

· 国家精品课程教材 ·

智能控制 实验与综合设计指导

师黎 陈铁军 李晓媛 姚利娜 编著

清华大学出版社



· 国家精品课程教材 ·

智能控制 实验与综合设计指导

师黎 陈铁军 李晓媛 姚利娜 编著

清华大学出版社
北京

内 容 简 介

《智能控制实验与综合设计指导》是与本书作者在清华大学出版社出版的《智能控制理论及应用》配套的实验教材。根据作者多年潜心研究与教学实践,在本指导书中安排了模糊控制、神经网络控制、遗传算法等方面的10个综合性实验,同时还提供了相应的MATLAB或C语言程序;结合《智能控制理论及应用》主要章节特点安排了11个综合设计题目,并给出了相应的设计结果。本书力求培养学生的实际动手能力、综合运用所学知识分析问题和解决问题的能力。

本书适合高等院校自动化、计算机和电气类专业的师生以及工程技术人员阅读。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

智能控制实验与综合设计指导/师黎等编著. —北京:清华大学出版社,2008.3

ISBN 978-7-302-16130-1

I. 智… II. 师… III. ①智能控制—实验—高等学校—教材 ②智能控制—课程设计—高等学校—教材 IV. TP273

中国版本图书馆 CIP 数据核字(2007)第 144659 号

责任编辑:刘 彤

责任校对:时翠兰

责任印制:何 芊

出版发行:清华大学出版社 地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编:100084

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印 装 者:北京市清华园胶印厂

经 销:全国新华书店

开 本:185×260 印 张:10.25 字 数:241 千字

版 次:2008 年 3 月第 1 版 印 次:2008 年 3 月第 1 次印刷

印 数:1~3000

定 价:19.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:024834-01

《智能控制实验与综合设计指导》是与《智能控制理论及应用》配套的实验教材。本指导书具有以下特点:

(1) 强调培养学生的工程实践能力。本指导书给出多个智能控制综合性实验,提供了 MATLAB 或 C 语言程序实现,结合智能控制基础主要章节特点给出了 11 个综合设计题目,并给出了设计结果。本书力求培养学生的实际动手能力、综合运用所学知识分析问题和解决问题的能力,达到增强学生的创新实践能力的素质教育要求。

(2) 以面向本科生的智能控制实验和综合设计为出发点,精心选择实验项目,对相关专业的毕业设计也具有一定的指导和参考价值。本指导书简明扼要,通俗易懂。

(3) 以立体形式呈现。本指导书是与《智能控制理论及应用》配套的教辅材料,呈现在师生面前的是教材—电子教案—多媒体课件—课后习题参考答案—实验指导书和综合设计指导书等多元信息资源,为师生构建了良好的教、学平台,使学生能更好地了解、掌握智能控制理论及应用,同时也大大方便了专业教师的教学工作。

本指导书中实验部分共安排了 10 个综合实验项目,每个实验由实验目的、实验设备及条件、实验原理、实验步骤、实验报告要求、思考题组成,并在每个实验的附录部分给出了各个实验的 MATLAB 或 C 语言程序。综合设计部分总结了 11 个综合设计,并给出了相应的设计结果。本实验和综合设计指导书从应用的角度出发,以实用性为重点,遵循由浅入深、循序渐进的原则,力求通俗易懂、方便学习。本书的出版得到研究生王治忠、邵国、王丽佳、欧阳玲、高云婷、王璇、王梓宇、王晓侃、王俊江、余姗姗、冯向荣等的大力支持和帮助,在此一并表示感谢。

