Master's Thesis

Title

Analyzing Steady State and Transient State Performance of Transmission Control Protocol in the Internet

Supervisor

Prof. Masayuki Murata

Author

Hiroyuki Hisamatsu

February 12th, 2003

Department of Informatics and Mathematical Science

Graduate School of Engineering Science

Osaka University

Master's Thesis

Analyzing Steady State and Transient State Performance of Transmission Control Protocol in the Internet

Hiroyuki Hisamatsu

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 thesis, 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. 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. We then extend the analytic approach to a more generic network, where multiple TCP connections are allowed to have different propagation delays. We derive the packet loss probability in the network, the throughput and the average round-trip time of each TCP connection in steady state.

Keywords

TCP (Transmission Control Protocol) Congestion Control, Control Theory, Steady State Behavior, Transient State Behavior

Contents

List of Figures

List of Tables

1 Introduciton

A feedback-based congestion control mechanism is essential to realize an efficient data transfer services in a packed-switched network. In the current Internet, a sort of feedback-based congestion control mechanisms called TCP (Transmission Control Protocol) has been used. TCP has two mechanisms called *packet retransmission mechanism* and *congestion control mechanism*. The packet retransmission mechanism of TCP realizes reliable data transfer between source and destination hosts by keeping track of lost packets in the network. The congestion control mechanism of TCP, on the contrary, realizes efficient utilization of network resources by dynamically adjusting the number of in-flight packets.

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 [2-4]. 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 thesis, 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 [2-4]. 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 does not 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 non-linear discrete-time system. On the other hand, the main objective of this thesis 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

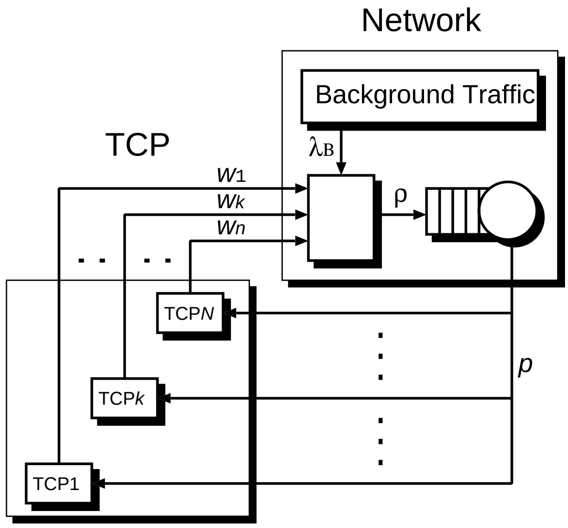


Figure 1: Analytic model as a feedback system consisting of TCP connections and a network.

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 dynamic 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.

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.

In the first part of this thesis, we analyze the steady state and the transient behavior of TCP. The main contribution of this thesis is to allow steady state analysis of TCP using the analytic model, and more importantly, to analyze the transient behavior of TCP using a rigorous manner based on control theory. We first derive the throughput of each TCP connection, the packet loss probability at the bottleneck router which is a router just before the bottleneck link, and the average queue length (i.e., the number of packets awaited in the buffer) at the bottleneck router or the round-trip time for TCP connections. 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.

In the second part of this thesis, we extend our analytic model to more generic network models. Using a similar modeling approach, we analyze the network with multiple TCP connections, where each TCP connection is allowed to have different propagation delays. We first present the steady state analysis for a network with a single TCP connection, and derive the TCP throughput, the TCP round-trip time, and the packet loss probability in the network. We then extend it to a network with heterogeneous TCP connections.

Organization of this thesis is as follows. In Section 2, we analyze and discuss the steady state and transient state behavior of TCP connections with identical propagation delays. In Section 3, by extending our analytic approach in Section 2, we analyze the steady state behavior of TCP connections with different propagation delays. Finally, in Section 4, we conclude the current thesis.

2 TCP Connections with Identical Propagation Delays

In this section, 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 [3, 7, 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). We then analyze the steady state and the transient behavior of TCP.

2.1 Analytic Model

2.1.1 Modeling Network using Queuing Theory

We assume that there exists only a single bottleneck link in the network. 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 2.2, 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 section, 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

$$\lambda = \sum_{i=1}^{N} \frac{w_i}{r_i} + \lambda_B$$

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

$$\rho = \frac{\lambda}{\mu}$$

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.

