Eng. & Tech. Journal, Vol.31, No.3, 2013

Design of On-Line Tuned Controller for Congestion Avoidance in Computer Networks

Saba T. Salim
Control and Systems Engineering Department, University of Technology/ Baghdad
Email:Saba_irq@yahoo.co.uk
Ali Majeed Mahmood
Control and Systems Engineering Department, University of Technology/ Baghdad

Received on: 16/10/ 2011 & Accepted on: 4/10/ 2012

ABSTRACT

Active Queue Management (AQM) applies a suitable control policy upon detecting congestion in networks. In this paper, a Proportional-Integral (PI) controller based on the optimization algorithm of MATLAB/Nonlinear Control Design Blockset (NCD) which is adapted as On-line tuning for controller parameters is applied to AQM for the objective of congestion avoidance and control in middle nodes. To present the methodology, a Proportional Integral (PI) controller is verified with the TCP/AQM model as an Active Queue Management (AQM) in internet routers The analytical results for linearized TCP/AQM model are presented in MATLAB/Simulink. From the obtained results, a faster response time as well as the regulation of the output to a constant value by the PI controller is clearly observed and it is noted that the PI controller based NCD provides good tracking performance for congestion avoidance in computer networks.

Keywords: Active Queue Management, NCD, computer networks.

تصميم مسيطر منغم آنيا لتجنب الاختناق في شبكات الحاسوب

الخلاصة

ان ادارة الطابور الفعال AQM يتم بتطبيق قوانين سيطرة مناسبه لأكتشاف الاختناق في شبكات الحاسوب. في هذا البحث تم تطبيق متحكم من نوع PI المعتمد على خوارزمية مثلى لأيجاد قيم المعاملات المناسبة الموجوده في برنامج Matlab وهي NCD او Nonlinear Controller Design و التي تتكيف آنيا للحصول على افضل قيم لمعاملات المتحكم كمنظم للطابور الفعال لغرض تجنب الاختناق و السيطرة في العقد الوسطى.

ولغرض التحقق من عمل المتحكم PI مع هذه الخوارزمية تم تطبيقه على الموديل الرياضي لل TCP / AQM في موجهات شبكات الانترنيت. النتائج التحليلية لهذا الموديل طبقت في برنامج Matlab. ولوحظ من النتائج التي تم الحصول عليها استجابه اسرع للنظام بالاظافه الى الحفاظ على الاخراج كقيمة ثابتة

https://doi.org/10.30684/etj.31.3A.14
2412-0758/University of Technology-Iraq, Baghdad, Iraq

بواسطة هذا المسيطر اللاخطي. و كذلك لوحظ بانه تم استخدام خوارزمية الحصول على قيم المعاملات المثلى NCD مع المسيطر PI تعطي اداء جيد في التتبع لقيمة الاخراج و بالتالي تجنب الاختناق في شبكات الحاسوب.

INTRODUCTION

Active queue management refers to the practice of manipulating the queue at an outbound interface in a router to bias the performance of flows that transit the router. The goals of active queue management are (1) to reduce the average length of queues in routers and thereby decrease the end-to-end delay experienced by packets, and (2) to ensure that network resources are used more efficiently by reducing the packet loss that occurswhen queues overflow [1]. Congestion in a computer network is a state in which performance degrades due to the saturation of network resources such as communication links, processor cycles, and memory buffers. Network congestion has well recognized as a resource-sharing problem. When too many packets are contending for the same link, the queue overflows and packets have to be dropped. When such drops become common events, the network is said to be congested. A brief description of these researches is submitted in the following paragraphs.

Jacobson and Karels [2] proposed the end-to-end congestion control is algorithms which forms the basic

for the TCP congestion control. Its content that a TCP sender keeps a sending window (packets) rate according to the rate of dropped packets when a buffer becomes full in the router queue.

Floyd and Jacobson [3] presented the RED. Its mechanism is that packets are randomly dropped before the buffer of queue overflows. And, Braden et al. [4] proposed the enhanced end-to-end congestion control for AQM.

