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, (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 (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 $\mathrm{USER}_s(U_s;\lambda)$. Then, with

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

the objective function of the problem $\mathrm{SYSTEM}(U, C)$ 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.

ACKNOWLEDGMENT

The authors wish to thank the editor and referees for their thorough review of earlier drafts of this note.

REFERENCES

[1] J. Alvarez and B. Hajek, "On the use of packet classes in communication networks to enhance congestion pricing based on marks," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 1020–1026, June 2002.

[2] S. Athuraliya, D. Lapsley, and S. H. Low, "An enhanced random early marking algorithm for Internet flow control," in *Proc. Infocom 2000*, Tel-Aviv, Israel, 2000, pp. 1425–1434.

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

[4] J. Crowcroft and P. Oechslin, "Differentiated end to end Internet services using a weighted proportionally fair sharing TCP," *ACM Comp. Commun. Rev.*, vol. 28, pp. 53–67, 1998.

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

[6] S. Floyd. (1994) TCP and explicit congestion notification. *ACM Comp. Commun. Rev.* [Online], pp. 10–23. Available: www.aciri.org/floyd/ecn.html

[7] R. J. Gibbens and F. P. Kelly. (1999) Resource pricing and the evolution of congestion control. *Automatica* [Online], pp. 1969–1985. Available: 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,. *Proc. Inst. Elect. Eng. 17th U.K. Teletraffic Symp.* [Online]. Available: www.statslab.cam.ac.uk/Reports/

[9] P. Hurley, J. Y. Le Boudec, and P. Thiran. (1999, Dec.) The asymmetric best-effort service. *Proc. IEEE GLOBECOM* [Online]. Available: http://icawww.epfl.ch/

[10] IETF Differentiated Services (Diffserv) Working Group [Online]. Available: http://www.ietf.org/html.charters/diffserv-charter.html

[11] R. Johari and D. K. H. Tan, "End-to-end congestion control for the Internet: Delays and stability," *IEEE/ACM Trans. Networking*, vol. 9, pp. 818–832, Dec. 2001.

[12] F. P. Kelly, "Charging and rate control for elastic traffic," *Euro. Trans. Telecommun.*, vol. 8, pp. 33–37, 1997.

[13] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control in communication networks: Shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, pp. 237–252, 1998.

[14] P. Key and D. McAuley, "Differential QoS and pricing in networks: Where flow control meets game theory," *Proc. Inst. Elect. Eng. Software*, vol. 146, pp. 39–43, 1999.

[15] L. Massoulié, "Stability of distributed congestion control with heterogeneous feedback delays," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 895–902, June 2002.

[16] A. M. Odlyzko. Paris metro pricing for the Internet. presented at *Proc. ACM Conf. Electronic Commerce*. [Online]. Available: www.research.att.com/~amo

[17] D. Wischik, "How to mark fairly," presented at the M.I.T. Workshop Internet Service Quality Economics, Cambridge, MA, 1999.

On the Use of Packet Classes in Communication Networks to Enhance Congestion Pricing Based on Marks

Juan Alvarez and Bruce Hajek

***Abstract*—This note explores the use of packet classes to enhance congestion pricing based on marks in the framework of Kelly and his coworkers. Two packet classes are used in order to exploit differences in the multidimensional quality-of-service (QoS) requirements within the population of users. Simple scenarios for a link with a finite buffer with and without burstiness are investigated. It is concluded that the use of packet classes provides a margin for error in provisioning and operating a packet network with congestion pricing.**

***Index Terms*—Congestion control, differentiated services, pricing in communication networks.**

I. INTRODUCTION

An important aspect of the Internet today is that it does not provide differentiation between packets. Therefore, the best-effort service that

Manuscript received October 31, 2000; revised June 7, 2001 and November 30, 2001. Recommended by the Guest Editors W. Gong and T. Başar. The work of J. Alvarez was supported in part by a Fulbright Graduate Fellowship to the first author, and in part by the National Science Foundation under Contract NSF ANR 99-80544, and DARPA under Contract F30602-00-2-0542. A version of this note was presented at the Conference on Information Sciences and Systems, Princeton University, Princeton, NJ, March 2000.

The authors are with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, the University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: alvarez@uiuc.edu; b-hajek@uiuc.edu).

Publisher Item Identifier S 0018-9286(02)05507-1.

currently exists is provided to all packets regardless of their diverse quality-of-service (QoS) requirements.

Kelly and his coworkers [4], [6] developed a framework for congestion pricing in which routers mark packets to indicate congestion. The packet transmission times of an individual user can be tailored to the time-varying, congestion dependent marking probabilities, and to the time-varying requirements of the user's applications. Therefore the framework offers users a rich menu of QoS options [4], [9], [10], [12].

Most of the work in this area focuses on one QoS measure: throughput. An exception is that for file transfers, the transfer time can be expressed in terms of throughput, encompassing delay as a QoS measure [4], [9], [12]. The work has also focused on using only a single class of packets. Yet, there are attractive reasons (see Clark [3]) to introduce a small number of service classes, in case some applications are sensitive to delay on a finer time scale than file transfer, such as interactive video applications. Bajaj *et al.* [2] found that the value of multiple service classes increases with the rigidity of delay sensitive applications and with the burstiness of background traffic, and the value depends on the mix of application types. We investigate the use of two packet classes within the congestion pricing framework.

Research on using a small number of service classes and no per flow information at the routers, called the differentiated services approach, is a topic of intense research. A competing approach to differentiated services, exemplified by ATM, RSVP, and Frame Relay standards, is to use reservations and enforcement based on per-flow state information within network routers. It is important to notice that as the number of packet classes allowed increases, the closer a differentiated services network comes to a per-flow controlled system.

