On Modeling Feedback Congestion Control Mechanism of TCP using Fluid Flow Approximation and Queueing Theory

Hiroyuki Hisamatsu† Hiroyuki Ohsaki‡ Masayuki Murata‡

†Graduate School of Engineering Science, Osaka University, Japan
E-mail: hisamatsu@ics.es.osaka-u.ac.jp

‡Cybermedia Center, Osaka University, Japan
E-mail: {oosaki, murata}@cmc.osaka-u.ac.jp

Abstract

The Internet uses a window-based congestion control mechanism in TCP (Transmission Control Protocol). In the literature, there have been a great number of analytical studies on TCP. Most of those studies have focused on the statistical behavior of TCP by assuming a constant packet loss probability in the network. However, the packet loss probability, in reality, changes according to packet transmission rates from TCP connections. In this paper, we explicitly model the interaction between the congestion control mechanism of TCP and the network as a feedback system. Namely, we model the congestion control mechanism of TCP as a dynamic system, where the input to the system is the packet loss probability and the output is the window size. Inversely, we model the network as a dynamic system, where the input is the window size and the output is the packet loss probability. The network is modeled by a $M/M/1/m$ queueing system by assuming an existence of a single bottleneck link. Using our analytic model, the transient behavior of TCP connections is quantitatively evaluated with several numerical examples.

1 Introduction

The most-widely deployed implementation of TCP called *TCP Reno* uses a packet loss in the network as feedback information from the network since a packet loss implies congestion occurrence in the network [1]. The fundamental operation of TCP Reno is summarized as follows. Until a packet loss occurs in the network, TCP Reno gradually increases the window size of a source host. As soon as the window size exceeds the bandwidth-delay product (i.e., the available bandwidth $\times$ the round-trip delay), excess packets are queued at the buffer of an intermediate router. When the window size increases further, the buffer of the router overflows, resulting in a packet loss. At the source host, TCP Reno conjectures the packet loss by receiving more than three duplicate ACKs. TCP Reno then decreases the window size for resolving congestion. After the reduction of the window size, congestion in the network is relieved, and TCP Reno increases the window size of the source host again. By repeating this control indefinitely, TCP Reno tries to efficiently utilize network resources as well as to prevent congestion in the network.

In the literature, there have been a great number of analytical studies on TCP [2, 3, 4]. In [2, 3], the authors have derived the average window size and the throughput of TCP Reno by assuming a constant packet loss probability in the network. However, the packet loss probability, in reality, changes according to packet transmission rates from TCP connections. Conversely, the window size of a TCP connection is dependent on the packet loss probability in the network. In this paper, we explicitly model the interaction between the congestion control mechanism of TCP and the network as a feedback system for investigating the transient behavior of TCP. For modeling the congestion control mechanism of TCP, we use four different analytic models presented in [2, 3, 4]. As a network model, we use a $M/M/1/m$ queueing system, where the input traffic is mixture of TCP traffic and background traffic (i.e., non-TCP traffic).

In [5], the authors have analyzed the performance of TCP by modeling the network as a $M/D/1/m$ queuing system. However, the authors have focused only on the steady state behavior of TCP; that is, the transient behavior of TCP has not been evaluated. In addition, their analytic model is not TCP Reno but TCP Tahoe, which doesn't have several important mechanisms found in TCP Reno. For instance, the effect of the fast retransmit mechanism in TCP Reno has not been investigated. In [4, 6], analytic models for TCP Reno and the RED (Random Early Detection) router have been presented, and the performance of TCP with the RED router has been analyzed. In [6], the primary focus of the analysis is in the steady state behavior of TCP. Only a qualitative discussion on the transient behavior has been presented. In [4], a control theoretical approach has been taken to analyze the stability and the transient behavior of TCP, where the RED router is modeled by a nonlinear discrete-time system. On the other hand, the main objective of this paper is to analyze the transient behavior of TCP with the Drop-Tail router, since most existing routers in the current Internet are Drop-Tail routers. We take a different approach of modeling the Drop-Tail router using a queuing theory.

In our analytic model, both TCP traffic and background traffic are taken account of. We model the interaction between the congestion control mechanism of TCP and the network as a feedback system; that is, both the congestion control mechanism of TCP running on a source host and the network seen by TCP are modeled by dynamic systems (Fig. 1). The congestion control mechanism of TCP is a window-based flow control mechanism, and it dynamically changes the window size according to occurrence of packet losses in the network. Hence, there exists a tendency that when the packet loss probability is small, the window size becomes large. On the contrary, when the packet loss probability is large, the window size tends to become small. We model the congestion control mechanism of TCP as a dy-

namic system, where the input to the system is the packet loss probability in the network and the output from the system is the window size of TCP.

