Charging For QoS

Domenico Ferrari and Luca Delgrossi

Center for Research on the Applications of Telematics
to Organizations and Society (CRATOS)
Università Cattolica, I-29100 Piacenza, Italy
{dferrari, ldgrossi}@pc.unicatt.it

Abstract

In an integrated-services packet-switching network (for example, the Internet of the future, which is expected to offer real-time as well as non-real-time services), the charging policy must be service-dependent. But how should charges vary with the type of service and with the quality of service (i.e., with the QoS parameters and with their respective values) ? We start from a list of the properties the policy should have, and derive from it a formula that satisfies most of them. We also briefly discuss the evaluation of the coefficients in the formula and the experiments that could be run for its validation.

1 Introduction

Selecting a pricing policy for a single-service network may not be very easy, but is much easier than for a multi-service one. In a network of the latter kind, the main problem is to decide the relative values of the various types of service (ToS), and, within each type, those of the various qualities of service (QoS). For example, in a network that offers both deterministic and probabilistic performance guarantees, how much more expensive should a connection with a deterministic delay bound of 100 ms be than an otherwise identical connection with a delay bound of 100 ms that is guaranteed to be satisfied for each packet with a probability of 0.99? Also, what should be the price difference between the former connection and an otherwise identical one with a deterministic delay bound of 200 ms?

Thus, the problem is complicated because of the presence of multiple services being offered by the same network, but is made even harder by the multiple QoS parameters (what are the relative impacts on the price of a communication of the delay bound, the jitter bound, the packet loss probability bound, and so on?) and the many different values each one of them can take. Pricing all types and qualities of service the same is not an acceptable solution to the problem: who would be able to convince users they do not need the best service with the best quality at all times? Without the deterrent of high prices, the network could be accessed by many fewer users and would be very inefficiently utilized. Thus, prices must be service-dependent, but how can they be set? Even if they were always set by the market, it would be important for the network managers and owners to know the relative costs of providing the various services. Also, there will be cases (for example, when the right of high-level-service providers to use a networking infrastructure owned by other subjects is regulated by a government agency) in which prices will have to be justified by the underlying costs or at least by rational, quantitative arguments instead of by references to obscure, uncontrollable "market forces".

Devising such arguments, if reasonably satisfactory ones can be found, is the main objective of this paper.

2 Principles and Properties

The context to which this paper refers is the one of integrated-services packet-switching internetworks (frequently to be called "networks" for simplicity in the sequel). The network is capable of carrying various types of traffic, all of which fall into one of the following two categories:

(a) non-real-time (*nrt*) traffic,
(b) real-time (*rt*) traffic.

Service to traffic in category (a) is the one called "best-effort transport": the internetwork layer does not provide any specific performance or reliability guarantees, though the client may use protocols at the transport or higher layer that will make the service reliable (i.e., the packet loss will be reduced to 0); unfortunately, one cannot do this on the performance side, for which guarantees cannot be offered to the user unless they are supported at all layers in the network's architecture.

Traffic in category (b) will be able to choose a real-time ToS, and to request and obtain performance as well as reliability guarantees within that ToS. In future networks, there might be an intermediate category between (a) and (b): the category of traffic that receives a service without guarantees but usually better than best effort (for instance, a service using measurement-based admission control). To keep our discussion simple, we shall ignore this possibility in the rest of this paper. For the same reason, we assume that there is a single ToS in category (b), namely, deterministic service, though with several QoS parameters.

The kind of network considered here will generally be able to offer multicast (one-to-many, many-to-many) service. However, again for the sake of simplicity, we assume here that only unicast communication is allowed.

As already recalled in Section 1, charges must be service-dependent (i.e., dependent on ToS and QoS). Making them also usage-sensitive is generally desirable but not compulsory: there might be good reasons, for instance economical ones, to prefer flat-rate charges; however, these charges must be different for different types and for different qualities of service. In this paper, we propose a usage-sensitive charging policy, but the usage-dependent terms in the formula could easily be replaced by a usage-independent one.

What are the desirable properties of a charging policy? From the viewpoint of the network manager, the most important ones are:

A. High probability of cost recovery,

B. Competitiveness of prices,

C. Encouragement (or discouragement) of client behaviors that will enhance (or degrade) the network's efficiency,

D. Low implementation costs,

E. Low usage costs.

From the viewpoint of the client, the main properties are:

F. Comprehensibility,

G. Controllability,

H. Predictability,

I. Stability,

J. Fairness.

Property A refers to the possibility that, if charges are too low, the network might not produce enough revenue to offset its costs and provide a fair profit margin even when it is reasonably loaded and well-run. Yet, charges should be competitive with those of networks offering similar services (Property B). The charging policy, states Property C, ought to reward efficient use and penalize inefficient use of the network. The costs of implementing the policy (e.g., the acquisition of tools for accounting data collection and for billing the users) and those of using it (e.g., the overhead of these tools) should be low (properties D and E, respectively).

