

杭州电子科技大学

硕士学位论文

题目: 电梯群控算法及评测系统的研究与设计

研究生 王瀚韬

专业 计算机应用技术

指导教师 李强 副教授

完成日期 2012年12月

杭州电子科技大学硕士学位论文

电梯群控算法及评测系统的研究与设计

研究生：王瀚韬

指导教师：李强 副教授

2012年12月

**Dissertation Submitted to Hangzhou Dianzi University
for the Degree of Master**

**Studying and Designing of Scheduling
Algorithm and Evaluation in Elevator
Group Control System**

Candidate: Wang Han Tao

Supervisor: Associate Prof. Li Qiang

December, 2012

摘 要

高层的商业或居住建筑内的交通复杂多变，为缩短人们的候梯时间，降低电梯的能耗，除了合理安装多台电梯之外，还需要采用优化的调度策略来管理多台电梯，以提高电梯群的运行效率和服务质量，这就引入了电梯群控系统。

本文以电梯群控系统的仿真为主要研究方向，结合国内外电梯领域的相关研究成果，分析了电梯群控系统组成和结构。在此基础上，提出了电梯群控系统仿真平台的系统设计，通过构建客流产生模块、电梯运行模块、电梯群控评测模块和群控调度算法模块，并提出一种基于电梯典型运行阶段的能耗计算方法，完成了仿真评测平台的设计。再从电梯群控的关键问题入手，在研究了 EGCS 的系统功能、系统构成、性能指标的基础上，提出了一种将电梯轿厢内人数作为调度因素，综合考虑乘客平均等待时间、平均乘梯时间、电梯能耗的人工鱼群算法智能调度算法。针对该算法实时性不强的问题，本文又提出了一种借助评价指标和各种电梯交通模式，用人工鱼群算法对模糊推理的线性回归进行优化，来提高派梯算法的运行效果。最后，本文在该仿真评测平台上对实用型群控调度算法及人工鱼群算法等智能算法进行评测。

本文首先阐述了电梯群控算法的国内外研究现状以及发展趋势，对电梯群控的工作原理以及核心技术进行了研究。其次，结合电梯交通系统动态特性的分析，介绍了 EGCS 的性能要求以及性能指标评价函数。接下来阐述如何将人工鱼群算法运用于电梯群控调度领域，通过改进鱼群的步长，并结合电梯运行的实际情况，把轿厢内人数作为调度因素，来解决电梯群控中的组合优化问题。接下来针对电梯群控调度问题，分析电梯群控系统易变因素和电梯群控的多目标，提出基于线性加权平均的电梯群控目标评价函数。在评价函数的计算方式中针对模糊推理缺乏学习性的特点，引入了人工鱼群算法多元线性回归。通过训练人工鱼群算法得到计算电梯可信度的决策函数，并将之利用到群控决策中。然后，在 Windows 操作系统环境下，以面向对象语言 Visual C# 为工具，开发了 EGCS 仿真软件，实现了电梯组优化调度的动态可视化仿真运行。接着通过建立电梯群控系统能耗的算法模型及电梯服务质量的其他评价指标验证算法。最后，通过该仿真评测系统，对比两种不同类型的调度算法。其中实用型调度算法中的积分算法在平均等待时间、平均乘梯时间、能耗上明显优于最小等待时间算法。智能算法中的基于改进型鱼群算法的电梯群控调度算法与基于遗传算法的电梯群控调度算法相比，前者的乘客平均候梯时间、平均乘梯时间、长候梯率较后者有明显降低。

关键词：电梯群控系统，人工智能算法，鱼群算法，电梯评测，电梯轿厢内人数

ABSTRACT

Transportation within high-rise commercial or residential buildings is complicated and diversified. In order to shorten people's waiting time and to reduce elevators' energy consumption, besides installing several elevators properly, it is necessary to adopt optimal scheduling policy to manage these elevators, so as to improve operating efficiency and service quality of elevator group, and elevator group control system (EGCS) is introduced.

This thesis focuses on EGCS simulation, analyzes the composition and structure of EGCS with relevant research results in the field at home and abroad. Based on this, the thesis puts forward the design scheme of EGCS simulation platform. The design of the platform is completed by building passenger flow generation module, elevator operation module, elevator group control evaluation module and group control scheduling algorithm module, with a kind of energy consumption computing method based on elevators' typical running stage. On the research basis of functions, structure and performance indicators of EGCS, a new elevator group scheduling algorithm is presented which applies Artificial Fish School Algorithm (AFSA), and takes the number of passengers inside the elevator car as scheduling factor, meanwhile taking passengers' average waiting time, average riding time and energy consumption into account. And this thesis propose the artificial fish algorithm based on a evaluation index , a variety of elevator traffic patterns, and combined with fuzzy reasoning for the optimize-EGCS dispatching strategy. Finally, some practical and intelligent group control scheduling algorithms such as artificial fish swarm algorithm are evaluated by this simulation & evaluation platform.

This thesis firstly describes the global research status and developing trend of elevator group control algorithm, and EGCS working principle and its core technology. Secondly, combined with the analysis of the dynamic characteristics of the elevator transportation system, this thesis introduces performance requirements of EGCS and its performance evaluation function. Next, it explains the method to apply AFSA to elevator group scheduling algorithm. This is mainly by adjusting the step length of the fish stocks, combining it with the actual elevator running status, and taking the number of passengers inside the car as scheduling factors to solve the combination optimization problem in EGCS. Next, for elevator group scheduling problems, this thesis analyzes variable factors in and multi-objective in the elevator group control system, and proposes Elevator Group Control objective evaluation function based on the linear weighted average. In the calculation of the evaluation function,

for fuzzy reasoning being characteristics of lack of inference learning, this thesis introduces the multiple linear regression of artificial fish algorithm. Through training artificial fish algorithm to calculate the credibility of the decision-making function of elevator, this thesis use the decision-making function in the group control decisions. Then, under Windows operating system environment, and with object-oriented language, Visual C#, the EGCS simulation software is implemented. It realizes the dynamic visual simulation of the elevator group optimal operation. After that, this thesis validates the algorithm by the establishment of the energy consumption of EGCS algorithm model and other elevator service quality factors. And finally, this thesis compares two different types of scheduling algorithms with this simulation & evaluation system. Thereinto, average waiting time, average operation time, and energy consumption of integral algorithm of practical scheduling algorithm are obviously superior to the minimum waiting time algorithm. By comparing the elevator group control algorithm based on improved fish swarm algorithm and the elevator group control scheduling algorithm based on genetic algorithm, the former is obviously inferior to the latter in average waiting time, average operation time, and long waiting rate.

Keywords: elevator group control system (EGCS), artificial fish school algorithm, artificial intelligence algorithms , evaluation of elevator performance , number of passengers in elevator car

目 录

摘 要	I
ABSTRACT	II
目 录	IV
第 1 章 绪论	1
1.1 选题背景及意义	1
1.2 电梯群控系统的现状及其发展方向	2
1.2.1 电梯群控系统及其发展	2
1.2.2 人工智能算法的应用	4
1.2.3 电梯群控虚拟仿真环境的研究现状及进展	6
1.3 论文主要内容与基本结构	9
第 2 章 电梯群控系统的理论基础	11
2.1 电梯群控系统的结构与组成	11
2.2 电梯群控系统动态特性分析	12
2.3 电梯群控的数学模型	14
2.3.1 客流模型	14
2.3.2 召唤模型	16
2.3.3 电梯模型	16
2.3.4 调度策略模型	17
2.4 电梯群控性能的评价指标	17
2.4.1 时间指标	17
2.4.2 系统能耗指标	18
2.4.3 载客能力指标	18
2.4.4 乘客舒适度指标	19
2.5 电梯系统的交通模式	19
2.5.1 上行高峰交通	20
2.5.2 下行高峰交通	22
2.5.3 两路交通	22

2.5.4 层间随机交通	22
2.6 小结	23
第 3 章 基于人工鱼群算法的群控调度策略设计及实现	24
3.1 人工鱼群算法的研究现状及进展	24
3.2 与其他算法的比较	25
3.3 人工鱼群算法描述	26
3.3.1 算法原理	26
3.3.2 行为描述	27
3.3.3 算法描述	28
3.4 基于鱼群算法的电梯群控调度策略设计	29
3.4.1 人工鱼个体设计	29
3.4.2 食物浓度的设计	29
3.4.3 人工鱼的行为设计	30
3.4.4 电梯轿厢内人数	33
3.4.5 参数分析	34
3.5 基于鱼群算法的电梯群控调度策略实现	37
3.6 小结	40
第 4 章 基于模糊推理的多目标控制派梯算法	41
4.1 群控多目标控制的目标函数确定	41
4.1.1 控制目标的提出	41
4.1.2 评价函数的确定	41
4.2 基于鱼群算法的模糊推理线性回归曲线的优化	43
4.2.1 输入量的计算	43
4.2.2 结合模糊推理获得学习样本	44
4.2.3 回归问题	48
4.2.4 人工鱼群算法多元线性回归的利用	48
4.3 小结	51
第 5 章 建立电梯群控虚拟仿真环境设计及仿真	52
5.1 电梯群控虚拟仿真环境的总体设计	52
5.1.1 电梯群控虚拟仿真环境总体流程	52

5.1.2 电梯群控虚拟仿真环境总体框架	53
5.1.3 电梯群类结构	53
5.2 电梯客流模块.....	55
5.3 电梯仿真模块.....	57
5.3.1 电梯动态运行的直观显示	57
5.3.2 电梯构件化	58
5.3.3 电梯运行控制	59
5.3.4 数据通信	65
5.4 电梯群控评测模块.....	67
5.4.1 平均候梯时间	67
5.4.2 平均乘梯时间	68
5.4.3 能耗	68
5.4.4 评测显示	72
5.5 群控调度算法模块.....	75
5.5.1 调度算法公共接口	75
5.5.2 积分算法	77
5.6 系统参数设置.....	80
5.7 调度算法的仿真比较.....	80
5.8 小结.....	85
第 6 章 总结与展望	86
6.1 全文总结.....	86
6.2 进一步的工作.....	87
致 谢	88
参 考 文 献	89
附 录	92

第 1 章 绪论

1.1 选题背景及意义

1854 年,世界博览会于 New York 水晶宫顺利举行,借着这一盛会,Elisha Graves Otis 向世界推出了他最新的研究成果——人类文明史上的首台升降梯。这一研究成果使世人对升降工具的利用发生了翻天覆地的变革。正是这种垂直代步工具的使用,才使得高层建筑的发展成为可能。

新工艺的开发、建筑水平的增强以及新型材料的应用,都使得高层建筑有了质的发展,为了更好的解决建筑内部货物以及人员的输送问题,各个建筑所安装的电梯通常不止一部,然而假使每部电梯都同时运行且彼此之间没有任何联系,那么就无法协调运行,这肯定会产生电梯空载运行或者多部电梯响应同一呼梯信号的现象,尤其是当电梯乘客很多时,必然会导致电梯聚群的问题,这会在很大程度上降低电梯系统的工作效率。所以,仅增加建筑中的电梯台数、加快每部电梯的工作速度、额定载重及其所允许搭乘的人员数量根本不能满足客流量改变较大的建筑的实际需要,而且这些做法还有可能增加电梯系统的能耗,但对于有效缓解客流高峰期所导致的候梯时间过长的问题毫无帮助^[1]。因此,最主要的问题是能不能结合每部电梯所处的实际楼层、轿厢内部的呼梯信号以及大厅的呼梯信号等实际情况,对各电梯进行优化调度和合理分配,由此产生了电梯群控调度问题^[2]。

早在上个世纪 50 年代,两家全球知名的生产电梯的公司(即 OTIS 以及 TECO-Westinghouse Motor)就提出了电梯群控方法,该法能使多部电梯协调工作,即进行统一管理和调度,从而极大的增强电梯的工作效率,其在 1949 年在联合国大楼内得到成功应用^[3]。相关数据显示,1960 年,在高层建筑所安装的电梯中,有 14%适用的使电梯群控,并且这一比例在其后连年攀升。发展到 1976 年,由于计算机、通信、人工智能等多种技术的发展和进步,电梯群控也逐步迈进现代群控系统时期。当今社会,高层建筑已发展为一个城市乃至国家实力水平的标志,由于电梯群控在电梯系统中占据着不可忽视的地位,因此也逐渐成为楼宇自动化的研究热点^[4]。本世纪初期,在所有的新建楼宇中,日本的智能建筑所占比例为 65%,而美国则高达 70%,中国智能建筑所占的比例不是很高,但也在逐渐上升,国外相关机构预测:在所有的新建智能建筑中,中国所占的比例将会高于 50%。

高层建筑中的电梯系统服务质量偏低,业已发展为中国未能很好解决、但却一定要解决的技术难题^[5]。从 2011 年电梯协会所发布的相关数据可以看出,我国每年生产的电梯数量、保有量和安装量均为世界第一,有 170 万台正在运行的电梯,并且在以每年 20%、

超过 35 万台、占全球一半多的增量发展，然而，与此形成鲜明对比的是，我国的电梯市场有超过 75% 的份额掌握在外资企业或者合资公司的手中，本土电梯公司仅仅分享了 25% 不到的份额，造成这一现象的根本就是个别发达国家完全垄断了电梯生产的核心技术^[6]。对于电梯群控系统来说，其核心技术即群控调度算法，因此针对这一算法开展大量的研究工作必然能够在一定程度上推动中国电梯行业的发展。从上世纪八十年代发展到现在，人工智能技术取得了长足的进步和发展，该法在处理模式识别或者复杂控制系统时，和以往的经典方法相比有着独特的优势，如何把它引进到电梯群控系统，从而解决电梯群的协调工作，不但当前群控系统的重点研究方向，而且是群控系统的重难点所在。

为了验证各类不同的群控调度算法，本文提出构建一个通用的电梯群控虚拟仿真系统。该系统将电梯群控算法与电梯运行系统严格分离，以达到为不同群控算法提供开放性的通用测试平台的目的，有利于不同群控算法之间的横向比较。同时，系统还可以方便的修改大楼参数、电梯参数，可以模拟各种典型的乘客到达情况，以构建各种仿真环境，有利于同一群控算法在不同仿真环境下的纵向比较。总之，通过构建通用的电梯群控仿真系统，可以有效地解决电梯群控研究中验证困难的问题，具有重大的理论意义。

1.2 电梯群控系统的现状及其发展方向

1.2.1 电梯群控系统及其发展

从首部安全升降梯被发明出来到现在，电梯控制技术的有了很大的进步和发展，群控技术也经历了几个不同的阶段。初始阶段即通过继电器接触控制 2~3 部电梯，然后随着集成电路的发展，使得借助部分复杂的调度算法协调多部电梯的运行过程得以实现，一直到当今社会的将计算机技术引进到群控系统里。详细的发展过程如下表所示（即表 1.1）。

表 1.1 电梯群控系统所经历的发展阶段

阶 段	电 梯 群 控 系 统		分 派 系 统	控 制 系 统
1941-1971 第一阶段	自动模式选择系统		区间分配系统	时间间隔控制
1971-1975 第二阶段	集成电路		呼梯分配系统	等待时间预测控制
1975 至今 第三阶段	计算机控制	最短待梯时间控制 综合评价系统应用 人工智能		带有学习功能的 预测

(1) 第一阶段。电梯群控系统在刚刚投入使用时，是借助继电器来实现对电梯的顺序控制的，该法结合电梯在各个时间段的具体的运行模式，从而确定相应的运行命令。根据时间间隔来确定控制方式，也就是说电梯的运行方式和呼梯信号没有关系，而是设置一个合理的时间间隔，当电梯从始点投入运行后，使其在每个楼层之间顺序起停。而

且为了减少电梯来回运动的周期，降低轿厢在每层楼的停梯次数，根据分区配置^[7-9]，使每部电梯的服务相应区域的呼梯信号。而且在分区时也可适用两种方式，即：一、静态分区；二、动态分区。如果为静态分区，聚集一定数量的楼层即可组建一个服务分区，也可以依照紧挨着的上行或者下行请求构建对应的需求分区。如果为动态分区，则是依照提前确定的原则，再加上每部电梯的实际位置、状态及其运行方向来确定分区的范围以及数量。动态分区并非一层不变的，而是时刻处于改变之中，因此其所用的算法也十分复杂^[10]，应用最多的是静态分区。在这一阶段，有效的缓解了每部电梯在建筑内部均匀分布的现象，然而其缺点和不足也十分明显：在楼宇的顶部或者底部，电梯等待分配耗时过多，电梯在顶部或者底部的停层一般没有任何意义，而且即使没有呼梯信号，电梯也依然不停的上下运行。这一阶段的群控系统硬件系统十分复杂、服务质量低下、可靠性不高。

(2) 第二阶段。上个世纪七十年代，集成电路技术取得了长足的发展和进步，这一技术也被逐渐的引进和应用到了电梯群控系统中，从而使其服务效率以及可靠性均大幅增强。不但使系统的硬件复杂度以及维护难度大幅下降，而且使很多较为复杂的算法得以实现。该阶段的控制方式能够根据各个呼梯信号依次派梯。如果在层站中新增了呼梯信号，那么系统就能依照每部电梯所处的实际位置以及运行状况来估计候梯时间，并依次为依据来确定最佳的响应该信号的电梯。其缺点即不能十分准确的计算等待时间。

(3) 第三阶段。从1975年开始，计算机技术便开始被逐步引入群控系统^[11]。借助不同的软件使各种常规控制算法得以实现，然后使其应用到群控系统里。众所周知，对于常规控制算法，其逻辑是一层不变的，这使其性能大幅降低，因此其并非最佳的方法，通过计算机能够编写和常规控制算法迥然不同的控制系统，依照确定的原则来计算各部电梯去应答呼梯信号所需的时间，然后依次为依据进行派梯，除此之外，该系统还要综合考虑运行方式、电梯运行优先级等问题。通过计算机技术，可以实现对算法所涉及参数的改变与控制，然后将变化之后的程序再次送进进算计，无需布线即可实现对算法的改进和完善。另外，计算机保存和分析数据的能力十分惊人，因此能够轻易的保存电梯的工作数据、性能指标和发生故障时的检修数据。

电梯的发明以及大力推广使世人的日常生活更为便捷，为了更好的迎合现代建筑和使用人员的实际需求，我们必须坚持改进和完善电梯技术，结合相关文献和资料，我们可以将群控技术的前进方向归纳为下述几点：

(1) 智能化。当今社会，科技的主题之一即为实现设备的智能化^[12]，也就是说，如果楼宇的功能出现改变，那么用户可以适当改变和调节相关参数或者系统能够借助在线学习及时改变相关参数，从而确保系统始终以最佳的状态运行。

(2) 节能化^[13-14]。随着人们环保意识的增强，绿色电梯应运而生，这就要求我们这电梯的生产、配置、安装以及运行的过程中设法降低其能耗及其造成的污染。一种可行很强的方案即借助群控算法来完成对交通流的模式识别，从而使电梯的起停数量降到

最低，即降低满足客流量所需的运行次数。

(3) 人性化。在不久的将来，高层建筑每层的乘客均能结合自身的实际需求定制对应的电梯服务，从而实现人性化服务，但其前提是确保对别的楼层的乘客没有影响。

计算机技术的进步和发展使得将人工智能被引入和使用到电梯群控系统里成为可能。借助智能算法（模糊控制、遗传算法、神经网络等）来描述电梯群控的各种性能使其各项性能得到大幅增强。通过和以往电梯系统的比较可以发现，电梯行业的相关技术已经取得了长足的发展和进步。对于电梯技术发展过程中所涉及的各项重要技术可以归纳为下图。

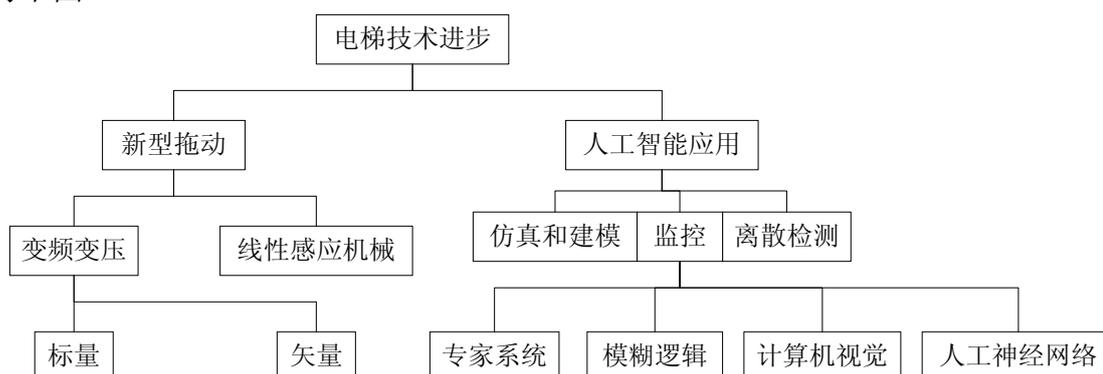


图 1.1 电梯技术的发展历程所涉及的主要技术

1.2.2 人工智能算法的应用

表 1.2 各公司不同特色的群控产品及特点

公司名	产品名	引用算法
OTIS	Elevonic Class	奖惩机制算法
MITSUBISHI	Sigma-AI2200	神经网络和模糊逻辑
HITACHI	FI-340G	遗传算法
SCHINDLER	AJTP	模糊控制与神经网络
FUJITEC	Flex8820/8830	模糊推理与自学习
KONE	TMS9000	模糊逻辑
TOSHIBA	EJ-1000	模糊逻辑与神经网络

人工智能技术是研究、开发用于模拟、延伸和扩展人的智能的理论、方法、技术和适用系统的新兴科目。1984年，日本的著名公司 **HITACHI**借助人工智能技术开发出了 **CIP-52000**系统^[15]，该系统能够依照楼宇中客流的变化而确定出最佳的控制参数。应用该系统的楼宇将能使用户的候梯时间降低10%，同时也将使电梯运行的能耗大幅降低。科学技术的不断发展使得部分大型企业先后推出了采用专家系统（Expert System）、模糊逻辑（Fuzzy Logic）、人工神经网络（Artificial Neural Network）及遗传算法（Genetic Algorithm）等人工智能技术的群控电梯。不同企业的各种群控产品如表1.2所示^[16]。

(1) 专家系统。该系统就是把专家的知识以及经验合理的合并到系统里^[17-18]，而且还将和优化派梯一起使用。它和群控系统一起使用的起源较早，比如MITSUBISHI的AI2100以及AI2200均是以专家系统为基础研制而来的。众所周知，电梯群控的控制过程十分复杂，其相关特性也很难借助数学模型准确反映，所以，如果可以合理的使用专家

的知识以及经验，并依此为依据来处理电梯群控系统将具有十分明显的优势。专家系统通常由以下部分组成：知识库、推理机、数据库、翻译缓解、搜集知识的环节以及人机接口。第一步即借助知识获取得到专家的知识 and 经验，然后将其凝练成确定的控制标准保存到知识库。而数据库的作用则是保存系统业已知晓的状态、乘客供应的事实以及推理的相关结果。比如楼宇中呼梯信号的分布状态、每部电梯的实际位置、电梯运行至呼梯信号所需的时间等。推理机的作用则是和专家系统一起管理组成部分的运行，它可以依照所获取的数据以及知识库里保存的知识或者经验，根据确定的推理算法来选择最合适的电梯。翻译环节即翻译系统的推理构造，同时还要很好的解答乘客的疑惑。图1.2即为专家系统的配置。然而，因为专家的经验 and 知识并不全面，其控制标准也不健全，因此派梯无法满足各种大楼的不同需求。

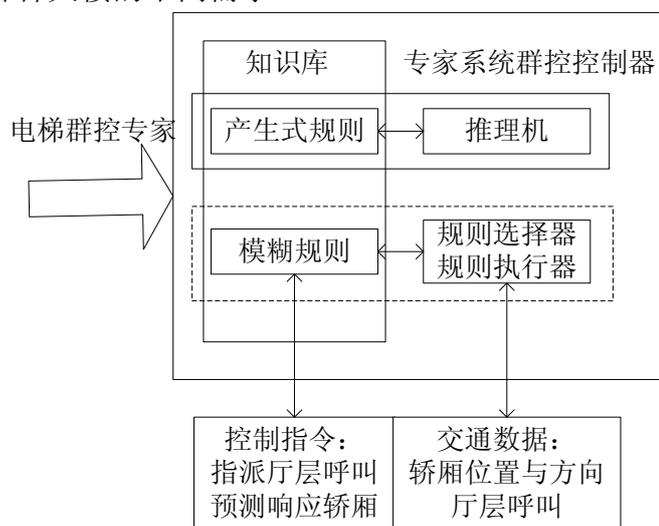


图 1.2 电梯群控专家系统配置

(2) 模糊控制^[19-20]。对于任意一个电梯群控系统，其中必然包含很多不确定性，且其复杂程度也十分惊人，再加上系统的状态无法合理的预测，因此对于群控系统来说，创建其准确的模型是十分困难的。在电梯群控系统中，客流量的规模、候梯时间、响应呼梯信号的速度等，都是很难用明确的数字来定义的。利用模糊控制技术^[21]，能够借助隶属度函数把上述一系列的不确定因素转变成清晰明了的状态来求解。富士通公司于上世纪八十年代末研制成功的FLEX-8800系统的基础即为模糊控制，其根据楼宇里所有电梯的运行状态以及交通情况，借助超过三十种的评价指标来实现多级决策。

(3) 人工神经网络。从上世纪九十年代初开始，人工神经网络即已经被应用到电梯群控系统里^[22-23]，该系统能够克服模糊群控系统缺乏自学习能力的缺点，能够正确高效的识别交通流量。神经网络与模糊逻辑相结合，构成模糊神经网络，能够实现动态分区控制以及建立交通流预报模型^[24]。采用神经网络技术的电梯群控系统如SCHINDLER公司研制的MICONIC VXTM/AITP和TOSHIBA公司研制的EJ-1000FN。带有神经网络技术的群控系统和模糊群控系统比较起来，可以看出它的候梯时间以及长时间候梯的概率都又了明显的改进。其结构图如图1.3所示。

(4) 遗传算法^[25-27]。遗传算法应用于电梯群控系统始于1993年，即日本HITACHI公司推出的FI-340G系列电梯，该算法根据当前电梯群的运行状态及内外召分布情况，在有限时间内搜索求解出相对最优的派梯方案。尽管遗传算法能够求出全局最优解，然而其本身的特性导致其搜索效率十分低下，这也会在很大程度上影响群控调度的性能。

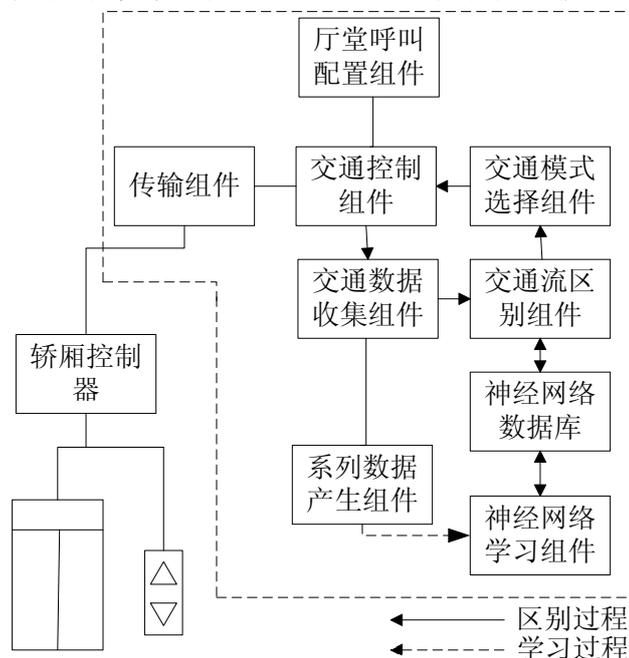


图 1.3 带有神经网络的群控系统结构

1.2.3 电梯群控虚拟仿真环境的研究现状及进展

目前，验证和分析群控系统群控算法一般通过两种途径实现，在电梯群控系统的研究中，最理想的试验途径是将新的群控算法应用到实际电梯服务系统中，并对实际运行的数据进行记录分析，从而进一步确定新的控制算法是否有效地提高了系统服务性能。同时，通过对运行数据的分析，还可以为后续的改进完善等工作提供依据。但出于运行安全性和试验成本的考虑，极少有研究者采用这一方式。只有一些大的电梯公司才有条件、有能力应用。目前，普遍的解决途径仍然是采用仿真技术。为了降低成本，一般先要在PC机上进行模拟测试^[28, 29, 30]，验证其有效性、正确性，然后再将群控算法应用于实际的电梯群控系统之中。目前国外各大电梯制造商均有自己的仿真软件，用以对群控调度策略进行模拟测试和试验^[31, 32, 33]，但均不对外公开。

对于群控仿真平台主要有两大类组成。

(1) 纯软件群控仿真平台。目前，商业化运作的电梯群控仿真软件只有Elevate[™]一款，但其价格相当昂贵，一般中小企业无法承受。ElevateIII能为电梯制造商提供开发和测试工具，能为建筑商提供电梯优化配置的依据，还能向用户提供购买电梯的技术咨询及服务。软件结构包括主界面、分析数据、建筑数据、电梯数据、乘客数据及工程数据等。它以数字和图表的形式给出电梯交通平均候梯时间、最长候梯时间、平均输送时间及最长输送时间数据，但不足之处是电梯交通性能指标较少，如缺少平均乘梯时间。

其运行界面如图1.4所示,在此仿真平台上,AlbertSo等人对电梯群动态分区进行了仿真测试^[34,35]。1999年天津大学的宗群^[36,37]等人在vc++语言环境下开发了一款电梯群控系统仿真软件,该软件主要用于大楼的电梯配置设计和群控算法的验证。2000年华南理工大学的李中华^[38]等人开发了电梯群控仿真平台,用于验证某些改进群控算法的有效性。2004年许玉格^[23]等人在分析电梯群控系统动态特性的基础上,把细胞自动机理论引入到电梯群控系统之中,使仿真电梯模拟真实的电梯运动情况,并在此平台下验证目的层预约群控调度策略的有效性。目前这些平台多用于电梯群及其调度方案的模拟配置,开放性较差,不利于新的调度算法的研发与测试。

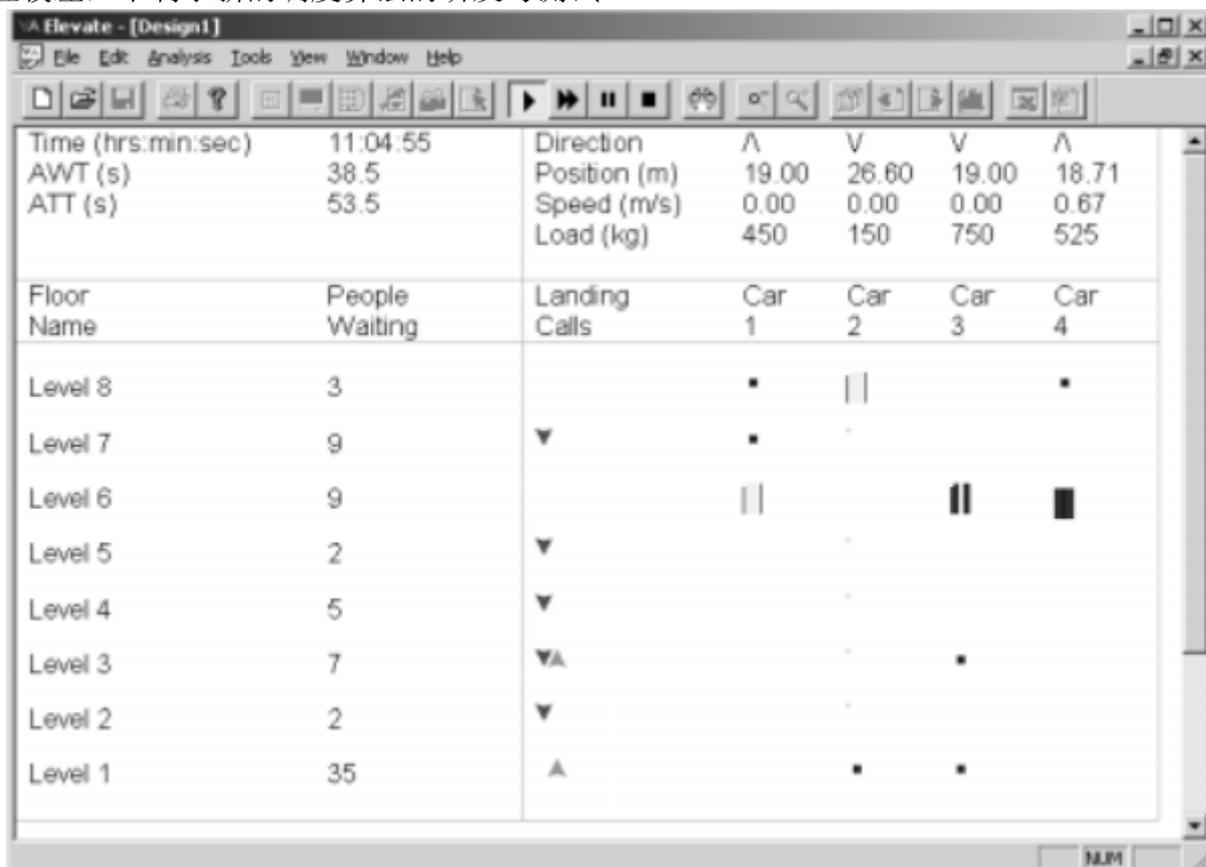


图1.4 Elevate运行主界面

(2) 软硬结合群控仿真平台。硬件系统采用分布式控制,单梯运行仿真由各自相应的微控制器单元完成,群控及实时运行监控功能采用面向对象技术在工控机上实现各功能模块之间采用CAN总线通信。2003年上海交通大学的余兴智^[39]等人研究的硬件试验台由四台群控电梯运行仿真器组成,共有32层站,可仿真电梯实时运行时的信号输入输出以及群控管理情况,控制系统包括外呼信号管理、轿内指令管理、电梯主控制器及基于PC机的群控控制器、输入输出部分包括轿内指令操作盘、外呼按键阵列和电梯运行状态显示屏等。图1.5所示为电梯群控系统仿真实验台的硬件结构图。图1.6上位机软件框图。然而,因为硬件设计上的局限性,不能满足各种大楼在电梯数量、楼层数等不同需求。

目前,电梯能耗的研究主要集中在国外的电梯制造商和各研究机构。文献[40]和文

献[41]使用电能表测量电梯在固定运行工况下的总能耗，但是不足以说明其他工况下的能耗；文献[42]对两种不同类型的电梯进行能耗计算；文献[43]对不同驱动类型电梯的日常能耗进行了测量，但不能说明电梯具体工况下的能耗。文献[44]利用经验公式对电梯年能耗进行计算。此外，中国特种设备检验协会在电梯能耗方面做了大量的工作^[45,46]。

国内外许多学者在电梯群控虚拟仿真环境及能耗分析上进行了研究，但不同学者的仿真环境、能耗计算方法均不同，终究没有一种统一的仿真测试平台和能耗建模方式，因此难以对不同群控调度算法的性能进行比较。

