

分类号_____ 密 级_____

UDC _____

学 位 论 文

模糊神经网络在电梯群控系统中的应用研究

作 者 姓 名： 张绍谦

指 导 教 师： 杨卫国 副教授

东北大学信息科学与工程学院

申请学位级别： 硕士 学 科 类 别： 工学

学科专业名称： 控制理论与控制工程

论文提交日期： 2009 年 6 月 30 日 论文答辩日期： 2009 年 7 月 4 日

学位授予日期： 2009 年 7 月 答辩委员会主席： 王建辉

评 阅 人： 周玮、张军

东 北 大 学

2009 年 6 月

A Thesis in Control Theory and Control Engineering

Application Research of Fuzzy Neural Network in Elevator Group Control System

by Zhang Shaoqian

Supervisor : Associate Professor Yang Weiguo

Northeastern University

June 2009

独创性声明

本人声明所呈交的学位论文是在导师的指导下完成的。论文中取得的研究成果除加以标注和致谢的地方外，不包含其他人已经发表或撰写过的研究成果，也不包括本人为获得其他学位而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示诚挚的谢意。

学位论文作者签名：

日 期：

学位论文版权使用授权书

本学位论文作者和指导教师完全了解东北大学有关保留、使用学位论文的规定：即学校有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人同意东北大学可以将学位论文的全部或部分内容编入有关数据库进行检索、交流。

作者和导师同意网上交流的时间为作者获得学位后：

半年 ☐ 一年 ☐ 一年半 ☐ 两年 ☐

学位论文作者签名：

导 师 签 名：

签 字 日 期：

签 字 日 期：

模糊神经网络在电梯群控系统中的应用研究

摘 要

随着高层建筑物的日益增多，电梯群在高层建筑和智能大厦中的作用越来越重要，电梯群控系统已经称为国内外研究的热点。本文的目的是根据电梯群控的理论和研究焦点，应用合适的智能算法进行电梯群控制。

本文首先回顾了电梯群控的发展历史和现状，介绍了电梯群控的基本理论和两种常用的电梯群控系统控制方法：模糊逻辑和神经网络，并分析了他们的优缺点。

然后，本文介绍了模糊逻辑和神经网络结合的模糊神经网络，并详细分析了模糊神经网络的结构和推理过程，然后针对本文采用的模糊神经网络模型，笔者编制了相应的 MATLAB 程序并对程序和模型进行了仿真实验以验证其有效性和可靠性。

在完成了前文的理论准备后，本文对模糊神经网络进行混合训练学习，首先使用了 k-均值算法初步确定了隶属函数的中心和宽度，然后使用顺序聚类方法提取模糊规则，最后使用带有动态惯性权值的 PSO 对隶属函数的中心和宽度进行了优化调整。经过混合训练学习方法对模糊神经网络的训练和优化后，可以得到较为完整的模糊神经网络。此处构建的模糊神经网络模型的方法将应用于随后的电梯群控模式识别模块和派梯调度模块。

按照上文提到的模糊神经网络模型构建和训练学习方法根据交通流模式识别的特点和要求建立相应的模糊神经网络，然后通过构建并训练完毕的模糊神经网络对当前交通流特征值分析以得到相应的交通流模式，从而可以根据当前交通流模式制定相应的调度控制策略。

之后按照上文提到的模糊神经网络模型构建和训练学习方法根据电梯群控派梯调度系统的特点和要求建立相应的模糊神经网络，根据前面得到的交通流模式和派梯调度策略，通过模糊神经网络对当前电梯运行状态数据进行的分析从而对呼梯信号序列进行调度。

最后对本文建立的电梯群控系统进行了仿真和数据验证，根据模拟实际系统的运行，验证了本文算法和模型的有效性。

关键词：电梯群控系统，模糊神经网络，PSO 算法，交通流模式识别，电梯群控派梯调度

Application Research of Fuzzy Neural Network in Elevator Group Control System

Abstract

With the development of high-rise buildings, elevator group is playing a more important role in high-rise buildings and intelligent buildings, so the elevator group control system (EGCS) is now the focus of the researchers at home and abroad. In this paper suitable intelligent algorithms are applied in this complicated system according to the theory of EGCS and research focus.

Firstly the history and status quo are reviewed, and the basic theory of EGCS is introduced, then two common methods in EGCS: Fuzzy Logic and Neural Network, are put forward and their advantages and Shortcoming are analyzed.

Then the combine of Fuzzy Logic and Neural Network: Fuzzy Neural Network, is put forward, and also the structure and reasoning process are analyzed. Then the MATLAB program is made use of to verify the effectiveness and reliability of the Fuzzy Neural Network model introduced in this paper.

After the preparation of theory, a mixed learn and train method is used in the Fuzzy Neural Network model. Firstly k-means algorithm is used to get the preliminary centers and widths of the membership functions, then the fuzzy rules are abstracted by the means of sequence clustering, at last, PSO with dynamic inertia weight is applied in the optimization and adjustment of centers and widths of the membership functions. When the mixed learn and train method is finished, a Fuzzy Neural Network model is built. The means used here to build and train a Fuzzy Neural Network will be applied in the following pattern recognition of traffic module and elevator scheduling module.

At first this means is used to build a FNN for the pattern recognition of traffic according to the request and feature of this module, then a result of pattern recognition of traffic through analysing the feature data of traffic is got, so the corresponding strategy will be designed according to the feature of traffic.

The the same means is used to built a FNN for the elevator scheduling module, according to the pattern recognition of traffic and the corresponding strategy, with the help of FNN, the elevator scheduling plan is got through analyzing the state data of elevators

At last, the EGCS is simulated and verified with experimental data, the result proves the effectiveness of the algorithm and model put forward and built in this paper.

Key words: elevator group control system, Fuzzy Neural Network, PSO algorithm, the pattern recognition of traffic, elevator scheduling

目 录

声明.....	I
摘要.....	II
Abstract.....	III
第 1 章 绪 论.....	1
1.1 电梯群控系统的发展和现状.....	1
1.2 电梯群控系统概述.....	1
1.2.1 电梯群控系统的特征.....	1
1.2.1 电梯群控系统的性能评价指标.....	3
1.2.1 乘客交通流数据获取方法.....	4
1.3 本文的目标和研究重点.....	4
1.4 本章小结.....	5
第 2 章 电梯群控系统控制方法.....	6
2.1 电梯群控系统控制思路.....	6
2.2 电梯群控系统调度方法.....	6
2.3 电梯群控系统常用的智能控制方法.....	7
2.3.1 模糊逻辑.....	7
2.3.2 人工神经网络.....	9
2.3.3 模糊神经网络.....	17
2.4 本章小结.....	16
第 3 章 模糊神经网络的学习和仿真方法.....	17
3.1 模糊神经网络的学习算法.....	17
3.1.1 确定隶属函数.....	18
3.1.2 形成模糊规则.....	20
3.2 使用粒子群优化算法 (PSO) 优化调整隶属函数参数.....	22
3.2.1 粒子群算法 (PSO) 的基本概念.....	22
3.2.2 粒子群算法 (PSO) 的算法原理.....	23

3.2.1 粒子群算法 (PSO) 的算法流程.....	24
3.2.1 带惯性权重的粒子群算法 (PSO)	25
3.2.1 动态惯性权值的粒子群优化算法 (DIPSO)	26
3.2.1 动态惯性权值 PSO 算法优化模糊神经网络参数.....	33
3.3 模糊神经网络的仿真.....	34
3.4 本章小结.....	36
第 4 章 电梯群控交通流模式识别.....	37
4.1 电梯群控系统交通模式识别概述.....	37
4.2 应用在交通流模式中的模糊神经网络的结构.....	38
4.2.1 模糊神经网络的输入及其模糊集合.....	38
4.2.2 模糊神经网络的输出.....	40
4.2.3 模式识别中模糊规则的表述.....	40
4.2.4 模式识别中的模糊推理.....	40
4.3 模糊神经网络的模式识别的训练和仿真.....	41
4.3.1 模糊神经网络的训练.....	41
4.3.2 电梯群控交通流模式识别仿真实例.....	43
4.4 本章小结.....	43
第 5 章 电梯群控派梯调度.....	45
5.1 电梯群控系统控制器.....	45
5.2 电梯群控派梯的调度方法.....	45
5.2.1 电梯群控系统的优化目标.....	45
5.2.2 电梯群控系统调度策略.....	46
5.2.3 电梯群控系统派梯调度算法流程.....	46
5.3 应用于电梯群控派梯调度的模糊神经网络结构.....	47
5.3.1 模糊神经网络的输入及其模糊集合.....	47
5.3.2 模糊神经网络的输出.....	49
5.3.3 模式识别中模糊规则的表述.....	50
5.3.4 模式识别中的模糊推理.....	52
5.2.5 模糊神经网络运算结果处理.....	52

5.4 电梯群控系统派梯调度的训练与仿真.....	52
5.4.1 模糊神经网络的训练.....	52
5.4.2 单个呼梯信号的电梯群控派梯调度仿真实例.....	54
5.4.3 针对呼梯信号序列的电梯群控派梯调度仿真.....	57
5.5 本章小结.....	58
第 6 章 结论与展望.....	59
参考文献.....	61
致谢.....	64

第1章 绪 论

1.1 电梯群控系统的发展和现状

自从世界上第一台电梯于问世以来,电梯已经成为人类高层建筑交通中不可缺少的工具,是当今高层建筑的重要组成部分。随着高层建筑的不断发展,楼层的高度越来越高,建筑规模也越来越大,需要合理安装多台电梯并进行集中统一的控制,这种多台电梯群的优化调度系统就是电梯群控系统(EGCS)。

电梯群控系统的发展历史经历了简易自动控制、集选控制和群控几个发展历程。

电梯发展的初期,人们采用了简易自动控制方式,使用的是继电器接触控制,称为“自动选择系统”,通过在上行,下行高峰以及平峰,双向选择运行命令来工作,这是群控的最简单形式:方向预选控制^[1]。但是由于这种方式不能同时响应多个呼梯信号,因此使用起来不方便,效率低下。后来出现了集选控制,这种控制方式能够记录所有的呼梯信号,并在前进方向上根据呼梯顺序停靠,但是这种调度方式效率仍然较为低下。

随着大型建筑物的发展,单台电梯不能很好的应付全部客流,因此需要设置多台电梯,为了控制安装在一起的多台电梯,电梯群控系统便应运而生。20 世纪 70 年代中期以后主要研究的是电梯交通系统的动态特性,并已经进入现代电梯群控系统阶段。1988 年,日本人 Hitoshi Aoki 把人工智能控制技术引入得到电梯群控系统中,建立了专家系统和模糊规则,这标志着电梯群控系统进入了人工智能技术的发展阶段,模糊控制开始应用于电梯群控系统中。1989 年,日本三菱公司把模糊逻辑应用于电梯群控系统中,用来确定乘客密度和电梯交通模式,紧接着,1990 年日本研制出带有模糊控制器的电梯群控系统 ELEX 系列。1992 年,神经网络技术开始应用于电梯群控系统中,其中的代表便是日本东芝公司开发的使用神经网络技术的 EJ-1000FN 系统。现在比较著名的电梯群控系统有:日本三菱公司的 AI-2100 系列和 AI-2200 系列,迅达电梯公司的 Miconic VX/AITP 装置等^[2]。

1.2 电梯群控系统概述

1.2.1 电梯群控系统的特征

电梯群控系统需要对多台电梯进行调度,使他们协同工作,其复杂性是由电梯群控系统的特性所决定的,具体表现在电梯群控系统所固有的非线性、不确定性和多目标性等几个方面。

(1) 电梯群控系统非线性因素

(a) 对同一组厅层呼叫,在不同的时间标度下,轿厢的分配是不同的,并且变化是

非线性的；

- (b) 能够用于分配的轿厢数目有限，受系统轿厢数目限制；
- (c) 轿厢容量是有限的，当轿厢容量达到饱和时，轿厢不会响应呼梯信号；

(2) 电梯群控系统的不确定性因素

- (a) 各层站的乘客数不确定；
- (b) 呼梯者的目的层不确定；
- (c) 呼梯信号的产生层不确定；
- (d) 建筑物内与环境因素有关的交通工况不确定；

(3) 电梯群控系统的不确定随机干扰

- (a) 乘客可能错误的登记了厅呼叫或目的层造成不必要的停站；
- (b) 乘客可能错误地对电梯进行操作，比如造成轿厢门不能关闭，而干扰系统的正常运行等；

- (c) 乘客的登记时间具有随机性；
- (d) 建筑物由于某些偶然因素而使建筑物的交通条件变化。

(4) 电梯群控系统中的模糊性

(a) 电梯轿厢中的乘客人数不能准确获得。虽然轿厢底部装有称重装置，但由于个人的个体体重差异较大，所以很难获得轿厢内乘客数的准确数据。

(b) 乘客进出轿厢的时间因个体不同而不同，同样不能获得准确数据；

(c) 乘客进入轿厢前，其目的层是不可知的。使对乘客乘梯时间的预测和对其他乘客候梯时间和乘梯时间的预测误差较大^[3]。

(5) 电梯群控系统的多目标性

(a) 平均候梯时间要求短。候梯时间是指从乘客按下层站呼梯按钮开始直到系统所指派的电梯到达此层停站所经过的时间。平均候梯时间是指所有候梯时间的平均值。平均候梯时间是评价电梯群控系统重要的性能指标。

(b) 长候梯率要求低。长候梯时间一般是指候梯时间超过 60s 的候梯时间。长候梯率是指长候梯时间发生的百分率。当候梯时间超过 60s，即出现所谓的长时间候梯时，乘客心理烦躁程度急剧上升，所以应尽量减少长时间候梯的发生。

(c) 系统能耗要求低。单台电梯的能耗与所选电梯的驱动方式、机械性能等有关。电梯能量的消耗特征是，电梯全速运行时所消耗的电能耗远远低于减速和加速时的电梯能耗。电梯停靠的次数越多，所消耗的电能耗就越多。所以电梯群控系统节能主要依靠群控系统合理地安排与调度梯群对呼梯信号的响应，尽量减少起停次数，同时起停次数的减少也会延长梯群的整体寿命。

(d) 平均乘梯时间要求短。乘客的乘梯时间是指从乘客进入电梯开始到乘客到达目

的层后离开电梯的时间。乘客乘梯时间的增长往往会使乘客感觉很不舒服、烦躁，所以乘客的乘梯时间应保持在一个特定的期限之内。

(e) 客流的输送能力要求高。电梯的输送能力是电梯的重要指标之一。输送能力的不足往往会造成乘客的拥挤，平均候梯时间长等不良性能。特别是在上行高峰期，客流密度极大，需要电梯系统迅速将乘客送往每个目的层。

(f) 轿厢内拥挤度要求小。轿厢内拥挤度的增大会给乘客带来极大的不方便，所以要减小轿厢内的拥挤度^[4]。

1.2.2 电梯群控系统的性能评价指标

(1) 时间评价指标

(a) 平均候梯时间 T_{aw}

所有乘客的平均候梯时间，用于评价电梯群控系统的性能和效率，要求尽可能短

$$T_{aw} = \frac{1}{N_p} \sum_{i=1}^{N_p} T_w(i) \quad (1.1)$$

$T_w(i)$ 第 i 个乘客的实际候梯时间

N_p 当前总乘客数

(b) 平均乘梯时间 T_{aj}

所有乘客的平均乘梯时间，用于评价电梯群控系统的性能和效率，要求尽可能短。

$$T_{aj} = \frac{1}{N_p} \sum_{i=1}^{N_p} T_j(i) \quad (1.2)$$

$T_j(i)$ 第 i 个乘客的实际乘梯时间。

N_p 当前总乘客数。

(c) 最长候梯时间 T_{mw}

电梯群控系统服务的所有乘客中实际候梯时间最长的，要求尽可能短。

$$T_{mw} = \max(T_w(i) | i=1, 2, \dots, N_p) \quad (1.3)$$

$T_w(i)$ 第 i 个乘客的实际候梯时间。

(2) 能耗评价指标

(a) 总运行距离 D_t

电梯群控系统控制的所有电梯运行距离的总和，用于衡量电梯群控系统的性能和成本，要求尽可能短。

$$D_i = \sum_{i=1}^{N_e} D(i) \quad (1.4)$$

D_i 第 i 台电梯的总运行距离。

N_e 系统电梯总数。

(b) 总能耗 E_i

电梯群控系统控制的所有电梯能耗的总和，用于衡量电梯群控系统的性能和成本，要求尽可能低。

$$E_i = \sum_{i=1}^{N_e} E(i) \quad (1.5)$$

E_i 第 i 台电梯的总能耗。

N_e 系统电梯总数。

1.2.3 乘客交通流数据获取方法

乘客交通流是由电梯群控系统的乘客数、乘客出现的周期以及乘客的分布情况来描述的，是表明大楼乘客流量的量。

具体应用中，交通流的获取主要有以下几种方法

(1) 称重装置。通常来说电梯底部装有称重传感器，以检测轿厢的负载状况，用此称重装置可以粗略的获得客流量信息，假定一个乘客的平均体重，就可以得到在某一楼层进入电梯及离开电梯的人数。但是用这种方法获得的信息是比较粗略的。

(2) 光电检测装置或红外检测装置。在电梯门两侧及顶部和底部装上光电或红外发射及检测装置，可以较为精确的检测乘客进出情况从而获得客流量信息。

(3) 目的楼层呼梯设备。这种设备使乘客在楼厅直接选择目的楼层，通过对呼梯的记录，可以准确地对客流的强度和流向进行跟踪。

(4) 计算机视觉设备。通过应用图像识别技术，可准确获取客流信息。

1.3 本文的目标和研究重点

随着社会的发展和进步，人们对电梯服务质量提出更高的要求。电梯系统不仅要缩短乘客的候梯时间，减少能量损耗，而且要考虑乘客的期望候梯时间和乘客心理等诸多因素。同时电梯的状态随时都是变化的，是一个随机过程，没有统一准确的数学模型进行描述，为了提供舒适的电梯乘坐环境，电梯系统的任务就是要对电梯当前运行状况做出分析评价，对电梯如何调度、如何控制做出相应的合理决策。

基于电梯群控系统的特点，电梯群控没有精确的数学模型描述，所以传统的控制理论在电梯群控制中难以进行有效的应用。所以本文的目标就是参考既有理论和方法，

探讨人工智能方法在电梯群控系统中的应用。

为了实现上述目标，本文通过分析电梯群控的基础理论和控制方法，选定模糊神经网络在电梯群控中的应用作为研究重点。首先介绍一种基于简单模糊推理方法的模糊神经网络，并针对模糊神经网络的训练学习引入了一种全局寻优的动态惯性权值 PSO 算法，然后应用前面提到的方法分别构建应用于电梯群控交通流模式识别模块和派梯调度模块的模糊神经网络，最后对前面构建的电梯群控系统进行仿真验证。

1.4 本章小结

电梯群控已经成为智能大厦工程和自动化控制的研究热点，本章简要的介绍了电梯群控系统的产生背景、发展历史，详细的阐述了电梯群控特点，性能评价指标和交通数据获取方法等电梯群控基本理论。最后本章提出了本文的研究重点和目标。

第2章 电梯群控系统控制方法

2.1 电梯群控系统控制思路

通常情况下电梯群控系统收集层站的呼梯信号及各台电梯的当前工作状态信息，然后按某种控制策略最合理地对各电梯进行工作调度和安排。在规模较大的建筑物内，通常是以4至8台电梯为一组进行群控。当某层站产生乘客召唤信号时，群控装置就会根据轿厢响应和轿厢位置等当前电梯运行状态为依据，通过某种控制策略判断出新产生的召唤信号应分配给哪台电梯。在上述过程中，控制策略往往要根据当前的客流状况来选择，不同的客流模式下，控制策略的侧重点不同。

2.2 电梯群控系统调度方法

(1) 固定程序调度原则

这种群控方式的特点是根据建筑物内乘客流的变化情况把电梯群控系统的工作状态分为几种固定的模式。国内电梯群控系统普遍采用以下的交通特征模式划分方法：

- (a) 上行客流高峰模式，即大量的客流量集中在电梯上行过程中，如上班时间。
- (b) 下行客流高峰模式，即大量的客流量集中在电梯下行过程中，如下班时间。
- (c) 层间客流平衡模式，即客流量均匀的分布在各层之间，如正常上班时间。
- (d) 空闲模式，即电梯很少使用的时间，如节假日或晚上时间。
- (e) 两路交通模式，即大部分的交通量集中在某两层楼层之间，如午餐时间。

