Design of AQM in Supporting TCP Based on the Well-Known AIMD Model

Ki Baek Kim,   Ao Tang,   Steven H. Low

Departments of CS and EE
California Institute of Technology, Pasadena, CA 91125
kkb@di.ens.fr, aotang@caltech.edu, slow@caltech.edu

Abstract— In this paper, we investigate how to design AQM with a low-pass filter (average queuing) in supporting TCP based on the well-known AIMD dynamic model sharing a common bottleneck router. Since we formulate the AQM design problem for the given TCP as a state-space model, we get two interesting and important results:

1) We derive PD-type (Proportional-Derivative) AQM structure in terms of queue length which supports the conjecture in the literature that the AQM RED is not enough to control TCP dynamic behavior, where RED can be classified as P-type AQM structure;

2) We propose a delay-dependent AQM structure to compensate for delays in congestion measure explicitly from the knowledge of RTT, capacity and number of sources, where the previous AQMs RED, REM/PI and AVQ are delay-independent controls.

As one trial to obtain stabilizing gains of the proposed structures, we minimize a LQ (linear quadratic) cost of the transients on average-queue length, average-queue length rate, jitter in the rate and the loss probability. Finally, we illustrate the above theoretical results through *ns* simulations for TCP Reno.

I. Introduction

Since TCP Reno/AQM Droptail has been proposed in [2], the current internet is still using Reno/Droptail and its variants as a congestion control strategy.

Droptail, which drops packets when the queue is full, is not appropriate as a feedback control for high-speed networks since links are likely to have large queues in high-speed networks and this may increase the average delay in the network. More importantly, Droptail can often cause the global synchronization [3], [4] and thus have low throughput. In order to overcome these problems, RED (Random Early Detection) has been suggested in [5], [6] which randomly drops packets proportionally to the average queue length. Since then, there have been a lot of investigations about how to tune design parameters in RED [7], [8], [9]. They show that RED is not enough to stabilize TCP and thus not easy to fully utilize the given network resources. As a result, new AQM algorithms such as BLUE [10], AVQ (Adaptive Virtual Queue) [11] and REM (Random Exponential Marking) [12] have been suggested. These AQM structures are designed to achieve additional performances such as buffer clearing. However,

all these papers did not address appropriately how to tune gains of their AQM structures for closed-loop stability since they lack of a TCP dynamic model, in spite that stabilizing the closed-loop dynamic behavior is necessary to get the maximum utilization. So, their AQM algorithms are only compared through simulations in the literature.

In order to address this problem, paper [13] has developed a dynamic model to reflect AIMD (Additive Increase Multiplicative Decrease) mode of TCP and paper [14] has applied the transfer function approach to the stabilizing RED design problem based on the AIMD model. As a sequence, papers [15], [16] have investigated how to scale gains of PI-type REM (Proportional-Integral) in terms of queue length and P-type AVQ in terms of aggregate, respectively, for closed-loop stability. The papers [17], [18] have suggested PI-type AQM in terms of aggregate in an inner loop. However, all these results based on the transfer function approach *do not consider what is a natural state-feedback control* to stabilize the given TCP and queue dynamics. In addition, they did not consider how to compensate for delays explicitly even if they know the previous dynamic information and delays, where this kind of control strategy is called delay-independent (or memoryless) control in the control literature. It is well-known that the delay-independent control has a limit on performance in the presence of a large delay [19], [20] compared with the delay-dependent (or memory) control, i.e., we cannot fully utilize the given network resources.

The present paper studies how to design AQM with a low-pass filter (average queuing) for stabilizing the AIMD model of [13] in the presence of arbitrary delays. This work builds upon the AIMD model introduced in [13], [14]. As a first step to get possible basic results for more realistic cases, we study a simplified version of this problem, simplified in three regards as in [14], [15], [17], [18], [21]. First, we assume single link, homogeneous sources with the same window size, and no short flows (Some extension of this paper to multiple links and heterogeneous delays is discussed in our companion paper [22]). Second, we consider the linearized version of the TCP function, so the stability means the local stability near the equilibrium, the variables denote perturbations around the equilibrium, and the cost measures the deviation from the equilibrium point. For example, a slower transient will incur a higher cost. Third, when we try to compensate for delays explicitly and get stabilizing optimal gains of the derived AQM structure, information of the networks is assumed to

be known accurately. In addition, we do not study the case of input/state constrained systems and uncertain systems which are intrinsically included in real networks and ns simulator. Thus, the simulation results of this paper via the non-deterministic nonlinear ns simulator cannot be considered as a direct illustration of the proposed results, but they clearly backup the proposed results qualitatively.

