Statistical Bandwidth Sharing: A Study of Congestion at Flow Level

S. Ben Fredj, T. Bonald, A. Proutiere, G. Régnié, J.W. Roberts
France Telecom R&D
38 rue du Général Leclerc
92794 Issy les Moulineaux, France
{slim.benfredj,thomas.bonald,alexandre.proutiere,gwenael.regnie,james.roberts}@francetelecom.com

ABSTRACT

In this paper we study the statistics of the realized throughput of elastic document transfers, accounting for the way network bandwidth is shared dynamically between the randomly varying number of concurrent flows. We first discuss the way TCP realizes statistical bandwidth sharing, illustrating essential properties by means of packet level simulations. Mathematical flow level models based on the theory of stochastic networks are then proposed to explain the observed behavior. A notable result is that first order performance (e.g., mean throughput) is insensitive with respect both to the flow size distribution and the flow arrival process, as long as "sessions" arrive according to a Poisson process. Perceived performance is shown to depend most significantly on whether demand at flow level is less than or greater than available capacity. The models provide a key to understanding the effectiveness of techniques for congestion management and service differentiation.

1. INTRODUCTION

The great majority of current Internet traffic is contained in TCP connections generated by applications requiring the transfer of some kind of digital document. This traffic is elastic and network quality of service is experienced mainly through the variable throughout achieved by the congestion control algorithms of TCP. Since this depends on the numbers of connections currently sharing the links of the network path, which vary as new flows begin and existing flows end, throughput performance can only be measured in statistical terms. In this paper we investigate how throughput performance depends on available capacity and the volume and characteristics of offered traffic. The ultimate objective is to derive provisioning rules and traffic controls accounting for the way performance depends on demand and available capacity. In the interests of simplicity, we ignore the impact of non-responsive flows and assume all traffic is elastic.

Characteristics of IP traffic at packet level are notoriously complex (see [17], for example). Arguably, however, these characteristics are less an exogenous expression of user demand than a result of the closed loop control implemented by TCP. A study of throughput performance more naturally calls for a characterization of traffic at the level of the documents whose transfer is necessary to accomplish the underlying applications (Web page, FTP file, e-mail,...). The present study focuses therefore on evaluating throughput performance as a function of the arrival process and size statistics of flows corresponding to individual document transfers.

We coin the term "statistical bandwidth sharing" to denote a form of statistical multiplexing where the rate of concurrent traffic streams is adjusted automatically to make optimal use of available bandwidth. Such sharing is achieved with a certain degree of fairness when all users implement TCP. The evaluation of statistical bandwidth sharing performance provides insight into the nature of congestion at flow level and clarifies the scope for quality of service differentiation. Understanding the relation between performance, capacity and traffic demand is also necessary for the development of performance-related network provisioning procedures.

There is relatively little work in the literature on the evaluation of throughput performance under statistical traffic assumptions. Heyman et al [12] consider the performance of a bottleneck link shared by a fixed number of homogeneous sources alternately emitting documents and remaining inactive during a random think-time. Their results confirm that TCP shares link bandwidth fairly and they derive an analytical model which accurately predicts throughput performance. A notable result is that expected performance depends only on the means of the document size and think-time and not on their precise distributions. Berger and Kogan [4] have further explored this model in an asymptotic heavy traffic regime.

Massoulié and Roberts [25] propose a model similar to that of Heyman where however the flow arrival process is Poisson. They identify the underlying fluid flow model as an $M/G/1$ processor sharing queue. The Poisson arrival assumption is more appropriate when the considered link receives traffic from a very large population of users. Kherani

SIGCOMM'01, August 27-31, 2001, San Diego, California, USA..
Copyright 2001 ACM 1-58113-411-8/01/0008 ...$5.00.

and Kumar [21] have studied the statistical bandwidth sharing performance realized by TCP and confirm that the processor sharing model provides accurate estimates when connection arrivals are Poisson. Bu and Towsley [8] incorporate a discriminatory processor sharing model in their study of TCP performance with finite size flows.

De Veciana et al [11] consider statistical bandwidth sharing in a network setting assuming Poisson flow arrivals. They notably highlight the potential for a form of congestion collapse when demand on any link exceeds capacity. A recent study by Fayolle et al [13] illustrates the difficulty of evaluating statistical throughput performance on a path containing multiple bottlenecks. Bonald and Massoulié [6] have further explored statistical bandwidth sharing in a network, notably illustrating the impact on stability of certain service differentiation mechanisms.

The main contribution of the present paper is to show that previous results, derived using simplified traffic models, are in fact valid under very general and realistic assumptions. The latter consist in supposing flows are grouped in user "sessions" whose starting times constitute a Poisson process. Session structure, including the number of flows, their size and any correlation in successive flow and think-time statistics, can be perfectly general. To prove this we apply theorems from the theory of stochastic networks not hitherto employed in the study of bandwidth sharing performance. While the mathematics necessary to demonstrate the generality of the derived performance results is quite sophisticated, it should be emphasized that the results themselves have a simple expression and provide clear insights into the nature of congestion and its impact on quality of service. We also provide an interpretation of these results in terms of bandwidth provisioning criteria and examine the important issue of performance under overload.

We begin by summarizing known characteristics of IP traffic at packet, flow and session levels. Detailed packet level simulations are then used to demonstrate how the slow start and congestion avoidance algorithms of TCP realize bandwidth sharing on a single bottleneck link. We then recall results for analytical fluid flow models derived under the assumption of Poisson arrivals and demonstate that these accurately predict the simulation results. It is in the following section that we apply results from the theory of stochastic networks to show how these models can be extended to account for very general and realistic flow arrival processes. Application of these models to deduce end-to-end performance is then discussed. Finally, we consider statistical bandwidth sharing on an overloaded link, evaluating the broad impact of user impatience and reattempt behavior on realized flow throughput and link goodput.

2. TRAFFIC CHARACTERISTICS

In this section we recall the known statistical characteristics of elastic traffic. Following a discussion on stationarity we present traffic characteristics in terms of packets, flows and sessions, respectively.

2.1 A stationary process

Traffic on network links averaged over a period of 5 to 10 minutes typically exhibit systematic variations as depicted in Figure 1. Intensity variations follow a certain daily pattern with a clearly identifiable busy period. During this period which can last several hours traffic intensity measured in bits/sec is approximately constant.

In the following sections we model traffic as a stationary stochastic process. This means we assume that traffic intensity remains constant for an indefinite period allowing the estimation of performance criteria as expected values. This is a classical approximation for which the main justification is the "eyeball" constancy of busy period traffic levels. We expect average performance measures derived under the stationarity assumption to be good approximations for the performance actually realized during a particular busy period.

Traffic intensity may be interpreted as the product of a packet arrival rate and the average packet size or in terms of higher level entities such as flows or sessions, as discussed below. The stationarity assumption applies equally to the arrival process of packets, flows and sessions.

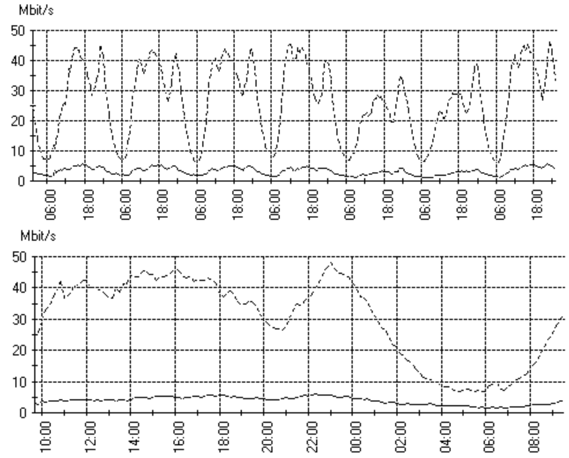

Figure 1: Weekly and daily utilization of a 155 Mbit/s link (both directions)

2.2 Packet level characteristics

It is now well known that Internet traffic at packet level is extremely variable over a wide range of time scales. This variability is manifested by asymptotic self-similarity and multi-fractal behavior at time scales smaller than that of the round trip time. A plausible expanation for self-similarity is the heavy-tailed nature of the size distribution of transferred documents while multifractal behavior appears to be a manifestation of the burstiness induced by TCP congestion control [17, 16].

The complexity of the packet arrival process is such that it proves very difficult to derive a packet level traffic characterization which is useful for performance modeling. We note further that most performance measures of interest for elastic traffic invoke higher level entities like the flow or the session such that it is more important to be able to describe traffic in these terms.

2.3 Flow level characteristics

A flow in the Internet is a loosely defined object representing a stream of packets having some criteria in common (IP addresses, port numbers,...). We use the term somewhat more restrictively to represent the packets corresponding to the transfer of a particular document. The document in question might be a Web page, an in-line object, a data file or an MP3 track. The defining feature is that the flow is manifested by a more or less continuous stream of packets using a considered network link or path. The flow may be realized by several overlapping TCP connections pertaining to the same document or by one period of activity of a sporadically used long-term TCP connection. It is characterized by its starting time and its size in bits. It may additionally be qualified by parameters such as the round trip time (RTT) or other external factors affecting the bandwidth it obtains on a shared link.

Measurements of the size of documents such as Web pages and FTP files show that their distribution has a heavy tail [10, 28]. The precise distribution clearly depends on the type of document considered. A reasonable fit to the form of the heavy tail is provided by the Pareto distribution:

$$\Pr[\text{size} \leq x] = 1 - \frac{k}{x^{\beta}}, \text{ for } x \geq k, \qquad (1)$$

with $1 < \beta \leq 2$, this distribution having a finite mean and infinite variance. The distribution has the property that a majority of flows are very small while most of the traffic in bytes is contained in large flows. We adopt the familiar shorthand of referring to very short flows as “mice” and to very long flows as “elephants”.

In many cases flows are generated within sessions. The flow arrival process thus tends to be bursty and has indeed been shown to be self-similar in certain cases [28, 15], a plausible explanation being that the number of flows per session has a heavy-tailed distribution. It may nevertheless be appropriate in certain circumstances to suppose flows arrive according to a Poisson process. This would be the case, for example, when flows correspond to a large number of independent sessions and the spacing of flows within a session is large compared to the average inter-flow interval.

2.4 Poisson session arrivals

As for flows, it is not immediately obvious how one can unambiguously define a session. Sometimes the session can be identified with an ISP modem call [16] or an FTP session [28]. Some authors have arbitrarily defined a session by partitioning a set of flows according to the inter-flow interval: a new session is assumed to start when this interval exceeds a certain threshold [27]. In all cases it is noted that session arrivals in any period where the traffic intensity is approximately constant are accurately modeled by a Poisson process. This observation is not surprising since a Poisson process is known to result from the superposition of a large number of independent user processes each of low relative intensity.

For present purposes, we consider a session to be composed of a set of flows whose statistical properties (arrival time, size,...) are independent of those of flows of any other session. We also assume that users generate sessions independently. If traffic intensity is constant and no single user contributes an excessive amount of traffic, the latter independence assumption will naturally lead to a stationary Poisson session arrival process whenever the number of users is large. The structure of a session is highly complex and varies depending on the underlying applications (Web, e-mail, FTP, etc.). Generically, it is composed of a succession of flows separated by an interval of inactivity which we call “think-time”.

In the next two sections we evaluate statistical bandwidth sharing on an isolated link assuming Poisson flow arrivals. This assumption simplifies analysis and provides useful insight. In the following section it is shown that most results derived for Poisson flow arrivals are also true with the weaker assumption of Poisson session arrivals.

3. BANDWIDTH SHARING REALIZED BY TCP

To gain insight into the way TCP realizes bandwidth sharing in statistical traffic, we have conducted a number of simulation experiments using the ns2 simulator [1]. These simulations illustrate the impact of packet level dynamics on the performance of bandwidth sharing at flow level, each flow being here assimilated to a TCP connection.

3.1 Simulation model

The simulated model is very simple consisting of just one bottleneck link handling packets from a certain number of TCP connections delivered to the link via a droptail buffer with a capacity of 50 packets. All packets are 1000 bytes long and the connections have a maximum receive window of 40 packets.

Statistical traffic variations result from an assumed Poisson connection arrival process and a Pareto size distribution (1) with parameters $\beta = 1.5$ and $k = 15$ packets. We have confirmed by other non-reported experiments that the exclusion of very small flows (less than 15 packets) does not affect the accuracy of reported results.

3.2 Slow start and the response time of “mice”

The response time of very small flows is constrained by TCP slow start. Even if the flow were alone on the link its throughput could not increase faster than allowed by a doubling of the congestion window every RTT. Assume the RTT is fixed and equal to RTT and that the transmission time of one packet is TR. Denote by P the bandwidth-delay product RTT/TR and by W the advertised receive window. Assume packets are acknowledged individually and that none are lost.

Let $R(n)$ be the time necessary to transfer and acknowledge n packets. $R(n)$ first increases geometrically until either the receive window is attained or the link is saturated and then increases at constant rate. Consideration of the schedule of packet emissions yields the following relations for $R(n)$. For $n < n^{\star} = 2^{\lceil \log_2 \min\{W,P\} \rceil + 1}$,

$$R(n) = (\lfloor \log_2 n \rfloor + 1)\,\text{RTT} + (n - 2^{\lfloor \log_2 n \rfloor})\text{TR}. \qquad (2)$$

[1] http://www.isi.edu/nsnam/ns/

For $n \geq n^\star$,

$$R(n) = \begin{cases} R(n-W) + \text{RTT}, & \text{if } W < P \\ R(n-1) + \text{TR}, & \text{if } W \geq P \end{cases} \quad (3)$$

The throughput $n/R(n)$ thus increases like $n/\log_2 n$ before flattening off and tending to the limit $\min\{W/\text{RTT}, 1/\text{TR}\}$ as $n \to \infty$.

3.3 Congestion avoidance and the throughput of "elephants"

The response time of very large flows, or elephants, depends more on TCP congestion avoidance than on the initial slow start phase. As long as the packet loss rate p is not too high and the receive window is not limiting, the throughput B achieved by a permanent flow is given by the approximate relation:

$$B(p) \approx \frac{K}{\text{RTT}\sqrt{p}} \quad (4)$$

where K is a constant that depends on second-order statistics of the loss process ($K = \sqrt{3/2}$ for periodic losses); see [1] and cited references.

When several permanent TCP connections use the same bottleneck link, the effect of congestion avoidance is to share the link bandwidth between them. Assuming all flows experience the same loss probability p, relation (4) suggests that bandwidth is shared in inverse proportion to RTT. We have verified this by an ns simulation of 20 connections sharing a link of 10 Mbit/s, 10 with an RTT of 50 ms and 10 with an RTT of 100 ms, together with 1 Mbit/s of on-off UDP traffic (included to attenuate undesirable synchronization effects due to the artificial homogeneity of simulated connections). Results in Table 1 confirm that on average the connections indeed share bandwidth in proportion to RTT and that there is no wasted bandwidth. For these results we have discarded the initial slow start phase before calculating the average throughput over a simulated duration of 1 minute.

RTT (ms)	Throughput of TCP flows (Kbit/s)	Total (Mbit/s)
50	646 570 578 605 577 629 642 592 535 667	6.04
100	273 277 376 352 248 320 311 306 252 288	3.00

Table 1: Bandwidth sharing of a 10 Mbit/s link with persistent UDP and TCP flows

It is possible to invert (4) to derive a relation $p(B)$. In other words, if B is known then we can deduce the packet loss rate p. Now, if we assume that TCP is efficient in using all the link capacity C and that each connection receives a share inversely proportional to its RTT, we can deduce the packet loss rate ($p = (\sum K/C\text{RTT}_i)^2$). This observation is significant in that it suggests that it is not necessary to take account of the complex packet arrival process discussed in Section 2. The loss rate and the multifractal scaling behavior both result from the way the congestion avoidance algorithm shares link bandwidth.

3.4 Statistical bandwidth sharing

Consider now the impact of random fluctuations in the number of TCP connections. In the rest of the section, all figures are bubble diagrams with each bubble depicting the realized throughput of a flow as a function of its size. The size given on the x-axis is also represented by the cross-section of the bubbles in order to reflect the relative weights of mice and elephants.

Non-limiting receive window, same RTT

Figure 2 presents results for a 1 Mbit/s link shared by users all having the same RTT. The curved envelope appearing in the top left hand corner is due to the rate limiting effect of slow start as described in Section 3.2. The figure shows that flow throughput for mice tends to be highly variable while that for elephants is more stable. Note that the dispersion of realized throughputs is roughly symmetrical about a size invariant mean of around 500 Kbit/s.

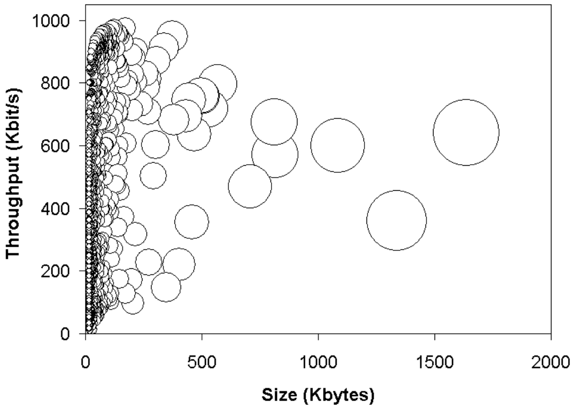

Figure 2: Throughput of TCP transfers (1 Mbit/s link, demand of 500 Kbit/s)

Limiting receive window, same RTT

Figure 3 corresponds to traffic on a link of 10 Mbit/s. In this case, flow throughput is limited mainly by slow start and the size of the receive window. An assumed RTT of 100 ms (40 ms maximum queuing delay + 60 ms fixed delay) with the 40-packet receive window would give a maximum bandwidth of 3.2 Mbit/s for a very long connection. Realized throughput is mainly limited by the envelope (2) and (3) which is not strictly defined, however, due to the fact that RTT is not constant.

Limiting receive window, different RTT

Figure 4 relates to a 10 Mbit/s link shared by two classes of connections distinguished by their maximum RTT: 50 ms and 100 ms, respectively. Each class contributes a demand of 4 Mbit/s. The darker bubbles correspond to the longer RTT connections which, as expected, achieve lower throughput. This discrimination is due both to the different slow start envelopes (2) and (3) and to the different shares obtained in congestion avoidance according to (4).

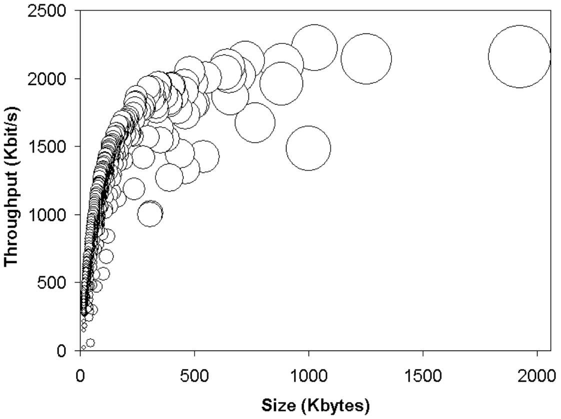

Figure 3: Throughput of TCP transfers (10 Mbit/s link, demand of 5 Mbit/s)

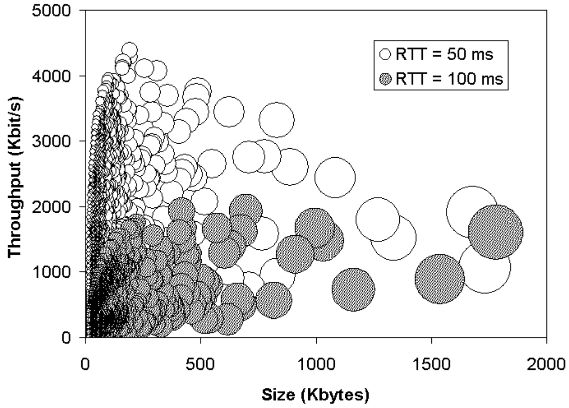

Figure 4: Throughput of TCP transfers with different RTT (10 Mbit/s link, demand of 8 Mbit/s)

The throughput of elephants of both types converges to a value somewhat less than 2 Mbit/s.

Prior to discussing the significance of these results with respect to statistical bandwidth performance achieved in a network, we present in the next two sections a number of mathematical models which help to explain the observed behavior.

4. MODELING STATISTICAL BANDWIDTH SHARING

Consider an isolated link of capacity C and assume flows arrive as a Poisson arrival process of rate λ with a size drawn independently from a common distribution of mean σ. Flows are modeled as a fluid whose rate adjusts instantly in response to changes in the number of flows in progress.

4.1 Fair sharing bottleneck

If all flows have the same RTT, TCP tends to share bandwidth equally among the flows in progress, at least for the larger flows. We assume here that fair sharing is realized immediately for all flows whatever their size. Let $\pi(n)$ be the probability n flows are in progress at an arbitrary instant and let $R(s)$ be the expected response time of a flow of size s. Let $\rho = \lambda\sigma/C$ denote the link load and assume $\rho < 1$.

With Poisson flow arrivals, the number of flows in progress behaves like the number of customers in an M/G/1 processor sharing queue [22] and we have immediately the well known results:

$$\pi(n) = \rho^n(1-\rho), \tag{5}$$

$$R(s) = \frac{s}{C(1-\rho)}. \tag{6}$$

Note that $R(s)/s$ is the mean of the inverse of the throughput received by flows of size s. Thus $\gamma(s) = s/R(s)$ is the *harmonic* mean throughput of flows of size s. For the present system $\gamma(s)$ is constant and equal to $C(1-\rho)$. The latter expression thus also represents the ratio *expected size* to *expected response time*, a measure of overall throughput performance [2].

It may be verified from the results of Figure 2 that $\gamma(s)$ provides a good approximation for the mean throughput achieved by TCP, $C(1-\rho)$ in this case being precisely 500 Kbit/s. While the mean value constitutes a useful estimate for the throughput of long flows, the distribution for shorter flows is more widely dispersed. This is not surprising since the throughput of mice is essentially determined by the (highly variable) number of flows present at their arrival. On the other hand, the throughput of elephants, which use all the capacity not used by other, shorter flows, is approximately equal to the residual capacity $C(1-\rho)$.

Figure 5 presents results comparable to those of Figure 2 derived from a simulation of the considered fluid system. Comparison of the figures confirms that the fluid model yields approximately the same behavior as TCP induced statistical sharing with the notable exception of the impact of slow start on the throughput of short flows.

The above formulas are insensitive to the nature of the flow size distribution. This is a highly significant result since it shows that first order bandwidth sharing performance is largely independent of this traffic characteristic.

4.2 Fair sharing with limited rate

The maximum throughput of flows on a network link is frequently limited by external constraints such as the user's modem speed, server capacity, bandwidth on other network links or the TCP receive window size, as illustrated in Figure 3. Assume all users have a common maximum rate limit $r < C$. This bandwidth sharing model can be recognized as a generalization of the processor sharing queue considered by Cohen in [9]. Corresponding results derived therein for $\pi(n)$ and $R(s)$ are as follows:

$$\pi(n) = (1-\rho)f(\rho) \times \begin{cases} \frac{m!}{n!}(\frac{\rho C}{r})^{n-m}, & \text{for } n < m, \\ \rho^{n-m}, & \text{for } n \geq m, \end{cases} \tag{7}$$

$$R(s) = s\left(\frac{1}{r} + \frac{f(\rho)}{C(1-\rho)}(1 - (\frac{C}{r} - m)(1-\rho))\right) \tag{8}$$

[2] For a discussion of alternative measures of throughput performance see [21].

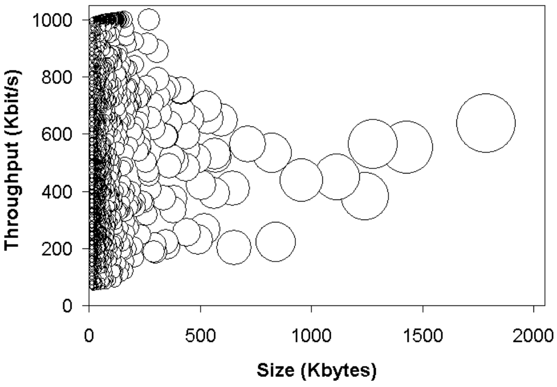

Figure 5: Throughput of flows (1 Mbit/s fair sharing link, 500 Kbit/s demand)

where m denotes the integer part of C/r and

$$f(\rho) = \frac{(\frac{C\rho}{r})^m/m!}{(1-\rho)\sum_{k=0}^{m-1}(\frac{C\rho}{r})^k/k! + (\frac{C\rho}{r})^m/m!}$$

is the probability the link is saturated.

Again, the mean throughput $\gamma(s) = s/R(s)$ does not depend on the flow size s. Figure 6 shows how γ depends on ρ with a rate limit $r = C/10$. It is clear from the figure that throughput on a high capacity link for which $r \ll C$ is equal to r except when the offered load ρ is very close to 1. We have the approximation: $\gamma \approx \min(r, C(1-\rho))$.

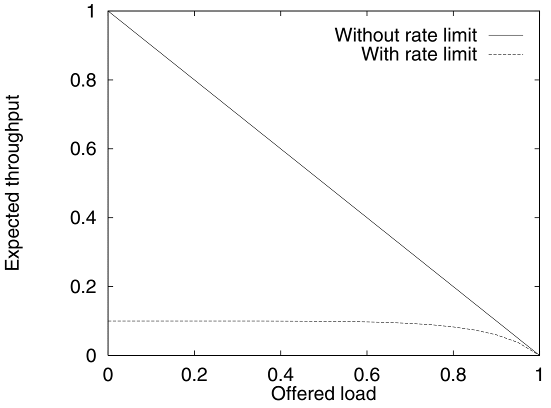

Figure 6: Expected normalized throughput against offered load in case of fair sharing

The above results are again insensitive to the flow size distribution. Unfortunately, this convenient property is lost if we wish to account for the fact that the rate limit is generally different for every flow and can vary during the transfer, or that bandwidth sharing is not perfectly fair. It is likely, however, that (8) still provides a good approximation for the expected response time, at least for elephants. This may be verified on the simulation results of Figure 4, for instance. The rate limit imposed by the receive window $W/$RTT is equal to 6.4 Mbit/s if RTT $= 50$ ms and 3.2 Mbit/s if RTT $= 100$ ms. Corresponding mean throughputs derived from (8) for each rate limit are 1.9 Mbit/s and 1.6 Mbit/s, respectively. These values correspond reasonably well to the throughput of elephants as shown in the figure.

4.3 Unequal sharing

In practice, bottleneck bandwidth is not shared perfectly fairly. One reason is the impact of different round trip times (see Fig. 4). Another is the fact that some flows may be transported by more than one TCP connection (e.g., with HTTP 1.0). To fully explore the implications of unequal sharing is beyond present scope. We note simply that evaluations using simulation and a discriminatory processor sharing model [14] reveal the following (see [6, 29]):

- discrimination in realized throughput is significant mainly at loads close to saturation;
- size dependent throughput $\gamma(s)$ is roughly the same for all s except for the very largest documents whose throughput tends to $C(1-\rho)$;
- throughput performance is roughly insensitive with respect to the document size distribution;
- naturally, the access rate r plays a significant equalizing role when $r \ll C$.

These observations suggest that the broad conclusions we draw from idealized equal sharing models are likely to be true also under more realistic assumptions of discriminatory sharing.

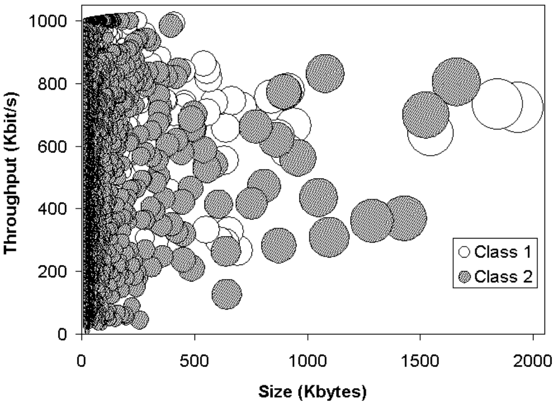

Figure 7: Throughput of flows (1 Mbit/s unequal sharing link, 500 Kbit/s demand, 10:1 bias)

Figure 7 illustrates the imprecise discrimination realized when flows of two classes of equal intensity share a bottleneck link with 50% load: class 1 flows receive 10 times the rate of concurrent class 2 flows. Despite this clear bias, the throughput performance realized by the two classes is quite close. The mean throughput predicted by the model of [14] is 690 Kbit/s for class 1 and 390 Kbit/s for class 2

[6]. Statistical variations obscur this difference, particularly for mice.

4.4 A transparent backbone link

We finally consider the case where the link capacity C is very large compared to the external rate limits and such that it is virtually transparent. By this we mean the probability of the sum of external rate limits of all flows in progress exceeding link capacity C is negligibly small. This assumption is reasonable for the large, moderately loaded links of major backbone providers.

The number of flows in progress is now unconstrained by the considered link which appears as an M/G/∞ queue. Flow duration is thus an independent random variable. Let θ be the mean duration and ν the mean number of flows in progress. By Little's law we have: $\nu = \lambda\theta$. The number of flows in progress has the Poisson distribution:

$$\pi(n) = \frac{\nu^n}{n!} e^{-\nu}, \text{ for } n = 0, 1, ... \tag{9}$$

Formula (9) is true for any flow duration distribution and thus for an arbitrary flow size distribution and rate limit.

5. ACCOUNTING FOR THE REAL FLOW ARRIVAL PROCESS

In this section we show that most of the analytical results derived above for Poisson flow arrivals also apply under much more realistic traffic assumptions. Our models apply to a fair sharing bottleneck link, with or without a common rate limit, or a transparent backbone link, as introduced in the previous section.

5.1 A stochastic network

The succession of document transfers and think-times constituting a session may be represented as a customer visiting two stations in a stochastic network of the kind considered notably by Kelly [19][3] (see Figure 8). The first and last station to be visited is the link, and successive visits to the link are separated by a visit to a think-time station. Outside arrivals are Poisson and every customer eventually leaves the network.

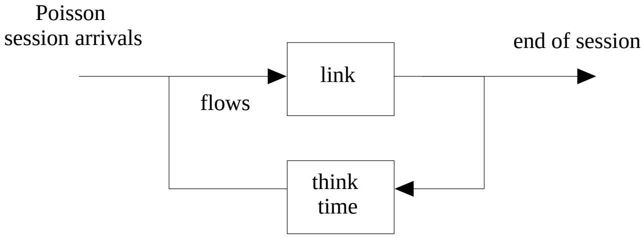

Figure 8: Flow arrivals modelled as a stochastic network

An essential characteristic of a customer in a stochastic network is its "class". This is a versatile attribute which allows us to distinguish different kinds of customer as well as the number of times the customer has previously visited the current station. In the present context we use the class to specify just the distribution of the customer's current service requirement and, either the fact that it leaves the network when it finishes that service, or the class it will acquire at the next station to be visited. In general, the customer changes class when it changes station.

Recall that the considered bottleneck or transparent backbone links have the property that, if customer arrivals are Poisson, the distribution of the number of customers present is insensitive to the form of the distribution of service requirements. The station representing the think-time also has this insensitivity property. In the terminology introduced by Kelly, all network stations are "symmetric queues" [19, Chapter 3].

The latter property coupled with the assumed class mechanism and Poisson customer arrivals from outside allows us to deduce that Theorems 3.7 and 3.10 in [19] apply to the considered network. Among other results, these theorems state that the distribution of the number of customers at each station is distributed *as if* all customer arrivals, including repeated visits, constitute a Poisson process. The implication for statistical bandwidth sharing performance is that as long as session arrivals are Poisson, the distribution of the number of flows in progress is given by (5), (7) or (9) for any flow arrival process which can be represented by the assumed class mechanism. This mechanism is sufficiently versatile to reproduce most of the observed characteristics of IP traffic at flow level, as explained below.

5.2 A general ¤ow arrival process

The set of classes is typically very large and may, in general, be countably infinite. It is natural, for example, to use distinct classes to represent different types of session (Web, FTP, e-mail,...). To distinguish successive flows within a session it is further necessary to attribute a new distinct class for each visit to a given station.

The class mechanism can be used to account for a particular distribution of the number of flows per session. Let λ be the overall arrival intensity of a particular type of session and denote by $p(i)$, for $i \geq 1$, the probability a session contains i flows. We define a distinct class c_{ij}, for $i \geq 1$ and $j < i$, for the j^{th} flow of an i-flow session. Sessions of class c_{i1} arrive according to a Poisson process of intensity $\lambda p(i)$ and mutate through classes $c_{i2}, c_{i3}, ..., c_{ii}$ before leaving the system. The distribution $p(i)$ clearly impacts the nature of the flow arrival process. It may be shown in particular that if the distribution $p(i)$ is heavy-tailed then the overall flow arrival process will be self-similar [7].

The flow size distribution can change depending on its position in the session. Imagine searching through a series of Web sites, generating small flows at each click, before downloading a possibly voluminous document as the last flow of the session. The flows corresponding to classes c_{ii} (the last visit to the link station) for such sessions would have a correspondingly larger mean size.

Dependence between successive flow sizes and think-times can be introduced by further subdividing the number of distinct session types. Assume for example that the size of the first flow of a given type of session is drawn from a

[3] Broadly equivalent results can be derived using the alternative formalisms of [2] or [9].

certain distribution and that the ensuing think-time is positively correlated with this flow size (a user spends more time looking at a complex Web page, say). This dependence can be accounted for by distinguishing initial customer classes according to the size of the first flow with distinct distributions for the subsequent think-time.

It is not suggested that in practice one would wish to distinguish so many different classes and sub-classes. The beauty of the stochastic network result is that the distribution of the number of flows in progress (of *any* class) does not depend on the underlying session structure but only on the overall mean traffic intensity. The notion of class is simply a device allowing us to verify this invariance by applying the above mentioned theorems due to Kelly.

5.3 Expected response time

We have so far only mentioned the insensitivity of the distribution of the number of flows in progress. The same property holds for the bottleneck response time results presented in the previous section.

By Theorem 3.10 in [19], the probability a given customer in a fair sharing station is of class c is equal to a_c/a where a_c is the demand (arrival rate × mean size) due to class c and a is the overall demand ($a = \rho C$). Suppose now that class c identifies customers with service requirement s (i.e., flows of size s)[4] and that their arrival intensity is λ_c so that $a_c = \lambda_c s$. The expected number of class c customers present is $E[n_c] = a_c/a \times E[n]$ where $E[n] = \sum_n n\pi(n)$ is the expected total number of customers present in the link station. Applying Little's formula, the expected response time is $E[n_c]/\lambda_c$. It may readily be verified by evaluating this expression with $\pi(n)$ given by (5) and (7) that we indeed have expressions (6) and (8) for the expected transfer time on a bottleneck link with and without rate limitation, respectively.

5.4 Finite user population

The Poisson session arrival assumption is appropriate when the considered link can be used by a very large number of users. In an access network or LAN it may be more approriate to account for the finite size of the user population. This is the assumption in the studies of statistical bandwidth sharing by Heyman et al [12] and Berger and Kogan [4]. The stochastic network approach allows us to generalize their results to account for more realistic traffic assumptions.

It is necessary to assume the succession of document transfers and think-times corresponding to the activity of a given user constitutes a stationary process. Let σ be the mean document size and τ the mean think-time. Assume we have N users, all with the same σ and τ. In the case of a bottleneck link without rate limitation, the probability $\pi(n)$ that n flows are in progress is given by:

$$\pi(n) = G(N)\frac{\sigma^n \tau^{N-n}}{(N-n)!} \qquad (10)$$

where $G(N)$ is a normalization constant. Similar results are available for the other types of link considered above. It is also possible to generalize (10) to account for different types of user and to mix finite user populations of a certain type with Poisson arrivals of other sessions of other types. The only condition is that the flows of all types are treated equitably.

Theorem 3.12 in [19] for closed stochastic networks shows that these relations hold true under very general assumptions regarding the structure of user activity: sessions of different types, general distribution for the number of flows per session, dependencies between successive flow sizes and think-times,... It is further possible to demonstrate that the expected response time is again proportional to the document size.

5.5 Extensions and generalizations

It is possible to generalize the model in certain directions, by including limits on the number of flows or sessions in progress, for example. Of course, as in the case of a link with Poisson flow arrivals, any modification which removes the "symmetry" of the assumed sharing policy (unequal shares, different rate limits,...) immediately destroys the analytical insensitivity property. It is likely, however, that an approximate insensitivity will be retained as in the case of Poisson flow arrivals.

6. END-TO-END PERFORMANCE

It is important, notably when designing provisioning procedures, to understand how statistical bandwidth sharing on all the links of a network path combines to determine end-to-end performance.

6.1 Notions of fairness

First consider the different possibilities for sharing bandwidth in a network when the number of connections is fixed. We represent the network as a set of links $\mathcal{L}$ with link l having capacity C_l. A route corresponds to a particular subset of links. Assume we have x_r permanent flows on route r and let B_r denote the bandwidth allocated to each of these flows. The B_r must satisfy the capacity constraints:

$$\sum_{r \ni l} x_r B_r \leq C_l, \quad \text{for } l \in \mathcal{L}. \qquad (11)$$

The issue of what constitutes a fair bandwidth allocation has been discussed in a number of recent papers [20, 26, 24]. A significant range of fairness notions can be expressed as the solution to the following optimization problem [26]:

$$\text{Maximize} \sum_r w_r x_r \frac{B_r^{1-\alpha}}{1-\alpha}, \text{ subject to constraints (11)}, \qquad (12)$$

where $\alpha \neq 1$ is a positive constant and the w_r are weights. This may be considered as a utility maximization problem with particular interpretations for the utility of individual allocations. Classical max-min fairness [5] arises in the limit $\alpha \to \infty$ with weights $w_r \equiv 1$, while weighted proportional fairness as discussed in [20] occurs as $\alpha \to 1$. As α decreases from ∞ towards 0, the different allocations give relatively less bandwidth to resource intensive long routes.

It is of considerable interest to understand what kind of allocation is achieved by the congestion avoidance algorithm of TCP. Let p_l be the packet loss rate on link l.

[4] We suppose for the sake of simplicity that demand from flows of size s is non-zero.

Assuming the p_l are small and the receive window does not limit throughput, expression (4) gives:

$$B_r \approx \frac{K}{\text{RTT}_r \sqrt{\sum_{l \in r} p_l}} \qquad (13)$$

Using the fact that $p_l > 0$ implies $\sum_{r \ni l} x_r B_r = C_l$, it may readily be verified by applying the Kuhn-Tucker theorem that the allocation satisfying (13) is the unique solution to optimization problem (12) with $\alpha = 2$ and $w_r = 1/\text{RTT}_r^2$ (see [30] for further discussion on the type of bandwidth sharing realized by TCP).

6.2 Statistical bandwidth sharing

Assume now that flows on route r arrive at rate λ_r and have mean size σ_r. Let $\rho_l = \sum_{r \ni l} \lambda_r \sigma_r / C_l$ denote the load offered to link l. We assume bandwidth is instantaneously allocated to solve (12) as the x_r change. A first question of interest is that of stability. Not surprisingly, as shown in [6], the number of flows on each route remains finite provided the usual traffic condition $\rho_l < 1$ for $l \in \mathcal{L}$ is satisfied [5].

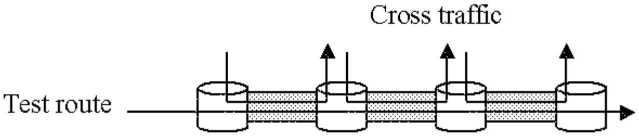

Figure 9: A 3-link linear network

It proves difficult to extend the performance results established for an isolated link to the case of a general network. A notable exception is the homogeneous linear network depicted in Figure 9 consisting of L links of capacity C shared by one end-to-end test route and L single hop routes. It was shown in [25] that it is possible to calculate the distribution of the number of flows in progress when the flow arrival process is Poisson and the bandwidth allocation realizes proportional fairness. The reasoning applied in Section 5 allows us to affirm that these results are also true with an assumption of Poisson *session* arrivals and general session structure. The mean throughput $\gamma(s)$ of flows of size s on the test route is independent of s. Denoting by ρ_r the offered load on this route, we find [6]:

$$\gamma = \frac{C(1-\rho_r)}{1 + \sum_{l=1}^{L} \frac{\rho_l - \rho_r}{1 - \rho_l}}. \qquad (14)$$

Simulations of networks sharing bandwidth according to different notions of fairness suggest the impact on statistical performance is slight for $\alpha \geq 1$ (i.e., ranging from proportional to max-min fairness). This is illustrated by the results in Figure 10 pertaining to a 3-link linear network with the same load on every route. The slight difference disappears almost completely when flows are also constrained by a maximum rate limit, as shown in the figure for a limit of $r = C/10$. The assumption of equal link loading may

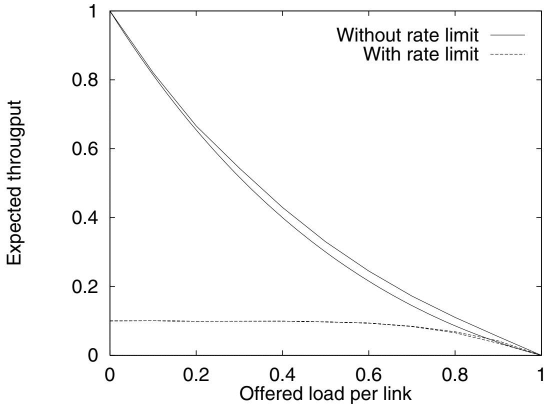

Figure 10: Expected normalized throughput on the test route against offered load for max-min fairness (top) and proportional fairness (bottom)

be considered to constitute a worst case for testing the hypothesis that the fairness notion does not significantly affect performance. Most network paths will have a clearly identifiable bottleneck whose bandwidth is shared identically with any choice of α in (12). As in the case of an isolated bottleneck, end-to-end throughput is generally well approximated by the expression $\gamma \approx \min(r, \min_{l \in r} C_l(1 - \rho_l))$.

6.3 Network provisioning

An important conclusion to be drawn from the previous results is that performance is generally satisfactory in a classical best effort network as long as link load is not too close to 100%. While the bandwidth achieved by small flows may vary widely due to random traffic fluctuations, as seen in the simulation results in Section 3, the throughput of a long flow is generally only loosely constrained by the expected available capacity $C_l(1 - \rho_l)$. By ensuring that this is appreciably greater than the limit imposed by the users' access network, a backbone link can be made virtually transparent.

On the other hand, it would be vain for a provider to make more precise guarantees with respect to throughput performance. Even when the mean throughput is controlled, performance of individual flows varies widely due to random fluctuations in the number of concurrent flows, as illustrated in the simulation results of Sections 3 and 4. In normal load conditions, this statistical effect is at least as significant as the deterministic discrimination due to different sharing weights (see Figure 7).

7. PERFORMANCE IN OVERLOAD

While providers generally aim to avoid congestion by adequate provisioning, it is clear that overload can occur on certain links. This may be due to planning errors, outages, traffic surges or deliberate underprovisioning of costly transoceanic routes or strategic peering links. We propose some simple preliminary models for an overloaded link accounting for user impatience and reattempt behavior.

[5] An alternative proof for the case of max-min fairness is given in [11]

7.1 Flow level congestion

Flows of mean size σ arrive at a link of capacity C at mean rate λ. When $\rho > 1$, the number of flows in progress tends to increase and their throughput tends to zero since the arrival rate λ is greater than the average rate of flow completion. This behavior is illustrated in Figure 11 which shows results from an ns simulation of a 10 Mbit/s link under 20% overload. The link is empty at time zero and flow sizes have a Pareto distribution. The dots represent the throughput realized by flows at the instant of their completion.

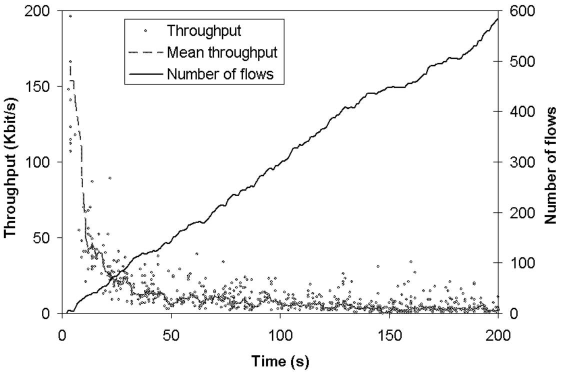

Figure 11: Bandwidth sharing performance during transient overload

It is interesting to note that the rate of increase in the number of flows actually depends significantly on the form of the size distribution [18]. The rate of increase is smaller as the proportion of mice is greater. The latter manage to complete although their throughput continues to decrese, while the response time of elephants tends rapidly to infinity. Thus, the heavy-tailed nature of flow sizes here may be said to have a positive impact on performance, albeit at the cost of an unfortunate discrimination against elephants.

Note that the degree of congestion on a network link may not be easy to detect simply by observing the packet arrival process. The packet arrival rate is controlled by TCP, ideally to a value close to the service rate, even though the number of contributing flows may be increasing rapidly. Elastic traffic congestion is manifested essentially at flow level rather than at packet level.

7.2 Impatience

In a real network, if demand exceeds capacity, the number of flows in progress does not increase indefinitely. As per-flow throughput decreases some flows or sessions will be interrupted, due either to user impatience or to aborts by TCP or higher layer protocols. In the following we use the term impatience for all causes of premature abandon.

We are unaware of any published statistics on user impatience. The phenomenon is clearly very difficult to observe in practice, notably because all flow aborts are not in reaction to excessive response time and because most impatience is manifested by the interruption of a session and may not be detectable as an abnormal event. However, to gain some insight into the phenomenon we propose below a simple hypothetical model.

We suppose a flow of size s will be interrupted if and when its response time exceeds a patience duration $\delta(s)$. It seems natural to assume that δ is an increasing but concave function of s since users have a response time expectation which increases with the flow size but need proportionally more throughput. We have observed in simulations that such impatience causes the number of flows in progress on a large capacity link to stabilize and vary slightly about a certain mean value. To simplify we assume here that the number is exactly constant so that each flow receives the same bandwidth share θ. A flow of size s is then completed if and only if $s \leq \theta\delta(s)$.

It follows from the concavity of δ that there exists a critical flow size $s^\star$ satisfying $s^\star = \theta\delta(s^\star)$ such that any flow of size smaller than $s^\star$ is completed while all the others are aborted. We can determine θ by arguing that, since the link is always saturated, we must have:

$$C = \lambda \int_0^\infty \min(s, \theta\delta(s)) dF(s) \qquad (15)$$

where $F(s)$ is the flow size distribution. The following simple closed formula may be derived from (15) in the particular case of Pareto distributed flow sizes (1) and constant patience duration, $\delta(s) = \delta$:

$$\theta = \frac{k}{\delta} \left(\frac{\rho}{\beta(\rho - 1)} \right)^{\frac{1}{\beta - 1}}. \qquad (16)$$

It is further possible in this case to derive the link goodput $U = \lambda \int_0^{s^\star} s dF(s)/C$, i.e., the fraction of link capacity used by flows that are effectively completed:

$$U = 1 - (\beta - 1)(\rho - 1) \qquad (17)$$

Note that link goodput may be equal to zero, meaning all flows are interrupted, in the extreme situation where the overload exceeds $1/(\beta - 1) \times 100\%$.

This model provides some useful insights into the impact of congestion. The results for constant patience are qualitatively representative of evaluations performed with different patience functions $\delta(s)$. Both realized throughput and link goodput deteriorate with increasing load, but are otherwise independent of link capacity. Realized throughput also decreases as users become more patient while goodput stays the same. On the other hand, confirming the positive impact of a heavy-tailed size distribution noted in Section 7.1, both throughput and goodput improve as β decreases from 2 to 1. This is explained by the fact that impatience discriminates against elephants and interrupts these large flows after only a small fraction of their data has been tranferred.

	analysis		ns simulation	
patience	throughput	goodput	throughput	goodput
10 s	23.8 Kbit/s	55%	18.5 Kbit/s	55%
20 s	11.9 Kbit/s	55%	9.9 Kbit/s	51%

Table 2: Per-flow throughput and link goodput during sustained overload

Table 2 compares evaluations of (16) and (17) with the results of ns simulations under the same impatience assumptions for a 1 Mbit/s link under 90% overload.

7.3 Reattempts

Aborted flows are not generally abandoned immediately as users will frequently make a repeat attempt. The impact of this behavior is to exacerbate the loss of goodput due to impatience as it is likely that the reattempts will also be interrupted. Consider the following simple model.

User behavior is modeled by a size dependent patience duration as introduced in the previous section. If a user aborts it reattempts with fixed probability p. Reasoning as above, in place of equation (15), the maximum completed flow size $s^\star$ now satisfies:

$$C = \lambda \int_0^{s^\star} s.dF(s) + \frac{\lambda}{1-p} \int_{s^\star}^{\infty} \theta\delta(s) dF(s). \qquad (18)$$

Figure 12 plots goodput U against p in the case of a 20% overload assuming constant patience duration and a Pareto size distribution.

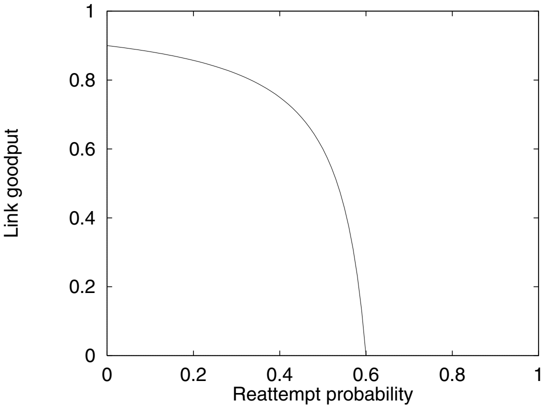

Figure 12: Impact of reattempts on link goodput

The figure shows that the loss of goodput can be considerable. While the model is overly simple, it does illustrate the negative impact user behavior can have in case of overload. Network efficiency would gain from a more proactive reaction to congestion.

7.4 Admission control

An alternative to allowing an overloaded link to stabilize through impatience is to perform admission control [23]. If flows arriving when the bandwidth they would achieve is less than an acceptable threshold were rejected immediately, there would be no cause for impatience. The advantage is that goodput is maintained close to 100% and is unaffected by reattempts. On the other hand, admission control would tend to increase the proportion of uncompleted flows since it applies equally to both mice and elephants. Since elephants may well be more important than mice, this is not necessarily a disadvantage. Indeed, one advantage of admission control is that it can be applied selectively with different admission thresholds applying to different classes of traffic (see [3]).

8. CONCLUSIONS

To evaluate the throughput performance of elastic transfers it is necessary to account for the dynamic nature of traffic. Traffic variations are most naturally modeled in terms of flows and sessions rather than packets whose complex arrival process is largely determined by the closed-loop control of TCP connections. We have demonstrated that fluid flow statistical bandwidth sharing models can accurately predict the results of ns packet-level simulations.

Using results from the theory of stochastic networks we have shown, in a number of ideally fair bandwidth sharing scenarios, that the distribution of the number of flows in progress and the expected flow throughput have very simple expressions which are valid under a wide range of realistic traffic assumptions. These expressions depend essentially only on expected demand and are independent of such characteristics as the heavy-tailed flow size distribution or the self-similar flow arrival process. Further evaluations nott reported here lead us to believe that the broad conclusions derived under an assumption of ideal fair sharing remain true under moderate discrimination due to different RTT, for instance.

The expected flow throughput achieved on a link of capacity C bits/s with utilization ρ is roughly equal to the minimum of the residual capacity $C(1 - \rho)$ and any rate limit arising from external causes such as the bandwidth available on other links, the user's modem speed or the size of the advertised TCP receive window. It follows that performance is generally satisfactory as long as demand is somewhat less than capacity (in which case demand is equal to the measured load $C\rho$). This justifies the usual provisioning procedures based on limiting utilization in the busiest period while suggesting currently used limits of 60%, say, may be overly conservative.

The stochastic network models are unstable when demand exceeds capacity (the number of flows in progress would grow indefinitely). In practice, when overload occurs, network utilization is necessarily stabilized through user impatience and other reasons for aborting a session or a flow. Since an incomplete user transaction generally implies bandwidth wastage, impatience leads to goodput which may be appreciably less than capacity. According to a simple model of user behavior, overload also brings discrimination against larger flows which are less likely to sustain the resulting low throughput.

We suggest that the key to quality of service is to apply adequate provisioning procedures coupled with traffic routing strategies designed to avoid demand overload. There appears little scope for service differentiation beyond the two broad categories of "good enough" and "too bad". Rather than relying on impatience to stabilize an overloaded link leading to quality which is too bad, it would be preferable to perform admission control at flow or session level, maintaining sufficient throughput for admitted flows and avoiding bandwidth wastage on incomplete transactions.

9. REFERENCES

[1] E. Altman, C. Barakat, and K. Avrachenkov. A stochastic model of tcp/ip with stationary random losses. In *Proc. of ACM SIGCOMM'00*, 2000.

[2] F. Baskett, K.M. Chandy, R.R. Muntz, and F.G. Pallacios. Open, closed and mixed networks of queues with different classes of customers. *Journal of ACM 22*, pages 248–260, 1975.

[3] S. Ben Fredj, S. Oueslati-Boulahia, and J.W. Roberts. Measurement-based admission control for elastic traffic. In *Proceedings of ITC17*, September 2001.

[4] A. Berger and Y. Kogan. Dimensioning bandwidth for elastic traffic in high-speed data networks. *IEEE/ACM Trans Networks*, 8(5):643–654, October 2000.

[5] D. Bertsekas and R. Gallager. *Data Networks*. Prentice-Hall International, 1992.

[6] T. Bonald and L. Massoulié. Impact of fairness on internet performance. In *Proceedings of ACM SIGMETRICS'01*, 2001.

[7] T. Bonald, A. Proutière, G. Régnié, and J.W. Roberts. Insensitivity results in statistical bandwidth sharing. In *Proceedings of ITC17*, September 2001.

[8] T. Bu and D. Towsley. Fixed point approximations for TCP behavior in an aqm network. In *Proceedings of ACM SIGMETRICS'01*, 2001.

[9] J.W. Cohen. The multiple phase service network with generalized processor sharing. In *Acta Informatica 12*, pages 245–284, 1979.

[10] M. Crovella and A. Bestravos. Self-similarity in world wide web traffic: Evidence and possible cause. In *Proceedings of ACM SIGMETRICS'96*, 1996.

[11] G. De Veciana, T.J. Lee, and T. Konstantopoulos. Stability and performance analysis of networks supporting services with rate control - could the internet be unstable? In *Proceedings of IEEE INFOCOM'99*, 1999.

[12] D.P.Heyman, T.V.Lakshman, and A.L.Neidhardt. New method for analyzing feedback protocols with applications to engineering web traffic over the internet. In *Proceedings of ACM SIGMETRICS'97*, 1997.

[13] G. Fayolle, A. de la Fortelle, J-M. Lasgouttes, L. Massoulié, and J.W. Roberts. Best-effort networks: modeling and performance analysis via large networks asymptotics. In *Proceedings of IEEE INFOCOM'01*, 2001.

[14] G. Fayolle, I. Mitrani, and R. Iasnogorodski. Sharing a processor among many classes. *Journal of the ACM*, 27:519–532, 1980.

[15] A. Feldman. Characteristics of TCP connection arrivals. K. Park, W. Willinger, editors, *Self-similar network traffic and performance evaluation*, J. Wiley & Sons, 2000.

[16] A. Feldman, A. Gilbert, P. Huang, and W. Willinger. Data networks as cascades: Explaining the multifractal nature of internet WAN traffic. In *Proceedings of ACM SIGCOMM'98*, 1998.

[17] A. Feldman, A. Gilbert, P. Huang, and W. Willinger. Dynamics of IP traffic: A study of the role of variability and the impact of control. In *Proceedings of ACM SIGCOMM'99*, 1999.

[18] A. Jean-Marie and P. Robert. On the transient behavior of the processor sharing queue. *Queueing Systems Theory and Applications*, 17:129–136, 1994.

[19] F.P. Kelly. *Reversibility and Stochastic Networks*. J. Wiley & Sons, 1979.

[20] F.P. Kelly, A. Maulloo, and D. Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49, 1998.

[21] A.A. Kherani and A. Kumar. Performance analysis of TCP with nonpersistent sessions. Preprint available at http://ece.iisc.ernet.in/~anurag/Anurag_Kumar.html, 2000.

[22] L. Kleinrock. *Queueing Systems, Volume 2*. J. Wiley & Sons, 1975.

[23] L. Massoulié and J.W. Roberts. Arguments in favor of admission control for TCP flows. In P. Key and D. Smith, editors, *Teletraffic Engineering in a Competitive World, Proceedings of ITC 16*, pages 33–44. Elsevier, 1999.

[24] L. Massoulié and J.W. Roberts. Bandwidth sharing: objectives and algorithms. In *Proceedings of IEEE INFOCOM'99*, 1999.

[25] L. Massoulié and J.W. Roberts. Bandwidth sharing and admission control for elastic traffic. *Telecommunication Systems*, 15:185–201, 2000.

[26] J. Mo and J. Walrand. Fair end-to-end window-based congestion control. In *Proceedings of SPIE'98 International Symposium on Voice,Video and Data Communications*, 1998.

[27] C.J. Nuzman, I. Saniee, W. Sweldens, and A. Weiss. A compound model of tcp arrivals. In *Proceedings of ITC Seminar on IP Traffic Modeling*, Monterey, 2000.

[28] V. Paxson and S. Floyd. Wide-area traffic: The failure of poisson modeling. *IEEE/ACM Trans. Networking*, 3(3):226–255, 1995.

[29] J.W. Roberts and S. Oueslati-Boulahia. Quality of service by flow-aware networking. *Phil. Trans. R. Soc. London A*, 358:2197–2207, 2000.

[30] M. Vojnovic, J.-Y. Le Boudec, and C. Boutremans. Global fairness of additive-increase and multiplicative-decrease with heterogeneous round-trip times. In *Proceedings of IEEE INFOCOM'00*, 2000.