1 Explicit Congestion Control: charging, fairness and admission management

Frank Kelly, Gaurav Raina

University of Cambridge, Indian Institute of Technology Madras

In the design of large scale communication networks, a major practical concern is the extent to which control can be decentralized. A decentralized approach to flow control has been very successful as the Internet has evolved from a small scale research network to today's interconnection of hundreds of millions of hosts; but, it is beginning to show signs of strain. In developing new end-to-end protocols, the challenge is to understand just which aspects of decentralized flow control are important. One may start by asking how should capacity be shared among users? Or, how should flows through a network be organized, so that the network responds sensibly to failures and overloads? Additionally, how can routing, flow control and connection acceptance algorithms be designed to work well in uncertain and random environments?

One of the more fruitful theoretical approaches has been based on a framework that allows a congestion control algorithm to be interpreted as a distributed mechanism solving a global optimization problem; for some overviews see [1, 2, 3]. Primal algorithms, such as the Transmission Control Protocol (TCP), broadly correspond with congestion control mechanisms where noisy feedback from the network is averaged at endpoints, using increase and decrease rules of the form first developed by Jacobson [4]. Dual algorithms broadly correspond with more explicit congestion control protocols where averaging at resources precedes the feedback of relatively precise information on congestion to endpoints. Examples of explicit congestion control protocols include the eXplicit Control Protocol (XCP) [5] and the Rate Control Protocol (RCP) [6, 7, 8].

Currently, there is considerable interest in explicit congestion control. A major motivation is that it may allow the design of a fair, stable, low loss, low delay and high utilization network. In particular, explicit congestion control should allow short flows to complete quickly, and also provides a natural framework for charging. In this Chapter we review some of the theoretical background on explicit congestion control, and provide some new results focused especially on admission management.

In Section 1.1 we describe the notion of proportional fairness, within a mathematical framework for rate control which allows us to reconcile potentially conflicting notions of fairness and efficiency, and exhibits the intimate relationship between fairness and charging. RCP uses explicit feedback from routers to allow fast convergence to an equilibrium and in Section 1.2 we outline a proportionally fair variant of the Rate Control Protocol designed for use in a network where queues are small. In Section 1.3 we focus on admission management of flows where we first describe a step-change algorithm that allows new flows to enter the network with a fair, and high, starting rate. We then study the robustness of this algorithm to sudden, and large, changes in load. In particular, we explore the key trade-off in the design of an admission management algorithm: namely the trade-off between the desired utilization of network resources and the scale of a sudden burst of newly arriving traffic that the network can handle without buffer overload. Finally, in Section 1.4, we provide some concluding remarks.

1.1 Fairness

A key question in the design of communication networks is just how should available bandwidth be shared between competing users of a network? In this Section we describe a mathematical framework which allows us to address this question.

Consider a network with a set J of *resources*. Let a *route* r be a non-empty subset of J, and write $j \in r$ to indicate that route r passes through resource j. Let R be the set of possible routes. Set $A_{jr} = 1$ if $j \in r$, so that resource j lies on route r, and set $A_{jr} = 0$ otherwise. This defines a $0-1$ incidence matrix $A = (A_{jr}, j \in J, r \in R)$.

Suppose that route r is associated with a *user*, representing a higher level entity served by the flow on route r. Suppose if a rate $x_r > 0$ is allocated to the flow on route r then this has *utility* $U_r(x_r)$ to the user. Assume that the utility $U_r(x_r)$ is an increasing, strictly concave function of x_r over the range $x_r > 0$ (following Shenker [9], we call traffic that leads to such a utility function *elastic* traffic). To simplify the statement of results, we shall assume further that $U_r(x_r)$ is continuously differentiable, with $U_r'(x_r) \to \infty$ as $x_r \downarrow 0$ and $U_r'(x_r) \to 0$ as $x_r \uparrow \infty$.

Assume further that utilities are additive, so that the aggregate utility of rates $x = (x_r, r \in R)$ is $\sum_{r \in R} U_r(x_r)$. Let $U = (U_r(\cdot), r \in R)$ and $C = (C_j, j \in J)$. Under this model the system optimal rates solve the following problem.

$SYSTEM(U, A, C)$:

$$\begin{aligned} &\text{maximize} && \sum_{r \in R} U_r(x_r) \\ &\text{subject to} && Ax \leq C \\ &\text{over} && x \geq 0. \end{aligned}$$

While this optimization problem is mathematically fairly tractable (with a strictly concave objective function and a convex feasible region), it involves utilities U that are unlikely to be known by the network. We are thus led to consider two simpler problems.

Suppose that user r may choose an amount to pay per unit time, w_r, and receives in return a flow x_r proportional to w_r, say $x_r = w_r/\lambda_r$, where λ_r could be regarded as a charge per unit flow for user r. Then the utility maximization problem for user r is as follows.

$USER_r(U_r; \lambda_r)$:

$$\begin{aligned} &\text{maximize } U_r\left(\frac{w_r}{\lambda_r}\right) - w_r \\ &\text{over} \qquad w_r \geq 0. \end{aligned}$$

Suppose next that the network knows the vector $w = (w_r, r \in R)$, and attempts to maximize the function $\sum_r w_r \log x_r$. The network's optimization problem is then as follows.

