Comparative Performance Analysis of Versions of TCP in a Local Network with a Mobile Radio Link

Anurag Kumar* and Jack Holtzman†

Abstract

The scenario is that a bulk data transfer is being performed over a TCP connection, from a host on a local area network (LAN) to a mobile host attached to the LAN by a radio link. In earlier work [10] we had assumed that packet losses in a TCP connection over a radio link are statistically independent. In this paper, we extend this analysis to a Rayleigh fading link, which we model by a two state Markov model. The bulk throughputs of TCP-OldTahoe and TCP-Tahoe are compared with and without fading, for various average signal-to-noise ratios. We also study the performance with a link protocol on the wireless link, and study the effect of varying the link packet size, the number of link packet attempts, and the vehicle speed. For the parameters of the BSD UNIX implementation, over a 1.5Mbps wireless link, we find that, with fading, a signal-to-noise ratio of at least 30dB is required to get reasonable throughput with TCP Tahoe or OldTahoe; this corresponds to at least 100 times more power than is needed without fading.

For fixed signal-to-noise ratio, as the vehicle speed varies there are roughly 3 regions of performance: at very low speeds (pedestrian speeds) the throughput is very good; at low vehicular speeds

*Work done while on sabbatical at WINLAB, Rutgers University. Permanent affiliation: Dept. of Electrical Communication Engg., Indian Institute of Science, Bangalore, 560 012, INDIA.

†WINLAB, Rutgers University, Piscataway, N.J., 08855, USA.

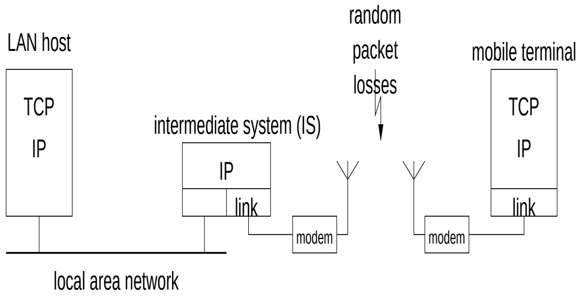

Figure 1: A LAN host with a TCP connection to a mobile host.

the throughput deteriorates, and again becomes very good at higher vehicle speeds. The speeds corresponding to the various regions depend on the parameters of the link protocol.

1 Introduction

The network scenario (Figure 1) is motivated by the many recent experimental studies of TCP performance over wireless mobile links [1, 2, 3]. We are interested in the data throughput, from the LAN host to the mobile terminal, that can be achieved by various versions of TCP, when the wireless link introduces random packet losses. In our earlier work on this problem [10] (see also [12], [14]), a simple loss model was assumed: TCP packets were transmitted in their entirety over the wireless channel, and each packet was lost independently of anything else with probability p (i.e., the Bernoulli loss model). The Bernoulli loss model would correspond to a nonfading channel with additive white Gaussian receiver noise. It is well known ([9, 15]) that when the mobile and the transmitter are not in line-of-sight, the mobile receiver antenna observes a superposition of multiple reflected and diffracted signals that are out of phase with each other. The phase relationship between the interfering multipath signals is continuously changing as the vehicle moves. Destructive interference of these signals leads to "fading". The fade durations depend on the velocity of the vehicle and the radio carrier frequency. For high speed radio transmission (e.g., 1.5Mbps), typical carrier frequencies (e.g., 900Mhz), and the usual vehicle speeds, the fade durations are comparable to the transmission times of TCP packets, or are multiples of transmission times of typical link packets. Thus the packet losses cannot be modelled as being independent of each other.

In this paper, we model the losses using a two state Markov model. We assume that a packet succeeds with probability 1 in the Good state, and with probability 0 in the Bad state. The TCP protocol alternates between packet transmission periods and loss recovery periods. It was shown in [10] that the TCP packet transmission process can be modelled as a Markov Renewal-Reward process [20], the Markov renewal instants being the epochs at which the first packet loss occurs in a transmission period, and the "reward" corresponding to successful packet transmissions. With the simple fading model, this Markov renewal-reward analysis is easily adapted. We obtain numerical results for the same parameters as in [10].

In previous work, Gilbert has used a two state Markov model to model bursty error rates in digital transmission links [7]. Wang and Moayeri [19] have studied multistate Markov chain models for radio channels, and have, in particular, developed a finite state Markov model for a Rayleigh fading channel. Recently, in [4], the authors analyse a wide area TCP connection in which the receiver end system is connected to the wireline network by a wireless mobile link. A Rayleigh fading model is assumed for the wireless link, and a two state Markov model is used. It is assumed that the link protocol on the wireless channel repeatedly retransmits the link packets until they succeed. Since this implies an increase in the effective transmission time of TCP packets, the main concern in the paper is the additional buffer requirement (at the wireline-to-wireless router) as a result of wireless channel losses. It is argued that for a Markovian fading model (implying exponentially distributed fade durations) TCP will yield reasonable throughputs if the buffer space grows only as fast as the logarithm of the bandwidth-delay product.

In our analysis, we assume that there are adequate buffers at the wireless router so that buffer overflow is not a concern. On the other hand, we permit the possibility of packet loss on the wireless link, either because there is no link protocol, or because the link protocol limits the number of retries. The effect of losses on the TCP throughput is studied. We provide results for throughput performance of TCP-OldTahoe and Tahoe as the signal-to-noise ratio varies, and compare the situations with and without Rayleigh fading. For fixed signal-to-noise ratio, we then study the variation of TCP throughput with the vehicle speed. Varying the vehicle speed effectively varies the duration of the fades and the good periods on the radio link, and results in interesting behaviour of the TCP throughput as a function of the vehicle speed.

This paper is organised as follows. In Section 2 we develop the two state Markov loss model. In Section 3 we describe TCP's window adaptation protocol. In Section 4 we adapt the throughput analysis developed in [10] to the Markov loss model. In Section 5 some numerical results and their discussion are presented.

