Submitted to SIGCOMM, January 1998.

A Simple Analytical Model for Wireless TCP Utilization

Stephen Jacobs, Huayan Wang, Alexandros Eleftheriadis and Mischa Schwartz

Department of Electrical Engineering
Columbia University, New York, NY 10027, USA

{sej,whycu,eleft,schwartz}@ee.columbia.edu

Abstract

TCP performance over wireless networks can be poor because TCP assumes losses occur in the network only when there is congestion. However, wireless networks often drop packets due to random errors on the wireless link. TCP performance can be severely degraded when packets with errors are mistaken for congestion. We present a very simple, tractable, and closed-form analytical model for determining the performance of TCP over hybrid wired/wireless networks. We compare our model with previous work in this area from an analytical perspective. We also compare our model with several simulations. The results show that our model matches both the previous work and the simulations very well. We also provide a simple heuristic for determining when wireless errors will severely degrade TCP performance.

1 Introduction

TCP is arguably the most important transport protocol today. This is primarily due to the huge growth in the use of the Internet, which uses TCP for most data applications, as well as variants of TCP for some real-time applications [7]. The expansion of the Internet has fueled the growth of all kinds of Internet access, including wireless Internet access. As wireless becomes a more common way of accessing the Internet, researchers have sought to provide the same level of service and performance that wireless users have come to expect from traditional wired networks.

TCP operates by continually probing the network for available bandwidth [11]. When TCP detects that the network has lost a packet, it reacts by reducing its transmission rate so as not to add to the congestion. The reasoning is that if a packet has been lost in the network due to congestion, an Internet-friendly protocol should react to this information by reducing its transmission rate. These techniques, called congestion detection and congestion avoidance, were originally designed for wired networks. This design decision violated protocol layer independence and has driven the need for network layer designers to be concerned with transport layer performance.

For transport protocols, the primary difference between wired and wireless networks is that wireless networks lose packets due to random errors on the wireless link. These errors could be caused by path loss, slow fading or fast fading. The result is that TCP cannot differentiate between a packet lost due to an error or due to congestion. Even the IP layer cannot determine the cause of a packet loss. TCP reduces its transmission rate by invoking its congestion avoidance algorithms in response to a packet loss that may not indicate congestion. In some cases, this leads to TCP under-utilizing the link.

In this paper, we develop a tractable model for TCP performance in the presence of wireless errors. The advantages of this model are that it is very easy to use and yet still maintains a high degree of accuracy and all of the heuristic content of current more complex models.

The next section provides an overview of previous work in analyzing TCP performance over wireless. Section 3 presents our analytical model for TCP performance in the presence of random wireless link errors. In Section 4 we compare the results of the model with both previous work and simulations, showing that our simple model yields quite favorable results. Finally, in Section 5 we present some concluding remarks.

2 Previous Work

There has been much research in the area of evaluating TCP performance over wireless networks. Much of the work has been aimed at techniques for improving performance [1, 2, 3, 4]. However, the analytical tools available to these researchers in evaluating the improvements are limited strictly to simulations. Analytical models of TCP performance over wireless that are easy to use should prove invaluable to the furtherance of this work.

Some models have been presented in the past. In [9, 10] the authors present an analytical model for measuring TCP performance on wired networks. Although the model is quite simple to use, it is only used for wired networks where loss is due to congestion alone.

In [8], Lakshman and Madhow present a more accurate model of TCP performance in wired networks for both TCP Tahoe and TCP Reno. The authors then extend this closed form solution to TCP performance in the presence of wireless errors. However, the wireless solution is not closed form and requires a numerical solution of an integral. In addition, many simplifying assumptions are made. Finally, in [6] the authors extend their previous work to incorporate a Markovian model of the channel.

In [5], the authors present a model of TCP dynamics in the presence of wireless errors by using a fluid flow approximation. Of course, the resultant equation for TCP utilization is quite complex and non-trivial to solve.

This paper differs from the previous work in that the end result is a simple closed form solution of TCP utilization in the presence of random errors. It unifies the simplicity of the work in [9] with a wireless model. We do make simplifying assumptions, but we clearly point out the limited regions over which these assumptions reduce the accuracy of the model.

3 Analytical Formulation

Our model consists of a TCP server located in the wired Internet and sending data across the Internet to a wireless client. The last link of the connection consists of a base station receiving data from the wired network and forwarding it to the client over the wireless link.

