Active Queue Management (AQM) for TCP/IP Networks Using Discrete Time Sliding Mode Control (DSMC)

Saeedi G. Sajad, Beheshti H. Mohamad, Monemi Mahdi,

***Abstract*— In order to preserve Quality of Service (QoS) in a computer network, implementation of Traffic Engineering (TE) techniques is critical to meet congestion control.**
In this paper a systematic control strategy based on discrete time sliding mode control considering the dynamics of the window based TCP flow is proposed. Robustness of the designed controller against disturbance and delay variations has been shown via formulas and simulation results.

I. Introduction

Generally there are two approaches for congestion control in a communication network that depends on the type of the network. First which is called rate based congestion control is mostly used in ATM and MPLS networks. In this method the bottleneck switches directly determine transmitting rates of the senders. Second one known as window based congestion control is often used in TCP/IP networks. In this method as congestion increases, Active Queue Management (AQM) algorithms try to step down senders' window size. Random Early Detection (RED) is one of the most famous AQM algorithms that use mark/dropping of packets. TCP Reno a version of TCP mechanism uses packet loss as feedback information because packet loss implies congestion in the network [1], [4]. Explicit Congestion Notification (ECN), a well-known IETF mechanism, notifies source hosts of congestion occurrence in the network explicitly [6]. There are different ways for its implementation in TCP/IP networks.
In this paper window based control has been used for congestion control using discrete time sliding mode control specialized for a special model of TCP/IP networks.

II. TCP Flow Model

Fig. 1. shows an analytic model of a simple TCP/IP network. The number N of source hosts is connected to corresponding destination hosts through bottleneck routers. The window based flow control mechanism changes its window size once every Round Trip Time (RTT), therefore the system has been considered a discrete time system with a sampling time that corresponds to the round trip time. Since RTT varies according to the network status, the length of sampling slots is not fixed.

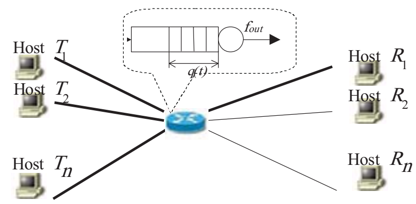

Fig. 1. Simple TCP/IP Network Topology

Suppose $w_n(k)$ be the window size of the source host n (Host $T_n, 1 \le n \le N$) at time slot k, that is the source host n can inject $w_n(k)$ packets into the network during k'th time slot. assume that each source host always has packets to transmit so that the number $w_n(k)$ of packets are sent at slot k. let $q(k)$ be the number of packets queued in the router's buffer at slot k. Bandwidth of the router is denoted by b. The RTT (i.e., sum of the source-destination delay and the destination-source delay) is denoted by τ which includes all propagation delays and processing delays but not queuing time at the router.
During an RTT, source host is allowed to consume the bandwidth being worth of its given window size. Considering that RTT of all connections are equal the number of packets in the buffer at slot $k+1$, $q(k+1)$ is:

$$q(k+1) = \max\left(\sum_{n=1}^{N} w_n(k) - br(k), 0\right) \tag{1}$$

Where $r(k)$ denotes the RTT at slot k and is given by:

$$r(k) = \tau + \frac{q(k)}{b}$$

The source host changes its window size based on the discrete time sliding mode control algorithm, performing active queue management. In another word $q(k)$ should track a predefined $q^{ref}(k)$ and according to the tracking error, control signal, $u(k)$, should be specified in the following equation:

$$w_n(k+1) = \alpha w_n(k) + u(k) \tag{2}$$

where α is a parameter denotes the increasing step of the window size [2], [5]. Following state

Manuscript received February 15, 2005. This work was supported by Iran Telecommunication Research Center (ITRC).
Saeedi G. Sajad is with Tarbiat Modares University, Tehran, Iran (phone: +98-912-249-3299; e-mail: sajadsaeedi@yahoo.com).
Beheshti H. Mohammad is with Tarbiat Modares University, Tehran, Iran. He is also with Iran Telecommunication Research Center (ITRC) (e-mail: mbehesht@modares.ac.ir).
Monemi mahdi is with Tarbiat Modares University, Tehran, Iran (e-mail: monemi_mahdi @yahoo.com).