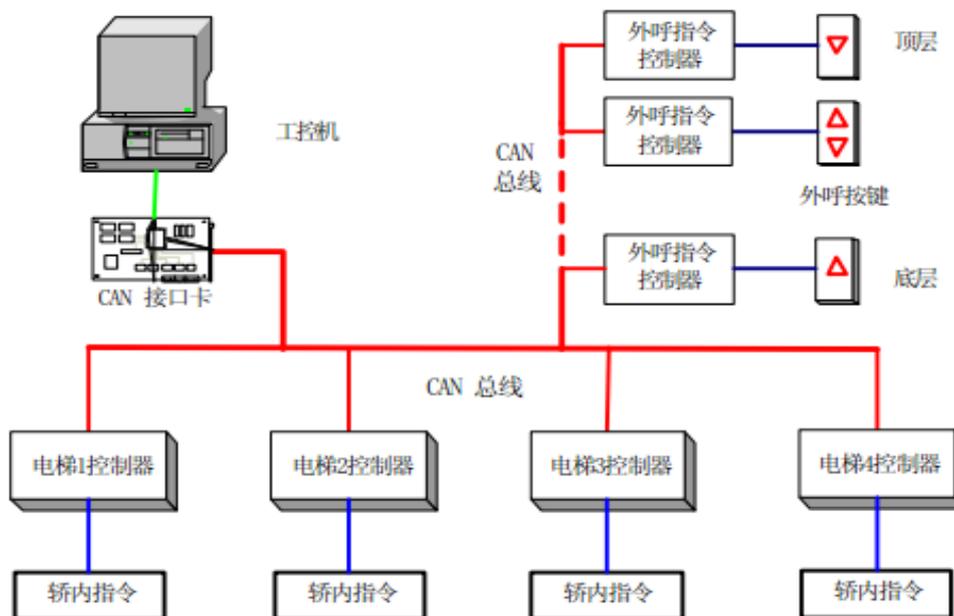


图 1.5 试验台硬件结构示意图

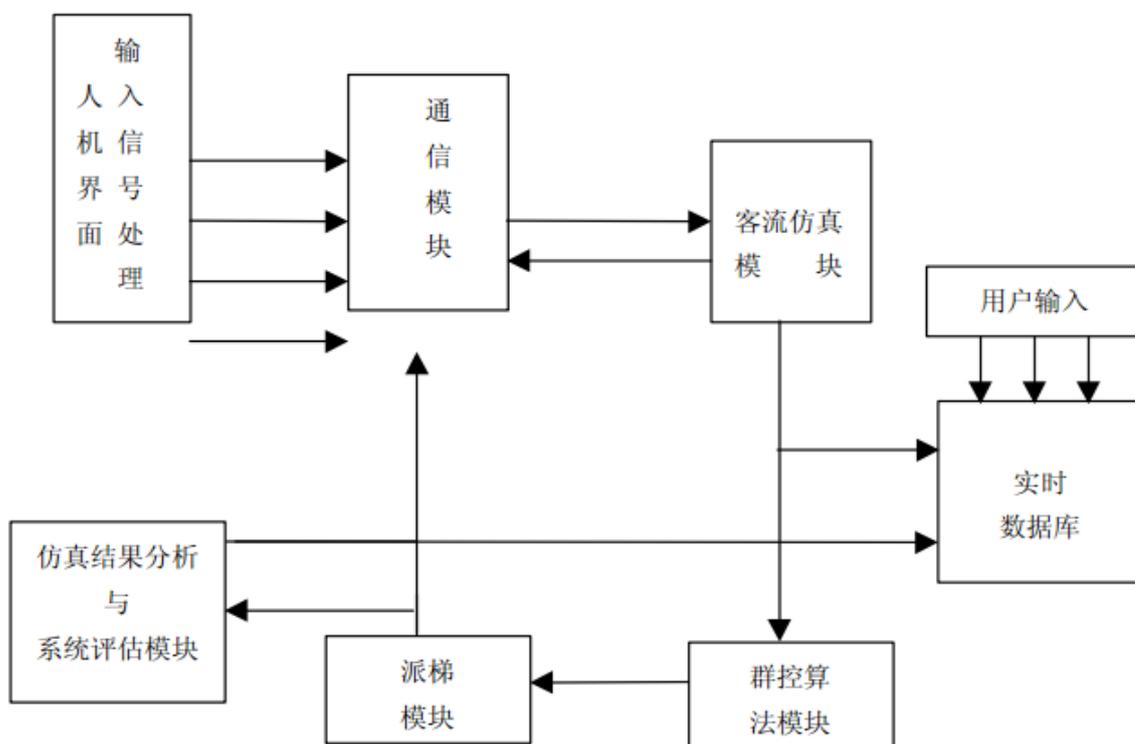


图 1.6 上位机软件框图

1.3 论文主要内容与基本结构

随着人工智能技术的蓬勃发展，人们不断的将其思想引入电梯群控系统，提出了各具特色的智能化调度算法。本文理论上研究的人工鱼群算法在电梯群控调度算法上的一次尝试，是人工智能技术的又一发展。本文的重点是，根据乘客到达的随机性以及大楼具体需求的多样性来设计仿真测试平台，该平台不仅可以对实用型群控调度算法进行评测，而且还可以对人工鱼群算法、遗传算法等智能算法进行验证。

本文针对电梯群控系统，从系统性能、系统构成以及运行模式等方面展开了全面的分析和研究。课题的主要工作可以归纳为下述几点：

(1) 引入人工鱼群算法 (Artificial fish school algorithm) 到电梯群控调度算法中。

(2) 提出了一种基于电梯典型运行阶段的能耗计算方法。

(3) 采用C#编写电梯群控虚拟仿真系统，实现动态加载c++编写的群控算法DLL文件。以通用、模块化的思想编写程序，建立单部电梯运行模型、客流生成器、评价体系、群控调度算法接口，模拟真实电梯群控的运行，从而为进一步分析电梯群控系统及其运行模式创建了优良的实验环境。

(4) 在楼宇和电梯配置参数完全一样的条件下，依次对实用型群控算法中的积分算法、最小等待时间算法进行评测，然后对鱼群算法等智能算法进行验证。

全文结构如下：

第一章 首先介绍了本文的选题背景和意义，接着描述了电梯群控技术的研究现状及其发展趋势，总结了人工智能算法在群控中的应用，阐明了直到今天仍未解决的难题以及具有良好借鉴作用的地方。最后总结了仿真测试平台的研究现状及进展，并分析了不同仿真平台的优缺点。

第二章 首先简单阐述了电梯群控系统的构成，分析了电梯群控系统的动态特性，接着描述了电梯群控的数学模型和电梯群控性能的各项评价指标，最后总结了电梯交通流的模式特征。本章阐述的理论为以后各章提供了研究基础。

第三章 首先介绍了人工鱼群算法的研究现状及其发展趋势，并在理论上分析其优越性，接着阐述如何将人工鱼群算法运用于电梯群控调度领域，通过对部分结构的改变，把轿厢内人数作为调度因素，来解决电梯群控中的组合优化问题。

第四章 首先针对电梯群控的多目标性提出基于线性加权平均的电梯群控目标评价函数。在评价函数的计算方式中针对模糊推理缺乏学习性的特点，引入了人工鱼群算法多元线性回归。通过训练人工鱼群算法得到计算电梯可信度的决策函数，并将之利用到群控决策中。

第五章 论述了关于电梯运行控制、数据通信模块、指标评测模块、群控调度算法模块等电梯群控虚拟仿真环境的设计情况及电梯群控系统能耗的算法模型。在该仿真平台上，对于本文提出的基于改进型鱼群算法的电梯群控调度算法与基于遗传算法的电梯

群控调度算法及实用型调度算法展开测试、对比与分析。

第六章 总结了上文所做的工作，同时展望了下一步的工作。

第 2 章 电梯群控系统的理论基础

2.1 电梯群控系统的结构与组成

电梯群控系统有多种多样的结构，尽管电梯的系统结构会因制造公司的不同而表现出一定的区别，然而其基本结构却是一样的。电梯群控系统通常是由二部及二部以上的电梯构建而成的，在这些电梯中，有一部是群控主机。

根据单体运行逻辑可以实现单体控制系统，以楼层呼梯顺序相应呼梯操作是单梯控制系统的控制方式。“自动定向，反向保号，顺向消号”是单梯控制系统必须遵循的基本原则^[47]。外召电脑板是外召信号的发送装置，在每个楼层都会安装，主要由以下几部分组成：按钮灯、CAN 通信模块、地址码设置按钮等。在楼宇的底层只有向上的呼梯按钮，在楼宇的顶层只有向下的呼梯按钮。

轿厢电脑板由三部分组成，即输入、输出和通讯端子，主要负责采集内召信号。主控板电脑是整个系统的核心，通过 CAN 串口与内召和外召电脑板相连，具有很强的运算和处理能力，实现数据的可靠传输。如图 2.1 为单梯的系统结构图。

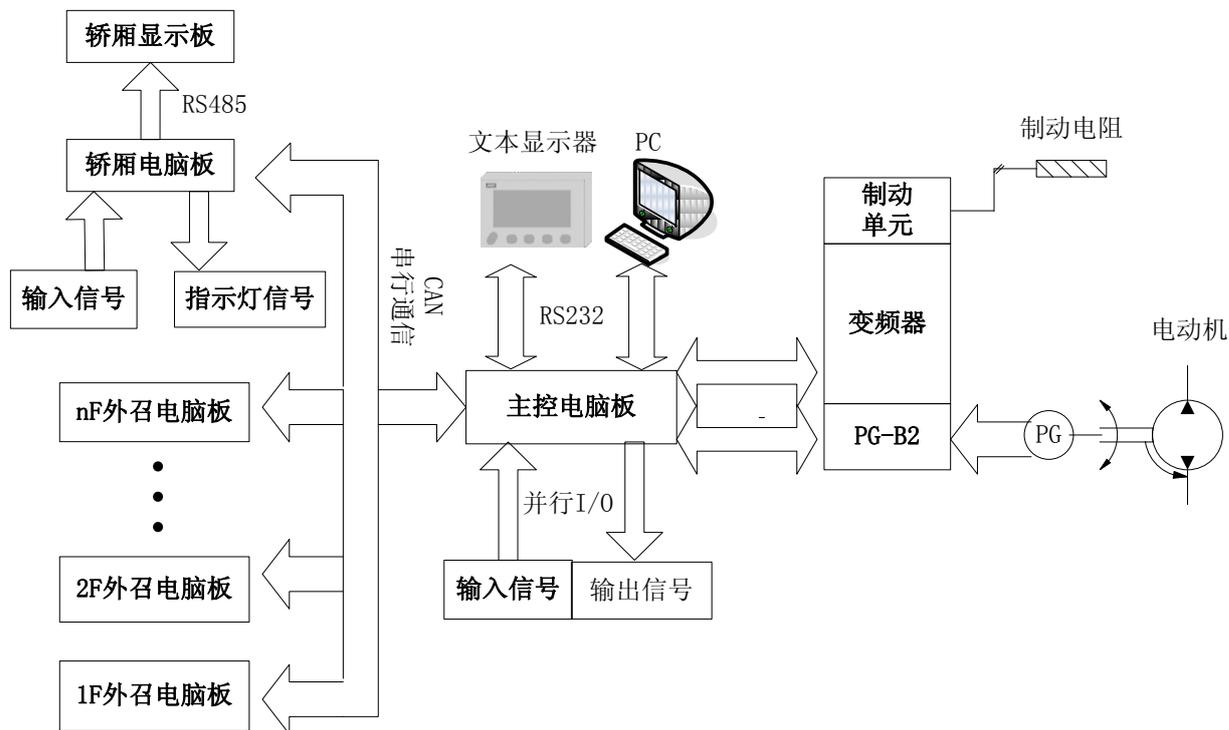


图 2.1 单梯系统结构图

每一楼层的外召信号系统对于参与群控的电梯来说是共享的，群控主机按照特定的群控算法选取最优的调度方案分配给电梯，要想合理的控制多台电梯，提高电梯的运作效率，需要选择合适的群控算法。电梯群控系统主要包括以下几个组成部分：轿厢、轿厢控制、群控和通讯系统。此外还包括一些辅助设备，如显示系统，呼叫按钮等。模块

化结构是最常用的群控系统软件设计方法。图 2.2 为群控系统层次图，由图可以看出，群控系统共分为三层，即群控层、接口层和设备层。群控主机通过接口层实现与单梯的双向通信，收集呼梯信号和各台电梯的选层信号，并负责发送调度控制指令。

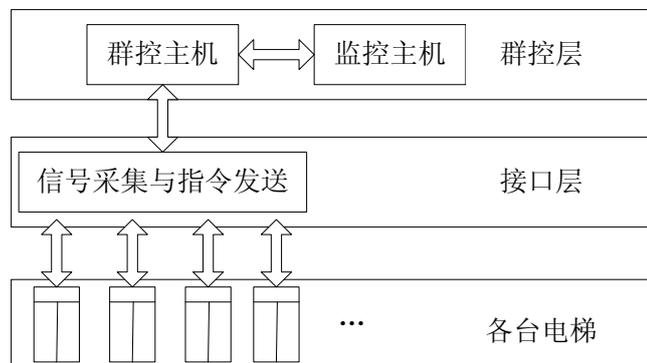


图 2.2 群控系统层次图

以电梯从 5 层运行到 20 层为例，电梯群控的工作流程可以分为以下七个步骤：

- (1) 在起始层 5 楼乘客按下上召按钮，发送呼梯信号；
- (2) 接收呼梯信号；
- (3) 根据呼梯信号所在楼层、运行方向、载重，合理制定调度方案，并按照最优调度方案执行电梯相应；

(4) 电梯群收到控制系统的调度指令以后，按照指令运行到 5 层。同时，利用显示设备实时监控电梯运行情况，保证电梯正常运作；

(5) 当轿厢运行至 5 楼，取消上行召唤指示灯，乘客进入轿厢并按下 20 层的电梯按钮，提交升/降梯信号；

(6) 电梯关闭电梯门，开始向上运行；

(7) 电梯成功将乘客送到 20 层。

上述七个步骤的重复运作便构成了电梯群控制系统。群控算法选择最优的调度方案，安排合适的电梯去响应召唤的过程称为派梯。

2.2 电梯群控系统动态特性分析

(1) 模糊性

实践表明，电梯群控系统存在明显的模糊性，产生模糊性的原因主要有以下几点：

- 1) 未知和不确定因素的存在。如：客流密集出现的时间和数量都是不确定的，乘客起始层和目的层的未知性；
- 2) 建筑物用途类型、规模等环境因素影响楼宇内交通情况；
- 3) 乘客出现的时间不确定。

这些都给电梯交通带来了模糊性。电梯群控系统是一个复杂的电机一体化设备，由多个相互独立又紧密联系的功能模块组成，系统的正常运行受多方面因素的影响和制约，如：建筑规模和用途，电梯类型、容客量和服务方式，客流量等，都能够影响电器

群控系统的性能。精确的数量定义很难描述未知、不确定的因素。为了更全面的描述电梯群控系统，定义和描述电梯群控系统时也使用了一些模糊概念，如：响应时间和速度、客流密集时间和数量、最大容量、候梯时间等。针对这类模糊问题，可以通过模糊数学^[48]很好的解决。电梯群控系统中的模糊概念可以模糊数学中的隶属函数形式进行描述，能够将复杂的问题简单化，将模糊的问题具体化，有效消除计算机芯片只能处理精确问题的限制，提高了计算机推理模糊逻辑的能力。如图 2.3 所示，是一个典型的模糊控制流程图。模糊逻辑能够将模糊概念具体化，确定未知因素，实现模糊逻辑的简化。

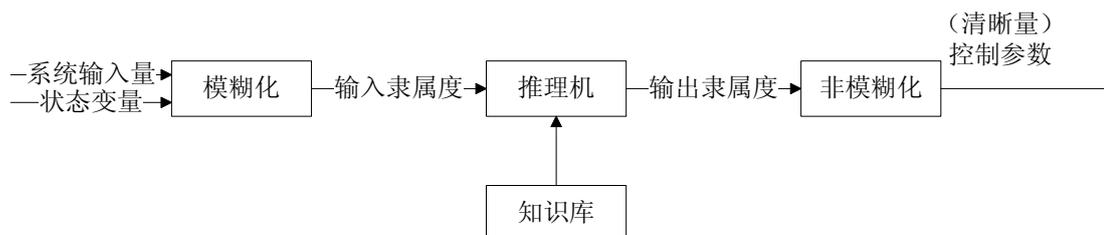


图 2.3 系统的模糊控制流程图

(2) 非线性

实践表明，电梯群控系统存在明显的非线性，产生非线性的原因主要有以下几点：

- 1) 最优轿厢的分配呈现明显的非线性关系。在不同的时间段下，呼叫同一楼层的电梯得到的结果是不同的。
- 2) 轿厢数目和分配状态呈明显的非线性关系。
- 3) 不同楼宇的轿厢分布呈现出明显的非线性和离散关系。
- 4) 在中间楼层时，电梯往往会改变反向。

这些问题是常规控制算法无法解决的问题。要想解决非线性特征问题，可以使用智能控制的方法。

(3) 扰动性

实践表明，电梯群控系统经常会随机出现扰动性，最常见的扰动性主要有以下几种情况：

- 1) 错误的内召信号和外召信号导致轿厢多余的停靠；
- 2) 电梯门之间存在障碍物或乘客在轿箱内长时间按下开门状态按钮使得电梯门无法关闭，导致行驶延误；
- 3) 因特殊原因导致临时改变目的楼层；
- 4) 内部和外部因素的干扰，会影响电梯的运行情况。外部因素主要波啊快：温度和湿度的变化，负载变化，误操作等人为因素，撞击、地震等自然因素；内部因素主要包括：设备的老化，系统功能的不稳定，软硬件不完整等。
- 5) 电磁干扰的影响。

可以从硬件设备和软件设计上入手，采取一些抗干扰措施来解决物理因素和环境因

素带来的扰动,其他影响因素并不能根本杜绝,但是可以通过预测手段进行防范。因此,误差预测能力是电梯群控系统十分重要的功能之一。为了应对各种非物理干扰,提高系统的稳定性和效率,满足不同系统的要求,要具有灵活制定控制决策的能力。

(4) 不完备性

实践表明,电梯群控系统具有明显的不完备性,最常见的不完备主要有以下几种情况:

1) 受各种未知和不确定因素的影响,如目的楼层、客流量等,系统的预测误差很大,无法精准的预测候梯时间、电梯运行时间。

2) 因客流量大小和个体差异不同使得乘客进出轿厢的时间不同,且无法获取准确的数据;

3) 每台轿厢都有额定的负载,受乘客体型、体重的限制,拥挤度和乘客数量的预测也十分不准确,系统也无法精确计算出候梯时间和电梯运行时间。

此外,电梯开关门的时间、目的楼层、运行时间等因素也是未知的,系统无法精确描述这些问题,这也增加了建立控制系统模型的难度。

(5) 多目标性

多目标性是另一个造成电梯群控系统结构复杂的重要因素^[49]。电梯群控系统的结构复杂,一方面电梯群控系统是由多个模块组成的,各模块功能既相互独立,又相互联系,必须综合考虑各模块之间的内在联系;另一方面,电梯系统具有很高的安全性、高效率的要求,这也增加了系统设计的难度。多样化目标是电梯群控系统的另一重要特征,其表现主要有以下几方面:

- 1) 尽量减小候梯时间,尽量不超过 90 秒;
- 2) 尽量减少候梯和乘梯时间,提高电梯运作效率;
- 3) 平均候梯时间尽可能小;
- 4) 电梯群能效尽可能小;
- 5) 电梯交通输送率尽可能大;
- 6) 轿箱内拥挤度适中,提高乘客舒适感。

上面几方面因素不是相互独立的,而是有紧密的内在联系,互相影响相互制约的。如何平衡各个指标和模块,提高电梯整体运作效率,是电梯群控系统的关键问题。

2.3 电梯群控的数学模型

电梯是电梯交通系统的服务台,乘客是服务对象。电梯交通系统的数学模型主要包括客流模型、电梯模型、召唤模型和调度策略模型,是一个典型的随机服务排队系统。

2.3.1 客流模型

通过理论研究和大量实践表明,泊松(Possion)分布可以很好的描述乘客到达的过程。

$$P\{\text{在时间}T\text{内到达}n\text{个乘客}\} = \frac{(\lambda T)^n e^{-\lambda T}}{n!} \quad (2.1)$$

其中, λ :乘客到达率;

T :时间长度。

则乘客到达的间隔时间 τ 服从参数为 λ 的负指数分布, 即

$$P\{\tau < t\} = \begin{cases} 1 - e^{-\lambda t}, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (2.2)$$

用以下递推公式得到乘客 1, 2, ... 的到达时间

$$\begin{cases} t_p(0) = 0 \\ t_p(i) = t_p(i-1) + \frac{\ln(r)}{\lambda} \end{cases} \quad (2.3)$$

式中, $t_p(i)$:第 i 名乘客的到达时间;

r : (0,1) 上服从均匀分布的随机数。

为了确定乘客的起点, 建立了乘客起点密度向量, 用 O 来表示; 为了确定终点楼层, 建立了终点矩, 用 OD 来表示。 O_i 表示占空比。第 i 层的人数用 A_i 来表示, 楼层数为 M , 则可以根据下面公式计算出 O_i 和 OD_{ij} :

$$\rho_i = \frac{A_i}{\sum_{j=2}^M A_j} \quad (i=2,3,\dots, M) \quad (2.4)$$

$$O_i = \sigma_1$$

$$O_i = (\sigma_2 + \sigma_3)\rho_i \quad (i=2,3,\dots, M)$$

σ_1 :起点为门厅时, 向上的乘客人数和所有乘客的比值

σ_2 :终点为门厅时, 向上的乘客人数和所有乘客的比值

σ_3 :起点与终点均非门厅时, 向上的乘客和所有乘客人数的比值

$$OD_{1j} = \begin{cases} 0, & j=1 \\ 100\rho_j, & j \neq 1 \end{cases} \quad (2.5)$$

$$OD_{i1} = \begin{cases} 0, & j=1 \\ 100\sigma_2 / (\sigma_2 + \sigma_3), & j \neq 1 \end{cases} \quad (2.6)$$

$$OD_{ij} = \begin{cases} 0, & j=1 \\ 100\sigma_3 / (\sigma_2 + \sigma_3), & j \neq 1 \end{cases} \quad (2.7)$$

$$\text{式中, } \beta_{ij} = \frac{A_j}{\sum_{k=2, k \neq j}^M A_k}$$

最后得到起点密度向量和起点-重点矩阵分别为:

$$O = (O_1, O_2, \dots, O_M) \quad (2.8)$$

$$OD = \begin{bmatrix} OD_{11} & OD_{12} & \dots & OD_{1M} \\ OD_{21} & OD_{22} & \dots & OD_{2M} \\ \dots & \dots & \dots & \dots \\ OD_{M1} & OD_{M2} & \dots & OD_{MM} \end{bmatrix}_{M \times M} \quad (2.9)$$

乘客的起点楼层和重点楼层可以通过两次蒙特卡洛抽样实验获得。在进行试验时，要以O和OD为依据。

通过上面的分析可以看出，我们已经能够知道起点楼层、终点楼层和到达时间这三个最重要的因素。

乘客到达率受客流高峰数量和时间的影 响。起点密度向量和矩阵受客流交通模式的影响和制约。各方面因素相互影响共同作用，在实际应用中要综合考虑各方面因素。

2.3.2 召唤模型

外部召唤信号是在电梯外部的召唤按钮产生的，与乘客所在的起点楼层有关；进入电梯轿厢后按下目的楼层，产生一个内部召唤。

$$P = (P_1, P_2, P_3) \quad (2.10)$$

式中， P_1 :产生召唤的时间

P_2 :起点楼层

P_3 :目的楼层

2.3.3 电梯模型

用于描述电梯配置的参数主要有以下几方面：电梯数量，用N来表示；电梯的负载和功率；电梯的运动学参数，如速度、加速度和变化率等。可以用电梯状态矩阵来描述电梯的状态，如下所示：

$$E[N][3] = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ \dots & \dots & \dots \\ e_{N1} & e_{N2} & e_{N3} \end{bmatrix} \quad (2.11)$$

式中， e_{ij} :第i台电梯所在楼层；

e_{i2} :第i台电梯轿厢内的乘客数；

$e_{i3} = 0,1,2,3,4,5,6,7$ 分别表示第i台电梯处于上行、下行、检修、超载、锁梯、暂停、消防、满载状态。

电梯轿厢内部召唤矩阵：

$$C[N][M] = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1M} \\ c_{21} & c_{22} & \dots & c_{2M} \\ \dots & \dots & \dots & \dots \\ c_{N1} & c_{N2} & \dots & c_{NM} \end{bmatrix} \quad (2.12)$$

$c_{ij} = 0,1,2$ 分别表示第i台电梯在第j层无召唤、有上行召唤、有下行召唤；M:楼层数。

2.3.4 调度策略模型

当有新的外部召唤产生时，电梯群控调度策略根据电梯群当前状态按一定的规则选择最优的调度方案分配给电梯群。通常以 F_i 表示第 i 种分配方案所付出的代价

$$F_i = F_i(E, C, H, P) \quad i = 1, 2, \dots, N \quad (2.13)$$

若 $F_k = \min(F_1, F_2, \dots, F_N)$ ，则选用第 k 种分配方案应用于该电梯群中。

电梯的运行效率和质量直接影响调度策略的应用效果。因此，群控电梯系统的关键和难点就是调度策略和合理选择。

2.4 电梯群控性能的评价指标

2.4.1 时间指标

电梯群控系统最重要的评价指标就是基于时间的评价指标^[50]。系统的性能直接受运行时间的影响。运行时间越短，运行速度越快，系统耗能就越少，系统的性能和效率就越高。可以按照下式定义时间指标：

(1) 电梯运行周期定义 (RTT)

RTT(round trip time)，电梯的运行周期，是衡量电梯运输能力的最主要的指标，是指电梯从基站出发，经过基站之间的全部路程，再次回到基站的时间总和。其可以根据下列公式计算。

$$T_p = \sum(t_{1,i} + t_{5,i}) \quad (2.14)$$

$$T_d = \sum(t_{2,i} + t_{4,i}) \quad (2.15)$$

$$T_r = \sum t_{3,i} \quad (2.16)$$

其中 $\sum t_{1,i}$ ——乘客进入轿厢的总时间；

$\sum t_{2,i}$ ——轿厢关门总时间；

$\sum t_{3,i}$ ——轿厢跑动总时间；

$\sum t_{4,i}$ ——轿厢开门总时间；

$\sum t_{5,i}$ ——乘客离开轿厢总时间。

运行周期 $RTT = T_p + T_d + T_r$ ，如图 2.4 所示，

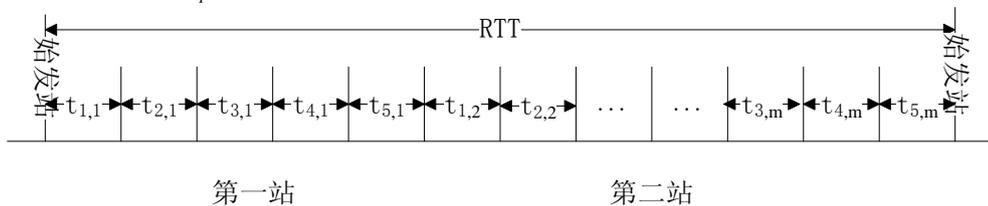


图 2.4 电梯运行周期

(2) 平均候梯时间 (T_{aw})

所有乘客的平均候梯时间 (average waiting time)

$$T_{aw} = \frac{1}{N_p} \sum_{i=1}^{N_p} T_w(i) \quad (2.17)$$

其中： N_p ——总的乘客人数；

$T_w(i)$ ——第 i 个乘客候梯所用的总时间

(3) 平均乘梯时间 (T_{aj})。

$$T_{aj} = \frac{1}{N_p} \sum_{i=1}^{N_p} T_j(i) \quad (2.18)$$

其中： $T_j(i)$ ——第 i 个乘客的总乘梯时间

(4) 平均花费时间 (T_{aa})

$$T_{aa} = T_{aw} + T_{aj} \quad (2.19)$$

(5) 最长候梯时间 (T_{mw})

$$T_{mw} = \max\{T_w(i) \mid i = 1, \dots, N_p\} \quad (2.20)$$

(6) 系统总运行时间 (T_{tr})

$$T_{tr} = T_{fs} - T_{ss} \quad (2.21)$$

其中： T_{fs} ——电梯的真正结束运行的时刻

T_{ss} ——电梯运行的初始时刻

(7) 电梯往返时间 (ARTT)

$$ATTR = \frac{\sum_{i=1}^{N_e} \sum_{j=1}^{R_j} RTT_{ij}}{\sum_{i=1}^{N_e} R_i} \quad (2.22)$$

其中： N_e ——电梯的台数

R_i ——第 i 台电梯上下运行的次数

RTT_{ij} ——第 i 台电梯运行的第 j 个周期

2.4.2 系统能耗指标

电梯的运行距离、加减速次数、负载等直接影响电梯的能耗，电梯的能耗与电梯的支出成正比^[31]。为了尽量降低电梯的耗能，提出了低碳的概念。可以根据下式评价电梯的耗能性能。

(1) 系统运行的总距离 (D_t)

$$D_t = \sum_{i=1}^{N_e} D(i) \quad (2.23)$$

其中： $D(i)$ ——第 i 台电梯运行的总距离

(2) 系统的加减速次数之和 (AD_t)

$$AD_t = \sum_{i=1}^{N_e} AD(i) \quad (2.24)$$

其中： $AD(i)$ ——第 i 台电梯加减速的次数之和

2.4.3 载客能力指标

可以通过下式评价垂直运输的效率。

(1) 5 分钟载客数 (CE_1)

假设 r_u 和 r_d 分别表示上行客流和下行客流的人数，则 5 分钟内电梯输送的乘客人数

为：

$$CE_1 = \frac{59 * 60(r_u + r_d)N}{RTT} \quad (2.25)$$

5 分钟载客率 (CE)

$$CE = \frac{CE_1}{N_p} \quad (2.26)$$

(2) 距离服务效率

$$\eta_d = \frac{CE}{D_t / D_f} \quad (2.27)$$

其中 D_f ——楼层高度。

(3) 能耗服务效率

$$\eta_e = \frac{CE}{AD_t} \quad (2.28)$$

表示单位能耗的 5 分钟载客率。

2.4.4 乘客舒适度指标

根据外召信号可以预测乘客的目的楼层。但是受未知因素的影响，如：中途新加呼梯召唤、新派任务等，实际运行时间会增加，但是预测的时候不会考虑到这一点，会影响预测结果的准确性。在实际应用中，这种事件的发生概率很高，是不容忽视的重要问题。过长的候梯时间会影响常客的心情。下列指标与乘客的心理变化有关。

(1) 平均候梯时延 (τ_{aw})

$$\tau_{aw} = \frac{1}{N_p} \sum_{i=1}^{N_p} (T_w(i) - T_{wo}(i)) \quad (2.29)$$

其中 $T_{wo}(i)$ ——第 i 个乘客预测的候梯时间。

(2) 平均乘梯时延 (τ_{aj})

$$\tau_{aj} = \frac{1}{N_p} \sum_{i=1}^{N_p} (T_j(i) - T_{jo}(i)) \quad (2.30)$$

其中 $T_{jo}(i)$ ——第 i 个乘客预测的乘梯时间。

(3) 平均时延 (τ_{aa})

$$\tau_{aa} = \tau_{aw} + \tau_{aj} \quad (2.31)$$

除了心理因素以外，受乘客体质和身体状况的不同，生理上对垂直方向运行方式也有不同的要求。不同的人承受加速度和减速度的限度是不同的。电梯在上升和下降过程中，会执行加速和减速操作。加速度的大小应该在人体承受范围之内，不会让人产生不舒适的感觉。经过大量实验表明，应该将加减速速度控制在 1.5m/s^2 以内。

2.5 电梯系统的交通模式

对高层建筑来说，其交通客流会因其功能的区别而呈现出不同的特点，甚至功能和作用完全一样的楼宇，其客流也会由于各个区域的生活规律、风土人情的区别而呈现出

很大的差异。一般来说，电梯交通流的研究和电梯系统方案、群控系统调度方法、楼宇电梯的配置及其现代化程度密切相关。尽管电梯的客流的不确定性十分明显，然而从其长期的运行状况来说，通常会表现出一定的规律。归纳来说，高层建筑的类型可以分成下述五类，即酒店、医院、办公楼、居民住宅楼、百货大楼。笔者以一办公楼为例展开研究，对办公楼来说，其电梯交通模式能够划分成以下几种：上行高峰、下行高峰、层间随机以及两路交通模式^[51]。

2.5.1 上行高峰交通

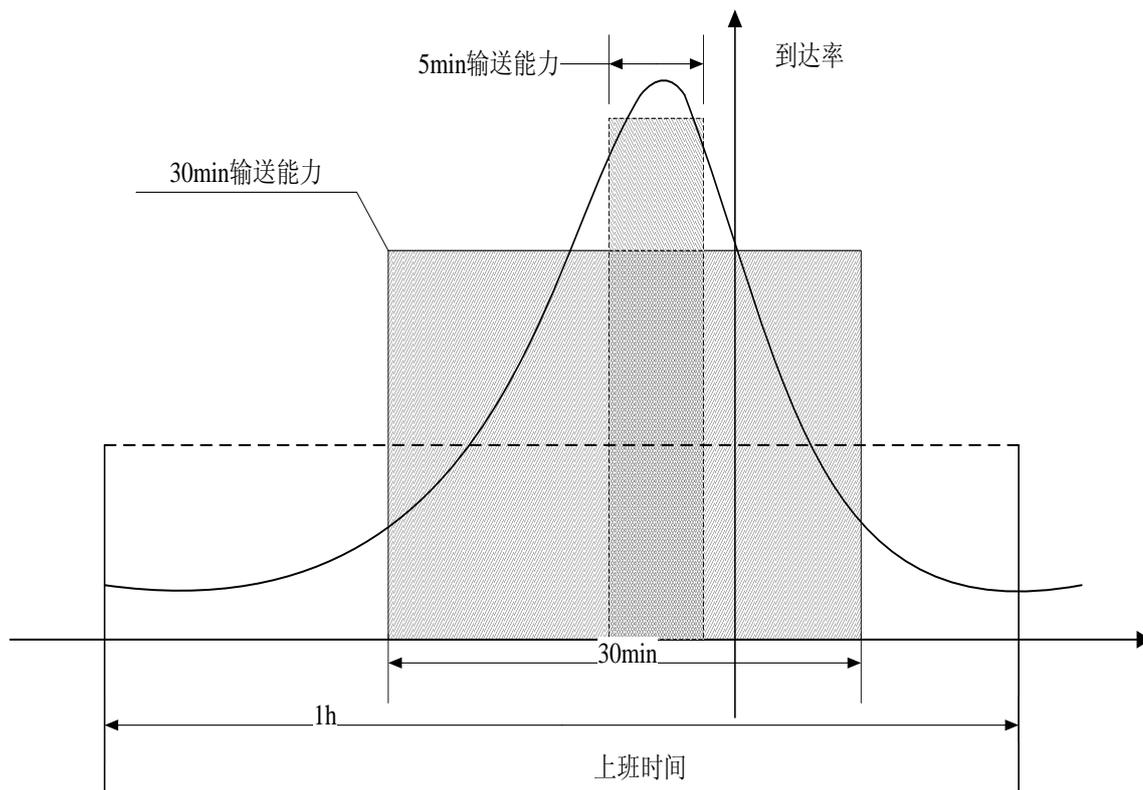


图 2.5 上行高峰乘客到达率曲线

上行高峰交通模式定义 (Up peak traffic condition): 当楼宇内主要的客流为上行时, 也就是说所有的或者大多数电梯用户均从楼宇的底层进入电梯, 其目的地在楼宇的不同楼层。

一般来说, 这种交通模式出现在上午即将开始上班的时期, 其主要集中在 8:30~9:30 之间, 在这一时期, 大多数乘客均从楼宇的底端乘坐电梯至建筑的不同楼层。形成这一现象的主要原因是公司职员一定要在公司制定的上班之前到达公司并展开一天的工作。在 1 个小时的时间内, 电梯所响应的呼梯次数即表示乘客到达率, 上图 (图 2.5) 即为从 8:30~9:30 这一时间段内的乘客到达率。由该图可以看出, 在公司制定的上班之前, 乘客到达率的曲线开始慢慢上升, 并在上班规定时刻前半小时左右出现高峰时期, 这一时期的时间仅有五分钟, 但其集中力却达到了 15% 左右 (集中率即某段时间内的候梯人数与使用电梯所有人数的比值), 一旦过了上班时间, 到达率即开始立即下

降。

从图 2.5 能够轻易的发现,除了上班前后这一段时间,建筑物并不要求电梯具有很强的运送能力,但是当公司规定时刻即将来临时要求较高。当乘客到达率较低时,电梯的运送能力远远大于乘客数,能够很好的迎合这一要求,然而,如果乘客到达率曲线比电梯的运送水平要高时,那么乘客等待电梯所需的时间便会上升,这将会导致排队等待现象的发生随着曲线的降低,当其低于 1 小时的运送水平之后,排队等待的问题将会得到缓解直至消失。由此不难看出,假如某些时刻电梯的需求更高,那么系统更无法迎合这一需求,唯一的选择就是多安装几部电梯,然而这却会使投入增加。因此一定要认真考虑利弊,接受合适的候梯时间。

结合 2.4.3 的相关定义, CE_t 与时间是紧密相连的,其值的大小会随着时间的变化而改变。众所周知,把客流到达率最大的 5 分钟的 CE 叫做 5 分钟乘客集中率 CE_a 。由此可以看出,只有当 $CE \geq CE_a$ 成立时方可确保乘客候梯时间的均值在一个合理的范围。

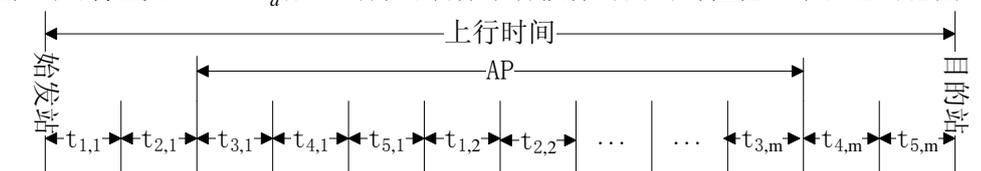


图 2.6 平均行程时间

表 2.1 用途不同的建筑在上行高峰期对电梯系统的服务水平

建筑物用途		5min 载客率 CE (%)	平均运行间隔 AI (s)	平均行程时间 AP (s)	
办公楼	公司专用楼	20-25	30s 以下为良好 30-40s 为较好 40s 以下为不良	60s 以下为良好 60-75s 为较好 75-90s 为较差 120s 为极限 住宅、医院和百货商店 可稍微长些	
	准专用楼	16-20			
	机关办公楼	14-18			
	分区出租办公楼	12-14			
	分层出租办公楼	14-16			
住宅楼		3.5-3.0	60~90s		
旅馆		10-15	30s 以下为良好 30-40s 为较好 40s 以下为不良		
医院	大型	人的交通	20		<60s
		车的交通	2		
	中小型	人的交通	20		<120s
		车的交通	2		
百货商店		16-18	60-90s		

在这种交通模式下,平均时隙时间 AI 以及行程时间均值 (AP) 均为体现电梯服务质量的重要参数。其中 AI 即二部相邻电梯到达基站的时间差的统计平均值。 AP 则为电梯轿厢从关门启动至抵达目的层站所用时间的均值,可用上图表示(图 2.6)。那么 $AP =$

上行时间 $-(t_{1,1} + t_{2,1} + t_{4,m} + t_{5,m})$.

