量 ΔE_i ,根据这些值决定以下各个量:

$$E_{\max} = \max |E_i|$$

$$\Delta E_{\max} = \max |\Delta E_i|$$

将 E_{\max} 和 ΔE_{\max} 分别乘上常数,即为 αE_{\max} 和 $\beta \Delta E_{\max}$,将它放在控制规则的前件作为 E 和 ΔE 模糊变量的隶属函数规格化常数。

⑤决定控制规则后件模糊变量隶属函数的规格化常数 为了决定控制规则后件模糊变量隶属函数的规格化常数,将学习以后的神经网络2输出的控制对象期望值输入,得到期望的控制输入波形。把这个输入波形在各个取样时刻的取样值写成 $U'_2(t_i)$ 或 $U'(t_i < t_h)$,而在当前实际输入控制对象的控制量在各个取样时刻的值写成 $U''_2(t_i)$ 或 $U''(t_i < t_h)$ 。这两者的差 $U_i(t_i) = U'(t_i) - U''(t_i)$ 就表示在各个取样时刻控制量过大或者是过小。

利用这一个差 $U_i(t_i)$,按照神经网络1进行的波形分析,就可以决定控制规则后件模糊变量的隶属函数的规格化常数。

⑥根据规则前件求后件 如果系统的控制规则形式如“if E is A_1 and ΔE is A_2 then u is B_i ”,而且,控制规则的前件和后件的模糊变量有7个隶属函数PB、PM、PS、ZE、NS、NM和NB。为了从前件的模糊变量标称来决定后件的模糊变量标称,可以把这些标称数值化:PB = 3, PM = 2, PS = 1, ZE = 0, NS = -1, NM = -2和NB = -3。用这些数值就可以决定后件模糊变量标称。

⑦根据波形分类分别处理 判断机构1将控制对象的输出波形送到神经网络1,以对波形进行分类。如果是A类输出波形,说明不需要进行调整,就转入第⑧步处理;若输出波形不是A类,就按照不同的类别进行不同的处理,返回第④步去调整控制量。

⑧生成最初的控制规则 由此便生成最初的控制规则,控制规则的次序则由判断机构1根据以下准则决定:

当取样时刻 $t_i \leq t_h$ 时,其步骤为:

A. 找出 $\Delta E_i = ZE$ 时与目标值的偏差 E_i ,从这些 E_i 中再找出最大的 E_j ,向判断机构2发送指令,以便构成对应于这一取样时刻 j 输出波形状态的控制规则。如果对于这一状态已经构成规则,就向判断机构2发送指令,以构成对应于 E_i 中第二大值状态的规则。

B. 找出 $E_i = ZE$ 时的 ΔE_i ,从这些 ΔE_i 中找出最大的 ΔE_j ,向判断机构2发送指令,以便构成对应于这一取样时刻 j 输出波形状态的控制规则。如果对于这一状态已经构成规则,就向判断机构2发送指令,以构成对应于 ΔE_i 中第二大值状态的规则。

由此反复交替使用这两个步骤,让判断机构2生成控制规则。如果要生成不满足上而步骤的规则时,则按以下步骤构成控制规则。

⑨在按步骤⑧构成规则时,前件的偏差 E_i 及其变化量 ΔE_i 的模糊变量的标称都为零,所以,只要决定了其中一个就决定了前件。由于前件模糊变量的隶属函数已由步骤④进行了规

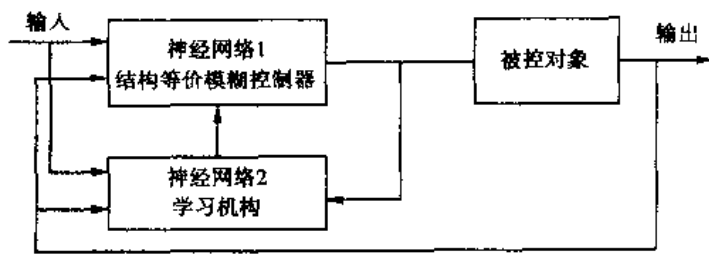


图 5.10 融合型神经模糊控制器

⑩反复进行步骤④~⑨的操作,直到没有步骤⑧的情况为止。

经过多次反复进行以上步骤,就可以自动生成控制规则。

5.3.2 融合型神经模糊控制器

如图 5.10 所示,融合型模糊控制器是以等价结构的神经网络直接组成控制器,这种控制系统由两个神经网络环节组成。其中,神经网络 1 是结构等价的模糊神经网络,用做模糊控制器;神经网络 2 是学习机构,它通过对系统控制过程的学习,去修改神经网络 1 的权系数。

5.3.3 模糊神经网络在倒立摆控制中的应用

作为智能控制研究中的一个经典对象,在倒立摆问题中应用神经网络方法,首推 Widrow 等人(1964—1987 年)的工作。较具有代表性的结果则主要是由美国加州大学柏克利分校 L. A. Zadeh 为首的“fuzzy group”做出的。如 C. C. Lee 与 H. R. Berenji 基于近似推理和再励学习的 ARIC(1989—1992 年)及 GARIC(1992—1994 年),以及 J. S. Jang 的基于自适应网络的模糊推理系统 ANFIS(1992 年)等。他们从 fuzzy logic 的角度分别发展了 Barto(1983 年)与 Anderson(1989 年)等人以及 Werbos(1990 年)等人的结果,而 Barto 特别是 Anderson(1989 年)后来的结果,则是研究有关倒立摆再励学习控制的早期经典之作。

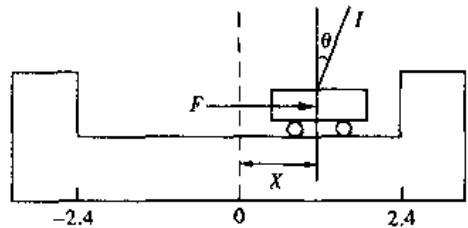


图 5.11 倒立摆示意图

图 5.11 给出了一级倒立摆的示意图。

考虑摩擦时倒立摆的运动方程可由如下非线性微分方程描述：

$$\ddot{\theta} = \frac{g \sin \theta + \cos \theta \left[\frac{-F - m_p l \dot{\theta}^2 \sin \theta + \mu_c \operatorname{sgn}(\dot{x})}{m_c + m_p} \right] - \frac{\mu_p \dot{\theta}}{m_p l}}{l \left[\frac{4}{3} - \frac{m_p \cos^2 \theta}{m_c + m_p} \right]}$$

$$\ddot{x} = \frac{F + m_p l [\ddot{\theta}^2 \sin \theta - \ddot{\theta} \cos \theta] - \mu_c \operatorname{sgn}(\dot{x})}{m_c + m_p}$$

其中,典型数据为 $g = 9.8 \text{ m/s}^2$ (重力加速度), $m_c = 1.0 \text{ kg}$ (小车质量), $m_p = 0.1 \text{ kg}$ (杆的质量), $l = 0.5 \text{ m}$ (杆的半长), $\mu_c = 0.0005$ (小车相对于导轨的摩擦系数), $\mu_p = 0.000002$ (杆相对于小车的摩擦系数), F 为作用于小车上的力,它相当于模糊神经网络控制器的输出。其他符

号与通常约定相同。

若令 $a_1 = (m_p/m)l \dot{\theta}^2 \sin\theta$, $a_2 = (m_p/m)l \ddot{\theta} \cos\theta$, $a = \ddot{x}$, $a_f = (1/m)F$, 这里, $m = m_c + m_p$, 则容易得到如下倒立摆方程的简化形式:

$$\ddot{\theta} = \frac{3}{4l}(g \sin\theta - a \cos\theta - \frac{1}{m_p l} \mu_p \dot{\theta})$$

$$\ddot{x} = a_f + a_1 - a_2 - \frac{1}{m} \mu_c \operatorname{sgn}(\dot{x})$$

进一步地, 设 $x = [\theta \ \dot{\theta} \ \dot{x}]^T$, 则有如下状态方程为:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = f_1 = \frac{3}{4l}(g \sin x_1 - a \cos x_1 - \frac{1}{m_p l} \mu_p x_2)$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = f_2 = a_f + a_1 - a_2 - \frac{1}{m} \mu_c \operatorname{sgn}(x_4)$$

式中, $f_1 = \ddot{\theta}$, $f_2 = a$ 。

在上述动态方程的仿真中, 采用了自动变步长 Runge-Kutta-Fehlberg 数值积分法。采样周期 $T = 20\text{ms}$ 。失败状态定义为 $|\theta| > 12^\circ$ 或 $|x| > 2.4\text{m}$ 。初始条件为 $\theta(t_0) = 2^\circ$, 其他均为 0。

本例只是从概念上说明模糊神经网络在自动控制中的应用。

第 6 章

模糊控制系统的开发

模糊控制系统的设计开发归结起来可分成两大部分：一是硬件系统设计，包括被控参数的检测、放大、A/D 转换、D/A 转换、执行部件等过程输入输出通道设计和模糊控制器设计，模糊控制器可选择数字单片机、模糊单片机和 PLC 等可编程控制器；二是软件设计，包括模糊控制算法设计及程序实现。另外，可用微机实现对模糊控制器和控制现场的通信和管理。

6.1 开发方法——恒温箱温度的模糊控制

6.1.1 控制对象简述

被控对象是恒温箱，恒温箱通过电压 220V 额定功率为 2000W 的电阻炉加热，采用热电偶检测温度和双向可控硅调节电阻炉加热功率。

6.1.2 模糊控制器的结构设计

模糊控制器采用如下两个模糊变量（如图 6.1 所示）：

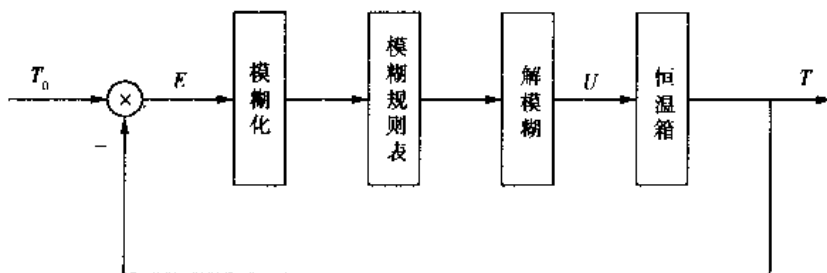


图 6.1 炉温模糊控制系统的原理示意图

图中： E ——温度误差， $E = T_0 - T$ ， T_0 是温度给定值，本例设为 100°C ， T 是实测值，设定恒温箱温度的模糊控制区为 $85 \sim 115^{\circ}\text{C}$ ，从室温到 85°C 和超过 115°C 为开关控制区或者人工控制区，可快速升温或降温，这时温差范围为 $\pm 15^{\circ}\text{C}$ 。

U ——双向可控硅控制角的增量，可根据实际情况而定，本例取 $\pm 45^{\circ}$ 。

其中, E 是输入变量, U 是输出变量。模糊控制器采用 AT89C51 单片机, 用单片机汇编语言实现模糊控制算法, 采样间隔为 1s。

6.1.3 模糊变量的论域及其隶属函数

把误差(E)论域量化为 7 级, 每级相差 5°C , 即 $\{-3, -2, -1, 0, +1, +2, +3\}$ 一般按对称方式取级数, 如果误差范围不是对称的, 则按以下公式转化为对称级数:

$$y = \frac{2n}{b-a} \left(x - \frac{a+b}{2} \right) \quad (6.1)$$

例如, 本例把温度范围 $[85, 115]$ 等价映射到对称的级数范围 $[-3, +3]$ 。误差变量 E 的模糊子集选取如下词集共 7 档: $\{\text{PB}, \text{PM}, \text{PS}, \text{Z}, \text{NS}, \text{NM}, \text{NB}\}$ 误差变量 E 的隶属度赋值如表 6.1。

表 6.1 误差变量 E 的隶属度赋值

E	-3	-2	-1	0	+1	+2	+3
PB	0	0	0	0	0.2	0.5	1
PM	0	0	0	0.2	0.5	1	0.5
PS	0	0	0.2	0.5	1	0.5	0.2
Z	0	0.2	0.5	1	0.5	0.2	0
NS	0.2	0.5	1	0.5	0.2	0	0
NM	0.5	1	0.5	0.2	0	0	0
NB	1	0.5	0.2	0	0	0	0

把控制量 U 的论域量化为 7 级, 每级相差 15°C , 即 $\{-3, -2, -1, 0, +1, +2, +3\}$, 控制量的模糊子集选取如下词集共 7 档: $\{\text{PB}, \text{PM}, \text{PS}, \text{Z}, \text{NS}, \text{NM}, \text{NB}\}$, 控制量 U 的隶属度赋值如表 6.2 所示。

表 6.2 控制量 U 的隶属度赋值

U	-3	-2	-1	0	+1	+2	+3
PB	0	0	0	0	0.2	0.5	1
PM	0	0	0	0.2	0.5	1	0.5
PS	0	0	0.2	0.5	1	0.5	0.2
Z	0	0.2	0.5	1	0.5	0.2	0
NS	0.2	0.5	1	0.5	0.2	0	0
NM	0.5	1	0.5	0.2	0	0	0
NB	1	0.5	0.2	0	0	0	0

6.1.4 根据实际操作经验给出以下控制规则

恒温箱的温度控制规则如下:

R_1 : if $E = \text{NB}$ then $U = \text{PB}$

R_2 : if $E = \text{NM}$ then $U = \text{PM}$

R_3 : if $E = \text{NS}$ then $U = \text{PS}$

R_4 : if $E = \text{Z}$ then $U = \text{Z}$

R_5 : if $E = \text{PS}$ then $U = \text{NS}$

R_6 : if $E = \text{PM}$ then $U = \text{NM}$

R_7 : if $E = \text{PB}$ then $U = \text{NB}$

从上述控制规则可以看出:如果实际温度小于给定值,则应加大可控硅导通角,差值越大,导通角变大越多,但不是线性的,反之亦然。

以上控制规则可归纳为 $E \times U$ 的 Fuzzy 关系 R :

$$R = R_1 \cup R_2 \cup R_3 \cup R_4 \cup R_5 \cup R_6 \cup R_7 \quad (6.2)$$

其中,求 $R_i (i=1,2,3,\dots,7)$ 的运算按下式进行:

$$R_i = E \times U = E^T \circ U \quad (6.3)$$

所以:

$$R_1 = (\text{NB})_E \times (\text{PB})_U = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.2 & 0.5 & 1 \\ 0 & 0 & 0 & 0 & 0.2 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

$$R_2 = (\text{NM})_E \times (\text{PM})_U = \begin{bmatrix} 0 & 0 & 0 & 0.2 & 0.5 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0.2 & 0.5 & 1 & 0.5 \\ 0 & 0 & 0 & 0.2 & 0.5 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

$$R_3 = (\text{NS})_E \times (\text{PS})_U = \begin{bmatrix} 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0 & 0 & 0.2 & 0.5 & 0.5 & 0.5 & 0.2 \\ 0 & 0 & 0.2 & 0.5 & 1 & 0.5 & 0.2 \\ 0 & 0 & 0.2 & 0.5 & 0.5 & 0.5 & 0.2 \\ 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

$$R_4 = (Z)_E \times (Z)_U = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 \\ 0 & 0.2 & 0.5 & 0.5 & 0.5 & 0.2 & 0 \\ 0 & 0.2 & 0.5 & 1 & 0.5 & 0.2 & 0 \\ 0 & 0.2 & 0.5 & 0.5 & 0.5 & 0.2 & 0 \\ 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

$$R_5 = (PS)_E \times (NS)_U = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 \\ 0.2 & 0.5 & 0.5 & 0.5 & 0.2 & 0 & 0 \\ 0.2 & 0.5 & 1 & 0.5 & 0.2 & 0 & 0 \\ 0.2 & 0.5 & 0.5 & 0.5 & 0.2 & 0 & 0 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

$$R_6 = (PM)_E \times (NM)_U = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0.5 & 0.2 & 0 & 0 & 0 \\ 0.5 & 1 & 0.5 & 0.2 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0.5 & 0.2 & 0 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

$$R_7 = (PB)_E \times (NB)_U = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0.2 & 0 & 0 & 0 & 0 \\ 1 & 0.5 & 0.2 & 0 & 0 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

于是:

$$R = R_1 \cup R_2 \cup \cdots \cup R_6 \cup R_7 = \begin{bmatrix} 0 & 0 & 0.2 & 0.2 & 0.5 & 0.5 & 1 \\ 0 & 0.2 & 0.2 & 0.5 & 0.5 & 1 & 0.5 \\ 0.2 & 0.2 & 0.5 & 0.5 & 1 & 0.5 & 0.5 \\ 0.2 & 0.5 & 0.5 & 1 & 0.5 & 0.5 & 0.2 \\ 0.5 & 0.5 & 1 & 0.5 & 0.5 & 0.2 & 0.2 \\ 0.5 & 1 & 0.5 & 0.5 & 0.2 & 0.2 & 0 \\ 1 & 0.5 & 0.5 & 0.2 & 0.2 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

6.1.5 控制输出

得到系统模糊控制规则 R 后,当采样得到偏差 E 作为模糊控制器的输入,就可得出相应的控制量 U :

$$U = E \circ R \quad (6.4)$$

若采样得到偏差:

$$E = PS = [0, 0, 0.2, 0.5, 1, 0.5, 0.2]$$

则控制量 U 可计算如下:

$$U = E \circ R = [0 \quad 0 \quad 0.2 \quad 0.5 \quad 1 \quad 0.5 \quad 0.2]_{1 \times 7} \circ$$

$$\begin{bmatrix} 0 & 0 & 0.2 & 0.2 & 0.5 & 0.5 & 1 \\ 0 & 0.2 & 0.2 & 0.5 & 0.5 & 1 & 0.5 \\ 0.2 & 0.2 & 0.5 & 0.5 & 1 & 0.5 & 0.5 \\ 0.2 & 0.5 & 0.5 & 1 & 0.5 & 0.5 & 0.2 \\ 0.5 & 0.5 & 1 & 0.5 & 0.5 & 0.2 & 0.2 \\ 0.5 & 1 & 0.5 & 0.5 & 0.2 & 0.2 & 0 \\ 1 & 0.5 & 0.5 & 0.2 & 0.2 & 0 & 0 \end{bmatrix}_{7 \times 7}$$

$$= [0.5 \quad 0.5 \quad 1 \quad 0.5 \quad 0.5 \quad 0.5 \quad 0.5]_{1 \times 7}$$

其结果按 Zadeh 记号表示为:

$$U = \frac{0.5}{-3} + \frac{0.5}{-2} + \frac{1}{-1} + \frac{0.5}{0} + \frac{0.5}{+1} + \frac{0.5}{+2} + \frac{0.5}{+3} \quad (6.5)$$

因此,“-1”级的隶属度最大,按最大隶属度法,控制量应取“-1”级,在本例中,可控硅导通角应减小 15° ,以克服温度偏高。按上述运算方法,可求出控制表,如表 6.3 所示,将此表存入单片机存储器,然后单片机采样温度计算偏差 E ,查表获取控制量。

表 6.3 控制表

E	-3	-2	-1	0	+1	+2	+3
U	+3	+2	+1	0	-1	-2	-3

6.2 数字单片机实现模糊控制

目前,由于模糊单片机的价格还相当昂贵,故模糊控制大多用数字单片机来实现。以本章 6.1 节恒温箱的温度模糊控制为例,硬件原理框图如图 6.2 所示,数字单片机采用 AT89C51 8 位 51 系列单片机,晶振频率采用 11.0592MHz 的晶振。

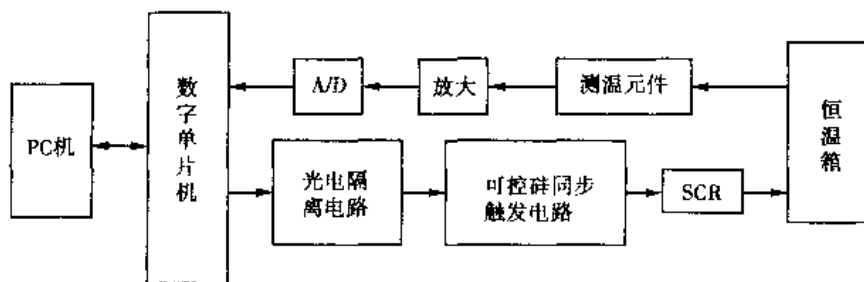


图 6.2 单片机模糊控制系统的硬件框图

本例采用单输入单输出方式,控制规则简单,设计程序时,可编写6个子程序来完成,基本思路是:温度的比较转化为数字量的比较,导通角的控制实际上是用单片机发出周期性脉冲,导通角如果为 90° ,由于220V 50Hz交流电的周期是0.02s,在交流电过零点后0.005s发出一定脉宽(如1ms)的脉冲,实现导通角 90° ,采样间隔设为1s;由于目标温度是 100°C ,在初始加热时,如果采样温度在 95°C 以下,则调用子程序0,用全功率加热;如果采样温度等于或者超过 115°C ,则调用子程序1,可控硅导通角减少 45° ;如果采样温度在 $110\sim 115^\circ\text{C}$ 之间,则调用子程序2,可控硅导通角减少 30° ;如果采样温度在 $105\sim 110^\circ\text{C}$ 之间,则调用子程序3,可控硅导通角减少 15° ;按照模糊控制规则以此类推。以上是只用温度误差 E 作输入的控制方式,若用误差 E 和误差变化率 C 作为系统输入,可采用如下方法设计控制方式。

$$U = - \langle aE + (1 - a)C \rangle \quad (6.6)$$

式中, E 是当前温差, C 是当前温差变化率, U 是可控硅导通角的增量, a 是 E 和 C 的加权调整参数,取值范围是0~1之间,运算符“ $\langle a \rangle$ ”表示取大于或者等于 $|a|$ 的最小整数,符号不变。

自组织模糊控制器的控制原理图和单片机构成的硬件框图如图6.2所示。

温度控制目标设为 100°C ,模糊控制区设为 $85\sim 115^\circ\text{C}$,从室温到 85°C 和超过 115°C 为开关控制区,温差范围为 $\pm 15^\circ\text{C}$,温差范围模糊化为 ± 3 级,即 $\{-3, -2, -1, 0, +1, +2, +3\}$,每级表示 5°C ,采样周期取10s,温差变化率 C 范围取 $\pm 3^\circ\text{C}/10\text{s}$,温差变化率范围也模糊化为 ± 3 级,即 $\{-3, -2, -1, 0, +1, +2, +3\}$,每级表示 $1^\circ\text{C}/10\text{s}$,控制量可控硅导通角增量 U 变化范围设为 $\pm 45^\circ$,1个控制量增量表示 15° ,这时,控制量模糊增量 $U = - \langle aE + (1 - a)C \rangle$,采用以上计算方法,可得到本例的控制表如下:

表 6.4 控制表

U		E						
		-3	-2	-1	0	+1	+2	+3
C	-3	+3	+2	+2	+1	0	-1	-1
	+2	+3	+2	+1	+1	0	-1	-2
	+1	+2	+2	+1	0	0	-1	-2
	0	+2	+1	+1	0	-1	-1	-2
	+1	+2	+1	0	0	-1	-2	-2
	+2	+2	+1	0	-1	-1	-2	-3
	+3	+1	+1	0	-1	-2	-2	-3

离线计算控制表的程序设计方法:

首先,把计算得到的控制表变成存储器易存储的表格,即分别把 E 和 C 的 ± 3 级变成 $\{0, 1, 2, 3, 4, 5, 6\}$,其他不变,如表6.5所示。

然后,每个控制量 U 的存储单元实际地址为:表首地址 + $E + C \times 7$,1个控制量用一个存储单元,即1个字节存放,字节的最高位为符号位,1表示负,0表示正。

根据采样温度计算得到的温差 E 和温差变化率 C 可直接判断为 $\{0, 1, 2, 3, 4, 5, 6\}$ 范围内,然后根据 E 和 C 直接查表求得控制量增量 U ,进而控制可控硅导通角即可。

表 6.5 控制表 2

U		E						
		0	1	2	3	4	5	6
C	0	+3	+2	+2	+1	0	-1	-1
	1	+3	+2	+1	+1	0	-1	-2
	2	+2	+2	+1	0	0	-1	-2
	3	+2	+1	+1	0	-1	-1	-2
	4	+2	+1	0	0	-1	-2	-2
	5	+2	+1	0	-1	-1	-2	-3
	6	+1	+1	0	-1	-2	-2	-3

由表 6.4 可知,根据采样计算得到的温差 E 和温差变化率 C 可直接计算控制量 U 。在这个方法中主要是解决避免小数运算的问题,其方法是:将产生小数的调整参数 a ($0 \sim 1$ 范围的 1 位小数)乘以 10 变成 $0 \sim 10$ 范围内的整数,即把公式 $U = - < aE + (1 - a)C >$ 变成:

$$U = - < 10 \times a \times E + (10 - 10 \times a)C > \quad (6.7)$$

根据温差 E 、温差变化率 C 和 $10 \times a$ 计算“ $< >$ ”内的值后,将该值除以 10,若余数不为 0,将商取绝对值加 1,可得到控制量 U ;若余数为 0,由商也可得到控制量 U 。

6.3 模糊单片机实现模糊控制

美国 Neuralogix 公司生产的模糊单片机 NLX230 模糊推理速度每秒达 3000 万条规则。它采用 Mamdani 法的极小极大合成运算进行推理,是一种有固定推理方式的模糊单片机。

RST	1		40	V_{DD}
DI_0	2		39	DO_0
DI_1	3		38	DO_1
DI_2	4		37	DO_2
DI_3	5		36	DO_3
DI_4	6		35	DO_4
DI_5	7		34	DO_5
DI_6	8		33	DO_6
DI_7	9		32	DO_7
V_{SS}	10	NLX230	31	V_{SS}
SK	11		30	MA_0
CS	12		29	MA_1
DI	13		28	MA_2
DO	14		27	STB
M/S	15		26	CLK
R/W	16		25	X0
NC	17		24	X1
NC	18		23	NC
NC	19		22	NC
V_{CC}	20		21	NC

图 6.3 NLX230 的引脚图

6.3.1 NLX230 的引脚及功能

NLX230 是 40 引脚的双列直插式集成芯片,如图 6.3 所示。

各引脚的功能简要介绍如下:

$DI_0 \sim DI_7$: 8 位数据输入端,在 NLX230 中,允许 8 个 8 位输入,哪一个数据进入哪一个模糊化器,由 $MA_0 \sim MA_2$ 控制输入选择器选择,由选通信号 STB 进行同步。

$DO_0 \sim DO_7$: 8 位数据输出端,输出寄存器由 $MA_0 \sim MA_2$ 选择并由 STB 同步。

STB: 选通信号,它和 $MA_0 \sim MA_2$ 相结合,用于选通输入和输出的数据。

$MA_0 \sim MA_2$: 多路地址信号,它们用于选择输入端和输出端。

SK: 串行时钟,用于把 NLX230 的配置数据送入 NLX230。

在 $M/S=0$ 时,称为从方式,这时,在 SK 同步之下,外部主计算机系统通过 DI 端下装数据到 NLX230,或者通过 DO 端从 NLX230 中读取数据。在 $M/S=1$ 时,称为主方式,这时,在 SK 同步下,NLX230 通过 DI 端读取外部 EEPROM 93C56 的内容。

