Chapter 14

Traffic Management

14.1 Introduction

Imagine a busy executive participating in a videoconference with other executives around the world. As the conference proceeds, she whips out a personal digital assistant, scrawls a note, and mails it to her colleague. Another participant brings up a spreadsheet that they edit together. While the meeting is in progress, an important telephone call interrupts one of the participants. At the end of the meeting, a video transcript of the proceedings is saved in the company archives. A librarian edits the transcript to create a meeting summary that is placed on the company Web site. Later, absentees use this summary to find out what went on in the meeting.

This scenario, though largely a fantasy in 1996, might be commonplace a decade from now. To make it happen, the participants must have access to a multimedia network that supports voice and video transport for videoconferencing, low end-to-end delays for telephony, reliable data transport for email, and bulk data transport for copying and editing the transcript. How can a network provider efficiently build such a network? How can it make sure that the network's services are available on demand, and that users are satisfied with the service they receive? How can the operator reduce the cost of providing service so that its services are competitively priced, yet profits are maximized? These are the broad questions that we will study in this chapter.

Traffic management is the set of policies and mechanisms that allow a network to efficiently satisfy a diverse range of service requests. The two fundamental aspects of traffic management, *diversity* in user requirements and *efficiency* in satisfying them, act at cross purposes, creating a tension that has led to a rich set of mechanisms. We have already

studied some of these mechanisms, such as scheduling and flow control, in Chapters 9 and 13. In this chapter, we will tie together these concepts to form a unified framework for traffic management.

Traffic management subsumes many ideas traditionally classified under *congestion control.* We say that a resource is congested when it is overloaded, so that a user experiences performance degradation. Congestion-control policies either restrict access to the resource or scale back user demand dynamically so that the overload situation disappears. Since an overload results in a loss of network efficiency, we can view congestion control as one aspect of traffic management. Traffic management is more general because it includes other mechanisms, such as scheduling and signaling, that are unrelated to congestion control.

We begin with an economic framework for traffic management in Section 14.2. To manage traffic, we need to understand traffic behavior (Section 14.3) and user behavior (Section 14.4). We introduce the notion of multiple time scales of management in Section 14.5. Subsequent sections deal with some mechanisms for traffic management, such as signaling, admission control, and capacity planning, in more detail. Finally, Section 14.12 summarizes the chapter.

14.2 An economic framework for traffic management

An economic formulation of the traffic management problem gives some useful insight. We model each network customer as having a *utility function* u that translates from a given *quality of service* or *QoS* (such as the bandwidth associated with a call[1] or the mean delay of packets sent during a call) to a degree of satisfaction, or *utility.* The greater the satisfaction, the greater the utility function of that quality of service. A user's utility function is known only to the user, and we will assume that *rational* users take actions that maximize their utility function. This modeling of user behavior, which is fundamental to the economic model, is general enough to capture a wide range of service requests, as the following example shows.

EXAMPLE 14.1

Suppose a particular user wants to transfer a file as soon as possible. We can model this with the utility function $u(t) = S - \alpha t$, where t is the time to transfer the file, S is the utility derived when the file transfer is infinitely fast, and α is the rate at which the utility declines as a function of time. Thus, as the file transfer takes longer and longer, the user's utility from the transfer linearly decreases. Note that when

[1] We use the term "call" to loosely refer to an association between a sender of data and its receiver. In an ATM network or a telephone network, a call is the same as a virtual circuit, or a physical circuit, respectively. On the Internet, with a connection-oriented transport layer such as TCP, a call is delimited by an explicit connection open and close. With a connectionless transport layer protocol such as UDP, a call is delimited by "long" idle times. (This definition is imprecise, but about the best we can do for connectionless networks!)

$t > S/\alpha$, the utility becomes negative. This reflects the fact that if the file transfer takes too long, the user feels that he or she is worse off than if the transfer had never been initiated. (Presumably this is because the file is delayed too long and the user has to pay for the transfer anyway!)

Suppose a user is participating in a videoconference. Then, to preserve interactivity, he or she may want *every* packet to arrive at the receiver before a deadline. Otherwise, the packet is too late and cannot be used to present an audio or video signal. We can model this user's requirements by:

$$\begin{aligned} u(t) = \text{if } & (t < D) \\ & \text{then } S \\ & \text{else } -\beta \end{aligned}$$

Here, t is the end-to-end delay experienced by a packet, D is the delay deadline, S is the satisfaction from a packet that meets the deadline, and $-\beta$ is the cost of missing a deadline. The penalty reflects the fact that the user has to pay for the packet even though it cannot be used.

A more sophisticated utility function measures not just the delay, but also the *probability* that a packet meets a certain delay or loss bound. For example, the utility function $u(\epsilon) = S(1 - \epsilon)$, where ϵ is the packet loss probability, reflects the fact that a user derives satisfaction S when no packets are lost, and as the probability of loss increases, the utility decreases. This loss probability here is an a priori loss probability: it might be advertised by the network provider and used by the user to choose a particular provider.