2 A Markov Model for Rayleigh Fading

2.1 Review of the Rayleigh Fading Model

The radio carrier is digitally modulated; for Pulse Amplitude Modulation (PAM), the transmitted signal is written as

$$x(t) = \sqrt{2}\ \mathrm{Re}\left\{ \sum_{k=-\infty}^{\infty} s_k e^{j\theta_k} p(t-kT) e^{j2\pi f_c t} \right\}$$

where $p(\cdot)$ is the baseband pulse, f_c is the carrier frequency, and (s_k, θ_k) is the complex modulating sequence (see, for example, [13]). Analysis of the superposition of multipath signals, in the presence of receiver mobility, yields the following model for the received signal ([16]):

$$\begin{aligned} y(t) = \sqrt{2}\ \mathrm{Re} \Bigg\{ \sum_{k=-\infty}^{\infty} & s_k r(t) e^{j(\theta_k+\phi(t))} \cdot \\ & p(t-kT) e^{j2\pi f_c t} \Bigg\} \end{aligned} \tag{1}$$

where $r(t)$ is the random attenuation and $\phi(t)$ is the random phase noise process. If the slower fading phenomena (power law attenuation, and log-normal fading [16]) are compensated for by power control, and the multipath phenomenon is spatially homogenous, then the process $r(t)$ is stationary with a Rayleigh marginal distribution R, with density

$$p_R(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right)$$

We note that $E(R^2) = 2\sigma^2$. The time variations in the process $r(t)$ are of the order of the Doppler frequency f_d, which is related to the carrier frequency f_c, and the vehicle speed v, by the formula

$$f_d = \frac{v f_c}{c} \tag{2}$$

where c is the speed of light. For a 900Mhz carrier frequency, for example, the above formula yields a Doppler frequency of 3Hz/m/sec, or 10.8Hz/km/hr. Thus, for the signalling rates used in high speed wireless transmission (e.g., Mbps), Rayleigh fading can be taken to be roughly constant over several bits (see also [16, pages 165–166]).

From Equation 1, the predetection signal-to-noise ratio (SNR), say $\psi(t)$, at the receiver is given by

$$\psi(t) \;=\; (r(t))^2 \left(\frac{E_b}{N_0}\right)_{xmit} \tag{3}$$

where $\left(\frac{E_b}{N_0}\right)_{xmit}$ is the "transmitted" SNR. Denote the marginal random variable for the stationary process $\psi(t)$ by Ψ; then we have

$$\Psi \;=\; R^2 \left(\frac{E_b}{N_0}\right)_{xmit} \tag{4}$$

where R is the marginal of the process $r(t)$, as defined above. The *average* effective received SNR is then given by

$$\begin{aligned} E(\Psi) &= 2\sigma^2 \left(\frac{E_b}{N_0}\right)_{xmit} \\ &=: \left(\frac{E_b}{N_0}\right) \end{aligned}$$

In dB units we can now write Equation 4 as

$$(\Psi)_{dB} \;=\; \left(\frac{E_b}{N_0}\right)_{dB} + (A)_{dB} \tag{5}$$

where $A := \frac{R^2}{2\sigma^2}$, and the distribution of A is known to be given by $P(A > a) = e^{(-a)}$ ([16]). As observed above, the SNR $\psi(t)$ varies slowly as compared to the signalling rate. When the SNR is low (i.e., a large negative values of $(A)_{dB}$), this situation persists for a while and the bit error rate (BER) is high; then the SNR improves and stays high for a while, yielding a low BER. This view motivates a Markov model for packet loss; we develop the model in the next subsection.

2.2 The Markov Packet Loss Model

For a given average SNR $\left(\frac{E_b}{N_0}\right)$ (say 30dB) we determine the amount by which the SNR must drop below the average so that the channel enters the "Bad" state (say 20dB, i.e., a factor of .01); call this "margin" α, in dB units. Then, defining $\delta := 10^{-\frac{\alpha}{10}}$ (i.e., $\delta = .01$ for $\alpha = 20$dB), the fraction of time that the channel is in the Bad state is given by $P(A \leq \delta) = 1 - e^{-\delta}$. Thus, fixing α

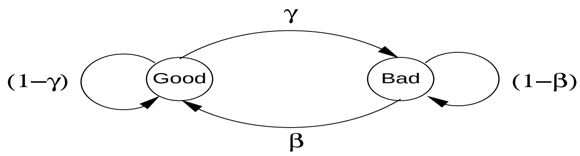

Figure 2: Transition structure of the Markov loss model.

fixes the fraction of time that the channel is in the Bad state. To obtain the mean durations in each state, we use results from level crossing analysis of the process $r(t)$ (see, [15]). Defining $G(\delta)$, and $B(\delta)$, as the mean durations in the Good and Bad states, respectively, the following formulas are obtained

$$G(\delta) = \frac{f_d^{-1}}{\sqrt{2\pi\delta}} \tag{6}$$

$$B(\delta) = \frac{f_d^{-1}(e^{\delta}-1)}{\sqrt{2\pi\delta}} \tag{7}$$

where f_d is given by Equation 2. For small δ, $e^{\delta}-1 \approx \delta$, hence Equations 6 and 7 yield

$$B(\delta) \approx \delta G(\delta). \tag{8}$$

The loss model is a discrete "time" Markov chain whose transitions are embedded at packet boundaries; thus we express the Good and Bad periods in units of packet transmission time on the link. The transition structure of the model is depicted in Figure 2. The chain is assumed to be embedded at the *beginnings* of packet transmissions. Thus, if a number of packets is being transmitted back to back, and if the chain is in the Good state when a packet is about to be transmitted then this packet will be Good, and the next packet will be Good or Bad with probabilities $1-\gamma$ and γ, respectively. It follows, from Equation 8, that

$$\frac{\gamma}{\beta} \approx \delta$$

In our calculations we will use the above approximation, as $\delta \leq 0.1$ (i.e., $\alpha > 10$dB) in our numerical results. Observe that we are making the tacit assumption that the durations during which the SNR is above or below the threshold level are exponentially distributed; this is not true (see [17]), but the minimal Markov model that we have considered cannot model any additional characteristic of the fading process.