CS:片选信号,低电平有效,用于启动 EEPROM 以进行数据串行读写。

DI:串行数据输入端。

DO:串行数据输出端。

M/S:主从方式设置端, $M/S=1$ 时,处于主方式; $M/S=0$ 时,处于从方式。

R/W:在从方式中,用做读写信号的输入端, $R/W=1$ 时,表示读; $R/W=0$ 时,表示写。主方式中无用。

XI:石英振荡器输入信号端。

XO:石英振荡器输出信号端。

CLK:内部振荡器的缓冲器输出。

RST:复位信号输入端,低电平有效。

V_{DD} : +5V 电源。

V_{SS} :电源地,NC:无用端,通常接地。

6.3.2 NLX230 的工作原理和功能特点

设 $y(k+1)$ 是第 $k+1$ 时刻的输出, $x(k+1)$ 是 $k+1$ 时刻的输入, R 是输入输出关系,其推理过程可用关系方程表示如下:

$$y(k+1) = y(k) \circ x(k+1) \circ R \quad (6.8)$$

当不考虑单片机 NLX230 的内部反馈信号时,则有:

$$y(k+1) = x(k+1) \circ R \quad (6.9)$$

(1) 隶属函数及其表示方法

在 NLX230 中,模糊量的隶属函数用项寄存器存放,每个项寄存器有 2 个字节,第一个字节存放隶属函数的中心值;第二个字节存放输入端选择信号和隶属函数的底宽。

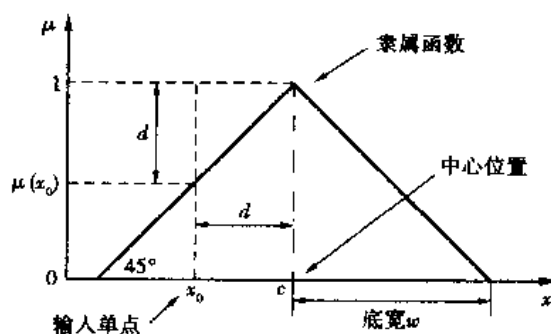


图 6.4 隶属函数的形状

(2) 控制规则的格式及意义

在 NLX230 的开发系统 ADS230 中,典型控制规则的文字描述如下:

NLX230

TERMS

Surface Temp is Cool 80 12 In

```

Surface Temp is Warm 140 12 In
Ambient Temp is NotHot 230 6 Ex
Ambient Temp is NotCold 20 20 Ex

```

OUTPUTS

```
Increased Flame 0 Ac
```

```

|
if Surface Temp is Cool and Ambient Temp is NotHot then 10
if Surface Temp is Warm and Ambient Temp is NotCold then -10
|

```

上面的控制规则文字描述称为规则集文件,开始的文字说明“NLX230”用于指明控制规则对应模糊单片机系列中的型号,规则集文件定义了控制规则所需的所有元素,也给出了控制规则。

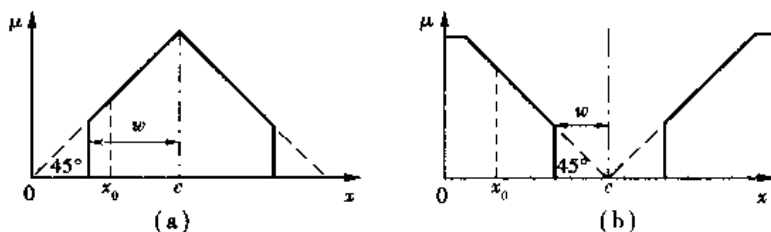


图 6.5 包含隶属函数和排斥隶属函数

(a) 包含隶属函数; (b) 排斥隶属函数

在 TERMS 段中,描述了在控制中要用到的输入物理量及其模糊量,例如,第一个句子表明输入物理量是表面温度 Surface Temp,它和模糊化器 Cool 有关。这个模糊量的中心值为 80,底宽为 12,隶属函数类型是包含型。第四个句子表明输入物理量是环境温度 Ambient Temp,它和模糊量 NotCold 有关,其中心值为 20,底宽为 20,是排斥隶属函数。

在 OUTPUTS 段中,给出了输出物理量是增加燃烧 Increased Flame,它的初始值为 0,输出值的修改方式是累加法 AC(Accumulate)。

用大括号“{}”括起来的句子是控制规则。第一条控制规则的意义为:如果表面温度 Surface Temp 为 Cool,并且环境温度 Ambient Temp 为 NotHot,那么把输出的原有值增加 10。第二条控制规则的意义为:如果表面温度 Surface Temp 为 Warm,并且环境温度 Ambient Temp 为 NotCold,那么把输出的原有值减少 10。

6.3.3 NLX230 的内部寄存器及功能

NLX230 寄存器地址至间为 00H ~ FFH,其中 00H ~ 0BH 保留,在输入时可全部清零。

(1) 输出方式寄存器(地址为 0CH)

该寄存器有 8 位,若某位为 0,则对应的 8 位输出以立即法输出,即输出值等于作用值加初始值;反之,则对应的 8 位输出以累加法输出,即输出值等于作用值加上次输出值。

(2) 输入配置寄存器(地址为 0DH)

该寄存器有 8 位,若某位为 0,则对应输入端的输入来自内部输出反馈信号;若某位为 1,则对应输入端的输入来自外部信号(DI₀ ~ DI₇)。

(3) 函数类型寄存器(地址为 0EH~0FH)

它是两个 8 位寄存器,共 16 位,对应控制 16 个模糊化器的隶属函数类型,若某位为 0,则对应模糊化器的隶属函数为包含隶属函数;反之,为排斥隶属函数。

(4) 初始值寄存器(地址为 10H~17H)

初始值寄存器有 8 个,每个有 8 位,在每个初始值寄存器中存放了用户定义的输出初始值,范围为 $-128 \sim +127$ 。在立即法中,初始值就是无需修改的额定输出值;在累加法中,初始值用于求取复位之后的第一个输出值。

(5) 规则边界寄存器(地址为 18H~1FH)

规则边界寄存器有 8 个,每个有 8 位,最高两位为 0, $D_5 \sim D_0$ 共 6 位为有效值。每一个输出会用到一定数量的连续规则,这个数量等于现行输出对应的规则边界寄存器内容与先前输出对应的规则边界寄存器的内容之差。每一个规则边界寄存器的内容就是相应输出所用到的最后一条控制规则的地址。每一个输出必须最少用到分配给它的规则数量的两条规则。应该注意的是:规则 0 是不能由用户编辑的,由于输出端有 8 个,而控制规则可达 64 条,平均每个输出端可以用 8 条规则。

(6) 项寄存器(地址为 20H~3FH)

20H	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
	MSB							LSB
21H	D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8
	Select 2	Select 1	Select 0	Width MSB				Width LSB

图 6.6 项寄存器

如图 6.6 所示,项寄存器有 16 个,每个有 16 位,占 2 个字节,即占 2 个地址。每个项寄存器的 $D_7 \sim D_0$,即第一字节存放隶属函数的中心值 $c(0 \sim 255)$, $D_{12} \sim D_8$ 存放隶属函数的底宽 $w(0 \sim 31)$, $D_{15} \sim D_{13}$ 存放选择码 $0 \sim 7$,用于选择输入。

(7) 作用值寄存器(地址为 40H~7FH)

作用值寄存器有 64 个,每个有 8 位,每个作用值寄存器存放补码,范围为 $-128 \sim +127$ 。

(8) 规则寄存器(地址为 80H~BFH 和 C0H~FFH)

规则寄存器有 64 个,每个规则寄存器有 2 组,每个有 16 位,占 2 个字节,一个规则寄存器由 80H~BFH 中的一个字节和 C0H~FFH 中相应的一个字节组成,如 81H 和 C1H 组成一个 16 位规则寄存器,其余类推。81H 中的字节用于选择低 8 个模糊化器,C1H 中的字节用于选择高 8 个模糊化器。规则寄存器中某位为 1,选中对应的模糊化器。

注意:规则 0 是不能由用户编辑的,所以,80H 和 C0H 应写入 0。也即是说,用户实际只能用 63 条控制规则。

6.3.4 NLX230 的操作及接口技术

(1) 主方式

当 M/S 引脚端接 +5V,这时,NLX230 处于主方式状态,此时产生下列控制信号:

①串行时钟 SK:所产生的 SK 信号作为外部 EEPROM 的时钟信号。它是石英振荡器输入信号的 32 分频。

②片选信号 CS:这时所产生的 CS 信号用于对外部 EEPROM 进行片选;

③串行数据输出口 DO:DO 信号是外部 EEPROM 的地址信号。

此外,NLX230 的串行数据输入端 DI 会接收来自外部 EEPROM 的读出数据。

NLX230 在主方式中的这种控制规则需要专门的外部 EEPROM 器件,NLX230 与 EEPROM 93C56 的主要连线如图 6.7 所示。

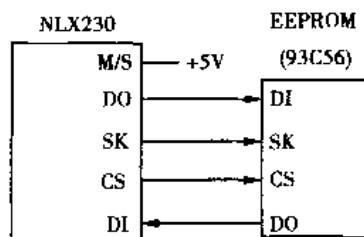


图 6.7 NLX230 与 EEPROM93C56 的主要连线图

(2) 从方式

当 M/S 端接电源地时,即地电平,则 NLX230 处于从方式的工作状态。在从方式下,NLX230 受外部逻辑电路控制的,接收外部逻辑电路来的下列信号:

①串行时钟 SK:由外部逻辑电路送给 NLX230,从 NLX 230 的 SK 端输入。

②片选信号 CS:由外部逻辑电路送向 NLX230 的控制信号,从 NLX230 的 CS 端输入。

③读写信号 R/W:由外部逻辑电路送来,从 NLX230 的 R/W 端输入。当 R/W = 1 时,表示读 NLX230 的内部结构;当 R/W = 0 时,表示写 NLX 230 的内部结构。

从方式有两种工作过程:即上装过程和下装过程。所谓下装,就是外部逻辑电路写 NLX 230;所谓上装,就是外部逻辑电路读 NLX230。无论在上装或是下装完成后,需要用一个复位信号 RST 对内部的数据进行初始化,才能启动 NLX230 的正常工作。

(3) 系统接口方法

当 NLX230 像单片机一样作为单独的核心处理器使用时(即主方式下),它只需要一个外部串行 EEPROM 存储配置信息就可以了。

当把 NLX230 和一般的微处理器连接使用时(即从方式下),就需要在它的 SK、CS、R/W、DI、DO 5 个引脚上提供所需的信号。根据输入输出的数目,还应在它的外部设置相应的输出锁存器和输入缓冲器。

(4) 初始化

在主方式时,初始化会使 NLX230 自动从外部 EEPROM 下装配置数据,这时,所有的输出端保持逻辑 0,直到配置数据下装结束;在从方式时,NLX230 等待外部逻辑电路下装配置数据,在复位之后,所有的输出端 $DO_0 \sim DO_7$ 都保持逻辑 0,直到数据从输入端 $DI_0 \sim DI_7$ 传送到输出端 $DO_0 \sim DO_7$ 为止。

6.3.5 NLX230 的应用实例

(1) NLX230 控制抽湿机的工作原理

NLX230 控制抽湿机的原理如图 6.8 所示,共有 3 个输入变量,它们就是空气湿度 H ,压缩机温度 T 和水箱的水量 W 。被控对象有 3 个,它们是压缩机、电风扇和电源。

模糊控制器由 NLX230 以及有关接口电路组成,通过分时方式分别接受不同的输入以及进行不同的输出,当空气湿度较大时,压缩机电机以较高速度工作,以便快速除湿;但是,在压缩机温度较高时,则要求降低压缩机电机速度,以保护压缩机。所以,对压缩机的控制采用如

下形式的条件推理语句:

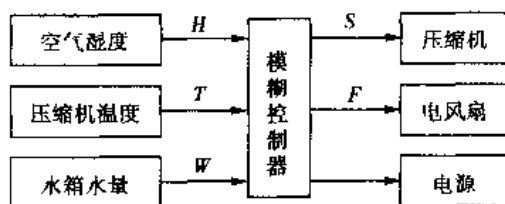


图 6.8 NLX230 控制抽湿机的系统原理图

if H and T then S

其中, H 是湿度, T 是压缩机温度, S 是压缩机电机速度。

压缩机的温度越高,则要求电风扇的转速越高,以实现压缩机的散热;同时,当空气较干燥时,压缩机的散热效果较差,所以,也要求电风扇有较高的转速。因而对电风扇的控制采用如下条件推理语句:

if T and H then F

其中: T 是压缩机的温度, H 是空气湿度, F 是电风扇电压。

湿度划分为 5 个模糊量,都采用包含型隶属函数,其底宽均为 20,中心值 36%,52%,68%,84%,100% 可分别用 0,32,64,96,127 表示,如图 6.9 所示。

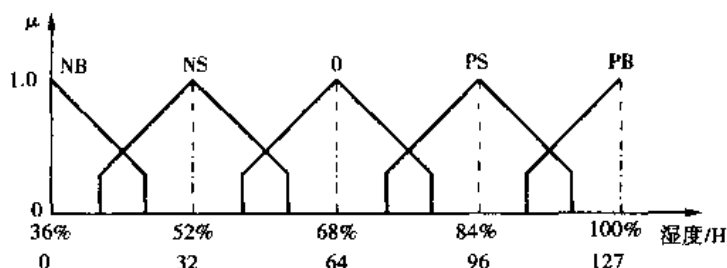


图 6.9 湿度的模糊化

温度划分为 5 个模糊量,其模糊量都采用包含型隶属函数,除“ M ”的底宽为 16 之外,其余为 24,如图 6.10 所示。

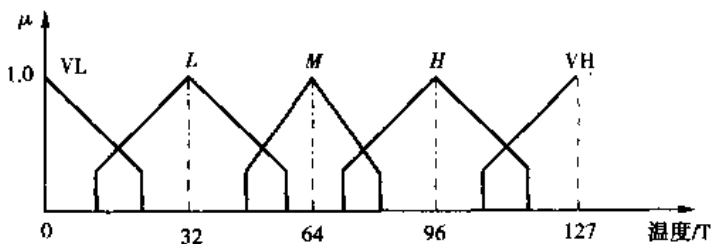


图 6.10 温度的模糊化

在对压缩机电机进行控制中,要进行电机的速度控制,控制采用变频方式。对压缩机的变频控制采用单点模糊量,也就是 10Hz、20Hz、30Hz、40Hz、50Hz 的频率。