Network clients want the charging formula to be easy to understand (Property F). According to Property G, they also would like to be able to control the total charges for a communication by modifying (to the extent that this is possible) the QoS requirements.

Predictability (Property H) has two aspects, both very important: on the one hand, clients want charges to be reproducible, so that they can predict the price of a communication identical or very similar to one they did in the past; on the other hand, for a "new" communication, it is very useful to obtain an estimate of the charges before irreversibly ordering the network to start it. Property I (Stability) postulates that most users prefer services whose prices do not change too often in time; this could be regarded as another dimension of predictability. It is also important that clients perceive the charges to be fair and reasonable (Property J): everybody should pay the same amounts for the same services; users who get more (resources, performance, reliability) from the network should pay more; communications over longer distances should be more expensive than similar communications over shorter distances; and so on.

In the next two sections, we shall present and discuss an approach to charging that is suitable for the context and satisfies most of the properties described in this section.

3 An Approach to Charging for QoS

An integrated-services internetwork such as the one described in Section 2 must offer connection-oriented services at the internetwork layer for the real-time components of its traffic, while connectionless services are desirable for the non-real-time ones at the same layer [1]. During connection establishment, resources are to be reserved along the path of the connection; these resources will be usable by non-real-time traffic whenever they are not occupied by the real-time connection to which they have been assigned, but will not be assignable to other real-time connections until this connection is torn down.

Thus, a real-time connection "possesses" network resources that are in finite supply, and its presence prevents other requests for real-time connections from being accepted: in fact, the admission control algorithm running during the establishment phase may reject a request if so much of a resource anywhere along the path chosen for the requested connection has been reserved that the resource cannot accommodate the new request. It is therefore fair to charge, for the resources reserved for them, the real-time connections that are admitted; these charges (to be called *reservation charges*) will have to be higher for larger amounts of each resource, and proportional to the duration of the reservation, i.e., of the connection's lifetime.

As more stringent values for QoS parameters generally require more resources to be reserved, charges for different QoS values could be obtained through those for the amounts of resources reserved. Different QoS parameters need different resources (or mixes of resources) to be reserved: hence, the relative charges for the various QoS parameters (e.g., for delay bounds vs. packet loss bounds) will be influenced by the relative charges chosen for the different resources.

Of course, these charges would be more directly comprehensible (Property F) and controllable (Property G) by the client if the formula had in it user-level QoS parameters and their values instead of network resources and their amounts, which the client is usually ignorant about. However, we do not see how the relative charges for the various QoS parameters and for their respective values could be rationally and quantitatively justified otherwise; to our knowledge, the only alternative to the resource-based approach is the market-based one, which, however, does not lend itself to such justification.

The resources our approach refers to are those being reserved along the path of the real-time connection. The longer the path, the more resources will be involved, and the higher this component of the total charge. Thus, everything else being equal, these charges will grow with the distance between source and destination, which is quite reasonable; as will be seen more clearly below, our notion of distance is much closer to the number of hops than to the Euclidean distance or to the length of the path.

Obviously, the non-real-time component of the traffic will not pay any reservation charges. Since it has to be charged something anyway, and we have chosen to make our formula usage-sensitive, that component will pay *usage charges*. These should be proportional to the number of bytes transported (charging for those received would be fairer to the client than charging for those transmitted) to account for the amount of bandwidth consumed, and to the number of packets to account for the header-processing work done by the routers along the path. The latter term ought to be proportional to the number of hops; it is a little more debatable whether this should be the case also of the former term, but it is certainly true that the buffer space and the fraction of the total link bandwidth consumed by a transmission grow with the number of hops.

Properties D and E may be viewed as calling for a simplification of the usage charges term. One way to do this is to observe that the number of packets is a monotonically non-decreasing function of the number of bytes, and that each hop may be assumed to contribute an approximately equal amount of bandwidth, processing time, and buffer space; thus, the term could be reduced to one proportional to the number of bytes times the number of hops, the coefficient of proportionality taking somehow into account the unit prices of all the per-hop and per-byte resources involved. Unfortunately, in the connectionless service case, the number of hops is not fixed, as packets belonging to the same communication may follow different routes to the destination. This difficulty may be resolved in an approximate way by using as the number of hops between source and destination its "normal" value, which is well-known.