Then, we formulate the AQM design problem with a low-pass filter of RED as a state-space model as in [21], [22], [23]. Thereby, we obtain the PD-type state-feedback AQM structure in terms of queue length which implies the structural deficiency of P-type RED for closed-loop stability, compensate for delays in congestion measure explicitly by using a memory control structure, and obtain a stabilizing gain of the proposed AQM structure by minimizing a linear quadratic (LQ) cost of the transients on the state variables and feedback control of the state-space model. Finally, we illustrate the proposed results via ns simulations.

II. DESIGN OF AQM WITH A LOW-PASS FILTER

Consider a single link of capacity c shared by N homogeneous TCP Reno sources and queue dynamics modeled by [13]

$$\dot{w}_s(t) = \frac{w_s(t - \tau_s(t))}{d_s + b(t - \tau_s(t))/c}(1 - p(t - \tau_s^b(t)))\frac{1}{w_s(t)}$$
$$- \frac{w_s(t - \tau_s(t))}{d_s + b(t - \tau_s(t))/c}p(t - \tau_s^b(t))\frac{w_s(t)}{2}$$

$$\dot{b}(t) = -c + \sum_{s=1}^{N} \frac{w_s(t - \tau_s^f(t))}{d_s + b(t - \tau_s^f(t))/c}, \qquad (1)$$

where $w_s(t)$ is (expected) TCP window size of source s at time t, $\tau_s^f(t)$ is the forward delay from source to router, $\tau_s^b(t)$ is the backward delay from router to source, $\tau_s(t)$ is the round trip time ($\tau_s(t) = \tau_s^f(t) + \tau_s^b(t) = d_s + b(t)/c$), d_s is the round trip propagation delay, $p(t)$ is the loss probability at time t that is implemented at links, $b(t)$ is the queue length at time t, and c is the link capacity, in packets/sec. In the above model, the first and second parts of the first equation represent AI and MD behaviors at sources, respectively, where the second equation represents the queue dynamics at link.

As in [14], we assume sources are identical $d_s = d$ and all have the common window $w_s(t) \equiv w(t)$; we assume forward delays are zero, $\tau_s^f(\cdot) = 0$, so that $\tau_s(t) = \tau_s^b(t)$. Let (w^*, b^*, p^*) be the equilibrium point. Then the equilibrium round trip time τ is related to b^* by $\tau = d + b^*/c$.

From (1), we can derive the following model of the linearized TCP and queue dynamics:

$$\delta\ddot{b}(t) = A_1\delta b(t) + A_2\delta\dot{b}(t) + B_1\delta p(t - \tau), \qquad (2)$$

where $\delta b(0)$, $\delta\dot{b}(0)$, and $\{\delta p(\sigma), \sigma \in [-\tau, 0]\}$ are given, $\delta b(t) = b(t) - b^*$, $\delta\dot{b}(t) = \dot{b}(t)$, $\delta\ddot{b}(t) = \ddot{b}(t)$, $\delta p(t) = p(t) - p^*$, $A_1 = -\frac{2cN}{\tau(2N^2 + c^2\tau^2)}$, $A_2 = -\frac{2cN\tau + 2N^2 + c^2\tau^2}{\tau(2N^2 + c^2\tau^2)}$, and $B_1 = -\frac{2N^2 + c^2\tau^2}{2\tau^2N}$. Refer to [21] for derivation of (2). Note that paper [21] does not consider the following average queuing dynamics in RED, where the additional dynamics can make the original stable TCP dynamics unstable.

Consider the dynamics of average queuing $\hat{b}(t)$ in RED which is called the low-pass filter in the control literature:

$$\dot{\hat{b}}(t) = -P_1\hat{b}(t) + P_1 b(t), \qquad P_1 > 0. \qquad (3)$$

Here, P_1 is a design parameter which decides the cut-off frequency [24]. From (2) and (3), we can get

$$\delta\dddot{\hat{b}}(t) = \hat{A}_2\delta\ddot{\hat{b}}(t) + \hat{A}_1\delta\dot{\hat{b}}(t) + \hat{A}_3\delta\hat{b}(t)$$
$$+ P_1 B_1\delta p(t - \tau), \qquad (4)$$