The key idea is that the utility function completely captures the user's requirements. Once we know a user's utility function, we know exactly how much he or she values a higher bandwidth over a lower delay, or lower loss rate over a lower price. This allows us to engineer a network to satisfy these requirements best. In other words, utility functions give us the vocabulary to talk about the diversity of performance requirements that we expect future applications to have.

How useful are utility functions?

Although an economic formulation of the traffic-management problem requires us to model users as having utility functions, do users really have such functions that they can express mathematically? Economists assume that even if users cannot come up with a mathematical formula, they can still express preferences for one set of resources (or one degree of performance) over another. These preferences can then be codified as a utility function. What is less clear is whether user utility func-

tions have the "nice" mathematical properties that economists want. Perhaps the best way to think about utility functions is that they allow us to come up with a mathematical formulation of the traffic-management problem that gives some insight. Although practical economic algorithms may never be feasible, policies and mechanisms based on these insights are still relevant.

14.2.1 Economic principles of traffic management

We will assume for the purposes of our discussion that a user knows his or her own utility function and tells it to the network. Given a set of user utility functions, it is desirable for the network provider to implement policies and mechanisms that try to optimize some metric on the ensemble of utility functions, such as maximizing their sum, while minimizing the cost of network infrastructure (this is often called *social welfare maximization*). The solution to this problem leads to three general principles for traffic management [Shenker 95]:

- The network should try to match its menu of service qualities to user requirements. Service menus that are more closely aligned with user requirements are more efficient. Intuitively, the more loosely a service menu matches user requirements, the more resources a network has to expend to achieve the same level of user utility. Thus, when building an integrated-services network, we should first determine the demands that will be placed on it.
- Building a single network that provides heterogeneous qualities of service is better than building separate networks for different qualities of service. For example, building a network that carries both voice and data is better than building separate networks for voice and data. Intuitively, this is because with an integrated network, the capacity not used by a voice call is available to carry data traffic, and vice versa. With separate networks, unused capacity lies idle.
- For typical utility functions, if network utilization remains the same, the sum of user utility functions increases more than linearly with an increase in network capacity. Thus, one way to increase overall user utility is to increase network capacity. Intuitively, the larger the network, the smaller the effect of statistical fluctuations. Thus, individual users experience fewer fluctuations in their service, making their use of the network more pleasant. This is a consequence of the law of large numbers.

EXAMPLE 14.2

This example is based on a similar one in reference [Shenker 95]. Consider a network consisting of a single switch with users A and B with utility functions $u(d) =$

$4 - d$ and $v(d) = 8 - 2d$, respectively, where d is the mean packet delay. B is more sensitive to delay, because its utility falls off more steeply with an increase in delay. Recall from Chapter 9 that the conservation law states that if $\rho()$ denotes a user's transmission rate, and the sum of these transmission rates is fixed, then for all delay allocations $d(A)$ and $d(B)$,

$$\rho(A)d(A) + \rho(B)d(B) = \text{constant}$$

Assume $\rho(A) = \rho(B) = 0.4$. Suppose a network does not distinguish between service qualities for the two users, so that $d(A) = d(B) = d$. Then, $0.4\,d + 0.4\,d = \text{constant} = C$. Thus, $d = C/0.8 = 1.25\,C$. The user utilities are $4 - 1.25C$ and $8 - 2(1.25C)$, so that their sum is $12 - 3.75C$.

Now, suppose the network can give a smaller delay to B and a larger one to A, through an appropriate choice of scheduling discipline. If the delay to B is $0.5C$, (which is smaller than its earlier delay of $1.25C$), then A's delay, from the conservation law, must be $(C - 0.5C * 0.4)/0.4 = 2.0C$. Thus, the sum of utilities is $4 - 2.0C + 8 - 2(0.5C) = 12 - 3C$. Clearly, this is larger than $12 - 3.75C$ for all $C > 0$. Thus, by giving higher priority to users that want lower delay, the network can increase its utility. Of course, in the process, A's utility decreases from $4 - 1.25C$ to $4 - 2C$. So, an overall increase in utility does not necessarily benefit all users. This trade-off between individual and global optimality must be made by the network operator. If the operator gives priority to B's traffic, it may compensate A for its higher delay by giving it a lower price, so that its overall utility (including its price) does not decrease.

Example 14.2 shows that the network operator, by aligning its service menu with user needs, can increase overall user utility. However, it can also increase overall user utility merely by increasing network capacity, which decreases the network's utilization and therefore its mean packet delays. To take a concrete example, suppose users of an online service complain about their response time when using interactive applications. The operator can either introduce mechanisms to give a higher priority and a lower delay to traffic from interactive applications, or increase capacity, decreasing delays to *all* applications. Both approaches will fix the problem, and the operator should choose the cheaper solution. This is a classic choice between "big and dumb" and "small and smart." How should the network operator choose between these alternatives?