Given α, equivalently δ, the individual values of γ and δ in the Markov model are obtained from the Doppler frequency f_d (Equation 2), and Equations 6 and 7. For example, consider packet transmission times of 7.5msec (e.g., 1.5Mbps link, and 1400byte packets), $f_c = 900$MHz, and $\alpha = 20$dB. For a speed of 5kmph, $f_d = 4.17$Hz, the mean good period is about 128 packets, the mean bad period is about 1.28 packets, and $\gamma = 0.0078$; at a speed of 100kmph, $f_d = 83.3$Hz, the mean good period is about 6.4 packets, the mean bad period is about 0.064 packets, and $\gamma = 0.156$. Thus, clearly, some care is needed in using this model for high vehicle speeds at which the fade duration is smaller than one packet length. Without a link protocol, whenever there is a fade during a packet transmission, that packet is lost, even though the channel is good during some parts of the packet. For high speeds, for which the fade durations are smaller than a packet (but not so high that $G + B <$ mean pkt xmission time), we take $\beta = 1$ (i.e., exactly one packet is lost in each fade), and γ is obtained from the following equation

$$\gamma = \left(\frac{G + B}{\text{mean pkt xmission time}} - 1 \right)^{-1}$$

Thus, at low speeds, the probability that a packet is bad is just $\frac{\delta}{1+\delta}$, whereas for high speeds, when the fades are shorter than the packet transmission time, the probability that a packet is bad *increases* with increasing speed, even though δ is fixed.

2.3 The Loss Model with a Link Protocol

The round trip propagation delay on terrestrial mobile radio links is typically smaller than the packet transmission time. Consequently, a stop-and-wait link protocol suffices. We characterise a link protocol by the link packet length and the number of times it attempts a packet.

Observe that with 128byte link packets, and 1.5Mbps transmission, the link packet transmission time is about 0.68msec. For 1400byte TCP packets, there are 11 link packets in a TCP packet, and for speeds above 30 or 40kmph the mean fade duration is less than 2 or 3 link packets. Thus, with a link protocol, we embed the Markov packet loss model (Figure 2) at the beginnings of link packet transmissions. We use the loss model to obtain the probability of TCP packet loss, and assume that TCP packet losses are independent at the TCP packet level. This assumption is adequate if the link packets are small, and the number of link packets in a TCP packet is large. We assume that the first link packet in a TCP packet finds the Markov model in its stationary distribution. The probability of TCP packet loss is then the probability that for at least one link packet all attempts fail. The link protocol also causes an increase in TCP packet transmission time; we use the link level Markov loss model to obtain the inflated mean TCP packet transmission time.

Let N denote the maximum number of attempts of a link packet; and L denote the number of link packets in a TCP packet. Recalling that the Markov process is embedded at the beginnings of link packet transmissions, we define

$p_n =$ Prob{at least 1 out of n link packets fails, given that the initial state is Good}

$q_n^{(k)} =$ Prob{at least 1 out of n link packets fails, given that the first link packet has already had $k(< N)$ bad attempts and the initial state is Bad}

$p =$ Prob{a TCP packet is bad }

Assuming that the first link packet in a TCP packet finds the Markov process in its stationary distribution, we have

$$p = \frac{\beta}{\gamma+\beta} p_L + \frac{\gamma}{\gamma+\beta} q_L^0$$

The values of $p_n, 1 \leq n \leq L$ and $q_n^{(k)}, 1 \leq n \leq L, 0 \leq k \leq N-1$ are easily obtained from the following equations by recursive substitution, with the boundary conditions: $p_1 = 0$, $q_n^{(N-1)} = 1, 1 \leq n \leq N$, and $q_1^{(k)} = (1-\beta)^{((N-1)-k)}, 0 \leq k \leq (N-1)$. For $n > 1$ and $0 \leq k < N-1$

$$\begin{aligned} p_n &= (1-\gamma)p_{n-1} + \gamma q_{n-1}^{(0)} \\ q_n^{(k)} &= \beta p_n + (1-\beta) q_n^{(k+1)} \end{aligned}$$

For obtaining the mean effective TCP packet transmission time, taking into account the link retransmissions, define

$l_n =$ mean number of link packets that will be sent for transmitting a TCP packet of length n link packets, given that the state at the beginning is Good

$m_n^{(k)} =$ mean number of link packets that will be sent for transmitting a TCP packet of length n link packets, given that k link packets have already been sent for the first link packet and the current state is Bad

$l =$ mean number of link packets that will be sent for transmitting a TCP packet of length L link packets

As for the loss probability, we have

$$l = \frac{\beta}{\gamma+\beta} l_L + \frac{\gamma}{\gamma+\beta} m_L^0$$

With the boundary conditions $l_1 = 1$ and $m_1^{(N-1)} = 1$, we have the following equations, for $n \geq 1$ and $0 \leq k < N-1$,

$$\begin{aligned} m_n^{(k)} &= 1 + (1-\beta) m_n^{(k+1)} + \beta l_n \\ l_n &= 1 + (1-\gamma) l_{n-1} + \gamma m_{n-1}^{(0)} \\ m_n^{(N-1)} &= 1 + (1-\beta) m_{n-1}^{(0)} + \beta l_{n-1} \end{aligned}$$

These equations can also be solved recursively.

3 TCP Window Adaptation

At any time t there is a *lower window edge* $A(t)$, which means that all data numbered up to, and including, $A(t) - 1$ has been transmitted and acknowledged. The transmitter can send data numbered $A(t)$ onwards. There is also the transmitter's *congestion window* $W(t)$, which, at time t, is the maximum amount of data that the transmitter is permitted to send, starting from $A(t)$. Under normal data transfer, $A(t)$ has nondecreasing sample paths, whereas the adaptive window mechanism causes $W(t)$ to increase or decrease, but never exceed W_{max}. The receipt of an ack packet that acknowledges some data will cause a jump in $A(t)$ equal to the amount of data acknowledged, but the change in $W(t)$ would depend on the particular version of TCP and the state of the congestion control process.

At the transmitter, round-trip time measurements of some packets are used to obtain a running estimate of the packet round-trip time (rtt) on the connection. Each time a new packet is transmitted, the transmitter starts a timer and *resets* the already running transmission timer, if any. The timer is set for a round-trip time-out (rto) value that is derived from the rtt estimation procedure. Whenever a time-out occurs, retransmission is initiated from the next packet after the last acknowledged packet (i.e., from the sequence number $A(t)$). It is important to note that the TCP transmitter process measures time and sets timeouts only in multiples of a *timer granularity*; for example, BSD UNIX based systems have a timer granularity of 500ms. Further, there is a minimum timeout duration of 2 or 3 timer "ticks" in most implementations. We will see, in the analysis, that coarse timers have a significant impact on TCP performance. For details on rtt estimation, and the setting of rto values, see [5] or [18].

The following basic window adaptation procedure [8] is common to all the TCP versions; our description follows that of [12]. At all times t the following are defined for all the protocol versions:

$W(t) =$ the transmitter's congestion window at time t

$W_{th}(t) =$ the slow-start threshold at time t

The evolution of these processes is triggered by acks (only first acks; not duplicate acks) and timeouts as follows.

1. If $W(t) < W_{th}(t)$, each ack from the receiver causes $W(t)$ to be incremented by 1. This is called the *slow start* phase.

2. If $W(t) \geq W_{th}(t)$, each ack from the receiver causes $W(t)$ to be incremented by $\frac{1}{W(t)}$. This is called the *congestion avoidance* phase.

3. *If timeout occurs* at the transmitter, at epoch t, $W(t^+)$ is set to 1, $W_{th}(t^+)$ is set to $\left\lceil \frac{W(t)}{2} \right\rceil$, and the transmitter begins retransmission from the packet after the last packet acknowledged.

Note that the transmissions after a timeout always start with the first lost packet. We will call the "window" of packets transmitted from the lost packet onwards, but before retransmission starts as the *loss window*.

If a packet is lost, then the ack number from the receiver will cease to be advanced, and *the transmitter at the LAN host will continue sending packets until the current window is exhausted, or a packet gets through to the receiver and an ack is received.*. The last packet transmitted will have a timeout associated with it; in TCP-OldTahoe, retransmission will start only upon the expiry of this timer. Later versions of TCP, i.e., Tahoe, Reno, and NewReno, implement a *fast-retransmit* scheme. Observe that, even if a packet is lost, if subsequent packets get through, the receiver will continue to send back acks, but the "expected packet" number is not incremented. If several (an integer parameter K, e.g., 3) such "duplicate" acks are received at the LAN host, then it can assume that the loss is a random loss, and not due to congestion. When the number of duplicate acks received reaches the threshold K, then the packet next expected by the receiver is retransmitted. A complicated recovery phase then follows. In particular, in TCP Tahoe [6] if the transmitter receives the Kth duplicate ack at time t, before the timer expires, then the transmitter behaves as if a timeout has occurred and begins retransmission, with $W(t^+)$ and $W_{th}(t^+)$ as given by the basic algorithm.

4 TCP Throughput Analysis with the Markov Loss Model

Assume that, at $t = 0$ the connection starts with $W(0) = 1$ and $W_{th}(0) = \left\lceil \frac{W_{max}}{2} \right\rceil$, where W_{max} is the maximum window limit set by the receiver; this usually depends on the receiver's buffer size (typical values are 32Kbytes or 64Kbytes, i.e., several 10's of TCP packets). The protocol starts at time 0 in slow start. Let ℓ_1 be the epoch at which the first packet loss occurs, and let $U_1 = W(\ell_1)$. As described above, timeout or fast recovery follows, and at some later epoch transmission resumes with the first lost packet. If this epoch finds the wireless channel in its Bad state, then a loss occurs immediately, the timeout is doubled, and the next cycle starts with $W = 1$ and $W_{th} = 2$, the minimum value of W_{th} ([18]). No successful packet is sent in such a period. Hence, we merge such periods, if any, into the recovery period of the previous cycle. Thus at the beginning of the next cycle, denoted by t_1, the channel is in the Good state. Again an epoch ℓ_2 can be identified at which the first loss occurs in this transmission cycle. For $k \geq 1$, let t_k denote the kth epoch at which a new transmission cycle starts as described above. For $k \geq 1$, we call the interval $(t_{k-1}, t_k]$ the kth cycle. In the kth cycle, let ℓ_k be the epoch at which the first packet is lost in the cycle (this is an end of a packet transmission epoch from the router). Further, for $k \geq 1$, let $U_k = W(\ell_k)$ denote the transmitter's window size at which packet loss takes place. We take $U_0 = W_{max}$. Since the first packet transmitted in each cycle is always good, the state space of the process $\{U_k\}$ is $\{2, 3, \ldots, W_{max}\}$.

If the first retransmission after a recovery period finds the channel in the good state, then for TCP-OldTahoe and Tahoe, the value of U_k determines the values of $W(t_k^+)$ and $W_{th}(t_k^+)$, according to $W(t_k^+) = 1$ and $W_{th}(t_k^+) = \left\lceil \frac{U_k}{2}, \right\rceil$ We know that at the beginning of the transmission of the packet that is lost, the channel is in the Bad state. It is thus clear that the evolution of the congestion window process after the first loss epoch in the kth cycle depends only on the value of U_k; thus the process $\{U_k\}$ is a discrete time Markov chain over the state space $\{2, \ldots, W_{max}\}$. We can obtain the stationary distribution of this chain. Note that a more complex analysis ensues if losses can occur in either state of the channel.

