EFFECTS OF FUZZY LOGIC METHODS OVER ATM NETWORKS

İman Askerbeyli[1] & Fidan Aybike Gedik[2]
[1] Department of Computer Engineering, Ankara University, Ankara, Turkey
[2] Department of Computer Engineering, Ankara University, Ankara, Turkey
Email: [1]Iman.Askerbeyli@eng.ankara.edu.tr, [2]fidanaybike.gedik@turktelekom.com.tr

ABSTRACT
In recent years, fuzzy logic methods has become one of the methods mostly used because of finding solution fast and low cost. One of these methods, the adaptive fuzzy logic method, adapted to the relevant technology and provide the optimum solutions. In this work a network traffic controller developed for minimizing error rate of multiple switches ATM Network with adaptive neuro- fuzzy method. This method was compared with a conventional fuzzy method to show the advantages and better results. ABR (Available Bit Rate) service of the ATM technology is used when designing and developing the controller. Results are discussed within the framework of today's technology requirements and problems.

Keywords: *Fuzzy logic, adaptive network fuzzy inference system, multiple switch traffic control, asynchronous transfer mode, available bit rate*

1. INTRODUCTION

Due to the nature of mathematics modeling arise spontaneously. Because mathematics has its own distinct rules, adjusting the problem to mathematics is difficult and inessential mostly. Therefore , we can try to adjust mathematics rules to problem. For example, there is no known mathematics rules for estimating nonlinear processes. We can access to solution of a nonlinear problem with comparing to a linear modeling. However, the linear models depends on input variables, such that If input values changes a little , the output values differs completely.

After the 1950s, the search for solutions to problems of systems analysis and probability theory has been used frequently. After a while, the fact that uncertainty overlaps with the real life started to accept. The concept was first articulated by Lotfi Zadeh in a paper published in 1965, ("Fuzzy Sets," Information and Control 8:3, 338-53) which provided the theoretical basis for fuzzy computer chips which appeared 20 years later. Unlike traditional logic, which attempts to categorize information into binary patterns such as hot and cold, tall and short. Fuzzy Logic pays attention to the "excluded middle" and tries to account for the "grays", the partially true and partially false situations which make up 99.9% of human reasoning in everyday life. [1]

Fuzzy logic has very broad application areas. The most valuable benefit is "experience of people specific learning". Fuzzy logic enables the modeling undefined concepts and explaining in mathematic with this benefit. Therefore, it is very suitable way to approaching nonlinear systems. There were applications that are developed with fuzzy logic in finances, geography, philosophy, agricultural processes, water treatment.

Communication systems is one of areas that developed solution with fuzzy. The steady increase in the usage of communication network on technological applications at the last years, managing traffic flow on the online network is a necessary issue today. However development the efficient management of network resources is a complicated task. With traditional network management methods it is difficult to obtain a comprehensive view of the state of the network and simultaneously discover important details from the traffic.

Looking more specific to communication network, The Asynchronous Transfer Mode (ATM) provides one approach that looks very promising from the perspective of traffic engineering. Recently, however, there is the popular belief that the Internet paradigm will dominate the end-to-end communication of advanced end systems. Nevertheless, there is still interest in the development of ATM, at least as the underlying technology of a high-speed backbone network.

ATM is a modern technology enabling the integration of different traffic types with in a single communication network. ATM Networks are being developed to carry data, video and voice traffics. Various service classes have been defined in ATM port he supports of traffic with different quality of service (QoS) requirements. These classes consist of constant bit rate (CBR), real-time variable rate (rt-VBR), non-real time variable bit rate (nrt-VBR), available bit rate (ABR) and unspecified bit rate(UBR). Of these, Available Bit Rate (ABR) service has been introduced to support highly busty traffic data applications.[2]

Traffic control management systems based on the decreasing the congestion over switches. If the congestion persists for long period of time, the buffer storage will be to maximum capacity and any additional data must be discarded on the ATM networks. In an effort to minimize such data loss, an ABR flow control scheme developed with various control techniques.

FERM (Fuzzy Explicit Rate Marking) is an explicit rate control algorithm, which means that it calculates the maximum transmission rate of each switches. FERM explicitly specifies this rate through the mechanism of resource management (RM) cells. [3]

FERM is set input functions according to network at the first time. This manual processing is done by an expert. They makes calculation overall network situation at any time or maybe makes prediction for maximum and minimum flow rates and then decided to premise parameters and functions to adapt to network most efficiently. This task takes a lot of time for research and development works. This means it , this task is very costly.

Therefore, setting input function only one time is not applicable for online networks so real world networks. Effective method is needed for tuning the membership functions according to flow rate of the network at some times. This method can minimize the output error measure and decreasing time and cost when adjusting parameters of the membership functions. Computational intelligence and artificial intelligence methods can be integrate to develop method like this. Using positive points of both intelligence methods, a new method can be generate that has representation of fuzzy logic and ability of adapt the knowledge base.

ANFIS (Adaptive Network Fuzzy Inference System) gives us an alternative approach to design a controller. ANFIS needs only initial membership functions and it's trained itself with training data. So there is no need for expert and costly research tasks. After training , adjusting input parameter so our input functions are changed for maximizing flow rate of network in an new situation.

