A Comprehensive TCP Stochastic Model

Dong Zheng
Department of Electrical Engineering
Arizona State University
Dong.Zheng@asu.edu

Georgios Y. Lazarou
TITL
Mississippi State University
glaz@ece.msstate.edu

Rose Hu
TITL
Mississippi State University
hu@ece.msstate.edu

ABSTRACT
Understanding the nature of TCP behavior is critical in order to properly engineer, operate, and evaluate the performance of Internet, as well as to properly design and implement future networks. In this paper, we propose a complete and more accurate TCP performance prediction model in terms of packet loss rate, round-trip time, and maximum congestion window size. We show that under a wide range of packet loss rates, our model is up to 75% more accurate than the model proposed in [1].

KEY WORDS
TCP, Stochastic Model, Performance Evaluation

1 Introduction

As the widely deployed transport protocol in the Internet, TCP's performance is believed to greatly influence Internet traffic behavior. Thus a well-designed TCP is of utmost importance to the level of satisfaction of Internet users. Hence, many stochastic models of TCP have been proposed [1, 2, 3, 4, 5, 6, 7, 8] to study the TCP behavior.

To the best of our knowledge, none of the steady-state models proposed so far account for the slow start phase which begins at the end of every single time-out. The work in [1] makes the assumption that the slow-start phase happens less frequently than the congestion-avoidance phase and the throughput in slow-start is less than that in congestion-avoidance phase, hence the slow start phase can be ignored. While this may be true for small loss rates, the assumption doesn't hold for higher loss rates. Empirical measurements show that 90% of the packet losses lead to time-outs [1]. Since TCP enters the slow-start phase every time a time-out occurs, accurate TCP performance models must take into consideration of the aggregate effects of the slow-start phases.

In this paper, we propose a complete and more accurate steady-state TCP performance prediction model. We accomplish this by first significantly improving the model proposed in [1] and then extending it by incorporating our slow-start phase model developed in [9]. This major improvement is achieved by relaxing key assumptions and correcting critical approximations.

The rest of the paper is organized as follows. Section 2 presents the assumptions we made in constructing our model, and Section 3 derives the analytical model expressions. Section 4 analyzes the accuracy of our model and compares it with the one proposed in [1]. The paper concludes with Section 5.

2 Assumptions

Like in [1], we develop our model based on the TCP Reno release. We assume that the link speed is very high, the round-trip time (RTT) remains constants at all times, and the sender sends full-sized segments whenever the congestion window ($cwnd$) allows. The advertised window is assumed to be always constant and large. Thus, the congestion window evolution alone determines the send rate.

We model the dynamics of TCP in terms of "rounds" as done in [1]. A round starts when a window of packets is sent by the sender and ends when one or more acknowledgments are received for these packets. The delayed acknowledgment's effect is taken into consideration, but neither the Nagle algorithm nor the silly window syndrome avoidance is considered. In addition, we assume that the packet losses are in accordance with the bursty model. The packet losses in different rounds are independent, but they are correlated within rounds, that is, if one packet in a round is lost, then the subsequent back-to-back in the same round are also assumed to be lost. It is an idealization of the packet loss dynamics observed in the paths where FIFO drop-tail queues are used. Finally, we assume that the sender has unlimited data to send.

3 Model Building

Fig. 1 shows our extended model, which includes the slow-start phase. It shows a typical congestion window's evolution over time. As shown by Fig. 1, when a time-out occurs due to lost packets, TCP enters the slow-start to recover from the perceived network congestion.

