The stationary behavior of idealized TCP congestion avoidance

Teunis J. Ott   J.H.B. Kemperman   Matt Mathis

August 22, 1996

Abstract

This note derives the stationary behavior of idealized TCP congestion avoidance. More specifically, it derives the stationary distribution of the congestion window size if loss of packets are independent events with equal probability. The mathematical derivation uses a fluid flow, continuous time, approximation to the discrete time process (W_n), where W_n is the congestion window after the n-th "good" acknowledgement packet (i.e. not a duplicate acknowledgement). We derive explicit results for the stationary distribution and all its moments.

Congestion avoidance is the algorithm used by TCP to set its window size (and indirectly its data rate) under moderate to light segment (packet) losses.

The congestion avoidance mechanism we model is idealized in the sense that loss of multiple packets does not lead to time-out phenomena. Such idealized behavior can be implemented using Selective Acknowledgements (SACKs). As such, our model predicts behavior of TCP with SACKs. It also is an approximate model in other situations.

Among the results is that if every packet is lost with a (small) probability p, average window size and long range throughput (when transporting a file large enough to reach stationarity) are of the order of $1/\sqrt{p}$.

1. Introduction

In this paper we study the stationary behavior of the congestion window size of an idealized form of the TCP congestion avoidance algorithm. TCP (Transport Control Protocol, see e.g. [4]) is the "Reliable Transport" Protocol in the Internet. Let W_n be the congestion window size immediately after the n-th "good" acknowledgement packet (non-duplicate acknowledgement) has been received by the source and the congestion window has been updated[1]. The congestion window size is halved upon receipt of abstract "congestion indicators". While those congestion indicators can have many forms, we conveniently think of them as lost packets (third duplicate acknowledgements), and we assume that packet losses are independent events with small but positive probability. This probabilistic assumption makes it possible to consider $(W_n)_{n=1}^{\infty}$ as a stochastic process, of which we are going to study the stationary distribution.

This paper is a logical precursor to [1] which appeared earlier. A version of this paper which contains more details can be found in [2].

We define *idealized* TCP congestion avoidance in the spirit of Van Jacobson's landmark paper [3]. When TCP receives a congestion signal (e.g. detects a missing segment) the congestion window is halved. As long as there is no congestion signal, the congestion window is slowly increased. If the Congestion Window is W Maximal Segment Sizes (MSSs), then the window is opened by $1/W$ for every received "good" TCP acknowl-

[1] We use the term *segment* to refer to TCP data units, and the term *packet* to refer to be the IP transmission units, including data and headers. Typically they are different only by the TCP headers.

edgement. In our idealized model, the first "good" acknowledgement after a congestion episode or congestion signal results in a congestion window half of that after the last "good" acknowledgement before the congestion period or congestion signal.

It is well known that existing versions of TCP do not achieve this idealized behavior. TCP-Reno (with fast retransmit and fast recovery algorithms) [4] comes close, but has problems if two or more segments are lost within one congestion window [10] or even when only one packet is lost while the congestion window is fewer than four MSSs. Under these conditions TCP-Reno often experiences superfluous retransmission time–outs and other misbehaviors. These behaviors, as well as possible TCP modifications to overcome them are under active investigation [5, 6, 7, 8].

As stated above, we model the situation where the congestion signals are independent from one another. It would be nice to assume that every lost segment signals a new congestion event, but this is somewhat unrealistic in the current Internet, where losses are often highly correlated. Future work will explore several avenues which bring this assumption into agreement with reality. First, the ubiquitous deployment of RED (Random Early Detection) [11] will greatly reduce correlated losses in the Internet. Second, newer TCP implementations [13, 14] tend to treat multiple closely spaced losses as a single congestion signal. Furthermore, other mechanisms besides segment loss can be used to signal congestion [12]. The model and analysis in the paper apply as long as the congestion signals are independent from one another. For ease of discussion we assume that they are indicated by RED style independent segment losses.

To study TCP congestion avoidance without interference from other mandatory TCP algorithms we must assume that TCP is operating in an environment where it is carrying unidirectional bulk data. While all or most of the assumptions below can be relaxed, together they are a good description of what we call the canonical situation.

1. There is always data pending at the sender, such that the sender can always send data as permitted by the congestion window.

2. The receiver's advertised window is sufficiently large to never constrain the congestion window.

3. TCP has previously negotiated the maximum segment size, (MSS), and all segments are exactly one MSS.

4. Every segment is individually acknowledged (the *delayed acknowledgement* algorithm is not in effect). Under delayed acknowledgement one acknowledgement often acknowledges two or more data segments. When we investigate delayed acknowledgement we must decide how much the window increases for such acknowledgements.

5. We assume that recovery from lost segments does not interact with congestion avoidance. This is part of saying that we consider an *idealized* version of TCP. Achieving this one of the design goals of future TCP implementations such as TCP-SACK.

Throughout this paper we assume that segment losses are independent (from segment to segment) events with all the same, positive but small, probability p.

Let W_n be the congestion window size right after the n-th "good" (i.e. non-duplicate) acknowledgement has been received. Our assumptions make it possible to consider $(W_n)_{n=1}^{\infty}$ as a stochastic process, of which we are going to study the stationary distribution.

The first and second assumptions imply that W_n can grow quite large. The third assumption means the congestion window is expressed in units of Maximal Segment Sizes (MSSs) segments, as opposed to, for example, bytes. Henceforth, we allow W_n to take on all values in $(0, \infty)$.

This means (W_n) evolves as follows:

Given W_n, with probability (1-p) W_{n+1} equals $W_n + (1/W_n)$, and with probability p it equals $W_n/2$. (The first part assumes the Congestion Window W is expressed in units equal to the Maximal Segments Size, and that all packets are equal to the Maximal Segment Size).

