σ

Journal of Engineering and Natural Sciences
Mühendislik ve Fen Bilimleri Dergisi

Sigma

2004/1

NEURAL NETWORKS APPLICATION IN THE NEXT STOPPING FLOOR PROBLEM OF ELEVATOR SYSTEMS

C. Erdem İMRAK*, Mustafa ÖZKIRIM

İstanbul Teknik Üniversitesi, Makine Fakültesi, Makine Mühendisliği Bölümü, Gümüşsuyu-İSTANBUL

Geliş/Received: 27.05.2003 Kabul/Accepted: 07.01.2004

ASANSÖR SİSTEMLERİNİN SONRAKİ DURAK KATI PROBLEMİNE SİNİR AĞLARI UYGULAMASI

ÖZET

Hızlı nüfus artışına paralel olarak yüksek katlı binaların ve binalarda asasnör kullanılması kaçınılmaz olmıştur. Asansörlerden beklenen taleplerin artmasıyla, asansör kontrol sistemlerinin detaylı incelenmesi ve optimum asansör tasarımı için klasik kontrol sistemlerinin yanı sıra bilgisayar teknolojisinin kullanıldığı kontrol sistemleri konularındaki araştırmalar yaygınlaşmıştır. Asansör sistemlerindeki sonraki durak katı problemi, tek bir kabinin bilinen konumu, yöneldiği doğrultusu ve ayrı ayrı kaydedilmiş yukarı ve aşağı kat çağrılarının giriş ile sonraki durak katlarının çıkış olduğu problem olarak ifade edilir. Bu çalışmada, asansör kontrol ünitesi içindeki sonraki durak katlarını belirlemede yapay sinir ağları kullanılmış ve asansör trafik kontrolüne bu yöntemin uygulanması ele alınmıştır. Simülasyon sonucu katlararası taleple değişen normalize edilmiş performans oranı elde edilmiş ve sabir sektörlü algoritmalar ve dinamik sektörlü algoritma gibi diğer trafik kontrol algoritmalarıyla karşılaştırılmıştır.
Anahtar Sözcükler: Sinir ağları, Asansör sistemi, Kontrol, Durak katı

ABSTRACT

According to the rapid population growth, it is inevitable to the use of the elevator in high-rise building. The research involves that the investigation of the control systems of elevator and the classical control systems for the optimum design of elevator besides the control systems that use PC technology by the demand waiting from elevators. The next stopping floor problem of elevator system is stated that given a single car system with a known position, direction commitment and seperatly registered up and down landing calls as inputs find the NSF as an output of the network. In this study, artificial neural network is using to determine the next stopping floor (NSF) in a lift control unit, and the method applied to the elevator traffic control were also determined. Normalized system performance figure according to interfloor demand from simulation runs proposed neural networks embaded control algorithm is evaluated and compared with other elevator traffic control algorithms such as priority timed fixed sectoring algorithms and dynamic sectoring algorithm.
Keywords: Neural networks, Elevator system, Control, Stopping floor

* Sorumlu Yazar/Corresponding Author: e-mail: imrak@itu.edu.tr ; tel: (0212) 243 4771

1. INTRODUCTION

The role of an elevator system as a means of vertical transportation becomes increasingly more important in modern world. It was not until the beginning of the 1900's that the high level control of lift systems became necessary, though the first passengers elevator has installed in 1857. This was the period when bigger and taller building started to be built. As a consequence of this, more than one car in a building was needed in order to meet the demand for higher capacity of vertical transportation due to the growing size of the building population. The last two decades saw the emergence of various elevator control systems designed mostly by elevator manufacturers. The advances on their sophistication show the needs for a better control system. The majority of the earlier systems are based on the simple control algorithms. In the newer systems, solid state devices are employed [1,2].

Artificial neural networks are programs that simulate the methods in which human neural networks operate and learn. They use structures of nodes and links, where each link has a certain weighting. It is the purpose of the learning mechanism to alter these weights according to the difference between the calculated outputs and the correct outputs [3]. Each neuron has one output, which is generally related to the state of the neuron, and receives several inputs. The inputs are the activation of the incoming neurons multiplied by the weights. All of the knowledge that a neural network possesses is stored in the weights of the connections between the neurons [4]. The Sigmoid function is generally used as a threshold function because of simulating biological neuron activities [5].