(2) 分区调度原则

对于高层建筑，分区是一种改进电梯系统服务性能的常用方法，同时也是电梯群控的一种常见派梯调度原则，其主要有点就是增加电梯系统的乘客输送能力。

分层调度具体可以分为两种调度方法：

(a) 固定分区调度

固定分区就是将建筑物层分成相应电梯运行区域。当无召唤时，各个电梯停靠在自己在服务区域的首层。当某区域出现呼梯信号时，就由负责该区域客流的电梯去响应该呼梯信号。每台电梯服务的区域不是固定不变的，可以根据召唤信号的不同对电梯的服务区域进行调整。这种调度方式相对简单，但是效率不高。

(b) 动态分区调度

动态分区调度就是按照一定的顺序把各个电梯的服务区域结成环状，当电梯系统开始运行后，每台电梯的服务区域随着电梯的位置和运动方向做动态的调整。这种调度原则能够使电梯群控系统的各台电梯能比较均匀的分布于大厦各个楼层。这种调度方式虽

然相对于固定分区调度效率要高，但是不利于节能，而且其控制方法也比较复杂^[5]。

(3) 基于心理待机时间评价的调度原则：

心理待机时间评价就是通过乘客等待时间来估计在等待电梯时间内乘客的心理影响。有相关研究证明，乘客待机焦虑感与待机时间成抛物线关系。采用心理待机评价调度方式，可以在层站产生呼梯信号时，根据某些既定原则进行计算，得到最合理的待机时间评价值，从而准确迅速的计算出最佳响应电梯。

心理待机时间评价有主要以下两种调度原则：

(a) 最小等待原则

根据层站召唤，预测各电梯应答的时间，从中选择应答时间最短的电梯去响应呼梯信号。

(b) 避免长时间侯梯调度原则

这种调度方式通常会根据电梯的速度，建筑物的楼层数、高度以及性质等因素规定一个长侯梯时间，如果乘客侯梯时间超过这个长侯梯时间则判断为长时间侯梯^[6]。

2.3 电梯群控系统常用的智能控制方法

2.3.1 模糊逻辑

在客观世界中，系统可分为确定性系统和模糊性系统。前者可用精确数学模型描述，而后者则不能。模糊逻辑系统是一种自动控制系统，是以模糊数学、模糊语言形式的知识表示和模糊逻辑的规则推理为理论基础，采用计算机控制技术构成的数字控制系统。

由于电梯群控系统本身存在着大量的不确定性，很难得到一个精确的数学模型，因而应用传统的控制算法并不能满足电梯群控系统的优化指标，不能使乘客获得更好的服务。模糊逻辑不需要被控对象的精确数学模型，而是基于专家的操作者的经验建立模糊逻辑模型，通过模糊逻辑推理完成控制决策过程，最后实现对被控对象的调节控制。

模糊逻辑基于固定的专家经验制定的规则进行推理和控制，模糊逻辑的效果和性能很大程度上取决于专家经验的准确性和精度。专家经验虽然是通过实践或实验得到，但是毕竟存在一定程度上的主观因素，而由人的主观获得的规则往往不可靠并且因人而异。

专家经验往往是对某个对象某段时间的描述，缺乏自适应和自我修正的能力，当对象状态改变或出现扰动时，专家经验制定的规则往往不能很好的发挥作用，从而导致电梯群控系统的效率和性能降低。

虽然模糊逻辑适于处理不确定的随机系统，特别是适用于处理缺乏精确数学模型的系统，但是专家经验获取规则的不可靠性、主观性和缺乏自适应性的这些特点限制了模

糊逻辑在电梯群控系统中的应用。

(1) 模糊集合表示方法

Zadeh 表示法：

$$A(x) = \frac{A(x_1)}{x_1} + \frac{A(x_2)}{x_2} + \dots + \frac{A(x_n)}{x_n} \quad (2.1)$$

偶序数表示法：

$$A(x) = \{(x_1, \mu_A(x_1)), (x_2, \mu_A(x_2)), \dots, (x_n, \mu_A(x_n))\} \quad (2.2)$$

向量表示法：

$$A(x) = (A(x_1), A(x_2), \dots, A(x_n)) \quad (2.3)$$

(2) 隶属函数

对于上面提到的模糊集合的概念， μ_A 称为模糊集合 A 的隶属度函数，表示模糊集合中的元素属于该集合的程度。若 A 中的元素用 x 表示，则 $\mu_A(x)$ 称为 x 属于 A 的隶属度。 $\mu_A(x)$ 的取值范围为闭区间[0, 1]，若 $\mu_A(x)$ 接近 1，则表示 x 属于 A 的程度高；若 $\mu_A(x)$ 接近 0，则表示 x 属于 A 的程度低。

隶属函数的主要形式有三角形、梯形、Gauss 型、钟形、S 型、正态型、柯西型等形式。

(3) 模糊蕴含运算

模糊逻辑中，模糊逻辑规则实质上是模糊蕴含关系。设存在模糊控制规则“if x is A then y is B”，则规则表示了 A 和 B 之间的模糊蕴含关系，记为 $A \rightarrow B$ 。常用的模糊蕴含关系是模糊蕴涵最小运算法(mamdani)^[7]。

(4) 模糊关系合成

设 S 是 A 到 B 的模糊关系，Q 是 B 到 C 的模糊关系，则称 $R = S \circ Q$ 为 S 和 Q 的合成关系，模糊关系合成可以使用 mamdani 或 larsen 中的一种或者其他的模糊蕴含关系，比较常见的是模糊蕴含最小-最大运算(Zadeh)，即有如下的表达方式，R 是 A 到 C 模糊关系： $\mu_R(a, c) = \bigvee_{b \in B} (\mu_S(a, b) \wedge \mu_Q(b, c))$ ^[8]。

(5) 模糊化和反模糊化

模糊化的方法：单点模糊集法、就近取整法、隶属度函数线性插值法、单值隶属度法、模糊向量真值修正法根据不同的模糊推理方法选择不同的模糊化方法

反模糊化方法：最大隶属度法、面积平均法、加权平均法

其中加权平均法最常用

(6) 典型的模糊逻辑方法:

(a) 对采集的数据进行模糊化，通过隶属函数的作用将数据模糊化为模糊量

- (b) 根据专家经验制定模糊规则，规则的表达形式为 if-then 形式
- (c) 根据模糊规则进行模糊蕴含运算
- (d) 根据命题要求对模糊量和模糊推理前件进行模糊关系运算的到模糊推理结果
- (e) 对模糊推理结果进行反模糊化运算，得到精确结果^[9]

2.3.2 人工神经网络

人工神经网络模型用于模拟人脑活动的过程，其中包括对信息的加工、处理、存储和搜索等过程，具有如下基本特点：

(1) 人工神经网络具有分布式存贮信息的特点。人工神经网络存储信息的方式与传统的计算机是不同的，信息不是保存在一个地方，而是分布在不同的位置。人工神经网络是用大量神经元之间的连结及对各连接权值的分布来表示特定的信息。因此，这种分布式存贮方式即使当局部网络受损时，仍具有能够恢复原来信息的特点。

(2) 人工神经网络对信息的处理过程具有并行的特点。每个神经元都可根据接收到的信息作独立的运算和处理，然后将结果传输出去。人工神经网络对于某一个输入，通过计算产生一个输出，各个输出节点代表的逻辑概念被同时计算出来。在输出模式中，通过输出节点的本身信号和本身信号的强弱而得到特定的解，同时排除多余的解。这体现了人工神经网络并行推理的特点。

(3) 人工神经网络对信息的处理具有自组织、自学习的特点。人工神经网络中各神经元之间的联结强度用权值大小来表示，这种权值可以事先定出，也可以通过大量的实际数据通过训练学习得来，同时可以适应周围环境而不断地变化。人工神经网络所具有的自学习过程模拟了人的形象思维方法，这是与传统符号逻辑完全不同的一种非逻辑非语言的方法。

总之，人工神经网络是以对信息的分布存贮和并行处理为基础的，具有自组织、自学习的功能的一种人工智能方法。在许多方面更接近人对信息的处理方法，具有模拟人的形象思维能力，反映了人脑功能的若干基本特性，但并不是人脑的逼真系统，而只是人脑的某种抽象、简化和模拟^[10]。

人工神经网络特有的非线性、适应性和信息处理能力，克服了传统人工智能方法对于知觉(如模式、语音识别、非结构化信息处理方面)的缺陷，使之在神经专家系统、模式识别、智能控制、组合优化、预测等领域得到成功的应用。人工神经网络与其他传统方法相结合，将推动人工智能和信息处理技术不断发展。近年来，人工神经网络正向模拟人类认知的道路上更及深入的发展，与模糊系统、遗传算法、进化机制等结合，形成计算智能，成为人智能的一个重要方向，将在实际中得到发展。

电梯群控系统是一个随机性很强的非线性系统，没有精确的数学模型，人工神经网络的特点正适合处理这类问题。在全世界研究者和电梯公司的共同努力下，人工神经网络

络用于电梯群控系统已经较为成熟。但是人工神经网络本身仍存在着结构未知，网络规模不易控制，容易陷入局部最优等缺点。

(1) 人工神经网络基本概念

(a) 神经元

人工神经网络是由大量神经元广泛互连而成的复杂网络系统。单一神经元可以有許多输入、输出。神经元之间的相互作用通过连接的网络权值体现。虽然单个神经元的结构和功能极其简单有限，而大量神经元构成的网络系统其行为却是丰富多彩的。

(b) 网络权值

表示神经元之间的连接强度和相互作用。

(2) 人工神经网络的分类

根据功能特性和学习特性来分主要有感知器、线性神经网络、BP 神经网络、径向基函数网络、自组织映射网络和反馈神经网络等。

根据网络的拓扑结构来分主要有前向网络、从输入到输出有反馈的网络、层内互联网络、互联网络等。

根据网络输入或网络层中是否有延迟或反馈环节可以分为静态网络和动态网络

(3) 人工神经网络的学习

处理人工神经网络的学习主要分为两个阶段：第一阶段是学习期，此时各计算单元的状态不变，各连接线上的权值可通过学习来修改；第二阶段是工作期，此时各连接权值固定，计算单元变化，以达到某种稳定状态。因此神经网络的学习是研究的主要方面。学习方式按环境所供信息的多少分为以下三种：

(a) 督导学习有教师学习

督导学习是最为常用的学习方式。这种学习方式需要外界存在一个“教师”它可对一组给定的输入数据提供应有的输出结果。这组已知的输入、输出数据称为训练样本集。学习系统可根据已知的输出与实际输出的差值来调整系统参数。

(b) 非督导学习无教师学习

非监督学习时不存在外部教师，学习系统完全按照环境所提供数据的某些统计规律来调节自身参数或结构，以表示外部输入的某种固有特性。

(c) 再励学习或强化学习

这种学习介于上述两种学习情况之间，外部环境对系统输出结果只给出评价奖或罚而不是给出正确答案，学习系统通过强化那些受奖励的动作来改善自身性能。有时也将这种学习方式归于监督学习^[11]。

2.3.3 模糊神经网络

神经网络具有并行计算, 分布式信息存储, 容错能力强以及具备自适应学习等一系列的优点。但一般来说, 神经网络不适合表达基于规则的知识, 因此在神经网络的训练时, 由于不能很好的利用既有经验和知识, 常常只能将初始权值取为零或随机数, 这种处理方式增加了网络的训练时间, 并且有可能陷入局部极值。这是神经网络的不足。另一方面, 模糊逻辑也是一种处理不确定性, 非线性等问题的有力工具。它比较适合于表达那些模糊或定性知识, 其推理方式比较适合于人的思维模式, 但是一般说来模糊逻辑系统缺乏学习和自适应能力。由此可以想到, 若能将模糊逻辑和神经网络适当的结合起来, 综合二者的长处, 应该可以得到比单独的神经网络系统或单独的模糊系统更好的系统。这样就诞生了模糊神经网络系统。

与传统的神经网络模型不同, 模糊神经网络的结构和权值都有一定的实际的逻辑含意义。在设计模糊神经网络结构时, 可以根据问题的复杂程度以及精度要求, 并结合既有经验和知识来构造相应的模糊神经网络模型。这样, 网络的学习速度会得到改善, 并在一定程度上回避了梯度优化算法带来的局部极值问题^[12]。

(1) 模糊神经网络的分类

根据模糊和神经网络连接的形式和使用功能, 两者的融合形态可归纳成以下五大类:

(a) 松散型结合。在同一系统中, 对于可用“if-then”规则来表示的部分用模糊系统描述, 而对很难用“if-then”规则表示的部分, 则用神经网络, 两者之间没有直接联系。

(b) 并联型结合模糊系统和神经网络在系统中按并联方式连接即享有共同的输入。按照两者在系统中所起的作用和地位, 还可分为同等型和补助型。

(c) 串联型结合。模糊系统和神经网络在系统中按串联方式联接, 即一方的输出成为另一方的输入, 这种情况可以看做是两段推理或串联中的前者作为后者的输入信号的预处理部分。

(d) 网络学习结合型。模糊系统由一个等价的人工神经网络表示。这里使用的人工神经网络不再是一个暗箱, 其所有的参数和神经元都有一定的意义, 对应模糊系统中的隶属函数或推理过程。结合以后的人工神经网络能够清晰的体现模糊逻辑, 这种结合方式最能体现人工神经网络与模糊逻辑的结合, 是应用最为广泛的模糊神经网络模型^[13]。

本文所采用的模糊神经网络模型基于网络学习结合型, 在保持模糊逻辑特点和优势的情况下, 使用神经网络模型来针对模糊逻辑缺乏自适应性的缺点进行改进, 整个网络的节点、权值和结构都符合模糊逻辑的基本概念, 能够清晰的表达出模糊逻辑。

(2) 本文采用的模糊神经网络算法基础

模糊神经网络要体现模糊推理的逻辑, 理应在网络结构上体现出的运算完全遵循模

糊推理过程。但是受到实际情况和具体应用的限制，and 连接的多重条件语句如果在运算网络化中严格的遵循模糊推理过程就会导致网络结构复杂庞大，并降低网络的运算效率和可靠性，所以本文在这里采用一种简化的模糊推理过程。

设输入向量 $x=[x_1, x_2 \dots x_n]^T$ ，每个分量 x_i 均为模糊语言变量，并设其中某个 x_i 的模糊集合为：

$$T(x_i) = \{A_i^1, A_i^2 \dots A_i^{m_i}\}, i=1, 2, \dots n \quad (2.4)$$

其中， $A_i^j (j=1, 2, \dots m_i)$ 是 x_i 的第 j 个语言变量值，是定义在论域 U_i 上的一个模糊集合，其相应的隶属度为 $\mu_{A_i^j}(x_i) (i=1, 2, \dots n; j=1, 2, \dots m_i)$

输出量 y 也是模糊语言变量且其模糊集合为

$$T(y) = \{B^1, B^2 \dots B^{m_y}\} \quad (2.5)$$

其中， $B^j (j=1, 2, \dots m_y)$ 是 y 的第 j 个语言变量值。相应的隶属度为 $\mu_{B^j}(y)$ 。

设第 i 条模糊规则 R_i 为

$$\text{If } x_1 \text{ is } A_1^{i_1} \text{ and } x_2 \text{ is } A_2^{i_2} \dots \text{ and } x_n \text{ is } A_n^{i_n}$$

$$\text{Then } y \text{ is } B_i$$

其中 $i=1, 2, \dots m$ ， m 表示规则总数， $m \leq m_1 m_2 \dots m_n$ ， $A_j^{i_j}$ 表示第 i 条规则的第 j 个模糊语言变量的第 i_j 模糊子集。

这样对于给定的输入 x ，可以求得对于每条规则的适用度：

$$\alpha_i = \mu_{A_1^{i_1}}(x_1) \wedge \mu_{A_2^{i_2}}(x_2) \dots \wedge \mu_{A_n^{i_n}}(x_n) \quad (2.6)$$

$$\text{或 } \alpha_i = \mu_{A_1^{i_1}}(x_1) \mu_{A_2^{i_2}}(x_2) \dots \mu_{A_n^{i_n}}(x_n) \quad (2.7)$$

对于给定的输入，只有在输入点附近的那些语言变量值才有较大的隶属度值，远离输入点的语言变量值的隶属度很小或者为 0，当隶属度函数很小时近似的取为 0。因此只有当输入 x 的模糊语言变量与第 i 条规则中对应的语言变量的模糊子集接近的时候，输入 x 对该规则适应度 α_i 才不为零。这样就完成了模糊前件和模糊规则的匹配。

得到模糊前件对第 i 条模糊规则的适应度之后，通过如下公式进行模糊逻辑推理：

$$\mu_{B_i}(y) = \alpha_i \wedge \mu_{B_i}(y) \quad (2.8)$$

$$\text{或 } \mu_{B_i}(y) = \alpha_i \mu_{B_i}(y) \quad (2.9)$$

其中，公式右边的 $\mu_{B_i}(y)$ 第 i 条规则中的输出，其值一般是输出量的模糊集合子集的中心值^[14]。

(3) 模糊神经网络模型

模糊表示的表示主要有两种：一种是模糊规则的后件是输出量的某一模糊集合，称为模糊系统的标准模型或 **mamdani** 模型；另一种是模糊规则的后件是输入语言变量的函数，典型的情况是输入变量的线性组合，称为 **Takagi-Sugeno** 模型。

本文采用的模糊神经网络模型为 **mamdani** 模型，其多输入单输出的 **MISO** 结构如图 2.1 所示：

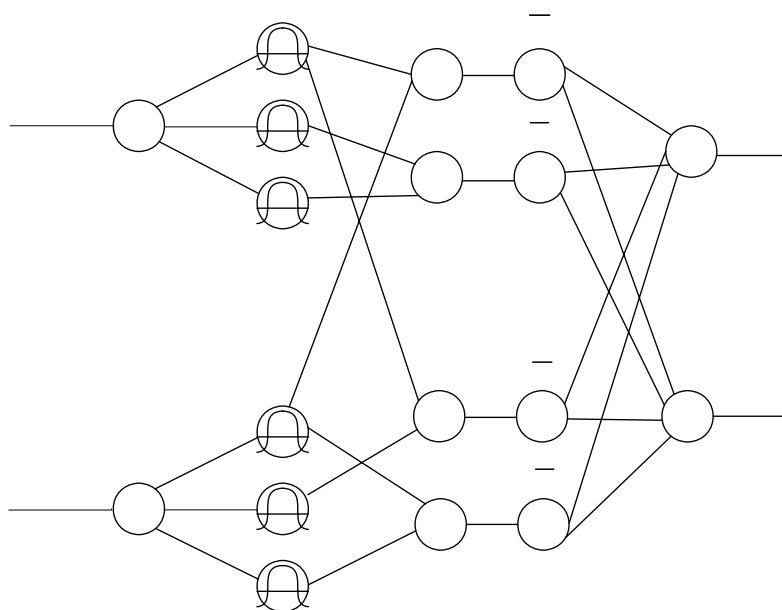


图 2.1 mamdani 模糊神经网络结构

Fig.2.1 Structrue of mamdani FNN

对于上述网络模型，定义一个神经元的基本功能，一个典型的神经网络由一组神经元组成。神经元的结构如图 2.2 所示

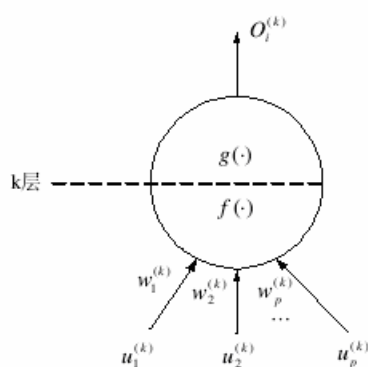


图 2.2 神经元结构

Fig.2.2 Structrue of Neurons

(4) 模糊神经网络网络结构

第一层：输入层。这一层直接把输入值传递到下一层，其神经元的个数 n_1 为输入变量的个数。

$$f_k^{(2)} = x_k^{(1)}, \quad g_k^{(1)} = f_k^{(1)} \quad (1 \leq k \leq n_1) \quad (2.10)$$

其中 $x_k^{(1)}$ 是第 k 个输入变量值, 连接权值 $\omega_k^{(1)} = 1$ 。

第二层: 输入语言变量层, 又称模糊化层。作用是计算各输入分量属于各语言变量值模糊集合的隶属函数。其神经元的个数 n_2 和输入变量个数以及每个输入变量的模糊子集个数有关, 如果选择每个输入变量的模糊子集个数相同 ($|T(x_i)| = t, i = 1, 2, \dots, n_1$), 则 $n_2 = n_1 * t$ 。每个神经元代表一个模糊子集, 如果选择高斯型函数作为隶属函数, 则

$$f_k^{(2)} = M_{xi}^j(m_{ij}, \sigma_{ij}) = -\frac{(x_i^{(2)} - m_{ij}^{(2)})^2}{\sigma_{ij}^{(2)}}, \quad g_k^{(2)} = e^{f_k^{(2)}} \quad (1 \leq k \leq n_2) \quad (2.11)$$

