

A survey on statistical bandwidth sharing

J. W. Roberts

*France Telecom R&D, 38 rue du Général Leclerc, 92794 Issy-Moulineaux Cedex,
France*

Abstract

The paper provides a survey of recent results on the performance of a network handling elastic data traffic under the assumption that flows are generated as a random process. We notably highlight the insensitivity results allowing a relatively simple expression of performance when bandwidth sharing realizes so-called “balanced fairness”. In normal load conditions flow throughput mainly depends on the users’ access line rate. There is little scope for optimizing performance, by means of size-dependent scheduling on network links, for instance. In overload, performance deteriorates rapidly. Pro-active admission control is then an arguably necessary traffic control to preserve performance. This paper is written as a tribute to the author’s friend Olga Casals.

Key words: Internet traffic theory, elastic traffic, processor sharing, balanced fairness, admission control.

1 Introduction

Data traffic is inherently elastic in that the rate of any transfer can be modulated depending on current demand, generally without detriment to user perceived performance. In the Internet, the congestion control algorithms implemented in TCP aim to fully exploit available capacity while adjusting the sending rate of competing transfers to realize a certain sharing of network bandwidth. Until recently the study of bandwidth sharing has generally been performed under the assumption that transfers last indefinitely. In fact, data connections are established at arbitrary instants and cease when the transfer of some finite size document is complete. This paper presents a survey of results on the performance of bandwidth sharing in this stochastic context.

The objective of this survey is to convey the most significant results on statistical bandwidth sharing and their significance for network design. Even though statistical bandwidth sharing is a relatively young field, it is rapidly gaining

popularity and there is already an abundant literature. The survey may not therefore be completely exhaustive and some of the author's interpretations are not universally accepted. The paper should be interpreted as an introduction to an important aspect of Internet traffic theory.

Statistical bandwidth sharing concerns traffic entities at a higher level than the IP datagram. Specifically we consider how link capacity is shared between concurrent *flows*. For present purposes a flow corresponds to the transfer of some digital document, like a Web page or an audio track, for instance. Such transfers are not usually generated in isolation but form part of a *session*.

While there may be some ambiguity in defining flows and sessions in practice, the following is a reasonable model of traffic in a backbone network. In the busy period, sessions occur as a stationary Poisson process. This results naturally from an assumption that individual sessions are independently generated by a large population of users and has been confirmed by measurements [26]. Each session is a finite succession of flows and inter-flow intervals that we call *think times*. Flow sizes and think time durations have distributions that reflect the underlying applications (mail, Web, P2P, ...). The number of flows in a session is also variable from session to session. Finally, successive flow durations and think times may be correlated. We call this model of backbone traffic the Poisson session model.

In the access network, a more appropriate traffic model takes account of the fixed number of traffic sources. Each user is assumed to generate an alternating sequence of flows and think times. The distributions of flow size and think time duration are again general and depend on the applications. It may be appropriate to identify user sessions delimited by longer think times or other artefacts introducing correlation.

The generality of the flow level processes described above can be taken into account simply in certain bandwidth sharing models. This is the beauty of so-called *balanced fair* sharing that provides performance results that are insensitive to detailed traffic characteristics. In evaluating statistical bandwidth sharing it is important to understand the necessary and sufficient conditions for insensitivity. The performance of alternative bandwidth sharing objectives can often be better understood in the light of the powerful insensitivity results.

A significant observation is that performance at normal traffic loads is generally very good and depends mainly on the average demand equal to the product of flow arrival rate and mean flow size. The balanced fair model allows an appraisal of what the "normal load" should be. In overload on the other hand, when demand exceeds capacity, performance is generally very poor and difficult to predict. It is mainly in such conditions that traffic control mechanisms like per-flow admission control can preserve network efficiency.

The next section introduces the insensitive model of bandwidth sharing on an isolated bottleneck. The impact of unfair sharing, deliberate or not, and the way one can evaluate the sharing realized in practice by TCP are considered in Section 3. Extensions to sharing bandwidth in a network are discussed in Section 4 where the notion of balanced fairness is defined. Section 5 considers how the performance of bandwidth sharing behaves under overload.

2 A fairly shared isolated bottleneck

In this section we consider the sharing of an isolated bottleneck assuming an ideal fluid model where link bandwidth is shared perfectly fairly with instantaneous adjustment as the number of active flows changes.

2.1 Ideal fair sharing

The assumption of perfect fair sharing allows the use of powerful results for processor sharing (PS) queues. This approach was proposed independently by Massoulié and Roberts [40], Nabe *et al.* [43] and Heyman *et al.* [30]. The generalization to the Poisson session traffic model was explained by Bonald and co-authors in two papers [6], [15]. Bonald and Proutière have since developed a unified approach based on so-called Whittle networks ([45], [13], [12]).

Whittle networks Whittle networks have the classical Jackson network form of a set \mathcal{N} of exponential servers visited by customers following a Bernoulli routing scheme. They are distinguished from Jackson networks by the fact that the service rate, $\psi_i(y)$, at node i can depend on the numbers of customers $\{y_1, y_2, \dots, y_N\}$ at all nodes of the network. The rates $\psi_i(y)$ are not arbitrary but satisfy:

$$\psi_i(y) = \frac{\Psi(y - e_i)}{\Psi(y)} \quad (1)$$

for a given positive function $\Psi(y)$ where e_i represents the vector with 1 in position i and 0 elsewhere. The rates are said to be “balanced” by $\Psi(y)$. Given condition (1), the network state probabilities $\chi(y)$ have the product form:

$$\chi(y) = \chi(0) \Psi(y) \prod_{i=1}^N a_i^{y_i}. \quad (2)$$

where demand a_i is the product of the flow arrival rate and the mean customer service requirement and $\chi(0)$ is determined by the normalization condition. The flow arrival rate at each node is derived from the classical routing equations of Jackson networks.

Processor sharing networks We consider a special case of Whittle networks where all nodes are processor sharing servers: all customers at node i are simultaneously served at rate $\psi_i(y)/y_i$. This definition includes the classical processor sharing server of rate C_i ($\psi_i(y) = C_i$), the processor sharing server with a maximum rate per customer c ($\psi_i(y) = \min\{y_i c, C_i\}$), and the infinite server ($\psi_i(y) = y_i$).