$NETWORK(A, C; w)$:

$$\begin{aligned} &\text{maximize } \sum_{r \in R} w_r \log x_r \\ &\text{subject to } Ax \leq C \\ &\text{over} \qquad x \geq 0. \end{aligned}$$

It is known [10, 11] that there always exist vectors $\lambda = (\lambda_r, r \in R)$, $w = (w_r, r \in R)$ and $x = (x_r, r \in R)$, satisfying $w_r = \lambda_r x_r$ for $r \in R$, such that w_r solves $USER_r(U_r; \lambda_r)$ for $r \in R$ and x solves $NETWORK(A, C; w)$; further, the vector x is then the unique solution to $SYSTEM(U, A, C)$.

A vector of rates $x = (x_r, r \in R)$ is *proportionally fair* if it is feasible, that is $x \geq 0$ and $Ax \leq C$, and if for any other feasible vector x^*, the aggregate of proportional changes is zero or negative:

$$\sum_{r \in R} \frac{x_r^* - x_r}{x_r} \leq 0. \tag{1.1}$$

If $w_r = 1, r \in R$, then a vector of rates x solves $NETWORK(A, C; w)$ if and only if it is proportionally fair. Such a vector is also the natural extension of Nash's bargaining solution, originally derived in the special context of two users [12], to an arbitrary number of users, and, as such, satisfies certain natural axioms of fairness [13, 14].

A vector x is such that the *rates per unit charge* are proportionally fair if x is feasible, and if for any other feasible vector x^*

$$\sum_{r \in R} w_r \frac{x_r^* - x_r}{x_r} \leq 0. \tag{1.2}$$

The relationship between the conditions (1.1) and (1.2) is well illustrated when $w_r, r \in R$, are all integral. For each $r \in R$, replace the single user r by w_r identical sub-users, construct the proportionally fair allocation over the resulting $\sum_r w_r$ users, and provide to user r the aggregate rate allocated to its w_r sub-users; then the resulting rates *per unit charge* are proportionally fair. It is straightforward to check that a vector of rates x solves $NETWORK(A, C; w)$ if and only if the rates per unit charge are proportionally fair.

1.1.1 Why proportional fairness?

RCP approximates the processor-sharing queueing discipline when there is a single bottleneck link, and hence allows short flows to complete quickly [15, 7]. For the processor-sharing discipline at a single bottleneck link, the mean time to transfer a file is proportional to the size of the file, and is insensitive to the distribution of file sizes [16, 15]. Proportional fairness is the natural network generalization of processor-sharing, with a growing literature showing that it has exact or approximate insensitivity properties [17, 18] and important efficiency and robustness properties [19, 20].

In their study of multihop wireless networks, Le Boudec and Radunovic [20] highlight that proportional fairness achieves a good trade-off between efficiency and fairness, and recommend that metrics for the rate performance of mobile ad hoc networking protocols be based on proportional fairness. We also highlight the two-part paper series [21] that study the use of proportional fairness as the basis for resource allocation and scheduling in multi-channel multi-rate wireless networks. Among numerous aspects of their study, the authors conclude that the proportional fairness solution simultaneously achieves higher system throughput, better fairness, and lower outage probability with respect to the default solution given by today's 802.11 commercial products.

Briscoe [22] has eloquently made the case for *cost fairness*, that is, rates per unit charge that are proportionally fair. As Briscoe discusses, it does not necessarily follow that users should pay according to the simple model described above; for example if users prefer ISPs to offer flat rate subscriptions. But to avoid perverse incentives, accountability should be based on cost fairness. For example, ISPs might want to limit the congestion costs their users can cause, not just charge them for whatever unlimited costs they cause.

In the next Section we show that the Rate Control Protocol may be adapted to achieve cost fairness, and further that it is possible to show convergence, to equilibrium, on the rapid time scale of round-trip times.

1.2 Proportionally fair rate control protocol

In this Section we recapitulate the proportionally fair variant of RCP introduced in [23]. The framework we use is based on fluid models of packet flows where the dynamics of the fluid models allows the machinery of control theory to be used to study stability on the fast time scale of round-trip times.

Buffer sizing is an important issue in the design of end-to-end protocols. In rate controlled networks, if links are run close to capacity, then buffers need to be large, so that new flows can be given a high starting rate. However, if links are run with some spare capacity, then this may be sufficient to cope with new flows, and allow buffers to be small. Towards the goal of a low delay and low loss network, it is imperative to strive to keep queues small. In such a regime, the queue size fluctuations are very fast – so fast that it is impossible to control the queue size. Instead, as described in [24, 25], protocols act to control the *distribution* of queue size. Thus, on the time-scale relevant for convergence of the protocol it is then the *mean* queue size that is important. This simplification of the treatment of queue size allows us to obtain a model that remains tractable even for a general network topology. Next we describe our network model of RCP with small queues, designed to allow buffers to be small.

Recall that we consider a network with a set J of resources and a set R of routes. A route r is identified with a non-empty subset of J, and we write $j \in r$ to indicate that route r passes through resource j. For each j, r such that $j \in r$, let T_{rj} be the propagation delay from the source of flow on route r to the resource j, and let T_{jr} be the return delay from resource j to the source. Then

$$T_{rj} + T_{jr} = T_r \qquad j \in r, r \in R, \tag{1.3}$$

where T_r is the round-trip propagation delay on route r: the identity (1.3) is a direct consequence of the end-to-end nature of the signalling mechanism, whereby congestion on a route is conveyed via a field in the packets to the destination, which then informs the source. We assume queueing delays form a negligible component of the end-to-end delay - this is consistent with our assumption of the network operating with small queues.

