Congestion Control of Active Queue Management Routers Based on LQ-Servo Control

Kang Min Lee, Ji Hoon Yang, and Byung Suhl Suh

Abstract— **This paper proposes the LQ-Servo controller for AQM (Active Queue Management) routers. The proposed controller structure is made by taking a traditional servo mechanism based on Linear Quadratic approach and by augmenting a new state variable to the feed forward loop. Since the controller structure is consists of a standard optimal feedback regulator and a feed forward controller, it is able to enhance the usefulness of resources and to reduce unnecessary memory reservations such as RAM (Random Access Memory) or SMA (Shared Memory Area) on ordinary router systems, respectively.**

Index Terms— **AQM Router, Congestion Control, LQ-Servo Controller, Trajectory Tracking.**

I. Introduction

It comes to more important issues that congestion controls for the information network under the circumstances as the more and the faster data through the information network increasing. Jacobson and Karels[1] proposed the end-to-end congestion control algorithms which forms the basic for the TCP(Transmission Control Protocol) congestion control. It is a 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. In the last 90's, Floyd and Jacobson[2] presented the RED(Random Early Detection). Its mechanism is that packets are randomly dropped before the buffer of queue overflows. And, Braden et al. [3] proposed the enhanced end-to-end congestion control for Active Queue Management.

In recent years, the more needs for the congestion controllers having enough ability which is more logically predictable and reliable are occurred. For this reason, the traditional control algorithms which have been used only for mechanical or electrical systems are adopted to the area of congested network and their performances which are known as relatively good. Consequently, the more controllers which have various featured types have been adopted to the network congestion control area using control theories

Manuscript received July 24, 2008.

Kang Min Lee is with the DSC Div. Advanced Development Group, SAMSUNG Techwin, Suwon, Korea (e-mail: kmry.lee@samsung.com , edwardjunior@hitel.net).

Ji Hoon Yang is with the Electrical Engineering Department, Hanyang University, Seoul, Korea (e-mail: openyj@hanyang.ac.kr).

Byung Suhl Suh is with the Electrical Engineering Department, Hanyang University, Seoul, Korea (corresponding author to provide phone: +82-2-2220-0364; fax: +82-2-2220-1856; e-mail: bssuh@hanyang.ac.kr).

On the issue of applying control theories to the network, especially AQM Router, Misra *et al.*[4] developed a methodology to model and obtain expected transient behavior of networks with Active Queue Management Routers supporting TCP flows. And Hollot *et al.* approximated its linearized model[5] using small-signal linearization about an operating point to gain insight for the purpose of feedback control, and designed the PI controller[6] based on the linear control theory. Its main contribution is to convert the congestion control algorithm into the controller design problem within the framework of control theory in AQM system. And also, [7] and [8] have used fundamental control theories to analyze and develop for AQM. More recently, Yang and Suh[9] have proposed the robust PID controller using LQ approach, the robust AQM controllers have developed by the optimal control theory based on the Linear Matrix Inequalities (LMI) [10] and the robust μ-analysis technique [11] for the stability and performance issues in AQM. These controllers maintain the queue size which is given at the starting moment of the system without any fluctuation. In these types of system operation case, it is impossible to use all resources (memories) being occupied by another software application because all those resources were privately dominated by previous one.

In order to cope with various unpredictable critical problems and share the memory resource peacefully, we induce congestion controller to more flexible one based on LQ-Servo structure of M. Athans [12]. Since it can follow the rectangular command input, we can change input reference command as the reference queue size in AQM system. Of course, we already knows the fact that the controllers don't have a servo structure can follow the varying reference command, however, it cannot have been satisfied the system stability, and most of all, system performances at all. Basically, this 'servo' operation enhances the usefulness of resources and furthermore, reduces unnecessary memory reservations. Above all, servo can induce condition that strengthen the system stability and can prevent router system from fatal error.

This paper is revised version of work appearing in [13]. The basic ideas are: we design LQ-Servo controller for the AQM router by combining two features of LQ optimization and servo structure. Consequently, we can get both satisfactions for the stability robustness and performance robustness. Each one is inborn attribute of LQ-Optimization and augmented servo structure.

This paper is organized as follows: A description of the rational linearized AQM model is reviewed in Section II.

Section III describes that a servo control structure using Linear Quadratic approach that is called LQ-Servo, and that the LQ-Servo structure is extended by augmenting a new state variable. We derive an implemented controller for AQM model using LQ-Servo structure in Section IV. Finally, some simulation results are presented in Section V.

