

Received August 30, 2020, accepted September 15, 2020, date of publication September 21, 2020, date of current version October 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3025377

An Adaptive Active Queue Management Based on Model Predictive Control

QIN XU¹, GAOYAO MA², KAIMENG DING¹, AND BIN XU³

¹School of Network and Communications Engineering, Jinling Institute of Technology, Nanjing 211169, China

²Jiangsu Sundry Law Firm, Nanjing 210017, China

³Shanghai Shenzhou Engineering Company Ltd., Shanghai 202161, China

Corresponding author: Kaimeng Ding (dkm@jit.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 41801303, in part by the Jiangsu Province Science and Technology Support Program under Grant BK20170116, in part by the Scientific Research Hatch Fund of Jinling Institute of Technology under Grant jit-b-201645, and in part by the Qing Lan Project.

ABSTRACT In recent years, active queue management (AQM) has gained more and more attention as an important part of network congestion control. Although there are many AQM algorithms, these algorithms show weaknesses to detect and control congestion due to the complexity and dynamics of the networks. Hence, this paper proposes a new AQM algorithm based on model predictive control (MPC) theory which has been widely applied in nonlinear and time-delay systems. In order to adjust the parameters of the MPC-based AQM algorithm adaptively according to network scenario variations, the adaptive mechanism is introduced into the new algorithm, named PHAQM, by using the Hebb learning rules from the neural network control theory. The simulation results show that the algorithm is effective in avoiding network congestion. Compared to the traditional AQM schemes, such as PI, REM, and GPC algorithm, the PHAQM has a faster convergence rate and smaller queue length fluctuations and outperforms especially under dynamically changing network situations.

INDEX TERMS Active queue management, adaptive control, model predictive control, network congestion control.

I. INTRODUCTION

The congestion control of the transmission control protocol (TCP) network is an important tool to improve the quality of service (QoS), which can prevent network collapse, avoid lockout behavior and effectively reduce the probability of control-loop synchronization [1]. To assist TCP's management of network performance, the active queue management (AQM) mechanism was introduced to allow the router involved. As an effective congestion control approach, the AQM has become an attractive research topic [2]. In fact, the AQM mechanism implemented in the router has significant performance development for the network, for instance, improving network utilization, reducing packet drops, and keeping the best-effort service with low-delay [3].

There are many related pieces of research focused on the AQM since the first proposed AQM algorithm named Random Early Detection (RED) [4], such as adaptive RED [5], BLUE [6], and YELLOW [7]. However, these algorithms were heuristic, which made the algorithms too sensitive to

parameter configuration. Fortunately, the fluid-flow model for the congestion control process in TCP networks was established by using stochastic theory [8], which helps the researches design and understand the behaviors of internet systems better. Based on the fluid-flow model, [9] first took the TCP/AQM system as a feedback system to be analyzed with control-theory, which has given a feedback control system depiction of AQM. The action of an AQM control law is to mark packets (with probability) as a function of measured queue length. Generally, when the TCP/AQM system is analyzed by control-theory as a feedback system, the probability of packet mark/drop in the router would be the control signal and the queue length would be the controlled variable. Afterwards, the control-theory-based approaches were used to solve the network congestion problem based on the model, for instance, PI [10], robust control [11], prescribed performance control [12], and state-feedback congestion controller [13]. These controllers improve the performance of the congestion control systems in wide network scenarios.

Among the control theories, it is known that model predictive control (MPC) has become a mature and advanced control strategy, which has been widely applied in nonlinear

The associate editor coordinating the review of this manuscript and approving it for publication was Dipankar Deb¹.

and time-delay systems. As the network congestion control system is a typical nonlinear and time-delay system, the MPC-based AQM algorithms were developed to determine the optimal control signal during each sampling time by predicting the future system dynamics. Generalized predictive control (GPC) is the first MPC that is used for network congestion control [14]. Then, predictive function control (PFC) was developed as a privileged AQM method in the high-speed networks [15]. MPAQM controller drop early packets at the router reasonably according to the predicted future queue length in the data buffer [16]. Data-AQM is proposed based on data-driven predictive control which can obtain the prediction directly basing on the input-output data alone without any explicit model of the system [17]. These researches demonstrate that MPC is able to handle system delay along with low computational load, and the MPC-based AQM algorithms outperform with a comparison of the traditional AQM algorithms.