Such problem does not exist for real-time connections, which will have to pay usage charges as well: the exact number of hops along the chosen path is known at the latest at the completion of the establishment phase. One can persuasively argue that, since real-time packets always have priority over non-real-time packets, their usage charges coefficient ought to be higher; how much higher is not obvious, and to be determined. Even among real-time packets there might be differences in priorities; for example, if packet scheduling in the routers is deadline-based, those with shorter deadlines (i.e., smaller end-to-end delay bounds) will have priority over those with longer deadlines. Thus, if one follows the same line of reasoning, the usage charges coefficient ought to be a function of the end-to-end delay bound (more precisely, of the per-hop delay bounds). In the interest of simplicity (Properties F and G), however, we may just let the different delay bounds influence only the reservation charges, and adopt the same usage charges coefficient for all real-time connections.

Reproducibility, one of the two aspects of Property H, requires that the charging formula do not include any traffic-sensitive terms: all terms must be dependent only on the characteristics of the communication the formula is being applied to, not on the rest of the network's load. So far in our discussion, no load-dependent terms have been introduced, with one exception, but the question is to be considered again after we specify in detail how each term is to be evaluated. The exception is an indirect one: the load (but sometimes also a hard failure, or the use of an unstable routing algorithm) may cause the path to change from an incarnation of a real-time connection to the next; this route oscillation will modify both the reservation charges and the usage charges, both of which are influenced by the number of hops. No such problem arises in the charging policy discussed so far for a non-real-time communication.

The other aspect of predictability, the advance knowledge of the charges, is easy to obtain with our approach: per-unit-time reservation charges can be computed and accumulated at each hop of the establishment message's return trip (we assume for ease of discussion that a real-time connection is set up by an establishment message issued by the source, which reaches the destination and returns to the source, where it provides a positive or negative answer to the sending client); for usage charges, the product of the coefficient times the number of hops (i.e., the per-byte charge) can be presented to the client, together with the per-unit-time reservation charges, at the time the client is notified that the requested connection has been set up. If the total number of bytes to be transmitted is known, and the duration of the connection can be estimated, the client can calculate the total charges before accepting the connection and starting transmission on it. Note that reservation charges are computed only once per connection, and that the cost of this computation (which is an additional establishment overhead) can be made negligible with respect to the cost of establishment, thereby satisfying Properties D and E. The cost of counting bytes, an operation that can be done at a single point of the path (e.g., at the receiver), is also very small; there will have to be mechanisms in each potential sender or receiver that can trigger, perform, and terminate the counting and ship the results for every communication, including the non-real-time ones (in this case, the best place to collect accounting data is probably the sender).

As far as the establishment operation is concerned, a decision should be made whether there ought to be a special charge for it. Following the telephone model, it would seem fair not to charge anything for unsuccessful setups. If, on the other hand, the establishment is successful, but the client for some reason (for instance, because the charge is too high) refuses the connection before having started using it, it would be reasonable to charge a small fee to account for the resources consumed during the setup and tear-down operations, as well as to discourage frivolous usage of real-time connection establishment services.

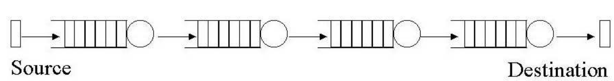
Stability (Property I) can be achieved by not changing the coefficients or the formula too often; there is nothing in our approach that would force or encourage network managers to do so. Proving that a charging policy is fair (Property J) is in general quite difficult. However, if charges are reproducible and stable, if they grow with the quality of the services, if the contributions to them of various factors (ToS, QoS parameters, parameter values) are reasonable and well-balanced, then the policy is likely to be perceived as fair by the clients.

The next section specifies the reservation charges in more detail by introducing a model of a real-time connection's path. In Section 5, we discuss methods to determine the coefficients of the complete charging formula.

4 The Charging Formula

The path of a real-time connection is assumed to be satisfactorily modeled by a cascade of queueing stations as in Figure 1. Each station contains a single server that represents either a header-processing resource or a packet-transmission one. In most cases, the correspondence between stations and routers is not one-one: in general, a router will be modeled as a series of two or more stations. Each station has its own queue, which is where packets (not only those traveling on the connection being considered, but all those needing the server) wait for its service. In general, the number of packets traveling on the connection for which there is room in the queue is limited by the number of buffers allocated to the connection in that particular station (or, if buffer allocation is dynamic, by the maximum number of buffers the connection can have). Each queue is scheduled by an algorithm that need not be the same for all queues; on the contrary, the algorithms may all be different, as long as they are capable of providing performance guarantees [2]. The lines connecting adjacent stations are characterized by either a zero or a nonzero propagation delay; the latter is the case of lines representing links connecting two routers, i.e., network links, while the former are models of intra-router connecting lines. The first and last stations are usually located inside the source host and the destination host, respectively.