II. AQM MODEL

This section describes a brief review for mathematical models of AQM based on a control theoretical approach.

A. Dynamic Model of TCP Behavior

A mathematical dynamic model of TCP behavior is developed by Misra *et al.* [4] using fluid-flow and stochastic differential equations. The model becomes as following (1):

$$\dot{W}(t) \approx \frac{1}{R(t)} + \big(1 - Q(W)\big)\left(-\frac{W(t)W\big(t-R(t)\big)}{2R\big(t-R(t)\big)}\right)p\big(t-R(t)\big) + \big(1+W(t)\big)Q(W)\frac{W\big(t-R(t)\big)}{R\big(t-R(t)\big)}p(t-R(t))$$

$$\dot{q}(t) = \sum_{i=1}^{N}\frac{W(t)}{R(t)} - C \tag{1}$$

where $\dot{W}(t)$ 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 = 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 function $Q(W)$ determines the probability that one loss is caused by a timeout, given that the window size W at the time of the loss.

And Hollot *et al.* [5] use a simplified version of (1) which ignores the TCP timeout mechanism. The non-linear differential equations are following as:

$$\dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)W\big(t-R(t)\big)}{2R\big(t-R(t)\big)}p\big(t-R(t)\big)$$

$$\dot{q}(t) = \frac{W(t)}{R(t)}N(t) - C \tag{2}$$

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 within $[0,1]$. Figure 1 shows these differential equations in the block diagram which highlights TCP window-control and queue dynamics.

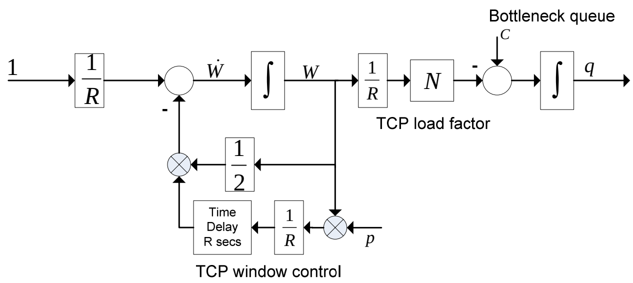

Figure 1. Block-diagram of TCP's congestion-avoidance flow-control mode

B. Linearization of AQM Model

For the control theoretical analysis, (2) was approximated as a linearized constant model by small-signal linearization about an operating point (W_0, q_0, p_0). Brief reviews for the linear model are shown below, and see [5] for linearization details.

The operating point is defined by $\dot{W} = 0$ and $\dot{q} = 0$ so that it is accomplished by conditions:

$$\dot{W}(t) = 0 \rightarrow W_0^2 p_0 = 2$$

$$\dot{q}(t) = 0 \rightarrow W_0 = \frac{R_0 C}{N} \tag{3}$$

And take the first chance of RTT as a input delay

$$R_0 \cong \frac{q_0}{C} + T_p \tag{4}$$

where T_p is propagation delay (seconds).

We finally get the dynamics of TCP/AQM router

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

$$\delta\dot{q}(t) = \frac{N}{R_0}\delta W(t) - \frac{1}{R_0}\delta q(t) \tag{5}$$

where

$$\delta W(t) \cong W - W_0$$

$$\delta q(t) \cong q - q_0$$

$$\delta p(t) \cong p - p_0 \tag{6}$$

Thus, the block diagram of linearized AQM control system is shown in Fig. 2. In this diagram, $P_{tcp}(s)$ is the transfer function of the TCP behavior, $P_{queue}(s)$ is transfer function of the queue dynamics, and C(s) denoted the transfer function of controller.

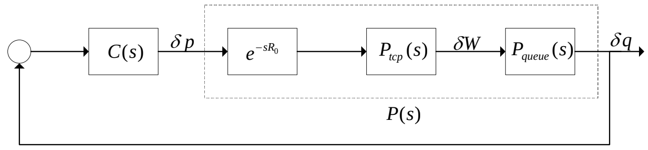

Figure 2. Block diagram of a linearized AQM as feedback control

The transfer functions of $P_{tcp}(s)$ and $P_{queue}(s)$ become as following, respectively,

$$P_{tcp}(s) = \frac{\frac{R_0C^2}{2N^2}}{s + \frac{2N}{R_0^2C}} \tag{7}$$

$$P_{queue}(s) = \frac{\frac{N}{R_0}}{s + \frac{1}{R_0}} \tag{8}$$

where R_0 = Round Trip Time (RTT), N = no. of active TCP session, C = link capacity (packet/sec)