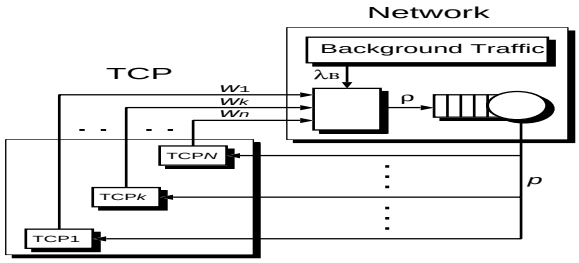

Figure 1: Analytic model as a feedback system consisting of TCP connections and network.

On the other hand, the network seen by TCP behaves such that when the number of packets entering the network increases, some packets are waited at the buffer of the router destined for the bottleneck link. This sometimes causes buffer overflow, resulting in a packet loss. So the packet loss probability becomes large when the number of packets entering the network increases. Thus, the network seen by TCP can be modeled by a dynamic system, where the input to the system is the window size and the output from the system is the packet loss probability.

Organization of this paper is as follows. In Section 2, we describe how the interaction between TCP and the network can be modeled as a feedback system. We then present four analytic models of the congestion control mechanism of TCP. In Section 3, comparing these four analytic models with simulation results, we discuss which analytic model is most suitable for analyzing the transient behavior of TCP. In Section 4, we analyze the transient behavior of TCP, and quantitatively show how the transient behavior of TCP is affected by the amount of background traffic and/or the propagation delay of the bottleneck link. In Section 5, we conclude this paper and discuss a few future works.

2 Analytic Model

2.1 Modeling Network using Queuing Theory

We assume that there exists only a single bottleneck link in the network. In the followings, the router just before the bottleneck link is called *bottleneck router*. We also assume that the bottleneck router adopts a Drop-Tail discipline. Provided that the network is stationary, the bottleneck router can be modeled by a single queue. Thus, once the packet arrival rate and the capacity of the bottleneck router are known, the packet loss probability and the average waiting time can be obtained from the queuing theory. Since the packet departure process from a source host is oscillatory, in reality, the network is not stationary. However, as we will show in Section 3, the network seen by TCP can be well modeled by a queueing system at a relatively large time scale (e.g., the round-trip time). In the rest of this subsection, we formally describe how the network seen by TCP can be modeled by a queueing system.

Let N be the number of TCP connections, and w_i and r_i be the window size and the round-trip time of ith ($1 \leq i \leq N$) TCP connection. Assuming that each TCP connection continuously sends packets, the transmission rate from ith TCP connection can be approximated by w_i/r_i. The average packet arrival rate at the bottleneck router, λ, is therefore given by $\sum_{i=1}^{N} w_i/r_i + \lambda_B$, where λ_B is the average arrival rate of background traffic at the bottleneck router. Let μ be the capacity of the bottleneck link, the offered traffic load at the bottleneck router ρ is given by $\rho = \lambda/\mu$. Depending on the packet arrival process, the distribution of the packet processing time, and the buffer capacity, there can be several queuing systems suitable for modeling the network seen by TCP. As a network model, we use a finite buffer queuing system, $M/M/1/m$, where m represents the buffer size of the bottleneck router.

2.2 Modeling TCP using Different Approaches

The congestion control mechanism of TCP is quite complicated since it performs several control mechanisms such as detecting packet losses in the network and retransmitting lost packets. It is therefore impossible to build an exact analytic model of TCP. In this paper, we model only the main part of the congestion control mechanism of TCP, and ignore the rest; that is, we model the essential behavior of TCP (i.e., the window-based flow control mechanism and the loss recovery mechanism including the fast retransmit mechanism of TCP Reno) in its congestion avoidance phase. In [3, 7, 4], several analytic models for the congestion avoidance phase of TCP have been presented, describing the relation between the packet loss probability in the network and the resulting window size of TCP. In what follows, we introduce four analytic models called A, A', B, and C, which are derived from different modeling approaches. In Section 3, we will discuss which model is suitable for analyzing the transient behavior of TCP.

