2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Switching congestion control for satellite TCP/AQM networks

Jinhua Cao and Margareta Stefanovic

***Abstract*— In this paper, a congestion controller using data-driven switching control theory is proposed to improve the dynamic performance of the satellite TCP/AQM networks. A PID controller, whose parameters are adaptively tuned by switching among members of a given candidate set using observed plant data, is presented and compared with a classical Random Early Detection AQM policy example. A cost-detectable cost function used by the switching law is considered and compared with a cost function more commonly used in the literature. Computational cost savings are considered. Simulations are presented to validate the theory.**

***Index Terms*— Control in communication networks, switching control, PID control.**

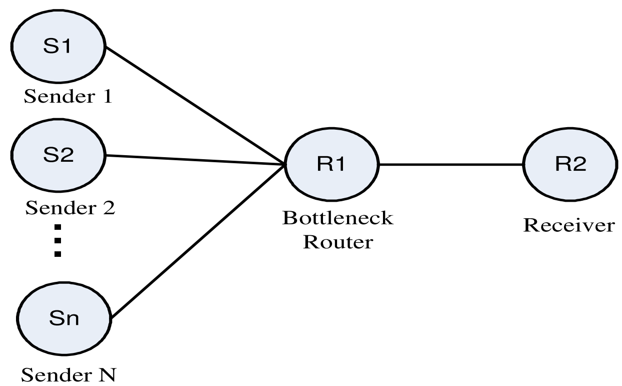

Fig. 1. Network topology

I. Introduction

Satellite networks play an important role in broadcasting data over large geographic locations, proving to be an essential means for reaching remote locations lacking in communication infrastructure. Many military and civilian applications have experienced significant benefit from the use of satellite networks. For example, national defense depends on satellite communications for robust, rapidly deployable and secure communications in adverse environments. Among civilian applications, supplying rural locations with high data rate, high fidelity communication services is currently feasible only through affordable satellite communications, due to the lack of availability in fiber networks.

One of the major communication problems hampering the use of the satellite networks is congestion. Transmission Control Protocol (TCP) congestion control algorithms use packet-loss and packet-delay measurements to detect congestion [1]. Recently, Active Queue Management (AQM) has been proposed to support the end-to-end congestion control in the Internet, by sensing impending congestion before it occurs and providing feedback information to senders by either dropping or marking packets, so that congestion that causes a significant degradation in network performance can be avoided [2]. In control theory, AQM can be considered as a nonlinear feedback control system with a delay.

The study of congestion problem within time-delay system framework has been successfully exploited using control theory. In [3], [4], [2], dynamical models of the average TCP window size and the queue size in the bottleneck router are derived and linearized about an equilibrium point, and the proportional-integral (PI) congestion controllers are designed. In [5], a delay dependent state feedback controller is proposed to compensate for the delay by using a memory feedback control. The latter methodology is an interesting theoretical contribution, but it has a dubious applicability on the more realistic scenarios (*e.g.*, heterogeneous delays). In [6], robust AQM algorithms are derived using a time delay system approach. However, only a single plant model was considered in the above cited work, which means that the designed controller is effective only in the vicinity of the nominal point.

In this paper, we focus on a new AQM algorithm for the *long fat* networks, in particular multi-layer satellite networks, using unfalsified control theory. From analysis, it can be shown that this algorithm can adaptively deal with heterogeneous delays. The main goal is to illustrate the potential impact of the Safe Switching Control methodology [7] on the congestion control problem of the TCP/AQM networks with dynamically varying parameters and time delays, which is the case of the long fat satellite networks.

The rest of the paper is organized as follows. In Section 2, a nonlinear network model is presented and linearized about nominal points. The *de facto* AQM standard, Random Early Detection (RED) [8], is discussed. In Section 3, the unfalsified safe switching data congestion control is presented. In Section 4, simulations are conducted, compared and discussed. The paper concludes with some remarks in Section 5.

II. Problem Statement

To design controllers for the network congestion problem, large scale networks are often simplified according to a well-known dumbbell topology, shown in Fig. 1. The network consists of N senders, a bottleneck router and a receiver,

This work was supported in part by the Department of Defense under contract DODAF41612.