It is known that the parameters of the Internet are time-varying in practice, such as load factor, round trip time, and link capacity. However, the traditional control parameters configured in particular network scenarios are not able to be adjusted, which causes the performance sensitivity to network scenario variations. Hence, the researches draw more attention to the adaptive technique [18]–[23], which is a powerful method to uncertain system, see [24]–[26], to name a few. Although there are some results on the AQM schemes based on adaptive control, the adaptive scheme for the MPC-based AQM is a challenge because the environment of the Internet is complex. Fortunately, the neural network control is developed as a useful tool to design adaptive sampled-data systems [27], [28]. As Hebb-learning rule is an unsupervised learning rule for neural-network, it can extract the statistical characteristics of the training set and is suitable to solve the adaptive problem for the MPC-based AQM. In order to adjust the parameters of MPC-based AQM schemes adaptively, this paper investigates an adaptive Hebb-learning rules for TCP/AQM systems with MPC control and proposes a new algorithm called PHAQM.

The remainders of this paper is organized as follows. The fluid model and the Laplace transfer function are presented in Section II. In Section III, a new AQM algorithm named PHAQM is proposed based on MPC and Hebb-learning rules. Section IV gives a simulation study to verify the proposed method. Finally, a conclusion is provided in Section V.

II. SYSTEM MODELS

Reference [8] had developed a dynamic model of TCP behavior by using fluid-flow and stochastic differential equation analysis. This fluid-flow model has been widely used in the design of the AQM algorithms, which can be described by the following coupled, nonlinear differential equations:

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

where $\dot{x}(t)$ denotes the time-derivative of x , W is the expected TCP window size, N is the load factor, C is the link capacity, q is the expected queue length, and R is the round-trip time. Note that the round-trip time can be expressed as $R(t) = T_p + \frac{q(t)}{C}$, where T_p is the propagation delay (secs), and p is probability of packet mark/drop. The queue length q and window-size W are positive, bounded quantities; i.e. $q \in [0, \bar{q}]$, and $W \in [0, \bar{W}]$ where \bar{q} and \bar{W} denote buffer capacity and maximum window size respectively. Also, the marking probability $p \in [0, 1]$.

In order to design the feedback control (AQM), [9] approximate these dynamics by their small-signal linearization about an operating point (W_0, q_0, p_0) as following:

$$\begin{cases} \delta \dot{W}(t) = \frac{-2N}{R_0^2} \delta W(t) - \frac{R_0 C^2}{2N^2} \delta p(t - R_0) \\ \delta \dot{q}(t) = \frac{N}{R_0} \delta W(t) - \frac{1}{R_0} \delta q(t) \end{cases} \quad (2)$$

where $\delta W(t) = W(t) - W_0$, $\delta q(t) = q(t) - q_0$, $\delta p(t) = p(t) - p_0$, $R_0 = T_p + \frac{q_0}{C}$ represent the perturbed variable about the operating point, and the constraints of δW , δq , and δp are ignored for simplicity. It is noted that the operating point (W_0, q_0, p_0) is denoted as following:

$$\begin{cases} \delta \dot{W}(t) = 0 \Rightarrow W_0^2 p_0 = 2 \\ \delta \dot{q}(t) = 0 \Rightarrow W_0 = R_0 \frac{C}{N} \end{cases} \quad (3)$$

Performing a Laplace transform on the differential (2), the transfer function $G_p(s)$ which relates the δp and δq can be obtained as:

$$G_p(s) = \frac{R_0 C^2}{(s + \frac{2N}{R_0^2})} \frac{\frac{N}{R_0}}{(s + \frac{1}{R_0})} e^{-sR_0}. \quad (4)$$

III. PHAQM ALGORITHM

A. AN AQM ALGORITHM BASED ON MPC

To design an AQM algorithm based on MPC, the traditional zero-order hold (ZOH) is used to discretize (4). Suppose the values of δp and δq are sampled in each interval T_s , then the transfer function which relates the δp and δq can be:

$$G(z) = \frac{(m_1 z^{-1} + m_2 z^{-2})}{(1 + n_1 z^{-1} + n_2 z^{-2})} z^{-d}, \quad (5)$$

where $d = \lceil \frac{R_0}{T_s} \rceil$ is the system delay, m_1, m_2, n_1, n_2 are determined by the network parameters and T_s as the following:

$$\begin{cases} m_1 = \frac{C^3 R_0^3 \left(\left(2 - 2e^{-\frac{T_s}{R_0}} \right) N + CR_0 \left(-1 + e^{-\frac{2NT_s}{CR_0^2}} \right) \right)}{4N - 2CR_0} \\ m_2 = \frac{C^3 e^{-\frac{(2N+CR_0)T_s}{CR_0^2}} R_0^3 \left(2 \left(-1 + e^{\frac{T_s}{R_0}} \right) N - CR_0 \left(-1 + e^{\frac{2NT_s}{CR_0^2}} \right) \right)}{2(-2N + CR_0)} \\ n_1 = -e^{-\frac{2NT_s}{CR_0^2}} - e^{-\frac{T_s}{R_0}} \\ n_2 = e^{-\frac{(2N+CR_0)T_s}{CR_0^2}} \end{cases} \quad (6)$$

Define the values of δp and δq as the input value u and output value y , respectively, then the dynamic model of TCP behavior can be described as the following input-output difference equation:

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t), \quad (7)$$

where $A(z^{-1}) = 1 + n_1z^{-1} + n_2z^{-2}$, $B(z^{-1}) = m_1z^{-1} + m_2z^{-2}$.

In order to derive the predictive value of δq after j th interval, i.e., jT_s seconds later, a Diophantine equation is introduced:

$$A(z^{-1})E(z^{-1}) + z^{-j}F(z^{-1}) = 1, \quad (8)$$

where $E(z^{-1})$ and $F(z^{-1})$ are the polynomials determined by $A(z^{-1})$ and j , which can be expressed as $E(z^{-1}) = e_0 + e_1z^{-1} + \dots + e_{j-1}z^{-(j-1)}$, $F(z^{-1}) = f_0 + f_1z^{-1}$.

Multiplying $E(z^{-1})$ to the two sides of (7), we have:

$$y(t+j) = z^{-d}B(z^{-1})E(z^{-1})u(t+j) + F(z^{-1})y(t). \quad (9)$$

Let $j = d + 1$, and transfer the above equation from Z domain to time domain, then

$$y(t+d+1) = f_0y(t) + f_1y(t-1) + g_1u(t) + g_2u(t-1) + \dots + g_{d+2}u(t-d-1), \quad (10)$$

where $g_1 = e_0m_1$, $g_2 = m_1e_1 + m_2e_0$, $g_3 = m_1e_2 + m_2e_1$, \dots , $g_{d+1} = m_1e_d + m_2e_{d-1}$, $g_{d+2} = m_2e_d$.

Note that $y(t) = \delta q(t) = q(t) - q_0$, and the reference queue length is $q(t+d+1) = q_0$, so $y(t+d+1) = 0$, then (10) can be simplified as:

$$u(t) = h_1y(t) + h_2y(t-1) + h_3u(t-1) + \dots + h_{d+3}u(t-d-1), \quad (11)$$

where $h_1 = -\frac{f_0}{g_1}$, $h_2 = -\frac{f_1}{g_1}$, $h_3 = -\frac{g_2}{g_1}$, \dots , $h_{d+3} = -\frac{g_{d+2}}{g_1}$.

B. ADAPTIVE SCHEME BASED ON HEBB-LEARNING THEORY

Whereas there are so many researches related to AQM scheme using the fluid-flow model, most of the existing works are based on fixed network arguments. Hence, the parameters of the corresponding AQM algorithm are fixed. However, the arguments of real networks are time-varying, which makes the traditional AQM algorithms perform unsatisfactorily in real networks. Besides, the fluid-flow model is not precise due to the complexity in considering all the situations in the real networks. So, the corresponding AQM algorithm obtained from the fluid-flow model is not reliable enough to bring good performance in real networks. To overcome these problems of conventional AQM algorithms, this paper introduces an adaptive scheme based on Hebb-learning theory to adjust the parameters h_i ($i = 1, 2, \dots, d+3$) shown in (11).