在对电风扇电机进行控制中,采用移相调压方式。对电风扇电机控制也采用单点模糊量。所以,控制电压分别取 140V、160V、180V、200V、220V。

考虑压缩机的控制情况,根据对压缩机的控制经验,采用的控制规则集如下:

if H is PB and T is VL then 50
 if H is PB and T is L then 50
 if H is PB and T is M then 50
 if H is PS and T is L then 40
 if H is PS and T is M then 40
 if H is Z and T is M then 30
 if H is NS and T is M then 20
 if H is NS and T is L then 20
 if H is NB and T is M then 10
 if H is NB and T is H then 10
 if H is NB and T is VH then 10

对于电风扇的控制,所用的控制规则如下:

if T is VL and H is PB then -40
 if T is VL and H is PS then -40
 if T is L and H is PB then -20
 if T is L and H is PS then -20
 if T is H then 20
 if T is VH then 40

(2) 控制系统的硬件结构

1) 输入结构

由 ADC0808 所构成的 A/D 转换器组成输入结构。在抽湿机控制中,被控对象不是一个高速响应的系统。所以,ADC0808 足以满足系统的速度要求。

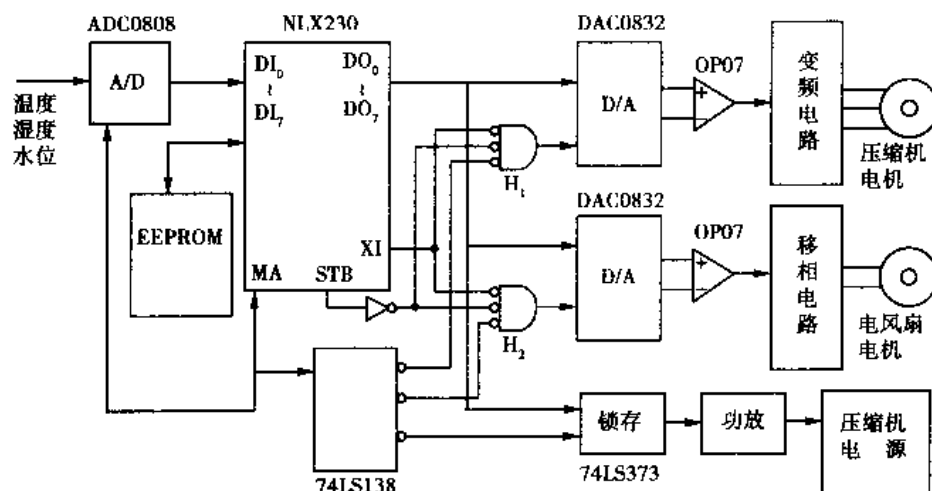


图 6.11 控制系统的硬件电路

2) 控制结构

由模糊单片机 NLX230 和 EEPROM、译码器 74LS138 以及有关附属电路组成控制结构。NLX230 的配置信息都存放在 EEPROM93C56 中,在 NLX230 中,配置信息用于设定输出方式寄存器、输入配置寄存器、隶属函数类型寄存器、初始值寄存器、项寄存器、作用值寄存器、规则

寄存器等的內容。 H_1 和 H_2 是“零与门”。它们用于把振荡输出信号 XI 、选通信号 STB 和 $MA_0 \sim MA_2$ 译码信号组合起来,去对输出信号进行选择。当 $MA_0 \sim MA_2$ 的值不同时,输出的选中信号就不同。在实际电路中,这些不同的输出就是控制压缩机电机、电风扇电机、压缩机电源通断的信号。

3) 输出结构

输出电路有三个:即压缩机控制、风扇控制和压缩机电源通断控制电路。压缩机控制由 DAC0832、OP07 和变频电路组成。其产生的模拟信号去控制变频电路工作,变频电路进而控制压缩机工作。

风扇电机控制则由 DAC0832、OP07 和移相电路组成,由 DAC0832 和 OP07 组成的电路把数字信号转换成模拟信号,去控制移相电路进行移相,进而控制电风扇的速度。

压缩机电源通断电路由锁存器、功放电路组成。锁存器 74LS373 接收 NLX230 输出的数字信号之后,其中,代表抽湿机水箱水位的一位状态通过功率放大后,去控制压缩机电源的通断。

(3) 抽湿机控制系统的开发

在开发过程中,利用软件输入控制规则和隶属函数,并对实际系统进行实时仿真调试。系统的隶属函数和控制规则如下:

NLX230

TERMS

H is NB 0 20 in

H is NS 32 20 in

H is 0 64 20 in

H is PS 96 20 in

H is PB 127 20 in

T is VL 0 24 in

T is L 32 24 in

T is M 64 16 in

T is H 96 24 in

T is VH 127 24 in

OUTPUTS

P O IM/ * Compressor Control */

{

if H is PB and T is VL then 50

if H is PB and T is L then 50

if H is PB and T is M then 50

if H is PS and T is L then 40

if H is PS and T is M then 40

if H is 0 and T is M then 30

if H is NS and T is M then 20

if H is NS and T is M then 20

```

if H is NB and T is M then 10
if H is NB and T is H then 10
if H is NB and T is VH then 10
{
OUTPUTS
V 80 IM/ * Fan Control */
{
if T is VL and H is PB then -40
if T is VL and H is PS then -40
if T is L and H is PB then -20
if T is L and H is PS then -20
if T is H then 20
if T is VH then 40
}

```

在上面的电风扇控制中,“V 80 IM”是表示输出是电压,初值为 80,采用立即输出方式。初值为 80 在实际中与 180V 电压相对应。

在实际中,一开始给出的条件语句可能并不理想,通过仿真之后,再根据结果对控制规则或隶属函数做一些修改,反复多次,最后求出的隶属函数和控制规则作为最终结果,把这些结果变换成配置信息写入外部 EEPROM,以实现控制系统的最终独立运行。

第 7 章

模糊控制应用实例

与传统控制理论相比,模糊控制有两大不可比拟的优点:第一,模糊控制在许多应用中可以有效且便捷地实现人的控制策略和经验,这一优点自从模糊控制诞生以来就一直受到人们密切的关注;第二,模糊控制可以不需要被控对象的数学模型即可实现较好的控制,这是因为被控对象的动态特性已隐含在模糊控制器输入、输出模糊集及模糊规则中。在本书的最后一章选取了一些模糊控制的应用实例,通过对这些实例的介绍,可以感受到模糊控制的多领域应用。

7.1 模糊控制全自动洗衣机

从控制角度看,传统全自动洗衣机实际上是一台按事先设定好的参数进行顺序控制的机器。从这个意义上说,其“全自动”并不具有任何智能,它不能根据情况和条件的变化来改变参数;而模糊逻辑控制的全自动洗衣机向真正的智能化的全自动迈进了一大步,它的目标则是要求根据所洗衣服的数量、种类和脏的程度来决定水的多少、水流的强度和洗衣的时间,并可以动态地改变参数,以达到在洗干净衣服的情况下还要尽量地不伤衣服、省电、省水、省时的目的;另外,要求操作简单,任何人都可以轻松地使用,即“一个按钮”一按就行;进一步还要求机器能“人性化”,对人“友好”,不要像黑箱子,而能把其工作情况和过程显示出来。比如,用液晶屏显示机器已经进入洗衣的哪个阶段,洗涤过程中衣服已经完成预定洗净目标的百分比,以及用倒计时方式显示还需多长时间完成整个任务。目前,已经用模糊逻辑控制技术开发出满足这些基本条件的全自动洗衣机,并还在不断提高其性能和智能水平。在其他条件相同的条件下,某些日本产的模糊全自动洗衣机已经可比普通洗衣机提高洗净度 20%;其洗涤水平可望达到有经验的老练主妇直觉判断的水准。

日本的 Matsushita、松下、三洋和日立公司先后都推出了自己的“一个按钮”的模糊控制洗衣机,但是,它们所考虑的因素和重点并不相同。例如,“三洋”洗衣机是调整溶于水的洗涤剂量,而“日立”洗衣机是判断衣物的重量和类型来设定合适的洗涤时间,其重点并不是放在衣物是否会被损坏上。下面以一种模糊控制洗衣机为例来分析其实现的原理和方法。

7.1.1 洗衣条件

要把衣物洗干净,去除污垢,与如下一些因素有关:衣物的质料、水的硬度、水的多少和温度、洗涤剂的性能和多少、机械力的大小和作用时间等。

(1) 衣服的质料

一般衣服质料纤维可分两大类:自然纤维的棉织品和人造化学纤维织品。棉制品的污垢不仅在表面,而且还渗透纤维内,所以,棉织品要比化学纤维难洗。

(2) 水

水可带走一般的灰尘和水溶性污垢,所以,不用洗涤剂也可能洗去部分污垢。水的硬度在用肥皂时也会影响洗涤效果,但影响最大的还是水的温度,在一定的水温范围内,温度越高,洗涤效果越好。图 7.1 所示为水温与洗净力之间的关系曲线。然而,温度也不宜太高,否则,高温会把附着在衣服上的蛋白质凝固,反而影响洗涤效果。

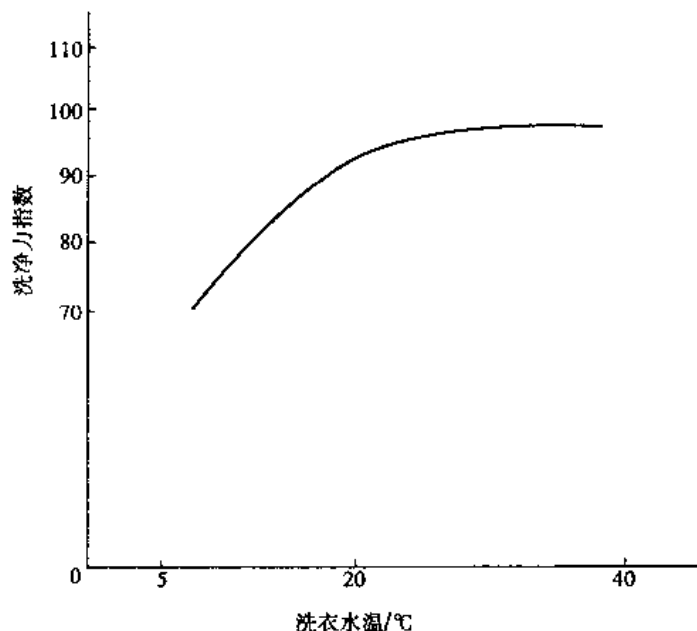


图 7.1 水温与洗净力之间的关系曲线

(3) 洗涤剂

洗涤剂的成分主要以烷基苯活性剂为主,不同的洗涤剂还会添加各种不同的辅助剂、酵素、荧光增白剂、香料等。

7.1.2 模糊控制洗衣机结构

图 7.2 所示为模糊控制洗衣机的机构剖面图。它由缸体、电动机、搅拌轮、进水阀、排水阀和各种传感器构成。

要对洗衣机进行控制,首先要用各种传感器不断地检测相关的状态,以作为控制的依据。下面介绍在模糊控制洗衣机中所用各种参数的检测原理和技术,在检测中要用到负载传感器、水位传感器和光电传感器等。

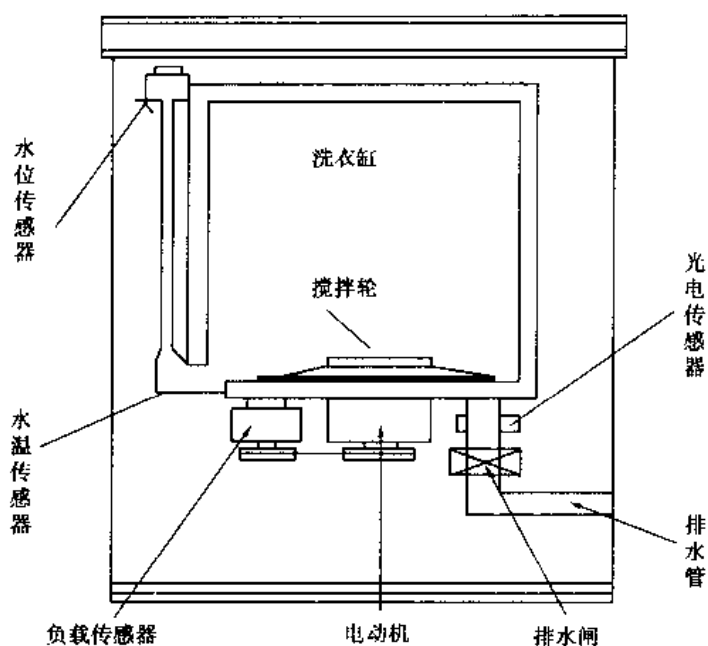


图 7.2 模糊控制洗衣机结构剖面图

(1) 负载检测

这主要用来检测所洗衣物的重量,以决定水位。这可用不同的方法实现。最容易想到的方法是用静态的压力传感器直接测量,但结构要稍复杂一些。目前,一般是用动态的间接测量方法,这是通过检测电动机负载来实现的。电动机负载既可用正常运转时的驱动电流来计算,也可用电动机断电后的反电动势的大小以及波形来计量。