A very simple observation gives us an "order of magnitude" for the size the congestion window W_n must have most of the time: Assuming one acknowledgement per data packet (no delayed ack), when the congestion window equals W the drift of the process equals

$$\text{Drift} = \frac{1-p}{W} - p\,\frac{W}{2}. \tag{1.1}$$

This drift is zero when

$$W = W^* = \sqrt{\frac{2(1-p)}{p}}. \tag{1.2}$$

Moreover, for values of W larger than W^* as in (1.2) the drift is negative and for values of W smaller than W^* the drift is positive: The drift is always toward W^* and gets stronger when we get farther away. This "proves" that in the long run the congestion window is almost always of the order of magnitude of W^* as in (1.2). In addition, in the long run the average value of the drift must be zero. Hence, by Jensen's inequality

$$0 = E[\frac{1-p}{W} - p\,\frac{W}{2}] \geq \frac{1-p}{E[W]} - p\,\frac{E[W]}{2}, \tag{1.3}$$

so that (1.2) becomes a (presumably usually sharp) inequality:

$$E[W] \geq \sqrt{\frac{2(1-p)}{p}}. \tag{1.4}$$

Result (1.2) has been known for a long time to at least the first author of this paper, and probably has been derived independently by many researchers. If the TCP destination uses delayed acknowledgements we lose the factor “2” in (1.2) and (1.4). A more exact analysis will generate yet another factor, see section 2. In the case of loss that is not random but exactly once every p^{-1} packets results similar to (1.2), (1.4) were obtained in [15] (formula 32), and related results can be found in [16]. [9] predicted that if one TCP flow goes through one bottleneck and another through n “very similar” bottlenecks (and therefore has a loss probability n times that of the first one), the ratio between the throughputs of the two flows is $\sqrt{n}$. [9] also verified this result by simulation.

A more interesting result is that for p approaching zero $\sqrt{p} \times W_n$ has a stationary distribution which is almost independent of p, and there is a limiting stationary distribution (in the sense of weak convergence) for $p \downarrow 0$. In the next few paragraphs we provide the setting for that result. That (limiting) stationary distribution will be derived in the sections 4 and 5 of this paper. Similar results can be obtained for the case of periodic loss (as opposed to random loss).

It must be noted that as long as the Maximal Window (Advertised Window) is reasonably large compared with W^* in (1.2), most of the results quoted still are good approximations. It is possible to do an exact analysis in that case. That analysis is complicated and not very interesting.

Since p is positive but small, we can re-scale time so that the epochs of packet failure become a Poisson Process. Apart from the re–scaling, time is still expressed in number of “good” acknowledgement packets received. We also rescale space to keep $W(.)$ from going to infinity. (1.2) tells us what spatial rescaling to use. Now, we have a continuous time stochastic process $(W(t))_{0<t<\infty}$ with state–space $(0, \infty)$. More explicitly, we define $W(t)$ by

$$W(t) = \sqrt{p}\, W_{\lfloor \frac{t}{p} \rfloor}, \tag{1.5}$$

(where $\lfloor\ \rfloor$ denotes floor or integer part). It is an abuse of notation to have simultaneously a discrete time process (W_n) where W_n is the window size after the arrival of the n-th "good" acknowledgement, and a continuous time process $(W(t))$ which is defined by (1.5). We do not expect confusion between the two. For p small, the process $W(t)$ approximately behaves as follows: There is a Poisson Process of intensity 1. In between the points of the Poisson Process $W(.)$ behaves as

$$(d/dt)W(t) = \frac{1}{W(t)}, \tag{1.6}$$

and in the points τ_k of the Poisson Process the behavior is that

$$W(\tau_k+) = \frac{1}{2}W(\tau_k-) \text{ (limit from right is half limit from left).} \tag{1.7}$$

With the construction (1.5) the Poisson Process has intensity one (in average one packet loss per unit of time: the original process has in average one packet loss per $1/p$ packets). For several reasons we allow the Poisson Process to have an intensity c_1 which need not be equal to one, and we slightly modify (1.6). The evolution of $W(.)$ now is as follows: There is a Poisson Process with intensity c_1, with points $(\tau_k)_{k=1}^{\infty}$. In between the points τ_k the process $W(.)$ behaves as

$$(d/dt)W(t) = \frac{c_2}{W(t)}, \tag{1.8}$$

and in the points τ_k the behavior still is as in (1.7).

The choice of c_1 and c_2 is related to our choice of "time". Thus far, time progressed by one for every "good" acknowledgement received. (Or a re-scaled version of that time). We could also let "time" increase by the increase in the "next expected" byte (but expressed in MSSs) since the previous "good" acknowledgement. As long as there is no delayed acknowledgement, and away from "congestion indicators", the two "times" increase at the same rate. Henceforth, when we increase time by one for every good acknowledgement we call it "ack-time" while if we increase time by the increase in "next expected byte" (in MSSs) we call it "packet-time". As we just saw, as long as there are no delayed acknowledgements, the two "times" increase at the same rate (away from congestion indicators).

If there is delayed acknowledgement, and assuming all acknowledgement packets acknowledge exactly two packets, we have a choice: we can increase time by one for every received "good" acknowledgement (this means essentially doubling the c_1) or we can use "packet-time" (this essentially means halving c_2).

Yet other values of c_1 and/or c_2 occur if we count "time" in bytes instead of in packets or acknowledgements, or if after a "good" acknowledgement the increase in the congestion window depends on the number of bytes acknowledged.

Formally, if we use "packet-time" instead of "acknowledgement-time" the continuous time process $(W(t))_{t=0}^{\infty}$ as in (1.7) and (1.8) is a good approximation (after rescaling) for the discrete time process $(W_n)_{n=0}^{\infty}$ as long as after an acknowledgement packet acking *Acked* MSSs, the Window W (MSSs) increases by $c_2 Acked/W$ MSSs, and that in a sequence of packets containing (together) N MSSs (N need not be integer), the probability of exactly one halving of the window is

$$c_1 Np + o(Np) \text{ if } Np \text{ is small,} \tag{1.9}$$

and the probability of two or more halvings is an order of magnitude smaller.