where $\delta\dddot{\hat{b}}(t) = \dddot{\hat{b}}(t)$, $\delta\ddot{\hat{b}}(t) = \ddot{\hat{b}}(t)$, $\delta\dot{\hat{b}}(t) = \dot{\hat{b}}(t)$, $\delta\hat{b}(t) = \hat{b}(t) - b^*$, $\hat{A}_1 = A_1 + A_2 P_1$, $\hat{A}_2 = A_2 - P_1$, $\hat{A}_3 = A_1 P_1$. The differential equation (4) can be represented as the following state-space model:

$$\dot{z}(t) = Az(t) + B\delta p(t - \tau), \qquad (5)$$

where $z(0)$ and $\{\delta p(\sigma), \sigma \in [-\tau, 0]\}$ are given,

$$z = \begin{bmatrix} \delta\hat{b} \\ \delta\dot{\hat{b}} \\ \delta\ddot{\hat{b}} \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ \hat{A}_3 & \hat{A}_1 & \hat{A}_2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ P_1 B_1 \end{bmatrix}.$$

The above state-space model is a minimal representation with state-space variables ($\delta\hat{b}(t)$, $\delta\dot{\hat{b}}(t)$, $\delta\ddot{\hat{b}}(t)$) of (4). The control strategy based on a state-space model is called modern control in the control literature, while that based on a transfer function model as in [14], [15], [16], [17], [18] is called classical control. Since the state-space approach was developed in 1950s, it has been widely investigated due to many advantages over the transfer function approach. Please refer to any control literature for more details [24].

From the above state-space model, we can naturally get the PD-type state-feedback AQM in terms of queue length

$$\delta p(t) = Hz(t) = H_1\delta\hat{b}(t) + H_2\delta\dot{\hat{b}}(t) + H_3\delta\ddot{\hat{b}}(t) \qquad (6)$$

if we ignore the time delay τ (i.e., $\tau = 0$) in the control $\delta p(t - \tau)$. Note that this structure (6) is not captured in the networking literature. Since RED has P-type control structure $\delta p(t) = H_1\delta\hat{b}(t)$, our derivation based on AIMD model (1) shows that RED is not enough to stabilize the TCP dynamics. How to obtain a pair of a stabilizing gain (H_1, H_2, H_3) is discussed later in this paper.

If we use a memoryless control, i.e., we use only the current state information $z(t)$ for the delayed control $\delta p(t - \tau)$, it is well-known that it is difficult to stabilize a dynamical system in the presence of a large delay [19], [20]. Thus, we cannot fully utilize the given network resources in the presence of large delays τ. Next, we overcome this problem by compensating for delays explicitly.

A. Explicit delay compensation by a memory control

As one way to compensate for large delays in congestion measure explicitly, we need a delay-dependent control that uses not only the current dynamic information at time t but

also the previous dynamic information from $t - \tau$ to t for delayed dynamical system (5). To this end, we define

$$
\begin{aligned}
\hat{A}_1 &= A_1 + A_2 P_1, \quad \hat{A}_2 = A_2 - P_1, \quad \hat{A}_3 = A_1 P_1 \\
\hat{B}_1 &= T_2 \hat{A}_3 \tilde{B}_1 + T_3 \tilde{B}_2 + T_4 \tilde{B}_3 \\
\tilde{B} &= [\tilde{B}_1^T, \quad \tilde{B}_2^T, \quad \tilde{B}_3^T]^T = e^{-A\tau} B \\
T_1 &= -\frac{T_2(\hat{A}_3 \tilde{B}_1 + \hat{A}_1 \tilde{B}_1 \tilde{B}_2^{-1} \tilde{B}_3 + \hat{A}_2 \tilde{B}_3 - \tilde{B}_2^{-1} \tilde{B}_3^2)}{D_1} \\
&\quad \text{for some} \quad T_2 \neq 0 \\
T_3 &= T_2 \hat{A}_1 + \frac{T_2(\hat{A}_3 \tilde{B}_2 - \hat{A}_3 \tilde{B}_1 \tilde{B}_2^{-1} \tilde{B}_3)}{D_1} \\
T_4 &= T_1 + \hat{A}_2 T_2 \\
D_1 &= \hat{A}_3 \tilde{B}_1^2 \tilde{B}_2^{-1} + \hat{A}_1 \tilde{B}_1 + \hat{A}_2 \tilde{B}_2 - \tilde{B}_3.
\end{aligned}
$$