Misra et al.[5] developed a methodology to model and obtain numerically expected transient behavior of networks with AQM routers supporting TCP flows. The solution methodology scales well to a large number of flows. This modeling/solution methodology has a great potential in analyzing and understanding various network congestion control algorithms.

Hollot et al. [6] used linearization to analyze a previously developed Non-linear model of the TCP/AQM. They linearized the model by using small-signal linearization about an operating point to gain insight for the purpose of feedback control to analyze a combined TCP and AQM model from a control theoretic standpoint and ability to present design guidelines for choosing parameters that lead to stable operation of the linear feedback control system.

Hollot et al. [7] proposed PI controller based on the linear control theory. The main contribution is to convert the congestion control Algorithm into the controller design problem within the framework of control theory in AQM system by studying a previously developed linearized model of TCP and AQM. The controller showed better theoretical properties than the well known RED controller.

Yuanwei Jing et al. [8] developed a new AQM algorithm based on Fuzzy Sliding Model Controller for the objective of congestion avoidance and control in middle nodes. The proposed controller combines the excellent characteristics of linear sliding model controller (LSMC) and the terminal sliding model controller (TSMC). The LSMC is used to speed up the error convergence when the error is greater than one, and the TSMC is adopted to guarantee the error convergence to zero in a finite time when the error is around the zero.

Yann et al. [9] Proposed an AQM based on the Lyapunov theory for time delay systems. With the help of Lyapunov-Krasovskii functional and using a state space representation of a linearized fluid model of TCP. Note that the proposed methods have been extended to the robust case where the delay in the loop is unknown.

Several researches have been done in the field of congestion avoidance in traffic of computer networks.

Kang et al. [10] proposed the LQ-Servo controller for AQM routers. The proposed controller structure is made by taking a traditional servo mechanism based on Linear Quadratic (LQ) approach. The proposed LQ-Servo controller can deal with a good tracking performance comparing with PI controller.

Kang et al. [11] developed the LQ-Servo controller based on loop shaping method for TCP/AQM router in order to meet such frequency domain design specifications as good disturbance rejection. The simulation results show that the proposed controller is more effective in getting the good tracking responses than PI controller for the varying reference queue size in AQM routers.

TCP/AQM MODEL

AQM has been extensively analyzed using control-theoretical methods. Control-theoretical approaches lead to stable, effective, and robust congestion control operation. In [5], the non-linear dynamic model for multiple TCP flows control has been developed based on fluid-flow theory to model the interactions of a set of TCP flows and AQM routers in computer networks which consist of a system of nonlinear differential equations. For the control theoretical analysis, it was approximated as a linearized constant model by small signal linearization about an operating point(W_0, q_0, p_0), see [6,12] for linearization details, which leads to the following :

$$\delta \dot{W}(t) = -\frac{2N}{R_0^2 C}\delta W(t) - \frac{R_0 C^2}{2N^2}\delta p(t-R_0) \qquad \ldots (1-a)$$

$$\delta \dot{q}(t) = \frac{N}{R_0}\delta W(t) - \frac{1}{R_0}\delta q(t) \qquad \ldots \qquad (1-b)$$

where $\delta W(t) \approx W - W_0$, $\delta q(t) \approx q - q_0$, $\delta p(t) \approx p - p_0$, $\dot{W}(t)$ denotes the time-derivative of $W(t)$, $\dot{q}(t)$ denotes the time derivative of $q(t)$, and

W :Expected TCP window size (packets)

q : Expected queue length (packets)

R_0:Round-trip time (seconds)