In summary, under pricing based congestion control the users communicate to the network routers by the rate at which they send packets. The routers communicate to the users by the delay, loss, and marks they impose on packets. If packets include class labels, the users can also use packet labels to communicate to the routers. The challenge is to use this two-way communication between users and network routers to allocate the network resources efficiently and provide multidimensional QoS.

II. Model of an Elastic User

This section presents the user model considered throughout this note. It is based on the "elastic" user model described by Gibbens and Kelly [4].

Consider a discrete-time system with time slot duration equal to one unit. Assume that all packets are of equal size, and that each packet takes one slot to be transmitted. Suppose that the charge imposed by the network consists of marks placed on packets and that there is just one packet class. Suppose the ith elastic user is willing to pay for w_i marks per unit time. It uses a state variable $x_i(n)$ to determine the number of packets it transmits in slot n. Specifically, the user transmits $X_i(n)$ packets, where $X_i(n)$ is an integer near $x_i(n)$ calculated as follows:

$$X_i(n) = \lfloor x_i(n) + z_i(n) \rfloor^+ \quad (1)$$

$$z_i(n+1) = x_i(n) + z_i(n) - X_i(n) \quad (2)$$

$$x_i(n+1) = x_i(n) + \kappa(w_i - f_i(n)) \quad (3)$$

where $f_i(n)$ is the number of marks received by user i in slot n, and κ is a constant, called the gain parameter.

The rounding method (1) makes the short term average of the integer variable X_i close to that of the real variable x_i. An interpretation of (3) can be obtained by adding up the updates from time 1 up to time N

$$\frac{x_i(N) - x_i(1)}{\kappa} = Nw_i - (f_i(1) + \cdots + f_i(N)). \quad (4)$$

The right-hand side of (4) is the amount the user is willing to spend over the first N slots minus the amount actually spent over the first N slots. If both sides are divided by N and $N \to \infty$, then the left-hand side of (4) goes to zero (if $x_i(N)$ is bounded), implying that the average number of marks received by user i is equal to the desired number of marks. So in the long run, user i pays for marks at its target rate w_i. Equations (3) and (4) indicate that a larger value of κ produces faster convergence but also higher variance around the stable point. Throughout this note propagation delay is neglected. It is assumed that a mark on a packet is received by the user in the slot just after the mark is generated.

Kelly, Maulloo, and Tan [6] and Gibbens and Kelly [4] explain how a user might adjust its expenditure rate w, based on a user utility function. However, this note assumes that w_i is a given constant for each user i, in order to focus on the quality of service received for fixed expenditure rates.

To aid the reader with the labeling, we briefly list here all the scenarios examined. Scenarios Q and DQ (for "queueing" and "differentiated queueing") are considered in Section III. Scenarios QU, DQU1, and DQU2 are considered in Section IV, and involve an unresponsive user. In the case of multiple packet classes, the unresponsive user can declare packets to be class 1 (Scenario DQU1) or class 2 (Scenario DQU2).

III. Buffered Link

This section begins with a description of a link with a queue, for which delay, throughput and loss are considered important QoS measures. Assume that there are N users of the type described in Section II trying to transmit their packets through a single link with a single server and a finite queue. The server is able to transmit one packet per slot, and the queue has capacity to hold B more packets. If the number of arrivals in a particular slot plus the number of packets carried over from the previous slot is $B+1+j$, then j of the arriving packets are dropped. The service provided is first-in first-out.

The following packet marking mechanism is used to notify users when they cause congestion. A busy period of the queue is an interval from the time of arrival of a packet arriving when the queue is empty until the time of departure of the first packet that leaves the queue empty again. Marks are placed on all packets that depart between the occurrence of the first loss in a busy period and the end of the same busy period. This marking mechanism was proposed by Gibbens and Kelly [4]. Other marking mechanisms are presented in [1], [7], [8], [10], and [13].

Consider the following nominal case, called Scenario Q. The link has a queue with capacity 10 packets $(B = 10)$. There are 40 users with expenditure rates given by

$$w_{2i} = w_{2i+1} = (i+1) * 0.001 \quad \text{for } i = 0, \ldots, 19$$

for a total expenditure rate of 0.42 marks/slot. Simulation of Scenario Q shows that the aggregate throughput is 0.92 packets/slot, the average loss is 3.9%, and the average delay is 5.99 slots/packet.

A. Delay/Loss Sensitive Users and a Buffered Link

Now suppose that half of the users (the even numbered users) in Scenario Q are delay or loss sensitive. For example, they may require a smaller average delay than the 5.99 slots experienced in Scenario Q, or they may require a loss smaller than the 3.9% loss in Scenario Q, such as a 1% loss. These users are called delay/loss sensitive users. The odd numbered users are called throughput sensitive users. Purely loss sensitive or purely delay sensitive users are not considered here, since delay and loss turn out to be highly correlated for the control mechanisms examined. A question that comes to mind is:

Is the network able to provide lower delay and 1% loss to the delay/loss sensitive users if only one packet class is used?