Let TDP be the period between two triple-duplicate (TD) losses, Z_i^{ss} be the time spent in the slow-start phase, Z_i^{TD} be the duration of the congestion control phase, and Z_i^{TO} be the time interval of the time-out phase. Let M_i be the number of packets sent during the total time S_i. Then,

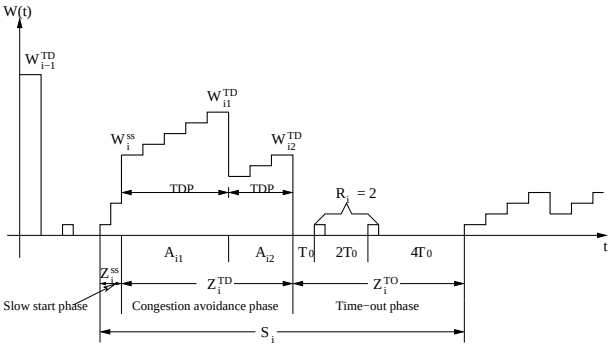

Figure 1. The extended steady state model - evolution of congestion window size when loss indications are triple-duplicate ACK's and time-outs.

we have that

$$M_i = Y_i^{ss} + \sum_{j=1}^{n_i} Y_{ij} + R_i, \quad (1)$$

$$S_i = Z_i^{ss} + Z_i^{TD} + Z_i^{TO} = Z_i^{ss} + \sum_{j=1}^{n_i} A_{ij} + Z_i^{TO}, \quad (2)$$

where Y_i^{ss} is the number of packets sent during the slow-start phase, A_{ij} is the duration of the jth TDP, n_i is the total number of the TDPs in the interval Z_i^{TD}, Y_{ij} is the number of packets sent during the jth TDP of interval Z_i^{TD}, and R_i is the number of packets sent during the time-out phase. W_i^{ss} is the window size at the end of a slow start and finally W_{ij}^{TD} is the window size at the end of the j^{th} TDP.

Assuming (S_i, M_i) to be a sequence of independent and identically distributed (i.i.d.) random variables, we determine the send rate as

$$B = \frac{E[M]}{E[S]}.$$

Considering n_i to be i.i.d. random variables and independent of Y_{ij} and A_{ij}, we have

$$B = \frac{E[Y^{ss}] + E[\sum_{j=1}^{n_i} Y_{ij}] + E[R]}{E[Z^{ss}] + E[\sum_{j=1}^{n_i} A_{ij}] + E[Z^{TO}]} = \frac{E[Y^{ss}] + E[n]E[Y] + E[R]}{E[Z^{ss}] + E[n]E[A] + E[Z^{TO}]}. \quad (3)$$

We next derive the closed form expressions for these expected values in the different TCP phases: the slow-start, the congestion-avoidance and the time-out phases.

3.1 The Slow-Start Phase

According to TCP Reno [10, 11], the current state of a TCP connection is determined based upon the values of $cwnd$ and the slow-start threshold ($ssthresh$). If $cwnd$ is less than $ssthresh$, TCP is in the slow-start phase, otherwise, it is in the congestion-avoidance phase.

From [9], we have that the expected number of rounds in the slow-start phase, $E[n^{ss}]$, can be obtained as follows:

$$E[n^{ss}] = log_g\left(\frac{E[Y^{ss}] + 2}{C}\right) - 2, \quad (4)$$

where $g = 1.618$ is the golden number, C is determined by the initial size of $cwnd$[1], and $E[Y^{ss}]$ is the expected number of packets sent in the slow-start phase.

Also, the expected congestion window size at the end of the slow-start phase, $E[W^{ss}]$, is given based on $E[Y^{ss}]$ [9]:

$$E[W^{ss}] = \frac{E[Y^{ss}] + 2}{g^2}. \quad (5)$$

If the slow-start phase is ended by a packet loss, the expected data that have been sent during this phase can be calculated as

$$E[Y^{ss}] = \frac{1-p}{p}, \quad (6)$$

where p is the loss rate.

Substituting the value of $E[Y^{ss}]$ in (5), we get:

$$E[W^{ss}]^* = \frac{1+p}{pg^2}. \quad (7)$$

This is the expected value of the congestion window when the slow-start phase ends due to a lost packet. Noting that when p is small, the expected value would be much bigger than the expected value of $ssthresh$, i.e.,

$$E[W^{ss}]^* \gg E[ssthresh] = \frac{E[W^{TD}]}{2}, \quad (8)$$

where the last equality comes from the fact that after each time-out, the slow-start threshold is set to half of the current congestion window W^{TD}.

