Overview

This book is about stochastic networks and their applications. Large-scale systems of interacting components have long been of interest to physicists. For example, the behaviour of the air in a room can be described at the microscopic level in terms of the position and velocity of each molecule. At this level of detail a molecule's velocity appears as a random process. Consistent with this detailed microscopic description of the system is macroscopic behaviour best described by quantities such as temperature and pressure. Thus the pressure on the wall of the room is an average over an area and over time of many small momentum transfers as molecules bounce off the wall, and the relationship between temperature and pressure for a gas in a confined volume can be deduced from the microscopic behaviour of molecules.

Economists, as well as physicists, are interested in large-scale systems, driven by the interactions of agents with preferences rather than inanimate particles. For example, from a market with many heterogeneous buyers and sellers there may emerge the notion of a price at which the market clears.

Over the last 100 years, some of the most striking examples of large-scale systems have been technological in nature and constructed by us, from the telephony network through to the Internet. Can we relate the microscopic description of these systems in terms of calls or packets to macroscopic consequences such as blocking probabilities or throughput rates? And can we design the microscopic rules governing calls or packets to produce more desirable macroscopic behaviour? These are some of the questions we address in this book. We shall see that there are high-level constructs that parallel fundamental concepts from physics or economics such as energy or price, and which allow us to reason about the systems we design.

In this chapter we briefly introduce some of the models that we shall encounter. We'll see in later chapters that for the systems we ourselves con-

Cambridge University Press
978-1-107-03577-5 - Stochastic Networks
Frank Kelly and Elena Yudovina
Excerpt
More information

struct, we are sometimes able to use simple local rules to produce macroscopic behaviour which appears coherent and purposeful.

Queueing and loss networks

We begin Chapter 1 with a brief introduction to Markov chains and Markov processes, which will be the underlying probabilistic model for most of the systems we consider. In Chapter 2 we study queueing networks, in which customers (or jobs or packets) wait for service by one or several servers.

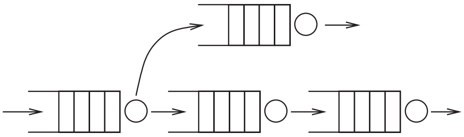

Figure 1 A network of four queues.

First, we look at a single queue. We shall see how to model it as a Markov process, and derive information on the distribution of the queue size. We then look briefly at a network of queues. Starting from a simplified description of each queue, we shall obtain information about the system behaviour. We define a traffic intensity for a simple queue, and identify Poisson flows in a network of queues.

A natural queueing discipline is first-come-first-served, and we also look at processor sharing, where the server shares its effort equally over all the customers present in the queue.

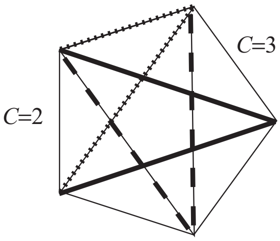

Figure 2 A loss network with some of the routes highlighted.

In Chapter 3, we move on to consider loss networks. A loss network consists of several links, each of which may have a number of circuits. The classical example of its use is to model landline telephone connections between several cities. When a telephone call is placed between one node and another, it needs to hold simultaneously a circuit on each of the links on a route between the two nodes – otherwise, the call is lost. Note the key differences between this and a queueing network, which are summarized in Table 1.

Table 1. *Queueing versus loss networks*

Queueing networks	Loss networks
Sequential use of resources	Simultaneous resource possession
Congestion $\Longrightarrow$ delay	Congestion $\Longrightarrow$ loss

First, we treat loss networks where routing is fixed in advance. We show that the simple rules for call acceptance lead to a stationary distribution for the system state that is centred on the solution to a certain optimization problem, and we'll relate this problem to a classical approximation procedure.

Next, we allow calls to be rerouted if they are blocked on their primary route. One example we consider is the following model of a loss network on a complete graph. Suppose that if a call arrives between two nodes and there is a circuit available on the direct link between the two nodes, the call is carried on the direct link. Otherwise, we attempt to redirect the call via another node (chosen at random), on a path through two links. (Figure 2 shows the single links and some of the two-link rerouting options.)