以计量断电后反电动势为例,可以通过电阻取样后经 A/D 转换器送微处理器进行数据处理。具体方法是:当衣服投入缸体中后,加入适量的水,启动电动机旋转若干圈后断电,测量电动机线圈两端的反电动势,经 A/D 转换器转换成数字量后送微处理器或者单片机处理判断,以决定衣物的重量。一般而言,衣物重,负载大,其反电动势也大,但跌落也快,即惯性转动的时间也短;反之亦然。

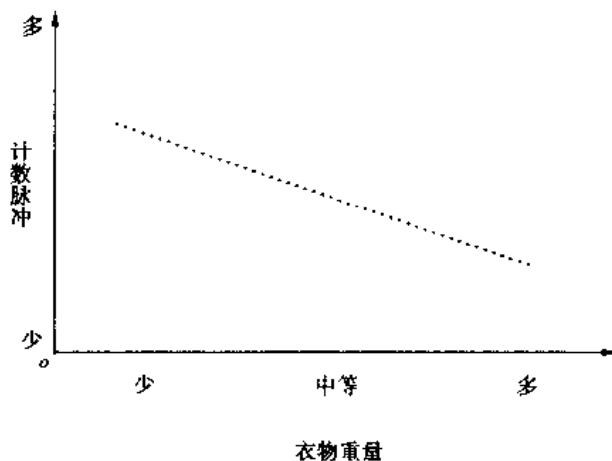


图 7.3 计数脉冲与衣物重量关系曲线

事实上,也可用另一种方法实现,不用 A/D 转换器,在放入要洗的衣物

后,放水启动电动机,反复用开 0.3s、关 0.7s 的脉冲电压驱动电动机 32s,在此过程中,用光电耦合器发送与接收的脉冲来计量在关断期间轮盘的惯性转动的转数。这时,衣物多,转动时间就短,计数就少,反之则多。计数脉冲与衣物重量关系曲线如图 7.3 所示。

(2) 质料检测

质料检测包括棉制品与化纤制品的区分和柔软布料与粗厚布料的区分。检测的具体方法如下。

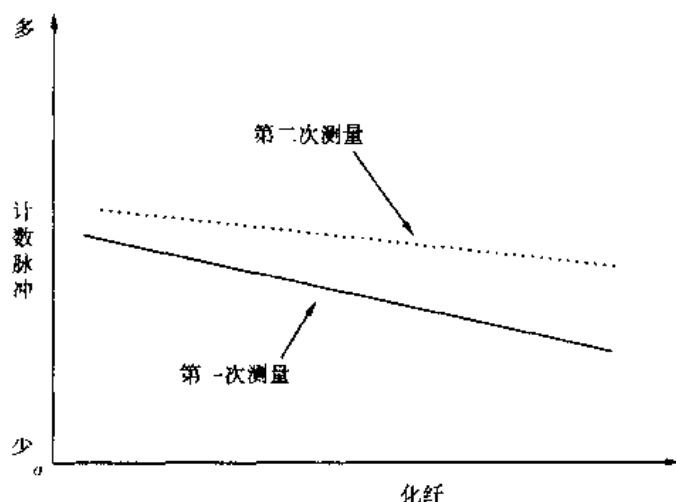


图 7.4 棉制品与化纤制品辨别曲线

在检测负载的基础上,把水放掉一点,同样用开0.3s、关0.7s的脉冲电压驱动电动机32s,记下脉冲数为 N ;若检测负载时的脉冲数是 M ,根据 $M-N$ 的值就可判断质料分布比例的大小情况。棉制品越多, $M-N$ 的值越大,反之则小。图7.4表示棉制品与化纤制品比例不同时两次测量脉冲的关系曲线。

同样都是棉制品,但对于像毛巾这样的柔软布料和牛仔布类这样的厚布料,其洗涤方法也是不同的,可用水位传感器来配合测量。具体方法是:在注水进行脉冲驱动32s后,比较启动前后水的变化量。若变化量较小,说明布料容易吸水,倾向于是毛巾类布料;反之,可能是牛仔布类厚布料。这是因为厚布料吸水慢,往往要搅动一段时间后才能充分吸水,这就会使水位变化量大。图7.5所示为在这两种情况下水位的变化曲线。

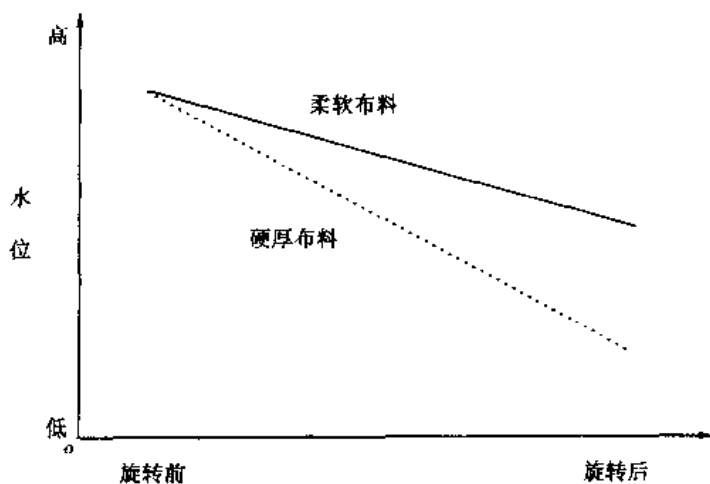


图 7.5 柔软布料和硬厚布料的水位变化曲线

如果测量的是驱动电流模拟量,也可通过对驱动电流的波形变化来识别。图7.6所示为

两种典型的不同软硬度布料情况下的驱动电流波形比较。

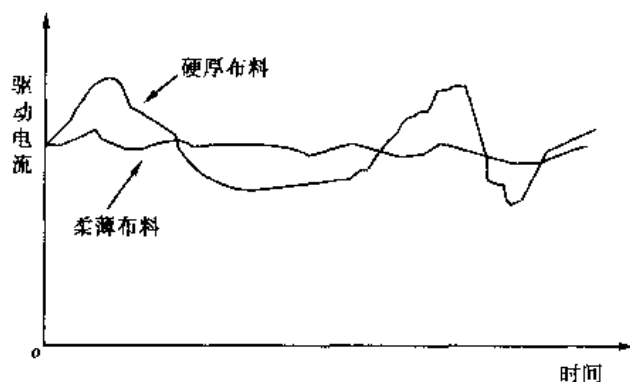


图 7.6 不同软硬度布料情况下驱动电流波形

(3) 水位检测

水位检测是用一种专用水位传感器实现的。这种水位传感器是一根与缸体等高的空管，它与缸体构成一个连通器，空管的上端有一个用压力膜隔开的差动电感器，当缸中有水注入时，管内的空气被压缩，使压力膜上压力增大，继而推动与它联动的铁心的移动，引起线圈的电感量变化。用此电感器构成 LC 振荡器的频率就能反映水位的高低，用这个传感器既可用于配合以上布料软硬度的检测，同时，也作为水位控制依据的检测装置。

(4) 水温检测

水温检测是通过热电耦测量的。它把洗衣机启动时的温度作为当时的室温，然后再检测供水的温度，以作为洗衣条件之一，根据需要可以对水加热且控制水温。

(5) 水的透光率检测

水的透光率是用光电传感器实现的。它被安排在排水管出口，发出二极管和光敏管分别相对着安装在管子的两边。发光二极管发出的光经聚焦后，透过水被光敏管接收，接收的强度就反映了水的透明度，也就是水的污垢程度。这是一种间接测量衣物污垢的方法，因为衣物脏的程度与洗涤水的污浊度有着正相关的关系。它的输出变化曲线如图 7.7、图 7.8、图 7.9 所示。

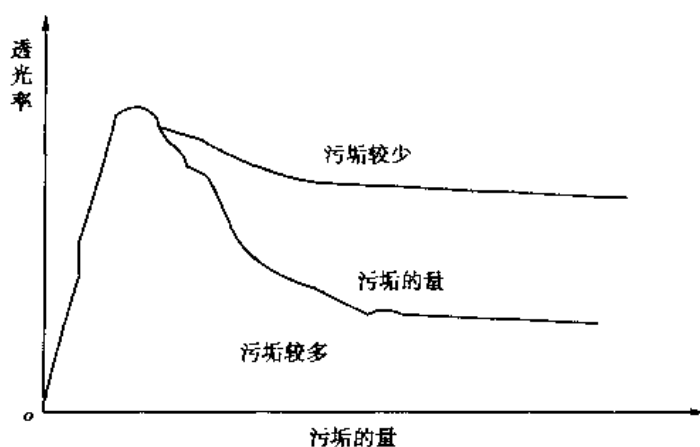


图 7.7 污垢程度透光率曲线

从图 7.7 和图 7.8 中可知，开始注入清水，透明度很高，随着污垢析出，水逐渐变浊，透明

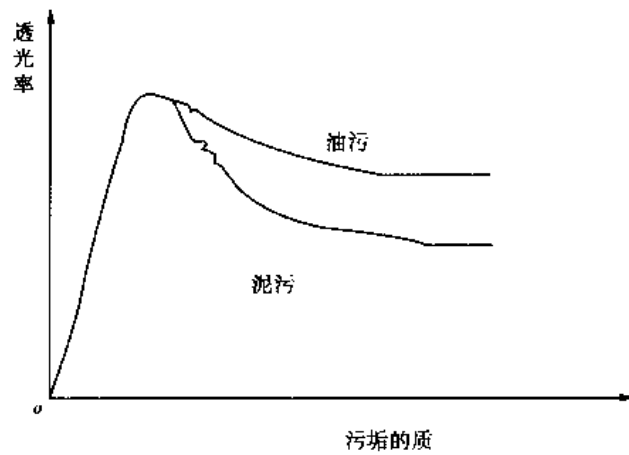


图 7.8 污垢性质透光率曲线

度下降,最后就有一个饱和稳定值。根据透光率的变化形态和过程,可以知道污垢的性质和程度。对于泥污类的污垢,一般分离得较快,较早进入水中,故其透光率进入饱和状态的时间较短;而油污类污垢相对分离得较慢,透光率的变化速率小,达到饱和值的时间也较长。另外,根据同样的思路(如图 7.9 所示),还可以从透光率的变化知道洗涤剂用的是固体粉末状态的还是液态的。总之,饱和值的高低反映了衣物脏的程度,透光率下降到饱和值的时间长短反映了污垢的性质。

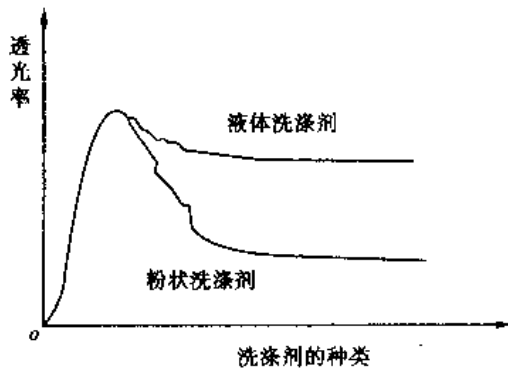


图 7.9 用不同洗涤剂时透光率曲线

7.1.3 控制电路设计

图 7.10 所示为模糊控制洗衣机的电路框图。它以单片机为核心,加上负载检测电路、水位检测电路、温度检测电路、光电检测电路以及键盘显示电路组成。

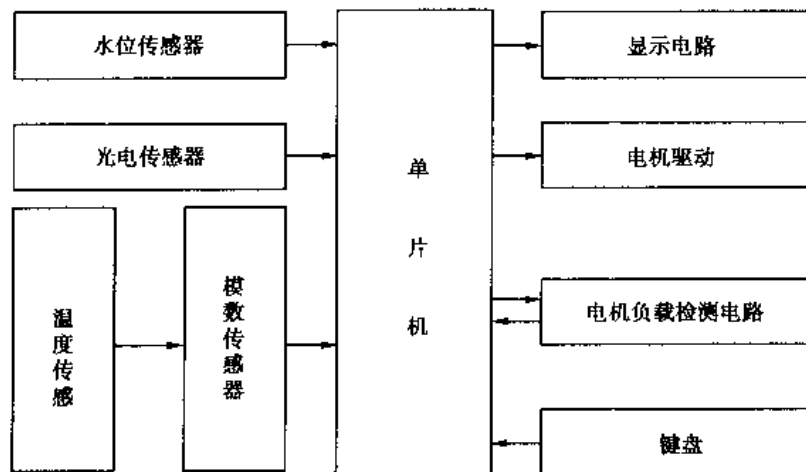


图 7.10 模糊控制洗衣机电路框图

1. *Journal of the American Medical Association*, 1997; 277: 1001-1005.



