Performance Modeling of TCP/IP in a Wide-Area Network*

Eitan ALTMAN, Jean BOLOT
Driss ELOUADGHIRI, Mohamed ERRAMDANI and Philippe NAIN
INRIA
2004, Route des Lucioles, B.P. 93
06902 Sophia-Antipolis Cedex
FRANCE

Patrick BROWN and Denis COLLANGE
France Telecom - CNET
905, rue Albert Einstein
06921 Sophia-Antipolis Cedex
FRANCE

Abstract

We examine the problem of evaluating the performance of TCP connections over wide area networks. Our approach combines experimental and analytic methods, and proceeds in three steps. First, we have used measurements taken over Renater to provide a basis for the chosen analytic model. This model turns out to be a shared bottleneck model, in which a finite buffer queue is shared by two connections, one being the reference TCP connection, and the other representing the exogenous traffic (i.e. all the other connections). This model is unlike previous models which did not explicitly consider the impact of exogenous traffic on the reference TCP connection. Second, we use fluid modeling to analyze the behavior of the reference TCP connection. We identify two modes of operation. For each mode, we derive closed-form expressions for the average throughput and delay as a function of buffering, roundtrip delay, and characteristics of exogenous traffic. Third, we use simulation to validate the model, and we find in general good correlation with the analytic results.

1 Introduction

The work in this paper is a response to concerns voiced by an operator about the performance of TCP (Transport Control Protocol [24]) over wide-area networks in general, and over the French

*The work in this paper was supported in part by grant 94-5B-012 from France Telecom/CNET.

research network Renater[1] in particular. Our specific goal was to obtain analytic expressions to analyze, and predict if possible, the time required to transfer large files over Renater. This analysis in turn would be used by the network operator for dimensioning purposes ("is the current bandwidth sufficient to provide reasonable end to end performance?") and for customer performance prediction purposes ("what kind of performance should a new customer expect given the current network configuration?").

Answering the questions above requires that one develop an understanding of TCP behavior as a function of network parameters, including both static parameters (such as buffer size or link bandwidth) and dynamic parameters (such as the dynamic aspects of the TCP control scheme or the characteristics of the traffic generated by other connections). While much is known about the behavior of TCP from observations of simulation and experimental results [23, 19, 21, 14], little is available in terms of closed-form expressions for the delay or throughput of a TCP connection in a wide area network. This is in large part because the complex dynamics of the TCP window flow control mechanism makes it difficult to analyze. Thus, much work so far has relied on experiments or simulation. Early analytical work examined simple models of a single connection in isolation over low and medium size/bandwidth networks [23], and was then extended to consider more complex but more accurate models [20], as well as connections with high bandwidth-delay products [15] and/or asymmetric characteristics [17]. In parallel, other work examined the impact of other connections on a reference TCP connection. This impact has been so far modeled as being that of an additional Bernoulli loss process [15].

Our work is closest in spirit and in approach to, and complements that of, reference [15]. We use a fluid modeling approach similar to that used there, however we explicitly model the impact of other connections on a reference TCP connection. Our general approach has proceeded in 3 steps. First, we have used measurements taken over Renater to provide a basis for the chosen analytic model. Our main result for the purpose of this paper is that the measurements can be interpreted using a shared bottleneck model. Not surprisingly, this result ties in well with that obtained in [4], which used measurements obtained over a variety of connections spanning the Internet.

Second, we have considered a shared bottleneck model of a TCP connection. In this model, a reference TCP connection shares a single bottleneck node with other connections while non bottleneck nodes are modeled by fixed delays. The traffic generated by these connections is assumed to be independent from the behavior of the reference TCP connection. We refer to this traffic as exogenous traffic, and to these connections as non controlled connections (even though some of them might be TCP connections, and hence might be flow controlled connections). We

[1] Renater is an acronym for REseau NAtional de télécommunications pour la Technologie, l'Enseignement, et la Recherche.

model these connections as a single exogenous connection that competes with the reference TCP connection for the resources of the bottleneck node. In practice, we model the exogenous and the reference connection using a fluid model and use it to analyze the behavior of the reference TCP connection. Fluid models have been found to be helpful in providing insights into the dynamic as well as stationary behavior of a variety of feedback control mechanisms [3, 7, 9, 26]. This is important because the dynamic behavior of the TCP flow control mechanism has an important impact on the performance of TCP connections.

Third, we have used simulation to validate the fluid approximation and the analytic results obtained in step 2.

The rest of the paper is organized as follows. In Section 2, we briefly describe the TCP flow control mechanism and our fluid model of it. This model turns out to have two modes of operation. We analyze these modes in Sections 3 and 4, respectively. In Section 5, we describe and analyze the simulation results obtained to validate the analytic results. Section 6 concludes the paper.

2 Modeling TCP flow control

The TCP flow control scheme

Flow control mechanisms regulate the flow of packets into a network to prevent network resources from becoming congested and to make sure that these resources are shared fairly between different users. This regulation is typically done using feedback information about the state (more or less congested) of the network. In TCP, the state of the network is characterized by packet losses and the regulation is done using a dynamic window scheme [11].

The specifics of the control mechanism are as follows. Packets are assigned increasing sequence numbers. The source TCP sends in each packet continuous data octets accompanied by the sequence number of the first octet. The destination TCP maintains a set of continuous sequence numbers. When it receives a packet, the destination TCP sends an acknowledgment (ack) packet indicating the value of the receive window and the next expected octet sequence number. The receive window indicates how much buffer space is available for this connection at the destination host. Its size is fixed. Data octets below the receive window have been passed on to the application. Data octets received out of sequence but within the receive window are buffered.

The source TCP maintains another window called the congestion window, equal to the maximum allowed number of unacknowledged packets. The congestion window size is adjusted dynamically in response to ack reception and packet loss. A packet loss in turn is detected either with the receipt of 3 duplicate acks (i.e. consecutive acks that have the same "next expected" sequence number), or with the expiration of a timer. The source then is allowed to send min(receive window, congestion window) packets.

The congestion window size adjustments work in cycles made up of two phases, namely the *slow-start phase* and the *congestion-avoidance phase*. The window size is initially set to 1. In the slow-start phase, the window size is increased by one every time an ack is received. Thus, when an ack arrives at the source, two packets are generated, one for the received ack and one because the window size is increased by one. This behavior causes an exponential increase of the window size as a function of time, and the amount of data in transit over the connection increases rapidly. The slow-start phase ends when the window reaches a certain level called the *slow-start threshold*.

At this point the congestion avoidance phase starts. The purpose of this phase is to slowly increase the load over the connection so as to adapt to and probe for the available bandwidth. This is done by increasing the current size W of the window by 1/[W] whenever a packet is acknowledged[2]. Thus, the window size increases by 1 when W packets have been received, i.e. roughly every roundtrip. When a packet is lost, the slow-start threshold is set to half the size of the window, the window is then set to 1, and a new cycle begins.

The control algorithm described above is referred to in the literature as "TCP Tahoe". Other versions such as "Reno" [12] and "Vegas" [6] have been proposed recently. They differ from Tahoe by slightly different window adjustments and packet loss detection schemes, and we will not consider them in this paper.