用途不同的建筑在这一阶段对电梯系统的服务质量的评价可以归纳如上, 即表 2.1。

2.5.2 下行高峰交通

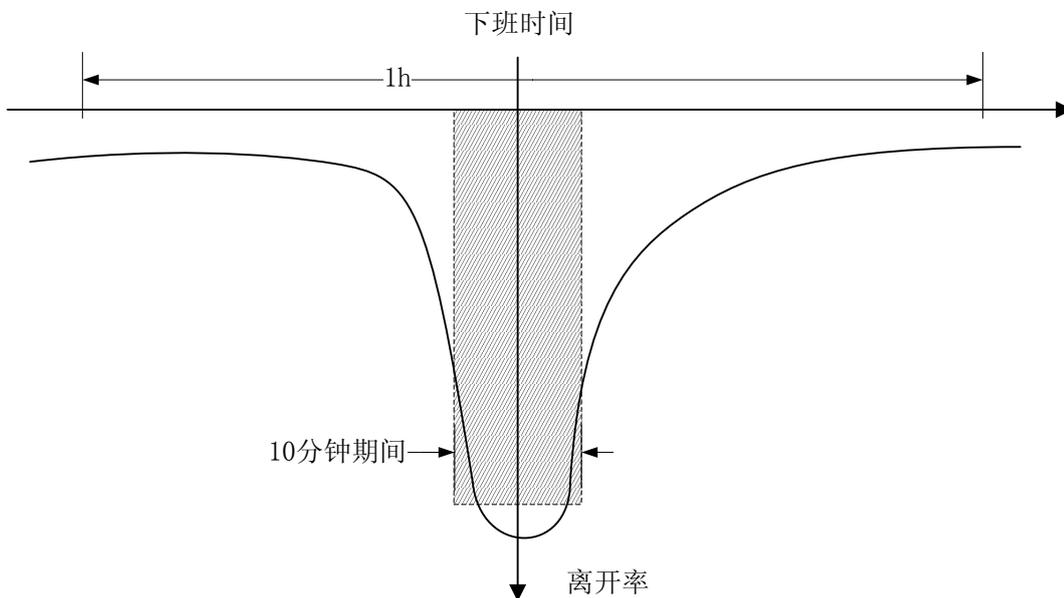


图 2.7 下行高峰乘客离开率曲线

这一交通模式就是当楼宇内主要的客流是下行方向, 即全部或大部分乘客从建筑物的内部各楼层进入电梯, 下行至大楼的门厅。

下行高峰交通模式通常发生下午下班的时间, 其和早高峰时一一对应的, 然而强度却要大很多, 持续的时间也要场很多。图 2.7 即为这一时期的乘客离开率。如果下行高峰强度过高, 因为乘客较多, 电梯在停靠一站或者二站之后就有可能满载, 所以为了让乘客的平均等待时间不宜过长, 必须合理的调度电梯群去迎接不同层的乘客。

2.5.3 两路交通

这一交通模式就是当建筑物里面乘客的流向是以某一特定楼层为起点层或目的层并且该层并非门厅。例如中午休息以及就餐的时间, 某幢建筑的餐厅设在 3 楼, 很多乘客便要乘梯前往该层就餐, 且就餐结束后离开该层。当两路交通模式发生时, 一定要使电梯群控系统针对某些楼层展开记忆和学习, 以此提高这些客流输送水平, 提高此类呼梯的运行效率和服务质量。

2.5.4 层间随机交通

这种交通模式主要出现在两段时间: 早上高峰后至中午下班之前、中午上班之后至下午高峰前。当层间交通模式出现时, 电梯群控系统必须有科学的停层方法, 以此来降低因电梯空跑而造成的能耗, 例如可将每部电梯均匀停靠在建筑物的不同楼层, 保障随机层的乘客尽快得到服务。

2.6 小结

虽然不同电梯生产商研制的电梯群控系统组成形式和功能特点各不相同，但在基本结构上是相似的。本章首先介绍了电梯群控系统的构成，然后针对其动态特性展开了重点介绍，比如模糊性、不确定性、非线性以及多目标性等，最后针对群控系统的评价指标进行了归纳。本章的内容主要是为后面的章节打下理论基础。

第3章 基于人工鱼群算法的群控调度策略设计及实现

3.1 人工鱼群算法的研究现状及进展

在1999年,我国女学者涂晓媛等人利用鱼的形态、习性和行为模型成功地创作了“人工鱼”,将“人工鱼”的基本特征——生物力学、运动、感知和行为引入到计算机动画的创作中,被学术界称为“Xiaoyuan' Fish”^[52]。2002年华人学者李晓磊等人提出了一种模拟鱼群觅食的仿生类算法,即人工鱼群算法(Artificial fish school algorithm, AFSA),AFSA是集群智能思想的一个具体的应用。它通过模仿构造鱼的基本行为,先让人工鱼单独执行觅食,再通过鱼与鱼之间的合作行为,最终找到全局最优解^[53],归纳起来该算法具有以下一些特点:

- (1) 赋予生命力的鱼群算法不易陷入局部极值;
- (2) 针对外部环境变动引起的极值点漂移问题,鱼群算法具备跟踪变化的能力;
- (3) 当求解的精度要求不高时,该算法能快速的得到一个相对最优解;
- (4) 算法的数学基础相对薄弱,且存在许多不完善和未涉及到的问题。

由于AFSA的良好性能,对它的研究应用已经渗透到多个应用领域,以下是近几年来人工鱼群算法的主要应用情况:

2003年李晓磊^[54]等人针对组合优化问题,提出了人工鱼群算法应用于离散事件的距离、中心和领域等概念,并给出了相应的描述。

2005年兰州大学信息科学与工程学的院俞洋,孔洁,田亚菲^[55]等人将人工鱼群算法用于用于低通、带通IIR数字滤波器的设计,并与粒子群算法进行了比较,仿真结果证明了AFSA的有效性,并且具有算法灵活、简单,全局收敛性好,收敛速度快的优点。

2007年北京交通大学的高德芳,赵勇,郭杨,赵海涛^[56]等人以用户需求与产品性能的相关度最大为优化目标建立了模块化产品配置设计模型;并提出了融合鱼群算法及蚁群算法的优化模型求解方法。以铁路客车内装模块化配置设计为应用,验证了该优化模型及其求解方法的可行性和有效性。

2008年南京理工大学电力系统及其自动化学院的袁远^[57]等人应用改进的人工鱼群算法对配电网进行无功优化,并以某城市实际的高压配电网和低压配电网进行了计算,结果表明改进算法,具有求解精度高,收敛速度快,能够全面地搜索解空间以达到最优解等优点。

2009年西安电子科技大学的杨丽^[58]等人提出了基于互信息和人工鱼群算法的图像准方配准方法,其主要思想是利用人工鱼群算法进行优化搜索,并采用归一化互信息作为配准相似性准则,实验结果表明,该算法具有高精度和高可靠性,且速度快,能有效

地进行图像配准。

2011年中南大学的刘李钊^[59]等人提出了将克隆遗传算法(CGA)与鱼群算法(AFSA)相结合,形成一种新型的智能优化算法——鱼群克隆遗传算法(AFCGA),其主要思想是利用克隆遗传算法的全局性和鱼群算法的方向性和快速性的特点,保证在全局内快速的找到最优解,结果表明,AFCGA应用于配电网重构不依赖于网络的初始条件,CGA后期运行速度慢的缺点也得到了有效克服。AFCGA有效地改善算法收敛性能,与CGA相比具有较高的搜索效率和较快的运行速度。

2011年哈尔滨工程大学的王芳,万磊,徐玉如,张玉奎等人^[60]将人工势场和人工鱼群算法相结合,应用于航班路径规划问题中。建立了引力和斥力的人工势场模型,用于单条改航路径规划;构造了单个人工鱼和适应度函数,采用人工鱼群算法优化改航路径;通过对改航路径的评估和修正,获得满足要求的改航路径。算例仿真结果显示,与现行的改航相比,基于人工势场-人工鱼群算法生成的改航路径总长度缩短了7.6%,表明算法是可行的和有效的。

2012年华北电力大学的杨淑露,韩奇,徐琳茜^[61]等人从污染物减排率、单位工业增加值减排量、治理工业污染投资总额、GDP相关指标、能耗下降率5个方面建立节能减排效果评价指标体系,分析BP神经网络与鱼群算法结合的可行性,对7个地区2006~2009年节能减排效果评价指标,运用神经网络及鱼群算法优化神经网络方法进行节能减排效果评价。研究表明:在收敛过程中,运用神经网络所得实际输出值与专家评分的误差长时间停留在0.7左右,而运用鱼群算法优化神经网络方法能够以较大的斜率快速收敛到期望误差;在误差为0.001时,前者经过202次训练后能够达到目标,而后者只需要75次训练就能达到目标,这表明鱼群算法优化神经网络具有准确、快捷、简易等优点。

2012年陕西师范大学的丁生荣,马苗,郭敏^[62]等人利用人工鱼群算法和二维Otsu阈值法,提出了一种自适应的图像对比度增强算法。该方法利用优化后的人工鱼群算法自动选取双阈值,依据图像的对比度自动搜索灰度变换斜率,得到最优的分段线性变换曲线,并用之对图像进行增强处理。实验表明,该方法可有效提高图像对比度,且优于直方图均衡化、反锐化掩模等传统的对比度增强方法。

这是近几年国内学者应用人工鱼群算法解决实际问题的一些研究,相信通过不断的改进和完善,人工鱼群算法将有更良好的应用前景。

3.2 与其他算法的比较

对比其它仿生算法,人工鱼群算法具有异曲同工之妙。与二进制编码遗传算法进行对比,它们之间的共同点是,首先均把单个个体描述为一种解决方案,再使各个智能个体并行运作,通过交互各自的信息,从而达到更好适应环境的目的。其次,人工鱼群算法中,聚群行为的寻找伙伴中心位置与遗传算法的算术交叉算子类似。在遗传算法中,交叉变换通过替换、重组两个父代个体的部分结构,从而产生新的两个子代个体;而在

鱼群算法中，人工鱼个体通过向视觉范围内的某个点移动，来更改当前人工鱼的状态。最后，人工鱼的随机移动行为可看做遗传算法中的变异变换，其变异概率由尝试次数 `try_number` 决定。人工鱼群算法中的聚群行为、追尾行为体现了与其他智能算法的共性，即适宜性：在变化着的环境中，通过个体的不断自学习，从而提高整体的适应能力。

与其他智能算法不同，人工鱼通过判别当前所处环境下的食物浓度来调整自身的状态，而并非产生新的人工鱼对其自身进行替换；在自适应的迭代过程中，人工鱼群算法不仅保留和利用当前位置，而且保留和利用周边环境的信息；人工鱼群算法采用自上而下的设计方法，使得人工鱼个体之间具有独立性和互补性，这使得算法具有相对较稳定的收敛性；且具有生命力的鱼群算法，可以轻易的跳出局部极值；这些特性使得与其他算法相比，人工鱼群算法更具优势。

3.3 人工鱼群算法描述

人工鱼群算法是李晓磊等人于 2002 年在对动物群体智能行为研究的基础上提出的一种新型仿生优化算法，该算法根据“水域中鱼生存数目最多的地方一般就是本水域中富含营养物质最多的地方”这一特点来模仿鱼群的觅食行为而实现寻优。人工鱼群算法采用自上而下的寻优模式，从构造个体的底层行为开始，通过鱼群的觅食、聚群及追尾行为改变自身的位置，经过一段时间各人工鱼移动到局部极值点，最终从人工鱼局部寻优中找到全局最优值^[44]。

3.3.1 算法原理

人工鱼群算法就是一种基于动物行为的自治体寻优模式，它是基于鱼类的活动特点构建起来的新型智能仿生算法。

每个人工鱼个体可以用面向对象的方法来构造，即把人工鱼的状态、属性和行为封装为一个对象，这样也便于工程实现。如下所示用 C++ 语言的伪代码表示：

```
class Artificial_fish
{
    float AF_x[n]; //AF的状态
    float AF_step; //AF的移动步长
    float AF_visual; //AF 的感知范围
    float AF_food consistence(); //AF当前状态的食物浓度值
    void AF_move(); //AF的移动行为
    float AF_follow(); //AF的追尾行为
    float AF_preay(); //AF的觅食行为
    float AF_swarm(); //AF的聚群行为
    int AF_evaluate(); //AF行为的自适应评价
    void AF_init(); //AF的初始化
};
```

通常人们可以观察到如下的鱼类行为：

(1) 觅食行为：即鱼通过嗅觉、视觉来感知水中的食物浓度，当发现某一方向的食物浓度比当前所处位置的食物浓度高时，就往这个方向游动，这是鱼趋向食物的最基本行为。

(2) 聚群行为：在自然进化过程中，为了躲避危险、寻找食物，随机离散的鱼会聚集到一起，从而形成这一聚集行为。鱼聚群时所遵守的规则有三条：

- 1) 内聚规则：尽量向视觉范围内伙伴的中心移动；
- 2) 对准规则：尽量与视觉范围内伙伴的平均方向一致；
- 3) 分隔规则：尽量避免与视觉范围内的伙伴过于拥挤。

(3) 追尾行为：当鱼发现某一食物时，会加速向其游动，它周边的鱼会根据其摆尾速度加快而追赶过来，进而更远处的鱼也会向食物处聚集。

(4) 随机行为：在未找到食物之前，为了寻觅食物或临近伙伴，鱼在水中的游动是随机的。随机行为实际上是觅食行为的一种缺省。

觅食行为主要是鱼趋向食物多的方向游动，在寻优算法中则是向较优方向前进的迭代方式。聚群行为能够很好地跳出局部极值，并尽可能搜索到其它的极值，最终搜索到全局极值。追尾行为有助于快速的向临近的最活跃者追逐，既能加快寻优速度，也能防止人工鱼陷入局部极值点。鱼群算法对以的三种行为进行评价后，选取最优的行为采取行动，从而形成一种高效快速的寻优策略。

觅食、聚群、追尾和随机行为与寻优问题的解决有着较密切的联系，为了能够让算法更好的实施，算法的详细描述和实现如下。

3.3.2 行为描述

在构建人工鱼模型之前，先介绍一下相关的一些定义：

- (1) 向量 $X_i = (x_1, x_2, \dots, x_n)$ 表示人工鱼个体的状态；
- (2) X_j 表示感知的下一个状态；
- (3) $Y = f(x)$ 表示人工鱼的食物浓度函数，其中 Y 为实物量；
- (4) $\delta (0 < \delta < 1)$ 表示拥挤度因子；
- (5) $Visual$ 表示人工鱼的感知距离；
- (6) $step$ 表示人工鱼移动的步长；
- (7) $d_{ij} = \|X_i - X_j\|$ 表示两条人工鱼个体之间的距离。

以优化问题为例，以下是人工鱼群算法基本行为的描述：

(1) 觅食行为：设人工鱼当前状态为 X_i ，在其视野范围内随机选择一个状态 X_j ，如果 $Y_i > Y_j$ ，则朝令 X_j 的方向进行游动；反之，再重新随机选择状态 X_j ，判断是否满足前进条件；试探 $trynumber$ 次后，如果仍不满足前进条件，则执行随机移动行为。

(2) 聚群行为：设人工鱼当前状态为 X_i ，探索其邻域的伙伴数目 n_f ，如果

$n_f < \delta, (0 < \delta < 1)$ ，则表明伙伴中心有较多的食物并且不太拥挤，如果此时 $Y_i > Y_c$ ，则人工鱼向中心位置前进一步；否则执行随机行为。

(3) 追尾行为：设人工鱼当前状态为 X_i ，探索其邻域内状态最优的邻居 X_{\min} ，如果 $Y_i > Y_{\min}$ ，并且 X_{\min} 的邻域内伙伴的数目满足 $n_f < \delta, (0 < \delta < 1)$ ，表明 X_{\min} 附近有较多的食物并且不太拥挤，则向 X_{\min} 的位置前进一步；否则执行随机行为。

(4) 随机行为：随机行为是觅食行为的一个缺省行为，它就是鱼在视野内随机选取一个状态，然后朝该方向移动。

根据目标函数，人工鱼群算法评价所处位置不同行为的食物浓度值，并针对不同问题，选择合适的行为采取行动。通常情况下，算法比较觅食、聚群、追尾行为的食物浓度值，选取优于当前食物浓度值的最优行为采取行动，若这三者行为的食物浓度值均不优于当前的浓度值，则采取随机移动行为。

3.3.3 算法描述

基于以上描述的人工鱼行为，每条人工鱼根据其自身所处的环境情况和它临近伙伴的情况，依照行为选择机制，选择最适合的行为采取行动，最终，人工鱼会集结在几个局部极值的周围。通常情况下，在讨论求极小问题时，具有较小食物浓度值的人工鱼会处于值较小的极值区域周围，这有助于获取全局极值域，而值较小的极值区域周围一般会集结较多的人工鱼，这有助于判断并获取全局极值，极大值问题亦然。

以下是人工鱼群算法的伪代码描述，算法根据特定的问题设置相应的参数，然后随机生成初始种群，接着每条人工鱼按照设定的行为进行寻优，一般直到算法达到规定的迭代次数或者连续所得值的均方差小于设定值。

```

Artificial_Fishschool_Algorithm()
{
    ::AF_init();
    while the result is not satisfied do
    {
        switch(::AF_evaluate())
        case value1:
            ::AF_follow();
        case value2:
            ::AF_swarm();
        default:
            ::AF_preay();
    }
    ::AF_move();
    get_result();
}

```

3.4 基于鱼群算法的电梯群控调度策略设计

群控系统管理 N 部电梯，有 p 个呼梯信号，则有 N^p 种派梯方案。针对这一具体问题，则要求人工鱼群算法在有限的时间内，要找出相对最优的一种派梯方案。所以在人工鱼建模时，要对其进行合理的设计，包括人工鱼个体的设计，即对电梯调度方案编码规则的设计；人工鱼食物浓度函数的设计，即电梯群控控制目标的设计；人工鱼状态受电梯轿厢内人数约束条件的设计；人工鱼觅食、聚群、追尾、随机行为的设计；人工鱼之间距离的计算方法，以及参数的选定。

3.4.1 人工鱼个体设计

每条人工鱼的状态描述运用一个二维数组 $x[2][2M-1]$ 表示，其中 M 为楼层数。 $x[0][1,2,..M-1]$ ， i 表示第 i 层的上行召唤。 $x[0][M,M+1,..2M-2]$ ， i 表示第 $i-M+2$ 层的下行召唤。 $x[0][i]$ 的取值范围为1或0，1表示该楼层有外部召唤，0表示无召唤， $x[1][i]$ 的取值范围 $[0,N]$ ，0表示不派梯， N 表示参与群控电梯数， $x[1][i]=j$ 表示第 i 个外部召唤由 j 号电梯响应(注：楼层数为16，第1层无下行召唤，第16层无上行召唤， $x[0][0],x[1][0]$ 舍弃不用)。

例如，数组 $x[2][2M-1]$ (设 $M=16,N=4$ ，当前外部上行召唤有5、8、9层，外部下行召唤有2、6、8、15、16层) 为

```
0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 1 1
0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 2 0 0 0 1 0 1 0 0 0 0 0 0 3 3
```

该数组表示的含义为：

- (1) 1号电梯响应第5、8、9层的上行召唤，同时响应第6、8层的下行召唤；
- (2) 2号电梯响应第2层的下行召唤；
- (3) 3号电梯响应第15、16层的下行召唤；
- (4) 4号电梯空闲；

本文对每条鱼的初始化采用如下策略：依次给 $x[0][i]=1$ 的 $x[1][i]$ 赋值，其值为在 $[1,N]$ 之间随机产生一个整数 j 。

3.4.2 食物浓度的设计

本算法以平均候梯时间、平均乘梯时间以及平均能耗作为电梯调度的主控指标。

(1) 平均候梯时间

以1号电梯为研究对象，对于1号电梯服务队列，从当前楼层到每个呼叫楼层（比如第 i 个呼叫楼层），我们总能找到四个峰值分别极为 $P1$ 、 $P2$ 、 $P3$ 和 $P4$ 。

当电梯处于停止或上升状态时，如图3.1(a)

$$T_{i1} = ((P2 - P1) + (P2 - P3) + (P4 - P3)) \times t + NS_i \times TS, i \in (1, n_1) \quad (3.1)$$

当电梯处于下降状态时，如图3.1(b)

$$T_{i2} = ((P1 - P2) + (P3 - P2) + (P3 - P4)) \times t + NS_i \times TS, i \in (1, n_1) \quad (3.2)$$

其中， t 表示电梯运行一层需要的时间；P1指目前电梯所在楼层；P2指在运行方向上的最顶层；P3指在运行反方向上的最顶层；P4是指呼叫楼层，NSi 指当前楼层到第i个呼叫楼层间的停靠次数，TS指平均停靠时间， n_i 为1号电梯服务队列的长度。

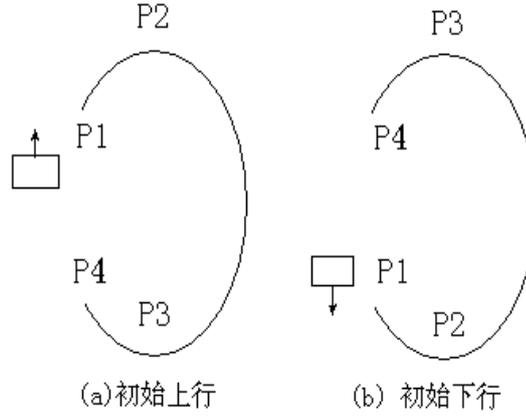


图3.1 单梯响应某服务的运行路线

由上面所得可知平均候梯时间：

$$AWT = \sum_{j=1}^N \sum_{i=1}^{n_i} T_{ji} / \sum_{i=1}^N n_i \quad (3.3)$$

其中， N 表示参与群控电梯数， n_i 表示i号电梯的响应服务的个数。

(2) 平均乘梯时间

通过上述求平均等待时间的计算方法来计算当前已产生的内部召唤呼梯信号，即可得出平均乘梯时间。

(3) 平均能耗

由于电梯匀速运行时的能耗远远小于加减速时的能耗，所以可以通过统计电梯启停的次数来估算能耗。

$$F_k = \omega_1 \times AWT + \omega_2 \times AJT + \omega_3 \times EW \quad (3.4)$$

其中 AWT 为平均等待时间， AJT 为平均乘梯时间， EW 为电梯能耗，

$\omega_1 + \omega_2 + \omega_3 = 1$, $\omega_1, \omega_2, \omega_3$ 分别为 AWT, AJT, EW 的加权系数，该系数由不同的交通模式确定。比如，在随机层间在随机层间交通模式下，我们希望顾客的候梯时间和乘梯时间小一些，则 ω_1, ω_2 应该偏大一些。问题到此就由求解 $\min(AWT, AJT, EW)$ 转化为求解 $\min(F_k)$ 。

3.4.3 人工鱼的行为设计

(1) 感知能力设计

对于人工鱼之间距离的计算，本文通过对比两条人工鱼的存储值得出，即对于任意两条人工鱼，将两者 $x[1][i]$ 中所有的值进行比较，相同取0，不相同取2，最后将所有比较后得出的值进行累加，即为两者之间的距离 d 。从人工鱼个体的设计可以看出， $x[1][i]$ 数组存储的值代表的是响应电梯的编号，如果运用通常的加减法或者欧几里得距离公式

算出的距离，非但没有实际意义，不能解决问题，反而会对人工鱼的视觉造成混乱。采用本文设计的相似度法，简单易行，即通过计算两者的相似度，来估计两者之间的距离，如果距离值越大则表明两者的相似度越小，当值大于给定的visual值时，则表示两条人工鱼之间是相互不可见的。对于有9个外部召唤的任意两条人工鱼，它们之间最大的距离为18，即表示两个派梯方案完全不相似。

(2) 觅食行为

从电梯群控调度问题可以看出，每条人工鱼的状态 $x[1][2M-1]$ 都是一种群控调度方案，其解空间是离散的，而非连续，即数组中的每一个值所代表的是响应电梯的编号，因此对于此离散问题，步长的设计可以省略。如果非要生搬硬套的像解决连续性问题那样设置步长，则会导致人工鱼寻优的混乱。所以，当人工鱼向前行动时，采取直接跳转到下一个更优的解决方案，既合理，同时算法收敛性效果也好。

设人工鱼当前状态为 X_i ，在其视野范围内随机选择一个状态 X_j ，如果 $Y_i > Y_j$ ，则令 $X_i = X_j$ ；反之，再重新随机选择状态 X_j ，判断是否满足前进条件；试探trynumber次后，如果仍不满足前进条件，则执行随机行为。

伪代码描述如下：

```
float Artificial_fish::AF_preay()
{
    for( i=0; i < try_number; i++)
    {
         $X_j = \text{Random}(N(X_i, \text{Visual}));$ 
        if(  $Y_i > Y_j$  )
        {
             $X_i = X_j$ 
            return AF_foodconsistence(  $X_i$  );
        }
         $X_i = X_j$ 
        return AF_fooddconsistence(  $X_i$  );
    }
}
```

$X_j = \text{Random}(N(X_i, \text{Visual}))$ 表示在 $[\text{Btnum} - \text{Visual}/2, \text{Btnum}]$ (Btnum 表示当前需响应的外部召唤个数) 之间随机产生一个整数 j ，在 $x[0][i]=1$ 的 $x[1][i]$ 中，选取 j 个点插入与 X_i 相同的位置，其余 $x[0][i]=1$ ， $x[1][i]$ 的位置随机选择分配。

(3) 聚群行为

设人工鱼当前状态为 X_i ，探索其邻域的伙伴数目 n_f ，如果 $n_f / N < \delta, (0 < \delta < 1)$ ，则表明伙伴中心有较多的食物并且不太拥挤，如果此时 $Y_i > Y_c$ ，则人工鱼向中心位置前进一步；否则执行随机行为。

伪代码描述如下:

```
float Artificial_fish::AF_swarm()
{
     $n_f = |N(X_i, Visual)|$ ;
     $X_c = \text{Center}(N(X_i, Visual))$ 
    if ( $n_f / N < \delta \ \&\& \ Y_i > Y_c$ )
         $X_i = X_c$ 
    else
         $X_i = \text{Random}(N(X_i, Visual))$ ;
    return AF_foodconsistence( $X_i$ );
}
```

$\text{Center}(N(X_i, Visual))$ 表示中心位置, 首先统计每个 $x[1][i]$ 位置上 $0 \sim 4$ 出现的次数, 其次将出现次数最多的数字填入 X_j , 这样, 赋值就尽量找到了伙伴的共同点。

(4) 追尾行为

设人工鱼当前状态为 X_i , 探索其邻域内状态最优的邻居 X_{\min} , 如果 $Y_i > Y_{\min}$, 并且 X_{\min} 的邻域内伙伴的数目满足 $n_f / N < \delta, (0 < \delta < 1)$, 表明 X_{\min} 附近有较多的食物并且不太拥挤, 则向 X_{\min} 的位置前进一步; 否则执行随机行为。

伪代码描述如下:

```
float Artificial_fish::AF_follow()
{
     $Y_{\min} = \text{Min}(f(X_{\min}), X_{\min} \in N(N_i, Visual))$ ;
     $n_f = |N(X_i, Visual)|$ ;
    if ( $n_f / N < \delta \ \&\& \ Y_i > Y_{\min}$ )
         $X_i = X_{\min}$ 
    else
         $X_i = \text{Random}(N(X_i, Visual))$ ;
    return AF_foodconsistence( $X_i$ );
}
```

(5) 生存机制

在大自然中, 鱼的时间寿命是有限的。一般情况下, 在食物多的地方, 鱼的生命力较长; 而食物稀少的地方, 鱼的生命力较短。根据这一特性, 在设计人工鱼时, 赋予它一定的生存力, 即使人工鱼在全局最优解附近生命力最长, 在局部极值点附近, 人工鱼会随着时间的推移而消亡。为了提高算法存储空间的使用率, 在鱼死亡之后, 随机产生一条新的人工鱼, 即可跳出局部极值。

生存机制的描述如下:

$$h = \frac{E}{\lambda T}; \begin{cases} h \geq 1, :: \text{AF_move} \\ h < 1, :: \text{AF_init} \end{cases} \quad (3.5)$$

其中, h 为生存指数, E 为AF当前所处位置的食物浓度值; T 为AF的生存周期, λ 为消耗因子, 即单位时间内消耗的能力值。

(6) 竞争机制

为了实时调整人工鱼的生存周期，引入竞争机制，其描述如下：

$$T = \varepsilon \frac{E_{\max}}{\lambda} \quad (3.6)$$

其中， E_{\max} 为当前所有人工鱼中食物浓度的最大值， ε 为比例系数。

3.4.4 电梯轿厢内人数

大多数群控系统在派梯过程中没有考虑各台电梯轿厢的人数，这种情况往往会导致某个电梯十分拥挤，而其他电梯的利用率不高。人工鱼根据其觅食行为、聚群行为和追尾行为进行全局寻优，当人工鱼向某一方向游动时，考虑各台电梯轿厢的人数和已有的内部召唤，可以使算法跳过一些不合理的解，从而搜索到更优的解。例如，第7层有外部上行召唤，1号电梯所在楼层为第5层，方向为上行，最近的内部召唤为第9层，当前轿厢内人数已到达额定人数18；2号电梯所在楼层为第3层，方向为上行，最近的内部召唤为第10层，当前轿厢内人数为7，未达到额定人数18；3、4号电梯都停在第1层。根据最小等待时间，就近原则，如不考虑电梯轿厢人数，将派1号电梯响应第7楼的外部上行召唤，但考虑轿厢人数之后，则派2号电梯响应该楼层外部上行召唤。

如表3.1所示，引入轿厢内人数因素的改进型鱼群算法在平均等待时间、长时间候梯率上明显优于不引入轿厢内人数因素的基本鱼群算法，深究其原因是因为引入轿厢内人数这一因素可以有效节省无效的停开门时间及其导致的加减速时间。

表 3.1 基本型与改进型对比

群控算法	基本的鱼群算法	引入轿箱内人数因素的鱼群算法
AWT (s)	27.73	25.39
MaxWT(s)	186	170
<20s 等待时间人数比例 (%)	48	49
20~40s 等待人数比例 (%)	31	33
40~60s 等待人数比例 (%)	12	12
60~90s 等待人数比例 (%)	6	5
>90s 等待人数比例 (%)	3	1

引入轿箱内人数的关键代码如下：

```
void Artificial_Fish::allocate(int i,int tmp[])//tmp放置具体方案，i表示第i层的方案需要分配
{
    bool rflag=false;
    int trynum;
    int iflag[M][M]={0};//用矩阵形式存放电梯分配情况，1表示不可分配，该梯满员
    retry:
        tmp[i]=(int)(rand()%(Myelenum-1)+1);//随机分配范围[1,4]
        //添加人数因素，判断是否符合
        if(i<My layernum)
        {
            //上行；轿厢内人数；轿厢的in；轿厢所在楼层，iflag[i][j]可否为第i个召唤分配电梯j；tmp分配
```

方案

```

    rflag=judge(true,i,ele[tmp[i]].People,ele[tmp[i]].BtnIn,ele[tmp[i]].Layer,iflag,tmp[i]);
    //ture:不可分配; false: 可分配
    trynum=0;
    for(int j=1;j<Myelenum;j++)
    {
        if(iflag[i][j])//在judge中, 不能分配此i处的电梯置为1
            trynum++;
    }
    if(trynum==(Myelenum))//所有电梯人数都满,不适合分配
    {
        rflag=false;
        tmp[i]=0;
    }
    if(rflag)//ture:满员, 不可分配, 则需要从新分配电梯
        goto retry;
    }
}

```

3.4.5 参数分析

目前还没有一般的解析方法能够对离散人工鱼群算法的trynumber、visual、number、maxgen等重要参数进行确定, 所以只能通过先保持其他参数不动, 然后再对每一参数进行调整来观测其对算法性能的影响。

(1) trynumber对算法性能的影响

表 3.2 trynumber 对算法性能的影响

		Sample (Visual=30, Option=24, Number=25, maxgen=100)						
		1	2	3	4	5	Avg	min
Trynumber=10	Time	0.573	0.552	0.556	0.599	0.743	0.6046	0.552
	Foodconsistence	222.4	220.9	221.2	222.1	220.2	221.36	220.2
Trynumber=20	Time	0.773	0.752	0.756	0.699	0.743	0.7446	0.699
	Foodconsistence	220.5	219.0	219.1	220.2	218.7	219.5	218.2
Trynumber=30	Time	0.871	0.889	0.893	0.868	0.901	0.8844	0.868
	Foodconsistence	215.1	214.9	213.9	214.2	214.5	214.52	214.9
Trynumber=40	Time	0.998	1.032	1.027	1.101	1.122	1.056	0.998
	Foodconsistence	214.8	214.6	214.4	214.5	215.4	214.74	214.4

由表3.2, 可知, 随着trynumber数目的增加, 虽然在求解的准确性和稳定性方面有不同的提高, 但与此同时所花费的时间也大大增加, 考虑电梯群控实时性的要求, 必须在0.6秒内的响应时间找到分配方案, 所以此处trynumber选10较为合适。

(2) visual对算法性能的影响

表 3.3 visual 对算法性能的影响

Sample (Trynumber =10, Option=24, Number=25, maxgen=100)		1	2	3	4	5	Avg	Min
Visual=22	Time	0.563	0.542	0.562	0.556	0.564	0.5574	0.542
	foodconsistence	250.4	250.5	251.2	252.1	248.4	250.52	248.4
Visual=24	Time	0.564	0.551	0.552	0.564	0.554	0.557	0.551
	foodconsistence	240.4	240.5	241.2	242.1	242.4	241.32	240.4
Visual=30	Time	0.573	0.552	0.556	0.599	0.743	0.6046	0.552
	foodconsistence	222.4	220.9	221.2	222.1	220.2	221.36	220.2
Visual=36	Time	0.881	0.899	0.873	0.868	0.921	0.8884	0.868
	foodconsistence	214.1	215.9	214.1	213.2	213.5	214.16	213.2
Visual=40	Time	0.998	1.032	1.027	1.101	1.122	1.056	0.998
	foodconsistence	214.8	214.6	214.4	214.5	215.4	214.74	214.4
Visual=44	Time	1.008	1.08	1.027	1.148	1.132	1.079	1.08
	foodconsistence	213.8	213.6	213.4	214.5	215.4	214.14	213.6

