Steady State and Transient Behavior Analyses of TCP Connections considering Interactions between TCP connections and Network

Hisamatu Hiroyuki
Graduate School of Engineering Science, Osaka University
E-mail: hisamatu@ics.es.osaka-u.ac.jp

Hiroyuki Ohsaki
Graduate School of Information Science and Technology, Osaka University
E-mail: oosaki@ist.osaka-u.ac.jp

Masayuki Murata
Cybermedia Center, Osaka University
E-mail: murata@cmc.osaka-u.ac.jp

Abstract

The Internet uses a window-based flow control mechanism in TCP (Transmission Control Protocol). In the literature, there have been a significant number of analytical studies on TCP. Most of those studies, however, have focused on the statistical behavior of TCP by assuming a constant packet loss probability in the network. In our previous work, we have presented an approach for modeling the network as a single feedback system using the fluid flow approximation and the queuing theory. In this paper, by utilizing and extending our previous work, we analyze the steady state behavior and the transient behavior of TCP. We first derive the throughput and the packet loss probability of TCP, and the number of packets queued in the bottleneck router. We then analyze the transient behavior of TCP using a control theoretic approach, showing the influence of the number of TCP connections and the propagation delay on its transient behavior of TCP. Through numerical examples, it is shown that the bandwidth–delay product of a TCP connection significantly affects its stability and transient performance. It is also shown that, contrary to one's intuition, the network becomes more stable as the number of TCP connections and/or the amounts of background traffic increases.

1 Introduction

The Internet uses a window-based flow control mechanism in TCP (Transmission Control Protocol), which is a sort of feedback based congestion control mechanisms [8]. TCP has two fundamental mechanisms: a packet retransmission mechanism and a congestion avoidance mechanism. The packet retransmission mechanism re-sends lost packets in the network for realizing reliable data transfer between source and destination hosts. The congestion avoidance mechanism controls the packet emission process from a source host according to the congestion status for utilizing network resources effectively.

In our previous work [6], we have proposed an approach for modeling the network including the TCP mechanism at a source host as a single feedback system. Our approach was to model the TCP congestion control mechanism and the network seen by TCP using the fluid flow approximation and the queuing theory. The congestion control mechanism of TCP changes the window size according to the occurrence of packet losses in the network. We have therefore modeled it as a SISO (Single-Input and Single-Output) system, where the input to the system is a measured packet loss probability in the network and the output is a TCP window size. On the other hand, when the number of packets entering the network increases, some packets are awaited in the buffer of the bottleneck router (i.e., the router connected to the bottleneck link). This causes a tendency of a high packet loss probability at the bottleneck router. We have therefore modeled the network seen by TCP as another SISO system, where the input to the system is the TCP window size and the output is the packet loss probability in the network. For modeling the TCP mechanism, we have compared four analytic approaches proposed in [13, 15, 18]. For modeling the network, we have used a $M/M/1/m$ queue where existence of the background traffic is taken account of.

In this paper, by extending the modeling approach proposed in [6], we analyze the steady state and the transient behavior of TCP. We first derive the throughput of each TCP connection, the packet loss probability at the bottleneck router, and the average queue length (i.e., the number of packets awaited in the buffer) at the bottleneck router. In the literature, there have been a great number of analytical studies on TCP (e.g., [2, 3, 4, 5, 7, 9, 10, 11, 12, 16, 17]). Most of those studies assume a constant packet loss probability in the network, and derive the throughput of TCP connections [2, 3, 11, 16] or the distribution of window sizes of TCP connections [9, 17, 10]. In this paper, we use a different approach, which models the interaction between TCP and the network more explicitly than those studies. In [4, 3], the authors have derived the average file transfer time [4] or the average queuing delay [3] without assuming a constant packet loss probability in the network for a network with multiple bottlenecks. However, the stability and the transient behavior of TCP have not been analyzed. In [5], the authors have modeled a RED (Random Early Detection) gateway as a feedback control system. However, they have only focused on the stability regions and the configuration of the RED gateway. In [12], the authors have modeled the interactions of a set of TCP flows and AQM (Active Queue Management) gateways, and they have showed a transient behavior of TCP. Since their methodology is based on the solution of the differential equations, they have not showed the transient behavior of TCP, rigorously. In [7], the authors have analyzed a combined TCP and AQM (Active Queue Management) gateways model from a control theoretic standpoint. However, they have only focused on the stability of the queue length of RED gateways. Hence, by utilizing the control theory, which has been developed in the control engineering, we analyze the transient behavior of TCP. We then show quantitatively how the stability and the transient behavior of TCP are affected by several system parameters: the number of TCP connections, the propagation delay, the bottleneck link capacity, and the buffer size of the bottleneck router.

