Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Design and analysis of a proportional-integral controller based on a Smith predictor for TCP/AQM network systems

Ouassim MENACER[1], Abderraouf MESSAI[1], Lazhar KASSA-BAGHDOUCHE[†‡2]

[1]*Department of Electronics, Faculty of Technology Sciences, Frères Mentouri University of Constantine 1, 25000 Constantine, Algeria*

[2]*Department of Electronics and Telecommunications, Faculty of Science and Technology, 8 May 1945 University of Guelma, 24000 Guelma, Algeria*

[†]E-mail: kassabaghdouche_lazhar@yahoo.com; kassabaghdouche.lazhar@univ-guelma.dz

Received May 23, 2020; Revision accepted Sept. 1, 2020; Crosschecked Jan. 17, 2022; Published online Feb. 4, 2022

Abstract: Active queue management (AQM) is essential to prevent the degradation of quality of service in TCP/AQM systems with round-trip time (RTT) delay. RTT delays are primarily caused by packet-propagation delays, but they can also be caused by the processing time of queuing operations and dynamically changing network situations. This study focuses on the design and analysis of an AQM digital controller under time-delay uncertainty. The controller is based on the Smith predictor algorithm and is called the SMITHPI controller. This study also demonstrates the stability of the controller and its robustness against network parameter variations such as the number of TCP connections, time delays, and user datagram protocol flows. The performance, robustness, and effectiveness of the proposed SMITHPI controller are evaluated using the NS-2 simulator. Finally, the performance of the SMITHPI controller is compared with that of a well-known queue-based AQM, called the proportional-integral controller.

Key words: Active queue management; Transport control protocol; Round-trip time delay; Smith predictor
https://doi.org/10.1631/FITEE.2000245 **CLC number:** TP13

1 Introduction

Internet-based services have evolved significantly in the past few years. Quality of service (QoS) is one of the most important aspects of Internet development. Transport Control Protocol (TCP) is the most common congestion-control protocol in Internet protocol (IP) networks; also called the "end-to-end" protocol, the principle of TCP involves controlling the transmission rates of each source with respect to the current traffic (Ramakrishnan et al., 2001). The basic operation of this protocol is simple: at every moment, the connected source competes for the access to resources and the size of the flows increases gradually until the network becomes congested. However, this control strategy can cause queues, thereby inducing large delay variations and alternating underutilization and saturation of links. Therefore, QoS of the network is considerably degraded. Several approaches have been proposed in the literature for queue management. Active queue management (AQM) is one such approach; it is designed for queue management at routers and is employed to ensure QoS in networks. The idea is to detect congestion before it actually occurs in networks by implicitly dropping packets and forcing TCP sources to respond. Many AQMs have been proposed for congestion control which are based on improving conventional methods through mathematical demonstration of network stability and multi-objective solutions. According to Ramakrishnan et al. (2001)

‡ Corresponding author
ORCID: Lazhar KASSA-BAGHDOUCHE, https://orcid.org/0000-0002-0013-1049

and Ma (2008), the proportional-integral (PI) technique is a control strategy that is frequently applied to the TCP/AQM network system. In the proposed technique, each part of the PI controller significantly contributes to achieving the target queue length. Integral action is required to accomplish a zero steady-state error, whereas proportional action is used in the transient stage to keep the output queue length close to the target value. The performance of several control strategies with the PI controller for AQM routers was discussed in Hollot et al. (2001) and Xiao et al. (2009). However, it is difficult to achieve significant control performance under all operational conditions with a linear PI controller due to large time delays (round-trip time, RTT). Moreover, it is difficult to estimate time delays correctly. Time delays are caused by the following: (1) propagation delays; (2) accumulation of a large number of TCP flows connected in series; (3) changes in dynamic network circumstances; (4) required processing time for queuing. Regardless of the progress in the area of the network system, networks with significant time delays are difficult to control using standard AQM controllers. The AQM control action requires time to elapse because of the changes in dynamic network circumstances. In addition, the AQM control action applied to the actual error attempts to control the queue length error that originated earlier.

In addition, re-tuning is required for a fixed PI-AQM controller to maintain a low buffer occupancy, which results in small queuing delays, low queue length fluctuation or jitter (stability), low packet loss, and high link utilization under heavy congestion. In the past, several researchers have addressed these problems by different control methods using a PI controller and time-delay compensators with model predictive control techniques. Barzamini et al. (2012) proposed an adaptive generalized minimum variance (AGMV) controller for dynamically varying TCP/AQM networks. The AGMV controller design combined real-time parameter estimation and generalized minimum variance (GMV), and then its performances were evaluated and compared with its adaptive minimum variance (AMV) counterpart under two distinct scenarios. The results obtained suggested that the AGMV controller is able to keep the queue length around the desired point. Wang DZ and Wu (2014) designed a congestion controller for TCP/AQM networks. In this work, the equilibrium of a class of TCP/AQM networks with time delay was investigated and the effect of communication time delay on the stability was addressed. The results showed that the nonlinear time-delay behavior of the system can be controlled using this approach. Kahe et al. (2014) proposed a compensated proportional-integral-derivative (PID) controller based on a new control strategy that addresses the phase lag and restrictions caused by the delay. The simulation results showed improvements, especially when the range of variation in delay and model parameters was drastic. Xu et al. (2016) derived a simplified network model based on the fluid-flow theory and stochastic differential equations. The simulation results demonstrated that the performance of the proposed controller is superior to that of the PI scheme in cases involving large delay and bursty data flow. Khoshnevisan and Salmasi (2016) proposed an internal model-control robust queue management scheme with two degrees of freedom. The authors introduced a system that entails two individual controllers for queue management and disturbance rejection, simultaneously. The results obtained demonstrated the effectiveness of the procedure and verified the analytical approach. Recently, Chebli et al. (2017) proposed a PID controller based on an extension of the Hermite-Biehler theorem applicable to quasi-polynomials, i.e., time-delay systems. The authors used an improved multi-objective genetic algorithm (GA) to find the optimal PID controller gains that minimize the performance indices of the integrated-time-squared error (ITSE), thereby ensuring the stability of the TCP network. Bisoy and Pattnaik (2017) proposed a proportional-differential-type feedback controller as a new kind of AQM to regulate the queue length with small oscillation. The results obtained showed that the proposed controller is stable and achieves a faster response in dynamic environments where the number of TCP connections, bottleneck capacity, and RTT keep changing. Manjunath and Raina (2019) studied compound TCP with a PI policy for queue management at Internet routers. The authors considered a nonlinear fluid model for the compound TCP-PI system. Fragility analysis of this model highlighted that even marginal variations in the PI parameters could induce instability. Alaoui et al. (2018) studied both single- and multi-bottleneck topologies of TCP/AQM network systems with mathematical models, in which multiple delays were introduced in TCP models. In this work, to ensure the asymptotic stabilization of the resulting closed-loop system, sufficient conditions for the existence of an admissible controller were established and a numerical example was given to illustrate the advantages of the proposed laxus feedback controller (LFC) with respect to some literature results. More recently, Kahe and Jahangir (2019) proposed a self-tuning compensated PID controller to address the time-varying nature of network conditions caused by parameter variations and unresponsive connections. The authors indicated that the proposed self-tuning controller can adapt itself reasonably well to different operating conditions, while preserving the simplicity of PI controllers. In addition, their work focused on estimating all the input-output data within such a complex system to determine the appropriate structure of the neural network controller under these circumstances. However, the self-tuning controller could not tolerate time-delay variation effects and demonstrated unstable behaviors due to large RTTs. More broadly, long queue length may unnecessarily increase the queuing delay of the packet, cause the timeout to occur frequently, and lead to undesired retransmission. In contrast, short queue length results in higher packet loss and lower link utilization. Such shortcomings can be overcome through regulation of the queue length to the desired value. Hence, regulating queue length to the desired value and avoiding overflow and underflow situations of the queue have been considered the main AQM objectives (Wang P et al., 2012; Alvarez and Martínez, 2013; Sheikhan et al., 2013; Zhou et al., 2013; Yazdi and Delavarkhalafi, 2018; Wang K et al., 2019) to improve the performance of high link utilization with low packet loss. Further, due to a certain level of instability in the TCP/AQM network system, caused by the time delay, a new controller is needed to compensate for the time delay in a TCP/AQM network system. In fact, taking into consideration the design of the feedback control system, the main objectives of the proposed controller can be specified through dynamics for tracking error; hence, various control objectives can be achieved accordingly. Although the time delay of TCP/AQM network systems has been improved in recent years, most improvements were achieved using a standard PI controller. However, it is possible to further improve the performance of such network systems.