其中, $m_{ij}^{(2)}$ 和 $\sigma_{ij}^{(2)}$ 分别为 x 的第 i 个输入变量的第 j 个模糊子集隶属函数的中心和宽度。连接权值 $\omega_k^{(2)} = m_{ij}^{(2)}$ 。此时, i 、 j 、 k 的关系为

$$i = \frac{k-1}{n_2+1}, \quad j = (k-1)\%n_2 + 1 \quad (2.12)$$

第三层: 规则层。这一层中的连线用来执行模糊逻辑规则前提条件的匹配, 计算出每条规则的适用度。其神经元的个数 n_3 等于规则数, 最大的规则数为 $n_2^{n_1}$ 。

$$f_k^{(3)} = \alpha_i = \mu_{A_1^{i_1}}(x_1) \wedge \mu_{A_2^{i_2}}(x_2) \dots \wedge \mu_{A_n^{i_n}}(x_n) \quad (2.13)$$

$$\text{或者 } f_k^{(3)} = \alpha_i = \mu_{A_1^{i_1}}(x_1) \mu_{A_2^{i_2}}(x_2) \dots \mu_{A_n^{i_n}}(x_n) \quad (2.14)$$

$$g_k^{(3)} = f_k^{(3)} \quad (1 \leq k \leq n_2) \quad (2.15)$$

其中, $u_{kj}^{(3)}$ 表示第 k 个神经元的第 j 个输入, 连接权值 $\omega_k^{(3)} = 1$ 。

对于给定的输入, 只有在输入点附近的那些语言变量值才有较大的隶属度值, 远离输入点的语言变量值的隶属度很小或者为 0, 当隶属度函数很小时近似的取为 0。因此第三层的所有输出中只有很少神经元的输出非 0, 而多数节点输出为 0。

第四层: 归一化层。这一层的神经元数目和第三层是一样的 $n_4 = n_3$, 其所实现主要功能是归一化计算, 以为后面的模糊推理和输出做准备。

$$f_k^{(4)} = \overline{\alpha_j} = \frac{\alpha_j}{\sum_{i=1}^m \alpha_i}, \quad j = 1, 2, \dots, m, \quad g_k^{(4)} = f_k^{(4)} \quad (1 \leq k \leq n_4) \quad (2.16)$$

其中 m 为规则数即 $m = n_4 = n_3$

第五层: 输出层。同时完成模糊逻辑推理的最后一步。

$$f_k^{(5)} = y_i = \sum_{j=1}^m \omega_{ij} \overline{\alpha_j}, i=1,2,\dots,r, \quad g_k^{(5)} = f_k^{(5)} \quad (1 \leq k \leq n_5) \quad (2.17)$$

与前面所给出的标准模糊模型的推理过程相比较,这里的 ω_{ij} 相当于 y_i 第 j 个语言值隶属函数的中心值, ω 的每一行对应输出 y 的模糊子集, ω 的每一列对应模糊规则的输出结果,写成向量的形式则为

$$y = \omega \circ \overline{\alpha} \quad (2.18)$$

其中

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_r \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega_{11} & \omega_{12} & \dots & \omega_{1m} \\ \omega_{21} & \omega_{22} & \dots & \omega_{2m} \\ \vdots & \vdots & \dots & \vdots \\ \omega_{r1} & \omega_{r2} & \dots & \omega_{rm} \end{bmatrix}, \quad \overline{\alpha} = \begin{bmatrix} \overline{\alpha_1} \\ \overline{\alpha_2} \\ \vdots \\ \overline{\alpha_m} \end{bmatrix} \quad (2.19)$$

(5) 模糊神经网络模型解释

上面的模糊神经网络标准模型严格的对应了模糊逻辑推理过程的各个步骤,清晰的反映了模糊逻辑各步骤之间的逻辑关系。

其中第一层输入层和第二层模糊化层对应的是模糊逻辑中的“对采集的数据进行模糊化,通过隶属函数的作用将数据模糊化为模糊量”这一条,其作用就是将对象的输入变量经过隶属函数的处理模糊化为模糊逻辑中使用的模糊量。整个模糊逻辑推理的过程使用的都是模糊量进行推理,本文采用的标准模糊神经网络结构也不例外,网络内部的所有运算量都以模糊量的形式出现。

第三层对应了模糊逻辑推理中的“根据模糊规则进行模糊蕴含运算”,本文所采用的标准模糊神经网络模型在网络的第三层使用简化的模糊推理方法进行模糊前件与模糊规则的匹配。

第五层对应了模糊逻辑推理中的“根据命题要求对模糊量和模糊推理前件进行模糊关系运算的到模糊推理结果”,采用前文叙述过的简化模糊推理算法,完成了模糊逻辑推理 $\mu_{B_i}(y) = \alpha_i \wedge \mu_{B_i}(y)$ 或者 $\mu_{B_i}(y) = \alpha_i \mu_{B_i}(y)$ 。

在网络训练过程中,对第三层和第四层的训练对应了“根据专家经验制定模糊规则,规则的表达式为 if-then 形式”。通过专家经验的数据对网络进行训练,从而得到相应的模糊规则,模糊规则的表达式就由第三层和第四层形成的逻辑完成。本文所采用的标准模糊神经网络模型与模糊逻辑推理在模糊规则获取方式不同之处就在于,模糊逻辑推理使用的规则是根据专家经验人为制定的,包含有很大主观因素,不同的人可能会由相同的专家经验得到不同的模糊规则,而模糊神经网络则通过网络学习的方法自行获取模糊

规则，最大程度的避免了人为主观因素的干扰，使模糊规则的获取只取决于网络结构和训练算法，从而使模糊规则的获取更具合理性和客观性，给模糊逻辑推理系统赋予了可贵的自适应性。

2.4 本章小结

本章在前一章电梯群控基本理论的基础上介绍了电梯群控系统常用的控制思路 and 调度原则，并详细说明了近年来在电梯群控系统中常用的智能算法：模糊逻辑，人工神经网络和模糊神经网络。通过对以上三种智能算法的分析，选择模糊神经网络作为本文主要的智能算法，然后详细的对本文采用的模糊神经网络模型的结构、算法进行了论述。

第3章 模糊神经网络的学习算法和仿真方法

前文的模糊逻辑介绍、神经网络介绍、模糊神经网络算法和结构的介绍都是本章的理论准备工作，本章将向读者展示一个模糊神经网络的学习算法和仿真的全过程，电梯群控系统模式识别和派梯调度都将以本章构建模糊神经网络的方法为基础。当一个mamdani网络的算法和结构确定之后，接下来的工作就是针对这个网络的学习和训练。

3.1 模糊神经网络的学习算法

在前文的研究中，模糊神经网络结合了模糊逻辑对缺乏准确数学模型的非线性问题的解决能力和神经网络的自适应性，形成了结构和功能上具有实际意义的网络结构。模糊神经网络构建以后要应用于实际必须使用训练数据进行相应的学习算法。

对于模糊神经网络，传统上使用的是混合学习算法，其基本思路是通过聚类算法来初步的确定模糊神经网络隶属函数的中心和宽度，然后使用初步确定的隶属函数通过规模规则生成算法来获取模糊规则，最后使用一种智能算法对结构已经完整的模糊神经网络的参数(如隶属函数的中心和权值)进行调整^[15]。

本文构建模糊神经网络采用常用的混合学习算法，首先通过k-均值方法确定隶属函数的初值；然后通过顺序聚类算法获取模糊规则；最后采用粒子群优化算法(PSO)学习算法优化隶属函数，以产生期望的输出。

具体的混合算法流程图如图3.1所示：

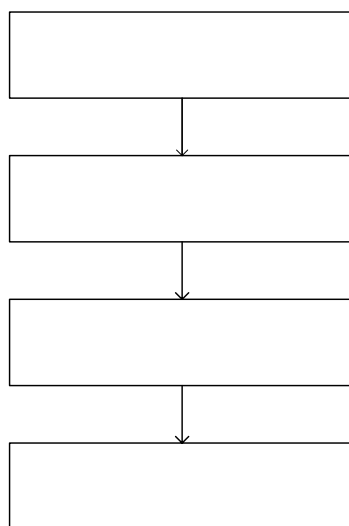


图 3.1 混合算法流程图

Fig.3.1 Flow diagram of mixed learn and train algorithm

3.1.1 确定隶属函数

根据输入样本初步决定高斯型隶属函数的中心 m_{ij} 和宽度 σ_{ij} ，从而大体上确定隶属函数的形状。本文采用 k-均值聚类算法来调整中心，此算法将训练样本集中的输入向量分为若干族，在每个数据族内找出一个高斯型隶属函数的中心向量，使得该族内各样本的向量距离该族中心的距离最小。

算法的流程图如下图 3.2 所示：

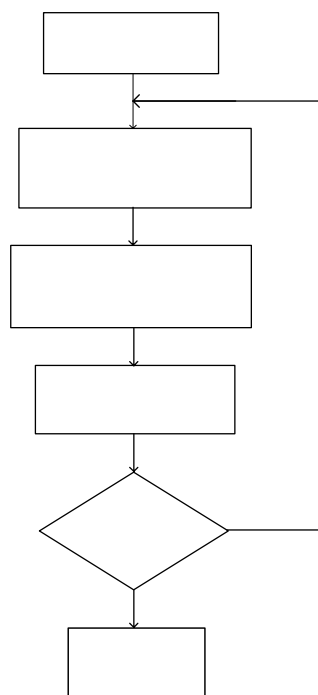


图 3.2 k-均值聚类算法流程图

Fig.3.2 Flow diagram of k-means algorithm

具体算法如下：

(1)从输入样本 $x(k)$ ， $k=1, 2, \dots, k$ (k 为输入样本总数)中选取 1 组数据作为初始的网络中心 $c_j(0)$ ， $j=1, 2, \dots, k$ 。但是要注意：因为 k-均值聚类算法对于初始的聚类中心比较敏感，所以在选取初始中心的时候，要根据实际情况也就是电梯客流量的特点按照时间顺序，以一定的时间段内样本的均值作为初始的中心，这样的选取可以有效的避免因初始聚类中心选取不当而对网络的逼近产生较大的误差。

(2)将输入的样本按照临近原则分组，即将每一个输入样本 $x(k)$ ， $k=1, 2, \dots, k$ (k 为输入样本总数)与各个聚类中心 $c_j(i)$ ，($j=1, 2, \dots, k$ ； $i=0, 1, \dots$)按照下式求出它们之间的距离

$$d_{kj}^{(i)} = \|x(k) - c_j^{(i)}\| \quad (3.1)$$

其中， i 为确定聚类中心的次数，由下面的(4)的算式确定，将输入样本 $x(k)$ 归为距

离最小的那个聚类中心的一组。

(3)按照下式计算新的聚类中心

$$c_j^{(i+1)} = \frac{1}{M_j^{(i)}} \sum_{k_j=1}^{M_j^{(i)}} x_j(k_j) \quad (3.2)$$

其中， $M_j^{(i)}$ 为输入样本 $x(k)$ 中在第 i 次确定网络中心时属于第 j 个类的样本数，这里的 $x_j(k_j)$ 为样本 $x(k)$ 中属于第 j 个类的样本。

(4)计算

$$\varepsilon_i = \|c_j^{(i)} - c_j^{(i-1)}\| \quad (3.3)$$

如果 ε_i 小于给定值，则 k -均值聚类算法完毕，否则转入第二步。

当经过上述处理后我们按照下式分别求出高斯型隶属函数的中心和宽度

$$c_j = \frac{\sum_{x \in \theta_j} x}{M_j} \quad (3.4)$$

$$\delta_j^2 = \frac{1}{M_j} \sum_{x \in \theta_j} (x - c_j)^2 \quad (3.5)$$

其中， M_j 是 j 组的样本数， θ_j 是第 j 组所有样本^[16]。

本文构建的模糊神经网络第二层神经元的激活函数采用图 3.3 的高斯型隶属函数。

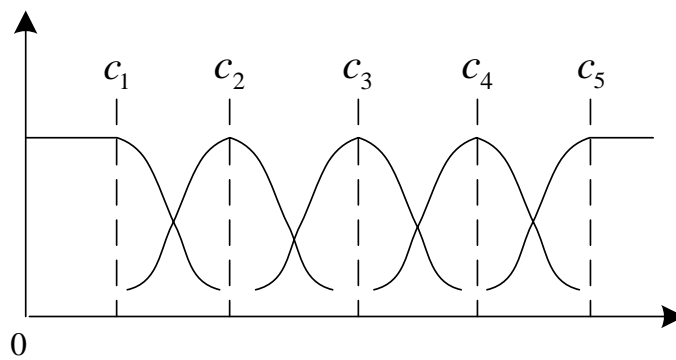


图 3.3 隶属函数基本图形

Fig.3.3 Basic shape of membership function

在上图中，隶属函数中叠加在一起的每一个高斯型函数对应模糊集合中的一个模糊子集，它们有各自的中心和宽度。

在训练中，我们要对每一个神经元的隶属函数中的高斯型函数单独进行训练，分别确定它们的中心和宽度。

基于以上思路对基本的 k-均值算法根据本文采用的模糊神经网络的特点进行应用。

在 k-均值算法中，第一步要选择几组样本数据作为中心，对于 k-均值算法初始的中心非常重要，所以在本文的具体实践中就选择电梯乘客流中典型的数据作为中心来进行 k-均值运算。一个输入对应的第二层神经元的数目即是输入量的模糊子集的数目，输入量的隶属函数有模糊子集就要根据实际电梯乘客流的情况选择几个典型数据作为中心，然后根据 k-均值算法对来初步的确定隶属函数的中心和宽度。

隶属函数的中心和宽度是模糊神经网络中重要的网络参数，由于其处在第二层，没有这部分参数网络后面的运算无法进行，所以需要首先将这部分参数确定。

具体算法应用本文将会在后面的电梯群控系统模式识别模块和派梯调度模块中使用详细的例子来进行叙述。

3.1.2 形成模糊规则

在隶属函数的参数确定以后，根据模糊规则的顺序聚类算法，确定出模糊逻辑规则。这种算法聚类数目事先未知，而是随着算法的进行生成新的聚类，所有的样本只参与算法一次。算法的思想是首先把第一个样本作为第一个聚类的中心，然后考察剩余的每个样本，根据样本到已经形成的聚类中心的距离，考虑将它分配到一个已有的聚类中，或者新生成一个聚类中心。该算法的最大优点是可以自动确定聚类的数目，并且算法非常直接和快速。缺点是算法的结果受样本的出场顺序影响较大。原因是对于每个样本的分配是在最后的聚类形成之前决定的，而最后的聚类是在所有的向量都处理后才形成的。

本文采用的模糊神经网络，需要从大量的数据中抽取模糊规则，模糊规则的表达方法就是第二层模糊化层与第三层规则层之间的权值连接和第四层和第五层之间的权值连接。

具体使用中，首先根据训练数据使用顺序聚类法将训练数据聚类，聚类的数据既是规则的数目，并根据聚类结果生成模糊规则，然后根据实际情况来对模糊规则进行合并和删除，并简化模糊规则，最后根据模糊规则确定第二层的模糊化层与第三层规则层之间的权值和第四层与第五层之间的权值^[17]。

算法的流程如图 3.4 所示：

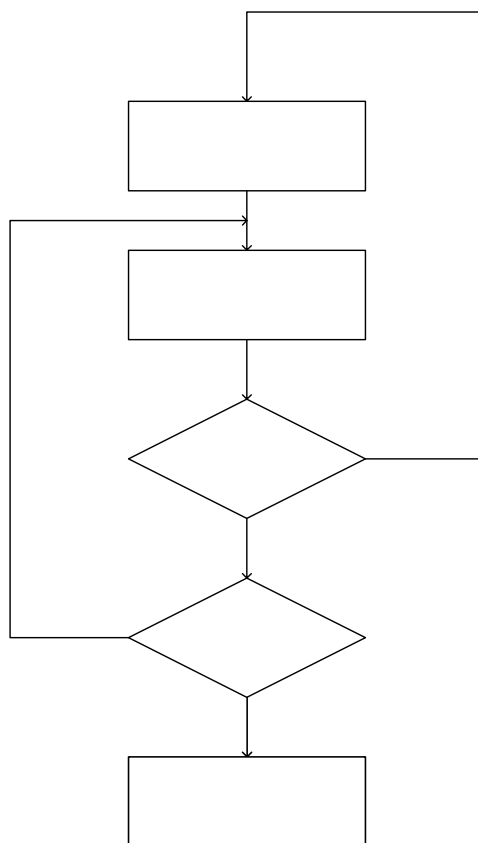


图 3.4 顺序聚类算法流程图

Fig.3.4 Flow diagram of sequence clustering algorithm

聚类的方法如下：

(1)对某一训练数据判断其是否属于某个类，对训练数据向量的每个值计算其与某个类中心的距离，并归入距离最小的类。

(2)如果训练数据与其他类的中心相隔太远，即训练数据向量的值与所有类中心的距离大于某一个值，则根据这个训练数据生成一个新的类，并确定其为新的中心。

(3)根据得到的聚类确定层与层之间的连接权值。

经过学习训练之后，第二次层与第三层的连接权表示了相应模糊规则的存在。在某输入量的所有连接模糊子集和模糊规则的连接权中，只有一个具有最大值，其它的将被删除。因此，每一个输入只有一个模糊子集的模糊量变量成为模糊规则输入。如果所有模糊子集节点和规则层节点的连接权都很小，则与之相应的所有连接将被删除，意味着这个输入的模糊子集与规则层节点有很少或没有关系。如果所有的输入与第三层规则层的某节点之间的连接均被删除，那么这一规则节点也将被删除，因此它几乎不影响输出，可以认为这条规则不存在。

3.2 使用粒子群优化算法(PSO)优化调整隶属函数参数

粒子群优化算法(Particle Swarm Optimization)以下简称 PSO 算法由 Jim Kennedy 于 1995 年提出的^[18]，它是对鸟类觅食过程的迁徙和聚群的模拟，由简单个体组成的群落以及个体之间的互动行为模拟搜索全局最优解。算法最初的设想是仿真简单的社会系统，研究并解释复杂的社会行为，后来发现粒子群优化算法可以用于实际优化问题尤其是工程优化问题的求解^[19]。

粒子群算法从正式提出到今天，仅有十多年的时间，这期间该算法有了质的发展，但是还有很多可以改进的地方。首先，粒子群算法主要适用于连续空间函数的优化问题。如何将其应用到离散空间优化问题，特别是一类非数值优化问题，将是需要在未来研究的。其次，微粒群算法有待于在收敛性、收敛速度估计等方面的数学证明。第三，微粒群算法与其他类进化算法的比较研究。再次，利用不同问题的特点设计出相应的有效改进算法，这也是把微粒群算法应用到更多领域的行之有效的好方法。

PSO 算法基于以下的基本理念：群体中的每个个体都可以通过对临近个体中的评价和以往经验中收益，进而得到进化。

粒子群优化算法采用速度—位置搜索模型。基于该模型的粒子在解空间进行搜索。每个粒子代表解空间的一个候选解，解的优劣程度由适应度函数决定。粒子的速度决定粒子在搜索空间单位迭代次数的位移。其中，适应函数根据优化目标定义^[20]

3.2.1 粒子群算法(PSO)的基本概念

(1) 粒子

类似于遗传算法中的染色体(Chromosomes)，PSO 中以粒子(Particle)为基本的组成单位，代表解空间的一个候选解。设解向量为 d 维变量，则当算法迭代次数为 t ，第 i 个粒子 $x_i(t)$ 可表示为 $x_i(t)=[x_{i1}(t), x_{i2}(t), \dots, x_{id}(t)]$ 。其中， $x_{ik}(t)$ 表示第 i 个粒子在第 k 维解空间中的位置，即第 i 个候选解中的第 k 个待优化变量。

(2) 种群

粒子种群(Population)由 n 个粒子组成，代表 n 个候选解。经过 t 次迭代产生的种群 $pop(t)=[x_1(t), x_2(t), \dots, x_i(t), \dots, x_n(t)]$ 。其中， $x_i(t)$ 为种群中的第 i 个粒子。

