Fuzzy based intelligent routing in high speed networks

M Mohan rao
M.Tech(Computer Science Engineering)
NRI Institute of technology
Vijayawada, Andhra Pradesh

Vamsi Krishna
Assistance prof in CSE dept
NRI Institute of technology
Vijayawada, Andhra Pradesh

Abstract: As the increase in usage of computing systems like computers, tablets as well as smart devices there is a large desire for the fast-growing online traffic. Dispensed traffic control frame work has been suggested, in which routers are integrated with smart data rate planners to handle the extreme traffic level. The traffic control prototype is distinctive as more traffic control methods have to determine network guidelines which require link latency, bottleneck bandwidth, packet reduction rate, or the amount of flows to calculate the permitted source delivering level. The fuzzy-logic based controller can determine queue size exclusively; it neglects assorted potential efficiency concerns coming up because parameter quotations as we alleviate much usage of calculation and memory resource in routers. As a system parameter, the queue size can be checked precisely and applied for generating proactive choice if activity should be used to control the source transmitting rate, thereby improves the strength of the system to traffic congestion. By the fuzzy logic strategy, QoS (Quality of Service) in communication is guaranteed by good shows of our scheme like max-min equity, low queuing delay and good robustness to system characteristics. The summary is that the outcomes and evaluations have confirmed the performance and made a provided a new benchmark that our traffic control scheme using fuzzy-logic can accomplish better efficiency than the established prototypes that count definitely on the evaluation of system parameter.

Keywords: Congestion control, fuzzy logic control, quality of service, max-min fairness, robustness, traffic management.

I. INTRODUCTION

Traffic congestion control is one of the efficient techniques to handle the network traffic [1], [2]. Network traffic control can restrict a network from extreme congestion as well as degradation in throughput wait efficiency. In contrast to other specific congestion planners that rely on the evaluation of network parameters (like link latency, bottleneck bandwidth, packet loss, or the amount of flows) to calculate the permitted source transmitting rate, the Intel Rate controller can prevent this while retaining good consistency as well as robustness. Simulation outcomes have validated these activities. Our review reveals that the Intel Rate controller can be approximated by a PI (Proportional-Integral) controller but with time-varying increases, which enables the controller to outperform its alternatives. Ultimately, by contrasting with the API-RCP (Adaptive PI Rate Control Protocol) utilising Opnet simulation, we experimentally demonstrate our summary [3].The back-pressure protocol is a well- known throughput-optimal algorithm. Anyhow, its delay efficiency may be rather poor even though the traffic burden is not near to system capability considering the appropriate two causes. First, every node has to preserve an individual queue for every product in the network, and just one queue is provided at a time. Second, the back-pressure routing algorithm can route a few packets on lengthy routes. The customer provides options to handle both of the above concerns, and thus, enhance the delay efficiency of the back-pressure algorithm. Among recommended systems also reduces the difficulty of the queuing information elements to be managed at every node [4]. Newly, explicit Control Protocol (XCP), Rate Control Protocol (RCP), and Adaptive Proportional- Integral Rate Control Protocol (API-RCP) have been projected for congestion control, with the main goal of obtaining fair as well as maximum bandwidth usage. Anyhow, reports display that both RCP and XCP may endure prolonged oscillations {due