First of all, I work on a single switch to measure the effect of the ANFIS and conventional Fuzzy Logic controller. Results are remarkable. However, our method must work for multiple switches network that will real one to apply for used ATM networks at the present day with current technology. In this project, I try to show the ANFIS traffic control mechanism over ATM networks that have multiple switches has better performance than traditional fuzzy logic control mechanisms.[4]

2. BACKGROUND WORK

2.1 ATM

ATM network management becomes much more difficult due to the various types of traffic expected on ATM networks; hence, guaranteeing the Quality of Service (QoS) is much more challenging. In order to meet their strict QoS requirements, CBR and VBR sources are guaranteed bandwidth when they are admitted into the network. UBR traffic is only minimally supported by the network. The functions of CAC and traffic isolation mechanisms are to preserve enough bandwidth for CBR and VBR services and to ensure the controllability of the whole network. ABR congestion control aims to maintain maximum ABR throughput by adjusting each ABR source rate based on the overall traffic status of the network than ever before. However, traffic controlling is discussed in this project, ABR flow control mechanisms mention in here as we used in our developed algorithm.

The flow control mechanism to indicate at which rate the source router can transmit uses explicit cell rate information written by the intermediate ATM NEs in the payload of the RM cell. The RM cell was generically defined for traffic control purposes, and ABR is simply a specific instance of its use. More specifically, with the explicit rate flow-control method, a source router places its current flow rate in the current cell rate (CCR) field. Intermediate switches explicitly communicate the rate at which the source is allowed to send at that given moment by placing a value in the ER field. The source router reads the ER field and adjusts its ACR(Allowed Cell rate) to match the ER as long as the calculated rate is not less than the minimum cell rate.[5]

2.2 Fuzzy Logic

Fuzzy control denotes the field in control engineering in which fuzzy set theory and fuzzy inference are used to derive control laws. The concept of a fuzzy set is an extension of the concept of an ordinary set, called a crisp set. For a crisp set X, an element either belongs to X and shown with number 1 or not and shown with number 0. However, for a fuzzy set F(X), and element has real number in the closed interval [0, 1]. Fuzzy set defined by membership functions. Any value between 1 and 0 can express the grade of membership function which an element belongs to this fuzzy set [6]. The concept of fuzzy sets makes it possible to use fuzzy inference. In the method of fuzzy inference, the knowledge of an expert in a field of application is expressed as a set of "IF-THEN" rules, leading to algorithms describing what action should be taken based on currently observed information. Rules are usually expressed in the form:

IF *variable* IS *property* THEN *action*

Fuzzy controllers are the applications of fuzzy sets and fuzzy inference in control theory [7]. Their operation is divided into four phases, called *fuzzification*, *rule base*, *decision making* and *defuzzification*.

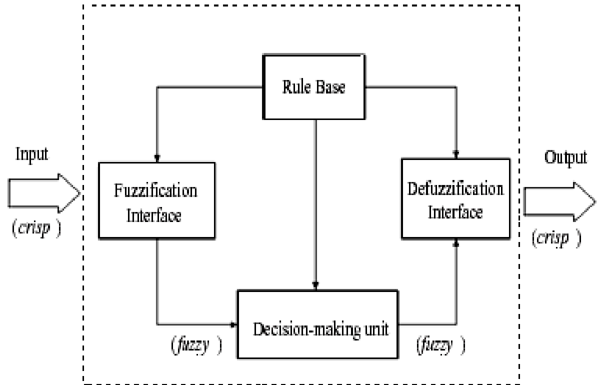


Figure 2.1 Fuzzy inference schemas

In order to obtain the control design for a nonlinear or complex dynamic system, there are four basic steps in designing a conventional fuzzy logic controller (FLC) for a physical system: fuzzification, the decision making of fuzzy control rules, fuzzy inference logic, and defuzzification. The inference operations upon fuzzy if-then rules performed by fuzzy inference systems are described as follows.[8]

In the fuzzification part, the in/out variables of a fuzzy controller can be divided into system variables, and linguistic variables. The fuzzy control rule is important to the successful operation of the fuzzy control system. The rule base (knowledge base), containing a number of fuzzy if-then rules, is composed as follows:

$$\mathrm{R}^{(\mathrm{m})}: \mathrm{IF}\ \mathrm{x}_1 \text{ is } \mathrm{A}_1^{\mathrm{m}} \text{ and} \dots \text{and } \mathrm{x}_\mathrm{n} \text{ is } \mathrm{A}_\mathrm{n}^{\mathrm{m}}\ \mathrm{THEN}\ \mathrm{y} = \mathrm{K}_\mathrm{m} \qquad (1)$$

Where $x = (x_1, ..., x_n)^T$ and y are the input and the output of the fuzzy logic system, respectively. $\mathrm{A}_\mathrm{i}^{\mathrm{m}}$ is the label of the fuzzy set in i, for $m = 1,2,...,M$, and K_m is the zero-order Sugeno parameter.

The sup-algebraic product compositional rule of inference is used in third part of the fuzzy inference system. In order to obtain the correct control input for this control system, it is necessary to defuzzify the fuzzy sets and aggregate the qualified consequent parts to produce a crisp output at the last part.

2.3 ANFIS

ANFIS (Adaptive Network Fuzzy Inference System) is an adaptive neuro- fuzzy network structure consisting of nodes and directional links through which the nodes are connected. Adaptive means output nodes depend on parameters pertaining to these nodes. Actionallty all nodes have not to be adaptive, part of them can be adaptive nodes. Learning rule specifies how these parameters should be change to minimize an error measure [9].