(3) 粒子速度

粒子速度表示粒子在单位迭代次数位置的变化即为代表解变量的粒子在 d 维空间的位移。 $v_i(t)=[v_{i1}(t), v_{i2}(t), \dots, v_{id}(t)]$ ，其中， $v_{ik}(t)$ 为第 i 个粒子在解空间第 k 维的速度。

(4) 适应度函数

适应度函数(fitness function)由优化目标决定，用于评价粒子的搜索性能，指导粒子种群的搜索过程。算法迭代停止时适应度函数最优的解变量即为优化搜索的最优解。

(5) 个体极值

个体极值 $p_i = (p_{i1}, p_{i2}, \dots, p_{id})$ 是单个粒子从搜索初始到当前迭代对应的适应度最优的解。

(6) 全局极值

全局极值 $g = (g_1, g_2, \dots, g_d)$ 是整个粒子种群从搜索开始到当前迭代对应的适应度最优的解^[21]。

3.2.2 粒子群算法(PSO)的算法原理

粒子群优化算法是基于群体智能理论的一种优化算法, 这种算法通过群体中粒子间的合作与竞争产生的群体智能来进行优化搜索。与进化算法比较, 粒子群保留了基于种群的全局搜索策略, 不过和进化算法不同的是 PSO 算法采用了一种相对简单的速度-位移模型, 避免了复杂的遗传算子操作。同时它特有的记忆功能使其可以动态跟踪当前的搜索情况而调整其搜索策略。因此, 粒子群算法是一种高效的并行搜索算法。

粒子群算法开始初始化一群随机粒子(随机解), 然后通过迭代寻找最优解。我们可以把每个优化问题的潜在解看作是 d 维搜索空间上的一个点, 用粒子的位置代表优化问题的解, 粒子由一个速度决定其飞行方向和速率。然后粒子就追随当前的最优粒子在解空间中进行搜寻。在每一次迭代中, 粒子通过跟踪两个极值来更新自己的位置。第一个是粒子本身所找到的最优解, 这个解叫做个体极值(pbest), 另一个极值是整个种群目前找到的最优解, 这个极值是全局极值(gbest)。粒子根据这两个值来更新自己的飞行速度和新的位置, 直到最后找到最优解。

设群体规模为 n , 在一个 d 维的目标搜索空间中, 群体中的第 $i(i=1, 2, \dots, n)$ 个粒子位置可以表示为一个 d 维矢量 $x_i(t)=[x_{i1}(t), x_{i2}(t), \dots, x_{id}(t)]$, 同时用 $v_i(t)=[v_{i1}(t), v_{i2}(t), \dots, v_{id}(t)]$ ($i=1, 2, \dots, n$) 表示第 i 个粒子的飞行速度。用 $p_i = (p_{i1}, p_{i2}, \dots, p_{id})$ 表示第 i 个粒子自身搜索到的最优点。而在这个种群中, 至少有一个粒子是最好的, 将其编号记为 g , 则 $g = (g_1, g_2, \dots, g_d)$ 就是当前种群所搜索到的最好点, 即种群的全局历史最优位置。基本粒子群算法更新迭代计算的公式如下:

$$v_{ij}^{k+1} = v_{ij}^k + c_1 r_{1j} (p_{ij}^k - x_{ij}^k) + c_2 r_{2j} (g_{ij}^k - x_{ij}^k) \quad (3.6)$$

$$x_{ij}^{k+1} = x_{ij}^k + v_{ij}^{k+1} \quad (3.7)$$

其中 $i=1, 2, \dots, n$, j 表示微粒的第 j 维, k 表示迭代次数, c_1, c_2 为加速常数, 一般在 $0 \sim 2$ 之间取值。 c_1 主要是为了调节微粒自身的最好位置飞行的步长, c_2 是为了调节微粒向全局最好位置飞行的步长。 $r_1 \in u(0,1)$, $r_2 \in u(0,1)$ 为两个相互独立的随机函

数。为了减少在进化过程中，微粒离开搜索空间的可能性， v_{ij} 通常限定于一定范围内，即 $v_{ij} \in [-v_{\max}, v_{\max}]$ 。如果问题的搜索空间限定在 $[-x_{\max}, x_{\max}]$ 内，则可设 $v_{\max} = k * x_{\max}$ ， $0.1 \leq k \leq 1.0$ 。

在式 $v_{ij}^{k+1} = v_{ij}^k + c_1 r_{1j} (p_{ij}^k - x_{ij}^k) + c_2 r_{2j} (g_1^k - x_{ij}^k)$ 所描述的速度进化方程中，其第一部分 v_{ij}^{k+1} 为粒子先前的速度；其第二部分 $c_1 r_{1j} (p_{ij}^k - x_{ij}^k)$ 为“认知”部分，它仅考虑了粒子自身的经验，表示粒子本身的思考，如果没有这部分，则粒子就没有了认识能力，也就是只有“社会”的模型，这样，粒子在相互作用下，有能力到达新的搜索空间，虽然它的收敛速度比基本粒子群算法更快，但对于复杂问题，则容易陷入局部最优点；其第三部分 $c_2 r_{2j} (g_1^k - x_{ij}^k)$ 为“社会”部分，表示粒子间的社会信息共享，如果没有这部分，则其性能变差，主要原因是不同的粒子间缺乏信息交流，则没有社会信息共享，粒子间没有交互，使得一个规模为 n 的群体等价于运行了 n 个单个粒子，因而得到最优解的概率非常小。总之，基本粒子群算法的速度进化方程有认知和社会两部分组成^[22, 23, 24]。

3.2.3 粒子群算法(PSO)的算法流程

- (1) 依照初始化过程，对粒子群的随机位置和速度进行初始设定；
- (2) 计算每个粒子的适应值；
- (3) 对于每个粒子，将其适应值与所经历过的最好位置 p_i 的适应值进行比较，若较好，则将其作为当前最好位置；
- (4) 对于每个粒子，将其适应值与全局所经历过的最好位置 g 的适应值进行比较，若较好，则将其作为当前的全局最好位置；
- (5) 根据两个迭代公式对粒子的速度和位置进行进化；
- (6) 如未达到结束条件通常为足够好的适应值或达到一个预设最大代数(G_{\max})，返回步骤(2)；否则执行步骤(7)；
- (7) 输出 $gbest$ ^[25]；

算法流程图 3.5 所示

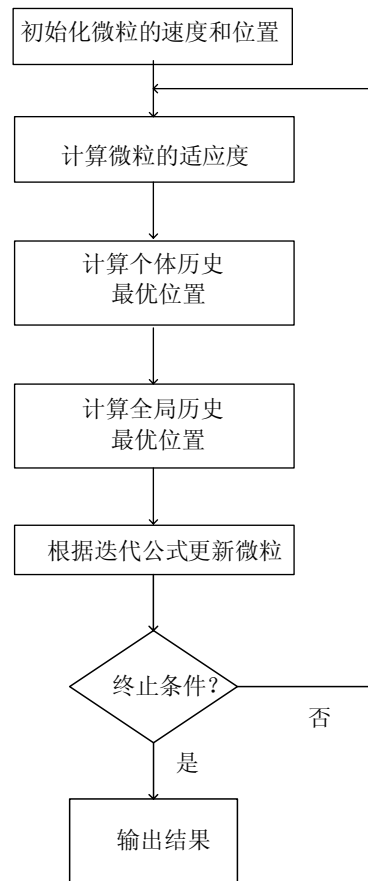


图 3.5 PSO 算法流程图

Fig.3.5 Flow diagram of PSO

3.2.4 带惯性权重的粒子群算法(PSO)

由公式 $v_{ij}^{k+1} = v_{ij}^k + c_1 r_{1j} (p_{ij}^k - x_{ij}^k) + c_2 r_{2j} (g_{ij}^k - x_{ij}^k)$ 可以看出, 公式的右边由三部分组成。第一部分是粒子更新前的速度, 而后两部分反映了粒子速度的更新。粒子群优化算法的全局搜索特性通过随机初始化的速度体现。惯性权重 w (inertia weight) 用于控制前一次迭代产生的粒子速度对本次迭代速度的影响。粒子群优化算法的全局搜索特性通过随机初始化的速度体现。惯性权重 $w \in [0, 1]$ 。Shi 与 Eberhart 通过实验证明, 较大的惯性权重有利与粒子群进行全局搜索, 而较小的惯性权重种群更倾向于局部搜索^[26]。在实际的优化问题求解过程中, 一般来说惯性权重随迭代次数线性递减, 这样就使粒子群在搜索的初始阶段, 能够以较大的概率在整个解空间进行搜索, 并以快速收敛到最优解所在的局部区域, 然后随着惯性权重的递减, 粒子种群在该区域内实现局部微调。

因此, 在公式 $v_{ij}^{k+1} = v_{ij}^k + c_1 r_{1j} (p_{ij}^k - x_{ij}^k) + c_2 r_{2j} (g_{ij}^k - x_{ij}^k)$ 的 v_{ij}^k 前使用惯性权重 w , w 较大算法具有较强的全索能力, w 较小则算法倾向于局部搜索。一般的做法是将 w 初始为 0.9 并使其随迭代次数的增加线性递减至 0.4, 以达到上述期望的优化目的^[27, 28]。基

本粒子群算法是 $w=1$ 的特殊情况。我们把带惯性权重的微粒群算法称之为标准粒子群算法。

引入惯性权重 w 可消除基本粒子群算法对 v_{\max} 的需要。当 v_{\max} 增加时, 可通过减少 w 来达到平衡搜索, 而 w 的减少可使得所需的迭代次数变小。所以, 可以将各维变量的 $v_{\max,d}$ 固定, 而只 w 进行调节。 w 越大, 粒子的飞行速度就越大, 它将以较大的步长进行全局搜索; w 越小, 粒子的速度步长越小, 粒子趋向于进行精细的局部搜索。 w 的变化趋势正好相当于粒子速度的变化趋势。所以带惯性权重的粒子群算法的改进之处就是将二者结合起来以使粒子可以尽快的向最优解区域靠拢, 而又不至于在到达最优解区域附近时飞越最优解。

目前关于粒子群算法的研究, 一般都是将带惯性权重的粒子群算法作为最基本的 PSO 算法模型。

3.2.5 动态惯性权值的粒子群优化算法(DIPSO)

粒子群优化算法最典型的缺陷就是其容易陷入局部最优解, 而且其搜索精度不高。随着人们对产品性能的不断追求, 对其产品的优化方法提出了更高的要求。因此, 各种各样的改进算法也应运而生。

前文已经提到标准粒子群优化算法, 其本质就是在基本粒子群优化算法的速度更新公式中的 v_{ij}^k 前面加上一个惯性权值 w , Shi 与 Eberhart 通过实验证明, 较大的惯性权重有利与粒子群进行全局搜索, 而较小的惯性权重种群更倾向于局部搜索。一般的做法是将 w 初始为 0.9 并使其随迭代次数的增加线性递减至 0.4, 以达到上述期望的优化目的

[29]。

一般做法的具体公式如下:

$$w(t) = 0.9 - \frac{t}{MaxNumber} \times 0.5 \quad (3.8)$$

其中, $MaxNumber$ 为最大迭代次数, 这样惯性权重 w 看成是迭代次数的函数, 可以从 0.9 到 0.4 线性减少。但在实际的问题解决中, 对于不同的问题, 其每一代所需的比例关系并不相同, 所以, 惯性权重 w 的线性递减只是对某些问题很有效。此外, 一旦在进化初期, 搜索不到最优点, 随着 w 的逐渐减小, 算法的局部收敛能力加强从而容易使算法陷入局部最优。并且, 如果在进化初期探测到次好点, 这时, w 的相对取小就可使算法很快搜索到最好点, 而 w 的线性递减减缓了算法的收敛速度。同时, PSO 在实际搜索过程中非线性是高度复杂的, 致使惯性权重 w 线性递减的策略不能反应实际的优化搜索过程^[30]。

由于 PSO 存在着容易陷入局部最优, 跳不出自身的循环, 因而必须对其进行改进。

根据上面提出的线性减少惯性权重所存在的弊端，提出一种改进惯性权值的思想。该思想的核心内容是，随着粒子群不断仿真过程的进行，逐渐达到其最优解，每相邻两次最优解比值的大小，能够说明其进化速度的快慢。迭代次数为 T 时，所有粒子的适应度值之和的平均值同当前微粒群的最优适应值的比值，表现了整个粒子群的运动状态，是分散还是比较集中。以这两个运动模型为依据，来动态的调整它们的运动状态。使其能够达到全局最优值，而不陷入局部最优值^[31]。

(1) PSO 算法惯性权值的改进方法

基于以上分析和想法，对带惯性权值的粒子群优化算法做出如下改进：

(a) 影响算法性能的一个重要因素便是进化速度因子，在优化过程中，全局最优值的值取决于个体最优值的变化，同时也反映了粒子群所有粒子的运动效果。在优化过程中，当前迭代的全局最优值总是要优于或至少等于上一次迭代的全局最优值。具体地说：如果优化目标是寻找极小值：

$$F(gbest_{T-1}) \geq F(gbest_T) \quad (3.9)$$

其中 $F(gbest_{T-1})$ 表示的是迭代过程中微粒群前一次的最优值， $F(gbest_T)$ 表示的是迭代过程中微粒群当前的最优值。

则定义：

$$Pspeed = \frac{1.0}{\exp(pb_{best} - pre_{best}) + 1} \quad (3.10)$$

其中 $Pspeed$ 表示进化速度因子， pb_{best} 表示当前微粒群的最优适应度值， pre_{best} 表示前一次微粒群的全局适应度值。

如果优化目标是寻找极大值

$$F(gbest_{T-1}) \leq F(gbest_T) \quad (3.11)$$

则定义：

$$Pspeed = \frac{1.0}{\exp(pre_{best} - pb_{best}) + 1} \quad (3.12)$$

综合这两种情况，进化速度因子($Pspeed$)可以表示为：

$$Pspeed = \frac{1.0}{\exp(\min((pre_{best} - pb_{best}), (pb_{best} - pre_{best}))) + 1} \quad (3.13)$$

根据上面的定义，可以得出 $0 \leq Pspeed \leq 1$ ，该参数考虑了算法的运行历史，也反映了粒子群的进化速度，即 $Pspeed$ 越小，进化速度越快。当经过了一定的迭代次数后，

P_{speed} 值将会保持 1，则可判定算法停滞或者找到了最优解。

(b) 影响算法性能的另一个因素是粒子群的聚集度因子。在算法中，全局最优值，总是优于所有个体的当前适应度值。具体地说：如果， \bar{F}_T 为所有粒子当前适应度值的平均值，则

$$\bar{F}_T = \frac{1}{N} \sum_{i=1}^N F(X_T[i]) \quad (3.14)$$

其中是粒子 $X_T[i]$ 在 i 在当前迭代次数为 T 时的位置， N 是粒子群的规模数。

分两种情况讨论：

如果优化目标是寻找极小值

$$F(gbest_T) \leq \bar{F}_T \quad (3.15)$$

其中 $F(gbest_T)$ 表示的是迭代过程中微粒群当前迭代次数为 T 时的最优值。则定义

$$ptogether = \frac{1.0}{\exp(pbest * popsize - paccount) + 1.0} \quad (3.16)$$

其中 $Ptogether$ 表示聚集度因子， $pbest$ 表示当前迭代次数为 T 代时微粒群的最优适应度值， $paccount$ 表示迭代次数为 T 代时所有粒子的适应度值之和。

如果优化目标是寻找极大值

$$F(gbest_T) \geq \bar{F}_T \quad (3.17)$$

聚集度因子就可以表示为：

$$ptogether = \frac{1.0}{\exp(paccount - pbest * popsize) + 1.0} \quad (3.18)$$

综合这两种情况，聚集度因子($Ptogether$)可以表示为：

$$ptogether = \frac{1.0}{\exp(\min((paccount - pbest * popsize), (pbest * popsize - paccount))) + 1.0} \quad (3.19)$$

根据上面的定义，可以得出 $0 \leq ptogether \leq 1$ ，该参数考虑了算法的运行历史，也反映了粒子群当前的聚集程度，同时也在一定的程度上也反映出粒子的多样性，即 $ptogether$ 越大，粒子群聚集程度也越大，粒子多样性也越小。当经过了一定的迭代次数后， $ptogether$ 值将会保持 1，粒子群中的所有粒子均具有同一性，如果此时算法陷入局部最优，则结果不容易跳出局部最优^[32]。

(2) 自适应惯性权值 w

在动态惯性权值的粒子群优化算法中, w 的大小应该随着粒子群的进化速度和粒子的逐渐聚集程度而改变, 即 w 可表示为进化速度因子(Pspeed)和聚集度因子(Ptogether)的函数。即

$$w = f(ps\text{peed}, p\text{together})$$

如果粒子群进化速度较快, 算法可以在较大的搜索空间内持续搜索, 粒子就可以保持大范围的寻优。当粒子群进化速度减慢时, 可以减少 w 的值, 使得微粒群在小空间内搜索, 以便更快地找到最优解。若粒子较分散, 粒子群就不容易陷入局部最优解。随着粒子群的聚集程度的提高, 算法容易陷入局部最优, 此时应增大粒子群的搜索空间, 提高粒子群的全局寻优能力。

综上所述, w 应随着粒子的聚集度的增大而增大, 随着进化速度的降低而减少, 因此它可以表示为:

$$w = 1.0 - ps\text{peed} * w_h + p\text{together} * w_s \quad (3.20)$$

从这个式子中可以看出, w 是随着进化速度因子和聚集度因子的变化而变化, 因此是一种动态的自适应惯性权重算法(DIPSO)^[33]。

对于 w_h 和 w_s 的选择, 一般认为 w_h 较大则有利于算法快速收敛, w_s 较大则有利于跳出局部最优, 就实际应用而言, 对于低维问题(如二维优化), w_h 一般选择 0.2, w_s 选择 0.2, 对于高维优化问题, w_s 应该相应增大。

(3) 动态惯性权值的粒子群优化算法流程

- (a)初始化粒子群的位置, 速度, 计算粒子群的适应度值;
- (b)初始化粒子群的个体最优和全局最优;
- (c)如果达到了粒子的收敛条件, 执行步骤(7), 否则执行步骤(4), (5);
- (d)用迭代速度和位置公式对粒子群进行更新, 并计算粒子的适应度值, 更新粒子的全局最优值和个体最优值;
- (e)将迭代次数加 1, 并执行步骤(3);
- (f)输出 gbest, 算法结束。

动态惯性权值的粒子群优化算法流程图 3.6 如下:

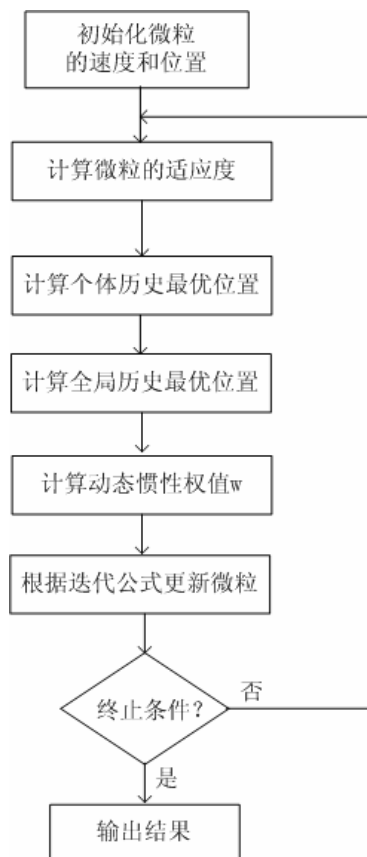


图 3.6 动态惯性权值 PSO 算法流程图

Fig.3.6 Flow diagram of PSO with dynamic inertia weight

(4) 动态惯性权值的粒子群优化算法的仿真

为了验证动态惯性权值的粒子群优化算法对于标准 PSO 算法的改进是否有效，本文选取以下两个函数最优化问题进行验证。

(a) 简单函数： $f(x) = (x_1 - 2)^2 + (x_2 - 2)^2$ 的极小值，这是一个典型的二维优化问题，并且其结果也是显而易见的，即 $x_1=2$ ， $x_2=2$ 的时候 $f(x)$ 有极小值 0。

在仿真的时候本文选取粒子为 (x_1, x_2) ，初始粒子为 $(8, 8)$ ，初始速度为 $(1, 1)$ ，初始局部最优位置为 $(8, 8)$ ，初始全局最优位置为 $(8, 8)$ ，初始局部最优值为 89，初始全局最优值为 89，最大迭代次数为 100 次，由于模型简单，所以对收敛条件暂时不做考虑， w_h 选为 0.2， w_s 选为 0.2。