The shared bottleneck model

We next turn to the problem of modeling and analyzing the performance of a reference Tahoe TCP connection in a wide-area network. As indicated in Section 1, we consider a shared bottleneck model. In this model, the reference TCP connection shares a finite buffer FIFO bottleneck mode with other connections. Thus, the arrival stream at the bottleneck queue is the superposition of two streams, namely the reference TCP stream and the exogenous stream, which is in turn the superposition of many streams active during the TCP connection lifetime. We assume in this section that the exogenous stream is a constant rate stream independent of the state of the network, i.e. independent of the behavior of the TCP connection under study and of the load in the bottleneck queue (we examine in Section 5 how the performance measures are impacted when the exogenous stream is a Poisson stream). We use a constant delay τ to

[2] [x] denotes the integer part of x.

model the fixed component of the round trip delay of the reference TCP packets. Refer to Figure 1.

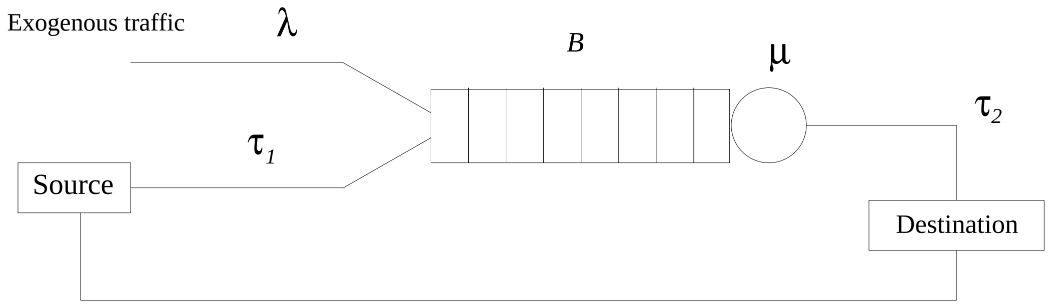

Figure 1: The shared bottleneck mode

The parameters of the model are

- $W(t) \triangleq$ window size at time t. We take $W(0) = 1$.
- $W_{\text{th}}(t) \triangleq$ value of the slow-start threshold at time t.
- $Q(t) \triangleq$ number of packets in the queue (from the TCP and exogenous stream) at time t.
- $B \triangleq$ maximum buffer size.
- $\mu \triangleq$ service rate of the queue.
- $\tau_1 \triangleq$ time between the transmission of a packet until the packet reaches the queue.
- $\tau_2 \triangleq$ time between the departure of a packet from the queue until the packet reaches the source via the destination.
- $\tau \triangleq$ propagation delay, i.e. the round trip delay (not including the service time) when the queue is empty.
- $\lambda \triangleq$ rate at which exogenous packets are transmitted. We assume $\lambda < \mu$.
- $T \triangleq \tau + \mu^{-1}$, the sojourn time of a packet in an empty system, i.e. the round trip delay plus the service time μ^{-1}.

For convenience, we introduce

- $\beta \triangleq B/[\tau(\mu - \lambda) + 1]$, the normalized buffer size (this definition matches that of [15] when $\lambda = 0$).

- $W_{\text{loss}} \triangleq$ window size at the end of a congestion avoidance phase before a loss occurs.

- $W_{\max} \triangleq$ maximal size attained by the window at the end of a congestion avoidance phase (just after a loss occurs).

Preliminary observations

The evolutions of the window size $W(t)$ are cyclical. Indeed, the window size initially grows rapidly during slow start until it reaches $W_{th}(t)$. At this point, congestion avoidance kicks in and the window size grows slowly until it reaches a maximum size $W_{\max}$, at which point there is a packet loss. The window size then drops to one and a new cycle starts. Thus, the long-term average throughput of the TCP connection can be computed as the number of TCP packets successfully transmitted in a cycle, divided by the cycle duration. Similarly, the long term average delay for TCP packets can be computed as the average delay of a TCP packet over a cycle. Let C denote the average duration of a cycle, N denote the average number of TCP packets successfully transmitted in a cycle, and R denote the average sojourn time of TCP packets successfully transmitted in a cycle. The performance measures of interest in this paper are the average throughput $\overline{thp}$ and the average round-trip delay $\overline{rtt}$ of the reference TCP connection defined by

$$\overline{thp} = \frac{N}{C} \tag{1}$$

$$\overline{rtt} = \frac{R}{N}. \tag{2}$$

Our goal then is to compute C, N, and R by analyzing the dynamic behavior of the window size. For convenience, we define in this paper a cycle to be a period starting just after the loss of a packet from the reference TCP connection, given that this loss occurred during the congestion avoidance phase, and ending just before the next such loss. A cycle thus always includes at least one slow-start phase, and it ends during a congestion-avoidance phase.

The cyclic behavior of $W(t)$ has been examined in [15] in the absence of exogenous traffic, i.e. when $\lambda = 0$. It was observed there that there are only two possible types of cyclic behavior, given the independence and the bottleneck assumptions above. One type of cyclic regime includes a single slow-start phase per cycle, the other two slow-start phases per cycle.

This results still essentially holds in the presence of exogenous traffic. We have found parameter values that yield three slow start phases in a cycle (see Figure 6 in Section 5). However in the vast majority of cases we find that the evolutions of $W(t)$ include a single or two slow start phases per cycle, as shown in Figures 2 and 3, respectively (the figures were obtained with the REAL simulator [13]).

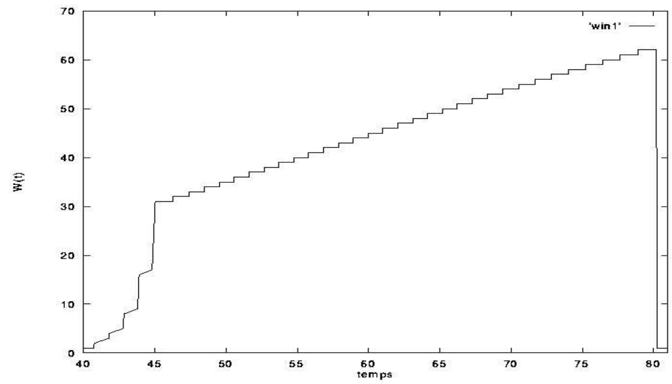

Figure 2: Evolutions of $W(t)$ with a single slow-start phase per cycle

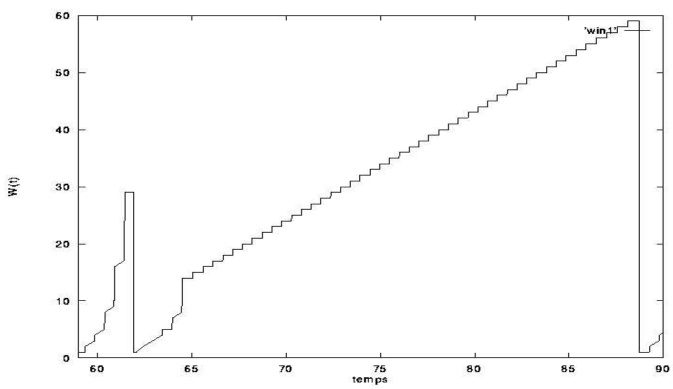

Figure 3: Evolutions of $W(t)$ with two slow-start phases per cycle

In the next two sections, we analyze the case of a single slow-start phase per cycle (Section 3) and the case of two slow start phases per cycle (Section 4), and we find the conditions when these cases occur.