Organization of this paper is as follows. In Section 2, we briefly explain the modeling approach proposed in [6], which uses the fluid flow approximation and the queuing theory. In Section 3, we derive the TCP throughput, the packet loss probability, and the queue length of the bottleneck router. We also validate our approximate analysis by comparing analytic results with simulation ones. In Section 4, we analyze the transient behavior of TCP using a control theoretic approach. Finally, in Section 5, we conclude the current paper and discuss future works.

2 Analytic Model and Derivation of State Transmission Equations

In this paper, we analyze the stability and the transient behavior of TCP Reno by extending the modeling approach proposed in [6]. In what follows, we briefly explain the modeling approach. For more detail, refer to [6].

Figure 1 illustrates our analytic model. We model the entire network, including TCP mechanisms running on source hosts, as a single feedback system, where the congestion control mechanism of TCP and the network seen by TCP interact each other. We separately model the TCP congestion control mechanism and the network by two discrete-time SISO systems; that is, the TCP congestion control mechanism and the network change their states at every unit time, which corresponds to the interval of two successive ACK packet arrivals at a source host. Table 1 summarizes the definition of symbols used throughout in this paper.

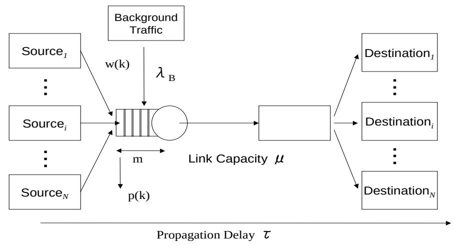


Figure 1: Analytic model.

Table 1. Definition of Symbols.

N	:	the number of TCP connections
μ	:	the bottleneck link capacity
τ	:	propagation delay between a source host and a destination host
λ_B	:	arrival rate of the background traffic
m	:	buffer size of the bottleneck router
$w(k)$	:	window size at slot k
$p(k)$	:	packet loss probability at slot k

The congestion control mechanism of TCP changes its window size according to the occurrence of packet losses in the network. Hence, we model the TCP congestion control mechanism as a SISO (Single-Input and Single-Output) system, where the input to the system is a packet loss probability in the network and the output is the TCP window size. We model the TCP congestion control mechanism using the fluid flow approximation [6]. Namely, the TCP window size $w(k)$ is updated according to the observed packet loss probability $p(k)$ in the network at every receipt of an ACK packet: i.e.,

$$\begin{aligned} w(k+1) &= w(k) + \frac{1-p(k+1-w(k))}{w(k)} \\ &\quad - \frac{(1-\hat{Q}(w(k),p(k)))\, p(k+1-w(k))\, w(k)}{2} \\ &\quad - p(k+1-w(k))\, \hat{Q}(w(k),p(k)) \qquad (1) \end{aligned}$$

In the above equation, $\hat{Q}(w,p)$ is a probability that the source host fails to detect one or more packet losses from duplicate ACKs, when the window size is w and the packet loss probability is p [18].

$$\hat{Q}(w,p) = \frac{(1-(1-p)^3)\,(1+(1-p)^3\,(1-(1-p)^{w-3}))}{(1-(1-p)^w)}$$

TCP TCP ł 4 ł TCP M/M/1/m [6]