2.1.2 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 thesis, 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 2.2, 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

$$w_A = \lambda_T\, r \tag{1}$$

• **Model A′**

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

$$w_A \simeq \sqrt{\frac{3}{2bp}} \tag{2}$$

• **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 CE bit of arriving packets, it informs source hosts of the congestion occurrence. In [7], the authors assume that the CE 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 CE bit is not 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

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

The analytic model presented in [7] is not for TCP, but can be easily applied. Namely, an ACK packet with the CE bit not set corresponds to a non-duplicate ACK in TCP (i.e., indication of no congestion). Similarly, an ACK packet with the CE 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

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

Note that we modify and extend Eq. (3) to include the timeout 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]

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

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

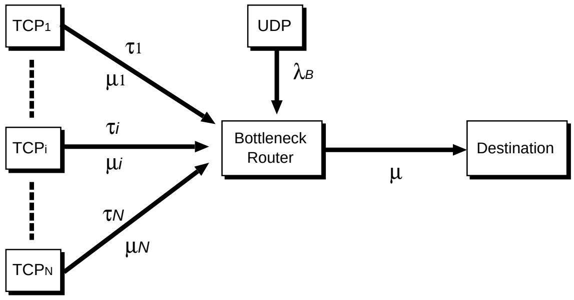


Figure 2: Simulation model of 10 TCP connections and a single bottleneck link.

suitable for analyzing the transient behavior of TCP. In the next subsection, we compare these four analytic models using numerical and simulation results.

2.2 Model Validation with Simulation

2.2.1 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].

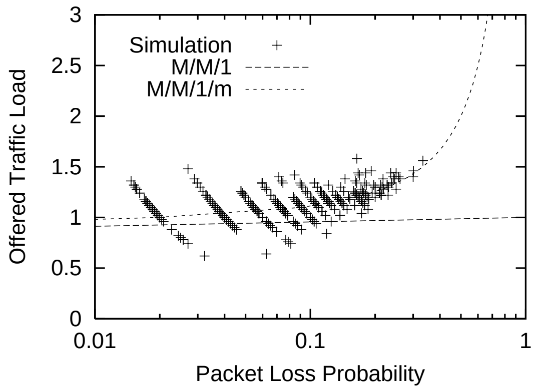


Figure 3: Comparison of $M/M/1/m$ queuing system with simulation result.

2.2.2 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 $M/M/1/m$ 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.

2.2.3 TCP Model

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

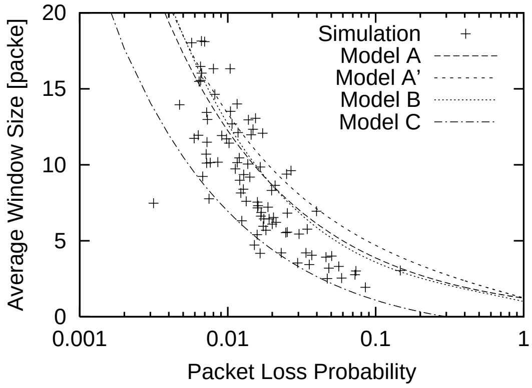


Figure 4: Comparison of four TCP models with simulation results.

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.

2.3 Derivation of State Transition Equations

In the rest of this section, we analyze the stability and the transient behavior of TCP by extending the modeling approach proposed in the first part of this section. In what

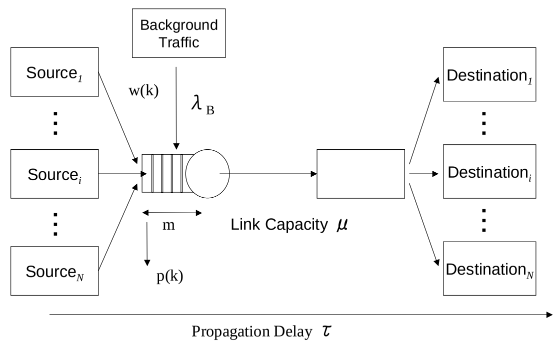


Figure 5: Analytic model of TCP connections with identical propagation delays.

follows, we briefly explain the modeling approach.

Figure 5 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 section.

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. 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], we first present a relatively simple analytic approach for simplicity. As have been explained

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

in Section 2.1.2, 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. 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.,

$$\begin{aligned} w(k+1) = w(k) + & \frac{1 - p(k+1-w(k))}{w(k)} \\ & -\frac{(1-\hat{Q}(w(k),p(k)))\, p(k+1-w(k))\, w(k)}{2} \\ & -p(k+1-w(k))\, \hat{Q}(w(k),p(k)) \end{aligned} \tag{6}$$

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 [3].

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

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. Namely,

$$p(k) \;=\; \frac{(1-\rho(k))\,\rho(k)^m}{1-\rho(k)^{m+1}} \tag{7}$$

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

$$\begin{aligned}
\rho(k) &= \frac{1}{\mu}\left(\frac{N\,w(k)}{r(k)}+\lambda_B\right)\\
r(k) &= 2\tau+\frac{\rho(k)\,(1-m\,\rho(k)^m+m\,\rho(k)^{m+1})}{\mu(1-\rho(k)^{m+2})\,(1-\rho(k))}
\end{aligned} \tag{8}$$

The use of the queuing model to analyze the steady state behavior of TCP is straightforward and promising. However, the application of the queueing model to analyze the transients behavior requires careful treatment since the queueing model was originally developed for analyzing not dynamic behavior but statistical behavior. However, we believe that the queueing model can give some insight on dynamic systems. Applicability of our approximate analysis will be validated in the latter sections by comparing analytic results with simulations results.

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. Note that our modeling approach can be further extended and applied to a network with heterogeneous TCP connections by, for example, using a similar approach proposed in [9]. However, it is beyond the scope of this thesis.

2.4 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 2.1. 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.,

$$w^* \equiv \lim_{k\to\infty} w(k) \tag{9}$$

$$p^* \equiv \lim_{k\to\infty} p(k) \tag{10}$$

These values can be numerically obtained by solving Eqs. (6) and (7) 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

$$T = \frac{w^*}{r^*} \tag{11}$$

$$\begin{aligned} L &= \rho^* \mu (r^* - 2\tau) \\ &= \frac{\rho^{*2}\,(1 - m\,\rho^{*m} + m\,\rho^{*(m+1)})}{(1-\rho^{*(m+2)})\,(1-\rho^*)} \end{aligned} \tag{12}$$

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. (11), (15), and (12), re-

Table 2: Parameter values for the simulation in Section 2.

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

spectively. Using ns-2 simulator [8], we run several simulation experiments at a packet level for the same network model with Fig. 5. 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, we model the background traffic by UDP packets, and the packet size of TCP and UDP packets is fixed at 1000 [byte]. The maximum window size of all TCP connections is fixed at a sufficiently large value, 10,000 [packets]. We use TCP version Reno on all source hosts.

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

Figure 6 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 [3] is shown in Fig. 6(a). In [3], 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 [3] 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

$$\begin{aligned} E[W] &= \frac{2+b}{3b} + \sqrt{\frac{8(1-p)}{3bp} + \left(\frac{2+b}{3b}\right)^2} \\ \hat{Q}(w) &= \frac{(1-(1-p)^3)(1+(1-p)^3(1-(1-p)^{w-3}))}{(1-(1-p)^w)} \\ f(p) &= 1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6 \end{aligned}$$

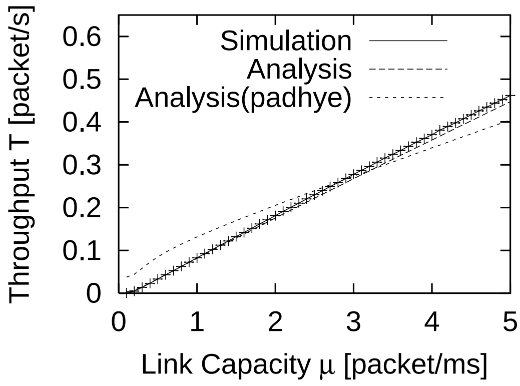
In this thesis, we calculate the TCP throughput from the above equation using the packet loss probability and the round-trip time obtained from the simulation. It can be found, in terms of the TCP throughput and the packet loss probability, both analytic and simulation results show a good agreement. In particular, in respect to the TCP throughput, it can be found that our analytic results show better agreement with simulation results than the value obtained from the expression in [3]. 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. 7, 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 bursty 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.

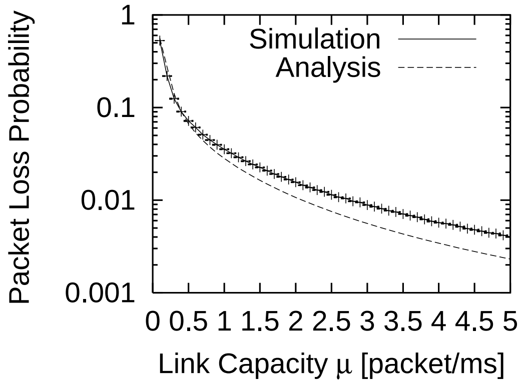
2.5 Trasient State Analysis

Using the analytic model presented in Section 2.1, 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.

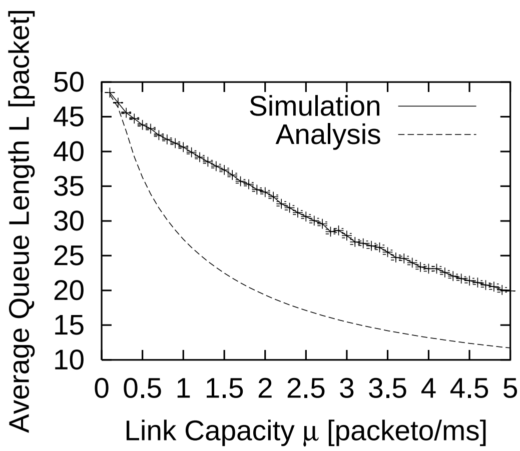
The state of the network at slot k is fully described by the window size $w(k)$ and the packet loss probability $p(k)$. For given initial values of the window size and the packet loss probability, the evolution of the window size and the packet loss probability can be numerically obtained. Although rigorous analyses of the stability and the transient behavior of TCP are possible using the approach presented in [4], we first present a relatively simple analytic approach for simplicity. 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



(a) TCP throghput

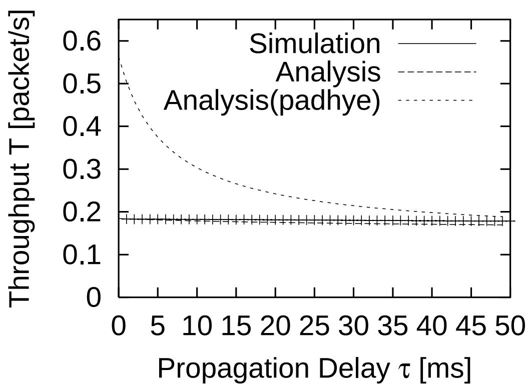


(b) Packet loss probability

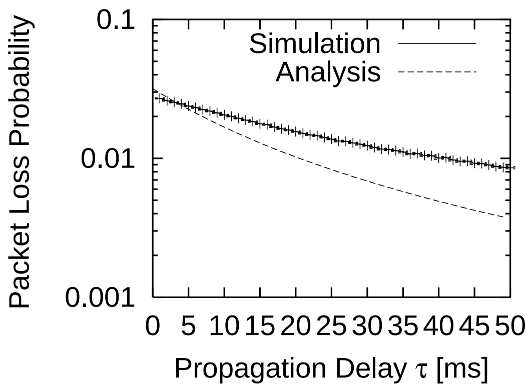


(c) Average queue length

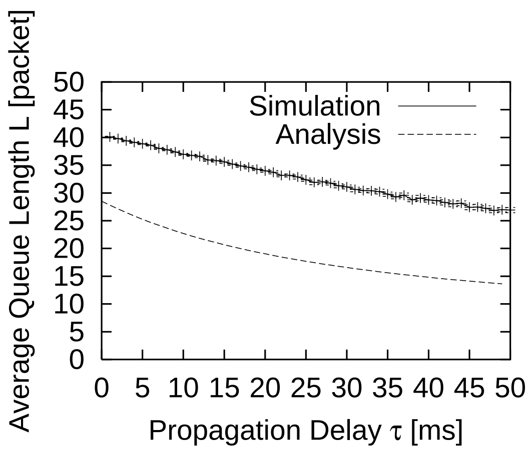
Figure 6: Analytic and simulation results for different bottleneck link capacities.



(a) TCP throghput



(b) Packet loss probability



(c) Average queue length

Figure 7: Analytic and simulation results for different propagation delays.

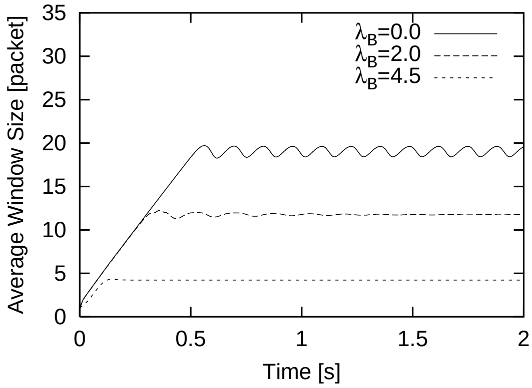


Figure 8: Transient behavior of TCP for different amount of background traffic.

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 8 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 9 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

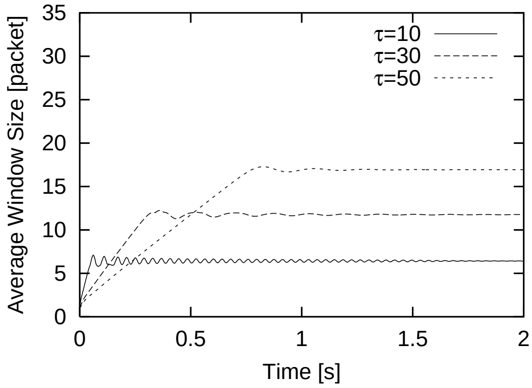


Figure 9: Transient behavior of TCP for different propagation delay of bottleneck link.

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.

In the rest of subsection, we analyze the TCP behavior in the transient state using state transition equations derived in Section 2.1. 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. (6) and (7) have non-linearity, we linearize them around their equilibrium points and write them in a matrix form

$$\mathbf{x}(k+1) = \mathbf{A}\,\mathbf{x}(k) \tag{13}$$

where $\mathbf{A}$ 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 [10]. 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 10 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 $\mathbf{A}$ for different bottleneck link capacities of $\mu =$ 0–5 [packet/ms] and propagation delays of $\tau =$ 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].

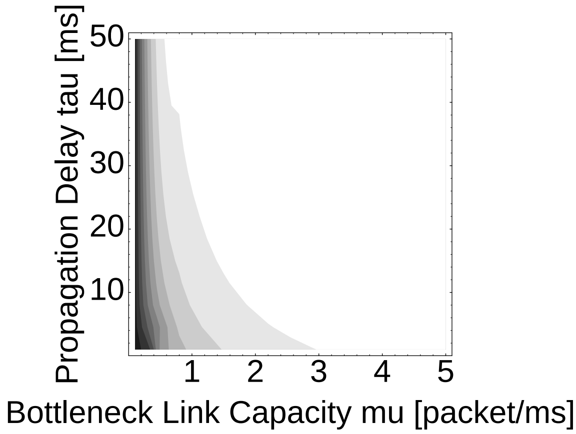
From Fig. 10, one can find that the maximum modulus of the eigenvalues is mostly determined by $\mu \times \tau$. This indicates that the stability and the transient behavior of TCP are determined by the bandwidth–delay product. This is because the congestion control mechanism of TCP is a window-based 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. 10(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.

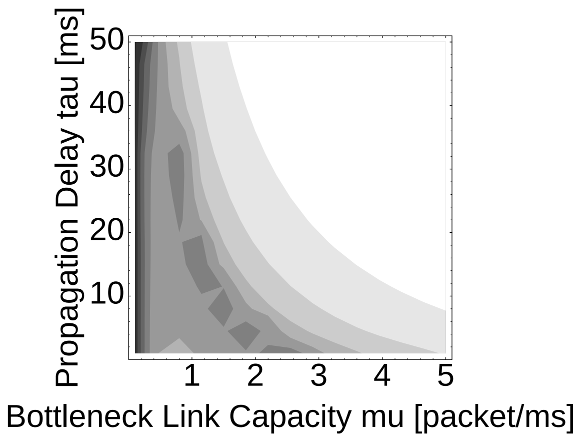
Figure 11 shows the maximum modulus of eigenvalues for the number of TCP connections $N = 10$ and different arrival rate of the background traffic, $\lambda_B = 0$, 0.2 and 0.5 [packet/ms]. By comparing Fig. 11(a)–(c), one can find that the stability region becomes slightly large, as λ_B becomes large. This is because the increase of the background traffic corresponds to the decrease of the available bandwidth to TCP connections. Namely, the 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 12 shows the window size and the average queue length obtained from our simulation for different bottleneck link capacities $\mu = 0.5$, 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.

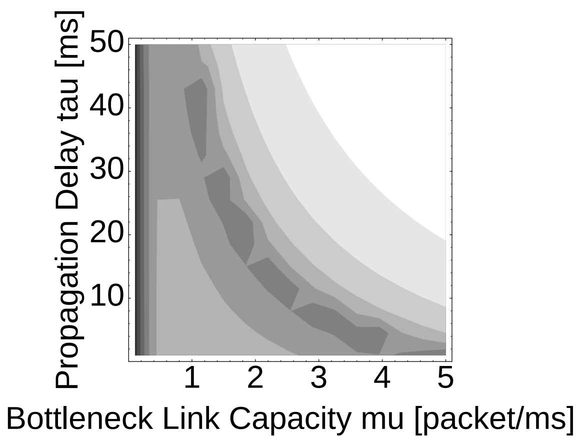
Using ns-2 simulator, we run simulations 50 times at a packet level for the same network model shown in Fig. 5, and investigate the evolution of the average TCP window size and



(a) $N = 5$

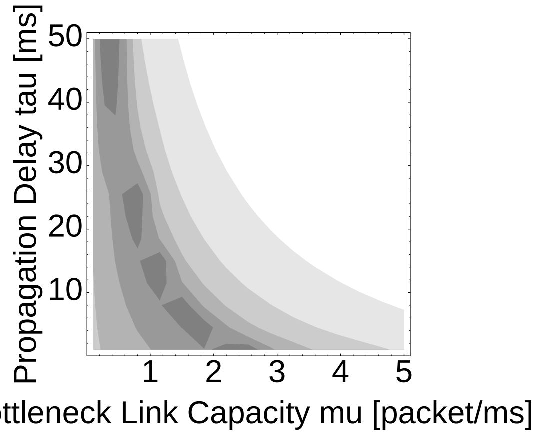


(b) $N = 10$

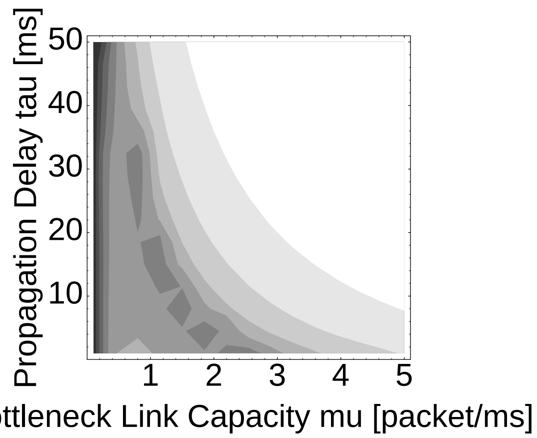


(c) $N = 15$

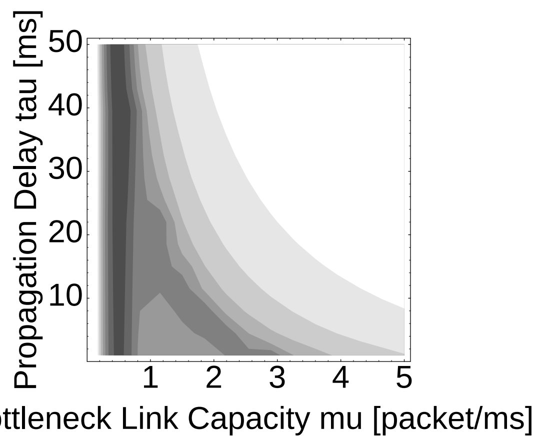
Figure 10: Maximum modulus of eigenvalues for different number of TCP connections.



(a) $\lambda_B = 0$ [packet/ms]

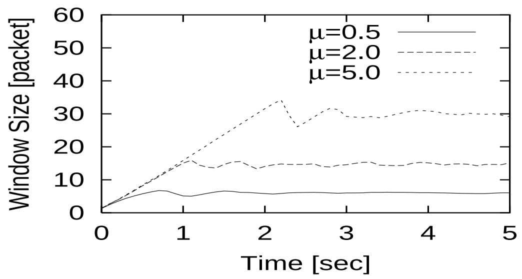


(b) $\lambda_B = 0.2$ [packet/ms]

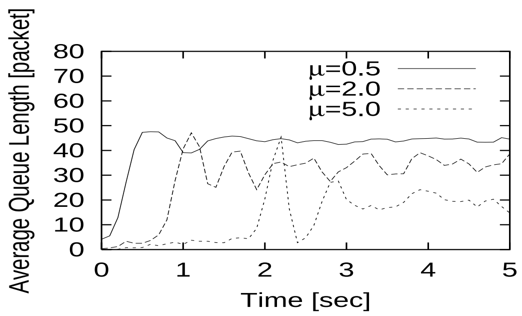


(c) $\lambda_B = 0.5$ [packet/ms]

Figure 11: Maximum modulus of eigenvalues for different arrival rate of background traffic.



(a) TCP window size



(b) Average queue length

Figure 12: Simulation results for different maximum moduli.

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.

3 TCP Connections with Different Propagation Delays

In this section, using a similar modeling approach to that of Section 2, we analyze the network with multiple TCP connections, where each TCP connection is allowed to have different propagation delay. We first present the steady state analysis for a network with a single TCP connection, and derive the TCP throughput, the TCP round-trip time, and the packet loss probability in the network. We then extend it to a network with heterogeneous TCP connections.

3.1 Analytic Model

Figure 13 illustrates our analytic model. The number N of TCP connections share the single bottleneck link, and each TCP connection has a different propagation delay τ_n. 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. Table 3 summarizes the definition of symbols used throughout this thesis. Note that the TCP throughput λ_n is defined as the packet transmission rate from the source host, and is different from the TCP goodput.

3.2 Steady State Analysis

3.2.1 Case of a Single TCP Connection

We first present the steady state analysis for the network with a single TCP connection. The TCP source host is modeled by a SISO (Single Input and Single Output) system; i.e., the packet loss probability p in the network is the input to the system, and the TCP throughput λ is the output. Let p and r be the packet loss probability in the network and the round-trip time of the TCP connection, respectively. The TCP throughput $\lambda(p, r)$ is

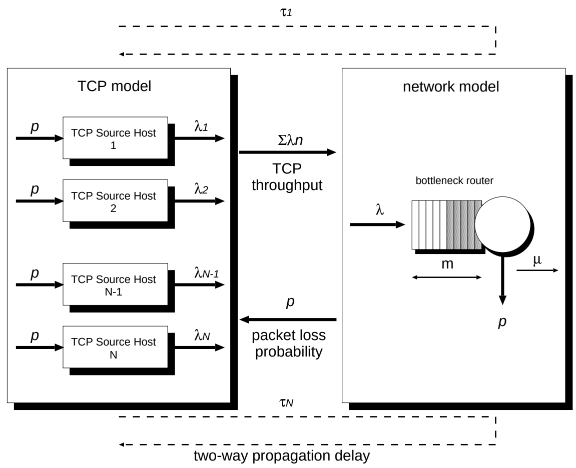


Figure 13: Analytic model of the network with different propagation delays.

approximately given by the following equation [3].

$$\lambda(p, r) \quad \simeq \quad \frac{1}{r}\sqrt{\frac{3}{2\,b\,p}} \tag{14}$$

In the above equation, b (usually $b = 1$ or $b = 2$) is the number of packets required for the TCP destination host to send an ACK packet.

We then model the network seen from the TCP source host by a single FIFO (First-In First-Out) queue. Provided that queueing delay occurs only at the bottleneck router, the bottleneck router is modeled by a M/M/1 queuing system. In other words, the network is modeled by a SISO system, where the input is the TCP throughput λ and the output is the packet loss probability p at the bottleneck router. Let λ and m be the TCP throughput and the buffer size of the bottleneck router. The packet loss probability in the network is approximated as

$$p(\lambda) \quad \simeq \quad 1 - \sum_{k=0}^{m} P_k(\lambda) \tag{15}$$

where $P_k(\lambda)$ is the state probability of a M/M/1 queuing system. By letting μ be the

Table 3: Definition of symbols.

N	the number of TCP connections
μ	packet processing speed of bottleneck router
m	buffer size of bottleneck router
τ_n	two-way propagation delay of nth TCP connection
r_n	round-trip time of nth TCP connection
λ_n	throughput of nth TCP connection
p	packet loss probability in the network

packet processing speed of the bottleneck router, $P_k(\lambda)$ is given by

$$P_k(\lambda) = \left(1 - \frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^k \tag{16}$$

Thus, $p(\lambda)$ is obtained from Eqs. (15) and (16) as

$$p(\lambda) \simeq \left(\frac{\lambda}{\mu}\right)^{1+m} \tag{17}$$

Since the window-based flow control mechanism of TCP operates around the 100% offered traffic load, $p(\lambda)$ in the above equation can be approximated by

$$p(\lambda) \simeq \frac{\lambda + \lambda m - m\mu}{\mu} \tag{18}$$

We then derive the round-trip time of the TCP connection. Let τ be the propagation delay of a TCP connection. We assume that the queueing delay occurs only at the bottleneck router. For a given TCP throughput λ, the TCP round-trip time $r(\lambda)$ is obtained using the average queue length $L = \lambda/(\mu - \lambda)$ of a M/M/1 queueing system and the Little's theorem $N = \lambda T$. Thus,

$$r(\lambda) = \frac{L}{\lambda} + \tau = \frac{1}{\mu - \lambda} + \tau \tag{19}$$

Let λ^*, r^*, and p^* be the TCP throughput, the round-trip time of the TCP connection, and the packet loss probability in the network in steady state, respectively. From Eqs. (14), (18), and (19), the following relations are satisfied in steady state.

$$\lambda(p^*, r^*) = \lambda^*$$

$$p(\lambda^*) = p^*$$

$$r(\lambda^*) = r^*$$

By solving these equations, we can obtain the TCP throughput λ^*, the round-trip time r^* of the TCP connection, and the packet loss probability p^* in steady state. For example, the TCP throughput λ^* in steady state is given by the solution of the following equation.

$$\frac{(\mu - \lambda)\sqrt{\frac{3\,\mu}{2\,b\,(\lambda + \lambda\,m - m\,\mu)}}}{1 - \lambda\,\tau + \mu\,\tau} = \lambda \tag{20}$$

3.2.2 Case of Multiple TCP Connections

In what follows, we extend the previous analysis to a network with heterogeneous TCP connections. Let N be the number of TCP connections, λ_n be the throughput of nth TCP connection, and r_n be the round-trip time of nth TCP connection. We assume random packet losses in the network. From Eq. (14), for a given packet loss probability p in the network, the throughput of each TCP connection is given by

$$\lambda_n(p, r_n) \simeq \frac{1}{r_n}\sqrt{\frac{3}{2\,b\,p}} \tag{21}$$

We then model the network seen from TCP source hosts as a MISO (Multi Input and Single Output) system, where the inputs are throughputs λ_n of TCP connections and the output is the packet loss probability p in the network. From Eq. (15), the packet loss probability in the network with multiple TCP connections, $p(\lambda_1, \cdots, \lambda_N)$, is obtained as

$$p(\lambda_1, \cdots, \lambda_N) \simeq 1 - \sum_{k=0}^{m} P_k(\sum_{n=1}^{N} \lambda_n) \tag{22}$$

Similarly, for a given TCP throughput λ_n, the round-trip time $r_n(\lambda_n)$ of nth TCP connections is obtained from Eq. (19).

$$r_n(\lambda_n) = \frac{1}{\mu - \sum_{n=1}^{N} \lambda_n} + \tau_n \tag{23}$$

where τ_n is the propagation delay of nth TCP connection.

Finally, we derive the equilibrium in steady state. Let λ_n^*, r_n^*, and p^* be the throughput of nth TCP connection, the round-trip time of nth TCP connection, and the packet loss probability in steady state, respectively. We have the following equations from Eqs. (21), (22), and (23).

$$\begin{aligned} \lambda_1(p^*, r_1^*) &= \lambda_1^* \\ &\vdots \\ \lambda_N(p^*, r_N^*) &= \lambda_N^* \\ p(\lambda_1^*, \cdots, \lambda_N^*) &= p^* \\ r(\lambda_1^*) &= r_1^* \\ &\vdots \\ r(\lambda_N^*) &= r_N^* \end{aligned}$$

By numerically solving these equations for λ_n^*, r_n^*, and p^*, the throughput and the round-trip time of nth TCP connection, and the packet loss probability in the network can be obtained.

4 Conclusion

In the frist part of this thesis, 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 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. 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.

In the second part of this thesis, we have extend our analytic approach to a more generic network, where multiple TCP connections are allowed to have different propagation delays. We have derived the packet loss probability in the network, throughput and average round-trip time of each TCP connection in steady state.

As a future work, it would be interesting to apply our approach to the more general heterogeneous network where several bottleneck routers or other congestion control mechanisms exist.

Acknowledgments

I would like to express my sincere appreciation to Professor Masayuki Murata of Osaka University, who introduced me to the area of computer networks including the subjects in this thesis. All works of this thesis would not have been possible without the continuous support of Associate Professor Hiroyuki Ohsaki of Osaka University. It gives me great pleasure to acknowledge his effortless assistance. He has been constant sources of encouragement and advice through my studies and preparation of this manuscript. Thanks are also due to Professor Hideo Miyahara of Osaka University who gave me many insightful advices. I am also indebted to Associate Professor Naoki Wakamiya of Osaka University, Associate Professor Go Hasegawa of Osaka University and Research Assistant Shin'ichi Arakawa of Osaka University who gave me helpful comments and feedbacks. Finally, I thank many friends and colleagues in the Department of Informatics and Mathematical Science of Osaka University for their support.

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*, pp. 303–314, Sept. 1998.

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

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

[6] V. Firoiu and M. Borden, "A study of active queue management for congestion control," in *Proceedings of IEEE INFOCOM 2000*, pp. 1435–1444, 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/`.

[9] K. Takagaki, H. Ohsaki, and M. Murata, "Analysis of a window-based flow control mechanism based on TCP Vegas in heterogeneous network environment," *IEICE Transactions on Communications*, pp. 89–97, Jan. 2002.

[10] 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*, pp. 1603–1606, Aug. 1999.