由表3.3, 可知, Visual的选取关系到鱼的视觉范围能力, 如果选择大了, 会因为增大了搜索范围, 导致算法收敛时间增加; 反之, 选择过小, 易使人工鱼陷入局部极值点, 最终导致算法无法收敛于全局最优, 因此, 选择合适的Visual很重要, 从本例子来看, visual=30最为适宜。

(3) 样本数目 (number) 对算法性能的影响

从表3.4 可以看出, 对于24个外部召唤的情况, 取number=25最为适宜, 如果进一步增加样本数目, 虽然会找到更优的解, 但同时也会增加收敛时间, 不符合电梯群控的实时性要求。

表3.4 样本数目 (number) 对算法性能的影响

		Sample (Visual=30, Option=24, Trynumber=10, maxgen=100)						
		1	2	3	4	5	Avg	Min
Number=20	Time	0.453	0.422	0.412	0.444	0.464	0.439	0.412
	foodconsistence	250.4	252.5	259.2	256.1	255.4	254.72	250.4
Number=25	Time	0.573	0.552	0.556	0.599	0.743	0.6046	0.552
	foodconsistence	222.4	220.9	221.2	222.1	220.2	221.36	220.9
Number=30	Time	0.851	0.879	0.873	0.858	0.881	0.8684	0.851
	foodconsistence	210.1	210.9	211.9	213.2	212.5	211.72	210.1
Number=35	Time	1.308	1.232	1.327	1.332	1.322	1.3042	1.232
	foodconsistence	210.8	214.6	210.4	209.5	204.4	209.94	210.4

(4) 迭代次数 (maxgen) 对算法性能的影响

从表3.5可以看出, 随着迭代次数的增加, 会进一步找到更优的解, 但同时所花费的时间也就越多, 根据电梯的实际情况, 这里maxgen取100较为适宜。

表3.5 迭代次数 (maxgen) 对算法性能的影响

		Sample (Visual=30, Option=24, Number=25, Trynumber=10)						
		1	2	3	4	5	Avg	Min
maxgen=80	Time	0.343	0.344	0.332	0.344	0.355	0.3436	0.332
	foodconsistence	260.4	262.5	259.2	263.1	259.4	260.92	259.2
maxgen=90	Time	0.455	0.442	0.472	0.463	0.432	0.4528	0.432
	foodconsistence	251.4	250.2	254.3	253.2	249.4	251.7	249.4
maxgen=100	Time	0.573	0.552	0.556	0.599	0.743	0.6046	0.552
	foodconsistence	222.4	220.9	221.2	222.1	220.2	221.36	220.2
maxgen=110	Time	0.831	0.845	0.843	0.874	0.824	0.8434	0.831
	foodconsistence	211.3	212.3	209.3	212.5	214.6	212	209.3

表 3.6 参数取值

召唤个数	Trynumber	Visual	Number	Maxgen
9	10	12	25	25
10	10	13	25	25
11	10	14	25	25
12	10	15	25	25
13	10	17	25	60
14	10	18	25	60
15	10	19	25	60
16	10	20	25	70
17	10	22	25	70
18	10	23	25	70
19	10	24	25	100
20	10	25	25	100
21	10	26	25	100
22	10	28	25	100
23	10	29	25	100
24	10	30	25	100
25	10	31	25	100
26	10	33	25	100
27	10	34	25	100
28	10	36	25	100
29	10	36	25	100
30	10	38	25	100

综上所述，trynumber反应了人工鱼尝试行为的强度，Visual表明感知能力的大小，number表示人工鱼个体数目，maxgen表示整个鱼群的迭代次数，对于这些参数的选取，必须针对具体问题进行分析，选择合适的参数，才能使寻优过程稳健、高效。对于楼层数为16，有4台电梯的大楼，8个及以下外部召唤，可采用穷举法处理，即可在600ms内找到最优解，当大于8个外部召唤时，穷举法计算的时间成指数增加，这时调用鱼群算法处理。由于电梯外部召唤的多态性，实时性，9个外部召唤与16个外部召唤的复杂度不同，相应离散鱼群算法的参数trynumber、visual、number、maxgen也会不同，本文给出9-30不同外部召唤调整好后的参数，见表3.6。

3.5 基于鱼群算法的电梯群控调度策略实现

鱼群算法总流程图3.2所示，算法具体步骤如下：

Step_1. 根据电梯群的当前状态，选择合适的鱼群参数，包括鱼群规模Number，试探次数Trynumber，最大迭代次数Maxgen，人工鱼的感知范围Visual等；

Step_2. 根据选定的参数，随机生成指定数目的人工鱼个体，用于鱼群的初始化；

Step_3. 选取鱼群中最小的食物浓度，将其放入到公告板中；

Step_4. 对于每条人工鱼执行以下操作：

- 1) 分别计算觅食行为、追尾行为、聚群行为的食物浓度，选取三者的最优浓

度值，若其值比当前当状态的浓度值更优，则向最优行为方向前进，否则执行随机行为。

2) 更新当前人工鱼的食物浓度值，如果其值小于公告板中的最小食物浓度值，则对公告板中的最小值进行替换。

Step_5. 判断是否所有人工鱼都进行了寻优，是则转Step_6；否则转Step_4；

Step_6. 调用基于人工鱼群算法的电梯群控调度策略完毕，返回公告板中的值，即将最优的派梯方案传送给电梯群。

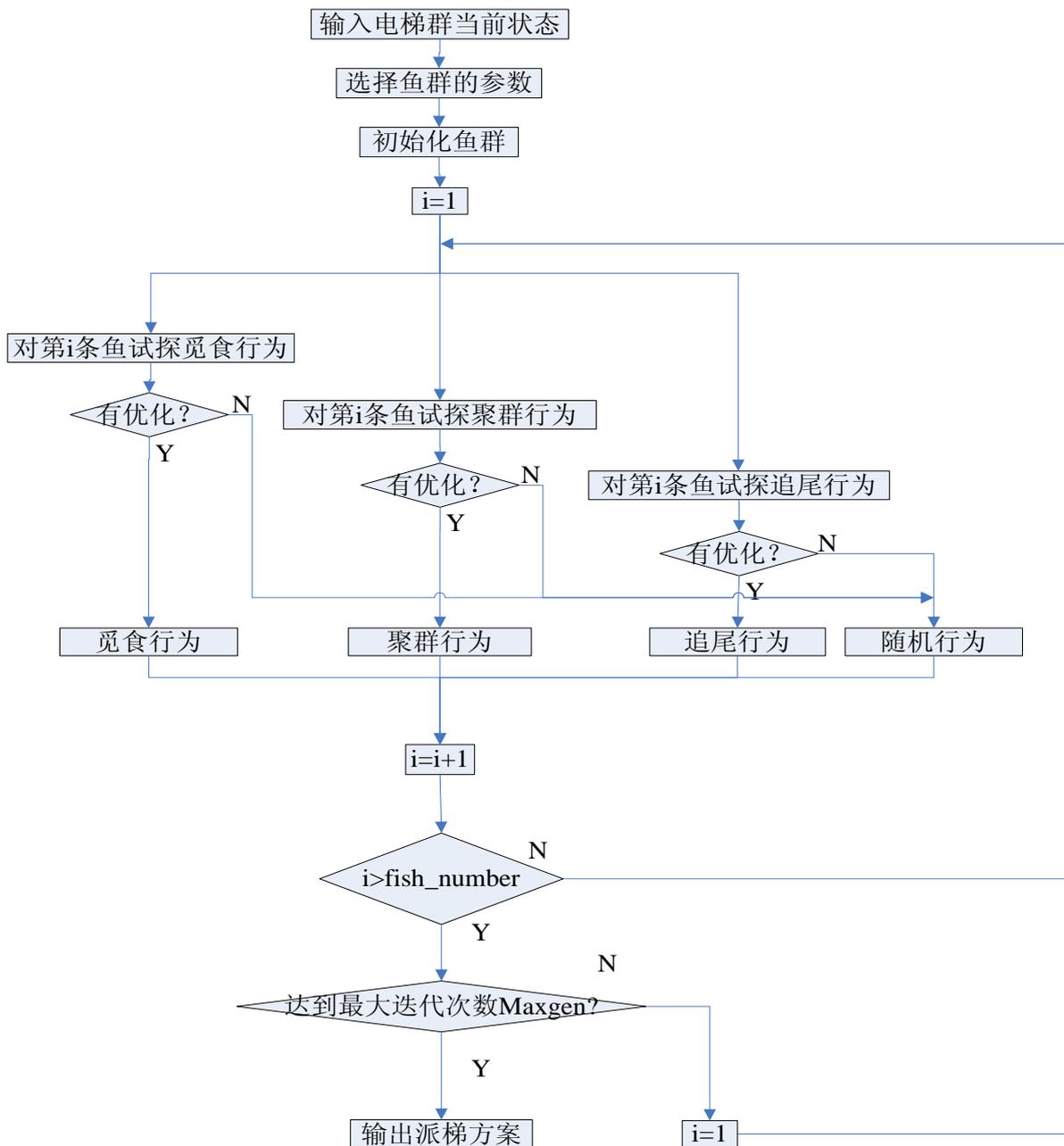


图3.2 算法总流程

追尾行为的关键代码如下：

```
double AF_Follow(Artificial_Fish &Self,Artificial_Fish *P)//追尾行为
{
    for(i=1;i<Self.fish_num;i++)
    {
        if(min_num!=P[i].num&&Dis(Xmin,P[i].AF_X)<Self.Visual)//统计视觉范围内的伙伴数
            Nf++;
    }
    if(Nf)
    {
        if(Nf/Self.fish_num<delta&&Self.food>Ymin)//不拥挤
        {
            for(int i=1;i<(My_layernum*2-1);i++)
                Self.F_X[i]=Xmin[i];//直接跳转到食物浓度最大值的伙伴处
            return Ymin;
        }
        else
            return Self.AF_Prey();//觅食行为
    }
}
```

觅食行为的关键代码如下：

```
double Artificial_Fish::AF_Prey()//觅食行为
{
    for(i=0;i<try_number;i++)//尝试次数
    {
        for(int j=0;j<(My_layernum*2-1);j++)//对所有需要响应的召唤进行分配
        {
            if(Btn[j])
            {
                allocate(j,Next); //Next[i]=(int)(rand()% 电梯数);
            }
            else
                Next[j]=0;
            Tmp[j]=0;//共同点是否已经分配过，这里是初始化
        }
        randnum=(int)(rand()%(1+Visual/2))+Btnum-Visual/2;
        //需响应的召唤个数减去视觉内的个数等于视觉外的个数
        for(int j=0;j<randnum;j++)//选取相同的个数
        {
retry:
            nrand=(int)(rand()% Btnum);
            Next[Rbtn[nrand]]=AF_X[Rbtn[nrand]];//随机选取相同个数的AF_X给Next赋值
            if(Tmp[nrand])//1表示已经赋值过，需重新计算
                goto retry;
            Tmp[nrand]=1;
        }
        Btnupdown(AF_X);//调度方案分配给电梯
    }
}
```

聚群行为的关键代码如下：

```
double AF_Swarm(Artificial_Fish &Self,Artificial_Fish *P)//聚群行为
{
    for(j=1;j<(My_layernum*2-1);j++)
```

```

{
    for(i=0;i<Myelenum;i++)
        Array[i]=Tmp[j][i]; //统计出现次数
        Xc[j]=bubble_sort(Array,Myelenum); //找出每个响应的最大值并赋值
    }
    if(Nf)//周围有伙伴
    {
        Self.Btnupdown(Xc);    Yc=Self.AF_Foodconsistence(false);
        if(Nf/Self.fish_num<delta&&Self.food>Yc)
            //伙伴中心食物浓度比当前人工鱼的食物浓度大，且不拥挤
            {
                for(int i=1;i<(Mylayernum*2-1);i++)
                    Self.S_X[i]=Xc[i]; //把伙伴中心的调度方案放置于S_X
                return Yc;
            }
        else
            return Self.AF_Follow(); //继续执行觅食行为
    }
}

```

3.6 小结

本章首先描述了人工鱼群算法的研究现状，接着对其应用领域进行阐述，并对比其它智能算法，在理论上分析其优越性，其次主要阐述了基于鱼群算法的群控调度方法，包括人工鱼的建模、适应度函数的构造、参数分析、人工鱼行为的设计及改进措施。在人工鱼的建模中，介绍了本算法的编码规则；在人工鱼的行为中，介绍了步长的改进策略，并结合电梯运行的实际情况，把轿厢内人数作为调度因素，以及如何根据电梯本身具有多态性的特性进行参数的选定，并给出算法步骤。

第 4 章 基于模糊推理的多目标控制派梯算法

4.1 群控多目标控制的目标函数确定

4.1.1 控制目标的提出

电梯群控系统最终要实现的目标是，在根据交通流识别交通模式的情况下合理调度派遣轿厢，在完成楼内交通的高效输送前提下舒适地运送乘客。群控系统实时掌握呼梯方向、呼梯楼层、电梯状态等信息，尽量满足电梯群控系统的多目标性（长候梯率要极小化、乘梯时间要极小化等）。为实现上述多目标的控制，我们综合考虑乘客的平均候梯时间（AWT）、乘客长候梯率（LWP）、能量消耗（RNC）三个主要目标，根据给予的不同权重，计算出一个最终值，并根据该值分配电梯。

AWT 定义见 2.4.1 小节。乘客到达率、轿厢的剩余空间和剩余载重能力是影响 AWT 的主要因素。当新乘客不断出现，轿厢的剩余空间较小或者电梯已经达到满载状态，群控系统就没有足够的剩余运输力去输送新乘客，甚至出现排队状况，导致乘客的平均候梯时间大大增加。

LWP 定义：候梯时间超过 60 秒的乘客数量占总乘客数的比例。根据心理学数据，候梯时间小于 60 秒是可以接受的，当候梯时间超过 60 秒，乘客的焦躁程度会增加。因此，基于人性化考虑，应尽量减少长候梯事件的发生，降低长候梯率。

RNC 定义见 2.4.2 小节。电梯的启动、加速、减速、停止是影响能量消耗的重要动作。在电梯运行过程中，减少此类动作是降低能耗的关键。

4.1.2 评价函数的确定

评价函数其实就是电梯群控调度依据的调度法则。本文采用的评价函数是基于线性的加权平均算法，在一个多目标控制的问题中，该方法可以以简单易懂的描述形式表达电梯群控调度的总的决策函数，其公式为

$$S_i = F(S_{AWT_i}, S_{LWP_i}, S_{RNC_i}) = w_1 S_{AWT_i} + w_2 S_{LWP_i} + w_3 S_{RNC_i} \quad (4.1)$$

其中， S_{AWT_i} ——第 i 台电梯的评价目标平均候梯时间的可信度；

S_{LWP_i} ——第 i 台电梯的评价目标长候梯率的可信度；

S_{RNC_i} ——第 i 台电梯的评价目标能量消耗的可信度；

S_i ——为第 i 台电梯的可信度；

w_i ——为权重， $w_i \in [0,1]$ 且 $\sum_{i=1}^3 w_i = 1$ ；

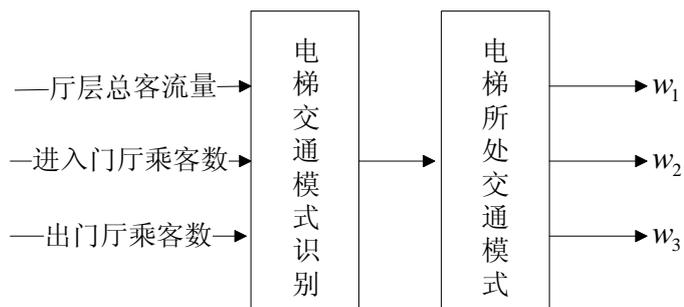


图 4.1 权重的获取

当楼内客客流量较大，电梯交通处于上行高峰或者下行高峰，此时基于疏导交通考虑，尽量减少候梯人数，必须将时间因素优先考虑，节约能源则作为次要因素，可以给予 AWT、LWP 较大的权重。反之，当楼内交通处于空闲期，呼梯人数不多，这种情况可以优先考虑节能，因为电梯此时能基本满足交通需求，所以 RNC 的权重可以大一些，AWT、LWP 的权重可以稍小。

因此，对于一次呼梯我们可以获得当前状态下所有电梯的可信度 S_1 、 S_2 ，……， S_n ，则

$$S_e = \max(S_1, \dots, S_n) \quad (4.2)$$

将呼梯信号分配给 S_e 对应的那台电梯。

通过以上分析以及借助电梯专家的经验，不同交通模式下的各个控制目标的权重推荐数值如表 4.1 所示。

表 4.1 各目标在不同交通模式下的权重^[47]

	w_1	w_2	w_3
上行高峰交通模式	0.4	0.5	0.1
下行高峰交通模式	0.4	0.5	0.1
午餐前交通模式	0.3	0.55	0.15
午餐后交通模式	0.15	0.6	0.25
层间随机交通模式	0.2	0.4	0.4

4.2 基于鱼群算法的模糊推理线性回归曲线的优化

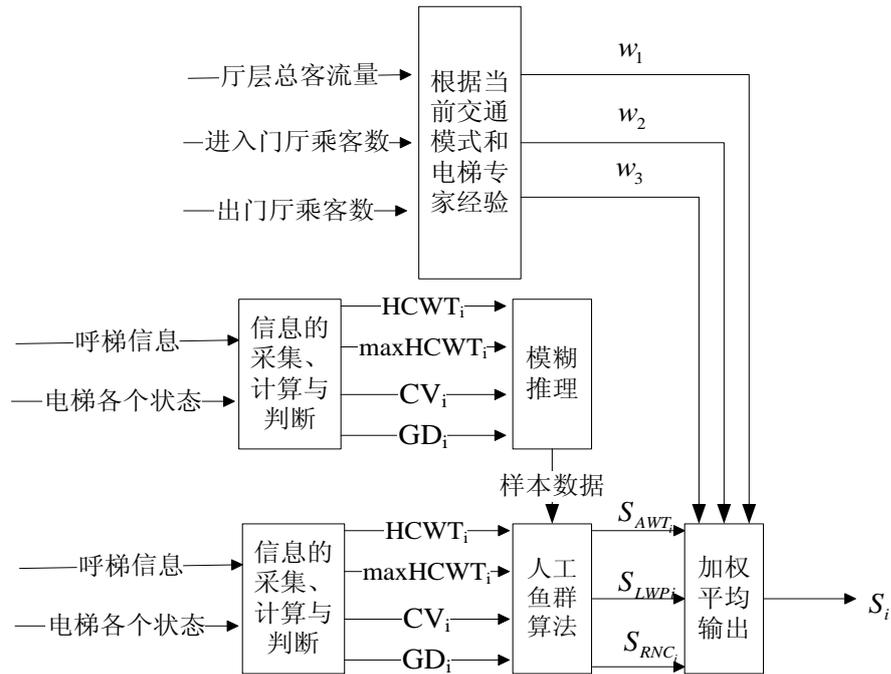


图 4.2 基于人工鱼群算法的群控多目标控制示意图

本节介绍基于人工鱼群算法来实现的电梯群控系统的优化调度。首先利用模糊推理得到一系列属于电梯各个目标的可信度数据，在获得部分数据后作为学习样本，针对各个目标分别利用人工鱼群算法进行训练，获得回归函数，克服模糊推理缺乏学习能力的缺点，接下来以回归函数作为计算可信度的标准，最后通过加权平均，将呼梯信号指派给可信度最高的那台电梯。本文提出的基于人工鱼群算法的群控多目标控制示意图如图 4.2 所示。图中的 $HCWT_i$ 、 $\max HCWT_i$ 、 CV_i 、 GD_i 将在下一个小节中进行详细介绍。

4.2.1 输入量的计算

由图 4.2 我们可以看到，与电梯的各个控制目标 AWT、LWP、RNC 有关的输入变量是 $HCWT_i$ 、 $\max HCWT_i$ 、 CV_i 、 GD_i 。首先给出各个量的物理定义。

我们假设新出现的呼梯信号分配给了第 i 台电梯，则

(1) 厅层召唤等待时间 ($HCWT_i$)：新的呼梯信号从产生到被第 i 台电梯响应的等待时间，包括电梯的运行时间 (T_{move}) 和轿厢的停靠时间 (T_{stop})。若

$$T_{move} = \text{加速时间} + \text{高速运行时间} + \text{减速时间} \quad (4.3)$$

$$T_{stop} = \text{电梯开门时间} + \text{乘客进出时间} + \text{关门时间} \quad (4.4)$$

则

$$HCWT_i = T_{move} + T_{stop} \quad (4.5)$$

(2) 厅层召唤最大等待时间 ($\max HCWT_i$)：第 i 台电梯已登记的所有召唤命令和新登记召唤命令的最大等待时间。

$$\max HCWT_i = \max \{ T_{assigned}, T_{new} \} \quad (4.6)$$

其中 $T_{assigned}$ —— 第 i 台电梯已登记的所有召唤命令的最大等待时间；

T_{new} ——新登记召唤命令的等待时间。

(3) 剩余响应能力 (CV_i): 第 i 台电梯登记了新的呼梯信号后, 对未来呼梯的响应能力, 它的大小与进出轿厢乘客的数量有关, 我们将该能力量化如下,

$$CV_i = 1 - \frac{err_person}{in_person} \quad (4.7)$$

其中, in_person 为单位时间内企图进入第 i 台电梯轿厢的人数;

err_person 为单位时间内未进入第 i 台电梯轿厢的人数。

(4) 召唤集中程度 (GD_i): 新的呼梯信号所在楼层位置与第 i 台电梯已经登记的所有呼梯命令 (包括外召和内召) 产生楼层位置的最短距离。电梯的能量消耗主要是在加速、减速阶段消耗较大, 电梯的加减速距离一般也大于一个楼层的高度, 通常召唤集中程度表示如下,

$$CD_i = \frac{4H}{MIN(distance)} \quad (4.8)$$

其中, 分母代表已经呼梯楼层与新呼梯信号楼层之间的最小距离, H 为楼层的高度。

4.2.2 结合模糊推理获得学习样本

人工鱼群算法的学习样本, 是通过模糊推理得到的一部分数据。进行模糊推理前, 必先对个输入变量进行模糊化, 即确定隶属函数, 模糊化要根据输入量的取值范围、系统的需求来确定。各个变量与各个目标的可信度推理关系如图 4.3 所示。下面分别确定 $HCWT_i$ 、 $\max HCWT_i$ 、 CV_i 和 GD_i 的隶属函数。

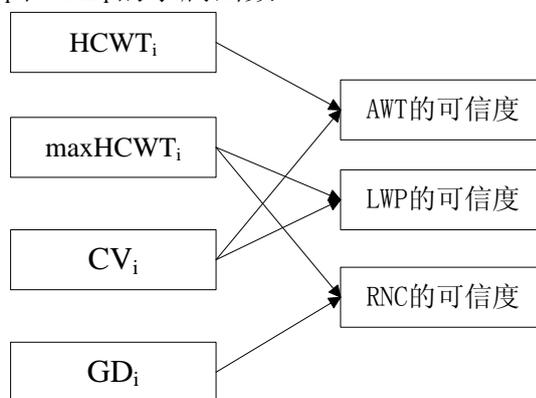


图 4.3 输入量与目标之间的关系^[47]

(1) $HCWT_i$ 的模糊化

根据调查, 乘客在候梯时心理的焦躁程度和候梯时间成一定关系, 具体描述如下:

I、0 秒 \leq 候梯时间 \leq 20 秒, 乘客的心理状态是好的;

II、20 秒 \leq 候梯时间 \leq 40 秒, 乘客心理状态一般;

III、40 秒 \leq 候梯时间 \leq 60 秒, 乘客心理开始变得焦躁。

我们用模糊术语 S (small)、M (medium)、L (lagre) 描述变量, 分别表示“小”、“中”、“大”, 这些模糊概念可以用图 4.4 所示的隶属度函数图来描述。S、M 和 L 的隶属度函数分别为

$$\mu(HCWT)_S = \begin{cases} 1, 0 \leq HCWT \leq 20 \\ \frac{(35 - HCWT)}{10}, 20 \leq HCWT \leq 30 \\ 0, HCWT \geq 30 \end{cases} \quad (4.9)$$

$$\mu(HCWT)_M = \begin{cases} 0, HCWT < 20 \\ \frac{(HCWT - 20)}{20}, 20 \leq HCWT \leq 40 \\ \frac{(60 - HCWT)}{20}, 40 \leq HCWT \leq 60 \\ 0, HCWT \geq 60 \end{cases} \quad (4.10)$$

$$\mu(HCWT)_L = \begin{cases} 0, HCWT < 50 \\ \frac{(HCWT - 50)}{10}, 50 \leq HCWT \leq 60 \\ 1, HCWT \geq 60 \end{cases} \quad (4.11)$$

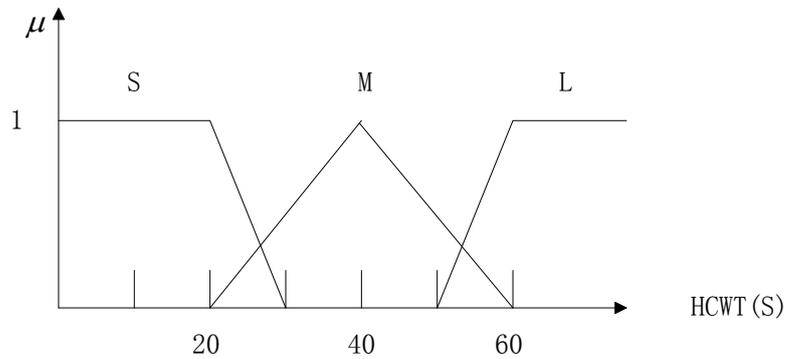


图 4.4 用模糊语言表述厅层召唤等待时间的隶属度函数

(2) $\max HCWT_i$ 的模糊化。

一般而言，群控系统的厅层召唤最大等待时间不应该超过 90 秒，当其值在 40 秒以内较优，在 60 秒左右乘客也是可以接受的。用模糊术语 S、M 和 L 描述，其隶属度函数如图 4.5。的隶属度函数表达式分别为

$$\mu(\max HCWT)_S = \begin{cases} 1, 0 \leq \max HCWT \leq 40 \\ \frac{(50 - \max HCWT)}{10}, 40 \leq \max HCWT \leq 50 \\ 0, \max HCWT \geq 50 \end{cases} \quad (4.12)$$

$$\mu(\max HCWT)_M = \begin{cases} 0, \max HCWT < 40 \\ \frac{(\max HCWT - 40)}{20}, 40 \leq \max HCWT \leq 60 \\ \frac{(80 - \max HCWT)}{20}, 60 \leq \max HCWT \leq 80 \\ 0, \max HCWT \geq 80 \end{cases} \quad (4.13)$$

$$\mu(\max HCWT)_L = \begin{cases} 0, \max HCWT < 70 \\ \frac{(\max HCWT - 70)}{10}, 70 \leq \max HCWT \leq 80. \\ 1, \max HCWT \geq 80 \end{cases} \quad (4.14)$$

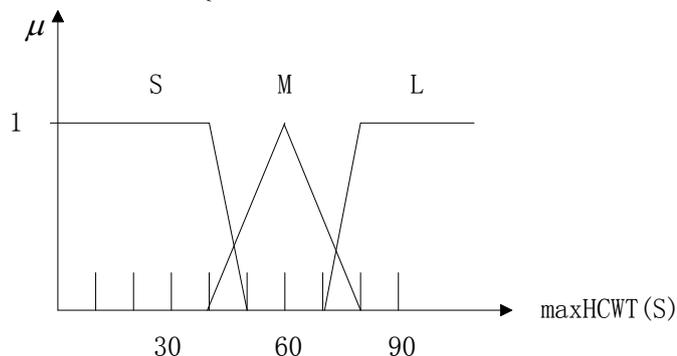


图 4.5 用模糊语言表述厅层召唤最大等待时间的隶属度函数

(3) CV_i 的模糊化

单位时间内未进入轿厢的人数越多，说明电梯的剩余响应能力越小，高峰期电梯的剩余响应能力一般不应超过 80%。用模糊术语 S、M 和 L 描述，其隶属度函数如图 4.6。的隶属度函数表达式分别为

$$\mu(CV)_S = \begin{cases} 1, 0 \leq CV \leq 0.2 \\ \frac{(0.3 - CV)}{0.1}, 0.2 \leq CV \leq 0.3 \\ 0, CV \geq 0.3 \end{cases} \quad (4.15)$$

$$\mu(CV)_M = \begin{cases} 0, CV < 0.2 \\ \frac{(CV - 0.2)}{0.2}, 0.2 \leq CV \leq 0.4 \\ \frac{(0.6 - CV)}{0.2}, 0.4 \leq CV \leq 0.6 \\ 0, CV \geq 0.6 \end{cases} \quad (4.16)$$

$$\mu(CV)_L = \begin{cases} 0, CV < 0.5 \\ \frac{(0.6 - CV)}{10}, 0.5 \leq CV \leq 0.6. \\ 1, CV \geq 0.6 \end{cases} \quad (4.17)$$

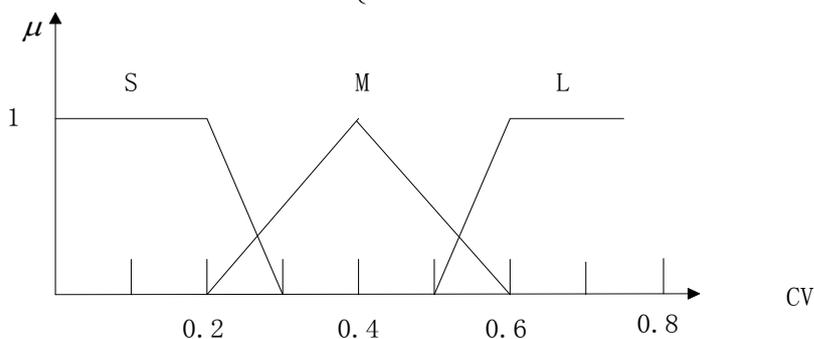


图 4.6 用模糊语言表述剩余响应能力的隶属度函数

(4) GD_i 的模糊化

从 GD 的定义公式，我们界定当 GD 的值小于 0.3，电梯的能耗比较理想，当大于 0.7 时，电梯的能耗会比较大。用模糊术语 S、M 和 L 描述，其隶属度函数如图 4.7。的隶属度函数表达式分别为

$$\mu(GD)_S = \begin{cases} 1, 0 \leq GD \leq 0.2 \\ \frac{(0.35 - GD)}{0.1}, 0.2 \leq GD \leq 0.35 \\ 0, GD \geq 0.35 \end{cases} \quad (4.18)$$

$$\mu(GD)_M = \begin{cases} 0, GD < 0.25 \\ \frac{(GD - 0.25)}{0.2}, 0.25 \leq GD \leq 0.45 \\ \frac{(0.65 - GD)}{0.2}, 0.45 \leq GD \leq 0.65 \\ 0, GD \geq 0.65 \end{cases} \quad (4.19)$$

$$\mu(GD)_L = \begin{cases} 0, GD < 0.55 \\ \frac{(0.7 - GD)}{10}, 0.55 \leq GD \leq 0.7 \\ 1, GD \geq 0.7 \end{cases} \quad (4.20)$$

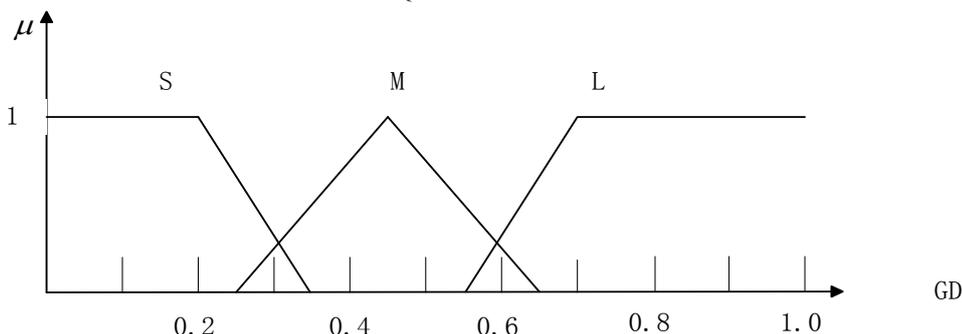


图 4.7 用模糊语言表述召唤集中程度的隶属度函数

结合电梯专家的知识 and 经验，AWT、LWP 和 RNC 的模糊规则归纳如下，如表 4.2，4.3，4.4 所示。表中的输入量均为采用上述方法模糊化后的值。

表 4.2 AWT 的模糊规则表^[47]

AWT _i		HCWT _i		
		大	中	小
CV _i	大	中		大
	中			
	小	小		中

表 4.3 LWP 的模糊规则表

LWP _i		maxHCWT _i		
		大	中	小
CV _i	大	中		大
	中			
	小	小		中

表 4.4 RNC 的模糊规则表^[47]

RNC _i		maxHCWT _i		
		大	中	小
CV _i	大	中		大
	中			
	小	小		中

4.2.3 回归问题

首先讨论一个简单的回归问题，如图 4.8 所示，考虑变量 x 和 y 的关系。假设已经获得若干样本及其对应的值 $T = \{(x_1, y_1), \dots, (x_l, y_l)\} \in (X \times Y)^l$ ，其中 $x_i \in X = R$ ， $y_i \in Y = R$ ， $i = 1, \dots, l$ 。尝试根据这 l 个样本值，推导 x 和 y 之间的函数映射关系 $y = f^*(x)$ 。

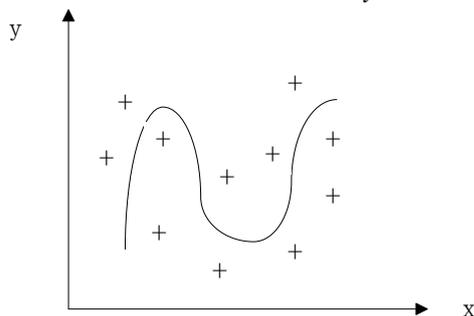


图 4.8 简单回归问题

把这 l 个样本值以图形化的方式描述出来，问题就转化为寻求一条曲线 $y = f(x)$ 。

解决这个问题一个直接的想法是，把 $y = f^*(x)$ 限制在一个不大复杂的函数集合中，在这个集合中找到尽可能满足

$$y_i = f^*(x_i), \quad i = 1, \dots, l \quad (4.21)$$

的 $f^*(x)$ 。通常使用一些拟合或回归的方法，经常使用的是线性回归方法，通过最小化误差的平方和寻找数据的最佳函数匹配。通常用平方损失函数

$$(y_i - f(x_i))^2 = (y_i - ax_i - b)^2 \quad (4.22)$$

来度量 $y = f(x) = ax + b$ 偏离 $y_i = f(x_i)$ 的程度。当这个值越小，说明偏差越小。所以要使该值尽可能小，可用最优化问题描述如下

$$\min_{a,b} \sum_{i=1}^l (y_i - ax_i - b)^2. \quad (4.23)$$

该问题的解 (a^*, b^*) 可以确定函数 $y = f(x) = ax + b$ 。

显然，可以将上述的回归问题和求解方法推广到 n 维空间。相对应的训练集为 $x_i \in X = R^n$ ， $y_i \in Y = R$ ， $i = 1, \dots, l$ 。相应的 $y = f^*(x)$ 所在的函数集合也可以推广为某个实函数集合 F 。

4.2.4 人工鱼群算法多元线性回归的利用

利用 4.2.2 小节的模糊推理，获得各种情况下平均候梯时间、长候梯率和能耗可信度数据，作为可信度数据为学习样本，本文借助人鱼群的全局搜索能力，获得平均候梯时间、长候梯率和能耗可信度计算的决策函数，克服了模糊推理缺乏学习能力的缺点。

人工鱼群算法在群控算法中的系统工作原理如图 4.9 所示。HCWT、maxHCWT、CV、GD 既是样本数据获取的基础，也是人工鱼群算法的输入，通过采集系统中呼梯信号和电梯的各个状态并根据相关公式去计算它们的值。当人工鱼群算法训练完毕，即可获得相应可靠度的决策函数。