On the other hand, when the number of packets entering the network increases, some packets are awaited in the buffer of the bottleneck router. This often causes a high packet loss probability. Therefore, we model the network seen by TCP as another SISO system, where the input to the system is the TCP window size and the output is the packet loss probability. We model the network by a $M/M/1/m$ queue, where existence of the background traffic is taken account of [6]. Namely,

$$p(k) = \frac{(1-\rho(k))\,\rho(k)^m}{1-\rho(k)^{m+1}} \tag{2}$$

where $\rho(k)$ and $r(k)$ are given by

$$\begin{aligned} \rho(k) &= \frac{1}{\mu}\left(\frac{N\,w(k)}{r(k)} + \lambda_B\right) \\ r(k) &= 2\tau + \frac{\rho(k)\,(1-m\,\rho(k)^m + m\,\rho(k)^{m+1})}{\mu(1-\rho(k)^{m+2})\,(1-\rho(k))} \end{aligned} \tag{3}$$

Note that, for simplicity, the above equations assume that propagation delays of all TCP connections are identical, and that window sizes of all TCP connections are changed synchronously. TCP [18]

3 Steady State Analysis

In this section, we derive the TCP throughput, the packet loss probability, and the average queue length in steady state using the state transition equations derived in Section 2. We then validate our approximate analysis by comparing analytic results with simulation ones.

The congestion control mechanism of TCP is an AIMD (Additive Increase and Multiplicative Decrease) based feedback control. When the propagation delay is non-negligible, the window size oscillates and never converges to a constant value. Note that the symbol $w(k)$ represents not the instant value of the oscillating TCP window size but the expected value of the TCP window size after a long period.

Let equilibrium values of the TCP window size and the packet loss probability in the network be w^* and p^*, respectively; i.e.,

$$w^* \equiv \lim_{k\to\infty} w(k) \tag{4}$$

$$p^* \equiv \lim_{k\to\infty} p(k) \tag{5}$$

These values can be numerically obtained by solving Eqs. (1) and (2) with equating $w(k+1) \equiv w(k)$ and $p(k+1) \equiv p(k)$. Using these equilibrium values, the TCP throughput T and the average queue length of the bottleneck router L are given by

Table 2. parameter values

$N = 10$	$\mu = 2$ [packet/ms]
$\tau = 30$ [ms]	$\lambda_B = 0.2$ [packet/ms]
$m = 50$ [packet]	packet size = 1000 [byte]

$$T = \frac{w^*}{r^*} \tag{6}$$

$$\begin{aligned} L &= \rho^*\mu(r^*-2\tau) \\ &= \frac{\rho^{*2}\,(1-m\,\rho^{*m} + m\,\rho^{*(m+1)})}{(1-\rho^{*(m+2)})\,(1-\rho^*)} \end{aligned} \tag{7}$$

where r^* and ρ^* are the equilibrium values of $r(k)$ and $\rho(k)$, respectively and decided by w^* and p^*. In the above equations, the TCP throughput T is approximated by the number of packet per a unit time emitted by source host, and the average queue length L is obtained from the number of customers waiting to be served of the M/M/1/m queue.

We next compare analytic results with simulation ones for validating our approximate analysis. In the following analytic results, we calculate the TCP throughput T, the packet loss probability p^*, and the average queue length L using Eqs. (6), (5), and (7), respectively. Using ns-2 simulator [1], we run several simulation experiments at a packet level for the same network model with Fig. 1. Each simulation experiment is continued for 24 seconds, and the last 20 seconds are used for calculating simulation results — the TCP throughput, the packet loss probability, and the average queue length. Each simulation experiment is repeated 50 times, and 95 % confidence intervals of all performance measures are calculated.

In obtaining the analytic and simulation results, we use the following parameters: the number of TCP connections $N = 10$, the bottleneck link capacity $\mu = 2$ [packet/ms], the propagation delay $\tau = 30$ [ms], the average arrival rate of the background traffic $\lambda_B = 0.2$ [packet/ms], and the buffer size of the bottleneck router $m = 50$ [packet]. In simulation experiments, The version of TCP is TCP Reno, and we model the background traffic by UDP packets, and the packet size of TCP and UDP packets is fixed at 1000 [byte]. And taking into account of Window-Scale option, the maximum window size is enough large value, 10000.

Table 2 summarizes parameters used in obtaining the analytic and simulation results. In simulation experiments, the background traffic is modeled by UDP traffic.