Therefore, in this study we analyze the performance of a TCP/AQM network system based on a new controller (SMITHPI controller) that can maintain the queue length at a certain threshold (to limit delays), while maximizing the throughput despite the changes in the network load and the presence of user datagram protocol (UDP) flows that do not include a congestion management mechanism. The Smith controller is designed to compensate for random time delays and tries to control the queue length error by considering the past reaction and variations in the queue length error in the previous cycle. The extensive simulation results demonstrate that the proposed SMITHPI controller achieves a faster control response and has good stability. Furthermore, compared to the PI controller, the implementation of the proposed SMITHPI controller in miniaturized devices like field programmable gate arrays and microcontrollers has a very small scale due to its complexity in implementation and its computation time requirement.

2 TCP/AQM system with a static PI controller

A linearized model for TCP congestion control, delays, and queues is expressed by the transfer function Plant(s), and the introduction of the PI controller results in the feedback control as shown in Fig. 1 (Hollot et al., 2001).

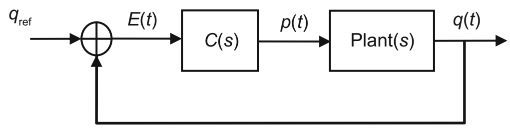

Fig. 1 The closed-loop system of the TCP/AQM linearized model Plant(s) with the proportional-integral (PI) controller $C(s)$

In the TCP/AQM system, q_{ref} is the target queue length, $p(t)$ the drop probability, and $E(t)$ the queue length error between the target queue length and the instantaneous queue length $q(t)$. $C(s)$ and Plant(s) are the Laplace transform of the controller and the dynamic model of the network, respectively.

The dynamic model of the TCP behavior was developed using fluid-flow and stochastic differential equation analysis (Hollot et al., 2001, 2002). This model has relation with the average value of the variable parameters of network and is described by the following coupled, nonlinear differential equations:

$$\begin{cases} \dot{W}(t)=\dfrac{1}{R(t)}-\dfrac{W(t)W(t-R(t))}{2R(t-R(t))}p(t-R(t)), \\ \dot{q}(t)=\dfrac{W(t)}{R(t)}N(t)-C, \end{cases} \tag{1}$$

where $W(t)$ is the TCP congestion window size (packet), $q(t)$ the queue length (packet), $R(t)$ the round-trip time (s), C the link capacity (Mb), $N(t)$ the traffic load (number of TCP connections), and $p(t)$ the packet drop probability. $\dot{X}$ denotes the time derivative of X.

The equilibrium values for the traffic load N and the round-trip delay R are described in the following linear model:

$$\begin{cases} \delta\dot{W}(t)=-\dfrac{2N_0}{CR_0^2}\delta W(t)+\dfrac{C^2R_0}{2N_0^2}\delta p(t-R_0), \\ \delta\dot{q}(t)=\delta W(t)\dfrac{N_0}{R_0}-\delta q(t)\dfrac{1}{R_0}, \end{cases} \tag{2}$$

where $\delta\dot{W}(t)$ and $\delta\dot{q}(t)$ are the perturbed variables over the operating point. N_0 and R_0 are the constants of traffic load $N(t)$ and delay term $R(t)$, respectively.

Consequently, the process plant is generally modeled with time delay and can be represented in a convenient notation as follows:

$$\text{Plant}(s)=\frac{\delta q(s)}{\delta p(s)}=\frac{K\mathrm{e}^{-sR}}{a_1s^2+a_2s+a_3}, \tag{3}$$

where $K=\dfrac{C^3R_0^3}{(2N_0)^2}$, $a_1=\dfrac{R_0^3C}{2N_0}$, $a_2=R_0+\dfrac{C^3R_0}{2N_0}$, and $a_3=1$.

The basic structure of controller $C(s)$ as the standard PID regulator is (Alvarez and Martínez, 2013)

$$\text{PID}=k_\text{p}+\frac{k_\text{i}}{s}+k_\text{d}s, \tag{4}$$

where k_p, k_i, and k_d are proportional, integral, and derivative parameters, respectively.

A fixed-gain PID is adequate for controlling a nominal TCP/AQM process; however, the requirement for high performance with changes in operational conditions or environmental parameters (R, C, N) makes the delay important and thus degrades the performance of a simple PID controller. In fact, the time delay introduces an additional phase lag in the TCP/AQM system, which can be represented as a linear transfer function according to the Padé approximation as follows:

$$\mathrm{e}^{-Rs}\cong\frac{\sum_{i=0}^{j}(-1)^i\dfrac{(2j-i)!}{(j-i)!i!}Rs^i}{\sum_{i=0}^{j}\dfrac{(2j-i)!}{(j-i)!i!}Rs^i}, \tag{5}$$

where R is the RTT consisting of the propagation time and queuing delay; in our case, we use a first-order Padé approximation. Using this approximation, the transfer function of the TCP/AQM system can be written as follows:

$$\text{Plant}(s)=\frac{K}{a_1s^2+a_2s+a_3}\cdot\frac{1-\frac{R_0}{2}s}{1+\frac{R_0}{2}s}. \tag{6}$$

The transfer function is used for tuning and obtaining PI control parameters. The performance of the designed PI controller is tested via the NS-2 simulator. The nominal system parameters are as follows: $N_0=30$ TCP connections, $C=15$ Mb, $q_0=150$ packets, and $R_0=0.32$ s. The queue length variations (Fig. 2) show a faster transient response in stabilizing the queue lengths around target values than the AQM tuned with the original model as presented in Hollot et al. (2002). It is possible to verify the delay time at the output.