It is clear, and will be made formal, that only the ratio of c_1 and c_2 matters. We define

$$\eta = \frac{c_1}{c_2}. \tag{1.10}$$

The stationary distribution of $W(.)$ must have most of its probability in the general area of $\sqrt{\frac{2}{\eta}}$. To see this we repeat the argument in (1.1) - (1.3): Let $D(w)$ be the drift of $W(t)$ if $W(t) = w$. Then

$$D(w) = \frac{c_2}{w} - \frac{c_1 w}{2}, \tag{1.11}$$

and

$$D(w) = 0 \text{ if } w^2 = \frac{2}{\eta}. \tag{1.12}$$

For smaller values of w the drift is positive, and for larger values of w the drift is negative. This gives the ad-hoc approximation

$$\overline{W} = \sqrt{\frac{2}{\eta}}. \tag{1.13}$$

In the remainder of this paper we will refine this observation. (1.13) is an ad-hoc approximation for the mean or average value of W in the process $W(.)$ as defined by (1.7), (1.8). Because of the definition (1.5), $\overline{W}$ therefore (apart from the factor $\sqrt{p}$) is an ad hoc approximation for the "acknowledgement–arrival" average of the window size in the original process W_n of interest. In section 3 we will see that for certain purposes we need the "time" average of the original. This is the average obtained by integrating over the clock time, as opposed to by summing over acknowledgement arrivals. The difference between "acknowledgement–arrival" and "clock–time" is essentially the same as the difference between "customer–arrival" averages and "time" averages in classical queueing theory. Since (in "clock–time"(!)) acknowledgements arrive at a higher rate when the congestion window is large, the "acknowledgement–arrival" average value of the window size is larger than the "time" average value. We will see that in fact (1.13) gives a value in between the time-average value and the acknowledgement–arrival average.

In the introduction above we chose W_n to be the congestion window right after the n-th "good" acknowledgement packet was received. This means that $W_n \geq 2$ (MSSs): at a time-out, the congestion window is reset to one, and a packet is re-transmitted. When the acknowledgement of that packet is received, W becomes 2 and that is the new value. (See also the discussion in the first paragraphs of section 7). In our theoretical treatment we disregard these details.

We could instead have considered the congestion window after the n-th data packet was acknowledged. This process has almost the same dynamics as the process (W_n), but its state becomes ill defined during fast recovery. We often make the (minor) error of regarding the two processes as identical. For $p \downarrow 0$ the rescaled versions of the two processes get the same dynamics.

In this context, it must be remembered that in all circumstances for the process $W(t)$ *time* is *not* "clock-time", but "ack-time": (some rescaled version of) total number of "good" acknowledgement packet received. The latter is almost, but not quite, the same as "packet-time": (some rescaled version of) total number of packets successfully transmitted. Thus, now we have "clock-time", "ack-time" and "packet-time". Let us call "clock-time" τ and let us call "ack-time" t. τ and t are increasing together, but they do not necessarily have the same rate of increase. For example, let us assume that the Round Trip Time (RTT) of packets is independent of the congestion window W and that the source always sends $\lfloor W \rfloor$ packets in one Round Trip Time (always has $\lfloor W \rfloor$ packets in flight). Then we have (approximately, and disregarding periods of fast recovery or slow start)

$$dt = \frac{W}{RTT} d\tau. \tag{1.14}$$

(1.14) is an approximate result before the time-rescaling (1.5) (it holds only over intervals of length large compared with RTT) but becomes exact if we first rescale time as in (1.5) and then let p go to zero. A similar result can be obtained for "packet-time" (where it actually is a bit sharper). Because we assume no delayed acknowledgements, away from fast recovery and slow start ack-time and packet-time increase at the same rate.

If the RTT *does* depend on W, we obviously have results similar to (1.14). We leave this as an exercise to the reader. Section 3 contains an example of how the stationary distribution in "clock-time" can be derived from the stationary distribution in "ack-time". This discussion of various kinds of time can be made fancier by calling t and τ "subjective times", where the ratio of the rates of increase of different "subjective times" depends on the state of the process. The choice of "ack-time" instead of "clock-time" is what makes the analysis of the window behavior quite transparent. In fancier language: A hard problem has been made easy by choosing the correct subjective time. The stationary distributions under various subjective times are related and usually can be obtained from one another by taking the sampling bias into consideration, see section 3.

In section 4 we will introduce a slightly larger class of stochastic processes than those as in (1.7) – (1.8), and in the sections 4 and 5 we will explicitly derive the stationary distributions and moments of this slightly larger class of processes. Section 6 gives a short proof of some auxiliary results.

Results specific to the Window Behavior that follow from the sections 4 and 5 will be quoted without proof in section 2.

In section 3 we will give a simple example of how the material developed in this paper can be used to predict the throughput of a TCP connection.

2. The stationary distribution of the congestion window size

The results quoted in this section are either obvious (e.g. (2.4), which follows from (2.1) and (2.3)) or special cases of results in the sections 4 and 5. In those sections "backward pointers" will be given on how to obtain the results below. In section 5 it will also be proven that the three different definitions in (2.2) respectively (2.7) are really the same.

Let W be a random variable with as distribution the stationary distribution of the process $W(.)$ in (1.7), (1.8). Then:

$$\frac{\eta}{2}E[W^2] = \frac{1}{1-\frac{1}{4}} = \frac{4}{3}, \quad E[W^2] = \frac{8}{3\eta}. \tag{2.1}$$

Define:

$$A = \prod_{j=1}^{\infty}\left(\frac{1-2^{-(2j+1)}}{1-2^{-2j}}\right) = \frac{\sum_{k=0}^{\infty}\frac{(-1)^k 2^{-k(k+2)}}{(1-2^{-2})(1-2^{-4})\ldots(1-2^{-2k})}}{\sum_{k=0}^{\infty}\frac{(-1)^k 2^{-k(k+1)}}{(1-2^{-2})(1-2^{-4})\ldots(1-2^{-2k})}} = 2\left(\sum_{k=0}^{\infty}2^{-\frac{1}{2}k(k+1)}\right)^{-1}. \tag{2.2}$$

Then:

$$\left(\frac{\eta}{2}\right)^{\frac{1}{2}}E[W] = \frac{A}{2}\sqrt{\pi}, \quad E[W] = A\sqrt{\frac{\pi}{2\eta}}. \tag{2.3}$$

Hence:

$$Var(W) = \frac{1}{\eta}\left(\frac{8}{3} - \frac{\pi A^2}{2}\right). \tag{2.4}$$

Also,

$$P\{W > w\} = \sum_{k=0}^{\infty} R_k(\frac{1}{4})exp\{-2^{2k-1}\eta w^2\}, \tag{2.5}$$

where for $|c| < 1$:

$$R_k(c) = \frac{1}{L(c)}\frac{(-1)^k c^{-\frac{1}{2}k(k+1)}}{(1-c)(1-c^2)\ldots(1-c^k)}, \tag{2.6}$$

$$L(c) = \prod_{k=1}^{\infty}(1-c^k) = \sum_{k=0}^{\infty}\frac{(-1)^k c^{\frac{1}{2}k(k+1)}}{(1-c)(1-c^2)\ldots(1-c^k)} = \sum_{k=-\infty}^{+\infty}(-1)^k c^{\frac{1}{2}k(3k+1)}. \tag{2.7}$$

For A in (2.2) and for $L(c)$ in (2.7) three expressions are given. Section 5 will prove, among other things, that those expressions are in indeed equivalent. For numerical purposes it seems the rightmost infinite sums are preferable over the middle infinite sums, which are preferable over the infinite products.

