Sri Lanka Association for Artificial Intelligence (SLAAI)
Proceedings of the fourth Annual Sessions
31st October 2007 - Colombo

Intelligent Elevator Group Control System

T. K. W. Senevirathna, H. Ekanayake and T. N. K. De Zoysa

University of Colombo School of Computing
Colombo, Sri Lanka
kanchana.senevirathna@gmail.com, {hbe,tnk}@ucsc.cmb.ac.lk

Abstract

High rising building is a common sight in most of the cities today. Fast and efficient elevator transportation is a key feature when creating these kinds of buildings. Lot of research has been carried out to build an intelligent system that satisfies the need of elevator control as it is a must to have intelligent elevators in the future. This paper proposes a destination control system for elevator group controlling which fully utilizes destination information. Fuzzy logic concepts are used to enable the elevator control system to make decisions. The design criteria include of optimizing movement of elevators with regard to several factors such as waiting time, riding time, energy, load, etc. Software simulation is done in order to capture the performance of the proposed system with compared to conventional approaches.

1. Introduction

Since the start of elevator systems, each and every one carried out the same operations just by using up and down hall call buttons and giving the destination inside a car, not allowing that much of processing of data to be done. However, this type of system provides some disadvantages such that the supervisory system of the elevators does not receive information on the destination of the passengers before they board the car. Consequently, car assignments are based on far less than 50% of the traffic information that passenger could supply earlier to the system and therefore assignments are obviously poor in quality. Such inherent disadvantage present in conventional system can be improved by using a destination oriented system.

The choice of optimization target in the cost function is important when considering the overall elevator performance and the service level. The most general optimization target in group controls has been the minimization of the average and maximum hall call times. Recently the cost functions have become more comprehensive. Instead of one target, multiple targets are optimized. A number of costs, such as call time, passenger waiting and journey times, car load factor, energy consumption, transportation capacity and number of starts, can be considered during the call allocation. When optimizing one target, the other features may suffer. For instance, when optimizing the energy consumption, the passenger waiting times may increase. Several optimization targets can be optimized within one control if the most suitable target is switched or prioritized according to the prevailing traffic pattern or requests.

Fuzzy logic [3][4] is used as an intelligent approach to optimize multiple targets such as waiting time, riding time, load and distance. This system determines the optimum car for a particular request. System requires the passengers to enter their destination before entering to the car, using a numeric keypad which is located in the requesting floor. The system then assigns the passenger to a car and displays the assigned car back to the user. The system is tested in simulating the traffic conditions of 3 cars in a 15 floor building.

2. Conventional Elevator Group Controller Architecture

A typical architecture of an elevator control system [1] is shown in figure 1. The control boards of an elevator group are usually situated in a machine room. The group control is the "brain" of the elevator system, deciding where the cars should go and stop. There can be one or several group controls in an elevator group. One of the group controls is the master that delivers the hall calls to the elevators, and the other group controls are backups. Other functions inside the car, e.g. registering and canceling of car calls, door control, and measurement of the car load, are handled by the elevator control. Using the latest microprocessor technology, part of the elevator control has been distributed among the elevator components. "Intelligent" elevator components communicate with each other through serial transmission using Control Area Network (CAN)[5]. Modern elevator controls provide built-in elevator monitoring devices or remote building monitoring systems to follow the elevator traffic. Typical control software for an elevator component includes an operating system, task-scheduling programs, input, output and communication programs, and programs for controlling and optimizing the function of the component.

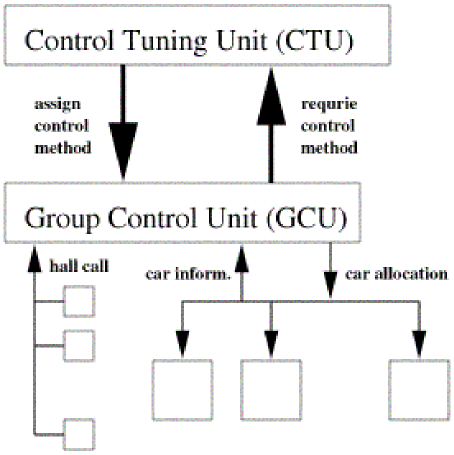


Figure 1: Architecture of conventional Elevator Group Control System

2. Design Criteria

The elevator configuration is chosen so that the generality of the problem is preserved and the simulation can be visualized which means that the building should not have too many or too less, number of floors and the number of elevators should be such that they operate smoothly under varied traffic intensity. Here, the number of elevators was fixed to stress more on the group control of elevators rather than the selection of optimum number of elevators. Thus 15 floor building with 3 cars is selected for testing and simulation. Since the system needed destination information of a passenger before entering the car, that information is fed to the system with a numeric keypad located in every requesting floors.