- **Model A**

In [3], by assuming a constant packet loss probability in the network (denoted by p), the authors have presented an analytic model describing the window size of a TCP connection in steady state. The authors have derived the average throughput of a TCP connection. In this model, the authors assume that the initial window size at the beginning of a congestion avoidance phase is equal to that at the beginning of the next congestion avoidance phase, and that TCP sends the number $1/p$ of packets in each congestion avoidance phase. In summary, the average throughput of a TCP connection, λ_T, is derived as

$$\lambda_T = \frac{\frac{1-p}{p} + E[W] + \hat{Q}(E[W])\frac{1}{1-p}}{r\left(\frac{b}{2}E[W]+1\right) + \hat{Q}(E[W])T_o\frac{f(p)}{1-p}}$$

where

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

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

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

and r is the average round-trip time of the TCP connection, and b is a parameter of delayed ACKs (i.e., a destination host returns an ACK packet for every b data packets). T_O is the length of TCP's retransmission timer. $\hat{Q}(w)$ is a probability that when the window size is w, the source host fails to detect a packet loss from duplicate ACKs. From these equations, the window size of TCP in steady state, w_A, is given by

$$w_A = \lambda_T\, r \qquad (1)$$

• **Model A′**

When the packet loss probability is very small ($p << 1$), Eq. (1) is approximated as [3]

$$w_A \simeq \sqrt{\frac{3}{2bp}} \quad (2)$$

• **Model B**

In [7], the authors have analyzed a congestion control mechanism using ECN (Explicit Congestion Notification). ECN is a mechanism to explicitly notify source hosts of congestion occurrence in the network. When a router experiences congestion, by setting the ECN bit of arriving packets, it informs source hosts of the congestion occurrence. In [7], the authors assume that the ECN bit of an ACK packet is set with a probability of p_E, and have derived a state transition equation for the window size. Let $w(k)$ be the window size at slot k (i.e., the time when kth ACK packet is received). Their analytic model is different from TCP; that is, when the ECN bit is set, the source host linearly increases the window size by $I(w(k))$. Otherwise, it multiplicatively decreases the window size by $D(w(k))$. By calculating the expected value of the window size at each receipt of an ACK packet, the evolution of the window size is given by

$$\begin{aligned} w(k) &= w(k-1) + (1-p_E)\, I(w(k-1)) \\ &\quad -p_E\, D(w(k-1)) \end{aligned} \quad (3)$$

The analytic model presented in [7] is not for TCP, but can be easily applied. Namely, an ACK packet with the ECN bit not set corresponds to a non-duplicate ACK in TCP (i.e., indication of no congestion). Similarly, an ACK packet with the ECN bit set corresponds to duplicate ACKs (i.e., indication of congestion). Thus, when the packet loss probability is p, the state transition equation for the window size, w_B, is given by

$$\begin{aligned} w_B(k) &= w_B(k-1) + (1-p)\,\frac{1}{w_B(k-1)} \\ &\quad -p\,(1-\hat{Q}(w_B(k-1)))\,\frac{w_B(k-1)}{2} \\ &\quad -p\,\hat{Q}(w_B(k-1))\,(w_B(k-1)-1) \end{aligned} \quad (4)$$

Note that we modify and extend Eq. (3) to include the time-out mechanism of TCP.

• **Model C**

In [4], the authors have derived the state transition equation for the window size in the congestion avoidance phase of TCP. This analytic approach uses a discrete-time model, where a time slot corresponds to the duration between two succeeding packet losses. However, their analytic model is not for the Drop-Tail router but for the RED router, where the router randomly discards arriving packets. In what follows, we describe a modification to the analytic model presented in [4] for analyzing TCP with the Drop-Tail router.

In [4], the authors have derived $\overline{X}(k)$, the expected number of packets passing through the RED router at slot k as

$$\overline{X}(k) = \frac{1/p_b(k)+1}{2}$$

where $p_b(k)$ is the packet dropping probability of the RED router at slot k. Let p be the packet loss probability of the Drop-Tail router, $\overline{X}(k)$ is changed to

$$\overline{X}(k) = \sum_{n=1}^{\infty} n(1-p)^{n-1}p = \frac{1}{p}$$

Thus, when the packet loss probability is p, the window size $w_{\underline{C}}$ at the beginning of slot k is obtained as [4]

$$w_{\underline{C}}(k) = \frac{1}{4}\left\{-1+\sqrt{(1-2w_{\underline{C}}(k-1))^2+\frac{8}{p}}\right\} \quad (5)$$

Note that Eq. (5) is derived by assuming that a packet loss probability is constant in a slot. Since the packet loss probability is, in reality, increased as the window size increases, this analytic model might overestimate the window size.