Thus, it is safe to assume that TCP enters the congestion-avoidance phase before a packet gets lost. That is, we assume that TCP always switches from the slow-start to congestion-avoidance phase when the congestion window reaches the value of $ssthresh$. We present a proof of this in [12].

As a consequence, we have that the expected congestion window size at the end of the slow start be constrained by the limitation of the slow-start threshold:

$$E[W^{ss}] = E[ssthresh] = \frac{E[W^{TD}]}{2}. \quad (9)$$

Using (9) in (5) and rearranging, we obtain the expected number of packets sent during the slow-start phase:

$$E[Y^{ss}] = \frac{E[W^{TD}]g^2}{2} - 2. \quad (10)$$

The time spent in the slow-start phase is obtained by multiplying the number of rounds described in (4) with RTT:

$$E[Z^{ss}] = log_g\left(\frac{E[W^{TD}]}{2C}\right) \cdot RTT. \quad (11)$$

[1] If $cwnd = 1$ initially, C is 0.724

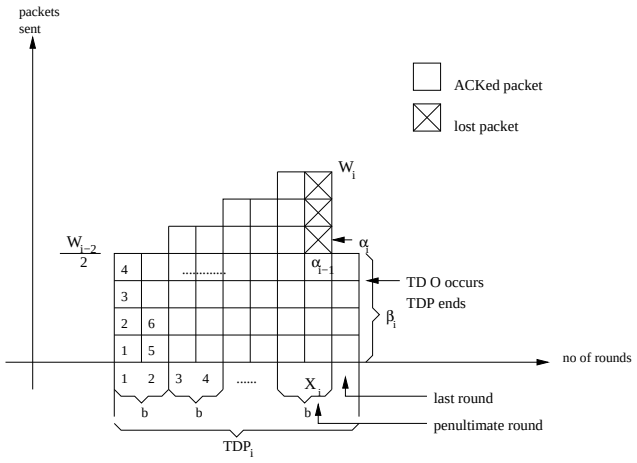

Figure 2. Packets sent during a TDP

3.2 The Congestion-Avoidance Phase

Let Y_i be the number of packets sent during the ith TDP, A_i be the duration, and W_i^{TD} be the window size at the end of the TDP. With reference to Fig. 2, we obtain the following relations [1][2]:

$$Y_i = \alpha_i + W_i^{TD} - 1, \quad (12)$$

$$A_i = \sum_{j=1}^{X_i+1} r_{ij}, \quad (13)$$

$$W_i^{TD} = \frac{W_{i-1}^{TD}}{2} + \frac{X_i}{b} - 1, \quad (14)$$

and

$$Y_i = \frac{X_i}{2}\left(\frac{W_{i-1}^{TD}}{2} + W_i^{TD} - 1\right) + \beta_i, \quad (15)$$

where X_i is the penultimate round in the TDP which experiences packet losses, r_{ij} is the round trip time, α_i is the number of packets sent in a TDP until the first loss happens, b is the number of packets acknowledged by a received ACK, and β_i is the number of packets sent in the fast retransmit phase, which is the last round [1].

Based on our assumptions, α_i is obviously geometrically distributed. Hence,

$$P[\alpha_i = k] = (1-p)^{k-1}p, \qquad k = 1, 2, \ldots \quad (16)$$

and therefore, we have that

$$E[Y] = \frac{1-p}{p} + E[W^{TD}]. \quad (17)$$

In addition, based on (14) and (15), we also have that

$$E[X] = b\left(\frac{E[W^{TD}]}{2} + 1\right) \quad (18)$$

$$E[Y] = \frac{E[X]}{2}\left(\frac{E[W^{TD}]}{2} + E[W^{TD}] - 1\right) + E[\beta], \quad (19)$$