The authors are with Dept. of Electrical and Computer Engineering, University of Wyoming, Laramie, WY, USA, 82071 jcao@uwyo.edu, mstefano@uwyo.edu

Corresponding author: M. Stefanovic.

978-1-4244-4524-0/09/$25.00 ©2009 AACC

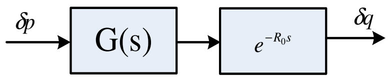

Fig. 2. Linearized system with delay

which in a cluster form of satellite networks correspond to the satellite nodes, cluster-head satellite of one cluster and a cluster-head of another communicating cluster, respectively. Cluster-heads of various clusters may be significantly far from each other (when the two clusters exchanging data belong to two distinct orbits, *e.g.*, LEO and MEO). In this topology, we assume that N TCP connections represent a homogeneous and long lived flow. In [9], a fluid-flow model describing the behavior of the average values of the window size and the queue size in the bottleneck router (Fig. 1) is given by the following pair of coupled nonlinear differential equations:

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

where $R(t) = \frac{q(t)}{C} + T_p$ is the Round-Trip Time (RTT), W is the TCP window size, p is the probability of dropping a packet, N is the number of connections or TCP sessions, C is the transmission capacity of the router, q is the queue length of the router buffer, and T_p is the propagation delay. To allow the use of the traditional control theory, a small-signal linearization is carried out about an equilibrium point (W_0, q_0, p_0), resulting in a nominal LTI plant model as follows:

$$G(s) = \frac{\frac{C^2}{2N_0}}{(s + \frac{2N_0}{R_0^2 C})(s + \frac{1}{R_0})}$$

where N_0 and R_0 are constants in the linearized model. The linearized system is shown in Fig. 2, together with the delay in the multiplicative decrease in the window size as a response to packet marking/dropping, where δp and δq represent perturbed variables about the equilibrium point. This nominal model relates how the packet marking probability dynamically affects the queue length. We first consider Random Early Detection (RED) scheme [8], a standard AQM policy example, and we compare it with our proposed scheme.

Typically, the router congestion control schemes can be classified as implicit and explicit feedback schemes. The main difference in these approaches is that the implicit schemes trigger packet dropping, while explicit schemes are notifications generated by the routers to the end hosts to "slow down".

In implicit feedback schemes, the source detects the existence of congestion by making local observations which include timeouts when acknowledgements are not received, acknowledgement reception with more than normal delay, duplicate acknowledgement reception, *etc*. The main underlying causes for all of the above observations are due to the packet delay and packet losses. Hence, in implicit schemes the routers simply drop packets during congestion and expect the source to respond to these lost packets by slowing their transmission rate. The design choices with the implicit schemes are the position in the queue from which the packet needs to be dropped and when a packet should be dropped. In this paper, the design choice concerning when a packet should be dropped is addressed, specifically the RED scheme belonging to the proactive dropping category also known as AQM.

The basic idea behind RED is to detect developing congestion early and convey congestion notification to the end-hosts, thereby allowing them to throttle their transmission rates before queues in the network overflow and packets are dropped. A router implementing RED continuously monitors the average queue length and when it exceeds a threshold, it randomly drops arriving packets with a certain probability, even though the buffer is not full. The dropping of packets serves as an early notification to the source to reduce its transmission rates. The RED algorithm uses the exponential weighted moving average (low pass filter) approach to calculate the average queue length.

A low-pass filter is used to calculate the average queue size, in order to play down the impact of the bursty traffic or transient congestion on the average queue size. Thus, the low-pass filter is designed as an exponentially weighted moving average (EWMA):

$$avg_q \leftarrow (1 - W_q)avg_q + W_q q$$

where W_q is the time constant for the low-pass filter. The packet marking (dropping) profile is a nonlinear gain:

$$p(avg_q) = \left\{ \begin{array}{cc} 0, & avg_q < min_{th} \\ \frac{avg_q - \min_{th}}{\max_{th} - \min_{th}} p_{\max}, & \min_{th} \leq avg_q < \max_{th} \\ 1, & \max_{th} \leq avg_q \end{array} \right\}$$

where $\min_{th}$, $\max_{th}$, and $p_{\max}$ are fixed parameters, determined in advance by considering the desired bounds on the average queue size.