ISSN: 2278 – 1323

to|because misestimating the bottleneck link capability, and API-RCP may encounter oscillations as of its PI adaptively strategy which requires switching nonlinearity. To prevent these issues, in the report a way of creating congestion control using the instant queue sizes in the routers is projected. The new strategy attains large link usage and smooth characteristics by clamping the bottleneck queue at a preferred size. And it preserves good equity by assigning the bottleneck bandwidth similarly to the challenging flows. Simulations are done to validate the efficiency of the technical layout [5].Next generation IP-based networks will provide quality of service (QoS) ensures by utilizing technologies like differentiated services (DiffServ) as well as multi-protocol label switching (MPLS) for traffic technology and network-wide supply management. In spite of the progress currently prepared, an amount of concerns still exist relating to edge-to-edge intra-domain as well as inter-domain QoS provisioning as well as control. This article will start by supplying credentials on techniques like DiffServ, MPLS and their potential combination for QoS support. It will subsequently introduce trends in service level agreements (SLAs) as well as service level specifications (SLSs) for the agreement to QoS-based providers It will then relocate to analyze architectures as well as frameworks for the management as well as control of QoS-enabled systems, incorporating the appropriate aspects: techniques as well as algorithms for off-line traffic technology as well as provisioning using specific MPLS paths or using hop-by-hop IP routing; techniques for dynamic resource control to manage with traffic variations outside the expected package; a provider control framework promoting a "resource provisioning cycle"; the derivation of estimated traffic need from subscribed SLSs as well as techniques for SLS invocation entrance control; a tracking design for scalable data selection promoting traffic technology as well as service control; and recognition concerns provided the present state-of-the-art of control protocols as well as tracking service. The article will also consist of protection of appearing work regarding inter-domain QoS provisioning, such as features such as: an inter-domain service model; consumer and peer supplier SLSs; design for the control as well as control of inter-domain providers; inter-domain off-line traffic technology; and QoS extensions to BGP for strong traffic technology. Appropriate developed strategies like IP field will be also protected. In all these segments, current research- work will be provided, with ideas to bibliography as well as a particularly designed Web page with further information [6].A new effective queue control scheme, fuzzy explicit marking (FEM), applied within the differentiated services (Diffserv) platform to supply the congestion control with a fuzzy logic control strategy. Network congestion control stays a critical as well as high priority concern. The fast growth of the Internet as well as increased need to use the Internet for time-sensitive voice as well as video programs require the create and usage of reliable congestion control algorithms, particularly for new architectures, like Diffserv. Thus, a amount of analysts are now looking at conversely schemes to TCP congestion control. RED (random early detection) as well as its variations are one of these choices to supply QoS in TCP/IP Diffserv networks. The projected fuzzy logic strategy for congestion control enables the usage of linguistic skills to obtain the characteristics of nonlinear possibility marking functions as well as offer excellent implementation, use of several inputs to obtain the status of the network more precisely, allow finer tuning for packet establishing actions for aggravated flows, and thus supply better QoS to various types of data streams, like TCP/FTP traffic, whilst preserving large usage [7].

II. RELATED WORKS

From the perception of network as well as service control, the aforesaid congestion control stratagies have QoS (Quality of Service) problems in that they cannot guarantee a certain level of performance under some situations due to design drawbacks. There are many different approaches to improve QoS. For example, admission control, as a network traffic management approach, can guarantee QoS by checking the availability of network bandwidth before establishing a connection, e.g., [8]. Service priority as another approach can be used to improve QoS by providing different service priorities to different users, e.g., [9]. Pricing or routing policies are also found to address QoS problems e.g., [10]. However, they are outside the scope of this paper that focuses on congestion control as an approach to address the QoS management problem.

FLC (Fuzzy Logic Control) [11] has been considered for IC (Intelligence Control). It is a methodology used to design robust systems that can contend with the common adverse synthesizing factors such as system nonlinearity, parameter uncertainty, measurement and modeling imprecision [12]. In addition, fuzzy logic theory provides a

ISSN: 2278 – 1323

convenient controller design approach based on expert knowledge which is close to human decision making, and readily helps engineers to model a complicated non-linear system.