[2]For details see [1]. Note that (14) is more accurate than the one presented in [1]

where we assume X_i and W_i^{TD} are mutually independent. Combining (17), (18), and (19), we get

$$\begin{aligned}\frac{1-p}{p} + E[W^{TD}] &= \frac{E[X]}{2}\left(\frac{E[W^{TD}]}{2} + E[W^{TD}] - 1\right) + E[\beta] \\ &= \frac{b(\frac{E[W^{TD}]}{2} + 1)}{2} \times \\ &\quad \left(\frac{E[W^{TD}]}{2} + E[W^{TD}] - 1\right) + E[\beta]. \quad (20)\end{aligned}$$

Since β_i is the number of packets sent when k packets in the penultimate round are ACKed, its value equals to k with probability

$$A(w,k) = \frac{(1-p)^k p}{1-(1-p)^w}. \quad (21)$$

Therefore,

$$\begin{aligned}E[\beta] &= E\Big[\sum_{k=0}^{w-1} k \cdot P(\beta = k)|w\Big] \\ &= E\Big[\sum_{k=0}^{w-1} \frac{k(1-p)^k p}{1-(1-p)^w}|w\Big] \\ &= E\left[\frac{(1-p)(1-pw(1-p)^{w-1} - (1-p)^w)}{p(1-(1-p)^w)}|w\right] \\ &\approx (E[W^{TD}] - 1)(1-p) \quad (22)\end{aligned}$$

for p small. Using (22) in (20) and rearranging, we get:

$$E[W^{TD}] = -\frac{2(b-2p)}{3} + \sqrt{\frac{4(bp + 2(1-p^2))}{3bp} + \left(\frac{2b-4p}{3b}\right)^2}. \quad (23)$$

Inserting (23) in (18), we obtain

$$E[X] = \frac{(2p+3)b - b^2}{3} + \sqrt{\frac{b^2p + 2b(1-p^2)}{3p} + (\frac{b-2p}{3})^2}, \quad (24)$$

and

$$\begin{aligned}E[A] &= (E[X]+1)E[r] \\ &= RTT\left(-\frac{b^2 - (2p+6)b}{3}\right. \\ &\quad \left. + \sqrt{\frac{b^2p + 2b(1-p^2)}{3p} + (\frac{b-2p}{3})^2}\right), \quad (25)\end{aligned}$$

where we assume r_{ij}'s to be i.i.d. and $E[r] \approx RTT$.

3.3 The Time-outs Phase

The probability that a loss indication is a time-out under the current congestion window size w, is given in [1] as

$$Q^{TD}(w) \approx \min\left(1, \frac{3}{w}\right).$$

Let Q^{TD} be the expected probability that a loss leads to a time-out at the end of the congestion-avoidance phase, then, we have that

$$Q^{TD} = E[Q^{TD}(w)] = \min\left(1, E[\frac{3}{W^{TD}}]\right). \quad (26)$$

In [1], Q^{TD} is approximated as

$$Q^{TD} = \min\left(1, \frac{3}{E[W^{TD}]}\right) \quad (27)$$

by using $E[1/W^{TD}] \approx 1/E[W^{TD}]$ approximation. However, as we show in [12]:

$$E[\frac{1}{W^{TD}}] \approx \frac{1}{E[W^{TD}]}(1 + \frac{\text{var}(W^{TD})}{E[W^{TD}]^2}) \approx \frac{\sqrt{3}}{E[W^{TD}]} \quad (28)$$

is a much better approximation. Therefore, we obtain the following enhanced approximation of Q^{TD}:

$$Q^{TD} \approx \min\left(1, \frac{3\sqrt{3}}{E[W^{TD}]}\right). \quad (29)$$

We then can compute the probability of the number of TDPs (n_i) by

$$p(n_i = k) = (1 - Q^{TD})^{(k-1)} \cdot Q^{TD}.$$

Therefore,

$$E[n] = \frac{1}{Q^{TD}}. \quad (30)$$

The expressions for the number of packets sent in the time-out phase, $E[R]$ and its duration, $E[Z^{TO}]$ are given in [1] as