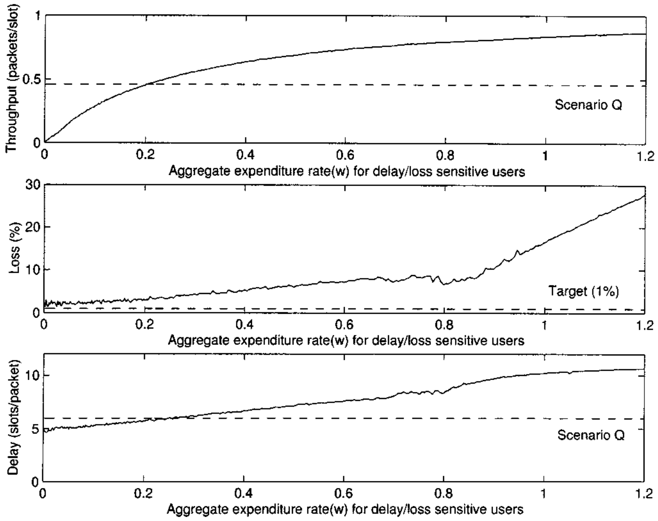


Fig. 1. Modification of Scenario Q: varying expenditure rates for delay/loss sensitive users. The QoS seen by the delay/loss sensitive users is shown.

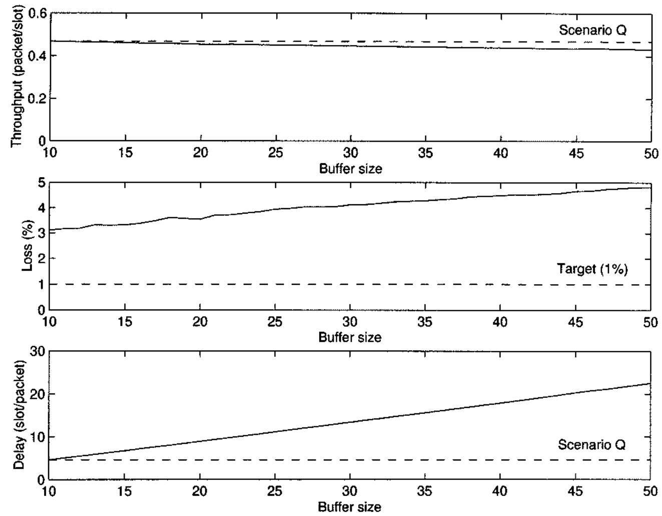


Fig. 2. Modification of Scenario Q: varying buffer size. The QoS seen by the delay/loss sensitive users is shown.

Under the conditions specified so far, the only way the delay/loss sensitive users can try to obtain a different performance from the resource is by changing their expenditure rate. Fig. 1 shows the performance of Scenario Q for various values of the aggregate expenditure rate of the delay/loss sensitive users.

The third plot in Fig. 1 shows that as the aggregate expenditure rate of the delay/loss sensitive users ranges from 0.0021 to 1.2 marks/slot, their average delay never drops below 4.5. Furthermore, to get the delay near the minimum level, the delay/loss sensitive users must substantially decrease their throughput. The problem, of course, is that as the delay/loss sensitive users back off, the other users step up their transmission rates to maintain a near level queue size. If the delay/loss sensitive users decide to use very large expenditure rates, they gain throughput but their increased pressure on the link causes both increased delay and loss. Similarly, the delay/loss sensitive users cannot achieve their target 1% loss rate. Therefore, if only one packet class is used in the system, it is not possible to satisfy the delay/loss sensitive QoS profile.

Of course, a possible solution to this dilemma would be for the network operator to provide more resources. Fig. 2 shows that increasing the buffer size is not helpful, and in fact is counterproductive in reducing loss and delay. However, Fig. 3 shows that increasing capacity does indeed help in reducing loss and delay.

In some situations, increased capacity may be available only on a long time scale, on the order of hours, days, weeks, or even months. Hence, we investigate alternatives that do not involve adding link capacity.

As a solution we include a class label in the packets in order to differentiate between two packet classes. This yields Scenario DQ, which is similar to Scenario Q, except the delay/loss sensitive users label all their packets as class 1 packets and the throughput sensitive users label all their packets as class 2. Class differentiation is achieved by charging higher prices for marks on class 1 packets and offering different service levels. Different prices can be easily incorporated into the user model by using the following modification of (3):

$$x_i(n+1) = x_i(n) + \kappa(w_i - p_i f_i(n)) \tag{5}$$

where p_i is the price per mark for the packet class of user i. Service differentiation is implemented using two different mechanisms: 1) buffer reservation and 2) packet scheduling.

In buffer reservation, the link reserves b_1 positions of the queue for class 1 packets and $b_2 = B - b_1$ positions for class 2 packets. Any

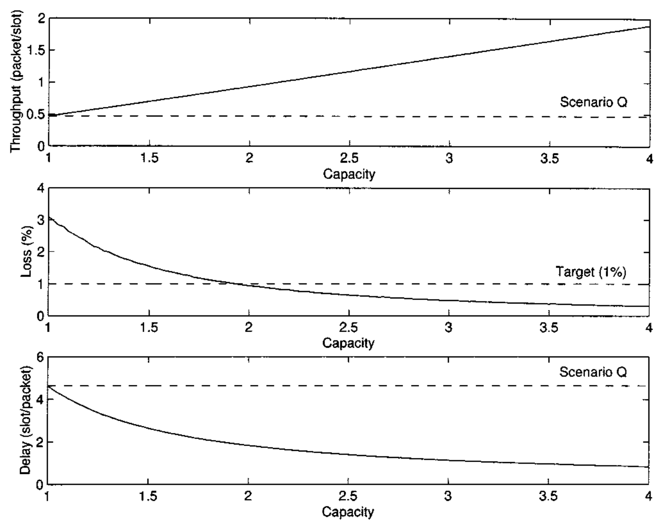


Fig. 3. Modification of Scenario Q: varying link capacity. The QoS seen by the delay/loss sensitive users in shown.

queue space unused by one class can be used by the other class. For example, if there are $n_1 < b_1$ class 1 packets in queue, then as many as $B - n_1 > b_2$ class 2 packets can be queued. However, if a class 1 packet arrives and there is no buffer space left, it will bump a class 2 packet out of the queue if $n_2 > b_2$. This mechanism is called buffer reservation $(b_1 : b_2)$ throughout the note.

