

Minimax based congestion control for TCP network systems with UDP flows

Yuanwei Jing^{1,*}, Zanhua Li^{1,2}, and Georgi Dimirovski^{3,4}

¹College of Information Science and Engineering, Northeastern University, Shenyang, 110004, China

²School of science, Shenyang Ligong University, Shenyang, 110168, China

³School of Engineering, Dogus University of Istanbul, Acibadem, Istanbul, 34722, Turkey

⁴School FEIT, St Cyril and St Methodius University, Karpos 2 BB, Skopje, 1000, Macedonia

Abstract. The congestion control problem for TCP network systems with UDP flows is considered. A nonlinear TCP network model with strict-feedback structure is established. The unknown UDP flow is considered as the disturbance to the system, and the maximum UDP flow is calculated by using the minimax approach. And then, a congestion control algorithm is proposed by using backstepping approach. Further, a state-feedback congestion controller is presented to make the TCP networks asymptotically stable. The simulation results show the superiority and feasibility of the proposed method.

Keywords. TCP network, congestion control, interference, minimax, backstepping

1 Introduction

With the rapid development of Cloud Computing and Internet of Things, the data has increased sharply in the word. So, the communication networks based on TCP have developed rapidly and been applied widely. The TCP protocol is regarded as the core of computer and communication network system, and thus in depth exploration of TCP networks becomes more important [1-2]. With the ever expanding scale of communication networks, the network congestion is also becoming increasingly serious [2-3]. Thus the issues of congestion control problems for network systems have become particularly important nowadays.

In turn, in order to cope with the traffic congestion in networks, the strategy of active queue management (AQM) is being used to reduce the packet loss rate, improve the coefficient of link utilization, reduce the queuing delay, and inhibit the oscillation rate, etc. The congestion control is one of the important means to solve the network congestion problem [3-7].

From the point of view of control theory, the network system model can be regarded as a feedback control system. At present, aiming at the transient response, stability and steady state error control of the network system, a number of research results have been obtained for overcoming network congestion problems. Among them are PI, PD and PID control [8-11]. The congestion control mechanism of the TCP protocol is considerably complex, in which many nonlinear distortion factors appear.

In view of the presence of nonlinearities, most of the existing AQM algorithms are designed on the grounds of the linearized system model at the operating equilibrium

point, i.e., the desired steady-state equilibrium. Though there are two restricting limitations to such controllers. One is that the linearization near the working point yields an approximation, the result of which the asymptotic stability is of local nature; hence the global stability behaviour cannot be extrapolated. The other one is that, after linearization, some essential nonlinear phenomena may will be lost. Therefore both the analysis and the synthesis usually deviate more or less from the objective facts.

As a net consequence, the AQM algorithm designed by using a linear control methodology could not capture the nonlinear dynamic nature of TCP networks. And typically the nonlinearity of network systems cannot be compensated properly. On the other hand, it is difficult to resist the load disturbance and the interference of unresponsive user-defined properties (UDP) traffic flows. Therefore the controller parameters are difficult to be set, the robustness is poor, and also the algorithmic stability of the overall network system is difficult to guarantee. In such case, the approach of proper choice is to design the AQM algorithm directly on the grounds of the nonlinear network system [12-13]. One of the promising methods to design the controller directly by using the nonlinear system model of the communication network traffic is the backstepping method [14-15].

While the research of congestion control strategy has made great progress, there are still some problems that have not been fully solved. One of the most important issues is to deal with the uncertainty and interference of the network system. Generally, there are two kinds of approach to deal with the interference. One is directly giving out the upper bound of the interference. However, this kind of approaches is not consistent with the actual

* Corresponding author: ywjing@mail.neu.edu.cn

situation, for the interference is often difficult to measure accurately and thus often uncertain. It is also difficult to determine the upper bound of the uncertainty of the network system. The other is shrinking the interference items in the process of designing controllers, which will strengthen the assumptions.