Figure 1: Path Model.



The model in Figure 1 contains three types of resource in each station, two visible and one invisible: buffer space B, computing capacity C, and schedulability D (called the "delay resource" in [3], [4]). The first two are easy to handle: in every station, there is a total amount of each resource, whose price per unit time is conventionally set to P(B) and P(C), respectively, and our connection requests fractions b and c of the two total amounts for its lifetime L: the reservation charges for those two resources in a given station will be L*[bP(B) + cP(C)].

Schedulability is a much harder resource type to deal with. It is necessary to consider this invisible and intangible resource since, when the connection has stringent delay bound requirements, it may be impossible to let it pass through a station if that station would be unable (because of the too many demanding connections already traversing it) to guarantee a small enough local delay bound. The establishment will have to be attempted along a different, sub-optimal route, and in some cases the request may even have to be rejected. Thus, this resource type, exactly like the other two, is capable of constraining the use of the others; unlike them, however, it will not block all admissions, but only some, and the decision will depend on the QoS requirements of the connections to be established. Furthermore, the form of this dependency will vary with the scheduling policy of the station. If, in an approximate way, one can assign a price per unit time P(D) to the schedulability resource and devise a method to compute the fraction d of it a new connection wants, then the charge for its reservation for time L will be L*[dP(D)].

We have begun an investigation of some of the most popular packet-scheduling policies to determine reasonable methods to estimate P(D) and d so that they will be traffic-independent (Properties F, G, and H), easy to understand (Property F), and inexpensive to realize (Property D) and to use (Property E). This will be the main subject of a forthcoming publication.

We are now able to write the charging formulae we have built step-by-step in Section 3:

(1) *Charges (rt) = ReservationCharges + + UsageCharges (rt),*

(1') *ReservationCharges = L*S[bP(B) + cP(C) + + dP(D)],*

(1") *UsageCharges(rt) = K(rt)*h*V,*

(2) *Charges(nrt) = UsageCharges(nrt) = K(nrt)*h*V,*

where

L = lifetime of the connection,

S = sum over all the stations along the connection's path,

b, c, d = fraction of the three resources in each station requested by the connection,

P(B), P(C), P(D) = prices per unit time of buffer space, computing capacity, and schedulability in each station,

K(rt), K(nrt) = usage charges coefficient for real-time and non-real-time communications, respectively,

h = number of hops in the path (rt case) or in the "normal" path (nrt case),

V = total volume of bytes involved in the communication.

5 Determining the Coefficients

The charging formula presented at the end of the previous section contains two types of coefficients: the *usage coefficients* K(rt) and K(nrt), which may be the same for all communications, and the *reservation coefficients* P(B), P(C), and P(D) for each station (thus, for example, a router modeled by two cascaded stations on each path traversing it has six such coefficients). The latter are the prices per unit time assigned to the three resources in the station: they can be kept and used there to compute the sum bP(B) + cP(C) + dP(D) after obtaining, either directly from the destination or from a local calculation based on some information kept locally and some propagated by the returning establishment message, the values of the fractions b, c, and d. This sum will be added to the partial sum received via the return message, and the updated partial sum will be written into the same message to be updated further by the stations between the current one and the source.

If the network contains groups of identical stations, it is reasonable to assume that all the stations in a group will have the same P(B), the same P(C), and the same P(D). This will simplify the formula of the reservation charges. If, for instance, each station belongs to one of two groups, and if the fractions requested by a connection are the same for all stations in a group, the formula becomes

(3) *ReservationCharges = = L*{n'*[b'P'(B) + c'P'(C) + d'P'(D)] + + n"*[b"P"(B) + +c"P"(C) + d"P"(D)]},*

where n' and n" are the numbers of stations along the path belonging to the first and the second group, respectively. These simplifications, if they can be made, will greatly reduce the number of unknowns, but this number will still be larger than the number of available equations.

A possible approach to determining the values of the coefficients consists of starting with values that appear reasonable and verifying that they result in equally reasonable charges. In the case of the two groups we have just described, since there are only six unknowns in the reservation charges formula (the prices per unit time), we could carefully choose six connections (so that they cover different path lengths, different areas of the network, different proportions of the two groups within the connections, if possible) and write six equations in which the other six unknowns, i.e., the reservation charges, have been assigned sensible values. The solutions of these six equations could then be used as initial values of the P's and P"s, to be adjusted by a trial-and-error procedure involving additional, carefully chosen connections and reasonableness checks for the corresponding charges.