Two different scheduling mechanisms will be tested for performance in Scenario DQ.

1) *Strict priority*: Service a class 2 packet only if there are no class 1 packets in the queue.
2) *WFQ*: Use weighted fair queueing to service the two packet classes. Assume that the weight for class 1 packets is W and for class 2 packets is 1.

Performance results for Scenario DQ are shown in Fig. 4 as p_1 is varied while keeping $p_2 = 1$.

The last three plots of Fig. 4 determine the fair region, which is the interval of p_1 for which the QoS of the throughput sensitive users is no worse than in Scenario Q, as follows. First, the aggregate throughput for the throughput sensitive users must be at least as large as before (in Scenario Q). In Scenario DQ with the WFQ weight $W = 1$ for

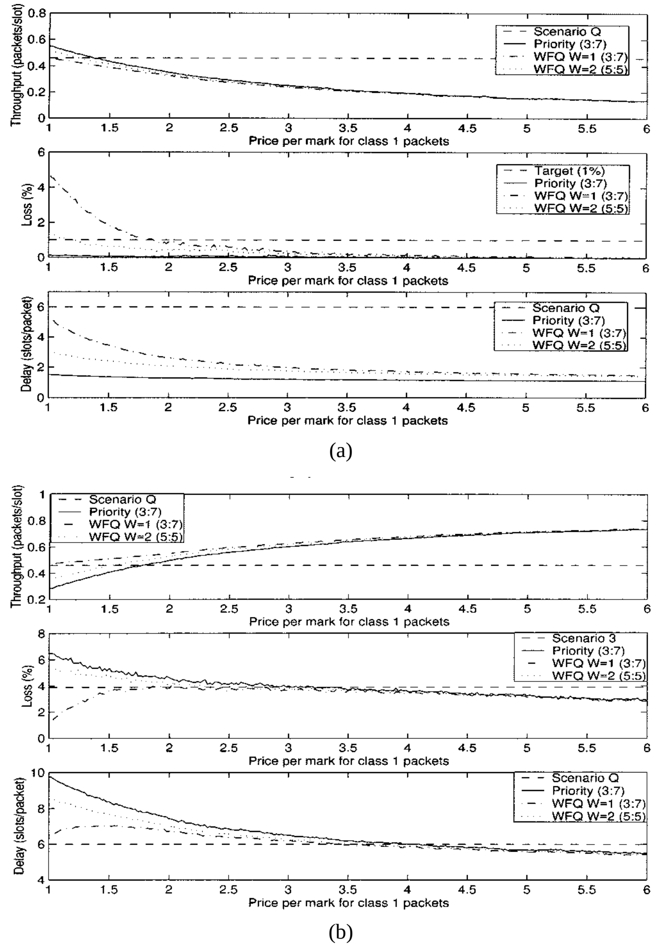


Fig. 4. Scenario DQ. (a) Delay/loss sensitive users. (b) Throughput sensitive users.

class 1 packets and capacity reservation $(3:7)$, this is achieved by any $p_1 > 1$. For WFQ $W = 2(5:5)$, this is achieved by $p_1 > 1.5$, and for strict priority $(3:7)$, for $p_1 > 1.8$. Since the three policies for buffer reservation and scheduling just mentioned were given in order of increasing service differentiation, it stands to reason that increasingly higher prices for the class 1 packet marks are needed to allow the throughput sensitive users to have the same throughput as if there was no class differentiation. However, the three policies offer similar performance in case of high prices per mark, because the total load imposed by the delay/loss sensitive users is greatly reduced.

Second, the loss experienced by the throughput sensitive users must be at most what they experienced in Scenario Q. By using WFQ $W = 1$ $(3:7)$ this is accomplished by $p_1 > 1$, while for WFQ $W = 2$ $(5:5)$ and strict priority $(3:7)p_1 > 3$ is needed, as shown in the second from last plot of Fig. 4. Again, higher prices are needed for the schemes with more aggressive service differentiation.

Third, the delay experienced by the throughput sensitive users must be at most what they experienced in Scenario Q. In this case, all three sets of control mechanisms require $p_1 > 4$, as shown in the last plot of Fig. 4. Here again, more aggressive service differentiation leads to higher delay for the throughput sensitive users.

Thus, the fair region is the interval $[4, \infty)$, and determination of the fair region is dominated by consideration of delay. The portions of the first three plots of Fig. 4 that lie in the fair region show which QoS vectors, i.e., (aggregate throughput, loss, delay), are available to the delay/loss sensitive users, without them negatively impacting the other users.

In summary, the simulations show that under any of the control mechanisms, a price per mark for class 1 packets in the region $[4, \infty)$ provides at least the same quality of throughput, delay and loss to the throughput sensitive users as in the original scenario, Scenario Q. Within this fair region of pricing, the delay/loss sensitive users can experience loss rates well under 0.1%, and delay under 2 slots/packet, at the expense of a 50% reduction in their throughput as compared to Scenario Q.

This section has shown that the combination of control mechanisms and multiple packet classes can achieve the goal of providing multidimensional QoS (lower delay or loss for some users at the expense of throughput for those users), while a single-class system cannot.

IV. Buffered Link With Unresponsive User

The scenarios considered in previous sections include only users with constant expenditure rates, so that the resulting traffic pattern is not very bursty. In real networks, bursty traffic patterns can be expected for numerous reasons: short file transfers common to web browsing, variable rate real-time data streams, and time-varying capacity of wireless links. For simplicity, in this section, burstiness is injected by introducing an unresponsive on–off user.