Learning rules are decided for minimizing error measure with changing these parameters. Gradient descent method is most basic learning rule in ANFIS. In this project, gradient method and least squares estimate (LSE) to identify parameters. For simplicity, our ANFIS structure has premise parameters and consequent parameters. If the parameter set P can be composed two set $P_{premise}$ and $P_{consequent}$

$P = P_{premise} + P_{consequent}$

Hybrid algorithm has two pass, forward pass and backward pass. Hybrid algorithm update premise parameters of ANFIS network at the forward pass with least square estimation (LSE). The output of a model y is given by the parameterized expression [8].

$$y = \theta_1 f_1(u) + \theta_2 f_2(u) + \cdots \theta_n f_n(u) \qquad (2)$$

where $\mathbf{u} = [\mathrm{u}_1 \dots \mathrm{u}_\mathrm{n}]^{\mathrm{T}}$ is the models input vector, $f_1,\dots\dots,f_n$ are known functions of u, and $\theta_1,\dots\dots,\theta_n$ are unknown parameters to be optimized. To identify these unknown parameters θ_i , usually a training data set of data pairs $\{(\mathrm{u}_\mathrm{i}, \mathrm{y}_\mathrm{i}), \mathrm{i} = 1, \dots, \mathrm{m}\}$ is taken; substituting each data pair in (2) a set of linear equations is obtained, which can be written as

$$A\theta = y \qquad (3)$$

in the matrix form. Where A is a mxn matrix

$$\mathrm{A} = \begin{bmatrix} \mathrm{f}_1(\mathrm{u}_1) & \dots & \mathrm{f}_\mathrm{n}(\mathrm{u}_1) \\ \vdots & \vdots & \vdots \\ \mathrm{f}_1(\mathrm{u}_\mathrm{m}) & \dots & \mathrm{f}_\mathrm{n}(\mathrm{u}_\mathrm{m}) \end{bmatrix} \qquad (4)$$

θ is nx1 unknown parameter vector

$$\theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_\mathrm{n} \end{bmatrix} \qquad (5)$$

Y is an mx1 output vector

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} \tag{6}$$

Since generally m>n, instead of exact solution of (3) an error vector e is introduced to account for he modeling error as

$$A\theta + e = y \tag{7}$$

and searched for $\theta = \ddot{\theta}$ which minimizes sum of squared error

$$E(\theta) = \sum_{i=1}^{m} \left(y_i - a_i^T\theta\right)^2 = e^T e \tag{8}$$

Where $E(\theta)$ is called the objective function. The squared error in (8) is minimized when $\theta = \hat{\theta}$, called Least squared estimator(LSE) that satisfied the normal equation

$$A^T A\hat{\theta} = A^T y \tag{9}$$

If $A^T A$ is non singular, $\hat{\theta}$ is unique and is given by

$$\hat{\theta} = (A^T A)^{-1} A^T y \tag{10}$$

In case of back propagation learning rule the central part concerns how to recursively obtain a gradient vector in which each element is defined as the derivative of an error measure with respect to a parameter. In this part $P_{consequent}$ are constant and $P_{premise}$ are updated with gradient descent method with output of the network is propagate through back. Following formula is used for updating premise parameter at back propagation pass.

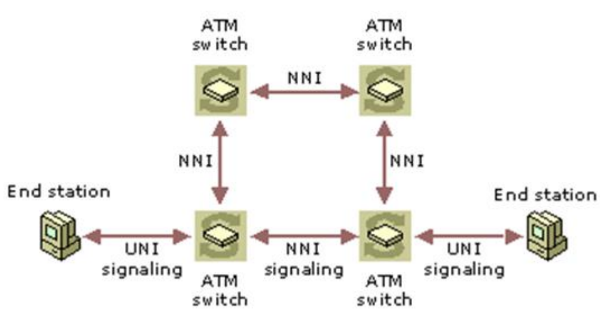
$$\Delta x = -\eta \frac{\partial E}{\partial x} \tag{11}$$

Where x is premise parameters, η training rate, E is error value at out of the network.

In the forward pass, when $P_{premise}$ is presented, the node outputs of the system are calculated layer by layer till the corresponding row in the matrices **A** and **y** of equation (3) are obtained. The process is repeated for all the training data pair to form the matrices **A** and **y** completely. Then the output parameters of set $P_{consequent}$ are calculated according to the equation (10) After this, the error measure for each training data pair is to be calculated. In the backward pass, these error signals propagate from the output end towards the input end. The gradient vector is found for each training data entry. At the end of the backward pass for all training data pairs, the input parameters are updated by steepest descent method as given by equation (11).

3. CONTROLLER DESIGN

ATM traffic controller has been thought of as consisting of three inputs and one output. It takes QLen (Average queue length) , DeltaQ (change in average queue length) and Ave.FFR (average Fractional Flow Rate of previous switches) as its input. An example ATM network with multiple switches is shown in Figure 3.1. In our method, third input of atm switch near the end station is calculated with arithmetic average of flow rate of other three atm switches. Queue length is averaged over an interval called as an Averaging Interval (AI). The queue growth rate or DeltaQ, is basically the difference between the QLen of two consecutive AIs. Output of controller is FFR (Fractional Flow Rate),which is the value between 0 and 1.



Şekil 3.0-1 An example ATM network with multiple switches

3.1 Conventional Fuzzy Controller

We discussed the explicit algorithm section 2. Explicit algorithm is developed with conventional fuzzy logic methods. In the explicit algorithm, explicit rate calculated with Fractional flow rate. FFR (Fractional Flow Rate) is multiplied with the link cell rate to get the ER (Explicit rate), which is then conveyed through the returning RM cell to the sources as the new rate. The sources then adjust their ACR (Allowed Cell Rate) according to the new ER conveyed.[10]

ER = FFR * Link Cell Rate

where, ER is Explicit Rate (Mb/sec),FFR is Fractional Flow Rate (between 0 and 1), Link Cell Rate is maximum speed of output physical link.

Let's start adapt to conventional fuzzy controller.. first step is fuzzification step of the fuzzy inference system. In this step we have three inputs, Qlenght, DeltaQ and ave FFR.

