ECN Protocols and the TCP Paradigm

Teunis J. Ott

June 30, 1999

Abstract

In TCP, packet loss is is used by users of the network to gauge congestion, and to set congestion windows. The fact that individual "signals" (packet losses) have other effects than transmitting information about congestion makes it desirable to keep the amount of information thus signaled quite low.

ECN (Explicit Congestion Notification, see e.g. [5]) makes it possible for Routers to give Endstations (including sources) more detailed information about congestion and (e.g.) about desirable congestion window sizes. This note describes and analyzes a class of protocols using this opportunity.

1 Introduction

Consider a flow of packets in IP with per packet acknowledgement. Assume the system allows " Explicit Congestion Notification" (ECN, see e.g. [4], [5]): When a router recognizes one of its buffers is getting close to congestion, it can set a "Congestion Indicator Bit" in packets flowing through this buffer. To avoid confusion with the similarly named bit in ATM with ABR, Floyd and Ramakrishnan call this the "Congestion Experienced"

(CE) bit. In this note we use that name (CE), but make different use of the bit. This setting of the CE bit can be done probabilistically, with a state–dependent probability p. The destination copies those CE bits into the ECN–Echo bit in acknowledgements. Thus, the source is informed of congestion. [4] and [5] discuss a number of of implementation issues, such a location of the bit and what to do when there is delayed acknowledgements, etc.

When the router sets the CE bit in a packet, we also say it marks the packet. The router can (for example) choose a state–dependent probability p and mark packets with probability p.

[4] and [5] repeatedly state the opinion that the source of traffic must react to a returning ECN–Echo bit that has been set ($= 1$) in (almost) exactly the same way a TCP source reacts to discovering a "congestion event" that includes loss of at least one packet.

This is an opinion the author of this note does not share.

Among the advantages of marking in stead of dropping (as in RED) are (i) that no retransmission is needed and therefore (ii) any marking probability $0 \leq p \leq 1$ is acceptable. Drop probabilities have to be at most not much more than .1, or (for example) TCP stops functioning. In addition, modern traffic endpoints (sources, destinations) have the ability to interprete the meaning of a stream of ECN–Echo bits quite carefully, and for example as a function of the type or class of service the packets belong to. The author of this note advocates that ECN–capable flows react to ECN–Echo bits in ways that still need to be defined and that may be quite different from the way a non ECN–capable flow reacts to dropped packets. Since fairness requires that if the source behavior is changed, also the router behavior must change, in a router the marking probability for ECN–capable flows similarly may or even must be different from the dropping probability for ECN non–capable flows. Even, the marking probability for ECN–capable flows may depend on the type (type of service, priority class, etc.) of the packet. Thus, the router has a number

Class	marking	dropping
Non–ECN	0	$p_{n,d}$
ECN Class 0	$p_{0,m}$	$p_{0,d}$
⋮	⋮	⋮
ECN Class L	$p_{L,m}$	$p_{L,d}$

Table 1: Marking and Dropping Probabilities

of "signaling congestion" parameters: The drop probability for ECN non–capable flows; the drop probability for ECN–capable flows (might be class dependent); and the marking probability for ECN–capable flows (might be class dependent). For example, for RSVP flows with a guaranteed rate (and that stay within their rate) it makes no sense to drop unless there is no choice, and it makes no sense to mark apart from as a warning signal that involuntary drop is imminent. Similarly, if in IP there is an option of distinguishing between "In–Rate" and "Out–of–Rate" packets (or "in–Profile" versus "Out–of–Profile" packets, using an "in–Rate" bit), "In–Rate" packets would be dropped only involuntarily, and would be marked only as a warning that involuntary drop is imminent. "Out–of–Rate" packets could be dropped as well as marked, with marking of course the prefered option. This implies that the signalling (e.g. carried by acknowledgements) from destination to source must separately signal markings of In–Rate and of Out–of–Rate packets. This router behavior is illustrated in Table 1.

In the foreseeable future, ECN–capable routers would set drop probabilities for ECN non–capable flows in a way consistent with RED or SRED or some such mechanism. ECN capable flows would react to drop in essentially the same way as ECN non–capable flows. This is necessary, because for some time there would be ECN–capable as well as ECN non–capable routers. Some changes would be allowed for special classes (say flows paying a "premium" tariff). ECN–capable routers could set the marking probability for

ECN–capable flows in just about any way, as long as Router Behavior and Flow Behavior are designed together.

Making sure that those new (marking) behaviors have been studied and implemented in (some) routers before many end–stations become ECN capable may very well be the only way to make an elegant transition to a new environment.

Once ECN is ubiquitous, at least in routers, (endstations may take longer!) the meaning of drop for ECN–capable flows in principle could change. This will be a hard transition to make and may very well be impossible, because by that time there will be many ECN–capable endstations that can not make a synchronized change.

By the same argument, the advent of ECN is an opportunity, quite likely the last opportunity, to modify the congestion algorithms in IP. We should use this opportunity to study Router Behavior (whether and when to mark packets) and Source Behavior (how to react to marked and unmarked packets) as two aspects of the same problem. The advent of ECN temporarily gives us a clean slate that we can fill in with new mechanisms, using what we have learned in the past 20+ years, and taking into consideration the greater capabilities in modern endstations, and the much higher bandwidths in the modern Internet. Since this is likely the last opportunity to do a significant amount of redesign of the control mechanisms, it is important to use the opportunity well!