1000

1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 26

7.1.4 模糊控制实现方法

(1) 基本结构和控制过程

图 7.13 所示为模糊控制洗衣机控制结构图。它是利用负载、质料、水位、水温 and 气温以及洗涤剂类型等检测所得到的信息,进行分段评估计算,使其模糊化,再根据模糊规则进行推理,最后根据所激活的规则进行解模糊判决,以决定最适当和明确的水流、水位、洗涤时间、清洗方法以及脱水时间。

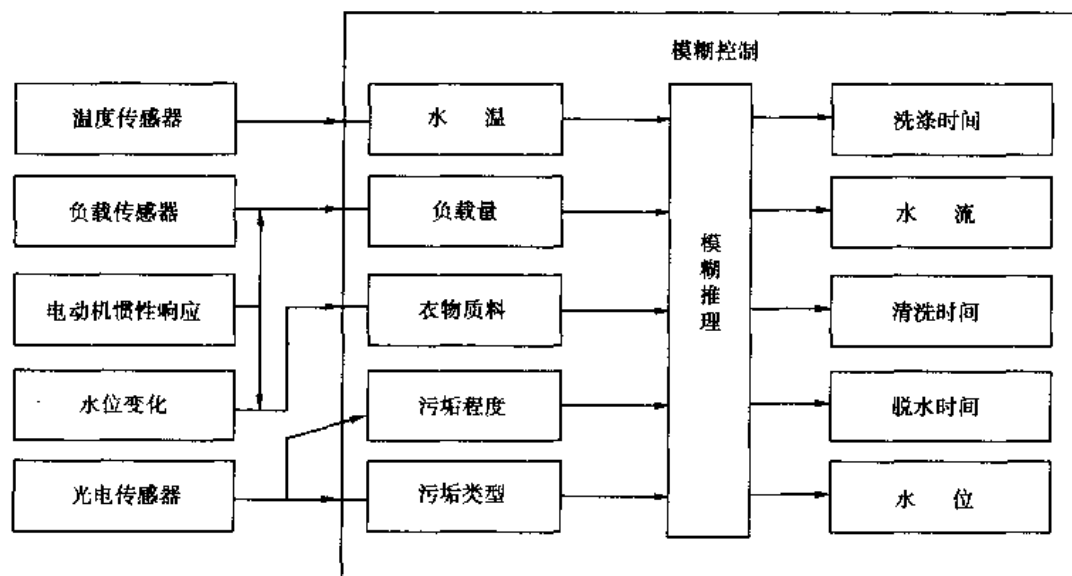


图 7.13 模糊控制洗衣机控制结构图

(2) 模糊规则

1) 输入变量(分三级)

①负载:大、中等、小。

②质料:棉制品偏多、棉和化纤制品各半、化纤制品偏多。

③水温:偏高、中等、偏低。

2) 输出变量(分四级)

①水流强度:很强、强、中、弱。

②洗涤时间:很长、长、中、短。

3) 模糊规则

根据输入变量和输出变量的分级组合,对于水流强度和洗涤时间就可用以下的 $27(3 \times 3 \times 3)$ 条模糊规则表示。

规则 1:如果负载小,质料化纤制品偏多,且水温偏高,那么就将水流调弱,洗涤时间调短。

规则 2:如果负载大,质料棉制品偏多,且水温偏低,那么就将水流调强,洗涤时间调长。

.....

以此类推,可写出其他规则,表 7.1 是输出变量与输入变量之间关系的模糊真值表。

表 7.1 模糊控制规则表

		棉制品偏多			棉和化纤制品各半			化纤制品偏多		
		偏低	中等	偏高	偏低	中等	偏高	偏低	中等	偏高
偏大	水流	特强	强	强	强	强	中	中	中	中
	时间	特长	长	长	长	长	中	长	中	中
中等	水流	中	中	中	中	中	中	中	弱	弱
	时间	长	中	短	长	中	中	中	中	短
偏小	水流	弱	弱	弱	弱	弱	弱	弱	弱	特弱
	时间	中	中	短	中	短	短	中	短	特短

(3) 模糊控制的隶属函数

采用最简单的三角形隶属函数表示。

1) 输入变量

图 7.14(a)、(b)、(c) 分别表示输入变量负载、质料和水温的隶属函数。

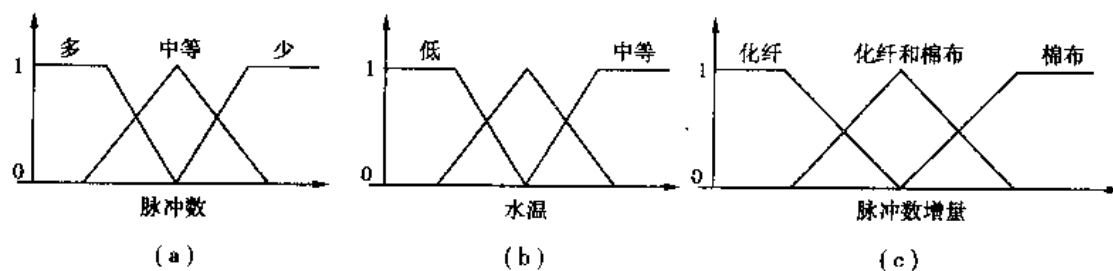


图 7.14 输入变量的隶属函数

(a) 负载量; (b) 水; (c) 质料

2) 输出变量

图 7.15(a)、(b) 分别表示输出变量水流强度和洗涤时间的隶属函数。

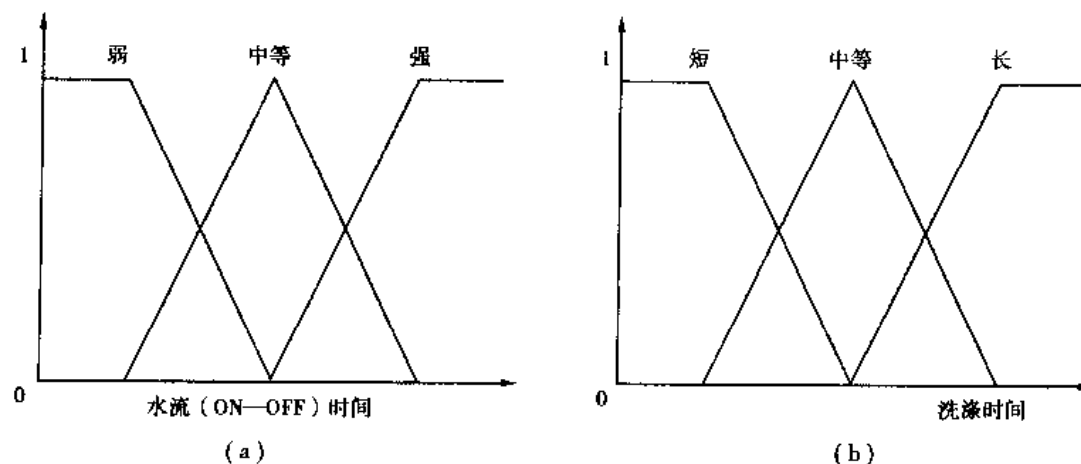


图 7.15 输出变量的隶属函数

(4) 解模糊判决结果

经过多次反复实验与模拟操作,最后对洗涤时间和水流确定了 264 种洗涤控制方法。据此又细分成如下的洗涤控制:

- ①水流 9 种;
- ②洗涤时间 16 种;
- ③清洗时间 6 种;
- ④脱水时间 6 种。

经实验对比,这种模糊控制的洗衣机在 5℃ 时,比一般洗衣机可提高洗净度 20%,并且可以避免衣物过度洗涤而遭到损伤,或者未能充分洗涤而去不掉污垢,既节约了时间,又节约了能源。

7.2 智能手机充电器

随着经济社会的高速发展,手机使用率越来越高。使用手机则需要对电池经常进行充电。手机的电池目前有镍铬电池、镍氢电池和锂离子电池三种。由于它们的充电方式完全不同,所以,在过去需要不同的充电器对其进行充电。模糊手机充电器是一种智能充电器,它可以对上述三种电池进行识别,然后根据电池的类型以相应的方式对其充电,使得电池能最有效、最恰当的充电。这样既能延长电池的寿命,又能一个充电器对多种电池进行充电。因此,模糊手机充电器是一种十分有效用的、性能优良的新型家用电器,是现代化生活的一种必需品。

7.2.1 智能充电原理

电池充电的关键在于实现有效充电的同时保证可以适时终止充电,防止过充电对电池造成损害;充电中应显示电量,以便了解充电情况;要有查错功能,在发现错误时及时停止充电,以保护电池和充电器。

镍铬电池适于恒流充电,这类电池可以允许以 $0.1C$ (C 是容量额定值,以“安时”计) 长时间过充电,而不会发生过大的温升。在快充电时,镍铬电池既需要主要充电终止法又需要使用辅助的充电终止法。系统采用 $-dv/dt$ 法。为防止其他误判断,还应增加最长充电时间限制。在以 $1C$ 的速度快速充电 1h 后,电池恢复了 90% ~ 95% 的电量,到完全充足电还需要几个小时。这是因为电池内把充电电流变换成可储存的电量的效率不是 100%。

同镍铬电池一样,镍金属氢化物电池也要求恒流充电。以标准的 $0.1C$ 速率慢充电时间不应超过 16h,达到这个时间后要改为涓流充电。

锂电池是用恒压限流电源进行充电的。这类电池易于受到过充电、深放电以及短路三种状态的损害,要特别加以注意。

不同类的电池其充电方式不同。充电时镍铬电池电压随电量的变化有着明显的变化,而在充电前期温度变化不明显;镍氢电池的温度随其电量的变化有相对较大的变化,可以利用它和镍铬电池在这点的不同将二者区分开来;锂电池在充电后期电压和电池的温度变化不明显,利用时间参数,可相对准确地确定其电量。根据以上对应关系可自动识别不同类型的电池,对应各类型进行准确的电量显示。如果不引进智能控制,而仅用电子器件根据这种非线性对应

关系完成准确显示显然是很困难的。

对电池充电还应有完善的充电电池保护措施。在充电前必须检测电池温度,若电池温度低于 10°C 后才使用快速充电。在充电过程中还应有过压、过热和过充保护措施,以保证充电电池不致因过充电而损坏。

7.2.2 手机充电器模糊控制的系统原理框图

系统原理框图如图 7.16 所示。

该手机电池充电模糊控制系统选用 Motorola 公司最新推出的功能强、价格低的 MC68HC05SR3 单片机作为核心器件。SR3 单片机具有较大的 ROM 和 RAM 空间,便于采用模糊控制方法,并带有 4 种 A/D 转换器,便于模拟量的检测。

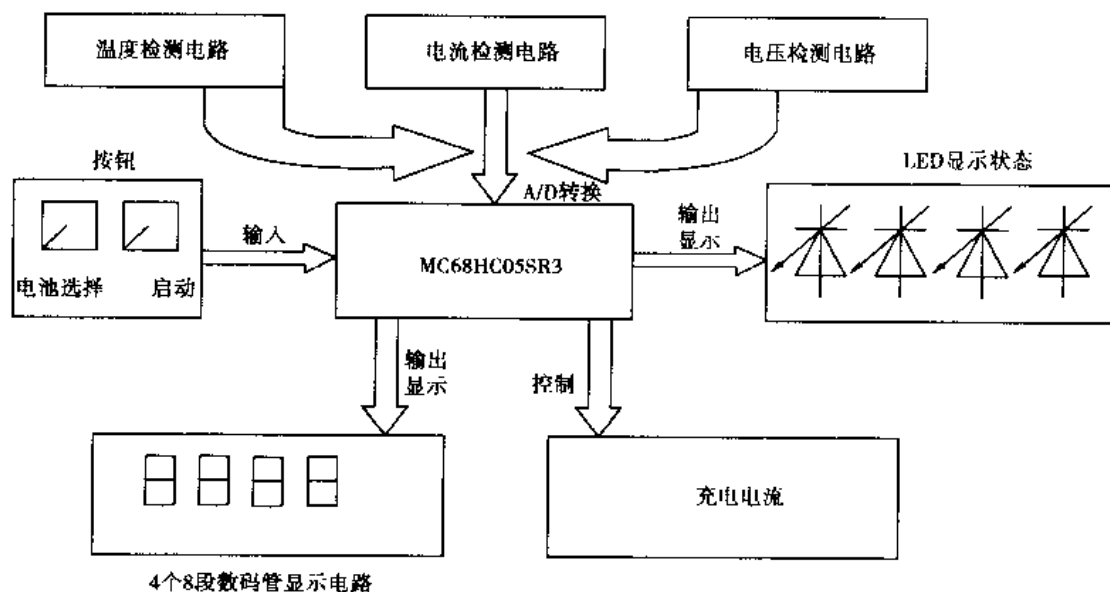


图 7.16 智能手机充电器系统原理框图

SR3 的时钟由 4MHz 晶振产生。两个功能键(开始充电及充电电池种类)接于 PA_0 、 PA_1 端口。4 位 LED 数码管显示及 4 个状态指示发光二极管由 PB 口和 PA_2 、 PA_3 、 PA_4 、 PA_5 、 PA_6 动态扫描驱动显示。

在电池充电过程中,要检测电池的电压、充电电流及温度。由于有两个电池,故共有 6 个模拟量要检测,而 SR3 只有 4 路 A/D 转换器,故而采用一个多路模拟开关,使两个电池的电压、电流的 A/D 转换器共用 AN_1 、 AN_0 ,两路温度检测量连接到 AN_2 、 AN_3 口。多路开关由 PC_2 、 PC_3 控制。温度的检测采用了负温度系数的热敏电阻。

两路电池的充电电流是由 PC_0 、 PC_1 口控制的,充电电流大小的调节方式采用 PWM 控制方式,SR3 通过控制 PC_0 、 PC_1 电平高低来控制三极管的开与关,而三极管开、关时间的长短决定了充电电流的大小。

7.2.3 充电控制原理

充电控制原理框图如图 7.17 所示。

整个控制系统实际上是一个给定的电池充电电压模型的跟随系统,根据充电电池的充电

原理,将电压模型曲线分为三段,分别采用不同的控制方式。电池电压在 5.2 ~ 5.6V 之间,这时电池内的电量已基本用完,为节省充电时间,采用恒定的大电流充电;电压在 5.6 ~ 6.0V 之间,这段为充电的关键阶段,若充电过多,会损坏电池,充电不够,会使电池产生恶性记忆,影响电池的容量,因此,这一段充电采用模糊控制方法,使电池电压按理想充电模式电压变化,从而达到最佳充电效果;电压大于 6.0V,这时电池充电已满,为防止电池放电,继续以点滴充电方式充电。

