Bandwidth sharing: objectives and algorithms

L. Massoulié and J. Roberts
CNET-France Télécom
38-40 rue du Général Leclerc,
92794 Issy-Moulineaux Cédex 9
France
{laurent.massoulie, james.roberts}@cnet.francetelecom.fr

Abstract

This paper concerns the design of distributed algorithms for sharing network bandwidth resources among contending flows. The classical fairness notion is the so-called max-min fairness; F. Kelly [8] has recently introduced the alternative proportional fairness criterion; we introduce a third criterion, which is naturally interpreted in terms of the delays experienced by ongoing transfers. We prove that fixed size window control can achieve fair bandwidth sharing according to any of these criteria, provided scheduling at each link is performed in an appropriate manner. We next consider a distributed random scheme where each traffic source varies its sending rate randomly, based on binary feedback information from the network. We show how to select the source behaviour so as to achieve an equilibrium distribution concentrated around the considered fair rate allocations. This stochastic analysis is then used to assess the asymptotic behaviour of deterministic rate adaption procedures.

1 Introduction

In a network like the Internet where a majority of traffic is generated by the transfer of "elastic" documents (files, Web pages, ...), user perceived performance depends critically on the way bandwidth is shared between concurrent flows. The objective is generally to use all available bandwidth to the full while maintaining a certain "fairness" in the allocations attributed to different flows. The most common understanding of fairness in a network is max-min fairness as defined, for example, in [2]: rates are made as equal as possible subject only to the constraints imposed by link capacities. In fact, there appears to be no clear economic reason why max-min sharing should be preferred over some other bandwidth allocation. More rational objectives would be to maximize overall utility accounting for

costs and perceived value or to minimize the expected response time of any transfer. In this paper we discuss possible bandwidth sharing objectives and the design of the flow control algorithms by which they can be achieved. Although we consider here that the network handles a fixed set of flows, it should be noted that bandwidth sharing is generally performed in the context of randomly varying demands as data transfers begin and end. Preliminary investigations on the impact of this random traffic are described in [10].

The appropriateness of the max-min allocation has already been questioned by Kelly [8] who argues that bandwidth should rather be shared so as to maximize an objective function representing the overall utility of the flows in progress. Assuming a logarithmic utility function where the value of a flow increases with allocated bandwidth λ in proportion to $\log \lambda$ results in so-called "proportional fairness". An alternative utility function with decreasing gradient is $(-1/\lambda)$ leading to the bandwidth sharing objective of minimizing the sum of the reciprocal of rates. This objective may alternatively be interpreted as minimizing the overall potential delay of the transfers in progress. All three objectives, max-min fairness, proportional fairness and minimum potential delay, can be generalized to account for deliberate bias in bandwidth allocations according to the value of weights which might, for instance, reflect different tariff options.

While max-min fairness is often the stated objective, it is widely recognized that this is imperfectly achieved by most network flow control protocols. In particular, it turns out that the additive increase, multiplicative decrease congestion avoidance principle [4], as implemented for instance in TCP [7], tends to realize proportional rather than max-min fairness [9]. Max-min fairness can be achieved by explicit rate calculation algorithms such as those studied in the context of the available bit rate (ABR) service class in ATM [1, 6]. However, experience suggests that it is difficult to achieve a satisfactory compromise between simplicity of the algorithm and resulting fairness which generally depends on all nodes implementing the same mechanisms.

Our focus in the present paper is mainly on distributed algorithms which can be implemented without the complexity of explicit rate calculations, either by means of fixed end to end window control or by rate adjustments performed by users in response to binary congestion signals. The study of fixed windows allows us to investigate how bandwidth sharing depends on the queue service discipline implemented in network nodes and to illustrate the impact of the round trip time of the different routes. To analyse algorithms based on users increasing and decreasing their rate in response to a binary congestion signal, we introduce a family of (hypothetical) random search algorithms, assuming instantaneous reactions (i.e., negligible round trip times). The properties of this family can be used to derive the precise

increase/decrease behaviour necessary to realize particular sharing objectives.

Practical bandwidth sharing algorithms must obviously take account of the packetized nature of individual flows and the resulting imprecision in the notion of rate. In the present study, however, we assume perfectly fluid flows, assimilating links to pipes and buffers to reservoirs. The above modelling devices allow a clearer evaluation of the different bandwidth sharing objectives and provide valuable intuition to guide the design of realistic packet-based algorithms. Clearly, however, more extensive investigations would be necessary to bring the present results to the stage of a practical proposition.

In Section 2 we recall the definition of max-min and proportionally fair sharing and their weighted generalizations, and propose an alternative minimal potential delay criterion. Some common bandwidth sharing algorithms are described in Section 3, notably the fixed end to end window for which we show how realized sharing depends on the service discipline implemented in network nodes. A new class of random search distributed algorithms which can achieve a target rate allocation with an arbitrarily small level of noise in response to a binary congestion indication is introduced in Section 4. The behaviour of these random schemes approximates in some sense that of more realistic deterministic schemes and thus allows an investigation of the rate sharing achieved by general increase/general decrease schemes. Section 5 presents preliminary conclusions drawn from the results of the studied bandwidth sharing models.

2 Bandwidth Sharing

In this section we introduce the considered network model with fluid flows and discuss possible bandwidth sharing objectives.

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 $\mathcal{L}$. We note $l \in r$ when route r goes through link l. Let $\mathcal{R}$ denote the set of routes. Note that some subsets of routes may use precisely the same set of links.

