Congestion control in networks with no congestion drops

Yi Lu, Stanford University. yi.lu@stanford.edu.
Rong Pan, Cisco Systems. ropan@cisco.com
Balaji Prabhakar, Stanford University. balaji@stanford.edu
Davide Bergamasco, Cisco Systems. davide@cisco.com
Valentina Alaria, Cisco Systems. valaria@cisco.com
Andrea Baldini, Cisco Systems. abaldini@cisco.com

***Abstract*— Congestion is intrinsic to the operation of networks and is usually handled by a combination of algorithms at the link and network/transport layers. Link level algorithms alleviate "transient congestion" caused by the temporary oversubscription of a link due to a burst of packets arriving at a switch or router buffer. Network or transport level algorithms alleviate "sustained congestion" which occurs when the long-term arrival rate at a link exceeds its capacity. Algorithms at the two levels interact to provide a scalable, stable and fair bandwidth allocation to the flows passing through the network.**

Link level algorithms are typically very simple: drop or mark packets with increasing probability as buffer congestion increases; moreover, if a packet arrives at a full buffer, drop it. These dropped or marked packets are used by the transport algorithms to adjust the transmission rate of sources.

In this paper we are concerned with networks in which packets cannot be dropped when there is congestion. In such networks a back-pressure mechanism "pauses" the link or links feeding a congested buffer, thus preventing further packets from arriving at the buffer. The links are later unpaused when the buffer becomes uncongested.

This paper is a theoretical study of the stability and fairness properties of network level congestion control when pause mechanisms operate at the link level to prevent packet drops. Our focus is on the Backward Congestion Notification (BCN) algorithm which is being considered by the IEEE 802.1 standards body for deployment in switched Ethernet networks.

I. Introduction

Congestion control is a basic operation in networking and has a rich history of algorithm development and theoretical study in wide area networks, such as the Internet. Congestion has two components: "transient" which is due to temporary, random fluctuations in the packet arrival process, and "sustained" which is due to a sustained oversubscription of a link's bandwidth. Transient congestion is effectively dealt with using buffers at the links and dropping the excess packets, while sustained congestion is alleviated by transport protocols like TCP which reduce the transmission rate of a flow when its packets are dropped or marked. This combination of algorithms provides stable, scalable and fair bandwidth sharing to flows which use the Internet. A representative sample of the literature may be found in the book [8], the survey papers [5], [9], and at the website [2].

Consider a different link level mechanism: when a buffer begins to fill up, a back pressure mechanism pauses the link(s) feeding the buffer, thereby preventing buffer overflow. This type of mechanism leads to no packet losses but, potentially, spreads congestion to upstream buffers and eventually to the sources. A detrimental effect of congestion spreading is that it is possible for a source, say S, which sends data at a high rate to cause a number of links to be paused and to adversely affect the throughput of other (non-congesting) sources which share these links with S. To mitigate this, it is desirable to have a mechanism for the network to signal to S that it should decrease its sending rate. Such a mechanism has been proposed: the Backward Congestion Notification (BCN) mechanism [1] which is currently being considered in the IEEE 802.1 Standards Committee for deployment in switched Ethernets.

It is the purpose of this paper to analyze, via theory and simulations, the stability and fairness properties of the BCN mechanism. We shall describe the details of the BCN mechanism that are relevant for the analysis. Other, important, details related to frame formats and signalling are out of the scope of the present treatment and we refer the interested reader to the BCN specification.[1]

Before proceeding further, it is useful to understand an anticipated use of link level pausing. Link level pausing is defined in the IEEE 802.3x standard and allows an Ethernet link to be paused by the switch which the link feeds. When used at all the links of a network, this feature makes the network "low-loss;" i.e., packets are not dropped because of congestion (packets may still be dropped because of data corruption). A low-loss network can help ensure that packets are not retransmitted or duplicated, thereby avoiding the use of large resequencing buffers and a potentially large retransmission time. Therefore, an Ethernet with link-level pausing can be used to carry traffic which require low-loss and low-latency links; for example, storage traffic which has been using the Fibre Channel (FC) networking technology [6], [7].

The BCN mechanism (described in detail later) is a way for mitigating the congestion spreading effects due to link-level pause by enabling the network to signal sources to reduce (or increase) their sending rates. The Ethernet environment for which the BCN mechanism is designed is to be contrasted with the Internet in the following crucial

[1] At the time of this writing the complete BCN specification was not yet available because the IEEE 802.1au Work Group was still working on the mechanism. A number of documents produced by the .1au WG can be found at [10].

ways:

1. A router in the Internet can signal a source to decrease its sending rate by marking or dropping its packets. This signal is conveyed by (essentially per-packet) acknowledgements from the receiver to the source; the router does not directly signal the sources. Ethernet is a layer-2 technology and cannot assume the existence of such end-to-end protocols. Therefore, BCN messages are directly generated by the switches and sent to the source.
2. A TCP source increases its sending rate (actually, its window size) voluntarily, probing for bandwidth in the network; current routers do not signal rate increases to sources. BCN allows switches to signal both rate *increase* and decrease messages to sources. The increase signals help sources consume available link bandwidth rapidly.[2] In this sense BCN is similar to the XCP [4] protocol proposed in the literature for the Internet.
3. New sources do not necessarily gently increase their sending rate, *a la* the slow-start mechanism of TCP. A source can initiate a transfer at the full line-rate (e.g. 10 Gbps).