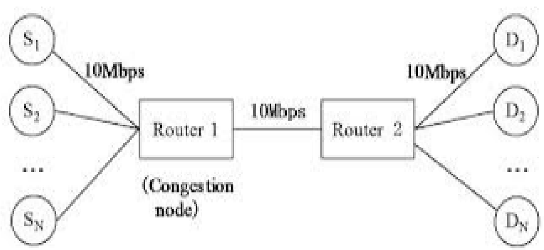

Fig. 1 Basic Configuration of a fuzzy logic system

In fact, fuzzy logic control has been widely applied in industrial process control and showed extraordinary and mature control performance in accuracy, transient response, robustness and stability [13], [14]. These control algorithms are explicit in nature, and they depend on absolute queue length (the maximum buffer size) instead of the TBO to adjust the allowed sending rate. Nevertheless, these early designs have various shortcomings including cell loss (even though cell loss is used as a congestion signal to compute the rate factor, e.g., [15]), queue size fluctuations, poor network latency, stability and low utilization. Later, FLC was used in RED (Random Early Detection) algorithm in TCP/IP networks, e.g., [16], to reduce packet loss rate and improve utilization. However, they are still providing implicit or imprecise congestion signaling, and therefore cannot overcome the throughput fluctuations and conservative behavior of TCP sources. We would like to integrate the merits of the existing protocols to improve the current explicit traffic congestion control protocols (like XCP, RCP, APIRCP and their enhancements) and form a proactive scheme based on some prudent design ideas such that the performance problems and excessive resource consumption in routers due to estimating the network parameters could be overcome. In this respect, a fuzzy logic controller is quite attractive because of its capability and designing convenience as discussed above. Specifically, the objectives of this paper are: 1) to design a new rate-based explicit congestion controller based on FLC to avoid estimating link parameters such as link bandwidth, the number of flows, packet loss and network latency, while remaining stable and robust to network dynamics ; 2) to provide maxmin fairness to achieve an effective bandwidth allocation and utilization; 3) to generate relatively smooth source throughput, maintain a reasonable network delay and achieve stable jitter performance by controlling the queue size; 4) to demonstrate our controller has a better QoS performance through case study. To achieve the above objectives, our new scheme pays attention to the following methodologies as well as the merits of the existing protocols. Firstly, in order to keep the implementation simple, like TCP, the new controller treats the network as a black box in the sense that queue size is the only parameter it relies on to adjust the source sending rate. The adoption of queue size as the unique congestion signal is inspired by the design experience of some previous AQM controllers (e.g., RED and API-RCP) in that queue size can be accurately measured and is able to effectively signal the onset of network congestion. Secondly, the controller retains the merits of the existing rate controllers such as XCP and RCP by providing explicit multi-bit congestion information without having to keep per-flow state information. Thirdly, we rely on the fuzzy logic theory to design our controller to form a traffic management procedure. Finally, we will employ OPNET modeler to verify the effectiveness and superiority of our scheme.

III. PRINCIPLE FOR TRAFFIC MANAGEMENT

We consider a backbone network interconnected by a number of geographically distributed routers, in which hosts are attached to the access routers which cooperate with the core routers to enable end-to- end communications. Congestion occurs when many flows traverse a router and because it's IQSize (Instantaneous Queue Size) to exceed the buffer capacity, thus making it a bottleneck in the Internet. Since any router may become bottleneck along an end- to-end data path, we would like each router to be able to manage its traffic. Below is the general operation

ISSN: 2278 – 1323

principle of our new traffic management/control algorithm. Inside each router, our distributed traffic controller acts as a data rate regulator by measuring and monitoring the IQSize. As per its application, every host (source) requests a sending rate it desires by depositing a value into a dedicated field *Req_rate* inside the packet header. This field can be updated by any router en route. Specifically, each router along the data path will compute an allowed source transmission rate according to the IQSize and then compare it with the rate already recorded in *Req_rate* field. After the packet arrives at the destination, the value of the *Req_rate* field reflects the allowed data rate from the most congested router along the path if the value is not more than the desired rate of the source. The receiver then sends this value back to the source via an ACK (ACKnowledgment) packet, and the source would update its current sending rate accordingly. If no router modifies *Req rate* field, it means that all routers en route allow the source to send its data with the requested desired rate. The following assumptions for the remainder of this paper pertain.1) Every source requests a desired sending rate from the network according to its application. 2) A destination always has enough buffer space to receive data from its source. This is because we do not want the destination to impose any constraint on the source sending rate when we verify the effect of our new control scheme in a bottlenecked router. 3) The propagation delay and the queuing delay along the data path are the two dominant components of the RTT while other components like processing delay of a packet in routers or hosts are negligible in comparison. 4) The queuing discipline of routers is FIFO (First-In-First-Out). 5) Long-lived flows with infinitely long files are used to approximate the greedy behavior of a source when active. This would generate the severest traffic in order for us to verify the robustness of the new scheme.

IV. THE INTELRATE CONTROLLER DESIGN

Figure 2 depicts the components of our fuzzy logic traffic controller for controlling traffic in the network system defined in Fig. 1. Called the IntelRate, it is a TISO (Two-Input Single- Output) controller. The TBO (Target Buffer Occupancy) q0 > 0 is the queue size level we aim to achieve upon congestion. The queue deviation e(t) = q0 -q(t) is one of the two inputs of the controller. In order to remove the steady state error, we choose the integration of *e(t)* as the other input of controller, i.e. *g (e* (t)) = *e* (t) *dt.* The aggregate output is *y* (t) =*ui (t - τi).* Under heavy traffic situations, the IntelRate controller would compute an allowed sending rate *ui(t)* for flow *i* according to the current IQSize so that q(t) can be stabilized around q0. In our design, IQSize q(t) is the only parameter each router needs to measure in order to complete the closed-loop control.