作者

2007 年 9 月

综合实验	1
实验一 双容水箱液位模糊控制	3
实验二 基于 BP 神经网络的自整定 PID 控制仿真	14
实验三 基于 MATLAB 的模糊 PID 参数自整定控制	20
实验四 用遗传算法求解函数最优化问题	28
实验五 基于 SIMULINK 的控制系统的双模糊控制	33
实验六 基于 RBF 神经网络整定的 PID 控制	38
实验七 单神经元 PID 控制	44
实验八 基于 ANFIS 的自适应噪声抵消	48
实验九 基于 MATLAB 的自调整因子模糊控制器的设计	52
实验十 基于干扰观测器的 PID 控制	55
综合设计	61
设计一 模糊控制算法研究	63
设计二 Speed Control Design for a Vehicle System Using Fuzzy Logic	73
设计三 模糊免疫 PID 控制设计	83
设计四 模型参考自适应模糊控制系统设计	93
设计五 遗传算法在机器人路径规划中的应用	98
设计六 神经网络用于字母的特征识别	105
设计七 双容水箱的模糊 PID 控制	109
设计八 基于 T-S 模糊模型的递归神经网络在函数拟合中的应用	124
设计九 基于模糊推理的自主移动机器人的避障控制	134
设计十 基于 RBF 神经网络的齿轮箱故障诊断	144
设计十一 Fuzzy Control of a Ball-Balancing System	151

综合实验

实验一

双容水箱液位模糊控制

一、实验目的

1. 熟悉双容液位控制系统的组成原理。
2. 通过实验进一步掌握模糊控制原理及模糊控制规则的生成。
3. 了解量化因子和比例因子对控制效果的影响。
4. 掌握解模糊方法及实现。

二、实验设备及条件

1. TKGK-1 型实验装置。
2. 万用表一只。
3. 计算机系统。

三、实验原理

图 1 所示为双容水箱液位控制系统。控制的目的是使下水箱的液位等于给定值,并能克服来自系统内部和外部扰动的影响。双容水箱液位系统结构如图 2 所示,该被控对象具有非线性和时滞性,要建立精确的数学模型比较困难;模糊控制不仅可以避开复杂的数学模型,通常还能得到比较好的性能指标。模糊控制器的组成如图 3 所示。

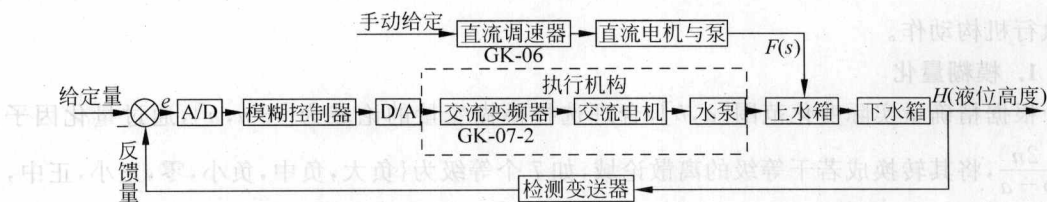


图 1 双容水箱液位模糊控制系统方框图

模糊控制器的输入为误差和误差变化率,误差 $e = r - y$, 误差变化率 $ec = de/dt$, 其中 r 和 y 分别为液位的给定值和测量值。把误差和误差变化率的精确值进行模糊化变成模糊量 E 和 EC , 从而得到误差 E 和误差变化率 EC 的模糊语言集合, 然后由 E 和 EC 模糊语言的子集和模糊控制规则 R (模糊关系矩阵), 根据合成推理规则进行模糊决策, 这样就可以得到模糊控制向量 U , 最后再把模糊量解模糊转换为精确量 u , 再经 D/A 转换为模拟量去控