In satellite networks, in particular multi-layer ones, data communication delays can be significantly large, making the classical RED method insufficiently effective. Also, the RTT between LEO and GEO (Geosynchronous Orbit) satellites is much larger than the delay among LEO satellites. Hence, traditional RED controller designed based on linearization about equilibrium points might not be effective in the multi-layer satellite networks, since both N and R are dynamically varying. Significant variations in the packet size and session duration afflicting streaming video traffic is another potential source of pitfalls of the traditional RED AQM controllers.

III. SAFE SWITCHING CONTROL

The Safe Switching Control system (SSC) based on the unfalsified control theory, used for the design of congestion control scheme, is described below and shown in Fig. 3. The

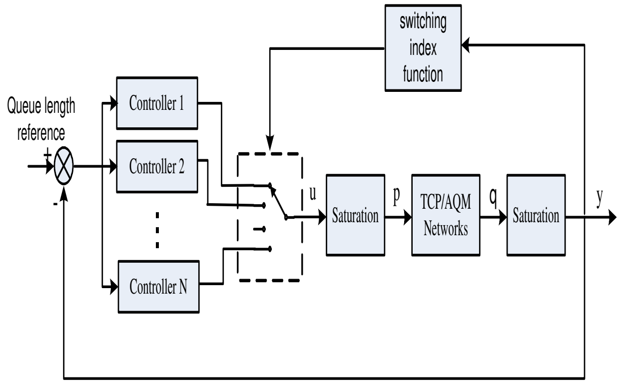

Fig. 3. Unfalsified congestion control structure

set of candidate controllers is taken to be of the proportional-integral-derivative (PID) structure, one of the most widely used methods in the industry. The theoretical framework for stability of the unfalsified SSC theory can be found in [10] and [7], while some of the extensions to the performance improvement are described in [11], [12]. To apply the SSC approach, we first construct a bank of candidate controllers (chosen here to be of the PID structure). The fictitious reference and error signals for individual PID controllers can be generated on-line. Given the fictitious reference signal, plant input signal, and plant output signal sets, the "best" (optimal) controller is selected from the candidate set using a properly designed cost function. The candidate controller parameters can be designed off-line using Integral Squared Error (ISE) optimization algorithms, by considering the linearized plant models [13].

Candidate controller construction: A candidate PID controller is to be chosen so as to minimize the ISE performance index corresponding to the linearized model [13]:

$$I = \int_0^\infty e^2(t)dt$$

where $e(t)$ is the queue length error signal. For the delay part, a 2^{nd} order direct frequency response series is applied as [13]:

$$e^{-s\tau} \approx \frac{1 - 0.49s\tau + 0.0954s^2\tau^2}{1 + 0.49s\tau + 0.0954s^2\tau^2}$$

Since the control performance optimization is non-convex, a local minimum might occur. To deal with this, the stability margin based on Ziegler-Nichols rules can be used for initial controller parameters [13]. The controllers' performance is evaluated before they are switched into the candidate controller set.

Fictitious reference signal: The ideal PID controller has an improper transfer function, hence for implementation we write it as:

$$u = (k_p + \frac{k_I}{s})(r - y) - \frac{sk_D}{\varepsilon s + 1}y$$

where r, u and y represent the queue length reference, dropping packet probability, and queue length. The parameter $\varepsilon > 0$ is added to approximate the derivative part. Then, the fictitious reference signal (FRS) for the candidate controller i can be calculated as:

$$\tilde{r}_i = y + \frac{s}{sk_{P_i} + k_{I_i}}(u + \frac{sk_{D_i}}{\varepsilon s + 1}y)$$

The fictitious error signal for the candidate controller i (defined as the error between its FRS and the actual plant output), can be computed as:

$$\tilde{e}_i = \tilde{r}_i - y$$

Cost function: An important step is to design a suitable cost function to adjust controller parameters based on the measured data alone. A popular form of the cost function used in the switching control literature contains a form of the accumulated error, such as the cost considered in [14]:

$$J_i(t) = \tilde{e}_i^2(t) + \int_0^t e^{-\lambda(t-\tau)}\tilde{e}_i^2(\tau)d\tau \qquad (1)$$

In [7], it was shown that this cost function is not cost-detectable, and therefore, it may in some cases discard the stabilizing controller and latch onto a destabilizing one.

A. Switched System Stability

To briefly recall the results on stability in a multi-controller unfalsified setting, consider the system $\Sigma : \mathfrak{L}_{2e} \longrightarrow \mathfrak{L}_{2e}$. We say that stability of the system $\Sigma : \boldsymbol{w} \mapsto \boldsymbol{z}$ is said to be *unfalsified* by the data $(\boldsymbol{w}, \boldsymbol{z})$ if there exist $\beta, \alpha \geq 0$ such that the following holds:

$$||\boldsymbol{z}||_\tau < \beta||\boldsymbol{w}||_\tau + \alpha, \forall \tau > 0;$$

Otherwise, we say that stability of the system Σ is *falsified* by $(\boldsymbol{w}, \boldsymbol{z})$.

As a swithing rule, we consider the cost minimization ε-hysteresis switching algorithm [15] together with the cost functional $J(K, z, t)$. This algorithm returns, at each t, a controller $\hat{K}_t$ which is the active controller in the loop:

ε-hysteresis switching algorithm A1

$$\hat{K}_t = \arg\min_{K \in \mathbf{K}}\{J(K, z, t) - \varepsilon\delta_{K\hat{K}_{t^-}}\}$$

where δ_{ij} is the Kronecker's δ, and t^- is the limit of τ from below as $t \to \tau$.

The switch occurs only when the current unfalsified cost related to the currently active controller exceeds the minimum (over the finite set of candidate controllers $\mathbf{K}$) of the current unfalsified cost by at least ε. The hysteresis step ε serves to limit the number of switches on any finite time interval to a finite number, and so prevents the possibility of the limit cycle type of instability. It also ensures a non-zero dwell time between switches.

Definition 1: [7] Let r denote the input and $z_d = \Sigma(\hat{K}_t, \mathcal{P})r$ denote the resulting plant data collected while $\hat{K}_t$ is in the loop. Consider the adaptive control system $\Sigma(\hat{K}_t, \mathcal{P})$ with input r and output z_d. The pair $(J, \mathbf{K})$ is said to be *cost detectable* if, without any assumption on the plant P and for every $\hat{K}_t \in \mathbf{K}$ with finitely many switching times, the following statements are equivalent:

1) $J(K_N, z_d, t)$ is bounded as t increases to infinity.
2) Stability of the system $\Sigma(\hat{K}_t, \mathcal{P})$ is unfalsified by the input-output pair (r, z_d).

Theorem 1: [7] Consider the feedback adaptive control system Σ, together with the hysteresis switching algorithm A1. Suppose the following holds: the adaptive control problem is feasible (there is at least one stabilizing controller in the candidate set), the associated cost functional $J(K, z, t)$ is monotone in time, the pair $(J, \mathbf{K})$ is cost detectable, and the candidate controllers have stable causal left inverses. Then, the switched closed-loop system is stable. In addition, for each z, the system converges after finitely many switches to the controller K_N that satisfies the performance inequality

$$J(K_N, z, \tau) \leq J_{true}(K_{RSP}) + \epsilon \quad \text{for all } \tau.$$

B. Cost-detectable Cost Function

A cost function that satisfies the cost-detectability properties can be chosen as

$$J_i(t) = -\rho + \max_{\tau \in [0,t]} \frac{\varepsilon \left\| \Delta u \right\|_\tau^2 + \left\| \widetilde{e_i} \right\|_\tau^2}{\left\| \widetilde{r_i} \right\|_\tau^2 + \alpha} + \gamma \left\| K \right\|^2 \qquad (2)$$

where Δu is the deviation of the congestion window size. The weighting parameter ε is a positive constant. Constant α is used in order to prevent $J_i(t)$'s denominator to be zero when $\widetilde{r_i} = 0$, $\|K\|$ is the is the Euclidean norm of the controller parametrization, and γ scales the importance of $\|K\|^2$. Also, $\|\cdot\|_t$ stands for the truncated 2-norm $\|x\|_t = \sqrt{\int_0^t (x(\tau))^2 d\tau}$.

A more standard form of the cost function can be found in [12]:

$$J_i(t) = -\rho + \int_0^t \Gamma_{spec}(\widetilde{r_i}(t), y(t), u(t))dt. \qquad (3)$$

However, instead of the ratio between input and output signals, a threshold ρ is employed here against which the controllers are falsified using the error between input and output signals:

$$\Gamma_{spec}(\widetilde{r_i}(t), y(t), u(t)) =$$
$$(w_1 * (\widetilde{r_i}(t) - y(t))^2 + (w_2 * u(t))^2 - \delta^2 - \widetilde{r_i}(t)^2$$

where w_1 and w_2 are weighting filters chosen by the designer, and σ is a constant representing the r.m.s. effects of noise on the cost.

Restriction: The probability of dropping the packets and the queue length are bounded quantities:

$$p = min(max(u, 0), 1)$$
$$y = min(max(q, 0), B)$$

where B is the total buffer size.

Computational cost saving: In satellite networks, cost saving is a consideration of even greater importance than in Earth-based networks. To save computational cost, two steps are applied: first, minimizing the number of the candidate controllers by designing each controller according to the chosen model of the satellite data network, rather than randomly choosing off-the-shelf controllers. Second, shutting off the cost monitor during the 'idle' periods, *i.e.* once the optimal stabilizing controller has been switched on, we reactivate the cost monitor for other candidate controllers only if/when it happens that after some pre-specified time period the current controller fails to meet the performance criterion and is consequently falsified.

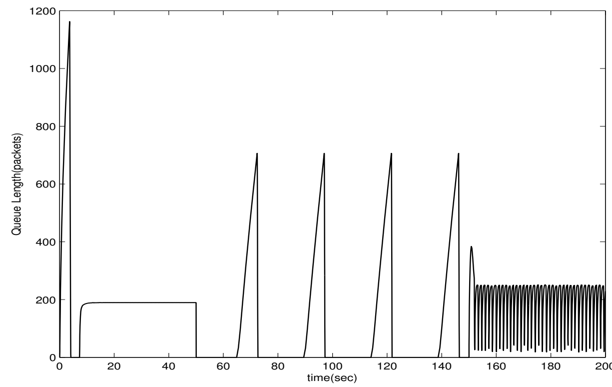

Fig. 4. Actual queue length with RED

The following steps summarize the safe switching unfalsified congestion control scheme, where the switch in step 4 occurs according to Algorithm A1:

1) Initialization: define a set of candidate controllers, and an initial controller in the loop at the beginning. Set initial cost function output to be 0. Initialize a timer $T = 0s$.
2) Measure Δu and y. Run timer.
3) Calculate $\widetilde{r_i}$ and $\widetilde{e_i}$, and $J_i(t)$.
4) Switch the controller $\arg \min_{1 \leq i \leq N} J_i(t)$ into the loop.
5) Measure y, and calculate $e = r - y$.
6) If $|e| > 5\, packets$, initialize the timer to 0, and go back to step 2.
7) If $|e| \leq 5\, packets$ and $T \leq 5$, go back to step 2.
8) If $|e| \leq 5\, packets$ and $T > 5$, stop the timer, initialize the cost function output to 0, and go back to step 5 (shut off the cost monitor).

IV. Simulation Results

We describe Matlab simulations of PID controller parameter adaptation based unfalsified congestion control designed above to demonstrate its effectiveness. Assume the transmission capacity of the router $C = 15Mbps$ (or 3750 $packets/s$ for an average packet size of 500 Bytes); the queue length reference $r = 200\,packets$. At the beginning, the number of connections is $N = 60$ with the propagation delay $T_p = 0.03\,s$; at $t = 50\,s$, T_p rises to $0.5\,s$ with N unchanged; at $t = 150\,s$, T_p falls back to 0.03 with N rising to 80. Based on the linearized models about the nominal points $N = 60, R = 0.08$ and $N = 60, R = 0.5$, the candidate controller set is designed off-line, using ISE optimization algorithm [13], to have the following parameters: For the first set, $K_P = 0.000951, K_I = 0.0032, K_D = 0.00004993$, for

