On packet marking at priority queues

R. J. Gibbens and F. P. Kelly

Abstract— **This note concerns charging, rate control and routing for a communication network using priority mechanisms at queues. It is argued that by appropriately marking packets at overloaded resources, end-systems can be provided with the information necessary to balance load across different routes and priorities.**

Keywords—**Congestion pricing, differentiated services, priority queues, rate control, routing.**

I. Introduction

The heterogeneity of Internet applications has led to interest in methods of providing differentiated services. Particular development effort has been devoted to the DiffServ proposal [10], whereby packets are classified into two (or more) classes and queues within the network treat differently packets of different classes.

An alternative, largely theoretical, framework ([2], [4], [7], [13], [14], [17]) has stressed the importance of feedback mechanisms based on explicit congestion signals [6], or marks, interpretable as shadow prices. The development of this approach has assumed simple first-in-first-out queues, with differentiation provided by the algorithms implemented in end-systems. Such a simple queueing discipline may well be appropriate in a network where queueing delays are insignificant compared with propagation delay. But one can envisage circumstances where a slow highly utilized link might cause unacceptable delay to some packets, and where users' perceptions of quality may be affected by their sensitivity to delay.

Alvarez and Hajek [1] extend the framework of [7] to consider a model where there are two classes of packet, the queue gives higher preference to one class of packet, and users pay a higher price for marks on packets of this class. The ratio of prices for marks on packets of different classes is set by the network. Alvarez and Hajek [1] show that it is possible for users willing to receive a reduced throughput to achieve an improved quality of service according to a different measure, such as delay or loss; in this sense the quality of service offered by the network is multidimensional.

Hurley, Le Boudec and Thiran [9] describe an asymmetric best-effort service, where applications mark packets as blue or green, and green packets, typically sent by real-time applications such as interactive audio, receive more losses during bouts of congestion than blue ones. In return, they receive a smaller bounded queueing delay. Queues within the network attempt to ensure that the throughputs on a blue and a green flow that share the same path are in a given ratio, set by the network, through an adaptive scheduler. The incentive to an application to choose blue or green is based on the nature of the application's traffic and on traffic conditions.

Odlyzko [16] describes an approach to pricing differentiated services based on a partition of the network into logically separate channels, which differ only in the prices paid for them. The price per packet would vary from channel to channel: channels with higher prices would attract less traffic and thereby provide a better service.

In this note we explore further some of the issues raised by the above papers. In particular we aim to show how a simple priority queue may be included within the theoretical framework of [7], [13], [14]. Our approach is to view a priority queue as a combination of two logical resources: a first logical resource used only by high priority packets, and a second logical resource used by both high and low priority packets. Each logical resource uses its own packet marking mechanism to indicate incipient congestion, and an end-system makes its own choice of how many packets of each class to send. The choice of packet class by an end-system is logically similar to a routing choice: a high priority packet goes through two logical resources, while a low priority packet goes through just one logical resource. High priority packets may or may not be more likely to be marked, depending upon which logical resources in a network are congested. An advantage of this approach is that no ratio of prices or of throughputs needs to be chosen by the network: the relative prices or throughputs of applications using different classes emerge from the aggregate choices of end-systems.

II. Experiments with a priority queue

A. The model

In this section we consider a single resource offered packets of unit length and consisting of a server capable of serving one such packet per unit time. We identify time slots with epochs occurring every unit of time. The resource can also mark packets during times of congestion, for example using Explicit Congestion Notification [6], and these marks are returned to the sender after some delay. Further details of the strategies used for marking are discussed below.