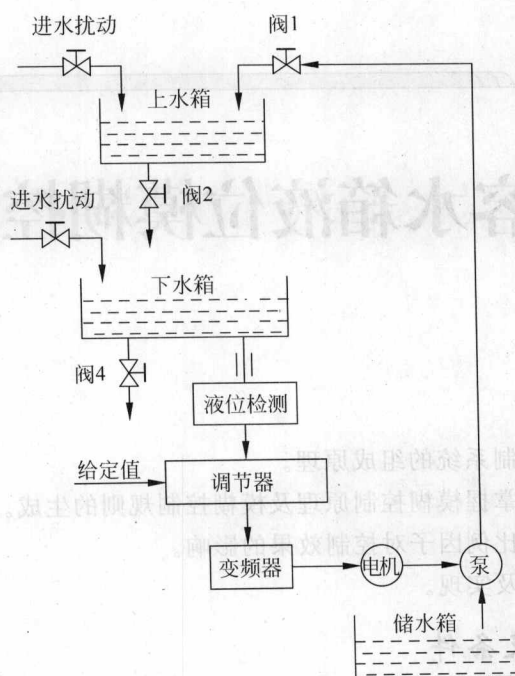


图2 双容水箱液位系统结构

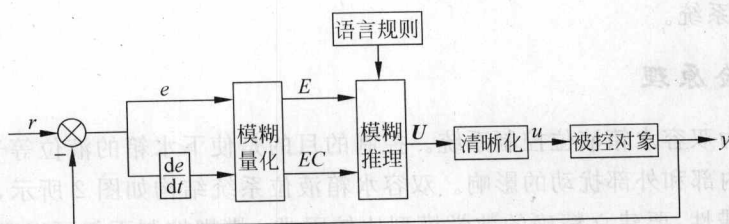


图3 模糊控制器组成原理

制执行机构动作。

1. 模糊量化

根据精确量实际变化范围 $[a, b]$ ，合理选择模糊变量的论域为 $[-n, n]$ ，通过量化因子 $k = \frac{2n}{b-a}$ ，将其转换成若干等级的离散论域，如7个等级为{负大，负中，负小，零，正小，正中，正大}，简称为{NB, NM, NS, O, PS, PM, PB}。确定模糊子集的隶属函数曲线，一般常采用三角形、梯形和正态分布等几种曲线。然后由隶属函数曲线得出模糊变量 E 、 EC 、 U 的赋值表。

根据经验， E 模糊子集的隶属度函数取正态分布曲线，则赋值见表1。

2. 模糊控制规则

模糊控制规则是操作经验和专家知识的总结，是进行模糊推理的依据。在设计模糊控制规则时，必须考虑控制规则的完备性、交叉性和一致性。既保证对于任意给定的输入，均有相应的控制规则起作用；控制器的输出值总是由数条控制规则来决定；控制规则中不存

在相互矛盾的规则。在总结专家经验和过程知识的基础上,可以得到如表 2 所列的控制规则。

表 1 变量 E 隶属度函数赋值表

E	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
NB	1.0	0.7	0.4	0.2	0.1	0	0	0	0	0	0	0	0
NM	0.3	0.6	1.0	0.6	0.3	0.1	0	0	0	0	0	0	0
NS	0	0.1	0.2	0.5	1.0	0.5	0.2	0.1	0	0	0	0	0
O	0	0	0	0	0.1	0.4	1.0	0.4	0.1	0	0	0	0
PS	0	0	0	0	0	0.1	0.2	0.5	1.0	0.5	0.2	0.1	0
PM	0	0	0	0	0	0	0	0.1	0.3	0.6	1.0	0.6	0.3
PB	0	0	0	0	0	0	0	0	0.1	0.2	0.4	0.7	1.0

表 2 控制规则表

$\begin{matrix} U \\ EC \end{matrix} \backslash E$		NB	NM	NS	O	PS	PM	PB
$\begin{matrix} NB \\ NM \\ NS \\ O \\ PS \\ PM \\ PB \end{matrix}$	NB	PS	PS	PS	PS	PM	PB	PB
	NM	NS	PS	PS	PS	PM	PM	PB
	NS	NM	NS	O	O	PS	PM	PM
	O	NB	NM	NS	O	PS	PM	PM
	PS	NB	NM	NS	O	O	PS	PM
	PM	NB	NB	NM	NS	NS	PS	PS
	PB	NB	NB	NM	NS	NS	NS	NS

3. 模糊推理

模糊推理是模糊控制器的核心,模糊控制系统目前常采用的有 CRI 推理的查表法、CRI 推理的解析法、Mamdani 直接推理法、后件函数法等。本实验可采用 Mamdani 直接推理法。Mamdani 直接推理法是先求出模糊关系 R ,再根据输入求出控制量,把控制量清晰化,可得控制查询表。