In the sequel we assume that the set of flows is fixed. We seek to allocate link bandwidth to the 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:

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

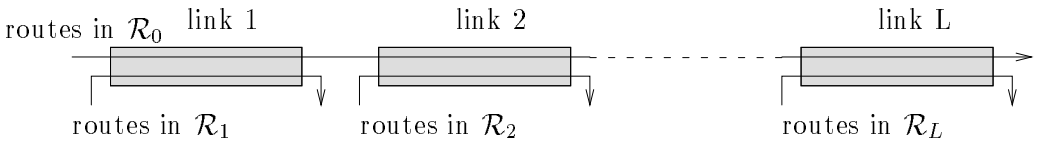

Figure 1: The linear network

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

To illustrate possible allocation strategies we consider the simple linear network depicted in Figure 1. The network consists of L unit capacity links with x_0 long routes which cross every link, and x_l routes which use link l alone, for $1 \leq l \leq L$. Denote by $\mathcal{R}_0$ the set of long routes and by $\mathcal{R}_l$ the set of routes using only link l.

2.2 Sharing Objectives

We now discuss possible objectives in fixing the bandwidth allocations λ_r. 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. More acceptable sharing objectives are discussed below.

2.2.1 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 [2]. The objective stated simply is indeed to maximize the minimum of $\{\lambda_r\}$ 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 $\lambda_{r'} < \lambda_r$. This leads to the following defining condition:

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

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

It is known that there exists only one such allocation when the number of resources and the number of routes are both finite. The max-min fair shares λ_r can then be computed by the following "filling procedure" (see e.g. [2]): start at time 0 with null rate allocations along each route. Increase linearly in time these rate allocations. When at some time the capacity limit is reached at some link, freeze the rate allocation of those routes that go through this link, but proceed with this linear filling for those routes not yet constrained. The desired rate allocation is obtained as the limit of this procedure.

The max-min allocation for the network of Figure 1 is as follows:

$$\lambda_r = \begin{cases} \frac{1}{x_0 + \max_{l \geq 1} x_l} & \text{for} \quad r \in \mathcal{R}_0, \\ \frac{1}{x_l}\left(1 - \frac{x_0}{x_0 + \max_{l \geq 1} x_l}\right) & \text{for} \quad r \in \mathcal{R}_l,\ l \geq 1,\ x_l > 0. \end{cases}$$

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

2.2.2 Proportional fairness

The appropriateness of max-min fairness as a bandwidth sharing objective has recently been questioned by Kelly [8] 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 (1). 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.,

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

Consider how this rate allocation works in the case of the linear network of Figure 1. First it is clear, by concavity of the log function, that all routes in the same set $\mathcal{R}_i$ must have the same allocation. Let γ_i be the allocation of routes in set $\mathcal{R}_i$ for $0 \leq i \leq L$. We necessarily have $x_0\gamma_0 + x_i\gamma_i = 1$ for $1 \leq i \leq L$: 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 $0 \leq i \leq L$, we deduce the allocation $\gamma_0 = 1/(L+1)$ and $\gamma_i = L/(L+1)$ for $i \neq 0$. 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.

2.2.3 Potential delay minimization

Recognizing that flows exist for the transfer of documents, a legitimate bandwidth sharing objective would be to minimize the time delay needed to complete those transfers. In the present static regime, it is more appropriate to consider a potential, rather than actual, flow transfer time equal to the reciprocal of the rate allocation, $1/\lambda_r$. In other words we would seek the allocations minimizing the total potential delay $\sum 1/\lambda_r$. This may alternatively be seen as a utility maximization where the utility function depends on λ_r through a term proportional to $1/\lambda_r$.

Consider the network of Figure 1. Easy calculations yield the following rates γ_i for those routes in $\mathcal{R}_i$:

$$\gamma_0 = \frac{1}{x_0 + \sqrt{\sum_1^L x_j^2}}$$

and

$$\gamma_i = \frac{1}{x_i} \frac{\sqrt{\sum_1^L x_j^2}}{x_0 + \sqrt{\sum_1^L x_j^2}}, \quad x_i > 0, \; i = 1, \ldots, L.$$

In the case where $x_i \equiv 1$, this reduces to $\gamma_0 = (1+\sqrt{L})^{-1}$ and $\gamma_i = \sqrt{L}/(1+\sqrt{L})$, hence an overall throughput of $L + 1 - \sqrt{L}$. This criterion is thus intermediate between the two pevious ones, in that it penalizes more (respectively, less) severely long routes than max-min (respectively, proportional) fairness, resulting in a larger (respectively, smaller) overall throughput.

2.2.4 Weighted shares

All three criteria admit natural generalizations with 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

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

As in the unweighted case, the corresponding allocation can be obtained through a filling procedure, but now the speed of increase of the rate along route r should be ϕ_r. In the case of a single bottleneck link, the allocation to each route is in proportion to its weight, i.e., we have $\lambda_r/\phi_r = \text{constant}$.

A weighted version of the proportional fairness criterion is described in [8]. The rates λ_r are then chosen so as to maximize $\sum_{\mathcal{R}} \phi_r \log \lambda_r$. Equivalently, for any other feasible allocations λ'_r, the aggregate of weighted proportional rate changes with respect to the optimum allocation $\sum_{\mathcal{R}} \phi_r(\lambda'_r - \lambda_r)/\lambda_r$ would be negative. Again, in the case of a single link, the weighted proportionally fair allocations are such that $\lambda_r/\phi_r = \text{constant}$.