Bell shaped membership function type is used for first input. The graphic of the bellshaped member ship function that is obtained with applying equation 12 is shown in the figure 3.2. The data interval is [0,1000] and QLenght is defined by three linguistic term; Empty, Moderate, Full.

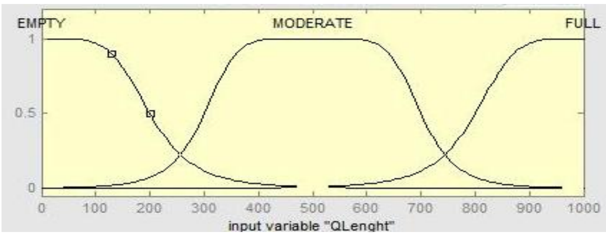


Figure 3.2 Membership function of the Qlenght input

$$\mu_{A_i}(x) = \frac{1}{1 + \left(\frac{x - c_i}{a_i}\right)^{2b_i}} \qquad (12)$$

Bell shaped membership function type is used for DeltaQ input, as well. The graphic of the DeltaQ member ship function is shown in the figure 3.3. In here, [200 -300] data interval is chosen for 5 Linguistic variables are *Decreasing Fast, Decreasing Slow, Zero, Increasing Slow, Increasing Fast*

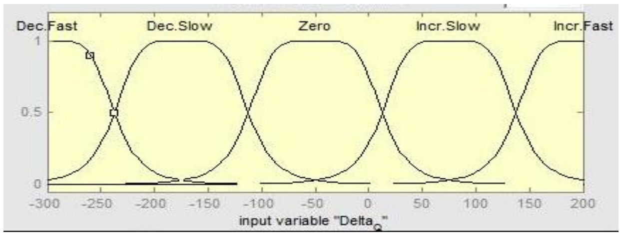


Figure 3.3 Membership function of the DeltaQ input

$$\mu_{A_i}(x) = \frac{1}{1 + \left(\frac{x - c_i}{a_i}\right)^{2b_i}} \qquad (13)$$

Bell shaped membership function type is used for ave FFR input, as well. The graphic of the Ave FFR member ship function is shown in the figure 3.4. In here, [0 1] data interval is chosen for 3 Linguistic variables are *Low, Normal, High*

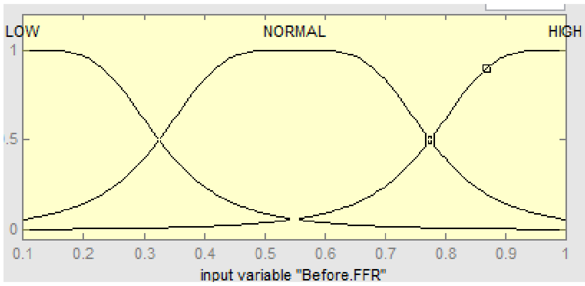


Figure 3.4 Membership function of the Ave.FFR input

$$Average\ Fractional\ Flow\ Rate(Awe.FFR) = \frac{1}{n}\sum_{i=1}^{n} FFR_i \qquad (14)$$

Second step is rule base. We have 29 fuzzy rules come by 3 linguistic terms for Qlength and 5 terms for DeltaQ and 3 term for Ave FFR. Following if-then rules shows these fuzzy rules constituted.

If Qlenght is EMPTY,flow rate of switch is 1
If Qlenght is MODERATE and DeltaQ is DEC.FAST and prev. Ave.FFR is LOW,flow rate of switch is 2
If Qlenght is MODERATE and DeltaQ is DEC.SLOW and prev. Ave.FFR is LOW,flow rate of switch is 2
If Qlenght is MODERATE and DeltaQ is ZERO and prev. Ave.FFR is LOW,flow rate of switch is 3
If Qlenght is MODERATE and DeltaQ is INC.SLOW and prev. Ave.FFR is LOW,flow rate of switch is 4
If Qlenght is MODERATE and DeltaQ is INC.FAST and prev. Ave.FFR is LOW,flow rate of switch is 5
If Qlenght is MODERATE and DeltaQ is DEC.FAST and prev. Ave.FFR is NORMAL,flow rate of switch is 2
If Qlenght is MODERATE and DeltaQ is DEC.SLOW and prev.Ave.FFR is NORMAL,flow rate of switch is 3
If Qlenght is MODERATE and DeltaQ is ZERO and prev. Ave.FFR is NORMAL,flow rate of switch is 4
If Qlenght is MODERATE and DeltaQ is INC.SLOW and prev.Ave.FFR is NORMAL,flow rate of switch is 5
If Qlenght is MODERATE and DeltaQ is INC.FAST and prev.Ave.FFR is NORMAL,flow rate of switch is 6
If Qlenght is MODERATE and DeltaQ is DEC.FAST and prev. Ave.FFR is HIGH,flow rate of switch is 3
If Qlenght is MODERATE and DeltaQ is DEC.SLOW and prev. Ave.FFR is HIGH,flow rate of switch is 4
If Qlenght is MODERATE and DeltaQ is ZERO and prev. Ave.FFR is HIGH,flow rate of switch is 5
If Qlenght is MODERATE and DeltaQ is INC.SLOW and prev. Ave.FFR is HIGH,flow rate of switch is 6
If Qlenght is MODERATE and DeltaQ is INC.FAST and prev. Ave.FFR is HIGH,flow rate of switch is 7
If Qlenght is FULL and DeltaQ is DEC.FAST and prev. Ave.FFR is LOW,flow rate of switch is 6
If Qlenght is FULL and DeltaQ is DEC.SLOW and prev. Ave.FFR is LOW,flow rate of switch is 7
If Qlenght is FULL and DeltaQ is ZERO and prev. Ave.FFR is LOW,flow rate of switch is 8
If Qlenght is FULL and DeltaQ is INC.SLOW and prev. Ave.FFR is LOW,flow rate of switch is 9
If Qlenght is FULL and DeltaQ is INC.FAST and prev. Ave.FFR is LOW,flow rate of switch is 10
If Qlenght is FULL and DeltaQ is DEC.FAST and prev. Ave.FFR is NORMAL,flow rate of switch is 7
If Qlenght is FULL and DeltaQ is DEC.SLOW and prev. Ave.FFR is NORMAL,flow rate of switch is 8
If Qlenght is FULL and DeltaQ is ZERO and prev. Ave.FFR is NORMAL,flow rate of switch is 9
If Qlenght is FULL and DeltaQ is INC.SLOW and prev. Ave.FFR is NORMAL,flow rate of switch is 10
If Qlenght is FULL and DeltaQ is INC.FAST and prev. Ave.FFR is NORMAL,flow rate of switch is 10
If Qlenght is FULL and DeltaQ is DEC.FAST and prev. Ave.FFR is HIGH,flow rate of switch is 8
If Qlenght is FULL and DeltaQ is DEC.SLOW and prev. Ave.FFR is HIGH,flow rate of switch is 9
If Qlenght is FULL and prev. Ave.FFR is HIGH,flow rate of switch is 10