Our small queues fair RCP variant is modelled by the system of differential equations

$$\frac{d}{dt} R_j(t) = \frac{aR_j(t)}{C_j \overline{T}_j(t)} \left(C_j - y_j(t) - b_j C_j p_j(y_j(t))\right) \tag{1.4}$$

where

$$y_j(t) = \sum_{r:j \in r} x_r(t - T_{rj}) \tag{1.5}$$

is the aggregate load at link j, $p_j(y_j)$ is the mean queue size at link j when the load there is y_j, and

$$\overline{T}_j(t) = \frac{\sum_{r:j\in r} x_r(t) T_r}{\sum_{r:j\in r} x_r(t)} \tag{1.6}$$

is the average round-trip time of *packets* passing through resource j. We suppose the flow rate $x_r(t)$ leaving the source of route r at time t is given by

$$x_r(t) = w_r \left(\sum_{j \in r} R_j(t - T_{jr})^{-1} \right)^{-1} . \tag{1.7}$$

We interpret these equations as follows. Resource j updates $R_j(t)$, the nominal rate of a flow which passes through resource j alone, according to equation (1.4). In this equation the term $C_j - y_j(t)$ represents a measure of the rate mismatch, at time t, at resource j, while the term $b_j C_j p_j(y_j(t))$ is proportional to the mean queue size at resource j. Equation (1.7) gives the flow rate on route r as the product of the weight w_r and reciprocal of the sum of the reciprocals of the nominal rates at each of the resources on route r. Note that equations (1.5) and (1.7) make proper allowance for the propagation delays, and the average round-trip time (1.6) of packets passing through resource j scales the rate of adaptation (1.4) at resource j.

The computation (1.7) can be performed as follows. If a packet is served by link j at time t, $R_j(t)^{-1}$ is added to the field in the packet containing the indication of congestion. When an acknowledgement is returned to its source, the acknowledgement feedbacks the sum, and the source sets its flow rate equal to the returning feedback to the power of -1.

A simple approximation for the mean queue size is as follows. Suppose that the workload arriving at resource j over a time period τ is Gaussian, with mean $y_j \tau$ and variance $y_j \tau \sigma_j^2$. Then the workload present at the queue is a reflected Brownian motion [26], with mean under its stationary distribution of

$$p_j(y_j) = \frac{y_j \sigma_j^2}{2(C_j - y_j)}. \tag{1.8}$$

The parameter σ_j^2 represents the variability of resource j's traffic at a packet level. Its units depend on how the queue size is measured: for example, packets if packets are of constant size, or Kilobits otherwise.

At the equilibrium point $y = (y_j, j \in J)$ for the dynamical system (1.4-1.8) we have

$$C_j - y_j = b_j C_j p_j(y_j). \tag{1.9}$$

From equations (1.8-1.9) it follows that at the equilibrium point

$$p_j'(y_j) = \frac{1}{b_j y_j}. \tag{1.10}$$

Observe that in the above model formulation there are two forms of feedback: rate mismatch and queue size.

1.2.1 Sufficient conditions for local stability

For the RCP dynamical system, depending on the form of feedback that is incorporated in the protocol definition one may exhibit *two* simple sufficient conditions for local stability; for the requisite derivations and associated analysis see [23].

Local stability with feedback based on rate mismatch and queue size. A sufficient condition for the dynamical system (1.4-1.8) to be locally stable about its equilibrium point is that

$$a < \frac{\pi}{4}. \tag{1.11}$$

Observe that this simple decentralized sufficient condition places *no* restriction on the parameters $b_j, j \in J$, provided our modelling assumptions are satisfied.

The parameter a controls the speed of convergence at each resource, while the parameter b_j controls the utilization of resource j at the equilibrium point. From (1.8-1.9) we can deduce that the utilization of resource j is

$$\rho_j \equiv \frac{y_j}{C_j} = 1 - \sigma_j \left(\frac{b_j}{2} \cdot \frac{y_j}{C_j} \right)^{1/2}$$

and hence that

$$\begin{aligned} \rho_j &= \left(\left(1 + \frac{\sigma_j^2 b_j}{8} \right)^{1/2} - \left(\frac{\sigma_j^2 b_j}{8} \right)^{1/2} \right)^2 \\ &= 1 - \sigma_j \left(\frac{b_j}{2} \right)^{1/2} + O(\sigma_j^2 b_j). \end{aligned} \tag{1.12}$$

For example, if $\sigma_j = 1$, corresponding to Poisson arrivals of packets of constant size, then a value of $b_j = 0.022$ produces a utilization of 90%.

Local stability with feedback based only on rate mismatch. One may also derive an alternative sufficient condition for local stability. If the parameters b_j are all set to zero, and the algorithm uses as C_j not the actual capacity of resource j, but instead a target, or virtual, capacity of say 90% of the actual capacity, then this too will achieve an equilibrium utilization of 90%. In this case the equivalent sufficient condition for local stability is

$$a < \frac{\pi}{2}. \tag{1.13}$$

Although the presence of a queueing term is associated with a smaller choice for the parameter a − note the factor two difference between conditions (1.11) and (1.13) − nevertheless the local responsiveness is comparable, since the queueing term contributes roughly the same feedback as the term measuring rate mismatch.

1.2.2 Illustrative simulation

Next we illustrate our small queue variant of the RCP algorithm with a simple packet level simulation in the case where there is feedback based only on rate mismatch.