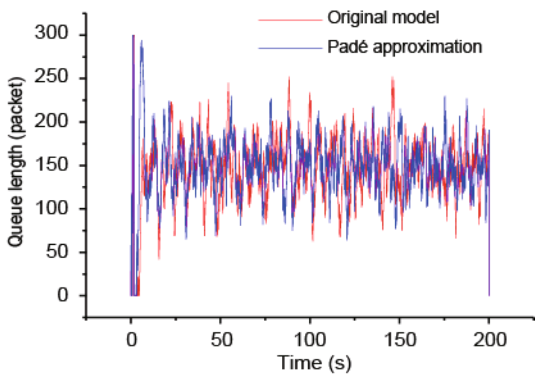

Fig. 2 Queue length variations (References to color refer to the online version of this figure)

3 TCP/AQM system with a SMITHPI controller

The Smith predictor (SP) algorithm takes the time-delay dynamic process into consideration. Recently, several studies were conducted to improve the performance of the SP. Ogata (2010) proposed a tool to control the system with time delays. Fig. 3 shows the control structure of the SP.

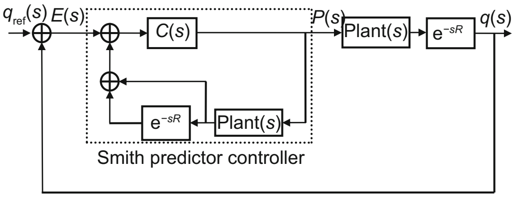

Fig. 3 Smith predictor structure

In Fig. 3, e^{-sR} represents the model of the plant time delay, and Plant(s) represents the plant model without delay. The SP controller is characterized by the input $E(s)$ and the output $P(s)$, where $E(s)$ is the queue length error between the target q_{ref} and the instantaneous queue length and $P(s)$ is the drop probability. It is possible to verify that $C(s)$ can be designed considering the transfer function Plant(s) without time delay. The transfer function of the complete system includes the design specification plus the dead time at the original plant. There is no time delay in the feedback, and thus the performance of the system is improved.

4 Digital implementation of the SMITHPI controller

There are several methods to convert differential equations into difference equations when implementing the controllers in discrete time (Isermann, 1989). Hollot et al. (2001) proposed a method for digital implementation of the PI controller, which we have used in the NS-2 simulator.

The equation that describes the TCP/AQM system with input $u(k)$ and output $y(k)$ is as follows (Barzamini et al., 2012):

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})} z^{-d} u(k) + \frac{C(z^{-1})}{A(z^{-1})} e(k), \qquad (7)$$

where d is the system delay, $A(z^{-1})$ a single polynomial of order n with coefficients a_i, $B(z^{-1})$ a general polynomial of order n with coefficients b_i, and $C(z^{-1})$ a general polynomial of order n with coefficients c_i. $u(k)$, $y(k)$, and $e(k)$ are the sampled process input, output signal, and the source noise, respectively.

$A(z^{-1})$, $B(z^{-1})$, and $C(z^{-1})$ can be computed using the following equations:

$$\begin{cases} A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_n z^{-n}, \\ B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \ldots + b_n z^{-n}, \\ C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \ldots + c_n z^{-n}, \end{cases} \qquad (8)$$

where $b_0 \neq 0$.

The algorithm used for digital implementation of the SMITHPI controller (Fig. 4) is sensitive to the choice or the estimate of the discrete delay $\hat{d}$. Thus, for estimating this time delay, Eqs. (7) and (8) can be approximated by the difference equation:

$$\hat{y}(k) = \underline{\boldsymbol{f}}^{\text{T}}(k, \hat{d}) \hat{\underline{\boldsymbol{P}}}(k), \qquad (9)$$

where

$$\begin{aligned} \underline{\boldsymbol{f}}(k, \hat{d}) =& [u(k-\hat{d}), u(k-\hat{d}-1), \ldots, \\ & u(k-\hat{d}-n-1), -y(k-1), \\ & -y(k-2), \ldots, -y(k-n)]^{\text{T}}, \end{aligned} \qquad (10)$$

$$\hat{\underline{\boldsymbol{P}}} = [\hat{b}_0, \hat{b}_1, \ldots, \hat{b}_{n+1}, \hat{a}_1, \ldots, \hat{a}_n]^{\text{T}}. \qquad (11)$$

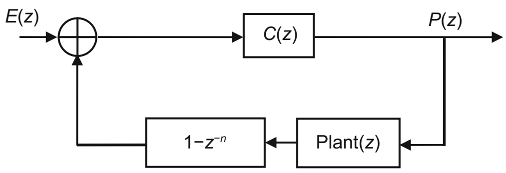

Fig. 4 Block diagram of a digital Smith predictor (Xu et al., 2016)

Kurz and Goedecke (1981) proposed a parameter vector optimization technique of containing more elements than the original $\hat{\underline{\boldsymbol{P}}}$ of Eq. (11). In the estimation technique, the longer parameter vector is the data vector. This modified structure is given by the following:

$$\hat{y}(k) = \left(\underline{\boldsymbol{f}}^x(k, \hat{d})\right)^{\text{T}} \hat{\underline{\boldsymbol{P}}}^x(k), \qquad (12)$$

where

$$\begin{aligned} \underline{\boldsymbol{f}}^x(k, d_{\max}) =& [u(k), u(k-1), \ldots, \\ & u(k-n-1-d_{\max}), -y(k-1), \\ & -y(k-2), \ldots, -y(k-n)]^{\text{T}}, \end{aligned} \qquad (13)$$

$$\hat{\underline{\boldsymbol{P}}}^x = [\hat{b}_0^x, \hat{b}_1^x, \ldots, \hat{b}_{n+1+d_{\max}}^x, \hat{a}_1, \hat{a}_2, \ldots, \hat{a}_n]^{\text{T}}, \quad (14)$$

$$\frac{\hat{B}^x(z^{-1})}{\hat{A}^x(z^{-1})} = \frac{\hat{b}_0^x + \hat{b}_1^x z^{-1} + \ldots + \hat{b}_{n+1+d_{\max}}^x z^{-n}}{\hat{a}_0 + \hat{a}_1 z^{-1} + \ldots + \hat{a}_n z^{-n}}, \qquad (15)$$

where $d_{\max}$ is the upper limit of the time delay.

For digital implementation, the z-transform of the controller transfer function is calculated for a sampling interval ΔT, the same as that for the PI controller in Hollot et al. (2002), namely, $\Delta T = 1/160$ s.

The $C(z)$ controller transfer function in the z-domain becomes

$$C(z) = \frac{a - bz^{-1}}{1 - z^{-1}}. \tag{16}$$

Similarly, the transfer function Plant(z) in the z-domain is given by

$$\text{Plant}(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{b_0 + b_1 z^{-1} + b_2 z^{-2}}. \tag{17}$$

The sample interval is adjusted to 1/160 s, because the length of the RTT is approximately greater. The time delay verified at the system can be easily modeled by measuring the number of samples during the total RTT. This measurement is represented as $n = \text{RTT}/T$ (T is the sampling time).

The feedback transfer function $P(z)/E(z)$ is expressed as Eq. (18) (on the bottom of this page).

The digital SMITHPI controller designed is represented by the transfer function (18) which can be converted to Eq. (19) (on the bottom of this page) for $t = kT$ s.

The digital implementation of Eq. (19) is performed with Algorithm 1.