And also, the plant transfer function which is denoted as $P(s) = P_{tcp}(s)P_{queue}(s)e^{-sR_0}$ can be expressed as:

$$P(s) = \frac{\frac{C^2}{2N}e^{-sR_0}}{\left(s + \frac{2N}{R_0^2C}\right)\left(s + \frac{1}{R_0}\right)} \tag{9}$$

III. LQ-Servo Systems

The objective of this section is to describe a servo control structure using Linear Quadratic approach that is called LQ-Servo. And the LQ-Servo structure is extended by augmenting with a new state variable.

A. LQ-Servo

The servo problem is the task of controlling a system so that the system output follows a reference command signal such as step changes. We consider a useful servo problem.

Suppose we are given the n-dimensional linear system having state equations as following:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{10}$$

$$y(t) = Cx(t) \tag{11}$$

where A is a system matrix, B is a input matrix, C is a output matrix, $x(t) \in R^n$, and the m entries of y(t) are linearly independent or, equivalently, the matrix C has rank m.

Without loss of generality, the optimal servo problem based on Linear Quadratic approach, that is called LQ-Servo, is to find the optimal control law u(t) for the linear system (10) by minimizing the cost functions

$$J = \int_0^\infty [x^T(t)Qx(t) + u^T(t)Ru(t)]dt \tag{12}$$

where a weighting matrix Q is symmetric and positive semi-definite, and a weighting factor R is positive value.

In order to design LQ-Servo systems that the output signal follows the reference command signal, an output variable is included with the state variables, and then the output vector becomes one part of the state vectors. So, the state variable x(t) can be divided as followings:

$$x(t) = \begin{bmatrix} x_r(t) \\ y_p(t) \end{bmatrix} \tag{13}$$

where $x_r(t) \in R^{n-m}$ and $y_p(t) \in R^m$.

Then, the state-space model of (10) and (11) becomes as:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{14}$$

$$y(t) = C_px(t) \tag{15}$$

$$x_r(t) = D_px(t) \tag{16}$$

where $C_p = [0_{m\times(n-m)} \quad I_{m\times m}]$ and $D_p = [I_{(n-m)\times(n-m)} \quad 0_{(n-m)\times m}]$.

It is well known that the following optimal control law is obtained

$$u(t) = -Gx(t) \tag{17}$$

and the gain matrix G can be partitioned as

$$G = [G_r \quad G_y] \tag{18}$$

where G_r is $m \times (n-m)$ matrix and G_y is $m \times m$ matrix.

Thus, the optimal control u(t) of (17) can be expressed as:

$$u(t) = -G_rx_r(t) - G_yy_p(t) \tag{19}$$

Substituting (19) into (14), the closed loop state-equation is obtained as:

$$\begin{aligned}\dot{x}(t) &= Ax(t) - BG_rx_r(t) - BG_yy_p(t) \\ &= \left[A - BG_rD_p - BG_yC_p\right]x(t)\end{aligned} \tag{20}$$

Figure 3 illustrates the block diagram of LQ-Servo which is considering both of the reference command r(s) and the disturbance d(s) of output.

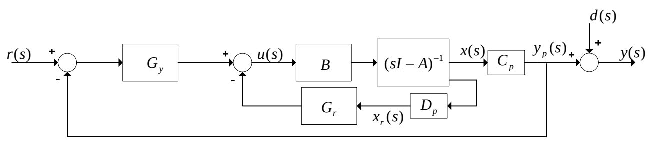

Figure 3. Block-diagram of LQ-Servo

It is noted that the optimal LQ-Servo system can be derived by using linear quadratic regulator theory. There is a result for a two-degree-of-freedom controller involving a standard optimal feedback regulator and a feed forward controller.

B. Augmented LQ -Servo

Since LQ-Servo can happen to a steady state error according to $\mathrm{r(s)}$ or $\mathrm{d(s)}$ in Fig. 3, a new integrator state variable $\mathrm{z_p(t)}$ is added to the feed forward loop of LQ-Servo structure to complement the steady state error.