To make the equation easier to understand, rewrite (11) as following:

$$u(t) = K \sum_{i=1}^{d+3} w_i(t)x_i(t), \quad (12)$$

where $x_1(t) = y(t)$, $x_2(t) = y(t-1)$, $x_3(t) = u(t-1)$, \dots , $x_{d+3}(t) = u(t-d-1)$, $K > 0$ denotes the proportional coefficient, and the values of $w_i(t)$ are the adaptive-tuned parameters. Using Hebb-learning theory to adjust the parameters, the adaptive-tuned parameters can be updated as following:

$$w_i(t) = w_i(t-1) + \eta_i e(t-1)x_i(t-1)u(t-1), \quad (13)$$

where $\eta_i > 0$ is the learning rate, $e(t-1) = q(t-1) - q_0$. In practice, as the arguments of real networks are time-varying, the values of h_i are hard to estimate. Fortunately, the initial values of $w_i(t)$ can be set as small non-zero values, and the simulation results show that the proposed AQM algorithm, i.e., PHAQM, is still effective to avoid network congestion.

IV. SIMULATION

In this section, we evaluate the performance and robustness of the proposed PHAQM algorithm by several simulations. The network topology used in the simulation is shown in Fig. 1, where the bottleneck link is located between Router B and Router C. Unless mentioned otherwise, the default values are set as follows: the bandwidth of all links is 45 Mb/s, the propagation delay is 100 ms, the number of TCP connections (TCP/Reno) is 500, the average size of TCP packets is 1000 Bytes, the buffer size is 1000 packets, router B adopts PHAQM as the congestion control strategy while other nodes adopt Drop-tail. The default parameters of PHAQM are set as follows: the sample interval is 0.01 s, $K = 0.01$, $d = 14$, $\eta_1 = \eta_2 = \dots = \eta_{17} = 0.001$, the expected queue length of Router B is 500 packets, and the initial values of $w_i(t)$ are $w_1 = w_2 = \dots = w_{17} = 0.001$.

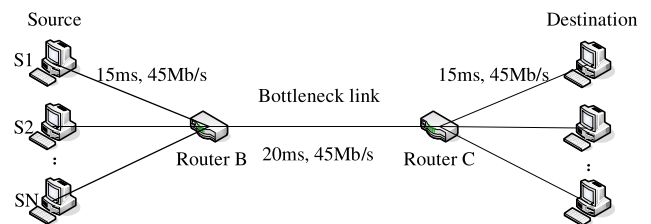


FIGURE 1. Simulation topology.

A. CONSTANT NUMBER OF TCP CONNECTIONS

Load sizes are changeable in real networks, so network congestion algorithms should be able to adapt to different numbers of TCP connections. This group of experiments mainly investigates the queue length of the PHAQM algorithm under different loads, i.e., different numbers of TCP connections. In the four simulations, the number of TCP connections is 200, 500, 1000, and 2000 respectively, and all TCP connections start sending data at the initial time. The simulation results are shown in Fig. 2. It can be seen that the network load has little effect on the performance of PHAQM. PHAQM can keep the queue length stable at the expected value under different TCP connection numbers.

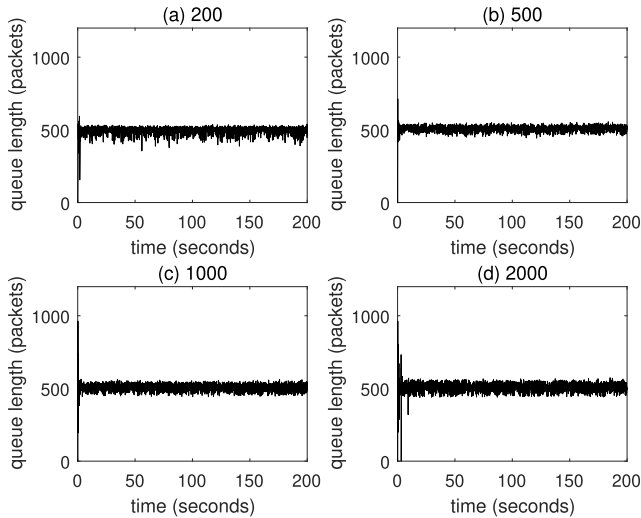


FIGURE 2. Queue lengths involving different numbers of TCP connections: (a) 200 TCP connections; (b) 500 TCP connections; (c) 1000 TCP connections; (d) 2000 TCP connection.