Algorithm 1 Iterative algorithm for digital implementation of the SMITHPI controller

q // Current instantaneous queue length
q_{ref} // Target queue length
pro[i] // Calculated packet drop probability
Calculate_p() // Calculated every t_m seconds
$P = 4.040\times10^{-5}(q-q_{\text{ref}})-4.063\times10^{-5}(q_{\text{old}}-q_{\text{ref}})+5.306\times10^{-5}(q_{\text{elder}}-q_{\text{ref}})-0.4658\times10^{-6}(q_{\text{eldest}}-q_{\text{ref}})+2.972\text{prob}-2.944\text{prob}[1]+0.9722\text{prob}[2]+2.056\times10^{-6}\text{prob}[40]+2.069\times10^{-6}\text{prob}[41]-2.036\times10^{-6}\text{prob}[42]-2.043\times10^{-6}\text{prob}[43]$
$q_{\text{eldest}} = q_{\text{elder}}$
$q_{\text{elder}} = q_{\text{old}}$
$q_{\text{old}} = q$
prob $=P$
pro[1]=prob
for (i=1; i<=43; i++) **do**
    pro[45−i]= pro[44−i]
end for
Enqueue() // Called upon each packet arrival
if p>1 **then**
    p=1
else if $p < 0$ **then**
    p=0
end if
random=uniform random(0, 1)
if buffer is full **then**
    Drop the packet
else if random > p **then**
    Enqueue the packet
end if

5 Simulation results and analysis

This section discusses the performance of the TCP/AQM network system with the SMITHPI controller using the NS-2 simulator (Riley and Henderson, 2010). The PI controller (Hollot et al., 2002) is chosen for validation and is compared with the proposed AQM. For the PI controller, we use the default parameters in NS-2 (Hollot et al., 2002). Certain default parameters are generally used in the studies of the AQM schema, and the parameters of SMITHPI are set as discussed later. The stabilization of the instantaneous queue size at a target value is one of the key performance indices to evaluate AQM. If the instantaneous queue can be regulated to remain close to a desirable target, then an increased demand can be achieved to use the Internet for high-speed and delay-sensitive applications in differing QoS

$$\frac{P(z)}{E(z)} = \frac{4.040\times10^{-5}z^{43} - 4.063\times10^{-6}z^{42} + 2.525\times10^{-6}z^{41} - 0.4658z^{40}}{z^{43} - 2.972z^{42} + 2.944z^{41} - 0.9722z^{40} - 2.056\times10^{-6}z^{3} - 2.069\times10^{-6}z^{2} + 2.036\times10^{-6}z - 2.043\times10^{-6}}. \tag{18}$$

$$\begin{aligned} P(k) =& 4.040\times10^{-5}e(k) - 4.063\times10^{-5}e(k-1) + 5.306\times10^{-5}e(k-2) - 0.4658\times10^{-6}e(k-3) \\ &+ 2.972P(k-1) - 2.944P(k-2) + 0.9722P(k-3) + 2.056\times10^{-6}P(k-40) \\ &+ 2.069\times10^{-6}P(k-41) - 2.036\times10^{-6}P(k-42) - 2.043\times10^{-6}P(k-43). \end{aligned} \tag{19}$$

requirements, especially when the target queue length is achieved independent of traffic conditions (Ryu et al., 2003a, 2003b).

Fig. 5 shows the proposed topology of the network system. The single-bottleneck link lies between the two routers R_1 and R_2 with a capacity of 15 Mb and a delay of 80 ms, whereas the other links are assumed to have sufficient capacity to carry their traffic. Router R_1 uses the PI controller and the SMITHPI controller (our controller), whereas other routers use Droptail. The packet size is 1000 bytes, the buffer size is 1000 packets, and the target queue length is set to 200 packets. The time of simulation is 200 s.

First, we test whether the SMITHPI controller can regulate the queue length around the expected target for different loads and link capacity with a variety of target choices. Thus, we investigate the robustness of our approach under variations in the TCP load (the number of TCP connections, N). The proposed PI and SMITHPI controllers are designed for nominal values of the TCP/AQM system parameters. These simulations are conducted to analyze the effect of the number of TCP connections on the queue length and the delay of the TCP/AQM network system. Such cases are depicted in Fig. 6. As can be seen from Fig. 6a, the performance of the SMITHPI controller is better than that of the PI controller. The PI controller becomes unstable (fluctuations with large amplitudes) over time. The PI controller regulates the queue length slowly, which results in a long disturbed queue. However, the SMITHPI controller remains available for stabilizing the queue length. In addition, Fig. 6b shows the plot of the delay against time. It can be seen that using the SMITHPI controller, a delay can be tuned and regulated by controlling the queue length nearly to the reference queue length. Thus, the delay is stable with regard to a reference value. In the PI controller, the delay increases with the increase of the number of TCP connections. The results show that the SMITHPI controller can control the queue length around the reference value. Therefore, it can be concluded that a stable reference delay can be maintained for a high number of TCP connections using the SMITHPI controller.

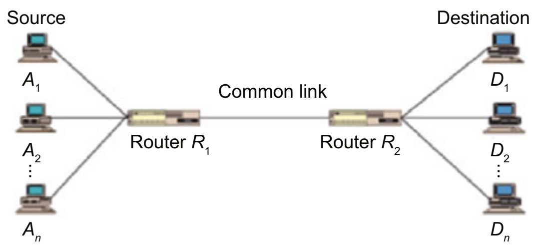

Fig. 5 Single-bottleneck topology

For the visual representation of the difference between the two controllers, Figs. 7a and 7b show the variations in the mean and standard deviation of queue length for different numbers of TCP connections, respectively. As the number of TCP connections increases, the mean of the queue length of the SMITHPI controller is nearly equal to the target queue length. In contrast, the mean and standard deviation of the queue length of the PI controller increase with the increase in the number of TCP connections. Moreover, the standard deviation of the queue length of the SMITHPI controller is smaller

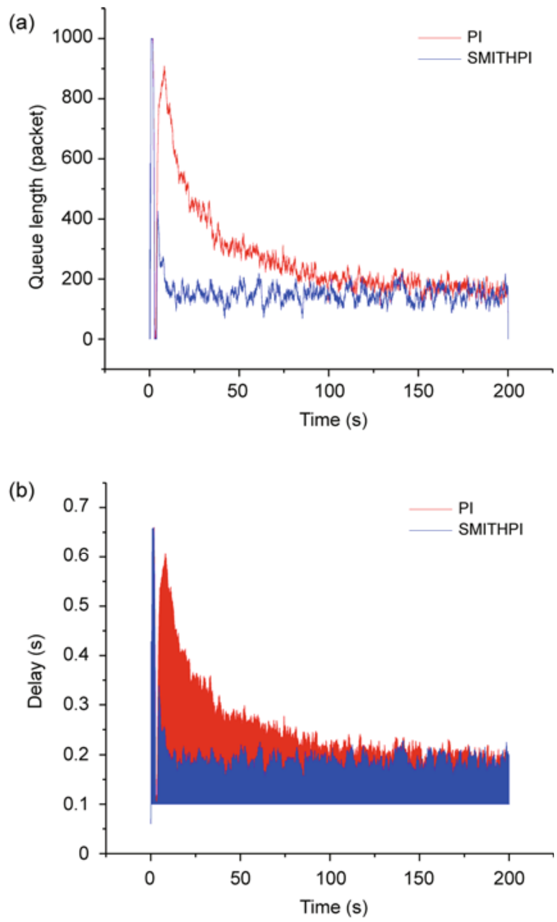

Fig. 6 Instantaneous queue length (a) and delay (b) as a function of time for 300 TCP connections (References to color refer to the online version of this figure)

and more stable than the one for the PI controller, thus supporting a good performance of the SMITHPI controller.