The new state variable $\mathrm{z_p(t)}$ is defined as:

$$\mathrm{z_p(t)} = \int_0^t \mathrm{y_p(\tau)d\tau} \tag{21}$$

So, the augmented state-variable descriptions are

$$\mathrm{x(t)} = \begin{bmatrix} \mathrm{x_r(t)} \\ \mathrm{y_p(t)} \\ \mathrm{z_p(t)} \end{bmatrix} \tag{22}$$

and also (14) and (15) become as following (23) and (24), respectively,

$$\dot{\mathrm{x}}\mathrm{(t)} = \mathrm{A_{aug}x(t)} + \mathrm{B_{aug}u_p(t)} \tag{23}$$

$$\mathrm{y(t)} = \mathrm{C_{aug}x(t)} \tag{24}$$

where $\mathrm{A_{aug}} = \begin{bmatrix} \mathrm{A} & 0 \\ \mathrm{C} & 0 \end{bmatrix}$, $\mathrm{B_{aug}} = \begin{bmatrix} \mathrm{B} \\ 0 \end{bmatrix}$ and $\mathrm{C_{aug}} = [\mathrm{C_p} \quad 0]$.

In LQ control design, , the optimal control input of (23) is obtained as:

$$\mathrm{u_p(t)} = -\mathrm{Gx(t)} \tag{25}$$

where $\mathrm{G} = -\mathrm{R^{-1}B_{aug}^T K}$ and $\mathrm{K} = \mathrm{K^T}$ is a solution matrix of the algebraic Riccatti's equation:

$$\mathrm{KA_{aug}} + \mathrm{A_{aug}^T K} + \mathrm{Q} - \mathrm{KB_{aug}R^{-1}B_{aug}^T K} = 0 \tag{26}$$

Suppose the gain matrix G is decomposed into $\mathrm{G} = [\mathrm{G_r} \quad \mathrm{G_y} \quad \mathrm{G_z}]$, the optimal control input of (25) can be expressed as

$$\mathrm{u_p(t)} = -\mathrm{G_r x_r(t)} - \mathrm{G_y y_p(t)} - \mathrm{G_z z_p(t)} \tag{27}$$

Therefore, we can modify the LQ-Servo structure as shown in Fig. 4 to ensure that zero steady state error is robustly achieved in response to a constant reference commands.

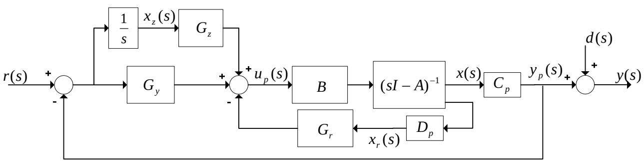

Figure 4. LQ-Servo with augmented state variable

IV. LQ-SERVO CONTROL FOR AQM

In this section, we present the controller implementation for AQM model using the LQ-Servo structure.

A. State-Space Model of AQM

Let the state variable $\mathrm{x(t)}$ of (5) be defined as:

$$\mathrm{x(t)} = \begin{bmatrix} \mathrm{x_r(t)} \\ \mathrm{y_p(t)} \end{bmatrix} = \begin{bmatrix} \mathrm{\delta W(t)} \\ \mathrm{\delta q(t)} \end{bmatrix} \tag{28}$$

The (4) can be represented with state-space model:

$$\dot{\mathrm{x}}\mathrm{(t)} = \mathrm{Ax(t)} + \mathrm{Bu(t - R_0)}$$

$$\mathrm{y(t)} = \mathrm{C_p x(t)} \tag{29}$$

where $\mathrm{y(t)(= \delta q(t))}$ is an output variable, $\mathrm{u(t)(= \delta p(t))}$ is a control input.

And also, the system matrix, input matrix and output matrix of (11) can be expressed as following, respectively:

$$\mathrm{A} = \begin{bmatrix} -\frac{2\mathrm{N}}{\mathrm{R_0^2 C}} & 0 \\ \frac{\mathrm{N}}{\mathrm{R_0}} & -\frac{1}{\mathrm{R_0}} \end{bmatrix}, \mathrm{B} = \begin{bmatrix} -\frac{\mathrm{R_0 C^2}}{2\mathrm{N}^2} \\ 0 \end{bmatrix},$$

$$\mathrm{C_p} = [0 \quad 1] \text{, and } \mathrm{D_p} = [1 \quad 0] \tag{30}$$

In this paper, $\mathrm{R_0}$ as an input delay on (29) is ignored because of its mere affection on entire control system. And one of the AQM system state variables W that can't be measured exactly on router itself is simply approximated to the average incoming packet size. In practice, router can observe and detect total incoming packets which have mixed form. Thus, it can count the total number of packets that comes from every end nodes at every sampling period.