In this note we study the consequences for traffic characteristics of certain "source behaviors", i.e. for certain ways for sources to react to packet marking probabilities. Later, we will use the insight gained to propose router behaviors, i.e. ways for routers to decide whether and when to mark packets.

In this note we do not yet consider class dependent marking. In particular we do not yet consider separate marking policies for In–Rate and Out–of–Rate packets. Mechanisms where "entry-routers" mark packets as either In–Rate or Out–of–Rate, where other routers

have different marking policies for such packets, and where the reaction of sources to marked packets depends on whether the packets were In–Rate or Out–of–Rate, look like a very promising area of future research.

The analyses done in this note are examples of the analyses that need to be done before a newly designed ECN can be finalized. There is no pretension of having achieved closure. The most important recommendation is that Router behavior and Source behavior must be designed together and must be class dependent, i.e. depend on the class of the packet.

2 The TCP Paradigm with general increases and decreases

Let us consider a Congestion Window based protocol where whenever an acknowledgement arrives at the source that acknowledges an unmarked data packet while the congestion window is W, then the congestion window increases by $incr(W)$, and when a marked packet is acknowledged the congestion window decreases by $decr(W)$. (Also, when a packet is marked while the W is small there will be a time-out, with time–outs probably increasing exponentially after repeated such markings. We do not consider such details in this note). At this point it does not matter whether W is expressed in bytes or in MSSs or in some other entity. Assuming packets are marked with probability p, and that "locally in time" p is constant, the drift per packet in the congestion window while the congestion window is W is

$$E[W_{n+1} - W_n | W_n = W] = drift(W,p) = (1-p).incr(W) - p.decr(W) = p.incr(W).\left(\frac{1-p}{p} - \frac{decr(W)}{incr(W)}\right). \quad (2.1)$$

Let us now consider the function

$$q(W) = \frac{decr(W)}{incr(W)}. \tag{2.2}$$

Assuming this function is reasonably smooth, and that p indeed is constant during a large number of packets sent, the congestion window will tend to spend most of its time at W values for which $q(W)$ is not too far from $\frac{1-p}{p}$.

It therefore is highly desirable that $q(.)$ is a strictly increasing function with

$$q(1) = 0, \quad \lim_{W \to \infty} q(W) = \infty. \tag{2.3}$$

In that case, if the marking probability p is constant, $q(W)$ will fluctuate around $\frac{1-p}{p}$. Thus we can predict the average window size: find $W(p)$ with

$$q(W(p)) = \frac{1-p}{p}. \tag{2.4}$$

Equation (2.3) ensures that (possibly with interpolation or rounding to an integer) there always is a solution to (2.4). The actual window size will tend to fluctuate around $W(p)$. We use $W(p)$ as approximation for the average. The functions $q(W)$ and $W(p)$ really are response surfaces of the sources to Router Behavior.

It is useful to note here a connection with "fairness": In the situation of (2.3), if two different flows that react in the same way to packet markings encounter the same marking probability p, they will tend to have the same (average) congestion windows.

Note that if the function $q(.)$ is almost constant over a long range of W values, we do not have fairness: if for a while $\frac{1-p}{p}$ happens to remain constant, equal to that $q(W)$ value, and two flows start with different W values in that range of W values, they will tend to keep their different congestion windows for a long time.

In the special case of TCP (without delayed acknowledgements, and dropping in stead of marking) we have (congestion windows are now measured in MSSs):

$$incr(W) = \frac{1}{W},\ decr(W) = \frac{W}{2}, \tag{2.5}$$

and thus

$$q(W) = \frac{W^2}{2},\ W(p) = \sqrt{\frac{2(1-p)}{p}}. \tag{2.6}$$

This is the basis for the celebrated square root formula, see e.g. [2].

The function $q(.)$ defines a "response surface" of W to p. We see that we can first choose $q(.)$ arbitrarily (subject to the monotonicity and (2.3)), and then still can choose $incr(.)$ and $decr(.)$ somewhat arbitrarily. Thus we have a choice of two modi operandi: we can choose $incr(.)$ and $decr(.)$ functions and find out what the resulting response surface is, or we can choose a response surface and then (with that degree of freedom gone) find $incr(.)$ and $decr(.)$ functions.

If we decide to first choose the response surface $q(W)$, (or $W(p)$), we can go one step further: after choosing this response surface we do not choose $incr(.)$ and $decr(.)$, but we let the source directly estimate the marking probability p, and adjust the congestion window W accordingly. This method has interesting consequences, that will be investigated in Sections 7 and 8. In fact, under certain circumstances it leads back to a system with $incr(.)$ and $decr(.)$ functions as above. Thus, it can be seen as a somewhat scientific way of choosing such functions.

For TCP, the desire to have multiplicative decrease governed the choice of $decr(.)$. This led to the somewhat unfortunate result that the congestion window occasionally halves. For connections with a long round trip time that therefore need a high congestion window this is a problem: after a halving of the congestion window, the congestion window increases quite slowly. This has a number of unfortunate consequences. One of these is that TCP traffic is not very good "background" traffic for other streams.

Choosing both $decr(W)$ and $incr(W)$ much smaller (but leaving the quotient the same) has the advantage that a flow following that behavior has the same average W value but behaves much more smoothly. There is the disadvantage that the flow reacts quite slowly to changing p.