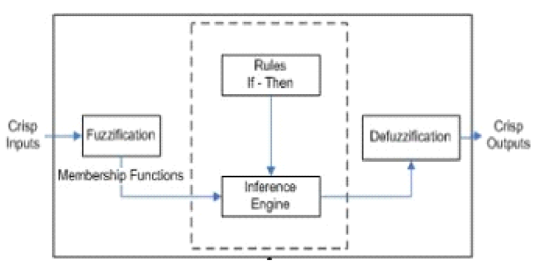

Fig.2 Intelrate Controller System

FLC is a non-linear mapping of inputs into outputs, which consists of four steps, i.e., rule base building, fuzzification, inference and defuzzification. The concepts of fuzzy set and logic of FLC were introduced in 1965 by Zadeh, and it was basically extended from two-valued logic to the continuous interval by adding the intermediate values between absolute TRUE and FALSE. Interested readers are referred to some standard tutorials/texts like [11], [17] for the details of the fuzzy logic theory. In the sequel, we formulate our new controller by following those four steps along with designing the fuzzy linguistic descriptions and the membership functions. The parameter design issues and the traffic control procedure are also discussed at the end of the section.

A. Linguistic Description and Rule Base

We define the crisp inputs *e(t). g(e(t))* and output *u(t)* with the linguistic variables *&(t).. §(e>(.tV)* respectively. There are N(N =1, *2, 3, . . .)* LVs (Linguistic Values) assigned to each of these linguistic variables. Specifically, we let = *;j — 1,2,... ,N}* be the input LVs with *i* = 1 for *S(t)* and *i* = *2* for *g(e (/)).* and let ***U = = 1,2*** ff] for itfe).

ISSN: 2278 – 1323

B. Membership Function, Fuzzification and Reference

Our IntelRate controller employs the isosceles triangular and trapezoid-like functions as its MFs (Membership Functions).Figure 3 describes the MFs used to determine the certainty of a crisp input or output.

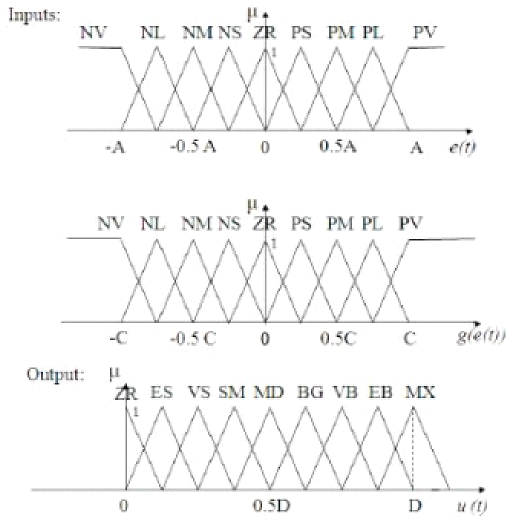

Fig.3 Membership functions without FS.

Note that Fig. 3 describes the usual way of designing MFs with FLC to determine the certainty of a crisp input or output, where e(t) and g(e(t)) have the same number of MFs and there are no boundaries for the input *g(e(t)).* While Fig. 4 is used to illustrate the design of a general FS, the designated values actually come from an example of $N = 9$ LVs with the absolute values of both the upper and lower limits of g(e(t)) set to mq0. Since e(t) is bounded by the physical size of a queue, we have the boundaries according to the limits $q0-B < e(t) <$ q0. The vertical dashed lines in Fig. 4 denote those boundaries of inputs or output.

C. *Defuzzification*

For the defuzzification algorithm, the IntelRate controller applies the COG (Center of Gravity) method to obtain the crisp output.

D. Design Parameters

From our design above, one can see there are different parameters which ultimately will affect the performance of our traffic controller.

E. The Control Procedure

The traffic-handling output u(t) and compare it with *Req rate.* (4) If an operation cycle *d* is over, update the crisp output *u(t)* and the output edge value of *D.* Note that this procedure actually allows the router to perform the max-min fairness in that the greedy flows are always restricted to *u(t)* by a router under heavy traffic conditions while those small flows procedure of the IntelRate controller in a router is , (1) Upon the arrival of a packet, the router extracts *Req_rate* from the congestion header of the packet. (2) Sample IQSize q(t) and update e(t) and g(e(t)). (3)

ISSN: 2278 – 1323

Compute the sending rate are smaller than u(t) along their data path have no such a restriction.