We assume that the bottleneck link over the entire TCP connection is the wireless network. In other words, all of the other links along the connection have sufficient bandwidth and buffering so that no packets are dropped due to congestion on those links. Packets are regularly dropped at the base station as TCP continually probes for available bandwidth in the network.

The analysis assumes that there is a given probability of error that the packet will be dropped on the wireless link due to random losses, P_r. We do not attempt to model the channel as a Markov chain switching between good and bad states with a correspondingly good and bad P_r. Our goal here is to evaluate the performance of TCP over wireless. Therefore, as long as the average time spent in a good state or bad state is large compared with the TCP round-trip time, we can assume that each time the channel changes state, TCP will reach a steady state before the channel changes state again. There will certainly be cases where this assumption is not valid; the user of this model could then pick a worst case P_r and use the model to generate a lower bound on performance.

3.1 Wired Network

In this section we first characterize the TCP stream as if it were operating in a wired network with loss only due to congestion [9, 10]. Then, we add the effects of loss due to errors on the wireless link to examine the performance degradation.

In a wired network, the bandwidth that a TCP connection uses is proportional to the size of the packets being sent, s, the round-trip time (RTT) and the frequency of lost packets due to congestion, P_c. The bandwidth used by TCP, given P_c, is:

$$\lambda_c = \frac{\alpha s}{\mathrm{RTT}\sqrt{P_c}}, \tag{1}$$

where α is dependent on the specific TCP implementation as well as the type of loss [10] and $P_c \ll 1$. A suggested value of $\alpha = 1.31$ is used. It should be noted that this equation is only valid for loss rates of less than 5% [9] and that it will produce inflated results beyond this range. We would like to determine the loss rate that TCP experiences due to congestion alone. To do this, we solve equation 1 for P_c.

$$P_c = \left(\frac{\alpha s}{\mathrm{RTT}\lambda_c}\right)^2 \tag{2}$$

In a wired network where there is very little random loss, TCP will fully utilize the bottleneck link if the buffering at the bottleneck node is larger than the bandwidth-delay product [8]. If the available bandwidth is 1 Mbps and the RTT is 50 milliseconds, then TCP will fully utilize the bandwidth if the buffering at the base station is more than 50 kilobits.

However, if the buffering is less than the bandwidth-delay product in a wired network, then the utilization must be calculated explicitly. In [8], the authors derive the TCP utilization for TCP Reno given the packet rate, μ, and buffering at the bottleneck node, b. In its most simplified form, the equation is as follows:

$$\lambda_c = \begin{cases} \frac{3\mu(b+t\mu)^2}{4(b^2+bt\mu+(t\mu)^2)}, & \text{if } b < \mu t \\ \mu, & \text{otherwise} \end{cases} \tag{3}$$

where $t = \tau + 1/\mu$ and τ is the propagation delay. The utilization, ρ_c, is then λ_c/μ. Although we focus on TCP Reno, results can also be derived for TCP Tahoe simply by replacing this equation with a formula for Tahoe utilization and continuing with the subsequent derivation. We continue the derivation assuming that $b < \mu t$ because this is the more complex case, as seen from equation 3.

Now that we know the throughput a TCP connection would receive in a wired network, we can substitute λ_c from equation 3 into equation 2 to calculate the loss rate experienced by this connection.

$$P_c = \left(\frac{4\alpha s(b^2 + bt\mu + (t\mu)^2)}{3t\mu(b+t\mu)^2}\right)^2, \tag{4}$$

where the RTT $= t$.

3.2 Wireless Network

P_c is the loss that the TCP connection experiences due to drops at the base station caused by congestion only, since the base station is the bottleneck node. Now we add the wireless random losses into the model. The losses occur at a rate P_r. The total loss seen by the TCP connection,

P_t, could be considered simply as the sum, $P_c + P_r$.[1] However, this might yield an incorrect value of P_t for two reasons.

Increasing P_r decreases P_c. For instance, a particular TCP connection has a loss rate due to congestion alone of P_c and there are virtually no errors due to the wireless link ($P_r = 0$). Suddenly, the wireless link becomes very noisy and P_r increases dramatically. Many more packets will be dropped due to these errors. TCP will detect these drops and reduce the rate at which it injects traffic into the network. This, in turn, will reduce the number of losses due to congestion (P_c) at the base station since the sender is transmitting at a lower rate and congesting the network less. Therefore, P_c is actually inversely related to P_r so that the sum of P_c and P_r is larger than the actual total loss rate seen by the TCP connection. The inflated value of P_t will create a lower utilization and will be particularly noticeable when P_r is on the same order of magnitude or larger than P_c.