Similarly, in its weighted version, the minimum potential delay allocation is that which minimizes $\sum_{\mathcal{R}} \phi_r/\lambda_r$. It coincides with the two previous allocations in the case of a single link.

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.

3 Classical bandwidth sharing algorithms

There are broadly two classes of adaptive bandwidth sharing algorithms which, following ATM terminology, we refer to as "explicit rate" and "congestion indication" algorithms. A simpler alternative is to employ a fixed end to end window on each route. Analysis of the latter algorithm illustrates the impact on allocation fairness of queue service disciplines.

3.1 Explicit rate calculations

By employing the filling procedure described in Section 2.2.1, it would be possible for an omniscient central controller to compute max-min fair shares for all routes and to update allocations as the number of flows or available bandwidth changes. Such a solution is, however, clearly impractical in any moderately large network. Practical explicit rate algorithms are based on the distributed calculation of rate allocations.

The algorithm described by Charny et al [3] converges in a finite number of iterations to an exact max-min fair rate allocation. The algorithm is based on users progressively discovering their rate allocation λ_r by comparison with the "advertised rate" of the links on its route. The advertised rate A_l of link l is given by the formula:

$$A_l = \frac{C_l - \sum_{r \in \Gamma_l} \lambda_r}{n_l - g_l}$$

where Γ_l denotes the subset of routes $r \ni l$ which are constrained (bottlenecked) by any link other than l, g_l is the number of routes in Γ_l and n_l is the total number of routes going through link l. The max-min allocation is characterized by the fact that $\lambda_r < A_l$ for $r \in \Gamma_l$ and $\lambda_r = A_l$ for $r \in l \setminus \Gamma_l$. At each step of an iterative process, the users update an estimate of their rate allocation, setting λ_r to the minimum advertised rate on their route. At the same time, the links progressively discover the members of set Γ_l for which $\lambda_r < A_l$.

Alternative explicit rate algorithms, studied in the context of ABR, are outlined by Arulambalam et al [1]. It appears difficult to find an optimal compromise between achieved fairness, stability, robustness, speed of convergence and link utilization. Explicit rate algorithms generally impose severe processing constraints on network nodes and rely for optimal efficiency on uniform implementation throughout the network.

3.2 Congestion indication

In view of the complexity of explicit rate algorithms, most network flow control protocols are based on simple binary indications of congestion issued independently by the network links. In practice, the condition for defining a state of congestion may depend on buffer occupancy, on measured average input rate or a combination of both.

By studying the impact on the sharing of a single link of various possible reactions to the presence or absence of congestion, Chiu and Jain have demonstrated the optimality of additive increase and multiplicative decrease algorithms [4]: in the absence of congestion, users increase their sending rate linearly until congestion occurs and then begin to decrease the rate exponentially. The rates of increase and decrease must be chosen to limit the amplitude of oscillations which can lead to inefficiencies in link utilization and to ensure rapid convergence when the population of active flows changes.

The additive increase, multiplicative decrease principle is widely implemented in proprietary and standardized protocols, notably in the congestion avoidance algorithms of TCP [7]. Standard user behaviour in ABR in response to the binary congestion indication signal is also based on this principle [1]. It is generally recognized in the ATM community that the congestion indication is less fair than explicit rate due to the so-called "beat down" effect:

flows routed over a long path are more often required to reduce their rate than flows on short routes and are consequently unable to compete fairly.

According to recent results from Kelly et al, the beat down effect may simply be another way of saying that congestion indication algorithms realize proportionally fair rather than max-min fair sharing [9]. More precisely, it is shown in [9] that, ignoring the feedback delay and assuming perfectly fluid traffic, it is possible to create weighted proportionally fair sharing using a common multiplicative decrease factor and an additive increase rate proportional to the required weight. In Section 4 below, we propose an alternative justification for the observation that classical flow control algorithms lead to proportional rather than max-min fair sharing.

3.3 Fixed end to end window control

Reliance on non–adaptive end to end windows is a feasible bandwidth sharing option when link buffers are sufficiently large to eliminate the possibility of data loss.

Assume route r has a window of size B_r (given in bytes, say) and let T_r denote the round-trip time associated with route r, excluding any queueing delay on the forward data transfer path. In general, the use of window control leads to fluctuating rates, i.e., the λ_r vary in time resulting in bursty traffic. However, for present purposes we shall assume that the network is equipped with additional mechanisms which smooth out the bursts, enabling the establishment of a static regime where the λ_r remain constant. In the assumed fluid model, FIFO queueing is sufficient to maintain such a static regime but some further device would be necessary to smooth out the bursts and ensure initial convergence. We do not further pursue the search for such a mechanism, the present aim being to explore how the fairness of the resulting allocations depend on B_r and T_r. We consider here how different sharing objectives are realized depending on the service discipline implemented in network nodes.

3.3.1 Proportional fairness

In the case of FIFO queueing, we have the following

Theorem 1 . *The fluid model under consideration, with non-adaptive end to end window control and FIFO queueing at each link, the window and round trip time of route r being B_r and T_r, respectively, has a unique static regime. The associated stationary rates λ_r on each route are characterized as the unique solution to the optimization problem*

$$\max \sum_{\mathcal{R}} B_r \log \lambda_r - \lambda_r T_r \tag{4}$$

under the constraints $\lambda_r \geq 0$, $\sum_{r \ni l} \lambda_r \leq C_l$.