Consider next Scenario QU, which is similar to Scenario Q, except for the inclusion of an unresponsive user. The unresponsive user alternates between being active for T_{active} slots and inactive for T_{inactive} slots. When active, the unresponsive user sends one packet per slot, and when inactive it sends no packets. This behavior is independent of the number of marks the user receives. There is only one class of packets in Scenario QU. Thus, the delay and loss seen by the delay/loss sensitive users (the even numbered users) is the same as seen by the throughput sensitive users, and the aggregate throughput seen by the delay/loss sensitive users is half of the throughput (not counting the throughput of the unresponsive user). As the system becomes more bursty, the gain parameter κ in the user (3) plays a greater role, so we look at the system behavior as κ ranges from 0.001 to 0.1. The QoS experienced by the delay/loss sensitive users is presented in Fig. 5 for $(T_{\text{active}}, T_{\text{inactive}}) = (40, 160)$.

Losses are high in Scenario QU mostly because the queue fills up during the active periods of the unresponsive user. Three potential methods for providing lower loss and delay are considered next: increasing the buffer size, increasing the link capacity, and using multiple packet classes.

In contrast to Scenario Q, we expect losses to decrease in Scenario QU as the buffer size is increased (up to some point) since a larger buffer can hold more packets while the unresponsive user is active. However, just as in Scenario Q, we expect delay to increase due to more packets being queued. This is in fact the resulting behavior, as shown in Fig. 6. Throughput is not improved by increasing the buffer size since once marking begins the server keeps marking packets until the buffer empties out.

Fig. 7 shows that increasing capacity does indeed help in providing lower loss and delay. However, in order to provide very low losses (say less than 1%) increasing capacity by a factor of four is needed to counteract the burstiness of the unresponsive user.

A. Unresponsive User Sending Class 1 Packets

We now look at the effect of using two packet classes in an attempt to improve the QoS for the delay/loss sensitive users in Scenario QU. This subsection explores Scenario DQU1, for which the unresponsive user sends class 1 packets.

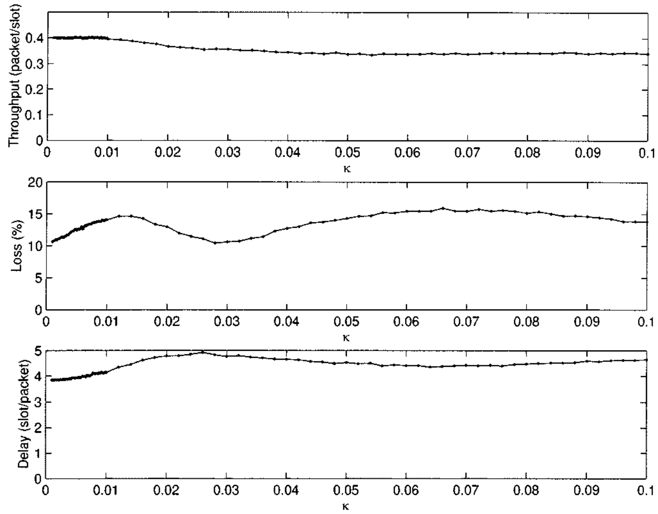


Fig. 5. Scenario QU: varying κ. $T_{\text{active}} = 40, T_{\text{inactive}} = 160$. The QoS seen by the delay/loss sensitive users is shown.

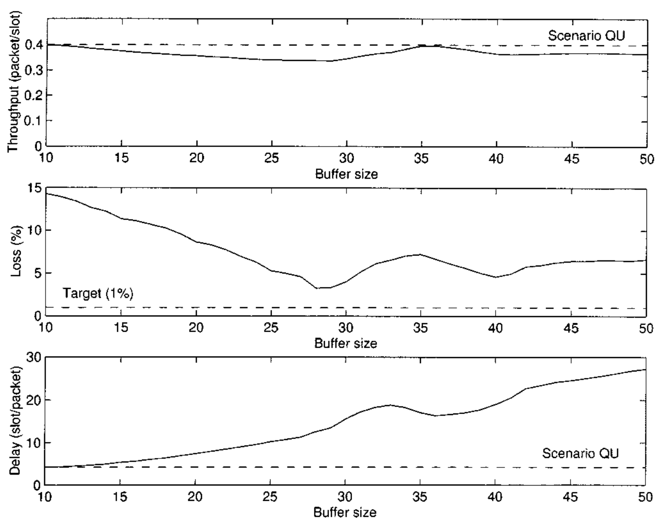


Fig. 6. Modification of Scenario QU: varying buffer size. $T_{\text{active}} = 40, T_{\text{inactive}} = 160$. The QoS seen by the delay/loss sensitive users is shown.

Figure 8 shows the behavior in this Scenario DQU1 with four values of p_1 using strict priority service with buffer reservation $(3 : 7)$, and compares it with the case where there is no packet class differentiation (Scenario QU).

The upper three plots in Fig. 8 show that multiple packet classes provide reductions in both loss and delay for the delay/loss sensitive users at a cost of reduced throughput, although losses are not reduced as much as this type of user might desire. The problem is that the unresponsive user also sends class 1 packets, and therefore, competes for the buffer space and capacity allocated to class 1 packets.

The bottom three plots of Fig. 8 show that a "fair region" cannot be achieved with these prices in this scenario since loss and delay for the throughput sensitive users are higher using packet classes than without using them. This is because the unresponsive user's packets have higher priority (class 1), and therefore, force the class 2 packets to stay queued for a long time when the unresponsive user is active, causing large delays and losses, no matter how large the price of class 1 packets.