To verify the validity of the proposed SMITHPI controller, several simulations are conducted. The variations in the mean and standard deviation of the delay under load variations are studied and analyzed. Figs. 8a and 8b show the mean and standard deviation of delays as a function of the number of TCP connections, respectively. These figures show that the delays of the SMITHPI controller are constant and close to the target delay, and that the delays of the PI controller are sensitive to the number of TCP connections. The simulation results show that the SMITHPI controller yields a good performance for the TCP/AQM network system.

The robustness of the proposed approach under variations in RTT is then examined. A similar approach is used to analyze the effect of the RTT on the queue length for the two types of controllers. Fig. 9 shows the variation in the instantaneous queue length as a function of time for different RTT values. We can see that the SMITHPI controller performs better than the PI controller for all RTT values. As the RTT increases, the PI controller becomes unstable (fluctuations with large amplitudes). When the RTT exceeds 200 ms, the PI controller slowly regulates the queue to the target value, which results in a long disturbed queue. However, the SMITHPI controller still results in a stable queue length even for high RTT values.

To analyze the performance of the TCP/AQM network system using the two types of controllers, the variation in the mean and standard deviation of the queue length against the RTT is presented in Fig. 10. As RTT increases, the mean queue length of the SMITHPI controller becomes nearly equal to the

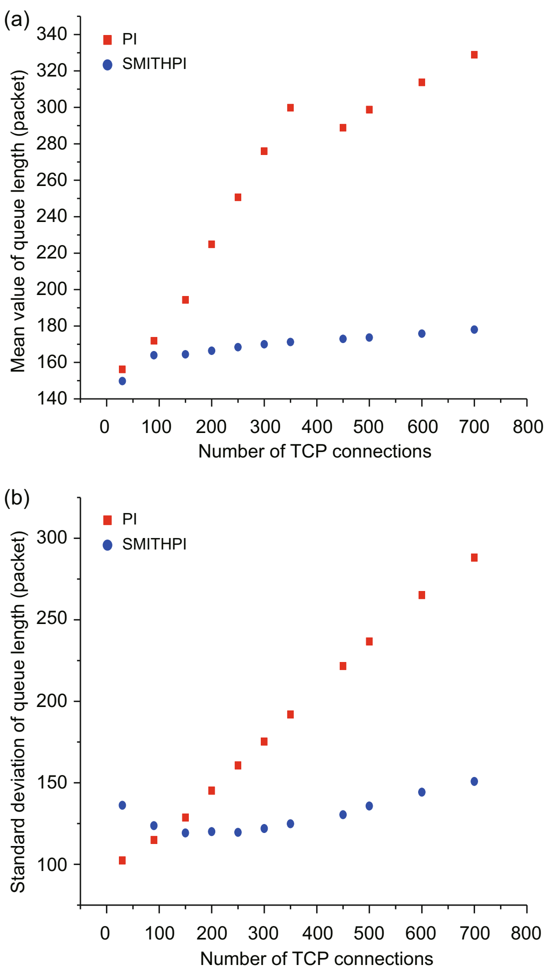

Fig. 7 Mean (a) and standard deviation (b) of the queue length under load variations

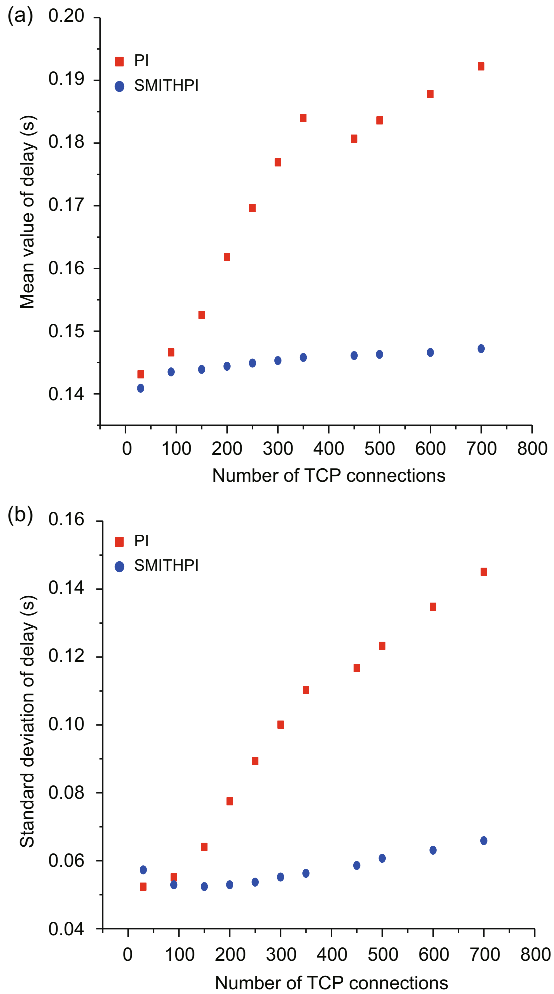

Fig. 8 Mean (a) and standard deviation (b) of the delay under load variations

target queue length, whereas the mean queue length of the PI controller changes with the variation in the RTT. The standard deviation of the queue length for the SMITHPI controller is smaller and more stable than the one for the PI controller.

To analyze the queue length of the TCP/AQM network system for a large number of TCP connections, Fig. 11 shows the queue length variations as a function of time for 500 TCP connections for the two types of controllers. As can be seen from this figure, the RTT is randomly distributed with a mean of 200 ms. The results of this study demonstrate that the SMITHPI controller is still effective in stabilizing the queue length around the target queue length with TCP connections having different random variations in RTT, and that all our previous observations apply.

Herein, the performances of the TCP/AQM network system using the two types of controllers are evaluated. The queue length of the network system is evaluated for different bottleneck link capacities

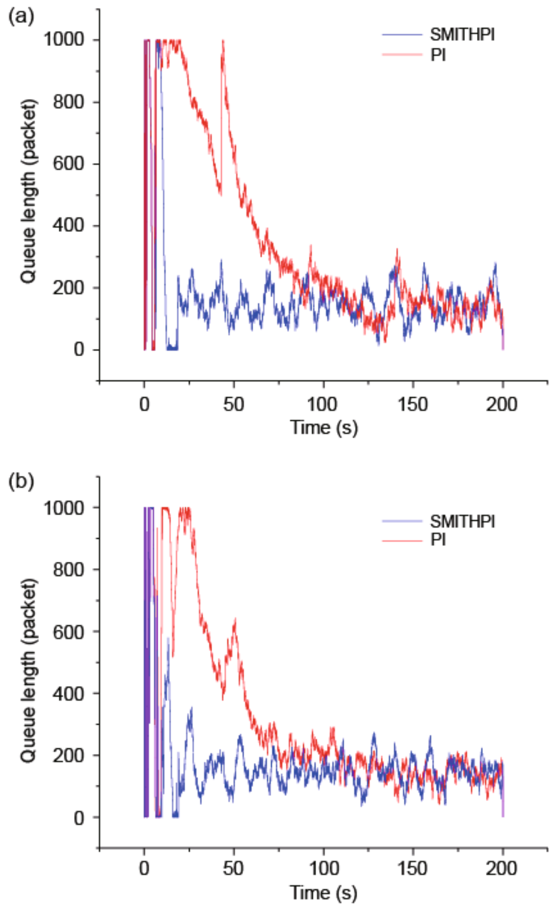

Fig. 9 Instantaneous queue length as a function of time for RTT=200 ms (a) and RTT=400 ms (b) (References to color refer to the online version of this figure)