With these preliminary remarks, we're ready to describe the contributions of the present paper. The BCN mechanism is described in the next section, Section II. It opens a congestion control loop from a source to a switch which is, essentially, the "bottleneck switch" for that source (the bottleneck may change for each source during the lifetime of a session). In Section III we present a stability analysis of this control loop. In Section IV we study the fairness properties of the BCN mechanism and its impact on flow completion times in a dynamic setting where flows come and go. Finally, Section V presents conclusions and outlines further work.

II. A Description of The BCN Mechanism

Overview. We describe the main components of the BCN mechanism, giving the details of congestion signalling and reaction later.

1. Congestion Point (CP): This refers to a switch buffer attached to a link that is being oversubscribed. The CP samples an incoming packet and generates a BCN message addressed to the source of the packet. The BCN message contains information regarding the extent of congestion to the source.
2. Reaction Point (RP): This refers to a rate limiter associated with the source which receives the BCN message. The RP adjusts its sending rate based on the feedback it received from the CP. An RP which has received a BCN message signalling a rate decrease from a CP becomes associated with that CP. Its subsequent packets contain a rate limited tag (RLT) containing the identity of the CP currently associated with the RP. The identity of the CP is referred to as CPID. An RP can be associated with multiple CPs over time; however, only one CP is associated with it at any given time (i.e., the last CP which sent a rate decrease BCN message to the RP).
3. The CP may send rate increase BCN messages to an RP associated with it as congestion decreases. The same sampling process picks out the packets for the RP to potentially send rate increase signals to. A CP shall not send rate increase signals to an RP not associated with it. Similarly, an RP shall not respond to rate increase signals from a CP with which it is not currently associated.
4. Finally, an RP may increase its sending rate voluntarily (even when not instructed to do so by its CP). Such a rate increase is typically small and serves two purposes: to gently probe for extra bandwidth, and to ensure fail-safe operation in the event that messages from its CP are not delivered. We shall see an additional important purpose of self-increase: it helps achieve bandwidth fairness, especially for short-lived flows.

[2]We note that BCN sources do have the ability to increase their rates without explicit increase signals from the network; thus, rate increase at a BCN source can occur either due to signals from a switch or due to self-increase.

The BCN Mechanism

As mentioned earlier, we shall only describe those parts of the BCN mechanism which are relevant for our analysis of stability and fairness. We omit a number of details important for an exact implementation, and refer those interested to the BCN specification.

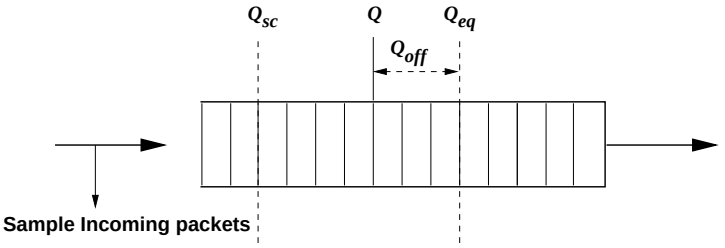


Fig. 1. *Congestion detection and BCN message generation.*

Figure 1 shows the buffer at a CP. The overall goal of BCN is to hold the buffer occupancy at a level Q_{eq} (for equilibrium queue level) so that the buffer is neither overrutilized nor underutilized. This is achieved using a proportional-integral contol: offsets from Q_{eq}, denoted Q_{off}, and the rate of change of the queue-size, denoted Q_δ, are used to adjust the sending rate at the RPs.

The CP Mechanism. The CP samples packets with a probability P.

Severe congestion: If Q exceeds a high threshold $Q_{sc} > Q_{eq}$, then the RP is sent the "severe congestion" message, BCN(0,0).

Decrease signal: If $Q_{eq} < Q < Q_{sc}$, compute $Q_{off} = Q - Q_{eq}$ and $Q_\delta = Q - Q_{old}$, where Q_{old} is the value of the queue at the previous sampling instant. Send BCN(Q_{off}, Q_δ).

Increase signal: If $Q_{eq} > Q$, then check to see if the packet has an RLT (rate-limited tag) and if the CPID in the RLT matches the identifier of the CP. If both of the above are true, send BCN(Q_{off}, Q_δ), where Q_{off} and Q_δ are computed as above; else, send no signals.