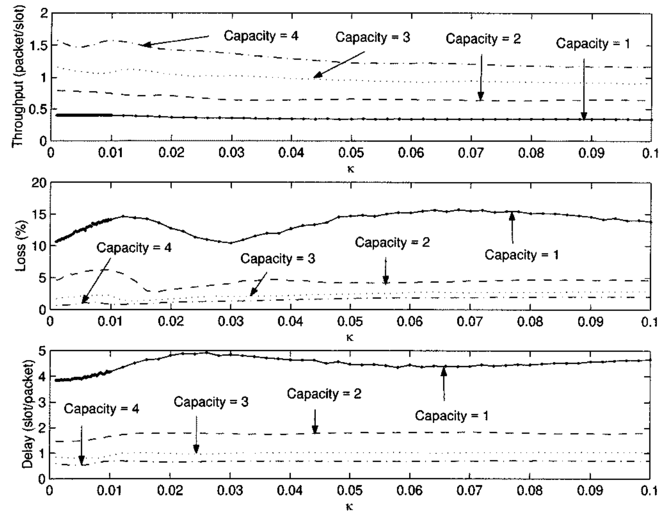
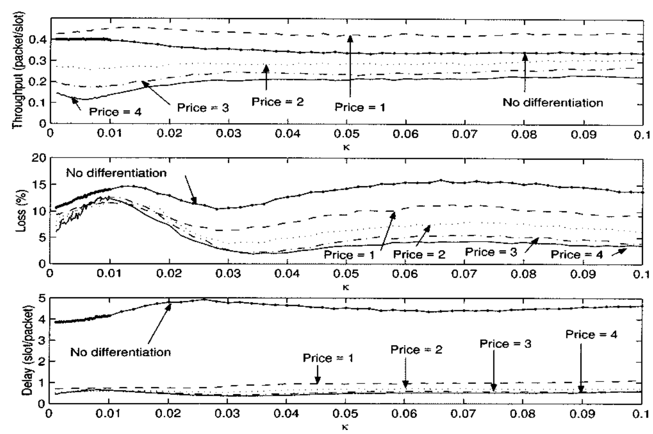
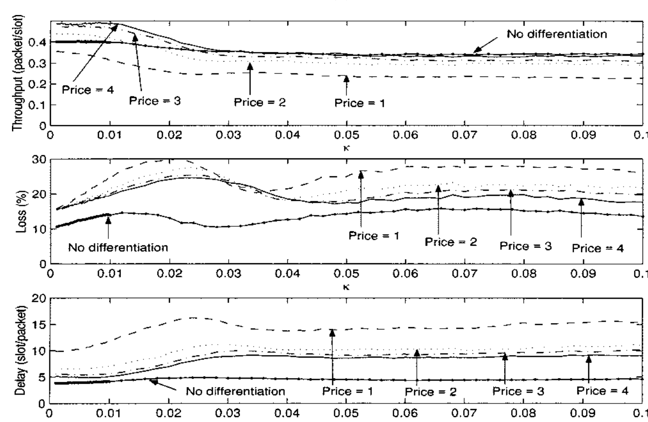


Fig. 7. Modification of Scenario QU: varying capacity. $T_{\text{active}} = 40, T_{\text{inactive}} = 160$. The QoS seen by the delay/loss sensitive users is shown.



(a)



(b)

Fig. 8. Scenario DQU1: varying κ and different prices for class 1 packets. $T_{\text{active}} = 40, T_{\text{inactive}} = 160$. (a) Delay/loss sensitive users. (b) Throughput sensitive users.

B. Unresponsive User Sending Class 2 Packets

Finally, consider Scenario DQU2, in which the unresponsive user sends class 2 packets. Fig. 9 shows the system behavior in Scenario DQU2 with four values of p_1 using strict priority service and buffer reservation $(3 : 7)$.

The top three plots in Fig. 9 show that multiple packet classes provides substantial reductions in both loss and delay for the delay/loss sensitive users at a cost of reduced throughput. These improvements are due to the fact that the multiple packet classes protect the delay/loss sensitive users from the unresponsive user, since the unresponsive user does not use their reserved buffer space and does not interfere with their successful transmission by the server. Moreover, the bottom three plots of Fig. 9 show that a 'fair region' of p_1 exists for small values of κ.

This section has shown that the combination of control mechanisms and multiple packet classes can significantly improve the multidimensional QoS (lower delay or loss for some users at the expense of throughput for those users), over the case of a single-class system. A caveat is that multiple packet classes can magnify the unfairness caused by an unresponsive user in the case such user sends class 1 packets, as in Scenario DQU1. The fact that the performance in Scenario DQU1 is better than in Scenrio DQU1 is consistent with an observation of Key *et al.* [11], that multiple classes are particularly attractive if delay sensitive traffic is less bursty than the other traffic.

V. Delay and Loss for Large Capacity Links

This section investigates the effect of increasing system size on delay and loss in a model with a buffer, similar to Scenario Q.

When queueing is present we can trade throughput against both delay and loss. We expect that the smaller the throughput, the smaller the delay and loss. This tradeoff is illustrated in Fig. 10, which shows the performance for an $M/M/1/B$ queue in continuous time with service rate μ, variable load and $B = 10\ \mu$.