Users generate packets which can be of two different classes. A user either generates packets from the high priority class, labelled H, or from the low priority class, labelled L. Let J^H be the set of users generating class H packets, J^L the set of users generating class L packets and set $J = J^H \cup J^L$. Each user, of whichever class, has a willingness to pay parameter, w, and operates an elastic user algorithm [1], [7] which transmits

$$X(t) = \lfloor x(t) + z(t) \rfloor^+ \tag{1}$$

RJG is with the Computer Laboratory, University of Cambridge, William Gates Building, JJ Thomson Avenue, Cambridge, CB3 0FD, UK and FPK is with the Statistical Laboratory, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge, CB3 0BW, UK. Emails: Richard.Gibbens@cl.cam.ac.uk and F.P.Kelly@statslab.cam.ac.uk.

packets in the slot $(t, t+1]$, where $x(t)$ and $z(t)$ are internal state variables updated as follows

$$z(t+1) = x(t) + z(t) - X(t) \tag{2}$$
$$x(t+1) = x(t) + \kappa(w - f(t)) . \tag{3}$$

Here $f(t)$ is the number of marks received at the end of slot $(t, t+1]$ and κ is a small positive constant (Johari and Tan [11] and Massoulié [15] treat the choice of the constant κ in a network context).

If we denote by $X_j(t)$ the number of packets produced by user j in time slot $(t, t+1]$ then

$$X^H(t) = \sum_{j \in J^H} X_j(t) \tag{4}$$
$$X^L(t) = \sum_{j \in J^L} X_j(t) \tag{5}$$

give the total numbers of packets of the two classes offered in time slot $(t, t+1]$.

The resource contains two packet buffers, labelled A and B, with occupancies Q^A and Q^B respectively. Buffer A is served with strict priority over buffer B. At time $t+1$ the $X^H(t) + X^L(t)$ packets generated by the users are offered to the resource and may be accepted into the appropriate buffers or lost according to the following procedure. The packets are first sorted into random order and considered in sequence as follows. If the packet is a class H high priority packet then it is accepted by buffer A only if $Q^A < B_1$ and $(Q^A + Q^B) < B_2$, in which case Q^A is increased by one. If the packet is not accepted by the buffer then it is lost. If the packet is a low priority packet of class L then it is accepted by buffer B only if $(Q^A + Q^B) < B_2$, in which case Q^B is increased by one. Again if the packet is not accepted then it is lost. Thus the resource operates as a priority queue with two logical constraints: a constraint on high priority traffic and a constraint on total traffic. A high priority packet of class H must meet both these constraints whereas a low priority packet of class L needs only to meet the constraint on total traffic. A consequence of the service discipline is that a packet accepted in buffer A is guaranteed service within a bounded delay of B_1 time slots. Figure 1 shows a schematic diagram of these two logical resources.

If a packet of either class is lost rather than accepted by the resource then a mark is returned to the user after a timeout period of T_{TO} time slots. Marks may also be given to accepted packets according to the operation of separate marking strategies for each logical constraint. For the high priority constraint, a class H packet is marked in an interval between when a class H packet violates the constraint $Q^A < B_1$ and the next time buffer A is empty. For the constraint on total traffic, a packet of class H or L is marked if it arrives in the interval between when a packet violates the constraint $Q^A + Q^B < B_2$ and the next time both buffers are empty (that is, $Q^A + Q^B = 0$). Thus high priority packets of class H experience two possibilities to be marked whereas low priority packets of class L experience just one. When a marked packet is served a mark is returned to the user after a round trip time of T_{RTT} time slots. Further discussion of marking strategies can be found in [2], [7], [8], [17].

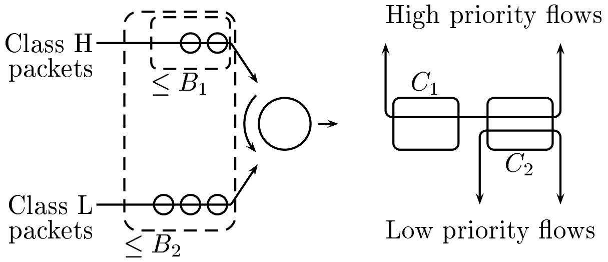

Fig. 1. Schematic diagrams of priority queueing system using two logical constraints. In the Appendix a network abstraction is considered where the priority queue is treated as two resources: a resource of capacity C_1 used by just high priority flows, and a resource of capacity C_2 used by both high and low priority flows.