In section 5 we will prove that

$$\left(\frac{2}{\eta}\right)^{\frac{1}{2}}E[\frac{1}{W}] = \frac{A}{2}\sqrt{\pi}, \quad E[\frac{1}{W}] = \frac{A}{2}\sqrt{\frac{\pi\eta}{2}}, \tag{2.8}$$

so that indeed the expected value of the drift is zero:

$$E[\frac{c_2}{W} - \frac{c_1}{2}W] = c_2 A\sqrt{\pi\eta}\left(\sqrt{\frac{1}{8}} - \sqrt{\frac{1}{8}}\right) = 0. \tag{2.9}$$

(2.8) shows that

$$\frac{1}{E[1/W]} = \frac{2}{A}\sqrt{\frac{2}{\pi\eta}}. \tag{2.10}$$

and in section 3 we will see that (under a common-sense assumption for source behavior) this is a good approximation for the "clock-time" average value of the process $W(.)$.

For numerical values of coefficients of interest we find (using the sums up to $k = 6$ only (!)):

$$A = 1.218\,299\,422\,132\,457, \quad L(\frac{1}{4}) = 0.688\,537\,537\,120\,339,$$
$$A\sqrt{\frac{\pi}{2}} = 1.526\,911\,889\,241\,912, \quad \frac{8}{3} - \frac{\pi A^2}{2} = 0.335\,206\,749\,158\,359,$$
$$\frac{st.dev(W)}{E[W]} = 0.379\,177\,361\,667\,205,$$

and finally

$$E[W].E[\frac{1}{W}] = 1.165\,729\,958\,754\,153.$$

Even though in practical situations at most the second decimal is of interest we gave in the numerical results above the full accuracy that can be obtained with at most 6 terms in the fast converging infinite sums.

These results are consistent with the statement that "much of the time" the process $W(.)$ remains between $E[W](1-.379)$ and $E[W](1+.379)$, which in its turn is consistent with the fact that

$$\frac{1.379}{0.621} = 2.222 > 2.$$

Combining (2.3), (2.8) with some of the results above, and comparing this with (1.13), we see that

$$\frac{1}{E[1/W]} = \frac{1.309\,833}{\sqrt{\eta}}, \ \overline{W} = \frac{1.414\,214}{\sqrt{\eta}}, \ E[W] = \frac{1.526\,912}{\sqrt{\eta}}, \text{ and}$$

$$\frac{1.309\,833}{1.414\,214} = .926\,192, \quad \frac{1.526\,912}{1.414\,214} = 1.079\,690\ .$$

$\overline{W}$ lies almost exactly in the middle between the "time" average and the "acknowledgement–arrival" average, and has distance about 7.6% to both of these.

3. Applying our results in the Internet

Consider two computers communicating over a very high bandwidth network, unidirectional (data A to B only, acks B to A only) with probability of loss p per MSS, and a Round Trip Time RTT. We want to use the material developed in the other sections of this paper to predict the throughput of the connection. This section is the only one in this paper where the assumption that the source always has material to send comes into play: The amount of data sent but not yet acknowledged is assumed to always equal one congestion window. Hence, we do not consider the possibility that the source keeps part of the congestion window as an un–exercised "right–to–send". The result below also requires that there is no dependence between number of MSSs acknowledged and instantaneous window size.

Under those circumstances, the throughput (rate) (in MSSs per unit of time) equals the quotient of congestion window W (in MSSs) and Round Trip Time (RTT): exactly one Congestion Window is acknowledged in one RTT.

$$\text{Throughput (rate)} = \frac{W}{RTT}. \tag{3.1}$$

Over longer periods both the window size W and the Round Trip Time RTT vary with time. For a simple example, assume the Round Trip Time is independent of time and of W. For example, assume the RTT is exactly equal to the sum of propagation delays (no queueing component). Let $W_t^{\#}$ denote the window size at *clock time* t (yet another abuse of notation: W_n, $W(t)$ and $W_t^{\#}$ are related but not the same!). W_n is the actual window size at em packet time n, $W_t^{\#}$ is the actual window size at *clock time* t, and $W(t)$ is the spatially rescaled window at temporally rescaled *ack time* t. The total throughput over the period $[0, n.RTT]$ is (by close approximation)

$$W_0^{\#} + W_{RTT}^{\#} + \ldots + W_{(n-1)RTT}^{\#}. \tag{3.2}$$

More generally, the total throughput (in MSSs) over the period $[0, T]$ will be

$$\left(\int_0^T W_t^{\#} dt \right) / RTT. \tag{3.3}$$

If we now accept that (2.10) indeed gives the time–average (this will be shown later in this section), we get

$$\text{Long Run Average Throughput} = \frac{2}{A.RTT}\sqrt{\frac{2}{\pi \eta p}} = \frac{1.309\,833}{RTT\sqrt{p}} \text{ MSSs/sec.} \tag{3.4}$$

It is interesting to compare (3.4), the long run average throughput in case of independent random packet loss, with the long run average throughput in case of deterministic, perfectly periodic packet loss. This means that for the process $W(.)$ in (1.1) etc. the process is halved at exactly the epochs $c_1^{-1}, 2c_1^{-1}, \ldots$. Instead of (2.3), (2.10) we now get

$$E[W] = \frac{2}{3} \times \frac{4-\sqrt{2}}{\sqrt{\eta}} = \frac{1.723\,858}{\sqrt{\eta}},$$

$$\frac{1}{E[\frac{1}{W}]} = \frac{1}{(2-\sqrt{2})\sqrt{\eta}} = \frac{1.707\,107}{\sqrt{\eta}}.$$

For the long run average throughput this gives

$$\text{Long Run Average Throughput} = \frac{1.707\,107}{RTT\sqrt{p}} \text{ MSSs/sec.} \tag{3.5}$$

(3.4), (3.5) give an indication of the sensitivity to the assumption of independence.