C :Link capacity (packets/second)
N : Load factor (number of TCP sessions)
p :Probability of packet mark/drop
t : Time
The expected queue length q and the expected TCP window size W are positive value and bounded quantities. And also, the probability of packet (mark /drop) p takes value only in [0 ,1] . Taking the Laplace transform of equation (1) and rearranging the following transfer functions are obtained:

$$P_{tcp}(s) = \frac{W(s)}{p(s)} = \frac{\frac{R_0 C^2}{2N^2}}{s + \frac{2N}{R_0^2 C}} \quad ..\ (2)$$

$$P_{queue}(s) = \frac{q(s)}{W(s)} = \frac{\frac{N}{R_0}}{s + \frac{1}{R_0}} \quad ..(3)$$

So, the overall plant transfer function becomes:

$$P(s) = P_{tcp}(s) P_{queue}(s) e^{-sR_0} \quad ..\ (4)$$

And can be expressed as:

$$P(s) = \frac{\delta q(s)}{\delta p(s)} = \frac{\frac{C^2}{2N} e^{-sR_0}}{(s + \frac{2N}{R_0^2 C})(s + \frac{1}{R_0})} \quad ..\ (5)$$

Thus, the block diagram of linearized AQM control system is shown in Fig.1. In this diagram *Ptcp* (s) denotes the transfer function from loss probability $\delta p(t)$ to window size $\delta W(t)$, *Pqueue* (s) denotes the transfer function from $\delta W(t)$ to queue length $\delta q(t)$, and $C(s)$ denotes the transfer function of controller.

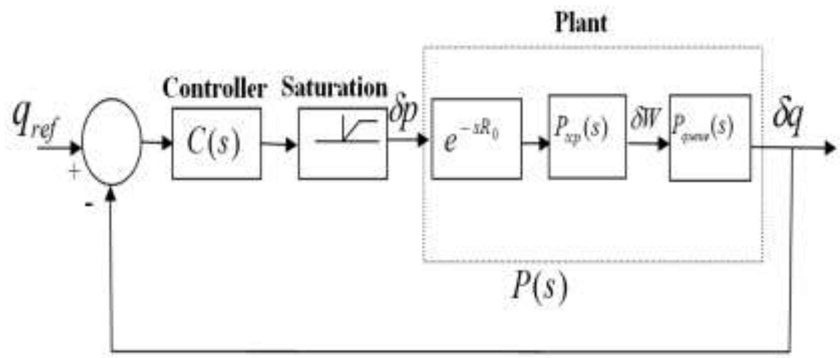

Figure (1): Block diagram of a linearized AQM as feedback control [12].

NONLINEAR CONTROL DESIGN (NCD) BLOCKSET

The (NCD) is a MATLAB tool that helps to tune design parameters in a nonlinear Simulink model by optimizing time-based signals to meet user-defined constraints by graphically placing constraints within a time-domain window. The NCD Blockset automatically converts time domain constraints into a constrained optimization problem and then solves the problem using the optimization routines taken from the Optimization Toolbox. The constrained optimization problem formulated by the NCD Blockset iteratively calls for simulations of the Simulink system, compares the results of the simulations with the constraint objectives, and uses gradient methods to adjust tunable parameters to better meet the objectives. The NCD Blockset allows introducing uncertainty into plant dynamics, specify lower and upper limits on tunable parameters, and alter termination criterion. The progress of an optimization while the optimization is running can be followed from command window, and the final results are available in the MATLAB workspace when an optimization is complete. Intermediate results are plotted after each simulation. It allows the user to terminate the optimization before it has completed, to retrieve the intermediate result or change the design.

NCD uses optimization algorithms to find parameter values that allow a feasible solution to the given constraints. NCD automatically converts the constraint bound data and tunable variable information into a constrained optimization problem. Basically, the NCD Blockset attempts to minimize error and generates constraint errors at equally spaced time points (with spacing given by the Discretization interval defined in the Optimization Parameters dialog box) beginning at the simulation start time and bound constraints, it is defined the constraint error as the difference between the simulated output and the constraint boundary. For lower bound constraints, it is defined the constraint error as the difference between the constraint boundary and the simulated output [13].