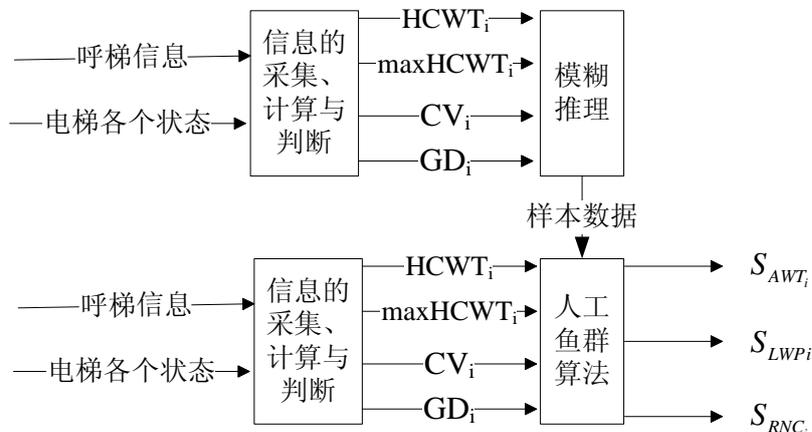


图 4.9 群控系统中人工鱼群算法的工作原理

已知由平均候梯时间、长候梯率和能耗可信度数据组成 l 个样本数据 $x_i \in X = R^n$ ， $y_i \in Y = R$ ， $i = 1, \dots, l$ ，试建立它们之间的线性回归方程，即决策函数：

$$\hat{y} = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + b_4 x_4 \quad (4.24)$$

以基于人工鱼群算法作为群控中采用的算法，并建立相应的决策机制，学习数据通过模糊推理得到，训练阶段的关键步骤如下：

Step_1. 采集呼梯信号以及各台电梯状态信息；

Step_2. 计算 HCWT、maxHCWT、CV、GD 的值；

Step_3. 模糊推理获得相应的 S_{AWT_i} 、 S_{LWP_i} 和 S_{RNC_i} 的值；

Step_4. 重复 Step_1 to Step_3，直到获得预定数量的数据后，根据图 4.3 中的关系，划分为 $(HCWT, CV, S_{AWT_i})$ ， $(maxHCWT, CV, S_{LWP_i})$ 和 $(maxHCWT, GD, S_{RNC_i})$ 三个集合的相应学习样本；

Step_5. 训练三个学习样本集合相应的人工鱼群直至收敛；

Step_6. 保存人工鱼群寻优相关结果。

相对于人工鱼群寻优的训练阶段，它在群控算法中真正起决策作用的步骤如下：

Step_1. 采集呼梯信号以及各台电梯状态信息；

Step_2. 计算 HCWT、maxHCWT、CV、GD 的值；

Step_3. 按训练好的人工鱼群多元回归决策函数去计算 S_{AWT_i} 、 S_{LWP_i} 和 S_{RNC_i} 的值；

Step_4. 计算出各台电梯的可信度；

Step_5. 根据各台电梯的可信度进行派梯。

在训练阶段，最重要的一个步骤是 Step_5，即人工鱼群多元回归决策函数的求解。本文采用 AFSA 算法优化这个多元线性回归分析问题中的 b_0 、 b_1 、 b_2 、 b_3 、 b_4 参数及食

物浓度函数值 $f = 1 / \sum_{i=1}^1 (y - \hat{y})^2$ ，设鱼群规模 $m=100$ ，最大迭代次数 $gen=100$ ，拥挤度因子 $\delta = 0.618$ ，感知范围 $Visual=10$ ，步长 $step=3$ ，试探次数 $Trynumber=200$ 。

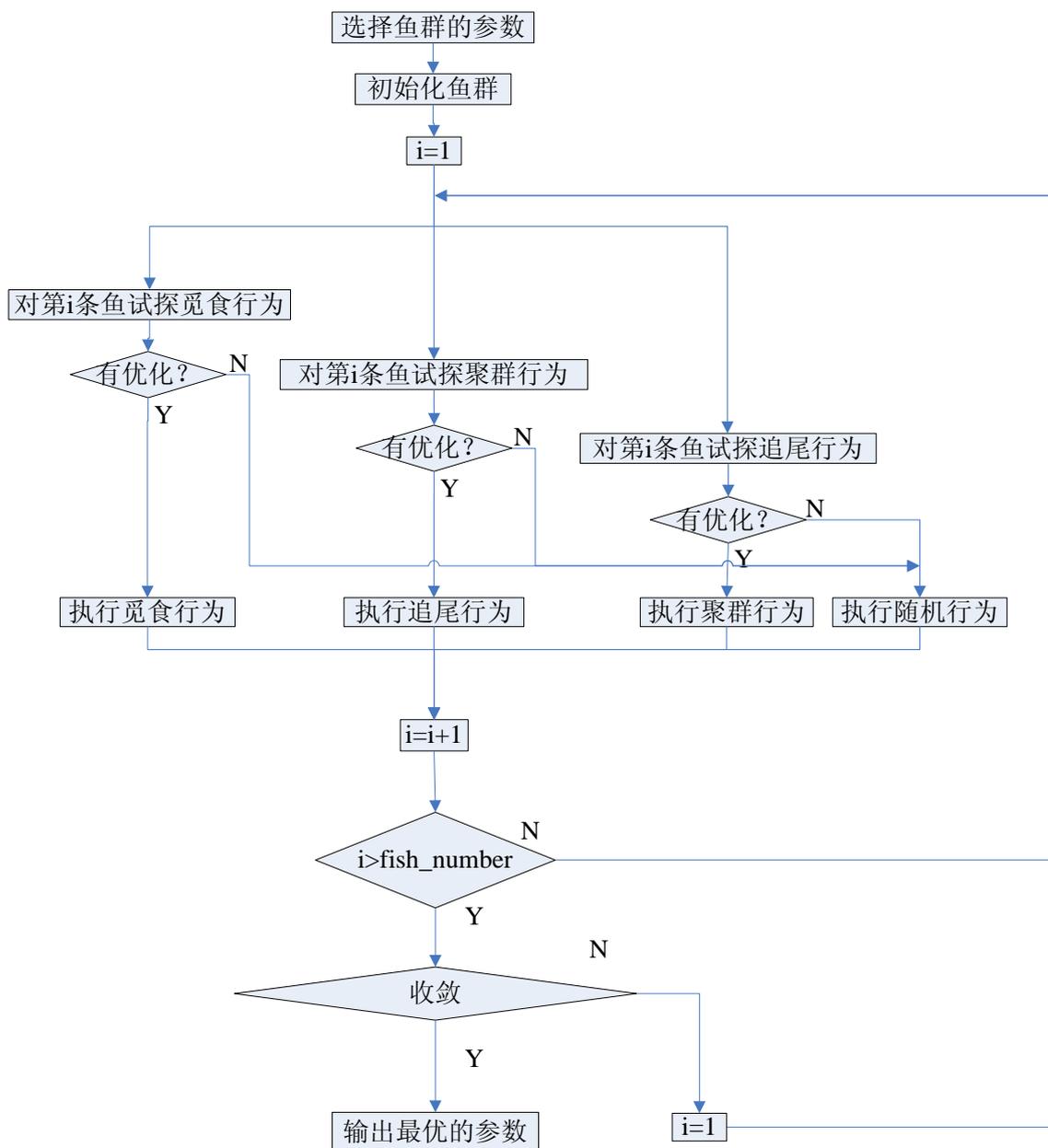


图4.10 算法总流程

鱼群算法总流程图 4.10 所示，算法具体步骤如下：

Step_1. 选择合适的鱼群参数，包括鱼群规模 $Number$ ，试探次数 $Trynumber$ ，最大迭代次数 $Maxgen$ ，人工鱼的感知范围 $Visual$ 等；

Step_2. 根据选定的参数，随机生成指定数目的人工鱼个体，用于鱼群的初始化；

Step_3. 选取鱼群中最大的食物浓度，将其放入到公告板中；

Step_4. 对于每条人工鱼执行以下操作：

1) 分别计算觅食行为、追尾行为、聚群行为的食物浓度，选取三者的最优浓

度值，若其值比当前当状态的浓度值更优，则向最优行为方向前进，否则执行随机行为。

2) 更新当前人工鱼的食物浓度值，如果其值小于公告板中的最大食物浓度值，则对公告板中的最大值进行替换。

Step_5. 判断是否所有人工鱼都进行了寻优，是则转Step_6；否则转Step_4；

Step_6. 调用基于人工鱼群算法完毕，返回公告板中的值，即将最优的告板中的值即为要求的最优值，其对应的人工鱼即为最优的参数。

4.3 小结

本章首先针对电梯群控的多目标性提出基于线性加权平均的电梯群控目标评价函数。在评价函数的计算方式中针对模糊推理缺乏学习性的特点，引入了人工鱼群算法多元线性回归。通过训练人工鱼群算法得到计算电梯可信度的决策函数，并将之利用到群控决策中。

第5章 建立电梯群控虚拟仿真环境设计及仿真

5.1 电梯群控虚拟仿真环境的总体设计

为了提高电梯群的运行性能，人们不断地将新的群控算法引入到各具特色的智能化电梯群控系统中。此时，如何准确而方便的评价群控方法的优劣便成为电梯群控研究中的一个重要问题，而构建一个电梯群控仿真平台则是解决这一问题的首选方案。

电梯群控虚拟仿真环境需要解决的一个关键问题是如何实现对客流的仿真。仿真平台必须做到：

- (1) 手工输入各种呼梯信号；
- (2) 自动产生随机客流数据；
- (3) 可制定不同交通模式下的客流数据。

考虑现实世界中的各种大楼，楼层数不同、层高不同、所拥有的电梯数不同。电梯群控虚拟仿真环境的另一个关键问题就是配置多样化的问题。要求仿真平台可根据用户需求定制不同数目的电梯群及不同楼层数的建筑物等信息。

为了有利于新的调度算法的研发与测试，调度算法接口必须具有开放性。

5.1.1 电梯群控虚拟仿真环境总体流程

仿真环境首先通过对建筑物及电梯配置参数的设置，自动生成相应的电梯群。当有

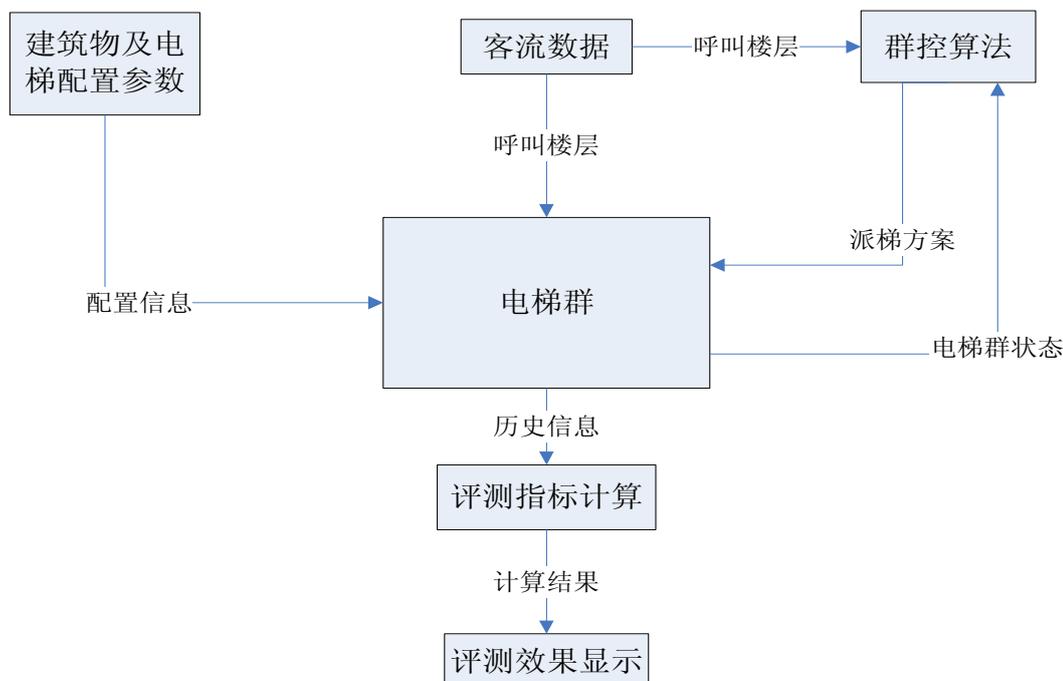


图5.1 电梯群控虚拟环境总体流程

新的乘客触发外部召唤时，群控算法模块获取电梯群信息及乘客所在召唤信息，经运算之后将派梯方案传送给电梯群。各台电梯获取需要响应的召唤链表，然后按照运行逻辑将乘客从起始楼层运送到目的楼层。与此同时，测评指标模块读取电梯及乘客的历史数据对各种指标进行计算，且把最终结果刷新显示在指定位置。电梯群控虚拟环境总体流程如图5.1所示。

5.1.2 电梯群控虚拟仿真环境总体框架

虚拟环境下的电梯必须严格遵守真实电梯的运行规则，并模拟电梯运行时的各种状态。该仿真软件的功能模块主要包括：客流设计模块、仿真模块、群控算法模块、评测模块。客流设计模块用来产生电梯交通流仿真数据；电梯仿真模块按照一定的运行逻辑和运行规律，完成群控系统分派的应答任务，将乘客从起始楼层运送到目的楼层。群控算法模块用来完成派梯选择，选择最合适的电梯去应答新产生的召唤；评测模块用来统计电梯及乘客的历史数据和评估电梯系统的性能指标。各个独立的模块承担不同的任务，协同工作，完成整个电梯系统的仿真运行。其群控虚拟环境总体框架如图5.2所示。

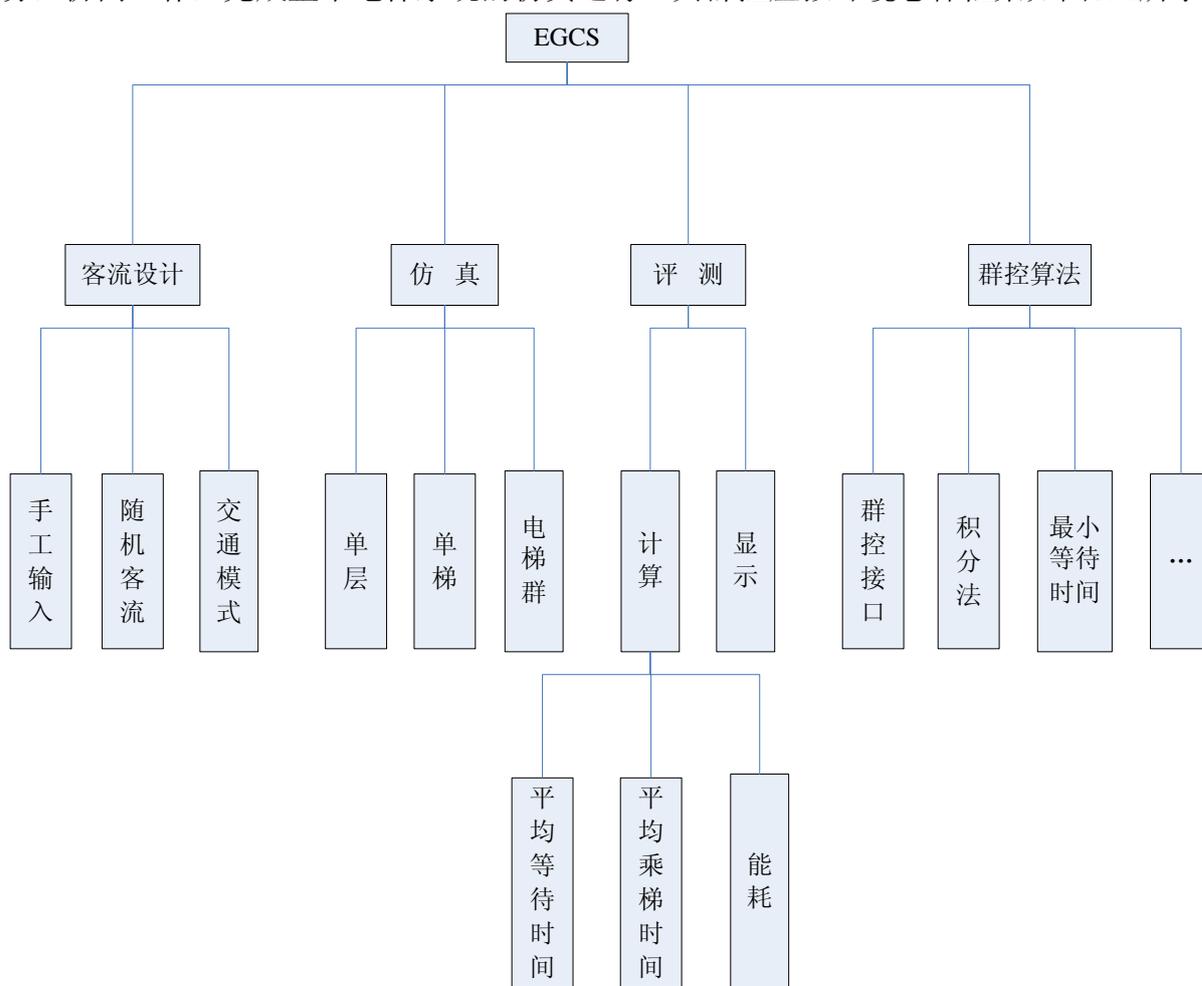


图5.2 电梯群控虚拟环境总体框架

5.1.3 电梯群类结构

电梯群控虚拟仿真环境采用面向对象编程技术，用 C#语言实现，调度算法借助于

C++实现, 开发工具为 Microsoft Visual Studio 2008, 下面主要介绍电梯群控系统类中的类, 仿真系统各类关系如图 5.3 所示。

(1) EGCEMain(主控中心类)

EGCEMain 类是电梯群控虚拟仿真平台的主窗体, 包含许多重要函数, 如: pasgFlowsting_click(客流设置), 可根据用户需求定制客流; myRealTime(实时评测), 该函数获取电梯群跑动过程中的客流状态, 分析各种评测指标; algorithm_click(算法调度), 通过鱼群算法为用户提供一系列最佳的电梯; getonlinePro(实时监控), 可动态获取实体电梯的数据。

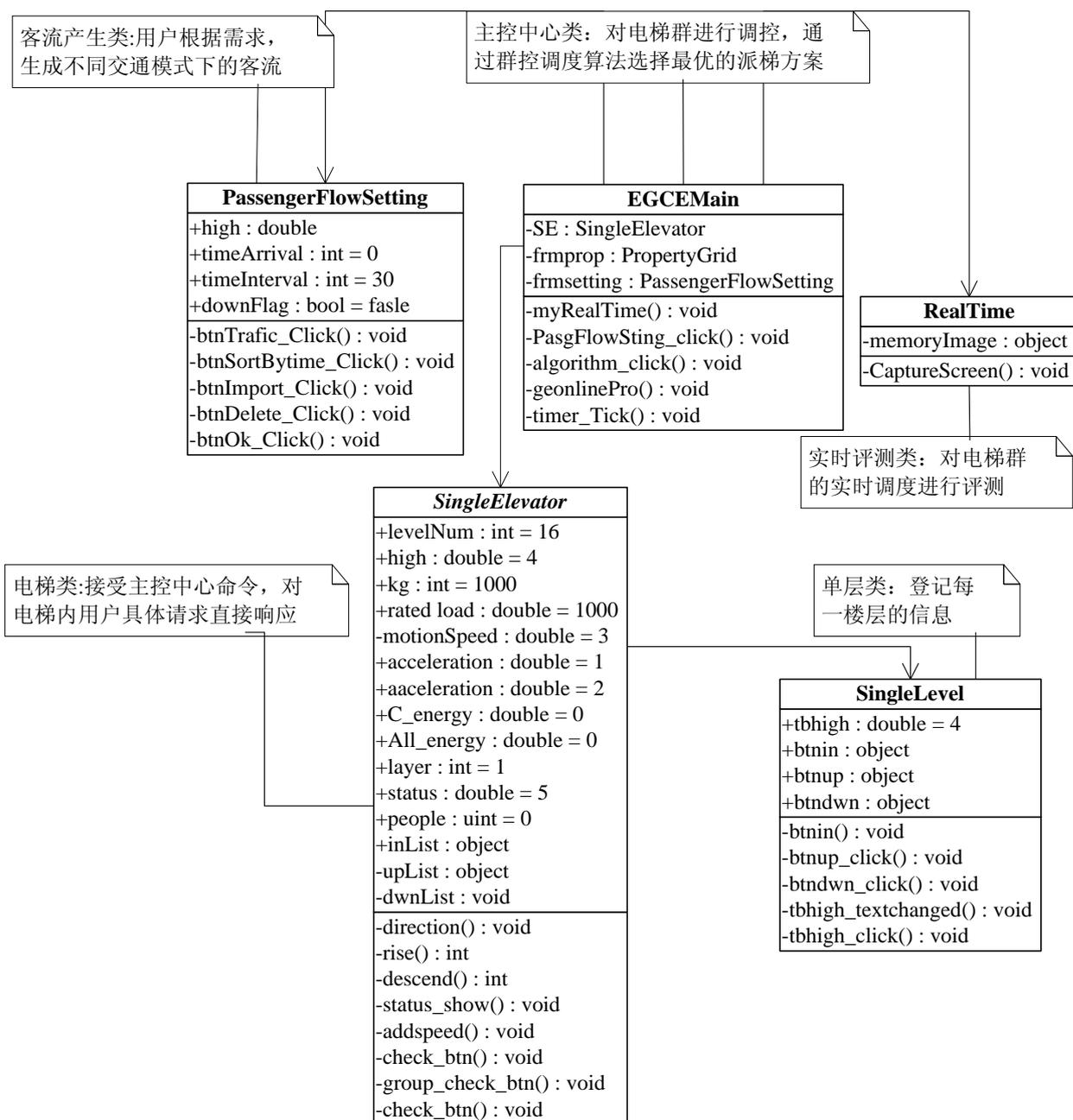


图 5.3 仿真系统各类关系

(2) SingleElevator(单部电梯类)

电梯类响应主控中心发来的命令请求，让电梯正常运作，电梯类包含许多重要函数，如：`check_btn`(接收函数),获取主控中心发来的命令请求，如上召、下召、内召信号。`rise`(上行函数)，电梯根据规则上行；`descend`(下行函数)，电梯根据规则下行；`status_show`(状态更新)，根据运行规则刷新电梯当前状态。

(3) SingleLevel(单层类)

单层类将楼层进行抽象，设置每一楼层的参数，如楼层号、层高、上召\下召按钮。

(4) RealTime(实时评测类)

实时评测类，根据电梯群运动过程中对乘客的接送及消耗的能耗进行分析，生成平均候梯时间、平均乘梯时间、长时间候梯率、电梯能耗、舒适度等曲线。

(5) PassengerFlowSetting(客流设置类)

客流设置类，根据不同用户的需求，设计不同交通模式的客流，便于仿真测试调度算法。

5.2 电梯客流模块

(Poisson) 分布可以很好的描述乘客到达的时间^[49]:

$$P\{\tau_{i+1} - \tau_i = t\} = \begin{cases} 1 - e^{-\lambda t}, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (5.1)$$

其中， t 是乘客到达电梯门口时的时间间隔最大值， λ 为乘客到达率， τ_i 表示第 i 个乘客的到达时刻。

交通客流的模拟，第一步需要确定乘客外部召唤信号的产生楼层和内部召唤的目的层。根据不同交通模式的特征，利用蒙特卡罗方法确定外召产生楼层和内召目的楼层的比例，从而确定乘客的起始楼层和目的层。第二步需要确定乘客的到达时刻。在 $(0,1)$ 的取值范围内，产生一个随机数 P ，接着计算乘客到达电梯门口时的时间间隔最大值 t 的值，即 $t = -\ln(1-p)/\lambda$ 。通过公式 $\tau_{i+1} = \tau_i + t$ 将间隔时间累加，这样就确定了乘客的到达时刻。

本文所设计的客流产生模块主要实现以下三个功能，客流产生界面如图 5.4 所示：

- (1) 手工输入。可以实现设定单个乘客产生的时间、呼叫楼层、目的楼层及体重。
- (2) 自动产生。可以实现通过设置乘客到达率、上行客流百分比、下行客流百分比、层间客流百分比来产生一定设定量的随机客流数据。
- (3) 交通模式。可用通过调整不同时间段如上行高峰、下行高峰时间及乘客上下行比例来产生不同交通模式下的客流数据。

上行高峰模式关键代码如下：

```
public void UpStream(Elevator ele, bool ing) {
    for (int i = 0; i < 60; i++)
    {
        for (int k = 0; k < ren[i]; k++)//ren[i]中存放的是一分钟的人流密度
        {
            time = start.AddSeconds(rd.Next(1, 60));//开始时间
```

```

weight = NormalRandom.GetWeightByNormal();
passengerNo++;
int percent = rd.Next(1, 101);//选取随机比例
if (percent < Convert.ToInt32(this.tbuppctup.Text))//符合上行比例
{
    Low = 1;//呼叫楼层
    High = rd.Next(2, height + 1);//目的楼层
}
else
if(percent<Convert.ToInt32(this.tbuppctup.Text) +
    Convert.ToInt32(this.tbuppctdown.Text))//下行比例
{//选取随机比例
    Low = rd.Next(2, Convert.ToInt32((height + 1) / 2));
    High = rd.Next(Convert.ToInt32((height + 1) / 2) + 1, height + 1);
}
else//层间上行
if(percent<Convert.ToInt32(this.tbuppctup.Text)+
Convert.ToInt32(this.tbuppctdown.Text) + Convert.ToInt32(this.tbuppu.Text))
{
    High = rd.Next(2, Convert.ToInt32((height + 1) / 2));
    Low = rd.Next(Convert.ToInt32((height + 1) / 2) + 1, height +
        1);
}
else//层间下行
{
    Low = rd.Next(2, height + 1);
    High = 1;
}
}
//将数据增加到listview，产生时间、呼叫楼层、目的楼层、体重
listview1.Items.Add(new ListViewItem(new string[] { passengerNo.ToString(), time.Hour + ":" +
time.Minute + ":" + time.Second, Low.ToString(), High.ToString(), weight.ToString() }));
}
start = start.AddMinutes(1); //产生时间加1
}

```



图5.4 客流设置

5.3 电梯仿真模块

利用计算机系统上的软件开发环境，开发电梯群控系统虚拟仿真软件，实现系统参数配置、多台电梯的可视化运行界面等功能，不仅可以及时发现群控调度算法的缺陷，而且还能比较各种算法的性能，极大地方便了算法应用前的测试工作。

5.3.1 电梯动态运行的直观显示

电梯群控的仿真环境必须与实际情况相一致，每台电梯轿厢配备一个内召信号板；每一楼层均配有向上外部召唤按钮和向下外部召唤按钮，但考虑到大楼底层和顶层的特殊性，这两层分别只装有向上外部召唤按钮和向下外部召唤按钮。电梯处于满载时，群控算法就不应该为其分配其他的外召响应任务；在不受限制的情况下，呼梯信号和目的楼层可以来自大楼的任一层。仿真平台下的单部电梯必须遵守真实电梯的运行规则，且必须具有真实电梯所包含的停层、加速、高速、减速等过程；如图 5.5 所示为电梯所有状态，如空闲、开门、装客、关门、匀加速、匀速运行等状态。

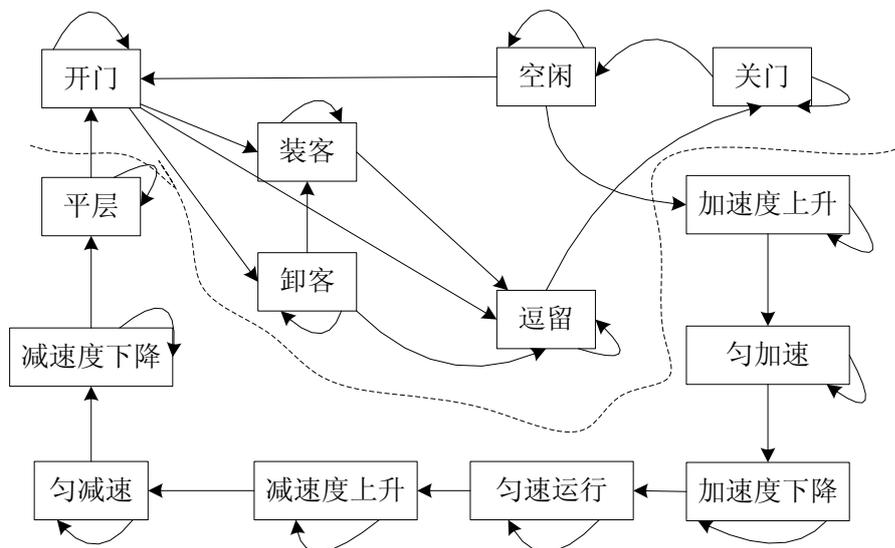


图 5.5 电梯状态转移图

电梯运行状况、其特有的一些特性、系统的不确定因素共同决定电梯状态，从图 5.5 中可以看出各个状态之间具有的逻辑关系。如加速度上升总是在空闲以后，匀速运行总是在加速度下降之前。电梯的门机控制系统控制电梯的开门、关门状态。当电梯没有指定任务时，会处于空闲状态。装客状态和卸客状态跟乘客乘梯习惯及大楼的客流量密切相关。电梯的逗留状态是指电梯开关门之间的时间，由电梯的服务对象、对象的进出动作决其逗留时间的长短。匀加速、匀减速和匀速运行状态的持续时间取决于电梯轿厢负载重量和电梯启动楼层与下一个停靠楼层的距离。电梯加速度、减速度上升和下降的状态持续时间由最大加速度、加速度变化率决定。

本文设计的群控仿真软件，能够动态显示的电梯群的调度过程，包括电梯的上行、下行、开门和关门等动态动作，每一部电梯的运行逻辑均采用单独的线程进行控制，

其动态显示效果如图 5.6 所示。

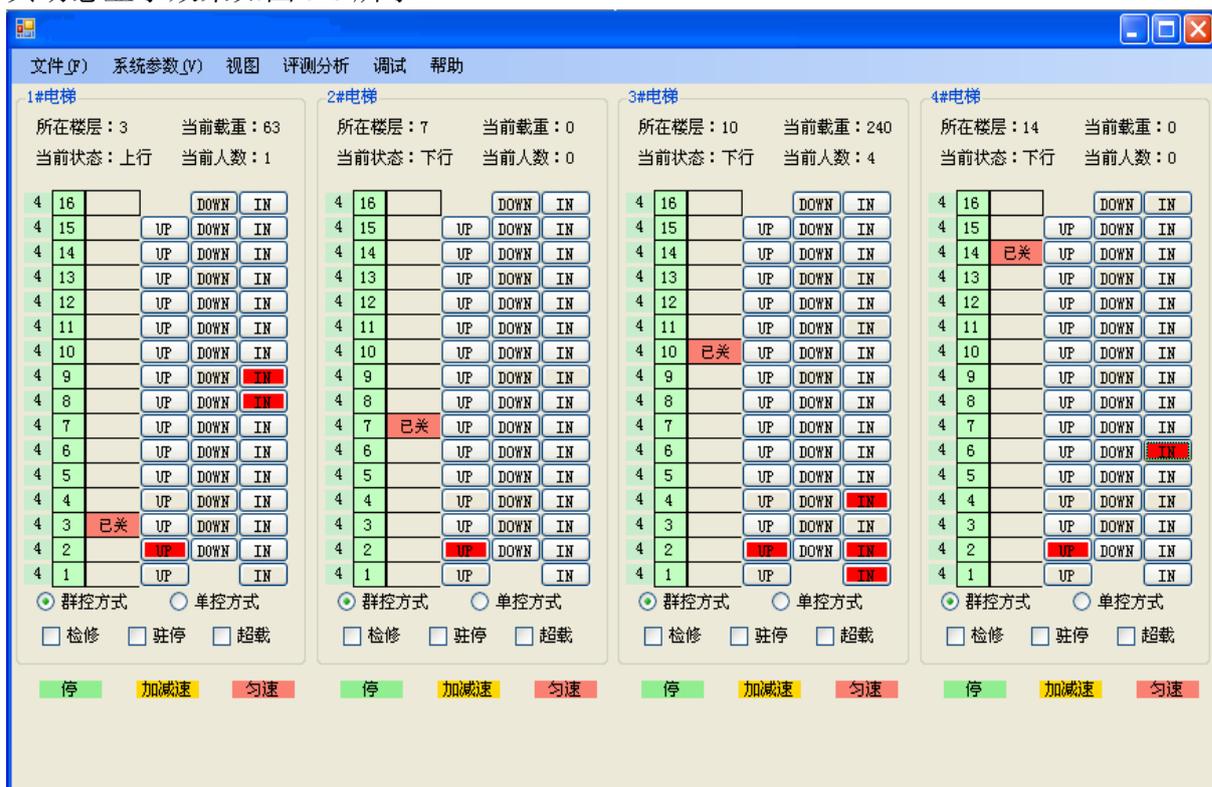


图5.6 软件动态显示界面

5.3.2 电梯构件化

为了构建电梯群，本文首先通过构件化的方式对单个楼层（Single Level）进行抽象，然后把多个楼层对象进行组建成单部电梯（Single Elevator），最后把多部电梯组建成一个电梯群（EGCS）。构件加载流程如图5.7所示。

以下是对控件进行设计时所需设置的一些属性。

1) 单层SingleLevel控件

单层SingleLevel控件所含信息包括：楼层高度、楼层号、电梯轿厢、上召信号、下召信号、内召信号。

2) 单部电梯SingleElevator控件

单部电梯SingleElevator控件所含信息包括：楼层数、最大额定载重(kg)、最大额定人数(人)、开/关门时间(s)、开门保持时间(s)、加速度 (m/s^2)、减速度(m/s^2)、匀速度 (m/s)、加速度能耗、匀速度能耗 (千瓦·时)、减速度能耗 (千瓦·时)。

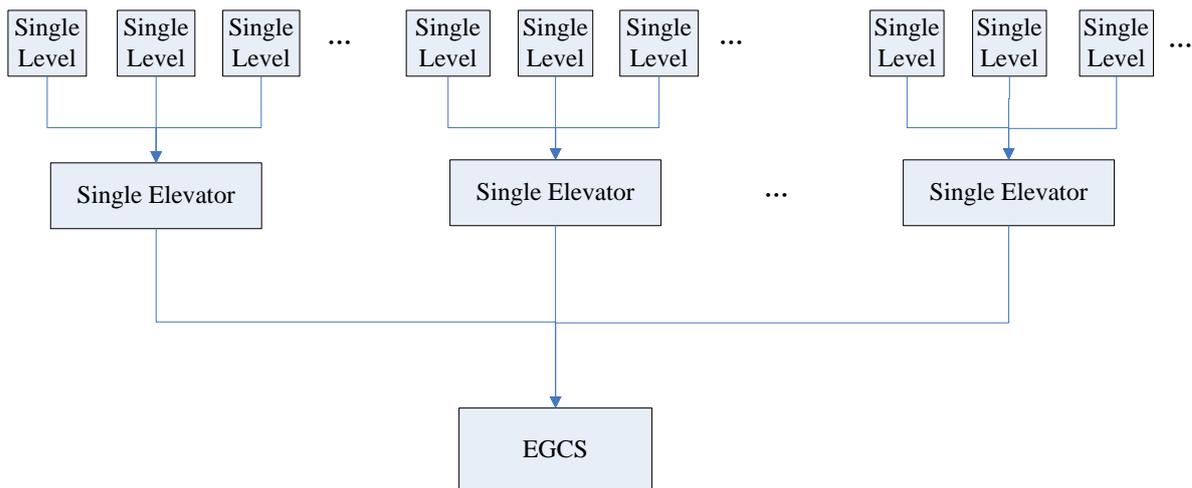


图5.7 构件加载流程

5.3.3 电梯运行控制

电梯运行模块按照一定的运行逻辑和运行规律，完成群控系统分派的应答任务，将乘客从起始楼层运送到目的楼层。电梯运行控制模块，根据电梯的内外召唤，由单梯内部检测机制中的单梯召唤检测函数`check_btn(Elevator ele)`实现，它把所有需要响应的Up, Down, In进行归类，上行逻辑的放于`allupList`；下行逻辑的放于`alldwnList`，其流程如图5.8所示；单部电梯响应机制可分为外部召唤响应机制和内部召唤响应机制，其流程图如图5.9所示，该流程图详细介绍了电梯控仿真机制。