Next, we prove the relationship between time average and acknowledgement arrival average that we stated and already used. Let $A(.)$ be the probability distribution function of the stationary distribution of window size at acknowledgement arrival epochs, and let $B(.)$ be the similar stationary distribution function for the continuous time process. In $[t, t + RTT]$ there are $W_t^{\#}$ acknowledgements. Hence:

$$dA(x) = \frac{x dB(x)}{\int_0^\infty y dB(y)}. \tag{3.6}$$

(2.10) etc. now follows readily by dividing both sides of (3.6) by x, and integrating from zero to infinity. In the special case of (3.6) we have a stronger result: Let $\phi_A(\cdot)$ respectively $\phi_B(\cdot)$ be the Laplace-Stieltjes Transforms of $A(\cdot)$, respectively $B(\cdot)$, and let μ_A^k respectively μ_B^k be the k-th moment of $A(\cdot)$ respectively $B(\cdot)$. Then:

$$\phi_A(s) = -\frac{\frac{d}{ds}\phi_B(s)}{\mu_B^1}. \tag{3.7}$$

Thus, if one of the two stationary distributions is known, both are known.

A fancier way of stating this is that if we have two different "subjective times" t and τ, and

$$dt = g(W)\, d\tau \;\; (g(W) > 0 \text{ for all } W), \tag{3.8}$$

then, if $A(\cdot)$ respectively $B(\cdot)$ are the stationary distributions of W under τ respectively t, then obviously

$$dA(x) = \frac{g(x)\, dB(x)}{\int g(y)\, dB(y)}. \tag{3.9}$$

This is actually a trivial result: think of $g(w)$ as the ratio of the sampling rates under the two subjective times.

The result above assumes all packets equal to one MSS and all data packets individually acknowledged. If there are delayed acks or smaller packets, there may be need for a factor $\sqrt{c_2}$. The actual factor must be obtained by a careful study of actual packet sizes, number of packets acked per acknowledgement, and window evolution upon arrival of an ack packet.

If part of the RTT is caused by congestion in a bottleneck link, the RTT becomes dependent on the instantaneous value of the time varying window size, and predicting throughput becomes more complicated. Ideas like (3.7) can still be used as long as $g(W) > 0$ for all $W > 0$. If there are other connections competing for bandwidth in that bottleneck link, the relationship becomes even more complicated. In that case we also need to know the RTTs of the other connections, and we must predict the throughputs of individual connections.

Random Early Drop will cause packet loss only if there is congestion in the bottleneck. In that situation, the packet loss probability becomes dependent on the buffer occupancy, i.e. on the level of congestion, in the bottleneck, and the (instantaneous) dropping probability becomes dependent on the (instantaneous) sizes of the congestion windows of the competing connections. This problem has been tackled in internal Bellcore research. In that situation the function $g(\cdot)$ is zero over certain intervals, and a different technique must be used.

With Random Early Detection (RED) the instantaneous packet loss probability becomes dependent on the history of the congestion windows, complicating the problem to the point that new techniques may be needed. Under certain circumstances diffusion approximations seem to work (Ott, work in progress).

4. The Central Result

In this section we study the Stochastic Process $(W(t))_{0<t<\infty}$ on $(0, \infty)$ which evolves as follows:

There is a Poisson Process of intensity $c_1 > 0$, with points $(\tau_k)_{k=-\infty}^{+\infty}$, with

$$\ldots < \tau_{-2} < \tau_{-1} < 0 < \tau_0 < \tau_1 < \ldots . \tag{4.1}$$

Between the points τ_k the process W(.) evolves as

$$(d/dt)W(t) = \frac{c_2}{W(t)^m}, \tag{4.2}$$

and in the points τ_k we have

$$W(\tau_k+) = \beta.W(\tau_k-), \tag{4.3}$$

where

$$0 < \beta < 1,\ m > -1,\ c_1 > 0,\ \text{and } c_2 > 0. \tag{4.4}$$

If $\beta = 1/2$ and $m = 1$ we are back in the situation of the sections 1 and 2. The results in the remainder of this section, and in the next section, indicate it is sufficient to require $m > -1$, see also the argument in section 1 based on (1.1), (1.2) or on (1.11), (1.12).

We define:

$$\eta = \frac{c_1}{c_2},\ c = \beta^{m+1}, \tag{4.5}$$

and we use the substitution

$$Z(t) = \frac{\eta}{m+1} W(t)^{m+1},\ \text{where } \eta \text{ is as in (4.5).} \tag{4.6}$$

It is easily verified that between the points τ_k

$$(d/dt)Z(t) = c_1, \tag{4.7}$$

and that in the points τ_k

$$Z(\tau_k+) = cZ(\tau_k-),\ \text{where } c \text{ is as in (4.5).} \tag{4.8}$$

We thus see that all processes $W(.)$ as above can be reduced to the special case with

$$c_2 = c_1 \text{ and } m = 0. \tag{4.9}$$

The central result in this paper is:

Theorem 1. If the random variable Z has the stationary distribution of the process $Z(.)$, then Z has the form

$$Z = \sum_{k=0}^{\infty} c^k E_k, \text{ where} \tag{4.10}$$

$(E_k)_{k=0}^{\infty}$ are independent, identically distributed random variables, all exponentially distributed with expected value 1.