It must be noted that not all versions of TCP are consistent in using multiplicative decrease. Some versions of TCP allow the window to be halved only once in a round trip time or only once per congestion episode. It must be noted that if we were to use $decr(W) = 1$ (MSS) in a scheme with marking instead of dropping, setting $p = 1$ during one round trip time closes all windows (reduces all windows to 1 MSS or less, time–out counting as a congestion window of less than one MSS): more draconic than halving all windows! Thus, because a marked packet is not lost (with all the unpleasant consequences of losing a packet), it is possible to send a strong signal by marking many packets in a short period. Choosing $decr(W) = 1$ also has the effect that acknowledgement of a marked packet does not cause transmission of a new packet. Thus, the router can fairly accurately predict the consequences of marking a packet. With multiplicative decrease, the router needs to know the window size to predict the consequences of marking a packet.

3 Outside the TCP Paradigm

In the previous section we had, as in TCP, $incr(.)$ and $decr(.)$ functions, and (in principle) a congestion window modification is made every time the source receives an acknowledgement, and the update uses the $incr(.)$ and $decr(.)$ functions. We call this situation the "TCP Paradigm". There are of course more general mechanisms. For example, the source could count "marked" and "unmarked" acknowledgements, and every now and then (say once every Round Trip Time RTT) update the congestion window. Such mechanisms are outside the TCP paradigm.

Mechanisms "on the boundary of" the TCP paradigm are for example those discussed in Section 7, where a "response surface" $q(.)$, $W(.)$ as in (2.3), (2.4) has been chosen, an estimate $\overline{p}$ for the marking probability p is maintained, and the congestion window actually used is $W(\overline{p})$. Such a scheme was already referred to in Section 2. In Section 8 a very interesting such mechanism will be described, of which we then find that it can be implemented in two different ways: one mechanism inside the TCP paradigm, the other outside. This can be seen as a "scientific" way of choosing $incr(.)$, $decr(.)$.

More research is needed to check whether there are pairs of router behavior– endstation behavior outside the TCP paradigm that are superior to all mechanisms inside the TCP paradigm.

4 A special class of $incr(.)$ and $decr(.)$ functions

In the remainder of this note we restrict ourselves to $incr(.)$ and $decr(.)$ of the form

$$incr(w) = c_1 w^{\alpha},\ decr(w) = c_2 w^{\beta}. \tag{4.1}$$

For these functions to make sense we obviously want $\alpha < 1,\ c_1 > 0$ and $\beta \leq 1,\ c_2 > 0$, and if $\beta = 1$ we clearly need $0 < c_2 < 1$. The reader who worries about implementability will be relieved to hear that at the end we conclude that (probably) optimal parameter values are $\alpha = 0$, $\beta = 1$, and that rules on how to choose c_1 and c_2 will be given. Until that point we consider general α and β etc. Many of the results in this section can be extended to more general $incr(.)$ and $decr(.)$ functions. At this point that is an exercise of limited interest.

With this choice of functions as in (4.1) we get the response surface

$$q(w) = \frac{decr(w)}{incr(w)} = \frac{c_2}{c_1} w^{\beta-\alpha}. \tag{4.2}$$

Thus, for packet marking probability p constant, and marking independent from packet to packet, when transporting a very big file the congestion window will tend to fluctuate around $w(p)$, defined as

$$w(p) = \left(\frac{c_1}{c_2}\frac{1-p}{p}\right)^{\frac{1}{\beta-\alpha}}. \tag{4.3}$$

When p is quite close to zero, (4.3) becomes

$$w(p) = \left(\frac{c_1}{c_2 p}\right)^{\frac{1}{\beta-\alpha}}. \tag{4.4}$$

If we choose that "more marking signals more congestion" we must have $\alpha < \beta \leq 1$.

In classical TCP we had $c_1 = 1,\ \alpha = -1,\ c_2 = \frac{1}{2},\ \beta = 1$.

In order to study behavior of the functions above with constant marking probability p, we study the evolution of W_n as a stochastic process. Using the same ideas as in [2] we get the following results:

Theorem 1. If

$$\alpha < \beta = 1,\ c_1 > 0,\ 0 < c_2 < 1, \tag{4.5}$$

then for $p \downarrow 0$ the process $(X(t))_{0 \leq t < \infty}$ defined by

$$X(t) = p\left(W_{\lfloor \frac{t}{p} \rfloor}\right)^{1-\alpha} \tag{4.6}$$

behaves as follows: there is a Poisson Process with intensity 1. In–between the points of the Poisson Process,

$$\frac{d}{dt}X(t) = c_1(1-\alpha), \tag{4.7}$$

and in the points of the Poisson Process (say point τ) we have

$$X(\tau^+) = (1-c_2)^{1-\alpha}X(\tau^-). \tag{4.8}$$

Hence, the stationary distribution of the process $X(.)$ is of the form

$$X = c_1(1-\alpha)Z, \tag{4.9}$$

where Z is a random variable the distribution of which does not depend on p or on c_1. Z has the form

$$Z = \sum_{k=0}^{\infty} (1 - c_2)^{k(1-\alpha)} E_k, \tag{4.10}$$

where $(E_K)_{k=0}^{\infty}$ are independent, identically distributed random variables, all exponentially distributed with expected value 1. The distribution of this infinite sum of random variables, including all its moments, was described in detail in [2]. For example, setting $c = (1 - c_2)^{(1-\alpha)}$ we have for all real μ

$$E[Z^{\mu}] = \Gamma(\mu + 1) \prod_{k=1}^{\infty} \left(\frac{1 - c^{\mu+k}}{1 - c^k} \right). \tag{4.11}$$