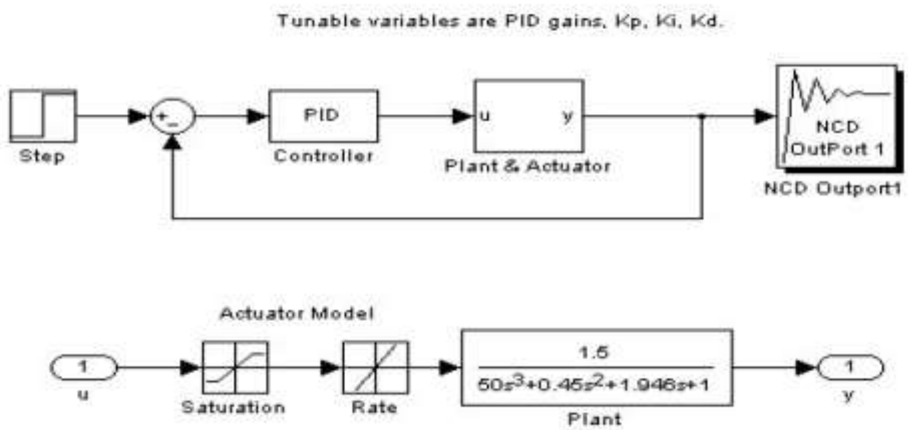

Figure (2): A Simulink model with NCD Outport Block[14].

CONTROLLER DESIGN

Conventional PI controller

The continuous-time PI controller is described by the following expression [14]:

$$u(t) = K_p \times e(t) + K_i \times \int e(t)dt \qquad ..(6)$$

Where K_p and K_i are the proportional and the integral gain coefficients. A block diagram for a PI controller is shown in Figure (2).

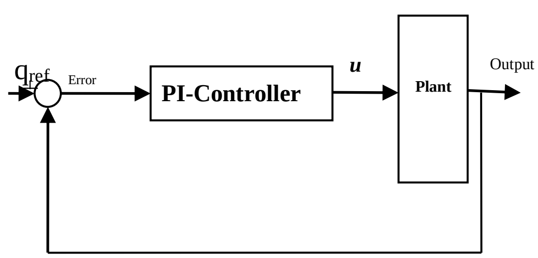

Figure (3): Block diagram of PI controller.

PI controller with NCD Blockset

To use the NCD Blockset, it only requires to include a special block, the NCD out port block, in Simulink diagram and to connect that block to any signal in the model to signify that user wants to place some kind of constraint on the signal. NCD outport block can be found under NCD within the Simulink Library Browser. Fig. (4,5) shows the block diagram of Engine Model with PI controller and NCD blockset.

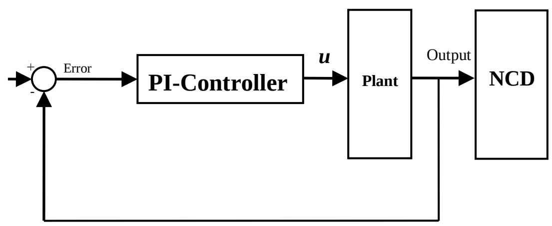

Figure (4): block diagram of Engine Model with PI Controller and NCD blockset.

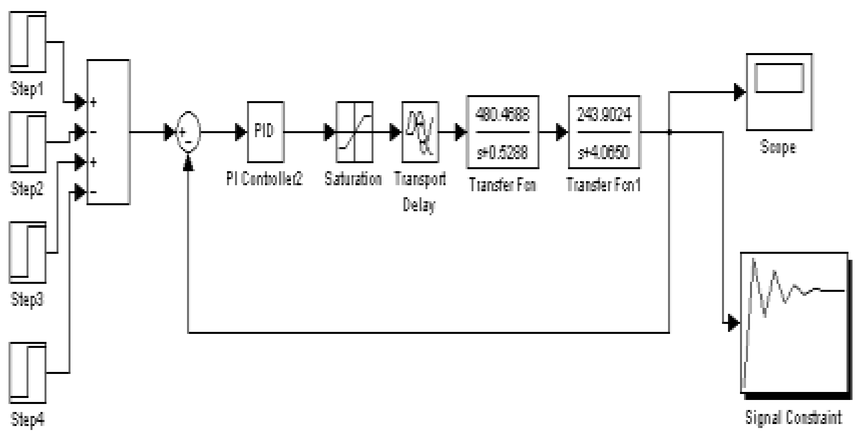

Figure (5): block diagram of TCP/AQM Model with PI controller and NCD blockset in MATLAB/SIMULINK.