Proof: Denote

$$Z(0) = z, \tag{4.11}$$

We define:

$$H_0 = \tau_0,\ H_k = \tau_k - \tau_{k-1} \text{ for } k \geq 1. \tag{4.12}$$

Clearly, $(H_k)_{k=0}^{\infty}$ are independent, identically distributed, all exponentially distributed with expected value c_1^{-1}. Also clearly,

$$Z(t) = z + c_1 t \text{ if } 0 \leq t < \tau_0, \text{ while} \tag{4.13}$$

$$Z(t) = c^{k+1} z + c_1 \left(c^{k+1} H_0 + c^k H_1 + \ldots + c H_k + (t - \tau_k) \right) \text{ if } \tau_k < t < \tau_{k+1}. \tag{4.14}$$

Using (4.14) to "look backward" from time t, we see that if

$$\text{if } \tau_j < t < \tau_{j+1} \text{ then} \tag{4.15}$$

$$Z(t) = \sum_{k=0}^{\infty} c^k E_k^{(t)}, \text{ where} \tag{4.16}$$

$$E_0^{(t)} = c_1(t - \tau_j),\ E_k^{(t)} = c_1(\tau_{j-k+1} - \tau_{j-k}) \text{ for } k \geq 1. \tag{4.17}$$

This proves the theorem. It is easily seen that the process $(Z(t))_{t=-\infty}^{+\infty}$ defined by (4.15) – (4.17) is stationary and is as in (4.7), (4.8). The proof also gives information about the relaxation time of the system: Very roughly speaking, in an amount of time T the system looses a fraction $1 - c^{T/c_1}$ of its memory. The characterization (4.7), (4.8) can also be used to explicitly find the transient behavior of the process $(Z(t))$ and therefore of the process $(W(t))$, this is work in progress at Bellcore.

A corollary of theorem 1 is that if Z and E are independent and E has the exponential distribution with expected value 1, then

$$Z \text{ and } E + cZ \text{ have the same distribution.} \tag{4.18}$$

Henceforth,

$$F(w,t) = P\{W(t) \leq w\} \tag{4.19}$$

for the "general" process $W(.)$ as in (4.2) – (4.4), and

$$G(z,t) = P\{Z(t) \leq z\} \tag{4.20}$$

for the "special" process $Z(.)$ as in (4.6) –(4.8), and $F(.)$ and $G(.)$ are the corresponding stationary distribution functions. W and Z are random variables with distribution functions F and G. Without loss of generality we assume that

$$Z = \frac{\eta}{m+1} W^{m+1}. \tag{4.21}$$

Hence:

$$F(w) = P\{W \leq w\} = P\{Z \leq \frac{\eta}{m+1} w^{m+1}\} = G(\frac{\eta}{m+1} w^{m+1}). \tag{4.22}$$

$G(.)$ is determined by (4.10), and $F(.)$ is determined by (4.22). In the next section we will discuss the distribution function $G(.)$, its Laplace Transform, and its Moments.

5. Transforms, Distribution Functions and Moments

Let $\phi_Z(.)$ denote the Laplace transform of the random variable Z in theorem 1. Clearly,

$$\phi_Z(s) = \prod_{k=0}^{\infty} \frac{1}{1 + c^k s}. \tag{5.1}$$

It is clear that the infinite product in the RHS of (5.1) converges for all $|c| < 1$ (unless s is of the form $s = -c^{-k}$). At this point it is useful to quote four classical results, three of which are due to Euler. The most important are the first two, both due to Euler: For $|c| < 1$,

$$\prod_{k=0}^{\infty} \frac{1}{1+c^k z} = \sum_{k=0}^{\infty} \frac{(-1)^k z^k}{(1-c)(1-c^2)\dots(1-c^k)}, \tag{5.2}$$

and

$$\prod_{k=0}^{\infty} (1+c^k z) = \sum_{k=0}^{\infty} \frac{c^{\frac{1}{2}(k-1)k} z^k}{(1-c)(1-c^2)\dots(1-c^k)}, \tag{5.3}$$

see Gasper and Rahman (1990), for example formulas (18) and (19) on page xiv, or the results on page 8, for more information on these and similar results. We give simple direct proofs of (5.2) and (5.3) in section 6 of this note. (5.2) gives the moments of the random variable Z. Later in this section we will give other expressions for those moments. (5.3) proves the "left" halves of (2.2), (2.7). The third result, also due to Euler, is (still $|c| < 1$):

$$L(c) = \prod_{k=1}^{\infty} (1-c^k) = \phi_Z(-c) = \sum_{k=-\infty}^{+\infty} (-1)^k c^{\frac{1}{2}k(3k+1)} = 1 - c - c^2 + c^5 + c^7 - c^{12} - c^{15} + \dots\ , \tag{5.4}$$

of which we do not give a proof (see e.g. Hurwitz and Courant (1964) page 207). In a slightly different form, (5.4) is also given in Gasper and Rahman, page xiv formula (15). For numerical purposes, (5.4) converges even faster than (5.3) with $z = -c$. (5.4) proves the rightmost equality in (2.7). The fourth result, which can be useful in evaluating expressions like (2.2), is (still $|c| < 1$):

$$\prod_{k=1}^{\infty} \frac{1-c^{2k}}{1-c^{2k-1}} = \sum_{k=0}^{\infty} c^{\frac{1}{2}k(k+1)} = 1 + c + c^3 + c^6 + c^{10} + \dots\ . \tag{5.5}$$

(5.5) can be obtained from Jacobi's triple product identity, see Gasper and Rahman (1990) page 12. (5.5) proves the rightmost equality in (2.2). (5.1) can be re-written as a partial fraction expansion:

$$\phi_Z(s) = \prod_{k=0}^{\infty} \frac{1}{1+c^k s} = \sum_{k=0}^{\infty} \frac{R_k(c)}{1+c^k s}, \tag{5.6}$$

where the "Residues" $R_k(c)$ are given by

$$R_k(c) = \lim_{s \to -c^{-k}} (1 + c^k s)\phi_Z(s) = \frac{1}{L(c)} \frac{(-1)^k c^{\frac{1}{2}k(k+1)}}{(1-c)(1-c^2)\dots(1-c^k)}, \tag{5.7}$$

where $L(c)$ is as in (5.4). (5.6) is not quite as obvious an identity as seems at first glance: to prove it we observe that the difference between the RHS and the LHS is analytic on the whole complex plane , and also is bounded on the whole complex plane. Liouville's theorem then shows that that difference is a constant function. By letting $s \to \infty$ we see that the constant is zero. We now have four different expressions for $\phi_Z(.)$:

$$\phi_Z(s) = \prod_{k=0}^{\infty} \frac{1}{1 + c^k s} = \sum_{k=0}^{\infty} \frac{(-1)^k s^k}{(1-c)(1-c^2)\dots(1-c^k)} = \sum_{k=0}^{\infty} \frac{R_k(c)}{1 + c^k s} =$$

$$\left(\sum_{k=0}^{\infty} \frac{c^{\frac{1}{2}(k-1)k} s^k}{(1-c)(1-c^2)\dots(1-c^k)} \right)^{-1}. \tag{5.8}$$