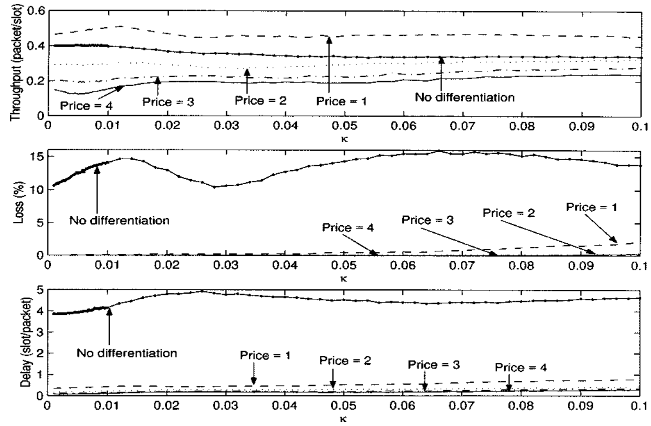
Taking $\mu = 1$ gives a reasonable approximation of Scenario Q. If a suitable marking mechanism is used, the total offered traffic, and hence the throughput, can be varied. Recall that for Scenario Q, the normalized throughput is 0.92, the average delay is 5.99 slots, and the loss probability is 0.039. These operating points for Scenario Q are indicated by the two stars in Fig. 10. According to the first plot, if we wished to cut the mean delay in half, down to 3, then the throughput would have to be reduced to about 70% of capacity.

The curves for $\mu = 10$ in Fig. 10 correspond to a ten fold increase in the buffer size and service rate, essentially having ten times the user population and resources as in Scenario Q. The curves for $\mu = 100$ correspond to a one hundred fold increase. Observe that as the system gets larger for a fixed normalized throughput, much smaller delays and loss can be achieved, with only a small sacrifice in throughput. For example, if $\mu = 100$, then the delay is less than 0.2 (corresponding to twenty queued packets) if the throughput is 95% of capacity or smaller, and the loss probability is less than 10^{-7} if the throughput is 98.8% of capacity or smaller. Thus, as the system gets larger, a smaller mean delay and a much smaller loss probability can be given to all users, while at the same time keeping the link nearly fully utilized. Clearly the delay and loss probabilities in Fig. 10 for large μ would not be as small if the user's were more bursty, but the benefits of statistical multiplexing for links serving a large number of users is still expected.

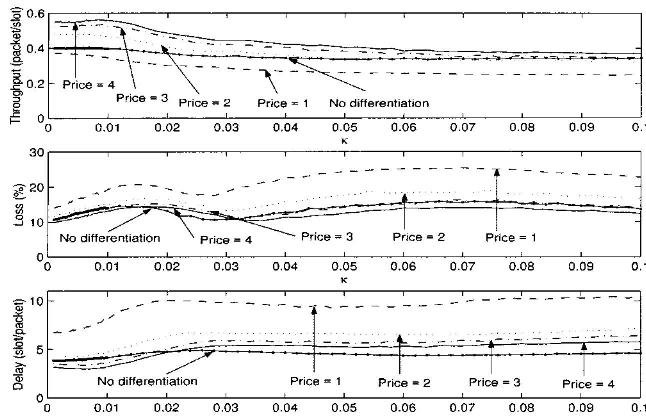
VI. Conclusion

This note considered a model for which loss, delay, and throughput were considered as separate components of QoS. It remains to be seen if quality of service is fundamentally multidimensional for large-scale networks and realistic traffic, or if instead, the dimensionality of the relevant QoS measures can be reduced to one in practice.

The original Scenario Q considered a link with fairly small capacity. Section V indicates that as the link buffer size and transmission capacity are scaled up, for the same type of elastic user model, the need for multiple packet classes disappears. That is because both the delay and loss probability decrease to levels small enough for all users, leaving throughput as the only remaining QoS parameter. Thus, the most likely place that multiple packet classes may be needed in future networks is



(a)



(b)

Fig. 9. Scenario DQU2: varying κ and different prices for class 1 packets. $T_{\text{active}} = 40, T_{\text{inactive}} = 160$. (a) Delay/loss sensitive users. (b) Throughput sensitive users.

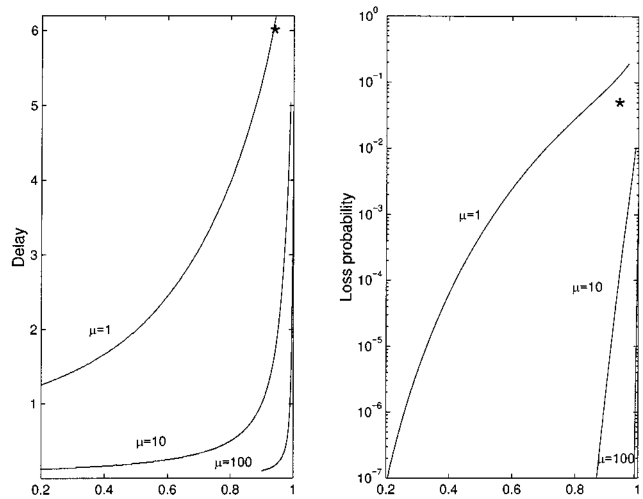


Fig. 10. Delay and loss versus throughput, for an M/M/1/B queue with service rate μ and $B = 10\ \mu$.

near the edges, where link bandwidth is limited. Perhaps wireless links, with their limited capacity, will be a major reason to implement multiple packet classes. Some wireless links, and some data streams are time varying with large time scales. Such time variations, coupled with the limitations imposed by large round trip times in congestion control loops, may forever insure that queuing delay, along with bandwidth, has to be explicitly addressed, pointing to a need for multiple packet classes.

Our goal for fairness is the following: provide at least the same level of satisfaction to throughput sensitive users when there are two packet classes as when there was only one class. An interesting question is how we can accommodate a multidimensional QoS profile within the notions of network-wide fairness. Another question is how to implement congestion pricing for multiple class networks. Progress in this direction was recently reported in [5] and [11].