Proof: Let $B_{l,r}$ denote the volume of traffic from route r currently in the buffer of link l. In the assumed static regime, these quantities, like the λ_r, are constant. Now, at any time, unacknowledged traffic emitted on route r is in one of three states: in transit on the forward path, queued at some link or at destination with an acknowledgement in transit on the backward path. The total volume of traffic in transit in the forward path or whose acknowledgement is in transit on the backward path is equal to $\lambda_r T_r$. We deduce the conservation equation

$$\lambda_r T_r + \sum_{l \in r} B_{l,r} = B_r \tag{5}$$

Assuming servers do not idle, it holds that

$$\sum_{r \ni l} \lambda_r < C_l \Rightarrow B_{l,r} = 0 \text{ for all } r \ni l$$

On the other hand, when the buffers are not empty, because of the assumed static regime and FIFO policy, the output rates are proportional to the buffer contents, i.e.,

$$\sum_{r \ni l} \lambda_r = C_l \Rightarrow \frac{B_{l,r'}}{B_{l,r}} = \frac{\lambda_{r'}}{\lambda_r}, \text{ forall } r, r' \ni l \tag{6}$$

Indeed, in order to maintain the static regime, data packets from different routes should be homogeneously interleaved in the buffer. Denote by $B(l)$ the total buffer content at link l, i.e., $B(l) = \sum_{r \ni l} B_{l,r}$. Summing the previous equation over r',

$$B_{l,r} = \lambda_r \frac{B(l)}{C_l}. \tag{7}$$

Substituting (7) into (5) yields

$$\lambda_r \left[T_r + \sum_{l \in r} \frac{B(l)}{C_l} \right] = B_r \tag{8}$$

where the λ_r and the $B(l)$ are non-negative, and such that for all l, $\sum_{r \ni l} \lambda_r \leq C_l$, and $\sum_{r \ni l} \lambda_r < C_l \Rightarrow B(l) = 0$. The Lagrangian associated with the optimization problem (4) is (the constraints $\lambda_r \geq 0$ need not be included here):

$$\mathcal{L} = \sum_{\mathcal{R}} B_r \log \lambda_r - \lambda_r T_r + \sum_{r \ni l} \mu_l (C_l - \sum_{r \ni l} \lambda_r).$$

According to the Kuhn-Tucker theorem, the optimum is the unique vector satisfying the constraints and such that

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial \lambda_r} = 0, \ \ r \in \mathcal{R}, \\ \mu_l \geq 0; \ \ \sum_{r \ni l} \lambda_r < C_l \Rightarrow \mu_r = 0. \end{cases}$$

The first condition reads

$$\frac{B_r}{\lambda_r} = T_r + \sum_{r \ni l} \mu_l.$$

Setting $\mu_l = B(l)/C_l$ and comparing this with (8) it may readily be verified that any vector of rates λ_r which correspond to a static regime for the fluid model under consideration is a solution of the above maximization problem. Since such a solution is unique, by strict concavity of the objective function, there exists only one such static rate allocation. □

Remark 1 . *When the round trip times are negligible, the objective function in (4) reduces to* $\sum B_r \log \lambda_r$, *so that the static rates constitute the proportionally fair rate allocation with weights given by the window sizes.*

Remark 2 . *When the round trip delays are non-negligible, their impact on the* λ_r *can be assessed from (4). Consider for instance a single link with unit capacity, shared by two routes with associated round trip times* T_i *and window sizes* B_i, $i = 1, 2$. *If* $B_1/T_1 + B_2/T_2 \leq 1$ *then one has* $\lambda_i = B_i/T_i$. *Otherwise, tedious but straightforward calculations yield*

$$\lambda_1 = \frac{2B_1}{T_1 - T_2 + B_1 + B_2 + \sqrt{(T_1 + T_2 - B_1 - B_2)^2 + 4(B_1T_2 + B_2T_1 - T_1T_2)}}$$

and a similar expression holds for λ_2.

3.3.2 Maximum throughput

Theorem 1 relies on the fact that the scheduling policy is FIFO. However, when one uses another policy instead, it turns out that an analogous result often holds, with a suitably modified objective function. This is illustrated by the following theorem.

Theorem 2 . *In the setting of Theorem 1, if each link implements per flow queueing with Longest Queue First (LQF) policy among queues, in any static regime of the system's behaviour, the corresponding stationary rates are uniquely characterized as the solution to the optimization problem*

$$\max \sum_{\mathcal{R}} B_r \lambda_r - \frac{1}{2} \lambda_r^2 T_r \tag{9}$$

under the usual non-negativity and capacity constraints.