P_r affects throughput more severely. Another problem with summing P_r and P_c is that random losses affect throughput more severely than congestion losses. The losses due to congestion happen periodically. TCP probes the network for available bandwidth by increasing its rate. In the beginning, TCP will be sending at a lower rate than the bottleneck rate so that the bottleneck buffer will be almost empty on average. Eventually, TCP will be sending at a rate which is faster than the bottleneck link and the buffer will begin to fill. After the buffer is completely full, a packet will be lost and TCP will start the cycle over again. In this way, the losses due to congestion happen periodically and only occur after the congestion window has reached the bandwidth delay product plus buffering.

However, random losses can occur at any time. If a random loss occurs when TCP is growing its window and increasing its transmission rate, then the transmission rate may not reach the bottleneck link rate during that cycle. This will severely impair the throughput.

[1] Of course, P_c and P_r must be small so that the sum will always be less than one.

Equation 1 assumes that losses are periodic. By using random errors in this equation, we are underestimating their effect on throughput. This will yield an artificially high throughput when P_r is large.

Nonetheless, we will use $P_t = P_c + P_r$. In Section 4 we will show that under most circumstances the effect of these inaccuracies is minor.

$$\lambda_t = \frac{\alpha s}{RTT\sqrt{P_t}} = \frac{\alpha s}{RTT\sqrt{P_c + P_r}} = \frac{\alpha s}{t\sqrt{\left(\frac{4\alpha s(b^2+bt\mu+(t\mu)^2)}{3t\mu(b+t\mu)^2}\right)^2 + P_r}} \quad (5)$$

The utilization is $\rho_t = \lambda_t/\mu$.

An important point to be observed from this equation is that it is the relationship between P_c and P_r which truly affects the throughput. If P_r is large, one might expect the performance to be degraded. This is not necessarily true because if P_c is even larger than P_r, the throughput will not be severely degraded. This is because P_c still dominates the square root in the denominator. Therefore, a simple heuristic in evaluating performance is simply to compare P_c with P_r. If P_r is larger than P_c then performance will be significantly degraded. This heuristic will be verified in the next section via simulations. We will also compare ρ_t with previous work in this area as well as simulations.

4 Results

4.1 Comparison with Previous Work

In this section, we compare equation 5 with the model presented by Lakshman and Madhow in [8]. We then compare both our model and Lakshman's against our simulations. Our goal in choosing simulation scenarios was to explore a range of values for the bandwidth-delay product, packet rate, buffering, and P_c.

Figure 1 shows the utilization as calculated by our model, Lakshman's model and obtained through simulations. The packet rate is 100 per second, the buffering is 80 packets and the round trip time is one second. The utilization in a purely wired network, ρ_c, would be 0.996 and $P_c = 0.00017$. The general shape of the curves are the same, though a significant deviation can be seen at $P_r = 10^{-4}$.

This deviation was explained in Section 3. As explained there, an increase in P_r will cause P_c to decrease. However, we do not take this effect into account in our model. For this reason, our model has an artificially high loss rate, resulting in an artificially lower utilization, when P_r (10^{-4}) is similar to or larger than P_c ($1.7 \cdot 10^{-4}$).

The other effect that our model ignores is the fact that wireless errors reduce utilization more than drops due to congestion. Because of this, our model will overestimate utilization at higher loss rates. These two inaccuracies serve to cancel each other out since the model simultaneously overestimates and underestimates the utilization at higher loss rates. At $P_r = 0.1$, there is also the effect that equation 1 overestimates utilization with loss rates higher than 5%.

In Figure 2 we have plotted the deviation of the model efficiencies from the simulated efficiencies,

$$\text{deviation} = \frac{\rho_{model} - \rho_{simulation}}{\rho_{simulation}} \tag{6}$$

It is clear from this figure that both models deviate from the simulations at higher loss rates and that the performance is quite comparable, even though the complexity of our model is much lower.

Figure 3 shows the same scenario except with less buffering, only 20 packets. The utilization in a purely wired network would be 0.871 and $P_c = 0.00023$. The same comments with respect to the deviation at $P_r = 10^{-4}$ can be made. In addition, the deviation of the models from the simulation is quite similar and is not shown here for that reason.

4.2 Comparison with Simulations

Figure 4 is a graph of our model of utilization and simulated utilization for a rate of 10 packets/sec, buffering of 10 packets and a round trip time of 1 second. This would yield a utilization of 1.0 on a purely wired network and a P_c of 0.01716. Again, the deviation from the simulation is worse at higher loss rates.