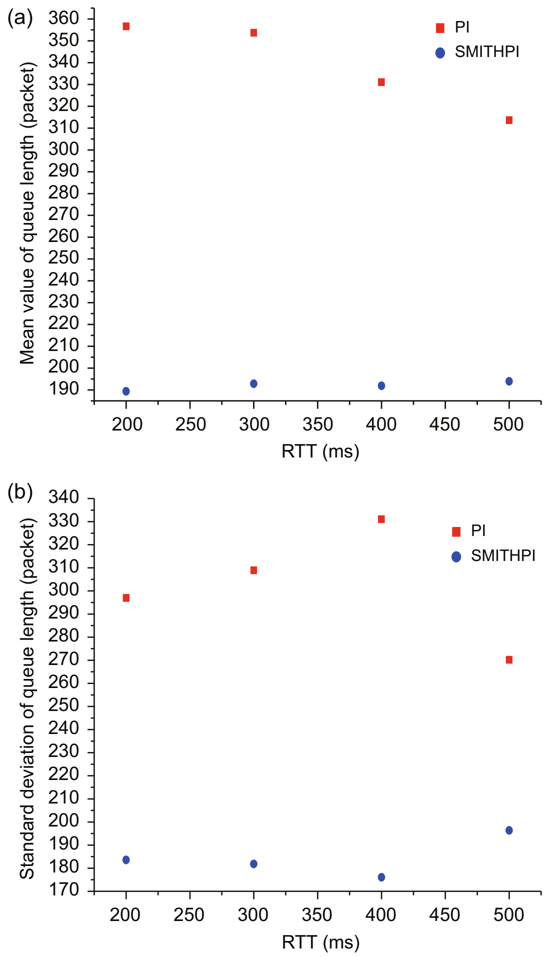

Fig. 10 Mean (a) and standard deviation (b) of the queue length under RTT variations

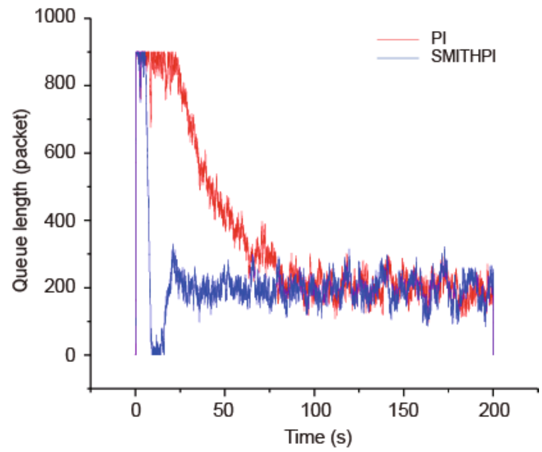

Fig. 11 Queue length variations for 500 TCP connections with RTT exponentially distributed with a mean of 200 ms (References to color refer to the online version of this figure)

(C): 5 and 45 Mb. The number of TCP connections is set to 500. Fig. 12 shows the plot of the variation in the queue length versus time for 500 TCP connections and different values of bottleneck link capacities. The SMITHPI controller yields the same performance when the link capacity changes. However, the instability of the PI controller decreases with the increase in the link capacity. As a result, the proposed SMITHPI controller works efficiently for different link capacities and performs better than the PI controller.

To justify the efficiency of the proposed SMITHPI controller, Table 1 shows the mean and standard deviation of the queue length for bottleneck link capacity C = 5 and 45 Mb. We observe that the mean queue lengths for the SMITHPI controller are 325.7449 and 312.2280 for C = 5 and 45 Mb, respectively, which are close to the target of 300. Moreover, the standard deviation of the queue length for the SMITHPI controller is lower than the one for the PI controller, which supports the good performance of the proposed SMITHPI controller.

Table 1 Mean and standard deviation of the queue length for C = 5 and 45 Mb

Controller	Mean		STD	
	C=5 Mb	C=45 Mb	C=5 Mb	C=45 Mb
PI	477.9299	361.9996	221.0805	153.8315
SMITHPI	325.7449	312.2280	145.6207	100.9769

STD: standard deviation

Furthermore, we evaluate the performance of the TCP/AQM network system for different target queue lengths. We set $q_{\rm ref}$ to 50 and 400 packets and the number of TCP connections to 500. Then we plot the variations in the queue length of the network system using the two types of controllers as a function of time for different values of target queue length. Such cases are depicted in Fig. 13. We can see that the PI controller becomes unstable (fluctuations with large amplitudes) around the target queue length. This controller regulates the length of the queue slowly, which leads to a long disturbed queue. However, the SMITHPI controller remains available for stabilizing the queue length even for a high value of the target queue length. Thus, the SMITHPI controller ensures successful stabilization of the queue length around any expected target value (any level of QoS). In addition, the mean and standard deviation of the queue length (Table 2) show a difference between the two controllers. The mean and standard deviation of the queue length are calculated for $q_{\rm ref}$ = 50 and 400 packets. We observe that the mean queue lengths for the SMITHPI controller are 76.8541 and 415.1725 for $q_{\rm ref}$=50 and 400 packets, respectively, which are close to the targets of 50 and 400 packets. Furthermore, the standard deviation of the queue length for the SMITHPI controller is lower than the one for the PI controller.

The aforementioned simulation results showed

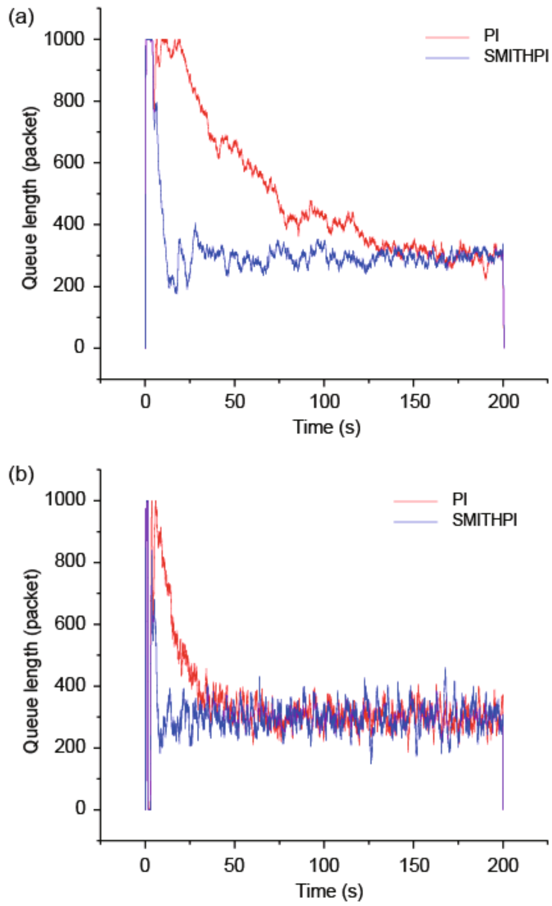

Fig. 12 Queue length variations for 500 TCP connections with the bottleneck link capacities of 5 Mb (a) and 45 Mb (b) (References to color refer to the online version of this figure)