The network simulated has a single resource, of capacity one packet per unit time and a 100 sources that each produce Poisson traffic. Let us motivate a simple calculation. Assume that the round-trip time is 10000 units of time. Then assuming a packet size of 1000 bytes, this would translate into a service rate of 100Mbytes/s, and a round-trip time of 100ms, or a service rate of 1 Gbyte/s and a round-trip time of 10ms. The figures bearing observations or traces from packet-level simulations were produced using a discrete event simulator of packet flows in RCP networks where the links are modelled as FIFO queues. The round-trip times that are simulated are in the range of 1000 to $100,000$ units of time. In our simulations, as the queue term is absent from the feedback, i.e. $b = 0$, we set $a = 1$ and replace C with γC for $\gamma \in [0.7, \cdots, 0.90]$ in the protocol definition. The simulations were started close to equilibrium.

Figure 1.1 show the comparison between theory and the simulation results, when the round-trip times are in the range of $1,000$ to $100,000$ units of time. Observe the variability of the utilization, measured over one round-trip time, for shorter round-trip times. This is to be expected, since there would remain variability in the empirical distribution of queue size. This source of variability decreases as the bandwidth-delay product increases, and in such a regime there is excellent agreement between theory and simulations.

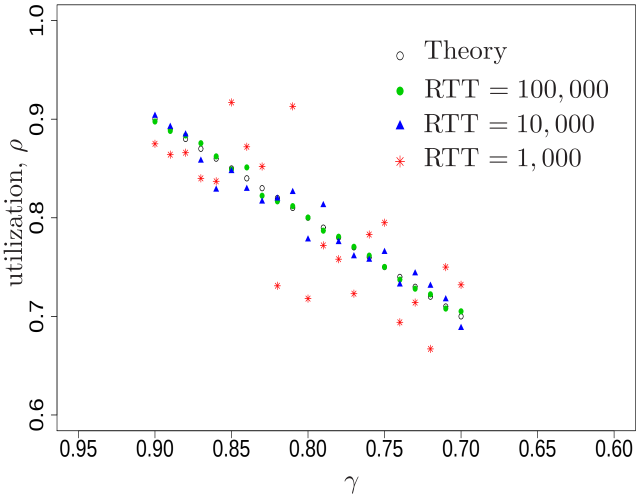


Figure 1.1 Utilization, ρ, measured over one round-trip time, for different values of the parameter γ with a 100 RCP sources that each produce Poisson traffic.

1.2.3 Two forms of feedback?

Rate controlled communication networks may contain two forms of feedback: a term based on rate mismatch and another term based on the queue size.

It has been a matter of some debate whether there is any benefit in including feedback based on rate mismatch *and* on queue size. The systems with and without feedback based on queue size give rise to different nonlinear equations; but, notwithstanding an innocuous-looking factor of two difference, they both yield decentralized sufficient conditions to ensure local stability.

Thus far, as methods based on linear systems theory have not offered a preferred design recommendation – note the simple factor two difference between conditions (1.11) and (1.13) – for further progress it is quite natural to employ nonlinear techniques. For a starting point for such an investigation see [27], where the authors investigate some nonlinear properties of RCP with a conclusion that favours the system whose feedback is based only on rate mismatch.

1.2.4 Tatonnement processes

Mechanisms by which supply and demand reach equilibrium have been a central concern of economists, and there exists a substantial body of theory on the stability of what are termed tatonnement processes [28]. From this viewpoint, the rate control algorithm described in this Section is just a particular embodiment of a *Walrasian auctioneer* searching for market clearing prices. The Walrasian auctioneer of tatonnement theory is usually considered a rather implausible construct; however, we showed that the structure of a communication network presents a rather natural context within which to investigate the consequences for a tatonnement process.

In this Section, we showed how the proportionally fair criteria could be implemented in a large scale network. In particular, it was highlighted that a simple rate control algorithm can provide stable convergence to proportional fairness per unit charge, and be stable even in the presence of random queuing effects and propagation time delays.

A key issue, however, is how new flows should be admitted to such a network, a theme that we pursue in the next Section. The issue of buffer sizing in rate controlled networks is a topical one and the reader is referred to [29], and references therein, for some recent work in this regard. However, our focus in this Chapter will be on developing the admission management process of [23].

1.3 Admission management

In explicit congestion controlled networks when a new flow arrives, it expects to learn, after one round-trip time, of its starting rate. So an important aspect in the design of such networks is the management of new flows; in particular, a key question is the scale of the step-change in rate that is necessary at a resource to accommodate a new flow. We show that, for the variant of RCP considered here, this can be estimated from the aggregate flow through the resource, without knowledge of individual flow rates.

We first describe, in Section 1.3.1, how a resource should estimate the impact upon it of a new flow starting. This suggests a natural step-change algorithm for a resource's estimate of its nominal rate. In the remainder of this Section we explore the effectiveness of the admission management procedure based on the step-change algorithm to large, and sudden, variations in the load on the network.

1.3.1 Step-change algorithm

In equilibrium, the aggregate flow through resource j is y_j, the unique value such that the right hand side of (1.4) is zero. When a new flow, r, begins transmitting, if $j \in r$, this will disrupt the equilibrium by increasing y_j to $y_j + x_r$. Thus, in order to maintain equilibrium, whenever a flow, r, begins R_j needs to be decreased, for all j with $j \in r$.