The key to deriving an explicit memory control for the delayed system (5) is to transform delayed system (5) to the equivalent nominal system

$$
\dot{s}(t) = A s(t) + \hat{B} \delta p(t), \tag{7}
$$

where $s(t) = [s_1(t), \ s_2(t), \ s_3(t)]^T$, $\hat{B} = [0, \ 0, \ \hat{B}_1]^T$

$$
\begin{aligned}
s_1(t) &= [T_1 - (T_3 - T_2 \hat{A}_1)\hat{A}_3^{-1}\hat{A}_1](\delta \hat{b}(t) + u_{1\tau}(t)) \\
&\quad + [T_2 - (T_3 - T_2 \hat{A}_1)\hat{A}_3^{-1}\hat{A}_2](\delta \dot{\hat{b}}(t) + u_{2\tau}(t)) \\
&\quad + (T_3 - T_2 \hat{A}_1)\hat{A}_3^{-1}(\delta \ddot{\hat{b}}(t) + u_{3\tau}(t)) \\
s_2(t) &= (T_3 - T_2 \hat{A}_1)(\delta \hat{b}(t) + u_{1\tau}(t)) \\
&\quad + T_1(\delta \dot{\hat{b}}(t) + u_{2\tau}(t)) + T_2(\delta \ddot{\hat{b}}(t) + u_{3\tau}(t)) \\
s_3(t) &= T_2 \hat{A}_3(\delta \hat{b}(t) + u_{1\tau}(t)) + T_3(\delta \dot{\hat{b}}(t) + u_{2\tau}(t)) \\
&\quad + T_4(\delta \ddot{\hat{b}}(t) + u_{3\tau}(t)) \\
\begin{bmatrix} u_{1\tau}(t) \\ u_{2\tau}(t) \\ u_{3\tau}(t) \end{bmatrix} &= \int_{-\tau}^{0} e^{-A(\sigma+\tau)} B \delta p(t+\sigma) d\sigma.
\end{aligned}
$$

Refer to Appendix for derivation of (7). Note that the nominal system is asymptotically stable if and only if original delayed system (5) is asymptotically stable.

From the above state-space model, we can get a memory state-feedback AQM

$$
\delta p(t) = H_1^* s_1(t) + H_2^* s_2(t) + H_3^* s_3(t).
$$

Next, we show how to obtain a stabilizing optimal gain (H_1^*, H_2^*, H_3^*) of proposed AQM structure (8).

B. A stabilizing optimal gain for the proposed structure

As a performance measure for linearized system (7), we consider the following optimization problem:

$$
\min_{\delta p(\cdot)} \int_t^{t+\infty} \left(s^T(\sigma) Q s(\sigma) + \delta p^2(\sigma) \right) d\sigma, \tag{8}
$$

where $Q = Q^T \geq 0$. By solving the above optimization problem, we can get stabilizing optimal and inverse optimal gains of the proposed PD-type AQM.

For ease of explanation of the above performance index, assume that $\tau = 0$ (i.e., $s(t) = z(t)$, $\hat{B} = B$). Then, we define the stabilizing optimal gain design problem as the problem of

choosing a feedback control $\delta p(t)$ that minimizes the cost of transient around an equilibrium:

$$
\min_{\delta p(\cdot)} \int_t^{t+\infty} \left(Q_1 \delta \hat{b}^2 + Q_2 \delta \dot{\hat{b}}^2 + Q_3 \delta \ddot{\hat{b}}^2 + \delta p^2 \right) d\sigma.
$$

Each term in the integrand penalizes the average-queue length, average-queue length rate, jitter in the rate, and the fluctuation of the loss probability, respectively. Hence the cost is a weighted sum of transients on average queue, average-queue rate, jitter in the rate, and fluctuation in probability, weighted by $Q_1 > 0$, $Q_2 > 0$, $Q_3 > 0$, and 1.

Proposition 1: The stabilizing optimal gain of proposed PD-type AQM (8), which minimizes transient cost (8), is given by

$$
\begin{aligned}
H_1^* &= -\frac{\hat{A}_3 + \sqrt{\hat{A}_3^2 + \hat{B}_1^2 Q_1}}{\hat{B}_1} \\
H_2^* &= \frac{-2\hat{A}_2 H_3^* - \hat{B}_1 H_3^{*2} + \hat{B}_1 Q_3}{2},
\end{aligned}
$$

and H_3^* is the positive solution of the following fourth order polynomial:

$$
\begin{aligned}
&-\hat{B}_1^4 H_3^{*4} - 4\hat{A}_2 \hat{B}_1^3 H_3^{*3} + (4\hat{A}_1 \hat{B}_1^2 - 4\hat{A}_2^2 \hat{B}_1^2 + 2\hat{B}_1^4 \\
&Q_3)H_3^{*2} - [8\hat{B}_1(\hat{A}_3 + \sqrt{\hat{A}_3^2 + \hat{B}_1^2 Q_1}) - 8\hat{A}_3 \hat{B}_1 - 8\hat{A}_1 \\
&\hat{A}_2 \hat{B}_1 - 4\hat{A}_2 \hat{B}_1^3 Q_3]H_3^* - 8A_2(\hat{A}_3 + \sqrt{\hat{A}_3^2 + \hat{B}_1^2 Q_1}) \\
&\qquad -4\hat{A}_1 \hat{B}_1^2 Q_3 + 4\hat{B}_1^2 Q_2 - \hat{B}_1^4 Q_3^2 = 0,
\end{aligned}
$$

where $H_1^* > -\frac{\hat{A}_3}{\hat{B}_1}$ and $H_2^* > -\frac{\hat{A}_1}{\hat{B}_1}$

Proof: The optimal control that minimizes (8) is given by $\delta p^*(t) = H^* s(t)$ [25]. $\qquad \square$

Proposition 1 implies that the solution of problem (8) is an AQM algorithm, specified by (H_1^*, H_2^*, H_3^*). Conversely, given any AQM of this structure, it solves problem (8) with appropriate weights Q_i, as the next result says. It can be easily proved from Proposition 1.

Proposition 2: Given a stabilizing AQM $\delta p(t) = [H_1 \ H_2 \ H_3]s(t)$, it solves problem (8) with weights

$$
Q_1 = \frac{H_1(2\hat{A}_3 + \hat{B}_1 H_1)}{\hat{B}_1}, \quad Q_3 = H_3^2 + \frac{2(\hat{A}_2 H_3 + H_2)}{\hat{B}_1}
$$

$$
\begin{aligned}
Q_2 &= H_2^2 + \frac{2(\hat{A}_1 \hat{B}_1 H_2 + \hat{A}_2 \hat{A}_3)}{\hat{B}_1^2} \\
&\quad + \frac{2(\hat{A}_2 + \hat{B}_1 H_3)\sqrt{\hat{A}_3^2 + \hat{B}_1^2 Q_1}}{\hat{B}_1^2}.
\end{aligned}
$$

Proposition 3: Given the eigenvalues λ_1, λ_2, and λ_3 of closed-loop system (7) with (8), $\delta p(t)$ solves (8) with weights

$$
Q_1 = \frac{\hat{\lambda}_3^2 - \hat{A}_3^2}{\hat{B}_1^2}, \quad Q_3 = \frac{-\hat{A}_2^2 - 2\hat{A}_1 + \hat{\lambda}_1^2 - 2\hat{\lambda}_2}{\hat{B}_1^2}
$$

$$
Q_2 = \frac{-\hat{A}_1^2 + \hat{\lambda}_2^2 + 2\hat{\lambda}_1 \sqrt{\hat{A}_3^2 + \hat{B}_1^2 Q_1} + 2\hat{A}_2 \hat{A}_3}{\hat{B}_1^2},
$$

where $\hat{\lambda}_1 = \lambda_1 + \lambda_2 + \lambda_3$, $\hat{\lambda}_2 = \lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_1\lambda_3$, and $\hat{\lambda}_3 = \lambda_1\lambda_2\lambda_3$.

We now interpret RED as an approximation of the proposed AQM. For ease of comparison, we assume that $\tau = 0$ (i.e., $s(t) = z(t)$, $\hat{B} = B$) for linearized model.

Remark *1:* The linear model motivated by RED is

$$\delta p^r(t) = H_1^r \delta\hat{b}(t)$$

for some nonnegative constant H_1^r. By Proposition 1, the proposed AQM has $\delta p(t) = H_1\delta\hat{b}(t) + H_2\delta\dot{\hat{b}}(t) + H_3\delta\ddot{\hat{b}}(t)$. Thus, if we set $H_1 = H_1^r$, $H_2 = 0$ and $H_3 = 0$, then we can get the above RED. Note that H_3 should be positive by Proposition 1. Thus, RED cannot be an optimal control structure of problem (8). We can also interpret the structural deficiency of RED as follows. From Propositions 1-3, the sum of eigenvalues of closed-loop system (7) with RED is given by

$$\hat{\lambda}_1 = \hat{A}_2,$$

while the proposed AQM structure in this paper is given by $\hat{\lambda}_1 = \hat{A}_2 + \hat{B}_1 H_3$. Thus, we cannot adjust the sum of eigenvalues with RED, i.e., cannot control the dynamic behavior of the closed-loop system arbitrary while we can do that with the proposed one.