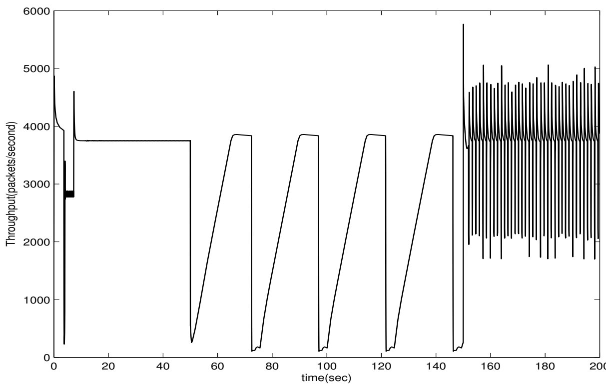

Fig. 5. Throughput with RED

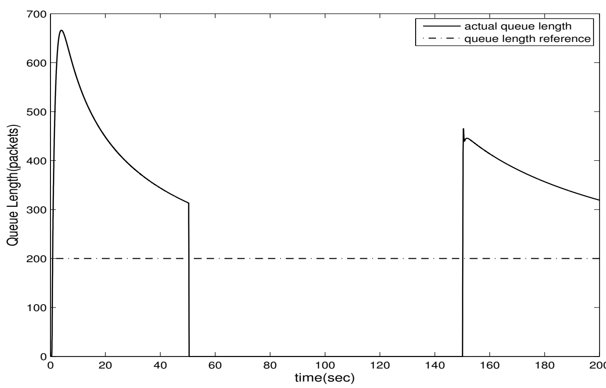

Fig. 6. Actual queue length with the cost function (3)

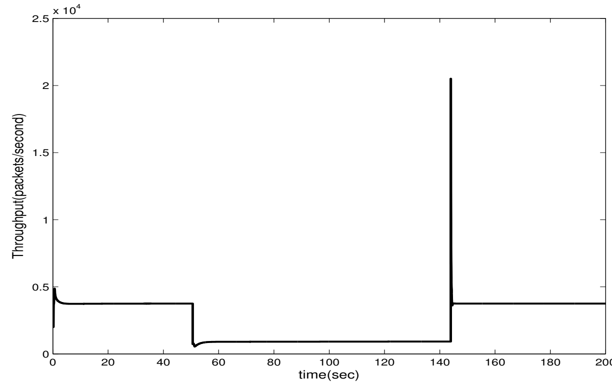

Fig. 7. Throughput with the cost function (3)

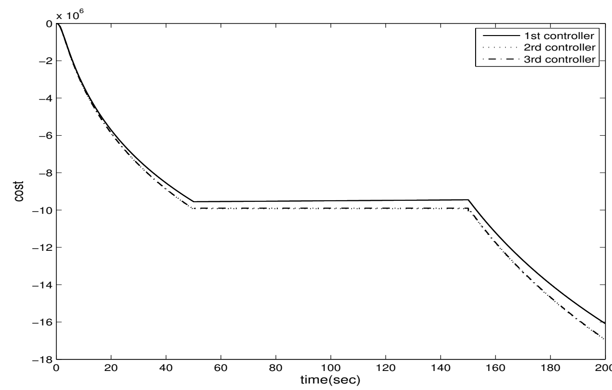

Fig. 8. Outputs of the cost function (3)

the second set $K_P = 0.000014, K_I = 1.6784 \cdot 10^{-6}, K_D = 0.00000639$. We also introduced a third controller $K_P = 4.8250 \cdot 10^{-4}, K_I = 0.0016, K_D = 2.8160 \cdot 10^{-5}$ as a linear interpolation of the 1^{st} and 2^{nd} controllers (the average of their parameters).

Simulation 1 (RED control scheme): The RED parameters are chosen as $W_q = 0.1$, $\min_{th} = 150$, $\max_{th} = 200$, $p_{\max} = 0.1$. The actual queue length is shown in Fig. 4. The throughput is shown in Fig. 5. In our simulation, the high frequency oscillation in low delay period can be damped, if we apply the piecewise function of p such as: $p(avg_q) =$

$$\begin{cases} 0, & avg_q < \min_{th} \\ \frac{avg_q - \min_{th}}{\max_{th} - \min_{th}} p_{\max}, & \min_{th} \leq avg_q < \max_{th} \\ \frac{avg_q - \max_{th}}{\max_{th}} (1 - p_{\max}) + p_{\max}, & \max_{th} \leq avg_q < 2\max_{th} \\ 1, & 2\max_{th} \leq avg_q \end{cases}$$

However, in large delay case, the RED scheme is not very effective.