$$E[R] = \frac{1}{1-p} \quad (31)$$

$$E[Z^{TO}] = T_0 \frac{f(p)}{1-p}, \quad (32)$$

where $f(p)$ is defined as

$$f(p) = 1 + p + 2p^2 + 4p^3 + 8p^4 + 15p^5 + 32p^6. \quad (33)$$

3.4 The Steady State Send Rate and Throughput

Substituting (10), (11), (17), (23), (25), (29), (30), (31) and (32) into (3), and taking into consideration the limitation of the window size[3], we finally derive the send rate as:

$$B = \begin{cases} \frac{\frac{E[W^{TD}]}{2} - 2 + \frac{1}{Q^{TD}(E[W^{TD}])}(\frac{1-p}{p} + E[W^{TD}]) + \frac{1}{1-p}}{\left(log_g(\frac{E[W^{TD}]}{2C_1}) + \frac{1}{Q^{TD}(E[W^{TD}])}(\frac{bE[W^{TD}]}{2} + b + 1)\right)RTT + \frac{f(p)T_0}{1-p}} & \text{when } E[W^{TD}] < W_m \\ \frac{\frac{W_m}{2} - 2 + \frac{1}{Q^{TD}(W_m)}(\frac{1-p}{p} + W_m) + \frac{1}{1-p}}{log_g(\frac{W_m}{2C_1})RTT + \frac{1}{Q^{TD}(W_m)}((\frac{b}{8}W_m + \frac{1-p}{pW_m} + 2) + 1)RTT + \frac{f(p)T_0}{1-p}} & \text{when } E[W^{TD}] \geq W_m. \end{cases} \quad (34)$$

This can be further simplified as

$$\min(\frac{W_m}{RTT}, \frac{1}{RTT\sqrt{\frac{2bp}{3}} + \min\left(1, 9\sqrt{\frac{bp}{8}}\right)p(\frac{RTT}{2}log_g(\frac{2}{3bpC_1^2}) + T_0(1 + 32p^2))}). \quad (35)$$

[3] We used the result from [1]

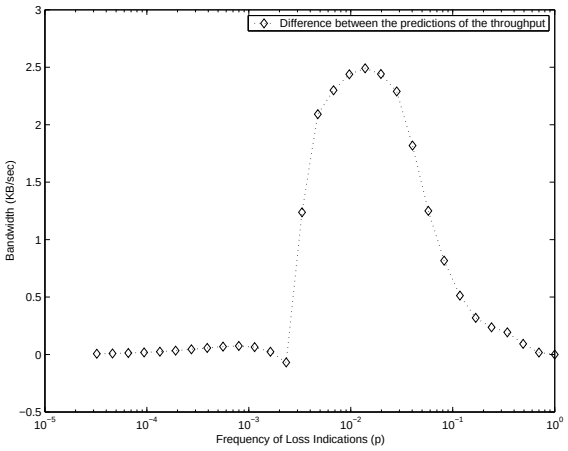

Figure 3. The difference of throughput predictions given by our proposed model and the model from Padhye's paper. The conditions are: $RTT = 200ms$, $MSS = 536bytes$, $w_1 = 1segment$, $T_0 = 1sec$, $W_m = 20segments$, $b = 2$.

To derive the throughput, we only need to change $E[Y]$, the expected size of packets that have been sent in a TDP, to $E[Y']$, the expected size of packets that have been received in a TDP. $E[Y']$ can be expressed as:

$$E[Y'] = E[\alpha] + E[\beta] - 1, \quad (36)$$

where $E[\alpha]$ is $1/p$ and $E[\beta]$ is given by (22). Also we substitute $E[R]$ with $E[R']$, the expected number of packets received in the time out phase, where [1]

$$E[R'] = 1.$$

Thus, the throughput can be formulated as:

$$H = \begin{cases} \frac{\frac{E[W^{TD}]g^2}{2} - 2 + \frac{1}{Q^{TD}(E[W^{TD}])}(\frac{1-p}{p} + (E[W^{TD}] - 1)(1-p)) + 1}{\left(log_g(\frac{E[W^{TD}]}{2C_1}) + \frac{1}{Q^{TD}(E[W^{TD}])}(\frac{bE[W^{TD}]}{2} + b + 1)\right)RTT + \frac{f(p)T_0}{1-p}} & \text{when } E[W^{TD}] < W_m \\ \frac{\frac{W_m g^2}{2} - 2 + \frac{1}{Q^{TD}(W_m)}(\frac{1-p}{p} + (W_m - 1)(1-p)) + 1}{log_g(\frac{W_m}{2C_1})RTT + \frac{1}{Q^{TD}(W_m)}((\frac{b}{8}W_m + \frac{1-p}{pW_m} + 1) + 1)RTT + \frac{f(p)T_0}{1-p}} & \text{when } E[W^{TD}] \geq W_m \end{cases} \quad (37)$$

which, when p is small, can be simplified as (35). This can be explained by noting that, if a loss seldom occurs, then the send rate should just equal to the throughput.

Fig. 3 compares our model against the one proposed in [1]. It shows the predicted throughput difference versus p for the case of $RTT = 200ms$, $MSS = 536bytes$, $w_1 = 1segment$, $T_0 = 1sec$, $W_m = 20segments$, and $b = 2$. From both models, when $p \to 0$, then $H \to W_m/RTT$. However, for $10^{-3} < p < 10^{-1}$, the model in [1] overestimates the throughput. Obviously, when $p \to 1$, again both models obtain the same values.

4 Model Validation through Simulation

In this section, we validate our proposed analytical models with simulation experiments. We performed all experi-

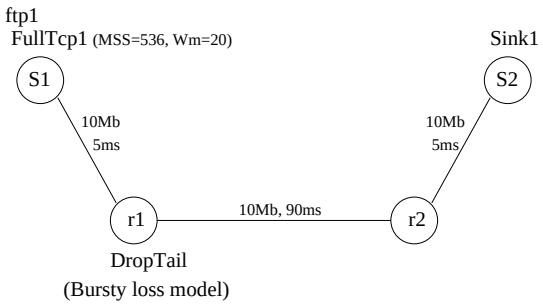

Figure 4. The simulation topology

ments in ns-2 [13] using the FullTCP agent. The FullTCP agent is modeled based on the 4.4BSD TCP implementation and can simulate all the important features of TCP Reno. The simulation topology is shown in Fig. 4.

We used FTP as the application for transmitting a 10MB file over a 10Mbps link. We set RTT to 200ms. For each value of p, we ran 1000 iterations.

Fig. 5 compares our model and the one derived in [1] against the simulation results for $0.005 < p < 0.1$. Clearly, the predicted values of the throughput at each p are much closer to the simulation values. To quantify the accuracy of our model, we computed the average error using the following expression taken from [1]:

$$\frac{\sum_{observations} |Th_{predicted}(p) - Th_{observed}(p)|/Th_{observed}(p)}{Number\ of\ observations},$$

where $Th_{predicted}$ is the throughput predicted by the models and $Th_{observed}$ is the throughput observed from the simulation. A smaller average error value indicates a better model accuracy. As shown in Fig. 6, our model is up to 75% more accurate than the model proposed in [1].

5 Conclusion

In this paper, we presented a more accurate stochastic model for predicting the throughput of TCP Reno. We first significantly improved the model developed in [1], and then extended it by incorporating our slow-start phase model we proposed in [9].

References

[1] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling TCP reno performance: A simple model and its empirical validation," *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp. 133–145, Apr. 2000.

[2] A. Kumar, "Comparative performance analysis of versions of TCP in a local network with a lossy link," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 485–498, 1998.