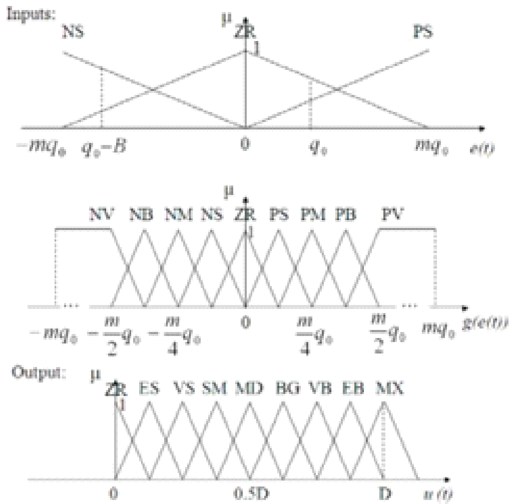

Fig.4 Membership functions with FS.

V. PERFORMANCE EVALUATION

The capability of the Intel Rate controller is demonstrated by performance evaluations through a series of experiments.

- *A. Simulated Network*

The controller is evaluated by the following performance measures.

1) Source throughput (or source sending rate) is defined to be the average number of bits successfully sent out by a source per second, i.e. bits/second. Here, a bit is considered to be successfully sent out if it is part of a packet that has been successfully sent. 2) IQ Size is the length of the bottleneck buffer queue seen by a departing packet. 3) Queuing delay is the waiting time of a packet in the router queue before its service.4) Queuing jitter is the variation of queuing delay due to the queue length dynamics, and is defined as the variance of the queuing delay.5) Link (or bottleneck) utilization is the ratio between the current actual throughput in the bottleneck and the maximum data rate of the bottleneck. It is expressed as a fraction less than one or as a percentage.6) Packet loss rate is the ratio between the number of packet dropped and the number of total packets received per second by the bottleneck.7) A feasible allocation of rates is 'maxminfair' if and only if an increase of any rate within the domain of feasible allocations must be at the cost of a decrease of some already smaller or equal rates

- *B. Robustness to Large Network Charges*

The real world Internet traffic is always dynamic. The performance of our controllers when faced with drastic network changes as in Fig.5 the variations of the number of flows or the available bandwidth is been investigated.

ISSN: 2278 – 1323

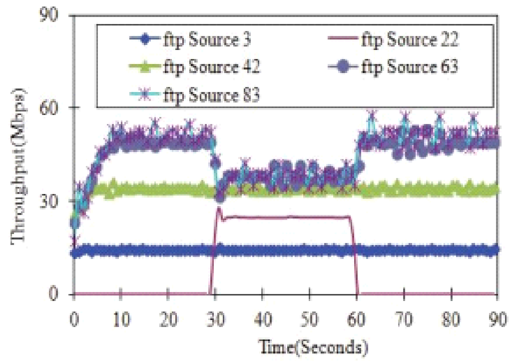

Fig. 5 Source and IQ Size dynamics under traffic change.

C. Queuing Jitter Control

One main source of the network latency oscillations comes from the dynamics of queuing delay in the routers. In Fig.6, we want to check under heavy traffic condition show the queuing delay fluctuates under the different TBO sin the Intel Rate controller, and how big the queuing jitters can be.

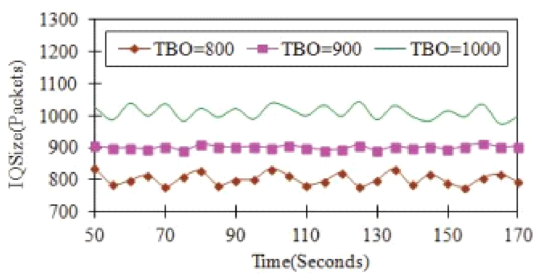

Fig.6 IQSize and queuing delay under different TBOs.

E. Effect of Short-Lived Traffic

The experiment shows that the long-lived flow scans accommodate the large short-lived http flows upon their arrivals. They simply regard the http flows as the long-lived ftp flows.

F. Utilization and Packet Loss Rate

The utilization and packet loss rate performance of the Intel Rate controller with respect to bottleneck bandwidth or the different settings of TBOs. First we check the system utilization and packet loss rate under the different bottleneck bandwidth from 45Mbpsto 10Gbps. The simulation results show that the Intel Rate controller is able to maintain the ideal zero packet loss rate with 100% link utilization despite the different bottlenecks(here we omit the plots due to space limit). Therefore, the buffer never overflows and packets are never lost upon heavy traffic. In the meanwhile, the stable feature in IQ Size and throughput guarantees the full bandwidth utilization.