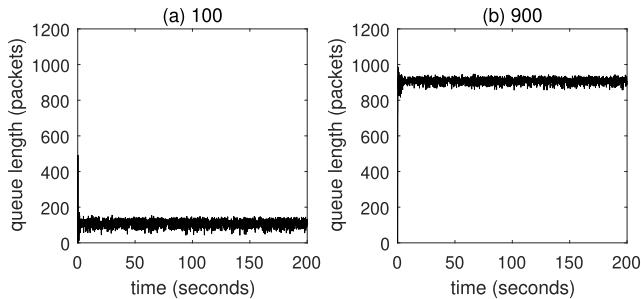


FIGURE 3. Queue lengths for the following queue length targets: (a) 100 packets; (b) 900 packets.

B. PERFORMANCE FOR DIFFERENT REFERENCE QUEUE LENGTH

The reference queue length should be set to reduce round-trip delay and avoid empty buffer to improve the utilization rate of the bottleneck link. The smaller the reference queue length is, the smaller the round-trip delay is, whereas the higher the probability of empty buffer is. Therefore, it is necessary to weigh the pros and cons carefully to select a reasonable reference queue length, which requires that the algorithm can stabilize the queue length at different values to adapt to different networks. This group of experiments investigates the ability of the PHAQM algorithm to stabilize the queue length at different values. The simulation results are shown in Fig. 3 when the reference queue lengths are set to 100 and 900, respectively. It can be seen that no matter what the target queue length is, PHAQM can stabilize the queue near the target value, with fast convergence speed and small queue length oscillation.

C. COMPARISON WITH OTHER AQM SCHEMES

In the actual network, new data flows may enter the network at each time period, and existing data flows may exit the

network. Therefore, the load of the network is dynamically changed. A good active queue management algorithm should be able to withstand the load dynamics. This section compares the performance of PI, REM, GPC, and PHAQM in the case of dynamic changes in the number of TCP connections using two sets of environments. The main control parameters of each algorithm are as follows: PI parameter, $a = 0.00001822$, $b = 0.00001816$; REM parameter, $\gamma = 0.001$, $\Phi = 0.001$; GPC parameter $C = 45$ Mb/s, $N = 500$, $R = 0.14$ s.

In the first set of experiments, we will compare the control effects of each AQM algorithm when the network load suddenly changes. In this experimental environment, 200 TCP connections pass through the bottleneck link at the initial time, and then 100 connections are added every 10 seconds until the 100th second, i.e., 900 TCP connections are added in 100 seconds. Then every 10 seconds, 100 connections are reduced at the same time, and a total of 900 connections are reduced at last. The results are shown in Fig. 4. It can be seen that the performances of PI and REM degrade noticeably if the network load suddenly changes, which means that PI and REM are very sensitive to mutation load. The variation of queue length in the REM method has the highest value, and REM takes a longer time to regular the queue length to the target value, especially when the network load suddenly decreases. Compared with PI and REM, GPC based on MPC reduces the sensitivity to mutation network, whereas the queue length oscillation is large. As an MPC-based algorithm, PHAQM can also absorb burst data streams well in the dramatically changing network environment, and it can maintain small queue length oscillation in addition. In fact, PHAQM gives less overshoot, shorter response time, and better stability than other algorithms. Conclusively, PHAQM is able to adapt to the environment involving sudden load reduction as quickly as possible while maintaining a small queue oscillation.

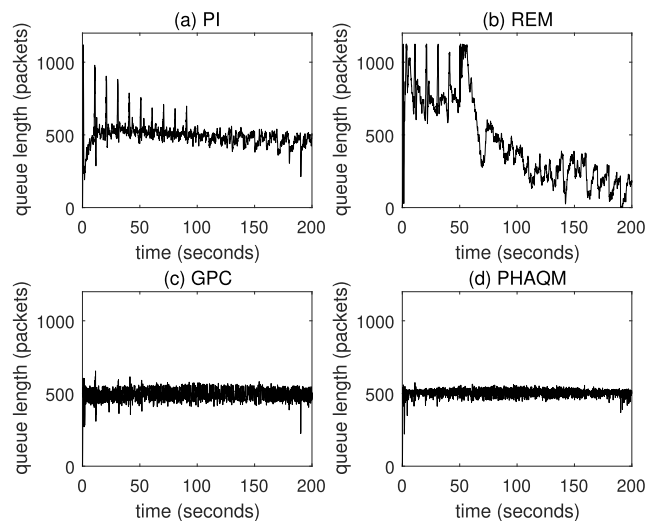


FIGURE 4. Queue lengths with different AQM algorithms when the network load suddenly changes: (a) PI; (b) REM; (c) GPC; (d) PHAQM.