Proof: Let $B_{l,r}$ denote the amount of connection r packets backlogged at the access of link l, in some candidate static regime and set $B(l) = \max_{r \ni l} B_{l,r}$. The policy is such that, when $B_{l,r} < B(l)$, one necessarily has $\lambda_r = 0$. When $B_{l,r} = B(l)$, on the other hand,

the policy puts a priori no constraint on the corresponding allocation λ_r. The Lagrangian associated with (9) reads

$$\mathcal{L} = \sum_{\mathcal{R}} B_r \lambda_r - \frac{1}{2}\lambda_r^2 T_r + \sum_l \mu_l (C_l - \sum_{r \ni l} \lambda_r) + \sum_{\mathcal{R}} q_r \lambda_r .$$

At the optimum, we have

$$\lambda_r T_r = B_r - \sum_{l \in r} \mu_l + q_r .$$

Identifying then the Lagrange multipliers μ_l with the maximal buffer contents $B(l)$, this equation is exactly the conservation equation for packets and acknowledgements on route r. Thus in any static regime the stationary rates λ_r solve (9); they therefore do not depend on the static regime under consideration, since (9), being a strictly concave maximization problem, has a unique solution, . □

Remark 3 . *When the round trip delays T_r are negligible, these stationary rates tend to maximize the sum of the throughputs λ_r, weighted by the window sizes B_r.*

3.3.3 Max-min fairness

A particularly interesting allocation results from the use of Fair Queueing scheduling policy. We interpret Fair Queueing in the considered fluid system to imply equal rates for all backlogged flows, and lesser rates for non-backlogged flows.

Theorem 3 . *In the setting of Theorem 1, if at each link one implements a per flow Fair Queueing policy, for any static system behaviour regime, the corresponding stationary rates are uniquely defined as the max-min fair shares of the network's resources with upper bounds B_r/T_r on the λ_r (that is to say, the λ_r are the max-min fair rate shares in a network identical to the one under focus where each route r crosses an additional dedicated access link of capacity B_r/T_r).*

Proof: Consider the conservation equation (5). It ensures that rate λ_r cannot exceed B_r/T_r. It also implies that if $\lambda_r < B_r/T_r$, there necessarily exists some link $l \in r$, such that $B_{l,r} > 0$. For this link l, it then holds that $\sum_{r \ni l} \lambda_r = C_l$, since the associated server is non-idling. Because service at each link is according to a Fair Queueing policy, it also holds that when $B_{l,r} > 0$, $\lambda_r = \max_{r' \ni l}\{\lambda_{r'}\}$.

Summarizing, for all $r \in \mathcal{R}$, $\lambda_r \leq B_r/T_r$, and

$$\lambda_r < B_r/T_r \Rightarrow \text{for some } l \in r, \sum_{r' \ni l} \lambda_{r'} = C_l \text{ and } \lambda_r = \max_{r' \ni l}\{\lambda_{r'}\}$$

Equivalently, these static rates are the max-min fair rate shares with an upper limit on λ_r of B_r/T_r. □

Remark 4 . *When every round trip time T_r is small when compared to the associated window size B_r, the bandwidth limits B_r/T_r are ineffective, so that the stationary rates are the unweighted max-min fair rates. This differs from the situation encountered in Theorems 1 and 2, where the window sizes have a greater impact on the stationary rates, as they translate into weights. In order to achieve stationary rates which correspond to the weighted max-min fair allocation, one should implement weighted fair queueing instead of fair queueing at each link.*

3.3.4 Minimum potential delay

To realize an allocation minimizing the sum of the potential delays as considered in Section 2.2.3, we must invent a rather peculiar queueing discipline.

Theorem 4 . *In the setting of Theorem 1, if at each link one implements per flow queueing with service rate being shared between queues at the prorata of the* ***square roots*** *of the corresponding buffer contents, then for any static regime of the system's behaviour, the associated stationary rates are uniquely characterized as the solution to the optimization problem*

$$\min \sum_{\mathcal{R}} \frac{B_r}{\lambda_r} + T_r \log \lambda_r \tag{10}$$

under the usual non negativity and capacity constraints, and in the domain $\lambda_r \leq B_r/T_r$, $r \in \mathcal{R}$.

Proof: The queueing policy at the prorata of the square roots of the buffer contents ensures that for all l, in some static regime either link l is not saturated and the $B_{l,r}$ are zero for all $r \ni l$, or it is saturated and then

$$\lambda_r = \frac{\sqrt{B_{l,r}}}{\sum_{r' \ni l} \sqrt{B_{l,r'}}} C_l, \quad r \ni l$$

Equivalently,

$$B_{l,r} = \mu_l \lambda_r^2$$

where

$$\mu_l = \left(\frac{\sum_{r' \ni l} \sqrt{B_{l,r'}}}{C_l} \right)^2$$

Substituting this the conservation equation (5) yields

$$B_r = \lambda_r T_r + \lambda_r^2 \sum_{l \in r} \mu_l, \quad r \in \mathcal{R} \tag{11}$$

Consider now the optimization problem (10). It is easily checked that the objective function to be minimized is convex in the domain $\lambda_r \leq B_r/T_r$ (note that stationary rates necessarily

satisfy this constraint) so that the Kuhn-Tucker theorem applies, allowing the following characterization of the optimal values λ_r:

$$\frac{B_r}{\lambda_r^2} - \frac{T_r}{\lambda_r} - \sum_{l \in r} \mu_l = 0, \quad r \in \mathcal{R}$$

where μ_l is the multiplier associated with the capacity constraint at link l, and is thus non negative, and necessarily zero if link l is not saturated. This expression is the same as (11), thus completing the proof of the theorem. □

Remark 5 . *With negligible round trip times, the static regime depicted in the previous theorem realizes the minimum of $\sum_{\mathcal{R}} B_r/\lambda_r$, and is thus the minimum potential sojourn time allocation, with weights given by window sizes.*

4 Random search and deterministic increase/decrease algorithms

Consider now a generic stochastic algorithm of the Metropolis type where routes individually adjust their sending rate according to the evolution of a random process and the assumed instantaneous knowledge of whether a proposed increase would lead to the saturation of any link on its path. The derived algorithms are not proposed as a practical network solution. However, as is shown in this section, their analysis can be used to gain some insight into the properties of deterministic algorithms such as TCP's additive increase/multiplicative decrease congestion avoidance mechanism.