We define PS *equivalence sets* of nodes to be subsets of \mathcal{N} such that all customers at any node of the set receives service at the same rate. Let \mathcal{N} be divided into K such subsets \mathcal{S}_k . We have:

$$\psi_i(y) = \frac{y_i}{x_k} \phi_k(x) \quad (3)$$

where $x_k = \sum_{i \in \mathcal{S}_k} y_i$ and $\phi_k(x)$ is the service rate attributed to the set.

Now if the $\phi_k(x)$ are balanced by a function $\Phi(x)$ (i.e., $\phi_k(x) = \Phi(x - e_k)/\Phi(x)$ for $x > 0$ and $k = 1, 2, \dots, K$), it is easy to show that the $\psi_i(y)$ are balanced by

$$\Psi(y) = \Phi(x) \prod_{k=1}^K \binom{x_k}{y_i, i \in \mathcal{S}_k}. \quad (4)$$

Let $\pi(x)$ be the stationary distribution of the number of customers in the equivalence sets. Recognizing that $\pi(x) = \sum_{\{y: \sum_{i \in \mathcal{S}_k} y_i = x_k\}} \chi(y)$ where $\chi(y)$ is given by (2), we deduce:

$$\pi(x) = \pi(0) \Phi(x) \prod_{k=1}^K A_k^{x_k} \quad (5)$$

where $A_k = \sum_{i \in \mathcal{S}_k} a_i$. Notice that these probabilities have exactly the same form as (2): the state probabilities are *as if* the equivalent sets were simple exponential servers.

Application to bandwidth sharing Consider now an isolated bottleneck link of capacity C shared fairly by flows arriving according to the Poisson session process defined in the introduction. Assume user access links have a common rate c limiting the maximum rate of any flow.

We represent this stochastic system as a network of two processor sharing service stations: station 1 represents flows in progress on the link, station 2 represents sessions currently in think time. The service rates of these stations are respectively: $\phi_1(x) = \min\{x_1 c, C\}$ and $\phi_2(x) = x_2$. These stations are in turn represented by two PS equivalence subsets of the nodes of a processor sharing Whittle network.

The nature of the Whittle network determines the traffic characteristics at flow level. By constructing a network as large and as complex as necessary

it is possible, for instance, to recreate any phase-type flow size distribution, any distribution of the number of flows per session or any degree of correlation between the sizes of successive flows and think times [13]. Any flow in progress is represented by one and only one customer in one of the nodes of the equivalence set. The point is not that one would wish to construct such a network but that we have exactly the same distribution of the number of flows in progress as in a simple Whittle network. Let A denote demand, the product of the flow arrival rate and the average flow size, and assume $A < C$. Let $p(n)$ be the probability n flows are in progress. It may readily be deduced from (5) that we have:

$$p(n) = p(0) \frac{A^n}{\prod_{1 \leq j \leq n} \min\{j, C\}}. \quad (6)$$

This result was derived in [6,15] as a consequence of the classical stochastic network theory of BCMP [5], Kelly [32] and Cohen [22]. This classical theory shows that a *sufficient* condition for insensitive performance in a processor sharing network is that the service rates $\phi_i(x)$ be balanced. A more recent result due to Bonald and Proutière [12] is that this condition is also *necessary*. The latter result is useful notably in evaluating bandwidth sharing in a network with multiple bottlenecks (see Section 4).

Insensitivity is a highly desirable property in a data network. It means that the only traffic characteristic necessary for predicting the performance of a fairly shared bottleneck is the expected demand A . The performance model is thus robust to the changing popularity of different data transfer applications (email, Web, P2P,...). This can be considered as a data network equivalent of the Erlang loss model for telephone networks where performance similarly depends only on expected demand.

2.2 Throughput performance

Users appreciate the performance of a bottleneck through the response time or, equivalently, the average throughput realized by their flow. Let $R(s)$ denote the response time realized by an arbitrary flow of size s . A simple measure of overall performance is the ratio:

$$\gamma = E[s]/E[R(s)]. \quad (7)$$

This quantity, termed the *equivalent capacity* in [12], is generally quite simple to calculate. In particular, when users have no access rate limit ($c \geq C$), $\gamma = C - A$. The equivalent capacity has some interesting properties in the case of fair sharing. Under the assumptions of the previous section, $R(s)$ is

proportional to s so that γ is a relevant performance parameter for flows of any size (i.e., $\gamma = s/E[R(s)]$).

The expected throughput $E[s/R(s)]$ is generally intractable, on the other hand, and depends on detailed traffic characteristics. Among a number of alternatives suggested by Kherani and Kumar [35] and by Litjens *et al.* [38], is the expected instantaneous throughput, $E[C/X]$, where the X is the random number of flows in progress just after a flow arrival. In the case of fair sharing this, like γ , is an insensitive measure and can be easily derived.

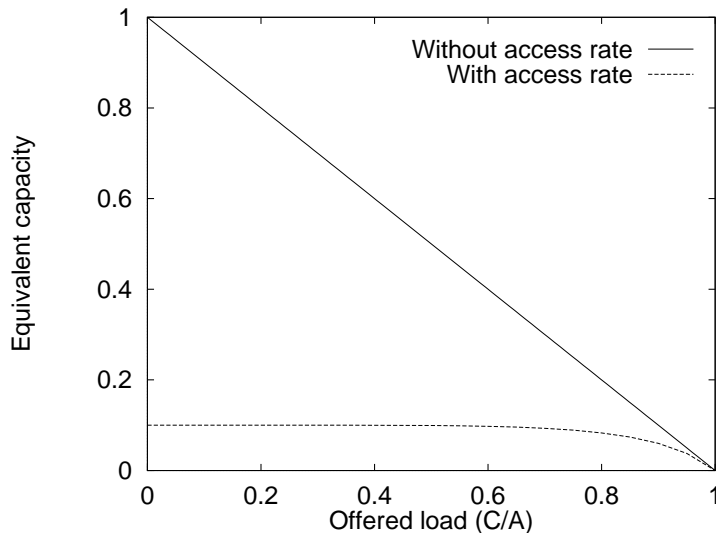


Fig. 1. Equivalent capacity γ against offered load (C/A) in case of fair sharing

In the following we use γ to illustrate the impact on performance of different parameters. Figure 1 presents γ as a function of demand A for different access rates c for the bottleneck defined in Section 2. A significant observation is that a link is virtually transparent with respect to its impact on flow throughput as long as the residual capacity $C - A$ is somewhat greater than the access rate c . For large backbone links ($C = 2.5$ Gbit/s, say) serving users equipped with DSL lines ($c = 2$ Mbit/s, say), the link is *not* a bottleneck until demand attains around 99% of link capacity.