According to (1.5)

$$y_j = \sum_{r:j\in r} w_r \left(\sum_{k \in r} R_k^{-1} \right)^{-1}$$

and so the sensitivity of y_j to changes in the rate R_j is readily deduced to be

$$\frac{\partial y_j}{\partial R_j} = \frac{y_j \overline{x}_j}{R_j^2} \tag{1.14}$$

where

$$\overline{x}_j = \frac{\sum_{r:j\in r} x_r \left(\sum_{k\in r} R_k^{-1} \right)^{-1}}{\sum_{r:j\in r} x_r}.$$

This $\overline{x}_j$ is the average, over all packets passing through resource j, of the unweighted fair share on the route of a packet.

Suppose now that when a new flow r, of weight w_r, arrives, it sends a request packet through each resource j on its route, and suppose each resource j, on observation of this packet, immediately makes a step-change in R_j to a new value

$$R_j^{new} = R_j \cdot \frac{y_j}{y_j + w_r R_j}. \tag{1.15}$$

The purpose of the reduction is to make room at the resource for the new flow. Although a step-change in R_j will take time to work through the network, the scale of the change anticipated in traffic from existing flows can be estimated from (1.14) and (1.15) as

$$(R_j - R_j^{new}) \cdot \frac{\partial y_j}{\partial R_j} = w_r \overline{x}_j \cdot \frac{y_j}{y_j + w_r R_j}.$$

Thus the reduction aimed for from existing flows is of the right scale to allow one extra flow at the average of the w_r-weighted fair share through resource j. Note that this is achieved without knowledge at the resource of the individual flow rates through it, $(x_r, r : j \in r)$: only knowledge of their equilibrium aggregate y_j is used in expression (1.15), and y_j may be determined from the parameters C_j and b_j as in (1.9).

We now describe an important situation of interest.

Large and sudden changes in the number of flows. It is quite natural to ask about the robustness of any protocol to sudden, and large, changes in the number of flows. A network should be able to cope sensibly to local surges in traffic. Such surges in traffic could simply be induced by a sudden increase in the number of users wishing to use a certain route. Or, such a surge may be induced by the failure of a link, where a certain fraction, or all of the load is transferred to a link which is still in operation.

1.3.2 Robustness of the step-change algorithm

In this subsection we briefly analyse the robustness of the admission control process based on the above step-change algorithm against large, and sudden, increases in the number of flows.

Consider the case where the network consists of a single link j with equilibrium flow rate y_j. If there are n identical flows, then at equilibrium $R_j = y_j/n$. When a new flow begins, the step-change (1.15) is performed and R_j becomes $R_j^{new} = y_j/(n+1)$. Hence equilibrium is maintained. Now suppose that m new flows begin at the same time. Once the m flows have begun, R_j should approach $y_j/(n+m)$. However, each new flow's request for bandwidth will be received one at a time. Thus the new flows will be given rates $y_j/(n+1), y_j/(n+2), \ldots, y_j/(n+m)$. So when the new flows start transmitting, after one round-trip time, the new aggregate rate through j, y_j^{new} will approximately be

$$y_j^{new} \approx n\frac{y_j}{n+m} + \int_n^{n+m} \frac{y_j}{u} du.$$

If we let $\epsilon = m/n$, we have

$$y_j^{new} \approx y_j \left(\frac{1}{1+\epsilon} + \log(1+\epsilon) \right). \tag{1.16}$$

For the admission control process to be able to cope when the load is increased by a proportion ϵ, we simply require y_j^{new} to be less than the capacity of link j. Direct calculation shows that if the equilibrium value of y_j is equal to 90% of capacity, then (1.16) allows an increase in the number of flows of up to 66%. Furthermore, if at equilibrium y_j is equal to 80% of capacity, then the increase in the number of flows can be as high as 122% without y_j^{new} exceeding the capacity of the link.

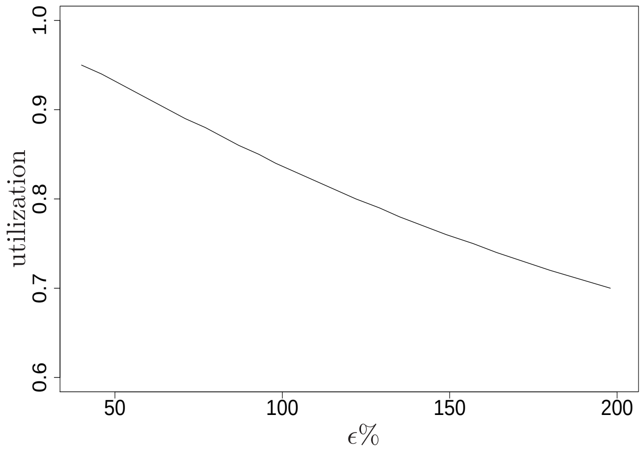


Figure 1.2 Utilization one can expect to achieve and still be robust against an $\epsilon\%$ sudden increase in load; numerical values computed from (1.16).

1.3.3 Guidelines for network management

Figure 1.2 highlights the trade-off between the desired utilization of network resources and the scale of a sudden burst of newly arriving traffic that the resource can absorb. The above analysis and discussion revolves around a single link, but it does provide a simple rule of thumb guideline for choosing parameters such as b_j or C_j. If one takes ϵ to be the largest plausible increase in load that the network should be able to withstand, then from (1.16), one can calculate the value of y_j which gives y_j^{new} equal to capacity. This value of y_j can then be used to choose b_j or C_j, using the equilibrium relationship $C_j - y_j = b_j C_j p_j(y_j)$. There are two distinct regimes that are possible after a sudden increase in the number of flows:

1. If, after the increase, the load y_j remains less than the capacity C_j, then we are in a regime where the queue remains stable. Its stationary distribution (1.8) will have an increased mean and variance, but will not depend on the bandwidth-delay product.
2. If, after the increase, the load y_j exceeds C_j, then we are in a regime where the queue is unstable, and in order to prevent packet drops it is necessary for the buffer to store an amount proportional to the excess bandwidth times the delay.

