Bandwidth sharing and admission control for elastic traffic

L. Massoulié [a] and J.W. Roberts [b]

[a] *Microsoft Research, St. George House, 1 Guildhall St., Cambridge CB2 3NH, UK*
E-mail: lmassoul@microsoft.com
[b] *France Télécom – CNET, 38 rue du Général Leclerc, 92794 Issy-Moulineaux Cédex 9, France*
E-mail: james.roberts@francetelecom.fr

We consider the performance of a network like the Internet handling so-called elastic traffic where the rate of flows adjusts to fill available bandwidth. Realized throughput depends both on the way bandwidth is shared and on the random nature of traffic. We assume traffic consists of point to point transfers of individual documents of finite size arriving according to a Poisson process. Notable results are that weighted sharing has limited impact on perceived quality of service and that discrimination in favour of short documents leads to considerably better performance than fair sharing. In a linear network, max–min fairness is preferable to proportional fairness under random traffic while the converse is true under the assumption of a static configuration of persistent flows. Admission control is advocated as a necessary means to maintain goodput in case of traffic overload.

1. Introduction

Traffic in a multiservice network is essentially composed of individual transactions or flows which can be broadly categorized as "stream" or "elastic". Stream flows typically carry voice or video and are characterized by a variable data generation rate which must be more or less preserved as the flow passes through the network. Elastic flows, on the other hand, are established for the transfer of digital objects which can be transmitted at any rate up to the limit imposed by link and system capacity. The digital object in question might be a file, a Web page or a video clip transferred for local playback. We refer to such objects simply as documents.

The way bandwidth is shared is defined by the network service model. In this paper we do not discuss the detailed mechanisms used (such as the TCP protocol) and consider different sharing objectives without regard to practical realization. We do assume that the network service model recognizes individual flows, each being established for the transfer of a single document, and not aggregates of flows constituting all the traffic from one LAN to another, for example.

It is frequently assumed that elastic flows have more relaxed or lower quality of service requirements than stream flows. Indeed, their quality of service is rarely seen as a design consideration, the notion of fairness being used instead as the criterion

for judging performance. Fairness may be generalized to incorporate weights used to introduce deliberate bias, depending on different tariff options, for example. In this paper we argue that fairness should be of secondary concern and that the network should be designed rather to fulfil minimal quality of service requirements. These requirements concern realized throughput. They are different but not less important than those pertaining to real time stream flows.

For an elastic flow, quality of service is manifested essentially by the time it takes to complete the document transfer. This time depends both on the way bandwidth is shared and on the random fluctuations in the number of flows in progress as flows begin and end. For example, taking account of random traffic we show below that throughput can be improved by actively discriminating in favour of shorter transactions. Conversely, sharing in proportion to weights determined by tariff options provides quite uncertain quality of service differentiation.

To ensure quality of service in case of traffic overload, it appears necessary additionally to employ admission control, with flow blocking appearing as a more acceptable quality degradation than diminishing throughput. The admission control mechanism for elastic flows must be particularly lightweight in view of the large number of very short transactions to be controlled.

In the next section we discuss common bandwidth sharing objectives and their performance under the assumption that demand consists of a fixed configuration of point to point routes with persistent data sources. The notion of random traffic is introduced in section 3 where the performance of an isolated bottleneck link is analysed for a number of bandwidth sharing schemes assuming Poisson arrivals. In section 4, we consider the throughput of a network under the random traffic assumption for the sharing objectives introduced in section 2. The desirability and feasibility of admission control for elastic traffic are discussed in section 5.

2. Bandwidth sharing objectives

Performance of elastic flows depends on how link bandwidth is shared between them. In this section we adopt the usual assumption that the network is used by a fixed set of routes with traffic generated on each route by a persistent source, i.e., the source always has data to send at whatever rate is assigned by the network. Possible sharing goals and algorithms to achieve these goals are discussed by the authors in [11]. Here we recapitulate the principle possibilities.

2.1. Network model

Consider a network as a set of links $\mathcal{L}$, where each link $l \in \mathcal{L}$ has a capacity $C_l > 0$. A number of flows compete for access to these links, each flow being associated with a route consisting of a subset of $\mathcal{L}$. In this section we focus on a short time scale where the set of flows is fixed. We seek to allocate link bandwidth to the

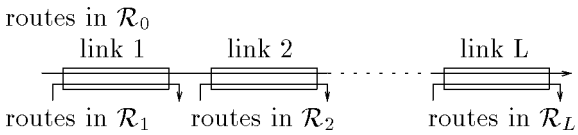

Figure 1. The linear network.

set of flows to meet some sharing objective. Let λ_r denote the allocation of route r. Feasible bandwidth allocations must satisfy the capacity constraints:

$$\sum_{r \ni l} \lambda_r \leqslant C_l, \quad l \in \mathcal{L}. \tag{1}$$

We assume here that flows are perfectly fluid and ignore the problems of granularity due to packet size.