本实验设计了容量为 7×7 条控制规则表,这个规则表可以用 27 条模糊条件语句来描述。如 if $E=NB$ and $EC=NB$ then $U=PS$ 。对应的模糊关系为 $R_1=A_1 \times B_1 \times C_1$ 。其中, A_1 是 E 的模糊集合; B_1 是 EC 的模糊集合; C_1 是 U 的模糊集合。

根据 $R_i(i=1,2,\cdots,27)$ 可以得到综合模糊矩阵 R 。如下计算模糊关系矩阵 R :

$$R = R_1 \cup R_2 \cup \cdots \cup R_{27}$$

E 和 EC 对应的模糊变量为和,此时可由下式求出:

$$U^* = (A^* \times B^*) \circ R$$

式中 $\circ$ 表示求合成运算。

4. 解模糊

可采用常用的重心法进行解模糊运算。

四、实验步骤

(1) 按图 1 连成控制系统图。其中被控对象为下水箱,被控制量为液位高度,控制器由

计算机系统组成,接口的驱动已在“计算机控制系统”试验中完成,这里不再赘述。

- (2) 启动工艺流程并开启相关仪器,调整传感器输出的零点与增益。
- (3) 运行模糊控制程序,并输入参数,记录实验数据和控制曲线。
- (4) 修改模糊控制规则,重复步骤(3)。
- (5) 待系统稳定后,加手动扰动,观察系统的动态变化,并记录结果。

五、实验报告要求

1. 给出模糊控制器的设计过程和程序清单。
2. 建立 EC 的隶属度函数赋值表。
3. 建立控制查询表。
4. 记录试验数据和曲线。
5. 分析试验结果。

六、思考题

1. 如果输入变量离散论域分为 5 个等级,对计算和控制效果有什么影响?
2. 模糊控制达到稳态后,稳态误差是否为零?为什么?

附录 语言程序清单

```
#include "graphics.h"
#include "conio.h"
#include "math.h"
#include "time.h"
#include "stdlib.h"
#include "dos.h"
#include "stdio.h"
#include "process.h"
#include "bios.h"
int sampt,key,v;
static float cy[600],cx[600],cs[6];
float datain,u,u1,e2,xset,i,Ku;
time_t start,end;
unsigned char dat[6],data_in[5][4];
unsigned char stat,data,data1;
int i1,j1;

main()
{
    int j,m,ntr,T,NUM;
    float adin(),fuzzycontrol(),max,tr,t0;
    void daout(),wct(),wtp(),wait();
    int gdriver,gmode,kk;
```

```

float x1,x2,x3;
char * t[] = {"sampled value:", "seted value:"};
char * tt1[] = {"L", "t"};
gdriver = VGA;
gmode = VGAHI;
initgraph(&gdriver, &gmode, "d:\\tc");
wait();
/* com init */
outportb(0x3fb, 0x80); /* SETB 0x3fb.7 = 1 */
outportb(0x3f8, 0x0C); /* LSB = 0CH BPS = 9600 */
outportb(0x3f9, 0x00); /* MSB = 00H */
outportb(0x3fb, 0x03); /* 8DATA 1STOP NO */
outportb(0x3fc, 0x03); /* SETB DTR = 0 RTS = 0 */
outportb(0x3f9, 0x00); /* SET IE = 0 */
star: inportb(0x3f8);
while(!((0x20 & inportb(0x3fd)) == 0x20));
f_f5:
while(!((0x20 & inportb(0x3fd)) == 0x20));
outportb(0x3f8, 0x66);
while(!inportb(0x3fd) & 0x01);
data1 = inportb(0x3f8);
while(!inportb(0x3fd) & 0x01);
data = inportb(0x3f8);
if(data != 0x35)
{ goto star; }
a_d:
while(!((0x20 & inportb(0x3fd)) == 0x20));
outportb(0x3f8, 0x61);
while(!inportb(0x3fd) & 0x01);
data = inportb(0x3f8);
if(data != 0x64)
{ exit(1); }
adin();
daout(0);

printf("Please Input...\n");
printf("\nSampt(3,4,5,6) = ");
scanf("%d", &sampt);
printf("\nXset = ");
scanf("%f", &xset);
printf("\nKu = ");
scanf("%f", &Ku);

NUM = 500;
setviewport(0, 0, 640, 350, 0);