The approach we advise is to select buffer sizes and utilizations to cope with the first regime, and to allow packets to be dropped rather than stored in the second regime. The second regime should occur rarely if the target utilization is chosen to deal with plausible levels of sudden overload.

1.3.4 Illustrating the utilization-robustness trade-off

We first recapitulate the processes involved in admitting a new flow into an RCP network. A new flow first transmits a request packet through the network. The links, on detecting the arrival of the request packet, perform the step-change algorithm to make room at the respective resources for the new flow. After one round-trip time the source of the flow receives back acknowledgement of the request packet, and starts transmitting at the rate (1.7) that is conveyed back. This procedure allows a new flow to reach near equilibrium within *one* round-trip time. We now illustrate, via some simulations, the admission management procedure for dealing with newly arriving flows.

We wish to exhibit the trade-off between a target utilization, and the impact at a resource of a sudden and large increase in load. Consider a simple network, depicted in Figure 1.3, consisting of 5 links where we do not include feedback based on queue size in the RCP definition and the end-systems produce Poisson traffic. In our simulations, as the queue term is absent from the feedback, i.e. $b = 0$, we replace C_j with $\gamma_j C_j$ for $\gamma_j < 1$, in the protocol definition, in order to aim for a target utilization. The value of a was set at $0.367 \approx 1/e$, to ensure that the system is well within the sufficient condition for local stability. In our experiments, links A, B, C and D each start with 20 flows operating in equilibrium. Each flow uses link X and one of links A, B, C or D. So, for example, a request packet originating from flows entering link C, would first go through link C and then link X before returning back to the source.

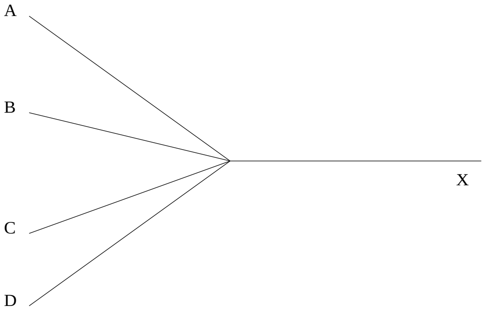


Figure 1.3 Toy network used, in packet-level simulations, to illustrate the process of admitting new flows into a RCP network The links labelled A, B, C, D and X have a capacity of $1, 10, 1, 10$ and 20 packets per unit time, respectively. The physical transmission delays on links A, B and X are 100 time units and on links C and D are 1000 time units.

The experiment we conduct is as follows. The target utilization at all the links is set at 90%, and the scenarios we consider are a 50%, a 100%, and then a 200% instantaneous increase in the number of flows. The choice of these numbers is guided by the robustness analysis above, which is illustrated in Figure 1.2. Since our primary interest is to explore the impact at the resource of a sudden, and instantaneous, increase in load we shall exhibit the impact at one of the ingress links, i.e. link C.

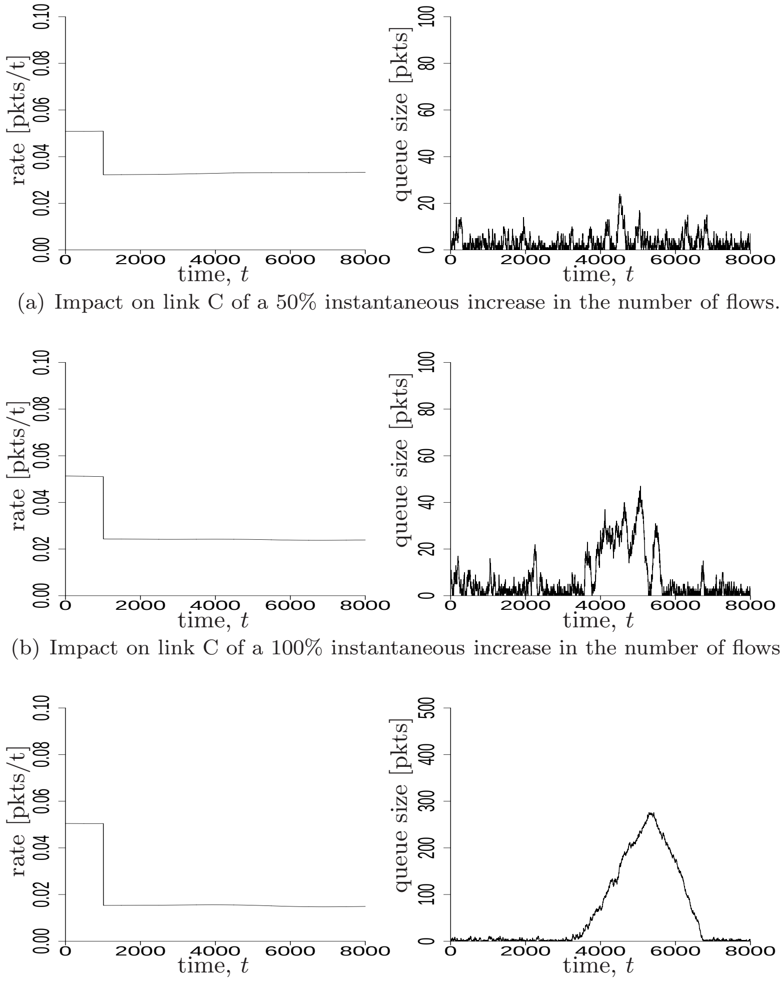
When dealing with new flows there are two quantities that we wish to observe at the resource: the impact on the rate, and the impact on the queue size. In Figure 1.4 (a) (b) (c), the necessary step-change in rate required to accommodate the new flows is clearly visible. The impact on the queue sizes is, however, more subtle. In Figure 1.4 (a), which corresponds to a 50% increase in the number of flows, observe the minor spike in the queue at approximately 4000 time units. The spike in queue size gets more visible when we have a 100% increase in the number of flows; see Figure 1.4 (b). The spike lasts for approximately 2200 time units which is twice the sum of the physical propagation delays along links C and X; the round-trip time of flows originating at link C. With a 200% increase in the number of flows, this spike is extremely pronounced and in fact pushes the peak of the queue close to 300 packets; see Figure 1.4 (c). However, the queue does return to its equilibrium state, approximately one round-trip time later.