NETWORK TOPOLOGY CASE STUDY

Figure (6) shows the network case study taken, where the simulation is conducted for a single link (Bottleneck link) that has a bandwidth capacity **C=3750** packets/sec (corresponds to a **15** Mbps with packet size **500** bytes), and the same bandwidth capacity is used at other links, the Round Trip Time (**R_0**) is **0.246** second . The number of TCP sessions (**N**) is **60** for source and destination, where applying the above parameters in Eq. (5) gets the overall TCP/AQM system transfer function as shown in equation (7). The maximum queue length in the Router1 is **800** packets and the desired queue size is **200** packets. The AQM mechanism (PI) is configured at Router1.

$$P(s)=\frac{q(s)}{p(s)}=\frac{117187.5e^{-s\,0.246}}{s^{2}+4.594s+2.15} \qquad ..\ (7)$$

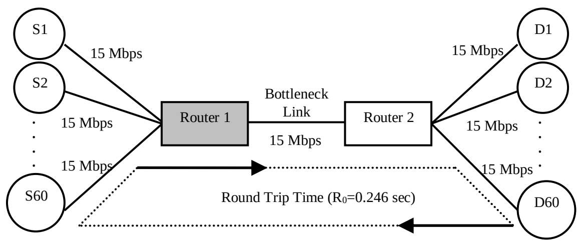

Figure (6): Network Topology Case Study [12].

LINEARIZED TCP/AQM MODEL SIMULATION RESULTS

The simulation of the linearized TCP/AQM model was done in MATLAB 7.0. Consider the TCP/AQM model with network parameters as set in pervious section and the reference input (queue size) which has rectangular form changes every 50 seconds as shown in Eq(8). First the simulation is done for the system without controller as shown Figure (7).

$$q_{ref}=\begin{cases}300 & 0<t<50;\\ 200 & 50<t<100;\\ 400 & 100<t<150;\\ 200 & 150<t<200;\end{cases} \qquad ..\ (8)$$

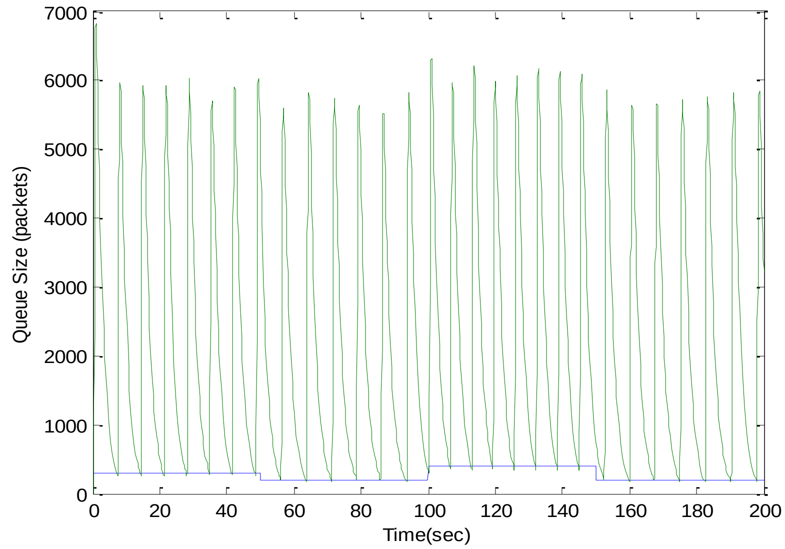

Figure (7): System Response without Controller.