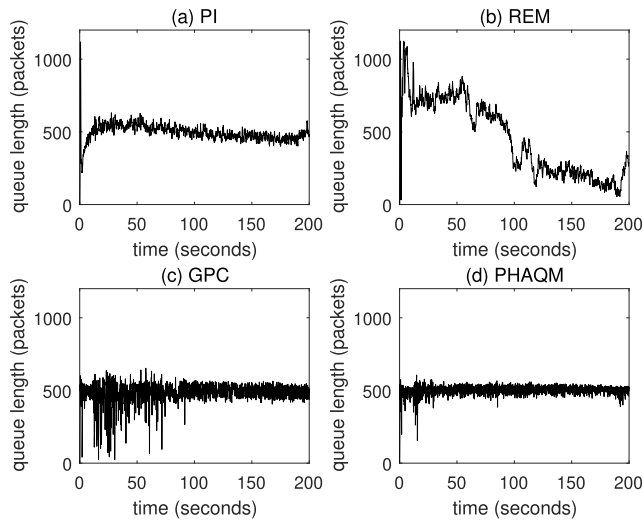


FIGURE 5. Queue lengths with different AQM algorithms when the network load uniformly changes: (a) PI; (b) REM; (c) GPC; (d) PHAQM.

In the second set of experiments, we will focus on the effect on the performance of different AQM algorithms when the network load changes uniformly. In the experimental environment, there are 200 TCP connections through the bottleneck link at the initial moment. During the 10s to 90s, 800 connections are evenly added, and 800 connections are evenly reduced between 110s and 190s. The results are shown in Fig. 5. It can be seen that in a slowly changing network environment, although PI maintains a small queue length oscillation, the queue length does not stabilize near the expected value. REM is very sensitive to the change of the network load, and its queue length is difficult to stabilize. GPC algorithm shows an acceptable behavior when the load uniformly decreases, but it performs poorly when the load uniformly increases. It can be seen that GPC has noticeable oscillations in the first 100s. Whatever, PHAQM can stabilize the queue length near the expected value, and the queue length oscillation is small, no matter the load increases or decreases.

D. MULTIPLE BOTTLENECKS

To evaluate the performance of PHAQM under the multi-bottleneck link, this section will give the simulation results with two bottleneck links. The network topology is presented in Fig. 6. The link between R1 and R2 and the link between R3 and R4 are two bottleneck links. The settings of the environments are shown in Fig. 6. PHAQM is employed as the AQM algorithm on R1 and R3. There are 300 TCP sources connected to R1, and the destinations of each 100 sessions are connected to R2, R3, and R4. Besides, there are 200 TCP sources connected to R2, and the destinations of each 100 sessions are connected to R3 and R4, respectively. Finally, 100 TCP sessions are traveling from R3 to R4. The results are shown in Fig. 7, where (a) and (b) are the queue lengths on R1 and R3, respectively. In Fig. 7, the queue length dynamics are shown to exhibit good robustness and

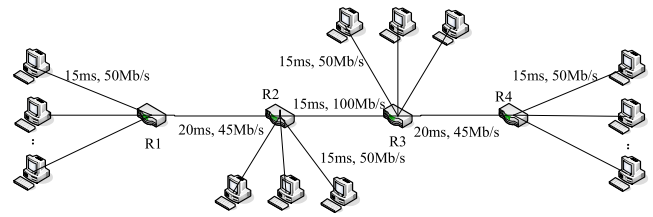


FIGURE 6. Topology of multiple bottleneck links.

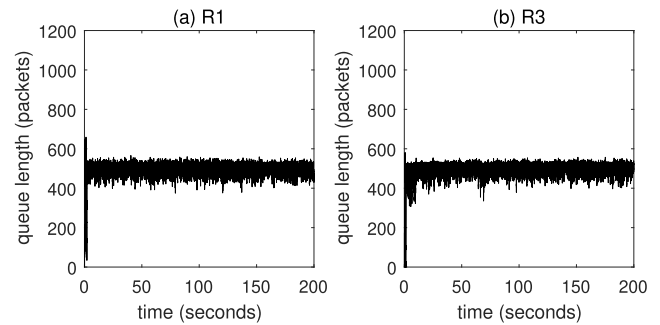


FIGURE 7. Queue lengths: (a) R1; (b) R3.

fast system response. It can be seen that PHAQM can also effectively stabilize the queue length near the reference value in a complex network.