When we reach a satisfactory set of values for the coefficients, we can determine whether Property B holds by comparing the charges of given connections on our network to those for the same connections on other networks. Verifying that Property A is satisfied is much harder. To be able to do this, even in a very rough, approximate way, we need to know (or predict) how much load the internetwork will have over, say, the next year; this difficult task is made ever more difficult by the dependence of the load on the competitiveness of prices (Property B), which may vary in time due to market forces or competitors' decisions. If real-time connections can be grouped into a relatively small number of categories, and the numbers of connections from each group that will be created over, say, the next year can be estimated together with the distribution of their path lengths and the volumes of information to be transmitted over them, the reservation revenue for real-time traffic and its components can be evaluated.

The total revenue for non-real-time traffic and that for the transport of real-time traffic can also be estimated if the total volume of data to be transported by the network can be predicted and reasonable values for coefficients K(nrt) and K(rt) can be found; the latter problem could be approached by selecting these two values so that usage charges will be competitive. In these evaluations, the fractions b, c, and d are best calculated by a suitable network simulator (or at least a path simulator) that models the real-time connection setup mechanism.

6 Conclusions

We have stated a few principles that any charging policy for packet-switching internetworks offering multiple services should obey, and listed 10 properties such a policy should ideally possess. From the principles and properties, we then derived a charging formula and discussed ways to determine the values of its coefficients, so that the formula can actually be used. Our formula does not fully satisfy all the properties we listed in Section 2. Perhaps a perfect formula does not exist. But it makes sense to ask how far would our formula be from the perfect one if such a perfect formula existed.

Ways to obtain the goal of Property A have been alluded to in Section 5. Also in the same section, a method to verify the competitiveness of prices (Property B) has been briefly described; the key to this verification is computing the charges for a real-time connection, e.g., by using a suitable simulator. Satisfaction of Property C (encouraging/discouraging client behaviors that enhance/degrade the network's efficiency) could be at least partially tested by verifying that the differences in price among the services and among their qualities are large enough to be significant. If it is, then we can expect most clients to choose the service type and quality that are barely sufficient for their purposes, thereby minimizing the waste of resources.

Properties D and E have to do with implementation costs and usage costs, respectively. These two properties have been invoked several times in the discussion of our approach: we believe that our charging policy is quite inexpensive on both counts.

Comprehensibility (Property F) and controllability (Property G) leave something to be desired on the side of the reservation charges for real-time traffic: when paths are long and traverse many routers, normal clients are easily lost, as the topology and most of the interesting details about the innards of the network are unknown to them. On the other hand, the formula is easy to understand, and a little training or experience will go a long way to familiarize the clients with the correspondences between QoS parameters and fractions of resources reserved. Predictability (Property H) is high since our approach has been based on the objective (among several others) that both aspects of predictability must be fully satisfied by it. It is unfortunate that different incarnations of the same connection may be assigned different routes, as this may cause a noticeable change in the total charges. This type of unpredictability is even more unfortunate since clients will not understand why their charges should depend on what they perceive as a network's decision they cannot control in any way (thus, it is also against Property G). The only justification a network manager can resort to is that, if the change of route is due to high traffic at some point along the usual path, the longer route (and the corresponding higher charges) can be regarded as the penalty for having chosen to transmit during a "rush hour".

Stability (Property I) is potentially quite high, as we saw in Section 3. Fairness (Property J) is another important property. We concentrated our attention on some of the many viewpoints one can adopt about fairness, and we succeeded in building an approach that looks fair from at least those viewpoints.

In summary, the approach we have presented in this paper is not an ideal one, but possesses to a satisfactory degree most of the properties we listed in Section 2. Much, of course, remains to be done before our charging policy can be used in practical networks.

References

[1] D. Ferrari, "Should an Integrated-Services Internetwork be Connectionless or Connection-Oriented? A Position Paper," *Proc. Sixth Int. Workshop on Network and Operating System Support for Digital Audio and Video,* Zushi, Japan, pp. 3-4, April 1996.

[2] D. Ferrari, "Real-Time Communication in an Internetwork," *J. of High Speed Networks*, vol. 1, no. 1, pp. 79-103, 1992.

[3] D. Ferrari and A. Gupta, "Resource Partitioning for Real-Time Communication," *Proc. 1993 IEEE Symp. on Global Data Networking,* Cairo, Egypt, pp. 128-135, December 1993.

[4] C. Parris and D. Ferrari, "A Resource Based Pricing Policy for Real-Time Channels in a Packet-Switching Network," Tech. Rept. TR-92-018, International Computer Science Institute, Berkeley, CA, March 1992.