This work is an exploration in the possibilities of the various fields in which neural networks can be applied in elevator control. Neural networks are usually applied in problems where no set of clear and decisive rules exist, and where the nature of the problem requires intuitive reasoning, like pattern recognition problems. Several training algorithms exist for training the neural networks, and the backpropagation method has been used in the program. Neural networks have been applied succesfully in many fields, like pattern recognition, process modeling and classification problems. The neural network approach has been applied to group control systems for improving passenger waiting time and a lift simulation software has been developed and implemented in order to assess the learning capability by measuring the performance of the control algorithm. The lift traffic analysis have been carried out by examining the simulation results obtained.

2. ELEVATOR CONTROL

The variation of the passenger movement results in a traffic pattern to be built up in a building. Although this is not the same for different buildings, there is usually a generalized pattern for a specific type of building. The conventional elevator system design method makes use of the up-peak traffic as a basis. This is based on an assumption that a system that can handle the up-peak traffic of a building would also be able to handle other traffic, namely the interfloor and down-peak. However, this does not guarantee that service provided to the latter will be acceptable, because it is dependent on the characteristics of particular control algorithm employed [6,7].

The control of the operation of an elevator system can be separated into levels, namely the low and high level controls. The former is concerned with commanding of individual cars to move up or down, to stop the car, and to open and close the doors, while the latter has the function of coordinating the activity of a group of cars working together where there is more than one car present in the system. The high level control works on the basis of logical rules defined by the designer. The primary task of an elevator control algorithm is to enable the cars to answer the car and landing calls in the most appropriate way.

The above problems are due to the random nature of the time and landing at which passengers arrive and request service. This means that control algorithm must be able to follow the change in passenger demand at all times. Elevator traffic control systems have become more complicated because of their intelligent nature. The aim of elevator group control is to facilitate effective operational management of elevators and to select the right cars to answer the hall calls. Elevator traffic control systems are used to control the cars as a group and take passengers to their destinations pleasantly and promptly. The selection and distribution of the most suitable cars in the building are a function of the assignment of calls. Neural networks have been applied to overcome this problem in the elevator control algorithm [8]. In elevator systems, the parameters such as number of passengers, number of calls and directions are changed not only in any service period but also in different building types. Elevator control, therefore, has to pose a flexible control module to adapt to these changes.

2.1. Design of Elevator Control Algorithm

The actual number of passengers using the lift system in different traffic conditions and the profiles of the traffic pattern have to be avaliable. The exact traffic pattern that exits in an individual building depends a great deal on its typicalities, of which the most important is the type or nature of use of the buildings. In the design of lift systems, the traditional method has been on calculating the round trip time, which relies on calculating the probable number of stops and the highest reversal floor. The probable number of stops is given as a function of the number of passenges boarding the car. It is proposed lose that this method can be reversed, in order to find the probable number of passengers boarding from the knowledge of the number of stops [9].

Design of control algorithm is not an easy task to accomplish. Elevator designers are presented with the problem of having to satisfy the random arrivals of passenger demand at all times. Several different techniques have been developed by manufacturers. Each technique has its own good and bad features. A very careful selection is therefore necessary when applying these existing technique. Landing and car calls with car positions are the input data of the control system. The building configuration also directly effects the control unit and neural networks unit. The outputs of the control system are the distribution of calls to the most suitable cars and car directions. There are two alternative methods which can be employed when designing the algorithm.

If the actual elevator installations were to be used, the simulation methods are employed to facilitate design studies, where it is too expensive or would cause disturbance. In the case of elevator systems on existing installations require long periods of observation to obtain meaningful data. Simulations can be performed by digital computer. The advent of time sharing digital computers and graphic terminals have made it possible for designer to obtain an extremely close contact with the design by means of conversational program techniques and by means of many types of graphical presentation.