The objective is to design a dispatching logic that achieves desirable results with respect to

- Minimize the waiting time of a passenger.
- Minimize the riding time of a passenger.
- Minimize the crowding of an elevator.
- Minimize the traveling distance of each elevator.

Also it assumes that a stop of an elevator will take 3 times the time taken to travel one floor. Maximum load for each car is assumed to be 20 persons.

3. Related Work

Although it is hard to implement and compare with most of other new approaches, following points gives a comparison with adaptation of the concepts by the proposed algorithm regarding some specific features.

3.1 Zoning

Zoning [5] has been a planning strategy from the start of high rising building elevator planning. Each car is assigned a zone of the building. It answers hall calls within its zone. The goal of the zoning approach is to keep the cars reasonably well separated and thus keep the interval down.

In fuzzy logic approach calculation of hall call area and destination call area helps to achieve dynamic zoning. If the call area is near to 0, algorithm will give a high priority to the lift when selecting the lift. This concept is further developed with a destination controlling system since it has the destination information before assigning a lift to a request.

3.2 Search based approaches

Another control strategy is to search through the space of possible car assignments, selecting the one that optimizes some criterion such as the average waiting time. Greedy search strategies [5] perform immediate call assignment, that is, they assign hall calls to cars when they are first registered, and never reconsider those assignments. Non-greedy algorithms postpone their assignments or reconsider them in light of updated information they may receive about additional hall calls or passenger destinations. Greedy algorithms give up some measure of performance due to their lack of flexibility, but also require less computation time.

Approach used in this project can be considered as a greedy approach. It assigns a car immediately when a request is made. Although this has the disadvantage of greedy algorithms, i.e. lack of flexibility, this has been overcome by various factors. Some of the factors are getting destination information before getting into a lift contrastingly to conventional systems and reducing psychological waiting time by displaying assigned lift earlier. (This is simulated in the simulation software) Also reassignment of lifts will result badly when it comes to dynamic zoning. Also avoiding non greedy approaches may result in less computation thus will be a less complex algorithm which is easy to understand and further develop.

3.3 Adaptive and learning approaches

Some of the optimization approaches use neural networks for learning traffic pattern to select the best rule set for fuzzy logic, to optimize parking [7], etc. However there are many situations in which training examples are costly or even impossible to obtain. Reinforcement learning [8] is more applicable in these more difficult situations where the only help available is a 'critic' that provide a scalar evaluation of the output that was selected, rather than specifying the best output or a direction of how to change the output. But in the applications of reinforcement learning or any other learning approaches it needs a large set of data to be trained on. This should be concerned regarding the application of real elevator system, since one would want to perform the initial training in simulation in any case, not only because of the large amount of experience needed, but also because performance would be poor during the early stage of training. Even though training will be done in a simulation, traffic generation of a simulator may not be accurate. Also elevator traffic depends so much on the type of the building [2]. Thus learning approaches are too much effort when building an algorithm for elevator group controlling. But this approach can be used for forecasting traffic and parking of elevators which will be a future addition to proposed algorithm in this project, which describes only immediate call allocation mechanism

4. Fuzzy controllers: An overview

Fuzzy controllers, [3] [4] contrary to classical controllers, are capable of utilizing knowledge elicited from human operators. This is crucial in control problems for which it is difficult or even impossible to construct precise mathematical models, or for which the acquired models are difficult or expensive to use. These difficulties may result from inherent nonlinearities, the time varying nature of the processed to be controlled, large unpredictable environmental disturbances, degrading sensors or other difficulties in obtaining precise and reliable measurements and a host of other factors. It has been observed that experience human operators are generally able to perform well under these circumstances.

The knowledge of an experienced human operator may be used as an alternative to a precise model of the controlled process. While this knowledge is also difficult to express in precise terms, an imprecise linguistic description of the manner of the controller can usually be articulated by the operator with relative ease. This linguistic description consists of a set of control rules that makes the use of fuzzy propositions. A typical form of these rules is exemplified by the rule.