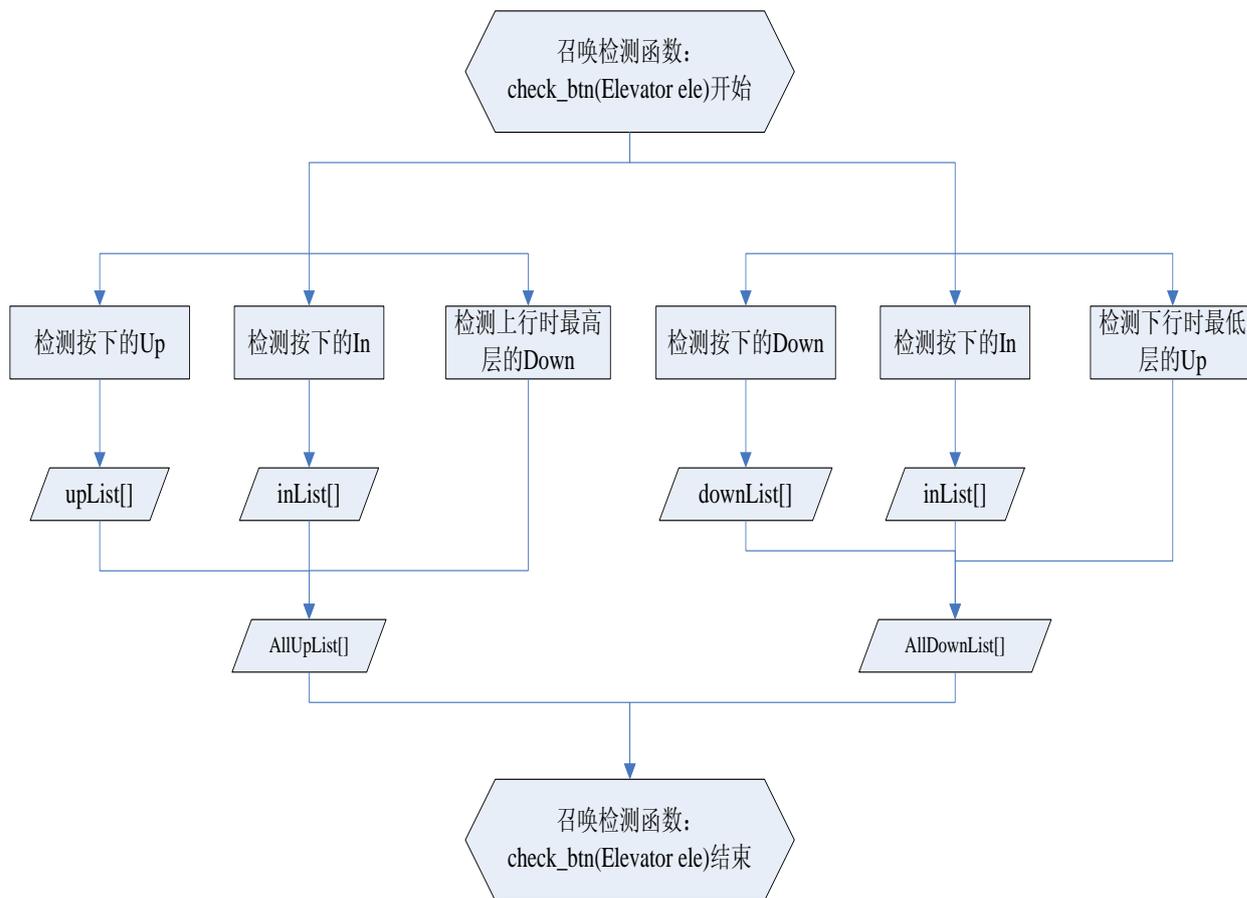
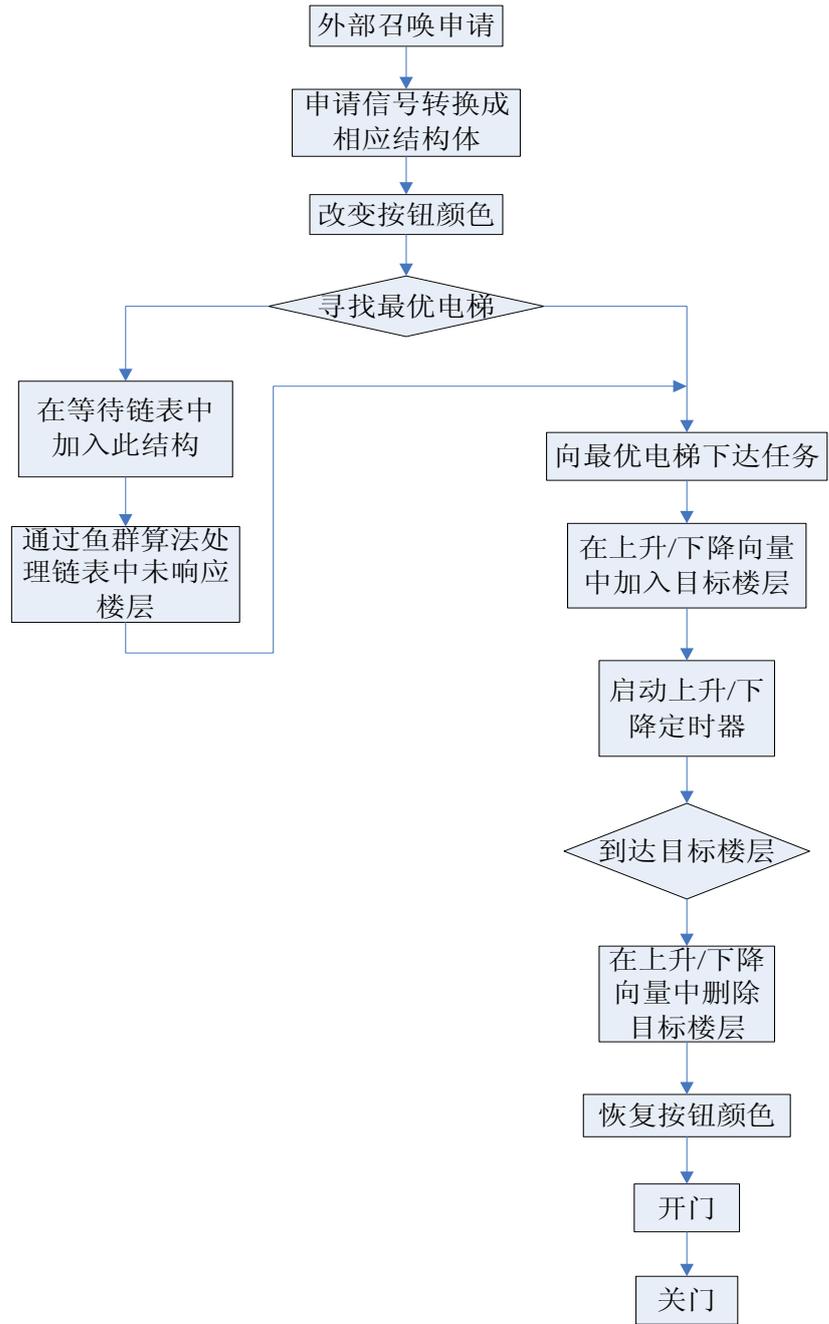
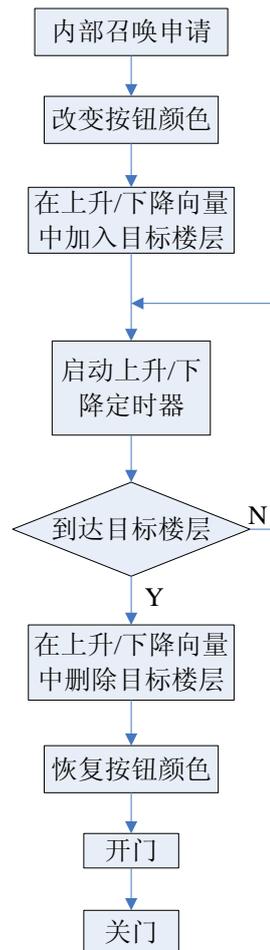


图5.8 单梯内部检测机制



(1) 外部召唤响应机制流程图



(2) 内部召唤响应机制流程图

图5.9 单部电梯响应机制

对于当前电梯处于静止状态，有新的召唤产生时，则电梯要选择方向，其流程如图5.10所示。经选择方向后，如果本层有召唤，则必须清除相应的召唤，如上行，则要清除向上召唤和当前层的内部召唤；反之下行，则要清除向下召唤和当前层的内部召唤。

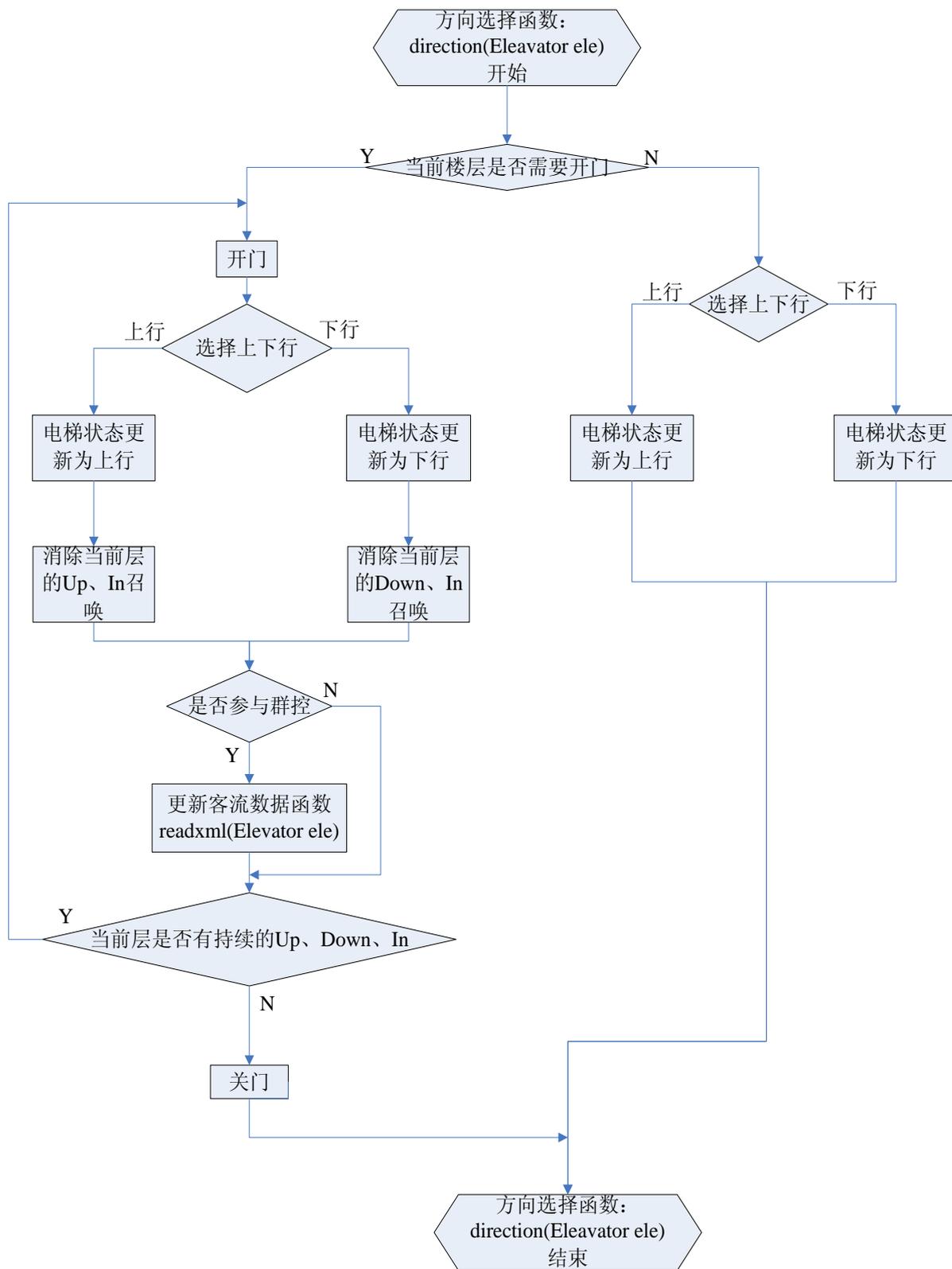


图5.10 方向选择机制

对于电梯的上行运行逻辑流程如图5.11所示，本文构件了rise函数，其主要逻辑判断步骤如下：

- (1) 循环检测是否已经响应全部召唤，是，则电梯驻停；否，则执行步骤2；
- (2) 是否继续向上，否，则电梯状态修改为向下并返回；是，则执行步骤3；
- (3) 是否上一楼层需要开门，否，则继续上行；是，则电梯状态修改为减速，到达上一楼层开门。

对于电梯的下行运行逻辑，与电梯的上行运行逻辑类似，本文不再叙述。

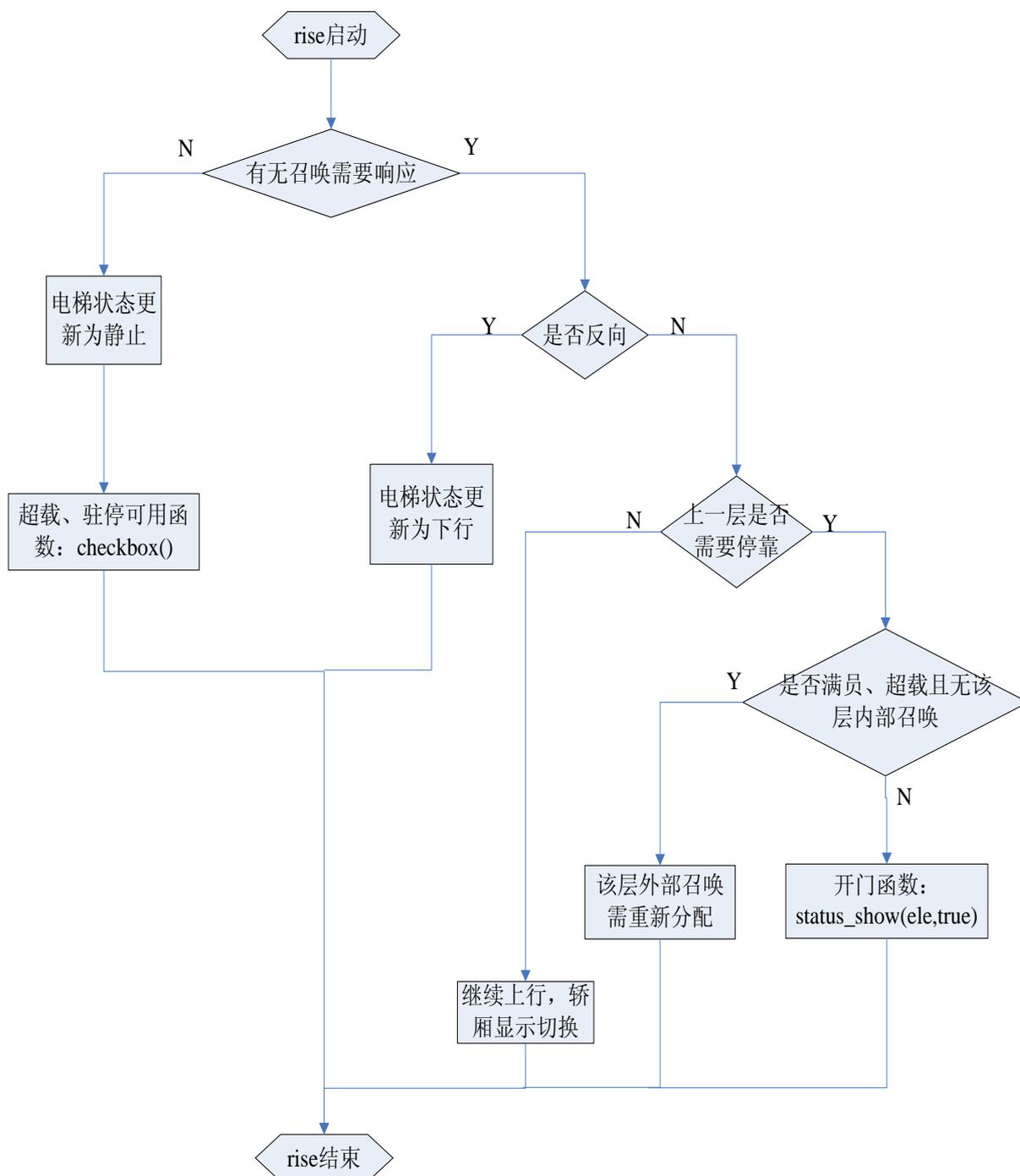


图5.11 向上运行逻辑

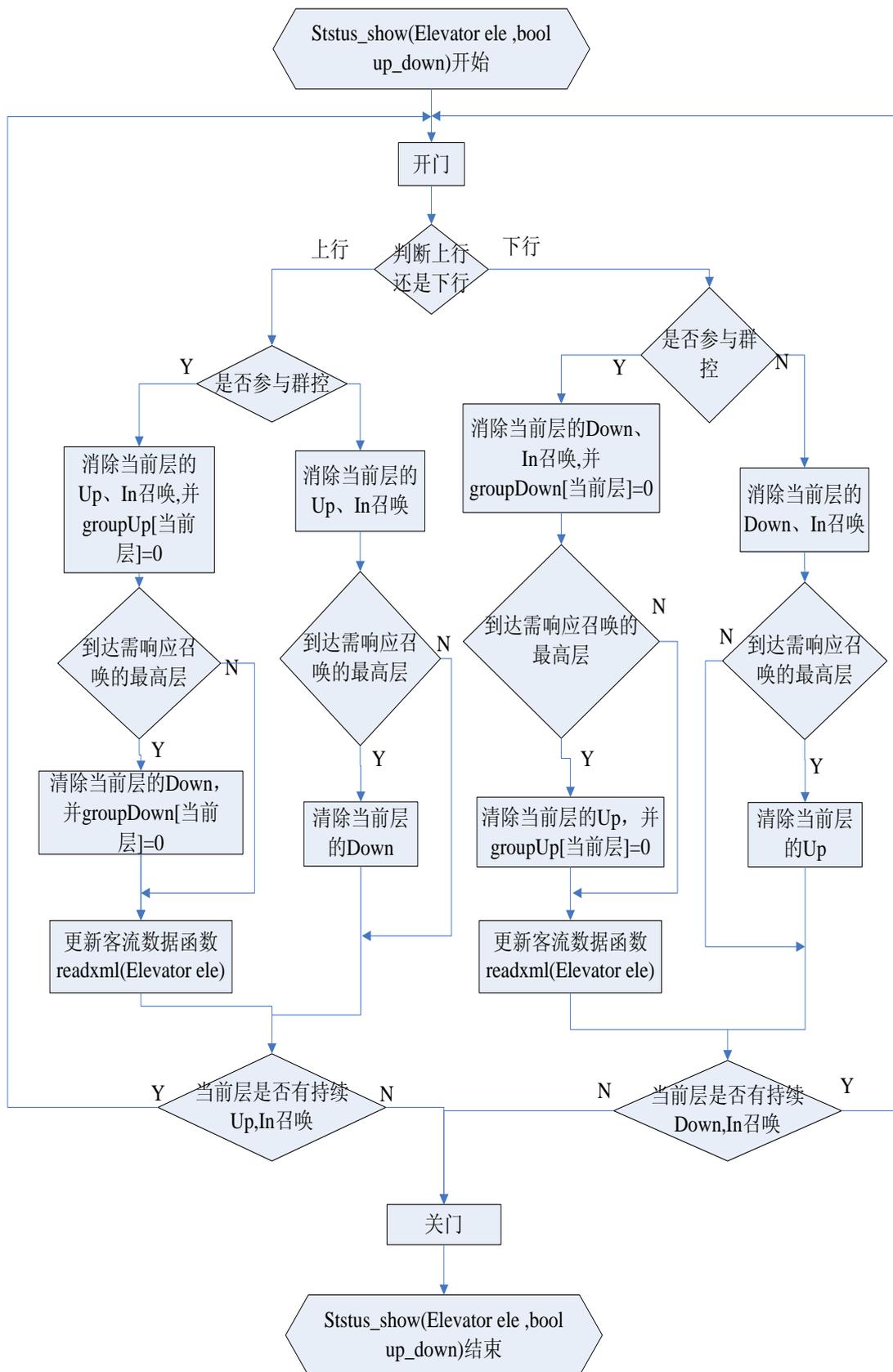


图5.12 开门状态更新

当电梯轿厢经过所要响应楼层时，则需要更新所在楼层的召唤。上行时，则清除向上召唤，和此层的内部召唤；下行时，则清除向下召唤，和此层的内部召唤；且当是群

控状态时需要记录乘客进出轿厢时的时间、人数等信息。开门状态更新流程如图5.12所示。其中更新客流数据函数为readxml(Elevator ele)，其主要作用就是记录电梯运行过程中的派梯运输乘客状况，同时更新电梯内的人数、轿厢内载重等信息，其更新流程如图5.13所示。

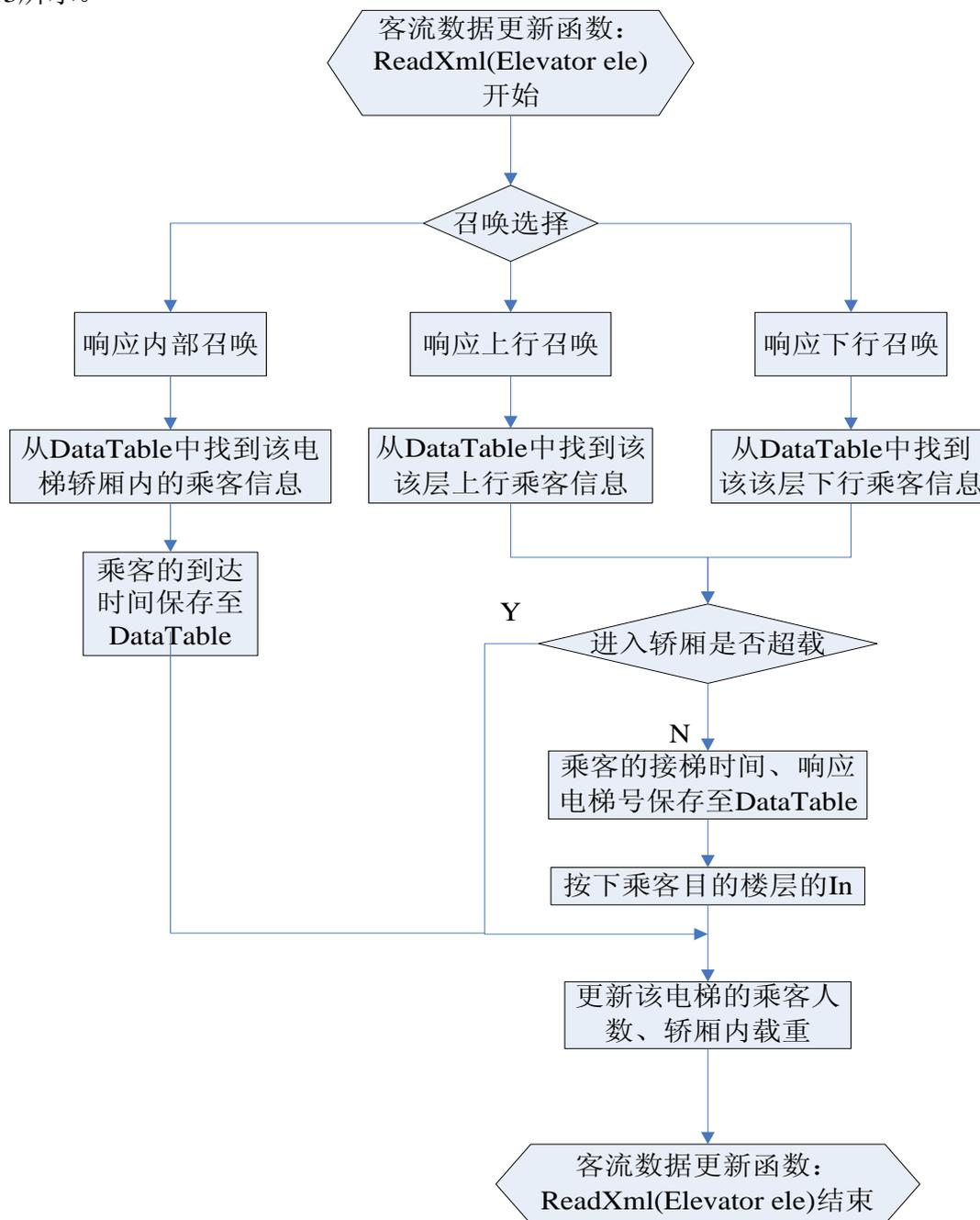


图5.13 客流数据更新流程图

5.3.4 数据通信

电梯是载人服务工具，需要实时采集乘客的乘梯要求和电梯系统本身的状态等输入信息，也需要实时发出派梯指令等输出信息，这就是数据通信模块的功能。

群控系统不仅需要从梯群内的电梯控制器采集电梯的轿厢内指令和当前状态，如当前运行方向，当前位置，当前轿厢内负荷(人数)以及运行状态；还需要收集厅外的召

唤信号。在群控算法做出派梯决策后，向电梯发出厅外召唤任务指令，让其去应答服务。

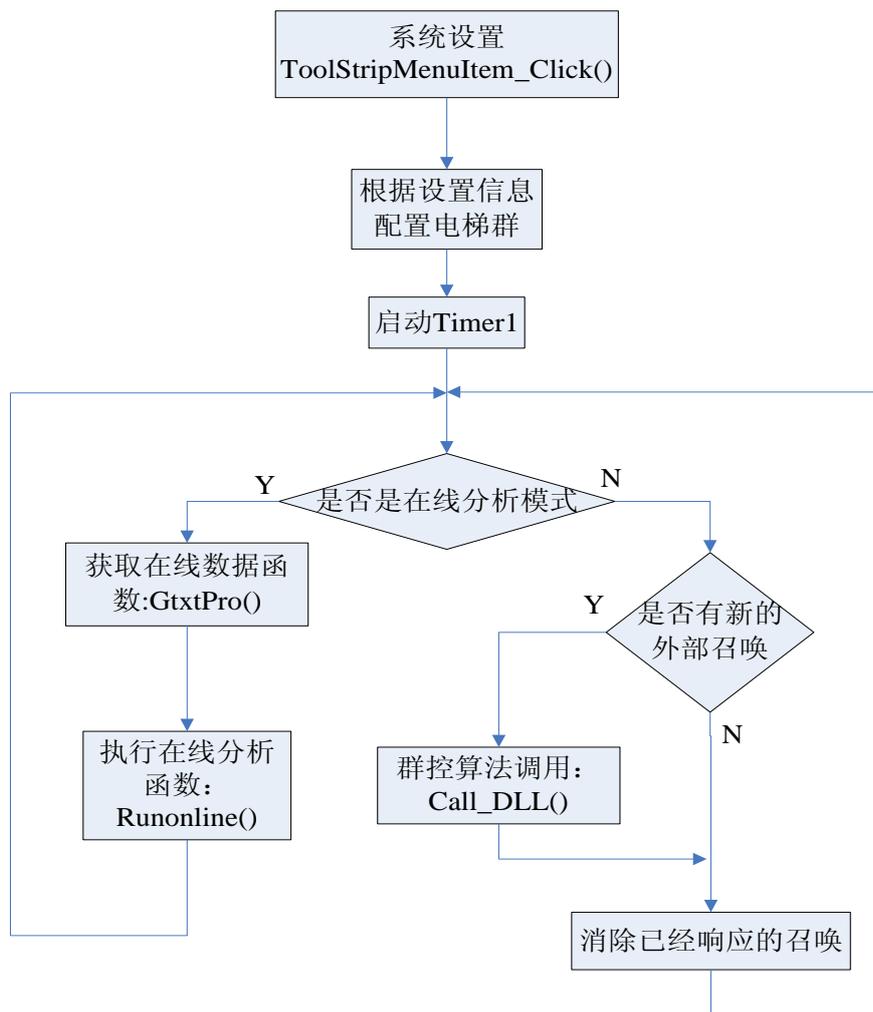


图 5.14 定时检测流程

由于在 PC 机上仿真电梯群控系统，数据通信变得相对简单。数据通信模块的存在，是为了用来更好地模拟实际中的电梯群控系统的运行。实际中的群控系统通常周期性地采集系统信息，周期性地发送任务指令。数据通信的间隔周期取决于电梯系统的硬件通信能力和群控算法的复杂程度。

本文的数据通信模块采用定时器的方式完成，定时器函数 `timer1_Tick(object sender, EventArgs e)` 流程如图 5.14 所示，其用途是：

- 1) 当模拟执行被选中时，有新的 Up\Down 召唤产生，则调用群控算法函数 (`Call_DLL()`) 进行派梯。
- 2) 已经响应此楼层的 Up(Down) 召唤时，则将所有同层的 Up(Down) 召唤都清除。

当按下模拟执行按钮时将触发读取客流信息函数 `readPassenger()` 其功能是，从客流文件中读取乘客的信息，定时按下呼叫楼层，其流程如图 5.15 所示。其中，启动线程调用 `CalcProcessStart` 函数用于实时刷新评测数据。

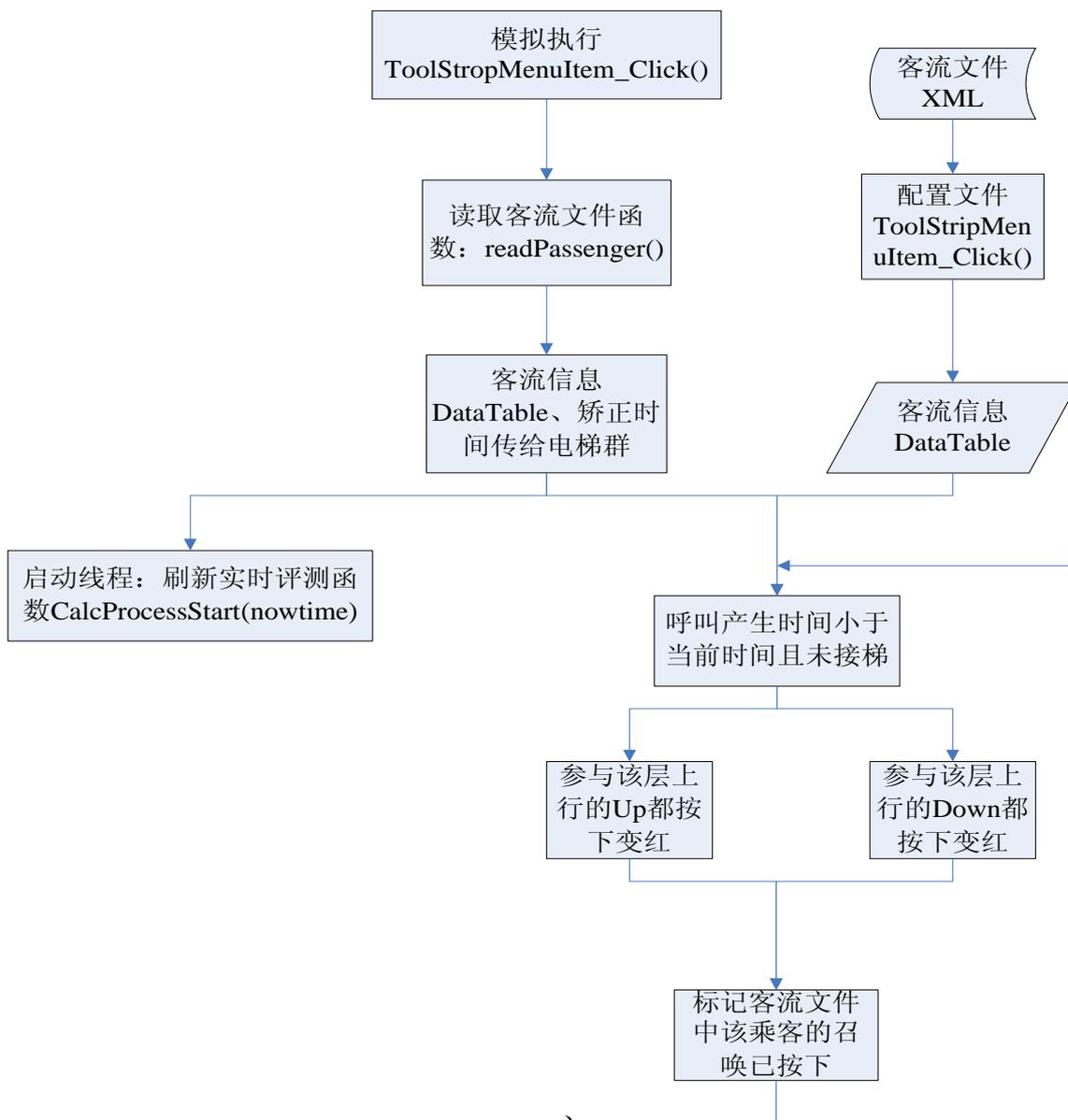


图 5.15 客流信息函数读取流程

5.4 电梯群控评测模块

指标评测模块的作用是对电梯的性能指标进行计算,对服务质量进行统计,评估乘客的满意度。以下是电梯平均候梯时间、平均乘梯时间、能耗的计算方法。

5.4.1 平均候梯时间

从乘客按下外召信号起,到电梯到达该楼层响应其呼梯信号的这段时间称为候梯时间。候梯时间函数定义为 $f_w(i,k)$ ^[26],表示电梯 k 响应第 i 个外召信号时所花的乘客候梯时间。电梯 k 响应所有分配的 N 个外召信号产生的平均候梯时间代价函数为:

$$\frac{1}{N} \sum_{i=1}^N f_w(i,k) \quad (5.2)$$

5.4.2 平均乘梯时间

从乘客进入电梯起,到电梯到达目的楼层,乘客离开电梯的这段时间称为乘梯时间。候梯时间函数定义为 $f_j(i,k)^{[26]}$,表示电梯 k 响应第 i 个外召信号时所花的乘客乘梯时间。因此,电梯 k 响应所有分配的 N 个外呼信号产生的平均乘梯时间代价函数为:

$$\frac{1}{N} \sum_{i=1}^N f_j(i,k) \quad (5.3)$$

5.4.3 能耗

电梯的总能耗包括运行阶段的能耗、电梯轿厢内照明设备能耗、控制器能耗、显示器能耗、通风设备的能耗以及内其他设备的能耗。

$$E = E_u + E_d + E_r \quad (5.4)$$

E 为电梯总能耗, E_u 为向上运行的总能耗, E_d 为向下运行的总能耗, E_r 为控制器和显示器、电梯轿厢内照明和通风系统以及电梯内其他设备的总能耗。

从简化的典型曳引系统出发,本文通过分析曳引系统的功能关系,从而建立典型运行阶段的能耗数学模型,即电梯轿厢负载、行程、运动状态与运行阶段总能耗之间的函数关系。并再讨论运行方式对运行阶段能耗的影响。如图5.16,即对电梯进行的受力分析。

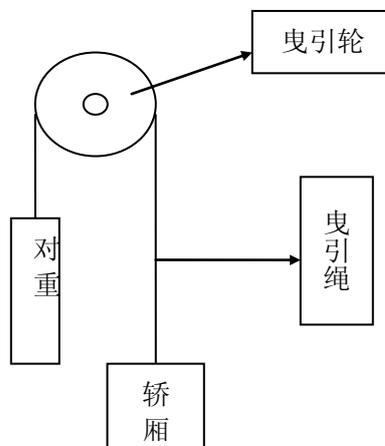


图 5.16 电梯简化受力分析

对轿厢进行受力分析,平衡时 $F_1 + F_2 = (M + m_i)g$ 。期中, F_1 为曳引力, F_2 为对重的重力(对重的重量等于轿厢自重加上电梯额定载荷的一半)。

(1) 电梯上行阶段,当 $m_i < \frac{1}{2}m$ 时,即 $m_{cwt} > M + m_i$,电梯靠对重产生对轿厢向上的拉力,不需要消耗很多电能去拉动轿厢上行,此时电梯的能耗可以忽略不计。期中, m_{cwt} 为电梯的对重, m 为额定载重, m_i 为轿厢内乘客总质量。