3. NEURAL NETWORKS APPLICATION IN NEXT STOPPING FLOOR PROBLEM

The availability of low cost computing power as offered by microcomputers opens up a new dimension in the control of elevator systems. The microcomputers can be used to gather data from the elevator system and then make decisions on how the cars should be driven. The artificial intelligence traffic modeling and prediction system using neural networks for elevator groups, allows a suitable allocation of landing calls to cars in order to serve the calls with regard to a function profile. This profile is defined by a desired combination and weighting of elements from a predetermined set of function requirements and in which a suitable landing call allocation corresponds to the waiting times [9,10].

When applied to elevator installations, neural network algorithms are able to dynamically learn the behavior of the building traffic characteristics and predict future events based on what has been learnt. The application of neural network algorithms in elevator traffic control systems may shorten the waiting time by forecasting car position and using call distribution laws. These systems can recognise the changes in traffic patterns during the day and automatically adapt the control system [11,12].

In particular neural networks have been used to solve the next stopping floor problem. The next stopping floor (NSF) problem is stated as: given a single car system with a known position, direction commitment and registered up and down landing calls as inputs, find the NSF as an output of the network.

The software using multiplayer network for solving the NSF because of the size of the training pattern, its kept in a separate file produced by the training pattern program. All the input patterns were introduced for several epochs. The next step is to calculate the number of inputs and outputs for such a system, as a function of the number of stops n. The following is a derivation of these two numbers and it assumes the use of linear decoding rather than binary decoding for the sake of simplicity.

The program uses these training patterns for training. Because of the size of the training pattern, it is kept in a separate file produced by the training pattern program. A training pattern consists of training pairs of inputs and outputs. The inputs are car calls, up and down landing calls and car direction. The outputs are the next stopping floors. This training pattern is generated depending on the number of floors and recorded in a binary coded format as shown in Figure 1.

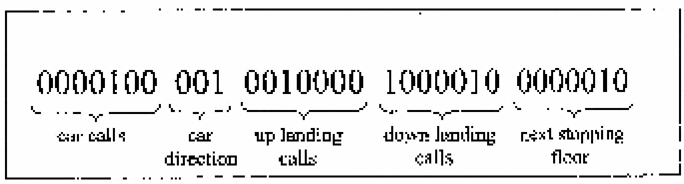

Figure 1. A sample file record representing a training pair

In order to train the neural network to be able to calculate the next stopping floor, the set of possible patterns has to be evaluated in advance, and stored in a file for a later use by the training program. The file contains the data which are obtained by running the algorithm of duplex/triplex control logic. In the file, each record consists of a binary code where "1" means that there is a call and "0" means that there is no call. In the sample file record given in Figure 1, the first group of digits represent the car calls, the next three digits represent the car direction ("001" for down direction and "100" for up direction), the third group of digits represent the up landing calls, the fourth group of digits represent down landing calls, and the last group of digits represent the next stopping floor assigned. The length of digits showing calls and the next stopping floors are determined by the number of floors. The program stops if the sum of the squares of the errors is less than 0.01 or after 1,000 epochs [10].

3.1. Structure of Simulation Program

The flow chart of traffic analysis and control module, having neural network algorithm is shown in Figure 2. In this control system, the algorithm learns the passengers' demand pattern throughout the day and sets the car movements in the building witha parking function. The traffic analysis and control module decides which car to be dispatched to which floor [8, 13].

The traffic analysis and control module is situated in the simulation program and allocates the calls to suitable cars. The main executive program generates landing and car calls as illustrated in Figure 2. The car capacity is a variable input as required. The aim of the module is to obtain the next stopping floors in order to allocate the calls to the cars. In the first step by considering the landing calls and car calls, the next arrival floors and next destination floors are determined respectively. The number of passengers, which is another component of next arrival floor, is determined using Inverse-Stop-Passenger (ISP) method [14]. Al-Sharif derived a formula expressing the probable number of passengers as a function of the number of stops as follows :