The above observation applies also when flows have relatively small but unequal access rates (DSL, modem, cable, etc.): throughput is practically unconstrained by the link until its load is close to 100%. However, there are no simple formulas for the equivalent capacity under usual definitions of fairness like max-min fairness when per-flow rates are made as equal as possible. An exact evaluation is possible, however, under the assumption of balanced fairness (see Section 4.2).

Results similar to those of the previous section can be derived for a finite source population. The formalism of Whittle networks may again be applied with the usual interpretation of server load in closed stochastic networks. It is again possible to demonstrate that the performance of a fairly shared link is insensitive to the form of the traffic process beyond the assumption that flows and think times occur alternately. Their size and duration can have any distributions and may be correlated. It is possible to define user sessions, for example, with a succession of closely spaced flows being followed by a long think time.

Heyman *et al.* [30] were probably the first to study this model. Berger and Kogan derive approximations valid when the link is always saturated [8]. Both of these works assume the flow – think time succession constitutes an alternating renewal process although this is not strictly necessary.

The models are most appropriate for studying the performance of an access network where a link handles the traffic of a certain number of users. This application is considered specifically in the paper by Bonald *et al.* [11]. In the following we summarize the main results derived from the latter reference.

Assume N users sharing a link of rate C have the same traffic characteristics and a common access rate c . Let V denote the average flow size, T the average response time and S the average think time. Define per-user demand as $a = V/(V/c + S)$ bits/s. This is the average rate a user would realize if he could always transmit at the access rate c . The performance of the access link can be measured through two further rate parameters: $b = V/(T + S)$, the expected per-user carried traffic, and $d = V/T$, the useful flow rate. Note that d corresponds to the equivalent capacity γ introduced in Section 2.2. We thus have the simple relation:

$$\frac{1}{a} + \frac{1}{d} = \frac{1}{b} + \frac{1}{c}. \quad (8)$$

Relation (8) allows the derivation of a simple approximation. We identify two operating regimes: a saturated regime when the link is always busy ($b = C/N$), and a transparent regime when the link does not restrict flow throughput ($b = a$). The approximation consists in evaluating d by ignoring the intermediate link states and assuming the regime changes abruptly from transparent to saturated when the per-user offered traffic increases beyond C/N .

It turns out that, for large enough N , this approximation is accurate compared to an exact evaluation derived from the stochastic network model. It yields a useful dimensioning formula: the required per-user capacity to attain a given

performance target d when offered traffic is a is:

$$C/N \approx f(a) = \frac{1}{1/a + 1/d - 1/c}. \quad (9)$$

Similar approximations may be derived accounting for different user traffic characteristics. It turns out that, because the function $f(a)$ defined in (9) is concave, it is conservative to assume the population is homogeneous: sizing using an average offered traffic a ensures all users have a useful rate no smaller than d . An interesting observation derived from the stochastic network model is that user performance measured by d is roughly independent of the individual traffic offered. This derives from the MUSTA (moving units see time averages) property for closed networks [45]: any user sees the network in the stationary state it would attain if that user were absent. The state probabilities, and therefore the realized throughput, are roughly the same when each user only counts for a small fraction of the total traffic.

3 Unfair sharing

The assumption of fair sharing allows the derivation of simple and general performance results as detailed in the previous section. In practice, fairness is not achieved for a variety of reasons. In this section we examine the impact of sharing that is unfair either deliberately or as a result of imperfections in the transport protocol.

3.1 Discriminatory sharing

In the previous sections we have assumed perfect fair sharing. In practice this is never achieved since the congestion control algorithms of TCP introduce bias due notably to differences in the round trip time. Deliberate bias may also be introduced by differentiated services mechanisms. Unfortunately, when sharing is not perfectly fair, the very attractive property of insensitivity, and much of the modelling tractability, disappears.

Discriminatory processor sharing, where the share attributed to a user of class k is proportional to a weight w_k , was evaluated by Fayolle *et al.* assuming a Poisson arrival process [25]. Throughput performance depends on the distribution of flow sizes. The equivalent capacities for a two-class system assuming exponential flow sizes are as follows:

$$\gamma_1 = \frac{C - w_1 A_1 - w_2 A_2}{C - w_1 A} (C - A) \quad (10)$$

$$\gamma_2 = \frac{C - w_1 A_1 - w_2 A_2}{C - w_2 A} (C - A) \quad (11)$$

where the A_i are per-class demands and $A = A_1 + A_2$. It turns out that performance is not highly sensitive to the flow size distribution except in extreme cases.

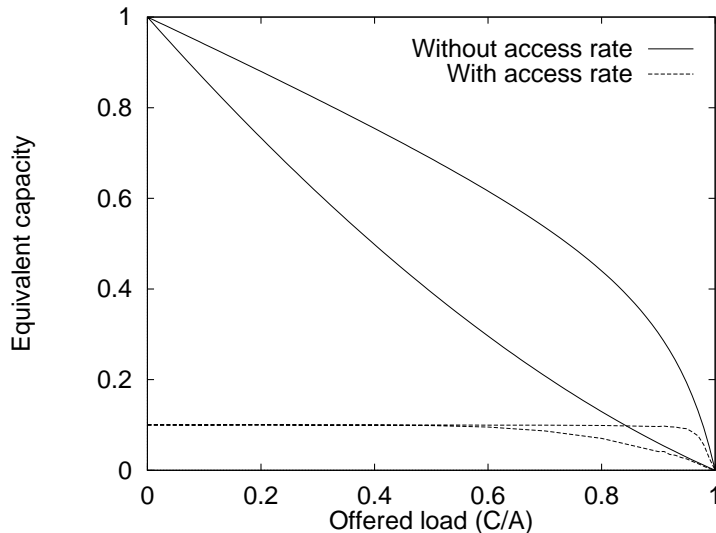


Fig. 2. Equivalent capacity γ against offered load (C/A) when flows of type 1 (top) receive ten times more bandwidth than flows of type 2 (bottom)

Figure 2 shows γ_1 and γ_2 when $w_1/w_2 = 10$. The ratio γ_1/γ_2 is generally much closer to 1 than 10. The figure also shows simulation results for an access rate $c = C/10$. The impact of discrimination is then negligible except when A is very close to C . This observation is significant when considering the possible efficacy of service differentiation mechanisms. There is little scope for realizing different quality of service classes in the network core since flow throughput is generally determined by other constraints exemplified here by the access line rate.