To illustrate possible allocation strategies we use the simple linear network depicted in figure 1. The network consists of L unit capacity links ($C_l = 1$) with x_0 long routes which cross every link, and x_l routes which use link l alone, for $1 \leqslant l \leqslant L$. Denote by $\mathcal{R}_0$ the set of long routes and by $\mathcal{R}_l$ the set of routes using only link l. The aim is to determine the set of allocations $\{\lambda_r\}$, satisfying the capacity constraints $\sum_{r \in l} \lambda_r \leqslant 1$ for each link l, which optimize some criterion.

2.2. *Max throughput*

A natural objective might be to choose the λ_r so as to maximize the global network throughput, that is to say, to maximize $\sum \lambda_r$. However, a significant drawback with this sharing objective is that it often leads to allocations where λ_r must be zero for some flows. For example, consider the linear network of figure 1 with one route on each link and one route end to end. For a given allocation λ_0, in order to maximize the overall throughput within the capacity constraints we should allocate $\lambda_r = 1 - \lambda_0$ to all the other routes giving a total throughput of $L - (L-1)\lambda_0$. This is maximal for $\lambda_0 = 0$ and is then equal to L.

2.3. *Max–min fairness*

Max–min sharing is the classical sharing principle in the domain of data networks as discussed, for instance, by Bertsekas and Gallager [3]. The objective stated simply is indeed to maximize $\min_{\mathcal{R}}\{\lambda_r\}$ subject to the capacity constraints. More formally, the allocations λ_r must be such that an increase of any λ_r within the domain of feasible allocations must be at the cost of a decrease of some $\lambda_{r'}$ such that $\lambda_{r'} \leqslant \lambda_r$. The max–min allocation is unique and is characterized by the condition:

- for every route r, there is at least one link $l \in r$ such that

$$\sum_{r' \ni l} \lambda_{r'} = C_l \quad \text{and} \quad \lambda_r = \max\bigl\{\lambda_{r'},\ r' \ni l\bigr\}. \tag{2}$$

The max–min allocation for the network of figure 1 is as follows:

$$\lambda_r = \begin{cases} \dfrac{1}{x_0 + \max_{l \geqslant 1} x_l} & \text{for } r \in \mathcal{R}_0, \\ \dfrac{1}{x_l}\left(1 - \dfrac{x_0}{x_0 + \max_{l \geqslant 1} x_l}\right) & \text{for } r \in \mathcal{R}_l,\ l \geqslant 1. \end{cases}$$

In the particular case where $x_i = 1$ for $i \geqslant 0$, the allocation to all routes is $1/2$ and the total throughput is $(L+1)/2$, considerably less than the maximum L.

2.4. Proportional fairness

The appropriateness of max–min fairness as a bandwidth sharing objective has recently been questioned by Kelly [7] who has introduced the alternative notion of proportional fairness. Rate allocations λ_r are proportionally fair if they maximize $\sum_{\mathcal{R}} \log \lambda_r$ under the capacity constraints. This objective may be interpreted as being to maximize the overall utility of rate allocations assuming each route has a logarithmic utility function (the law of diminishing returns).

Again, in the case of finitely many links and routes, the vector of proportionally fair rate shares λ_r is unique. It may be characterized as follows. The aggregate of proportional rate changes with respect to the optimum of any other feasible allocation λ'_r is negative, i.e.,

$$\sum_{\mathcal{R}} \frac{\lambda'_r - \lambda_r}{\lambda_r} \leqslant 0.$$

Consider how this rate allocation works in the case of the linear network of figure 1. First it is clear that all routes in the same set $\mathcal{R}_i$ must have the same allocation. Let γ_i be the allocation to routes in set $\mathcal{R}_i$ for $0 \leqslant i \leqslant L$. We necessarily have $x_0\gamma_0 + x_i\gamma_i = 1$ for $1 \leqslant i \leqslant L$: this sum is the capacity used at link i and must, therefore, be less than or equal to one; however, for any rate allocation such that this sum is less than one, γ_i can be increased without violating the capacity constraints and this results in an increase in the objective function to be maximized. It follows that to determine the optimal rate allocation we must find the value γ_0 which maximizes

$$x_0 \log(\gamma_0) + \sum_{i=1}^{L} x_i \log\left(\frac{1 - x_0\gamma_0}{x_i}\right).$$

Differentiating, we have that at the optimum

$$\frac{x_0}{\gamma_0} = \sum_{i=1}^{L} \frac{x_i x_0}{1 - x_0\gamma_0},$$

giving

$$\gamma_0 = \frac{1}{x_0 + \sum_{i=1}^{L} x_i}.$$

In the particular case where $x_i = 1$ for $0 \leqslant i \leqslant L$, we deduce the allocation $\lambda_0 = 1/(L+1)$ and $\lambda_r = L/(L+1)$ for $r \neq 0$. This corresponds to an overall throughput of $L - (L-1)/(L+1)$. It is clear from this example that proportional fairness penalizes long routes more severely than max–min fairness in the interest of greater overall throughput.