仿真工具使用 MATLAB，使用自行编写的 PSO 算法进行运算，算法完成后将迭代次数作为横坐标，每次迭代的全局最优值作为纵坐标使用“o”表示在平面坐标系上。

首先考虑不带动态惯性权值的粒子群优化算法，对于以上输入可以得到如图 3.7 结果，本图为一仿真结果，从仿真结果图可以看到标准 PSO 算法在第 20 代的时候收敛完成迭代。

图 3.8 为在标准 PSO 算法的基础上增加动态惯性权值的粒子群优化算法的 MATLAB 仿真图，从图中可以看出使用了动态惯性权值的粒子群优化算法后，算法在

第 12 代就收敛完成了迭代，并且在迭代过程中的第六步开始对比标准 PSO 算法明显可以看到动态惯性权值的作用，即在粒子速度下降的时候减小粒子速度，同时在粒子趋向于聚集的时候增大了粒子的搜索空间防止其陷入局部最优。

在针对上文简单函数的多次仿真结果中，使用标准 PSO 算法出现过陷入局部最优，算法失效不收敛的情况，而使用带动态惯性权值的粒子群优化算法则未出现上述问题，这说明带动态惯性权值的粒子群优化算法对于提升算法速度和防止算法陷入局部最优是有效的。

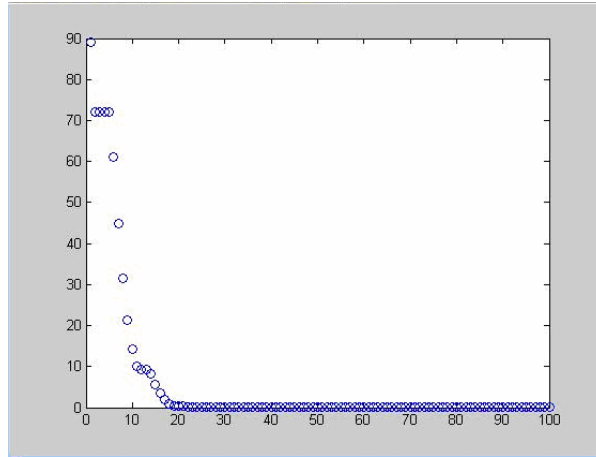


图 3.7 标准 PSO 算法仿真结果

Fig.3.7 Result of standard PSO

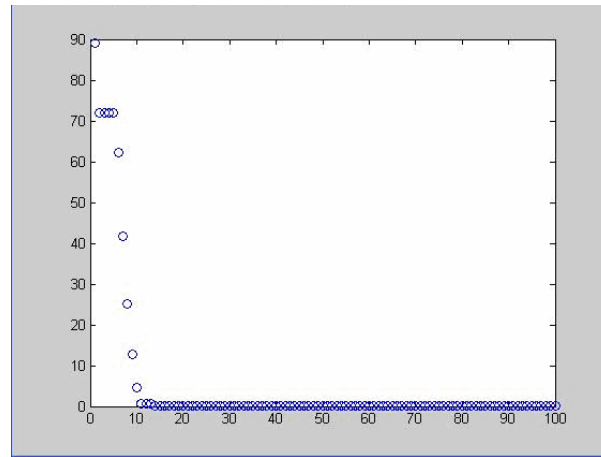


图 3.8 动态惯性权值 PSO 算法仿真结果

Fig.3.8 Result of PSO with dynamic inertia weight

(b) Freudenstein-Roth 函数，求

$$f(x) = [-13 + x_1 + ((5 - x_2)x_2 - 2)x_2]^2 + [-29 + x_1 + ((x_2 + 1)x_2 - 14)x_2]^2 \quad (3.21)$$

的极小值。这也是一个典型的二维优化问题，问题的答案也是显而易见的，极小值为 0。相对于前文简单的函数，这个函数将自变量的最高次数提高到 3，从而使优化问题复杂化，由于函数较为复杂，不通过计算我们无法大体判断其解的位置，所以我们随机选取粒子的初始位置(3, 3)，初始速度为(0.5, 0.5)，初始局部最优位置为(2.5, 2.5)，

初始全局最优位置为(2.5, 2.5)，初始局部最优值为 100，初始全局最优值为 100，最大迭代次数为 100 次，收敛条件就是最大迭代次数， w_h 选为 0.2， w_s 选为 0.2。

仿真工具使用 MATLAB，使用自行编写的 PSO 算法进行运算，算法完成后将迭代次数作为横坐标，每次迭代的全局最优值作为纵坐标使用“o”表示在平面坐标系上。

图 3.9 为使用标准 PSO 算法对 Freudenstein-Roth 函数进行处理的结果，本次结果在第 55 次迭代开始收敛。

图 3.10 为使用动态惯性权值的粒子群优化算法对 Freudenstein-Roth 函数进行处理的结果，本次结果在第 28 次迭代后开始收敛。可以看出动态惯性权值的粒子群优化算法的收敛速度明显比标准 PSO 算法收敛速度快。在使用两种算法对 Freudenstein-Roth 函数进行处理的过程中，两种算法均出现多次不能收敛的情况，但总体上，动态惯性权值的粒子群优化算法更容易收敛。

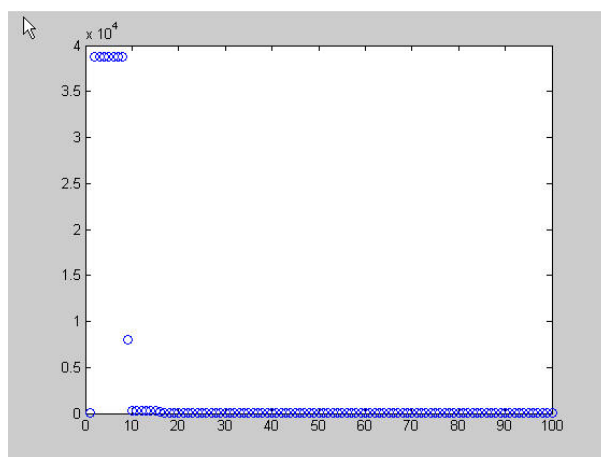


图 3.9 标准 PSO 算法结果

Fig.3.9 Result of standard PSO

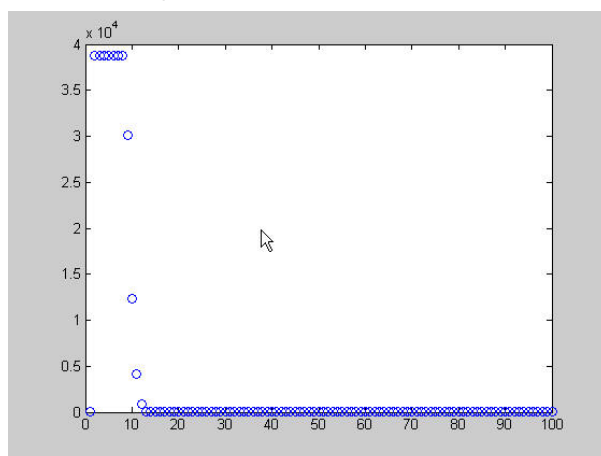


图 3.10 动态惯性权值 PSO 算法结果

Fig.3.10 Result of PSO with dynamic inertia weight

3.2.6 动态惯性权值 PSO 算法优化模糊神经网络参数

前文介绍了动态惯性权值的粒子群优化算法，这种方法可以在保存 PSO 标准算法全局搜索和模型简单特点的基础上很大程度的避免了 PSO 算法容易陷入局部最优解的缺陷。本文接下来的将使用这种方法对已经搭建起来的模糊神经网络隶属函数的中心和宽度进行优化和调整。

假设我们选择的输入量数目为 m ，按照专家主观经验确定的每个输入量的隶属函数的模糊子集为 n ，则需要优化调整的隶属函数数目为 $m*n$ ，隶属函数需要优化调整的参数是隶属函数的中心，于是需要优化调整的参数总数为 $m*n$ 。按照上文的分析，PSO 算法中的粒子的维数就是 $m*n$ ，粒子的具体表示方式为：

$$x_i(t) = [m_{11}, m_{12}, \dots, m_{ij}] \quad 1 \leq i \leq m, 1 \leq j \leq n \quad (3.22)$$

其中 $x_i(t)$ 为第 i 个粒子在第 t 次迭代的坐标， m_{ij} 为第 i 个输入的隶属函数的第 j 个高斯型函数的中心，其中 $1 \leq i \leq m, 1 \leq j \leq n$ 。

粒子的维数和形式确定后，就需要根据具体应用确定 PSO 算法的适应度函数。具体到本文使用的 PSO 算法，其适应度函数选取为输出端均方误差：

$$e(t) = \frac{1}{S} \sum_{i=1}^s \sum_{j=1}^n (\bar{y}_{ij} - y_{ij})^2 \quad 1 \leq i \leq s, 1 \leq j \leq n \quad (3.23)$$

其中， $e(t)$ 输出端均方误差， S 为训练样本总数， n 为输出模糊量个数， \bar{y}_{ij} 为第 i 个样本的第 j 个输出量的理想值， y_{ij} 为第 i 个样本的第 j 个输出量的实际值。

在本文采用的模糊神经网络系统中，训练样本是以模糊量的形式给出的，输入为单值的精确量，输出为向量形式的模糊量，具体实例会在后面的电梯群控系统模式识别和调度中详细的给出。

假设一个模糊神经网络有两个输入 a 和 b ，每个输入有三个模糊子集，分别表示为 $[L, M, B]$ ，有一个输出 c ，输出有三个模糊子集，分别表示为 $[L, M, B]$ ，并有如下的简单模糊规则：

If a is B , b is B then c is L

If a is B , b is M then c is M

If a is B , b is L then c is B

If a is M , b is L then c is L

If a is M , b is M then c is B

If a is M , b is B then c is B

动态惯性权值 PSO 算法优化目标为模糊神经网络隶属函数的中心，粒子初始化为 $[0.2, 0.5, 0.8, 0.2, 0.5, 0.8]$ ，速度初始化为 $[0.01, 0.01, 0.01, 0.01, 0.01, 0.01]$ ，初始化局部最优位置为 $[0.25, 0.55, 0.85, 0.25, 0.55, 0.85]$ ，初始化全局最优位置为 $[0.25,$

0.55, 0.85, 0.25, 0.55, 0.85], 选用输出端均方误差作为适应度函数, 最大迭代次数 100 次。

在动态惯性权值 PSO 算法中使用 20 组样本训练数据对模糊神经网络隶属函数的中心进行优化, 结果如图 3.11 所示, 其中纵坐标表示全局最优点的适应度值, 横坐标为迭代次数, 最终优化结果为[0.2609, 0.5859, 0.7202, 0.1000, 0.4622, 0.8148]

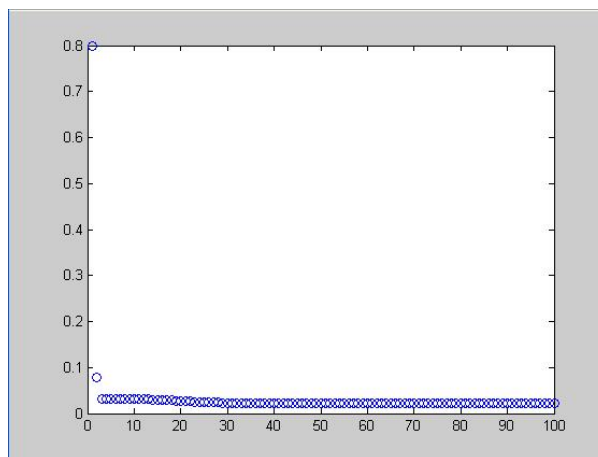


图 3.11 动态惯性权值 PSO 算法对模糊神经网络隶属函数的优化

Fig.3.11 The optimization result of PSO with dynamic inertia weight

3.3 模糊神经网络的仿真

神经网络作为当前的研究热点, 具有并行计算, 自适应性, 不需要准确数学模型等特点, 将模糊逻辑与神经网络进行有机融合的模糊神经网络在继承了神经网络特点的同时又具有了模糊逻辑的特性, 可以处理模糊语言量进行模糊推理。

在实验室中, 使用计算机仿真技术可以在一定程度上实现对模糊神经网络的仿真。在神经网络的诸多特性中, 计算机无法仿真的是神经网络的并行计算, 在实际中, 并行计算往往是通过 CPU 的堆砌或 CPU 的多线程来实现的, 就本文而言, 由于网络规模不大, 对速度要求不高, 所以可以忽略这个影响。

MATLAB 是 1984 年由美国 MathWorks 公司的产品。早在 20 世纪八十年代中期, MATLAB 就曾在我国出现, 但真正大规模流行是九十年代中期以后的事。现在, MATLAB 已被从事科学研究、工程计算的广大科技工作者、高校师生确认为必须掌握的计算工具, 从理论通向实际的桥梁, 和最可信赖的科技资源之一。

MATLAB 相对于其他高级编程语言最大的优势就是对矩阵的处理能力, 并且 MATLAB 还集成了大量的当前先进的数值分析方法和实用的工具箱, 另外 MATLAB 可以任意的定义其他高级编程语言难以获得的动态数组, 这对于编程者来说也是非常宝贵的特性。但是 MATLAB 也有其自身缺点, 那就是缺乏结构性, 很难形成像其他高级编程语言那样的结构化软件包, 并且 MATLAB 本质上是一种脚本语言, 只适应于科研仿

真。

为了验证本文构建的模糊神经网络的有效性和可靠性，下面使用一个简单的 MISO 模型来对模糊神经网络进行验证。

假设一个系统有两个输入 $[x_1, x_2]$ 和一个输出 y 。每个输入和输出都有三个模糊子集分别记为 $[L, M, B]$ 。

每个输入和输出的隶属函数中心宽度矩阵假设为：

$$w = \begin{bmatrix} 0.2 & 0.2 \\ 0.5 & 0.2 \\ 0.8 & 0.2 \end{bmatrix} \quad (3.24)$$

假设具体的模糊规则有以下四条

If x_1 is L x_2 is L then y is L

If x_1 is L x_2 is M then y is L

If x_1 is M x_2 is L then y is M

If x_1 is B x_2 is B then y is B

则作为模糊神经网络模糊规则输入 $w1$ 的矩阵为

$$w1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.25)$$

作为模糊神经网络模糊推理输入 $w2$ 的矩阵为

$$w2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.26)$$

模糊神经网络应用于电梯群控系统交通模式识别

现在给定一个特定的输入 $x=[0.18, 0.2]$ ，将 $n=2$ ， $m=3$ ， $s=3$ ， $t=4$ ， w ， $w1$ ， $w2$ 带入编制好的模糊神经网络仿真程序 FNN 中，可以得到输出

$$y=[0.3073, 0.1845, 0]$$

在 y 的所有模糊子集中取最大值作为结果，结果符合先前定义的模糊规则。

再次给定一个特定输入 $[0.45, 0.12]$ ，将 $n=2$ ， $m=3$ ， $s=3$ ， $t=4$ ， w ， $w1$ ， $w2$ 带入前文的模糊神经网络仿真程序 FNN 中，可以得到输出

$$y=[0.0235, 0.3016, 0]$$

在 y 的所有模糊子集中取最大值作为结果，结果符合先前定义的模糊规则。

再次给定一个特定输入 $[0.75, 0.82]$ ，将 $n=2$ ， $m=3$ ， $s=3$ ， $t=4$ ， w ， w_1 ， w_2 带入前文的模糊神经网络仿真程序 FNN 中，可以得到输出

$$y=[0.0663, 0.0537, 0.3619]$$

在 y 的所有模糊子集中取最大值作为结果，结果符合先前定义的模糊规则。

经过多次试验可以证明本文所采用的模糊神经网络的模型在模糊推理上的正确性。

3.4 本章小结

本章详细叙述了如何使用混合训练方法对模糊神经网络进行学习训练，即：通过 k -均值方法初步获取隶属函数中心和宽度、通过顺序聚类算法获取模糊规则，通过动态惯性权值的 PSO 算法优化隶属函数的中心和宽度。本章重点介绍了一种基于速度-位移模型的粒子群优化算法(PSO)，然后在此基础上引入一种带有动态惯性权值的 PSO 算法，并针对标准 PSO 算法和动态惯性权值的 PSO 算法进行了仿真比较。最后本章针对完成学习训练的模糊神经网络进行了仿真以验证有效性和可靠性。

第4章 电梯群控交通流模式识别

4.1 电梯群控系统交通模式识别概述

模式识别是从模式空间到类别隶属空间的正确映射。模式识别的方法有传统模式识别和智能模式识别。传统的模式识别包括统计模式识别和结构模式识别。智能模式识别包括模糊推理、神经网络和遗传算法等。在智能模式识别方法中,模糊模式识别可以反映模式空间整体的、主要的特征。但是,由于准确的隶属函数难以建立,并且模糊规则方法无自学习功能,所以限制了它的应用。神经网络模式识别是一个非线性动态系统,通过对样本的学习建立记忆,然后用将来模式判断为最接近的记忆,它可以处理一些环境信息十分复杂,背景知识不清楚,推理规则不明确的问题。神经网络模式识别系统具有学习功能,而且对输入信息的不完备或特征的缺损不太敏感,但是用神经网络方法制定样本困难,而且网络训练非常耗时。

在电梯控制中,控制电梯运行的方法必须根据电梯状况、交通流量和安装环境的变化条件而变化,与安装环境相适应的电梯条件等是可观测的或是已知的,然而交通流量是难以观测的,掌握它是个难题。交通流量是表明电梯状态的一个概念,是影响电梯群控算法性能的一个重要因素,对其进行研究非常必要,电梯群控系统的普遍发展趋势是能识别交通流量变化。交通流是由电梯服务系统的乘客数、乘客出现的周期以及乘客的分布情况来描述的状态量,它反映电梯群控系统所处的交通状况。对于一个电梯群控系统,交通流可以用许多种数据来表示,而交通分析中用到的仅仅是一部分能够反映建筑物内交通特征的数据,它们分别是单位时间间隔内进入门厅人数,离开门厅人数,建筑物内总的客流量,层间客流最大层的比例。交通流的出现情况主要与建筑的用途有关。对于典型办公大楼,根据其客流量的分布特点可以把交通情况分成几种模式:上行高峰交通模式,下行高峰交通模式,层间交通模式,空闲交通模式,两路交通模式^[34, 35]。

(1)上行高峰交通模式(UP)。这个交通模式的特征是客流强度很大,当主要的(或全部的)客流是上行方向时,即全部或者大多数乘客在建筑物的门厅进入电梯且上行,这种状况被定义为上行高峰交通模式。上行高峰以相当大的到达率将发生在早晨上班时刻(主要针对办公大楼),此时乘客进入电梯上行到大楼的上层上班;其次,强度稍小的上行高峰发生在午间休息结束时刻。

(2)下行高峰交通模式(DP)。这个交通模式的特征是客流强度很大,当主要的(或全部的)客流是下行方向时,即全部或者大多数乘客乘电梯下行到门厅离开电梯,这种状况被定义为下行高峰交通模式。下行高峰主要发生在下班时刻,而在午间休息开始时形成强度较弱下行高峰。

(3) 层间交通模式(BT)。这个交通模式的特征是客流强度为中等或较繁忙阶段，当上行和下行乘客数量大致相同时，并且各层之间的交通需求基本平衡，这种交通模式是一种通常的交通状况，存在于一天的大部分时间。

(4) 空闲交通模式(IT)。这个交通模式的特征是客流量极少，且是间歇性的(如假日和深夜)。

(5) 两路交通模式(DT)。当主要的客流是朝着某一层或从某一层而来，而该层不是门厅，则此时存在的是两路交通模式。

4.2 应用在交通流模式中的模糊神经网络的结构

4.2.1 模糊神经网络的输入及其模糊集合

根据对各种交通模式的分析，确定交通模式识别的特征值：本时间段的总客流量，进入门厅的客流量，离开门厅的客流量；最大楼层客流量，次大楼层的客流量。此时，门厅特指一楼大厅。这些特征值反映了某一时间段的交通特征，应用这些特征值对交通模式进行识别。