The window size is

$$\mathrm{W} = \text{Incoming Packets} / N \tag{31}$$

B. LQ-Servo control for AQM

In order to follow a reference input command trajectory which is continuously varying, our controller must have at least one integrator which is associated with proper state variables. For the AQM system model, the reference queue size is the reasonable state variable. In this paper, rectangular reference input commands are used for the test of continuous varying command. Thus, just one integrator is adapted to AQM system.

Integrator adoption process which is finished to augmented system matrix $\mathrm{A_{aug}}$ is formed via hiring new state variable

$$\mathrm{z_p(t)} = \int_0^t \mathrm{q(\tau)d\tau} \tag{32}$$

So, the augmented state-variable descriptions become

$$x(t) = \begin{bmatrix} x_r(t) \\ y_p(t) \\ z_p(t) \end{bmatrix} = \begin{bmatrix} \delta W(t) \\ \delta q(t) \\ \int_0^t q(\tau)d\tau \end{bmatrix} \quad (33)$$

and the state-space model of (9) becomes as following:

$$\dot{x}(t) = A_{aug}x(t) + B_{aug}u(t - R_0)$$

$$y(t) = C_{aug}x(t) \quad (34)$$

And also, the system matrix, input matrix and output matrix can be presented as, respectively,

$$A_{aug} = \begin{bmatrix} -\frac{2N}{R_0^2 C} & 0 & 0 \\ \frac{N}{R_0} & -\frac{1}{R_0} & 0 \\ 0 & 1 & 0 \end{bmatrix}, B_{aug} = \begin{bmatrix} -\frac{R_0 C^2}{2N^2} \\ 0 \\ 0 \end{bmatrix}$$

$$\text{and } C_{aug} = [0 \quad 1 \quad 0] \quad (35)$$

It is certain that both system matrices A and A_{aug} has same eigen values.

Figure 5 illustrates the block-diagram of LQ-Servo controller for AQM. And, the proposed LQ-Servo controller parameters as the gain matrix of (25) are achieved by solving the algebraic Riccatti's equation (26).

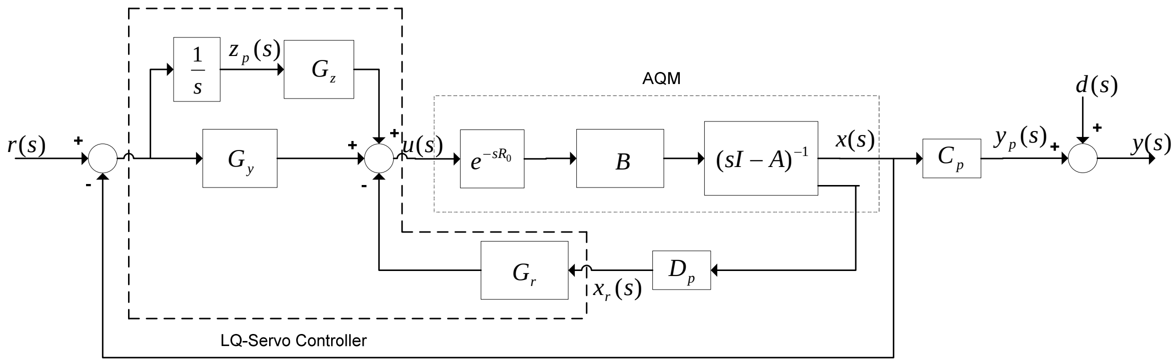

Figure 5. LQ-Servo Controller for AQM

Remark 1. It is noted that we can set the weighting matrix Q to the identity matrix I and the weighting factor R to 2 for convenience in this paper.

Remark 2. In practical implementation, conditions (6) must be verified. At first, W_0 can be derived from certain constant values R_0 , C , N . And q_0 have to be reference input that is continuous varying. The state variables x_r and y_p are a value from observing the window size (31) and the queue length, respectively. And also, the state variable z_p is a value of an accumulated error between the reference queue size and $\delta q(t)$. This z_p may strengthen the performance robustness on tracking issue.

V. SIMULATION

We can verify the tracking performance of some network congestion controller by simulation using the ns-2 simulator [14].

Experiments are a mixture of FTP and HTTP sources as following Fig. 6.