2.5. *Weighted shares*

Both max–min and proportional fairness criteria can be generalized on introducing weighting factors ϕ_r associated with each route r such that an increase in this weight leads to an increase in the received share λ_r. The general definition of max–min fairness is then:

For all r, there is at least one link $l \in r$ such that

$$\sum_{r' \ni l} \lambda_{r'} = C_l \quad \text{and} \quad \frac{\lambda_r}{\phi_r} = \max\left\{ \frac{\lambda_{r'}}{\phi_{r'}} \colon\ r' \ni l \right\}. \tag{3}$$

In the case of a single bottleneck link, the allocation to each route is in proportion to its weight, i.e., we have $\lambda_r/\phi_r =$ constant.

A weighted version of the proportional fairness criterion is described in [7]. The rates λ_r are then chosen so as to maximize $\sum_{\mathcal{R}} \phi_r \log \lambda_r$. Again, in the case of a single link, the weighted proportionally fair allocations are such that $\lambda_r/\phi_r =$ constant.

The use of weights has been advocated as a means for users to express the relative value of their traffic with the assumption that they pay more for a higher value of ϕ_r. Note, however, that the variation of the optimal allocation λ_r with ϕ_r is not straightforward: the increase in λ_r is approximately proportional to ϕ_r only when the number of routes sharing a link is large and the individual allocations are small.

Weighted proportional fair sharing appears in [8] as a means to achieve an allocation with optimal utility when users dynamically express the utility they attach to an allocation through the value they attribute to the weight parameters.

3. Flow throughput in random traffic

Fairness of bandwidth sharing is an issue which is relevant mainly at the small time scale during which the number of flows in contention remains fixed. In practice, this number is a random process, varying as flows begin and end, and the throughput achieved by a given flow depends as much on this process as on the bandwidth sharing algorithm employed. User perceived quality of service may be measured by the response time of a given document transfer or, equivalently, by the realized throughput equal to the document size divided by the response time. The fact that this throughput

was attributed "fairly" is largely irrelevant and, moreover, totally unverifiable by the user.

3.1. Traffic model

We assume that traffic to be handled by the network bandwidth sharing protocol appears as a succession of requests for the immediate transfer of a certain document. The arrival process of requests for document transfer on a given network route is assumed to be Poisson. This process results naturally when a large population of users emits requests independently, each at a relatively low intensity.

The size of digital documents is highly variable. Observations on Web traffic indeed reveal that the tail of the document size distribution behaves like that of a Pareto distribution [1,4]:

$$\Pr\{\text{size} > x\} \sim \left(\frac{k}{x}\right)^{\alpha},$$

where the exponent α satisfies $1 < \alpha \leqslant 2$ ($\alpha \leqslant 1$ leads to a distribution with infinite mean). In numerical evaluations below we have assumed a Pareto distribution with parameter values $\alpha = 1.4$ and $k = 1$ Kbyte.

In this section we consider the performance of a single bottleneck link under the above traffic model. The link has capacity c and is offered Poisson traffic of intensity λ with a mean document size of $1/\mu$. We denote link utilization by ρ, i.e., $\rho = \lambda/\mu c$.

3.2. A processor sharing model

We assume for simplicity that traffic is perfectly fluid and that, when the number of flows in progress at time t, $X(t)$, changes to n, the flow control protocol instantaneously adjusts the service rate of each flow to c/n. These assumptions define the classical processor sharing queue for which a number of interesting performance results are well known [9].

If $\rho < 1$, the stationary distribution of $X(t)$ is geometric:

$$\Pr\{X(t) = n\} = \rho^n(1-\rho),$$

and the expected response time $R(p)$ for a document of size p is

$$R(p) = \frac{p}{c(1-\rho)}.$$

The above results are insensitive to the document size distribution. If admission control is used to limit the number of flows in progress to $n_{\max}$ say, the distribution of $X(t)$ is given by truncating and renormalizing the above geometric distribution. In particular, for any load ρ, the probability a new request is blocked is given by

$$B = \Pr\{X(t) = n_{\max}\} = \frac{\rho^{n_{\max}}(1-\rho)}{1-\rho^{n_{\max}+1}}. \tag{4}$$

Further simple refinements to the above model are a common limit on the maximum rate of any flow (due to access line capacity, for example) and a state dependent arrival rate. The case where the arrival rate results from a finite number of sources is considered by Heyman et al. as a model of a link shared using the congestion avoidance algorithm of TCP [6]. The insensitivity property, together with much of the model's tractability, is lost if flows are not homogeneous, having different minimum or maximum rates, for instance.