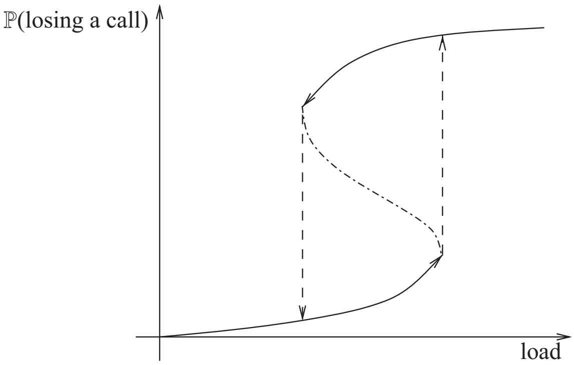

Figure 3 Hysteresis.

What is the loss probability in such a network as a function of the arrival rates? In Figure 3 we sketch the proportion of calls lost as the load on the system is varied. The interesting phenomenon that can occur is as follows: as the load is slowly increased, the loss probability increases slowly to begin with, but then jumps suddenly to a higher value; if the load is then slowly decreased, the loss probability does not trace the same curve, but drops back at a noticeably lower level of load. The system exhibits *hysteresis*, a phenomenon more often associated with magnetic materials.

How can this be? We model the network as an irreducible finite-state Markov chain, so it must have a *unique* stationary distribution, hence a *unique* probability of losing a call at a given arrival rate.

On the other hand, it makes sense. If the proportion of occupied circuits is low, a call is likely to be carried on the direct route through a single link; if the proportion of occupied circuits is high, more calls will be carried along indirect routes through two circuits, which may in turn keep link utilization high.

How can both of these insights be true? We'll see that the resolution concerns two distinct scaling regimes, obtained by considering either longer and longer time intervals, or larger and larger numbers of nodes.

We end Chapter 3 with a discussion of a simple dynamic routing strategy which allows a network to respond robustly to failures and overloads. We shall use this discussion to illustrate a more general phenomenon, *resource pooling*, that arises in systems where spare capacity in part of the network can be used to deal with excess load elsewhere. Resource pooling allows systems to operate more efficiently, but we'll see that this is sometimes at the cost of losing early warning signs that the system is about to saturate.

Decentralized optimization

A major practical and theoretical issue in the design of communication networks concerns the extent to which control can be decentralized, and in Chapter 4 we place this issue in a wider context through a discussion of some ideas from physics and economics.

In our study of loss networks we will have seen a network implicitly solving an optimization problem in a decentralized manner. This is reminiscent of various models in physics. We look at a very simple model of electron motion and establish Thomson's principle: the pattern of potentials in a network of resistors is just such that it minimizes heat dissipation for a given level of current flow. The local, random behaviour of the electrons results in the network as a whole solving a rather complex optimization problem.

Of course, simple local rules may lead to poor system behaviour if the rules are the wrong ones. We'll look at a simple model of a road traffic network to provide a chastening example of this. Braess's paradox describes how, if a new road is added to a congested network, the average speed of traffic may fall rather than rise, and indeed everyone's journey time may lengthen. It is possible to alter the local rules, by the imposition of appropriate tolls, so that the network behaves more sensibly, and indeed road traffic networks have long provided an example of the economic principle that externalities need to be appropriately penalized if the invisible hand is to lead to optimal behaviour.

In a queueing or loss network we can approach the optimization of the system at various levels, corresponding to the two previous models: we can dynamically route customers or calls according to the instantaneous state of the network locally, and we can also measure aggregate flows over longer periods of time, estimate externalities and use these estimates to decide where to add additional capacity.