In practice, the expectations and requirements of network users almost always involve delay and loss, whether or not explicitly stated in a service level specification or agreement. Perhaps increasing network resources in a timely fashion, relying on statistical multiplexing, and controlling admission to networks will someday make throughput the only relevant QoS measure. However, if some users are more tolerant to delay and loss than others, and if burstiness of aggregate traffic streams in some links cannot be avoided, the use of multiple packet classes can typically improve the margin of protection against unexpected or unavoidable stresses on the network. Similar conclusions are reached in [2].

References

[1] S. Athuraliya, D. Lapsley, and S. Low, "An enhanced random early marking algorithm for Internet flow control," in *Proc. INFOCOM'2000*, Israel, 2000.

[2] S. Bajaj, L. Breslau, and S. Shenker, "Is service priority useful in networks?," in *Proc. ACM Sigmetrics'98*, Madison, WI, June 1998, pp. 66–77.

[3] D. D. Clark, "Adding service discrimination to the Internet," *Telecommun. Policy*, vol. 20, pp. 33–37, 1996.

[4] R. J. Gibbens and F. P. Kelly, "Resource pricing and the evolution of congestion control," *Automatica*, vol. 35, 1999. 1969–1985.

[5] ——, "A note on packet marking at priority queues," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 1016–1020, June 2002.

[6] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control in communication networks: Shadow prices, proportional fairness and stability," *J.Oper.Res.Soc.*, vol. 49, pp. 237–252, 1997.

[7] F. P. Kelly, "Models for a self-managed Internet," *Phil.Trans.Royal Soc.A358*, pp. 2335–2348, 2000.

[8] ——, "Mathematical modeling of the Internet," in *Mathematics Unlimited—2001 and Beyond*. Berlin, Germany, 2001, pp. 685–702.

[9] P. B. Key and D. R. McAuley, "Differential QoS and pricing in networks: Where flow control meets game theory," in *Proc. Inst. Elect. Eng. Software*, vol. 146, 1999, pp. 39–43.

[10] P. Key, D. McAuley, P. Barham, and K. Laevens, "Congestion pricing for congestion avoidance," Microsoft Research, Cambridge, U.K., Tech. Rep. MSR-TR-99-15, 1999.

[11] P. Key, L. Massoulié, and J. Shapiro, "Service differentation for delay-sensitive applications: An optimization-based approach,", Cambridge, U.K., Tech. Rep. MSR-TR-2001-115, Nov. 2001.

[12] L. Massoulié and J. Roberts, "Bandwidth sharing: Objectives and algorithms," *Proc. IEEE INFOCOM'99*, pp. 1395–1403, Mar. 1999.

[13] D. Wischik, "How to mark fairly," presented at the M.I.T. Workshop Internet Service Quality Economics, Cambridge, MA, 1999.

On Localized Control in QoS Routing

Srihari Nelakuditi, Srivatsan Varadarajan, and Zhi-Li Zhang

***Abstract*—In this note, we study several issues in the design of *localized* quality-of-service (QoS) routing schemes that make routing decisions based on *locally collected* QoS state information (i.e., there is no network-wide information exchange among routers). In particular, we investigate the granularity of local QoS state information and its impact on the design of localized QoS routing schemes from a theoretical perspective. We develop two theoretical models for studying localized proportional routing: one using the link-level information and the other using path-level information. We compare the performance of these localized proportional routing models with that of a global optimal proportional model that has knowledge of the global network QoS state. We demonstrate that using only coarser-grain path-level information it is possible to obtain near-optimal proportions. We then discuss the issues involved in implementation of localized proportional routing and present some practical schemes that are simple and easy to implement.**

***Index Terms*—Localized proportional routing, quality-of-service (QoS) routing.**

I. Introduction

In quality-of-service (QoS)-based routing [2], [6], [23], paths for flows are selected based upon knowledge of the resource availability (referred to as *QoS state*) at network nodes (i.e., routers) and the QoS requirements of the flows. This knowledge, for example, can be obtained through (periodic) information exchange among routers in a network. Under this approach, which we refer to as the *global* QoS routing approach, each router constructs a global view of the network QoS state by piecing together the QoS state information obtained from other routers, and performs path selection based on this global view of the network state. Examples of the global QoS routing approach are various QoS routing schemes [4], [23] based on QoS extensions to the OSPF routing protocol as well as the ATM PNNI routing protocol. Global QoS routing schemes work well when each source node has a reasonably *accurate* view of the network QoS state. However, as the network resource availability changes with each flow arrival and departure, maintaining an accurate network QoS state is impractical, due to the prohibitive communication and processing overheads entailed by frequent QoS state information exchange. In the presence of *inaccurate* information regarding the network QoS state, global QoS routing schemes may suffer degraded performance as well as potential instability [22], [14].

As a viable alternative to the global QoS routing approach, in [15] we proposed a novel *localized* approach to QoS routing. Under this localized QoS routing approach, instead of (periodically) exchanging information with other routers to obtain a global view of the network QoS state, a *source* router attempts to *infer* the network QoS state from *locally collected flow statistics* such as flow arrival/departure rates and flow blocking probabilities, and performs path selection based on this local information. As a result, the localized QoS routing approach avoids the drawbacks of the conventional global QoS routing approach such as degraded performance in the presence of inaccurate routing information. Furthermore, it has several important advantages: *minimal* communication overhead, *no* processing overhead at *core* routers, and *easy* deployability.

Manuscript received October 19, 2000; revised June 30, 2001 and November 25, 2001. Recommended by the Guest Editors M.-B. Gong and T. Başar.

The authors are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455 USA.

Publisher Item Identifier S 0018-9286(02)05508-3.