VI CONCLUSIONS

A creative traffic control scheme, named the Intel Rate controller, has been projected to handle the Internet congestion in order to assure the quality of service for various service programs. The controller is developed by giving consideration to the drawbacks and the benefits of the existing congestion control prototypes. As a dispensed operation in networks, the Intel Rate controller utilizes the instant queue size only to efficiently throttle the source transmitting rate with max-min fairness. The back-pressure algorithm, while being throughput-optimal, is not

ISSN: 2278 – 1323

beneficial in apply for adaptive routing since the delay efficiency can be actually harmful. In the paper, we have provided an algorithm that routes packets on quickest hops when feasible and decouples routing and organizing using a probabilistic splitting algorithm built on the approach of shadow. Probabilistic routing table that depends gradually over time, real packets do not have to examine long routes to enhance throughput; this efficiency is carried out by the shadow "packet..

A packet-by-packet wavelength-routing interlink strategy for converting method with three- stage design has been shown. The strategy employs an optical wavelength section multiplexing link as well as dynamic bandwidth-sharing among wavelengths.

REFERENCES

[1] M. Welzl, *Network Congestion Control: Managing Internet Traffic.* JohnWiley & Sons Ltd., 2005.
[2] R. Jain, "Congestion control and traffic management in ATM networks: recent advances and a survey," *Computer Networks ISDN Syst.,* vol. 28, no. 13, pp. 1723-1738, Oct. 1996.
[3] J. Liu and O. Yang, "Characterization of the IntelRate controller for high-speed networks," in *Proc. 2011 Commun. Netw. Services Research Conf.,* pp. 63-68.
[4] W. Hu and G. Xiao, "Design of congestion control based on instantaneous queue size in the routers," in *Proc. 2009 IEEE GLOBECOM,* pp. 1-6.
[5] G. Pavlou, "Traffic engineering and quality of service management for IP-based NGNs," in *Proc. 2006 IEEE/IFIP Netw. Operations Manage. Symp.,* p. 589..
[6] C. Chrysostomou, A. Pitsillides, G. Hadjipollas, *et al.,* "Fuzzy explicit marking for congestion control in differentiated services networks," in *Proc. 2003 IEEE Int. Symp. Computers Commun.,* vol. 1, pp. 312-319.
[7] A. Vashist, M. Siun-Chuon, A. Poylisher, *et al.,* "Leveraging social network for predicting demand and estimating available resources for communication network management," in *Proc. 2011 IEEE/IFIP Int.Symp. Integrated Netw. Manage.,* pp. 547-554.
[8] D. Toelle and R. Knorr, "Congestion control for carrier ethernet using network potential," *Proc. 2006 IEEE/IFIP Netw. Operations Manage. Symp.,* pp.1-4.
[9] M. Charalambides, P. Flegkas, G. Pavlou, *et al.* , "Policy conflict analysis for diffserv quality of service management," *IEEE Trans. Netw. Service Manage.,* vol. 6, no. 1, pp. 15-30, Mar. 2009.
[10] G. Kesidis, "Congestion control alternatives for residential broadband access," *Proc. 2010 IEEE Netw. Operations Manage. Symp.,* pp. 874- 877.
[11] K. M. Passino and S. Yurkovich, *Fuzzy Control.* Addison Wesley Longman Inc., 1998.
[12] E. Jammeh, M. Fleury, C. Wagner, *et al.,* "Interval type-2 fuzzy logic congestion control for video streaming across IP networks," *IEEE Trans. Fuzzy Syst.,* vol. 17, no. 5, pp. 1123-1142, 2009.
[13] T. W. Vaneck, "Fuzzy guidance controller for an autonomous boat," *IEEE Control Syst. Mag.,* vol. 17, no. 2, pp. 43-51, Apr. 1997.
[14] T. Kiryu, I. Sasaki, K. Shibai, *et al.,* "Providing appropriate exercise levels for the elderly," *IEEE Eng. Med. Biol. Mag.,* vol. 20, no. 6, pp. 116-124, 2001.
[15] R. Chang and C. Cheng, "Design of fuzzy traffic controller for ATM networks," *IEEE/ACM Trans. Netw.,* vol. 4, no. 3, pp. 460-469, June 1996.
[16] H. Aoul, A. Nafaa, D. Negru, and A. Mehaoua, "FAFC: fast adaptive fuzzy AQM controller for TCP/IP networks," in *Proc. 2004 IEEE GLOBECOM,* vol. 3, pp. 1319-1323.
[17] S.Kaehler.Available:
http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html

ISSN: 2278 – 1323