Table 2 Mean and standard deviation of the queue length for target queue length $q_{\rm ref}$ = 50 and 400 packets

Controller	Mean		STD	
	$q_{\rm ref}$ = 50	$q_{\rm ref}$ = 400	$q_{\rm ref}$ = 50	$q_{\rm ref}$ = 400
PI	212.9753	510.0294	261.1631	179.7765
SMITHPI	76.8541	415.1725	150.7211	102.0215

STD: standard deviation

that the performance of the SMITHPI controller is better than that of the PI controller. Furthermore, two simulations are conducted to examine the performance of the TCP/AQM network system using the two types of controllers when the number of TCP connections varies with time. The first simulation consists of 500 TCP connections; we divide them into two groups and the number of TCP connections increases abruptly. In this case, we observe the queue length variations after 100 s, and evaluate the instantaneous queue length of the PI and SMITHPI controllers after there are sudden traffic changes in 100 s. Fig. 14 shows the variations in the queue length as a function of time under sudden traffic changes. The figure shows a heavy traffic change at 100 s. In addition, the SMITHPI controller quickly regulates the queue length to the expected target value and effectively shortens the buffer overflow. On the contrary, worse stability and robustness to varying load and longer buffer overflow of the PI controller are observed.

In the second simulation, we examine the queue length of the two controllers when the number of TCP connections increases randomly from 100 to 1000. The results obtained are compatible with the previous results. Fig. 15 shows that the SMITHPI controller is still successful in stabilizing the queue length around the target value with a faster transient response and smaller oscillations under random variations in the number of TCP connections compared to the PI controller. Furthermore, the effect of UDP flow disturbances on the performance of the TCP/AQM network system is examined when the number of TCP connections is fixed at 500 and mixed with 400 UDP flows. The RTT of TCP connections is exponentially distributed with a mean of 200 ms, and the propagation delays of all UDP flows are exponentially distributed with a mean of 200 ms. Each of the UDP flows follows an exponential ON/OFF

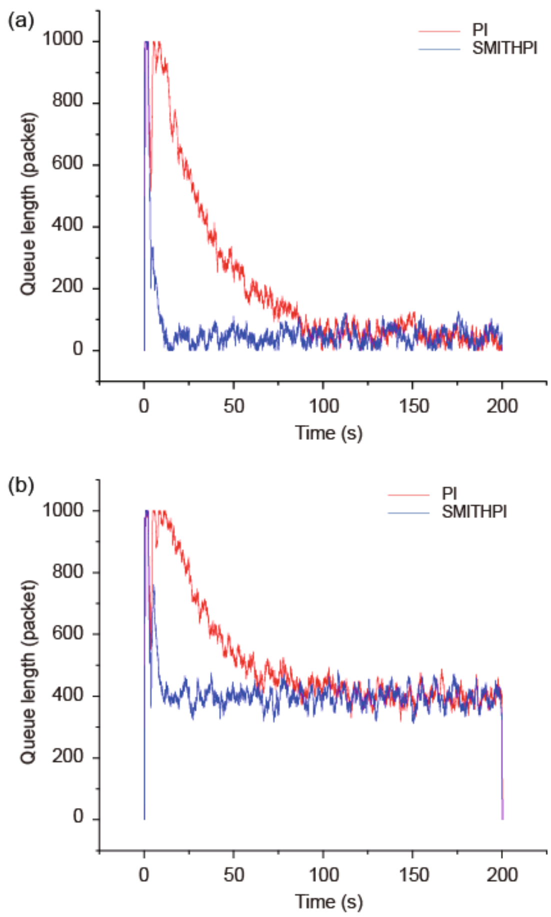

Fig. 13 Queue length variations for 500 TCP connections with the target queue length of 50 packets (a) and 400 packets (b) (References to color refer to the online version of this figure)

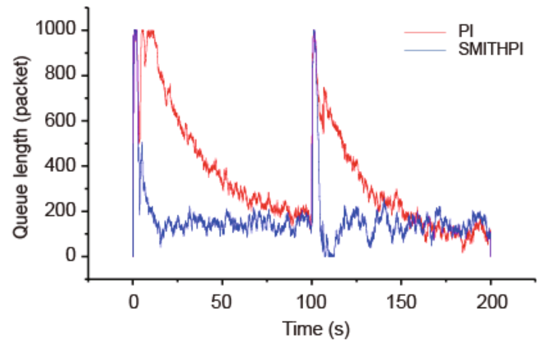

Fig. 14 Queue length variations under sudden traffic changes, i.e., the number of TCP connections increasing abruptly (References to color refer to the online version of this figure)

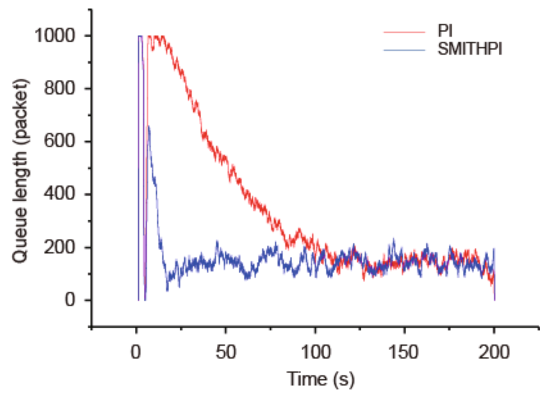

Fig. 15 Queue length variations when the number of TCP connections increases randomly from 100 to 1000 (References to color refer to the online version of this figure)

traffic model. Both idle and burst times have a mean of 0.5 s. The sending rate during ON time is 64 kb/s with the packet size of 500 bytes. Fig. 16 depicts the variations in the queue length for 500 TCP connections with 400 UDP flows added. The figure shows that the SMITHPI controller is successful in stabilizing the queue length around the target value with a faster transient response and smaller oscillations under UDP disturbances compared to the PI controller.

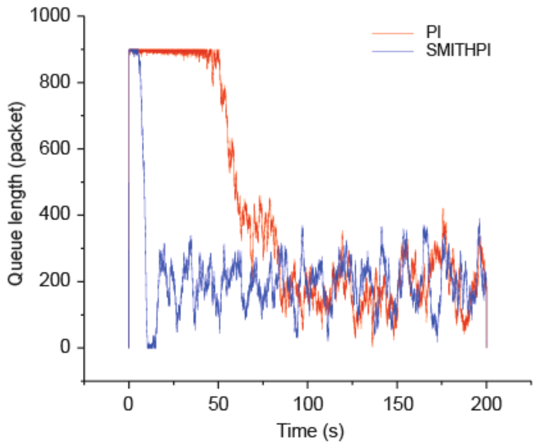

Fig. 16 Queue length variations for 500 TCP connections adding 400 user datagram protocol (UDP) flows (References to color refer to the online version of this figure)

6 Conclusions

In this study we have proposed the design and analysis of a new congestion controller (SMITHPI controller) to efficiently control the queue length and improve the stability of the TCP/AQM network system, thereby improving the quality of service (QoS). The performance of the proposed controller has been computed using the NS-2 network simulator, and its stability was analyzed against different network parameters such as the number of TCP connections, bandwidth, round-trip time (RTT), and target queue length. Furthermore, its stability was analyzed under sudden changes in the TCP load and with non-responsive user datagram protocol (UDP) connections. The simulation results demonstrate that the designed controller achieves faster stabilization of the queue length around the given target and maintains the queue length close to the target value. Therefore, such a controller has a lower average delay and average jitter and an improved QoS performance. These properties are required to control time-sensitive applications such as video or radio broadcast via HTTP.