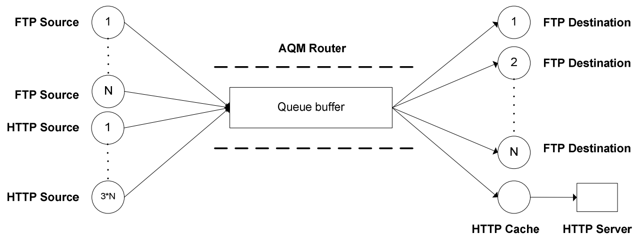

Figure 6. AQM Router Diagram

For the first time, PI controller [6] which has non trajectory tracking structure is tested as a comparative reference. Figure 7 shows its performance for the continuous varying reference command. The reference input (queue size) has rectangular form and changes every 50seconds. Initial value start with queue size 300 and then, reference queue size is changes to 200 and 500, 200. Sampling time parameter is set to 1/160second for all experiments

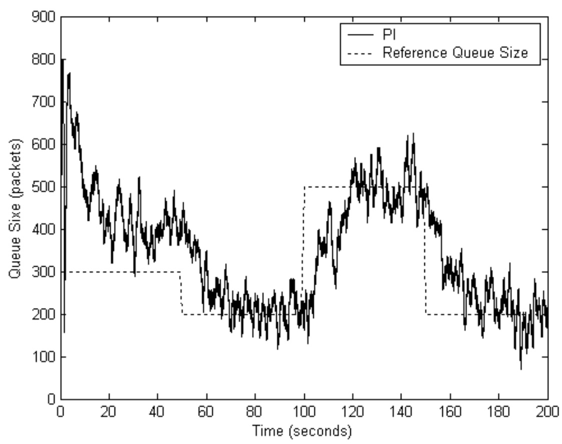

Figure 7. Tracking performance of the PI controller

Next, some experiments can verify the tracking performance of the LQ-Servo controller with some different AQM model parameters.

A. Experiment I

To evaluate tracking performance, we compare LQ-Servo with PI controller at first on the ns-2. (35), the continuous (analog) system and input matrices are

$$A_{aug} = \begin{bmatrix} -0.5288 & 0 & 0 \\ 243.9024 & -4.0650 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B_{aug} = \begin{bmatrix} -480.4688 \\ 0 \\ 0 \end{bmatrix}$$

In order to simulate, this continuous model must convert to discrete one. Simply, command 'c2d' makes continuous model into appropriate one on Matlab.

Finally, we can get the control gain for the experiment 1.

$$G = [-0.4373 \quad -0.1607 \quad -0.1662]$$

Both comparative reference PI and LQ-Servo controllers have same values of the network parameters.

$R_0 = 0.246$ (second)

$C = 15$Mbps (3750 packets/sec)

$N = 60$

The reference input (queue size) also has rectangular form and changes every 50seconds. Initial value start with queue size 300 and then, queue size is changes to 200 and 500, 200. The result is shown in Fig. 8.

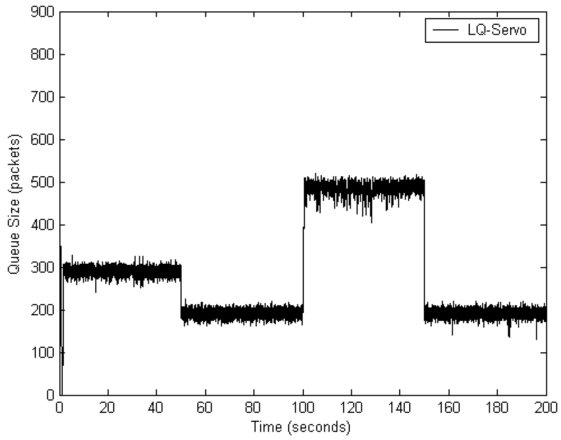

Figure 8. Tracking performance of LQ-Servo controller for experiment I

B. Experiment II

Simulation is accomplished with similar to experiment I, except that more load factors (N) are added and also link capacity (C) is reduced.

$R_0 = 0.246$ (second)

$C = 8$Mbps (2000 packets/sec)

$N = 120$

The continuous (analog) system and input matrices are

$$A_{aug} = \begin{bmatrix} -1.9829 & 0 & 0 \\ 487.8049 & -4.0650 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B_{aug} = \begin{bmatrix} -34.1667 \\ 0 \\ 0 \end{bmatrix}$$

Control gain for the experiment II.

$$G = [-3.4654 \quad -0.4220 \quad -0.4440]$$

The reference input (queue size) also has rectangular form and changes every 50seconds. Initial value start with queue size 300 and then, queue size is changes to 400 and 200, 350. Figure 9 shows the simulation result.