The mean number of packets successfully transmitted in each cycle, before the first lost packet, is just $\frac{1}{\gamma}$. When a loss occurs at the lossy link, there would be some packets queued in the lossy link transmitter, and the TCP transmitter at the LAN host will continue sending packets till the congestion window (U_k) is exhausted (even if fast-retransmit is implemented, by the time 3 duplicate acks are received, a fast sender would already have exhausted the window). Some of the packets that are transmitted on the lossy link, *after* the first lost packet, will get through, and will not need to be resent. We denote the mean number of such packets as $residual\_good$.

Thus, for each TCP version, we have a Markov renewal reward process [20], embedded at the epochs $t_0, t_1, t_2, \ldots$, the reward being the number of good packets put through in each cycle. Then the throughput T is given by

$$T = \frac{\frac{1}{\gamma} + residual\_good}{mean\_cycle\_duration} \tag{9}$$

With a link protocol, and high vehicle speeds, we assume a Bernoulli loss model at the TCP packet level. Hence, for this case the analysis in [10] carries through directly, after accounting for the fact that TCP packet transmission times are longer.

We will assume that the minimum timeout is large compared to the other time scales in the local network; this is true for the numerical parameters that we used in [10] (0.75sec minimum timeout, as in BSD, and 7.5msec packet transmission time). Then, if the loss in a cycle is followed by a timeout, we assume that the beginning of the next cycle finds the Markov loss process in its stationary distribution. This is always the case for OldTahoe, since it recovers only after a timeout. In the case of Tahoe, however, if fast retransmission takes place then we need to know the state of the Markov loss process at the beginning of the next cycle.

The following analysis also assumes that the LAN host can transmit much faster than the wireless link. Hence, during the packet transmission periods in a cycle, the wireless link is assumed to be continuously busy. This assumption facilitates the carrying over of the channel state from one packet to the next.

4.1 Analysis of the Markov Chain $\{U_k\}$

Owing to the above viewpoint, the transition probabilities of $\{U_k\}$ are slightly different from those in [10]. For $M \in \{2, \ldots, W_{max}\}$, define

$\theta_M =$ Prob{the next cycle starts with $W_{th} = \lceil \frac{M}{2} \rceil$, given that the loss window in this cycle is M}

$\overline{\theta}_M =$ Prob{the next cycle is forced to start with $W_{th} = 2$, owing to additional timeouts, given that the loss window in this cycle is M $(= 1 - \theta_M)$}

Following the development in [10] we can now write down the transition probabilities. Given that $U_k = M \in \{2, \ldots, W_{max}\}$ (and $m := \max(2, \left\lceil \frac{M}{2} \right\rceil)$), recalling the Markov loss model, and writing $\overline{\gamma} = 1-\gamma$, define two transition probability matrices P and Q as follows:

$$
\begin{aligned}
p_{M,j} &= \begin{cases} \overline{\gamma}^{(j-1)-1}\gamma & 2 \le j \le m-1 \\ \overline{\gamma}^{d(k)-1}(1-\overline{\gamma}^{m+k}) & j = m+k, \\ & 0 \le k \le (W_{max}-1-m) \\ \overline{\gamma}^{d(W_{max}-m)-1} & j = W_{max} \end{cases} \\
q_{M,j} &= p_{3,j} (= p_{2,j} = p_{4,j}) \quad \text{for } 2 \le j \le W_{max}
\end{aligned}
$$

where

$$
d(k) = (k+1)((m-1)+\frac{k}{2})
$$

Note that P corresponds to the case in which the next cycle starts with $W_{th} = \lceil \frac{M}{2} \rceil$, and Q corresponds to the case in which the next cycle is forced to start with $W_{th} = 2$. Finally, we define the vector

$$
\underline{\Theta} = (\theta_2, \ldots, \theta_{W_{max}}),
$$

and then the transition probability matrix of $\{U_k\}$ is written as

$$
\text{diag}(\underline{\Theta})P + (I - \text{diag}(\underline{\Theta}))Q, \tag{10}
$$

where I is the $(W_{max}-1) \times (W_{max}-1)$ identity matrix.

The matrices P and Q are common to the OldTahoe and Tahoe versions with the same values of W_{max} and Markov loss model transition probabilities γ and β; as shown below $\underline{\Theta}$ depends on γ and β for both OldTahoe and Tahoe.

OldTahoe: Since we assume that after a timeout the Markov channel model is found in its stationary distribution, for $M \in \{2, \ldots, W_{max}\}$,

$$
\theta_M = \frac{\beta}{\gamma+\beta} = 1 - \theta_M
$$

Tahoe: Define, for $M \in \{2, \ldots, W_{max}\}$,

$\phi_M =$ Prob{the good transmission period in a cycle is followed by fast retransmission, if the loss window is M}

$\phi_M^{(G)} =$ Prob{the good transmission period in a cycle is followed by fast transmission, and at this epoch the channel is in the Good state}

$\phi_M^{(B)}$ = Prob{the good transmission period in a cycle is followed by fast transmission, and at this epoch the channel is in the Bad state (i.e., $\phi_M^{(B)} = \phi_M - \phi_M^{(G)}$)}

It follows that

$$\theta_M \quad = \quad \phi_M^{(G)} + (1 - \phi_M)\frac{\beta}{\gamma + \beta}$$

Recursive equations were developed for computing $\phi_M^{(G)}$ and $\phi_M^{(B)}$; owing to lack of space here we refer the reader to the full paper [11].