We note that models A and A' are built based on the window size in steady state. It is therefore expected that these models are not suitable for analyzing the transient behavior of TCP. On the contrary, models B and C describe the dynamic behavior of the window size in the congestion avoidance phase. Thus, it is expected that models B and C are suitable for analyzing the transient behavior of TCP. In the next section, we compare these four analytic models using numerical and simulation results.

3 Model Validation with Simulation

3.1 Simulation Model

The simulation model is shown in Fig. 2. In this model, 10 TCP connections share the bottleneck link. The propagation delay of ith TCP connection is $5+i$ [ms], and the link capacity from the ith source host to the router is $5+0.5i$ [packet/ms]. We model the background traffic as UDP packets, where the packet arrival of UDP packets is modeled by a Poisson process with the average arrival rate of $\lambda_B = 2$ [packet/ms]. Unless explicitly noted, we use the following parameters in all simulations: both TCP and UDP packet sizes are fixed at 1000 [byte], the capacity of the bottleneck link μ is 5 [packet/ms], and the propagation delay of the bottleneck link τ is 5 [ms]. Note that with these simulation parameters, 1 [packet/ms] corresponds to about 8 [Mbit/s]. We run every simulation for 30 seconds using ns2 [8].

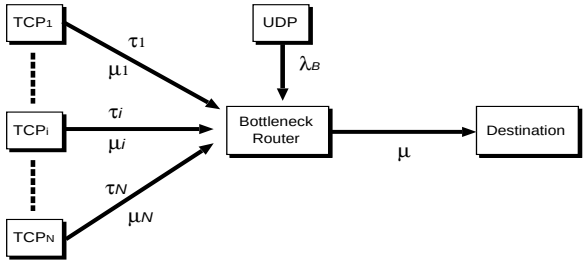

Figure 2: Simulation model of 10 TCP connections and a single bottleneck link.

3.2 Network Model

Figure 3 shows the relation between the offered traffic load and the packet loss probability. These values are measured at the bottleneck router for every 10 [ms]. Namely, these values are rough estimation of the *instantaneous* offered traffic load and the *instantaneous* packet loss probability. In the figure, the packet loss probabilities obtained from

well-known results of $M/M/1/m$ and $M/M/1$ are also plotted. This figure shows that the dynamics of the network at a relatively small time scale can be well modeled by the $M/M/1/m$ model. Note that the queuing theory is for analyzing the statistical behavior, not the dynamical behavior. Note also that UDP and TCP packet sizes are fixed at 1000 [byte]. This figure indicates that $M/M/m/1$ could be usable for analyzing the transient behavior of TCP. However, simulation results are scattered around the result of $M/M/1/m$. This means that the packet loss probability has a variability even when the offered traffic load at the bottleneck router is fixed.

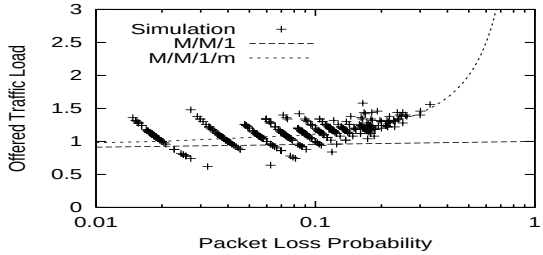

Figure 3: Comparison of $M/M/1/m$ queuing system with simulation result.

3.3 TCP Models

By comparing with simulation results, we discuss how accurately four analytic models of TCP capture the relation between the window size and the packet loss probability. Figure 4 shows the relation between the packet loss probability and the window size obtained using models A, A', B and C, respectively. In this figure, the window size for a given packet loss probability is obtained using Eqs. (1), (2), (4), and (5). Note that in the model A, the analytic result is calculated by assuming no timeout (i.e., $\hat{Q}(w) = 0$). Also note that in the model C, Eq. (5) gives the window size at the beginning of a slot, and not the average window size. For comparison purposes, the average window size is calculated and plotted in the figure. Refer to [4] for more detail. We also plot simulation results; that is, points corresponding to the average window size and the packet loss probability. As with Fig. 3, these values are *instantaneous values* of the average window size and the packet loss probability, which are measured at the bottleneck router for every 1 [s]. This figure shows that when the packet loss probability is less than 0.02, analytic models A, A', and B show good agreement with simulation results. On the other hand, when the packet loss probability is more than 0.03, analytic models B and C show good agreement.