equation can be concluded from the equations (1) and (2):

$$\begin{cases} \begin{bmatrix} w_n(k+1) \\ q(k+1) \end{bmatrix} = \begin{bmatrix} \alpha & 0 \\ N & -1 \end{bmatrix} \begin{bmatrix} w_n(k) \\ q(k) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k + \begin{bmatrix} 0 \\ -b\tau \end{bmatrix} \\ s(k) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} w_n(k) \\ q(k) \end{bmatrix} \end{cases} \quad (3)$$

The above equations can be abbreviated as this:

$$\begin{cases} X_{k+1} = AX_k + Bu_k + E\tau_k \\ s_k = CX_k \end{cases} \quad (4)$$

Where

$X_k = \begin{bmatrix} w_n(k) \\ q(k) \end{bmatrix}$ and

$A = \begin{bmatrix} \alpha & 0 \\ N & -1 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $E = \begin{bmatrix} 0 \\ -b \end{bmatrix}$, $C = \begin{bmatrix} 0 & 1 \end{bmatrix}$

and E is considered as input disturbance matrix. The aim is determining u_k in such a way that s_k should track s_k^{ref} in the presence of disturbance according to the Fig. 2.:

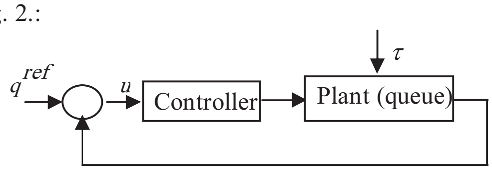

Fig. 2. System Block Diagram

III. Control Design

A. Discrete Time Sliding Mode Control (DSMC)

Sliding mode control is a very powerful tool for control design. Most sliding mode approaches are based on finite-dimensional continuous-time models and lead to discontinuous control action. Generally for a continuous-time system, in the sliding phase by discontinuous control switching at theoretically infinite frequency, system order will reduce, but in practice because of limitations in switching frequency, we have to use a special approach consisting with these limitations. However a discontinuous control designed for a continuous time system model would lead to chatter when implemented without modifications in discrete time model considering finite sampling rate.

Suppose the following system where $u(k)$ is the control signal:

$x_{k+1} = F(x_k, u_k)$, $x_k = x(kT)$, $k = 1, 2, 3, \ldots$

and $s(k)$ as switching surface:

$s_k = CF(x_k)$

where C is a designing matrix.

The discrete time sliding mode control is defined as following:

At $t = 0$, $u(x(t = 0), T)$ should be designed in such a way that $s(x(t = T)) = 0$.

At $t = T$, $u(x(t = T), T)$ should be designed in such a way that $s(x(t = 2T)) = 0$.

Generally, at $t = kT$, $u(x(t = kT), kT)$ should be designed in such a way that $s(x(t = (k+1)T)) = 0$, $k = 1, 2, \ldots$.

In other words, at each sampling point k, $u(k)$ should be designed in such a way that $s(x(k+1)) = 0$ [3].

B. Control Algorithm

Consider the system (4), where E is a disturbance matrix and is unmatched (i.e., the disturbance does not enter the system through the matrix B). The sliding surface is defined as:

$$s(k+2) = q(k+2) - q^{ref}(k+2)$$
$$= C(X_{k+2} - X_{k+2}^{ref}) \quad , \quad C = [0 \quad 1]$$

Considering that $CB = 0$, it is concluded that:

$$s(k+2) = C(A^2 X_k + ABu_k + AE\tau_k + E\tau_{k+1} - X_{k+2}^{ref})$$

Regarding to the mentioned control algorithm, control signal is extracted as:

$$u_{keq} = -(CAB)^{-1}(CA^2 X_k + CAE\hat{\tau}_k + CE\hat{\tau}_{k+1} - CX_{k+2}^{ref})$$

Where $\hat{\tau}$ is an estimation of τ and CAB should have inverse form. From above equations tracking error can be calculated easily as:

$$Error(k+2) = B((\tau(k) - \hat{\tau}(k)) - (\tau(k+1) - \hat{\tau}(k+1)))$$

It means that error decreasing is related to RTT estimation precision.