V. CONCLUSION

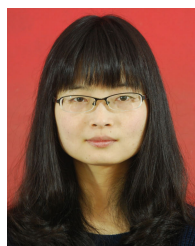
There are many pieces of research devoted to improving the performance of the congestion control in the Internet. However, it is still a challenge to detect and control congestion due to the complexity and dynamics of the networks. Hence, this paper is aimed to solve the above problem by applying the MPC theory and employing the adaptive scheme. In this paper, a TCP/AQM dynamic model is used to obtain a prediction model. An adaptive AQM algorithm, named PHAQM, is proposed based on the model by using MPC theory, and the adaptive scheme is designed with Hebb learning rules.

A large number of simulation experiments demonstrate the effectiveness of the PHAQM algorithm. The developed controller not only can guarantee that the queue length tracks the desired queue length, but also can be adaptive to network scenario variations. It is shown by experiments that the PHAQM has better robustness, and it has significant advantages in a dynamic network environment compared with traditional algorithms, such as PI, REM, and GPC. In particular, PHAQM can regulate the queue length around the expected value and obtain a good control effect under multiple bottleneck link scenarios.

Future work will cover the theoretical analysis of the stability and convergence speed of PHAQM and the extension of the simulation environment from numerical simulations to real network experiments.

REFERENCES

- [1] V. Jacobson, "Congestion avoidance and control," in *Proc. Symp. Proc. Commun. Archit. Protocols SIGCOMM*, 1988, pp. 314–329.
- [2] S. Patel, "Nonlinear performance evaluation model for throughput of AQM scheme using full factorial design approach," *Int. J. Commun. Syst.*, vol. 33, no. 8, May 2020, Art. no. e4357.
- [3] R. Adams, "Active queue management: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1425–1476, 3rd Quart., 2013.
- [4] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Netw.*, vol. 1, no. 4, pp. 397–413, Aug. 1993.
- [5] J. Aweya, M. Ouellette, D. Y. Montuno, and A. Chapman, "Enhancing TCP performance with a load-adaptive RED mechanism," *Int. J. Netw. Manage.*, vol. 11, no. 1, pp. 31–50, 2001.
- [6] W.-c. Feng, K. G. Shin, D. D. Kandlur, and D. Saha, "The BLUE active queue management algorithms," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 513–528, Aug. 2002.
- [7] C. Long, B. Zhao, X. Guan, and J. Yang, "The yellow active queue management algorithm," *Comput. Netw.*, vol. 47, no. 4, pp. 525–550, Mar. 2005.
- [8] 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 *Proc. Conf. Appl., Technol., Archit., Protocols Comput. Commun. SIGCOMM*, 2000, pp. 151–160.
- [9] C. V. Hollot, V. Misra, D. Towsley, and W.-B. Gong, "A control theoretic analysis of RED," in *Proc. IEEE INFOCOM Conf. Comput. Commun., 20th Annu. Joint Conf. IEEE Comput. Commun. Soc.*, Apr. 2001, pp. 1510–1519.
- [10] Y. Hong, O. W. W. Yang, and C. Huang, "Self-tuning PI TCP flow controller for AQM routers with interval gain and phase margin assignment," in *Proc. IEEE Global Telecommun. Conf. GLOBECOM*, Nov. 2004, pp. 1324–1328.
- [11] C. Zhou, J. He, and Q. Chen, "A robust active queue management scheme for network congestion control," *Comput. Electr. Eng.*, vol. 39, no. 2, pp. 285–294, Feb. 2013.
- [12] Y. Liu, X. Liu, Y. Jing, and S. Zhou, "Adaptive backstepping H_∞ tracking control with prescribed performance for Internet congestion," *ISA Trans.*, vol. 72, pp. 92–99, Jan. 2018.
- [13] Z. Li, Y. Liu, and Y. Jing, "Design of adaptive backstepping congestion controller for TCP networks with UDP flows based on minimax," *ISA Trans.*, vol. 95, pp. 27–34, Dec. 2019.
- [14] X.-H. Zhang, K.-S. Zou, Z.-Q. Chen, and Z.-D. Deng, "Stability analysis of AQM algorithm based on generalized predictive control," in *Proc. Int. Conf. Intell. Comput.* Berlin, Germany: Springer, 2008, pp. 1242–1249.
- [15] N. Bigdeli and M. Haeri, "Predictive functional control for active queue management in congested TCP/IP networks," *ISA Trans.*, vol. 48, pp. 21–107, 12 2008.