Simulation 2 (SSC with the cost function (3)): Here we demonstrate the effectiveness of unfalsified control with the standard cost function (3). The actual queue length is shown in Fig. 6, and the throughput in Fig. 7. From Fig. 8, we see that all controllers are falsified.

Simulation 3 (SSC with the cost-detectable cost function (2)): Here we use the cost-detectable cost function (2). The actual queue length is shown in Fig. 9, the throughput is shown in Fig. 10. The cost output is shown in Fig. 11. The RTT is shown in Fig. 12. By comparing Figs. 6, 7 with Figs. 9, 10, we observe that using one fixed controller or using switching unfalsified control method with the standard cost function does not result in satisfactory tracking the queue length reference, and the bandwidth utility is low. Instead, the proposed SSC unfalsified controller can track the reference signal, stabilize the system, and achieve high bandwidth utility in different situations. Even when the number of connections rises by 20, and there is no controller in this nominal point in the candidate set, the proposed cost function can still choose the best controller (according to the algorithm A1), and track the reference signal, demonstrating its robustness. Fig. 9 in particular demonstrates the improvement achieved by using the cost-detectable switching controller. It shows that the queue length is kept low which is very good from the router perspective. Fig. 12 shows that the unfalsified controller can keep the RTT at a relatively low level. In Fig. 11, the time intervals when all the cost functions drop to 0 signify the case when the queue length error with the currently active controller has been lower than 5 packets for 5 s, and so the cost monitor has been shut off to save the computational cost. When the cost functions become nonzero again signifies the case when the current error is

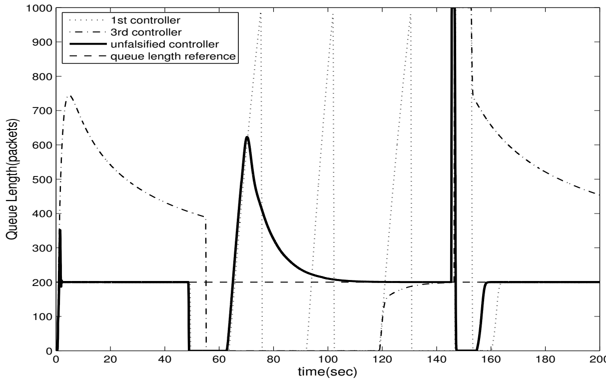

Fig. 9. Actual queue length with the cost-detectable cost function (2)

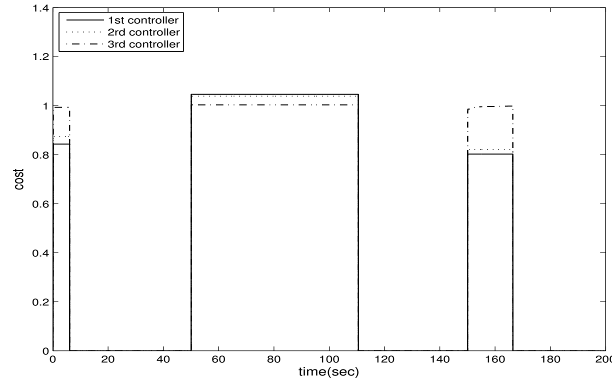

Fig. 11. Outputs of the cost-detectable cost function (2)

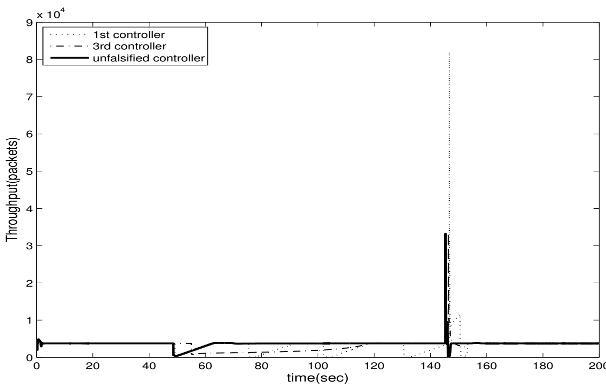

Fig. 10. Throughput (packets/s) with the cost-detectable cost function (2)