Thus the transition probabilities of the process $\{U_k\}$ can be calculated by first computing $\phi_M, \phi_M^{(G)}, \phi_M^{(B)}$, and θ_M, and finally using Equation 10. The stationary distribution can be obtained by any of the many standard techniques. Denote the stationary distribution by

$$u_M, 2 \leq M \leq W_{max}$$

4.2 Throughput Analysis

Given that the loss window is M, the probability that $k \leq (M-1)$ of the remaining packets get through is just $g_{M-1}^{(k)} + b_{M-1}^{(k)}$. Given the stationary distribution $u_M, 2 \leq M \leq W_{max}$, the mean number of good packets transmitted in a cycle after the first loss occurs can thus be calculated. This is what we had called *residual_good* (see Equation 9).

We adopt the approximate mean cycle time analysis discussed in Section 4.2 of [10]. Letting Z denote the mean of the loss window distribution, we take the average window during packet transmissions to be $0.75Z$, and take the network throughput to be $r = r(\lambda, 0.75Z)$ (λ is the LAN packet transmission rate normalised to the wireless link transmission rate), where $r(\lambda, w) = \frac{(\lambda^{(w+1)} - \lambda)}{(\lambda^{(w+1)} - 1)}$, if $\lambda \neq 1$, and $r(1, w) = \frac{w}{1+w}$.

We assume that the timeout is always the minimum timeout (rto_min) (a good assumption for a large coarse timeout and the local network situation). It follows that the recovery time due to repeated, exponentially growing timeouts, is given by $\frac{rto\_min}{1 - 2\frac{\gamma}{\gamma+\beta}}$ (for this equation to hold, we must have $\frac{\gamma}{\gamma+\beta} <$ 0.5).

It follows that the throughput of OldTahoe is given by

$$T_{OldTahoe} = \frac{\frac{1}{\gamma} + residual\_good}{\frac{1}{\gamma r} + \frac{rto\_min}{1 - 2\frac{\gamma}{\gamma+\beta}}}$$

where the first term in the denominator is the mean time for transmitting $\frac{1}{\gamma}$ good packets. The throughput of Tahoe is given by

$$\begin{aligned} T_{OldTahoe} &= (\frac{1}{\gamma} + residual\_good) \div \\ & \left(\frac{1}{\gamma r} + \sum_{M=2}^{W_{max}} u_M \left((1-\phi_M)\frac{rto\_min}{1-2\frac{\gamma}{\gamma+\beta}} + \right.\right. \\ & \left.\left. \phi_M^{(G)}\frac{M}{r} + \phi_M^{(B)}(\frac{M}{r} + \frac{rto\_min}{1-2\frac{\gamma}{\gamma+\beta}})\right)\right) \end{aligned}$$

Here the second term in the denominator is an expectation over the loss window distribution; the term corresponding to each M is the sum of the recovery durations for three possibilities: the first if fast retransmit fails, the second if fast retransmit succeeds and the next cycle finds the channel in the Good state, and the last if fast retransmit succeeds but the channel is found in the Bad state, necessitating a timeout based recovery.

5 Numerical Results and Discussion

We present numerical results for the same numerical parameters as in [10]. The wireless link speed is 1.5Mbps (all rates are normalised to this rate) and the LAN host can transmit at 5 times the wireless link rate (i.e., $\lambda = 5$). The TCP packet length is 1400 bytes; i.e., its transmission time on the wireless link is 7.5ms; various times are normalised to this transmission time. The minimum timeout is 750ms; or 100 packet service times on the wireless link. The fast retransmit threshold is $K = 3$. The carrier frequency f_c is taken to be 900Mhz.

We consider DPSK (differential phase shift keying) modulation. In this analysis we do not consider forward error correction coding, or bit interleaving. The link protocol operates directly over the modulation scheme, and has an error detection code. With AWGN the BER for DPSK is given by

$$\epsilon = 0.5 \exp(-\frac{E_b}{N_0}) \tag{11}$$

Thus without a link protocol, in AWGN, the TCP packets can be assumed to be lost independently with probability p given by

$$p = 1 - (1-\epsilon)^{packet\_length} \tag{12}$$

We will provide numerical results with AWGN for comparison purposes. In

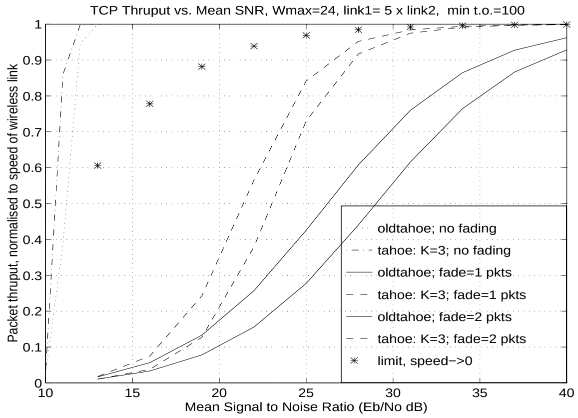

Figure 3: Throughput of versions of TCP vs. mean signal to noise ratio; no link protocol; K is the fast retransmit threshold.

Figure 3 we plot the throughput of TCP OldTahoe and Tahoe versus the average SNR, with and without fading (i.e., AWGN). There is no link protocol; thus, whenever a packet encounters a fade that packet is certainly lost. In the case of fading, two curves are shown for each version, corresponding to average fade durations of 1 and 2 packets. The Bad state corresponds to an SNR $\leq$ 10dB.

Figure 3 shows that with AWGN an SNR of about 12dB yields very good throughput, whereas, with fading, an SNR of about 30dB is required to obtain a throughput better than 90% from TCP Tahoe. TCP OldTahoe requires an SNR of 40dB. For a given SNR, increasing the fade length appears to improve the TCP throughput. This observation is explained as follows. The "fade limit" of 10dB, taken together with the average SNR, fixes α (see Section 2.2), and hence fixes δ; i.e., for a given value of $\frac{E_b}{N_0}$ the ratio of the good and the bad periods is fixed (see Equation 8). It follows that, for fixed $\frac{E_b}{N_0}$, if the fade duration is increased from 1 to 2 packets then the good periods also increase by a factor of two. Thus, although increasing the fade duration results in a greater frequency of consecutive losses, since the good duration also increases, the TCP window can build to large values before losses do occur. This yields a larger throughput for increasing fade duration. These observations hold for low speeds at which the fade durations are comparable to the packet transmission times (e.g., $\frac{E_b}{N_0}$=20dB implies speeds of about 10 to 20kmph).