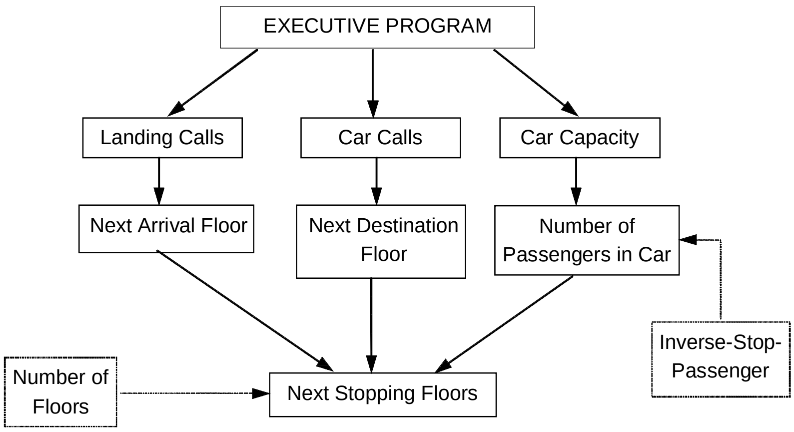

Figure 2. The flow chart of traffic analysis and control module

$$P = \frac{\ln\left(\frac{N-S}{N}\right)}{\ln\left(\frac{N-1}{N}\right)} \tag{1}$$

where *S* is the probable number of stops, *N* is the number of flows served above the main terminal, and *P* is the number of passengers. ISP method which is developed by Al-Sharif using equation (1) to determine the probable number of passengers in the car according to assigned car calls. When the algorithm developed decides which car should be allocated to the hall calls. It considers the number of passengers estimated by this method as well as other parameters such as car direction and car calls.

In fully connected neural networks, input and output layers are configured by number of floors, and two hidden layers are used. The Sigmoid non-linear function is used as threshold function. To improve the generalization of the backpropagation algorithm, the number of hidden layers are double that of the number of input layer nodes. The software uses the multi-layer backpropagation method for determining the NSF. Because of the size of the training pattern, it is kept in a separate file produced by the training pattern program. All the input patterns were introduced for several epochs. The next step is to calculate the number of inputs and outputs for such a system, as a function of the number of stops *n*. The derivation of these two numbers assumes the use of linear decoding rather than binary decoding for the sake of simplicity.

The flow chart of the simulation program which employs both conventional control and the neural network algorithm is shown in Figure 3 [10,13]. The algorithm also learns the passenger's demand pattern throughout the day and sets the car movements in the building with a parking function.

In order to predict future values of the traffic patterns, past values are needed. Neural networks with hidden layers have been used to build a one step ahead predictor for lift traffic patterns. The response of the lift system can then be predicted using simulation. The program calculates the outputs by multiplying the input matrix by the weight matrix to produce the output vector, and then it sets the highest output to “1”, while setting the all other outputs to zero. It then compares this answer with the correct answer and then adjust the weights. Several aspects of the lift system behavior have been recorded: the round trip time, the interval and the average passenger waiting time.

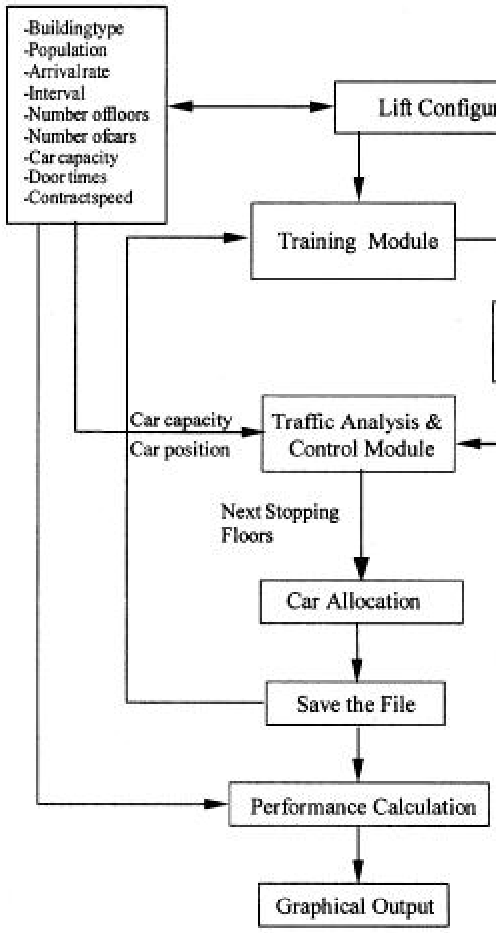

Figure 3. The flow chart of simulation program

3.2. Backpropagation

Backpropagation is a systematic method for training multilayer artificial neural networks. It has a mathematical foundation that is strong if not highly practical. The invention of the backpropagation algorithm has played a large part in the resurgence of interest in artificial neural networks. In general, neural networks can be though of as “black box” devices that accept inputs and produce outputs. One of the operations that neural networks perfom is control. In control an input pattern represents the current state of a controller and the desired response for the controller and the output is the proper command sequence that will create the desired response.