Figures 5 and 6 show the final two simulations. The first one is for a packet rate of 1000 per second and buffering of 1000 packets. The second has a packet rate of 10 per second and buffering of 100 packets. In this case, since the buffering is larger than the bandwidth delay product, equation 3 no longer holds. For this reason, we can assume that there is enough buffering for TCP to completely utilize the available bandwidth and we can use $\lambda_c = \mu$ from equation 3. Finally, Figure 7 shows the deviation from simulation for Figures 4 through 6.

Table 1 summarizes the simulations. The values for the last column (P_r) were chosen by identifying the point at which the slopes of the utilization curve (both simulated and modeled) were most steep. The table demonstrates the validity of the heuristic presented: the most significant degradation in utilization occurs when P_r and P_c are of the same order of magnitude. In every case, the most significant degradation in utilization occurs when P_r is of the same order as P_c. As mentioned earlier, this can be seen very clearly in our model since it is the sum of P_c and P_r which affects the utilization.

Packet Rate (packets/sec)	Buffering (packets)	Bandwidth Delay Product, μt	Wired Utilization	P_c	P_r with Steepest Slope of Utilization
100	80	100	0.996	$1.73 \cdot 10^{-4}$	10^{-4}
100	20	100	0.871	$2.26 \cdot 10^{-4}$	10^{-4}
10	10	10	1.000	$1.72 \cdot 10^{-2}$	10^{-2}
1000	1000	1000	1.000	$1.72 \cdot 10^{-6}$	10^{-6}
10	100	10	1.000	$1.72 \cdot 10^{-2}$	10^{-2}

Table 1: Summary of results. The last column indicates the random loss rate that decreases the utilization most dramatically. In all simulations the round trip time is 1 second.

5 Conclusions

We have developed a very simple model for analyzing the effect of random errors on TCP performance. Although our accuracy is marginally lower than that of the previous work in this field, we have the added benefit of an extremely easy-to-use closed form solution. This should be a benefit to researchers working in the field of improving TCP performance over wireless networks. In addition, we have presented a heuristic for the determination of when TCP performance is significantly impaired; this heuristic has been verified by simulation.

Our model does not account for all of the subtleties in TCP performance over wireless, but it is our contention that the slight degradation in accuracy is outweighed by the dramatic improvement in simplicity. The model is also interesting in that the path to obtaining it is much different than that of previous work in the field.

References

[1] B.S. Bakshi, P. Krishna, N.H. Vaidya, and D.K. Pradhan. Improving performance of TCP over wireless networks. In *Proc. of the 17th International Conference on Distributed Computing Systems*, pages 365–373, May 1997.

[2] H. Balakrishnan, V.N. Padmanabhan, S. Seshan, and R.H. Katz. A comparison of mechanisms for improving TCP performance over wireless links. In *Proc. ACM SIGCOMM Int. Conf. Communications Architecture, Protocols and Applications*, pages 256–269, Stanford, CA, October 1996.

[3] P. Bhagwat, P. Bhattacharya, A. Krishna, and S.K. Tripathi. Enhancing throughput over wireless LANs using channel state dependent packet scheduling. In *Proc. IEEE INFOCOM*, volume 3, pages 1133–1140, San Francisco, CA, March 1996.

[4] S. Biaz and N Vaidya. Using end-to-end statistics to distinguish congestion and corruption losses: A negative result. Technical report, Texas A and M University, Department of Computer Science, August 1997.

[5] A. Chan, D. Tsang, and S. Gupta. TCP (transmission control protocol) over wireless links. In *Proc. of IEEE 47th Vehicular Technology Conference*, pages 1326–1330, Phoenix, AZ, May 1997.

[6] H. Chaskar, T.V. Lakshman, and U. Madhow. On the design of interfaces for TCP/IP over wireless. In *Proc. of MILCOM '96 IEEE Military Communications Conference*, volume 1, pages 199–203, McLean, VA, October 1996.

[7] S. Jacobs and A. Eleftheriadis. Adaptive video applications for non-QoS networks. In *Proc. IFIP Fifth International Workshop on Quality of Service*, New York, NY, May 1997.

[8] T.V. Lakshman and U. Madhow. The performance of TCP/IP for networks with high bandwidth-delay products and random loss. *IEEE/ACM Trans. Networking*, 5(3):336–350, June 1997.