4.1 Distributed random search algorithms

Assume each route r sends data at rate $\lambda_r = \delta\nu_r$, where ν_r is integer-valued and fluctuates between 0 and $n_{\max}$ and δ is a fixed bandwidth unit. The rates ν_r change in a Markovian fashion, jumping from n to $n-1$ with rate d_n, and from n to $n+1$ with rate b_n on condition that this will not lead to capacity being exceeded at some link. First, consider the auxiliary process where each ν_r evolves in a Markovian fashion, jumping from n to $n-1$ at rate d_n and from n to $n+1$ at rate b_n, and this independently of the link status. Clearly, the individual processes ν_r are independent and the joint process has a reversible measure proportional to the weights

$$\pi'(n_1, \ldots, n_R) = \prod_{\mathcal{R}} \frac{b_0 b_1 \cdots b_{n_r - 1}}{d_1 \cdots d_{n_r}}$$

where $R = |\mathcal{R}|$. Now, the process under focus is obtained from this auxiliary process by setting to zero those transition rates which would lead to a violation of some capacity

constraint. Again by standard results for reversible Markov processes, a stationary measure for this process is then given by restricting π' to the configurations which do not violate any capacity constraint. The stationary distribution of the ν_r is thus proportional to the measure

$$\pi(n_1, \ldots, n_R) = \prod_{\mathcal{R}} \frac{b_0 b_1 \cdots b_{n_r - 1}}{d_1 \cdots d_{n_r}} \prod_{\mathcal{L}} \mathbf{1}_{\sum_{r \ni l} \delta n_r \leq C_l}.$$

Different bandwidth sharing objectives can be satisfied by an appropriate choice of b_n and d_n.

4.1.1 Maximum throughput

A first choice for the parameters b_n and d_n is to make them independent of n: $b_n \equiv b$, $d_n \equiv d$. The measure π then takes the form:

$$\left(\frac{b}{d}\right)^{\sum_{\mathcal{R}} n_r} \prod_{\mathcal{L}} \mathbf{1}_{\sum_{r \ni l} \delta n_r \leq C_l}.$$

Thus, when b/d becomes large, the stationary distribution concentrates on those rate allocations which maximize the total throughput $\sum_{\mathcal{R}} \lambda_r$.

4.1.2 Proportional fairness

A second choice consists in setting $b_n = (n+1)^a$, $n \geq 0$, and $d_n = (n-1)^a$, $n \geq 1$, for some parameter $a > 0$. The measure π then reads

$$\exp a \sum_{\mathcal{R}} \log n_r \prod_{\mathcal{L}} \mathbf{1}_{\sum_{r \ni l} n_r \leq C_l}.$$

Thus, when the parameter a increases, the stationary distribution concentrates on the rate allocations which maximize the sum of the logarithms of the rates, within the capacity constraints, i.e., the distribution concentrates on the proportionally fair rate allocations.

4.1.3 Max-min fairness

In order to approximate max-min fair rate sharing we select b_n and d_n such that for all $n \geq 1$, $b_{n-1}/d_n = \exp A^{M-n}$, where A and M are two positive parameters. We could, for instance, set $b_n \equiv \exp A^{M-n-1}$ and $d_n \equiv 1$. We then have

$$\pi(n_1, \ldots, n_R) \propto \exp \sum_{\mathcal{R}} \sum_{i=1}^{n_r} A^{M-i} \prod_{\mathcal{L}} \mathbf{1}_{\sum_{r \ni l} \delta n_r \leq C_l}.$$

Assume now that M has been chosen sufficiently large so that, for any feasible rate allocation δn_r, $M > n_r$. Consider two feasible allocations $\{\delta n_r\}$and $\{\delta m_r\}$, such that for some r_0, $n_{r_0} < m_{r_0}$, and for any other r, either $n_r \leq m_r$ or $m_r > n_{r_0}$. In view of the definition of

max-min fairness, if $\{\delta m_r\}$ is the max-min fair rate allocation, for any other rate allocation $\{\delta n_r\}$ there exists such an r_0. The ratio of probabilities $\pi(m_1,\ldots,m_R)/\pi(n_1,\ldots,n_R)$ is easily seen to be larger than

$$\exp\{A^{M-m_{r_0}} - R\sum_{i=m_{r_0}+1}^{n_{\max}} A^{M-i}\}$$

Thus, when A tends to infinity, this ratio also tends to infinity. In other words, the probability distribution π concentrates on the max-min fair rate allocation as $A \to \infty$.

4.1.4 Minimum potential delay

In order to approximate the minimum potential delay allocation, choose rates b_n and d_n such that

$$\frac{b_{n-1}}{d_n} = \exp -a\left[\frac{1}{n} - \frac{1}{n-1}\right] = \exp\frac{a}{n(n-1)}, \quad n > 1$$

(take for instance $d_n \equiv 1$ and $b_n = \exp a/[n(n+1)]$ for $n \geq 1$, and $d_1 = 0$). The stationary measure π is then proportional to

$$\exp -a\sum_{\mathcal{R}}\frac{1}{n_r} \quad \prod_{\mathcal{L}} \mathbf{1}_{\sum_{r\ni l}\delta n_r \leq C_l}$$

and thus concentrates as $a \to \infty$ on the feasible allocations which minimize the total potential delay $\sum 1/\lambda_r$.