In particular (as is easier seen directly!)

$$E[Z] = \frac{1}{1 - c}, \; Var(Z) = \frac{1}{1 - c^2}, \tag{4.12}$$

and therefore

$$Coeff.Var(Z) = \frac{st.dev(Z)}{E[Z]} = \sqrt{\frac{1 - c}{1 + c}}. \tag{4.13}$$

Thus, we know that the stationary distribution of the congestion window has the form

$$W = p^{-\frac{1}{1-\alpha}} \left(c_1 (1 - \alpha) Z \right)^{\frac{1}{1-\alpha}}. \tag{4.14}$$

That stationary distribution, including all its moments, therefore is explicitly known. Among other results, we have

$$coeff.var(W) = \frac{st.dev(W)}{E[W]} \sim coeff.var(Z^{\frac{1}{1-\alpha}}), \tag{4.15}$$

independent of p and c_1.

This leads to a clean expression for $Coeff.Var(W)$ only if $\alpha = -1$ or $\alpha = 0$. If $\alpha = 0$ we get

$$Coeff.Var(W) \sim \sqrt{\frac{c_2}{2 - c_2}}. \tag{4.16}$$

The proof of Theorem 1, and of the corollaries above, is left as an exercise for the reader. Hint: copy the corresponding proofs in [2].

Theorem 2. If

$$\alpha < \beta < 1,\ c_1 > 0,\ c_2 > 0, \tag{4.17}$$

then for $p \downarrow 0$ the process $(X(t))_{0 \le t < \infty}$ defined by

$$X(t) = p^{\nu_1} \left(W_{\lfloor \frac{t}{p^{\nu_2}} \rfloor} - \left(\frac{c_1(1-p)}{c_2 p} \right)^{\frac{1}{\beta - \alpha}} \right), \tag{4.18}$$

with

$$\nu_1 = \frac{(1+\beta)}{2(\beta - \alpha)},\ \nu_2 = \frac{1-\alpha}{\beta - \alpha}, \tag{4.19}$$

becomes the Ornstein–Uhlenbeck process with local drift

$$E[X(t+\Delta) - X(t) | X(t) = x] = -\Delta . x . (\beta - \alpha) c_1^{-\frac{1-\beta}{\beta-\alpha}} c_2^{\frac{1-\alpha}{\beta-\alpha}} + o(\Delta)\ (\Delta \downarrow 0), \tag{4.20}$$

and local dispersion

$$Var(X(t+\Delta) | X(t) = x] = \Delta . c_1^{\frac{2\beta}{\beta-\alpha}} c_2^{-\frac{2\alpha}{\beta-\alpha}} + o(\Delta)\ (\Delta \downarrow 0). \tag{4.21}$$

Thus for $p \downarrow 0$ the stationary distribution of $X(t)$ becomes the normal distribution with expected value zero and variance

$$Var(X) = \frac{c_1^{\frac{1+\beta}{\beta-\alpha}} c_2^{-\frac{1+\alpha}{\beta-\alpha}}}{2(\beta - \alpha)}. \tag{4.22}$$

This theorem will be proven in Appendix A.

The stationary distribution of $X(.)$ immediately translates into a stationary distribution for W_n: For $p \downarrow 0$, the stationary distribution of W_n has

$$E[W] \sim \left(\frac{c_1(1-p)}{c_2 p} \right)^{\frac{1}{\beta - \alpha}} \sim \left(\frac{c_1}{c_2 p} \right)^{\frac{1}{\beta - \alpha}},$$

$$\text{st.dev}(W) \sim \frac{c_1^{\frac{1+\beta}{2(\beta-\alpha)}} c_2^{-\frac{1+\alpha}{2(\beta-\alpha)}}}{\sqrt{2(\beta-\alpha)}} p^{-\frac{1+\beta}{2(\beta-\alpha)}}, \tag{4.23}$$

$$coeff.var(W) = \frac{st.dev(W)}{E[W]} \sim \frac{c_1^{-\frac{1-\beta}{2(\beta-\alpha)}} c_2^{\frac{1-\alpha}{2(\beta-\alpha)}}}{\sqrt{2(\beta-\alpha)}} p^{\frac{1-\beta}{2(\beta-\alpha)}}. \tag{4.24}$$

(4.15) and (4.24) show that there is a certain charm to choosing $\beta = 1$: with that choice, and that choice only, the coefficient of variation of W becomes independent of p for $p \downarrow 0$ (i.e. when the congestion window is allowed to be very large). In fact, for that choice the distribution of $\left(\frac{p}{c_1}\right)^{\frac{1}{1-\alpha}} W$ becomes independent of p and c_1 for p small. The non–dependence on p may seem no big deal, but non–dependence on p implies non–dependence on the average value of W: scale invariance!

Any smaller value of β makes the (stationary) window size almost deterministic when $p \downarrow 0$ (when the average value of W becomes large).

In the next section we will draw some conclusions about the number of marked packets in a flow per Round Trip Time (RTT).