(2) 电梯上行阶段,当 $m_i \geq \frac{1}{2}m$ 时, $F_1 = (M + m_i - m_{cwt})g$ 。

(3) 电梯下行阶段，当 $m_i > \frac{1}{2}m$ 时，即 $m_{cwt} < M + m_i$ ，轿厢依靠其自身的重力即能下行，曳引轮控制速度，此时电梯消耗的电能可忽略不计。

(4) 电梯下行阶段， $m_i \leq \frac{1}{2}m$ 时， $F_1 = (m_{cwt} - M - m_i)g$ 。

将向上运行阶段分为匀速阶段和变速阶段，向上运行阶段能耗为： $E_u = E_{ue} + E_{uc}$

其中， E_{ue} 为向上匀速运行能耗， E_{uc} 为向上变速运行能耗。

(1) 向上匀速运行能耗分析

$$E_{ue} = \sum_{i=1}^n E_{ue}(i) \quad (5.5)$$

期中，n为向上匀速运行段数

$$E_{ue}(i) = F_i S_i = (M + m_i - m_{cwt})g S_i \quad (5.6)$$

期中， F_i 为轿厢和乘客重力， S_i 为电梯匀速运行的位移， M 为电梯轿厢质量， m_i 为轿厢内乘客总质量， m_{cwt} 为电梯对重， m 为电梯额定载重， g 为引力常量。

(2) 向上加速阶段能耗分析

根据以下关系

$$v = dx/dt \quad (5.7)$$

$$a = dv/dt \quad (5.8)$$

$$jerk = da/dt \quad (5.9)$$

$$Yank = mj = dF/dt \quad (5.10)$$

可得到

$$a = \int jdt = jt + c_1 (t=0, a=0, c_1=0) \quad (5.11)$$

$$v = \int adt = \int jtdt = \frac{1}{2}jt^2 + c_2 (t=0, v=0, c_2=0) \quad (5.12)$$

$$s = \int vdt = \int \frac{1}{2}jt^2 dt = \frac{1}{6}jt^3 + c_3 (t=0, x=0, c_3=0) \quad (5.13)$$

$$F = \int Ydt = \int mjdt \quad (5.14)$$

电梯经加加速，匀加速，可以达到最大加速度和额定速度，如图5.17所示。

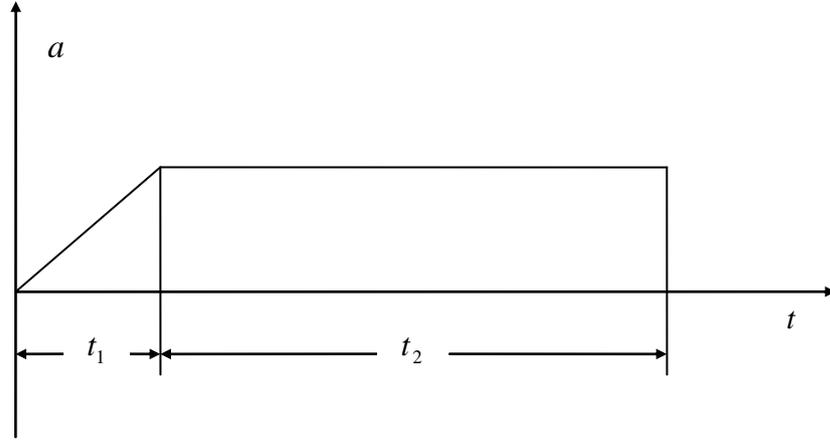


图 5.17 加速分析

假设：额定速度 $v = 2.5m/s$ ，加速度 $a = 1m/s^2$ ，加加速度 $j = 1.6m/s^3$

1) 加加速度阶段

当 $0 \leq t \leq t_1$ 时，即电梯处于加加速时：

在最初的 $t_1 = \frac{a}{j} = 0.625s$ ，电梯轿厢的速度将达到 $v_1 = \frac{1}{2}jt_1^2 = 0.3125m/s$ ，此时电

梯向上运行 $s_1 = \frac{1}{6}jt_1^3 = 0.0651m$

$$E_{uj} = \int_0^{t_1} F_1(t)S_1(t) = \int_0^{t_1} (M + m_i - m_{cwt})(jt + g)\left(\frac{1}{2}jt^2\right)dt \quad (5.15)$$

2) 加速度阶段

当 $t_1 \leq t \leq t_1 + t_2$ 时，即电梯处于匀加速时：

电梯轿厢以 $a = 1m/s^2$ 作匀加速度运动，在 $t_2 = \left(v - \frac{a^2}{2j}\right)/a = 2.1875s$ 内，从 $0.3125m/s$ 增

加到 $v = 2.5m/s$ ；电梯运行距离 $s_2 = v_1t_2 + \frac{1}{2}at_2^2 = 3.0762m$

整个加速过程将持续 $0.625 + 2.1875 = 2.8125s$ ，期间电梯上升 $s = s_1 + s_2 = 3.1413m$ 。

$$E_{ua} = \int_{t_1}^{t_1+t_2} F_2(t)S_2(t) = \int_{t_1}^{t_1+t_2} (M + m_i - m_{cwt})(a + g)(v_1 + at)dt \quad (5.16)$$

电梯相邻层间停靠时。达不到额定速度但达到最大加速度时的能耗为 E_o 。

相邻层停靠时电梯达不到额定速度，也达不到最大加速度时的能耗为 E'_o 。由计算得

$E'_o \ll E_o$ 因此 E'_o 可忽略不计。

综上所述，电梯加速向上的总能耗为：

$$E_{uc} = \sum_{i=1}^{N_1} (E_{uj} + E_{ua}) + \sum_{i=1}^{N_2} E_o \quad (5.17)$$

其中， N_1 为向上做一般加速运行的次数， N_2 为向上邻层停靠次数。

3) 向上减速阶段能耗分析

当临近目的楼层时，匀速向上运行的电梯，便要开始减速。这一阶段的能耗即 $E_a \ll E_{ue}$ ； $E_a \ll E_{uc}$ 。由于匀速或加速上升阶段时的能耗远远大于此阶段的能耗，所以该阶段的能耗可以忽略不计，因此电梯变速向上阶段的能耗为：

$$E_u = \sum_{i=1}^{N_1} (E_{uj} + E_{ua}) + \sum_{i=1}^{N_2} E_o + \sum_{i=1}^{N_3} E_a \approx \sum_{i=1}^{N_1} (E_{uj} + E_{ua}) + \sum_{i=1}^{N_2} E_o \quad (5.18)$$

期中， N_3 为向上减速次数。

向下运行阶段能耗分析即为向上运行的逆行为，在此省略。

电梯轿厢内照明设备能耗、控制器能耗、显示器能耗、通风设备的能耗以及内其他设备能耗的建模

$$E_r = Ct \quad (5.19)$$

其中，电梯运行时间为 t ，电梯轿厢内照明设备能耗、控制器能耗、显示器能耗、通风设备的能耗以及内其他设备能耗常量为 C 。

综上所述，电梯运行阶段的总能耗为：

$$\begin{aligned} E &= E_u + E_d + E_r \\ &= \sum_{i=1}^{N_1} (E_{uj} + E_{ua}) + \sum_{i=1}^{N_2} E_o + E_d + Ct \end{aligned} \quad (5.20)$$

能耗计算流程图如图5.18所示，首先对参与群控的电梯进行序号的初始化，接着从第一部电梯开始依次计算每部电梯的能耗。对于第 k 部电梯能耗的计算，系统首先读取该梯当前的运行数据，并开设计算暂存区，接着根据电梯所处运行阶段的计算公式，计算该梯响应召唤需要消耗的能量，再根据公式 (5.18)，将上述结果逐个累加至暂存区，暂存区中最后存储的数据即为第 k 部电梯的总能耗。最后将每部电梯的能耗相加，即得电梯群组的总能耗。

单部电梯能耗计算关键代码如下：

```
public void calcEnergy(Elevator ele, bool ing)
{
    for (int i=1; i < this.elvator.levelNum; i++)
    {
        distance += Convert.ToInt32(ele.singleLevel[j].tbhigh.Text); //运行距离
    }
    if (tkg > (1 / 2 * this.elvator.kg))
    {
        tmpE += tkg * (distance - 3.14) * g; //3.14是加速距离，匀速能耗
        tmpE += tkg * (1 / 8 * g * this.elvator.aacceleration *
Math.Pow(Convert.ToDouble(this.elvator.aacceleration / this.elvator.acceleration), 4) + 1 / 6 * Math.Pow(g,
2) * Math.Pow(Convert.ToDouble(this.elvator.aacceleration / this.elvator.acceleration), 3)); //加加速阶段能耗
        double t2 = (this.elvator.even_pace - Math.Pow(this.elvator.acceleration, 2) / 2 /
this.elvator.aacceleration) / this.elvator.acceleration;
        double v1 = 1 / 2 * Math.Pow(this.elvator.acceleration, 2) /
this.elvator.aacceleration;
        tmpE += tkg * (this.elvator.acceleration + g) * (v1 * t2 + 1 / 2 * this.elvator.acceleration * Math.Pow(t2, 2)); //加
速度能耗
    }
}
```

```

//计算邻层停靠能耗
for (int i=1; i < this.levator.levelNum; i++)
{
    if (upE[i].status == 1 && upE[i+1].status == 1)
    {
        tmpE += E0;
    }
}
}

```

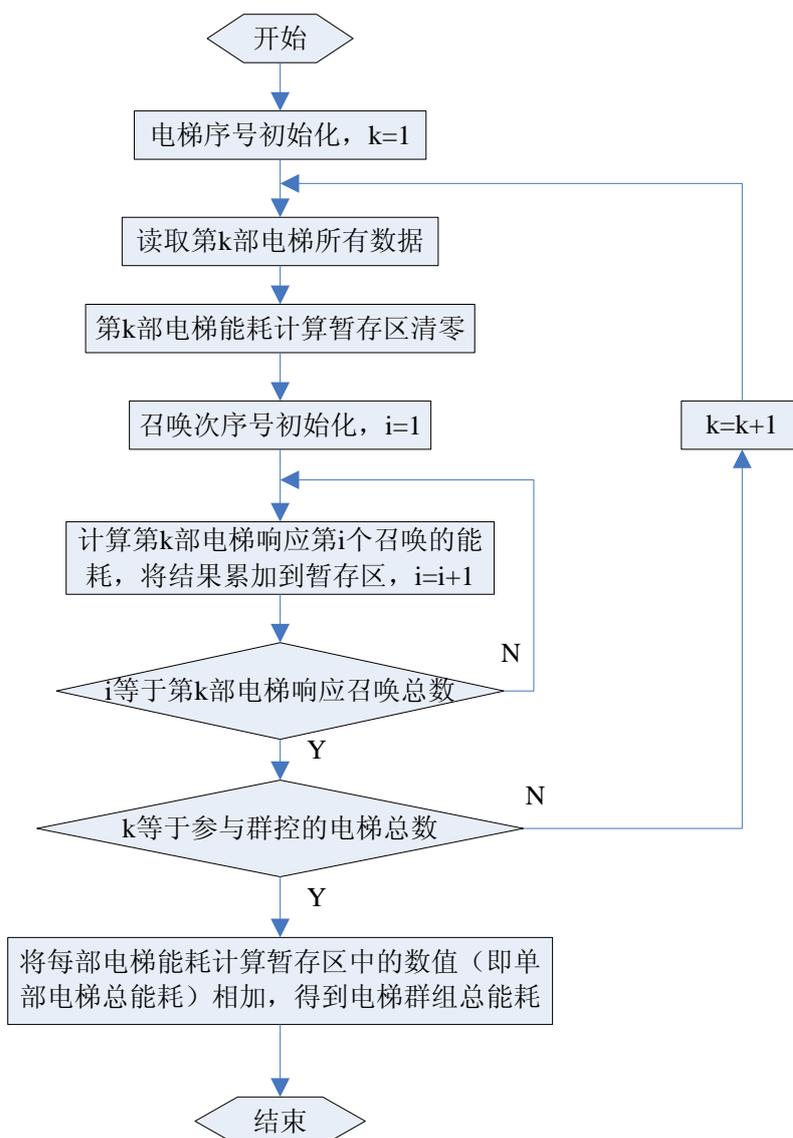
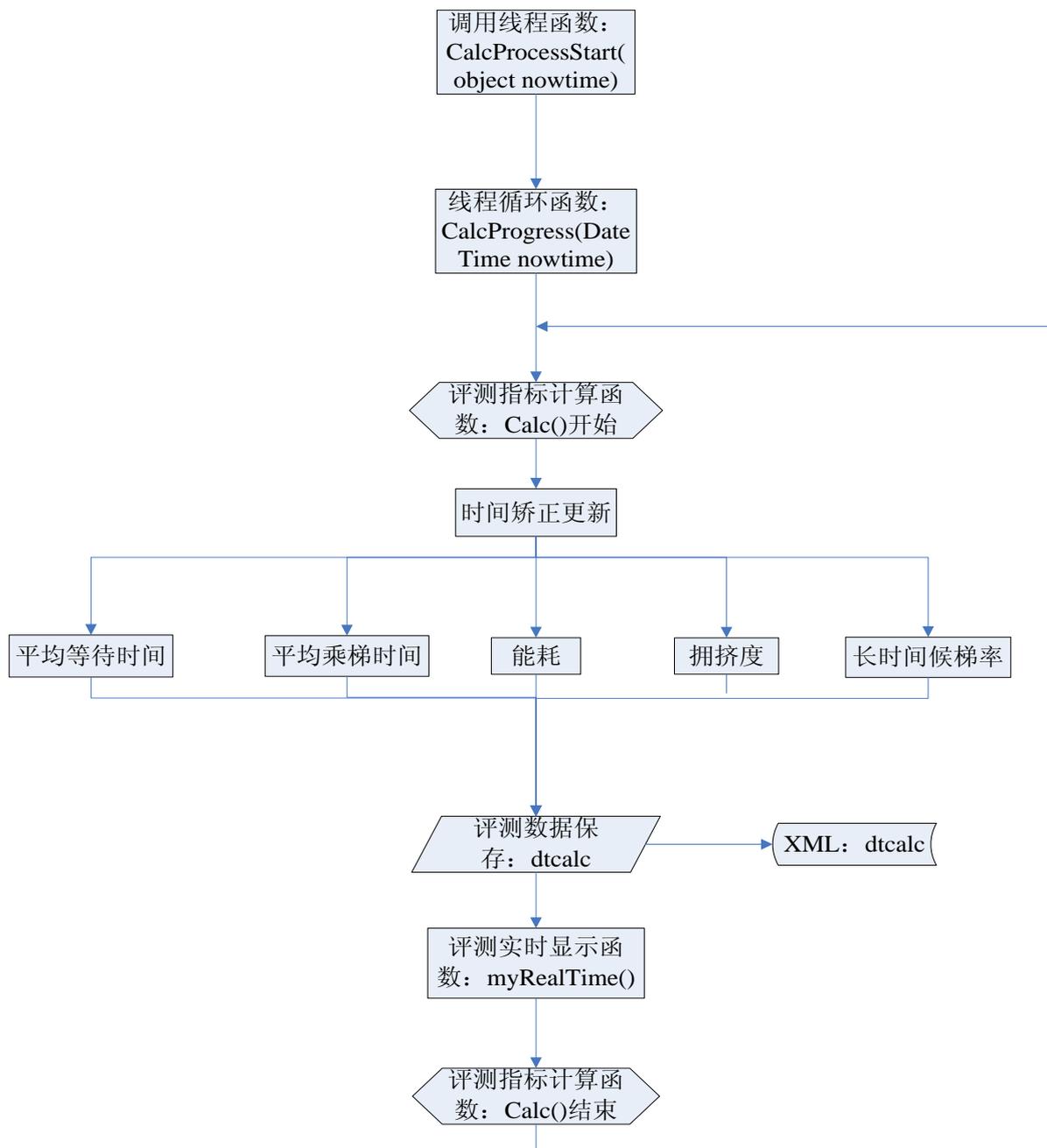


图5.18 能耗计算流程图

5.4.4 评测显示

在本仿真平台中,除主要对乘客的平均候梯时间、平均乘梯时间、长时间候梯率和电梯能耗进行统计计算外,还可以针对具有学习能力的群控算法,提供周期性学习的电梯运行数据,以便群控系统调整群控参数,或者改变群控策略,为乘客提供更加快捷的服务。乘客的平均候梯时间、平均乘梯时间和能耗,以5分钟为计算单位。群控系统的周期性学习,可以分每天、每周、每月进行。评测模块从电梯及乘客的历史

数据，获取如产生时间、接梯时间、到达时间等计算平均候梯时间、平均乘梯时间等信息。其流程如图 5.19 所示。



5.19 评价指标计算流程

鉴于本文所开发的仿真平台是采用 VS2008 实现，顾涉及的评测显示采用 .NET 3.5 框架下的图表控件 MSChart 实现。如图 5.20 为实时评测显示图。

评测显示关键代码如下：

```

private void myRealTime()//实时显示评测指标
{
    string[] xValues = new string[dtcalc.Rows.Count];//时间轴
    double[] yValues = new double[dtcalc.Rows.Count];//数据轴
  
```

```
for (int j = 1; j < 5; j++)//评测指标的类型
{
    for (int i = 0; i < dtcalc.Rows.Count; i++)
    {
        xValues[i] = Convert.ToString(dtcalc.Rows[i][0]);
        //(dtcalc 中的存放的数据为评测指标计算后的结果
        yValues[i] = Convert.ToDouble(dtcalc.Rows[i][j]);//纵坐标数据源
    }
    switch (j)
    {
        //平均候梯时间
        case 1: rt.chtawt.Series["AWT"].Points.Clear();
        rt.chtawt.Series["AWT"].Points.DataBindXY(xValues, yValues); break;
        //平均乘梯时间
        case 2: rt.chtajt.Series["AJT"].Points.Clear();
        rt.chtajt.Series["AJT"].Points.DataBindXY(xValues, yValues); break;
        //长时间候梯时间
        case 3: rt.chtlwp.Series["LWP"].Points.Clear();
        rt.chtlwp.Series["LWP"].Points.DataBindXY(xValues, yValues); break;
        //能耗
        case 4: rt.chtrc.Series["RC"].Points.Clear();
        rt.chtrc.Series["RC"].Points.DataBindXY(xValues, yValues); break;
    }
}
}
```

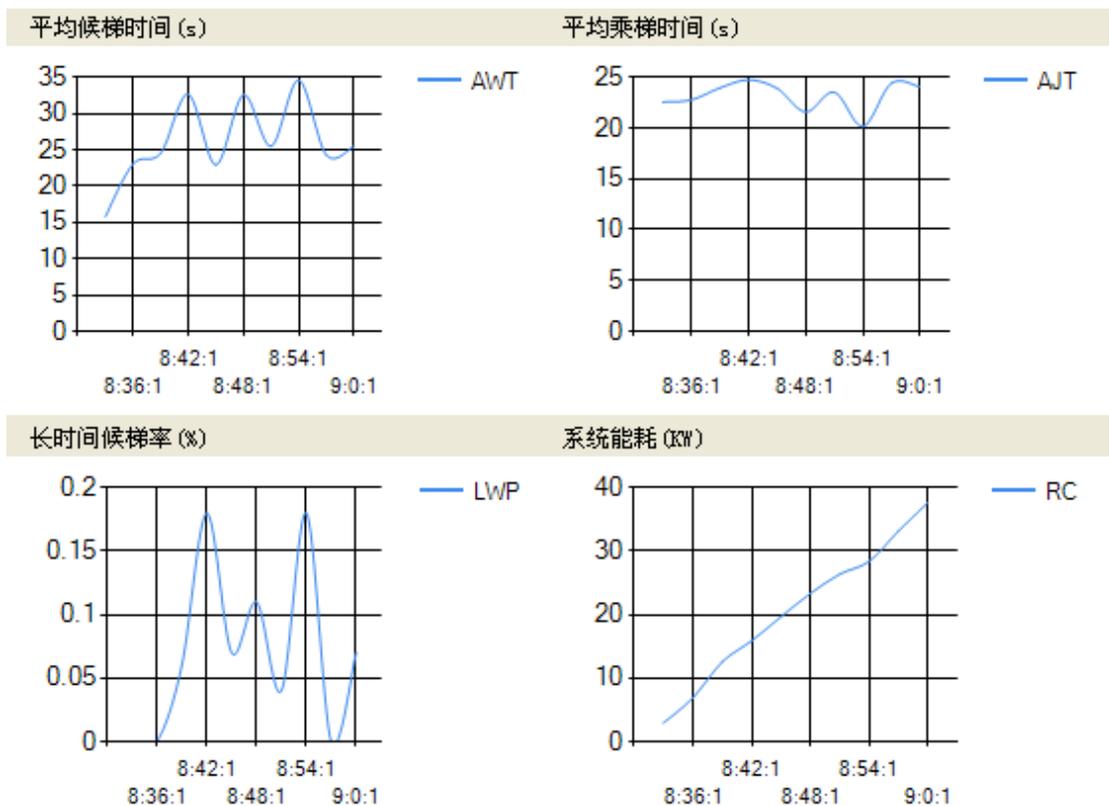


图5.20 评测显示

5.5 群控调度算法模块

5.5.1 调度算法公共接口

为了对不同的群控算法进行评测，本文设计了统一的群控算法公共接口。本文的仿真平台由 C# 编码实现，群控算法则采用 C++ 动态链接库 (Dynamic Link Library) 方式编码。

对于 C# 调用 C++ DLL 问题，由于 C# 中使用 `DllImport` 是不能像动态 `load/unload assembly` 那样，所以只能借助 API 函数了。在 `kernel32.dll` 中，与动态库调用有关的函数包括：

- ① `LoadLibrary` (或 MFC 的 `AfxLoadLibrary`)，装载动态库。
- ② `GetProcAddress`，获取要引入的函数，将符号名或标识号转换为 DLL 内部地址。
- ③ `FreeLibrary` (或 MFC 的 `AfxFreeLibrary`)，释放动态链接库。

它们的原型分别是：

```
HMODULE LoadLibrary(LPCTSTR lpFileName);
FARPROC GetProcAddress(HMODULE hModule, LPCWSTR lpProcName);
BOOL FreeLibrary(HMODULE hModule);
```

虽然可以用 `IntPtr hModule=LoadLibrary("Count.dll");` 来获得 DLL 的句柄，用 `IntPtr farProc=GetProcAddress(hModule, "_count@4");` 来获得函数的入口地址。但是，知道函数的入口地址后，如何调用这个函数就存在一些问题。因为在 C# 中是没有函数指针的，

没有像 C++ 那样的函数指针调用方式来调用函数，所以我们得借助其它方法。经过研究，发现可以通过结合使用 `System.Reflection.Emit` 及 `System.Reflection.Assembly` 里的类和函数达到目的。群控算法添加过程如图 5.22 所示。

DLL 的入口函数为：

```
int EGCS_DLL(int Button[],int Status[],int Layer[],int inn[][M],int people[],int X[],int
eenum, int layernum,int inpeo)
```

其中形参：

- 1) Button 用于存放电梯的所有外部召唤；
- 2) Status 用于存放电梯的状态；
- 3) Layer 用于存放电梯所在楼层；
- 4) inn 用于存放电梯内部召唤；
- 5) people 用于存放轿厢内人数；
- 6) X 用于存放最后的调度方案；
- 7) eenum 表示电梯数；
- 8) layernum 表示楼层数；
- 9) inpeo 表示最大轿厢人数荷载。

调用 DLL 时需要用到 `Call_DLL` 函数，其主要功能是将参与群控的信号进行编码，并传给群控算法 DLL 进行调度分配计算，同时对于静止的电梯需给定选择方向，流程如图 5.21 所示。最佳调度方案数存放于数组 `X[]`，见表 5.1。

表 5.1 最佳调度方案数组

下标	0	1	2	3	4	5	6	7	8
所派电梯	0	1	0	1	0	1	2	2	3

其中所派电梯 0 表示无电梯； i (1,2,3,4...) 表示第 i 部电梯去响应召唤；调度方案分拆函数 (`Btupdown`) 作用是将派梯方案分拆给每部电梯的上下行待响应链表。对表 5.1 进行分拆后的数据为表 5.2。

表 5.2 电梯上下行待响应链表

电梯号	待响应召唤	待响应召唤	待响应召唤	...
1#	1	3	5	
2#	6	7		
3#	3			
4#				

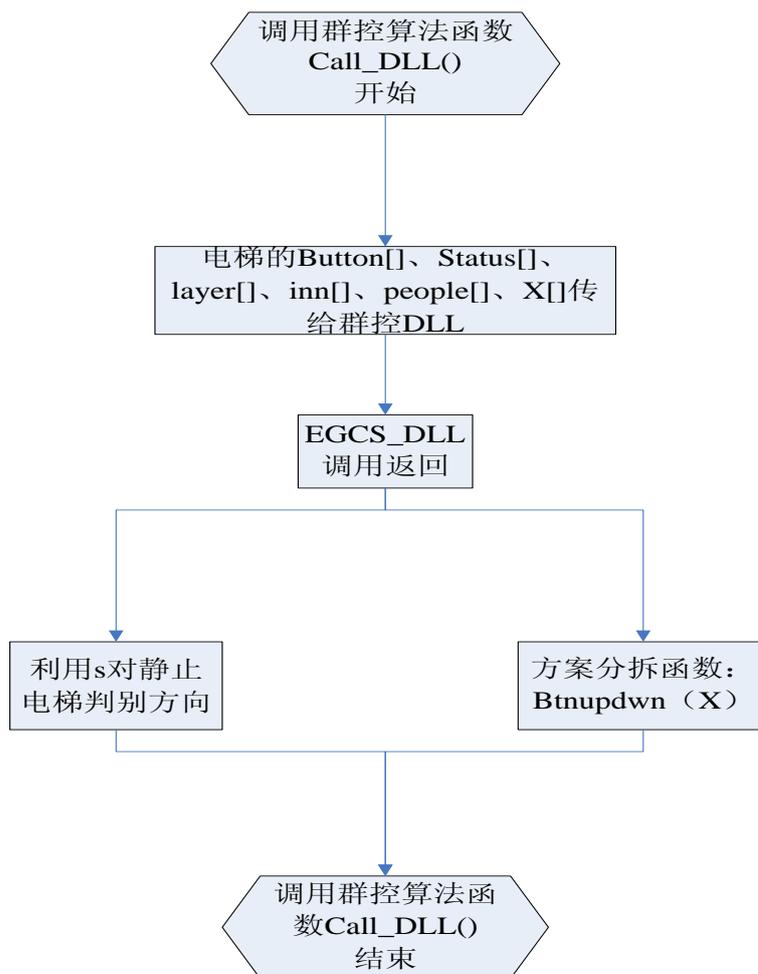


图5.21 Call_DLL函数流程



(1)选择需添加的算法文件

(2)添加后的群控算法库

图5.22 群控算法添加过程

5.5.2 积分算法

最小等待时间调度原则是根据所产生的层站召唤，预测各电梯应答时间，从中选择

应答时间最短的电梯去响应召唤。积分算法其实是乘客召唤信号的最小等待时间的一种优化方法，其流程见图 5.23。

积分算法的原则设计如下：

1) 距离积分 $D(i)$ ：

根据为了响应该召唤信号，电梯根据目前已登记的召唤信号，计算出运行多少层楼后到达该召唤楼层。通常每层楼为 1 分。

2) 指令或召唤积分 $O(i)$ ：

考虑到电梯响应每一个已登记的指令或召唤信号时需花去不少时间，所以在计算积分过程中，在计算每一个召唤信号对每台电梯的积分时，对于该电梯在与那个信号之间的每一个已登记的指令或已分配到的同向召唤，都给一个 3 分的积分。

3) 反向积分 $P(i)$ ：

考虑电梯运行中的顺向优先原则，在计算积分过程中，增加一个反向积分。

4) 超越积分 $E(i)$ ：

为了提高电梯的运行效率，减少电梯相互之间的超越现象，在计算每一个按钮对每台电梯的积分时，还要加上超越积分。通常，如果某台电梯在前方有一台同向运行的电梯时，更前方的召唤信号对后面一台电梯的积分要加 6 分。

5) 同向后方指令积分 $B(i)$ ：

对于某台电梯来说，新出现的同向召唤信号来不及处理（即同向后方召唤，例如电梯上行目前在 5 楼，6 楼出现上召信号，这时电梯已经过了最小减速距离，来不及减速去响应 6 楼的上召信号），需要给该电梯增加一个同向后方指令积分，该积分要大些，可设为 1000。

6) 累计积分 $AC(i)$ ：

根据各台电梯目前已经登记的召唤信号，计算各台电梯累计需要停层的次数，通常停层一次加 8 分。

7) 节能积分 $A(i)$ ：

为了提高节能效果，减少同时启动多台电梯去响应命令，在节能优先的情况下，为未启动的电梯增加一个节能积分 20 分，可根据需要调整该积分大小。

每一台电梯针对每一个召唤信息，根据以上原则累计一个总的积分。每一个召唤信号对应每台梯都有一个积分，相互比较每台电梯积分的大小，将该召唤的资格分配给具有最小积分的电梯。

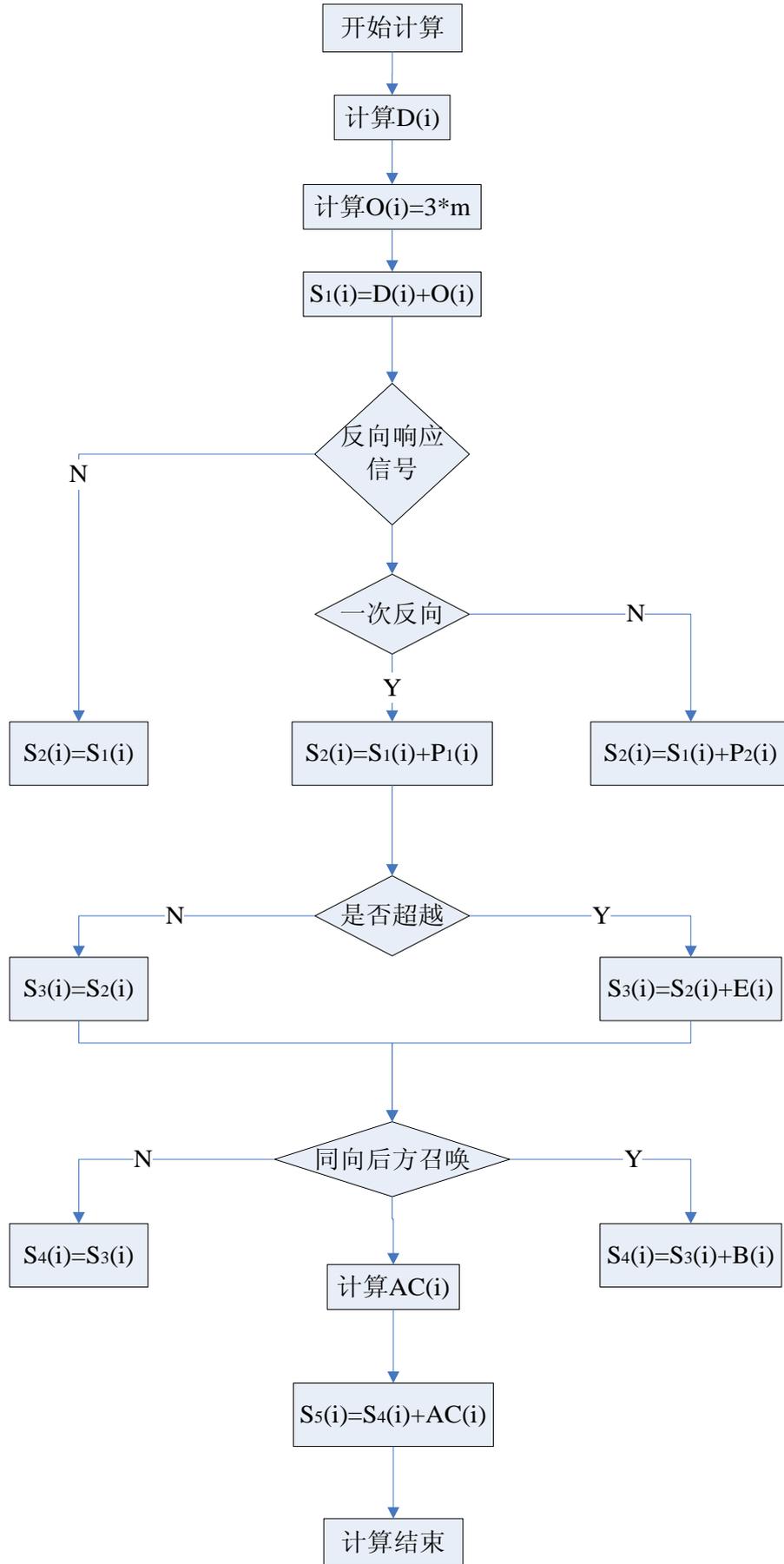


图5.23 积分算法流程

5.6 系统参数设置

系统提供一些默认参数供用户设置，可通过参数设置页面如图 5.24 进行设置。对参数取值范围进行一定的限制，如楼层数范围选取从 6 层至 156 层；电梯数目选取从 1 部到 16 部；每层楼高度可以不相同，楼层高度范围选取从 2m 至 5m；轿厢加速度，范围选取从 $0.5m/s^2$ 至 $1.5m/s^2$ ；加加速度：即轿厢加速度变化率，设定范围从 $0.5m/s^3$ 至 $2m/s^3$ ；轿厢匀速度，范围从 2.5m/s 至 5.5m/s。

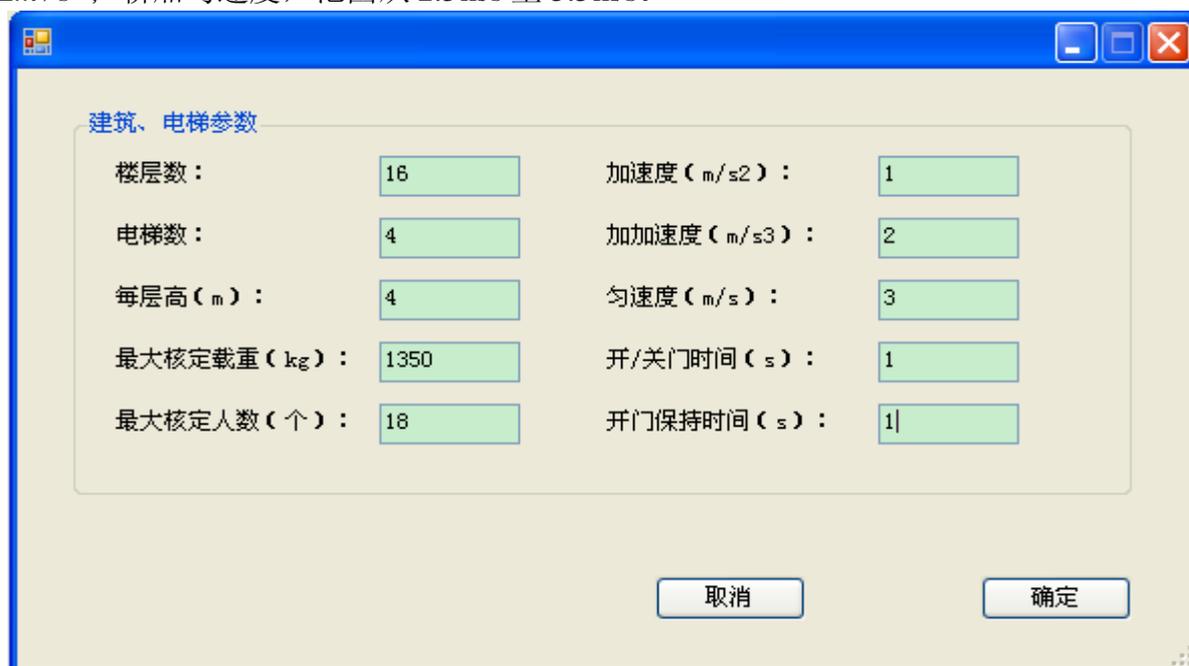


图5.24 参数设置

5.7 调度算法的仿真比较

本文所设计的仿真平台，不仅可以对实用型群控调度算法进行评测，而且还可以对人工鱼群算法、遗传算法等智能算法进行验证。人工鱼群算法与遗传算法都是仿生型的智能算法，虽然两者均能够求出全局最优解，然而在实际电梯的嵌入式环境下，由于CPU运算能力有限，且实时性要求极高，算法本身的特性会导致其搜索效率低下，本文所探讨的人工鱼群算法仅在理论研究上有意义，在实际运用中还存在一定差距。