Figure 2 shows the TCP throughput, the packet loss probability, and the average queue length for the different bottleneck link capacities. For comparison purposes, another analytic result of the TCP throughput from [18] is shown in Fig. 2(a). In [18], the TCP throughput is derived as a function of the round-trip time and the packet loss probability for a TCP connection. More specifically, the TCP throughput T' derived in [18] is given by

$$T' = \frac{\frac{1-p}{p} + E[W] + \hat{Q}(E[W])\frac{1}{1-p}}{r\left(\frac{b}{2}E[W]+1\right) + \hat{Q}(E[W])T_o\frac{f(p)}{1-p}}$$

where

$$E[W] = \frac{2+b}{3b} + \sqrt{\frac{8(1-p)}{3bp} + \left(\frac{2+b}{3b}\right)^2}$$

$$\hat{Q}(w) = \frac{(1-(1-p)^3)(1+(1-p)^3(1-(1-p)^{w-3}))}{(1-(1-p)^w)}$$

$$f(p) = 1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6$$

In this paper, we calculate the TCP throughput from the above equation using the packet loss probability and the round-trip time obtained from the simulation. It can be found, in terms of the TCP throughput and the packet loss probability, both analytic and simulation results show a good agreement. In particular, in respect to the TCP throughput, it can be found that our analytic results show better agreement with the simulation results than the value obtained from the expression in [18]. However, in terms of the average queue length, it can be found that our analytic results are much smaller than simulation results. Such a disagreement between analytic and simulation results is probably caused by our assumption that the packet arrival at the bottleneck router follows a Poisson process. In running the simulation, the average arrival rate of the background traffic is fixed at $\lambda_B = 0.2$ [packet/ms]. Hence, the amount of the TCP traffic becomes relatively larger than the amounts of the background traffic as the bottleneck link capacity becomes large. As a result, the packet arrival process at the bottleneck router cannot be modeled by a Poisson process.

In Fig. 3, both analytic and simulation results are shown for different propagation delays. Similarly to the previous case, it can be found that both analytic and simulation results show a good agreement in terms of the TCP throughput. However, as the propagation delay increases, the packet loss probability obtained from our analysis deviates from the corresponding simulation result. It can also be found that, in respect to the average queue length of the bottleneck router, our analytic results are much smaller than simulation ones. Such disagreement between analytic and simulation results is probably caused by our assumption that the packet arrival at the bottleneck router follows a Poisson process. Since TCP uses a window-based flow control mechanism, the packet emission process from the source host becomes more clumpy as the propagation delay becomes large. Hence, as the propagation delay becomes large, the Poisson process becomes in sufficient for modeling the arrival process of the background traffic at the bottleneck router is.

4 Transient Behavior Analysis

In this section, we analyze the TCP behavior in the transient state using state transition equations derived in Section 2. Specifically, by applying the control theory, we show how the TCP window size and the packet loss probability converge to their equilibrium points.

Let $\mathbf{x}(k)$ be the difference between ($w(k)$, $p(k)$) and (w^*, p^*).

$$\mathbf{x}(k) \equiv \begin{bmatrix} w(k) & - & w^* \\ p(k) & - & p^* \end{bmatrix}$$

Since Eqs. (1) and (2) have non-linearity, we linearize them around their equilibrium points and write them in a matrix form

$$\mathbf{x}(k+1) = \mathbf{A}\,\mathbf{x}(k) \tag{8}$$

where **A** is a state transition matrix. Eigenvalues of the state transition matrix determine the stability and the transient behavior of the feedback system around the equilibrium point [14]. It is known that the system is stable if the maximum modulus is less than one. It is also known that the smaller the maximum modulus is, the better the transient behavior becomes. In the followings, we show several numerical examples to reveal how the stability and the TCP transient behavior are affected by several system parameters — the number of TCP connections, the propagation delay, the bottleneck link capacity, and the buffer size of the bottleneck router.

Figure 4 shows the maximum modulus of the eigenvalues for different numbers of TCP connections of N = 5, 10, and 15. In this figure, we plot the maximum modulus of eigenvalues of the state transient matrix **A** for different bottleneck link capacities of μ = 0–5 [packet/ms] and propagation delays of τ = 0–5 [ms]. The buffer size of the bottleneck router m is fixed at 50 [packet] and the average arrival rate of the background traffic λ_B is fixed at 0.2 [packet/ms].