When resources are scarce, then no matter how efficiently a network operator manages the network, user utility will still be low. However, once the network capacity increases beyond a minimum, one can argue that implementing intelligent traffic management is more effective than increasing capacity. As a case in point, consider a hypothetical network that provides the same quality of service (synchronous, 64-Kbps circuits with minimal jitter and delay) to all its users. Suppose some users wanted to send data at 1 Mbps instead. One solution would be to give *all* users a 1-Mbps connection, of which

most use only a 64-Kbps portion. A more efficient solution is for the network to provide heterogeneous QoS, where some users get 64 Kbps, and others get 1 Mbps. Similarly, consider another hypothetical network where no user gets a guarantee on bandwidth or delay. If some users wanted to get a guarantee of at least 8 Kbps (measured over some interval) for their calls, one solution would be to increase capacity so that *all* calls get more than 8 Kbps. The alternative, to reserve bandwidth only for these calls, usually proves more efficient, and thus is cheaper.

The choice between overprovisioning ("big and dumb") and intelligent traffic management ("small and smart") is still a matter of much debate. Some people feel that network operators should concentrate on increasing the bandwidth available to all users, and the rising tide will raise all ships. Although it is still too early to pass judgment, the ultimate decision depends on whether it will prove cheaper to build intelligent traffic management schemes, or use the same money to increase the raw capacity of the network. In this chapter, we will assume that intelligent traffic-management schemes are desirable and worth studying.

The call

The argument between the two camps led me to write this ditty, sung to the tune of Pink Floyd's "The Wall."

We don't need no reservation
We don't need ad-mission control
All applications must be adaptive
The Net works just fine, so leave it alone
Hey! Professor! Leave the Net alone!

We don't need no traffic management
Overprovision bandwidth for all
The only true god is TCP/IP
The Net isn't broken, so leave it alone
Hey! Professor! Leave the Net alone!

All we want is just flat rate pricing for all

14.2.2 Pricing

In this subsection, we will briefly study the problem of network pricing, that is, how much a public network should charge for its services. This area is fraught with complications, as the following example shows.

Consider a network with a fixed capacity, where increased usage causes every user's individual utility to decrease (for example, with increased usage, every user's mean delay might increase, thus decreasing his or her utility). If we charge users a flat fee for access, then, under some general assumptions, it can be shown that a few bandwidth hogs, who are insensitive to

delays, dominate the network and displace delay-sensitive users [MV 95]. Intuitively, with a flat fee, there is no incentive for a user to limit his or her use of the network. Thus, users who are sensitive to delay cede the network to users who are insensitive to delay. Another way to view this is to observe that the network is congested from the perspective of delay-sensitive users, but not from the perspective of delay-insensitive users, biasing the network to support only delay-insensitive users. This is probably not desirable for the network provider, because it loses revenue from delay-sensitive users.

To avoid this situation, a network operator could use *congestion pricing*, where the operator charges a user according to the *disutility* his or her traffic causes to other users. Thus, if a user sends a lot of traffic when the network is already loaded, causing delays and packet losses to other users, it is charged heavily for the discomfort it causes. If it sends the same load at an off-peak time when the network is underloaded, then because it causes almost no discomfort to other users, its price is low. With this scheme, the eventual equilibrium is optimal in the sense that it not only maximizes the network operator's revenue, but also each user's utility [MV 95]. Congestion fees discourage excessive network usage when the network is loaded, thus making the network usable by delay-sensitive users.

However, congestion pricing in the simple form just stated requires the network provider to know every user's utility function, something a network provider has no way to directly determine. To begin with, users may not even know what traffic their application might generate, much less their utility function. Even if users knew these things, and had some way of informing a network of their utility function, they could lie. Thus, a simple-minded approach to congestion pricing, which, unfortunately, is the only kind of approach amenable to analysis, cannot be used in a network setting right away. We can only use the insights gained from the analysis when developing pricing schemes.

Here is another example of a complication we run into with network pricing. Suppose we claim that a network provider can infer users' utilities from their willingness to pay for services. The idea is that the more utility a user obtains from using the network, the higher the price he or she is willing to pay. Thus, the network could charge different prices for different services, and users' willingness to pay this price would reveal their utility functions. However, when we use this principle with congestion pricing, we run into a chicken-and-egg problem. The network operator chooses a price depending on each user's inferred utility function (which allows it to determine the disutility caused by each user) and the inferred utility function itself depends on the price! If we are not careful, this can lead to a situation where the price charged from each user oscillates wildly over time.

Despite these complications, the key point is that by setting a price for usage, the network can *control* user demand, at least broadly, thus modifying the traffic load on the system. Therefore, pricing can be used as a tool for traffic management. This is still an area of ongoing research.

14.3 Traffic models

To effectively manage traffic, a network provider must know not only the requirements of individual applications and organizations, but also their "typical" behavior. For example, an Internet service provider must guess at how long a "typical" user uses a modem line, so that modem pools have sufficient modems to keep the probability of a blocked call low. A *traffic model* summarizes the expected behavior of an application or an aggregate of applications.

Traffic models fall into two broad categories. Some models are obtained by detailed traffic measurements of thousands or millions of connections over days or years. Others