3.2 Size-dependent scheduling

Fairness is not necessarily a desirable networking objective. Gains in performance can in particular be achieved by scheduling flow transmissions depending on their size [40]. The *shortest remaining processing time first* (SRPT) service discipline is known to optimize the overall expected response time [46,2]. The longest flows are actively discriminated against in order to improve the throughput of short flows. However, when the flow size distribution is heavy-

tailed (e.g., a Pareto distribution), Bansal and Harchol-Balter have shown that the penalty incurred by long flows is negligible or non-existent, except under heavy load [4].

The advantage of size-dependent scheduling can be most readily exploited in the context of Web servers where the flow size is generally available [23,29]. Its use has also been proposed for bandwidth sharing in the network notably by Yang and de Veciana [49] and by Rai *et al.* [44]. In this case, it is necessary to use somewhat less efficient scheduling disciplines, based on the volume of data already transmitted, for instance, since it is impossible to know the size of a flow in advance. The performance of SRPT and derived service disciplines is sensitive to traffic characteristics but is tractable for Poisson flow arrivals and general flow size distributions. Performance tends to be better as the variance of the latter distribution increases.

Despite the performance advantage, the appeal of size-dependent scheduling is somewhat mitigated in the case of backbone links. Flow throughput is then limited in general by the user's access rate, except when link load is close to saturation. When demand exceeds capacity, the largest flows suffer denial of service. This does preserve the performance of the majority of smaller flows but it is not obvious that a network operator would wish in practice to discriminate systematically against the longest flows. Note also that the gain in throughput due to size-based scheduling is realized mainly by the smallest flows. For many applications, this gain may be negligible as the overall response time includes much more significant incompressible components such as the TCP handshake or a DNS look-up.

3.3 *Sharing realized by TCP*

The processor sharing network model of bandwidth sharing ignores many features of the real world where flows are controlled by TCP. The slow-start algorithm of TCP introduces bias against short flows that do not have time to ramp up to their fair share. The additive increase – multiplicative decrease congestion avoidance algorithm introduces bias since the rate realized by a long flow is inversely proportional to its round trip time. Some authors have endeavoured to more faithfully take account of the packet level mechanisms of TCP to derive more accurate estimates of throughput performance.

Kherani and Kumar [35] propose a model taking account of the way TCP adjusts flow rates while retaining the fair sharing assumption of the PS model. The approach of Garetto *et al.* [28] is to represent the system as an open multi-class queueing network where each node is an infinite server station representing some state in the evolution of a flow such as the initial slow-start

or a time-out phase. Baccelli and Hong [3] have developed an original approach using the notion of billiards to precisely model the evolution of the TCP rate. Other authors, notably Gibbens et al. [27], Bu and Towsley [20] and Lassila et al. [37], have developed models at packet level, using a fixed-point method to relate packet loss rates and the number of active flows.

The above modelling approaches are clearly necessary if the objective is to evaluate the performance of TCP or to design more efficient active queue management algorithms. They tend to be complex, however, and do not yield convenient closed form formulas. The advantage of the analytical PS queueing network model presented in Section 2.1 above is that it more clearly reveals the impact of the underlying traffic process.

Even though the conditions for the insensitivity properties of this model are not realized in practice, we can be fairly confident that actual performance does not depend significantly on detailed flow and session characteristics, given the assumption of Poisson session arrivals. Similarly, the preponderant impact of a relatively low access rate is also present in the real network.

4 Bandwidth sharing in a network

In this section we return to the fluid model of Section 2.1 in considering bandwidth sharing in a network. We represent the network as a set of links $\mathcal{L} = \{1, \dots, L\}$ where link $l \in \mathcal{L}$ has a capacity $C_l > 0$. We distinguish a number K of classes where flows of class k use the same route r_k , consisting of a particular set of links, and have maximum rate c_k (corresponding to an access line rate, for example). We only consider allocations that are fair between flows of the same class. Possible allocations differ depending on the way link bandwidth is shared between classes. Let x_k be the number of class k flows in progress and denote by $\phi_k(x)$ the overall service rate of class k in state $x = \{x_1, \dots, x_K\}$. The allocation must satisfy the capacity constraints:

$$\sum_{k:l \in r_k} \phi_k(x) \leq C_l, \quad l = 1, \dots, L, \quad \text{and} \quad \phi_k(x) \leq x_k c_k, \quad k = 1, \dots, K. \quad (12)$$

4.1 Utility-based fairness

Bandwidth sharing in a network is frequently evaluated in terms of a utility function, as introduced by Kelly *et al.* [33]. In this case, the utility of a flow of class k , $U_k(\lambda)$, is assumed to be an increasing concave function of its rate λ . The objective is to realize the allocation that maximizes overall utility, i.e.,

for a given flow population x , to chose $\phi_k(x)$ to maximize:

$$\sum_{k=1}^K x_k U_k \left(\frac{\phi_k(x)}{x_k} \right), \quad (13)$$

Examples of possible utility functions are $U_k = \log \lambda$, leading to so-called proportional fairness of Kelly et al. [33], and $U_k = \lambda^{1-\alpha}/(1-\alpha)$ for $0 < \alpha < \infty$, leading more generally to α -fairness defined by Mo and Walrand [41]. Max-min fairness arises in the limit $\alpha \rightarrow \infty$ while proportional fairness corresponds to $\alpha \rightarrow 1$. In the limit $\alpha \rightarrow 0$, the objective is to maximize overall throughput to the detriment of fairness. More general notions of weighted fairness can be defined by multiplying the utility function by a class-dependent weight.