The backpropagation algorithm is a model-free function estimation with neural networks. It is possible to derive the backpropagation algorithm with a few iterated applications of the chain rule to differential calculus. The backpropagation algorithm converges to a local error minimum, if it converges at all. The Sigmoid function defines the nonlinearities. The backpropagation algorithm has generally three layers of neurons: an input layer, a hidden layer and an output layer. The hidden layer size is somewhere between the input layer size and the output layer size [15].

Multilayer backpropagation networks have a layer of input neurons at the beginning, any number of hidden layers and a layer of output neurons at the end. In general connections within a layer or from a higher to a lower layer are forbidden, but connections may skip intermediate layers. The architecture of mutlilayer backpropagation network is shown in Figure 4.

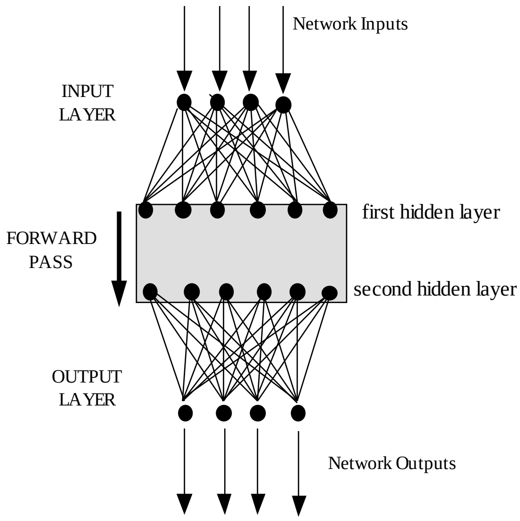

Figure 4. The architecture of multiplayer backpropagation

To begin with, the network learns a predefined set of input-output example pairs by using a two-phase propagation adoption cycle. After an input pattern has been applied as a stimulus to the first layer of network units, it is propagated though each upper layer unit until an output is generated. This output pattern is then compared to the desired output, and an error signal is computed for each output unit [4]. The errors are then transmitted backwords from the output layer to each node in the intermediate layer that contributes directly to the output.

After training when presented with an arbitrary input pattern the units in the hidden layers of the network will respond with an active output if the new input contains a pattern that resembles the feature the individual units learned to recognize during training.The input layer configuration depends on the possible parameters that effect the network outputs. The backpropagation training algorithm is an iterative gradient descent algorithm that attempts to minimize the mean square error between the actual network output and the desired one. The invention of the backpropagation algorithm has played a large part in the resurgence of interest in neural networks. Backpropagation is a systematic method for training multilayer neural networks [15].

3.3. Learning Algorithm

In backpropagation algorithm, the training set is first presented to the network and then secondly the error at the output nodes is reduced along the steepest descent direction. The initial weights and the thresholds are randomly generated at the beginning [1]. Adjusting the two sets of weights between the pairs of layers and re-calculating the outputs is an iterative process that is carried out until the errors fall below a tolerance level. Increasing the number of hidden layers may improve the generalization capacity. Two hidden layers are preferred to achieve good convergence. Learning rate parameters scale the adjustments to weights. Learning rate is used for convergence and this convergence is drawn according to epoch with constant momentum. The input and output parameters are normalized with learning rate within the range 0.2 to 0.8.