Random access networks

In Chapter 5, we consider the following model. Suppose multiple base stations are connected with each other via a satellite, as in Figure 4. If a base station needs to pass a message to another station it sends it via the satellite, which then broadcasts the message to all of them (the address is part of the message, so it is recognized by the correct station at the end). If transmissions from two or more stations overlap, they interfere and need to be retransmitted. The fundamental issue here is contention resolution: how should stations arrange their transmissions so that at least some of them get through?

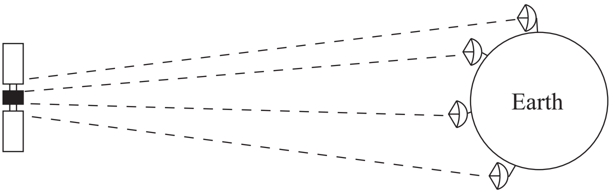

Figure 4 Multiple base stations in contact via a satellite.

If stations could instantaneously sense when another station is transmitting, there would be no collisions. The problem arises because the finite speed of light causes a delay between the time when a station starts

Cambridge University Press
978-1-107-03577-5 - Stochastic Networks
Frank Kelly and Elena Yudovina
Excerpt
More information

transmitting and the time when other stations can sense this interference. As processing speeds increase, speed-of-light delays pose a problem over shorter and shorter distances, so that the issue now arises even for distances within a building or less.

Consider some approaches.

- We could divide time into slots, and assign, say, every fourth slot to each of the stations. However, this only works if we know how many stations there are and the load from each of them.
- We could implement a token ring: set up an order between the stations, and have the last thing that a station transmits be a “token” which means that it is done transmitting. Then the next station is allowed to start transmitting (or it may simply pass on the token). However, if there is a large number of stations, then simply passing the token around all of them will take a very long time.
- The ALOHA protocol: listen to the channel; if nobody else is transmitting, just start your own transmission. As it takes some time for messages to reach the satellite and be broadcast back, it is possible that there will be collisions (if two stations decide to start transmitting at sufficiently close times). If this happens, stop transmitting and wait for a *random* amount of time before trying to retransmit.
- The Ethernet protocol: after k unsuccessful attempts, wait for a random time with mean 2^k before retransmitting.

We shall study the last two approaches, which have advantages for the challenging case where the number of stations is large or unknown, each needing to access the channel infrequently.

We end Chapter 5 with a discussion of distributed random access, where each station can hear only a subset of the other stations.

Broadband networks

Consider a communication link of total capacity C, which is being shared by lots of connections, each of which needs a randomly fluctuating rate. We are interested in controlling the probability that the link is overloaded by controlling the admission of connections into the network.

If the rate is constant, as in the first panel of Figure 5, we could control admission as to a single-link loss network: don’t admit another connection if the sum of the rates would exceed the capacity. But what should we do if the rate needed by a connection fluctuates, as in the second panel? A conservative approach might be to add up the peaks and not admit another

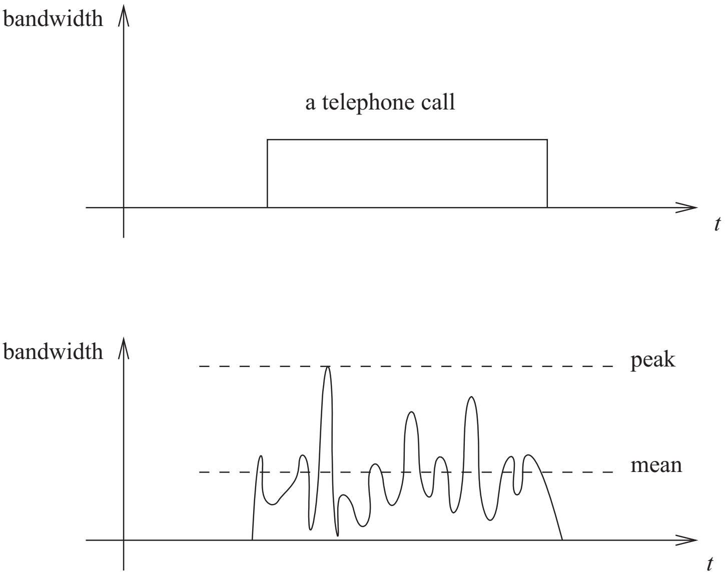

Figure 5 Possible bandwidth profiles; mean and peak rates.

connection if the sum would exceed the capacity. But this looks very conservative, and would waste a lot of capacity if the peak/mean ratio is high. Or we might add up the means and only refuse a connection if the sum of the means would exceed the capacity. If there is enough buffering available to smooth out the load, we might expect a queueing model to be stable, but connections would suffer long delays and this would not work for real-time communication.