5 The Number of Marked Packets per Round Trip Time

If a flow has a marking probability of p per packet and a congestion window of W packets, it will in average see pW marked packets per Round Trip Time. In the situation of Section 4, when a flow has been in existence long enough to have reached stationarity, this gives us the following results:

Theorem 3. In the situation of theorem 1 ($\beta = 1$), when a flow is in existence for a long time and the marking probability p is constant and close to zero, it has in average

about

$$p^{-\frac{\alpha}{1-\alpha}}(c_1(1-\alpha))^{\frac{1}{1-\alpha}}E[Z^{\frac{1}{1-\alpha}}] \tag{5.1}$$

marked packets per Round Trip Time. The distribution of Z depends only on c_2 and on α, not on p.

Theorem 4. in the situation of theorem 2 ($\beta < 1$), when a flow is in existence for a long time and the marking probability p is constant and close to zero, it has in average about

$$p^{\frac{\beta-\alpha-1}{\beta-\alpha}}\left(\frac{c_1}{c_2}\right)^{\frac{1}{\beta-\alpha}} \tag{5.2}$$

marked packets per Round Trip Time. We see that in both cases there is a factor $p^{\frac{\beta-\alpha-1}{\beta-\alpha}}$. We see that it is highly desirable that

$$\beta - \alpha \leq 1. \tag{5.3}$$

Namely, in that case, the number of marked packets per Round Trip Time will not go to zero when $p \downarrow 0$, at least as long as the flow is allowed very large windows (as large as the "response surfaces" permit). That way, the router can signal relatively subtle changes in desired rates. When the number of marked packets per Round Trip Time falls (significantly) below 1, it becomes hard or impossible for routers to signal a desired minor change in congestion window. Classical TCP is an extreme case, with only in the order of $\sqrt{p}$, i.e. much fewer than 1, "marked" packets per Round Trip Time (if the reader prefers it, we can restate this as about one "marked" packet per $p^{-\frac{1}{2}}$, i.e. many, RTTs).

A sensible choice seems to be

$$\beta - \alpha = 1, \tag{5.4}$$

and either

$$\beta = 1,\ \alpha = 0,\ c_1 E[Z] = \frac{c_1}{c_2} = O(1) \tag{5.5}$$

($E[Z] = \frac{1}{c_2}$ because $\alpha = 0$) or

$$\beta < 1,\ \alpha = \beta - 1,\ \frac{c_1}{c_2} = O(1) \tag{5.6}$$

The $O(1)$ in (5.5) and (5.6) probably should be chosen in the range .2 (in average one marked packets per 5 RTTs) to 5 (in average 5 marked packets per RTT). Research on the appropriate optimality criteria is desirable. It must be noted that in both cases above, more generally as long as (5.4) holds, the response surface has the form

$$q(w) = \frac{c_2}{c_1} w,\ w(p) = \frac{c_1}{c_2} \frac{1-p}{p}. \tag{5.7}$$

This "ideal" response surface will be seen in Section 8 to have an interesting and attractive additional characteristic.

In the situations of (5.5) or (5.6), the source gets, when the marking probability is low and stationarity has been achieved, a low but sufficient number (say $\frac{1}{5}$ to 5) of marked packets per Round Trip Time. This enables the routers to give adequate signals without the need on the part of the source to make strong changes.

From the point of view of easy implementability, the first choices to be considered are ($\beta = 1,\ \alpha = 0$) and ($\beta = 0,\ \alpha = -1$).

It must be noted that the first option ($\beta = 1,\ \alpha = 0$) is what happens during slowstart in classical TCP, and always in those versions of TCP that have the famous bug mentioned in [12], page 977, but with different c_1 and c_2.

From the point of view of having "enough" marked packets to transmit detailed information, it is attractive to also consider the case $0 < \beta - \alpha < 1$. If a convenient implementation can be designed that mimmicks such behavior without causing punitive computational overhead, it becomes attractive to at least study such schemes.

6 Relaxation Times

There are two other, related, criteria in choosing a feedback mechanism. We want a feedback mechanism that, if p remains constant and once stationarity has been reached, has fairly constant congestion windows. We also want a mechanism where if p changes the congestion window quickly moves to the new equilibrium value. The two desires are somewhat contradictory.

Large β and α cause very quick adjustments when p changes, but also relatively wild swings while p is constant. Small β and α cause very "smooth" behavior while p is constant, but make the system adapt relatively slowly to changing p.

We already saw that even if β is as low as 0, a router can reduce all windows to 1 (or "less": time–out) by setting $p = 1$ for ${c_2}^{-1}$ Round Trip Time. This should be adequate in all (?) circumstances. For larger values of β the downward adjustment in the congestion window is more rigorous, but also reaction to randomly marked packets is wilder.

For upward adjustments in the congestion window it similarly seems desirable that $\alpha \geq 0$.

Together with the material in the previous section, this makes the choice

$$\alpha = 0,\ \beta = 1 \tag{6.1}$$

seem more and more desirable. Choices of c_1 and c_2 then must be used to take the sharp edges off this behavior. (4.16) shows that in that case c_2 controls the coefficient of variation of the stationary distribution of the congestion window. A value $c_2 = \frac{1}{8}$ or $\frac{1}{16}$ looks extremely plausible.

In addition there are implementation details that can be used to further smooth behavior without decreasing the rate of adjustment to changing circumstances, see Section 10. Section 8 will give another argument in favor of the choice (6.1).