The idea of minimax optimum control can be used to avoid the limitations and conservatism of the methods mentioned above, which is an effective method to suppress interference [16-17]. Compared with other existing methods, the minimax control method is aimed at calculate the worst case interference and uncertainty, and then discuss how to achieve the controlled stability and performance of the system. This is being carried out by making the integrated index minimum with regard to the system state error, the control energy consumption, the interference, and the uncertainty during the whole processing time.

A nonlinear AQM algorithm based on backstepping and minimax is proposed in this paper on the grounds of a suitable nonlinear dynamic model of TCP network system with actual situation of UDP flow interference. The rest of this paper is arranged as follows. Section 2 illustrates the representation of TCP network with unresponsive UDP flows. The design of backstepping controller based on the minimax idea is presented in Section 3. Then in Section 4, there are presented the simulation experiments with two illustrative examples. Thereafter conclusion and references are presented.

2 State space representation of network

Consider the dynamic model of TCP network system as follows [8].

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

where, $W(t)$ is the window size of TCP source, $q(t)$ is the instantaneous queue length of the router, $N(t)$ is the TCP network load, C is the link bandwidth being a constant, $R(t)$ is the round-trip time and satisfies

$$R(t) = \frac{q(t)}{C} + T_p,$$

T_p is the propagation delay, and $p(t)$ is the probability of congestion indication at the update step and takes value in the interval $[0,1]$.

In order to use the control theory to discuss the congestion control problem, the state space expression of the TCP network should be given. Taking the queue length and the window size as the state variables and the congestion indication probability as the control input of the network system, let

$$x_1 = q(t) - q_d, \quad x_2 = W(t), \quad u(t) = p(t).$$

where q_d is the expected queue length.

Denote $\mathbf{x}=(x_1, x_2)^T$. Then, neglecting system delay, the system (1) can be rewritten in the state equation as

$$\begin{aligned} \dot{x}_1 &= a(t)x_2 - C \\ \dot{x}_2 &= f(t) + g(t, \mathbf{x})u(t) \\ y &= kx_1 \end{aligned} \quad (2)$$

where, k is the output parameter, and

$$a(t) = \frac{N(t)}{R(t)}, \quad f(t) = \frac{1}{R(t)}, \quad g(t, \mathbf{x}) = -\frac{x_2^2}{2R(t)},$$

The system of differential equation (2) is in the lower-triangular form, which can be dealt with by using the back-stepping control design method [14].

If we consider the situation with the UDP flow interference [1], an interference item should be added to the above system representation model. Then we have

$$\begin{aligned} \dot{W}(t) &= \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t)} p(t-R(t)) + \rho U(t, W, q) \\ \dot{q}(t) &= \frac{N(t)}{R(t)} W(t) - C \end{aligned} \quad (3)$$

where, $U(t, W, q)$ is the UDP flow interference and ρ is the interference coefficient. The state space model of (3) is as follows.

$$\begin{aligned} \dot{x}_1 &= a(t)x_2 - C \\ \dot{x}_2 &= f(t) + g(t, \mathbf{x})u(t) + \rho\omega(t, \mathbf{x}) \\ y &= kx_1 \end{aligned} \quad (4)$$

where term $\omega(t, \mathbf{x})$ represents the system disturbance, *i.e.* the UDP flow interference $U(t, r, q)$.

The purpose of the research is to design a controller to make x_1 and x_2 tending to their expected values, *i.e.*, stabilizing the TCP network system (2) or (4) at their equilibrium points.

3 Backstepping controller design

In this section, we deal with the problem on designing backstepping controller to the system (2) and system (4) respectively.

3.1 Case without interference

According to the design idea of the backstepping method, firstly consider the subsystem with state variable x_1 of the system (2). Define the state transformation as follows

$$\begin{aligned} e_1 &= x_1 \\ e_2 &= x_2 - x_2^* \end{aligned} \quad (5)$$

Then the subsystem becomes

$$\begin{aligned} \dot{e}_1 &= \dot{x}_1 = a(t)x_2 - C \\ &= a(t)(e_2 + x_2^*) - C \end{aligned} \quad (6)$$

where, x_2^* represents the virtual control variable of the subsystem (6), e_2 represents the error variable between the system state x_2 and the virtual control x_2^* .