B. Experiment 1

In this example we vary the sets J^H and J^L of users of the two different classes while keeping their choice of willingness to pay, w_j, held fixed. Suppose that we have $|J| = 40$ users with $w_j = 0.0001 * j$ $(j = 1, \ldots, |J|)$ and the same gain parameter $\kappa = 0.001$. We initially suppose that all users produce packets of class H $(J = J^H, J^L = \emptyset)$ and then choose in random order a member of J^H and transfer it to J^L. We continue with this procedure until all users generate packets of class L $(J = J^L, J^H = \emptyset)$[1].

The resource has a single server which serves packets at unit rate and has loss thresholds of $B_1 = 5$ and $B_2 = 10$. The delays for returning marks to the sending user are $T_{TO} = 200$ and $T_{RTT} = 100$ time slots.

For each of the randomly chosen configurations of users we simulate the behaviour of the resource. Figure 2 shows the long-run proportions of time slots which serve packets of class H and L. The diagonal line shows the constraint on total traffic given by a fully utilized server. The figure shows the nature of the two logical constraints. There is a sloping constraint given by the constraint on total traffic and a vertical constraint given by the constraint on high priority traffic.

Figure 3 shows the marking probabilities for the two logical constraints as a function of the proportion of high priority demand, $\sum_{j \in J^H} w_j$. The marking probabilities for the two constraints are estimated by the ratio of the number of packets marked by the constraint to the number of packets subject to the constraint. (We note that very few high priority packets, less than 0.3%, were marked by *both* constraints. Each such packet carries back only a single mark to the user.)

Also shown in Figure 3 are the loss probabilities for the two packet classes. We can see that loss of high priority packets rises to a potentially unacceptable level of around

[1]In a more general framework, such as that considered in the Appendix, we might allow a single user to send packets in each class, and to vary the proportions: but with 40 users this refinement would have little effect on network level performance statistics.

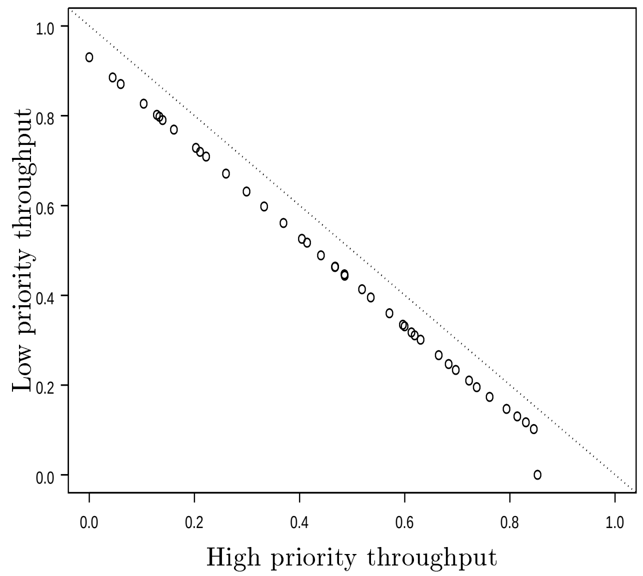

Fig. 2. Class H and L carried loads given by the long-run proportion of time slots which serve packets of high and low priority respectively. Each simulation point corresponds to a different mixture of high and low priority users. The diagonal line shows the constraint on throughput given by a fully utilized server.

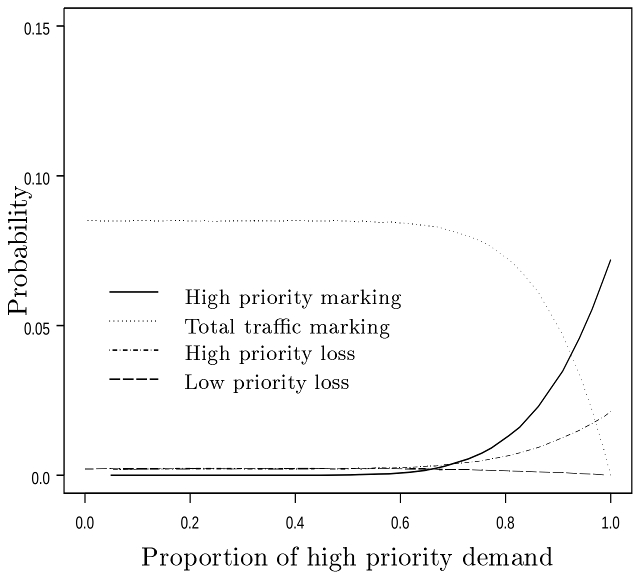

Fig. 3. Marking and loss probabilities as a function of the proportion of high priority demand. The marking probability for the high priority constraint shows the proportion of high priority packets that were marked by the constraint on high priority traffic. The marking probability for the total traffic shows the proportion of packets marked by the constraint on total traffic.