3.4. Simulation

To investigate the improvement produced by this control system, neural networks were introduced into a duplex/triplex elevator control algorithm and optimization is carried out on many parameters such as waiting time and various performance criteria [9,13]. The program uses these training patterns, in order to train the neural network to be able to calculate the NSF. The set of possible patterns has to be evaluated in advance, and stored in a file to be used later by the training program. The simulation program is executed for a test building whose technical specifications are given as follows:

Building type	: Office	Door type	: Side open
Building population	: 1000	Door with	: 800 mm
Arrival rate	: % 15	Door opening time	: 2 s
Interval	: % 30	Door clossing time	: 2.6 s
Number of floors	: 9	Interfloor distance	: 3.3 m
Number of cars	: 3	Single flight time	: 5 s
Car capacities	: 6	Acceleration	: 0.5 m/s^2
Contract speed	: 1.0 m/s	Passenger transfer time	: 1.2 s

In this program interfloor distance, contract speed, time for opening and closing the doors, passenger transfer time are chosen by users. The program runs on a 486PC with 8 MB RAM, 256MB hard disk configuration.

In the design of lift traffic and calculating the performance of a lift system, the traditional method is to calculate the round trip time (RTT), which relies on calculating the average number of stops made, the average highest reversal floor and the average number passengers carried [8, 9]. According to Barney [9], lift systems are sized for the uppeak traffic pattern by using the following formulae:

$$RTT = 2 \cdot H \cdot t_v + (S+1) \cdot t_s + 2P \cdot t_p \qquad (1)$$

where *H* is highest reversal floor, *S* is number of stops, *P* s number of passengers, t_v is single floor transit time, t_s is stopping time and t_p is passenger transfer time.

Then the number, car capacity and the speed of the elevators can be derived in order to provide a reasonable interval (RTT / number of lifts) and five minute handling capacity (300 x P / interval). The uppeak interval is

$$UPPINT = \frac{RTT}{L} \qquad (2)$$

where *L* is the number of lift cars witin a lift group. The rated car capacity is

$$RC = \frac{300 \cdot P}{UPPINT} \,. \tag{3}$$

For car loads less than 50 % the awerage waiting time is

$$AWT = 0.4 \cdot UPPINT \,, \tag{4a}$$

and for car loads over 50 % it is possible to develop an equation for the *AWT* as [9]:

$$AWT = \left[0.4 + \left(\frac{1.8 \cdot P}{RC} - 0.77 \right)^2 \right] \cdot UPPINT \,. \tag{4b}$$

Simulations are run for several levels of interfloor traffic demand for the improved control system (ICS). At the end of each simulation, the average waiting time is recorded and the performance figure calculated by dividing the average waiting time by the interval to normalise it. The normalized system performance figure is defined as the percentage of the simulated average passenger waiting time divided by calculated uppeak interval. The interfloor demand (β) is defined as the interfloor arrival rate divided by calculated uppeak handling capacity. The normalized performance figures, with respect to the interfloor demand are compared with other traffic control algorithms which are priority timed fixed sectoring algorithm (FS4), bi-directional fixed sectoring algorithm (FS0), dynamic sectoring algorithm (DS), computer group control algorithm (CGC), hall call allocation algorithm (ACA) as shown in Figure 5 [8].