III. *ns* SIMULATIONS

In this section, we illustrate the proposed results via two simulation examples. In each example, we compare the proposed PD-type memoryless AQM with P-type RED and compare the proposed memory PD-type AQM with the proposed memoryless one in the presence of large propagation delays.

For ease of selection of stabilizing optimal gains in Section II-B, we set all eigenvalues of the closed-loop system to be equal (i.e., $\lambda = \lambda_i$ for all i). Then we have only to design $\lambda < 0$ in order to get the stabilizing optimal gains. The dynamic behavior of the closed-loop system depends on how to design λ. A larger $|\lambda|$ makes the system move faster, i.e., makes the system go to the equilibrium faster. However, if the value of $|\lambda|$ is too high, the closed-loop system can also easily deviate from the equilibrium point due to nonlinearity and randomness of *ns* simulator. This problem is more critical for P-type RED than for the proposed PD-type AQM in this paper because of the structural deficiency of RED as discussed in Remark 1. Since the stabilizing gains in this paper are derived from the deterministic linearized model near the equilibrium point, the value of $|\lambda|$ should not be so high. RED-like AQM is obtained by setting $H_2 = H_3 = 0$ after getting the gains H_1, H_2, and H_3 of the proposed stabilizing AQM.

Similarly to [14], [15], we simulate a single bottleneck link with capacity $c = 4000$ pkts/sec shared by $N = 100$ Reno sources. The AQM at the bottleneck link uses ECN marking. Real queue length is limited by $b_{\max} = 800$ pkts. Marking probability of each AQM is updated every 2ms, i.e., the sampling time is $T_s = 2$ms, where $\hat{b}(t) = \hat{b}(t - T_s) + T_s P_1\big(-\hat{b}(t - T_s) + b(t - T_s)\big)$. The simulation duration is 40 sec, where this simulation time is enough to illustrate the proposed results.

A. *RED-like AQM and the proposed memoryless PD-type AQM*

In this example, we compare performances of RED-like AQM and the proposed memoryless PD-type AQM without explicit delay compensation in the presence of delays. We assume that the propagation delay is $d_s = 150$ms, where the round trip time is $\tau^* = d_s + b^*/c$ and the target queue length is $b^* = 200$ pkts.

Figure 1 shows their performances when $\lambda = -3.75$. It illustrates Remark 1 that the proposed AQM is easier to stabilize TCP than RED-like AQM. The average queue lengths of RED-like AQM and the proposed AQM are 434.8 pkts and 205.3 pkts, respectively and utilization of RED-like AQM and the proposed AQM are 93 % and 95 %, respectively. Thus, the proposed PD-type AQM has a smaller queuing delay but a larger throughput than RED-like AQM.

B. *Performance of a memory stabilizing AQM*

In this example, we illustrate the performance of a memory control structure by comparing a memory PD-type AQM with a memoryless one. Here, we hold all simulation parameters of the previous subsection except changing the propagation delay into $d_s = 350$msec.

Figure 2 illustrates that the proposed memory AQM is much closer to the target queue length ($b^* = 200$ pkts) than the proposed memoryless AQM as the propagation delay increases, while they have similar performances when the delay is small. Utilization of the memoryless AQM and the memory AQM are 83 % and 91 %, respectively. Thus, the proposed memory AQM has a larger throughput than the proposed memoryless AQM. We have the same results for other large propagation delays.

IV. CONCLUSION

In this paper, we studied the design of AQM with a low-pass filter (average queuing) for stabilizing the given TCP. In this study, we showed what is a state feedback control structure for the given TCP model, how to compensate for delays explicitly,
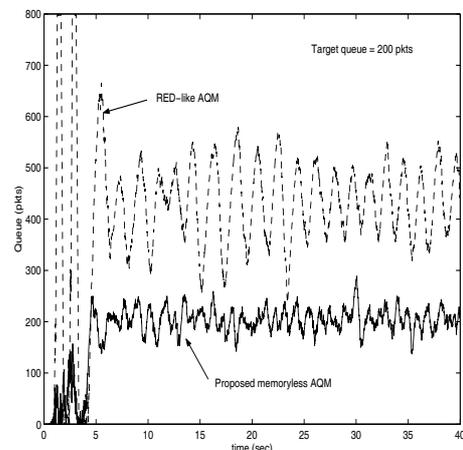