If the *temperature* is very high
AND the *pressure* is slightly low
THEN the *heat change*
should be slightly negative,

Where temperature and pressure are the observed state variables of the process, and heat change is the action to be taken by the controller. The vague terms very high, slightly low and slightly negative can be conveniently represented by fuzzy sets defined on the universe of discourse of temperature values, pressure values and heat changes values, respectively. This type of linguistic rule has formed the basis for the design of a great variety of fuzzy controllers described in the literature.

A general fuzzy controller consists of four modules: a fuzzy rule base, a fuzzy inference engine and fuzzification/ defuzzification modules.

5. Applying fuzzy logic to Elevator control system

In this system destination details are obtained before a passenger load into the car through a numeric keypad located in the floor we can use these additional knowledge when applying fuzzy logic. For fuzzification 6 base variables has been identified: 5 input variables and 1 output variable. Following quantities are given as input to the Fuzzy interpreter:

- **Waiting time** - Total time an elevator needs to travel from its current position to the new hall call.
- **Riding time** - Total time a passenger spent in the elevator until he/she reached at his/her destination.
- **Loading** - Number of passengers in an elevator.
- **Hall call area** - The area weight of the elevator which goes to the floor where a new hall call is generated.
- **Destination call area** - The area weight of the elevator which goes to the floor where the destination of the new hall call is generated.

"Priority" is the output of the fuzzy controller and the elevator with highest priority value will be assigned for the given request.

5.1 Fuzzification and Fuzzy Reasoning

Each fuzzy system [6] is realized in the form of fuzzy rules as in the following example**:**

Rule 1: If X is A_1 and Y is B_1 then Z is C_1
Rule 2: If X is A_2 and Y is B_2 then Z is C_2

Where X and Y are variables of the condition part, and Z is the variable of the action part. A_i, B_i and C_i are fuzzy parameters characterized by membership functions.

The condition parts of control rules make use of measurements which are usually real numbers. e.g. x^0 and y^0 are matched to their corresponding fuzzy variables by determining their membership values defined as figure 2.

Suppose that $X = x^0$ and $Y = y^0$, the reasoning is derived as follows:

- Define the linguistic variables as described in fuzzification section.
- Compute firing levels by mathematical interpretation of these rules, as follows:

  $$\alpha_1 = A_1(x^0) \cap B_1(y^0)$$
  $$\alpha_2 = A_2(x^0) \cap B_2(y^0)$$

  Where x^0, y^0 are actual inputs to the system and A_1, A_2, B_1, B_2 are fuzzy sets and α_1, α_2 are firing levels. Inputs are now fuzzified and embedded in the firings of each rule.

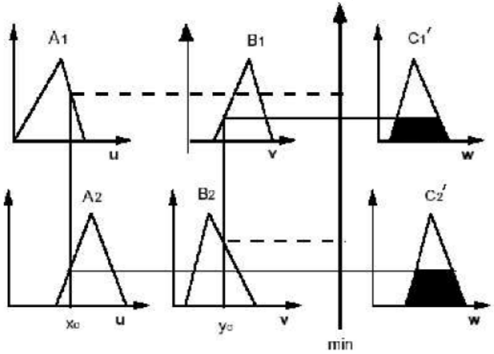


Figure 2: Mamdani Inference Mechanism

- Apply Mamdani inference mechanism [3] to get individual out fuzzy sets for each rule and a final consequence set as shown in the figure 2. Mathematically,

  $$C'_1 = \alpha_1 \cap C_1$$
  $$C'_2 = \alpha_2 \cap C_2$$

  Where C'_1, C'_2 are individual output fuzzy sets for each rule and C is the final consequence set.

5.2 Membership Functions

In order to build membership functions (MF) it is needed to identify the ranges where each base variable is spread out.

Waiting time = distance to request + no. of stops x3
Riding time=distance to destination + no. of stops x 3

Since waiting time and riding time is evaluated with above mentioned formula range of the membership function depend on the maximum number of floors in the building. As in this system there are 15 floors maximum number of distance to arrival/destination is 14 floors. Also maximum number of stops is 14. So the range for waiting time or riding time will be from 0 to 56.

Their membership functions for waiting time and riding time are shown in figure 3a and 3b.

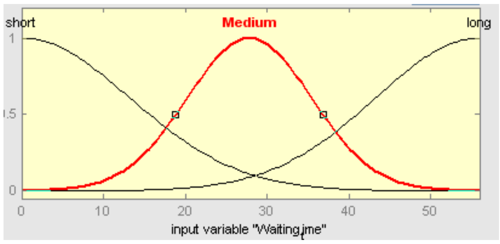


Figure 3a: Membership functions for waiting time.

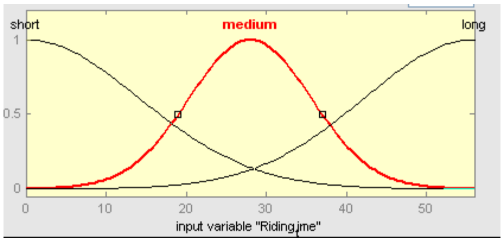


Figure 3b: Membership functions for riding time.

As maximum number of passenger for the lift is 20, the range for number of passengers is from 0 to 20. Figure 3c shows it's membership functions.

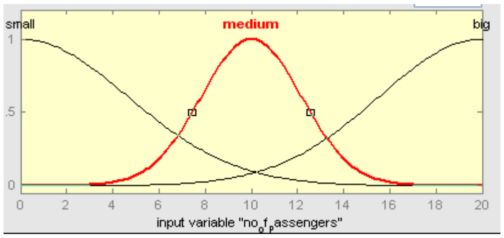


Figure 3c: Membership functions for load.

Destination call area weight and hall call area weight values represent its direction as well. So the range consists of -14 to 14. Figure 3d displays its membership functions.

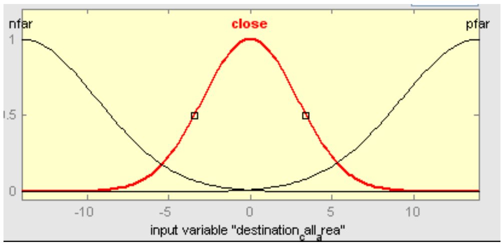


Figure 3d: Membership functions for call areas.

Membership functions small, medium and high for output 'priority' are displayed in figure 3e.

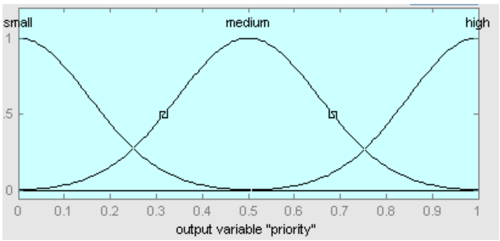


Figure 3e: Membership functions for priority.

5.3 Rule Base

Depending on the fuzzy inputs and the rule bases, the output fuzzy set, `priority' is computed using an inference scheme. Several inference schemes are available like Mamdani, Sugeno etc. For the present simulator, the Mamdani scheme has been adopted.

In this application each rule has a single input mapped to a single output to avoid complexities involved by considering all the inputs in a single rule. There are different rules to achieve different objectives as described below.

1. If waiting time short then priority is high.
2. If waiting time is medium then priority is medium.
3. If waiting time is long then priority is small.
4. If riding time short then priority is high.
5. If riding time is medium then priority is medium.
6. If riding time is long then priority is small.
7. If loading is small then priority is high.
8. If loading is medium then priority is medium.
9. If loading is high priority is small.
10. If hall call area is close priority is high.
11. If hall call area is positively far priority is medium.
12. If hall call area s negatively far priority is small.
13. If destination call area is close priority is high.
14. If destination call area is positively far priority is medium.
15. If destination call area is negatively far priority is small

Every rule has a weight (a number between 0 and 1), which is applied to the number given by the antecedent. Generally this weight is 1 and so it has no effect at all on the implication process. From time to time system may want to weight one rule relative to the others by changing its weight value to something other than 1.

5.4 Defuzzification

The input for the defuzzification process is a fuzzy set (the aggregate output fuzzy set) and the output is a single number. As much as fuzziness helps the rule evaluation during the intermediate steps, the final desired output for each variable is generally a single number. However, the aggregate of a fuzzy set encompasses a range of output values, and so must be defuzzified in order to resolve a single output value from the set. In this system centroid method is being used.

The centroid of a plane figure can be given as the integral,

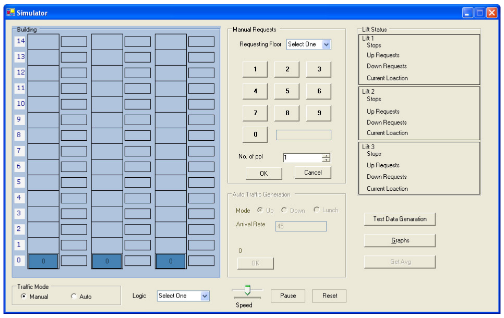
$$\frac{\int x f(x)\, dx}{\int f(x)\, dx}$$

where f(x) is the vertical extent of the object at abscissa x. This formula can be derived from the first moment about the y-axis of the area.

6. Implementation

Simulator is developed using C#.Net and MATLAB. C#.Net is used to create the graphical user interface and calculate performance of the algorithm compared to conventional algorithms. Also calculating input parameters to fuzzy logic is done in the .net application. The dynamic link library that was created by MATLAB to evaluate the fuzzy inputs and to give the output feed the output to the .net application. In the simulator following functionalities are implemented:

- Auto traffic generator
- Simulation of few different algorithms
- Graph controller for test result analysis
- Elevator movement visualization



Requesting Floor: Select One
No. of ppl: 1
Mode: ● Up ○ Down ○ Lunch
Arrival Rate: 45
Traffic Mode: ● Manual ○ Auto
Logic: Select One

Figure 4: Snapshot of the simulator GUI

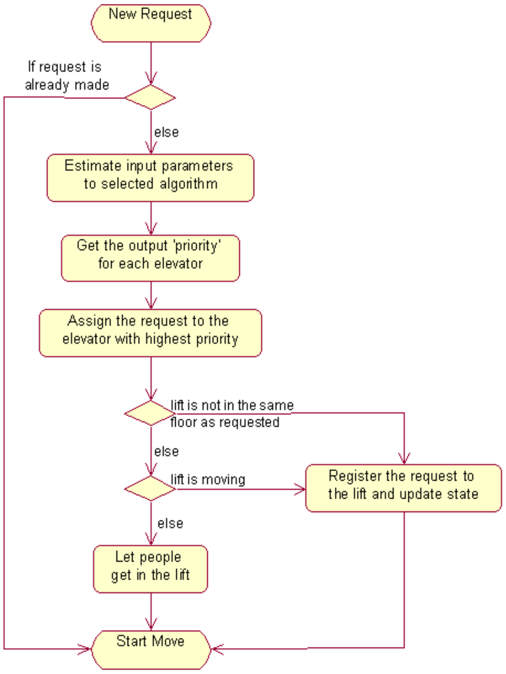


Figure 5: Request registering events flow

The above figure describes how the lift maintains its states and functions accordingly. When a new request is made system checks whether the request already exists or not. If not system estimates the input parameters to the relevant algorithms considering each lift's status. Then considering these values it estimates the priorities. If the lift with the highest priority is not in the same floor as requested or moving, it registers the request to the lift and start moving. If the lift is in the requested floor and stationary it allows people to get in and then starts moving.

7. Testing and Evaluation

In the simulation average waiting time, average riding time and average distance is mapped to graphs in order to evaluate algorithms by comparison. Figure 6a, 6b and 6c show the comparison with regard to each performance criteria. In order to have smooth line graph simulation should be done several times for each arrival rate (number of requests/ 5 minutes). In these results simulation has run 2-3 cycles for the smoothing of the results. For every graph x axis is arrival rate (number of requests per 5 minutes time period) and y axis will be the performance criteria i.e. average waiting time, average riding time, etc.

7.1 Average Waiting Time

Figure 6a: Average waiting times in up peak traffic mode

Since the minimum waiting time method optimize the average waiting times it is visible that minimum waiting time method has the best performance when considering waiting times. Since fuzzy logic uses waiting time as an input to the system, it also minimize the waiting time as possible. Since minimum distance consider only the distance not the stops between the current location of the lift and the destination, its waiting times are quite high compared to other two approaches.

7.2 Average Riding Time

Fuzzy logic has used riding time as an input to the system. Figure 6b shows that fuzzy approach is being fairly good when compared to other approaches. Results are not that convenient due to that the traffic is randomly generating and simulation ran only couple of time for each arrival rate since it is time consuming.

Figure 6b: Average riding times in lunch traffic mode

7.3 Average Distance

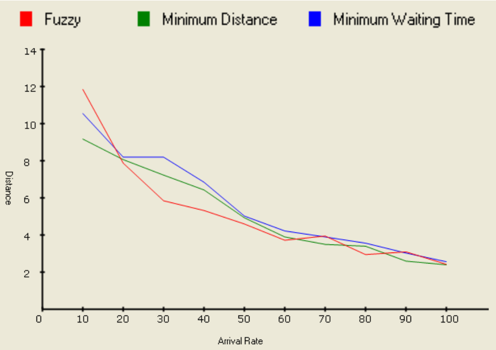


Figure 6c: Average distances in down peak traffic mode

Fuzzy logic is proven to be good in average distances as well. Around 30 – 70 arrival rates fuzzy approach is the best approach in all the traffic modes when compared to others.

When comparing all graphs fuzzy logic has optimized multiple targets very effectively unlike other approaches when one performance is optimized other performance act badly. Since in this project weight of the fuzzy rules and rules itself has been fixed, there might be some performance lost regarding some parameters. But as a future improvement if this system is enabled with dynamic rule allocation and weight changes according to the traffic mode this will optimize the needed performance at needed time.

8. Conclusion

Fuzzy logic has been proven to fit in to most of the complex situations in decision making. Elevator group controlling is one such situation where we can use fuzzy logic, i.e. rule based approaches, very efficiently. This algorithm proves to be a very good option when talking about multiple target optimizations since in new generation elevators, waiting time or riding time optimization may not be the first objective since elevator systems in high rising buildings have developed mechanically to increase elevator speeds. Power consumption may be one key thing that will look into when developing this type of system. Also fuzzy logic has shown to be quite powerful, yet computationally less intensive technique to deal with in this kind of problems.

Although destination control systems have not been installed in old buildings, it is proven to be far more effective to have destination information before entering to an elevator. With the new systems, riders can log in their destination at central lobby and the guidance programs use this information to determine which cars and how many of them will travel to specific floors. An LCD display directs riders to the proper elevator. Without traffic guidance systems, tall buildings are vulnerable to queuing, a condition that produces frustration among riders while they wait for available cars and after they jam into one, because they come into cars at random, riders end up stopping at practically every floor. And that means it's going to take a long time to come back to the lobby to pick up the next group of people which leads to more queues. But since in this system people with similar destinations ride in the same car, the number of stops the elevator has to make decreases significantly.

8.1 Future Work

This project addressed broad issues related to elevator control and was more general in nature since elevator controlling has many areas such as planning, group controlling, traffic pattern recognition, etc. Hence, there remains significant scope of improvement in most of the aspects which requires much more focused and detailed work. Some of them are listed below:

- Adaptive and learning approaches may be included to system for traffic pattern identification and parking of the elevators.
- Checking the effect of the new request assignment to the old requests and assigning lifts with minimum effect.
- Implementing fuzzy approach of group controlling in a fuzzy embedded microcontroller for the validation of the algorithm.

9. Acknowledgments

We are greatly thankful Dr. Lalith Premarathna for taking keen interest in this research and for giving valuable inputs and encouragement. Finally we are thankful to all our fellow researchers at University of Colombo School of Computing for their constant support.

References

[1] Tan Kok Khiang , Marzuki Khalid, and Rubiyah Yusof - *Intelligent Elevator control by ordinal structure Fuzzy logic Algorithm* ICARCV 97, Singapore, Dec. 1996.

[2] Marja-Liisa Siikonen - *Elevator Group Control with Artificial Intelligence*. Technical Report, Systems Analysis Laboratory Research Reports, A67, Helsinki University of Technology, 1997.

[3] Fuzzy sets and Fuzzy Logic - Theory and Applications by George J. Klir and Bo Yuan - ISBN 81-203-1136-1

[4] An Introduction to Fuzzy sets - Analyse & Design by Witold Pedrycz and Fernando Gomide - ISBN 81-203-2465-x

[5] Marja-Liisa Siikonen - *Planning and Control Models for Elevators in High-Rise Building*. Research Report A68. System Analysis of Technology, Helsinki University of Technology. 1997.

[6]http://en.wikipedia.org/wiki/Fuzzy_control_system

[7] Matthew Brand Daniel and Nikovski *Optimal Parking in Group Elevator Control* IEEE International Conference on Robotics & Automation (ICRA)

[8] Robert H. Crites and Andrew G.Barto *Improving Elevator Performance using Reinforcement Learning* Adv. Neural Inform. Process. Systems 8, 1996