7 First estimating p

With more powerful endstations it is possible to have a more sophisticated algorithm. The main proposal in this section is to first choose a response function $q(.)$ and then explicitly estimate p. The desired W is computed from the estimated p using (2.4). p could be estimated using exponential smoothing, but there may be problems doing this: Let

$$Zap(k) = \begin{cases} 0 \text{ if the } k-\text{th packet is unmarked} \\ 1 \text{ if the } k-\text{th packet is marked} \end{cases} \tag{7.1}$$

and let

$$\overline{p}_k = (1-r)\overline{p}_{k-1} + r\,Zap(k). \tag{7.2}$$

be the estimate for p. (7.2) has the disadvantage that when the estimate $\overline{p}$ is small compared with the smoothing parameter r, a single "zapped" (i.e. marked) packet increases $\overline{p}$ far too much. It is desirable to let r depend on $\overline{p}$, for example a well chosen positive constant times $\overline{p}$. However, this may lead to problems when $\overline{p}$ becomes extremely small. A comprehensive solution seems to be:

Choose a minimal value for $\overline{p}$. For example, choose a maximal acceptable value W_{max} for the congestion window W (say the receive window). From the chosen response surface $q(.)$, compute p^* such, that $W_{max} = W(p^*)$. Now choose p_{min} "appropriately small" (to be defined) compared with p^*. Choose a positive constant c_3, $0 < c_3 < 1$, for example $c_3 = \frac{1}{8}$ or $\frac{1}{16}$. Now, instead of (7.2) use

$$\overline{p}_k = max\left((1 - c_3\overline{p}_{k-1})\overline{p}_{k-1} + c_3\overline{p}_{k-1}Zap(k),\ p_{min}\right). \tag{7.3}$$

This way, when $\overline{p}$ is small, it takes in the order of $(\log(1+c_3))^{-1}$ (log base 2) marked packets in relatively quick succession (much faster than probability $\overline{p}$ per packet) to double the value of $\overline{p}$.

Every time $\overline{p}$ has been re–computed, recompute W from

$$W = min\left(W(\overline{p}), W_{max}\right). \tag{7.4}$$

Thus, as long as $p_{min} \leq \overline{p} \leq p*$, W remains at W_{max}. When $\overline{p}$ increases above $p*$, W decreases below W_{max}. As long as $\overline{p}$ remains below p^*, randomly marked packets do not affect W. It seems to make sense to choose $p_{min} = \frac{p*}{2}$. In that situation about $(\log(1+c_3))^{-1}$ marked packets in quick succession always start decreasing W.

8 Estimating p with the "Ideal" Response Function

In this section we choose the response function as in (5.7):

$$q(W) = \frac{W}{c_4}, \tag{8.1}$$

where $c_4 = \frac{c_1}{c_2}$. c_1 and c_2 no longer have meaning by themselves. Window evolution is done as in Section 7, with parameter c_3. We replace (approximate) the response surface by

$$W(p) = \frac{c_4}{p}. \tag{8.2}$$

Next we analyze the evolution of W in the domain where $W < W_{max}$, $p* < \overline{p}$. In other words, we always have

$$W_k = \frac{c_4}{\overline{p}_k}. \tag{8.3}$$

Since we have

$$\overline{p}_k = \begin{cases} (1 - c_3\overline{p}_{k-1})\overline{p}_{k-1} & \text{if } Zap(k) = 0, \\ (1 + c_3 - c_3\overline{p}_{k-1})\overline{p}_{k-1} & \text{if } Zap(k) = 1, \end{cases} \tag{8.4}$$

we also have

$$W_k - W_{k-1} = \begin{cases} \frac{c_3 c_4}{1 - c_3\overline{p}_{k-1}} & \text{if } Zap(k) = 0, \\ -\frac{c_3(1-\overline{p}_{k-1})}{1+c_3(1-\overline{p}_{k-1})} W_{k-1} & \text{if } Zap(k) = 1, \end{cases} \tag{8.5}$$

Thus, we see that for $p* < \overline{p} << 1$ the evolution of W is as in Section 4 with $\alpha = 0,\ \beta = 1,\ c_1 = c_3 c_4$, and $c_2 = \frac{c_3}{c_3+1}$.

Since $\beta = 1,\ \alpha = 0,\ \frac{c_1}{c_2} = \frac{c_4}{c_3+1}$ is the desired number of marked packets per Round Trip Time (once stationarity has been reached). For every marked packet, the congestion window is decreased from W to $\frac{W}{c_3+1}$. If $c_3 = 1$ every marked packet halves the window. A less draconic choice is $c_2 = \frac{1}{8}$ or even $\frac{1}{16}$. There now are two trains of thought that can be used to set c_2 or c_3: the one based on how fast the estimate for p is changing when there are marked packets, and the one based directly on how fast the congestion window must change when packets are marked.

9 Router Behavior

This note does not study router behavior. It is however possible to make some relevant observations that may be the start of a later serious study.

A router can estimate, for all its buffers, the number of active flows of class i that are using that buffer. This can be done, for example, by the methods described in [3]. Let N_i be the estimated number of active flows of class i. If the router also knows that all class i flows are ECN–capable, and that all sources of class i flows are using the "$c_1,\ c_2,\ \alpha,\ \beta$" policy (with c_1 etc of course depending on i), it can for example set the marking probability $p^{(i)}$ for class i packets in that buffer in the order of

$$p^{(i)} = c_5 . N_i^{\beta_i - \alpha_i}. \tag{9.1}$$

In (9.1) the constant c_5 can depend on the buffer occupation etc. We again see that the case $\beta_i - \alpha_i = 1$ has a certain charm: the dependence of the probability p on the estimated number N of flows is smoother than for TCP. A small error in the estimate N has less serious consequences.