Window–Scale optionbandwidth-delay product TCP ACK Window–Scale option

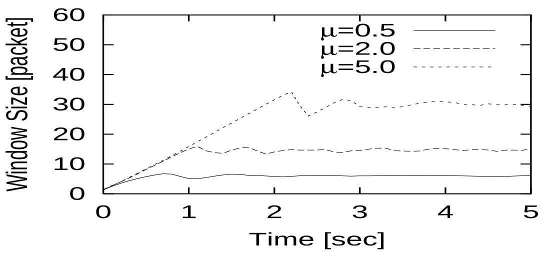
From Fig. 4, one can find that the maximum modulus of the eigenvalues is mostly determined by $\mu \times \tau$. This indicates that the stability and the transient behavior of TCP are determined by the bandwidth–delay product. This is because the congestion control mechanism of TCP is a window-based mechanism, and it changes the window size at every receipt of an ACK packet. Provided that the packet size is fixed, the number of ACK packets in the network during a round-trip time is proportional to the bandwidth–delay product. In the control engineer's view, increase of the propagation delay means decrease of the feedback gain or the feedback delay. Hence, the stability and the transient behavior of TCP are determined by the bandwidth–delay product.

By comparing Figs. 4(a)–(c), one can find that as the number of TCP connections increases, the stability region becomes large. This is because the larger the number of TCP connections becomes, the smaller the bandwidth–delay product of each TCP connection becomes. The small bandwidth–delay product means that a source host receives a small number of ACK packets which carry feedback information. As a result, the increase of the number of TCP connections has the same effect with decrease of the feedback delay and/or the feedback gain.

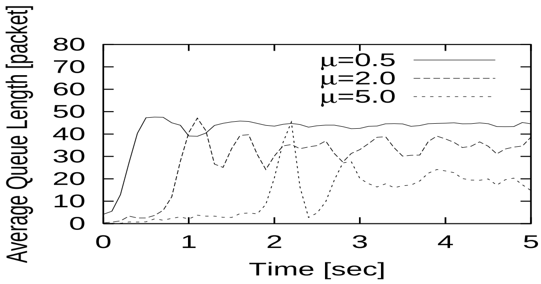
Figure 5 shows the maximum modulus of eigenvalues for the number of TCP connections $N = 10$ and different arrival rate of the background traffic, λ_B = 0, 0.2 and 0.5 [packet/ms]. By comparing Fig. 5(a)–(c), one can find that the stability region becomes slightly large, as λ_B becomes large. This is because increase of the background traffic corresponds to decrease of the available bandwidth to TCP connections. Namely, decrease of the available bandwidth to TCP connections results in the smaller bandwidth–delay product, which means a little feedback gain. Because the little feedback gain makes system sensitivity to the changes of the environment low, the reduction of the available bandwidth would bring the larger stability region.

To validate our transient behavior analysis, we next show how the TCP transient behavior changes for different maximum moduli using simulation experiments. Figure 6 shows the window size and the average queue length obtained from our simulation for different bottleneck link capacities μ = 0.5, 2.0, and 5.0 [packet/ms]. Note that when the bottleneck link capacity μ is 0.5, 2.0, and 5.0, the maximum modulus of eigenvalues of the state transient matrix is 0.619, 0.780, and 0.923, respectively. We use the same values with Tab. 2 for all parameters except the bottleneck link capacity.

Using ns-2 simulator, we run simulations 50 times at a packet level for the same network model shown in Fig. 1, and investigate the evolution of the average TCP window size and the average queue length. More specifically, we calculate the average TCP window size and the average queue length every 100 ms. From this figure, one can find that the smaller the maximum modulus is (the smaller the bottleneck link capacity is), the better the transient behavior becomes. From these observations, we conclude that our transient behavior analysis using the control theory accurately captures the dynamics of TCP.



(a) TCP window size



(b) Average queue length

Figure 6: Simulation results for different maximum moduli.

5 Conclusion and Future Work

In this paper, by extending our previous work, we have analyzed the steady state and the transient behavior of TCP. We have derived the throughput of each TCP connection, the packet loss probability, and the average queue length at the bottleneck router. We have also analyzed the TCP transient behavior by using the control theory. As a result, we have found that the bandwidth–delay product mostly determines the stability and the transient behavior of TCP. We have also found that the network becomes stable as the number of TCP connections or the amounts of the background traffic increases.