因此应用在电梯群控系统交通模式识别的模糊神经网络的输入应该是：

单位时间内总的客流量 x_1

单位时间内上行的客流量 x_2

单位时间内下行的客流量 x_3

单位时间内客流量最大楼层的客流量 x_4

其中一般选取 5 分钟作为单位时间。

这几个客流量特征值相对容易搜集并且基本上可以清楚的表示出电梯群控系统当前的客流量状态。

作为模糊网络的输入，客流量特征值需要进一步的处理，这里本文使用归一化对客流量特征值进行处理，以方便在训练和优化隶属函数。

归一化处理如下：

$$\text{单位时间内总的客流量 } x_1' = \frac{x_1}{x_{\max}}$$

$$\text{单位时间内上行的客流量 } x_2' = \frac{x_2}{x_{\max}}$$

$$\text{单位时间内下行的客流量 } x_3' = \frac{x_3}{x_{\max}}$$

$$\text{单位时间内客流量最大楼层的客流量 } x_4' = \frac{x_4}{x_{\max}}$$

其中 x_{\max} 是单位时间内总客流量的最大值^[35]。

每一个输入的客流量对应一个模糊集合，每个模糊集合都有数个模糊子集构成，为了简化问题，本文规定所有的模糊集合都有同样数量的模糊子集。

对于上面的输入，每个分量 x_i 均为模糊语言变量，并设其中某个 x_i 的模糊集合为：

$$T(x_i) = \{A_i^1, A_i^2 \dots A_i^{m_i}\}, i = 1, 2, \dots, n$$

其中， A_i^j ($j=1, 2, \dots, m_i$) 是 x_i 的第 j 个语言变量值，是定义在论域 U_i 上的一个模糊集合，其相应的隶属度为 $\mu_{A_i^j}(x_i)$ ($i=1, 2, \dots, n; j=1, 2, \dots, m_i$)

在这里我们具体的假设每个客流量特征值的模糊集合有 5 个模糊子集，即 $m_i = 5$ 。使用模糊语言可以表述为{小，较小，中，较大，大}，为了方便表述我们使用 {L, ML, M, MB, B} 来表示。

本文后面的模式识别过程和派梯调度过程中使用模糊规则也使用上述的表示方法表达。

精确的客流量数据要完成模糊化过程需要通过隶属函数的运算，本文采用的隶属函数形式都是高斯型隶属函数的相互叠加。

但是具体到本文的应用对象和前面规定的模糊集合的自己数目，本文采用如图 4.1 所示的隶属函数模型：

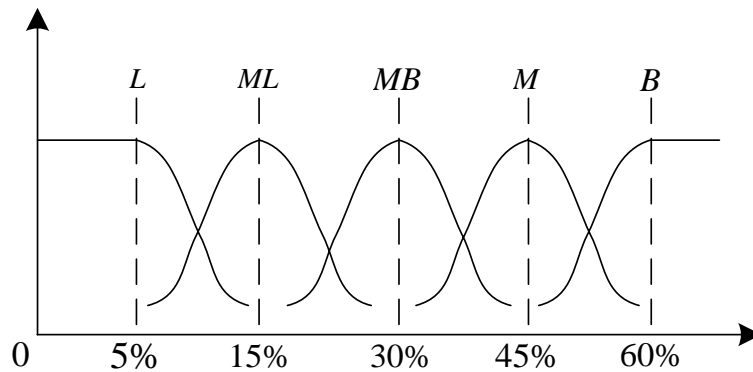


图 4.1 隶属函数图形

Fig.4.1 Shape of membership function

上图中，L 和 B 对应的高斯函数只有一半，其意义是当人数少到一定程度的时候直接判断为小，当人数多到一定程度的时候直接判断为大。每个高斯型隶属函数都有自己的中心，图中的 5%，15%，30%，45%，60% 就是高斯型隶属函数的中心，图中标明的中心只是举例说明，具体在实际中，不同的输入的模糊子集对应不同的中心，需要通

通过对模糊神经网络的训练和优化获得。通过隶属函数得到的隶属度是一个大于 0 小于 1 的实数。

4.2.2 模糊神经网络的输出

对应于上文提到的五种交通模式：上行高峰交通模式(UP)、下行高峰交通模式(DP)、层间交通模式(BT)、空闲交通模式(IT)、两路交通模式(DT)，网络的输出 $y = [y_{UP}, y_{DP}, y_{BT}, y_{DT}, y_{IT}]$ 。

其模糊集合为 $T(y) = \{B^1, B^2 \dots B^{m_y}\}$ 。

其中， $B^j (j=1, 2, \dots, m_y)$ 是 y 的第 j 个语言变量值。相应的隶属度为 $\mu_{B^j}(y)$ 。

在这里我们具体的假设输出量的模糊集合有 5 个模糊子集，即 $m_y = 5$ 。使用模糊语言可以表述为{小，较小，中，较大，大}，为了方便表述我们使用{L, ML, M, MB, B}来表示。

4.2.3 模式识别中模糊规则的表述

本文使用的模糊神经网络的规则表达主要靠前文的网络结构中的第二、三、四、五层之间的网络权值的连接来实现的。通过这几层之间权值的连接形成的拓扑结构可以形成类似于模糊逻辑中的模糊规则。

比如：

If x_1' is B and x_2' is B and x_3' is L and x_4' is M

Then y is UP

以上的规则是一般模糊逻辑中的表示方法，在实际推理和本文的模糊神经网络中使用模糊量的隶属度来表示模糊规则，比如

If $x_1' = [0, 0, 0, 0, 1]$ and $x_2' = [0, 0, 0, 0, 1]$ and $x_3' = [1, 0, 0, 0, 0]$ and $x_4' = [0, 0, 1, 0, 0]$

Then $y = [1, 0, 0, 0, 0]$

上面的使用隶属度表示的规则只是一个例子，真正的应用中因为本文采用的隶属函数是高斯型隶属函数，同时本文的模糊神经网络使用的模糊规则是通过使用数据训练来抽取模糊规则的，所以一个输入的子集得隶属度不会如上式中那么规则。

4.2.4 模式识别中的模糊推理

本文的电梯群控系统所使用的模糊神经网络采用前文提出的简化模糊推理方法，即对于给定的输入，只有在输入点附近的那些语言变量值才有较大的隶属度值，原理输入点的语言变量值的隶属度很小或者为 0，当隶属度函数很小时近似的取为 0。因此只有当输入 x 的模糊语言变量与第 i 条规则中对应的语言变量的模糊子集接近的时候，输

入 x 对该规则适应度 α_i 才不为零。这样就完成了模糊前件和模糊规则的匹配。

输入通过模糊神经网络的推理会得到一个结果，结果的表示形式：

$$y = [y_{UP}, y_{DP}, y_{BT}, y_{DT}, y_{IT}] \quad (4.1)$$

其中的模糊子集的隶属度表示与某模式的匹配度。

4.3 模糊神经网络的模式识别的训练和仿真

4.3.1 模糊神经网络的训练

实际中的建筑物和电梯群有着这样那样的限制和客观条件，本文为了讨论问题方便，采用一种简化的模型，并且忽略某些不重要的细节。

本文所采用的模糊神经网络的训练数据是根据实际情况有目的的选择一天中各个典型时间段的典型乘客流数据。选取训练数据样本的原则是随机的选取，同时兼顾一天中的典型时间段。

通过使用根据前文提到的 k -均值聚类自己编写的 MATLAB 程序可以得到输入模糊神经网络仿真程序的高斯型隶属函数中心和宽度矩阵。

用于初步形成模糊神经网络高斯型隶属函数的单个样本形式为：

$$x_i' = [x_1', x_2', x_3', x_4', y] \quad (4.2)$$

样本总容量定为 50 个。训练数据均来自于对某大楼的某几天不同时间段的采样数据作为训练样本。使用的数据只取数据格式的输入部分。

各个模糊里函数的初始中心均选定为 $[0.1, 0.3, 0.5, 0.7, 0.9]$ 。

通过对训练数据的处理就可以得到交通流模式识别模糊神经网络的中心和宽度矩阵的输入 w 。然后就可以使用这个初步确定的中心和宽度矩阵来对模糊神经网络做进一步的训练。

用于形成模糊神经网络模糊规则的单个样本形式为：

$$x_i' = [x_1', x_2', x_3', x_4', y] \quad (4.3)$$

样本总容量定为 50 个。训练数据均来自于对某大楼的某几天不同时间段的采样数据作为训练样本。使用的数据和上文用于 k -均值算法的数据是一样的。

通过使用根据前文提到的聚类方法自己编写的 MATLAB 程序可以得到输入模糊神经网络仿真程序的模糊规则。

在分析采集数据的时候，由于不能得知该大楼电梯的控制策略，所以作为结果的交通模式是由笔者根据有关资料给定的。为了简化模型，笔者对采集的数据进行了一定的处理，排除了一些在聚类运算中元素较少的类，同时对一些相似的类进行的合并和简化。

通过对顺序聚类法得到的结果进行合并简化得到如下的交通模式识别模糊规则：

If x_1' is B x_2' is B x_3' is L x_4' is B then $y=[1, 0, 0, 0, 0]$

If x_1' is B x_2' is MB x_3' is L x_4' is MB then $y=[1, 0, 0, 0, 0]$

If x_1' is B x_2' is M x_3' is L x_4' is M then $y=[1, 0, 0, 0, 0]$

If x_1' is MB x_2' is MB x_3' is L x_4' is MB then $y=[1, 0, 0, 0, 0]$

If x_1' is MB x_2' is M x_3' is L x_4' is M then $y=[1, 0, 0, 0, 0]$

If x_1' is B x_2' is L x_3' is B x_4' is B then $y=[0, 1, 0, 0, 0]$

If x_1' is B x_2' is L x_3' is MB x_4' is MB then $y=[0, 1, 0, 0, 0]$

If x_1' is B x_2' is L x_3' is M x_4' is M then $y=[0, 1, 0, 0, 0]$

If x_1' is MB x_2' is L x_3' is MB x_4' is MB then $y=[0, 1, 0, 0, 0]$

If x_1' is MB x_2' is M x_3' is M x_4' is M then $y=[0, 0, 0, 0, 1]$

If x_1' is MB x_2' is M x_3' is M x_4' is M then $y=[0, 0, 0, 0, 1]$

If x_1' is M x_2' is ML x_3' is ML x_4' is ML then $y=[0, 0, 0, 0, 1]$

If x_1' is M x_2' is L x_3' is L x_4' is L then $y=[0, 0, 1, 0, 0]$

If x_1' is ML x_2' is L x_3' is L x_4' is B then $y=[0, 0, 1, 0, 0]$

If x_1' is L x_2' is L x_3' is L x_4' is B then $y=[0, 0, 0, 1, 0]$

模糊关系合成推理方阵为：

$$w2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad (4.4)$$

这样就可以的到交通流模式识别模糊神经网络的第二层与第三层权值矩阵 $w1$ 和第四层与第五层权值矩阵 $w2$ 。

用于训练模糊神经网络的高斯型隶属函数的单个样本形式为：

$$x_i = [x_1', x_2', x_3', x_4', y] \quad (4.4)$$

样本总容量为 50 个。训练数据均来自于对某大楼的某几天不同时段采样数据作为训练样本。是用的数据和上文用于获得模糊规则的数据是一样的。

经过动态惯性权值的粒子群优化算法优化后，就可以得到完整的交通流模式识别模糊神经网络模型。图 4.2 反映的是动态惯性权值 PSO 算法对模糊神经网络隶属函数中心的优化，其中横坐标为迭代次数，纵坐标为全局最优适应度值。

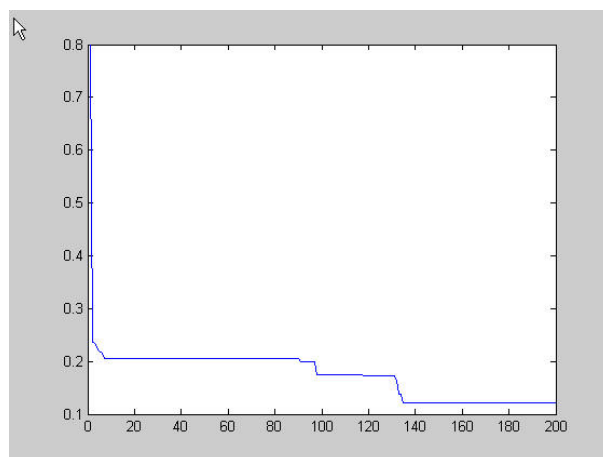


图 4.2 动态惯性权值 PSO 算法对模糊神经网络隶属函数的优化

Fig.4.2 The optimization result of PSO with dynamic inertia weight

对于建筑物某时刻下的某输入 x ，将上面得到的 w ， w_1 ， w_2 以及输入个数 $n=4$ ，输出个数 $m=5$ ，模糊规则个数 $t=14$ ，每个输入的模糊子集个数 $s=5$ 带入前文的模糊神经网络 MATLAB 仿真程序就可以得到每种交通模式的可能性，最后取可能性为 1 的交通模式作为当前交通流特征的交通流模式识别结果。

4.3.2 电梯群控交通流模式识别仿真实例

本文选择了一组交通流特征输入

$$x_1' = 0.8$$

$$x_2' = 0.7$$

$$x_3' = 0.1$$

$$x_4' = 0.7$$

得到的识别结果是 $y = [0.9666, 0, 0, 0, 0]$ ，即上行交通模式。

选取另外一组交通流特征输入

$$x_1' = 0.9$$

$$x_2' = 0.46$$

$$x_3' = 0.42$$

$$x_4' = 0.43$$

得到的识别结果是 $y = [0, 0, 0, 0, 0.9656]$ ，即两路交通模式。

这些结果都是符合常识和逻辑的，可见本文构建的交通流模式识别系统的正确性和可靠性

4.4 本章小结

本章的主要内容是电梯群控系统的模式识别。通过前面章节的介绍和论述，完整的模糊神经网络模型已经建立起来，本章根据前文构建的模糊神经网络模型结构和训练学习方法，针对电梯群控系统模式识别模块的要求和特点，将模糊神经网络模型应用于电梯群控系统的模式识别模块，并详细的叙述了应用于模式识别的模糊神经网络的结构和参数，最后针对模式识别具体应用进行了相应的仿真。

第5章 电梯群控系统派梯调度

5.1 电梯群控系统控制器

在完成了交通流模式识别以后，就可以得到当前的乘客流的交通模式，从而为下一步的电梯群控系统派梯调度提供依据。

一般的电梯群控系统结构图如下图 5.1 所示：

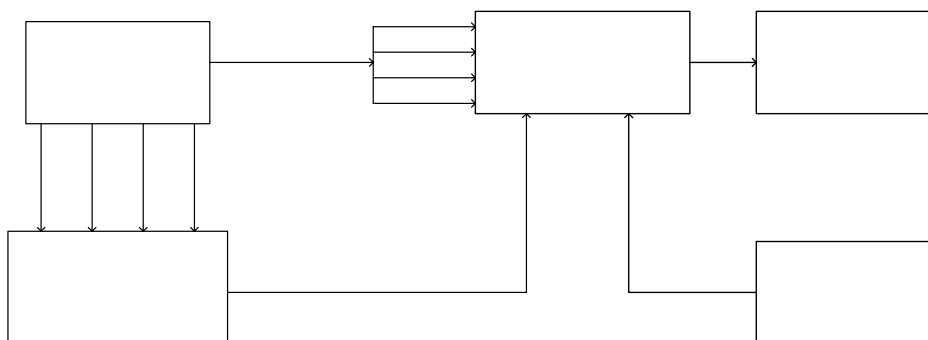


图 5.1 电梯群控系统结构图

Fig.5.1 Structrue of EGCS

上图中通过乘客流模式识别模块可以得到当前的交通模式，确定交通模式后就确定了电梯群控系统的派梯调度的策略。电梯系统派梯调度模块根据派梯调度的策略对每一个呼梯信号进行调度，指派最合适的电梯去响应该呼梯信号。

5.2 电梯群控派梯的调度方法

通过乘客流模式识别模块可以得到分析、处理模块确定交通模式后就确定了电梯群控系统的派梯调度的策略。电梯系统派梯调度模块根据派梯调度的策略对每一个呼梯信号进行调度，指派最合适的电梯去响应该呼梯信号。在电梯控制系统中，厅层呼叫分配部分要考虑厅层呼叫的楼层和方向，各个电梯的状态，包括电梯的位置、轿厢内的人数、已分配的厅层呼叫和未来的厅层呼叫等，这些参数影响各台电梯在评价指标下的可信度。当产生厅层呼叫后，就计算各台电梯的指标 AWT、LWP 和 RNC^[37]。

5.2.1 电梯群控系统的优化目标

在电梯群控系统中，考虑最多的因素就是候梯时间，在本文中，综合考虑乘客平均候梯时间 AWT、乘客长时间候梯率 LWP 和能量损耗 RNC 这几个因素，并给出这三个性能指标的目标函数。

$$S_i = \omega_1 S_{AWT_i} + \omega_2 S_{LWP_i} + \omega_3 S_{RNC_i} \quad \omega_1 + \omega_2 + \omega_3 = 1 \quad (5.1)$$

其中， S_i 为指派第 i 台电梯去响应当前呼梯信号的可信度。 ω_j ($j = 1, 2, 3$) 为权重，

通过上一章中的交通模式识别结果来获得。当前客流处于上行高峰或下行高峰时，这时候梯人数多，所以此时主要考虑的因素是乘客的候梯时间最短而节约能源作为次要因素，即平均候梯时间和长候梯率的权重大，能耗的权重小；当客流处于空闲状态时，这时候梯人数少，所以此时主要考虑的因素是节约能源而乘客的候梯时间作为次要因素，即能耗的权重大，平均候梯时间和长候梯率的权重小。 S_{AWT_i} 是指派第 i 台电梯去响应信号的平均候梯时间(AWT)可信度， S_{LWP_i} 是指派第 i 台电梯去响应信号的乘客长时间候梯率(LWP)可信度， S_{RNC_i} 指派第 i 台电梯去响应信号的能量损耗(RNC)可信度。

通过上式计算可以获得响应呼梯信号所指派的 n 台电梯的可信度，然后在所有可信度中选择最大的的电梯去响应呼梯信号。

$$S_e = \max(S_1, S_2 \dots S_n) \quad (5.2)$$

其中 e 为响应呼梯信号的电梯号^[38]。

5.2.2 电梯群控系统调度策略

前面的乘客流模式识别模块得到交通模式对应了不同的调度策略。对于公式 5.1 的目标优化有如下调度策略：

当前客流处于上行高峰或下行高峰时，这时候梯人数多，所以此时主要考虑的因素是乘客的候梯时间最短而节约能源作为次要因素，即平均候梯时间和长候梯率的权重 ω_1 和 ω_2 大，能耗的权重 ω_3 小。对应于这种模式，本文暂时定为 $\omega = [0.45, 0.45, 0.1]$ 。

当客流处于空闲状态时，这时候梯人数少，所以此时主要考虑的因素是节约能源而乘客的候梯时间作为次要因素，即能耗的权重 ω_3 大，平均候梯时间和长候梯率的权重 ω_1 和 ω_2 小。对应于这种模式，本文暂时定为 $\omega = [0.3, 0.3, 0.4]$ 。

当客流量处于层间交通模式时，此时建筑物各层之间的交通量比较平均，这种情况下对平均候梯时间、长候梯率和能耗的要求比较均衡，所以三个权重 ω_1 、 ω_2 、 ω_3 应当接近。对应于这种模式，本文暂时定为 $\omega = [0.35, 0.35, 0.3]$ 。

当客流量处于两路交通模式时，说明在建筑物的某两层之间发生了大量的乘客流，但是其流量一般不会超过上行或下行交通模式，这种情况下一般要求乘客的候梯时间要短，同时一定程度上兼顾能耗指标，所以平均候梯时间和长候梯率的权重 ω_1 和 ω_2 较大，能耗的权重 ω_3 较小。对应于这种模式，本文暂定为 $\omega = [0.4, 0.4, 0.2]$ 。

以上的调度策略不是一成不变的，需要根据实际情况进行修正，修正的时候可以根据专家经验进行修正，也可以通过智能算法基于统计数据进行修正^[39, 40]。

5.2.3 电梯群控系统派梯调度算法流程

- 电梯群控系统派梯调度具体流程如下：
- (1) 更新呼梯信号序列，通过交通流模式识别模块获取当前交通流模式并确定调度策略。
 - (2) 通过交通流数据采集模块获取当前电梯运行状态数据。
 - (3) 对呼梯信号序列中的每一个呼梯信号计算其每台电梯响应的可信度。
 - (4) 当电梯发生客流量或呼梯信号变动则跳转到(1)。