In real implementation there are limitations on the control signal like the following:

$$u_k = \begin{cases} u_{keq} & |u_{keq}| < u_0 \\ u_0 \dfrac{u_{keq}}{|u_{keq}|} & |u_{keq}| > u_0 \end{cases}$$

Considering this limitation there is a question that "In which situation, for each initial value, the tracking error will decrease?". This question will be studied for transient response because the steady state error will be in a specified range.

It can be showed that:

$$s_{K+2} = s_{K+1} - C(AX_k + Bu_k + E\tau_k - X_{k+1}^{ref})$$
$$+ C(A^2 X_k + ABu_k + AE\tau_k + E\tau_{k+1} - X_{k+2}^{ref})$$

By subtending u_{keq} in the above equation and considering that $CB = 0$ and after some simplifications, it can be concluded that:

$$s_{K+2} = s_{K+1} + C(-AX_k - E\tau_k + X_{k+1}^{ref}) +$$
$$(1 - \frac{u_0}{|u_{keq}|})(C(A^2 X_k + AE\hat{\tau}_k + E\hat{\tau}_{k+1} - X_{k+2}^{ref})) +$$
$$C((AE\tau_k + E\tau_{k+1}) - (AE\hat{\tau}_k + E\hat{\tau}_{k+1}))$$

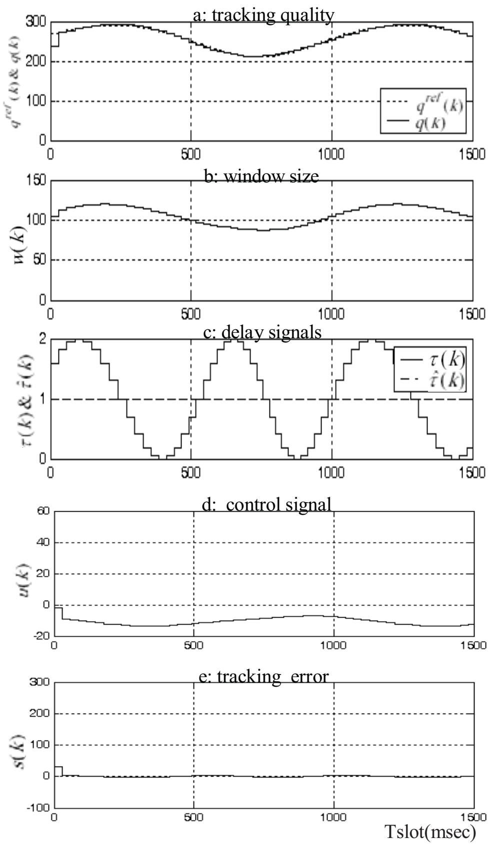

Fig. 3. simulation results for ${}_0 = 60$

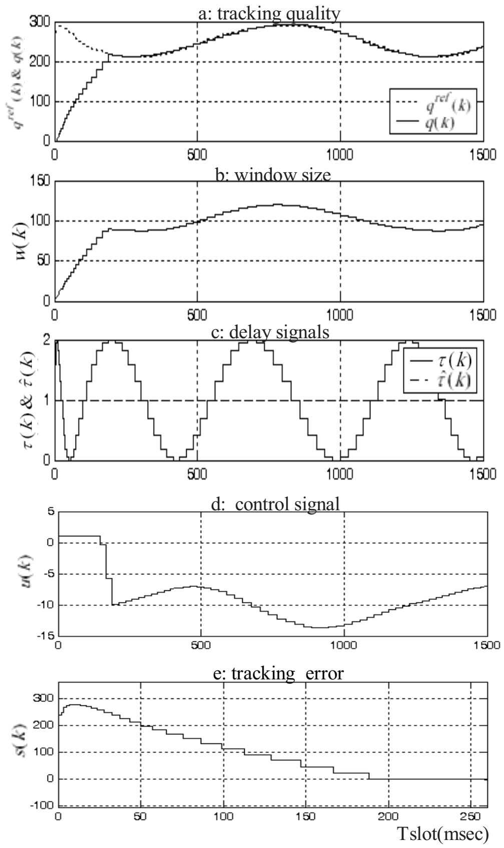

Fig. 4. simulation results for $u_0 = 1$

Thus:

$$\left|s_{K+2}\right| < \left|s_{K+1}\right| + \left|C(-AX_k - E\tau_k + X^{ref}_{k+1})\right| + (1 - \frac{u_0}{\left|u_{keq}\right|})\left|C(A^2 X_k + AE\tau_k + E\tau_{k+1} - X^{ref}_{k+2})\right| + \left|C((AE\tau_k + E\tau_{k+1}) - (AE\hat{\tau}_k + E\hat{\tau}_{k+1}))\right|$$

Considering u_{keq} we have:

$$\left|s_{K+2}\right| < \left|s_{K+1}\right| + \left|C(AX_k - X^{ref}_{k+1})\right| + \left|C(A^2 X_k - X^{ref}_{k+2})\right| + \left|CE\tau_k\right| + \left|C(AE\tau_k + E\tau_{k+1})\right| + \left|\begin{matrix} C((AE\tau_k + E\tau_{k+1}) \\ -(AE\hat{\tau}_k + E\hat{\tau}_{k+1})) \end{matrix}\right| - \frac{u_0}{\left|(CAB)^{-1}\right|}$$

Hence in order to decrease tracking error monotonically, following limitation should be satisfied:

$$u_0 > \left|(CAB)^{-1}\right| \left(\left\{ \begin{matrix} \left|C(AX_k - X^{ref}_{k+1})\right| + \\ \left|C(A^2 X_k - X^{ref}_{k+2})\right| \end{matrix} \right\} + \left\{ \begin{matrix} \left|CE\tau_k\right| + \left|C(AE\tau_k + E\tau_{k+1})\right| \\ + \left|\begin{matrix} C((AE\tau_k + E\tau_{k+1}) - \\ (AE\hat{\tau}_k + E\hat{\tau}_{k+1})) \end{matrix}\right| \end{matrix} \right\} \right)$$

In the discussed TCP/IP congestion control case, it can easily be concluded that the mentioned limitation should have the following specification:

$$u_0 > \frac{1}{N} \left(\left\{ \begin{matrix} \left|-Nw_k + q^{ref}_{k+1}\right| + \\ \left|N(\alpha - 1)w_k + q_k - q^{ref}_{k+2}\right| \end{matrix} \right\} + \sup \left\{ \begin{matrix} \left|b\tau_k\right| + \left|b(\tau_k - \tau_{k+1})\right| + \\ \left|b((\tau_k - \hat{\tau}_k) - (\tau_{k+1} - \hat{\tau}_{k+1}))\right| \end{matrix} \right\} \right)$$

If we can hold the above equation, we can guarantee error decreasing.

IV. SIMULATION

In order to simulate the proposed controller, a network topology like fig. 1. has been considered, in which N right handed computers want to communicate their data with N

left handed computers and it is supposed that computers always have data to send. For simplification of simulation, the initial window sizes of all source hosts are assumed identical. The parameters introduced at (3) are:

$b = 10$ [$packet/m\sec$] , $N = 5$

$\alpha = 1.3$, $u_0 = 60$

τ as the real propagation considered a sinusoidal signal and our estimation for it is a constant signal as $\hat{\tau} = 1 m\sec$, with the estimation error of $\pm 1 m\sec$, as it is shown in Fig. 3.c. Using trial and error, u_0 has been selected such that it could satisfy the limitation condition. q^{ref} as a reference queue length and q as an indicator of the current queue length have been showed in Fig. 3.a. Window size, $w(k)$, has been showed in Fig. 3.b. According to Fig. 3.a tracking has good performance. Finally control and tracking error signals, $u(k)$ and $e(k)$, have been showed in Fig. 3.d, Fig. 3.e. In the Fig. 4., the simulation has been repeated considering $u_0 = 1$ that can not satisfy u_0 limitation condition. In this case the error signal does not decrease in the transient state and also the reaching time will increase in comparison to the previous state.