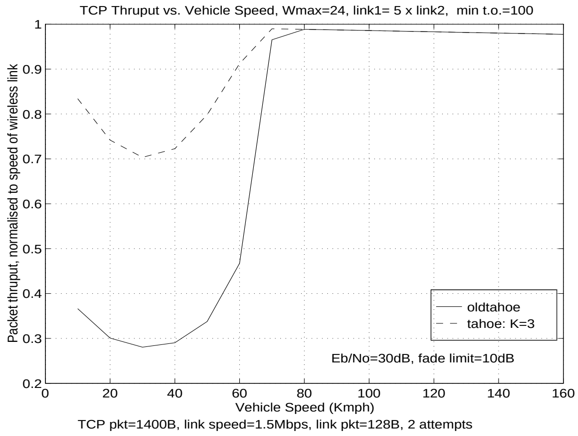

Figure 4: Throughput of versions of TCP vs. vehicle speed, for fixed $\frac{E_b}{N_0}$ = 30dB, and fade limit = 10dB; each link packet is attempted 2 times.

Consider what would happen if the vehicle speed was allowed to reduce to zero. For example, if $\frac{E_b}{N_0}$ = 30dB then the probability that a connection *starts* in the bad state is 0.01 (see Equation 8). For very low speeds, during the entire duration of the connection, the channel will be in the same state as it was found at the beginning of the connection. Thus if the initial state found is bad the throughput is 0, otherwise the throughput is 1. Hence, averaged over connections, the average throughput seen by a TCP connection will be 0.99. It follows that in Figure 3, for a fixed SNR of 30dB, as the vehicle speed decreases the throughput of both Tahoe and OldTahoe will converge to 0.99. Similar calculations can be done for each value of SNR; these results are plotted as "limit, speed $\rightarrow$ 0" in Figure 3. Note that we have no link protocol in the model for which results are shown in Figure 3. With a link protocol, the throughput should be no worse than without one. Thus, we see that for speeds approaching zero the average throughput increases and converges to the throughput obtained from a "quasistatic" analysis assuming very long fade durations and good durations but with the appropriate probabilities. This observation is consistent with the results with a link protocol that we present next. We now fix the average SNR (to 30dB or 25dB), and, for a fixed fade limit of 10dB, we study TCP throughput as a function of vehicle speed. We now include the link protocol model in our analysis (see Section 2.3).

In Figures 4 and 5 we show the TCP throughout versus vehicle speed. The link packet length is 128 bytes; thus there are 11 link packets in a TCP

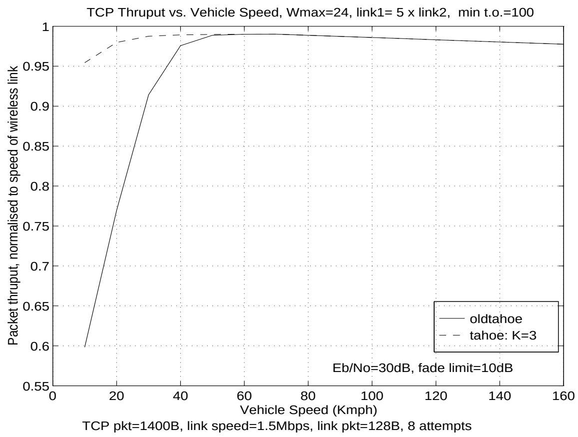

Figure 5: Throughput of versions of TCP vs. vehicle speed, for fixed $\frac{E_b}{N_0}$ = 30dB, and fade limit = 10dB; each link packet is attempted 8 times.

packet. In Figure 4 each link packet is attempted $N = 2$ times, whereas in Figure 5 each packet is attempted $N = 8$ times.

Figure 4 shows that, as the vehicle speed increases from a small value, first the throughput (for both versions) decreases, hits a minimum, and then it increases. This is explained as follows. Note that if the fade duration is longer than the number of attempts made for a link packet then that link packet and the corresponding TCP packet is surely lost. To the left of the minimum point, the mean fade duration is longer than 2 link packets, and increases as the speed decreases; however, as observed earlier, the good period length increase too, hence the throughput increases. To the right of the minimum point, the fade duration is smaller than 2 packets, and for larger vehicle speeds there is increasing probability that not all attempts of a link packet fail.

We note that the behaviour of throughput versus vehicle speed, for fixed SNR, in Figure 4 is consistent with our discussion for low speeds in the context of Figure 3. The throughput does increase for low vehicle speeds. If the curves in Figure 4 are extrapolated as speed goes to zero, however, the limit will not match that obtained in the context of Figure 3. This is because the model used to obtain the results in Figure 4 (fading model at the link packet level, but Bernoulli losses across TCP packets) does not apply for very low vehicle speeds for which there is significant fade correlation between TCP packets.

Figure 5 shows that each link packet needs to be attempted 8 times for

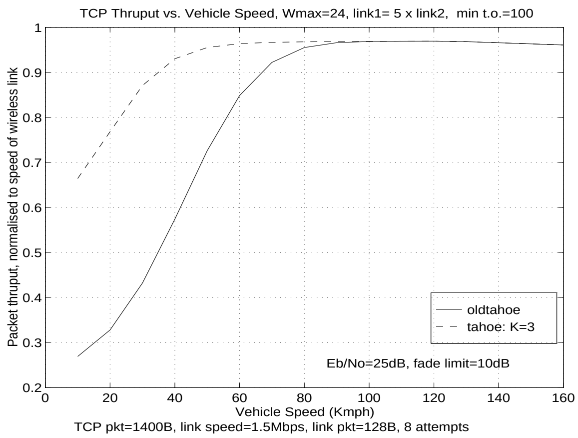

Figure 6: Throughput of versions of TCP vs. vehicle speed, for fixed $\frac{E_b}{N_0} =$ 25dB, and fade limit = 10dB; each link packet is attempted 8 times.