Third part of the FIS is Fuzzy inference logic. In here for applying AND logic to inputs the sup-algebraic product compositional rule of inference is employed in this work.
Last part of FIS is defuzzification. Takagi-surgeno fuzzy inference system is used in this Project. Crisp values are calculated using weighted average method.

3.2 Anfis Controller

The input membership functions are initially same with the conventional fuzzy controller membership functions. The graphs of them shown in Figure 3.5 ,Figure 3.6 and Figure 3.7 are same with graph of conventional fuzzy controller in Figure 3.3 ,Figure 3.2 and Figure 3.4 Therefore the result of two models can be more comparable.

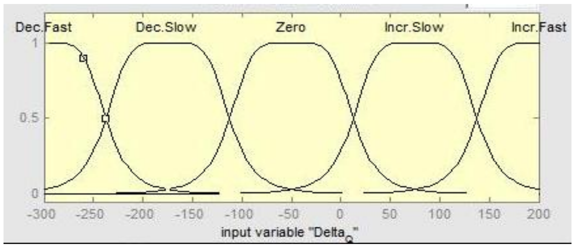


Figure 3.5 Membership function of the Qlenght

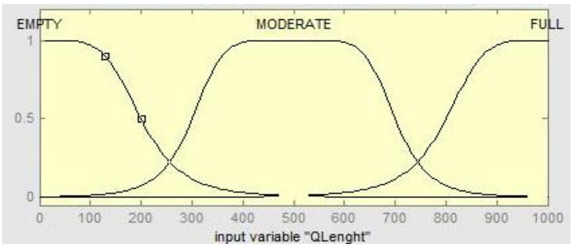


Figure 3.6 Membership function of the DeltaQ

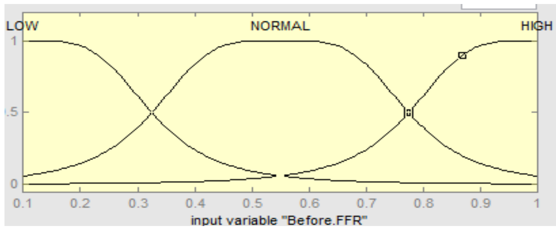


Figure 3.7 Membership function of the DeltaQ

Fuzzy rules and if-then rules are same with conventional fuzzy controller, too. These rules can be seen from Table 3.0.1 and they are table form of conventional fuzzy controller fuzzy rules at mentioned in section 3.1. Difference of the models is starting in here; output variable is not created at this time. It will create after training step with neural network so two output data of the models will be comprised after training.

Table 3.0.1: Rule matrix of the anfis

ΔQ FFR QL	DeltaQ	Desc. Fast	Desc. Slow	Zero	Inc. Slow	Inc. Fast
Qlenght	FFR					
Empty	L	1				
	N					
	H					
Mode rate	L	2	2	3	4	5
	N	2	3	4	5	6
	H	3	4	5	6	7
Full	L	6	7	8	9	10
	N	7	8	9	10	10
	H	8	9	10		

3.2.1 Design anfis architecture

Integrated neurofuzzy system combines the advantages of ANN and FIS. While the learning capability is an advantage from the viewpoint of FIS, formation of knowledge base is an advantage from the viewpoint of ANN. One such neuro-fuzzy algorithm called Adaptive Neuro Fuzzy Inference System (ANFIS) has been used to build neuro-fuzzy controllers. ANFIS is a feed forward type adaptive network, which is functionally equivalent to an FIS. It basically consists of five layers. These layers are shown in Figure 3.6

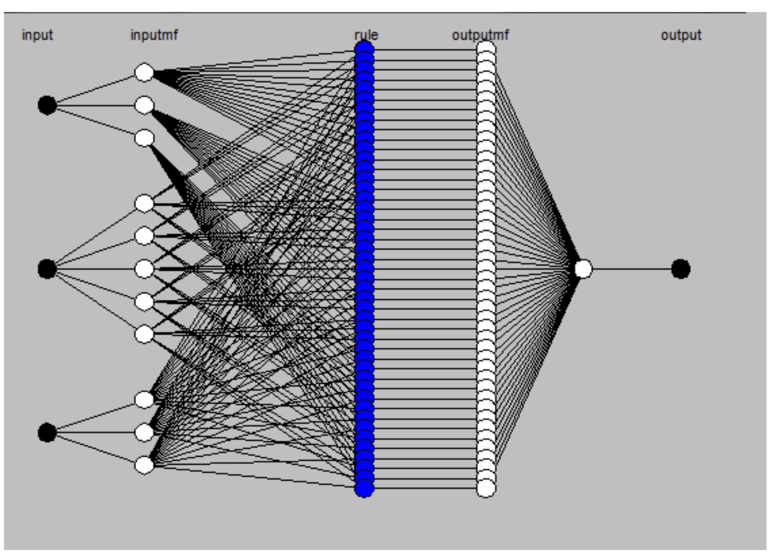


Figure 3.6 Anfis Architecture

Layer I: Every node in this layer computes the degree of membership of the input. Each node is using bell-shaped membership function as given in equation