3.3. Unequal shares

A particularly interesting case of heterogeneity arises when the flows do not share bandwidth equally but in proportion to a weight attributed to their pre-assigned service class. Let flows of class i be assigned a weight ϕ_i such that when the number $X^j(t)$ of flows of class j is n_j, for $j = 1, \ldots, m$, the service rate of each class i flow is: $\phi_i / \sum n_j \phi_j$. With no further restrictions on service rates, these assumptions define the discriminatory processor sharing model considered by Fayolle et al. [5]. They derive expressions for the expected response time when document size distributions of the different service classes have a rational Laplace transform. Figure 2 shows the normalized response time $R(p)/p$ as a function of p for a unit capacity ($c = 1$), two-class system with $\phi_1 = 1$ and $\phi_2 = 2$. The same traffic is generated by each class, server load is $2/3$. The mean document size is 1 and we present results for two size distributions: exponential (full lines) and hyperexponential (dashed lines). We

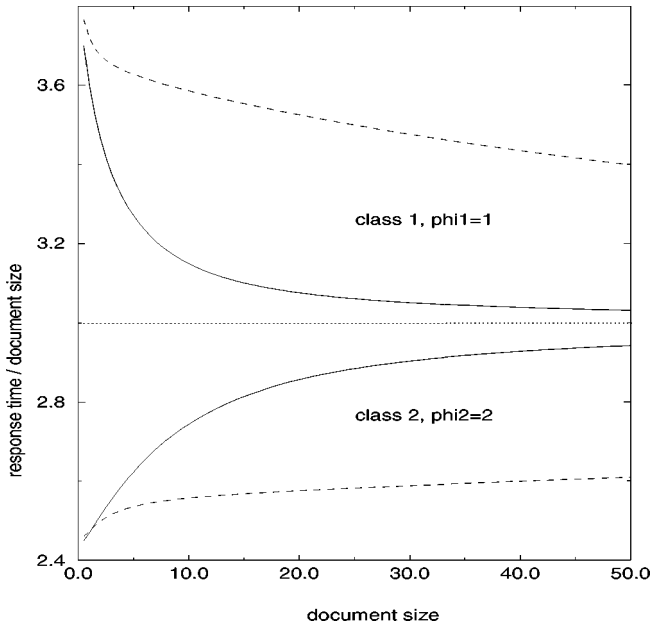

Figure 2. Normalized response time $R(p)/p$ for discriminatory processor sharing.

have chosen a large variance of 250 for the hyperexponential distribution to give an indication of the impact of the heavy tailed distribution observed in practice.

From these and other numerical evaluations we derive the following observations:

- throughput depends on the document size and the document size distribution;
- the weights ensure effective service discrimination for short documents but, for all classes, expected throughput $p/R(p)$ tends to $c(1-\rho)$ as document size increases;
- the distribution of the number of flows in progress from each class is roughly insensitive with respect to the document size distribution.

The second observation reflects the fact that an exceptionally long document utilizes all the capacity left available by the transfer of shorter documents starting and ending within the transfer time of the former. With the Poisson arrivals assumption, this remaining capacity is indeed, on average, equal to $c(1-\rho)$. A possible motivation for attributing different weights to flows sharing a network may be pricing: users pay more for a greater share of bandwidth (e.g., [7]). This simple discriminatory processor sharing model suggests that the gain in realized throughput may be very slight in relation to the price paid, particularly for large documents for which response time is a particularly significant measure of performance.

3.4. Priority to short documents

Discrimination between flows on a class basis may be more effective from a performance point of view if the class distinguishes document size. Indeed, it is known that the throughput performance of a single server is optimized on employing the "shortest remaining processing time" preemptive resume scheduling algorithm (SRPT):

- the server is assumed to know the remaining volume of data of all documents to be transferred and devotes its capacity exclusively to the smallest;
- if a new arrival concerns a document whose size is less than that of the document in service, the latter is preempted;
- any preempted transfer resumes service where it left off as soon as its remaining volume is again smaller than that of any other pending request.

The throughput performance of SRPT was studied by Schrage and Miller [15]. They notably derive expressions for the response time $R(p)$ of a document of size p under an assumption of Poisson arrivals and general service time distribution. Figure 3 shows a numerical evaluation of their formulas for exponential and Pareto distributed document sizes, respectively. Link load is 0.66, as in the example considered in figure 2. The p axis, in units of the mean document size, is on a log scale to capture the heavy tail particularity of the Pareto distribution.

The results clearly illustrate that SRPT considerably improves the response time of short documents. In the case of exponential document sizes, the response time of longer documents (size greater than 5 times the mean) increases marginally with

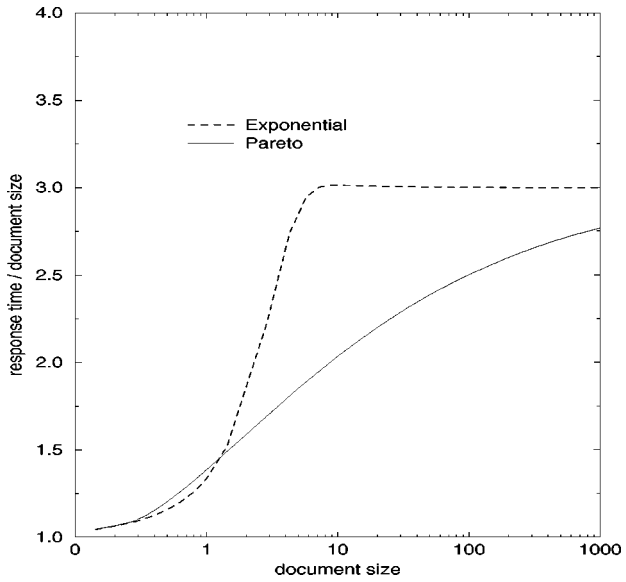

Figure 3. Normalized response time $R(p)/p$ for SRPT scheduling.

respect to that achieved with fair sharing, $p/(1-\rho)$. All response times are significantly reduced, however, in the practically significant case of the Pareto distribution.