Figure 1.4 (a) illustrates the first regime described in Section 1.3.3: after the increase in load the queue remains stable, albeit with an increased mean and variance. Figures 1.4 (b) and (c) illustrate the second regime, where after the increase the load y_j exceeds the capacity C_j. In Figure 1.4 (b) the excess load is relatively small, and there is only a gentle drift upwards in the queue size, with random fluctuations still prominent. In Figure 1.4 (c) the excess load, $C_j - y_j$, causes an approximately linear increase in the queue size over a period of length one round-trip time. Recall that these two cases correspond with respectively a doubling and a tripling of the number of flows.

The above experiments serve to illustrate the trade-off between a target utilization and the impact a large and sudden load would have at a resource. The step-change algorithm helps to provide a more resilient network; one that is capable of functioning well even when faced with large surges in localised traffic. A comprehensive performance evaluation of the step-change algorithm, which forms an integral part of the admission management process, to demonstrate its effectiveness in rate controlled networks is left for further study.

1.3.5 Buffer sizing and the step-change algorithm

TCP is today the de facto congestion control standard that is used in most applications. It's success, in part, has been due to the fact that it has mainly operated in wired networks where losses are mainly due to the overflow of a router's buffer. TCP has been designed to react to, and cope with, losses; the multiplicative decrease component in the congestion avoidance phase of TCP provides a risk averse response when it detects the loss of a packet. Losses, however, constitute damage to packets. This concern is expected to get compounded in environments where bit error rates may not be negligible; a characteristic usually exhibited in wireless networks.



(a) Impact on link C of a 50% instantaneous increase in the number of flows.

(b) Impact on link C of a 100% instantaneous increase in the number of flows.

(c) Impact on link C of a 200% instantaneous increase in the number of flows.

Figure 1.4 Illustration of a scenario with a 50%, 100% then and a 200% increase in the flows which instantaneously request to be admitted into the network depicted in Figure 1.3. The target utilization for all the links in the simulated network is 90%.

In rate controlled networks, loss gets decoupled from flow control and it is possible to maintain small queues in equilibrium and also in challenging situations. A consequence of this is that buffers, in routers, can be dimensioned to be much smaller than the currently used bandwidth-delay product rule of thumb [25] without incurring losses. The role played by the step-change algorithm in ensuring that the queue size remains bounded is exhibited, and so it forms a rather natural component of system design; for example, in developing buffer sizing strategies to minimize packet loss and hence provide a high grade quality of service.

1.4 Concluding remarks

Traditionally, stability has been considered an engineering issue, requiring an analysis of randomness and feedback operating on fast time-scales. On the other hand, fairness has been considered an economic issue, involving static comparisons of utility. In networks of the future this distinction, between engineering and economic issues, is likely to lessen and will increase the importance of an inter-disciplinary perspective. Such a perspective was pursued in this Chapter where we explored issues relating to fairness, charging, stability, feedback and admission management in a step towards the design of explicit congestion controlled networks.

A key concern in the development of modern communication networks is charging and the mathematical framework described enabled us to exhibit the intimate relationship between fairness and charging. Max-min fairness is the most commonly discussed fairness criteria in the context of communication networks. However, it is not the only possibility and we highlighted the role played by proportional fairness in various design considerations ranging from charging, stability and admission management.

Analysis of the fair variant of RCP on the time scale of round-trip times reveals an interesting relationship between the forms of feedback and stability. Incorporating both forms of feedback, i.e. rate mismatch and queue size, is associated with a smaller choice for the RCP control parameter. Nevertheless, close to the equilibrium we expect the local responsiveness of the protocol to be comparable, since the queueing term contributes approximately the same feedback as the term measuring rate mismatch. Analysis of the system far away from equilibrium certainly merits attention; however, it is debatable if both forms of feedback are indeed essential and this issue needs to be explored in greater detail.

As networks grow in both scale and complexity, mechanisms that may allow the self regulation of large scale communication networks are especially appealing. In a step towards this goal, the automated management of new flows plays an important role in rate controlled networks and the admission management procedure outlined does appear attractive. The step-change algorithm that is invoked at a resource to accommodate a new flow is *simple*, in the sense that the requisite computation is done without knowledge of individual flow rates. It is also *scalable*, in that it is suitable for deployment in networks of any size. Additionally, using both analysis and packet level simulations, we developed insight into a fundamental design aspect of an admission management process: there is a trade-off between the desired utilization and the ability of a resource to absorb, and hence be robust towards, sudden and large variations in load.