The rationale for utility-based fairness is clear for permanent flows where indeed the utility may reasonably be assumed to depend on the (constant) flow throughput. However, under the more realistic assumption adopted here where flows have a finite size, utility is more reasonably a function of response time.

Among the class of α -fair utility maximizing allocations, it is not clear which is preferable in terms of overall response time performance. Note, however, that the throughput maximizing allocation (corresponding to $\alpha \rightarrow 0$) can lead to catastrophic response time performance for classes with long routes as the underlying queueing system becomes unstable [10].

There are very few analytical results available for the throughput performance of α -fair allocations under random traffic. This is mainly because the performance of these networks is not insensitive and depends significantly on detailed traffic characteristics such as the flow size distribution [13]. Even under the simplest Markovian assumptions, it appears virtually impossible to derive practically useful results for any but the most trivial networks, as illustrated in the paper by Fayolle *et al.* [24]. Fortunately, the notion of Whittle processor sharing networks introduced in Section 2.1 provides an alternative powerful notion of fairness allowing the evaluation of general networks, as discussed in the next section.

4.2 *Balanced fairness*

Consider the following processor sharing Whittle network. An equivalence set of nodes \mathcal{S}_k , for $k = 1, \dots, K$, represents different phases of the different kinds of flows of class k . The total population of all nodes in the equivalence set \mathcal{S}_k is the number of class k flows in progress x_k . An additional equivalence set \mathcal{S}_{K+1} represents the think times.

This is a straightforward extension of the stochastic network defined for a single bottleneck link in Section 2.1. Recall that the equivalence sets can have a very general structure with transitions between any pairs of nodes in any sets, as necessary to model general phase-type distributions and arbitrary correlation between successive flow sizes and think times. The only requirement is that all customers at nodes of set \mathcal{S}_k receive the same service rate.

Bonald and Proutière show that a necessary and sufficient condition for insensitivity is that the rates $\phi_k(x)$ be balanced by some positive function $\Phi(x)$, i.e., $\phi_k(x) = \Phi(x - e_k)/\Phi(x)$, for $k = 1, \dots, K$ [13]. In this case, the joint probability $\pi(x)$ that the number of class k flows in progress is x_k , for $k = 1, \dots, K$, is given by (5) where the A_k are the class k demands. The number of sessions in think time can similarly be derived.

Any positive function $\Phi(x)$ such that the capacity constraints (12) are satisfied defines an insensitive allocation. It is possible to define a unique most efficient allocation by ensuring that the capacity constraints are attained on at least one link in any state x . This allocation is called the *balanced fair* allocation and is defined recursively by $\Phi(0) = 1$ and, for $x > 0$,:

$$\Phi(x) = \max \left(\max_l \left\{ \frac{1}{C_l} \sum_{k:l \in r_k, x_k > 0} \Phi(x - e_k) \right\}, \max_{k:x_k > 0} \left\{ \frac{1}{a_k x_k} \Phi(x - e_k) \right\} \right). \quad (14)$$

Formulas (14) and (5) allow the derivation of performance parameters like the equivalent capacity γ_k of each flow class, at least for some toy network topologies. Efficient formal and numerical algorithms have been developed by Virtamo and co-authors [16]. In particular, an algorithm is provided for a single bottleneck with multiple flow classes distinguished by their access rate c_k . This constitutes a bandwidth sharing equivalent of the multi-rate Erlang loss system.

4.3 Bounds on performance

The formulas of the previous section do not constitute practically useful network design tools for operational IP networks. It is necessary to dispose of simple rules applicable to isolated links. Bounds on the performance of networks with a balanced fair allocation derived by Bonald and Proutière in [14] provide such rules. It is expected that the same rules apply for other allocations realizing a similar degree of fairness such as proportional fair and max-min fair allocations.

It is proved in [14] that the expected response time $R_k(s)$ of a class k flow of

size s satisfies:

$$\max_{l \in r_k} \left\{ \frac{s}{c_k}, \frac{s}{C_l - A_l} \right\} \leq R_k(s) \leq \frac{s}{c_k} + \sum_{l \in r_k} \frac{s}{C_l - A_l}. \quad (15)$$

The lower bound states that the response time is greater than that which would result if any link or the access line were used in isolation. The upper bound corresponds to the response time that would arise if the network were used in a store and forward fashion with transfers on each hop fairly sharing the link bandwidth. These bounds are intuitively appealing though not easy to prove (particularly the upper bound). They very clearly illustrate that the bottleneck is defined by the link with the smallest residual capacity (or the access line) and that if a clear bottleneck exists, the impact of other links on throughput is negligible.

5 Performance in overload

Under demand overload, the infinite source model of bandwidth sharing becomes unstable in that the number of flows increases indefinitely. In this section we discuss stability conditions and consider system performance under overload.

5.1 Stability conditions

The single bottleneck system of Section 2.1 is stable if and only if demand A is less than link capacity C . In a network with a fair allocation of bandwidth the natural condition for stability is:

$$A_l < C_l, \quad \forall l \in \mathcal{L}. \quad (16)$$

Lee *et al.* [47] and Fayolle *et al.* [24] proved that condition (16) is indeed necessary and sufficient for a network realizing max-min fairness under the assumptions of Poisson arrivals of independent exponentially distributed flow sizes. Bonald and Massoulié extended this result for all α -fair allocations for $\alpha > 0$ (i.e., excluding the throughput maximizing allocation) [10] under the same traffic assumptions. In the case of balanced fairness, Bonald and Proutière have shown that condition (16) is necessary and sufficient for stability under the general Poisson session traffic [13].

In networks where flows of some classes have priority over others, instability can occur under conditions that are more restrictive than (16). Bonald and Massoulié exhibit examples where the population of active flows increases

indefinitely under link loads considerably less than 100% [10]. This occurs when the priority mechanism discriminates against flows on the longer paths.

SRPT and SRPT-like scheduling on an isolated bottleneck preserves the performance of flows less than a certain critical size σ . The value of σ is such that demand due to flows no greater than σ is less than link capacity C . The price of stability is service denial for the larger flows. Note that size-dependent scheduling might lead to instability under traffic conditions more restrictive than (16) if there is correlation between flow size and path length resulting a priority system like those considered in [10].