Implementation of SRPT in the case of a single link would, of course, be very complex and the appropriate extension of this principle to a network remains unclear. However, it does provide a clear illustration that fairness, or weighted fairness, is not necessarily a useful objective in bandwidth sharing. In particular, both users and network provider stand to gain by employing a flow control protocol which discriminates in favour of short documents.

4. Flow throughput in the network

The throughput of flows in a network depends in a complicated way on the traffic on all links. To investigate the impact on performance of the different bandwidth sharing objectives discussed in section 2, we consider the simple linear network of figure 1.

4.1. Max–min sharing

It proves very difficult to extend the simple processor sharing model described above to the case of a network realizing max–min fair sharing. Consider the simple linear model of figure 1 with the following traffic assumptions for each route i:

- requests for transfers arrive according to a Poisson process of rate λ_i;
- the document size distribution is exponential with mean μ_i^{-1}.

The vector $X(t)$ giving the number of flows on each route at time t is then a Markov process with transition rates:

$$\begin{aligned} q(x, x + e_i) &= \lambda_i, \\ q(x, x - e_0) &= \mu_0 \frac{x_0}{x_0 + \max_{1 \leqslant i \leqslant L} x_i}, \\ q(x, x - e_i) &= \mu_i \left[1 - \frac{x_0}{x_0 + \max_{1 \leqslant i \leqslant L} x_i} \right], \end{aligned}$$

where e_i denotes the $(i+1)$th unit vector in $\mathbb{R}^{L+1}$.

Solution for the stationary distribution of $X(t)$ proves intractable in general. Some insight into the throughput on long routes may be deduced from the limit case where $\mu_0 \to \infty$. This assumption renders the individual links virtually independent and the probability of having more than one flow on the long route (i.e., $X_0 > 1$) negligible. Let $\rho_i := \lambda_i / \mu_i$. We deduce, as in [13],

$$E[X_0] = \rho_0 \left[1 + \sum_{k=1}^{L} (-1)^{k+1} \sum_{|\Gamma|=k} \frac{\prod_\Gamma \rho_i}{1 - \prod_\Gamma \rho_i} \right] + o(\rho_0),$$

where Γ denotes a subset of links. When $\rho_i = \rho$ for $1 \leqslant i \leqslant L$, we derive the asymptotic estimate as $L \to \infty$:

$$E[X_0] \sim \rho_0 \left(-\frac{1}{\log \rho} \right) \log L.$$

The assumption of independent geometric distributions for the X_j is pessimistic for the throughput of the end to end flow. We deduce, therefore, that the expected throughput of route 0 transfers, with the initial traffic assumptions, decreases more slowly than $1/\log L$ as the number of hops L increases, in the case of max–min sharing.

4.2. *Proportionally fair sharing*

Proportionally fair sharing is (surprisingly) amenable to analysis in the particular case of the linear network with the same traffic assumptions as introduced above. In section 2, we derived the proportionally fair rate allocations corresponding to a given state $\{x_0, x_1, \ldots, x_L\}$. We deduce the following transition rates for the Markov process $X(t)$:

$$\begin{aligned} q(x, x + e_i) &= \lambda_i, \\ q(x, x - e_0) &= \mu_0 \frac{x_0}{x_0 + \sum_{i=1}^{L} x_i}, \\ q(x, x - e_i) &= \mu_i \frac{\sum_{i=1}^{L} x_i}{x_0 + \sum_{i=1}^{L} x_i}. \end{aligned} \tag{5}$$

The following theorem provides the explicit stationary behaviour of the process $X(t)$.

Theorem 1. Under the stability condition $\sup_{1\leqslant i\leqslant L} \rho_0 + \rho_i < 1$, the process $X(t)$ is reversible, with equilibrium distribution given by

$$\pi(x_0, \ldots, x_L) = C^{-1} \binom{\sum_{i=0}^{L} x_i}{x_0} \prod_{i=0}^{L} \rho_i^{x_i}, \tag{6}$$

where the normalization constant C equals

$$C = \frac{(1-\rho_0)^{L-1}}{\prod_{i=1}^{L}(1-\rho_0-\rho_i)}. \tag{7}$$

The corresponding generating function ψ defined by

$$\psi(z) = \sum_{x_0,\ldots,x_L \geqslant 0} \pi(x_0, \ldots, x_L) \prod_{i=0}^{L} z_i^{x_i}$$

may be written as

$$\psi(z) = \left(\frac{1-\rho_0 z_0}{1-\rho_0}\right)^{L-1} \prod_{i=1}^{L} \left(\frac{1-\rho_0-\rho_i}{1-\rho_0 z_0 - \rho_i z_i}\right). \tag{8}$$