From Figure (7) it is shown that the system without controller is unable to track the queue length around the queue length to the desired level, where the system goes into a sustained oscillation with high congestion exceeding the maximum buffer size. In order to eliminate this sustained oscillation and get better tracking performance a classical control (PI-controller) is applied. As it is shown in Figure (8) below. The system response with PI controller has the ability to get the desired performance, where the desired queue length is achieved within the buffer size of the router. The PI controller parameters are selected by trial and error method. Thus the PI coefficients are K_p=٠,٠٠٠٠٢٥ and K_i= ٠,٠٠٠٠٠٨respectively. Although the PI controller shows good performance the system response is slow, so to overcome this drawback and to get the best parameters to controller and for enhancing the system response the NCD is used **as** on-line tuning method to get best Parameters for PI controller to speed up the system response. Figure (9) shows the system response with PI and PI based NCD. Using NCD enhances the system response especially in decreasing the rising time and the settling time which means that the PI with NCD could speed up the system response, as a result it gives better congestion avoidance compared with trial and error method and, as shown in Fig. (9) and Table (1).

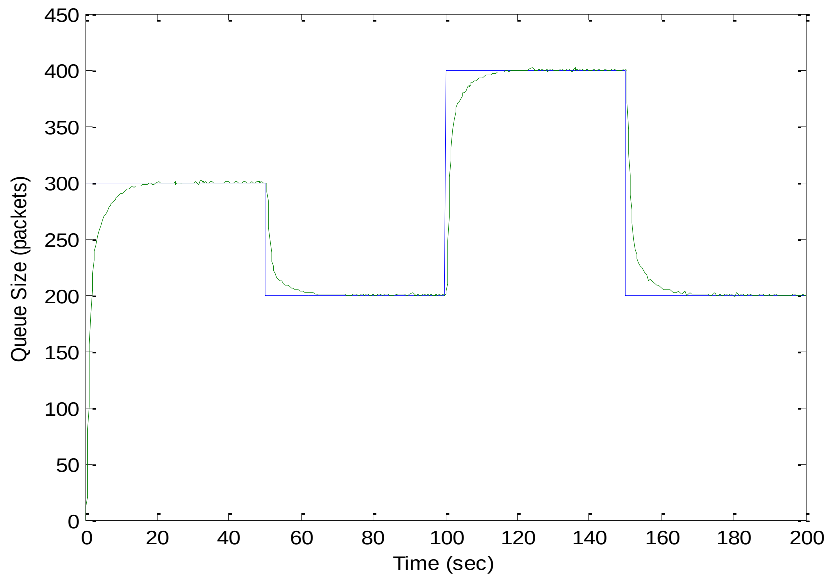

Figure (8): System Response with PI Controller.

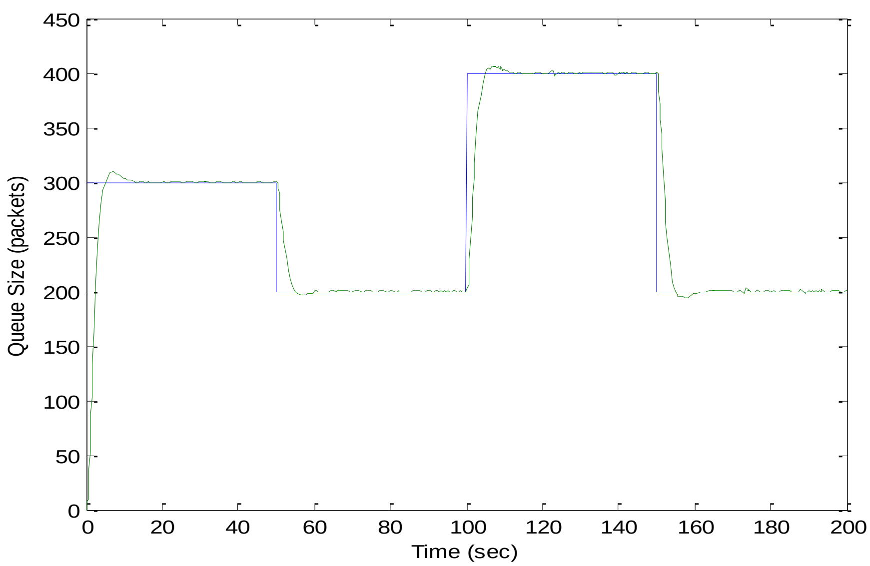

Figure (9): System Response with PI Controller based on NCD.