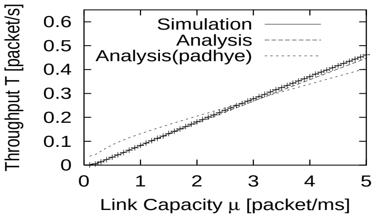
As a future work, it would be interesting to apply our approach to the more general heterogeneous network where several bottleneck routers exist, and improve the accuracy of the approximate analysis by using the rigorous analytic model of a $M/D/1/m$ queue.

References

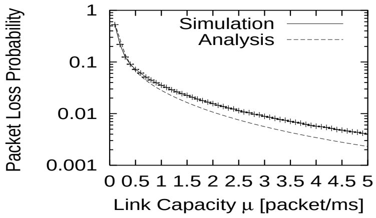
[1] The network simulator. available at `http://www.isi.edu/nsnam/ns/`.

[2] B.Sikdar, S.Kalyanaraman, and K.S.Vastola. Analytic models and comparative study of latency and steady-state throughput of TCP Tahoe, Reno nad SACK. In *Proceedings of IEEE GLOBECOM2001*, Nov. 2001.

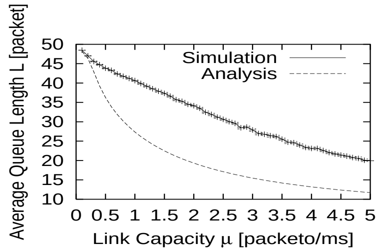
[3] C. Casetti and M. Meo. A new approach to model the stationary behavior TCP connections. In *Proceedings of IEEE INFOCOM 2000*, pages 367–375, Mar. 2000.



(a) TCP throghput

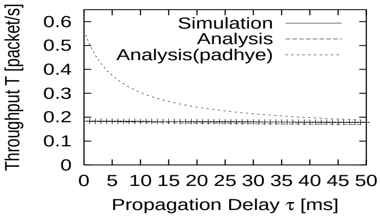


(b) Packet loss probability

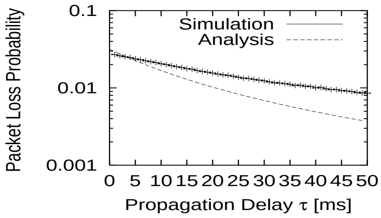


(c) Average queue length

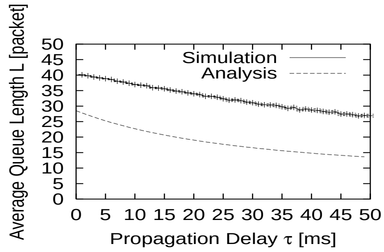
Figure 2: Analytic and simulation results for different bottleneck link capacities.



(a) TCP throghput



(b) Packet loss probability



(c) Average queue length

Figure 3: Analytic and simulation results for different propagation delays.

[4] E.Alessio, M.Garetto, R.Cigno, M.Meo, and M.Marsan. Analytical estimation of completion times of mixed NewReno and Tahoe TCP connections over single and multiple bottleneck networks. In *Proceedings of IEEE GLOBECOM2001*, Nov. 2001.

[5] V. Firoiu and M. Borden. A study of active queue management for congestion control. In *Proceedings of INFOCOM2000*, pages 1435–1444, 2000.

[6] H. Hisamatu, H. Ohsaki, and M. Murata. On modeling feedback congestion control mechanism of TCP using fluid flow approximation and queueing theory. In *Proceedings of 4th Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT2001)*, pages 218–222, Aug. 2001.

[7] C. V. Hollot, V. Misra, D. F. Towsley, and W. Gong. A control theoretic analysis of RED. In *Proceedings of INFOCOM2000*, pages 1510–1519, 2001.

[8] V. Jacobson and M. J. Karels. Congestion avoidance and control. In *Proceedings of SIGCOMM '88*, pages 314–329, Nov. 1988.

[9] A. Kumar. Comparative performance analysis of versions of TCP in a local network with a lossy link. *IEEE/ACM Transactions on Networking*, 6(4):485–498, 1998.