Although there was such a big changes in delay variation but control signal was designed so that system could keep its performance against delay that was considered and simulated as a system disturbance

In order to qualify DSMC approach, integral control design, a powerful tool for disturbance rejection has been designed. This kind of controller is just a kind of error accumulator and then control signal is designed according to this error signal. Integral control principles can be found in every basic control book. Briefly, augmented system using error accumulator should be structured then using simple linear design approaches such as Ackerman feedback design approach, a feedback controller should be designed to restrict the error accumulator to a fixed value or to converge the tracking error to zero [7].

Fig. 5. includes simulation results for integral control design. Also these two approaches were compared from numerical view point. The squared sum of errors for DSMC is 5 but for integral control is 21, so the error in DSMC approach is lower than integral control approach.

V. Conclusion

This paper was focused on a window based flow control mechanism, using discrete time sliding mode control (DSMC) for TCP/IP networks. A condition for tracking error convergence to zero has been extracted and this condition qualified using simulation results.

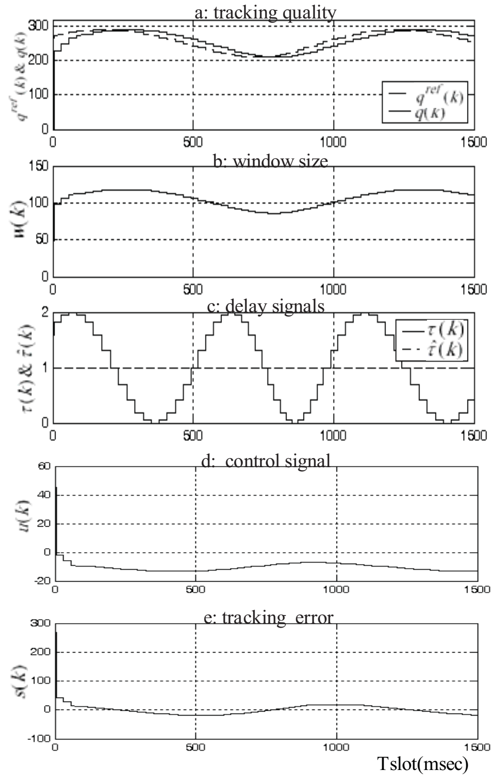

Fig. 5. simulation results for integral control

This paper can be extended to more generic network topologies. Here it is assumed that all connections have identical propagation delays, and there is only a single bottleneck router in the network. In real TCP/IP networks, propagation delays of all connections are not identical and the bottleneck router may change its position dynamically as time changes. It is very important to examine this algorithm in such generic network configuration.

Acknowledgment

This work has been supported by Iran Telecommunication Research Center (ITRC) under grant No. T/500/4118, Ministry of Communication and Information Technology, Tehran Iran.

References

[1] H. Beheshti, M. Monemi, S. Saeedi, "Robust Congestion Control Design for AQM in Differentiated Services Communication Networks", Int'l Symposium on telecommunication (IST2003), 2003

[2] H. Ohsaki, M. Murata, H. Miyahara, "Application of Control Theory to a Window-Based Flow Control Mechanism with ECN Routers"

[3] V. Utkin, J. Guldner, J. Shi, Sliding Mode Control in Electromechanical Systems, 1999, Taylor & Francis Inc.

[4] V. Jacobson, "Congestion Avoidance and Control", SIGCOMM'88, Aug 1988.

[5] H. Sirisena, M. Hassan, A. Haider, "Optimal TCP Congestion Control"

[6] K. Ramakrishnan, "A Proposal to add Explicit Congestion Notification (ECN) to IP" RFC 2481, Jan. 1999.

[7] G. Franklin, et. Al., Digital control of Dynamic Systems, 3rd Ed. 1998, Addision-Wesley