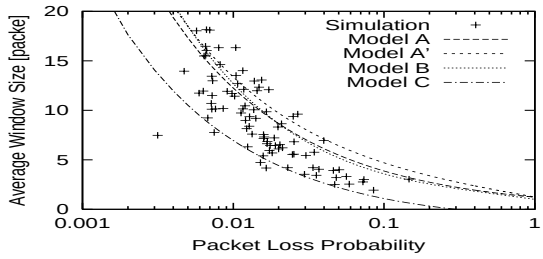

Figure 4: Comparison of four TCP models with simulation results.

4 Transient Behavior Analysis of TCP

Using the analytic model presented in Section 2, we analyze the transient behavior of TCP in the congestion avoidance phase. By the word *transient behavior*, we mean the dynamics of the window size from its initial value to its equilibrium value. TCP changes the window size according to the occurrence of a packet loss in the network. Since a packet loss occurs probabilistically, the window size can be thought of as a random variable. By focusing on the *average behavior* of TCP, we analyze the transient behavior of TCP. More specifically, we analyze the transient behavior of TCP by investigating how the expected value of the window size changes.

We model both the congestion control mechanism of TCP and the network as interconnected discrete-time systems, where the states of these models change at every unit time. The state of the network at slot k is then fully described by the window size $w(k)$ and the packet loss probability $p(k)$. For given initial values of the window size and the packet loss probability, the evolution of the window size and the packet loss probability can be numerically obtained. In what follows, we present analytic results for the combination of the model B for TCP and a $M/M/1/m$ queuing system for the network. Although rigorous analyses of the stability and the transient behavior of TCP are possible using the approach presented in [4], in this paper, we present a relatively simple analytic approach for simplicity. As have been explained in Section 2.2, the model B describes the change of the window size every receipt of an ACK packet. Hence, in the following analysis, the duration between two succeeding ACK packets corresponds to a unit time. We assume that the propagation delays of all TCP connections are identical, and that the widow sizes of all TCP connections change synchronously.

Let τ be the propagation delay between source and destination hosts. Since a source host sends the number $w(k)$ of packets during its round-trip time, the round-trip time corresponds to $w(k)$ slots. On the contrary, it takes the round-trip time for the change in the packet loss probability to propagate to the source host. The window size $w(k)$ is therefore determined by $w(k-1)$, the previous window size, and $p(k-w(k-1))$, the packet loss probability before the round-trip time. From these observations, the model B (Eq. (4)), and well-known results of the $M/M/1/m$ queuing system, the state transition equations are obtained as

$$\begin{aligned} w(k) &= w(k-1) + \frac{1 - p(k - w(k-1))}{w(k-1)} \\ &\quad - \frac{p(k - w(k-1))\, w(k-1)}{2} \\ p(k) &= \frac{(1-\rho(k))\,\rho(k)^m}{1-\rho(k)^{m+1}} \end{aligned}$$

where $\rho(k)$ is the offered traffic load at the bottleneck router at slot k. This value is defined as

$$\rho(k) = \frac{1}{\mu}\left(\frac{N\, w(k)}{r(k)} + \lambda_B\right)$$

where $r(k)$ is the round-trip time of the TCP connection:

$$\begin{aligned} r(k) &= 2\tau + \frac{1-\rho^m}{\mu(1-\rho^{m+1})}\left(\frac{1}{1-\rho} + \frac{m\rho^m}{1-\rho^m}\right) \\ &\simeq 2\tau + \frac{m}{\mu} \end{aligned}$$

In the above equation, we assume that all packet losses can be detected by duplicate ACKs (i.e., $\hat{Q}(w) = 0$ in Eq. (4)). Since TCP usually operates around high traffic load, the round-trip time $r(k)$ can be approximated by the sum of the propagation delay and the buffer size divided by the capacity of the bottleneck link. Recall that $w(k)$ is not the instant value of the window size, but the average value of the window size. Using these equations and calculating the evolutions of $w(k)$ and $p(k)$, the transient behavior of TCP can be analyzed.

We next present several numerical examples, showing how the amount of background traffic λ_B and the propagation delay τ of the bottleneck link affect the transient behavior of TCP. In the following numerical examples, unless explicitly noted, the initial window size is 1 [packet], the initial packet loss probability is 0, the number of TCP connections N is 10, the capacity of the bottleneck link μ is 5 [packet/ms], the propagation delay τ is 15 [ms], and the buffer size of the bottleneck router m is 50 [packet].