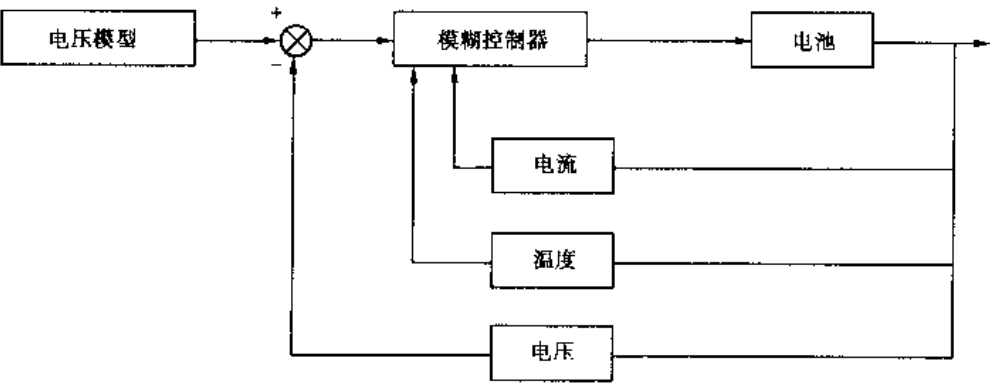


图 7.17 手机模糊充电控制器系统原理框图

充电电流的模糊逻辑控制器以理想电压和实际电压之差 ΔV 和 ΔV 的变化率 $\Delta V/\Delta t$ 作为输入,充电电流大小作为输出,构成一个两输入单输出的模糊控制系统。将 ΔV 和 $\Delta V/\Delta t$ 均划分为 5 个模糊状态,即“负大”(-2)、“负小”(-1)、“零”(0)、“正小”(+1)、“正大”(+2)。模糊控制系统对以上两个输入量进行决策,求出的模糊控制表如表 7.2 所示。

表 7.2 模糊控制规则表

$I \begin{matrix} \Delta V/\Delta t \\ \Delta V \end{matrix}$	-2	-1	0	+1	+2
-2	-2	-2	-1	-1	0
-1	-2	-1	-1	0	1
0	-1	-1	0	+1	+1
+1	-1	0	0	1	+2
+2	0	+1	+1	+2	+2

在表 7.2 中,“I”为 0,表示维持现在充电电流;“+1”表示充电电流增大一个等级;“+2”表示充电电流增大两个等级;“-2”表示充电电流减小两个等级。

由于手机充电器采用 Motorola 公司的 MC68H705SR3 单片机作为中心控制器件,利用该单片机集成了 A/D 通道等特点,充分地发挥智能控制的优越性,以尽量少的外围器件实现了恒压、恒流两路充电。恒流充电电路实现对镍铬和镍氢电池的充电,恒压充电电路实现对锂电池充电。系统可通过按键进行功能设定,选择慢充、快充和放电三种工作方式之一,对两路的控制实现了准并行处理,除功能设定时互斥地使用同一按键外,相互间无影响。另外,除可由功能设定按键选定执行放电功能外,也可由系统自动按缺省方式先执行放电功能。先放电再

进行充电,可以保证电池足额充电。考虑到在开路情况下,电池也会由于内部自放电造成电量的减少,所以,引入了涓流充电电路,以在充电完成后尚未取出电池的情况下,对电池进行小电流充电,从而保证电池电量不会降低。

电压检测电路用于将两路电池的电压值分别进行处理后送单片机的 A/D 口,单片机用 A/D 转换的结果判断电池是否插入,电池电压是否过高,充电是否完成等。温度检测电路将电池内电阻随电池温度的变化转换为电压的变化送至单片机,用于电池温度超界检测及电池类别的判定等。在完成充电前期类型检测后,系统自动地依三类电池的不同特性而显示其容量。为消除在测电池电压及温度时的外部干扰,除硬件措施外,软件中还采用了“四取二防”脉冲干扰的数字滤波技术。

恒流充电电路模块的结构如图 7.18 所示。实现恒流输出的主要器件是可控三端稳压器 LM317。LM317 的基本工作方式如图 7.19 所示。

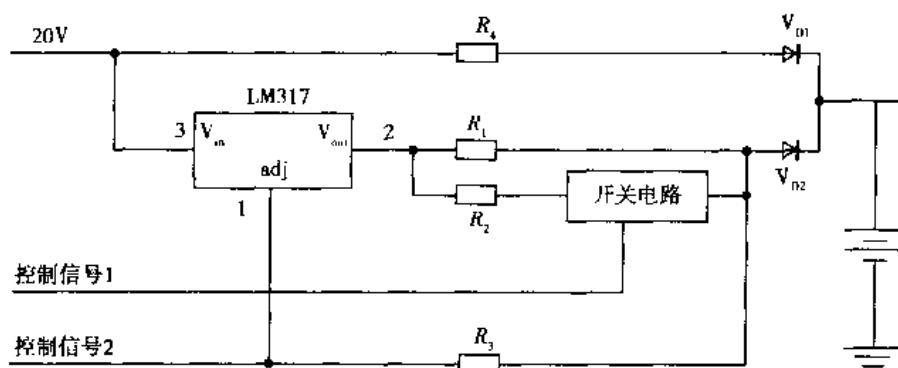


图 7.18 恒流充电模块原理图

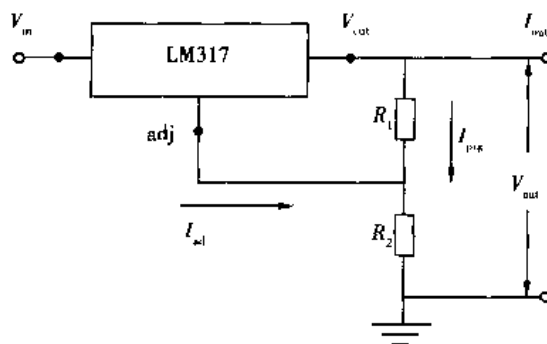


图 7.19 LM317 的基本工作电路

在正常工作时,这个器件输出脚 (V_{out}) 到校正脚 (adj) 间可保持稳定的 1.25V 参考电压 V_{ref} ,参考电压经电阻 R_1 转换为可调节的电流 I_{PRG} 。这一稳定的电流经过电阻 R_2 接地。输出恒流值的计算公式如式(7.1)所示:

$$V_{out} = V_{ref} \left(\frac{R_2}{R_1} \right) + I_{adj} R_2 + 1.25 \quad (7.1)$$

由于从 adj 端来的电流 I_{adj} 引入了电压偏差,所以,在设计 LM317 时, I_{adj} 被控制在小于

0.1mA的范围之内,而且其值保持恒定。为实现这一要求,内部工作电流都被强制流向 V_{out} 端口。同时,也决定了负载电流的最小值,小于这个最小值时输出电压会升高。LM317 是浮动恒压源,在实际工作时它仅需要保持 V_{out} 和 adj 之间的电压差值,所以,可以工作在对地有较高电压的情况。利用其恒压特性也可实现恒流输出。当图 7.18 中的控制信号 2 为低时,二极管 V_{D1} 阳极的电压低于阴极电压,不对电池充电;当控制信号 2 为高而控制信号 1 为低时,对电池进行慢充电,输出电流按式(7.2)计算。

$$I_{out} = \left(\frac{V_{ref}}{R} \right) + I_{adj} \approx \frac{1.25V}{R} \quad (7.2)$$

此时, R 值即为 R_1 阻值(5Ω),所以,输出电流为 250mA。

当控制信号 1、2 同时为高时, V_{D1} 、 V_{D2} 同时导通,则式(7.2)中的 R 值等于 R_1 和 R_2 (1.25Ω) 并联后的等效阻值(1Ω)。此时进行快速充电,快充电流为 1250mA。

7.2.4 控制软件

系统主流程框图如图 7.20 所示。

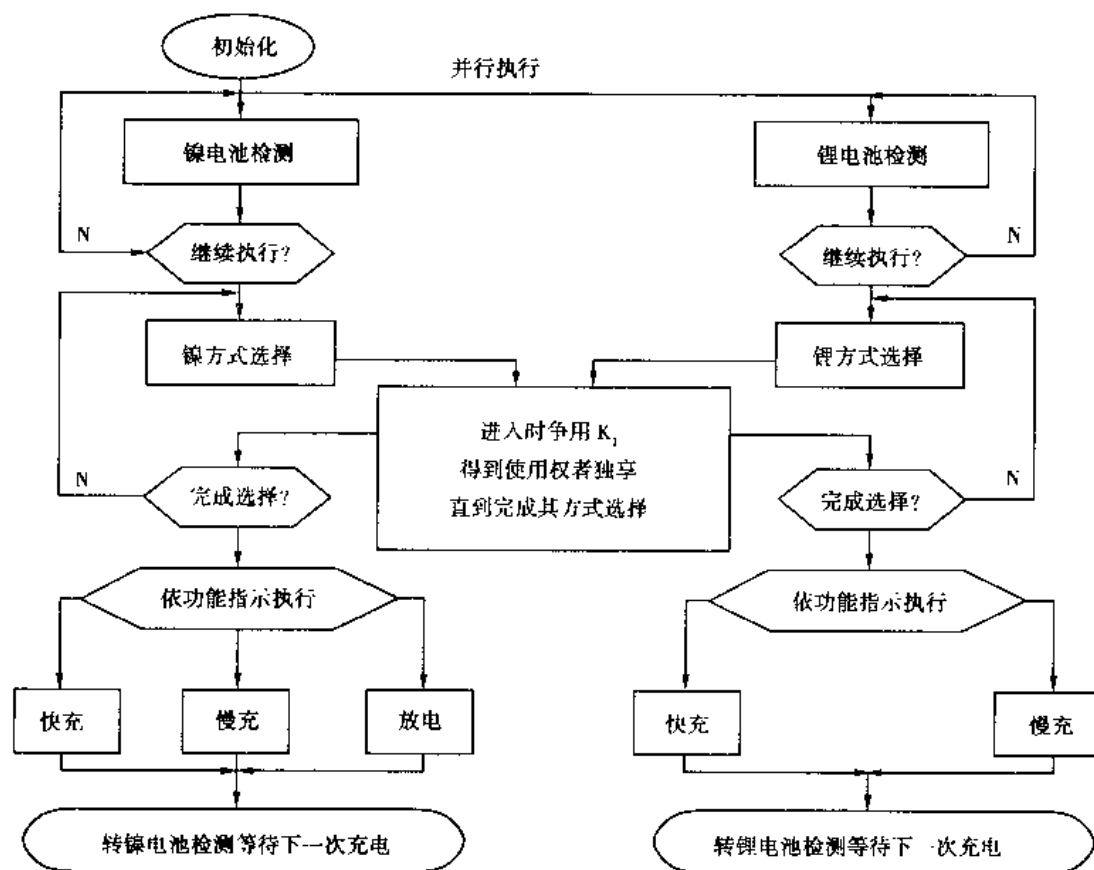


图 7.20 主流程框图

单片机加电后先执行公用初始化程序段,然后镍(Ni)组锂(Li)组并行执行。检测到已插入电池且无错误的组可争用功能按键 K_1 ,争用到后即可独享 K_1 进行功能选择。完成选择后开放对 K_1 的独享,并依所设定功能或缺省方式(等待时间内一直无按键信号来时)工作。在充放电过程中,调用子程序进行电压、电量显示,并进行监测,以便在功能完成或出错时及时退出,以保护电池。在快充开始第 5min 时,进行电池类别判定,依判定结果决定电量显示选择,同

时,若发现恒压和恒流类电池插反,即停止充电并报警。正常结束之后,各组返回其他电池检测处等待取出已充、放完电的电池,开始本组下一充、放电过程。

7.3 地铁机车模糊控制器

日本的日立公司为仙台市研制的模糊控制地铁电力机车自动运输系统是当今世界上相当先进的地铁系统,它自1983年开始试验和测试,历时4年,于1987年7月正式投入运行。它是模糊逻辑应用于控制领域的一座里程碑,也是模糊控制的一个十分闪光的范例。该车非常平稳、舒适,且可以很高的精确性停靠于站台。经过系统运行1万次以上的试验行驶、进站停车统计,停车误差在30cm以上的还不到1%,标准差是10.6cm;另外,还能比传统PID控制系统节省10%的燃料,且对于停靠站台的控制值变化次数,模糊控制只有PID控制的1/3。

根据机车运行控制的具体特点,采用了预测型的模糊推理。下面简单介绍日本仙台市地铁机车模糊控制的一些情况。

7.3.1 对评价指标的定义

为了对评价指标进行定义,这里先对所有符号进行说明。

用6个符号表示有关论域:

A: (Accuracy of stop gap) 停车准确度

C: (Comfort of riding) 乘坐舒适性

E: (Energy saving) 节约能源

R: (Running time) 行驶时间

S: (Safety) 安全性

T: (Traceability of speed) 速度跟踪性

用5个符号表示模糊概念的等级:

VG: (Very Good) 非常好

G: (Good) 好

M: (Medium) 中等

B: (Bad) 差

VB: (Very Bad) 非常差

如果要表示停车准确度非常好,在规则中就可用 $A = VG$ 表示,在隶属函数中就用 AVG 表示;同样以此类推,就用 $S = B$ 和 SB 表示安全性差。用这些符号就可对评价指标进行定义。

(1) 停车准确度

停车准确度是用停车目标位置与预测停车位置的相对距离 N_p 来描述的,用 N_p 定义停车准确度的模糊子集的隶属函数。

(2) 乘坐舒适度

如何对乘坐舒适度来进行评价呢?据研究表明,人对前后方向的震动并不敏感,但对上下震动却比较敏感;当速度控制阀频繁切换时,就会产生较高频率的震动而引起乘客感到不舒服。所以,这里是用行驶中速度控制阀变化的段数 N_c 的函数 $C(N_c)$ 和该控制阀在切换后

所维持的时间 T_c 来描述的,它们反映了乘坐的舒适性好坏。

(3) 节约能源

可用如下方法定义节约能源:

在车站与车站之间设定某个特定的地点 X_k ,如果从目前所在地点到 X_k 利用惯性来行驶,计算出可能要增加的时间,用这个可能要增加的时间与还剩余的时间做比较,来决定是否允许利用惯性行驶一段时间。