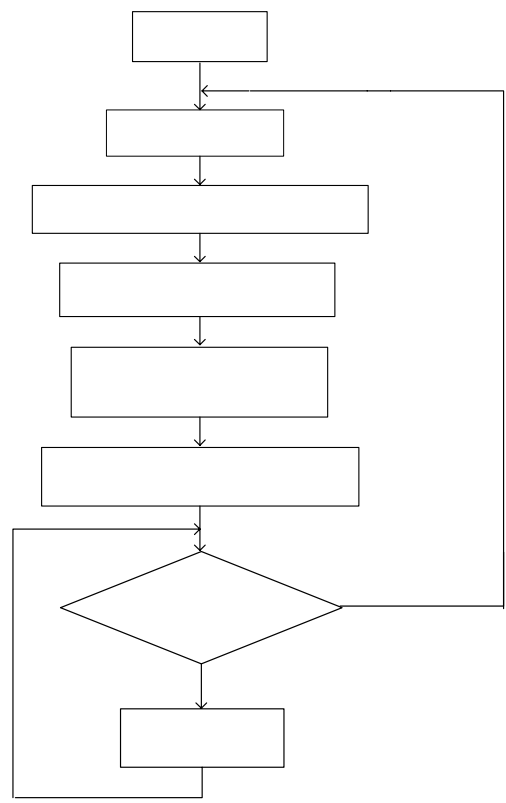


图 5.2 电梯群控系统调度
Fig.5.2 The scheduling of EGCS

5.3 应用于电梯群控派梯调度的模糊神经网络结构

5.3.1 模糊神经网络的输入及其模糊集合

- (1) $HCWT_i$ ：层站召唤信号从产生到被第 i 个电梯响应的等待时间。

$$HCWT_i = t_{move} + t_{stop}$$

其中 t_{move} ：电梯运行时间，包括匀速时间、加速时间和减速时间。

t_{stop} ：电梯停车时间，包括开门时间、客流进出时间和关门时间。

- (2) $\max HCWT_i$ ：分配第 i 个电梯的所有层站召唤信号最大等待时间。

$$\max HCWT_i = \max(t_{assigned}, t_{new}) \quad (5.4)$$

其中 $t_{assigned}$ ：已分配给第 i 个电梯的乘客的等待时间。

t_{new} ：新的厅呼叫分配给第 i 个电梯的乘客的等待时间。

(3) CV_i ：第 i 个电梯响应未来层站召唤的能力。

$$CV_i = 1 - \frac{in_person}{default_person} \quad (5.5)$$

其中 in_person ：当前电梯 i 的乘客数。

$default_person$ ：电梯 i 的额定载客数。

(4) GD_i ：新的层站召唤信号的产生位置与被第 i 个电梯响应的所有召唤信号(包括层站召唤和梯内召唤)产生位置之间的最短距离^[41, 42, 43]。

为了方便运算起见， GD_i 直接采用楼层作为计数单位。

以上数据为了处理方便均进行归一化处理，即对每一个输入量都事先设定设个输入量的最大值，然后用输入量与最大值进行比较从而得到归一化处理量。

对于一个层数为 20 的建筑物，假设每台电梯通过单位楼层的时间为 1s 则有：

层站召唤信号从产生到被第 i 个电梯响应的等待时间($HCWT_i$)的输入最大值为 40s。

分配第 i 个电梯的所有层站召唤信号最大等待时间($\max HCWT_i$)的输入最大值为 60s。

第 i 个电梯响应未来层站召唤的能力(CV_i)本身就是一个比例量，可以直接作为网络的收入。

新的层站召唤信号的产生位置与被第 i 个电梯响应的所有召唤信号产生位置之间的最短距离(GD_i)输入的最大值就是建筑物的楼层数，此处为 40。

以上的输入都可以由电梯运行状态数据计算得到

每一个输入的电梯运行状态参数对应一个模糊集合，每个模糊集合都有数个模糊子集构成，为了简化问题，本文规定所有的模糊集合都有同样数量的模糊子集。

对于上面的输入，每个分量 X_i 均为模糊语言变量，并设其中某个 X_i 的模糊集合为：

$$T(X_i) = \{A_i^1, A_i^2 \dots A_i^{m_i}\}, i=1, 2, \dots, n \quad (5.6)$$

其中， A_i^j ($j=1, 2, \dots, m_i$) 是 X_i 的第 j 个语言变量值，是定义在论域 U_i 上的一个模糊集合，其相应的隶属度为 $\mu_{A_i^j}(X_i)$ ($i=1, 2, \dots, n; j=1, 2, \dots, m_i$)

在这里我们具体的假设每个客流量特征值的模糊集合有 5 个模糊子集，即 $m_i=5$ 。使用模糊语言可以表述为{小，较小，中，较大，大}，为了方便表述我们使用

{L, ML, M, MB, B}来表示。

精确的电梯运行状态参数数据要完成模糊化过程需要通过隶属函数的运算，本文采用的隶属函数形式都是高斯型隶属函数的相互叠加。

如下图 5.2 所示是对应于层站召唤信号从产生到被第 i 个电梯响应的等待时间 ($HCWT_i$) 的模糊集合及其隶属函数图。L 和 B 对应的高斯函数只有一半，其意义是当时时间短到一定程度时直接判断为小，当时间长到一定程度时直接判断为大。每个高斯型隶属函数都有自己的中心，图中的 10%，30%，50%，70%，90% 就是高斯型隶属函数的中心，图中标明的中心只是举例说明，具体在实际中，不同的输入的模糊子集对应不同的中心，需要通过对模糊神经网络的训练和优化获得。通过隶属函数得到的隶属度是一个大于 0 小于 1 的实数。

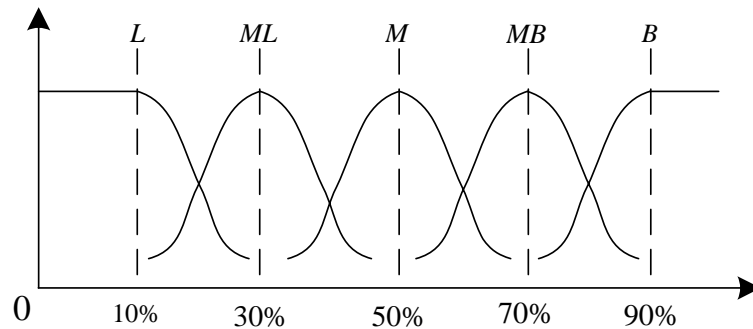


图 5.3 隶属函数图形

Fig.5.3 Shape of membership function

5.3.2 模糊神经网络的输出

电梯群控系统派梯调度所使用的模糊神经网络输出分别是平均候梯时间、长候梯率和能耗的可信度 $Y = [Y_{AWT}, Y_{LWP}, Y_{RNC}]$ 。

其模糊集合为 $T(Y_i) = \{B^1, B^2 \dots B^{m_y}\}$ 。

其中， i 为输出量 Y 的第 i 个分量， B^j ($j=1, 2, \dots, m_y$) 是 Y_i 的第 j 个语言变量值。相应的隶属度为 $\mu_{B^j}(Y_i)$ 。

在这里我们具体的假设每个输出量的模糊集合有 5 个模糊子集，即 $m_i = 5$ 。使用模糊语言可以表述为{小，较小，中，较大，大}，为了方便表述我们使用{L, ML, M, MB, B}来表示。

5.3.3 派梯调度中的模糊规则

本文使用的模糊神经网络的规则表达主要靠前文的网络结构中的第二、三、四、五

层之间的网络权值的连接来实现的。通过这几层之间权值的连接形成的拓扑结构可以形成类似于模糊逻辑中的模糊规则。

由于电梯群控派梯调度的复杂性，本文直接采用成熟的相关规则用于模糊神经网络中，数据表示形式为向量形式。

相关规则表述如下：

有关平均侯梯时间 **AWT** 的可信度的规则，从日常生活经验可以得知当层站召唤信号从产生到被电梯响应的等待时间 **HCWT** 越小，平均侯梯时间 **AWT** 越小，而当电梯响应未来层站召唤的能力 **CV** 越大，也就是说这部电梯更有可能会去响应其他的呼梯信号，所以平均侯梯时间 **AWT** 越大，相关规则表述如下：

If **HCWT** is [1, 0, 0, 0, 0] and **CV** is [1, 0, 0, 0, 0] then **AWT** is [1, 0, 0, 0, 0]

If **HCWT** is [0, 0, 1, 0, 0] and **CV** is [1, 0, 0, 0, 0] then **AWT** is [0, 1, 0, 0, 0]

If **HCWT** is [0, 0, 0, 0, 1] and **CV** is [1, 0, 0, 0, 0] then **AWT** is [0, 0, 1, 0, 0]

If **HCWT** is [1, 0, 0, 0, 0] and **CV** is [0, 0, 1, 0, 0] then **AWT** is [0, 1, 0, 0, 0]

If **HCWT** is [0, 0, 1, 0, 0] and **CV** is [0, 0, 1, 0, 0] then **AWT** is [0, 0, 1, 0, 0]

If **HCWT** is [0, 0, 0, 0, 1] and **CV** is [0, 0, 1, 0, 0] then **AWT** is [0, 0, 0, 1, 0]

If **HCWT** is [1, 0, 0, 0, 0] and **CV** is [0, 0, 0, 0, 1] then **AWT** is [0, 0, 1, 0, 1]

If **HCWT** is [0, 0, 1, 0, 0] and **CV** is [0, 0, 0, 0, 1] then **AWT** is [0, 0, 0, 1, 0]

If **HCWT** is [0, 0, 0, 0, 1] and **CV** is [0, 0, 0, 0, 1] then **AWT** is [0, 0, 0, 0, 1]

有关长侯梯率 **LWP** 的可信度规则，从日常生活经验可以得知当分配的电梯的所有层站召唤信号最大等待时间 **maxHCWT** 越小，长侯梯率 **LWP** 就会越小，而当电梯响应未来层站召唤的能力 **CV** 越大，也就是说这部电梯更有可能会去响应其他的呼梯信号，所以长侯梯率 **LWP** 越大，相关规则表述如下：

If **maxHCWT** is [1, 0, 0, 0, 0] and **CV** is [1, 0, 0, 0, 0] then **LWP** is [1, 0, 0, 0, 0]

If **maxHCWT** is [1, 0, 0, 0, 0] and **CV** is [0, 0, 1, 0, 0] then **LWP** is [0, 1, 0, 0, 0]

0, 0]

If maxHCWT is [1, 0, 0, 0, 0] and CV is [0, 0, 0, 0, 1] then LWP is [0, 0, 1, 0, 0]

If maxHCWT is [0, 0, 1, 0, 0] and CV is [1, 0, 0, 0, 0] then LWP is [0, 1, 0, 0, 0]

If maxHCWT is [0, 0, 1, 0, 0] and CV is [0, 0, 1, 0, 0] then LWP is [0, 0, 1, 0, 0]

If maxHCWT is [0, 0, 1, 0, 0] and CV is [0, 0, 0, 0, 1] then LWP is [0, 0, 0, 1, 0]

If maxHCWT is [0, 0, 0, 0, 1] and CV is [1, 0, 0, 0, 0] then LWP is [0, 0, 1, 0, 0]

If maxHCWT is [0, 0, 0, 0, 1] and CV is [0, 0, 1, 0, 0] then LWP is [0, 0, 0, 1, 0]

If maxHCWT is [0, 0, 0, 0, 1] and CV is [0, 0, 0, 0, 1] then LWP is [0, 0, 0, 0, 1]

有关能耗 RNC 的规则：从日常生活经验可以得知分配第 i 个电梯的所有层站召唤信号最大等待时间 maxHCWT 越小，电梯响应呼梯信号越积极，相应的电梯启动停止就越频繁，导致能耗 RNC 越大，而新的层站召唤信号的产生位置与电梯响应的所有召唤信号产生位置之间的最短距离 GD 越大，表明电梯频繁启停的次数越少，从而能耗 RNC 越小，相关规则表述如下：

If maxHCWT is [1, 0, 0, 0, 0] and GD is [1, 0, 0, 0, 0] then RNC is [0, 0, 0, 0, 1]

If maxHCWT is [1, 0, 0, 0, 0] and GD is [0, 0, 1, 0, 0] then RNC is [0, 0, 0, 1, 0]

If maxHCWT is [1, 0, 0, 0, 0] and GD is [0, 0, 0, 0, 1] then RNC is [0, 0, 1, 0, 0]

If maxHCWT is [0, 0, 1, 0, 0] and GD is [1, 0, 0, 0, 0] then RNC is [0, 0, 0, 1, 0]

If maxHCWT is [0, 0, 1, 0, 0] and GD is [0, 0, 1, 0, 0] then RNC is [0, 0, 1, 0, 0]

If maxHCWT is [0, 0, 1, 0, 0] and GD is [0, 0, 0, 0, 1] then RNC is [0, 1, 0, 0, 0]

If maxHCWT is [0, 0, 0, 0, 1] and GD is [1, 0, 0, 0, 0] then RNC is [0, 0, 0, 0, 0]

1, 0]

If maxHCWT is [0, 0, 0, 0, 1] and GD is [0, 0, 1, 0, 0] then RNC is [0, 1, 0, 0, 0]

If maxHCWT is [0, 0, 0, 0, 1] and GD is [0, 0, 0, 0, 1] then RNC is [1, 0, 0, 0, 0]

5.3.4 派梯调度中的模糊推理

本文的电梯群控系统所使用的模糊神经网络采用前文提出的简化模糊推理方法，即对于给定的输入，只有在输入点附近的那些语言变量值才有较大的隶属度值，原理输入点的语言变量值的隶属度很小或者为 0，当隶属度函数很小时近似的取为 0。因此只有当输入 x 的模糊语言变量与第 i 条规则中对应的语言变量的模糊子集接近的时候，输入 x 对该规则适应度 α_i 才不为零。这样就完成了模糊前件和模糊规则的匹配。

5.3.5 模糊神经网络运算结果处理

将模糊神经网络运算得到的结果进行如下处理

$$S_i = \omega_1 S_{AWT_i} + \omega_2 S_{LWP_i} + \omega_3 S_{RNC_i} \quad \omega_1 + \omega_2 + \omega_3 = 1 \quad (5.7)$$

通过上式计算可以获得响应呼梯信号所指派的 n 台电梯的可信度，然后在所有可信度中选择最大的电梯去响应呼梯信号。

$$S_e = \max(S_1, S_2, \dots, S_n) \quad (5.8)$$

其中 e 为响应呼梯信号的电梯号。

5.4 电梯群控系统派梯调度的训练与仿真

5.4.1 模糊神经网络的训练

实际中的建筑物和电梯群有各种限制和客观条件，为了讨论方便，采用一种简化的模型，忽略某些不重要的细节。基于以上的简化模型，本文所采用的模糊神经网络的训练数据是根据实际情况有目的的选择一天中各个典型时间段的典型乘客流数据。选取训练数据样本的原则是随机的选取，同时兼顾一天中的典型时间段。

用于初步形成模糊神经网络的高斯型隶属函数的单个样本组织形式为：

$$X_i = [X_1, X_2, X_3, X_4, Y_{AWT}, Y_{LWP}, Y_{RNC}] \quad (5.9)$$

样本总容量定为 50 个。训练数据均来自于对某大楼的某几天不同时间段的采样数据作为训练样本。使用的数据只取数据格式的输入部分。

输入 X_1 的 K-均值算法的初始中心为[0.1, 0.3, 0.5, 0.7, 0.9]。

输入 X_2 的 K-均值算法的初始中心为[0.1, 0.3, 0.5, 0.7, 0.9]。

输入 X_3 的 K-均值算法的初始中心为[0.1, 0.3, 0.5, 0.7, 0.9]。

输入 X_4 的 K-均值算法的初始中心为[0.1, 0.3, 0.5, 0.7, 0.9]。

通过 k-均值算法可以得到模糊神经网络输入隶属函数的中心和宽度矩阵 w 。

用于形成模糊神经网络规则的单个样本组织形式为：

$$[HCWT, CV, AWT] \quad (5.10)$$

$$[\max HCWT, CV, LWP] \quad (5.11)$$

$$[\max HCWT, GD, RNC] \quad (5.12)$$

通过使用前文提到的规则进行训练，可以得到模糊神经网络的模糊规则矩阵 $w1$ 和模糊关系合成矩阵 $w2$ 。

用于对模糊神经网络的高斯型隶属函数进行优化调整的单个样本组织形式为：

$$X_i = [X_1, X_2, X_3, X_4, Y_{AWT}, Y_{LWP}, Y_{RNC}] \quad (5.13)$$

样本总容量定为 50 个。训练数据均来自于对某大楼的某几天不同时段采样数据作为训练样本。

通过使用 PSO 算法根据输入的训练样本数据对模糊神经网络输入隶属函数进行优化和调整可以得到完整的派梯调度算法的模糊神经网络模型。图 5.4~5.6 分别反映了分解开的三个动态惯性权值 PSO 算法对模糊神经网络隶属函数中心和宽度优化，其中横坐标为迭代次数，纵坐标为全局最优适应度值。

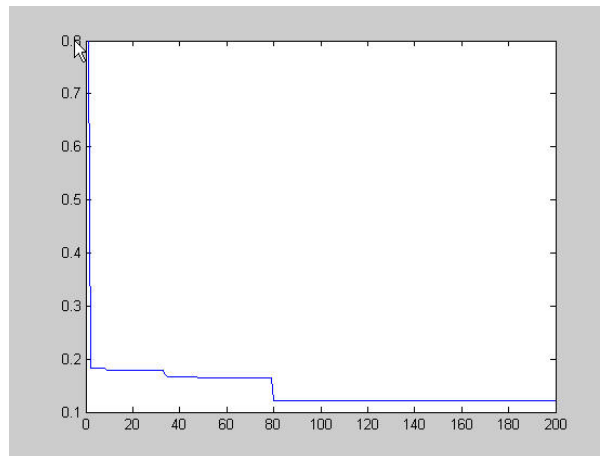


图 5.4 动态惯性权值 PSO 算法对 AWT 模糊神经网络隶属函数的优化

Fig.5.4 The AWTFNN optimization result of PSO with dynamic inertia weight

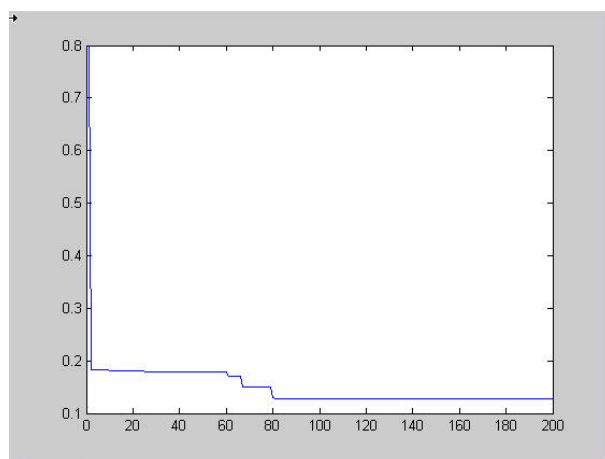


图 5.5 动态惯性权值 PSO 算法对 LWP 模糊神经网络隶属函数的优化

Fig.5.5 The LWPfNN optimization result of PSO with dynamic inertia weight

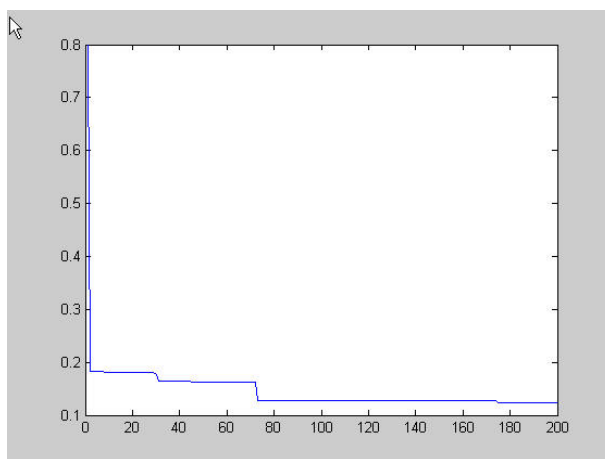


图 5.6 动态惯性权值 PSO 算法对 RNC 模糊神经网络隶属函数的优化

Fig.5.6 The RNCfNN optimization result of PSO with dynamic inertia weight

5.4.2 单个呼梯信号的电梯群控派梯调度仿真实例

(1) 单个呼梯信号调度实例 1

假设建筑物有 20 层，系统有四台电梯同时运行，分别标号 1，2，3，4，每台电梯额定载客数为 20，当前时刻各台电梯载客数分别为：1 号电梯 10 人，2 号电梯 8 人，3 号电梯 18 人，4 号电梯 5 人。四台当前时刻位置分别为：1 号在第 10 层上行，2 号在第 3 层下行，3 号在第 15 层下行，4 号在第 12 层上行。为了方便描述问题，一台电梯通过单位楼层的平均时间为 1s，每次停站平均耗时 2s。具体运行状况如下图 5.3 所示：