The infinite sum in the RHS of (5.6) shows that

$$P\{Z > x\} = \sum_{k=0}^{\infty} R_k(c) exp\{-c^{-k} x\}. \tag{5.9}$$

For $x \downarrow 0$ this probability must go to one: this is indeed guaranteed by (5.6) with $s = 0$, and also by (5.3) with $z = -c$. (5.9) also gives us the probability distribution of the original random variable W:

$$P\{W > w\} = P\{\frac{\eta W^{m+1}}{m+1} > \frac{\eta w^{m+1}}{m+1}\} = \sum_{k=0}^{\infty} R_k(c) exp\{-\frac{c^{-k}\eta}{m+1} w^{m+1}\}, \tag{5.10}$$

and the density is given by

$$(d/dw)P\{W \leq w\} = \sum_{k=0}^{\infty} a_k(c) \eta w^m exp\{-\frac{c^{-k}\eta}{m+1} w^{m+1}\}, \tag{5.11}$$

where

$$a_k(c) = R_k(c) c^{-k} = \frac{1}{L(c)} \frac{(-1)^k c^{\frac{1}{2}(k-1)k}}{(1-c)(1-c^2)\dots(1-c^k)}. \tag{5.12}$$

This completes the proof of (2.5), (2.6).

We are now ready to obtain the moments of W and Z (with distribution functions F and G). We define:

$$\mu(\alpha) = E[W^\alpha],\ \nu(\alpha) = E[Z^\alpha], \tag{5.13}$$

so that

$$\nu(\alpha) = \left(\frac{\eta}{m+1}\right)^\alpha \mu(\alpha(m+1)). \tag{5.14}$$

(5.1) and (5.2) immediately show that for $k \in \{0, 1, 2, \ldots\}$

$$\nu(k) = \frac{k!}{(1-c)(1-c^2)\ldots(1-c^k)}. \tag{5.15}$$

On the other hand, (5.9) shows that for all (non-negative) α

$$\nu(\alpha) = \Gamma(\alpha+1)\sum_{k=0}^{\infty} R_k(c)c^{k\alpha}. \tag{5.16}$$

Using (5.3) with $z = -c^{\alpha+1}$ we see that (5.16) implies that in fact

$$\nu(\alpha) = \Gamma(\alpha+1)\prod_{k=1}^{\infty}\left(\frac{1-c^{\alpha+k}}{1-c^k}\right) = \Gamma(\alpha+1)\frac{\phi_Z(-c)}{\phi_Z(-c^{\alpha+1})}. \tag{5.17}$$

(5.17) is a generalization of (5.15) to values of α outside $\{0, 1, \ldots\}$. For $\alpha \in \{0, 1, \ldots\}$ (5.17) is an alternative proof of (5.15). (5.17) with $\alpha = 1/2$ proves (2.2), and with $\alpha = 1$ it proves (2.3). More interestingly: the middle expression in (5.16) is a meromorphic function of α, so that $\nu(\alpha)$ is finite for all real values of α. (The simple poles of $\Gamma(\alpha+1)$ are eliminated by the simple zeros of $\prod(1-c^{\alpha+k})$). Using a result from Whittaker and Watson (section (12.14)) we get, for $\alpha > 0$:

$$\nu(-\alpha) = \frac{\pi}{\Gamma(\alpha)\sin(\pi\alpha)}\prod_{j=1}^{\infty}\left(\frac{1-c^{-\alpha+j}}{1-c^j}\right). \tag{5.18}$$

Both (5.17) and (5.18) hold for all real α. (5.17) with $\alpha = -1/2$ (or (5.18) with $\alpha = +1/2$) proves (2.8). For $k \in \{1, 2, \ldots\}$ we have

$$\nu(-k) = \frac{(1-c)(1-c^2)\ldots(1-c^{k-1})}{(k-1)!c^{\frac{1}{2}(k-1)k}}\log(\frac{1}{c}) < \infty, \tag{5.19}$$

Since $\nu(\alpha) < \infty$ for all real α we know that for $z \downarrow 0$, $G(z) \downarrow 0$ faster than z^α for every real α. In section 7 we will refine this result.

6. The proofs of (5.2) and (5.3)

This section provides direct proofs of (5.2) and (5.3). Define

$$q_k(c) = \frac{(-1)^k}{(1-c)(1-c^2)\dots(1-c^k)}, \; f(z) = \sum_{k=0}^{\infty} q_k(c) z^k, \tag{6.1}$$

$$r_k(c) = \frac{c^{\frac{1}{2}(k-1)k}}{(1-c)(1-c^2)\dots(1-c^k)}, \; g(z) = \sum_{k=0}^{\infty} r_k(c) z^k. \tag{6.2}$$

We see that

$$(1-c^k) q_k(c) = -q_{k-1}(c). \tag{6.3}$$

Multiplying this with z^k and summing over k yields

$$f(z) = \frac{1}{1+z} f(cz) = \frac{1}{(1+z)(1+cz)} f(c^2 z) = \dots\, , \tag{6.4}$$

which (with $f(0) = 1$) proves (5.2). Similarly, we have

$$(1-c^k) r_k(c) = c^{k-1} r_{k-1}(c). \tag{6.5}$$

Multiplying this with z^k and summing over k yields

$$g(z) = (1+z) g(cz) = (1+z)(1+cz) g(c^2 z) = \dots\, . \tag{6.6}$$

This completes the direct proof of (5.3).

7. $P\{Z \leq x\}$ for x small

For small window sizes the dynamic becomes different than described in the paragraphs preceding (1.1).

One factor is that when the congestion window is less than four MSSs, a single lost packet leads to the situation that no three duplicate acknowledgements will accumulate, so that the TCP source goes into time-out. This considerably changes the throughput behavior. As long as we only look at arrival epochs of "good" (i.e. non-duplicate) acknowledgements this complication does not greatly change the dynamic of the process (W_n): if the congestion window is four or less, and a packet is lost, then after the next good acknowledgement the congestion window is 2 (MSSs). (Again, use of the appropriate subjective time saves the day!). In this case, a more complicated sampling bias must be used to translate the "arrival epoch" stationary distribution into a "continuous time" stationary distribution and into a prediction of throughput.

Another factor is that when the congestion window is small the approximation (1.6) becomes questionable.