Tahoe to yield a good throughput for a large range of vehicle speeds. In Figure 6 we show the effect of reducing the SNR by 5dB. We find that 8 attempts are no longer enough even for Tahoe. The number of link packet attempts need to be increased to 16 for Tahoe to provide reasonably good performance (see [11]).

6 Conclusion

For the default parameters of the BSD implementation of TCP, over a 1.5Mbps wireless link, as a general observation, we find that (without physical level enhancements, such as forward error correction and bit interleaving[1]) an SNR of 25dB to 30dB is required to obtain from Tahoe a throughput greater than 90% of the wireless link speed. Without a link protocol, OldTahoe requires 40dB; and with a link protocol OldTahoe suffers at low vehicle speeds, for which the fade durations are large. When using a link protocol, the choice of link packet length and the number of attempts for each packet are important parameters. Basically, the attempts have to "outlast" the fade. Since very large link packets defeat the idea of using link packets, small link packets have to be attempted several times. For a fixed SNR, a given link packet

[1] In any case, forward error correction and interleaving are not much help when there is slow fading.

length, and a given number of link packet attempts, we find that throughput varies in an interesting way with vehicle speed. At very low speeds (e.g., pedestrian speeds) the throughput is high; it decreases with increasing vehicular speed until the fade durations become shorter than the number of link packet attempts. Beyond this speed, again the throughput increases, and is limited only by the expansion of the TCP packet transmission time owing to link-level retransmissions.

Much work remains to be done to develop more comprehensive protocol analyses even with the simple two state Markov loss model. Our analysis at present does not apply to many situations of interest; e.g., link protocols at very low speeds, or with large link packets. The Markov model can also be enhanced to include a nonzero loss probability in the Good state. Both these situations will require a more complex analysis, as the state of the Markov loss model will need to be included in the evolution of the TCP window process. Such enhancements of the analysis will be fruitful as future research. Other TCP versions, such as Reno and NewReno, also need to be analysed.

References

[1] A. Bakre and B.R. Badrinath, "I-TCP: Indirect TCP for mobile hosts," *Proc. 15th IEEE International Conf. on Distributed Computing Systems (ICDCS)*, pp 136-143, May 1995.

[2] H. Balakrishnan, V.N. Padmanabhan, S. Seshan, and R.H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Transactions on Networking*, Vol 5, No. 6, pp 756-769, December 1997.

[3] R. Cáceres, and L. Iftode, "Improving the performance of reliable transport protocols in mobile computing environments," *IEEE Journal on Selected Areas in Communications*, Vol. 13, No. 5, pp 850-857, June 1995.

[4] H. Chaskar, T.V. Lakshman, U. Madhow, "On the design of interfaces for TCP/IP over wireless," *Proc. IEEE MILCOM'96*.

[5] A. Desimone, M.C. Chuah, and O.C. Yue, "Throughput performance of transport layer protocols over wireless LANs." *Proc. IEEE Globecom'93*, December 1993.

[6] K. Fall, and S. Floyd, "Comparisons of Tahoe, Reno, and Sack TCP," manuscript, ftp://ftp.ee.lbl.gov, March 1996.

[7] E.N. Gilbert, "Capacity of a burst-noise channel," *Bell Systems Technical Journal*, Vol. 39, pp. 1253-1265, Sept. 1960.

[8] V. Jacobson, "Congestion avoidance and control," *Proc. ACM Sigcomm '88*, August 1988.

[9] William C. Jakes, *Microwave Mobile Communications*, John Wiley and Sons, New York, 1974.

[10] Anurag Kumar, "Comparative performance analysis of versions of TCP in a local network with a lossy link," WINLAB Technical Report No. 129, Rutgers University, October 1996. Also to appear in *IEEE/ACM Transactions on Networking*, Vol 6, No. 4, August 1998.

[11] Anurag Kumar and J.M. Holtzman, "Comparative performance analysis of versions of TCP in a local network with a lossy link: Rayleigh Fading Mobile Radio Link," WINLAB Technical Report No. 133, Rutgers University, November 1996.

[12] T.V. Lakshman and U. Madhow, "The peformance of TCP/IP for networks with high-bandwidth delay products and random loss," *IEEE/ACM Transactions on Networking*, Vol. 5, No. 3, pp 336-351, June 1997.

[13] Edward A. Lee and David G. Messerschmitt, *Digital Communication*, Kluwer Academic Publishers, Boston, 1988.

[14] P.P. Mishra, D. Sanghi, and S.K. Tripathi, "TCP Flow Control in Lossy Networks: Analysis and Enhancements," in IFIP Transactions C-13 *Computer Networks, Architecture and Applications*, S.V. Raghavan, G.v. Bochmann, and G. Pujolle (eds.), Elsevier North Holland, pp. 181-193, Amsterdam, 1993.

[15] J.D. Parsons, *The Mobile Radio Propagation Channel*, Pentech Press, London, 1992.

[16] T.S. Rappaport, *Wireless Communications*, IEEE Press, New York, 1996.

[17] S.O. Rice, "Distribution of the duration of fades in radio transmission," *Bell Syst. Tech. Jour*, Vol. 37, pp. 581-635, May 1958.

[18] W.R. Stevens, *TCP/IP Illustrated, Volume 1*, Addison-Wesley, Reading, Mass., Nov. 1994.

[19] Hong Shen Wang and Nader Moayeri, "Finite-state Markov channel – a useful model for radio communication channels," *IEEE Transactions on Vehicular Technology*, vol. 44, no. 1, pp 163-171, February 1995.

[20] R.W. Wolff, *Stochastic Modelling and the Theory of Queues*, Prentice Hall, Englewood Cliffs, N.J., 1990.