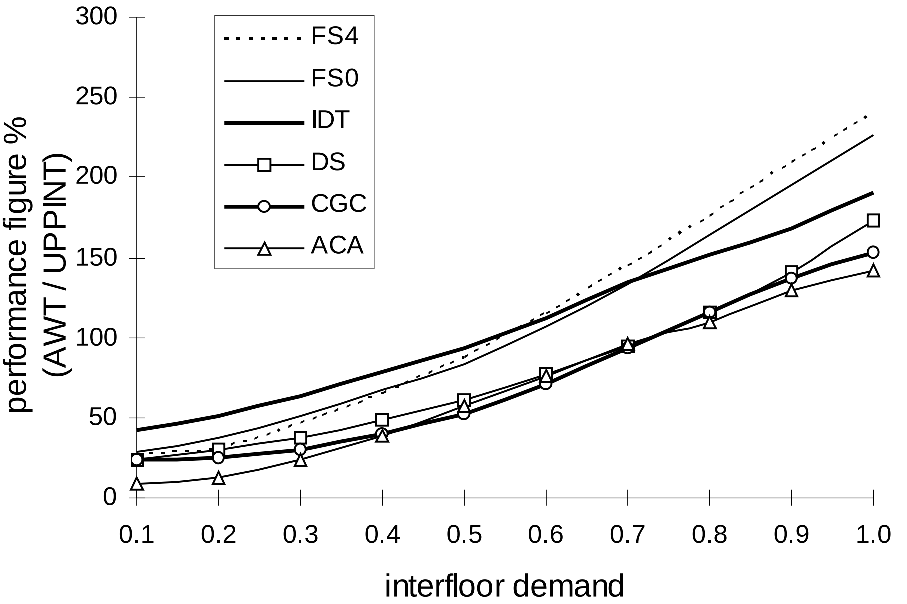

Figure 5. Interfloor performance of the control algorithms

Figure 5 gives the comparison of the performance figure of the control algorithms. IDT algorithm shows improvement over the FS4 and FS0 control algorithm over 55% interfloor demand. It also shows that improved duplex/triplex system is less effective for balanced interfloor traffic compared with DS, CGC and ACA algorithms.

5. CONCLUSIONS

The reliable service of the lift system of any modern building is necessary for the efficient functioning of any building. The service provided by the lift system has to be not just reliable, but also convenient. For these reasons, in order to ensure reliability of operation, it is necessary to make sure that they operate continuosuly without interuptions. The possible use of control theory in lift systems is outlined in detail. The application of neural network algorithms in elevator traffic control systems may shorten the waiting time by forecasting car position and using call distribution laws.

Under a heavy interfloor demand, the neural network embaded IDT system shows better performance than the fixed sectoring systems (FS4 and FS0), but it still has a poorer performance against the other computer-based traffic control systems. In low rise buildings, it gives a better performance against the conventional traffic control systems. To improve IDT interfloor performance, the neural network algorithm will be improved or changed backpropagation algorithm with other one.

REFERENCES

[1] Barney G.C., Dos Santos S.M. “Elevator Traffic Anaylsis Design And Control. Peter Perigrinus Ltd., 1985, 85-146.

[2] Mukherjee A., Deshpande J.M. “Application of Artificial Neural Networks in Structural Design Expert Systems” Computers&Structures Vol.54, 3, 367-375, 1995.

[3] Mulvney D., Hamdi M., “Real–time Dynamic Scheduling and Its Application to Lift Systems”, Proceedings of ELEVCON 2002, Elevator Technology 12, IAEE Publ. 2002, 207-216.

[4] Antsaklis P.J. “Neural Networks in Control Systems”, IEEE Control Systems Magazine Vol.12, 2, 8-10, 1992.

[5] Rumelhart D.E., Hinton G.E., Williams R.J., “Learning Representations by Backpropagation Errors”, Nature, Vol.13, 533-536, 1986.

[6] Swindells W. “Software for The Computer Control of Lift Systems”, MSc Thesis, UMIST, 1975.

[7] Closs G.D., “The Computer Control of Passenger Traffic in Large Lift Systems”, PhD. Thesis, UMIST, 1970.

[8] İmrak C.E., “Asansör Sistemlerinin Trafik Analizi, Dizaynı ve Simülasyonu”, PhD. Thesis, Inst. Of Sci. & Tech. İTÜ.,1996.

[9] Barney G.C., “Elevator Traffic Handbook Thoery and Practice”, Spon Press, London. 2003, 269-284.

[10] İmrak C.E., “Asansör Kontrol Sistemleri ve Trafik Analizi”, 7.Uluslararası Makina Tasarımı ve İmalatı Kongresi”, ODTÜ, Ankara, Eylül 1996, 351-361.

[11] Powell B.A., Sirag D.J., Whitehall B.L., “Artificial Neural Networks in Elevator Dispatching”, Lift-Report, Vol.2, March/April 2001, 14-19.

[12] Sasaki K., Markon S., Nakagawa M., “Elevator Group Supervisory Control Systems Using Neural Networks”, Elevator World, 1996, 81- 86.

[13] İmrak C.E., Barney G.C., “Application of Neural Networks on Traffic Control”, Proceedings of ELEVCON’98, Elevator Technology 9, IAEE Publ, 1998, 140-148

[14] Al-Sharrif L., “Prdedictive Methods in Lift Traffic Analysis”, PhD. Thesis, UMIST, 1991.

[15] Lloyd J.A., “An Introduction to Artificial Neural Networks”, MSc. Thesis, UMIST, 1990.