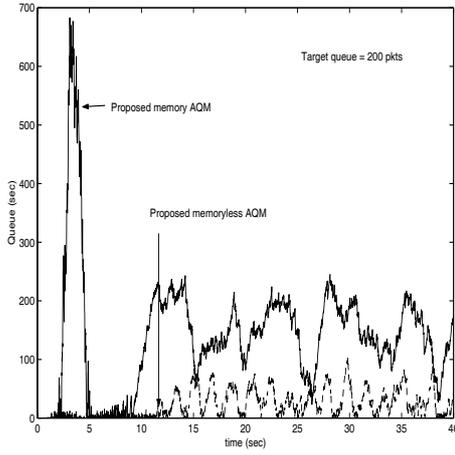Fig. 1. Queue $b(t)$ trajectories

Fig. 2. Queue $b(t)$ trajectories

and one way to get a stabilizing gain of the derived control structure. Finally, we illustrated the proposed results via *ns* simulations. In simulations, we used only one parameter (λ) for designing the proposed AQMs.

Our result based on the AIMD model in [13] shows that P-type memoryless (or delay-independent) AQM like RED is not enough to stabilize the AIMD dynamic behavior of TCP Reno. We expect that this work will be helpful to design a new AQM control structure which fully supports any kind of TCP in the future.

REFERENCES

[1] K. B. Kim, A. Tang, and S. H. Low, "Design of AQM in supporting TCP based on the well-known AIMD model." Caltech Technical Report caltechCSTR:2003.001, To appear in Proc. of IEEE Globecom'03, 2002. http://cisl.snu.ac.kr/~kkb,http://caltechcstr.library.caltech.edu/.

[2] V. Jacobson, "Congestion avoidance and control," in *Proceedings of ACM/SIGCOMM*, 1988. ftp://ftp.ee.lbl.gov/papers/congavoid.ps.Z.

[3] F. Baccelli and D. Hong, "AIMD, fairness and fractal scaling of TCP traffic," tech. rep., INRIA, Paris, France, 2001. RR 4155.

[4] F. Baccelli and D. Hong, "Interaction of TCP flows as billiards," tech. rep., INRIA, Paris, France, 2002. RR 4437.

[5] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, 1993. ftp://ftp.ee.lbl.gov/papers/early.ps.gz.

[6] S. Floyd, "TCP and explicit congestion notification," *ACM Computer Communications Review*, vol. 24, pp. 10–23, 1994. ftp://ftp.ee.lbl.gov/papers/tcp_ecn_4.ps.gz.

[7] W. Feng, D. Kandlur, D. Saha, and K. Shin, "A self–configuring RED gateway," in *Proceedings of IEEE/INFOCOM*, March 1999. http://www.eecs.umich.edu/~wuchang/work/infocom99.ps.Z.

[8] T. J. Ott, T. V. Lakshman, and L. Wong, "SRED: Stabilized RED," in *Proceedings of IEEE Infocom*, March 1999. ftp://ftp.bellcore.com/pub/tjo/SRED.ps.

[9] M. Christiansen, K. Jeffay, D. Ott, and F. D. Smith, "Tuning RED for web traffic," in *Proceedings of ACM/SIGCOMM*, 2000.

[10] W. Feng, K. G. Shin, D. Kandlur, and D. Saha, "Stochastic Fair Blue: A queue management algorithm for enforcing fairness," in *Proceedings of INFOCOM*, April 2001.

[11] S. Kunniyur and R. Srikant, "End–to–end congestion control schemes: utility functions, random losses and ECN marks," in *Proceedings of IEEE/INFOCOM*, 2000. http://www.ieee-infocom.org/2000/papers/401.ps.

[12] S. Athuraliya, V. H. Li, S. H. Low, and Q. Yin, "REM: active queue management," *Submitted for publication*, 2000. http://netlab.caltech.edu/pub.html.

[13] V. Misra, W.-B. Gong, and D. Towsley, "Fluid-based analysis of a network of AQM routers supporting tcp flows with an application to RED," in *Proceedings of ACM/SIGCOMM*, September 2000.

[14] C. Hollot, V. Misra, D. Towsley, and W.-B. Gong, "A control theoretic analysis of RED," in *Proceedings of IEEE/INFOCOM*, April 2001. http://www-net.cs.umass.edu/papers/papers.html.