2%. In the next section we address how a marking strategy can give an early warning of the onset of congestion so as to reduce the likelihood of packet loss.

C. Virtual queue marking

In a network where routes comprise many hops, packet loss may not just inconvenience the users concerned, it may also damage the network by causing congested links to occupy themselves sending packets which will only be dropped later in the network [5]. In this section, following [7], we amend the marking strategy to give lower packet loss. Suppose the resource implements two virtual queues, labelled 1 and 2 with occupancies V^1 and V^2, respectively. Associated with virtual queue, i, is a single parameter θ_i with $0 < \theta_i \leq 1$. Virtual queue 1 is used to implement a marking strategy for the logical constraint on high priority traffic and is offered high priority packets only. Virtual queue 2 is used to implement a marking strategy for the constraint on total traffic and is offered packets of both classes accordingly.

The virtual queues are each served at slower rates θ_i $(i = 1, 2)$ than the real queues. A packet offered to virtual queue 1 is accepted if $V^1 + 1 \leq \theta_1 B_1$, in which case V^1 is increased by one. A packet offered to virtual queue 2 is accepted if $V^2 + 1 \leq \theta_2 B_2$, in which case V^2 is increased by one. A major effect of the parameters θ_1 and θ_2 is determining the trade-off between the utilization of resources and packet losses — for further discussion of this issue see [8].

Packets accepted by the real queue are then marked according to the sample path of the virtual queue. Each virtual queue marks arrivals to it in each interval between a virtual loss and its next being empty.

D. Experiment 2

Figure 4 shows the marking and loss probabilities when virtual queues are used with parameters $\theta_1 = 0.8$ and $\theta_2 = 0.9$. The loss probabilities are now insignificantly small with correspondingly higher marking probabilities.

Figure 5 shows the queueing delays experienced by high and low priority packets as a function of the proportion of high priority demand. The queueing delay of high priority packets accepted by the resource is guaranteed to be at most 5 and the figure shows that the 99 percentile of the delay distribution is comfortably less than 5 over most of the range. There is no such bound on the queueing delay of low priority packets as low priority packets must wait until there are no high priority packets before receiving service. Figure 5 shows that as the proportion of high priority demand increases, there is an increase in the variability of delay for low priority packets.

Figure 5 shows that, as the proportion of high priority demand increases from 0 to 0.8, the mean delay increases for low priority packets, *and* for high priority packets. The mean delay over *all* packets is however fairly stable throughout this range. This is consistent with the previous observation, since as the proportion of high priority demand increases, the mix of traffic changes, with fewer packets incurring the larger delays associated with low priority traffic.

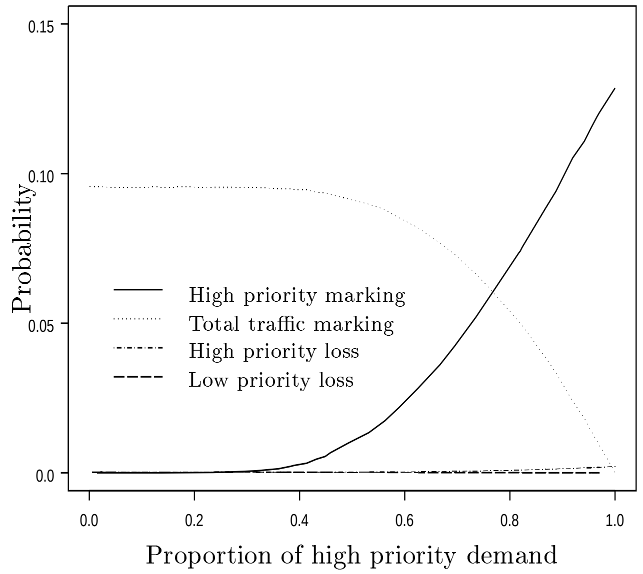

Fig. 4. Marking and loss probabilities with virtual queues. The marking strategy for each logical constraint now follows the sample paths of the virtual queues. The factors, θ_i, can be tuned to give early warning of the onset of congestion.