The mean number of calls in progress along route i in the stationary regime is given by

$$E[X_0] = \frac{\rho_0}{1-\rho_0}\left(1 + \sum_{i=1}^{L} \frac{\rho_0}{1-\rho_0-\rho_i}\right) \tag{9}$$

and for $i \geqslant 1$,

$$E[X_i] = \frac{\rho_i}{1-\rho_0-\rho_i}. \tag{10}$$

Proof. First check that π as defined in (6) is indeed stationary and reversible for the transition rates (5). This is true if, for all $x \in \mathbb{N}^{L+1}$ and all $j \in \{0, \ldots, L\}$,

$$\pi(x) q(x, x+e_j) = \pi(x+e_j) q(x+e_j, x)$$

which may be readily verified.

The normalization constant C is

$$C = \sum_{x_0,\ldots,x_L \geqslant 0} \binom{\sum_{i=0}^{L} x_i}{x_0} \prod_{i=0}^{L} \rho_i^{x_i}.$$

Applying the negative binomial formula

$$(1-z)^{-d} = \sum_{n \geqslant 0} \binom{d-1+n}{n} z^n,$$

to the summation over x_0 gives

$$C = \sum_{x_1,\ldots,x_L \geqslant 0} (1-\rho_0)^{-(1+\sum_{i=1}^n x_i)} \prod_{i=1}^{L} \rho_i^{x_i}$$

and expression (7) easily follows.

Expression (8) is obtained similarly since the same sum has to be computed with $\rho_i z_i$ instead of ρ_i. The expressions for $E[X_i]$ are then derived using

$$E[X_i] = \frac{\partial}{\partial z_i} \log \psi(1,\ldots,1). \qquad \square$$

Note that, due to the reversibility of process $X(t)$, the above results are insensitive to the document size distribution and do not rely on the exponential assumption. We have unfortunately not been able to make any nontrivial generalizations of this theorem to other network configurations. The results do nevertheless illustrate some interesting aspects of the qualitative behaviour of proportional fairness.

Expression (10) illustrates that the effect of long path traffic on the throughput of single hop flows is, on average, simply to reduce the available bandwidth. Throughput on the long path, on the other hand, is quite severely restricted by the cross traffic: expected throughput ($= \rho_0/E[X_0]$) decreases like $1/L$. This contrasts with the corresponding result for max–min fairness (derived in the limit $\mu_0 \to \infty$) where throughput decreases like $1/\log L$.

4.3. *Overall throughput*

In section 2 we showed that, in the static regime, overall throughput of the linear network is better with proportional fair sharing than with max–min fair sharing. This observation does not necessarily carry over to the present dynamic traffic regime where the number of flows in progress is random and depends on the rate allocations.

Consider the linear network in the simplest case where all traffic loads ρ_i are equal to ρ. As a measure of overall throughput we use the expected number of flows in progress $\overline{X} = \sum E[X_r]$ which, by Little's law, is proportional to the response time of an arbitrary flow. The analytic results of theorem 1 allow us to calculate $\overline{X}$ in the case of proportional fair sharing:

$$\overline{X}_{\text{propfair}} = \frac{\rho}{1-\rho} + \frac{\rho}{1-2\rho}\,\frac{1}{1-\rho} L. \tag{11}$$

For max–min fair shares we have used simulation. The results are presented in figure 4.

The figure reveals that proportional fair sharing is in fact *less* effective than max–min sharing. Reducing the rate attributed to the end to end flows liberates more capacity for short path flows in a static regime but, with the considered random traffic process, the reduction tends to increase the number of flows in progress leading to lower overall throughput.

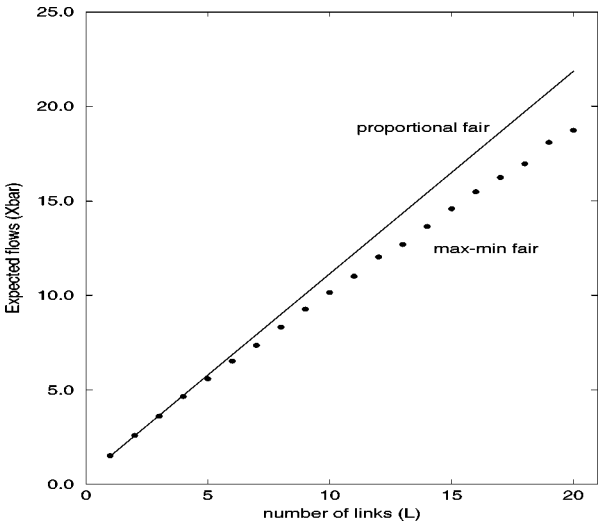

Figure 4. Overall expected number of flows $\overline{X}$.