3.1.1 Step 1

The aim of this step is to design virtual feedback control x_2^* making $e_1 \rightarrow 0$. Thus select the following Lyapunov function

$$V_1 = \frac{1}{2}\alpha e_1^2, \quad \alpha > 0. \quad (7)$$

The derivative of V_1 to the time t along the trajectory of the subsystem (6) is

$$\begin{aligned} \dot{V}_1 &= \alpha e_1 \dot{e}_1 \\ &= \alpha e_1 [a(t)(e_2 + x_2^*) - C] \\ &= \alpha a(t)e_1 e_2 + \alpha a(t)e_1 x_2^* - \alpha e_1 C. \end{aligned} \quad (8)$$

Choose an appropriate virtual control law as

$$x_2^* = -\frac{1}{a(t)}(l_1 e_1 + C) \quad (9)$$

to make $\dot{V}_1 < 0$, where l_1 is a positive design parameter.

Substituting (9) into (8) yields

$$\dot{V}_1 = -l_1 \alpha e_1^2 + \alpha a(t)e_1 e_2. \quad (10)$$

If $e_2 = 0$, we can see that

$$\dot{V}_1 = -l_1 \alpha e_1^2 < 0$$

and the state e_1 is asymptotically stable. In general, the state e_2 cannot be guaranteed to be zero. So we need to consider the subsystem with state variable x_2 and design the control u to make the error state variable e_2 having the expected asymptotic stability.

3.1.2 Step 2

In this step, we design a feedback control u making the state equation

$$\begin{aligned} \dot{e}_2 &= \dot{x}_2 - \dot{x}_2^* \\ &= f(t) + g(t, \mathbf{x})u(t) + l_1 x_2 - \frac{C}{a(t)} \end{aligned} \quad (11)$$

asymptotically stable at $e_2 = 0$.

Construct a Lyapunov function

$$V_2 = V_1 + \frac{1}{2}\beta e_2^2, \quad \beta > 0. \quad (12)$$

The derivative of V_2 to the time t along the trajectory of the system (6) and (11) is

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + \beta e_2 \dot{e}_2 \\ &= -l_1 \alpha e_1^2 + \alpha a(t)e_1 e_2 \\ &\quad + \beta e_2 [f(t) + g(t, \mathbf{x})u(t) + l_1 x_2 - \frac{C}{a(t)}] \\ &= -l_1 \alpha e_1^2 + \beta e_2 [\frac{\alpha}{\beta} a(t)e_1 + f(t)] \\ &\quad + \beta e_2 [g(t, \mathbf{x})u(t) + l_1 x_2 - \frac{C}{a(t)}]. \end{aligned} \quad (13)$$

If the control law is designed as

$$u(t) = -\frac{1}{g(t, \mathbf{x})} [\frac{\alpha}{\beta} a(t)e_1 + f(t) + l_1 x_2 - \frac{C}{a(t)} + l_2 e_2] \quad (14)$$

where l_2 is a positive design parameter. We can get

$$\dot{V}_2 = -l_1 \alpha e_1^2 - l_2 \beta e_2^2 < 0 \quad (15)$$

From the analytic induction above, we know that the system (2) is asymptotically stable under the designed controller, and then the system (1) realizes the active queue management congestion control.

3.2 Case with UDP flow interference

Consider the state space model of TCP network system with UDP flow interference described in equation (4). We also use the backstepping method to deal with the problem.

Consider the subsystem with state variable x_1 of the system (4) firstly. Define the state transformation as same as in equation (5). The subsystem becomes

$$\dot{e}_1 = a(t)x_2 - C = a(t)(e_2 + x_2^*) - C \quad (16)$$

where, x_2^* represents the virtual control variable of the subsystem (16). The meaning of e_2 is as same as in (6). The backstepping design process is divided into two-step stages.

The first step is to design the virtual control to the subsystem (16). The design idea is similar to that of subsection 3.1.1, and it is omitted here.