```

```

clearviewport();
setbkcolor(3);
setcolor(1);
cleardevice();
settextjustify(LEFT_TEXT, TOP_TEXT);
settextstyle(1, 0, 2);
settextjustify(0, 2);

settextstyle(0, 0, 1);
outtextxy(410, 20, t[1]);
outtextxy(410, 40, t[0]);
line(37, 85, 40, 80);
line(43, 85, 40, 80);
line(585, 287, 590, 290);
line(585, 293, 590, 290);

for (j = 1; j < 9; j++)
{
    line(40, 290 - j * 25, 45, 290 - j * 25);
    line(40 + j * 50, 285, 40 + j * 50, 290);
}

line(490, 285, 490, 290);
line(540, 285, 540, 290);
setlinestyle(0, 0, 3);
line(40, 80, 40, 295);
line(35, 290, 590, 290);
settextjustify(0, 2);
settextstyle(1, 0, 1);
outtextxy(25, 70, tt1[0]);
outtextxy(590, 290, tt1[1]);

setcolor(4);
setlinestyle(0, 0, 1);
line(536, 48, 558, 48);
wct(3, 200., 14, 88);
wct(3, 150., 14, 138);
wct(3, 100., 14, 188);
wct(3, xset * 10., 14, 288 - (int)(xset * 10.));
wct(3, 50., 14, 238);
wct(3, 100., 135, 300);
wct(3, 200., 235, 300);
wct(3, 300., 335, 300);
wct(3, 400., 435, 300);
wct(3, 500., 535, 300);
wtp("Xset = ", xset, 5, 23);
wtp("Ku = ", Ku, 25, 23);

```

```

/* wtp("I=",ki,45,23);
wtp("D=",kd,60,23); */
max = 0.0;ntr = 0;

for(i = 0;i<NUM;i++)
{
    cx[i] = xset;
    start = time(NULL);
    t0 = i * sampt;
    x1 = adin( );
    x2 = adin( );
    x3 = adin( );
    cy[i] = (x1 + x2 + x3)/3;
    e2 = xset - cy[i];

    u = fuzzycontrol(e2);

    daout((int)(u));
    putpixel(42 + i,290 - cx[i] * 10,62);
    putpixel(42 + i,290 - cy[i] * 10,4);
    putpixel(42 + i,290 - u * 0.29,1);
    wtp("cy[i]=",cy[i],5,21);
    wtp("Time=",t0,60,21);
    wtp("u=",u,45,21);
    wtp("Num=",i + 1.0,25,21);
    if (i>= 2)
    { if(cy[i]>max) max = cy[i];
      wtp("max=",max,5,25);
    }

    if(ntr == 0)
    { if(cy[i]>= xset)
      {tr = i * sampt;
       wtp("tr=",tr,25,25);
       ntr = 1;
      }
    }

    if(kbhit() != 0)
    {
        key = bioskey(0);
        key = (key>>8)&0xff;

        if(key==44) goto loop2;
    }

loop1:    end = time(NULL);
        if((difftime(end,start))<sampt) goto loop1;
}