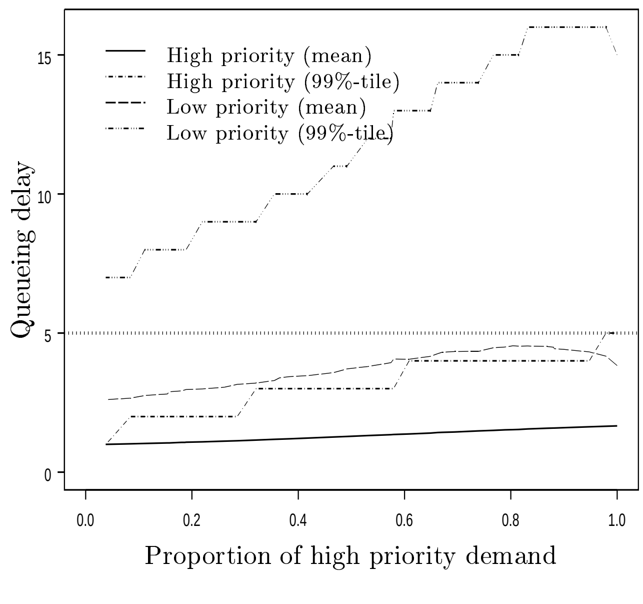

Fig. 5. Queueing delay as a function of the proportion of high priority demand. The 99 percentile of the queueing delay of the high priority packets is comfortably less than the guaranteed bound of 5 over most of the range. As the proportion of high priority demand increases, there is an increase in the variability of delay for low priority packets.

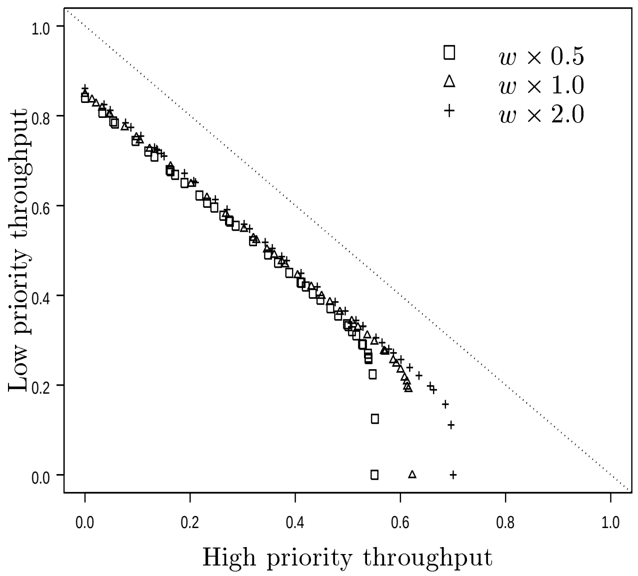

Fig. 6. Effect of uniform scaling of demand by factors of $0.5, 1.0$ and 2.0. The effect on the constraint on total traffic is minimal compared with the effect on the constraint on high priority traffic.

Figure 6 shows the effect of scaling the willingness-to-pay parameters. In the three scenarios the parameters are uniformly scaled by factors of $0.5, 1.0$ and 2.0. The position of the high priority constraint is noticably affected by the scaling, although the low priority constraint is relatively insensitive.