3 Case 1: A single slow-start phase per cycle

Our main result in this section are analytic expressions for the average throughput $\overline{thp}$ and the average delay $\overline{rtt}$ as a function of system parameters. This result is expressed in the theorem below.

Theorem 3.1 *We have*

$$W_{\rm loss} = (B/\mu + T)\,(\mu - \lambda) + \lambda/\mu,$$

$$W_{\max} = W_{\rm loss} + 2, \tag{3}$$

$$W_{\rm th} = W_{\max}/2,$$

$$\overline{thp} = \frac{N_1 + N_2 + N_3}{T_1 + T_2 + T_3}, \tag{4}$$

$$\overline{rtt} = \frac{N_1 R_1 + N_2 R_2 + N_3 R_3}{N_1 + N_2 + N_3}, \tag{5}$$

where N_i and T_i $(i = 1, 2, 3)$ are given by the following, and $\tilde{W} = T(\mu - \lambda)$:

(i) If $\tilde{W} < W_{\rm th}$ then

$$T_1 = T\log(\tilde{W}), \quad N_1 = \tilde{W} - 1, \quad \textit{and} \quad R_1 = T,$$

$$T_2 = \frac{W_{\rm th} - \tilde{W}}{\mu - \lambda/2}, \; N_2 = W_{\rm th} - \tilde{W}, \quad \textit{and} \quad R_2 = \frac{W_{\rm th} + \tilde{W} - 2}{2\mu - \lambda} + \frac{1}{\mu} - \tau_1, \tag{6}$$

$$T_3 = \frac{3}{2}\frac{W_{\rm th}^2}{\mu - \lambda}, \; N_3 = \frac{3}{2}W_{\rm th}^2, \quad \textit{and} \quad R_3 = \frac{1}{\mu - \lambda}\left(\frac{14}{9}W_{\rm th} - \frac{\lambda}{\mu}\right).$$

(ii) If $\tilde{W} \geq W_{\rm th}$ then

$$T_1 = T\log(W_{\rm th}), \; N_1 = W_{\rm th} - 1, \quad \textit{and} \quad R_1 = T,$$

$$T_2 = T(\tilde{W} - W_{\rm th}), \; N_2 = \frac{\tilde{W}^2 - Wt^2}{2}, \quad \textit{and} \quad R_2 = T,$$

$$T_3 = \frac{4W_{\rm th}^2 - tW^2}{2(\mu - \lambda)}, \quad N_3 = 4W_{\rm th}^2 - \frac{\tilde{W}^2}{2}, \quad \textit{and} \quad R_3 = \frac{1}{\mu - \lambda}\left(\frac{2}{3}\frac{4W_{\rm th}^2 + 2W_{\rm th}\tilde{W} + \tilde{W}^2}{2W_{\rm th} + \tilde{W}} - \frac{\lambda}{\mu}\right).$$

Proof: We first discuss the evolution of the window size over time [11]. If an acknowledgment arrives at time s, then

$$W(s) = \begin{cases} W(s^-) + 1 & \text{if } W(s^-) < W_{\text{th}}(s^-) \text{ (slow-start phase)} \\ W(s^-) + 1/\lfloor W(s^-) \rfloor & \text{otherwise (congestion avoidance phase)} \end{cases} \tag{7}$$

where $\lfloor \cdot \rfloor$ denotes the integer part of the argument. If a loss is detected at time s, then

$$W_{\text{th}}(s) = \frac{W(s^-)}{2} \tag{8}$$

and $W(s)$ is set to one.

Let $ack(s)$ denote the number of acknowledgements that have returned by time s. We will approximate the dynamics (7) by

$$\frac{dW}{dack} = \begin{cases} 1 & \text{if } W < W_{\text{th}} \\ W^{-1} & \text{if } W \geq W_{\text{th}} \end{cases} . \tag{9}$$

Define the following quantities:

- $thp_{in}(s) \triangleq$ input rate of fluid originating from the controlled source.
- $thp_{out}(s) \triangleq$ rate of fluid originating from the controlled source, at the output of the queue.
- $\lambda_{out}(s) \triangleq$ rate of fluid originating from the exogenous sources, at the output of the queue.
- $rtt(s) \triangleq T + Q(s)/\mu$, the total sojourn time of a packet, i.e. the round trip delay plus the queuing delay (taking into account that the queue is nonempty).

Note that the rate at which acknowledgments arrive back to the source is equal to thp_{out} as long as there are no losses of acknowledgements. The total number of packets N transmitted successfully during a cycle can be expressed as

$$N = \int_0^C thp_{out}(s)\ ds. \tag{10}$$

To obtain $thp_{out}(s)$, we observe that

$$\frac{dW}{dt} = \frac{dW}{dack}\frac{dack}{dt} = \frac{dW}{dack}\ thp_{out} \tag{11}$$

where dW/dt is the rate at which the window grows as a function of time. Together with (9), this implies

$$\frac{dW}{dt} = \begin{cases} thp_{out} & \text{if } W < W_{\text{th}} \\ thp_{out}/W & \text{if } W \geq W_{\text{th}}. \end{cases} \tag{12}$$

As long as the queue is empty, we have

$$thp_{out}(t) = thp_{in}(t) = W(t)/T \tag{13}$$

and $\lambda_{out} = \lambda$. When the queue starts building up, we have

$$thp_{out}(t) + \lambda_{out} = \mu.$$

Hence, it follows that the queue starts building up at time $\hat{t}$ when W reaches the value

$$W(\hat{t}) = \tilde{W}. \tag{14}$$

When the queue is nonempty, the output rates of both the controlled as well as the exogenous traffic are smaller than the input rates. It is then reasonable to assume that the output rates are proportional to the input rates, namely

$$thp_{out}(t) = \mu \, \frac{thp_{in}(t)}{thp_{in}(t) + \lambda} \qquad \text{and} \qquad \lambda_{out}(t) = \mu \, \frac{\lambda}{thp_{in}(t) + \lambda}. \tag{15}$$

Therefore,

$$thp_{in}(t) = thp_{out}(t) \, \frac{\lambda}{\mu - thp_{out}}.$$

Another equation that relates the input and output rates of the controlled traffic is obtained by noting that the input rate is the sum of the output rate and of the rate at which the window size grows. By using the relation

$$thp_{in}(t) = \frac{dW}{dt} + \frac{dack}{dt} = \left(1 + \frac{dW}{dack}\right) thp_{out}(t)$$

we obtain

$$thp_{out}(t) = \mu - \frac{\lambda}{1 + \dfrac{dW}{dack}}. \tag{16}$$

We make the reasonable assumption that $W^{-1} << 1$ during the congestion avoidance phase, which, by (9), implies that $dW/dack$ can be neglected in (16) (provided that λ is not too close to μ). When the queue is non empty, we obtain

$$thp_{out}(t) = \begin{cases} \mu - \lambda/2 & \text{during the slow-start phase} \\ \mu - \lambda & \text{during the congestion-avoidance phase.} \end{cases} \tag{17}$$

We will discuss implications of (17) below. Collecting all the results obtained so far, we get