$$\mu_{A_i}(x) = \frac{1}{1 + \left(\frac{x - c_i}{a_i}\right)^{2b_i}} \tag{15}$$

where, $\{a_i, b_i, c_i\}$ is a parameter set. The parameters in this layer are referred to as premise parameters. The hybrid algorithm adjusts these premise parameters to achieve the optimal shape of the member functions.

Layer II: Every node in this layer multiplies the incoming signals and sends the product out, as shown in equation,

$$O_i^2 = w_i = \mu_{A_j}(x_1) \times \mu_{B_k}(x_2) \times \mu_{C_l}(x_3) \tag{16}$$

Product T-norm operator has been used to perform fuzzy AND operation. Algebraic product:

$$T_{ap}(a, b) = ab \tag{17}$$

Layer III: The ratio of ith rule firing strength to the sum of all rules' strengths is calculated over here, given in equation

$$O_i^3 = \overline{w}_i = \frac{w_i}{\sum_i w_i} \tag{18}$$

Layer IV: Every node i in this layer have a node function as given in equation

$$O_i^4 = \overline{w}_i \underline{f_i} = \overline{w}_i(p_i x_1 + q_i x_2 + r_i x_3 + t_i) \tag{19}$$

Where is the output of layer 3 and $\{p_i, q_i, r_i, t_i\}$ is consequent parameters. x_1 is AveQLen (Average Queue Length) and x_2 is DeltaQ (Change in Queue Length). The parameters $\{p_i, q_i, r_i, t_i\}$ are adjusted through RLSE (Recursive Least Square Estimator).[6]

Layer V: Single fixed node in this layer is labeled Σ. The overall output is computed as the summation of all incoming signals, as given in equation

$$O_i^5 = y = \sum_i \overline{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{20}$$

3.2.2 Training

The designed architecture at the previous section is applied according to the supplied training data with the hybrid learning method of gradient descent [7] and least square estimator. This module basically requires rough initial premise parameters for the membership functions of the layer 1 and supplied training data [8]. Consequent parameters are learnt during forward pass of the hybrid learning algorithm using recursive least square estimator and premise parameters are learnt using gradient descent method. The initial premise parameters and consequent parameters after designed anfis controller are shown in Table 3.3. The member ship functions of anfis controller are changed after trained. These are can be seen from Figure 3.8, Figure 3.9 and Figure 3.10.

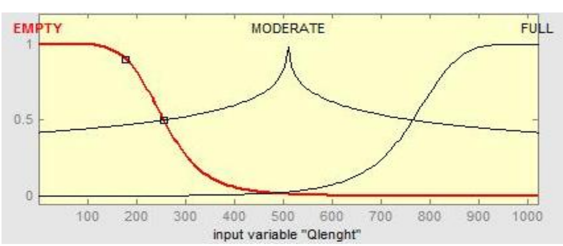


Figure 3.8 Membership function of the Qlenght

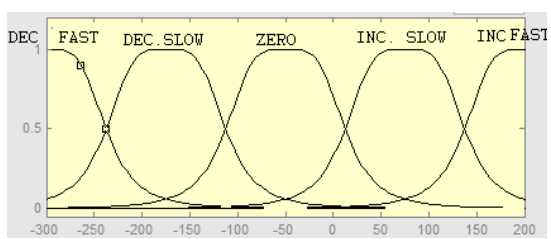


Figure 3.9 Membership function of the DeltaQ

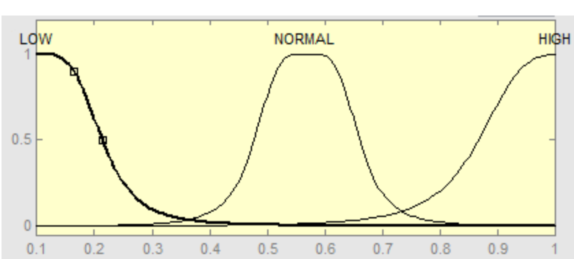


Figure 3.10 Membership function of the Avearage FFR

Table 3.2 Premise Parameters

A_i	B_i	C_i
250	2	0
250	2	500
250	2	1000
62.5	2	-300
62.5	2	-175
62.5	2	-50
62.5	2	75
62.5	2	200
0.225	2	0.1
0.225	2	0.5
0.225	2	1

Table 3.3 Consequent parameter of input

A_i	B_i	C_i
250.2	3.107	0.1025
250.1	0.4396	500
250	02.Haz	1000
62.5	1.998	-300
62.5	1.993	-175
62.5	1.996	50
62.5	2.004	75
62.5	1.999	200
11.179	2.012	0.09592
0.09198	2.023	0.5692
0.166	2.001	1.034