Figure 5 shows the evolution of the window size in the congestion avoidance phase for the amount of background traffic λ_B of 0, 2.0, and 4.5 [packet/ms]. From this figure, one can find that the window size in steady state becomes small as the amount of background traffic increases, indicating that TCP suffers less throughput. One can also find that the convergence speed (i.e., in this case, the increase rate of the window size) of the window size is independent of the amount of background traffic. This is because, in the congestion avoidance phase, TCP increases the window size by one packet per a round-trip time, which is essentially irrelevant to the TCP throughput.

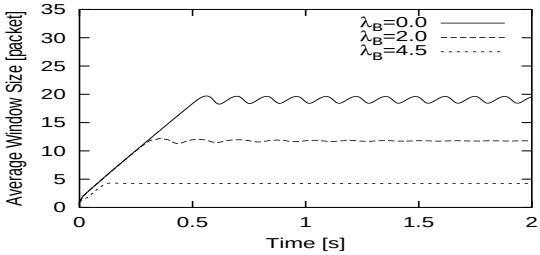

Figure 5: Transient behavior of TCP for different amount of background traffic.

Figure 6 shows the evolution of the window size in the congestion avoidance phase for the propagation delay τ of 10, 30, and 50 [ms]. One can find that the window size becomes large as the propagation delay increases. This can be intuitively understood from the increased bandwidth-delay product. In addition, as the propagation delay becomes large, one can find that the convergence speed of the window size becomes slow, and that the ramp-up time of the window size becomes short. In general, as the feedback delay becomes large, the transient behavior is degraded and the system becomes less stable. However, the latter shows a contrary result. In particular, when the propagation delay τ is small (e.g., 10 [ms]), the window size oscillates for long (e.g., more than 1.5 [s]). This is because, from a control theoretical viewpoint, the feedback gain in the congestion avoidance phase of TCP is changed according to the round-trip time. Namely, in the congestion avoidance phase of TCP, the window size is incremented by one packet for every round-trip time. Thus, increasing the propagation delay implies decreasing the feedback gain.

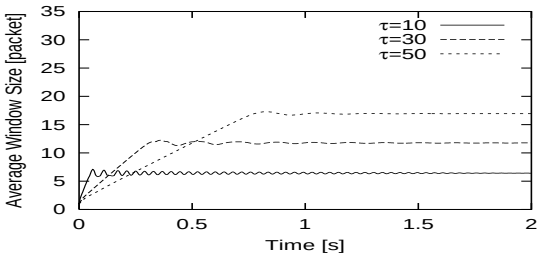

Figure 6: Transient behavior of TCP for different propagation delay of bottleneck link.

5 Conclusion and Future Work

In this paper, we have modeled both the congestion control mechanism of TCP and the network as a feedback system, and have analyzed the transient behavior of TCP. We have shown that the transient behavior is heavily dependent on the propagation delay of the bottleneck link, but is almost independent of the amount of background traffic. We have also shown that the operation of TCP in the congestion avoidance phase is less stable when the amount of background traffic is small or the propagation delay of the bottleneck link is short.

In this paper, we have analyzed the transient behavior of TCP by iteratively calculating state transitions. However, rigorous analyses of the stability and the transient behavior of TCP are possible using the approach presented in [4]. We are currently working on such rigorous analyses.

Acknowledgement

This work was supported in part by Research for the Future Program of Japan Society for the Promotion of Science under the Project "Integrated Network Architecture for Advanced Multimedia Application Systems" (JSPS-RFTF97R16301).

References

[1] V. Jacobson and M. J. Karels, "Congestion avoidance and control," in *Proceedings of SIGCOMM '88*, pp. 314–329, Nov. 1988.

[2] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Reno performance: a simple model and its empirical validation," *IEEE/ACM Transactions on Networking*, vol. 8, pp. 133–145, Apr. 2000.

[3] 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*, 1998.

[4] H. Ohsaki, M. Murata, and H. Miyahara, "Steady state analysis of the RED gateway: stability, transient behavior, and parameter setting," submitted to *IEICE Transactions on Communications*, May 2001.

[5] C. Casetti and M. Meo, "A new approach to model the stationary behavior of TCP connections," in *Proceedings of IEEE INFOCOM 2000*, Mar. 2000.

[6] V. Firoiu and M. Borden, "A study of active queue management for congestion control," in *Proceedings of IEEE INFOCOM 2000*, Mar. 2000.

[7] T. J. Ott, "ECN protocols and the TCP paradigm," in *Proceedings of IEEE INFOCOM 2000*, pp. 100–109, Mar. 2000.

[8] "The network simulator — ns2." available at `http://www.isi.edu/nsnam/ns/`.