III. Discussion

In Section II we described a marking strategy for a simple priority queue that was interpretable in terms of two logical resources. The choice of packet class by an end-system is then logically similar to a routing choice: a high priority flow goes through two logical resources, while a low priority flow goes through just one logical resource (Figure 1). The theoretical treatment of [12], [13] does include routing choices, but assumes that the utility to a user is a concave function of the sum of the flows achieved by the user along the different routes. In the Appendix we extend the treatment of [12], [13] to allow the utility to a user to be a concave function of the entire vector of flows along the different routes available to a user. This extension is appropriate for the circumstance where propagation delays or bounds on queueing delays differ significantly from route to route.

The major difficulty with using queueing mechanisms for service discrimination was described in the key early paper of Clark [3]: that a simple priority scheme has no means to balance the demands of the various classes. In the approach of this paper this balancing is left entirely to end-systems: high priority flows may or may not be more likely to be marked, depending upon which logical resources in the network are congested.

Acknowledgements

We are grateful to the EPSRC for its support with computing facilities under grant number GR/M09551. RJG is grateful to the Royal Society for the funding of his University Research Fellowship. The authors wish to thank the editor and referees for their thorough review of earlier drafts of this paper.

References

[1] J. Alvarez and B. Hajek, On using marks for pricing in multiclass packet networks to provide multidimensional QoS. Submitted to IEEE Trans. on Automatic Control http://www.comm.csl.uiuc.edu/~hajek/

[2] S. Athuraliya, D. Lapsley and S. H. Low (2000) An enhanced Random Early Marking algorithm for Internet flow control. Proc. Infocom 2000, Israel. p 1425–34.

[3] D. D. Clark (1996). Adding service discrimination to the Internet. *Telecommunications Policy*, **20**, 33-37. http://ana-www.lcs.mit.edu/anaweb/abstracts/TPRC2-0.html.

[4] J. Crowcroft and P. Oechslin (1998) Differentiated end to end Internet services using a weighted proportionally fair sharing TCP. *ACM Computer Communications Review* **28**, 53–67.

[5] S. Floyd and K. R. Fall (1999) Promoting the use of end-to-end congestion control in the Internet *IEEE/ACM Trans Networking*, **7(4)**, 458–472 .

[6] S. Floyd (1994) TCP and Explicit Congestion Notification, *ACM Computer Communication Review* **24**, 10–23. www.aciri.org/floyd/ecn.html

[7] R. J. Gibbens and F.P. Kelly (1999) Resource pricing and the evolution of congestion control, *Automatica* **35**, 1969–1985. www.statslab.cam.ac.uk/~frank/evol.html

[8] R. J. Gibbens, P. B. Key and S. R. E. Turner (2001) Properties of the virtual queue marking algorithm, *IEE Proc 17th UK Teletraffic Symposium, Dublin.* Statistical Laboratory Research Report 2001-10 www.statslab.cam.ac.uk/Reports/

[9] P. Hurley, J. Y. Le Boudec, P. Thiran (1999) The Asymmetric Best-Effort Service. Proceedings of IEEE Globecom, Rio de Janeiro, Brazil, December 1999. http://icawww.epfl.ch/

[10] IETF Differentiated Services (diffserv) working group. http://www.ietf.org/html.charters/diffserv-charter.html

[11] R. Johari and D. K. H. Tan (2000) End-to-end congestion control for the Internet: delays and stability. *To appear IEEE/ACM Trans Networking.* Statistical Laboratory Research Report 2000-2 www.statslab.cam.ac.uk/Reports/

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

[13] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan (1998) Rate control in communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society* **49**, 237–252.

[14] P. Key and D. McAuley (1999) Differential QoS and pricing in networks: where flow control meets game theory. *IEE Proc Software* **146**, 39-43.

[15] L. Massoulié (2000) Stability of distributed congestion control with heterogeneous feedback delays. Microsoft Research Technical Report 2000-111. Submitted to IEEE Trans. on Automatic Control.

[16] A. M. Odlyzko (1999) Paris Metro Pricing for the Internet. Proc. ACM Conference on Electronic Commerce, 140–147. www.research.att.com/~amo.

[17] D. Wischik (1999) How to mark fairly. Workshop on Internet Service Quality Economics, MIT 1999.