Table 3.4 Consequent Parameters of Output

p_i	q_i	r_i	t_i
1.132e-005	-0.003359	-0.001379	0.003997
1.563e-005	-0.00332	0.0164	0.0003693
7.388e-006	-0.0002485	1.212	0.001167
-2.757e-005	-0.005243	0.003402	0.03422
-2.743e-005	-0.005378	0.0965	0.00312
-1.209e-005	0.03913	8.906	0.009847
0.0004035	0.001456	0.9358	1.494
0.0002499	-0.01184	0.05283	0.134
0.0003321	0.01604	0.9499	0.4402
0.001031	0.008165	-0.5	0.6849
-0.004914	0.01137	6.994	0.0615
0.0001766	-0.01276	1.429	0.2017
0.00131	0.006656	0.0265	0.01997
-0.00106	-0.01154	0.9535	0.001738
0.01054	0.05292	0.1634	0.006129
0.0003448	-0.002795	0.0222	0.0007463
0.0003226	-0.002417	0.2004	-0.001601
0.0003368	-0.002603	0.04895	0.009906
0.0002278	-0.004225	1.079	0.006347
0.0004766	-0.003187	0.3049	-0.01377
0.0003016	-0.003815	0.05435	0.08499
0.000553	-0.004062	-0.433	0.2761
-0.0002307	-0.01337	0.154	-0.6033
0.0002042	-0.01434	-4.83	3.711
0.000595	-0.003209	1	0.1265
0.002014	-0.04307	4.263	-0.277
0.002849	0.01072	-1.44	1.701
0.0003833	-0.001123	0.01756	0.003676
0.006457	0.02809	0.4267	-0.007941
-0.0205	-0.1	0.004479	0.04903
0.0003257	0.0002543	-0.01843	-0.00212
0.0003074	-0.0003894	-0.2883	0.001632
0.0003191	-0.0001352	-0.1094	-0.01138
0.0001838	-0.0009579	-0.9931	-0.01818
0.0003313	-0.002133	-1.024	0.01399
0.000129	-0.003541	-0.5165	-0.09768
0.0004789	-0.003069	0.3032	-0.7947
0.0002541	0.0023	-0.747	0.6111
8.137e-005	0.00129	5.006	-4.269
0.0006851	0.0167	-6.551	0.2803
-0.0005496	0.001023	1.216	-1.957

4. RESULT

After applying all these fuzzy inference system steps to our inputs, the out fractional flow rate has crisp values to calculate the explicit rate in the ATM switch. The result FFR values are shown at the figure 4.1 with respect to input values, Qlenght and Avearage Flow rate.

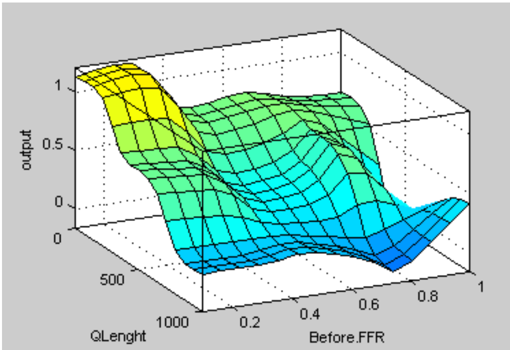


Figure 4.2 Output of the anfis controller

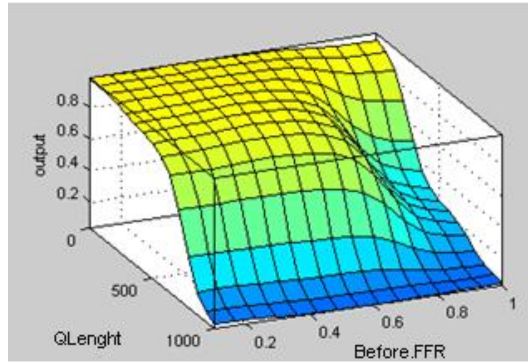


Figure 4.1 Output of the fuzzy logic controller

Result surface after training can be seen from figure 4.2. There is more adaptive shape according to condition of the network.

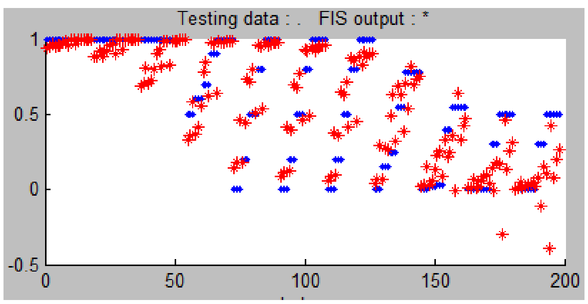


Figure 4.2 Fuzzy controller output resp. to desired data

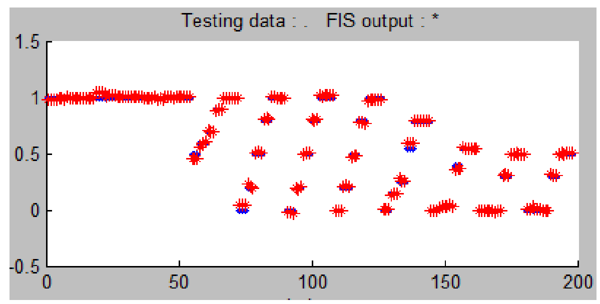


Figure 4.4 Anfis controller output resp. to desired data

Sample data is tested on conventional Fuzzy inference system and results can be seen from figure 4.3. Result is not bad but there is some not match point at the plot.
The data that is used in the conventional controller is tested in trained anfis controller[12]. The plot shows this data and controller output. Match results in here is better if we compare the results of conventional fuzzy controller

Table 4.3 Error values of fuzzy controller

Conventional fuzzy	
Test Error	%0.18673

Table 4.2 Error values of Anfis controller

Anfis	
Train Error	% 0.016018
Test Error	% 0.016185

Results of two models are shown in Table 4.0.1 and Table 4.0.2. Error result of the ANFIS controller is better than conventional fuzzy controller because the membership functions are set only one time and all online network is not remain same conditions. So ANFIS method gives us better performance according to conventional method. However fuzzy logic cannot handle difficulties and differences of the online network traffic.