The router can always drastically reduce congestion windows by setting $p = 1$ for a significant fraction of a Round Trip Time. Since the router is marking, no packet loss ensues. It is desirable to do this only if the router can predict the effect of markings: If it does this "until the effect is noticeable", most congestion windows have been reduced to one MSS or less.

10 Implementation Issues

Making sure that a flow has at least a small handful of marked packets per Round Trip Time, (say between $\frac{1}{5}$ and 5), has the advantage that the congestion window can be controlled more tightly than in a system where the flow has a marked packet only once every many RTTs. It may have the disadvantage that the flow appears "jittery": there is a downward adjustment for every marked packet. There are many ways to get around this. One of these is the following:

- When the number of outstanding (unacknowledged) bytes is larger than the current congestion window, but less than the advertised window, the source still is allowed to transmit 1 MSS for every 2 MSSs acknowledged.
- When the number of outstanding (unacknowledged) bytes is smaller than the current congestion window (and therefore smaller than the advertised window), the source can transmit at most two packets (at most one MSS each) for every packet acknowledged.

Hence "locally in time" there is additive increase as well as additive decrease. Over slightly larger timescales the congestion window is the actual constraint.

The "real" solution probably must combine ideas as above with ideas already being discussed for TCP (such as larger starting congestion windows, etc.).

Multiplicative decrease ($\beta = 1$) with small c_2 has the disadvantage that for small congestion windows the decrease becomes small. It may be advisable (this needs study) to decrease the congestion window (for every marked packet) by (for example)

$$min(c_2 W, \frac{1}{2}) \text{ MSSs.} \tag{10.1}$$

If the choice $\beta = 1$, $\alpha = 0$ is made, we saw that there are at least two ways to implement the flow control: One as in Section 2, one as in Section 8. There may be more ways of achieving the same goal. The two methods described are not quite identical for larger values of p. More investigation is desirable on the relative performances of these implementations. In particular, the implementation in Section 8 mimmics slowstart in the situation where the congestion window is "much too small", and thus may make construction of a explicit initialization phase unnecessary.

11 Conclusions

In this paper we study mechanisms in the Internet where Routers give feedback about their state of congestion to endstations (say sources) by ECN (Explicit Congestion Notification). We argue that Router Behavior (e.g. whether and when to mark packets) and Source Behavior (e.g. how to modify congestion windows in reaction to marked and unmarked packets) must be designed together. We argue that the advent of ECN is an opportunity, quite possibly the last opportunity, to modify the TCP feedback system (in the short term: give different interpretations to "drop" and "mark", and make the interpretation of "mark" dependent on the type of IP packet).

We discuss the general TCP Paradigm, where there are general $incr(.)$ and $decr(.)$ functions. We then restrict our attention to a smaller class of such schemes, where $incr(w) = c_1 w^\alpha$ and $decr(w) = c_2 w^\beta$. For these functions we predict source behavior,

including the stationary behavior of congestion window sizes, as function of the marking probability p.

Based on the number of marked packets per Round Trip Time we recommend $\beta - \alpha \leq 1$, and based on implementability we recommend, at least for the time being, $\beta - \alpha = 1$. We show that there is an alternative way of thinking about congestion window evolution, with a corresponding different implementation outside the "TCP Paradigm", that at least for small marking probabilities has the same effect as the TCP Paradigm with $\beta = 1$, $\alpha = 0$. The two ways of thinking about (essentially) the same scheme make it possible to come to a more systematic way of setting parameter values.

While $\beta = 1$, $\alpha = 0$ is the current favorite, other parameter values must not be discarded. In fact, there is an interesting question on what the appropriate objective to be optimized really is.

Acknowledgement. I thank Christian Huitema for suggesting to differentiate between In–Rate and Out–of–Rate packets.

A The Ornstein – Uhlenbeck Approximation

In the situation of Theorem 2,

$$W_{n+1} = \begin{cases} W_n + c_1 {W_n}^{\alpha} & \text{with probability } p, \\ W_n - c_2 {W_n}^{\beta} & \text{with probability } 1-p, \end{cases} \tag{A.1}$$

with $\alpha < \beta < 1$, $c_1 > 0$, $c_2 > 0$. For the process

$$X(t) = p^{\nu_1} \left(W_{\lfloor \frac{t}{p^{\nu_2}} \rfloor} - \left(\frac{c_1(1-p)}{c_2 p} \right)^{\frac{1}{\beta - \alpha}} \right), \tag{A.2}$$

we therefore have:

$$\frac{1}{p^{\nu_2}} E[X(t+p^{\nu_2}) - X(t) | X(t) = x] = p^{\nu_1 - \nu_2} \left(W_{\lfloor \frac{t}{p^{\nu_2}} + 1 \rfloor} - W_{\lfloor \frac{t}{p^{\nu_2}} \rfloor} \right) =$$

$$p^{\nu_1-\nu_2}\left((1-p)c_1 W^{\alpha}_{\lfloor \frac{t}{p^{\nu_2}} \rfloor} - pc_2 W^{\beta}_{\lfloor \frac{t}{p^{\nu_2}} \rfloor}\right) =$$

$$p^{\nu_1-\nu_2}\left((1-p)c_1\left(\left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{1}{\beta-\alpha}} + p^{-\nu_1}x\right)^{\alpha} - pc_2\left(\left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{1}{\beta-\alpha}} + p^{-\nu_1}x\right)^{\beta}\right).$$