- [16] P. Wang, H. Chen, X. Yang, and Y. Ma, "Design and analysis of a model predictive controller for active queue management," *ISA Trans.*, vol. 51, no. 1, pp. 120–131, Jan. 2012.
- [17] P. Wang, D. Zhu, and X. Lu, "Active queue management algorithm based on data-driven predictive control," *Telecommun. Syst.*, vol. 64, no. 1, pp. 103–111, Jan. 2017.
- [18] Y. Fan, Z.-P. Jiang, and S. S. Panwar, "An adaptive control scheme for stabilizing TCP," in *Proc. 5th World Congr. Intell. Control Autom.*, Jun. 2004, pp. 1410–1414.
- [19] J. Sun, M. Zukerman, and M. Palaniswami, "A stable adaptive PI controller for AQM," in *Proc. Int. Symp. Commun. Inf. Technol.*, Oct. 2007, pp. 707–712.
- [20] R. Barzamini, M. Shafiee, and A. Dadlani, "Adaptive generalized minimum variance congestion controller for dynamic TCP/AQM networks," *Comput. Commun.*, vol. 35, no. 2, pp. 170–178, Jan. 2012.
- [21] Y. Su, L. Huang, and C. Feng, "QRED: A Q-learning-based active queue management scheme," *J. Internet Technol.*, vol. 19, no. 4, pp. 1169–1178, 2018.
- [22] K. Wang, Y. Liu, X. Liu, Y. Jing, and S. Zhang, "Adaptive fuzzy funnel congestion control for TCP/AQM network," *ISA Trans.*, vol. 95, pp. 11–17, Dec. 2019.
- [23] M. Baklizi, "Stabilizing average queue length in active queue management method," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 3, pp. 77–83, 2019.
- [24] R. Patel and D. Deb, "Adaptive backstepping control of single chamber microbial fuel cell," in *Proc. 17th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2017, pp. 574–579.
- [25] R. Patel, D. Deb, H. Modi, and S. Shah, "Adaptive backstepping control scheme with integral action for quanser 2-dof helicopter," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2017, pp. 571–577.
- [26] R. Patel and D. Deb, "Parametrized control-oriented mathematical model and adaptive backstepping control of a single chamber single population microbial fuel cell," *J. Power Sources*, vol. 396, pp. 599–605, Aug. 2018.
- [27] Y. Wang, H. Shen, and D. Duan, "On stabilization of quantized sampled-data neural-network-based control systems," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3124–3135, Oct. 2017.
- [28] Y. Wang, W. Zhou, J. Luo, H. Yan, H. Pu, and Y. Peng, "Reliable intelligent path following control for a robotic airship against sensor faults," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2572–2582, Dec. 2019.



QIN XU received the B.S. and Ph.D. degrees from the Nanjing University of Science and Technology, in 2009 and 2015, respectively. She is currently a Lecturer with the School of Network and Communications Engineering, Jinling Institute of Technology. Her current research interests include big data, network congestion control, and quality control.



GAOYAO MA received the B.E. and M.D. degrees from the Nanjing University of Science and Technology, Nanjing, China, in 2011 and 2014, respectively. She is currently in intellectual property work with Jiangsu Sundry Law Firm, Nanjing. She has engaged in research and development of network equipment for nearly two years. Her research interest includes network congestion.



KAIMENG DING received the B.E. degree in computer science and technology from Jiangsu Ocean University, Lianyungang, China, in 2009, and the Ph.D. degree in geographic information system (GIS) from Nanjing Normal University, Nanjing, China, in 2015. He is currently an Associate Professor with the School of Networks and Communications Engineering, Jinling Institute of Technology. His current research interests include network reliability, network information security, and application of deep learning.



BIN XU received the bachelor's degree in electrical engineering from Donghua University, Shanghai, China, in 2011, and the master's degree in engineering management from Shanghai Jiaotong University, in 2017. He is currently a Engineer with Shanghai Shenzhou Engineering Company Ltd. He is currently engaging in project management of financial network industry. His current research interests include data, network congestion control, and quality control.

...