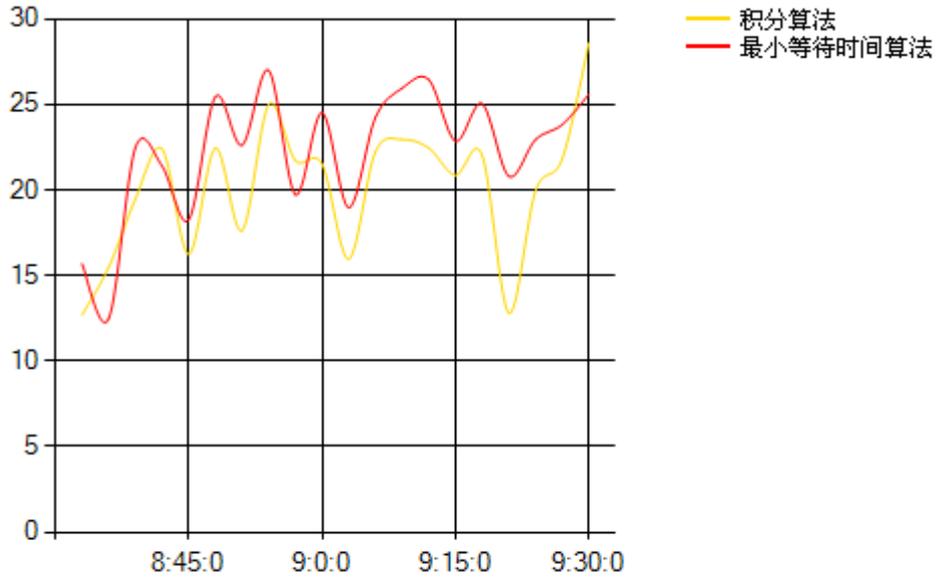
本文借助于Visual C++强大的运算处理能力进行最优解的搜索，在电梯群控虚拟仿真环境下，按如下参数设计进行对几种群控调度算法进行仿真比较：

建筑物层数为16/层，楼层高度为4 m，电梯数为4台，电梯额定载客量为18人。额定速度为 $3m/s$ ，加速度为 $1m/s^2$ ，加加速度率为 $2m/s^3$ ，开门时间为 $2s$ ，关门时间为 $1s$ ，保持时间 $2s$ 。

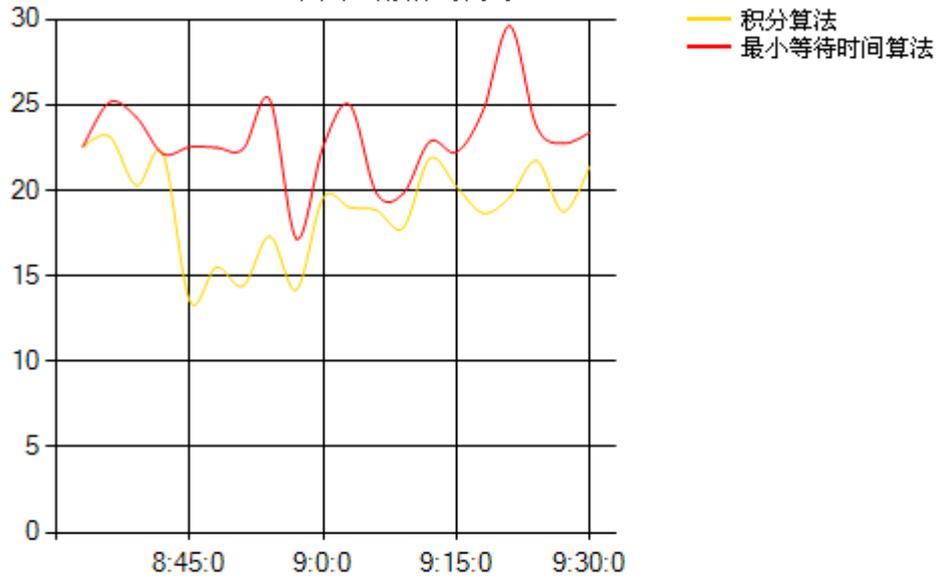
交通流数据：上行高峰（8:30~9:30，总1000人/次）。

(1) 实用型算法对比：

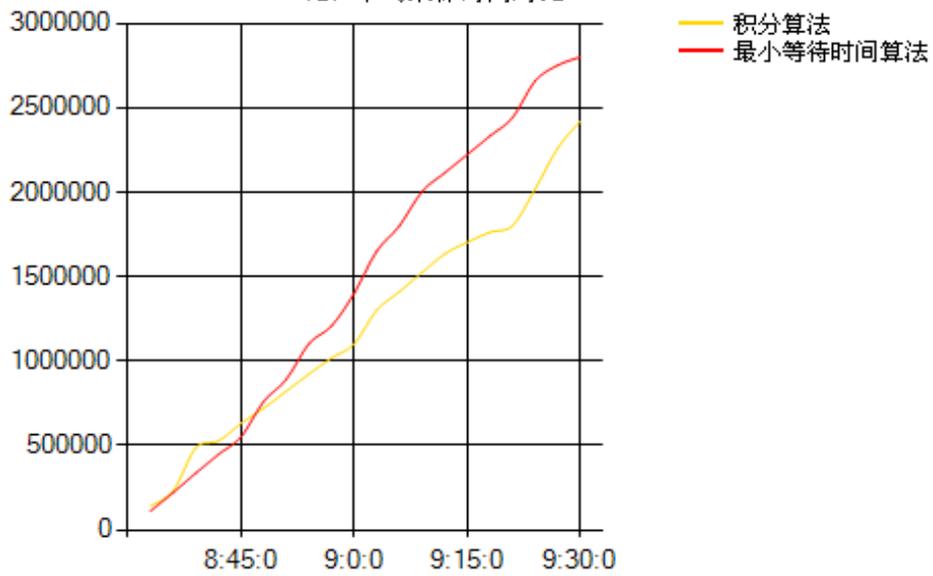
- 1) 最小等待时间算法；
- 2) 积分算法。



(1) 平均候梯时间对比



(2) 平均乘梯时间对比



(3) 能耗对比

图5.25 实用型算法仿真结果对比

表5.3 实用型算法性能指标比较

群控算法	积分算法	最小等待时间算法
AWT (s)	20.51	24.28
AJT (s)	18.82	23.35
MaxWT(s)	132	144
<20s 等待时间人数比例 (%)	65	48
20~40s 等待人数比例 (%)	24	35
40~60s 等待人数比例 (%)	8	10
60~90s 等待人数比例 (%)	2	4
>90s 等待人数比例 (%)	1	3

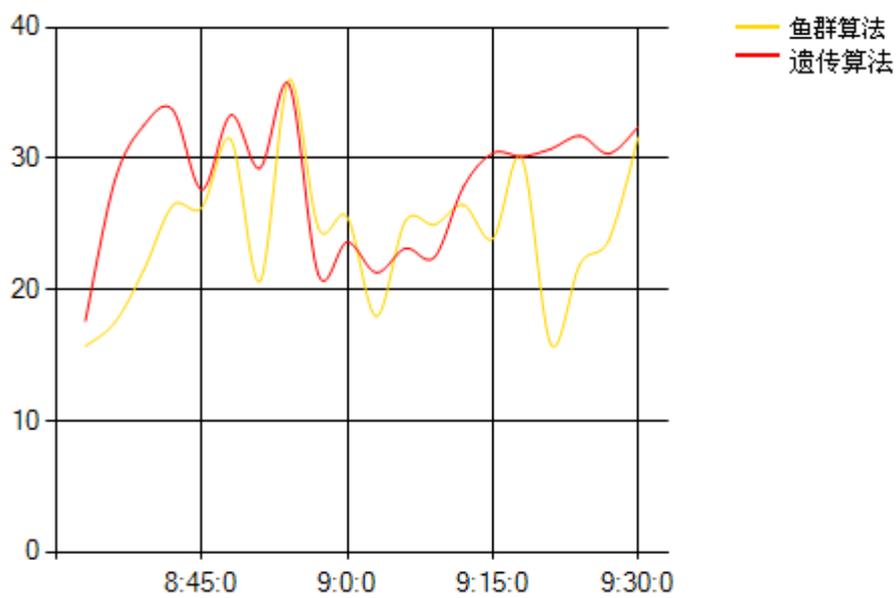
仿真结果如图5.25，2种实用型算法性能对比结果见表5.3。分析得，积分算法相比最小等待时间的电梯群控调度算法的平均候梯、平均乘梯时间、长候梯率分别减少15.52%、19.4%、57.14%，电梯能耗减少11.78%。可见相比最小等待时间算法，积分算法具有明显的优化性。

(2) 智能算法对比：

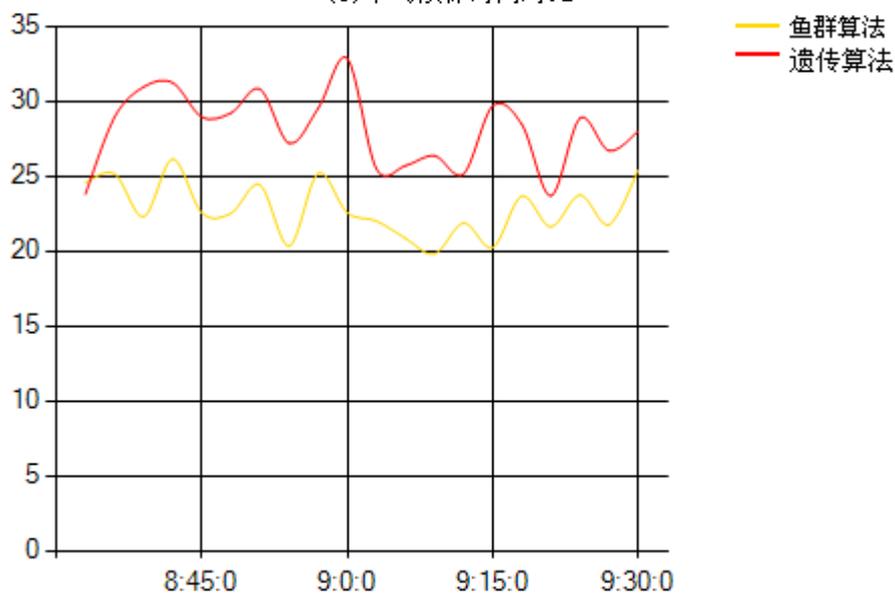
- 1) 引入轿箱内人数因素的鱼群算法；
- 2) 基于遗传算法的电梯群控调度算法基本参数设置：
 - ①种群大小：30；
 - ②代数：50；
 - ③交叉概率：0.3；
 - ④变异概率：0.01。

表5.4 智能算法性能指标比较

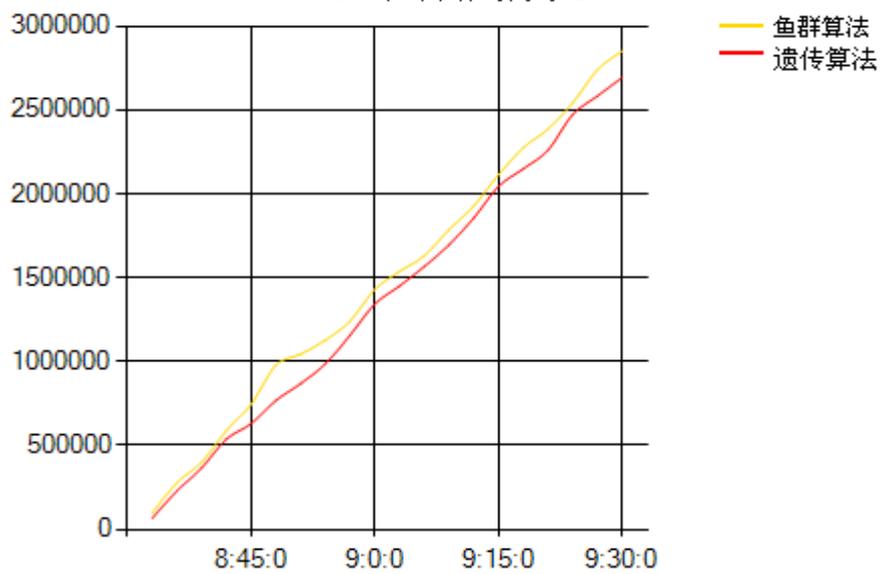
群控算法	引入轿箱内人数因素的鱼群算法	基于遗传算法的电梯群控调度算法
AWT (s)	22.81	26.78
AJT (s)	21.38	28.07
MaxWT(s)	137	141
<20s 等待时间人数比例 (%)	51	49
20~40s 等待人数比例 (%)	30	28
40~60s 等待人数比例 (%)	12	13
60~90s 等待人数比例 (%)	5	7
>90s 等待人数比例 (%)	2	3



(1) 平均候梯时间对比



(2) 平均乘梯时间对比



(3) 能耗对比

图5.26 智能算法仿真结果对比

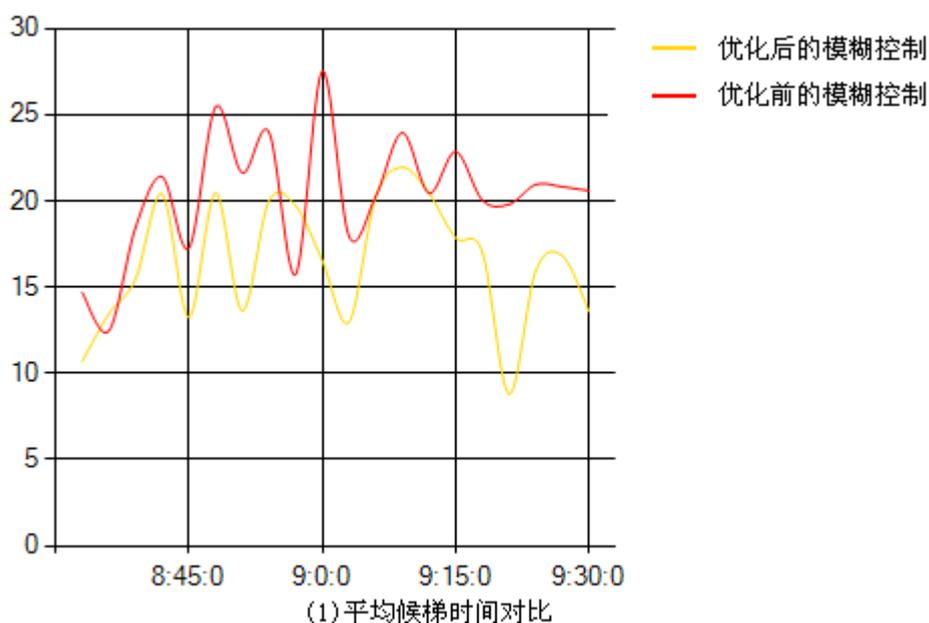
仿真结果如图5.26, 2种智能算法性能对比结果见表5.4。分析得, 引入轿箱内人数因素的鱼群算法相比基于遗传算法的电梯群控调度算法的平均候梯、平均乘梯时间、长候梯率分别减少14.82%、23.83%、30%, 但电梯能耗增加4.7%。可见相比遗传算法, 鱼群算法存在一定优越性。

(3) 优化前后的模糊控制算法对比:

- 1) 优化后的模糊控制;
- 2) 优化前的模糊控制。

表5.5 优化前后的模糊控制算法性能指标比较

群控算法	优化后的模糊控制	优化前的模糊控制
AWT (s)	16.81	19.21
AJT (s)	17.32	22.21
MaxWT (s)	117	121
<20s 等待时间人数比例 (%)	65	55
20~40s 等待人数比例 (%)	22	26
40~60s 等待人数比例 (%)	8	10
60~90s 等待人数比例 (%)	4	6
>90s 等待人数比例 (%)	1	3



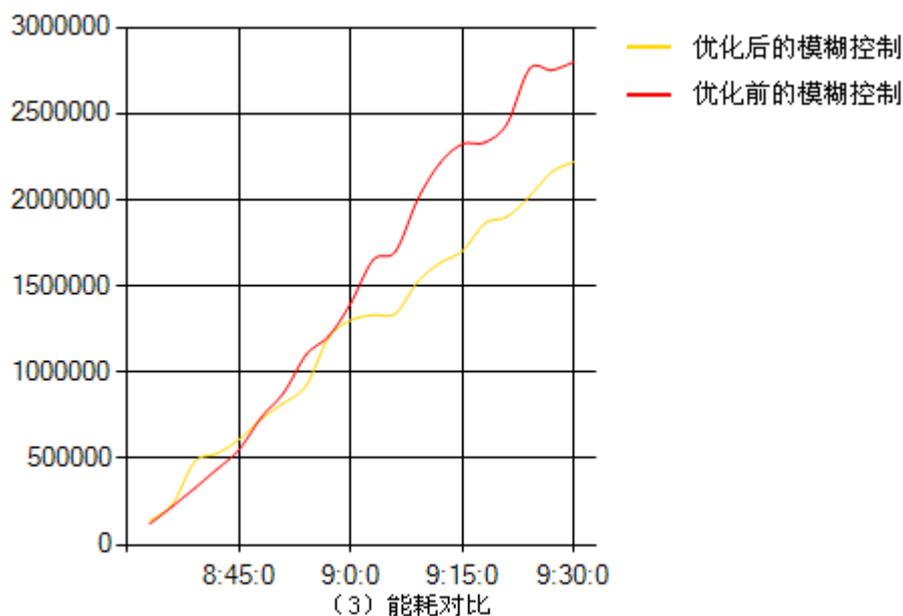
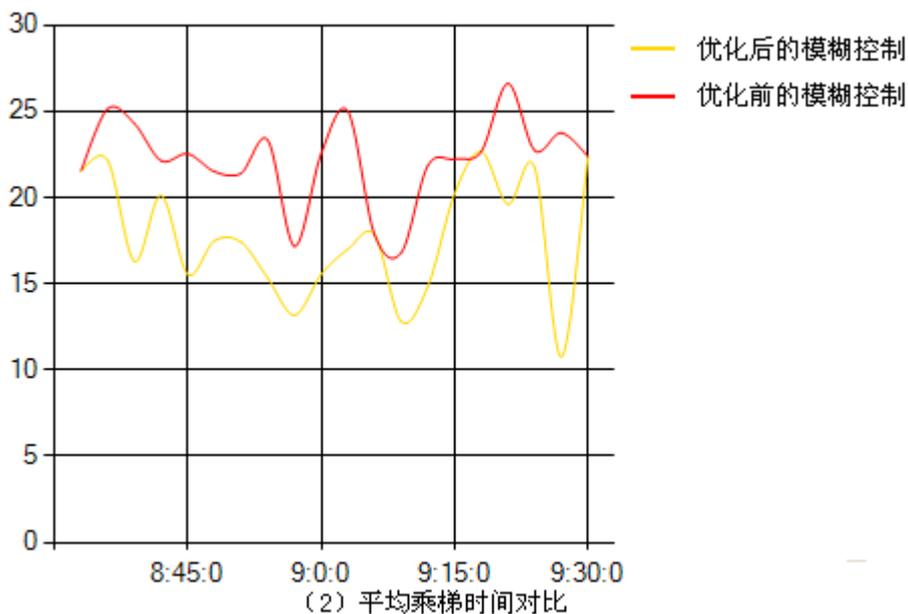


图5.27 仿优化前后的模糊控制算法仿真结果对比

仿真结果如图5.27，2种优化前后的模糊控制算法性能对比结果见表5.5。分析得，优化后的模糊控制算法相比优化前的模糊控制算法的平均候梯、平均乘梯时间、长候梯率分别减少12.49%、22.11%、44.44%，且电梯能耗减少18.51%。可见相比优化前的模糊控制算法，优化后的模糊控制算法存在一定优越性。

5.8 小结

本章详细介绍了电梯群控虚拟仿真环境的模块划分、实现原理、软件功能等。首先介绍了仿真软件的框架，然后详细阐述了关于电梯运行控制、数据通信模块、指标评测模块、群控调度算法模块等模块的设计，接着描述了相关控制目标的计算方法，并建立了典型运行阶段的能耗计算方法，其次采用图文结合的方法详细介绍了软件系统的各个操作界面及其实现原理等。最后在此环境下对几种不同的群控调度算法进行了仿真测试。

第6章 总结与展望

6.1 全文总结

电梯群控调度算法是群控系统的重要组成部分，群控评测系统是验证群控调度算法有效性的重要工具。本文首先通过分析电梯群控系统的结构及其特征，结合人工鱼群算法的特性，对人工鱼的步长进行改进，将人工鱼群算法引入到电梯群控系统中。并根据电梯运行的实际情况，把轿厢内人数作为调度因素，引入到基于人工鱼群的群控调度算法中。针对模糊推理缺乏学习性的特点，引入了人工鱼群算法多元线性回归。通过训练人工鱼群算法得到计算电梯可信度的决策函数，并将之利用到群控决策中。其次基于电梯实际运行逻辑，对群控系统的电梯运行控制、数据通信、指标评测、群控调度算法等模块进行建模，同时对相关控制目标的计算方法，如平均等待时间、平均候梯时间、典型运行阶段的能耗计算进行数学建模。最后采用C#与C++共同编程的方式将仿真评测系统编码实现。最后通过该仿真评测系统，对比三种不同类型的调度算法。其中实用型调度算法中的积分算法在平均等待时间、平均乘梯时间、能耗上明显优于最小等待时间算法。智能算法中的基于改进型鱼群算法的电梯群控调度算法与基于遗传算法的电梯群控调度算法相比，前者的乘客平均候梯时间、平均乘梯时间、长候梯率较后者有明显降低。相比优化前的模糊控制算法，优化后的模糊控制算法存在一定优越性。

本文主要的工作总结如下：

(1) 在阅读大量国内外相关文献的基础上，先对电梯群控的研究现状与发展趋势进行归纳总结，介绍了人工智能算法在电梯群控中的应用情况，并分析了不同算法的优缺点。其次归纳总结了现有仿真测试平台的研究现状及意义，并分析了不同平台的优缺点。

(2) 深入研究电梯群控系统的结构及其特征，并对电梯群控系统的模糊性、非线性、扰动性、不完备性和多目标性等动态特性进行重点分析，在此基础上介绍了电梯群控的数学模型及电梯各项性能指标，为建立仿真环境及衡量电梯群控效率的方法提供了良好的理论基础。

(3) 针对电梯群控调度问题，建立相应的人工鱼个体，分析电梯群控系统易变因素和电梯群控的多目标，提出基于线性加权平均的食物浓度评价函数。并简化鱼群算法的步长，把轿厢内人数作为调度因素，使鱼群算法更加适用于电梯调度方案的求解。再分析电梯群控系统易变因素和电梯群控的多目标，提出基于线性加权平均的电梯群控目标评价函数。在评价函数的计算方式中针对模糊推理缺乏学习性的特点，引入了人工鱼群算法多元线性回归。通过训练人工鱼群算法得到计算电梯可信度的决策函数，

并将之利用到群控决策中。对比实验表明，基于改进型鱼群算法的电梯群控调度算法在乘客平均候梯时间、平均乘梯时间、长候梯率等指标上比基于遗传算法的电梯群控调度算法有明显优势。相比优化前的模糊控制算法，优化后的模糊控制算法存在一定优越性。

(4) 建立模拟实际电梯系统的电梯群控虚拟仿真平台。根据电梯的运行过程，对群控系统的电梯运行控制、数据通信、指标评测、群控调度算法等模块进行建模。分析电梯运行阶段曳引系统的功能关系，提出了一种基于电梯典型运行阶段的能耗计算方法，与平均等待时间、平均乘梯时间等指标一起建立群控调度评价体系，为降低群控算法的评测成本提供了一套切实可行的评测平台。

6.2 进一步的工作

随着科学技术的进步，电梯群控系统作为一个非线性的复杂系统，其交叉融合的跨科学研究特征越加明显，我国对电梯群控算法的研究起步较晚，但受到了越来越多的关注。在解决模糊性、非线性、复杂性问题时，人工智能技术较传统的最小等待时间调度算法等具有明显的优势，因此如何将人工智能算法应用到电梯群控系统中，到达减少电梯群的平均等待时间、平均乘梯时间、长时间候梯率的目的，已成为当前一个研究趋势。本文所做的工作是以通用、模块化的思想编写仿真测试平台，建立单部电梯运行模型、客流生成器、评价体系、群控调度算法接口，模拟真实电梯群控的运行，从而为进一步分析不同类型的群控调度算法创建优良的实验环境，但仍需要有进一步完善的地方，概括起来为：

(1) 提高群控算法的可扩展性和易用性。考虑到电梯群控系统的复杂性，对于不同用途的大楼，大楼层高不同，所需控制的电梯数目不同，面对的客流也不同，所以控制算法的设计结构是个难点，需要进一步完善其扩展性和易用性。

(2) 电梯群控虚拟仿真平台是验证群控调度算法有效性的重要工具，可以大大降低企业的运行成本，所以对该仿真平台的研究必须进一步强化。

(3) 高层建筑电梯群控应急疏散配置。随着高层建筑的普及，人们对防灾、减灾意识的提高，电梯的应急疏散作用也必将纳入电梯群控系统之中，因此对高层建筑应急情况下的调度策略的进一步研究工作也是必须要做的。

(4) 应对人工鱼群算法用于实用型调度算法如积分算法、专家系统的参数优化进行研究。

致 谢

本文是对我攻读硕士学位期间工作的一个总结。首先，衷心感谢我的导师李强副教授，本论文的大部分研究工作都是在他的直接指导下完成的。在攻读硕士学位期间，李老师无论在学业上还是在生活上都给予了我很大的关心和帮助。导师渊博的知识、敏捷的思维、勤勉的作风以及宽厚正直的为人永远值得我敬仰和学习！

在论文进行过程中，感谢严义教授对论文提出了许多建设性的意见和建议，严老师严谨的治学态度使我受益匪浅；感谢包健教授在开题、预答辩时提出的宝贵意见；感谢邬惠锋、赵建勇老师对论文的审核和论证。

感谢本实验室的各位师兄、师弟和师妹，他们是卢金鑫、孙大群、吴盼盼、尤仙、金丽娟、马超、万真龙、沈凌飞、应俊、张跃喜。一直以来和他们融洽的生活，共同营造了一个良好的学习、研究环境，并且同他们的讨论中受益匪浅。感谢潘必平、虞彦、裴晨昕等朋友的支持和鼓励。感谢好友赖海明的帮助以及在论文写作中给出很好的建议。

特别感谢千里之外的父母家人在我漫长的求学生涯中对我长期的鼓励和支持。

参 考 文 献

- [1] 刘战国. 智能控制在建筑空调控制系统及电梯群控系统中的应用研究[D]. 重庆大学, 2008.
- [2] Barney G. C. and Dos Santos S. M. Elevator Traffic Analysis Design and Control[M]. Peter Peregrinus. 1985.
- [3] 马福军. 电梯群控技术的研究[D]. 浙江工业大学, 2003.
- [4] 张绘敏. 基于模糊控制的电梯群控算法研究[D]. 兰州交通大学, 2011.
- [5] 杨祯山, 邵诚. 电梯群控技术的现状与发展方向[J]. 控制与决策. 2005, 20(12): 1321-1331.
- [6] 李东, 王伟, 邵诚. 电梯群控智能系统与智能控制技术[J]. 控制与决策, 2001, 05(05): 513-517.
- [7] Albert T. P. So, Janson K. L. Yu and W. L. Chan. Dynamic Zoning Based Supervisory Control for Elevators. International Conference on Control Applications. 1999: 1591-1596.
- [8] Albert T. P. So and W. L. Chan. Comprehensive Dynamic Zoning Algorithms[J]. Elevator World. 1997, 45(9): 99-103.
- [9] Beielstein T, Markon S and Preuss M. A Parallel Approach to Elevator Optimization Based on Soft Computing. Proc of the 5th Metaheuristics Int Conf. 2003: 25-28.
- [10] 宗群, 王振世, 尚晓光. 电梯上高峰模式下动态分区算法的研究[J]. 系统工程与电子技术. 2001(12): 97-100.
- [11] 刘登科. 电梯技术最新发展趋势[J]. 科技信息, 2011, 08(04): 810-811.
- [12] 李彦华. 智能电梯群控系统的研究与设计[D]. 厦门大学, 2008.
- [13] 朱晓红. 智能建筑设备节能优化运行控制技术研究[D]. 重庆大学硕士学位论文. 2005.
- [14] Z Li, Y Ruan. A Novel Energy Saving Control System for Elevator Based on Supercapacitor Bank Using Fuzzy Logic. Proceedings of 11th International Conference on Electrical Machines and System. 2008, 22(5): 2717-2722.
- [15] Sakai Y and Kurosawa K. Development of Elevator Supervisory Group Control System with Artificial Intelligence. Hitachi Review. 1984, 33(1): 25-30.
- [16] 李伙友. 智能电梯群控系统的研究[D]. 厦门大学, 2007.
- [17] K Igarashi, S Take and T Ishikawa. Supervisory Control for Elevator Group with Fuzzy Expert System. Proceedings of the IEEE International Conference on Industrial Technology. 1994: 133-137.
- [18] 赵顺, 顾战松. 基于专家系统的电梯群控的实现[J]. 电气自动化. 2003, 25(1): 28-29.
- [19] Ho M and Robertson B. Elevator Group Supervisory Control Using Fuzzy Logic[J]. Proc of Canadian Conf on Electrical and Computer Engineering. 1994: 825-828.
- [20] 张昆, 段其昌, 张从力. 基于模糊控制的多目标电梯群控技术[J]. 仪器仪表学报. 2004, 25(4): 248-251.
- [21] 曾国伟, 赵国军, 邢海潇, 陆鑫森. 基于模糊控制技术的电梯群控系统的设计[J]. 浙江工业大学学

报.2009,37(1):115-118.

[22] J C Liu and Y Y Liu. Ant Colony Algorithm and Fuzzy Neural Network-based Intelligent Dispatching Algorithm of An Elevator Group Control System. 2007 IEEE International Conference on Control and Automation. 2007:2306-2310.

[23] 许玉格,罗飞,曹建忠. 目的层预约的模糊神经网络电梯群控策略[J]. 华南理工大学学报(自然科学版),2007,01(12):13-18.

[24] W L Chan, Albert T P So and K C Lam. Dynamic zoning for intelligent supervisory control Int J of Elevator Engineering. 1995,4(2):47-59.

[25] Toshimitsu Tobita, Atsuya Fujino, Kazuhiro Segawa et al. A parameter tuning method using genetic algorithms for an elevator group control system. Proceedings of the 1996 IEEE IECON 22nd International Conference.1996 (2):823-828.

[26] 何万里,李桂芝,钱伟懿.改进的遗传算法在电梯群控中应用[J].渤海大学学报.2007,28(1):46-50.

[27] 唐桂忠.基于遗传算法的现代群控电梯交通客流控制[D].南京工业大学硕士学位论文.2005.

[28] Siikonen ML. Elevator traffic simulation[J]. Simulation. 1993, 61(4): 257-267.

[29] Siikonen ML, Susi T'Hakonen H. Passenger traffic flow simulation in tall buildings[J]. Elevator World.2001, 49(8): 117-123.

[30] ZongQ, XueL, Wang Z. Virtual simulation environments of elevator group control systems[J]. Elevator World.2001,49(12): 90-93.

[31] DieRerich T G. Hierarchical reinforcement learning with the MaXQ value function decomposition[J].Journal of Artificial Intelligence Research. 2000,13:227-303.

[32] Bao G Cassandras C G,et al. Elevator dispatchers for down peak traffic[R]. Boston: ECE Department Technical Report, University of Massachusetts, 1994.

[33] Lustig A. Lift simulation program[J]. Elevator World. 1986, 34(3): 74-79.

[34] So AT P, Chan W L. Comprehensive dynamic zoning algorithms[J]. Elevator World. 1997,45(9): 99-103.

[35] So AT P, Chan W L. Intelligent supervisory control for lifts: dynamic zoning[J]. Building Serv Eng Res Technol. 2001,22(1): 14-33.

[36] 宗群, 尚晓光, 岳有军. 电梯群控系统虚拟仿真环境设计[J].制造业自动化.1999,21(5): 24-25,31.

[37] 宗群,王维佳,刘文静. 电梯群控与监视系统实验平台的研究. 实验室研究与探索.2005, 24(增刊): 291-294.

[38] 李中华,朱燕飞,李春华.面向对象的电梯群控系统仿真平台开发与实现[J]. 计算机仿真. 2005,22(2):156-159,170.

[39] 余兴智.电梯群控系统的试验平台设计与研究[D].上海交通大学硕士论文.2003.

[40] Donald Cooper. Energy Consumption of Various Elevator Drives[J]. Elevator World. 1987, 35(3):20-23.

[41] Ashok B Kulkarni. Energy Consumption Analysis for Geared Elevator Modernization: Upgrade from

- DC Ward Leonard System to AC Vector Controlled Drive.IEEE Trans. Ind.Applat. 2000,4(8-12):2066-2070.
- [42] White LE. Energy Consumption : Hydraulic Elevators and Traction Elevators[J]. Elevator World.1984,32(4):76-77,79.
- [43] Y. Lu,S. Mabu,T. T. Zhang, et al.Multi-car elevator group supervisory control system using Genetic Network Programming. IEEE Congress on Evolutionary Computation . 2009. 12(5):242-255.
- [44] Hirasawa K,Eguchi T,Zhou J,Yu L,Hu JL,Markon S.Adouble-deck elevator group supervisory control system usinggenetic network programming.[J] Applica-tions and Reviews . 2008,28(04):535-549.
- [45] 孙立新. 关于电梯能效评价的探讨. 中国电梯, 2008,19(4):43-45.
- [46] 陈志溪,余志林,叶剑刚.电梯工作特性与能耗形式分析.中国电梯.2008,19(16):45-49.
- [47] 秦臻.基于 SVM 的电梯群控系统 (EGCS) 算法的研究[D].杭州电子科技大学硕士学位论文,2011.
- [48] 孙增圻等.智能控制理论与技术[M].清华大学出版社,广西科学技术出版社. 2008.
- [49] 赵小翠.电梯群控系统的多目标智能优化调度研究[D].华南理工大学硕士学位论文,2011.
- [50] 林琳.群控电梯多目标智能最优调度算法的研究[D].东北大学硕士学位论文. 2006.
- [51] 王芳,宗群,张景龙,李俊芳. 电梯群的可调整鲁棒优化调度[J]. 控制理论与应用,2012,01(2):27-33..
- [52] TuXiaoyuan. Artificial Animals for Computer Animation: Biomechanics, Locomotion, Perception, and Behavior[D], Dissertation for doctor degree. 1999.
- [53] 王闯.人工鱼群算法的分析及改进[D].上海海事大学硕士学位论文. 2008.
- [54] 李晓磊, 邵之江, 钱积新. 一种基于动物自治体的寻优模式: 鱼群算法[J]. 系统工程理论与实践. 2002,22(11):32-38.
- [55] 俞洋,孔洁,田亚菲. 基于人工鱼群算法的 IIR 数字滤波器设计[J].通信理论与技术新进展——第十届全国青年通信学术会议论文集. 2006(08):301-305.
- [56] 高德芳,赵勇,郭杨,赵海涛.基于混合鱼群-蚁群算法的模块化产品配置设计[J].机械.2007,22(01):32-35.
- [57] 袁远.基于改进人工鱼群算法的配电网无功优化[D]. 南京理工大学硕士学位论文.2008.
- [58] 杨丽.基于人工鱼群算法的图像配准的研究与实现[D]. 西安电子科技大学硕士学位论文.2009.
- [59] 刘李钊. 基于鱼群克隆遗传算法的配电网重构研究[D]. 中南大学硕士学位论文.2011.
- [60] 王芳,万磊,徐玉如,张玉奎. 基于改进人工势场的水下机器人路径规划[J].华中科技大学学报(自然科学版).2011,39(2):184-187.
- [61] 杨淑露,韩奇,徐琳茜.鱼群算法与神经网络结合的节能减排效果评价[J].中南大学学报(自然科学版).2012,43(4):1538-1544.
- [62] 丁生荣,马苗,郭敏.人工鱼群算法在自适应图像增强中的应用[J].计算机工程与应用.2012,48(2):185-187.

附录:

作者在读期间发表的学术论文及参加的科研项目

- [1] 王瀚韬, 李强. 基于鱼群算法的电梯群控调度算法[J]. 机电工程, 2013.