It is tempting to conclude that bandwidth sharing would be more efficient on attributing even more capacity to the end to end flows. That this is not true can be seen on considering the following sharing scheme: in the network of figure 1, attribute all capacity to the end to end flows as long as $X_0 > 0$; share equally between flows on the same path. In this scheme, each link behaves like a preemptive resume $\mathrm{M}/\mathrm{M}/1$ queue and it is possible to derive the value of $\overline{X}$ from known formulas for this system (see [2, p. 192]). It turns out that the expression for $\overline{X}$ is precisely the same as that for proportional fairness (11), although the individual means $E[X_i]$ are clearly different.

These examples illustrate the difficulty in choosing a network sharing scheme in the case of random traffic. Throughput in the static regime is not a useful measure of performance when the dynamic nature of flow composition is taken into account. Max–min sharing appears as the preferred choice for the simple linear network but we have no evidence to suggest this is the case in general.

4.4. Alternative sharing policies

Bandwidth sharing schemes derived from the static models of section 2 give equal weight to flows of arbitrary size. For a single link we saw in section 3 that SRPT service discipline gave considerably better performance than fair sharing. While this result cannot be immediately generalized to the case of a network, it is certainly true to say that max–min or proportionally fair bandwidth sharing is not optimal from the performance point of view. It remains preferable to give priority to short documents since their response time can be improved without any detrimental effect on that of long documents.

While one aim of bandwidth sharing is to provide the best possible quality of service to users, the absolute service quality level depends more on the relation between

available capacity and demand. We argue below that the network bandwidth sharing policy should incorporate the means to limit the numbers of flows in progress in order that each is guaranteed an acceptable minimum throughput.

5. Admission control to limit the number of flows in progress

Admission control is generally accepted as necessary for flows requiring hard performance guarantees with respect to network delays. For elastic traffic, however, it is more commonly assumed that when a new demand appears it is better to reduce the throughput of ongoing flows than to reject the new flow: the utility of a flow as a function of its throughput is assumed positive and strictly concave everywhere so that overall utility increases as more flows are admitted [16]. We contest this assumption and argue that admission control is also extremely desirable for elastic flows.

5.1. Overload control

The simple performance models discussed above rely on the assumption that offered load ρ is less than one. In this case, in the processor sharing model, the number of documents in transit is usually very small and expected throughput $c(1-\rho)$ is high. Throughput performance would be approximately the same if the number of admitted flows were limited (to 50, say, for a load $\rho = 0.8$) and the probability of rejection, given by (4), would be very small (less than 10^{-5} in the above example).

Admission control is useful mainly when the offered load is greater than 1. In this case, the processor sharing models are unstable, the number of documents in transit tending to infinity as their allocated rate gets smaller and smaller. In practice, as the rate they receive becomes very low, users begin to abandon their transactions through impatience (or higher layer protocols, interpreting excessive acknowledgement delays as a sign of link failure, interrupt the connection). This is in contradiction with the assumption that the utility function is strictly concave: there *is* a minimum throughput below which utility is zero (or negative).

Goodput (corresponding to completed document transfers) may be very low even though the link is observed to be fully utilized. By limiting the number of flows such that each has an acceptable throughput (20 Kbit/s, say), an admitted flow is almost always completed and link capacity continues to be used efficiently. Goodput is maximized for a particular threshold whose value depends on user behaviour as well as link capacity and traffic intensity. The above arguments are further developed by the authors in [12].

5.2. Flow routing

The network functions necessary for admission control are also necessary for intelligent traffic routing. In the present Internet, packets are forwarded (virtually) independently of the capacity and congestion status of links. It is possible that a

congested link is repeatedly used for new flows even though an alternative path could offer much greater throughput. It is likely at times that offered traffic on some link included in the forwarding paths determined by the Internet routing protocols is indeed greater than capacity leading to saturation and loss of goodput as outlined above.

Observation of link load alone does not reveal the extent of the congestion since the mechanisms of TCP maintain this less than one. The more relevant measure of traffic is the number of flows currently in progress. This is precisely the information necessary for admission control. Indeed, admission control appears as a particular form of intelligent routing: forward the flow to an alternative route of zero capacity rather than further reduce throughput on a saturated link.

5.3. Pricing

In [14] we argued the case for "transaction pricing" where users pay in relation to the volume of bytes transmitted in any flow (stream or elastic). In this context admission control appears as an essential network attribute in order to ensure that users receive good quality of service for the fee they pay.

On the other hand, as flows rejected by admission control do not produce revenue, the network has the necessary incentive to provide sufficient capacity to limit blocking to an acceptably low level.

5.4. Implementation

Admission control does not necessarily imply a complex flow set up stage with explicit signalling exchanges between user and network nodes. This would be quite unacceptable for most elastic flows which are of very short duration. We envisage a network rather similar to the present Internet where users simply send their data as and when they wish. However, nodes implementing admission control would keep a record of the identities of existing flows currently traversing each link in order to be able to recognize the arrival of a packet from a new flow. Such a packet would be accepted and its identifier added to the list of active flows if the number of flows currently in progress were less than a threshold, and would otherwise be rejected. A flow would be erased from the table of existing flows if it sent no packets during a certain time out interval.