[10] M. Mathis, J. Semke, and J. Mahdavi. The macroscopic behavior of the TCP congestion avoidance algorithm. *Proceedings of ACM SIGCOMM Communication Review*, 27(3), July 1997.

[11] A. Misra and T. J. Ott. The window distribution of idealized TCP congestion avoidance with variable packet loss. In *Proceedings of IEEE INFOCOM '99*, pages 1564–1572, 1999.

[12] V. Misra, W.-B. Gong, and D. F. Towsley. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED. In *Proceedings of ACM SIGCOMM*, pages 151–160, 2000.

[13] H. Ohsaki and M. Murata. Steady state analysis of the RED gateway: stability, transient behavior, and parameter setting. *IEICE Transactions on Communications*, E85-B(1):107–115, Jan. 2002.

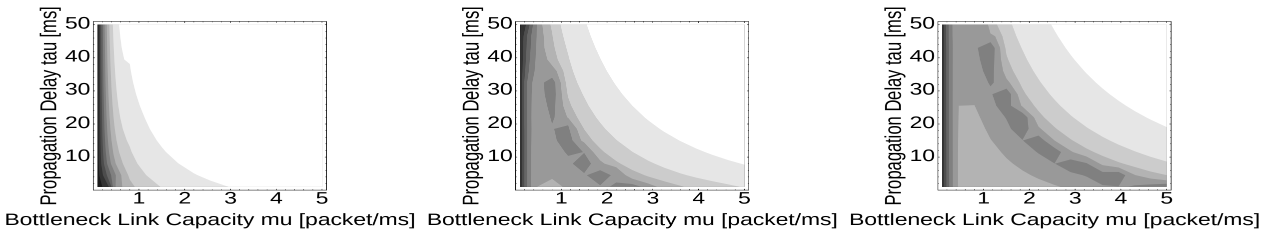
[14] H. Ohsaki, M. Murata, T. Ushio, and H. Miyahara. Stability analysis of window-based flow control mechanism in TCP/IP networks. *1999 IEEE International Conference on Control Applications*, pages 1603–1606, Aug. 1999.

[15] T. J. Ott. ECN protocols and the TCP paradigm. In *Proceedings of IEEE INFOCOM 2000*, pages 100–109, Mar. 2000.

[16] T. J. Ott, J. Kemperman, and M. Mathis. The stationary behavior of ideal TCP congestion avoidance. 1996.

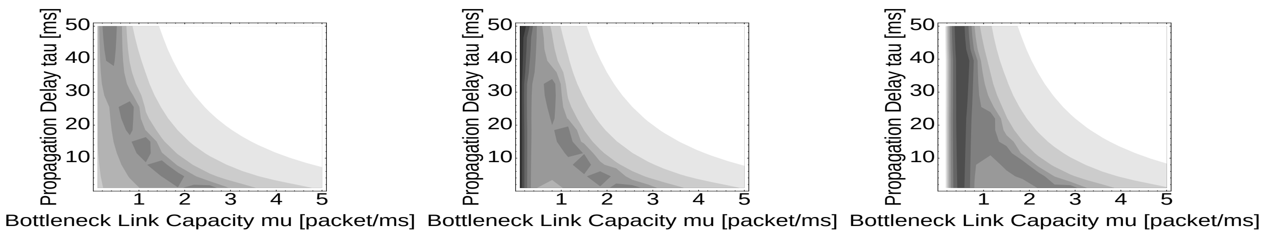
[17] J. Padhye, V. Firoiu, and D. Towsley. A stochastic model of TCP Reno congestion avoidance and control. Technical report, CMPSCI Technical Report 99-02, 1999.

[18] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP throughput: a simple model and its empirical validation. In *Proceedings of ACM SIGCOMM '98*, pages 303 – 314, Sept. 1998.



(a) $N = 5$ (b) $N = 10$ (c) $N = 15$

Figure 4: Maximum modulus of eigenvalues for different number of TCP connections.



(a) $\lambda_B = 0$ [packet/ms] (b) $\lambda_B = 0.2$ [packet/ms] (c) $\lambda_B = 0.5$ [packet/ms]

Figure 5: Maximum modulus of eigenvalues for different arrival rate of background traffic.