Appendix

Consider a network with a set J of *resources*. Let a *route* r be a non-empty subset of J, and write R for the set of possible routes. Let y_r be the flow on route r, and suppose that resource j incurs a cost $C_j(\sum_{r:j\in r} y_r)$ dependent on the flow through that resource, where $C_j(\cdot)$ is an increasing, strictly convex, differentiable function. For example the function $C_j(\cdot)$ may rapidly increase as its argument increases towards the capacity C_j of resource j so that it acts as a penalty function for the capacity constraint. Write $y = (y_r, r \in R)$ for the collection of all flows.

Let $s \in S$ label a user, and suppose s is identified with a subset of R, the routes available to serve the user s, where distinct members of S identify disjoint subsets of R. Write $x_s = (y_r, r \in s)$ for the collection of flows serving user s. Assume the utility to user s of this collection, $U_s(x_s)$, is a real valued strictly concave function. To simplify the statement of results assume further that the derivative of $U_s(x_s)$ with respect to y_r, $U_s^r(x_s)$, is continuous, with $U_s^r(x_s) \to \infty$ as $y_r \downarrow 0$ and $U_s^r(x_s) \to 0$ as $y_r \uparrow \infty$ for $r \in s$.

Consider the following optimization problems (the formulation varies from that of [12], [13] in that the utility $U_s(x_s)$ is a function of a vector, rather than a function a single real variable).

SYSTEM(U,C):

$$\begin{array}{ll} \text{maximize} & \displaystyle\sum_{s\in S} U_s(x_s) - \sum_{j\in J} C_j\Big(\sum_{r:j\in r} y_r\Big) \\ \text{subject to} & x_s = (y_r, r\in s), s\in S \\ \text{over} & y_r \geq 0, r\in R. \end{array}$$

NETWORK$(C;w)$:

$$\begin{array}{ll} \text{maximize} & \displaystyle\sum_{r\in R} w_r \log y_r - \sum_{j\in J} C_j\Big(\sum_{r:j\in r} y_r\Big) \\ \text{over} & y_r \geq 0, r\in R. \end{array}$$

USER$_s(U_s;\lambda)$:

$$\begin{array}{ll} \text{maximize} & \displaystyle U_s(x_s) - \sum_{r\in s} w_r \\ \text{subject to} & x_s = (y_r, r\in s), s\in S \\ \text{and} & w_r = \lambda_r y_r, r\in s \\ \text{over} & w_r \geq 0, r\in s. \end{array}$$

Theorem 1: There exist vectors $\lambda = (\lambda_r, r \in R)$, $w = (w_r, r \in R)$ and $y = (y_r, r \in R)$ such that
(i) $(w_r, r \in s)$ solves USER$_s(U_s;\lambda)$, for $s \in S$;
(ii) y solves NETWORK$(C;w)$;
(iii) $w_r = \lambda_r y_r$ for $r \in R$.
The vector y then also solves SYSTEM(U,C).

Proof: The conditions on the functions U_s, C_j ensure that each of the above optimization problems has a unique optimum, interior to the positive orthant, and that it can be located by first-order stationarity conditions.

The stationarity condition for the optimization problem SYSTEM(U,C) is

$$U_s^r(x_s) = \sum_{j\in r} \mu_j, r\in R, \tag{6}$$

where

$$\mu_j = C_j'\Big(\sum_{r:j\in r} y_r\Big), j \in J \tag{7}$$

and we recall throughout that μ_j is a function of y. The unique vector y fulfilling the stationarity condition, with $x_s = (y_r, r \in s), s \in S$, is the solution to SYSTEM(U, C).

The stationarity condition for the optimization problem NETWORK$(C; w)$ is

$$\frac{w_r}{y_r} = \sum_{j\in r} \mu_j, r \in R. \tag{8}$$

while that for the optimization problem USER$_s(U_s; \lambda)$ is

$$U_s^r(x_s) = \lambda_r, r \in s. \tag{9}$$

Thus if

$$\lambda_r = \sum_{j\in r} \mu_j, r \in R \tag{10}$$

where y is the solution to SYSTEM(U, C), and if $w_r = \lambda_r y_r$, then λ, w, y satisfy the conditions of the Theorem. Conversely if λ, w, y satisfy the conditions of the Theorem they identify a solution to the first-order stationarity condition for the problem SYSTEM(U, C), and hence y solves that problem.