```

```

    getch( );
loop2:    daout(0);
    outportb(0x3fc,0x01);
    closegraph( );
    exit(0);
}

float adin ( )
{
    b_d;
    while(!((0x20&inportb(0x3fd)) == 0x20));
    outportb(0x3f8,0x62);
    while(!(inportb(0x3fd)&0x01));
    data = inportb(0x3f8);
    for(i1 = 0; i1 < 5; i1++)
        {for(j1 = 0; j1 < 4; j1++)
            { while(!(inportb(0x3fd)&0x01));
              data_in[i1][j1] = inportb(0x3f8);
            }
        }
    for(i1 = 0; i1 < 5; i1++)
        {for(j1 = 0; j1 < 4; j1++)
            { data_in[i1][j1] = data_in[i1][j1] - 0x30; }
        }
    if(data != 0x64)
        { goto b_d; }
    datain = data_in[0][0] * 10 + data_in[0][1] + data_in[0][2] * 0.1 + data_in[0][3] * 0.01;
    return(datain);
}

float fuzzycontrol(e2)
float e2;
{
    float Xec,Yu,e20,Ke,Kec;
    int E,EC,U,a[13][13];
    int m0 = 6,n = 6;
    int out[13][13] = {
        { 7, 7, 7, 6, 5, 4, 4, 4, 4, 4, 4, 4, 4 },
        { 7, 6, 6, 5, 4, 4, 4, 4, 4, 4, 4, 4, 4 },
        { 7, 6, 5, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4 },
        { 6, 5, 4, 4, 4, 4, 4, 2, 2, 2, 2, 2, 2 },
        { 5, 4, 4, 4, 4, 4, 4, 2, 0, 0, 0, 0, 0 },
        { 4, 4, 4, 2, 2, 2, 2, 2, 0, 1, 1, 1, 1 },
    }
}

```

```

{ 4, 4, 4, 2, 0, 0, 0, 0, 0, -1, 1, 1, 1},
{ 2, 2, 2, 2, 0, 1, 1, 1, 1, 1, 1, 1, 1},
{ 0, 0, 0, 0, 0, -1, 1, 1, 1, 1, 1, 1, 0},
{ 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, -2},
{ 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, -2},
{ 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, -2, -2, -2},
{ 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, -2, -2, -2}};

e20 = xset - cy[i - 1];
Xec = (e2 - e20)/sampt;
Ke = n/2;
Kec = m0/1;

if(e2>2) e2 = 2;
if(e2<-2) e2 = -2;
    if(Xec>1) Xec = 1;
    if(Xec<-1) Xec = -1;

E = Ke * e2;
EC = Kec * Xec;

U = out[E + 6][EC + 6];
    /* printf("U = %d",U); */
Yu = Ku * U;
    if(Yu>255.0) Yu = 255.0;
    if(Yu<-255.0) Yu = -255.0;
    /* printf("Yu = %f",Yu); */
    /* Yu = abs(Yu) * /;
return(Yu);

}

void daout(x)
int x;
{int x1,x2,x3;
c_d:
x1 = (int)(x/100.0);
x2 = (int)((x - x1 * 100)/10.0);
x3 = (int)(x - x1 * 100 - x2 * 10);
dat[0] = x1;dat[1] = x2;dat[2] = x3;
dat[3] = 0;dat[4] = 0;dat[5] = 0;
for(i1 = 0;i1<6;i1++)
    {dat[i1] = dat[i1] + 0x30;}
while(!((0x20&inportb(0x3fd)) == 0x20));

```

```

outportb(0x3f8,0x63);
while(!(inportb(0x3fd)&0x01));
data = inportb(0x3f8);
if(data!= 0x64)
{ exit(1);}
while(!((0x20&inportb(0x3fd)) == 0x20));
outportb(0x3f8,0x64);
for(i1 = 0;i1<6;i1++)

```

```

{
while(!((0x20&inportb(0x3fd)) == 0x20));
outportb(0x3f8,dat[i1]);
}
while(!(inportb(0x3fd)&0x01));
data = inportb(0x3f8);
if(data!= 0x64)
{ exit(1);}
}

```

```

void wct(b,xx,x2,y2)
int b,x2,y2;
float xx;
{char mb2[5];
setcolor(1);
gcvt(xx,b,mb2);
settextjustify(0,2);
settextstyle(0,0,1);
outtextxy(x2,y2,mb2);
}

```

```

void wtp(xc,xt,x1,y1)
char xc[ ];
float xt;
int x1,y1;
{textcolor(9);
gotoxy(x1,y1);
textcolor(1);
printf(" % s % 7.2f",xc,xt);
}

```

```

void wait( )
{int i,j;
printf("\n\n Please Wait For 1 minute !!! \n");

setbkcolor(3);
setcolor(4);

```