Contributors

Ouassim MENACER and Abderraouf MESSAI designed the research. Abderraouf MESSAI guided the research. Ouassim MENACER and Abderraouf MESSAI conducted the simulations, derived the main results, and drafted the paper. Lazhar KASSA-BAGHDOUCHE helped organize and revised the paper.

Compliance with ethics guidelines

Ouassim MENACER, Abderraouf MESSAI, and Lazhar KASSA-BAGHDOUCHE declare that they have no conflict of interest.

References

Alaoui SB, Tissir EH, Chaibi N, 2018. Active queue management based feedback control for TCP with successive delays in single and multiple bottleneck topology. *Comput Commun*, 117:58-70. https://doi.org/10.1016/j.comcom.2018.01.003

Alvarez T, Martínez D, 2013. Handling the congestion control problem of TCP/AQM wireless networks with PID controllers. In: Yang GC, Ao SL, Gelman L (Eds.), IAENG Transactions on Engineering Technologies: Special Volume of the World Congress on Engineering 2012. Springer, Dordrecht, p.365-379. https://doi.org/10.1007/978-94-007-6190-2_28

Barzamini R, Shafiee M, Dadlani A, 2012. Adaptive generalized minimum variance congestion controller for dynamic TCP/AQM networks. *Comput Commun*, 35(2): 170-178. https://doi.org/10.1016/j.comcom.2011.08.010

Bisoy SK, Pattnaik PK, 2017. Design of feedback controller for TCP/AQM networks. *Eng Sci Technol Int J*, 20(1):116-132. https://doi.org/10.1016/j.jestch.2016.10.002

Chebli S, Elakkary A, Sefiani N, 2017. Multi-objective genetic algorithm optimization using PID controller for AQM-TCP networks. *Int Rev Autom Contr*, 10(1):33-39. https://doi.org/10.15866/ireaco.v10i1.11143

Hollot CV, Misra V, Towsley D, et al., 2001. On designing improved controllers for AQM routers supporting TCP flows. Proc 20th Annual Joint Conf of the IEEE Computer and Communications Society, Conf on Computer Communications, p.1726-1734. https://doi.org/10.1109/INFCOM.2001.916670

Hollot CV, Misra V, Towsley D, et al., 2002. Analysis and design of controllers for AQM routers supporting TCP flows. *IEEE Trans Autom Contr*, 47(6):945-959. https://doi.org/10.1109/TAC.2002.1008360

Isermann R, 1989. Digital Control Systems, Vol. 1, Fundamentals, Deterministic Control (2nd Ed.). Springer-Verlag, Berlin, Germany.

Kahe G, Jahangir AH, 2019. A self-tuning controller for queuing delay regulation in TCP/AQM networks.

Telecommun Syst, 71(2):215-229.
https://doi.org/10.1007/s11235-018-0526-1

Kahe G, Jahangir AH, Ebrahimi B, 2014. AQM controller design for TCP networks based on a new control strategy. *Telecommun Syst*, 57(4):295-311.
https://doi.org/10.1007/s11235-013-9859-y

Khoshnevisan L, Salmasi FR, 2016. A robust and high-performance queue management controller for large round trip time networks. *Int J Syst Sci*, 47(7):1586-1597. https://doi.org/10.1080/00207721.2014.941959

Kurz H, Goedecke W, 1981. Digital parameter-adaptive control of processes with unknown dead time. *Automatica*, 17(1):245-252.
https://doi.org/10.1016/0005-1098(81)90099-6

Ma XY, 2008. The Research of Internet Congestion Control Algorithm. MS Thesis, Lanzhou University of Technology, Lanzhou, China (in Chinese).

Manjunath S, Raina G, 2019. Stability and performance of compound TCP with a proportional integral queue policy. *IEEE Trans Contr Syst Technol*, 27(5):2139-2155. https://doi.org/10.1109/TCST.2018.2855141

Ogata K, 2010. Modern Control Engineering (5^{th} Ed.). Prentice Hall, Upper Saddle River, USA.

Ramakrishnan K, Floyd S, Black D, 2001. The Addition of Explicit Congestion Notification (ECN) to IP. RFC 3168. https://doi.org/10.17487/RFC3168

Riley GF, Henderson TR, 2010. The NS-3 network simulator. In: Wehrle K, Güneş M, Gross J (Eds.) Modeling and Tools for Network Simulation. Springer, Berlin, Heidelberg.
https://doi.org/10.1007/978-3-642-12331-3_2

Ryu S, Rump C, Qiao CM, 2003a. Advances in Internet congestion control. *IEEE Commun Surv Tut*, 5(1):28-39. https://doi.org/10.1109/COMST.2003.5342228

Ryu S, Rump C, Qiao CM, 2003b. A predictive and robust active queue management for Internet congestion control. Proc 8^{th} IEEE Symp on Computers and Communications, p.991-998.
https://doi.org/10.1109/ISCC.2003.1214245

Sheikhan M, Shahnazi R, Hemmati E, 2013. Adaptive active queue management controller for TCP communication networks using PSO-RBF models. *Neur Comput Appl*, 22(5):933-945.
https://doi.org/10.1007/s00521-011-0786-0

Wang DZ, Wu SJ, 2014. Design of the congestion control for TCP/AQM network with time-delay. *Math Probl Eng*, 2014:834698. https://doi.org/10.1155/2014/834698

Wang K, Liu Y, Liu XP, et al., 2019. Adaptive fuzzy funnel congestion control for TCP/AQM network. *ISA Trans*, 95:11-17. https://doi.org/10.1016/j.isatra.2019.05.015

Wang P, Chen H, Yang XP, et al., 2012. Design and analysis of a model predictive controller for active queue management. *ISA Trans*, 51(1):120-131.
https://doi.org/10.1016/j.isatra.2011.08.006

Xiao LS, Wang ZX, Peng XH, 2009. Research on congestion control model and algorithm for high-speed network based on genetic neural network and intelligent PID. Proc 5^{th} Int Conf on Wireless Communications, Networking and Mobile Computing, p.1-6.
https://doi.org/10.1109/WICOM.2009.5302733

Xu S, Fei MR, Yang XH, 2016. A new scheme for network congestion control based on modified adaptive Smith predictor. *Int J Simul-Syst Sci Technol*, 17(28):30.1-30.6.

Yazdi MN, Delavarkhalafi A, 2018. Robust stability and design of state feedback controller for straightforward active queue management. *Int J Anal Appl*, 16(5):654-672.

Zhou C, He JW, Chen QW, 2013. A robust active queue management scheme for network congestion control. *Comput Electr Eng*, 39(2):285-294.
https://doi.org/10.1016/j.compeleceng.2012.11.008