(i) If $\tilde{W} < W_{\text{th}}$

$$\frac{dW}{dt} = \begin{cases} W/T & \text{if } W \le \tilde{W} \\ \mu - \lambda/2 & \text{if } \tilde{W} < W < W_{\text{th}} \\ (\mu - \lambda)/W & \text{if } W \ge W_{\text{th}} \end{cases} \tag{18}$$

(ii) If $\tilde{W} \ge W_{\text{th}}$

$$\frac{dW}{dt} = \begin{cases} W/T & \text{if } W \le W_{\text{th}} \\ 1/T & \text{if } W_{\text{th}} < W < \tilde{W} \\ (\mu - \lambda)/W & \text{if } W \ge \tilde{W}. \end{cases} \tag{19}$$

Let time $\tilde{t}$ denote the time at which $W_{\max}$ is reached. The queue is then full, so that the number of packets at time $\tilde{t}$ originating from the controlled source in the queue is equal to

$$B \, \frac{thp_{in}(\tilde{t})}{thp_{in}(\tilde{t}) + \lambda}.$$

W_{loss} is then obtained as follows:

$$W_{\text{loss}} - \tau_1 thp_{in}(\tilde{t}) - \left(\tau_2 + \frac{\tau}{2}\right) thp_{out}(\tilde{t}) - 1 = B \, \frac{thp_{in}(\tilde{t})}{thp_{in}(\tilde{t}) + \lambda}.$$

Since by (15) and (17) we have $thp_{in} = thp_{out}(\tilde{t}) = \mu - \lambda$, equation (3) follows. When the queue is nonempty, the number of controlled packets in it is given by

$$W(t) - \tau_1 thp_{in}(t) - \left(\tau_2 + \frac{\tau}{2}\right) thp_{out}(t) - 1 = Q(t) \, \frac{thp_{in}(t)}{thp_{in}(t) + \lambda}.$$

From (18) and (19) we may conclude that there are three periods within a cycle: one when the window grows exponentially fast, another one when it grows linearly, and a third one when it grows sub-linearly.

Define now

- $T_i \triangleq$ duration of the ith such period ($i = 1, 2, 3$).
- $t_i \triangleq$ time at which the ith period ends.
- $N_i \triangleq$ number of packets transmitted in period i.

- $R_i \triangleq$ average delay of a packet in period i.

We have $W_{\text{th}} = W_{\max}/2$, where $W_{\max}$ is given in (3). The average throughput is given by (4), and the average delay by (5), with

$$N_i = \int_{t_{i-1}}^{t_i} thp_{out}(s)\, ds \tag{20}$$

and

$$R_i = T + \frac{1}{T_i} \int_{t_{i-1}}^{t_i} \frac{Q(s)}{\mu}\, ds. \tag{21}$$

In order to determine N_i and R_i we will need to distinguish the following two cases:

Case 1: $\tilde{W} < W_{\text{th}}$

We solve the system (18). The condition $\tilde{W} < W_{\text{th}}$ means that the queue starts to build up during the slow-start phase. This case decomposes into three subcases:

- $W \leq \tilde{W}$ (slow-start phase)

  This subcase begins at time 0 with $W(0) = 1$ and ends at time t_1 with $W(t_1) = \tilde{W}$. Integrating (18-a) with the initial condition $W(0) = 1$ yields $W(t) = \exp(t/T)$ as long as the queue is empty. Since the queue starts to build up at time T_1 we have

  $$e^{t/T} = W(T_1) = \tilde{W} \tag{22}$$

  so that $T_1 = T\,\log(\tilde{W})$. Combining now (13) and (22) yields $thp_{out}(t) = \exp(t/T)/T$ for $0 < t < T_1$, so that $N_1 = \tilde{W}$ from (20). Since during that time $Q(t) = 0$, we have $R_1 = T$.

- $\tilde{W} < W < W_{\text{th}}$ (slow-start phase)

  We consider the interval between t_1 and t_2, with boundary conditions $W(t_1) = \tilde{W}$ and $W(t_2) = W_{\text{th}}$. Integrating (18-b) yields

  $$W = (\mu - \frac{\lambda}{2})(t - t_1) + \tilde{W}.$$

  T_2 in (6) is then obtained by substituting the boundary condition $W(t_2) = W_{\text{th}}$. N_2 is obtained by combining (17-c) with (20), and R_2 is obtained from (21) by noting that during this period

  $$Q(t) = \frac{\mu}{\mu - \lambda/2}(W(t) - 1) - \mu(\tau + \tau_1). \tag{23}$$

- Congestion avoidance phase

  We have $W \geq W_{\text{th}}$. Hence $W(t_2) = W_{\text{th}}$ and $W(t_3) = W_{\max}$. Integrating (18-c), we obtain

  $$W = \sqrt{2(t - t_2)(\mu - \lambda) + W_{\text{th}}^2}$$

  which yields T_3 by using the boundary condition $W(t_3) = W_{\max}$. N_2 is obtained by combining (17-b) with (20), and R_2 is obtained from (21) by noting that during this period

  $$Q(t) = \frac{\mu}{\mu - \lambda}(W(t) - 1) - \mu\tau. \tag{24}$$

Case 2: $\tilde{W} \geq W_{\text{th}}$

We solve the system (19). The queue starts to build during the congestion avoidance phase. Again, this case decomposes into three subcases.

- Slow-start phase

  The results are obtained in the same way as in the first subcase of (i), except that the boundary condition changes to $W(t_1) = W_{\text{th}}$.

- $W_{\text{th}} < W < \tilde{W}$ (congestion avoidance phase) We have $W(t_1) = W_{\text{th}}$ and $W(t_2) = \tilde{W}$, We integrate (19-b) and obtain $W = (t - t_1)/T + W_{\text{th}}$. Substituting $W(t_2)$ we obtain T_2. This, together with (13) and (20) yields N_2. $R_2 = T$ since the queue is empty.

- $W \geq \tilde{W}$ (congestion avoidance phase)

  We have $W(t_2) = \tilde{W}$ and $W(t_3) = W_{\max}$. We integrate (19-c) and obtain

  $$W = \sqrt{2(t - t_2)(\mu - \lambda) + \tilde{W}^2}.$$

  Substituting $W(t_3)$, we obtain T_3. N_3 is obtained by using (17-b) with (20), and R_3 is obtained by using (21) and by noting that during this period

  $$Q(t) = \frac{\mu}{\mu - \lambda}(W(t) - 1) - \mu\tau.$$

This concludes the proof of our main result. ■

Observe that we have made the assumption in our analysis that losses are detected immediately after they occur. This assumption is reasonable with the duplicate ack (refer to Section 2) or a selective ack scheme. However, when losses are detected with a timer expiration, the length of a cycle is equal to $T_1 + T_2 + T_3 + rto$, where rto is the value of the timer (rto =500 ms with the coarse TCP timer [24]). Also, a whole window worth of packets (of size $W_{\max}$) can still be transmitted between the actual loss and the time at which the loss is detected. Therefore, equation (4) then becomes

$$\overline{thp} = \frac{N_1 + N_2 + N_3 + W_{\max}}{T_1 + T_2 + T_3 + rto}. \tag{25}$$

We next illustrate the utilization of the above analytical results. We consider the fraction of the available throughput used by TCP, i.e. $thp/(\mu - \lambda)$. This quantity indicates how well TCP uses the residual capacity left by the exogenous traffic. In an ideal situation, it should be close to one. A value far below one indicates link underutilization (as we see is the case when b is small in figure 5), while a value (even slightly) above one indicates that the TCP connection does not allow for the exogenous traffic to flow. In all cases, we take the unit of time to be one bottleneck queue service time.

Fig. 4 shows the fraction of the available throughput used by TCP as a function of the round-trip time τ for different values of λ when the buffer size is $B = 20$. For small values of τ, TCP uses a higher percentage of the available throughput for higher exogenous traffic rates, even exceeding a ratio of one. For large values of τ the opposite is true. In all cases this ratio decreases with τ and this at a rate which increases with the exogenous traffic intensity.

In Fig. 5, τ is set to 30 while the buffer size varies. The fraction of the available throughput used by TCP is shown for different values of λ. As we saw was the case for high values of τ, TCP uses more effectively the available capacity for smaller values of λ. Furthermore, we observe that this is true independent of the buffer size B. Also, we note that the TCP throughput never exceeds 1. However, for low values of B, the underutilization can be as high as 30%.

The above observations suggest that we examine in more detail the dynamic behavior of the throughput, which is given by equation (17). We note that the exogenous traffic behaves as if it had full priority over the TCP traffic during the congestion avoidance phase, Indeed, its throughput is equal then to λ, i.e. to the input rate of exogenous traffic. We have actually observed this in simulations. This behavior is not there in the slow-start phase though, because two consecutive controlled packet are transmitted at every ack arrival, so that the controller is more aggressive in using the available bandwidth at the expense of the exogenous traffic. This "hogging" of bandwidth has an important impact in practice. Consider for example the case when the exogenous traffic is constant rate UDP audio traffic. Then the result above says that the audio data in a way gets priority over the TCP data most of the time since the

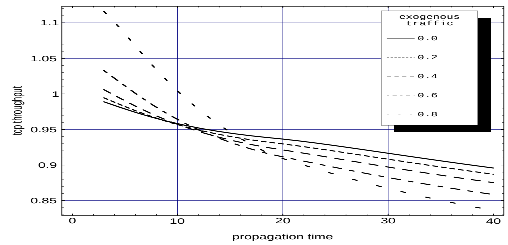

Figure 4: Available throughput utilization vs. propagation time for $B = 20$ and different λ

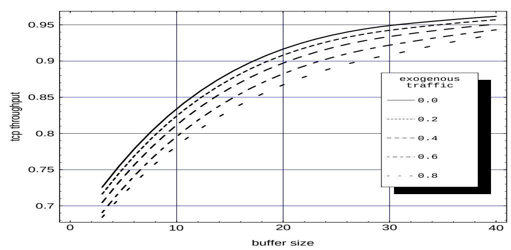

Figure 5: Available throughput utilization vs. buffer size for $\tau = 30$ and different λ

congestion avoidance phases are typically much longer than the slow-start phases. This could be thought of as a nice feature since it gives extra resources to traffic that is sensitive to delay and loss. However, this also implies that the widespread use of audio tools (not an unreasonable occurrence at all [8]) might shut down or at least severely impact TCP connections such as mail or Web transfer connections. This in turn is an additional motivation for incorporating flow control mechanisms in audio (but also video and other UDP-based) tools [5, 18].

4 Case 2: Two slow-start phases per cycle

The analysis of case 2 requires that we start with a more detailed analysis of the window behavior so as to

(i) examine under which conditions a loss occurs in a slow-start phase. We will show that the condition is

$$\beta \leq (3 - \lambda/\mu)^{-1}. \tag{26}$$

This condition reduces to the condition $\beta \leq 1/3$ obtained in [15] for the case $\lambda = 0$.

(ii) when such loss is possible, we will predict the time (and the corresponding value of the window size W) at which the loss occurs. We will then use a fluid approach to compute the performance of the system.

A very detailed analysis of losses during a slow-start phase is given in [15] in the absence of exogenous traffic. We next present an analysis which extends that in [15] to include exogenous arrivals.

Assume first that there is no exogenous traffic. One may define mini-cycles (as in [15]), the duration of which is equal to the round trip time $T = \tau + \mu$ of a packet. When a cycle starts, a single packet is transmitted. The first mini-cycle lasts until the ack is received at the source. After time T, the second mini-cycle starts when two packets are transmitted one after the other. As long as we are in the slow-start phase and the queue is not full, we observe that 2^{n-1} packets are transmitted during the nth mini-cycle. The acknowledgments for the 2 packets transmitted consecutively during the 2nd mini-cycle arrive back at the source separated by a delay equal to a service time $1/\mu$. Since each acknowledgment received during the slow-start phase results in the immediate transmission of two packets, the rate at which packets are transmitted is thus 2μ. Iterating this type of argument shows that at the beginning of the nth mini-cycle, a train (or a burst) of 2^n packets is transmitted at rate 2μ. Note that when the trains arrive to the queue, whose server has a rate μ, the queue builds up at a rate of $2\mu - \mu = \mu$.

For the above decomposition into mini-cycles to be valid, it is necessary that the time it takes for the queue to build up and then empty be shorter than the round trip delay τ. A sufficient condition for this to hold is $2B/\mu < \tau$, or $\beta < 1/2$.

Next we consider the case of exogenous traffic with rate $\lambda < \mu$. The trains of TCP packets now have a rate lower than 2μ, since acknowledgments come back at the source on average at a rate smaller than μ (since exogenous packets are also served at the queue). Let α_n be the rate at which packets are transmitted by the controlled source during the nth mini-cycle. We make the natural assumption that the service capacity μ is shared between the exogenous traffic and the controlled one proportionally to their input rates. Hence, *during a burst*, the throughput of controlled packets is given by

$$\mu \times \frac{\alpha_n}{\alpha_n + \lambda}.$$

This is also the rate at which acknowledgments arrive back at the source, so that the rate at which packets are transmitted during the next mini-cycle is

$$\alpha_{n+1} = \frac{2\alpha_n}{\alpha_n + \lambda}\mu. \tag{27}$$

The solution α_n of the recursion (27) is

$$\alpha_{n+1} = \frac{\alpha_1}{\left(\frac{\lambda}{2\mu}\right)^n + \frac{\alpha_1}{2\mu}\left(\frac{1-(\lambda/2\mu)^n}{1-\lambda/2\mu}\right)}.$$

Since $(\lambda/\mu) < 1$, we observe from the above that α_n converges rapidly toward $2\mu - \lambda$. Hence, we will assume that during the bursts, the input rate of the exogenous traffic is equal to $2\mu - \lambda$.

In order to determine when a loss occurs, we note that during a burst, the total input rate into the queue is 2μ, while the output rate is μ. Hence, the queue builds up at rate μ, and it takes time B/μ from the moment when the burst arrives at the queue until a loss occurs (provided the burst is at least that long). At time B/μ, the amount of controlled packets that have arrived during that burst is

$$W_b = (2\mu - \lambda)\ \frac{B}{\mu}. \tag{28}$$

Hence, a loss occurs during a slow-start phase if W_b is less than or equal to the slow-start threshold W_{th} computed in (3). The number of mini-cycles n_b until the loss occurs is calculated as follows. The time from the beginning of the mini-cycle till the loss occurs was shown to be B/μ, during which 2^{n_b-1} (controlled) packets should have been transmitted approximately at a rate of $2\mu - \lambda$. Therefore,

$$2^{n_b-1}\left(\frac{2\mu-\lambda}{\mu}\right) = B$$

so that

$$n_b = \log_2\left(B\left(\frac{2\mu-\lambda}{\mu}\right)\right).$$

This value, as well as (28), coincide with that derived in [15] for the case $\lambda = 0$. The condition $W_b \le W_{\text{th}}$ is equivalent to

$$2(2-\lambda/\mu)B \le B(1-\lambda/\mu) + T(\mu-\lambda) + \mu/\lambda$$

so that

$$\frac{B}{\tau(\mu-\lambda)+1} \le \frac{1}{3-\lambda/\mu}$$

which is the condition (26) for a loss to occur in a slow-start phase.

We are now ready to analyze the fluid model for our system. We use the methodology and notation from section 3. We assume that $W_b \leq W_{\text{th}}$ (i.e. that (26) holds). The first part of the cycle is a slow-start phase, during which a loss occurs when the window size reaches W_b. The window size then drops to one and a second part of the cycle starts: another slow-start phase starts; when the window size reaches the new threshold $\tilde{W}_{\text{th}} = W_b/2$, a congestion-avoidance phase starts; the cycle ends when $W(s)$ reaches $W_{\max}$ (see (3)), and then a loss (of a controlled packet) occurs. All calculations for quantities during the second part of the cycle are the same as those in the previous section, except that $\tilde{W}_{\text{th}}$ now replaces W_{th}. It thus only remains to determine the behavior of the window during the first part of the cycle. The calculations of this part follow again those for the slow-start phase in the previous section. Specifically, this phase contains either only one type of behavior of the window (i.e. exponential growth), or also a linear growth. The phases of the whole cycle change at times $\tilde{t}_1 \leq \tilde{t}_2 < t_1 < t_2 < t_3$, where t_3 is the instant at which the cycle ends, and $\tilde{t}_2$ is the instant where the first slow-start phase ends. We have to determine the quantities $\tilde{T}_i, \tilde{N}_i, \tilde{R}_i,\ i = 1, 2$, and $T_i, N_i, R_i,\ i = 1, 2, 3$. We obtain the following results:

- $\tilde{W} \geq W_b$: In this case, the performance measures corresponding to the first slow-start phase can be calculated as if the queue was always empty (note that this contradicts the more granular approach, used above to compute W_b). We obtain

$$\tilde{T}_1 = T \log(W_b), \qquad \tilde{N}_1 = W_b - 1, \quad \text{and} \quad \tilde{R}_1 = 0.$$

  We may set $\tilde{T}_2 = \tilde{N}_2 = \tilde{R}_2 = 0$.

- $\tilde{W} < W_b$: We obtain
  We obtain

$$\tilde{T}_1 = T \log(\tilde{W}), \quad \tilde{N}_1 = \tilde{W} - 1, \quad \text{and} \quad \tilde{R}_1 = T,$$

$$\tilde{T}_2 = \frac{W_b - \tilde{W}}{\mu - \lambda/2}, \quad \tilde{N}_2 = W_b - \tilde{W}, \quad \text{and} \quad \tilde{R}_2 = \frac{W_b + \tilde{W} - 2}{2\mu - \lambda} + \frac{1}{\mu} - \tau_1.$$

- $\tilde{W} < \tilde{W_{\text{th}}}$: We obtain

$$T_1 = T\log(\tilde{W}), \quad N_1 = \tilde{W} - 1, \quad \text{and} \quad R_1 = T,$$

$$T_2 = \frac{\tilde{W_{\text{th}}} - \tilde{W}}{\mu - \lambda/2}, \quad N_2 = \tilde{W_{\text{th}}} - \tilde{W}, \quad \text{and} \quad R_2 = \frac{\tilde{W_{\text{th}}} + \tilde{W} - 2}{2\mu - \lambda} + \frac{1}{\mu} - \tau_1,$$

$$T_3 = \frac{4W_{\text{th}}^2 - \tilde{W_{\text{th}}}^2}{2(\mu - \lambda)}, \quad N_3 = \frac{4W_{\text{th}}^2 - \tilde{W_{\text{th}}}^2}{2}, \quad \text{and} \quad R_3 = \frac{1}{\mu - \lambda}\left(\frac{2}{3}\frac{4W_{\text{th}}^2 + \tilde{W_{\text{th}}}^2 + 2W_{\text{th}}\tilde{W_{\text{th}}}}{2W_{\text{th}} + \tilde{W_{\text{th}}}} - \frac{\lambda}{\mu}\right).$$

- $\tilde{W} \geq \tilde{W_{\text{th}}}$: We obtain

$$T_1 = T\log(\tilde{W_{\text{th}}}), \quad N_1 = \tilde{W_{\text{th}}} - 1, \quad \text{and} \quad R_1 = T,$$

$$T_2 = T(\tilde{W} - \tilde{W_{\text{th}}}), \quad N_2 = \frac{\tilde{W}^2 - \tilde{W_{\text{th}}}^2}{2}, \quad \text{and} \quad R_2 = T,$$

$$T_3 = \frac{4W_{\text{th}}^2 - \tilde{W}^2}{2(\mu - \lambda)}, \quad N_3 = \frac{4W_{\text{th}}^2 - \tilde{W}^2}{2}, \quad \text{and} \quad R_3 = \frac{1}{\mu - \lambda}\left(\frac{2}{3}\frac{4W_{\text{th}}^2 + \tilde{\tilde{W}}^2 + 2W_{\text{th}}\tilde{W}}{2W_{\text{th}} + \tilde{W}} - \frac{\lambda}{\mu}\right).$$

The average throughput is then given by

$$\overline{thp} = \frac{\tilde{N_1} + \tilde{N_2} + N_1 + N_2 + N_3}{\tilde{T_1} + \tilde{T_2} + T_1 + T_2 + T_3}$$

and the average round trip is given by

$$\overline{rtt} = \frac{\tilde{N_1}\tilde{R_1} + \tilde{N_2}\tilde{R_2} + N_1R_1 + N_2R_2 + N_3R_3}{\tilde{N_1} + \tilde{N_2} + N_1 + N_2 + N_3}.$$

5 Simulation results

We next present some results from extensive simulation work with the REAL simulator [13] done to validate the analytical results above. The differences between the simulation and the analytic models are i) the simulation model deals with discrete-sized packets (while the simulation model deals with fluid flows), and ii) the simulation model incorporates the exact and complete TCP Tahoe flow control mechanism (while the analytic model incorporates an approximation of it).

We have carried out simulations for different values of the input parameters λ, μ, τ_1, τ_2, and B so as to cover all cases described in the analysis in Sections 3 and 4. We introduce the additional parameter BW_{in} which denotes the bandwidth of the link that connects the source to the bottleneck queue. For the sake of conciseness, we only examine a few cases here (more complete results are available in [1, 2]). In all cases here, the packet size is constant and equal to 576 bytes.

We first consider the case when the exogenous traffic is a periodic process. Table 1 below shows the values of $\overline{thp}$ and $\overline{rtt}$ obtained with the simulation and analytic model, and the relative error between them, for different input parameters. Each set is associated a "case number".

	Parameters						$\overline{thp}$			$\overline{rtt}$		
case	B	BW_{in}	μ	λ	τ_1	τ_2	Ana.	Sim.	%err	Ana.	Sim.	%err
1	05	300	170	10	0.05	0.05	082.1	089.9	07.7	0.222	0.214	3.3
2	15	300	170	40	0.06	0.04	122.9	133.6	08.0	0.248	0.240	3.4
3	10	290	160	10	0.07	0.08	082.7	091.1	09.2	0.332	0.321	3.5
4	10	150	060	22	0.15	0.15	034.9	038.2	08.6	0.685	0.675	1.4
5	10	150	060	22	0.08	0.15	036.1	039.0	07.4	0.557	0.539	3.1
6	05	150	080	10	0.11	0.11	036.2	038.7	06.4	0.481	0.473	1.6
7	05	100	060	20	0.06	0.06	036.7	038.0	03.4	0.312	0.303	2.9
8	10	100	060	20	0.08	0.05	039.3	040.8	03.6	0.379	0.353	7.3
9	05	100	060	05	0.15	0.10	028.9	030.7	02.3	0.560	0.547	2.3

Table 1: Results with deterministic exogenous traffic

In general, we find that simulation results show good correlation with the analytic results for both the average round trip delay $\overline{rtt}$ (with errors smaller than 7.5%) and the average throughput $\overline{thp}$ (with errors smaller than 11%). Furthermore, it turns out that in all cases the criterion (26) for deciding whether a single slow-start phase or two slow-start phases occur in a cycle also held in the simulations. In cases 1, 3, 6 and 9 we have $W_b < W_{\max}/2$, and two slow-start phases indeed occur; in cases 2, 4, 5, 7 and 8, we have $W_b > W_{\max}/2$ and there is indee only one slow-start phase per cycle (the graphs showing the evolutions of $W(t)$ in all 9 cases are available in [2]).

However, there are cases when the match between simulation and analysis is mediocre. We have found three such cases, which are illustrated in the table below. Case 1 corresponds to a case when the finite value of the bandwidth BW_{in} impacts the performance measures. Recall that BW_{in} is not considered in the analytic models. This suggests, and we have indeed observed this, that the match between simulation and analysis will degrade as the value of BW_{in} decreases below $\mu - \lambda$, i.e. as the source-bottleneck link becomes as much a bottleneck as the bottleneck queue.

	Parameters						$\overline{thp}$			$\overline{rtt}$		
case	B	BW_{in}	μ	λ	τ_1	τ_2	anal	simu	%err	anal	simu	%err
1	20	020	060	10	0.06	0.06	054.9	019.9	179.0	0.550	0.314	74.8
2	10	150	060	50	0.15	0.15	008.9	037.9	076.4	0.689	0.674	02.0
3	05	150	060	22	0.15	0.15	018.2	025.3	025.0	0.658	0.642	02.6

Table 2: Cases with mediocre match

Case 2 corresponds to a case when the value of λ is very large. In general, we have found that the error, especially for the average throughput, increases as λ increases. This is a result of our assumption that the bottleneck bandwidth is shared in proportion to the intensities of the flows of the streams sharing the bottleneck resources (refer for example to equation 15). This assumption would be adequate with a queue with the processor sharing discipline [25] or a RED-like scheme, but it is not always appropriate for the FIFO discipline.

Finally, case 3 corresponds to a case which we briefly hinted at earlier in Section 2, but did have not consider in our analytic model. In this case, we observe three slow start phases per cycle, as illustrated in Figure 6. We have not yet found a criterion for predicting when this type of behavior occurs. In practice, we have found that it occurs rarely, and that the error between simulation and analysis is moderate (for example it is equal to 25% for $\overline{thp}$ and 2.6% for $\overline{rtt}$).

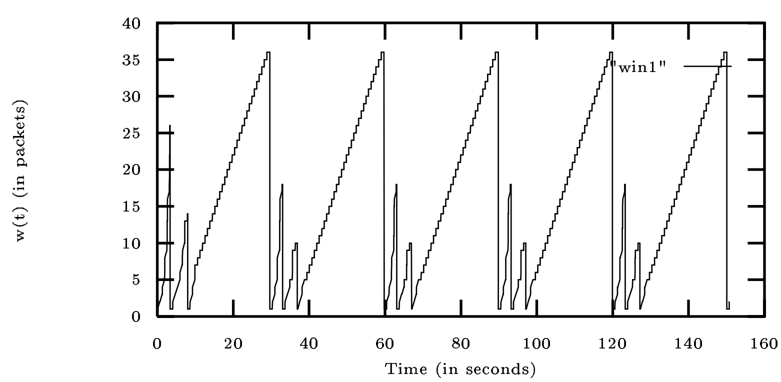

Figure 6: Evolutions of $W(t)$ in case 3 of Table 2

We now consider the case when the exogenous traffic is not periodic, but rather is a Poisson process. The main difference between the deterministic and the Poisson cases is of course that the periodic evolution of $W(t)$ is "broken" in the latter case by the stochastic nature of the exogenous arrival stream. This is illustrated in Figure 7 below (which corresponds to parameter values of case 5 in Table 3).

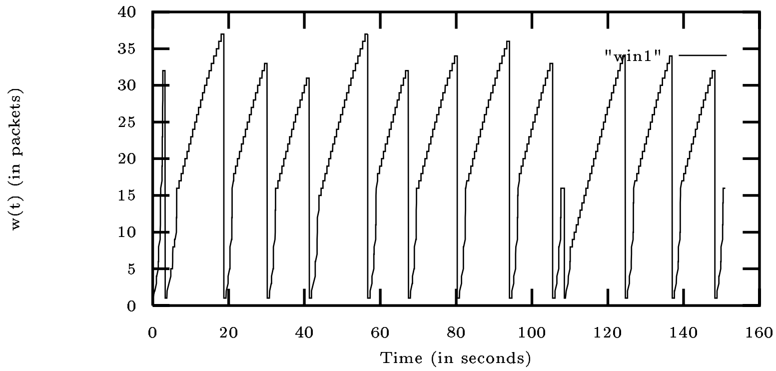

Figure 7: Evolutions of $W(t)$ in case 5 of Table 3

Table 3 shows the same results as Table 1 with the deterministic exogenous traffic now replaced by Poisson traffic. Note that the analytic results are identical in Tables 2 and 3 since the parameter λ which appears in the model is the same for both Poisson and deterministic process traffic streams. However, the simulation results are slightly different, but the fit with the analytic results remains good with the Poisson exogenous traffic.