We guess that

$$|p^{-\nu_1}x| << \left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{1}{\beta-\alpha}} \tag{A.3}$$

and do binomial expansions of the inner expressions. This gives

$$\frac{1}{p^{\nu_2}}E[X(t+p^{\nu_2}) - X(t)|X(t)=x] \sim$$

$$p^{\nu_1-\nu_2}\left\{(1-p)c_1\left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{\alpha}{\beta-\alpha}} + (1-p)c_1\alpha\left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{\alpha-1}{\beta-\alpha}}.p^{-\nu_1}x\right.$$

$$\left. -pc_2\left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{\beta}{\beta-\alpha}} - pc_2\beta\left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{\beta-1}{\beta-\alpha}}.p^{-\nu_1}x\right\}$$

The highest–order terms drop out and we get

$$\frac{1}{p^{\nu_2}}E[X(t+p^{\nu_2}) - X(t)|X(t)=x] \sim$$

$$-x(\beta-\alpha)c_1{}^{-\frac{1-\beta}{\beta-\alpha}}c_2{}^{\frac{1-\alpha}{\beta-\alpha}}(1-p)^{-\frac{1-\beta}{\beta-\alpha}}p^{\frac{1-\alpha}{\beta-\alpha}-\nu_2}$$

We see that to get a "useful" (Ornstein–Uhlenbeck type) result for $p \downarrow 0$ we need

$$\nu_2 = \frac{1-\alpha}{\beta-\alpha}. \tag{A.4}$$

Repeating the process for second moments we see that the Ornstein–Uhlenbeck result holds as long as in addition to (A.4) also

$$\nu_1 = \frac{1+\beta}{2(\beta-\alpha)}. \tag{A.5}$$

With (A.4) this yields the condition (4.19).

Since $\alpha < \beta$, for p small downward jumps in the process $X(.)$ are (much) larger than upward jumps. By first approximation, the quotient of downward jump sizes and standard deviation of $X(.)$ is

$$\left(\frac{\text{Jump}}{\text{St.Dev}}\right) = \sqrt{2(\beta-\alpha)}c_1{}^{-\frac{1-\beta}{2(\beta-\alpha)}}c_2{}^{\frac{1-\alpha}{2(\beta-\alpha)}}(1-p)^{\frac{\beta}{\beta-\alpha}}p^{\frac{1-\beta}{2(\beta-\alpha)}}, \tag{A.6}$$

which shows that for $p \downarrow 0$ the paths of the process $X(.)$ become continuous. The smaller the expression in (A.6), the more "almost continuous" the paths of the process $X(.)$.

It must be noted that since in Theorem 2 $\alpha < \beta < 1$, and hence

$$\nu_2 = \frac{1-\alpha}{\beta-\alpha} > 1, \tag{A.7}$$

the speed-up of the process $X(t)$ compared with the process W_n is higher in theorem 2 than in theorem 1.

The "guess" (A.3) is proven to be correct by the same idea as used in (A.6):

$$|x| = O(\text{st.dev}(X)) = O(1) << p^{\nu_1}\left(\frac{c_1(1-p)}{c_2 p}\right)^{\frac{1}{\beta-\alpha}} = \left(\frac{c_1(1-p)}{c_2}\right)^{\frac{1}{\beta-\alpha}} p^{-\frac{1-\beta}{2(\beta-\alpha)}}.$$

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.
Submitted to Perf. Eval. Rev. This paper has circulated on the Web since August 1996. see ftp://ftp.bellcore.com/pub/tjo/TCPwindow.ps for a copy.

[3] Ott, T.J., Lakshman, T.V. and Wong, L.H. (1999) SRED: Stabilized RED. *Proceedings of IEEE INFOCOM'99*, pp 1346 - 1355, March 1999.

[4] Floyd, S. (1994) TCP and Explicit Congestion Notification. *ACM Computer Communications Review* 21 no 5, pp 8-23.

[5] Ramakrishnan, K.K. and Floyd, S. (1998) A Proposal to add Explicit Congestion Notification to IPv6 and to TCP. *Internet Draft*, draft-kksff-03.txt, Oct 1998.

[6] Padhye, J., Firoiu, V., Towsley, D, and Kurose, J. (1998) Modeling TCP Throughput: a Simple Model and its Empirical Validation, Proceedings of *Sigcomm'98*, Sept 1998.

[7] Kumar, A. (1998) Comparative Performance Analysis of Versions of TCP in a Local Network with a Lossy Link, *IEEE/ACM Transactions on Networking*, August 1998.

[8] D Chiu and R Jain (1989) Analysis of the Increase/Decrease Algorithms for Congestion Avoidance in Computer Networks, *Journal of Computer Networks and ISDN*, June 1989.

[9] Feng, W. Kandlur, D. Kang, G. and Saha, D. (1998) Adaptive Packet Marking for Providing Differential Services in the Internet. *ICNP*. October 1998.

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

[11] Stevens, W.R. (1994), *TCP/IP Illustrated*, volume 1, Addison-Wesley, Reading MA, 1994.

[12] Wright, G.R. and Stevens, W.R. *TCP/IP Illustrated*, volume 2, Addison-Wesley, Reading MA, 1995.

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

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

[15] 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.

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

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

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

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

[20] 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.

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

[22] A. Neidhardt *Private Communication* 1997.

[23] 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.

[24] 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.*