From Figure (9) above it is shown that the PI controller based NCD speeds up the response comparing with PI controller, where PI controller parameters are get from NCD as follows K_p=١,٢٨٩٣ e-005, K_i=٩,٥٠٣٣ e-006.

Table (1): TCP/QAM System Response Performance of PI, and PI Based NCD with error criteria 5%.

Controller	**Rise Time (t_r) (sec)**	**Overshoot M_p%**	**Peak Time (t_p) (sec)**	**Settling Time (t_s) (sec)**
PI	20	-	-	22
PI based NCD	5	1.5	7	13
Improvement%	75%	-	-	50%

CONCLUSIONS

From the design and the simulation results, it can be concluded that:

1- The applied PI controller based NCD method can deal with congestion problem with a good tracking performance as shown from the parameters obtained in Table (1) about the desired queue size with high link utilization and faster system response observed.

2- By using the NCD to tune PI parameters the system response is improved as shown in table () which prove the efficiency of NCD method.

3- NCD is adapted for real-time executions and it can tune any controller without the mathematical model knowledge of the system it is controlling.

REFERENCES

[1]. Menighed, K. C. Join and C. Aubrun, “Multi-Red Controller for Router Accommodation” BP-239, 54506 Vandoevre-lès-Nancy, France, 2004.

[2]. Jacobson and M. J. Karels, V. "Congestion Avoidance and Control", *In SIGCOMM’ss*, 1988.

[3]. Floyd and V. Jacobson, S. "Random early detection gateway for congestion avoidance", IEEE/ACM Transactions on Networking Vol.1, No. 4, August 1993.

[4]. Braden,B. D. Clark and J. Crowcroft, "Recommendations on Queue Management and Congestion Avoidance in the Internet", Network Working Group ,Request for Comments: 2309 *(RFC2309),* April 1998.

[5]. Misra, V. W.-B. Gong, and D. Towsley, "Fluid-Based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED", in Proc. ACM SIGCOMM, 2000. pp. 151-160.

[6]. Hollot,C. V. V. Misra, D. Towsley, and W. B. Gong, "A Control Theoretic Analysis of RED", In Proceedings of IEEE INFOCOM, 2001.

[7]. Hollot, C. V. V. Misra, D. Towsley, and W. B. Gong, "On Designing Improved Controllers for AQM Routers Supporting TCP Flows". In Proceedings *of IEEE* INFOCOM, pages 1726–1734, 2001.

[8]. Jing, Y. N. Yu, Z. Kong*, G. Dimirovski "Active Queue Management Algorithm Based on Fuzzy Sliding Model Controller",College of Information Science and Engineering, Northeastern University, Shenyang, Liaoning, 110004, P.R. of China**,** 2008 IFAC Korea, 2008, 332-338.

[11]. Lee, K. M. J. H. Yang and B. S. Suh,"Lq-Servo Control Design Based On Loop-Shaping Method For TCP/AQM Router" International Conference Modeling , Identification, and Control MIC February 16 -18 ,2009 Innsbruck ,Austria

[12]. Mahmood, A. M. "Design of Fuzzy-Genetic Controller for Congestion Avoidance in Computer Networks", March, 2010.

[13]. MathWorks Inc., (2002), MATLAB® Nonlinear Control Design Blockset User's Guide, Version 1.1.6, July 2002.

[14]. Katsuhiko Ogata, "Modern Control Engineering", third edition, 1997.