4.2 Deterministic general increase/general decrease algorithms

The above random search framework allows us to derive more practical *deterministic* rate adjustments realizing particular bandwidth sharing objectives.

4.2.1 Additive increase/multiplicative decrease

We first devise rates such that the stochastic algorithm of the previous subsection mimics the additive increase/multiplicative decrease mechanism. Our choice consists in setting $d_n = n$ and $b_n = n + \alpha/\delta$, where α is a positive constant. When upwards transitions are feasible, the drift for λ_r is constant and equal to α, producing a linear increase in the absence of saturation. On the other hand, when upwards transitions are impossible, the drift at some point $x = n\delta$ is exactly $-x$, producing an exponential decay during saturation. In the limit $\delta \to 0$, the rates evolve continuously in a deterministic fashion according to this additive increase/multiplicative decrease mechanism.

Consider two feasible rate vectors $x = \{x_r\} = \{\delta n_r\}$ and $y = \{y_r\} = \{\delta m_r\}$. We investigate the ratio $\pi(x)/\pi(y)$ of the probabilities of each vector in the limit $\delta \to 0$. It is

easily seen that this ratio equals

$$\exp \sum_{\mathcal{R}} \left\{ \sum_{j=x_r/\delta+1}^{(x_r+\alpha)/\delta-1} \log j - \sum_{j=y_r/\delta+1}^{(y_r+\alpha)/\delta-1} \log j \right\}.$$

The value of this expression is not changed on adding $\log \delta$ to each term $\log j$ of both sums which may then be recognized as Riemann sums. The exponent is thus equivalent to:

$$\frac{1}{\delta} \sum_{\mathcal{R}} \int_{x_r}^{x_r+\alpha} \log x dx - \int_{y_r}^{y_r+\alpha} \log x dx.$$

In the limit $\delta \to 0$, the distribution π thus concentrates on the feasible rate configuration which maximizes $\sum_{\mathcal{R}} \int_{x_r}^{x_r+\alpha} \log x dx$. We may conclude that, for small α, this configuration is close to the proportionally fair rate allocation, because the objective function is then equivalent to $\alpha \sum_{\mathcal{R}} \log x_r$. These arguments add support to the belief that additive increase/multiplicative decrease algorithms realize a proportionally fair rate sharing, as has already been advanced by Kelly et al. [9], using a different approach.

4.2.2 General increase/general decrease.

Consider now the following deterministic control policy: rate λ_r increases at speed $f_r(\lambda_r)$ in the absence of congestion and decreases at speed $g_r(\lambda_r)$ under congestion. The previous paragraph dealt with the case $f_r(x) \equiv \alpha$ and $g_r(x) \equiv x$. Applying the same method yields the following result:

Theorem 5 . *The deterministic congestion avoidance algorithm with increase and decrease functions f_r and g_r for route r, $r \in \mathcal{R}$, has equilibrium points at those rate allocations at which the function*

$$\sum_{\mathcal{R}} \int_0^{\lambda_r} \log \frac{f_r(u) + g_r(u)}{g_r(u)} du \tag{12}$$

is maximal.

Proof: Approximate this deterministic system behaviour by that of the stochastic algorithm of the previous subsection, where λ_r jumps from δn to $\delta(n+1)$ at rate $(f_r(\delta n) + g_r(\delta n))/\delta$ in the absence of congestion, and jumps from δn to $\delta(n-1)$ at rate $g_r(\delta n)/\delta$. When δ tends to zero, the behaviour of this system is the same as that of the deterministic system under focus. Let us investigate the limiting behaviour of the stationary distribution π as δ goes to zero. Given two feasible rate allocations $\{\lambda_r\}$, $\{\mu_r\}$, we have

$$\frac{\pi(\lambda_r)}{\pi(\mu_r)} = \exp \sum_{\mathcal{R}} \sum_{n=1}^{\lceil \lambda_r/\delta \rceil} \log \frac{f_r(\delta n) + g_r(\delta n)}{g_r(\delta n)} - \sum_{n=1}^{\lceil \mu_r/\delta \rceil} \log \frac{f_r(\delta n) + g_r(\delta n)}{g_r(\delta n)}$$

As $\delta \to 0$, recognizing Riemann sums in the exponent in the right-hand side, the latter is equivalent to

$$\frac{1}{\delta} \sum_{\mathcal{R}} \int_{\mu_r}^{\lambda_r} \log \frac{f_r(u) + g_r(u)}{g_r(u)} du$$

so that the distribution π concentrates on those allocations for which $\sum_{\mathcal{R}} \int_0^{\lambda_r} \log[(f_r(u) + g_r(u))/g_r(u)]du$ is maximal and the result of the Theorem follows. □

Remark 6 . *If the objective function has a unique global maximum in the domain of admissible rate allocations and no other local maximum, one would expect the deterministic increase/decrease algorithm to converge indeed to that maximising point. However if there are multiple local maxima, it is likely that the deterministic mechanism will get trapped in any such local maximum.*