The RP Mechanism.

Severe congestion: When an RP receives a BCN(0,0) message, it sets its current rate R to 0 and starts a timer which will expire after a random amount of time. When the timer expires, it sets its rate to a small value R_{min} and sends packets which will hopefully trigger a rate increase signal from its CP.

When the RP receives a non-BCN(0,0) message, it computes the quantity $F_b = -(Q_{off} + wQ_\delta)$, where w is a weight parameter chosen as explained later.

Decrease: If $F_b < 0$, then it performs multiplicative decrease and sets its current rate R to

$$R \leftarrow R(1 - G_d|F_b|),$$

where G_d is the decrease gain.

Increase: If $F_b > 0$, then it performs additive increase, setting

$$R \leftarrow R + G_i F_b R_u,$$

where G_i is the increase gain and R_u is a constant that gives the correct rate units (one can fold R_u into G_i, we're being faithful to the specification).

Note: The parameters w, G_d, G_i and R_u are chosen to make the control loop stable, responsive and scalable. The specification prescribes ranges for the parameters w, G_d, G_i and R_u. In the next few sections we show how a control theoretic analysis of the control loop with feedback delays yields ranges for the parameters. We shall then present simulations with parameter values obtained from the analysis.

III. Stability Analysis

This section presents a local stability analysis of the BCN control loop. Due to a shortage of space, we are unable to present all the work, especially the simulation studies of various scenarios and configurations. We shall present it in forthcoming publications.

We analyze the BCN mechanism where sources do not increase voluntarily; that is, all rate decrease and increase at an RP is due to signals from its CP. The analysis when sources self-increase their rates follows as an extension.

First, note that when sources do not self-increase there are *multiple* equilibrium points for the control system. An illustrative example: consider one link of capacity C shared by two sources with rates R_1 and R_2. Any set of values such that $R_1 + R_2 = C$ is an equilibrium point because in this case the switch does not signal rate changes (there is no congestion or free capacity) and the sources do not self-increase. The following theorem, stated without proof, generalizes the above observation to many flows and arbitrary networks.

Theorem 1: Let $\mathbf{R}$ be the vector of flow rates, $\mathbf{A}$ the routing matrix and $\mathbf{C}$ the vector of link capacities. For a general network, without self-increase, the equilibrium set of rates $\mathbf{R}$ is the set of boundary points of $\{\mathbf{R}|\mathbf{AR} \leq \mathbf{C}\}$. In particular, there is no unique equilibrium point for the flow rates.

Given the lack of a unique equilibrium point around which to linearize the control system, we shall choose the "fair share" equilibrium point: $R^* = \frac{C}{N}$, $Q^* = Q_{eq}$ and $Q_\delta = 0$. This equilibrium point is robust with respect to stochastic perturbations in the arrival process which is the realistic scenario.

A. The System Equations and Their Linearization

We shall now write down the basic equations which describe the BCN mechanism, linearize them about the above equilibrium point and then obtain the stability region.

Link (CP)

$$\frac{dq(t)}{dt} = N \times R(t) - C.$$

Source (RP)

$$F_b(t) = -\left[(q(t) - q_{eq}) + \frac{wS}{CP} \times \frac{dq(t)}{dt}\right]/S$$

If $F_b(t-\tau) > 0$,

$$\frac{dR(t)}{dt} = [G_i R_u \times F_b(t-\tau) \times R(t-\tau) \times P]/S$$

If $F_b(t-\tau) < 0$,

$$\frac{dR(t)}{dt} = [R(t) \times G_d \times F_b(t-\tau) \times R(t-\tau) \times P]/S$$

Notice $F_b(t)$ is measured in terms of a fixed cell-size S.

Substituting for $F_b(\cdot)$ in the right hand side of the above equations and by making the substitution: $\frac{dq(t)}{dt} = N \times R(t) - C$, we get

If $F_b(t-\tau) > 0$,

$$\frac{dR(t)}{dt} = G_i R_u \left(\frac{q_{eq}P}{S^2} + \frac{w}{S}\right) R(t-\tau) \quad (1)$$

$$- \frac{G_i R_u w N}{CS} (R(t-\tau))^2 \quad (2)$$

$$- \frac{G_i R_u P}{S^2} R(t-\tau) q(t-\tau). \quad (3)$$

If $F_b(t-\tau) < 0$,

$$\frac{dR(t)}{dt} = G_d \left(\frac{q_{eq}P}{S^2} + \frac{w}{S}\right) R(t-\tau) R(t) \quad (4)$$

$$- \frac{G_d w N}{CS} (R(t-\tau))^2 R(t)$$

$$- \frac{G_d P}{S^2} R(t-\tau) q(t-\tau) R(t). \quad (5)$$

Linearization

Linearizing equations (3) and (5) about the equilibrium point $R^* = \frac{C}{N}$, $q^* = q_{eq}$, and $\frac{dq}{dt}^* = 0$ is now straightforward, and we get the following linear equations:
If $F_b(t-\tau) > 0$,

$$\delta\dot{R}(t) = -\frac{G_i R_u w}{S}\,\delta R(t-\tau) - \frac{G_i R_u PC}{NS^2}\delta q(t-\tau). \quad (6)$$