It is interesting at this point to note that the bandwidth sharing system undergoes a kind of phase change as demand increases beyond the stability limit. Consider, for example, the isolated bottleneck of Section 2.1. In the absence of an access rate limit ($c \geq C$), the distribution of the number of flows in progress when $A < C$ is $p(n) = (1 - A/C)(A/C)^n$. For a link of relative load 0.9, the probability of seeing more than 100 active flows is 3×10^{-5} .

That backbone Internet links actually see orders of magnitude more than 100 simultaneous flows when their load is considerably less than 0.9 is due to the fact that the throughput of these flows is limited elsewhere, by their access line for instance. To study fair sharing of an isolated link under an assumption that the number of flows and the link capacity are both very large is therefore of relative interest, despite the attractiveness of resulting diffusion [34] or mean field approximations [21]. If the number of flows competing for a fair share is large, this is rather a sign that the link or network is in overload. Alternative modelling approaches are then appropriate as discussed below.

5.2 *Transient behaviour in overload*

In this section we consider the transient behaviour of a bottleneck link under demand overload. Somewhat surprisingly, analysis of this system yields simple analytical results under the assumption of fair sharing or discriminatory fair sharing.

The overloaded processor sharing server with Poisson flow arrivals and independent identically distributed customer requirements was evaluated by Jean-Marie and Robert [31]. They notably showed that, unlike the stable processor sharing queue, performance depends significantly on the customer service distribution $F(s)$. As the duration of an overload persists, the rate at which customers complete service is shown to attain an asymptotic limit. This rate

μ is the solution to the equation:

$$\mu = \lambda \int_{s \geq 0} e^{-(\lambda - \mu)s/C} dF(s) \quad (17)$$

where λ is the flow arrival rate.

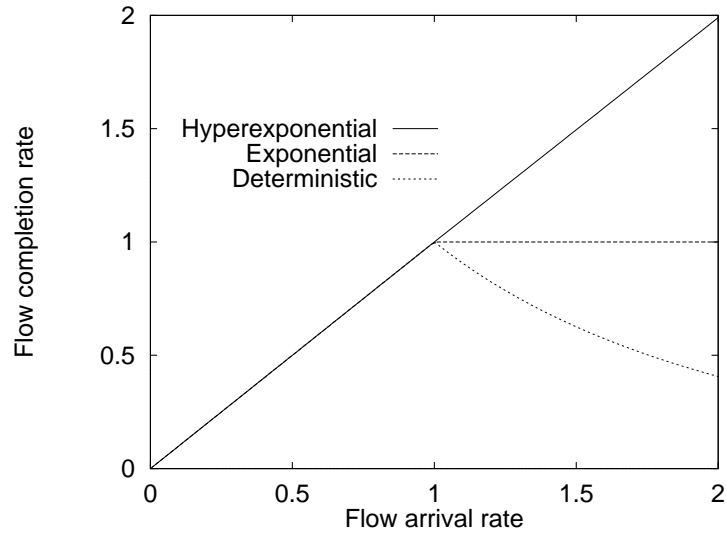


Fig. 3. Asymptotic flow completion rate against arrival rate for different document size distributions

Figure 3 shows how μ depends on the server load ($\lambda \int s dF/C$) for three distributions: deterministic, exponential and hyperexponential with a very high coefficient of variation (≈ 200). The figure shows that the completion rate remains high in the practically interesting case where the variance of F is large (Internet flow sizes are known to have a heavy-tailed distribution). Brown and Collange have noted that this implies that overload in the Internet can be very slow to have an impact [19].

Altman *et al.* have recently generalized the above results to the case of discriminatory sharing, as considered in Section 3.1 [1]. They also demonstrate that equation (17) and its multi-class equivalent apply for any arrival process having a rate and not just for a Poisson process.

5.3 Impatience

In the event of prolonged overload, per-flow throughput tends to zero. Eventually, some flows will be interrupted either because a user loses patience or the underlying application can no longer be sustained. Here we refer to all such causes of interruption as impatience.

A number of models of a fairly shared single link accounting for user impa-

tience have been proposed in the literature: Massoulié and Roberts propose a Markovian model [39], Yang and de Veciana evaluate the impact of several hypothetical patience behaviours by simulation [48], Boyer *et al.* evaluate a reduced service rate approximation [18], Bonald and Roberts exploit the fact that the number of active flows is fairly constant in this system to evaluate the impact of different traffic and impatience models [17].

All these approaches rely on hypothetical models of user impatience, none of which has been verified by measurements on real networks. In fact, it appears virtually impossible to observe impatience in practice by analysing packet traces. For instance, the models assume impatience implies the interruption of a *flow* whereas a more frequent reaction to slow response times would be the early termination of a *session*. The latter event is impossible to recognize. Impatience behaviour also depends on the application with recent peer to peer transfers being particularly resilient.

Despite this imprecision, the following qualitative conclusions drawn from the cited studies should remain true:

- impatience stabilizes the performance of overloaded links but leads to loss of efficiency since capacity is wasted on non-completed flows,
- the number of interrupted flows and the amount of wasted capacity waste due to incomplete transfers decrease as the variance of flow sizes increases,
- to increase patience is generally bad for performance since the proportion of wasted capacity increases,
- accounting for the renewal of interrupted flows exacerbates the negative impact of overloads and user impatience.

5.4 Admission control

The authors of [18] and [17] recognize that bandwidth sharing efficiency in overload would be improved if it were possible to perform pro-active admission control rather than relying on user impatience to stabilize the system. Admission control consists in rejecting a new flow on its arrival in order to preserve the performance of flows already in progress. Several proposals have been made for lightweight admission control adapted to the rapid dynamics of Internet traffic [36,42,7]. In this section we simply point out the possible impact of admission control.

When a link is in overload, its overall throughput is virtually equal to its capacity C . Under fair sharing assumptions, if the number of flows in progress is N , each has an instantaneous rate C/N . In this ideal system, admission control might be employed to ensure N never exceeds a certain value N_{\max} , say. If N_{\max} is large (≥ 100 , say), in practice the variation of N is relatively

small and we can say with a reasonable conservative approximation that each flow experiences an equivalent capacity of $\gamma = C/N_{\max}$.