■

Next consider the system of differential equations

$$\frac{d}{dt} y_r(t) = \kappa_r \left(w_r(t) - y_r(t) \sum_{j\in r} \mu_j(t) \right) \tag{11}$$

for $r \in R$, where

$$\mu_j(t) = p_j \left(\sum_{r:j\in r} y_r(t) \right) \tag{12}$$

and $p_j(\cdot)$ is a positive continuous increasing function, for $j \in J$. We interpret the relations (11)–(12) as follows. Suppose that resource j marks a proportion $p_j(z)$ of packets with a feedback signal when the total flow through resource j is z; and that user r views each feedback signal as a congestion indicator requiring some reduction in the flow x_r. Then equation (11) corresponds to a response by user r that comprises two components: a steady increase at rate proportional to $w_r(t)$, and a steady decrease at rate proportional to the stream of feedback signals received.

It is shown in [13] that if $w_r(t) = w_r$ for $r \in R$ then the system of differential equations (11)–(12) has a stable point, to which all trajectories converge. The variable $\mu_j(t)$ can be viewed as the *shadow price* per unit of flow through resource j at time t, and at the stable point

$$y_r = \frac{w_r}{\sum_{j\in r} \mu_j}. \tag{13}$$

The rates y determined by equation (13) have an interpretation as a set of rates that are *proportionally fair per unit charge*, as discussed in [12] and [13].

Next suppose that user s is able to monitor the rates $y_r(t), r \in s$, continuously, and to vary smoothly the parameters $w_r(t), r \in s$, so as to satisfy

$$w_r(t) = y_r(t) U_s^r(x_s(t)) : \tag{14}$$

this would correspond to a user who observes a charge per unit flow of $\lambda_r = w_r(t)/y_r(t)$ on routes $r \in s$, and chooses $w_r = w_r(t), r \in s$, to solve the optimization problem USER$_s(U_s; \lambda)$. Then, with

$$C_j(y) = \int_0^y p_j(z)dz, \tag{15}$$

the objective function of the problem SYSTEM(U, C) is a Lyapunov function for the system of differential equations (11)–(12), (14), and the vector y maximizing the objective function is a stable point of the system, to which all trajectories converge.

Richard Gibbens studied mathematics at the University of Cambridge, graduating with an MA in mathematics, a diploma in mathematical statistics and a PhD in 1988 in the Statistical Laboratory. His doctorial thesis work, a collaboration between the University of Cambridge and British Telecom, was on the design and analysis of dynamic routing strategies in telecommunication networks, resulting in the development of the dynamic alternative routing strategy, which is now in operation in the British Telecom trunk network. He has worked in the area of mathematical modelling of communication networks, mainly at the Statistical Laboratory, University of Cambridge, but he was also a visiting consultant at AT&T Bell Laboratories, Murray Hill, NJ, during 1989. He was appointed to a Royal Society university research fellowship in 1993 and in 2001 he joined the Computer Laboratory, University of Cambridge as a lecturer.

Frank Kelly received his B.Sc. in 1971 from the University of Durham, and his Ph.D. in 1976 from the University of Cambridge. He has held positions in the Engineering and Mathematics Faculties at Cambridge, and served as Director of the Statistical Laboratory from 1991 to 1993. He is currently Professor of the Mathematics of Systems. His main research interests are in random processes, networks and optimization, and especially in applications to the design and control of communication networks. His current research is directed at understanding methods of self-regulation of the Internet.

Frank Kelly has been awarded the Guy Medal in Silver of the Royal Statistical Society, the Lanchester Prize of INFORMS and the Naylor Prize of the London Mathematical Society. He is a Fellow of the Royal Society. He has chaired the Advisory Board of the Royal Institution/Cambridge Mathematics Enrichment Project, and the Management Committee of the Isaac Newton Institute for Mathematical Sciences. He currently serves on the Scientific Council of EURANDOM and the Conseil Scientifique of France Telecom.