If $F_b(t-\tau) < 0$,

$$\delta\dot{R}(t) = -\frac{G_d wC}{NS}\delta R(t-\tau) - \frac{G_d P}{S^2}\left(\frac{C}{N}\right)^2 \delta q(t-\tau) \quad (7)$$

$$\delta\dot{q}(t) = N\delta R(t). \quad (8)$$

B. Stability Analysis of Linearized System

Note that there are two equations for the rate, R, in the feedback system. The linearized versions of these equations are described by equations (6) and (7), respectively. The sign of $F_b(t-\tau)$ determines which equation is to be used at any time t. We first show that the stability of each of the two linearized systems separately is a *sufficient condition* for the stability of the overall system. We then derive the conditions for the stability of each of the two linearized systems; together these conditions give conditions on the stability of the overall linearized system.

There is a symmetry in the model: all sources behave the same way; their rates are equal at all times. Hence it suffices to consider the rate R of a single source in the following theorem. Let System 1 refer to the system governed by the pair of equations (6) and (8); i.e., the sytem enforced by $F_b(t-\tau) > 0$. Let System 2 refer to the one governed by equations (7) and (8); the sytem enforced by $F_b(t-\tau) < 0$.

Theorem 2: Suppose Systems 1 and 2 are individually stable, and the flow rate R and queue size q in each system converges to the same equilibrium values R^* and q_{eq}, respectively. Then, the overall system, defined by equations (6), (7) and (8) is stable. Moreover, R converges to R^* and q converges to q_{eq} in the overall system.

Proof: Note, from equations (6) and (7), that $\delta\dot{R}(t) = 0$ iff

$$w\delta R(t-\tau) + \frac{PC}{NS}\delta q(t-\tau) = \frac{PC}{NS}\delta F_b(t-\tau) = 0;$$

i.e., iff $F_b(t-\tau) = 0$. Hence the switching between the two systems occurs only when $\delta\dot{R}(t) = 0$.

Since System 1 and 2 are second-order, they either have two real poles or two complex poles. If at least one system has two real poles and both systems are stable, switching can happen at most once, and the system with two real poles will bring $\delta R(t)$ to 0.

Now suppose both System 1 and System 2 have two complex poles and are stable (i.e., underdamped). Since each is stable, the trajectory of $\delta R(t)$ in each system (under an impulse input) will oscillate about 0 with exponentially decreasing peaks and increasing troughs, before eventually converging to 0.

Consider the overall system when Systems 1 and 2 have two complex poles. Without loss of generality assume (under an impulse input) that System 1 brings $\delta R(t)$ to a peak P_1. Since $\delta\dot{R}(t) = 0$ at all peaks and troughs, the two systems only switch at peaks and troughs. In particular, System 2 will start from P_1 with gradient 0. Since System 2 is stable on its own, it will bring $\delta R(t)$ to a trough T_1 such that $|T_1| = \alpha_2|P_1|$, with $\alpha_2 < 1$. [3] As the switching repeats, we see that the trajectory of $\delta R(t)$ has the property $|P_1| = |T_1|/\alpha_2 = |P_2|/\alpha_2\alpha_1 = |T_2|/\alpha_2^2\alpha_1 = ...$, bounded by two alternating exponential decaying envelops. Hence the overall system is stable.

Since $\delta R(t)$ converges to 0, $R(t)$ converges to R^*. Also, $\dot{q}(t) = NR(t) - C$ converges to 0. Since the gradient of $R(t)$ goes to 0, $F_b(t-\tau)$ goes to 0. Thus we deduce $\delta q(t)$ goes to 0 and $q(t)$ goes to q_{eq}. ■

Theorem 3: The linearized BCN system is stable if the following conditions hold:
(i) $\frac{G_i R_u w}{S} \le \frac{1}{a\tau}$, (ii) $G_i R_u w^2 > \frac{PC}{b\sqrt{b^2+1}}$,
(iii) $G_d w \le \frac{SN}{aC\tau}$, and (iv) $G_d w^2 > \frac{PN}{b\sqrt{b^2+1}}$,
where $a \ge 1$ and $\frac{b}{a} + \arctan b = \frac{\pi}{2}$.

Proof: Conditions (i) and (ii) are sufficient for the stability of System 1, while (iii) and (iv) are sufficient for the stability of System 2. The Laplace transforms of equations (6) and (8) are:

$$\delta R(s) = -\frac{\frac{G_i R_u PC}{NS^2}}{s + \frac{G_i R_u w}{S}e^{-s\tau}}e^{-s\tau}\delta Q(s) \quad \text{and} \quad \delta Q(s) = \frac{N}{s}\delta R(s).$$

As a first step, we would like the pole in the equation to be simple; i.e., for $\frac{G_i R_u w}{S}e^{-s\tau} \approx \frac{G_i R_u w}{S}$. It is shown in Appendix I that this is true if

$$\frac{G_i R_u w}{S} \le \frac{1}{\tau}. \quad (9)$$

We introduce the design parameter a to control the size of the margin between the pole $\frac{G_i R_u w}{S}$ and the stability region. In particular, we shall require that

$$\frac{G_i R_u w}{S} \le \frac{1}{a\tau}, \quad \text{for } a \ge 1. \quad (10)$$

Now, assuming that the pole is at the above bound; that is, the pole is at $p = \frac{1}{a\tau}$. We obtain the frequency s^* at which the phase of the transfer function equals $-180°$ as the solution to the equation:

$$-90° - \frac{s}{pa} - \arctan\frac{s}{p} = -180°.$$

Write $b = \frac{s^*}{p}$. We would like the magnitude of the transfer

[3] α_2 is solely determined by the position of the complex poles of system 2. In particular, for poles at $-a \pm ib$, $\alpha_2 = exp(-\pi a/b)$.

function at $s^* = bp$ to be less than 1. Or, we require:

$$\begin{aligned} & \left\| \frac{\frac{G_i R_u PC}{NS^2}}{s + \frac{G_i R_u w}{S}} e^{-s\tau} \frac{N}{s} \right\|_{s=j\frac{bG_i R_u w}{S}} \\ = & \left\| \frac{\frac{PC}{wS}}{(\frac{sS}{G_i R_u w} + 1)s} \right\|_{s=j\frac{bG_i R_u w}{S}} \\ = & \frac{PC}{b\sqrt{b^2+1} G_i R_u w^2} \quad < \quad 1 \\ \Rightarrow & \quad G_i R_u w^2 > \frac{PC}{b\sqrt{b^2+1}}. \end{aligned} \tag{11}$$

Equations (10) and (11) correspond to conditions (i) and (ii) of the theorem.

In an entirely similar fashion the transfer function of System 2 yields conditions (iii) and (iv) of the theorem. This completes the proof. ■

The four conditions in Theorem 3 provide guidelines for choosing the parameters $G_i R_u$, G_d and w when P is fixed. By varying a, one can obtain the complete set of parameters that ensure the stability of the linearized system.

Figure 2 shows a simulation designed for a single link network with $N = 50$ and $\tau = 200\mu s$. We choose the paramters as follows: $G_i = 4$, $R_u = 1e6$, $w = 2$, $Gd = 1/128$, which are obtained when $a = 5$ and $b = 2.2$ and $P = 0.01$. The pole for increase is at 666 rad/s and the pole for decrease is at 521 rad/s.

Remark about the simulation environment: It is important to point out that all the simulations in this paper are run without the sources obeying the "severe congestion" signal; that is, the sources only perform additive increase and multiplicative decrease. The behavior expected with the severe congestion mode (which will be presented in future work) will be more stable, but perhaps with a more sluggish response in terms of flow completion times. Moreover, packets arriving to a full buffer are dropped and will be retransmitted by the source. Therefore, a flow which has K packets to send will transmit exactly K packets successfully; dropped packets are requeued at the source. Even though we have not incorporated link-level pause in our simulations, it is clearly not necessary for the basic single-link topology studied in this paper; the difference in performance is negligibly small. We plan to include link-level pause and severe congestion in future work where we shall simulate more complex topologies. The control-theoretic analysis of this paper is fully captured by our present simulations.

The simulation shows 400 sources arriving during the simulation run-time, bursts of 50 arriving every 0.2 seconds, each source starts at a rate of 100 Mbps thus bringing an extra load of 5 Gbps every 0.2 seconds. The desired q_{eq} is 16 packets, all packets are 1500 bytes long. As Figure 2 shows, both the queue-size and the total rate are quite stable, especially when considering that the system was loaded with 8 times more flows than the $N = 50$ value used for choosing the parameters!

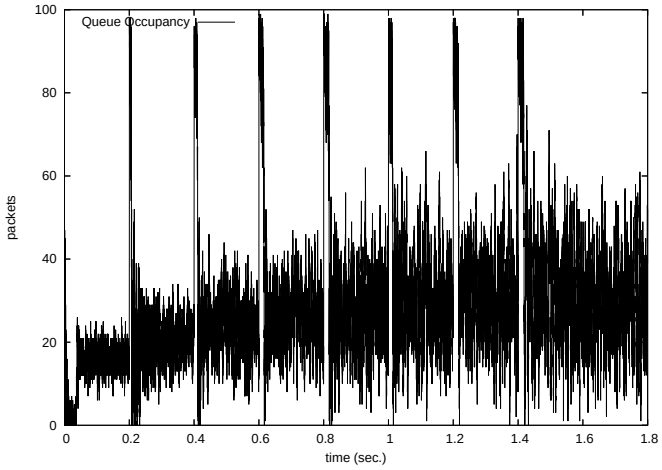


Fig. 2. *Queue-size at the buffer attached to a 10Gbps link; no self-increase.*

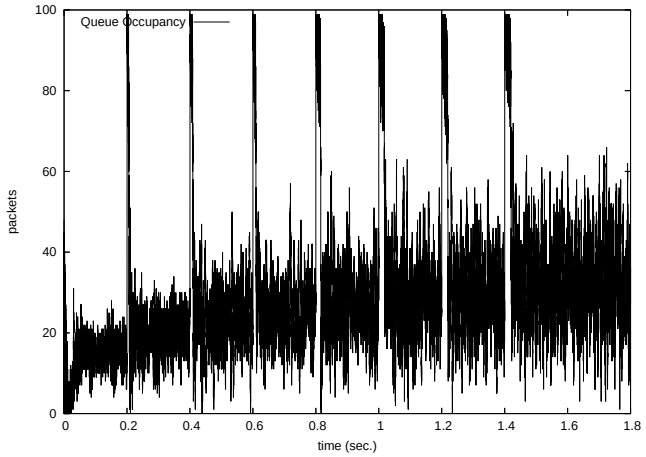
C. Self-increase Algorithms

We present three algorithms that allow a source or RP to increase its sending rate, *even though* it does not receive an increase signal from its CP. As mentioned earlier, such increases ensure fail-safe operation should BCN messages be lost. We shall see in the next section how they also affect the fairness in bandwidth allocation.

Algorithm: SI-1. This algorithm allows an RP to increase its sending rate by an amount of I bps per second (denoted henceforth as I bps/s). Under this algorithm, if the network has N sources, then the total arrival rate increases by NI bps every second. For a large enough N, this algorithm is clearly unstable. This is evident from the plots shown in Figure 3. The parameters and arrival patterns are all as in the setup of Figure 2. As we can see, when we have a gentle self-increase of $I = 10$Mbps/s the system is much more stable than when $I = 500$Mbps/s.

Algorithm: SI-2. This algorithm allows an RP to increase its sending rate by a constant factor. If the current rate is R and rate increases occur every T seconds, then the rate at the next increment time will equal $R.i.T$. This scheme enjoys the advantage that the the net arrival rate on any link can only increase by the factor i in fixed intervals of time, *regardless* of the number of sources. This makes the algorithm more stable compared to SI-1. Figure 4 (left) shows the queue-sizes under SI-2. The factor i was chosen to be 10, which is a pretty aggressive factor. Nevertheless, the figure shows that the scheme is well-behaved.

Algorithm: SI-3. This algorithm increases the rate additively. The amount of increase is inversely proportional to the number of *negative feedback* signals obtained by a source. For example, if rate increases occur every T seconds, then the current rate R will increase to $R + \frac{I.T}{\#NegF_b}$, where I is the increase amount and $\#NegF_b$ equals the number of



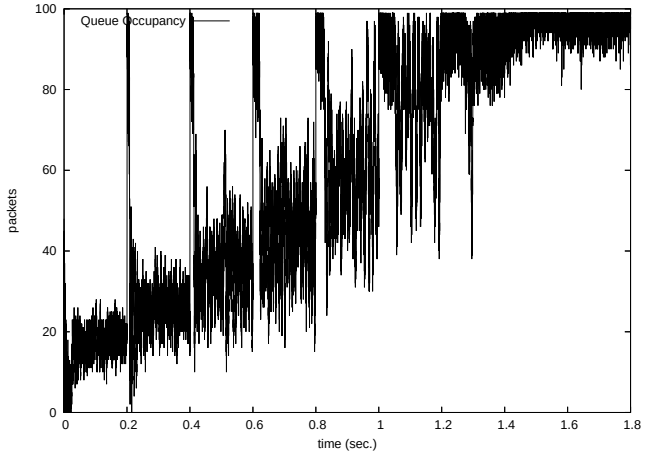
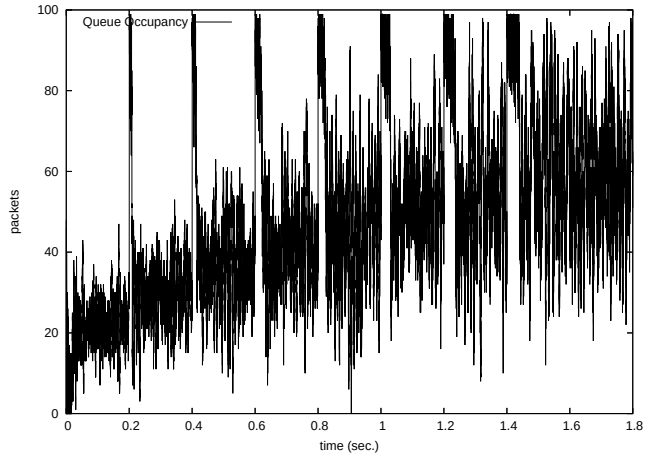


Fig. 3. *Queue-sizes under algorithm SI-1. Gentle increase of 10 Mbs/s (top) and aggressive increase of 500 Mbps/s (bottom).*



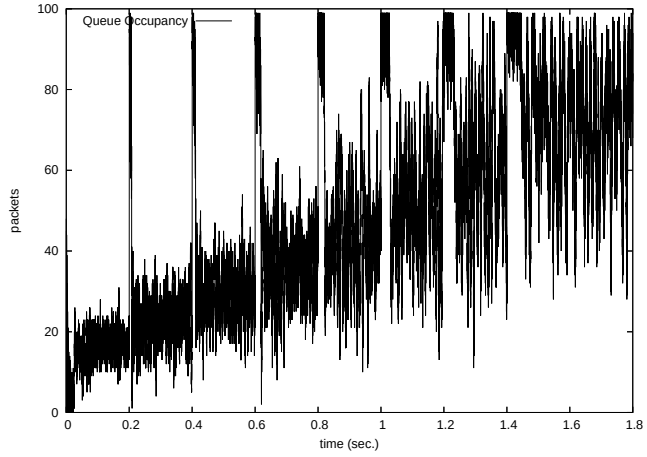


Fig. 4. *Queue-sizes under algorithm SI-2 (top), and under algorithm SI-3 (bottom).*

negative F_b signals received by the RP in the increase interval of T seconds. This algorithm is similar to SI-1, but fairer: it does not let large sources increase by as much as SI-1 does. Figure 4 (right) shows the queue-sizes under this algorithm.

IV. Fairness

In this section we shall consider the fairness properties of the schemes presented in the previous section. We shall call the scheme with no self increase SI-0. As seen in Theorem 1 SI-0 gives rise to multiple equilibria, only one of which gives an equal bandwidth allocation in the single-link case. This equilibrium is, therefore, fair and is stable under stochastic perturbations such as randomness in packet arrival processes. While equilibrium bandwidth shares are one indication of fairness, a more important figure of merit for any congestion control algorithm is the speed with which it achieves fairness when sources start off with an unequal bandwidth allocation. Several "fairness indices" have been defined in the literature (see [3], for example) to measure the fairness of an algorithm. We have performed the appropriate simulations, but do not present them here due to a lack of space. We have also found these indices not informative in practice because they are concerned with the situation when infinite sources send traffic.

The more realistic scenario corresponds to finite-sized flows arriving and departing. In this case one can consider the flow completion times and say that an algorithm is "fair" if it makes the completion times of similar-sized flows equal. We note that since bandwidth is shared by all flows, one flow completing transmission quickly implies that another must take longer. Hence, when considering similar (or equal) sized flows, one cannot expect the mean flow completion time to be an indication of fairness. This suggests defining a "fair" algorithm as one which reduces the *variance of flow completion times* across flows of similar size. We now

investigate the fairness of the schemes SI-0,...,SI-3 in the sense described above.

We consider flows whose sizes are drawn from a Pareto distribution with parameter 1.8 and mean flow size equal to 1 MB. The arrival rate is Poisson with mean 1125 flows/sec. Hence the average traffic in the network is 9 Gbps. The starting rate of all flows is 1 Gbps. Figure 5 compares the mean and the normalized standard deviation of the flow completion times. As expected, whereas all algorithms obtain the same mean completion time, SI-0 is markedly inferior in controlling the variance of the flow completion time. Thus, the other algorithms are fairer.

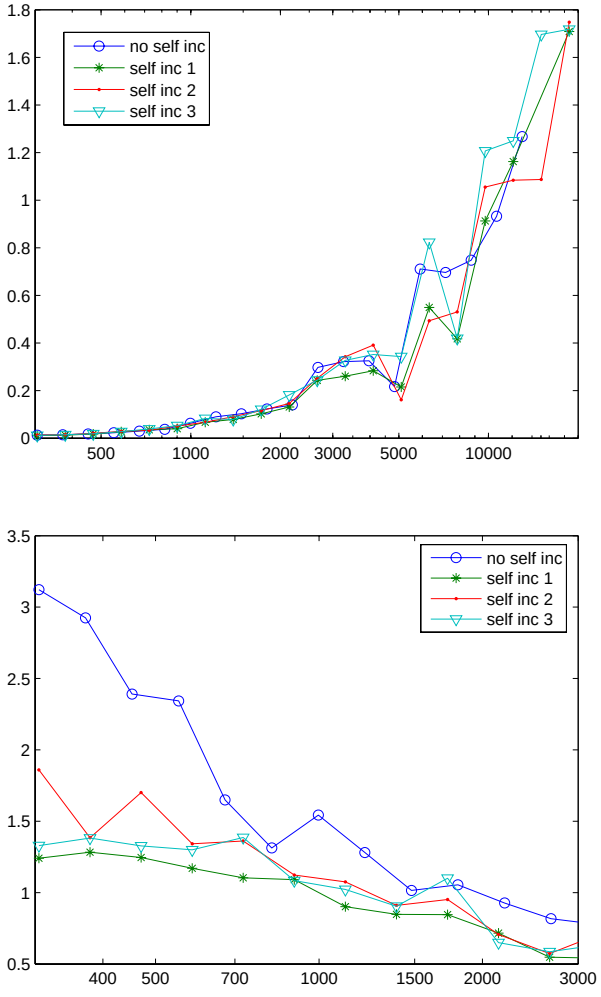


Fig. 5. *A comparison of flow completion times under the algorithms SI-0,...,SI-3. Means (top) and normalized standard deviation (bottom).*

V. Conclusion and Further Work

This paper presented the stability analysis of the BCN mechanism being considered by the IEEE standardization process. It points out, and analyzes, a subtlety that arises in this context: the overall system switches between two separate sets of equations depending on the sign of the feeback variable, F_b. In addition to stability, the fairness aspects of the BCN mechanism have been considered in detail. Notably, we have put forth a new notion of fairness involving the variance of flow completion times to compare algorithms in a more realistic scenario.

There is much ongoing and future work. We have already mentioned simulations which include the severe congestion mode, especially with more complex network topologies. It would be interesting to understand just how well link-level pause and BCN work as network size scales. It would also be very interesting to understand how BCN interacts with TCP, which performs end-to-end congestion control and relies on packet drops to get congestion notices.

Acknowledgment: We thank Paul Cuff for helping write the ns-2 simulator used in this paper.

References

[1] D. Bergamasco, R. Pan, "Backward Congestion Notification Version 2.0", IEEE 802.1 Interim Meeting, Garden Grove (CA), Sep 2005, http://www.ieee802.org/1/files/public/docs2005/new-bergamasco-bcn-september-interim-rev-final-0905.ppt

[2] S. Floyd. Papers on congestion control research. http://www.icir.org/floyd/papers.html

[3] R. Jain, W. Hawe, D. Chiu, "A quantitative measure of fairness and discrimination for resource allocation in Shared Computer Systems." DEC-TR-391, 1984.

[4] D. Katabi, M. Handley, C. Rohrs. "Internet Congestion Control for High Bandwidth-Delay Product Networks." ACM Sigcomm, Pittsburgh, August, 2002.

[5] F.P.Kelly. "The mathematics of traffic in networks." In "The Princeton Companion to Mathematics" (Editors W.T. Gowers and J. Barrow-Green) forthcoming, Princeton University Press.

[6] R.W.Krembel, R.Cummings. "The Fibre Channel Consultant: A Comprehensive Introduction." Northwest Learning Associates, 2000.

[7] R.W.Krembel. "Fibre Channel Switched Fabric." Northwest Learning Associates, 2001.

[8] R. Srikant. "The Mathematics of Internet Congestion Control." Birkhauser, 2004.

[9] D. X. Wei, C. Jin, S. H. Low, S. Hegde. "FAST TCP: motivation, architecture, algorithms, performance." IEEE/ACM Trans. on Networking, to appear Feb 2007.

[10] IEEE 802.1au Work Group Public Documents Repository, http://www.ieee802.org/1/pages/802.1au.html

Appendix I

We look at a transfer function of the form $\frac{1}{s+ae^{-s\tau}}$, or in the standard frequency form $\frac{1}{jw+ae^{-jw\tau}}$. Its magnitude is equal to

$$\frac{1}{\sqrt{a^2+w^2-2aw\sin w\tau}} \simeq \frac{1}{\sqrt{a^2+w^2}},$$

which is magnitude of

$$\frac{1}{jw+a}$$

when $w\tau$ is small.

Its phase is given by the expression

$$\arctan\frac{w-a\sin w\tau}{a\cos w\tau} \simeq \arctan\frac{w-aw\tau}{a(1-\frac{(w\tau)^2}{2})}.$$

When $a\tau \leq 1$, we solve

$$1 - \frac{(w\tau)^2}{2} < 0 \Rightarrow w > \frac{\sqrt{2}}{\tau} > a$$

so $a\tau \leq 1$ is a sufficient condition to closely approximate $\frac{1}{s+ae^{-s\tau}}$ by $\frac{1}{s+a}$. Note that when $a\tau > 1$, there is positive feedback from the beginning.