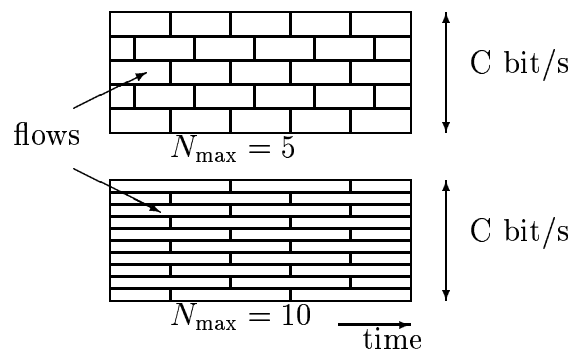


Fig. 4. Impact of admission threshold N_{\max} on realized flow throughput

Figure 4 depicts flows saturating a link with two possible choices of N_{\max} differing by a factor of two. The figure represents flows as rectangles whose left and right sides correspond to the flow start and completion, respectively. The height gives the realized throughput so that the flow size is equal to the area of the rectangle. In the figure, the flows have constant size and their starting times are synchronized to make the drawings look like brick walls. The aim of the figure is to illustrate the impact of choosing a higher or lower value for N_{\max} .

In the upper diagram, the flows have double the throughput of flows in the lower diagram. This improvement in performance is obtained without loss of efficiency (the link is saturated in both cases) and without any increase in the blocking probability. It thus appears preferable to set N_{\max} to as low a value as possible. In practice, there is a lower limit below which there is a perceptible loss of efficiency and increase in blocking probability. Evaluations performed by Ben Fredj et al. suggest a reasonable value of N_{\max} is around 100 [7].

If admission control is not applied, the height of the "bricks" in Figure 4 tends to stabilize at a low value corresponding to the limit of patience of the portion of flows that are abandoned. It is preferable to choose an admission threshold such that the throughput of admitted flows is sufficiently high to avoid users experiencing impatience. Of course, practical admission control schemes cannot use an admission condition based only on the number of flows since, as we have seen, many flows are limited in rate by their access line and could not attain a throughput as high as C/N_{\max} . The scheme described in [7] uses a measurement-based estimate of the bandwidth a new flow could attain accounting for all the limitations of flows in progress.

6 Conclusions

The study of statistical bandwidth sharing, occurring when network capacity is shared by finite size elastic flows occurring as a random process, is an important part of Internet traffic theory. In this survey we have highlighted the appealing and practically interesting insensitivity properties of a form of fair sharing called balanced fairness. Expected flow throughput performance is then a simple function of the demand and capacity of each link.

The impact of real world protocols and their imperfections tends to be of secondary importance. Similarly, nominally optimal size-dependent sharing schemes like SRPT tend to have a minor impact on performance. In conditions of normal load, performance is governed by the link with the tightest residual capacity, i.e., the smallest difference between its bit rate and expected demand. This bottleneck is frequently the user's own access line.

A phase change in performance occurs whenever demand on any network link exceeds its capacity. Per-flow throughput tends to zero as the arrival rate of new flows exceeds the completion rate. Impatience stabilizes the stochastic system but typically at a sub-optimal level of performance. Pro-active flow-level admission control avoids throughput deterioration and constitutes an effective overload control. Size-dependent scheduling realizes a form of admission control since the largest flows are effectively denied service. However, while this minimizes the number of blocked flows, it is by no means obvious that the outcome is desirable from a network performance point of view.

While much progress has been made on understanding statistical bandwidth sharing, there remains considerable scope for further research. The behaviour of a network realizing fair sharing objectives like max-min or proportional fairness appears to be fairly insensitive and similar to that of a balanced fair network. It would nevertheless be useful to understand the limits of their dependence on detailed traffic characteristics. Bandwidth sharing when a flow may choose between alternative routes or can use several routes simultaneously (as with certain peer to peer protocols) largely remains to be evaluated. While this survey has assumed elastic traffic shares dedicated bandwidth, the reality is that network links also handle incompressible streaming flows. The capacity available for statistical bandwidth sharing is then itself a random process. Finally, the environment of wireless networks poses yet further challenges since the capacity of the shared medium is not measured in terms of bandwidth. It is rather a complex function of available power and inter-user interference.

References

- [1] E. Altman, T. Jimenez, D. Kofman, DPS queues with stationary ergodic service times and the performance of TCP in overload, Proceedings of IEEE Infocom 2004.
- [2] F. Baccelli, P. Bremaud, *Elements of Queueing Theory*, Applications of Mathematics 26, Springer-Verlag, 1994.
- [3] F. Baccelli and D. Hong, Interaction of TCP Flows as Billiards, Proceedings of IEEE Infocom 2003.
- [4] N. Bansal, M. Harchol-Balter, Analysis of SRPT Scheduling: Investigating Unfairness. Proceedings of ACM Sigmetrics 2001.
- [5] F. Baskett, K.M. Chandy, R.R. Muntz and F.G. Palacios, Open, closed and mixed networks of queues with different classes of customers, J. Assoc. Comput. Mach. 22 (1975) 248–260.
- [6] S. Ben Fredj, T. Bonald, A. Proutière, G. Régnié, J. Roberts, Statistical bandwidth sharing: a study of congestion at flow level, in: *Proc. of ACM SIGCOMM 2001*.
- [7] S. Ben Fredj, S. Oueslati-Boulahia, and J.W. Roberts. Measurement-based Admission Control for Elastic Traffic. In *Teletraffic Engineering in the Internet Era*. ITC 17, Elsevier, December 2001.
- [8] A. Berger, Y. Kogan, Dimensioning bandwidth for elastic traffic in high-speed data networks, IEEE/ACM Trans. on Networking, Vol 8, No 5, October 2000.
- [9] D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, 1987.
- [10] T. Bonald and L. Massoulié, Impact of fairness on Internet performance, in: *Proc. of ACM SIGMETRICS 2001*.
- [11] T. Bonald, P. Olivier, J. Roberts, Dimensioning high speed IP access networks, Proceedings of ITC 18, Elsevier, 2003.
- [12] T. Bonald, A. Proutière, Insensitivity in processor-sharing networks, Performance Evaluation 49 (2002) 193–209.
- [13] T. Bonald, A. Proutière, Insensitive bandwidth sharing in data networks, Queueing Systems 44-1 (2003) 69–100.
- [14] T. Bonald, A. Proutière, On performance bounds for balanced fairness, Performance Evaluation 55 (2004) 25-50.
- [15] T. Bonald, A. Proutière, G. Régnié, J.W. Roberts, Insensitivity results in statistical bandwidth sharing, in: *ITC 17th 2001*.
- [16] T. Bonald, A. Proutière, J. Roberts, J. Virtamo, Computational aspects of balanced fairness, Proceedings of ITC 18, Elsevier, 2003.