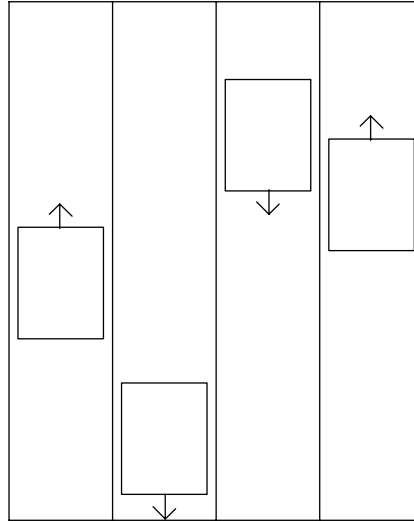


图 5.7 电梯运行状态

Fig.5.7 Working state diagram of elevators

当前经过交通流模式识别得到交通模式为层间模式，所以其权值为 $\omega = [0.35, 0.35, 0.3]$ 。

假设当前第 8 层有上行呼梯信号，则电梯群控系统依次对每一台电梯计算其 $HCWT$, LWP , RNC 的可信度，计算方法就是将电梯运行状态数据 $HCWT$, $\max HCWT$, CV , GD 作为派梯调度模糊神经网络的输入：

$HCWT_1=0.95$	$HCWT_2=0.275$	$HCWT_3=0.575$	$HCWT_4=0.9$
$\max HCWT_1=0.5$	$\max HCWT_2=0.2$	$\max HCWT_3=0.8$	$\max HCWT_4=0.3$
$CV_1=0.5$	$CV_2=0.6$	$CV_3=0.1$	$CV_4=0.75$
$GD_1=0.95$	$GD_2=0.275$	$GD_3=0.575$	$GD_4=0.9$

然后通过前面建立的模糊神经网络来计算其可信度。

$S_{AWT_1}=0.1072$	$S_{AWT_2}=0.2590$	$S_{AWT_3}=0.2596$	$S_{AWT_4}=0.0376$
$S_{LWP_1}=0.2000$	$S_{LWP_2}=0.2663$	$S_{LWP_3}=0.1902$	$S_{LWP_4}=0.0970$
$S_{RNC_1}=0.2502$	$S_{RNC_2}=0.0389$	$S_{RNC_3}=0.2722$	$S_{RNC_4}=0.2181$

将加权值应用在上述可信度后将其相加，可以得到每台电梯响应呼梯信号的可信度：

$$S_1=0.35*S_{AWT_1}+0.35*S_{LWP_1}+0.3*S_{RNC_1}=0.1536$$

$$S_2=0.35*S_{AWT_2}+0.35*S_{LWP_2}+0.3*S_{RNC_2}=0.2627$$

$$S_3=0.35*S_{AWT_3}+0.35*S_{LWP_3}+0.3*S_{RNC_3}=0.2246$$

$$S_4=0.35*S_{AWT_4}+0.35*S_{LWP_4}+0.3*S_{RNC_4}=0.0673$$

综上所述对于当前状况下第 8 层的上行呼梯信号，应该使用第 2 台电梯对其进行响应。从实际情况分析，第 2 台电梯无论从乘客数量还是响应时间都更有优势，第 3 台电

下行
8人
3层

梯的长候梯率可信度较小，这意味着在此次呼梯信号之前有的乘客已经等待第 3 台电梯响应较长时间，故第 3 台电梯应该优先去响应此前的呼梯信号，所以在这种情况下选择第 2 台电梯。

(2) 单个呼梯信号调度实例 2

假设建筑物有 20 层，系统有四台电梯同时运行，分别标号 1, 2, 3, 4，每台电梯额定载客数为 20，当前时刻各台电梯载客数分别为：1 号电梯 12 人，2 号电梯 14 人，3 号电梯 3 人，4 号电梯 8 人。四台当前时刻位置分别为：1 号在第 6 层上行，2 号在第 3 层上行，3 号在第 15 层下行，4 号在第 12 层上行。为了方便描述问题，一台电梯通过单位楼层的平均时间为 1s，每次停站平均耗时 2s。具体运行状况如下图 5.4 所示：

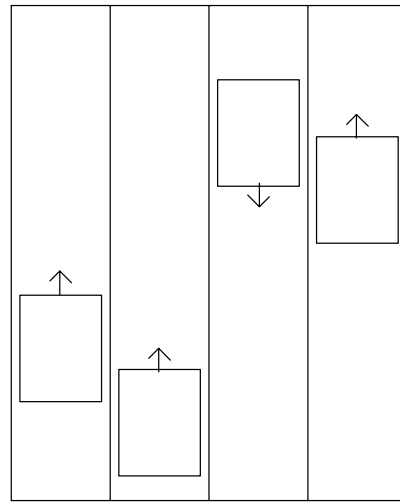


图 5.8 电梯运行状态

Fig.5.8 Working state diagram of elevators

当前经过交通流模式识别得到交通模式为上行模式，所以其权值为 $\omega = [0.45, 0.45, 0.1]$ 。

假设当前第 9 层有上行呼梯信号，则电梯群控系统依次对每一台电梯计算其 AWP, LWP, RNC 的可信度，计算方法就是将电梯运行状态数据 HCWT, maxHCWT, CV, GD 作为派梯调度模糊神经网络的输入：

$HCWT_1=0.05$	$HCWT_2=0.125$	$HCWT_3=0.575$	$HCWT_4=0.5$
$\max HCWT_1=0.1$	$\max HCWT_2=0.1$	$\max HCWT_3=0.2$	$\max HCWT_4=0.6$
$CV_1=0.2$	$CV_2=0.15$	$CV_3=0.85$	$CV_4=0.6$
$GD_1=0.05$	$GD_2=0.125$	$GD_3=0.575$	$GD_4=0.5$

然后通过前面建立的模糊神经网络来计算其可信度。

$S_{AWT_1}=0.2380$	$S_{AWT_2}=0.3329$	$S_{AWT_3}=0.1152$	$S_{AWT_4}=0.1086$
$S_{LWP_1}=0.2663$	$S_{LWP_2}=0.3424$	$S_{LWP_3}=0.0395$	$S_{LWP_4}=0.1810$

$$S_{RNC_1}=0.0357 \quad S_{RNC_2}=0.0389 \quad S_{RNC_3}=0.2722 \quad S_{RNC_4}=0.2534$$

将加权值应用在上述可信度后将其相加，可以得到每台电梯响应呼梯信号的可信度：

$$S_1=0.35*S_{AWT_1}+0.35*S_{LWP_1}+0.3*S_{RNC_1}=0.2522$$

$$S_2=0.35*S_{AWT_2}+0.35*S_{LWP_2}+0.3*S_{RNC_2}=0.2364$$

$$S_3=0.35*S_{AWT_3}+0.35*S_{LWP_3}+0.3*S_{RNC_3}=0.0774$$

$$S_4=0.35*S_{AWT_4}+0.35*S_{LWP_4}+0.3*S_{RNC_4}=0.1448$$

综上所述对于当前状况下第 8 层的上行呼梯信号，应该使用第 1 台电梯对其进行响应。虽然第 1，2 台电梯处于繁忙的上行运输中，调用它们去响应该呼梯信号会影响它们上行运输的效率，但是第 3 台，第 4 台电梯可以迅速回到低层去响应上行呼梯信号。

5.4.3 针对呼梯信号序列的电梯群控派梯调度仿真

假设建筑物有 20 层，系统有四台电梯同时运行，分别标号 1，2，3，4，每台电梯额定载客数为 20，当前时刻各台电梯载客数分别为：1 号电梯 12 人，2 号电梯 14 人，3 号电梯 3 人，4 号电梯 8 人。四台当前时刻位置分别为：1 号在第 6 层上行，2 号在第 3 层上行，3 号在第 15 层下行，4 号在第 12 层上行。为了方便描述问题，一台电梯通过单位楼层的平均时间为 1s，每次停站平均耗时 2s。具体运行状况如下图 5.4 所示：

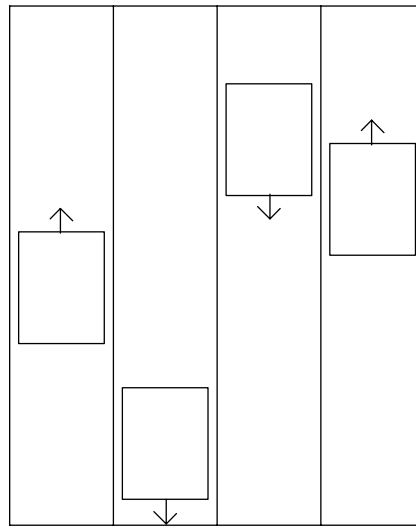


图 5.9 电梯运行状态

Fig.5.9 Working state diagram of elevators

当前经过交通流模式识别得到交通模式为层间模式，所以其权值为 $\omega=[0.35,0.35,0.3]$ 。

假设现在有这样的呼梯信号序列：[-8, 1, 8, -20, 4, -4, 2, -3, 11, -10]，其中“+，-”是为了在 MATLAB 程序中方便计算而人为设置的，分别表示上行和下行。

通过电梯群控派梯调度模块分析，可以得到针对当前电梯群运行状况的派梯调度方

案: [4, 2, 2, 4, 2, 1, 2, 3, 1, 4]。结果中很少使用第 3 台电梯, 因为这台电梯当前载人太多接近饱和, 考虑到电梯内乘客的感受应该尽量让第 3 台电梯尽快完成输送乘客的任务。

针对此次呼梯信号序列的派梯方案的各项性能指标如下:

平均侯梯时间 $T_{aw}=12.4s$

最长侯梯时间 $T_{mw}=36s$

运行总距离 $D_t=248m$

而使用相同的调度策略、隶属函数和模糊规则的模糊逻辑方法对相同的呼梯信号序列进行调度, 不同之处在于模糊逻辑方法的隶属函数采用的是人为事先规定好的高斯型隶属函数, 可以得到模糊逻辑的调度策略: [4, 2, 2, 4, 4, 3, 2, 3, 1, 2], 其性能评价结果如下所示:

平均侯梯时间 $T_{aw}=14.1s$

最长侯梯时间 $T_{mw}=40s$

运行总距离 $D_t=262m$

可见采用模糊神经网络算法的派梯调度方案性能指标分别要比单纯使用模糊逻辑的派梯方案性能提升 13%、5% 和 5.6%, 在性能上有明显的改善。

使用模糊神经网络作为派梯调度控制器相对于模糊逻辑的优势在于模糊神经网络可以使用训练数据通过网络学习来优化调整隶属函数的中心和宽度, 使得隶属函数更加符合实际, 从而模糊神经网络所进行的模糊化和模糊推理更加符合逻辑。并且模糊神经网络还可以通过学习训练更新模糊规则的表达, 使得模糊规则更加符合建筑最近一段时间的实际情况, 考虑到这一情况, 模糊神经网络的优势会更加明显。

5.5 本章小结

本章的主要内容是电梯群控系统的派梯调度。通过前面章节的介绍和论述, 完整的模糊神经网络模型已经建立起来, 并获得了模式识别提供的交通流模式。本章将模糊神经网络模型应用于电梯群控系统的派梯调度, 并详细的叙述了应用于派梯调度的模糊神经网络的结构和参数, 最后针对拍提调度这个具体应用进行了相应的仿真。

第6章 结论与展望

由于电梯群控本身的特点与复杂性,传统的控制理论与方法很难对这种缺乏精确数学模型的对象进行分析和控制。电梯群控的研究重点和热点在于智能算法的应用,本文将模糊神经网络引入电梯群控系统中,介绍了模糊神经网络的算法基础、结构、学习方法,并将以上理论和方法具体应用于电梯群控系统的模式识别模块和派梯调度模块,然后通过 MATLAB 仿真对本文采用的方法进行了有效性和可靠性验证。

(1) 结论

(a) 电梯群控的特点是非线性、不确定性、模糊性和多目标性,基于这些特点,电梯群控缺乏精确的数学模式,这导致传统的控制理论与方法很难发挥作用,需要引入智能算法来解决电梯群控中遇到的问题。

(b) 模糊神经网络融合了模糊逻辑与人工神经网络,同时具备这两种智能算法的优点。相对于模糊逻辑,模糊神经网络具有自适应性,可以自动生成模糊规则并优化调整参数;相对于人工神经网络,模糊神经网络结构简单并且可以处理规则。

(c) 通过混合学习方法对模糊神经网络进行学习和训练,通过三个阶段的学习和训练完成模糊神经网络的构建。首先使用 k-均值算法初步获得隶属函数的中心和宽度,然后使用顺序聚类法获取模糊规则,最后使用一种动态惯性权值的 PSO 算法对网络参数进行优化和调整。

(d) 电梯群控基本的调度方法是首先通过模式识别模块对交通流数据进行分析获取当前交通流的交通模式,根据这个交通模式制定响应的调度策略,然后通过派梯调度模块结合电梯状态数据对呼梯信号序列进行派梯调度。

(e) 交通流模式识别模块和电梯派梯调度模块均使用本文提到的模糊神经网络构建方法构建的模糊神经网络作为主要的逻辑推理机构。最后本文采用 MATLAB 作为仿真工具对模式识别结果和派梯调度结果进行了仿真和验证。

(2) 展望

电梯群控无论作为实际工程还是理论探讨都有着重大的研究价值,未来电梯群的性能、可靠性和舒适性将会进一步提高。

(a) 采用多智能算法来解决电梯群控遇到的问题,将智能控制的最新理论研究成果应用于电梯群控,开辟电梯群控研究的新方法、新领域。

(b) 对于本文采用的模糊神经网络可以在模糊推理方法和人工神经网络上对其进行结构与算法上的改进与完善。

(c) 针对模糊神经网络的学习方法,可以采用其他的方法来完成隶属函数的确定和

模糊规则的生成，使之更有效率更加符合实际。**PSO** 算法作为一种新兴的智能算法，其理论与应用方法都有很多值得研究和改进的地方，藉此效率和性能也可以进一步提高。

参考文献

1. Hitoshi Aoki. Group supervisory control system assisted by artificial intelligence[J], Elevator World, 1990, 2(3):70-80.
2. 杨祯山, 邵诚. 电梯群控技术的现状与发展方向[J], 2005, 12(5): 2-4.
3. 朱德文, 付国江. 电梯群控技术[M], 北京: 中国电力出版社, 2006, 9-11.
4. 张巍. 基于查分演化算法的电梯群控智能控制系统研究[D], 沈阳: 东北大学, 2008.
5. WL Chan. Comprehensive dynamic zoning algorithms[J], Elevator World, 1977, 9(7):99-103.
6. 王含. 基于人工智能的电梯群控算法的研究[D], 沈阳: 东北大学, 2007.
7. mamdani E H, Assilian S. An experiment in linguistic synthesis with a fuzzy logic controller[J], Int. J. Machine Studies, 1994, 7(11):1-13.
8. zadeh L A. Outline of a new approach to the analysis of complex system and decision[J], IEEE Trans System, 1973, 3(1):28-44.
9. 付国江. 电梯群控模糊派梯系统研究与设计[D], 沈阳: 东北大学, 2000.
10. 刘晓环. 基于模糊神经网络的电梯群控系统模式识别[D], 成都: 四川大学, 2003.
11. 郑未名. 模糊神经网络的学习算法[D], 武汉: 武汉大学, 2005.
12. 唐海燕. 基于模糊神经网络的电梯群优化控制研究[D], 哈尔滨: 哈尔滨工业大学, 2006.
13. 王剑. 基于模糊神经网络的模糊规则自动生成方法[D], 北京: 中国科学院研究生院(计算机研究所), 1997.
14. 李国勇. 神经模糊控制理论及应用[M], 北京, 电子工业出版社, 287-289.
15. 宗群, 尚晓光等. 模糊神经网络的电梯系统交通模式识别[J], 系统工程学报, 2001, 16(3): 418-424.
16. 罗秋敏. 基于模糊神经网络的模糊规则获取方法[D], 大连: 大连交通大学, 2004.
17. 郝敏. 基于模糊聚类算法的自适应模糊神经网络研究[D], 哈尔滨: 哈尔滨理工大学, 2007.
18. Kennedy J, Eberhart RC, Shi Y. Swarm Intelligence[M], San Fransico, Morgan Kaufman Publishers, 2001, 11-15.

19. Mataric M. Designing and Understanding Adaptive Group Behavior[J], Adaptive Behavior, 1995, 4(4):14-17.
20. 张顶学, 关治洪, 刘新芝. 基于 PSO 的 RBF 神经网络学习算法及其应用[J], 计算机工程与应用, 2006, 20: 13-15.
21. 周驰. 粒子群优化算法应用研究[D], 武汉: 华中科技大学, 2007.
22. 薛婷. 粒子群优化算法的研究与改进[D], 大连: 大连海事大学, 2008.
23. 金欣磊. 基于 PSO 的多目标优化算法研究及应用[D], 杭州: 浙江大学, 2006.
24. 杨诗琴. 微粒群优化算法 PSO 的改进研究[D], 无锡: 江南大学, 2008.
25. 李明, 杨承, 舒宇. 一种全局收敛的神经网络 PSO 训练算法[J], 仪器仪表学报, 2007, 28(6): 533-537.
26. Eberhart RC, Shi Y. A Modified Particle Swarm Optimization [J], Proceedings of the IEEE International Conference on Evolutionary Computation, 1998,3(10): 69-73.
27. Carlisle A, Dozier G. An Off-the-shelf PSO[M], Indianapolis, Purdue School of Engineering and Technology IUPUI(inpress), 2001, 21-26.
28. 孙书芳. PSO 神经网络及其在板形板厚综合控制中的应用研究[D], 沈阳: 东北大学, 2006.
29. Angeline P J. Using selection to improve particle swarm optimization[M], Anchorage, IEEE Piscataway, 1998, 84-89.
30. Angeline P J. Evolutionary optimization versus particle swarm optimization philosophy and performance differences[M], Berlin, Springer-Verlag, 1998, 601-610.
31. 王晓丽. 粒子群优化算法的研究及其应用[D], 太原: 太原科技大学, 2008.
32. 张学良, 温淑花. 仿生智能算法及其在机械工程中的应用[M], 北京: 中国科学技术出版社, 2004.
33. 张选平, 杜玉平, 秦国强, 覃征. 一种改变惯性权的自适应粒子群算法[J], 西安交通大学学报, 2005, 39(10): 1039-1042.
34. Barney G C, Dos Santos S M. Elevator traffic analysis, design and control[M], England, Peter Peregrinus, 1985.
35. 周玮. 电梯群控系统的调度算法研究[D], 武汉: 华中科技大学, 2004.
36. 宗群, 尚晓光, 岳有军, 雷小锋. 电梯群控系统的交通模式识别[J], 控制与决策, 2001, 16(5): 163-166.

37. 宗群, 王朝阳, 岳有军, 雷小锋. 电梯群控系统多目标模糊控制算法的设计[J], 制造业自动化, 2000, 22(6): 23-25.
38. 姚玉刚, 柏逢明. 模糊控制在电梯群控系统中的应用[J], 长春理工大学学报, 2008, 21(2): 107-110.
39. 杨冬梅. 电梯群控算法网络集成[D], 大连: 大连海事大学, 2001.
40. 万健如, 刘春江, 刘洪池. 基于前向神经网络最佳派梯智能控制[J], 系统工程与电子技术, 2003, 25(12): 466-468.
41. 马福军. 电梯群控技术研究[D], 杭州: 浙江大学, 2000.
42. 吴蕾. 基于神经网络的电梯群控系统职能调度研究[D], 武汉: 武汉理工大学, 2007.
43. 李东, 王伟等. 电梯群控智能系统与智能控制技术[J], 控制与决策, 2001, 16(9): 513-517.

致 谢

值此论文完成之际，谨向所有关心、支持我的老师、同学和朋友们表达最诚挚的谢意。

首先要对我的导师杨卫国副教授表示最真诚的感谢和敬意。在我研究生学习的两年时光里，杨老师在学习、科研、为人各方面都给了我莫大的影响和帮助。杨老师渊博的学识，严谨的治学态度，一丝不苟的科研精神和精益求精的工作作风都为我树立了榜样，使我受益匪浅，在我的学习生涯中留下了难以磨灭的印记。杨老师对我的启迪和教诲都将成为我人生一笔宝贵的财富。

感谢东北大学自动化研究所所有的老师和同学们，感谢他们对我的帮助和启发，是你们让我感受到良好的学术氛围和朝气蓬勃的思想。感谢孟冬冬、田佳健、郑亮、马俊伟在实验室对我的帮助。

感谢我的家人和朋友们，谢谢你们对我一如既往的支持和关心。

最后我要将论文献给我的母亲。

本文使用了 24 个图，44 篇参考文献，共 64 页