Table 4.3 Error Rates Of By Increase The Number Of Switches

Switches	**Error Rate%**
1. Switch	0.025144
2. Switch	0.020489
3. Switch	0.01749
4. Switch	0.016259
5. Switch	0.016018
……	
Test result	0.08641

In an ATM network, there may be a lot of switches. We test our algorithm by increasing the number of switches , we can see that error rate decrease to specific number switch. This result may change at any moment of the network according to flow rate on it. However, error rate of this method stays stable for very large ATM network rather than decreasing.

If we look the ANFIS controller result with a graph shown in Figure 4.5, we can see clearly ANFIS controller error values (blue points) are smaller than fuzzy controller error values (red points). In conclusion, we can say that; if we trained the switch possible traffic conditions, switches can schedule data packet more proper and fast with ANFIS models.

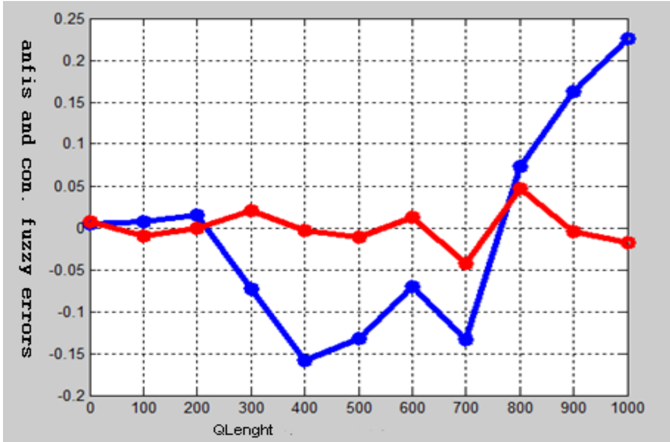


Figure 4.3 Error values of fuzzy controller and ANFIS controller

5. CONCLUSIONS

In daily life , very critical datas is moved over ATM networks, like bank datas. When Customer deposit his money to bank , he wants to see it on his bank account at same moment. if any delay has been occurred when one deposit processing, the other deposits are processed very late. Because of that , atm networks must not has a bottleneck. Switches must be updated continuously and then switches orients the packets on the network for optimization of flow rate.

Conventional fuzzy logic methods cannot provide updating and adapting systems because of setting input mf functions only one time at the beginning. To find the most adapting function to an ATM network is an expert job and very expensive. At the other hand ANFIS methods can update the switches according to network. Also, ANFIS updates input functions according to flow rate of any switches. Therefore our ANFIS controller adapting to online ATM networks very cheap and without an expert.

When results of the both method are compared, we can see that the error rate of the fuzzy logic is about ten times bigger than error rate of ANFIS. This shows us, the flow rate on the network faster with ANFIS controller. A bank that use ANFIS controller on ATM networks waits customers less.

Finally, we can say that, ANFIS models combine the advantages of neural network and fuzzy logic and offers good results.

REFERENCES

[1] Thomas F. Edgar (2006), Recursive Least Squares Parameter Estimation for Linear Steady State and Dynamic Models http://www.cse.sc.edu/~gatzke/, 20 May 2010.

[2] Held G. (1999), Asynchronous Transfer Mode. http://technet.microsoft.com, 10 May 2010.

[3] Pitsillides A, Sekercioglu YA, Ramamurthy G (1997). Effective control of traffic flow in ATM networks using Fuzzy Explicit Rate Marking (FERM). IEEE Journal on Selected Areas in Communications 1997; *15(2):* 209-225.

[4] Askerbeyli İ, Gedik F.A.(2011), Neuro-Fuzzy Approach For Solving Communication Network Problems, International Journal of Electric & Computer Sciences IJECS-IJENS Vol: 11 No: 01, 41-51

[5] Ahmad, Khalid**.** *Sourcebook of ATM and IP Internetworking.* New York : IEEE Press, 2002. pp 78-88

[6] Dubois, D. Prade H. Fuzzy Sets and Systems. Theory and Applications, Academic Press, New York, 1980.

[7] Liu T-L. A Fuzzy PD controller for structural optimization of axis symmetric shells,Master Thesis YUAN ZE University,2003.

[8] ABREU, Gustavo Luiz C. M. de and RIBEIRO, José F.. On-line control of a flexible beam using adaptive fuzzy controller and piezoelectric actuators. Sba Controle & Automação. 2003, vol.14, n.4, pp. 377-383.

[9] Jang, R. Shing, J. ANFIS: Adaptive-network-based fuzzy inference system. IEEE Transactions Systems Man and Cybernetics. v23. 665-685.

[10] Mondragon E.H.,The Design of a Fuzzy Traffic Controller for ATM1 Networks, HCS Research Lab,University of Florida, May 1999

[11] Edgar T. F. (2006), Recursive Least Squares Parameter Estimation for Linear Steady State and Dynamic Models http://www.cse.sc.edu/~gatzke/, 20 May 2010.

[12] Aseri,T.C.,Deepak Bagai D. (2008) , Traffic Control in Unicast ATM ABR Service using Adaptive Approach. International Journal of The Computer, the Internet and Management Vol. 16.No.3,pp 48-63.

[13] Y.A. Sekercioglu, A. Pitsillides, "Fuzzy control of ABR traffic flow in ATM LANs," iscc, pp.227, IEEE Symposium on Computers and Communications (ISCC'95), 1995

[14] Saito H., (1994). Teletrafic Technologies in ATM Networks, Artech House, London.