[9] J. Mahdavi and S. Floyd. TCP-friendly unicast rate-based flow control. Technical report, Internet Engineering Task Force, 1997.

[10] M. Mathis, J. Semke, J. Mahdavi, and T. Ott. The macroscopic behaviour of the TCP congestion avoidance algorithm. *Computer Communication Review*, 27(3):67–82, July 1997.

[11] J. Postel. Transmission control protocol. Internet RFC 793, Internet Engineering Task Force, September 1981.

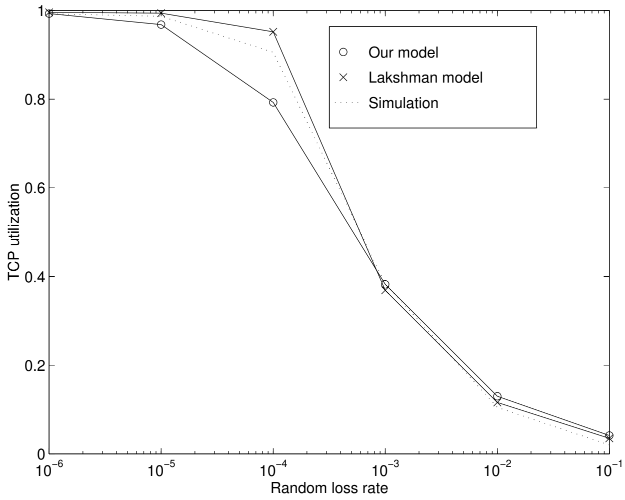

Figure 1: Utilization as calculated by our model, Lakshman's model, and simulation for 100 packets/sec, 80 packet buffering, a one second round trip time and increasing loss rate.

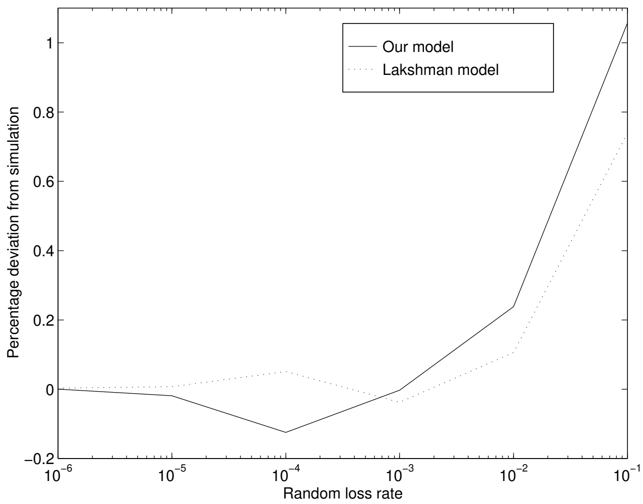

Figure 2: Deviation of utilization model from simulation for our model and Lakshman's model at 100 packets/sec, 20 packet buffering, a one second round trip time and increasing loss rate.

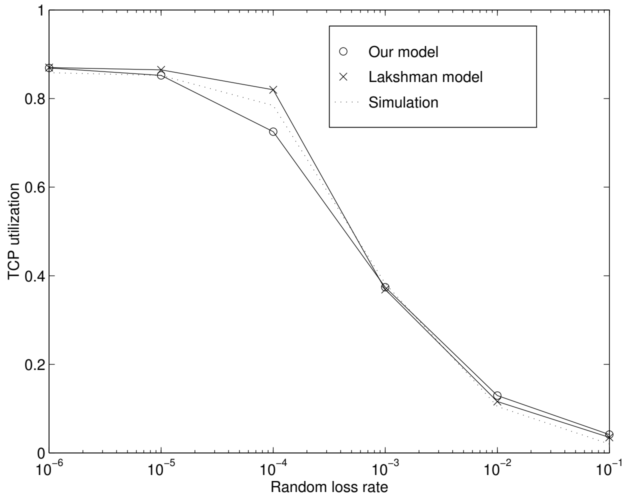

Figure 3: Utilization as calculated by our model, Lakshman's model, and simulation for 100 packets/sec, 20 packet buffering, a one second round trip time and increasing loss rate.