We Proceed straight into the second step design. Our aim is to design a feedback control law u making the state equation

$$\dot{e}_2 = \dot{x}_2 - \dot{x}_2^* = f(t) + g(t, \mathbf{x})u(t) + \rho \omega(t, \mathbf{x}) + l_1 x_2 - \frac{C}{a(t)} \quad (17)$$

asymptotically stable at $e_2 = 0$.

Construct a Lyapunov function

$$V_2 = V_1 + \frac{1}{2} \beta e_2^2, \quad \beta > 0. \quad (18)$$

The derivative of V_2 to the time t along the trajectory of the system (16) and (17) is

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + \beta e_2 \dot{e}_2 \\ &= -l_1 \alpha e_1^2 + \alpha a(t) e_1 e_2 \\ &\quad + \beta e_2 [f(t) + g(t, x)u(t) + \rho \omega(t, x) + l_1 x_2 - \frac{C}{a(t)}] \\ &= -l_1 \alpha e_1^2 + \beta e_2 [\frac{\alpha}{\beta} a(t) e_1 + f(t) + g(t, x)u(t)] \\ &\quad + \beta e_2 [\rho \omega(t, x) + l_1 x_2 - \frac{C}{a(t)}]. \end{aligned} \quad (19)$$

The unknown interference term $\omega(t, x)$ appears in equation (19). So we cannot directly design the control law like as equation (14). We have to deal with the disturbance before designing controller. There are two ways of interference treatment generally. One is that the upper bound of interference is assumed artificially. But the interference is often difficult to measure accurately and is often uncertain. Therefore, such way may not be in accordance with physical reality.

The other way is to reduce the interference related items in the Lyapunov function during the design of controller, which will lead to hypothetical condition enhancement.

To avoid the limitations and conservatism mentioned above, prior to the design of the controller, we first deal with the interference terms based on the minimax theory. That is, the worst effect of interference on the system will be calculated.

Consider the following index performance

$$J = \frac{1}{2} \int_0^\infty (y^2 - \gamma \omega^2) dt \quad (20)$$

where, $\gamma > 0$ is the interference suppression constant. In order to calculate the worst interference, the following critical function is constructed

$$\phi = \dot{V}_2 + \frac{1}{2} (y^2 - \gamma \omega^2) \quad (21)$$

Substituting (19) into (21) yields

$$\begin{aligned} \phi &= -l_1 \alpha e_1^2 + \beta e_2 [\frac{\alpha}{\beta} a(t) e_1 + f(t) + g(t, x)u(t)] \\ &\quad + \beta e_2 [\rho \omega(t, x) + l_1 x_2 - \frac{C}{a(t)}] + \frac{1}{2} (y^2 - \gamma \omega^2). \end{aligned} \quad (22)$$

The first derivative of ϕ about ω is

$$\frac{\partial \phi}{\partial \omega} = \beta \rho e_2 - \gamma \omega. \quad (23)$$

Let it be zero. We obtain

$$\omega^*(t, \mathbf{x}) = \frac{\beta \rho}{\gamma} e_2. \quad (24)$$

Then we can get the second derivative of ϕ , which is less than zero.

$$\frac{\partial^2 \phi}{\partial \omega^2} = -\gamma < 0. \quad (25)$$

It can be seen, from the discussion above, that ϕ has the maximum value at $\omega^*(t, \mathbf{x})$, i.e.

$$\max_{\omega} \phi = \dot{V}_2(\omega^*) + \frac{1}{2} (y^2 - \gamma (\omega^*)^2). \quad (26)$$

For the critical function (21), the maximum value is

$$\max_{\omega} \phi = \max_{\omega} [\dot{V}_2 + \frac{1}{2} (y^2 - \gamma \omega^2)]. \quad (27)$$

Integrating both sides of (27) simultaneously yields

$$\max \int_0^\infty \phi dt = \max [\int_0^\infty \dot{V}_2 dt + \frac{1}{2} \int_0^\infty (y^2 - \gamma \omega^2) dt] \quad (28)$$

Let

$$\bar{\phi} = \int_0^\infty \phi dt \quad (29)$$

Then

$$\max_{\omega} \bar{\phi} = \max_{\omega} [V_2(\infty) - V_2(0) + J] \quad (30)$$

It means that $\max_{\omega} \bar{\phi}$ is equivalent to $\max_{\omega} J$. Therefore, $\omega^*(t, \mathbf{x})$ that makes the critical function ϕ get the maximum also makes the index performance J get the maximum. Then we can say that $\omega^*(t, \mathbf{x})$ is the worst disturbance to the system.

Substituting the worst interference $\omega^*(t, \mathbf{x})$ into the derivative of V_2 , the equation (19), we have

$$\begin{aligned} \dot{V}_2 &= -l_1 \alpha e_1^2 + \beta e_2 [\frac{\alpha}{\beta} a(t) e_1 + f(t) + g(t, \mathbf{x})u(t)] \\ &\quad + \beta e_2 [\frac{\beta \rho^2}{\gamma} e_2 + l_1 x_2 - \frac{C}{a(t)}]. \end{aligned} \quad (31)$$

If the control law is designed as

$$\begin{aligned} u(t) &= -\frac{1}{g(t, \mathbf{x})} [\frac{\alpha}{\beta} a(t) e_1 + f(t) + \frac{\beta \rho^2}{\gamma} e_2] \\ &\quad - \frac{1}{g(t, \mathbf{x})} [l_1 x_2 - \frac{C}{a(t)} + l_2 e_2] \end{aligned} \quad (32)$$

where l_2 is a positive design parameter, we can get

$$\dot{V}_2 = -l_1 \alpha e_1^2 - l_2 \beta e_2^2 < 0 \quad (33)$$

From the above presented induction analytic synthesis, we know that the system (4) is asymptotically stable under the designed controller, and then the system (3) with the UDP flow interference has realized the active queue management congestion control.

Theorem 1. To any given $\gamma > 0$, the system (4) is asymptotically stable under the control law (32).

Corollary 1. If the control law (32) is designed properly, the system (3) can achieve the expected queue length and the proper window size.

4 Simulation experiments

The effectiveness of the proposed minimax based backstepping approach is illustrated by two simulation examples in this section. The first one is provided to check the validity of the results in the stability context, while the second one explains the superiority of the designed nonlinear active queue management controller compared with PI controller.

4.1 Illustrative example 1

The system parameters considered in the simulation is as follows.

$$N = 60, C = 3750 \text{ packets/s}, q_d = 375 \text{ packets}, \\ \gamma = 0.2, \alpha = 0.2, \beta = 0.25, l_1 = 0.5, l_2 = 1.$$

We assume that the network system has different round-trip delays, and is divided into four parts under the multi-protocols.

In the case of the same propagation delay $T_p = 0.3s$, we perform the queue length simulations for four different round-trip delays, *i.e.* 250ms, 275ms, 300ms, and 325ms, respectively.

The simulation results are drawn in Fig. 1 to Fig. 4.

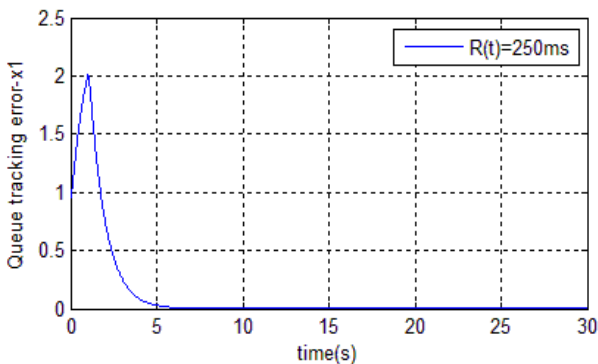


Fig. 1. Queue tracking error in 250ms of round-trip delay.

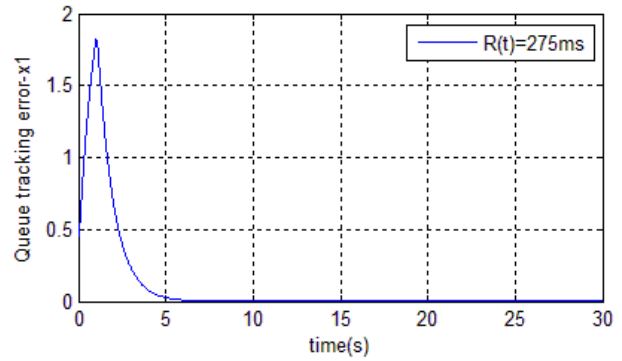


Fig. 2. Queue tracking error in 275ms of round-trip delay.

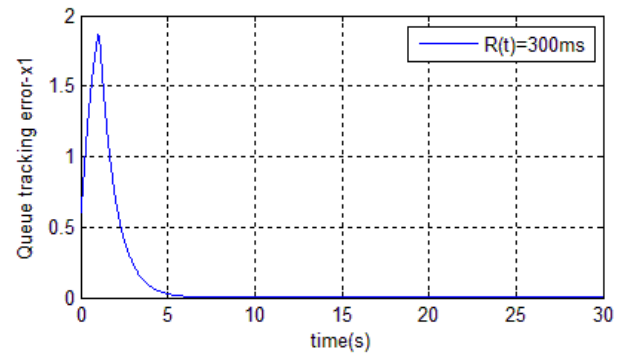


Fig. 3. Queue tracking error in 300ms of round-trip delay.

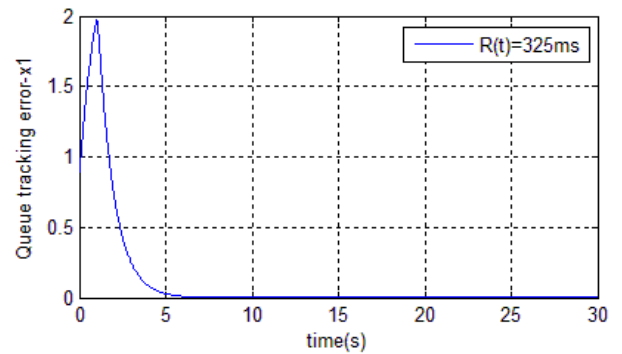


Fig. 4. Queue tracking error in 325ms of round-trip delay.

From the figures, we can see that although the different round-trip delays produce the different values for queue length errors, the perturbations are still in the ideal range, and the queue length errors converge to zero in a short time.

4.2 Illustrative example 2

The system parameters considered in the simulation is as follows.

$$N = 60, C = 3750 \text{ packets/s}, q_d = 375 \text{ packets}, \\ W_0 = 8.333, \alpha = 0.2, \beta = 0.25, \gamma = 0.2, l_1 = 0.5, l_2 = 1.$$

where W_0 is the expected window size. And the values of parameters for PI controller are given as in existing works.

$$k_p = 0.5, \quad k_i = 0.8.$$

Fig. 5 and Fig. 6 depict the simulation results of queue length and the window size, respectively.

From Fig. 5, we can see that, in the first 5 seconds of the beginning, the error of queue length under the designed minimax-backstepping (MB) controller is smaller than PI controller, and then converges to 0 faster.

Fig. 6 shows that, during the first 5 seconds of the beginning, the perturbation of the window size under the designed minimax-backstepping controller is much smaller than PI controller, and it is in the ideal range.

The reason for the difference is caused by linearizing the nonlinear network system.

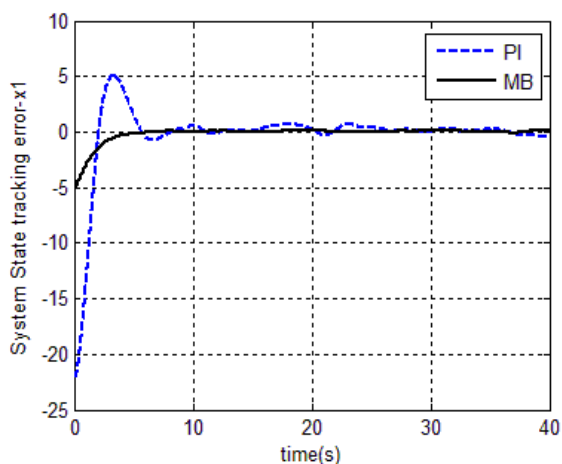


Fig. 5. Queue tracking errors under MB and PI controllers.

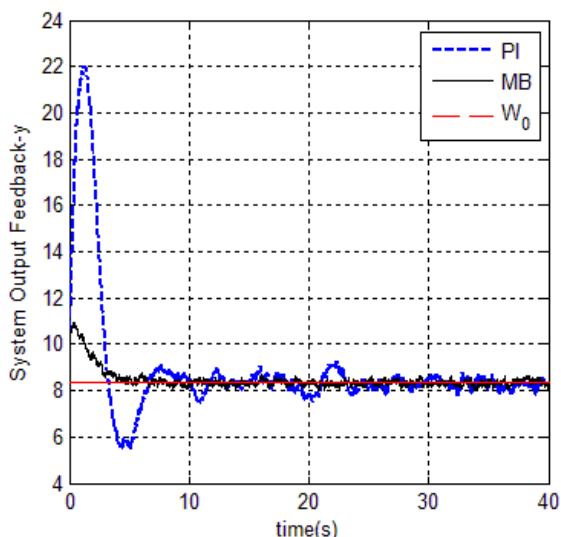


Fig. 6. Window size under MB and PI controllers.

The controller obtained by minimax method has good suppression effect on the UDP flow interference. And the minimax-backstepping controller makes the queue length and the window size in the TCP network system reaching the expected value very quickly.

5 Conclusions

In this work, the congestion control problem is discussed to a class of nonlinear TCP network system by using backstepping technology with the minimax control method. An AQM control algorithm is presented to avoid the influence of retracting interference terms and neglecting feature of systems. To the worst interference calculated with the minimax method, a robust controller is designed. The simulation results show that the method proposed has strong robustness, and can get the faster system response and smaller overshoot.

The research work is supported by the National Natural Science Foundation of China (61773108). In part, it is also supported by Dogus University Fond for Science.

References

1. A. Leon-Garcia, I. Widjaja, *Communication Net-works: Fundamental Concepts and Key Architectures* (New York, NY: McGraw-Hill 2000)
2. ATM Forum, Technical Committee, *Traffic Management Specification* (Version 4.1, af-tm-0121.000, 43-55, 1999)
3. O. C. Imer, T. Basar, *Eur. J. Contr.* **7**, 2-3 (2001)
4. T. Ren, C. Wang, X. Luo, Y. Jing, G.M. Dimirovski, *Int. J. Innovative Computing, Inf. and Contr.* **5**, 8 (2009)
5. F. Zheng, J. Nelson, *Automatica* **45** (2009)
6. Y. Liu, X. P. Liu, Y. W. Jing, S. W. Zhou, *ISA Transactions* **72** (2018)
7. N. El Fezazi, F. El Haoussi, E. H. Tissir, T Alvarez, *J. Franklin Institute* **354**, 17 (2017)
8. C. V. Hollot, V. Misra, D. Towsley, *IEEE Trans. Automa. Contr.* **47** (2002)
9. H. U. Unal, D. Melchor-Aguilar, D. Ustebay, S. I. Niculescu, H. Ozbay, *Comput. Communications* **36** (2013)
10. P. Wang, D. J. Zhu, X. H. Lu, *Telecom. Systems* **64** (2017)
11. Q. Chen, O. W. Yang, *IEEE Trans. Automa. Contr.* **52** (2009)
12. W. J. Changa, Y. T. Menga, K. H. Tsai, *Intl. J. Systems Science* **43**, 12 (2012)
13. E. Abolfazli, V. Shah-Mansouri, *IEEE Int. Conf. Mobile & Wireless Networks*, 1840-1844 (2015)

14. M. Krstic, I. Kanellakopoulos, P. Kokotovic, *Nonlinear and Adaptive Control Design* (New York, NY: Wiley-Interscience, 1995)
15. P. V. Kokotovic, M. Arcak, *Automatica* **37**, 637-662 (2001).
16. M. M. Kogan, *IEEE Trans. Automa. Contr.* **43**, 5 (1998)
17. X. D. Yuan, Y. W. Jing, *Neural Comput. Appl.* **25** (2014)