Based on these insights we expect that the "square root law" we established will hold best if we look at the system at epochs just after "good acknowledgements", hold almost (but not quite) equally well if we look at the system at "arbitrary packet times" (after the definition of "congestion window" has been fixed during fast recovery), and holds less well or not at all if we look in "clock-time". The last is for two reasons: Round Trip Times can depend on the congestion window, and time-outs can become important.

For these reasons it is desirable to have some assurance that for p reasonably small small values of W_n are unlikely, or even extremely unlikely. This assurance can indeed be given. We will derive upper bounds for $P\{Z < x\}$ for x positive but small. The reader can translate these (using (4.6) and (1.5)) into "approximate upper bounds" for $P\{W_n < w\}$ for w positive and small compared with $p^{+\frac{1}{2}}$.

In section 5 we already saw that

$$\lim_{x \downarrow 0} x^{\alpha} P\{Z \leq x\} = 0 \text{ for all real } \alpha. \tag{7.1}$$

In this section we give a bound for the actual rate.

Using the idea of the Bienaimee-Chebychev inequality we see that for $\alpha > 0$, $x > 0$:

$$\nu(-\alpha) = \int_0^\infty y^{-\alpha} dG(y) > \int_0^x y^{-\alpha} dG(y) > x^{-\alpha} G(x), \tag{7.2}$$

and hence

$$P\{Z \leq x\} < x^\alpha \nu(-\alpha) = \frac{\pi x^\alpha}{\Gamma(\alpha)\sin(\pi\alpha)} \prod_{j=1}^{\infty} \left(\frac{1 - c^{-\alpha+j}}{1 - c^j} \right). \tag{7.3}$$

(7.3) uses (5.18). For the sharpest possible bound, we should find the best possible α as function of x. This seems hard. Quite a good bound is obtained by choosing α integer, $\alpha = K + 1$. (7.3) then reads:

$$P\{Z \leq x\} < \frac{\left(xc^{-\frac{1}{2}K}\right)^{K+1}}{K!} \log(\frac{1}{c}) \prod_{j=1}^{K} (1 - c^j). \tag{7.4}$$

In (7.13), the choice $K = K(x)$ defined by

$$K(x) = \text{integer part of } \frac{\log(x)}{\log(c)}, \text{i.e. } c^{K(x)+1} < x \leq c^{K(x)} \tag{7.5}$$

gives for $0 < x < c < 1$

$$P\{Z < x\} < \frac{x^{\frac{1}{2}\frac{\log(x)}{\log(c)}}}{\Gamma(\frac{\log(x)}{\log(c)})} \log(\frac{1}{c}). \tag{7.6}$$

A value of α somewhat larger than $K(x) + 1$ may very well give a somewhat better (i.e. lower) upper bound.

References

[1] Mathis, M., Semke, J. Mahdavi, J. and Ott, T.J. (1997) The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm. *Computer Communications Review 27* (3), pp 67 - 82 (July 1997).

[2] Ott, T.J., Kemperman, J.H.B., and Mathis, M. (1996) The Stationary Behavior of Idealized TCP Congestion Behavior.
ftp://ftp.bellcore.com/pub/tjo/TCPwindow.ps

[3] Van Jacobson. Congestion avoidance and control, *Proceedings of ACM SIGCOMM '88*, August 1988.

[4] W. Stevens, *TCP/IP Illustrated*, volume 1, Addison-Wesley, Reading MA, 1994.

[5] Kevin Fall and Sally Floyd, Simulations-based comparisons of tahoe, reno and SACK TCP, *Proceedings of ACM SIGCOMM '96*, May 1996.

[6] Janey C. Hoe, Improving the start-up behavior of a congestion control scheme for TCP, *Proceedings of ACM SIGCOMM '96*, August 1996.

[7] Lawrence S. Brakmo, Sean W. O'Malley, and Larry L. Peterson, TCP vegas: New techniques for congestion detection and avoidance, *Proceedings of ACM SIGCOMM '94*, August 1994.

[8] Lawrence S. Brakmo and Larry L. Peterson, Performance problems in BSD4.4 TCP, *Proceedings of ACM SIGCOMM '95*, October 1995.

[9] Sally Floyd, Connections with Multiple Congested Gateways in Packet-Switched Networks Part I: One Way Traffic. *CCR* **21** no 5 pp 30 - 47.

[10] Sally Floyd, TCP and successive fast retransmits, February 1995, Obtain via ftp://ftp.ee.lbl.gov/papers/fastretrans.ps.

[11] Sally Floyd and Van Jacobson, Random early detection gateways for congestion avoidance, *IEEE/ACM Transactions on Networking*, August 1993.

[12] Sally Floyd, TCP and explicit congestion notification, *ACM CCR*, 24(5), October 1994.

[13] Matthew Mathis, Jamshid Mahdavi, Sally Floyd, and Allyn Romanow, TCP selective acknowledgement options, May 1996, Internet Draft ("work in progress") draft-ietf-tcplw-sack-02.txt.

[14] Matthew Mathis, Jamshid Mahdavi, Forward Acknowledgment: Refining TCP Congestion Control, *Proceedings of ACM SIGCOMM '96*, August 1996.

[15] Lakshman, T.V., and Madhow, U. (1997) The Performance of TCP/IP for Networks with high Bandwidth-Delay products and random loss. *Trans of Netw* 1997.

[16] Lakshman, T.V., Madhow, U. and Suter, B. (1997) Window-based error recovery and flow control with a slow acknowledgement channel: a study of TCP/IP performance *Infocom '97*.

[17] Jessen, B. and Wintner, A (1935) Distribution Functions and the Riemann Zeta Function, *Trans of the AMS* **38** pp 48–88.

[18] Gasper, G. and Rahman, M. (1990) *Basic Hypergeometric Series*, Encyclopedia of Mathematics and its Applications, Cambridge University Press.

[19] Hurwitz, A. and Courant, R. (1964) Vorlesungen ueber allgemeine Funktionentheorie und elliptische Funktionen, Springer Verlag (4-th edition).

[20] Loeve, M. (1955) *Probability Theory*, Van Nostrand.

[21] Stevens, W.R. (1994) *TCP/IP Illustrated*, Volumes I and II. Addison Wesley.

[22] Titchmarsh, E.C. (1932) *The Theory of Functions*, Oxford University Press.