Theorem 5 may be used either to gain insight into the nature of the equilibria achieved by existing increase/decrease mechanisms, as in the previous paragraph, or conversely to design new increase/decrease mechanisms with pre-specified equilibrium properties. Let us illustrate this by devising functions f and g so that the associated equilibrium points minimize the total potential sojourn time $\sum_{\mathcal{R}} 1/\lambda_r$. The corresponding f and g should be such that $\int_0^{\lambda} \log[(f(u) + g(u))/g(u)]du = -1/u$. Differentiating, we should therefore set

$$f(u) + g(u) = g(u) \exp \frac{1}{u^2} \tag{13}$$

There is a minor difficulty here: for such f and g the corresponding integral diverges at zero. However, it is easy to extend Theorem 5 to the case where some $\log[(f_r(u) + g_r(u))/g_r(u)]$ fails to be integrable at zero, the result then being that the function maximised at equilibrium has the derivative $\sum_{\mathcal{R}} \log[(f_r(u) + g_r(u))/g_r(u)]$.
Returning to (13), if we want to keep the multiplicative decrease half of the TCP congestion avoidance mechanism, we have $g(u) = u$, and thus

$$f(u) = u \left(\exp \frac{1}{u^2} - 1 \right)$$

For large values of u, we have $f(u) \sim 1/u$. Assuming that to set $f(u) = 1/u$ instead of the above does not significantly change the system equilibrium, the following statement makes sense: "logarithmic increase/multiplicative decrease mechanisms lead to rate shares that minimize the total potential delay". Logarithmic increase could be realized by increasing the window size on route r as follows: just after the window size has been increased to n packets, wait 2^n time units before increasing it to $n + 1$.

One might wonder whether for appropriately chosen increase and decrease functions f and g the objective function is maximised at the max-min fair rate allocation. It turns out

that there do not exist functions which guarantee this property to hold for an arbitrary network configuration.

5 Conclusion

The way network bandwidth is shared between contending flows has a significant impact on user perceived performance. We have considered a variety of bandwidth sharing objectives including max-min fairness, proportional fairness and overall delay minimization. In the present work we have concentrated on the protocols and distributed algorithms used to realize these objectives for a given set of flows each having a fixed network route.

The algorithms currently used in data networks generally aim to realize max-min sharing although precision in realizing this objective is often sacrificed in the interest of simplicity. There is evidence that classical congestion indication algorithms based on additive increase / multiplicative decrease tend to produce allocations which are proportionally fair rather than max-min fair. We have illustrated through a simple example how proportional fairness tends to produce smaller allocations on routes using a large number of hops to the advantage of greater overall throughput. Minimizing potential delay as a sharing objective provides an intermediate solution between max-min and proportional fairness, penalizing long routes less severely than the latter.

We have demonstrated that a simple fixed window flow control produces different sharings depending on the scheduling discipline employed in network nodes. For example, FIFO tends to produce weighted proportional fairness, with weights given by the respective window sizes, while fair queueing leads naturally to max-min fairness.

We have approached the problem of designing a distributed algorithm realizing a given sharing objective through the study of a family of so-called Metropolis algorithms. The rate of individual flows varies randomly and independently of the rate of other flows except for the condition that transitions to infeasible states (where link capacities would be exceeded) are barred. By appropriately choosing transition probabilities, it is possible to ensure that the random process concentrates on the rate allocation which realizes the required sharing objective. More practical algorithms are derived as deterministic limits of the stochastic processes. In particular, it is shown by this means that the additive increase / multiplicative decrease algorithm tends to realize proportional sharing, as already shown in [9]. In fact, as in the cited work, the sharing objective is realized under the (unrealistic) assumptions that rate adjustments in response to congestion signals are immediate and that the multiplicative decrease factor tends to one (i.e., rate fluctuations occur in a very limited neighbourhood of the congested state).

To complete the study of how bandwidth sharing algorithms affect user-perceived performance, it is necessary to consider the impact of random changes in the number of flows in progress. Indeed, the bandwidth sharing algorithm has its own impact on this number since the transfer time of a given flow (i.e., a given document) clearly depends on the rate allocated to it. Preliminary investigations on the throughput performance of bandwidth sharing algorithms are reported in [10]. In this context, the natural rate sharing objective would be to minimize the number of transfers in progress and thus, by Little's law, minimize the mean transfer time. This is the motivation behind the potential delay minimization bandwidth sharing introduced here.

References

[1] A. Arulambalam and X.Q. Chen, Allocating fair rates for available bit rate service in ATM networks. IEEE Communications Magazine (1996) 92-100.

[2] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall, 1987.

[3] A. Charny, D. Clark and R. Jain, Congestion control with explicit rate indication. *Proc. ICC '95*, June 1995.

[4] D.M. Chiu and R. Jain, Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Computer Networks and ISDN Systems* 17 (1989) 1-14.

[5] S. Floyd and K. Fall, Router mechanisms to support end-to-end congestion control. Lawrence Berkeley National Laboratory, Preprint (1997).

[6] E. J. Hernandez-Valencia, L. Benmohamed, S. Chong and R. Nagarajan, Rate control algorithms for the ATM ABR service. Europ. Trans. Telecom. Vol 8 (1997), 7-20.

[7] V. Jacobson, Congestion Avoidance and Control. In *Proc. SIGCOMM '88*, 314-329.

[8] F. Kelly, Charging and rate control for elastic traffic. Europ. Trans. Telecom. Vol 8 (1997), 33-37.

[9] 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).

[10] J. Roberts and L. Massoulié, Bandwidth sharing and admission control for elastic traffic, Submitted, 1998