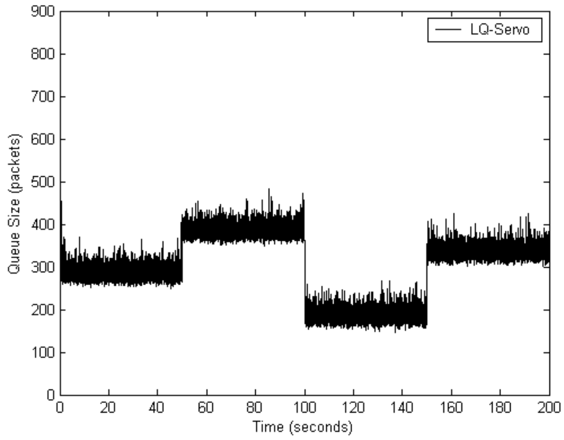

Figure 9. Tracking performance of LQ-Servo controller for experiment II

It is remarked that Figure 7 and Figures 8~9 indicate the simulation results of both the PI controller and the proposed LQ-Servo controller with respect to changed reference queue size, respectively. These results show that the proposed controller has the better performance in a trajectory tracking problem in AQM routers.

VI. CONCLUSION

This paper presents the LQ-Servo controller dealing with a good tracking performance. This controller structure is made by taking a traditional servo mechanism based on Linear Quadratic approach and by augmenting a new state variable to the feed forward loop of LQ-Servo structure. 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. On the LQ-Servo simulation result, not only steady state wave form has a stable tendency but also it shows the better performances at every reference changing point.

REFERENCES

[1] V. Jacobson and M. J. Karels. "Congestion Avoidance and Control," In SIGCOMM'ss, 1988.G.

[2] S. Floyd and V. Jacobson. “Random Early Detection gateways for Congestion Avoidance,” IEEE/ACM Transactions on Networking, vol 1, no. 4, 1997.

[3] B. Braden et al. “Recommendations on Queue Management and Congestion Avoidnace in the Internet,” RFC2309, April 1998.

[4] V. Misra, W.-B. Gong, and D. Towsley. “Fluid-based analysis of a nerwork of AQM routers supporting TCP flows with an application to RED,” in Proc. ACM SIGCOMM, 2000. pp. 151-160.

[5] C. V. Hollot, V. Misra, D. Towsley, and W. B. Gong, “A control theoritic analysis of RED,” in Proc. IEEE INFOCOM, 2001.

[6] C. V. Hollot, V. Misra, D. Towsley, W.-B. Gong. “Analysis and Design of Controller for AQM Routers Supporting TCP Flows,” IEEE Transcations on Automatic Control, vol. 47, no. 6, pp. 945-959. 2002.

[7] J. Aweya, M. Ouellette and D. Y. Monyuno, “A control theoretic approach to active queue management,” Computer networks, vol. 36, issues 2-3, pp. 203-235, 2001.

[8] F. Ren, C. Lin, X. Ying, X. Shan and F. Wang, “A robust active queue management algorithm based on sliding mode variable structure control,” proc. IEEE/INFOCOM, 2002.

[9] J. H. Yang and B. S. Suh. “Robust PID Controller for AQM based on Linear Quadratic Approach,” IAENG International Conference on Communication Systems and Applications, Hong Kong, pp. 1083-1087, March 19-21, 2008.

[10] M. M. A. E. Lima, N. L. S. Fonseca and J. C. Geromel. “An Optimal Active Queue Managemnet Controller,” IEEE Communications Society, pp. 2261-2266. 2005

[11] Q. Chen and O. W. W. Yang. “On Designing Traffic Controller for AQM Routers Based on Robust u-Analysis,” IEEE Globecom, pp. 857-861, 2005.

[12] M. Athans, “Multivarible Control System,” Athan’s Control Lecture Note, MIT.

[13] K. M. Lee, J. H. Yang and B. S. Suh. “Design of LQ-Servo Controller for Active Queue Management Routers” IAENG International Conference on Control and Automation, Hong Kong, pp. 1256-1260, March 19-21, 2008.

[14] “ns-2 Network Simulator,” Obtain from http://www.isi.edu/nsnam/ns/.