In Chapter 6, we use a large deviations approach to define an *effective bandwidth* somewhere between the mean and the peak; this will depend on the statistical characteristics of the bandwidth profiles and on the desired low level of overload probability. Connection acceptance takes on a simple form: add the effective bandwidth of a new request to the effective bandwidths of connections already in progress and accept the new request if the sum satisfes a bound. This approach allows the insights available from our earlier model of a loss network to transfer to the case where the bandwidth required by a connection fluctuates.

If the connections are transferring a file, rather than carrying a real-time video conversation, another approach is possible which makes more efficient use of capacity, as we describe next.

Internet modelling

What happens when I want to see a web page? My device (e.g. computer, laptop, phone) sends a request to the server, which then starts sending the page to me as a stream of packets through the very large network that is the Internet. The server sends one packet at first; my device sends an acknowledgement packet back; as the server receives acknowledgements it sends further packets, at an increasing rate. If a packet is lost (which will happen if a buffer within the network overflows) it is detected by the endpoints (packets have sequence numbers on them) and the server slows down. The process is controlled by TCP, the *transmission control protocol* of the Internet, implemented at the endpoints (on my device and on the server) – the network itself is essentially dumb.

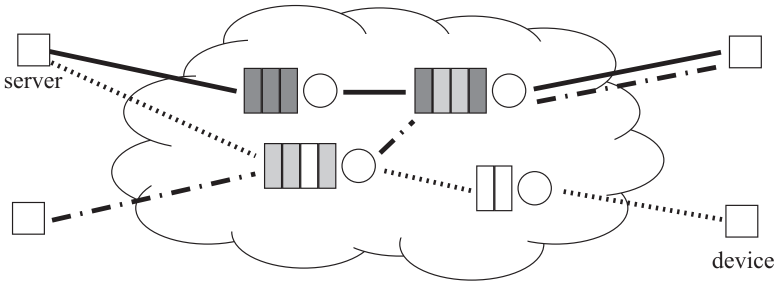

Figure 6 A schematic diagram of the Internet. Squares correspond to devices and servers, the network contains resources with buffers, and many flows traverse the network.

Even this greatly simplified model of the Internet raises many fascinating questions. Each flow is controlled by its own feedback loop, and these control loops interact with each other through their competition for shared resources. Will the network be stable, and what does this mean? How will the network end up sharing the resources between the flows?

In Chapter 7, we discuss various forms of fairness and their interpretation in terms of optimization problems. We will study dynamical models of congestion control algorithms that correspond to primal or dual approaches to the solution of these optimization problems.

Over longer time scales, flows will come and go, as web pages and files are requested or their transfers are completed. On this time scale, the network's overall behaviour resembles a processor-sharing queueing system, with congestion decreasing flow rates and lengthening the delay before a flow is completed. We'll look at models of this behaviour in Chapter 8.

Cambridge University Press
978-1-107-03577-5 - Stochastic Networks
Frank Kelly and Elena Yudovina
Excerpt
More information

We shall see that to understand the Internet's behaviour we need to consider many time and space scales, depending on the level of aggregation and the time intervals of interest. At the packet level the system looks like a queueing network, and we need to model packet-level congestion control algorithms and scheduling disciplines; but if we look on a longer time scale, a flow makes simultaneous use of resources along its route, and flow rates converge to the solution of an optimization problem; and over longer time scales still, where the numbers of flows can fluctuate significantly, the system behaves as a processor-sharing queue.