Although many additional practical considerations would need to be addressed, such a control procedure does seem largely feasible technically given recent developments in router technology [10].

6. Conclusions

In this paper we have considered the performance of a network handling elastic traffic: documents are transferred at a rate determined by available bandwidth. We have argued that the design of schemes for sharing bandwidth between elastic flows

should take more account of absolute user perceived performance than the relative notion of fairness.

Bandwidth sharing is usually considered for a static configuration of flows. In this case it is possible to distinguish different sharing schemes by the overall utility of the resulting bandwidth allocations. We have shown on a simple linear network how proportional fairness outperforms max–min fairness in this sense by allocating less bandwidth to routes using a large number of links.

In practice, traffic in a network is not static and realized throughput depends more on the randomly changing number of flows in progress than on the way bandwidth is shared between the set of flows present at any given time. We have illustrated the impact of random traffic in the simplest case of a single bottleneck link. Notable results are that the attribution of sharing weights (corresponding to different tariff options, for example) has fairly unpredictable consequences on realized throughput, the response time of very long documents, in particular, being largely independent of the attributed weight. Performance of bandwidth sharing can be improved by actively discriminating in favour of the transfer of shorter documents by implementing a scheduling policy like SRPT, for example.

Throughput performance in a network under random traffic proves very difficult to evaluate. However, a comparative study in the case of a simple linear network allows us to conclude that the relative performance of max–min and proportional fair bandwidth sharing schemes is inversed with respect to that obtained in the static configuration. While it makes sense in a static regime to reduce the rate of flows using a large number of links (as in a proportional fair allocation), in random traffic this reduction leads to increased response times and thus an increased expected number of such flows.

In the random traffic environment there may be situations where demand (arrival rate of new flows $\times$ their mean size in bytes) exceeds link capacity. Such overload provokes congestion leading to a reduction in goodput (the combined rate of successfully completed document transfers) as flows are prematurely interrupted due to user impatience or the actions of higher layer protocols. We have suggested that this type of overload should be avoided by means of admission control: no link should admit more than the number of flows compatible with a minimum acceptable throughput. Recent developments in router technology suggest that admission control is feasible even accounting for the particular nature of elastic traffic characterized by large numbers of very small document transfers. The same mechanisms used for admission control would also be necessary to perform intelligent flow routing allowing saturated links to be avoided when alternative paths are available.

References

[1] M.F. Arlitt and C. Williamson, Web server workload characterization: The search for invariants, in: *Proc. of ACM Sigmetrics '96* (1996).

[2] F. Baccelli and P. Brémaud, *Elements of Queueing Theory*, Applications of Mathematics, Vol. 26 (Springer, New York, 1994).
[3] D. Bertsekas and R. Gallager, *Data Networks* (Prentice-Hall, Englewood Cliffs, NJ, 1987).
[4] M. Crovella and A. Bestavros, Self-similarity in World Wide Web traffic: Evidence and possible causes, in: *Proc. of ACM Sigmetrics '96* (1996).
[5] G. Fayolle, I. Mitrani and R. Iasnogorodski, Sharing a processor among many jobs, Journal of the ACM 27(3) (1980) 519–532.
[6] D. Heyman, T. Lakshman and A. Neidhart, A new method for analysing feedback-based protocols with application to engineering Web traffic over the Internet, in: *Proc. of ACM Sigmetrics '97* (1997).
[7] F. Kelly, Charging and rate control for elastic traffic, European Transactions on Telecomunications 8 (1997) 33–37.
[8] F. 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).
[9] L. Kleinrock, *Queueing Systems*, Vol. 2 (Wiley, New York, 1975).
[10] V.P. Kumar, T.V. Lakshman and D. Stiliadis, Beyond best effort: Router architectures for differentiated services of tomorrow's Internet, IEEE Communications Magazine 36(5) (1998).
[11] L. Massoulié and J. Roberts, Bandwidth sharing: Objectives and algorithms, in: *Proc. of IEEE Infocom '99* (1999).
[12] L. Massoulié and J. Roberts, Arguments in favour of admission control for TCP flows, in: *Proc. of ITC 16*, eds. P. Key and D. Smith, *Teletraffic in a Competitive World* (North-Holland, Amsterdam, 1999).
[13] E. Oubagha, L. Massoulié and A. Simonian, Delay analysis of a credit-based control for ABR transfer, in: *IEEE ATM '97 Workshop* (1997).
[14] J. Roberts, Quality of service guarantees and charging in multiservices networks, IEICE Transactions on Communications, Special issue on ATM Traffic Control and Performance Evaluation E81-B(5) (1998).
[15] L. Schrage and L. Miller, The M/G/1 queue with the shortest remaining processing time first discipline, Operations Research (1966) 670–684.
[16] S. Shenker, Fundamental design issues for the future Internet, IEEE Journal on Selected Areas in Communications 13(7) (1995).