(4) 行驶时间

从进站标志到停车位置一般只有几十米,这段距离所花时间不会太长,而且大体相同,所以,行驶时间可用出发时间至到达进站标志点的时间作为行驶时间。

(5) 安全性

安全性被定义为当目前机车速度超过限定速度时,从该速度回到限定速度以下所需要的时间 T_s 。

(6) 速度跟踪性

速度跟踪性被定义为预测速度与目标速度的一致性。

7.3.2 对机车的运动特性模拟实验

由于道路的变化和其他因素的影响,使得刹车阈值产生的减速度与系统预定的值会有大约 $\pm 30\%$ 的误差。为了更好地控制机车,就必须对相应的减速度进行推算。其方法是:司机在机车上用一定的控制阈值行驶一段时间(例如 3s),用所行驶的距离就可计算出加速度或减速度。据此即可推得驱动和刹车系统的特性,对加减速度进行修正,以便更准确地预测。

7.3.3 模糊控制规则的制定

根据熟练司机的经验法则和模糊表述方法,制定出如下 24 条预见型模糊控制规则。在规则中:

N :表示控制阈值;

NC :表示相对于当前的控制阈值的变化量;

P_n :表示行驶控制刻度盘上的刻度, P_f 表示最大控制值;

B_n :表示刹车刻度盘上的刻度;

B_{max} :表示紧急刹车;

$N(t)$ 是当前控制值。

1) 站间定速行驶规则

富有经验的司机提供了如下操作经验的语言描述:

规律 1:为了确保安全性和乘坐的舒适,当速度高于所限速度时,把控制值调到当前控制值与紧急刹车控制值之间的中间值,如果需要紧急刹车,冲击就会减小。

规律 2:为了节约能源,当可以确保行驶时间时,就利用惯性运行,这时既不加速也不刹车。

规律 3:为了缩短行驶时间,当速度小于所限速度时,则可用最大加速。

规律 4:为了乘坐舒适,如果用当前控制值就可保持车速跟踪目标速度,就可保持当前控制值。

规律5:为了保证速度跟踪性,如果在当前控制下不能达到目标值,就应该在 $\pm n$ 个控制值范围内选择适当的控制值来调节车速,以达到目标值。同时,还要考虑到乘坐舒适,避免加速过大。

根据这些控制规律,就可制定出如下满足模糊控制要求的控制规则:

规则1:如果 $N=0$ 时, $S=G$ 且 $C=G$ 且 $E=G$, 那么 $N=0$;

规则2:如果 $N=P7$ 时, $S=G$ 且 $C=G$ 且 $T=B$, 那么 $N=P7$;

规则3:如果 $N=B7$ 时, $S=B$, 那么 $N=(N(t)+B_{\max})/2$;

规则4:如果 $NC=4$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=4$;

规则5:如果 $NC=3$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=3$;

规则6:如果 $NC=2$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=2$;

规则7:如果 $NC=1$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=1$;

规则8:如果 $NC=0$ 时, $S=G$ 且 $T=G$, 那么 $NC=0$;

规则9:如果 $NC=-1$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=-1$;

规则10:如果 $NC=-2$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=-2$;

规则11:如果 $NC=-3$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=-3$;

规则12:如果 $NC=-4$ 时, $S=G$ 且 $C=G$ 且 $T=VG$, 那么 $NC=-4$ 。

2) 车站停车控制规则

操作经验的语言描述为:当列车通过车站前放置的停车标志后,指示可以开始控制停车定位,但同时要考虑到乘坐舒适。具体根据以下要求来选择控制值。

规律1:为了乘坐舒适性,在通过标志时,应该保持当前的控制值,以避免惯性冲击。

规律2:为了缩短行驶时间,同时考虑到乘坐舒适性,在标志前不要刹车,过了标志开始缓慢刹车。

规律3:为了精确定位,在过了标志后,就应该在 $\pm n$ 个控制值范围内选择适当的控制值来调节车速,以便准确地停车,同时要避免发生惯性冲击。

根据这些控制规律,就可制定出如下满足模糊控制要求的控制规则:

规则1:如果 $NC=+3$ 时 $R=VG$ 且 $C=G$ 且 $A=VG$, 那么 $NC=3$;

规则2:如果 $NC=+2$ 时 $R=VG$ 且 $C=G$ 且 $A=VG$, 那么 $NC=2$;

规则3:如果 $NC=+1$ 时 $R=VG$ 且 $C=G$ 且 $A=VG$, 那么 $NC=1$;

规则4:如果 $NC=0$ 时 $R=VG$ 且 $A=G$, 那么 $NC=0$;

规则5:如果 $NC=-1$ 时 $R=VG$ 且 $C=G$ 且 $A=VG$, 那么 $NC=-1$;

规则6:如果 $NC=-2$ 时 $R=VG$ 且 $C=G$ 且 $A=VG$, 那么 $NC=-2$;

规则7:如果 $NC=-3$ 时 $R=VG$ 且 $C=G$ 且 $A=VG$, 那么 $NC=-3$;

规则8:如果 $N=P7$ 时 $R=VB$ 且 $C=G$ 且 $S=G$, 那么 $N=P7$;

规则9:如果 $N=P4$ 时 $R=B$ 且 $A=B$ 且 $S=G$, 那么 $N=P4$;

规则10:如果 $N=0$ 时 $R=M$ 且 $C=G$ 且 $S=G$, 那么 $N=0$;

规则11:如果 $N=B1$ 时 $R=G$ 且 $C=G$ 且 $S=G$, 那么 $N=B1$;

规则12:如果 $N=B7$ 时 且 $S=VB$, 那么 $N=0$ 。

7.3.4 模糊控制的实现

机车的刹车特性是随着车上人数的多少(即负载)而变化的,但是,控制阈值所对应的驱动能力和刹车的控制性能是可以推算出来的。模糊控制器不断地根据控制规则推出指令,进站定位停车还要受到其他因素的影响,所以,必须要灵活地加以处理。这种预见型模糊控制的推理过程可用图 7.21 来表示。

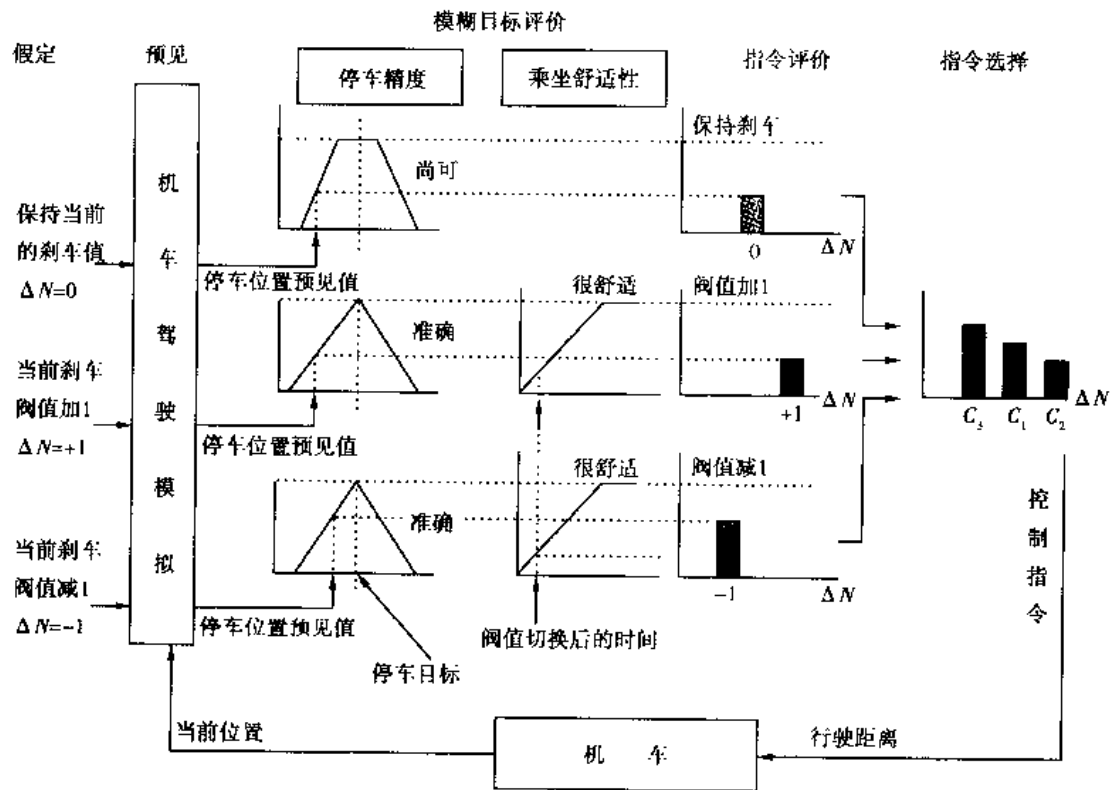


图 7.21 预见型模糊停车控制的推理过程

参考文献

- [1] 黄布毅, 崔光照. 模糊控制技术在家用电器中的应用. 北京: 中国轻工业出版社, 1998
- [2] 李洪兴, 汪群. 工程模糊数学方法及应用. 天津: 天津科学技术出版社, 1993
- [3] 章卫国, 杨向忠. 模糊控制理论与应用. 西安: 西北工业大学出版社, 1999
- [4] 余永权, 曾碧. 单片机模糊逻辑控制. 北京: 北京航空航天大学出版社, 1995
- [5] 何平, 王鸿绪. 模糊控制器的设计及应用. 北京: 科学出版社, 1997
- [6] 章正斌, 吴汝善, 于健. 模糊控制工程. 重庆: 重庆大学出版社, 1995
- [7] 涂承宇, 涂承媛. 模糊控制理论与实践. 北京: 地震出版社, 1998
- [8] 张文修, 梁广锡. 模糊控制与系统. 西安: 西安交通大学出版社, 1998
- [9] 韩启纲, 吴锡祺. 计算机模糊控制技术与仪表装置. 北京: 中国计量出版社, 1999
- [10] 余永权. 模糊控制技术与模糊家用电器. 北京: 北京航空航天大学出版社, 2000
- [11] 李士勇. 模糊控制·神经控制和智能控制论. 哈尔滨: 哈尔滨工业大学出版社, 1996
- [12] 张乃尧, 阎平凡. 神经网络与模糊控制. 北京: 清华大学出版社, 1998
- [13] 谢宋和. 单片机模糊控制系统设计与应用实例. 北京: 电子工业出版社, 1999
- [14] 张晋格. 计算机控制原理与应用. 北京: 电子工业出版社, 1995
- [15] 余永权. 单片机模糊逻辑开发软件. 北京: 北京航空航天大学出版社, 1995
- [16] 冯冬青, 谢宋和. 模糊智能控制. 北京: 化学工业出版社, 1998
- [17] 应浩. 关于模糊控制理论与应用的若干问题. 自动化学报, 2001, 27(4)
- [18] 窦振中. 模糊逻辑控制技术及其应用. 北京: 北京航空航天大学出版社, 1995
- [19] 戎月莉. 计算机模糊控制原理及应用. 北京: 北京航空航天大学出版社, 1995
- [20] 冯保成. 模糊数学实用集粹. 北京: 中国建筑工业出版社, 1991
- [21] 李友善, 李军. 模糊控制理论及其过程控制中的应用. 北京: 国防工业出版社, 1993
- [22] 窦振中. 模糊控制新器件新技术和开发工具. 计算机研究与发展, 1995
- [23] 窦振中. 模糊控制新器件新技术和开发工具. 第二届全国智能控制专家讨论会论文集. 清华大学, 1994(8)
- [24] 王士同, 夏祖勋, 陈剑夫. 模糊数学在人工智能中的应用. 北京: 机械工业出版社, 1991
- [25] 戎月莉, 孙山东. 德国在模糊控制技术方面的发展. 信息与控制, 1993, 22(4)

- [26] R. Turner. 赵沁平译. 人工智能中的逻辑. 北京: 北京大学出版社, 1990
- [27] 贺仲雄, 赵人勇, 李建文, 肖伟中. 模糊数学及其派生决策方法. 北京: 中国铁道出版社, 1992
- [28] J. B. Kiszka, M. E. Kochanska and D. S. Sliwinska. The influence of some parameters on the accuracy of fuzzy model. in Industrial Applications of Fuzzy Control, M. Sugeno, Ed. Amsterdam, North-Holland, 1985
- [29] D. Dubois and H. Prade. Unfair coins and necessity measures; Toward a possibilistic interpretation of histograms. Fuzzy Sets Syst, Vol. 10, No. 1, 1985
- [30] K. L. Tang and R. J. Mulholland. Comparing fuzzy logic with classical controller designs. IEEE Trans. Sys. Man cybern., Vol. SMC-17, No. 1, 1987
- [31] T. Takagi and M. Sugeno. Derivation of fuzzy control rules from human operator's control actions. in Proc. of the IFAC Symp. on Fuzzy Information, Knowledge Representation and Decision Analysis, Marseilles, France, July, 1985
- [32] 何玉彬, 李新忠. 神经网络控制技术及其应用. 北京: 科学出版社, 2000
- [33] 焦李成. 神经网络的应用与实现. 西安: 西安电子科技大学出版社, 1993
- [34] 孙曾祁. 智能控制理论与技术. 清华大学出版社, 广西科学技术出版社, 1997
- [35] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, Proc. Acad. Sci., USA, Vol. 79, 2554 and physical systems with emergent collective computational abilities, Proc. Acad. Sci., USA, Vol. 79, 2554 ~ 2558, 1982
- [36] 冯冬青. 模糊智能控制. 北京: 化学工业出版社, 2000
- [37] 韦巍. 智能控制技术. 北京: 机械工业出版社, 1999
- [38] Yamakawa T. High speed fuzzy controller hardware system. In Proc. 2nd Fuzzy System Symo. Japan, 1986. 122 ~ 130.