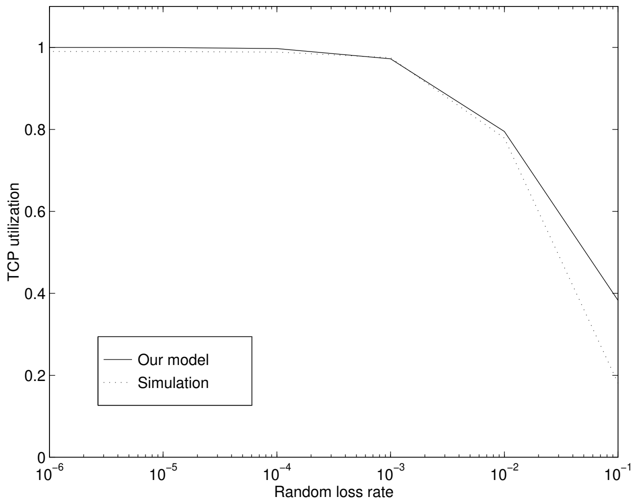

Figure 4: Utilization as calculated by our model, and simulation for 10 packets/sec, 10 packet buffering, a one second round trip time and increasing loss rate.

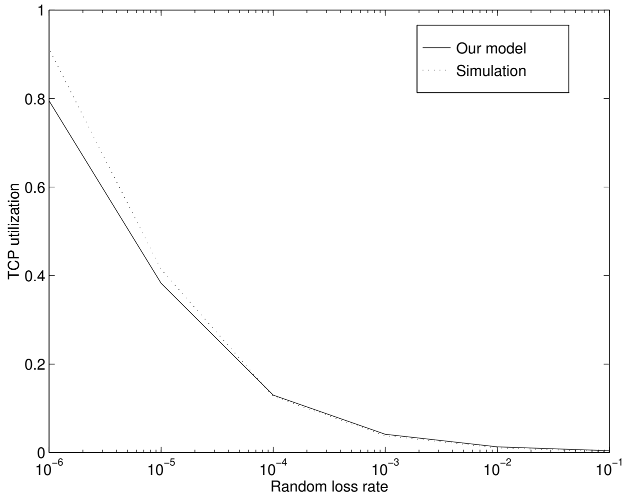

Figure 5: Utilization as calculated by our model, and simulation for 1000 packets/sec, 1000 packet buffering, a one second round trip time and increasing loss rate.

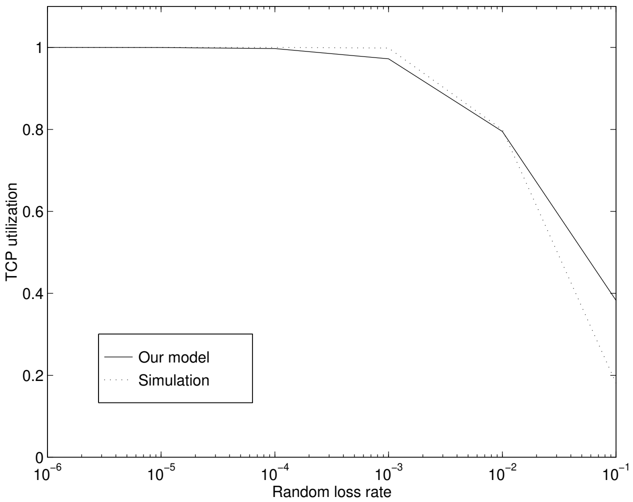

Figure 6: Utilization as calculated by our model, and simulation for 10 packets/sec, 100 packet buffering, a one second round trip time and increasing loss rate.

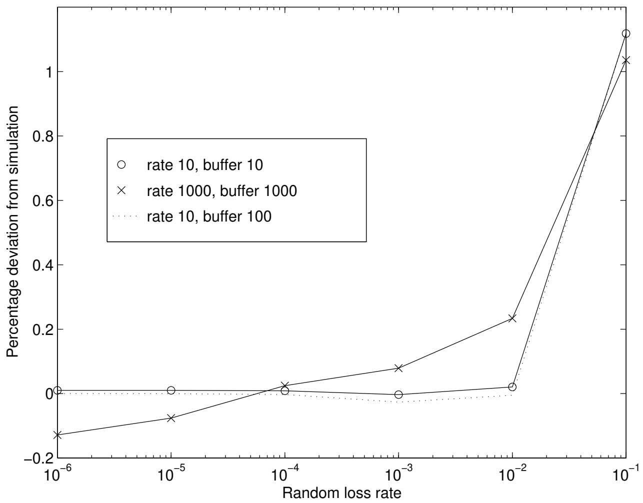

Figure 7: Deviation of utilization model from simulation for our model over varying rates and bufferings, a one second round trip time and increasing loss rate.