	Parameters						$\overline{thp}$			$\overline{rtt}$		
case	B	BW_{in}	μ	λ	τ_1	τ_2	Ana.	Sim.	%err	Ana.	Sim.	%err
1	05	300	170	10	0.05	0.05	082.1	091.4	10.1	0.222	0.215	3.2
2	15	300	170	40	0.06	0.04	122.9	129.7	05.2	0.248	0.238	4.2
3	10	290	160	10	0.07	0.08	082.7	092.6	10.6	0.332	0.322	3.1
4	10	150	060	22	0.15	0.15	034.9	035.2	00.8	0.685	0.666	5.1
5	10	150	060	22	0.08	0.15	036.1	038.7	06.7	0.557	0.543	0.8
6	05	150	080	10	0.11	0.11	036.2	040.0	09.5	0.481	0.472	2.5
7	05	100	060	20	0.06	0.06	036.7	037.4	01.8	0.312	0.298	4.6
8	10	100	060	20	0.08	0.05	039.3	041.8	05.9	0.379	0.353	7.3
9	05	100	060	05	0.15	0.10	028.9	031.5	08.2	0.560	0.546	2.5

Table 3: Results with Poisson exogenous traffic

6 Conclusion

We have examined a model of a reference TCP connection sharing resources with exogenous connections in a wide area network, and derived analytic expressions for the mean throughput and delay of this reference TCP connection. This work has shown several interesting things.

First, the results of the fluid model tie in well with simulation. Second, our results with the shared bottleneck model are consistent with those obtained by others using different models, in particular models which represent the impact of exogenous traffic as that of an additional loss process. Third, the analysis has brought out interesting behavior, such as the "quasi priority" given to constant rate exogenous packets over TCP packets during congestion avoidance phases, and the application of this to audio and video connections in the Internet.

This work can be extended in several directions, and we are indeed considering some of them. Obvious extensions (at least conceptually) would be to consider more complete analytic models, and for example to explicitly consider the interactions between multiple TCP sources. Another extension is to use the analytic results in Sections 3 and 4 to infer λ from measured values of $\overline{thp}$ and $\overline{rtt}$, and more generally to estimate parameters of a network from end-to-end measurements obtained with TCP packets. Unlike constant rate probing schemes such as that in [4], a TCP-based scheme would not unduly overload the connections being examined. However, the trace measurements would be more difficult to interpret because of the interaction between TCP and exogenous packets. Our analysis would clearly help in untangling this interaction, and we are pursuing work in this area.

References

[1] E. Altman, F. Boccara, J. Bolot, P. Nain, P. Brown, D. Collange, and C. Fenzy, "Analysis of the TCP/IP Flow Control Mechanism in High-Speed Wide-Area Networks", *IEEE Conf. on Decision and Control (CDC' 95)*, pp. 368-373, New Orleans, LA, USA, Dec. 1995.

[2] E. Altman, D. Elouadghiri, M. Erramdani, J. Bolot, P. Nain, P. Brown and D. Collange, "Performance of TCP/IP over the French Research Network: Measurements and Analysis", France Telecom - CNET Internal Report No. RP/PAA/ATR/ORE/4678, 06921 Sophia-Antipolis Cedex, France, 1996.

[3] J-C. Bolot, A. U. Shankar, "Analysis of a fluid approximation to flow control dynamics", *Proc. IEEE Infocom '92*, pp. 2398-2407, Florence, Italy, May 1992.

[4] J.-C. Bolot, "End-to-end delay and loss behavior in the Internet", *Proc. ACM Sigcomm '93*, pp. 289-298, San Fransisco, CA, USA, Sep. 1993.

[5] J-C. Bolot, A.V. Garcia, "Control mechanisms for packet audio in the Internet", *Proc. IEEE Infocom '96*, San Fransisco, CA, pp. 232-239, Apr. 1996.

[6] L. S. Brakmo, S. O'Malley, L. Peterson, "TCP Vegas: New techniques for congestion detection and avoidance", *Proc. ACM Sigcomm '94*, pp. 24-35, London, UK, Sep. 1994.

[7] D. W. Browning, "Flow control in high-speed communication networks", *IEEE Trans. on Communications*, vol. 42, no. 7, pp. 2480-2489, Jul. 1994.

[8] CNET review, http://www.cnet.com/Content/Reviews/Compare/Webphone/

[9] K. W. Fendick, M. Rodrigues, A. Weiss, "Analysis of a rate-based control strategy with delayed feedback", *Proc. ACM Sigcomm '92*, pp. 136-147, Baltimore, MD, USA, Sep. 1992.

[10] S. Floyd, "Connections with multiple congested gateways in packet switched networks Part 1: One-way traffic", *CCR*, vol. 21, no. 5, pp. 30-47, Oct. 1991.

[11] V. Jacobson, "Congestion avoidance and control", *Proc. ACM Sigcomm '88*, Stanford, CA, USA, Aug. 1988.

[12] V. Jacobson, "Berkeley TCP evolution from 4.3-Tahoe to 4.3-Reno", presentation at the 18th IETF meeting, Aug. 1990.

[13] S. Keshav, "REAL: A network simulator", see also http://www.cs.cornell.edu/Info/People/skeshav/

[14] S. Klivansky, A. Mukherjee, C. Song, "On long-range dependence in NSFNET traffic", Georgia Tech report GIT-CC-94-61, Dec. 1994.

[15] T. V. Lakshman, U. Madhow, "Window-based congestion control for networks with bandwidth-delay products and random loss: a study of TCP/IP performance", *Proc. HPN '94 (High Performance Networking)*, pp. 133-147, Grenoble, France, Jun. 1994.

[16] T. V. Lakshman, A. Neidhart, T. Ott, "The Drop-from-front strategy in TCP over ATM and its interworking with other control features", *Proc. IEEE Infocom'96*, San Fransisco, CA, pp. 1242-1250, April 1996.

[17] T. V. Lakshman, U. Madhow, B. Sutter, "Window-based congestion control over asymmetric channels", technical report in preparation, 1997.

[18] J. Mahdavi, S. Floyd, "TCP-Friendly Unicast Rate-Based Flow Control", *Draft posted on end2end mailing list*, Jan. 8, 1997.

[19] J. Mogul, "Observing TCP dynamics in real networks", *Proc. ACM Sigcomm'92*, Baltimore, MD, pp. 305-317, Sep. 1992.

[20] T. J. Ott, J. H. B. Kemperman, M. Mathis, "The stationary behavior of ideal TCP congestion avoidance", Aug. 1996.

[21] V. Paxson, "Empirically derived analytic models of a wide-area TCP connection", *IEEE/ACM Trans. Networking*, vol. 2, no. 4, pp. 316-336, Aug. 1994.

[22] The Renater Web page is at URL http://www.urec.fr/Renater/.

[23] S. Shenker and L. Zhang, "Some observations on the dynamics of a congestion control algorithm", *Computer Communication Review*, pp. 30-39, Oct. 1990.

[24] W. Richard Stevens, *TCP/IP illustrated*, Addison Wesley, Vol. 1, 1994.

[25] A. Weiss, D. Mitra, "A transient analysis of a data network with a processor sharing switch", *AT&T Technical Journal*, pp. 162-176, Sept/Oct. 1988.

[26] N. Yin, M. Hluchyj, "On closed-loop rate control for ATM cell relay networks", *Proc. IEEE Infocom '94*, pp. 99-109, Toronto, Canada, Jun. 1994.

[27] Z. Zhang, S. Shenker, D. D. Clark, "Observations on the dynamics of a congestion control algorithm: the effects of two-way traffic", *Proc. ACM Sigcomm'91*, pp. 133-147.