[15] C. Hollot, V. Misra, D. Towsley, and W. B. Gong, "On designing improved controllers for AQM routers supporting TCP flows," in *Proceedings of IEEE/INFOCOM*, 2001. http://www-net.cs.umass.edu/networks/publications.html.

[16] S. Kunniyur and R. Srikant, "Analysis and design of an adaptive virtual queue (AVQ) algorithm for active queue management," in *Proceedings of ACM/SIGCOMM*, 2001. http://comm.csl.uiuc.edu/~srikant/pub.html.

[17] H. Lim, K. J. Park, E. C. Park, and C. H. Choi, "Active queue management algorithm with a rate regulator," in *Proc. of 15st IFAC World Congress on Automatic Control*, vol. T-We-M02, (Barcelona, Spain), 2002.

[18] E. C. Park, H. Lim, K. J. Park, and C. H. Choi, "Anaysis of the virtual rate control algorithm in TCP networks," in *Globecom*, vol. Nov, (Taipei, Taiwan), 2002.

[19] J. H. Lee, S. W. Kim, and W. H. Kwon, "Memoryless H_∞ controllers for state delayed systems," *IEEE Trans. on Autom. Contr.*, vol. 39, pp. 159 – 162, 1994.

[20] P. Park, "A delay-dependent stability criterion for systems with uncertain time-invariant delays," *IEEE Trans. Automat. Contr.*, vol. 44, no. 4, pp. 876–877, 1999.

[21] K. B. Kim and S. H. Low, "Analysis and design of AQM for stabilizing TCP." Caltech Technical Report caltechCSTR:2002.009, March 2002. http://cisl.snu.ac.kr/~kkb,http://caltechcstr.library.caltech.edu/,http://netlab.caltech.edu.

[22] K. B. Kim and S. H. Low, "Design of receding horizon AQM in stabilizing TCP with multiple links and heterogeneous delays," in *Proc. of 4th Asian Control Conference*, vol. WA-1, (Singapore), 2002. http://cisl.snu.ac.kr/~kkb.

[23] K. B. Kim, A. Tang, and S. H. Low, "A stabilizing AQM based on virtual queue dynamics in supporting TCP with arbitrary delays," in *Proc. of IEEE Conference on Decision and Control*, vol. Dec., (Maui, Hawaii), 2003.

[24] G. F. Franklin, J. D. Powell, and A. E. Naeini, *Feedback Control of Dynamic Systems*. Addison-Wesley, 1994.

[25] D. E. Kirk, *Optimal Control Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1970.

APPENDIX

We note that system (5) can be written as $z(t + \tau) = e^{A\tau}[z(t) + \int_{-\tau}^{0} e^{-A(\sigma+\tau)} B\delta p(\sigma + t)d\sigma]$. Define $h(t) = z(t) + \int_{-\tau}^{0} e^{-A(\sigma+\tau)} B\delta p(t + \sigma)d\sigma = z(t) + \begin{bmatrix} u_{1\tau}(t) \\ u_{2\tau}(t) \\ u_{3\tau}(t) \end{bmatrix}$. Then, system (5) can be rewritten as

$$\dot{h}(t) = Ah(t) + \tilde{B}\delta p(t), \qquad (A.9)$$

where $\tilde{B} = [\tilde{B}_1^T, \tilde{B}_2^T, \tilde{B}_3^T]^T = e^{-A\tau}B$.

Let $s(t) = Th(t)$, where $T = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_3 - T_2\hat{A}_1 & T_1 & T_2 \\ T_2\hat{A}_3 & T_3 & T_4 \end{bmatrix}$ with $T_{11} = T_1 - (T_3 - T_2\hat{A}_1)\hat{A}_3^{-1}\hat{A}_1$, $T_{12} = T_2 - (T_3 - T_2\hat{A}_1)\hat{A}_3^{-1}\hat{A}_2$ and $T_{13} = (T_3 - T_2\hat{A}_1)\hat{A}_3^{-1}$. T_2 should be selected so that $\det(T) \neq 0$ (T_2 cannot be zero). Then system (A.9) can be rewritten as $\dot{s}(t) = TAT^{-1}s(t) + T\tilde{B}\delta p(t) = As(t) + \begin{bmatrix} 0 \\ 0 \\ \hat{B}_1 \end{bmatrix} \delta p(t)$, where $\hat{B}_1 = T_2\hat{A}_3\tilde{B}_1 + T_3\tilde{B}_2 + T_4\tilde{B}_3$.