- [17] T. Bonald, J. Roberts, Congestion at flow level and the impact of user behaviour, *Computer Networks* 42 (2003) 521-536.
- [18] J. Boyer, F. Guillemin, P. Robert, B. Zwart, Heavy tailed M/G/1-PS queues with impatience and admission control in packet networks, *Proceedings of IEEE Infocom 2003*.
- [19] P. Brown, D. Collange, Sharing resources in overload with TCP, submitted paper, 2004.
- [20] T. Bu, D. Towsley, Fixed Point Approximation for TCP behavior in an AQM Network *Proceedings of ACM SIGMETRICS 2001*
- [21] A. Chaintreau, F. Baccelli, D. de Vleeschauwer, D. McDonald, A Mean Field Analysis of Interacting HTTP Flows, to appear in *Proceedings of ACM Sigmetrics 2004*.
- [22] J.W. Cohen, The multiple phase service network with generalized processor sharing, *Acta Informatica* 12 (1979) 245–284.
- [23] M. Crovella, R. Frangioso, and M. Harchol-Balter, Connection Scheduling in Web Servers, *USENIX Symposium on Internet Technologies and Systems (USITS '99)*, Boulder, Colorado, October 1999, pp. 243-254.
- [24] 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: *Proc. of IEEE INFOCOM 2001*.
- [25] G. Fayolle, I. Mitrani, R. Iasnogorodski, Sharing a processor among many job classes, *J. of ACM*, vol 27, No 3, July 1980, 519-532.
- [26] S. Floyd, V. Paxson, Difficulties in simulating the Internet, *IEEE/ACM Trans. on Networking*, 9 (4) 393-403, August 2001.
- [27] R. Gibbens, S. Sargood, C. Van Eijl, F. Kelly, H. Azmoodeh, R. Macfadyen, N. Macfadyen, Fixed-point models for the end-to-end performance analysis of IP networks, in *13th ITC Special Seminar: IP Traffic Management, Modeling and Management*, 2000.
- [28] M. Garetto, R. Lo Cigno, M. Meo, M. Ajmone Marsan, Modeling short-lived TCP connections with open multiclass queuing networks, *Computer Networks* 44 (2004) 153-176.
- [29] M. Harchol-Balter, B. Schroeder, N. Bansal, M. Agrawal. "Size-based Scheduling to Improve Web Performance." *ACM Transactions on Computer Systems* , Vol. 21, No. 2, May 2003.
- [30] D. P. Heyman, T. V. Lakshman, A. L. Neidhardt "A new method for analyzing feedback based protocols with applications to engineering web traffic over the internet," *Proceedings of IEEE SIGMETRICS'97*, pp. 2438, February 1997.
- [31] A. Jean-Marie, P. Robert, On the transient behaviour of the processor sharing queue, *Queueing Systems Theory and Applications*, 17 (1994) 129-136.

- [32] F.P. Kelly, *Reversibility and Stochastic Networks*, Wiley, 1979.
- [33] 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).
- [34] F. Kelly, R. Williams, Fluid model for a network operating under a fair bandwidth sharing policy, to appear, *Annals of Applied Probability*.
- [35] A. Kherani, A. Kumar, Stochastic models for throughput analysis of randomly arriving elastic flows in the Internet, *Proceedings of IEEE Infocom 2002*.
- [36] A. Kumar, M. Hegde, and S.V.R. Anand. NETMASTER : Experiences in Using Nonintrusive TCP Connection Admission Control for Bandwidth Management of an Internet Access Link. *IEEE Communications Magazine*, May 2000.
- [37] P. Lassila, H. van den Berg, M. Mandjes, R. Kooij, An integrated packet/flow model for TCP performance analysis, *Proceedings of ITC 18*, Elsevier, 2003.
- [38] R. Litjens, H. van den Berg, R. Boucherie, Throughput measures for processor sharing models, COST 279 document (03)022, 2003.
- [39] L. Massoulié and J.W. Roberts. Arguments in Favor of Admission Control for TCP Flows. In *Teletraffic Engineering in a Competitive World*, pages 33–44. ITC 16, Elsevier, June 1999.
- [40] L. Massoulié and J.W. Roberts, Bandwidth sharing and admission control for elastic traffic, *Telecommunication Systems* 15 (2000) 185-201.
- [41] J. Mo and J. Walrand, Fair end-to-end window-based congestion control, *IEEE/ACM Transactions on Networking* 8-5 (2000) 556-567.
- [42] R. Mortier, I. Pratt, C. Clark, and S. Crosby. Implicit Admission Control. *IEEE Journal on Selected Areas in Communications*, December 2000.
- [43] M. Nabe, M. Murata, H. Miyahara, Analysis and modeling of World Wide Web traffic for capacity dimensioning of Internet access lines, *Performance Evaluation* 34 (1998) 249-271.
- [44] I. A. Rai, G. Urvoy-Keller, and E. W. Biersack. Analysis of LAS Scheduling for Job Size Distributions with High Variance. In *ACM Sigmetrics 2003*, pages 218–228, June 2003.
- [45] R. Serfozo, *Introduction to stochastic networks*, Springer, 1999.
- [46] L. Schrage, L. Miller, The queue M/G/1 with shortest processing time first discipline, *Operations Research* 14 (1966) 670-684.
- [47] G. de Veciana, T.-J. Lee and T. Konstantopoulos. Stability and Performance analysis of networks supporting ABR and best effort-like services. *IEEE/ACM Transactions on Networking*, Volume: 9 ,Issue: 1 ,Feb 2001 Pages:2 - 14
- [48] S. Yang, G. de Veciana, Bandwidth sharing: the role of user impatience, *Proceedings of Globecom*, 2001.

- [49] S. Yang, G. de Veciana, Size-based adaptive bandwidth allocation: optimizing the average QoS for elastic traffic, Proceedings of IEEE Infocom 2002.