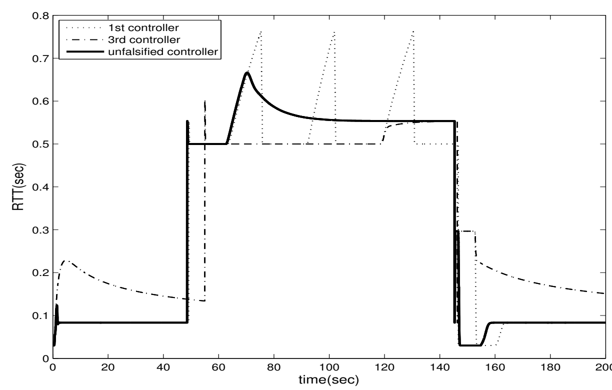

Fig. 12. Round-Trip Time

larger than 5 packets, and the current candidate controller might be falsified. Hence, the cost monitoring is reactivated to choose the optimal controller.

V. Conclusion

This paper presents a framework for the data congestion control system for satellite TCP/AQM networks using safe switching unfalsified control. In this approach, the optimal controller can be adaptively selected according to the varying network conditions; an accurate plant model is helpful for designing candidate controllers, but otherwise not required for guaranteeing stability. The candidate controllers chosen for this application are designed in the PID form. Computational cost saving scheme further saves the energy, which is very important in satellite networks. The simulation results demonstrate the validity of the proposed framework.

VI. Acknowledgment

We thank Dr. Suresh Muknahallipatna of the University of Wyoming for fruitful discussions on computer networks.

References

[1] A. S. Tanenbaum. *Computer Network*. 4th, Ed., Pearson Education, 2003.

[2] D. Agrawal and F. Granelli. Redesigning an active queue management system. *IEEE Globecom*, vol. 2,:pp. 702–706, Dec. 2004.

[3] S. Low, F. Paganini, Z. Wang, and J. Doyle. A new tcp/aqm for stable operation in fast networks. *Proceedings of IEEE INFOCOM*, 2003.

[4] C. V. Hollot, V. Misra, D. Towsley, and W. Gong. Analysis and design of controllers for aqm routers supporting tcp flows. *IEEE Trans. Autom. Control*, Vol. 47, No. 6:pp. 945–959, 2002.

[5] K. B. Kim. Design of feedback controls supporting tcp based on the state space approach. *IEEE Trans. Autom. Control*, vol. 51 (7), July 2006.

[6] S. Manfredi, M. di Bernardo, and F. Garofalo. Robust output feedback active queue management control in tcp networks. *In IEEE Conf. on Decision and Control*, pages 1004–1009, Dec. 2004.

[7] M. Stefanovic and M. G. Safonov. Safe adaptive switching control: Stability and convergence. *IEEE Trans. Autom. Control*, 53(9):2012 – 2021, October 2008.

[8] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Trans. Networking*, 1:397–413, August 1993.

[9] V. Misra, W. Gong, and D. Towsley. Fluid-based analysis of network of aqm routers supporting tcp flows with an application to red. *In Proc. of ACM/SIGCOMM*, 2000.

[10] M. G. Safonov and T. C. Tsao. The unfalsified control concept and learning. *IEEE Trans. Autom. Control*, 42(6):843–847, Jun 1997.

[11] J. Cao and M. Stefanovic. Performance improvement in unfalsified control using neural networks. *In Proc. of 17th IFAC World Congress, Seoul, Republic of Korea*, July 2008.

[12] M. Jun and M. G.Safonov. Automatic pid tuning: An application of unfalsified control. *In Proc. IEEE CCA/CACSD, Kohala Coast-Island of Hawaii, HI*, pages 328–333, Aug 1999.

[13] A. O'Dwyer. *Handbook of PI And PID Controller Tuning Rules*. Imperial College Press, 2006.

[14] J. Balakrishnan and K. S. Narendra. Adaptive control using multiple models. *IEEE Trans. Autom. Control*, 42(2), Feb 1997.

[15] A. S. Morse, D. Q. Mayne, and G. C. Goodwin. Applications of hysteresis switching in parameter adaptive control. *IEEE Trans. Autom. Control*, 37(9), Sep 1992.