In the design of any new end-to-end protocol there is considerable interest in how simple, local and microscopic rules, often involving random actions, can produce coherent and purposeful behaviour at the macroscopic level. Towards the quest for desirable macroscopic outcomes, the architectural framework described in this Chapter may allow the design of a fair, stable, low loss, low delay, high utilization and a robust communication network.

Acknowledgement

Gaurav Raina acknowledges funding by the Defense Research Development Organisation (DRDO)-IIT Madras Memorandum of Collaboration.

References

[1] M. Chiang, S.H. Low, A.R. Calderbank and J.C. Doyle. Layering as optimization decomposition: a mathematical theory of network architectures. *Proceedings of the IEEE*, **95** (2007) 255–312.

[2] F. Kelly. Fairness and stability of end-to-end congestion control. *European Journal of Control*, **9** (2003) 159–176.

[3] R. Srikant. *The Mathematics of Internet Congestion Control*, (Boston: Birkhauser, 2004).

[4] V. Jacobson. Congestion avoidance and control. *Proceedings of ACM SIGCOMM* (1988).

[5] D. Katabi, M. Handley and C. Rohrs. Internet congestion control for future high bandwidth-delay product environments. *Proceedings of ACM SIGCOMM* (2002).

[6] H. Balakrishnan, N. Dukkipati, N. McKeown and C. Tomlin. Stability analysis of explicit congestion control protocols. *IEEE Communications Letters*, **11** (2007) 823–825.

[7] N. Dukkipati, N. McKeown and A.G. Fraser. RCP-AC: Congestion control to make flows complete quickly in any environment. *High-Speed Networking Workshop: The Terabits Challenge*, Spain (2006).

[8] T. Voice and G. Raina. Stability analysis of a max-min fair Rate Control Protocol (RCP) in a small buffer regime. *IEEE Transactions on Automatic Control*, **54** (2009) 1908–1913.

[9] S. Shenker. Fundamental design issues for the future Internet. *IEEE Journal on Selected Areas of Communication*, **13** (1995) 1176–1188.

[10] R. Johari and J.N. Tsitsiklis. Efficiency of scalar-parameterized mechanisms. *Operations Research*, **57** (2009) 823–839.

[11] F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, **8** (1997) 33–37.

[12] J.F. Nash. The bargaining problem. *Econometrica*, **28** (1950) 155–162.

[13] R. Mazumdar, L.G. Mason and C. Douligeris. Fairness and network optimal flow control: optimality of product forms. *IEEE Transactions on Communications*, **39** (1991) 775–782.

[14] A. Stefanescu and M.W. Stefanescu. The arbitrated solution for multiobjective convex programming. *Revue Roumaine de Mathématiques Pures et Appliquées*, **20** (1984) 593–598.

[15] N. Dukkipati, M. Kobayashi, R. Zhang-Shen and N. McKeown. Processor sharing flows in the Internet. *Thirteenth International Workshop on Quality of Service*, Germany (2005).

[16] S. Ben Fredj, T. Bonald, A. Proutière, G. Régnié and J.W. Roberts. Statistical bandwidth sharing: a study of congestion at flow level. *Proceedings of ACM SIGCOMM* (2001).

[17] L. Massoulié. Structural properties of proportional fairness: stability and insensitivity. *The Annals of Applied Probability*, **17** (2007) 809–839.

[18] J. Roberts and L. Massoulié. Bandwidth sharing and admission control for elastic traffic. *ITC specialists seminar*, Yokohama (1998).

[19] T. Bonald, L. Massoulié, A. Proutière and J. Virtamo. A queueing analysis of max-min fairness, proportional fairness and balanced fairness. *Queueing Systems*, **53** (2006) 65–84.

[20] J.-Y. Le Boudec and B. Radunovic. Rate performance objectives of multihop wireless networks. *IEEE Transactions on Mobile Computing*, **3** (2004) 334–349.

[21] S.C. Liew and Y.J. Zhang. Proportional fairness in multi-channel multi-rate wireless networks - Parts I and II. *IEEE Transactions on Wireless Communications*, **7** (2008) 3446–3467.

[22] B. Briscoe. Flow rate fairness: dismantling a religion. *Computer Communication Review*, **37** (2007) 63–74.

[23] F. Kelly, G. Raina and T. Voice. Stability and fairness of explicit congestion control with small buffers. *Computer Communication Review*, **38** (2008) 51–62.

[24] G. Raina, D. Towsley and D. Wischik. Part II: Control theory for buffer sizing. *Computer Communication Review*, **35** (2005) 79–82.

[25] D. Wischik and N. McKeown. Part I: Buffer sizes for core routers. *Computer Communication Review*, **35** (2005) 75–78.

[26] J.M. Harrison. *Brownian Motion and Stochastic Flow Systems*, (New York: Wiley, 1985).

[27] T. Voice and G. Raina. Rate Control Protocol (RCP): global stability and local Hopf bifurcation analysis, preprint, online `http://www.statslab.cam.ac.uk/∼gr224`, (2008).

[28] H.R. Varian. *Microeconomic Analysis*, (New York: Norton, 1992).

[29] A. Lakshmikantha, R. Srikant, N. Dukkipati, N. Mckeown and C. Beck. Buffer sizing results for RCP congestion control under connection arrivals and departures. *Computer Communication Review*, **39** (2009) 5–15.

Index