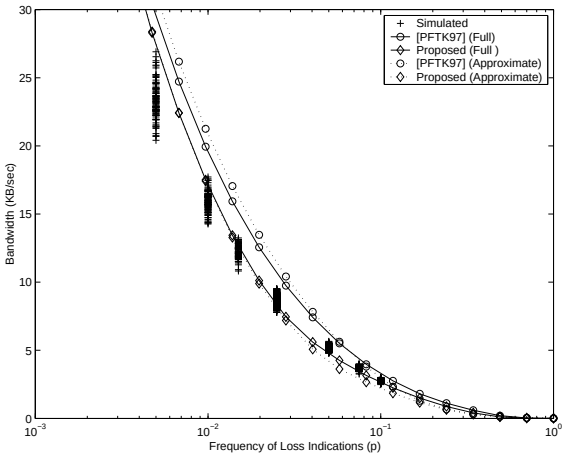

Figure 5. Comparing the steady state throughput predicted by the models in the middle-to-high loss rate range. The parameters are: $RTT = 200ms$, $MSS = 536bytes$, $w_1 = 1segment$, $T_0 = 1sec$, $W_m = 20segments$.

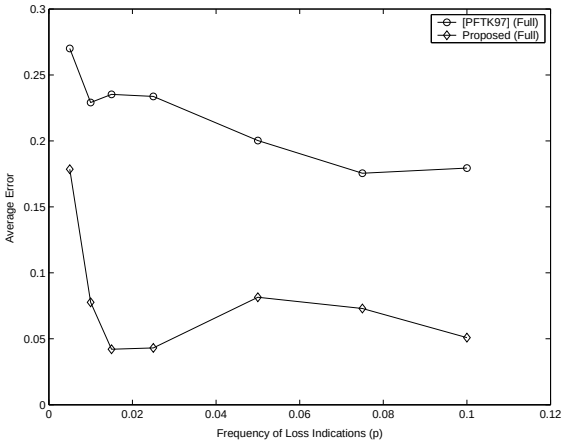

Figure 6. Average error comparison of the models in the middle-to-high loss rate range. The parameters are: $RTT = 200ms$, $MSS = 536bytes$, $w_1 = 1segment$, $T_0 = 1sec$, $W_m = 20segments$.

[3] T. Lakshman and U. Madhow, "The performance of TCP/IP for networks with high bandwidth-delay products and random loss," *IEEE/ACM Trans. Networking*, vol. 5, no. 3, pp. 336–350, July 1997.

[4] J. Heidemann, K. Obraczka, and J. Touch, "Modeling the performance of HTTP over several transport protocols," *IEEE/ACM Trans. Networking*, vol. 5, no. 5, pp. 616–630, Oct. 1997.

[5] M. Mathis, J. Semske, J. Mahdavi, and T. Ott, "The macroscopic behavior of the tcp congestion avoidance algorithm," *Computer Communication Review*, vol. 27, no. 3, July 1997.

[6] C. Casetti and M. Meo, "A new approach to model the stationary behavior of TCP connections," in *Proc. IEEE INFOCOM '2000*, Tel Aviv, Israel, Mar. 2000, pp. 367–375.

[7] B. Sikdar, S. Kalyanaraman, and K. S. Vastola, "An integrated model for the latency and steady-state throughput of TCP connections," *Performance Evaluation*, vol. 46, no. 2-3, pp. 139–154, 2001.

[8] N. Cardwell, S. Savage, and T. Anderson, "Modeling TCP latency," in *Proc. IEEE INFOCOM '2000*, vol. 3, Tel Aviv, Israel, Mar. 2000, pp. 1742–1751.

[9] D. Zheng, G. Y. Lazarou, and R. Hu, "A stochastic model for short-lived tcp flows," to be presented at the IEEE ICC '2003, Anchorage, Alaska, USA, May 2003.

[10] V. Jacobson, "Congestion avoidance and control," in *Proc. SIGCOMM '88*, Stanford, CA, 1988, pp. 314–329.

[11] W. R. Stevens, *TCP/IP illustrated*. Reading, MA: Addison-Wesley, 1994, vol. 1.

[12] D. Zheng, "On the modeling of TCP latency and throughput," Master's thesis, Mississippi State University, Miss. State, 2002. [Online]. Available: http://titl.ece.msstate.edu/publications.html

[13] UCB/LBNL/VINT. (2002) The network simulator ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns/