HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Engineering Physics and Mathematics
Degree programme of Engineering Physics

Aleksi Penttinen

Mathematical Models for Marking in Congestion Pricing

Master's thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology

Supervisor: professor Jorma Virtamo
Instructor: professor Jorma Virtamo

Espoo, 9th August, 2001

Author: Aleksi Penttinen
Department: Department of Engineering Physics and Mathematics
Major subject: Systems analysis and operations research
Minor subject: Teletraffic theory

Title of thesis: Mathematical Models for Marking in Congestion Pricing
Finnish title: Matemaattiset mallit ruuhkahinnoittelun merkinnässä
Date: 9th August, 2001 **Pages:** 73

Chair: S-38 Teletraffic theory
Supervisor: Professor Jorma Virtamo
Instructor: Professor Jorma Virtamo

Congestion pricing is a simple control scheme for a packet network such as the Internet. It can be implemented e.g. as follows: Network signals the local (overflow related) congestion costs to users by marking traversing packets at the resources. Every marked packet implies a small charge to its sender and so, in case of congestion, the increasing flow of marks, i.e. the rising price, acts as an incentive for users to reduce their load. How the packets should be marked, however, has remained largely an open question.

This thesis considers the marking issue from a modelling point of view. The general congestion pricing scheme is discussed from the underlying philosophy to some practical implementations. Especially the theoretical framework developed by Kelly et al. [21] and existing heuristic marking schemes are analysed.

Three different approaches based on the M/M/1/K queuing model are presented to point out that the correct prices can be achieved by marking each packet with the probability that the buffer overflows during that busy period. The risk of congestion can be itself interpreted as a continuous price stamp. This way the users are provided an early warning of congestion and even some packet losses can be avoided.

Based on this observation, a marking strategy is also suggested: In case of an overflow, all the packets in the resource are marked; otherwise, unmarked packets are marked according to the price-probability. This method is called the predictive marking.

The state-dependent overflow probabilities are further calculated for M/G/1/K and a simple priority queue models. Regardless of the model, the form of price as a function of state remains roughly the same. Hence, the form can be approximated for GI/GI/1/K models using a simple exponential formula derived from the diffusion approximation. Furthermore, in the light of a few numerical examples the accuracy of the approximation seems promising.

Despite the need of estimating parameters at resources, the predictive marking approach solves the marking problem in simple mathematical models. The assumption on non-correlated traffic, however, may limit the direct applicability of the method in practice.

Keywords: Resource marking, congestion pricing, Internet congestion control, differentiated services

Accepted: **Library code:**

Tekijä: Aleksi Penttinen
Osasto: Teknillisen fysiikan ja matematiikan osasto
Pääaine: Systeemi- ja operaatiotutkimus
Sivuaine: Teleliikenneteoria

Työn nimi: Matemaattiset mallit ruuhkahinnoittelun merkinässä
English title: Mathematical Models for Marking in Congestion Pricing
Päivämäärä: 9.8.2001 **Sivumäärä:** 73

Professuuri: S-38 Teleliikenneteoria
Työn valvoja: Professori Jorma Virtamo
Työn ohjaaja: Professori Jorma Virtamo

Ruuhkahinnoittelu on yksinkertainen hallintaperiaate Internet-tyyppiselle pakettiverkolle. Se voidaan toteuttaa mm. seuraavasti: Verkko signaloi paikalliset (ylivuodoista riippuvat) ruuhkakustannukset merkitsemällä hintaleimoja reitittimen läpi kulkeviin paketteihin. Merkityistä paketeista peritään korvaus paketin lähettäjältä ja näin ruuhkatilanteissa kasvava merkkien vuo, hinta, kannustaa käyttäjiä vähentämään aiheuttamaansa kuormitusta. Kuinka paketit tulisi sitten merkitä?
Tässä työssä syvennytään merkintäongelmaan matemaattisten mallien avulla. Aluksi analysoidaan koko järjestelmää yleisemmin, lähtien taustalla olevasta filosofiasta ja päätyen vaihtoehtoisiin käytännön toteutusmenetelmiin. Tarkemmin syvennytään ylläkuvattuun teoreettiseen mallikehyseen (Kelly et al. [21]), sekä heuristisiin merkintätekniikoihin.
Työssä esitetään kolme erilaista M/M/1/K-jonomalliin perustuvaa lähestymistapaa ja osoitetaan, että oikeiden hintojen saamiseksi paketti voidaan merkitä sillä todennäköisyydellä, millä kyseisen kiirejakson aikana tapahtuu ylivuoto. Ruuhkariski voidaan myös sellaisenaan tulkita jatkuva-arvoiseksi hintaleimaksi. Näin tarjotaan käyttäjille ennakkovaroitus uhkaavista ylivuodoista ja jopa vältetään pakettien menetyksiä.
Merkintään suositellaan seuraavaa politiikkaa: Ylivuodon tapahtuessa kaikki systeemin paketit merkitään ja muulloin merkitsemättömän paketin poistuessa systeemistä se saa merkin senhetkisellä hintatodennäköisyydellä. Menetelmää kutsutaan ennakoivaksi merkinnäksi.
Tilariippuvat ylivuototodennäköisyydet lasketaan myös M/G/1/K- ja prioriteettijonomalleille. Huomataan, että hinta tilan funktiona pysyy mallista riippumatta jokseenkin samanmuotoisena. Tätä muotoa approksimoidaan GI/GI/1/K malleille yksinkertaisella eksponentiaalisella kaavalla, joka on johdettu diffuusioapproksimaation avulla. Approksimaation tarkkuus havaitaan numeeristen esimerkkien valossa lupaavaksi.
Parametrien estimointitarpeesta huolimatta ennakoiva merkintä ratkaisee merkintäongelman yksinkertaisten mallien puitteissa. Oletus korreloimattomasta liikenteestä saattaa kuitenkin rajoittaa menetelmän suoraa sovellusta käytäntöön.

Avainsanat: pakettimerkintä, ruuhkahinnoittelu, Internetin ruuhkanhallinta differentioidut palvelut

Hyväksytty: **Työn sijaintipaikka:**

Acknowledgements

This thesis was written in the Networking Laboratory of Helsinki University of Technology during the spring 2001 and was funded by the COM^2 project of the Academy of Finland.

First of all, I would like to express my gratitude to the supervisor of this thesis, professor Jorma Virtamo, not only for his valuable comments and guidance, but also for providing an interesting topic for this work. The process of writing was very rewarding and encouraged me to continue my studies.

My other workmates deserve a special mention for being always open to my numerous questions. They have also managed to create it fun and enjoyable to work here.

Finally, I feel obliged to thank all of my friends, both foreign and domestic, for they have kept me going during the process.

Espoo, 9$^{\text{th}}$ August, 2001

Aleksi Penttinen

Contents

1 Introduction **1**

2 Congestion Pricing **5**

2.1 Foundations 5

2.1.1 Economic background 5

2.1.2 Congestion and the Internet 9

2.2 Proportionally Fair Pricing 11

2.2.1 Mathematical framework 12

2.2.2 Discussion 19

2.3 Other related schemes 21

3 Marking **23**

3.1 Basics of marking 23

3.1.1 Sample Path Shadow Prices 24

3.2 Different marking schemes 27

3.2.1 Mark after loss 27

3.2.2 Threshold 27

3.2.3 Virtual Queue 28

3.2.4 RED 29

3.2.5 REM 30

4 Predictive Marking **31**

4.1 Mathematical modelling 31

4.1.1 Simple M/M/1/K queuing model 32
4.1.2 Alternative approaches 36
4.2 On marking mechanisms 40
4.2.1 Is one bit enough? 41
4.2.2 Time of marking 41
4.2.3 Predictive marking for approximating SPSP 43
4.3 General queuing models with Markovian properties 44
4.3.1 Models with embedded Markov chain 44
4.3.2 Diffusion approximation 50
4.4 Fractional Brownian motion 54

5 Analysis **58**
5.1 Comparison of different methods 58
5.1.1 Effects of the process 59
5.1.2 Diffusion approximation 61
5.2 Simulation experiment 64
5.2.1 Differences with SPSP 64

6 Conclusions **66**
6.1 Further work 68

Chapter 1

Introduction

There is no need to commend the scale or importance of the current Internet. It has established its firm status on the communications area and more and more services are installed to be available world-wide through the network. In terms of technology, however, the Internet is becoming old – so old that the first symptoms of the age are already turning up. Although the available capacity is growing with developing technologies, the demand for the same capacity is growing even faster. New services, such as multimedia applications and distributed computing, have emerged on large scale to co-exist with the more traditional data traffic such as e-mail and file transfer. The principles, standards and the software, which have successfully reigned over the Internet for more than a decade, are becoming inescapably outdated in many ways. It is not only that the bandwidth is becoming inadequate, but the heterogeneity of the new demands which is forcing the foundation of the network, the principles of controlling it, to shape up to meet the challenge.

Increasing utilization of real-time services puts new demands on the resources consisting of preferences on delays and packet losses, i.e. on Quality of Service (QoS). These demands are completely strange to the way Internet was designed to operate and so the requirements cannot usually be met. Thus, a user running a real-time application in the Internet is given an incentive to bypass the mechanisms controlling the flows which, to a large extent, threatens the stability of the whole network. Hence, it is quite natural that there is a wide consensus that changes in the current congestion control principles are unavoidable.

The intense ongoing research is looking for alternatives to enable the transport of audio, video, real-time, and classical data traffic within a single network infrastructure. Within the Internet Engineering Task Force (IETF), which is a large open international community of researchers concerned with the evolution of the Internet, there are two working groups addressing the problem from different perspectives. The Integrated Services (intserv) approach is to provide connection-oriented schemes, where the QoS requirements are met with admission control and resource reservation. The other approach, Differentiated Services (diffserv), is aiming at to provide QoS by priority classes and queuing disciplines within a connectionless environment.

In this thesis we shall explore a different approach to the Internet congestion control. Suppose that the network generates charges to direct users actions. Rising prices act as an incentive for the users to reduce their load and so the aggregate flows evolve towards a goal set by the network. A natural selection for this goal would be the maximization of the resource usage which means that the charges can be interpreted as the congestion costs occurred in the network. In this case the users themselves decide the fair allocation of the resources by defining their own reactions to the prices. This means roughly that if you want better service than another user you must be ready to pay more than that user. These are the principles of congestion pricing.

Pricing can also be motivated from a somewhat different premise. Any scheme equipped with service differentiation must include some form of pricing or other incentive to avoid the "tragedy of commons", lack of any reason to use anything else than the best possible priority or service. Thus, the future Internet must apply some form of pricing in order to serve ever altering demands of diverse traffic flows. Optimally the pricing would be implemented so that it will allow new services to be developed without the need of changing the whole pricing system every time there is a new type of demand emerging.

The proposals for the future Internet are many and varied, even within the congestion pricing principle, but one of the most elegant approaches is the Proportionally Fair Pricing by Kelly et al. Proportionally Fair Pricing (PFP), described in detail in the next chapter, is able to provide fairness, stability and arbitrarily differentiated services all in one simple network model. It seems especially appealing for the reason that it can be implemented by using

existing Internet standards and proposals.

The principle of the scheme is straightforward; users send their traffic and the network generates feedback signals which are small charges to the users. All the complexity is left on the end users who may behave as they wish, knowing that they will be charged accordingly. The congestion control and fairness is so implemented solely by the self interest of the users. The network is left only with the task to generate these price signals reflecting the congestion costs in each resource. How should the prices be calculated?

This work is intended to introduce a parallel interpretation to these prices and to be a survey of mathematical models related to determining the prices of individual packets within PFP. From this starting point one cannot hope to derive an ultimate solution to the problem and so the aim is merely to explore the possibilities and limitations such models pose and to bring new aspects into the discussion. We shall provide a solid groundwork for further development believing that the issues considered here are relevant also for problems appearing in other DiffServ proposals.

The organization of the thesis is as follows. Chapter 2 sheds light on the background of congestion pricing. We start from the economic philosophy behind the scheme, continue by outlining some incentives to improve the current Internet technology, and then describe in detail the mathematical framework, Proportionally Fair Pricing by Kelly et. al, as an example to put all this in practice. The chapter is concluded with a brief survey on the other proposed implementations of Congestion Pricing.

Chapter 3 is devoted to packet marking procedures, how the end-nodes can be made aware of the congestion prices at resources by piggy-backing the information on the traversing packets and especially how this information should be determined. In Chapter 4, after relating the packet price to the overflow probability, we shall look into calculations required to reveal this probability in the context of various mathematical models.

In Chapter 5 we examine the robustness and the behaviour of price under different assumptions. Without going into any detail with traffic modelling we are able to find the general form of the pricing function easily described by a functional form that requires only a few parameters to be estimated.

Finally we conclude in Chapter 6 and discuss further possible developments available in this field of study.

Chapter 2

Congestion Pricing

2.1 Foundations

The concept of congestion pricing emerged from idealistic economic models, where the aim was to provide fair resource allocation between competing instances by means of pricing. Concurrently, when these models were suggested for Internet traffic management, a pragmatic development on congestion control was under way within the TCP/IP protocol suite to fight the menace of congestion collapse of the furiously expanding Internet. As the premises of both the economists and the engineers were roughly the same, it was natural that the mathematical theory of congestion pricing evolved by combining ideas from both camps. This section will shed light on the foundations of congestion pricing, both in economics and in the Internet world and then follow the evolution which lead to various independent theoretical proposals, of which the *proportionally fair pricing* scheme by Kelly et al. [11, 21, 17] will play an essential role in this presentation.

2.1.1 Economic background

Fairness in networks

When a network becomes congested, the limited resources must be shared between users and some data must be rejected or subjected to delay. Although

the principle of sharing could be arbitrary, maximum efficiency and stability is in this case achieved by a socially accepted, *fair*, resource allocation. This can be motivated by noting that an allocation that is not fair could provoke cut-throat competition among the users, essentially including some sort of greedy behaviour resulting in increased losses at the resource and hence inefficiency in the network.

However, the fairness issue is far from unambiguous. From the network point of view, dividing the resource equally among the users would seem fair, but it is the users' point of view solely that can have any effect on their demands. An equal share seems not fair to users as they may have very different requirements on the resource. For example, consider a situation where a user is watching an important real-time video footage and his share may not be enough for performing the task smoothly while some other may be transferring a large file and gone to have a cup of coffee meanwhile and would not mind if the transfer took a few seconds more. If the video is that important would it not be fair to allocate a larger share for that service than the file transfer? Maybe yes, but the network is not generally aware of the importance of the data it carries and thus not well placed to decide for the allocation. In the classic cake sharing analogy, it is not the cake which can decide how it should be divided but the people eating it.

Pricing for fairness

Users' requirements and preferences on a network may differ in quantity and quality both in location and time. Obviously only users themselves can announce their own needs, but they cannot be assumed to cooperate voluntarily. Furthermore, the possibility of users negotiating the allocation themselves is far too slow and complex to be implemented in a communications network. The simple solution is that the users announce the importance of their own data to the network which decides the fair allocation based on the information received from the users. Naturally, if we allow any kind of priorization in the network there *must* be an incentive for a user not to use a better service than what is fair. Otherwise everybody could announce, just for convenience, that their information is of utmost importance and the whole construction of importance classification breaks down. This incentive should be so strong that

even the possible misbehaving users causing congestion on purpose would have to seriously consider their actions.

The economic approach is to put a price on the social cost of congestion. All the users having a share of the resource pay a constant price proportional to their share, a price that depends only on how much and how important data cannot be carried due to the congestion. If all the traffic can be transmitted there will be no costs to the users. In economic terms congestion is an extrenality to users – a factor that has an effect on one's welfare but is under somebody else's control. Setting an appropriate price for the congestion, users' welfare is changed and congestion becomes their concern by limiting the increase in welfare in the presence of congestion.

The effect of pricing is threefold: It provides an incentive to avoid congestion and even to balance the load over the time scale so that the utilization is maximised and finally it provides an arbitrarily differentiated set of services as users have total control on their data and only the charges differ. This is the general principle of pricing; information (prices) is conveyed to direct consumption. If the prices are set to correspond the marginal cost of upgrading the resource (so that it would be able to handle all the offered traffic), the market equilibrium is reached.

Resource pricing has always been commonly used in other areas of economics, think about electric power markets, airlines or even common market places, but in the Internet context this view was first introduced by economists McKie-Mason and Varian [28], who proposed an online packet auction, smart market. Their work is further discussed in section 2.3.

Fairness of pricing

Whilst congestion pricing is a result of a very clear line of thought, it raises some justified doubts on the underlying assumptions which should be discussed.

First of all, is there a demand for the congestion pricing scheme in the Internet altogether? The framework provides a somewhat different set of services than traditional pricing for flat rate, admission or carried traffic. In those users can *predict the cost* of their actions, while in congestion pricing they are not

only uncertain about the rate they receive but also about the price they have to pay. However, although the costs cannot be forecasted, one can transfer files with arbitrary criteria, for example "as fast as possible", "by the next full hour" or "with minimum cost" which are not supported in the common best-effort network with predictable prices. It is hard to predict whether or not the freedom of congestion pricing is preferred to the predictability of traditional pricing in the process of standardization.

Assuming that congestion means higher price for the users and hence higher revenue to the network provider, what if the network provider is deliberately allowing congestion in order to gain profits? To answer this question we have to go back into the "real world" analogies as the problem is the one of *monopoly*. If the monopolist has enough freedom over pricing it can maximise its revenue anyway regardless of the pricing method used. It could be argued that in a competitive world such monopolies cannot exist; users will change their operator if experiencing bad quality of service and high prices.

Another area of concern is about the pricing or more accurately use of money associated with fairness. If one gets his traffic through to network only by paying, will the Internet then become the property of the wealthy? What happens to the universities and non-profit organizations? This is a problem of distribution of wealth, not of the method. It should be noted that the prices do not have to be directly counted in money as long as it provides an incentive for users to react. Some kind of distributed mint could provide the solution. This is an obvious and interesting problem, which cannot be neglected, but due to its philosophical nature it is out of the scope of this thesis.

In summary, as the amount of resource cannot be increased in the operating time scale, the network controls the demand by pricing; users are expected to react to alterations in price by changing their transmission rate. Congestion causes prices to rise and higher prices should attenuate demand. From the users' point of view, a user selects his own preferences from the network by deciding how much he is willing to pay for the packets to be carried. In order to get a larger share of the resource or decreased blocking probability, user must be ready to pay a higher unit price.

Congestion pricing is a part of vast field of Internet economics (for a compact overview see e.g. [29]) but it should not be confused with any business model

of a network provider. Although it could finance some upgrading, it is not intended to cover any infrastructure or operating costs but merely encourage the end-users to avoid causing congestion by economical means.

2.1.2 Congestion and the Internet

Another major incentive to the development of congestion pricing has been the technology of the Internet. It is a typical example of a packet switching network where the data is transmitted between two end nodes in individual packets of varying sizes each containing the source and destination addresses. Packets travel trough the network independently via routers which handle the switching operations. At a router, in case of heavy traffic, packets are buffered until the router is able to forward them and, if the buffer becomes full, the arriving packets will be discarded.

TCP/IP and congestion

In the current Internet the TCP/IP protocol suite handles the traffic management operations on the logical links (for detailed description see e.g. [40]) between routers and end-nodes all over the world. Internet Protocol (IP) is responsible for providing connectionless service between end systems whilst the connection establishment/termination and the actual data transfer is done above it on the transport layer. The transport layer protocols defined in the TCP/IP stack are the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP).

From the congestion point of view the most important feature of transport level protocols is the flow control. In TCP the flow control comprises of a window system with the slow start mechanism and the congestion avoidance algorithm of Jacobson [13]. Next we will outline the principles how it handles congestion situations.

All successfully received packets are confirmed by destination-nodes in form of acknowledgments (ACK). Each sender has a certain number, allowed window, of packets which can be sent without acknowledgment. This corresponds roughly to the transmission rate. Sender's allowed window is the smaller of

the dynamically adjusted congestion window (`cwnd`) and the allowed window announced by the receiver. TCP aims to avoid congestion by changing the size of `cwnd` and thus the sending rates of users. Congestion window updating was originally implemented followingly [41]; when initializing a connection `cwnd` is increased by one segment for each received ACK (this is called slow start, although the growth is actually exponential) and in the case a segment (packet) loss is detected due to a timeout, a threshold, `ssthresh`, is set to be half of the current congestion window. After that `cwnd` is set to one and is again increased by one segment for each ACK until the threshold is reached. Above `ssthresh` the window grows by one per round trip time (that is, linearly). Again, at any time if a timeout occurs `ssthresh` is set to be half of `cwnd` and the transmission starts with `cwnd`=1.

This is the standard flow control procedure in TCP, but it soon proved to be too conservative and later variants of the protocol include also other features. Jacobson [14] proposed two improvements in the basic algorithm, fast retransmit and fast recovery. Fast retransmit enables TCP to quickly replace a damaged packet within the flow without the need of waiting for the timeout. Fast recovery means that in case of a lost segment retransmission is done and then, instead of starting with the slow start, `cwnd` is cut half and then increased linearly.

In this presentation there is no need to go into further details of the protocol (an excellent collection of related papers can be found from [1]), but to notice that the flow control is essentially implemented by detecting packet losses. Although the data integrity is secured by effective retransmission policies, this makes TCP slow and usually unable to provide e.g. constant amount of bandwidth.

Naturally TCP cannot provide satisfying service for some types of traffic, especially real-time applications, and so the use of the user datagram protocol (UDP) is becoming popular. UDP, however, does not have any built-in end-to-end congestion control or error recovery and hence is able to compete unfairly against TCP for resources. Increase in these unresponsive flows (ones that fail to reduce their offered load at a router when experiencing an increased packet drop rate) could lead to congestion collapse of the Internet as argued by Floyd and Fall [7]. Furthermore, there is an increasing demand for applications with

different requirements on QoS or delay, while the current TCP/IP is suitable only for "best-effort" type traffic.

These problems have stimulated intense research on the future Internet (as well as other network) protocols to increase performance and robustness and on the other hand to provide increased flexibility for the services. Although many of the problems and solutions to them have been recognised a long time a ago, the fast development of the commercial network overruled the early schemes, such as [37], which did not receive much attention at the time. Next we will briefly discuss some of the more recent proposals which are indeed reaching the implementation phase.

Emergence of ECN

Floyd [6] presented Explicit Congestion Notification (ECN), described in detail in [35, 36], to provide congestion awareness extension into the current TCP/IP suite. The idea of the method is as follows; a single ECN-bit in the IP header is marked in a router in the imminence of congestion and thus the congestion is detected before a significant amount of packets are lost. Floyd and Jacobsson also suggested an algorithm, Random Early Detection (RED) [8], for the implementation of a buffer control applying the ECN, which is discussed in some detail in the next chapter.

The engineering approach has provided elegant and practical methods to improve the Internet. However, the solutions are severely constrained by the current protocols and the fundamental question of fairness is completely evaded. Is the TCP/IP combination a valid choice for the diverse requirements on the future Internet? What prevents the users from neglecting the ECN-marks or modifying the TCP-protocol to fight more aggressively for available bandwidth?

2.2 Proportionally Fair Pricing

Motivated by the problems of the Internet, a body of work following Gibbens and Kelly [11] has emerged to provide a different approach to network congestion control. Their premise is that the the queuing delays are becoming

smaller compared to propagation times due to the evolving technology [18] and so there would be no need for priorities within the network.

The principle of this approach, named *Proportionally Fair Pricing* (PFP), is that the users have a complete freedom in sending their packets but the network supplies feedback to users in form of prices reflecting the congestion costs. This simple congestion pricing scheme is able to provide differential quality of service [23] (services defined by the users) and the advantages of the smart market in a simple and robust fashion in any packet network. It is not bounded to any particular transmission protocol and the users may behave as they wish knowing that they will be charged accordingly by the network.

If using a packet marking technology similar to ECN, PFP is compatible with the TCP and RED. Next we shall look briefly into the mathematical model behind the proposal.

2.2.1 Mathematical framework

Next presentation of the underlying theory is based on the paper of Kelly et al. [21]. The discussion here will merely outline the theory and thus emphasise the important concepts rather than proofs.

The model

Assume a set of resources ($\mathcal{J}$) and routes ($\mathcal{R}$) indexed by j and r, respectively. Each route, a subset of the resources $r \subset \mathcal{J}$, can be seen as a user who has a transmission rate x_r and each resource has the capacity c_j. Denote

$$\mathbf{A} = \{a_{jr}\} = \begin{cases} 1 & \text{if } j \in r \\ 0 & \text{otherwise,} \end{cases} \tag{2.1}$$

so that the flow through the resource j can be written as

$$y_j = \sum_r a_{jr} x_r. \tag{2.2}$$

Let $\mathbf{x} = (x_r, r \in \mathcal{R})$ be a vector of rates and $\mathbf{c} = (c_j, j \in \mathcal{J})$. $\mathbf{x}$ is said to be *feasible* if $\mathbf{x} \geq 0$ and fulfills capacity limitations in the network, i.e.

$$\mathbf{Ax} \leq \mathbf{c}. \tag{2.3}$$

Each user r has a utility denoted by U_r depending on the rate x_r. Suppose that the functions $U_r(x_r)$ are concave, continuously differentiable, with $U_r'(x_r) \to \infty$ as $x_r \downarrow 0$ and $U_r'(x_r) \to 0$ as $x_r \uparrow \infty$. This means that when the prices depend linearly on the rate, there is an unique utility maximum. Adaptive traffic having this kind of utility is called *elastic* traffic following Shenker [39].

Before going into the details of the actual problem we need to define some important concepts related to fairness.

Definitions of fairness

The common fairness definition, much discussed by philosophers, is the *max min fairness*.

It is defined as follows: $\mathbf{x} = (x_r, r \in \mathcal{R})$ is *max min fair* if it is feasible and for each $r \in \mathcal{R}$ x_r cannot be feasibly increased without decreasing some other x_{r^*} which is smaller or equal to x_r.

Whereas max min fairness is seen to provide a fair resource allocation in context of political sciences, when talking about bandwidth sharing it gives an absolute priority to the smaller flows. The problem is, a decrease, no matter how small, in a smaller flow cannot reimburse an increase, no matter how large, in a larger flow. Such an extreme situation may occur in e.g. case of multiple bottlenecks in the network and therefore max min fairness may not be the best or the most effective alternative here. This is not a flaw of the definition but merely an intentional choice to define what is fair. An alternative definition, suggested by Kelly [17], weights the small flows somewhat less and is thus a more convenient criterion for the problem of bandwidth sharing.

The *proportional fairness* criterion is defined as follows: $\mathbf{x} = (x_r, r \in \mathcal{R})$ is proportionally fair if it is feasible and for any other feasible $\mathbf{x}^*$ the aggregate

of proportional changes is zero of negative:

$$\sum_{r\in\mathcal{R}} \frac{x_r^* - x_r}{x_r} \leq 0. \tag{2.4}$$

Further, let $\mathbf{w} = (w_r, r \in \mathcal{R})$ be a vector of weights and define *weighted proportional fairness* as: $\mathbf{x} = (x_r, r \in \mathcal{R})$ is *weighted proportionally fair* if for any other feasible $\mathbf{x}^*$,

$$\sum_{r\in\mathcal{R}} w_r \frac{x_r^* - x_r}{x_r} \leq 0. \tag{2.5}$$

If we use the interpretation that a weight is an amount to pay per unit time this definition means that the allocation is proportionally fair *per unit charge*. This means that the resource is divided among the users depending roughly on how much they are willing to pay.

Three optimization problems

PFP aims at maximising the aggregate utility, which we can write as an optimization problem SYSTEM($U, \mathbf{A}, \mathbf{c}$):

$$\begin{aligned} \max_{\mathbf{x}} \quad & \sum_{r\in\mathcal{R}} U_r(x_r) \\ & \mathbf{A}\mathbf{x} \leq \mathbf{c} \\ & \mathbf{x} \geq 0. \end{aligned}$$

As the users' utilities are generally not known to the network we consider the users and the network separately. Each user determines his preferences from the network by choosing an amount to pay per unit time, a parameter called *willingness-to-pay*, w_r. In return each user receives a flow $x_r = w_r/\lambda_r$, where λ_r can be seen as the cost per unit flow and time on the route r. Now the USER($U_r; \lambda_r$) becomes

$$\begin{aligned} \max_{w_r} \quad & U_r\left(\frac{w_r}{\lambda_r}\right) - w_r \\ & w_r \geq 0. \end{aligned}$$

On the other hand the network attempts to share its resources fairly to the users. We can assume that the users' preferences, the vector $\mathbf{w} = (w_r, r \in \mathcal{R})$, is known to the network and we can select the NETWORK($\mathbf{A}, \mathbf{c}; \mathbf{w}$) problem

as

$$\begin{aligned} \max_{\mathbf{x}} \quad & \sum_{r\in\mathcal{R}} w_r \log x_r \\ & \mathbf{A}\mathbf{x} \leq \mathbf{c} \\ & \mathbf{x} \geq 0. \end{aligned}$$

Now that the problems have been presented, it is necessary to motivate the choices made so far. It straightforward to verify that a vector $\mathbf{x}$ solves the problem NETWORK$(\mathbf{A}, \mathbf{c}; \mathbf{w})$ if and only if the rates are proportionally fair per unit charge. Consider deviating $\mathbf{x}$ so that $x_r^* = x_r + \delta x_r$ with all $r \in \mathcal{R}$. The corresponding increase in the objective function of the network problem is

$$\begin{aligned} & \sum_{r\in\mathcal{R}} w_r \left(\log\left(x_r + \delta x_r\right) - \log x_r\right) \\ = & \sum_{r\in\mathcal{R}} w_r \left(\log\left(1 + \frac{\delta x_r}{x_r}\right)\right) \\ = & \sum_{r\in\mathcal{R}} w_r \frac{\delta x_r}{x_r} + o(\delta\mathbf{x}) \\ = & \sum_{r\in\mathcal{R}} w_r \frac{x_r^* - x_r}{x_r} + o(\delta\mathbf{x}) \end{aligned} \tag{2.6}$$

Due to the convexity of the feasible region and the strict concavity of the objective function this increase is always zero or negative at maximum. This is actually an equivalent definition to the proportional fairness (2.5) and thus the reason for selecting the logarithmic objective function.

Solution of the problems

Here we shall sketch the solution for the optimization problems described above by using standard tools of constrained nonlinear optimization. References to the methods used here can be found, e.g. in [4].

First we take the users' position. Solution of the USER$(U_r; \lambda_r)$ problem

$$\frac{\partial}{\partial w_r}\left[U_r(\frac{w_r}{\lambda_r}) - w_r\right] = \frac{1}{\lambda_r} U_r'\left(\frac{w_r}{\lambda_r}\right) - 1, \tag{2.7}$$

suggests that, in order to maximise their net utility, the users should select the w_r (and thus the rate) where the derivative of their utility equals to the sum of the shadow prices along the route, that is

$$U_r'\left(\frac{w_r}{\lambda_r}\right) = \lambda_r. \tag{2.8}$$

Assume then that the network now shares the resource to users using its fairness criteria based on the w_r. The Lagrangian for the NETWORK$(\mathbf{A}, \mathbf{c}; \mathbf{w})$ is

$$L_{\text{network}}(\mathbf{x}, \mathbf{z}; \boldsymbol{\mu}) = \sum_{r \in \mathcal{R}} w_r \log x_r + \boldsymbol{\mu}^T(\mathbf{c} - \mathbf{A}\mathbf{x} - \mathbf{z}), \tag{2.9}$$

where the $\mathbf{z} \geq 0$ is a vector of slack variables and $\boldsymbol{\mu}$ is a vector of Lagrange multipliers $\boldsymbol{\mu} = (\mu_j, j \in \mathcal{J})$ associated with the capacity limits of each resource $j \in \mathcal{J}$. The Lagrange multipliers have an interpretation of *shadow prices* of the resources or *implied costs* per unit flow at the resources [17]. Now the solution to the network problem can be found by derivation of the Lagrangian (2.9)

$$\frac{\partial L_{\text{network}}}{\partial x_r} = \frac{w_r}{x_r} - \sum_{j \in r} \mu_j, \tag{2.10}$$

and so the unique optimum is

$$x_r = \frac{w_r}{\sum_{j \in r} \mu_j}. \tag{2.11}$$

If the prices are right, namely if the cost along the route r is given by

$$\lambda_r = \sum_{j \in r} \mu_j, \tag{2.12}$$

and the users are acting to maximise their utility doing the selection 2.8, the solution of the NETWORK$(\mathbf{A}, \mathbf{c}; \mathbf{w})$ equals to the solution of the whole the system. The Lagrangian of the SYSTEM$(U, \mathbf{A}, \mathbf{c})$ is

$$L_{\text{system}}(\mathbf{x}, \mathbf{z}; \boldsymbol{\mu}) = \sum_{r \in \mathcal{R}} U_r(x_r) + \boldsymbol{\mu}^T(\mathbf{c} - \mathbf{A}\mathbf{x} - \mathbf{z}), \tag{2.13}$$

and so by derivation

$$\frac{\partial L_{\text{system}}}{\partial x_r} = U_r'(x_r) - \lambda_r. \tag{2.14}$$

In summary, the system problem (or its dual problem) may be solved by de-

composing it to separate network and user problems which are solved simultaneously. Formally, we may write

Problem decomposition: There exist vectors $\boldsymbol{\lambda}$, $\mathbf{w}$ and $\mathbf{x}$ so that $w_r = \lambda_r x_r$ for all $r \in \mathcal{R}$ and w_r solves the corresponding USER$(U_r; \lambda_r)$ while $\mathbf{x}$ solves the NETWORK$(\mathbf{A}, \mathbf{c}; \mathbf{w})$ and the SYSTEM$(U, \mathbf{A}, \mathbf{c})$.

The model presented above is in a sense quite general and tells nothing about the actual solution mechanism in a dynamic network environment. Next we shall look into one way to solve the optimization problems dynamically.

An implementation

In practice, solution of the model above in a network environment would essentially require conveying information on prices to users and correspondingly weights w_r to the resources. The network problem could be processed in such centralised fashion but a more simple and robust approach would be decentralised as the delays and failures would be problems of individual end-nodes rather than of the system. That is, the computational effort is placed on the users themselves, while the network does only the pricing by generating congestion signals. Users may try to control, e.g. the rate of cost they have to pay, i.e. w_r, by dynamically adjusting their rate.

The strategy is then to design user-algorithms to solve the network problem implementing proportional fairness. One possibility would then be to use following strategy for controlling the sending rate

$$\frac{d}{dt}x_r(t) = \kappa\left(w_r(t) - x_r(t)\sum_{j\in r}\mu_j(t)\right), \tag{2.15}$$

where κ is a parameter controlling the rate of convergence. In this model the resource j sends feedback signals at rate $y_j\ \mu_j(t) = y_j\ p_j(y_j(t))$ ($p_j(\cdot)$ is a load dependent marking function) of which user r receives the proportion x_r/y_j. Now equation 2.15 describes the user's behaviour: it increases rate linearly proportionally to w_r and multiplicatively decreases it at rate proportional to the received feedback signal flow [11]. It can be shown [21] that the system of differential equations of type (2.15) has a stable point (2.11) by noting that

the expression

$$\mathcal{U}(\mathbf{x}) = \sum_{r \in \mathcal{R}} w_r \log x_r - \sum_{j \in \mathcal{J}} \int_0^{y_j} p_j(\xi) d\xi, \tag{2.16}$$

provides a Lyapunov function for the differential equation (2.15).

The algorithm controls the rate attempting to equalize the aggregate cost of a flow with a target value w_r. $p_j(y_j)$ can be seen as the cost per unit flow at the resources. It is important to note that the algorithm is only *approximative*; the functions $p_j(\cdot)$ can be chosen that (2.16) is arbitrarily close to NETWORK$(\mathbf{A}, \mathbf{c}; \mathbf{w})$. In fact, we are actually dealing with relaxations of the original constrained optimization problem. This point of view will be explained and motivated next.

Suppose that under a heavy load the network incurs some utility cost, in terms of delay or loss. In this sense it can be interpreted that the algorithm is penalising the proximity to the capacity constraint for each resource j, i.e. costs are incurred at the rate

$$C_j(y_j) = \int_0^{y_j} p_j(\xi) d\xi. \tag{2.17}$$

We call the value of the function

$$p_j(y_j) = \frac{d}{dy_j} C_j(y_j), \tag{2.18}$$

the *shadow price* of the resource j. Note that if

$$p_j(y_j) = \begin{cases} \infty & y_j > c_j \\ 0 & y_j \leq c_j \end{cases}, \tag{2.19}$$

the distributed algorithm becomes the problem NETWORK$(\mathbf{A}, \mathbf{c}; \mathbf{w})$. This, however, cannot be applied directly as the stability is compromised by high values of $p_j'(y_j)$.

What is done here is actually the relaxation of SYSTEM$(U, \mathbf{A}, \mathbf{c})$ (and thus also the network problem), which can be seen as seeking the *total net utility* (given that the costs and utilities are additive, this assumption will be subjected to discussion later)

$$\max_{\mathbf{x}} \sum_{r \in \mathcal{R}} U_r(x_r) - \sum_{j \in \mathcal{J}} C_j(y_j). \tag{2.20}$$

In optimization terms, the constraints of the problem have been replaced by a penalty function which depends on the load. It can be shown that under mild regularity conditions on the functions $p_j(y_j)$ ($p_j(\cdot)$ nonnegative, continuous and smoothly increasing function) the problem decomposition still holds under the identification

$$\lambda_r = \sum_{j \in r} p_j(y_j). \tag{2.21}$$

2.2.2 Discussion

We have described the development of mathematical background of congestion pricing above following the works of Kelly et al. Based on a simple decomposition of an optimization problem, we showed that it is possible to share network resources *fairly* among the users even when the network is not explicitly aware of the users' utilities. Further, we showed that this optimization process can be implemented by decentralised simple rate control algorithms. The focus of this work is to study the network part of this framework and how the network should set prices on the flows. Before that, some discussion on the model is required.

The original decomposition required the knowledge on the Lagrange multipliers, the shadow prices. In the relaxed dynamic solution model it is not possible to include these as defined without compromising stability and so they are replaced with utility-additive penalty functions. Although on the flow level it should be relatively easy to find penalty functions which, in theory, force the problem to converge to the feasible optimum, in the spirit of congestion pricing the cost should be relative to the utility of the information that could not be carried. *The original model does not tell what happens if the capacity constraints are temporarily exceeded* before reaching stability after each change in the flow. Probably some data will then be lost, but which flows suffer losses and which do not? What is the value of lost data?

It makes no sense to define the value of lost data using the users' utility functions. For example, imagine a compressed multicast real time video stream which incurs some packet loss due to congestion. If the stream cannot be decompressed the value of lost data is closer to the value of the whole stream than just of the loss rate. We would actually need another set of utility func-

tions describing each user's preferences on the quality of service. The model does not take this into account and we are forced to make the assumption that all such packet losses are equal in utility. This can be motivated so that the cost function is actually the cost to the *network* or to the whole system and not to individual users who care only about their allowed rate.

We selected the cost to the network, the social cost of congestion, be the rate of lost packets in a resource in this presentation. The selection is arbitrary but quite natural; each lost packet has to be typically retransmitted. Similarly we could follow, for example, delays instead of lost packets.

Now that the cost is explicitly defined, the implementation of PFP turns to the world the Internet. The congestion signalling is implemented by marking packets in RED/ECN-style and in this sense we can see the algorithm (2.15) as a variant of TCP, see, e.g. [19, 24] for comparison. The marking decisions and thus the shadow prices are calculated on the *packet level*, which will be thoroughly discussed in the next chapter.

The marking on the packet level is well motivated not only by the compatibility with the existing standards but also by the fact that the unavoidable averaging (estimating flow or prices) becomes a problem of the end-nodes rather than of the network. This seems desirable under the very diverse round trip times in the Internet environment.

We have not addressed here the behaviour of a large network implementing this scheme. Stability, convergence, random effects, time-lags and such are essential in the framework though and have received a lot of attention. The interested reader should see e.g. [21, 42]. An important observation is that in the presence of different round trip times (RTT) the equilibrium is not changed but the stability may be compromised by high values of $p_j'(y_j)$. The lower the buffer level where the marking occurs, the lower the chance of oscillatory behaviour of the solution [18].

Finally, suppose there is a non-adaptive user, against the assumptions put on the utilities above, who will not react to the cost signals received. The (pre-emptive) congestion control has to be done then before the user is admitted to the network. The PFP framework provides a method for this situation using distributed admission control [20]. When a call arrives, a number of probe

packets are transmitted along the preferred route and the call is accepted only if none of the probe packets are marked or lost.

Areas for further research are mainly new user policies to solve the network problem (see e.g. [22]) and how the shadow prices should be implemented. This is one of the key issues of this presentation. In the next chapter we shall look into the principles of conveying the price information, but before that a brief introduction to other congestion pricing proposals is presented.

2.3 Other related schemes

Various other approaches parallel to PFP have been suggested for implementing the congestion control of the future Internet. We shall next give a brief overview on the important ones based on pricing.

Smart market by McKie-Mason and Varian [28], was one of the first of the kind. It is an auction where the data with highest bids are carried at the market-clearing price (first rejected), always lower than all the admitted bids. Users can decide how much they are willing to pay for their data to be carried. If the prices determined by the network are right, the social optimum is found: Users get their fair share and from the network point of view the resource utilization is optimum. Although the elegance of this approach is alluring, it is rather impractical to implement from the technical perspective: It is unrealistic to assume that users would bid on packet-by-packet basis in the fast moving Internet. Furthermore, major new investments in router hardware should be made and the stability of such auction would be very hard to predict. Hence it is unlikely that the approach ever makes it into reality.

The Optimization Flow Control (OFC) approach described by Low and Lapsley, ([27], [26]), is a close relative to PFP. Basically the same maximum aggregate source utility is solved but in somewhat different manner: network calculates a price vector from the rates, information which is then conveyed to the users which decide their next transmission rate. This distributed optimization algorithm causes the price vector to converge to a proportionally fair allocation of resources. The main difference to PFP is that in OFC the users decide their rates and pay what the network charges whereas in PFP the users

decide their payments.

Paris Metro Pricing (PMP) by Odlyzko [34] is the simplest differentiated services solution where pricing is used to control traffic. It is based on the former pricing used in the Paris Metro system. The cars were divided into 1st and 2nd class cars which differed from each other only in price; 1st class was twice as expensive as the 2nd class. This way only the passengers who did not want to experience congestion (wanted to get a seat, avoid crowd or noisy teenagers etc.) selected the expensive 1st class which was much less congested. In similar fashion PMP provides two (or more likely three or four) similar (logical) subnetworks without any technical differences. As in the paragon, the improved quality in it is achieved only by the natural behaviour of the self-interested users. Users are assumed to select the route capable of meeting their requirements at lowest possible cost. Setting the prices and capacities for each class is a difficult problem. Also some research suggest that PMP would have difficulties to survive under competitive market situation [10] as it would not emerge naturally in such environment.

Chapter 3

Marking

3.1 Basics of marking

As discussed in the previous chapter, the end-nodes are informed on the congestion costs by congestion signals sent by the network. All the complexity and calculations are offloaded onto users and so the core of the network is kept simple to increase robustness. However, there are still two important functions the network have to perform: determining the right congestion information and conveying it to end-nodes.

The users can be made aware of the congestion prices several different ways. A classic approach would be the use of a separate (logically or physically) signalling network, but in the Internet context it is more convenient either to send separate "price packets" or more practically to piggyback the congestion information onto the transferred packets themselves when they pass through the congested resources. To this end, each packet should contain a data field in the network layer protocol header or trailer for this information.

Writing or updating the congestion data field in a packet is called *marking* following the convention from ECN. In the simplest form, marking means indeed setting a single bit in the network layer protocol header, and, to keep things simple, we will refer to this simple form of marking in the following sections. That is, if a packet is marked its congestion bit is set to 1 and otherwise it will be 0. Here the congestion price will be associated with the marking probability. Later it will be discussed whether this is enough for

implementing the PFP framework.

As the premise of the PFP framework was that the queuing delays are becoming small, we may omit the detriments caused by the delays and define the congestion cost to be solely the amount of information lost due to congestion. When packets traverse the network, each resource marks packets according to the local congestion costs so that the information at the receiving end-node corresponds the sum of the shadow prices along the route of the packet. Next example from [11] will motivate the marking mechanisms.

3.1.1 Sample Path Shadow Prices

A simple slotted-time model

Assume a slotted time system where N packets are handled in a time slot and a number of users, indexed by r, with Poisson distributed independent loads in each slot with means x_r. The aggregate load Y at the resource is then also Poisson distributed with the mean $y = \sum x_r$. Expected number of lost packet per a slot (cost) is given by

$$C(y) = \mathsf{E}[Y - N]^+ = \sum_{n \geq N} (n - N) e^{-y} \frac{y^n}{n!}, \tag{3.1}$$

and thus the shadow price

$$p(y) = C'(y) = \sum_{n \geq N} e^{-y} \frac{y^n}{n!}. \tag{3.2}$$

If an overflow occurs in a time slot, all the Y arrived packets are marked, see Figure 3.1. If the resource occupation is n, the user r has a binomially distributed number of packets in the system, $(X_r | Y = n) \sim \mathrm{Bin}(n, x_r/y)$. Thus, the expected number of received marks (or lost packets, both are treated

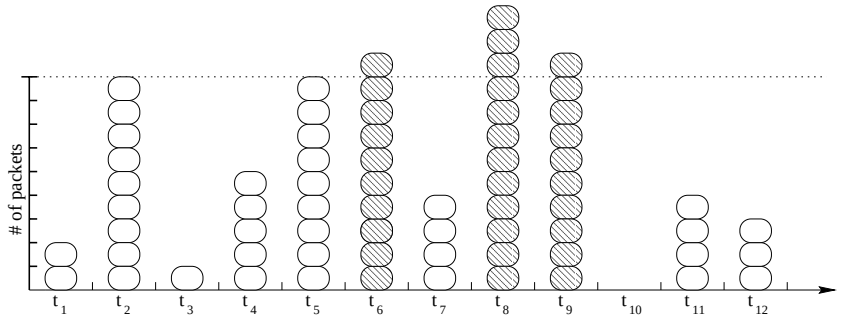

Figure 3.1: SPSP marking in slotted time

equally) by that user per unit time is

$$\begin{aligned}
\mathsf{E}[X_r|Y > N] &= \sum_{n>N} \mathsf{P}(Y=n)\mathsf{E}[X_r|Y=n] \\
&= \sum_{n>N} n\frac{x_r}{y}e^{-y}\frac{y^n}{n!} \\
&= \sum_{n>N} x_r e^{-y}\frac{y^{(n-1)}}{(n-1)!} \\
&= x_r \sum_{n\geq N} e^{-y}\frac{y^n}{n!} \\
&= x_r p(y). \qquad (3.3)
\end{aligned}$$

Interpretation is straightforward; for Poisson statistics, marking every packet when the resource is overloaded gives precisely the correct price information.

It is discussed in [11] that the relationship between the expected increase in system cost caused by a load increment, and the expected charge to that increment is more profound; it does not require any distributional assumption on the increment. This leads to the natural definition of the sample path shadow price.

Sample path shadow price (SPSP) of a packet is one if deleting it causes one less packet drop at the resource.

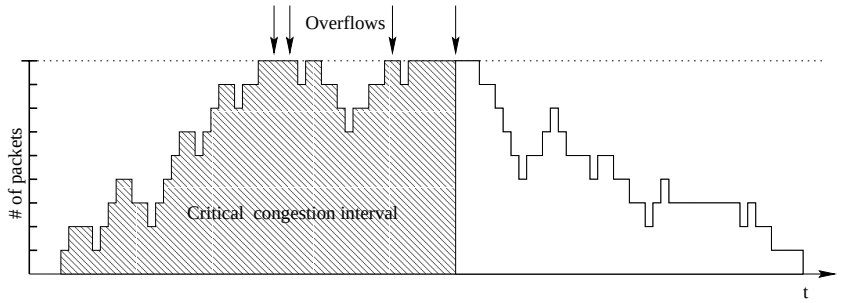

Figure 3.2: SPSP in continuous time: Critical congestion interval

SPSP in continuous time

More realistic models of the Internet resources include finite buffers. To be able to define SPSP in this environment let us recall some definitions on queuing models. *Busy period* is the time between an arrival to an empty system and the first departure that leaves the system empty. *Critical congestion interval* is the period between the start of the busy period until the last packet loss. Generalising the SPSP to continuous time queuing models we should ideally mark all the packets arriving during the critical congestion interval. See Figure 3.2 for illustration. However, there is a problem with the queuing model; some of the packets arriving during the interval may already have left the system before any congestion is detected. It is generally *impossible* to say whether the resource first overflows or empties when the packet is in the queue. Despite this drawback, several marking algorithms have been suggested in the literature. Some of them are presented in the next section.

The ideal marking algorithm can be implemented also in a parallel way giving exactly the same information on average and providing a convenient early warning of congestion. Instead of concentrating on marking exactly the right packets with the price one we give *each* packet a price which (locally) is a real number between $[0, 1]$. (In the single bit marking scheme this can be implemented by associating the price with the marking probability and on the user's end the price is estimated from the flow of marks.). We shall show in the next chapter that this price is essentially related to predicting overflows during the ongoing busy period. Therefore refer to this as *predictive marking*.

3.2 Different marking schemes

Several marking schemes have been discussed in the literature, mainly in the context of ECN-style marking. In this section some different approaches are presented and discussed. However, our aim is not to describe the methods in detail but merely to emphasise important design aspects related to approximating the SPSP scheme in buffers.

3.2.1 Mark after loss

Mark after loss, proposed in [11], is based on the observation that it seems to be enough to mark a correct number of packets on average instead of selecting just the ones arriving during the critical congestion interval. In this approach the resource keeps track on the number of packets arrived since the start of the current busy period. In case of an overflow, the resource starts to mark packets until to correct number of marks are placed.

Alternatively, the resource could mark packets from the overflow until the buffer becomes empty. Based on the argument on correct number of packets, this makes sense if the queue size were a reversible stochastic process and the busy periods were to contain overflows as the distribution of packets arriving before the last packet drop would be the same as of the packets leaving after the first drop.

By definition these schemes are obviously not fair as "they close the stable doors after the horse has bolted, and then blame the horses left inside for running away!" as colourfully described by Wischik [44]. Further, if the feedback delays are relatively short or the aim is to reach low packet drop ratio, the packets have to be market somewhat earlier in the busy period so that the losses could be prevented.

3.2.2 Threshold

The simplest possible predictive marking method is the use of a marking threshold – if the buffer occupancy exceeds a certain predefined limit the arriving packet receives the congestion mark. The problem with this method

is obviously that it makes no difference whether the occupancy lies relatively close to the threshold or if the buffer is almost empty. So even if there was only a small amount of mark-free buffer space left, users behave essentially in same way as when the buffer is empty. Naturally this leads to large bursts in traffic above the threshold, which now emphasises the main tradeoff made in setting the limit: If the threshold is high compared to the physical capacity of the resource we are likely to have bursts of lost packets and the early warning feature (and the whole idea of congestion control scheme) is lost and stability will also be jeopardised. On the other hand if the threshold is set low we will lose in the efficiency as users are actually aiming at using the resource up to the threshold instead of the resource capacity. This method, however, could be considered if delays play an important role in the system compared to the actual transmission capacity. That is, the limit is set to keep the queuing delays short rather than preventing packet loss.

3.2.3 Virtual Queue

The Virtual Queue, first proposed in [11] was designed as an alternative way to anticipate congestion. As the name implies this approach is based on the idea that instead of tracking the actual buffer, a separate virtual queue is maintained at the resource. It has exactly the same arrival process as the real queue but the service rate (and maybe the capacity) is scaled down by a factor κ. This means that a virtual overflow will happen before the real buffer becomes congested.

The marking is performed by setting the congestion bit in the packets which would receive mark using, e.g. the mark after loss or the threshold principle in the fictitious queue. In fact, the users are then notified about the shadow prices of the virtual system but by setting the scaling factor appropriately this should give the right information.

For instance, using an M/M/1 queuing model with the traffic intensity ρ for the actual buffer, the rate of packets arriving into a full system (K or more packets present) is $\rho p_K(\rho) = \rho^{K+1}$ and the shadow prices are given by

$$\frac{d}{d\rho}\rho p_K(\rho) = (K+1)\rho^K. \tag{3.4}$$

Suppose that the virtual system serves at a fraction of κ of the real buffer. If marking occurs when the arriving packet finds K or more packets from the virtual queue which happens at the probability $p_K(\rho/\kappa)$. Equating this with the shadow prices (3.4) Kelly et. al. [20] proposed a factor for the service rate:

$$\kappa = (K+1)^{-1/K}. \tag{3.5}$$

When compared to the mark after loss scheme, virtual queue reacts faster to threatening congestion. Still, it tends to blame the innocent, as the overflow mechanism in the virtual queue is essentially different from that of the real buffer. This feature is emphasised with small values of κ. Furthermore, the same tradeoff between utilization and the threshold must be made if the threshold marking scheme is used in the virtual queue.

3.2.4 RED

Random Early Detection (RED) by Floyd and Jacobsson [8] was developed as an independent buffer management algorithm for ECN and TCP/IP, but there is no reason why it could not be applied to provide correct information for SPSP scheme. The exponentially weighted moving average of the queue size is maintained and then compared to two thresholds. No packets are marked below the lower threshold and every packet receives a mark (or is dropped in case of non-cooperative sources) while above the higher one. Between the limits a linear function of the average queue size is used to give the marking probability which is then weighted with a coefficient depending on the packet count since the last marked packet.

The essential parameters here are the weighting used to determine the average queue size and the location of the lower threshold. Although these parameters could be set to approximate the SPSP, decision based on average queue size is generally too slow to react to random fluctuations and the method is thus prone to zigzag-effect as all the users are given incentives to react similarly at the same time.

3.2.5 REM

Random Exponential Marking by Athuraliya et. al [3] is a variant of RED. Although originally developed for the flow control scheme discussed in Section 2.3, it can be applied as a full queue management scheme [2]. A price information is updated explicitly at the resource and the marking probability depends exponentially on the price. The motivation for this form is that the end-to-end marking probability becomes exponentially increasing in the sum of the link prices along the path. In effect, if the resource has a workload W present at the packet arrival, the packet is marked with the probability $1 - \phi^{qW}$ with some parameter ϕ.

Chapter 4

Predictive Marking

4.1 Mathematical modelling

The whole theoretical framework of congestion pricing leans on the assumption that a network is able to provide the correct congestion price information to its end-nodes. As the exact prices for all the traffic are impossible to determine, as noted before, the pragmatic marking methods discussed in the previous section were mainly concerned about providing an early warning of congestion. This work, conversely, takes an essentially different approach to Sample Path Shadow Prices through mathematical modelling, aiming at to give the right price to each packet *on average*.

By dealing with these idealizations, however, we face the obvious controversy. Whilst mathematical models are able to provide tractable, exact and unarguable answers within their domain, the domains themselves are impossible to perfectly fit to cover the dynamic reality to be modelled. In this context we are essentially modelling the traffic going through a network resource. By modelling we simplify the reality, force it to follow some predefined laws, which may at best give some kind of approximation or capture some dominant feature from the real traffic. Generally there is a trade-off between the models which describe the reality accurately and the models which can give answers to the preset questions. In other words, the model's complexity can invalidate its usability.

Each model is based on a set of assumptions. When the assumptions are valid,

the obtained results are valid given that they are interpreted correctly. This gives to each model a certain limited range of validity where it can be applied. In this section our intention is, on the one hand, to discuss and motivate certain models and, on the other hand, to provide solutions for them. As decisions of marking have to be fast and robust, the simplicity of the results a model offers plays a significant part in the selection of the model.

The most natural selection for the model of an Internet router is a finite queue operating in continuous time. Although the Sample Path Shadow Prices are impossible to calculate exactly in such environment, with stochastic queuing models it is indeed possible in the sense of expectations. Our premise is that this itself may be enough for the implementation of congestion pricing scheme on the flow level. We are pricing the *risk* of overflow. Moreover, by deploying available information extensively the models can get very close to imitating the ideal SPSP-marking scheme also in packet-per-packet basis.

We shall start the journey, however, by a simple example shedding light on the SPSP scheme in continuous time. The profound observation is that the Sample Path Shadow Prices can be determined by calculating the *probability of overflow* during the busy-period when the packet has arrived.

4.1.1 Simple M/M/1/K queuing model

Consider an M/M/1/K queue model for a network resource. Packets arrive independently according to a Poisson process with the parameter λ and the service times are exponentially distributed with the parameter μ. Denote the traffic intensity $\rho = \lambda/\mu$. The queue works with the first-in-first-out (FIFO) discipline with one packet served (transmitted) at a time. If an arriving packet finds the system with K packets it will be discarded. From the user point of view we shall make no difference between whether the packet was discarded or merely just marked.

First, for purposes of determining the probability of overflow from a given state before the end of the busy period an embedded Markov chain, so called jump chain, is constructed. The states of this chain are the queue occupancy after any change (arrival or departure) in the queue. It has the transition

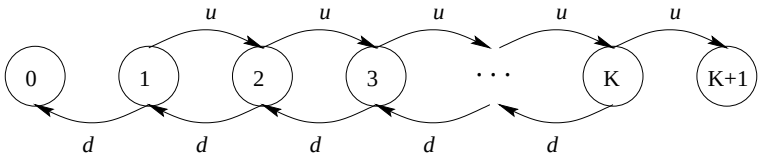

Figure 4.1: Embedded Markov chain of the M/M/1/K-queue.

probabilities

$$u = \frac{\lambda}{\lambda + \mu} = \frac{\rho}{\rho + 1}, \tag{4.1}$$

$$d = \frac{\mu}{\lambda + \mu} = \frac{1}{\rho + 1}, \tag{4.2}$$

upwards and downwards, respectively, and absorbing states at 0 and $K + 1$, see Figure 4.1. This method is often referred to as the first step analysis [5].

Calculation of the overflow probability is now straightforward. Regardless of the state from where the chain starts, the absorption will happen *almost surely*, that is with probability 1. This corresponds to that the buffer becomes empty or overflows in the original model. The interesting event here is the absorption to the state $K+1$ when starting from a given state of the system. Denote the probability of absorption to the state $K + 1$ from the state n by p_n. Due to the Markovian property of the chain, it holds that

$$\begin{aligned} &p_n = u\ p_{n+1} + d\ p_{n-1} \quad n \in \{1, \ldots, K\}, \\ &p_0 = 0,\ p_{K+1} = 1. \end{aligned} \tag{4.3}$$

This linear homogenous difference equation can be solved by using standard methods [38]. First the characteristic equation is formed by setting $p_n = r^n$ and dividing by r^n, which leads to

$$ur^2 - r + d = 0. \tag{4.4}$$

The characteristic equation is of quadratic form and thus has the roots

$$r_1 = \frac{1 + \sqrt{1 - 4ud}}{2u}, \tag{4.5}$$

$$r_2 = \frac{1 - \sqrt{1 - 4ud}}{2u}. \tag{4.6}$$

Due to the linearity, the solution of (4.3) is of the form

$$p_n = Ar_1 + Br_2, \tag{4.7}$$

for some constants A and B which can be determined using the boundary conditions, that is

$$\begin{cases} p_0 = A + B = 0, \\ \\ p_{K+1} = Ar_1^{K+1} + Br_2^{K+1} = 1. \end{cases} \tag{4.8}$$

The solution of the problem then becomes

$$p_n = \frac{r_1^n - r_2^n}{r_1^{K+1} - r_2^{K+1}}, \tag{4.9}$$

which can be simplified by using the identities (4.1) and (4.2), leading to the simple form given by

$$p_n = \rho^{K+1-n}\frac{(\rho^n - 1)}{(\rho^{K+1} - 1)}. \tag{4.10}$$

Next we shall generalise the result from the slotted time model of Gibbens and Kelly discussed in section 3.1.1. Consider an M/M/1/K system described above with a number of users, indexed by r, with a Poisson arrival process with with parameters x_r. The aggregate arrival process to the resource is then also Poisson with mean $y = \sum x_r$. The service rates are exponentially distributed with the mean service time $1/\mu$. Denote the traffic intensity of user r by $\rho_r = x_r/\mu$ and the total intensity $\rho = y/\mu$. The steady state probability of state i is

$$\pi_i = \rho^i \frac{1-\rho}{1-\rho^{K+1}}. \tag{4.11}$$

Expected rate of loss of data, the congestion cost, is given by arrivals to the system in the state K, i.e. at the bit rate

$$C(\rho) = \rho \mathrm{P}(\text{system full}) = \rho\rho^K \frac{1-\rho}{1-\rho^{K+1}}, \tag{4.12}$$

and thus the shadow price is obtained by derivation

$$p(\rho) = C'(\rho) = \frac{\rho^K(\rho^{K+2} - (K+2)\rho + K + 1)}{(\rho^{K+1} - 1)^2}. \tag{4.13}$$

Assume that upon arrival to the system, a packet is given the price determined by the equation (4.10) when *after* the arrival the system is in state n. That is, we *predict* whether the packet will be marked or not. Assume also that the system is at equilibrium and so the expected rate of marked data to the user r is

$$\begin{aligned}
\rho_r \sum_{i=0}^{K} \pi_i p_{i+1} &= \rho_r \sum_{i=0}^{K} \rho^i \frac{1-\rho}{1-\rho^{K+1}} \rho^{K+1-(i+1)} \frac{(\rho^{i+1}-1)}{(\rho^{K+1}-1)} \\
&= \rho_r \rho^K \frac{\rho-1}{(\rho^{K+1}-1)^2} \sum_{i=0}^{K} \left(\rho^{i+1}-1\right) \\
&= \rho_r \rho^K \frac{\rho-1}{(\rho^{K+1}-1)^2} \left(\rho \frac{\rho^{K+1}-1}{\rho-1} - 1 - K\right) \\
&= \rho_r \frac{\rho^K(\rho^{K+2}-(K+2)\rho+K+1)}{(\rho^{K+1}-1)^2} \\
&= \rho_r p(\rho). \qquad (4.14)
\end{aligned}$$

This means that by marking each arriving packet with the price (or marking the ECN-bit in the packet with the price-probability) (4.10) gives the desired information within this Markovian model. This scheme could be called *predictive marking*.

It should be noted, however, that although the expected value agrees with the proportion of marked data, the price is now more equally divided among the packets. In fact, there are almost no free of charge transmissions and the full price is also a rarity. This is in clear contradiction with the principles of congestion pricing in which essentially congestion free traffic should cost nothing. The phenomenon cannot be completely avoided as the uncertainty caused by the unknown future is always present.

However, this probabilistic marking scheme is not only an approximation of SPSP, it is also an alternative formulation of shadow prices within a mathematical model and can be used to implement PFP as it is. If each resource is able to provide an estimate of the overflow probability, the user receives the estimate of the overflow probability along its route. It has the advantage of being implementable but the drawback of enjoining the resource to do some averaging in order to estimate model parameters. Instead of pricing the congestion itself we price the *risk* of congestion. Here the prices can rise before any packet is lost and if some users are fast enough they may even be able

to react before any physical harm is done. Hence, it could be expected that this would contribute to more stable behaviour than in case of pure congestion pricing where the users observe essentially on-off type congestion.

We could even take one step further. It would be even more appealing to convey the information on the relative occupancy only. In that case the *users* have to estimate the traffic behaviour in the resources along their route. This could be perhaps implemented by using some known exponential form of marking probability. The difference is that now the prices are not constant for each mark but they are merely extracted from the flow of marks knowing exactly how the marking is implemented in the network. The approach could be called *utilization pricing*; prices depend on the buffer occupancies only. This topic, however, is left for further study. It should be noted though that then the users would be using the same kind of results we derive in this thesis for the network resources.

From now on we shall mostly refer to approximating the SPSP scheme. It has no impact on the calculations presented if we instead talked about the risk pricing as it is actually only another perspective to the same matter. SPSP provides also a suitable benchmark for the more limited mathematical models.

Before going into the details of marking, we present two alternative approaches in determining the shadow prices of an M/M/1/K model.

4.1.2 Alternative approaches

While the approach described above gave a simple result, it is useful to examine other ways of determining the shadow prices both for getting a better insight into the matter and for obtaining facultative methods for computation for more complex systems.

Based on the argumentation on the equality of overflow probability and shadow price we shall bring forward another way of calculating this probability by inspecting the net flow through the states in equilibrium. After that we deduce the shadow prices by a heuristic argument originating from the context of Markovian decision processes (MDP). In this case the discussion on the theory of these processes is omitted and the emphasis is on the heuristics.

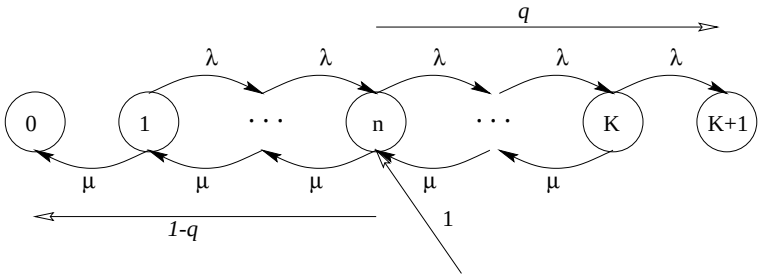

Figure 4.2: Flows from the state n.

Net flow

Consider the same absorbing Markov chain discussed above. What is the probability of absorption from the state n?

Assume that a constant flow of 1 is brought into the state n and the time has passed so that the system can be considered to be in equilibrium. Now we start examining the net flow q and $1-q$, upwards and downwards, respectively. In the steady state there is a flow of q from the state n towards the absorbing state $K+1$ and correspondingly a flow of $1-q$ towards state 0, see Figure 4.2. Each system state has the transition intensities λ and μ upwards and downwards, respectively. Denote the probability of finding the system in the state i by p_i. Now we can write the balance equations

$$\begin{cases} \lambda \cdot p_{i-1} - p_i \cdot \mu &= q & i > n, \\ p_i \cdot \mu - \lambda \cdot p_{i-1} &= 1-q & i < n, \end{cases}$$

with the boundary conditions

$$\begin{cases} p_0 = 0, \\ p_{K+1} = 0, \\ p_n^- = p_n^+. \end{cases} \tag{4.16}$$

The last condition is included to emphasise the fact that the value p_n fixes the value of q when solving the equations starting from both ends at the same time.

Solution is straightforward. We start at the state $p_{K+1} = 0$ and obtain $p_K =$

q/λ. Generally, when we have come down to state $K-i$ for some i we have

$$p_{K-i} = \frac{q}{\lambda}\sum_{j=0}^{i}\frac{1}{\rho^j} = \frac{q}{\lambda}\frac{\rho^{-i-1}-1}{\rho^{-1}-1}. \tag{4.17}$$

That is, for state n

$$p_n = \frac{q}{\lambda}\frac{\rho^{n-K-1}-1}{\rho^{-1}-1}. \tag{4.18}$$

Correspondingly, starting from state 0 we have

$$p_n = \frac{1-q}{\mu}\sum_{j=0}^{n-1}\rho^j = \frac{1-q}{\mu}\frac{\rho^n-1}{\rho-1}. \tag{4.19}$$

By setting both expressions equal at n, we are able to solve q, the net flow over the states towards overflow:

$$\begin{aligned}
\frac{1-q}{\mu}\frac{\rho^n-1}{\rho-1} &= \frac{q}{\lambda}\frac{\rho^{n-K-1}-1}{\rho^{-1}-1} \quad \| \cdot\mu \\
(1-q)(1-\rho^n) &= q(\rho^{n-K-1}-1) \\
q(\rho^n-\rho^{n-K-1}) &= \rho^{n-1} \\
q &= \rho^{K+1-n}\frac{(\rho^n-1)}{(\rho^{K+1}-1)}.
\end{aligned} \tag{4.20}$$

This is, as it should be, the same as obtained before in (4.10).

Relative costs

An elegant heuristic approach to the problem is based on the Markovian decision processes. Consider a packet arriving into the M/M/1/K system with n packets present and the load $\rho = \lambda/\mu$. In principle we are allowed to admit the packet into the system or reject it. The shadow price will rise from the comparison of these decision alternatives as it will be motivated below.

If the packet is accepted, the system will be at state $n+1$ after the arrival. Shadow prices were defined to be the cost increment for a marginal increase in load. Hence the interesting property is *increase in cost*, i.e. the increase in the expected number of blocked packets in the future, due to the arrived packet.

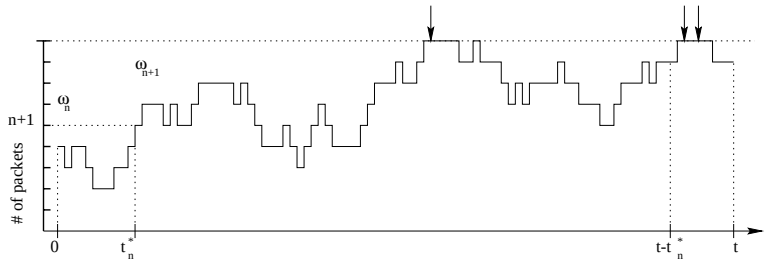

Figure 4.3: The effect of the starting state on overflows.

(In MDP terms, we calculate the difference of the *relative costs* of the states n and $n+1$, but it is not necessary to go further into the MDP terminology here).

The cost of a sample path representing the queue occupancy at a time t is the number of blocked packets before t. What is the expected increase in cost when starting from the state $n+1$ instead of n? Consider two arbitrary paths, ω_n starting from the state n, and ω_{n+1} starting from state $n+1$.

Due to the Markovian property of the process, from the point ω_n reaches the state $n+1$ for the first time (denote the time this happens by t_n^*) it is statistically identical to ω_{n+1}. Naturally, there cannot be any overflows on ω_n before t_n^* as the process have pass through $n+1$ in order to reach higher states and so also the blocking state. In other words, at any given time $t > t_n^*$, ω_n is stochastically equal to what ω_{n+1} was the time t_n^* ago and so one can expect an equal number of overflows during $(0,t)$ on ω_n and $(0,t-t_n^*)$ on ω_{n+1}. Thus, the increase in cost at the time t, the expected difference in numbers of overflows on the paths, is the expected number of lost packets on the ω_{n+1} between $(t-t_n^*,t)$. This is illustrated in Figure 4.3.

When $t \to \infty$, effects of the initial value vanish, the system is at equilibrium and the probability that the system is full (and all the arriving packets are blocked) is given by

$$\mathrm{Ovf}(K,\rho) = \frac{\rho^K}{\sum_{i=0}^{K}\rho^i} = \rho^K \frac{(1-\rho)}{(1-\rho^{K+1})}. \tag{4.21}$$

On average $\lambda \mathsf{E}[t_n^*]$ packets arrive during the time $(t-t_n^*,t)$ and hence the

increase in expected number of lost packets becomes

$$p_n = \lambda \mathsf{E}[t_n^*]\ \mathrm{Ovf}(K, \rho). \tag{4.22}$$

$\mathsf{E}[t_n^*]$ can be determined by observing an M/M/1/n system. Immediately after a blocked packet the system is full, i.e. at the state n. Next time when blocking happens corresponds the transition from state n into $n+1$. Hence, t_n^* is distributed as the time between subsequent overflow events in an M/M/1/n system (arrivals into full system) and the expectation

$$\mathsf{E}[t_n^*] = \frac{1}{\lambda \mathrm{Ovf}(n, \rho)} \tag{4.23}$$

Now we can state our result: the increase in cost – the shadow price – is given by

$$p_n = \frac{\mathrm{Ovf}(K, \rho)}{\mathrm{Ovf}(n, \rho)}, \tag{4.24}$$

which is exactly the same as (4.10) noting that n represents here the state *before* the packet is accepted into the system while the price (4.10) is calculated *after* the packet has been placed into the queue.

We have shown that the congestion prices arise from the overflow probability. In practice, however, the actual implementation of marking is limited by the technology. In the next section this question will be briefly addressed.

4.2 On marking mechanisms

While the "single bit in the header at arrival"-marking is usually taken as an obvious choice following the ECN-technology, there is a wide variety of options and possibilities to explore. The essence of predictive marking framework is predicting the future so that it could be predicted whether a certain packet contributes to congestion cost or not. Although the exact knowledge is not available at the time of decision, information is needed for the best possible prediction. The marking mechanism is the key to this information. In this section we shall outline the effects of the choices in the technical implementation of the marking.

4.2.1 Is one bit enough?

ECN-type marking comprises of setting a single congestion bit in the packet. Naturally it is advantageous to use as little space as possible for this purpose; in order to be suited to small packet size, the overhead cannot be large in comparison to the payload. Furthermore, this is indeed enough for implementation of the idealistic SPSP. However, if we are using predictive marking and if the packet size may be large it is necessary to estimate the correct price information faster and more bits are necessary. On the flow level the shadow price is estimated from the *rate* of incoming marks. Therefore, further bits would require less packets to get the right information and it would take less time for the user to react as there is decreased need to do averaging at the end-node. (Obviously some averaging is required for the single bit scheme, otherwise the end-node behaviour would be of the on-off style).

Moreover, the shadow prices are cumulative along the route. A single bit is enough to provide correct information only if the congestion is reasonably rare and so it is highly unlikely to have two or more marks on the same packet. Thus, it may be necessary to expand the price above the maximum of one. On the other hand, this may be necessary if the packet sizes are very different or if the users can change their packet size. Large packets could easily cause more losses than just one.

4.2.2 Time of marking

The time of marking is the most essential thing to choose and it provides many alternatives with differing motivations. It is closely related to the location of the ECN-bit in a data packet as after that part of the packet is transmitted it is impossible to change the congestion information. If the transmission times of the packets are long compared to inter-arrival times it is essential to mark as late as possible. For example, the overflow may already have happened during the time packet resides in the system or the packet leaves the system almost full when the overflow is highly likely to happen.

The common mark at arrival is not the most informative as during the waiting time we are likely to get crucial information on overflow. However, this

method would still provide some kind of steady state information of the buffer occupancy (cf. Poisson Arrivals See Time Averages property). Here it does not matter where in the packet the actual congestion data field is located.

Another alternative is to mark when the transmission of the packet is started. This corresponds roughly marking the header when it is send to lower protocol layers for transmission and thus is easy to implement. The main motivation for this is, however, that this moment is the last possibility of marking the header.

The third approach has a pure mathematical motivation. If the marking information of all the packets in the system is updated at each arrival, the final mark depends on the situation at the last arrival during the time packet is waiting or being transmitted. This way the stochastic properties of the system at the arrivals could be preserved and it might ease the prediction computation. However, the implementation of this system may be complicated.

The fourth, and in some sense optimal option is to mark the packet just after its departure, roughly in the trailer. This way the most current information becomes available and again the observation moment preserves some stochastic properties.

There is still one approach, even more difficult to implement, but which would provide almost the correct information, given that the round-trip times are long compared to busy periods. Assuming that each packet is confirmed by acknowledgment, the information on the shadow price of the original packet could be piggy-packed on the acknowledgment traversing back towards the source. This would require that the same route is used on the way back and huge tables of users and their traffic are maintained at the resource. Therefore, it seems quite remote in the light of current Internet technology, and we will not pursue this option further.

Other limitations on the available information reside in the software and hardware in routers. Many predictive methods require local parameter estimates and the accuracy of the estimates may be crucial. Further, the current state (or even history) of a queue is essential information anyway. Whether it is given by the number of bits or number of packets or even both, has an effect on the prediction process. Similarly the capacity of the buffer can be expressed in

packets or in raw memory and it may not be in same units than the occupancy. This may increase the complexity further.

Predictive marking can be used to implement the PFP scheme as it is and there the time of marking does not play a significant role (given that the model is right), but if we are to mimic the ideal SPSP scheme it does. On the other hand, all the available information should be used as it increases robustness of in case of inadequate models. Thus, we propose the following rules.

4.2.3 Predictive marking for approximating SPSP

The examples in the previous section showed that the absorption probability and SPSP agrees in the sense of expectations. In practice, however, there are only paths that lead to overflow and paths that do not. Thus fairness is compromised if the price is shared equally among the packets arriving at the same state. To improve the situation while approximating SPSP we propose the following principles in marking:

1. If an overflow occurs all the packets in the buffer are marked.

2. Non-marked packets are marked as late as possible before completely leaving the system, at the start of transmission (header) or preferably at the end of it (trailer). The price or the marking probability is given by the overflow probability during the busy period.

The first point ensures that those who are guilty will be charged and so we are left with only those that are on the verge of escaping the system. If we want to provide any early warning on oncoming congestion, these packets must carry a price reflecting the congestion situation. By waiting until the very last moment before marking we are able to convey the information on the most current situation. Calculations related to the overflow probability do not change, given that the Markov property holds for the model. The expected rate of marked bits remains the same; now the marking happens essentially with the probability related to the system state *without* the actual packet, but we are more likely to mark the right packets. This is the best we can do to get close to SPSP.

The function of the mathematical models is then to capture only those unmarked who are able to leave the system before the current busy period ends. This increases the robustness of the system as the significance of the model is thus reduced. The next section will concentrate on the calculation of the overflow probabilities in various common queuing models.

4.3 General queuing models with Markovian properties

4.3.1 Models with embedded Markov chain

Embedded Markov chain can be constructed to all single server queuing models with Poisson arrival process. This, however, leads to more tedious calculations with potentially very large matrices. The idea is again to form a chain from the system states *after the departures* so that all the overflow and system-empty states act as absorbing states. The absorption probability corresponds the shadow price of the state which is marked to the packet as if the packet had just left the system.

Absorption probabilities

First we recall some results from the theory of Markov chains, for a more detailed account see, e.g. [15]. Consider a discrete time Markov chain X_t, $t = \{0, 1, \ldots\}$. The transition probability matrix is denoted by

$$\mathbf{P} = \{p_{ij}\}, \tag{4.25}$$

where the elements are the transition probabilities $p_{ij} = \mathrm{P}(X_{t+1} = j | X_t = i)$. If we have s transient states and r absorbing states we can partition the matrix $\mathbf{P}$ as

$$\mathbf{P} = \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix},$$

where the $s \times s$ matrix $\mathbf{Q}$ contains the mutual transition probabilities of the transient states, and the $s \times r$ matrix $\mathbf{R}$ has the one step transition probabilities

into the absorbing states. This is called the canonical form of $\mathbf{P}$. The n-step transition probabilities of the chain can be obtained by

$$\mathbf{P}^n = \begin{pmatrix} \mathbf{Q}^n & \mathbf{?} \\ \mathbf{0} & \mathbf{I} \end{pmatrix},$$

where $\mathbf{?}$ stands for the matrix containing n-step transition probabilities into the absorbing states, which has no simple presentation at this point. $\mathbf{Q}^n \to \mathbf{0}$ when $n \to \infty$. This means that the absorption will happen almost surely, with probability 1.

Let now b_{ij} be the probability of absorption into an absorbing state j when starting from the transient state i, and $\mathbf{B} = \{b_{ij}\}$. Naturally, letting $\mathcal{K}$ be the set of transient states, for each absorbing state j

$$b_{ij} = p_{ij} + \sum_{k \in \mathcal{K}} p_{ik} b_{kj}. \tag{4.26}$$

By writing this in matrix form using the canonical form we get

$$\mathbf{B} = \mathbf{R} + \mathbf{QB}, \tag{4.27}$$

which leads to the solution

$$\mathbf{B} = (\mathbf{I} - \mathbf{Q})^{-1}\mathbf{R}. \tag{4.28}$$

Basically, if it is possible to construct a time homogenous absorbing Markov chain, the rest is straightforward calculation to determine the absorption probability. Next we will present the generation of the transition probability matrix for a few common models.

M/G/1/K

Consider a M/G/1/K model with the arrival intensity λ and the service time distribution defined by the distribution function $f_S(t)$. The system is observed just after the departures and the occupancy has the values $\{0, \ldots, K\}$. At the next observation the state of the chain is the current added with the arrivals during the service time minus the one departing just before the observation. If

there are no packets left at any point the chain has absorbed. The probability of i arrivals during a service time is thus essential in forming the transition probability matrix and is given by

$$t_i = \int_0^\infty \frac{(\lambda t)^i}{i!} e^{-\lambda t} f_S(t) dt. \tag{4.29}$$

Denote the probability of i or more arrivals in a service time by

$$t_\infty^i = \sum_{j=i}^{\infty} t_j = 1 - \sum_{j=0}^{i-1} t_j. \tag{4.30}$$

Let the indices $\{1, \ldots, K-1\}$ refer to the corresponding states, K to the overflow and $K+1$ to the absorbing zero state. Note that the *state* K is impossible to reach. If the system were in this position after a departure, there would have been at least $K+1$ packets present just before the departure and overflow would already have happened. Now we can write the transition probability matrix in the canonical form

$$\mathbf{P} = \begin{pmatrix} t_1 & t_2 & t_3 & \ldots & t_{K-1} & 0 & t_\infty^K & t_0 \\ t_0 & t_1 & t_2 & \ldots & t_{K-2} & 0 & t_\infty^{K-1} & 0 \\ 0 & t_0 & t_1 & \ldots & t_{K-3} & 0 & t_\infty^{K-2} & 0 \\ \vdots & \vdots & \vdots & \ddots & & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & t_0 & t_1 & t_\infty^2 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.31}$$

Using this matrix the absorption probabilities are easy to calculate with the method described above. The first column of $\mathbf{B}$ gives the probabilities of overflow from each starting state before the end of the busy period.

The corresponding results can also be derived for the G/M/1/K model where the transitions of the embedded chain are induced at the times of arrival of new packets.

A limited M/M/1 priority queue

Another example of applications of the embedded Markov chains is a limited M/M/1 pre-emptive queue with priority classes. This corresponds roughly to a single resource providing differentiated services. Now it should be noted that packets having a priority over others are not responsible only for the overflow among their priority class but also for all the classes with lower priority. This is a rather arbitrary selection as losses in a lower class would likely be less expensive (or completely free) for a higher class.

Consider a simple model with two priority classes 1 and 2 with pre-emptive FIFO queuing discipline and separate limited waiting queues. This means that for the class 1 traffic the queue works as a normal $\mathrm{M/M/1/K_1}$ queue, where there are room for $\mathrm{K_1} - 1$ packets to wait for the service. For class 2, however, the behaviour is more complicated as all the class-1 packets will be served before any of the class-2 packets is taken into service. Furthermore, a class-2 packet's service is interrupted by any arrival in the class 1 and there is a maximum limit of $\mathrm{K_2} - 1$ packets waiting in this class.

We can form a Markov chain consisting of the system occupancy after each change, arrival or departure, in the system. In Figure 4.4 class 1 is represented as the horizontal states and class 2 will be served only if the horizontal state is zero. The horizontal state $\mathrm{K_1} + 1$ and the vertical state $\mathrm{K_2} + 1$ are absorbing states.

Denote class-1 (horizontal) and class-2 (vertical) arrival probabilities when the *horizontal state* is i by a_i^1 and a_i^2, respectively. The corresponding departures are denoted by d_i^1 and d_i^2. If the arrival and the service rates are given by λ_1 and μ_1 for class 1, and λ_2 and μ_2 for class 2, we can write

$$a_i^1 = \begin{cases} \dfrac{\lambda_1}{\lambda_1 + \lambda_2 + \mu_2} & i = 0, \\ \dfrac{\lambda_1}{\lambda_1 + \lambda_2 + \mu_1} & 0 < i \leq \mathrm{K_1}, \end{cases} \tag{4.32}$$

$$d_i^1 = \dfrac{\mu_1}{\lambda_1 + \lambda_2 + \mu_1} \quad 0 < i \leq \mathrm{K_1}, \tag{4.33}$$

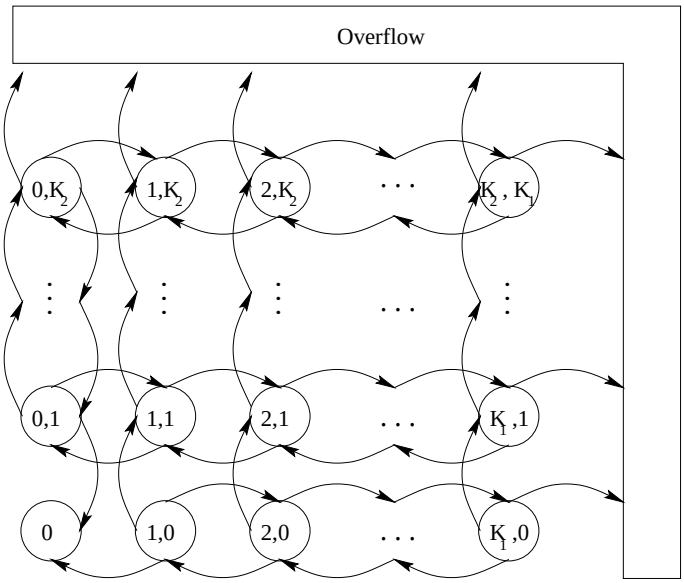

Figure 4.4: Markov chain of the priority model.

$$a_i^2 = \begin{cases} \dfrac{\lambda_2}{\lambda_1+\lambda_2+\mu_2} & i = 0, \\ \dfrac{\lambda_2}{\lambda_1+\lambda_2+\mu_1} & 0 < i \leq \mathrm{K}_1, \end{cases} \tag{4.34}$$

$$d_i^2 = \frac{\mu_2}{\lambda_1+\lambda_2+\mu_2} \quad i = 0. \tag{4.35}$$

For the transition probability matrix we get the block presentation

$$\mathbf{P} = \begin{pmatrix} Q_{0x} & Q_{1x} & \mathbf{0} & \dots & & & B_0 \\ Q_{2x} & Q_0 & Q_1 & \mathbf{0} & \dots & & B_1 \\ \mathbf{0} & Q_2 & Q_0 & Q_1 & \mathbf{0} & \dots & B_2 \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & & \ddots & & B_2 \\ & & & & & Q_0 & B_3 \\ \mathbf{0} & \dots & & & \dots & \mathbf{0} & I \end{pmatrix}, \tag{4.36}$$

where the Q_i-blocks are of size $K_1 \times K_1$, B_i-blocks of size $K_1 \times 2$, and I is the identity matrix of size 2×2. The notation Q_{ix} means a Q_i-block without the first row and column. Using the notation presented above the blocks are given

by

$$Q_0 = \begin{pmatrix} 0 & a_0^1 & 0 & \dots & 0 \\ d_1^1 & 0 & a_1^1 & & 0 \\ 0 & d_2^1 & 0 & & \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \dots & & & 0 \end{pmatrix}, \tag{4.37}$$

$$Q_1 = \begin{pmatrix} a_0^2 & 0 & 0 & \dots & 0 \\ 0 & a_1^2 & 0 & \dots & 0 \\ 0 & 0 & a_2^2 & & \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & \dots & & & 0 \end{pmatrix}, \tag{4.38}$$

$$Q_2 = \begin{pmatrix} d_0^2 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & & 0 \end{pmatrix}. \tag{4.39}$$

Transition probabilities into the absorbing states are given by

$$\begin{aligned} B_0 &= \begin{pmatrix} 0 & d_1^1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 0 & d_1^2 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix}, \\ B_2 &= \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ a_{K_1}^1 & 0 \end{pmatrix}, \quad B_3 = \begin{pmatrix} a_0^2 & 0 \\ a_1^2 & 0 \\ \vdots & \vdots \\ a_{K_1-1}^2 & 0 \\ a_{K_1}^2 + a_{K_1}^1 & 0 \end{pmatrix}. \end{aligned} \tag{4.40}$$

Now that we have a complete transition probability matrix, we are able to partition it and calculate the absorption probabilities again as presented in the beginning of this section.

Other possibilities

The two examples discussed above give an idea of the great versatility of the method. Further generalizations are available for instance into quasi birth-and-death processes; the interested reader is referred to the original work of Neuts [30].

Although providing simple and elegant solutions, simple queues with some Markovian properties have a fundamental problem. They are always bound to some particular traffic model and cannot be assumed to perform well if the traffic changes. In search of more robust methods we have to step outside the nice and neat queuing model mindset and explore some approximations having a theoretical ground for robustness.

4.3.2 Diffusion approximation

Poissonian arrival process may have been an appropriate approximation in early telephone networks, but it is highly questionable in modern Internet traffic. In this section we shall present a more plausible, although approximative, model for calculation of the overflow probabilities.

A typical router in the Internet may have a buffer of size of thousands of packets and hundreds of kilobytes in each output. Therefore, it has become excessive to model the buffer in terms of separate packets as the discontinuous leaps in the occupancy at an arrival or departure are small in comparison with the average state. We are dealing with relatively small changes, so small that the buffered data can be seen as a continuous flow of fluid entering a piping system. This model, called the fluid approximation, however, whilst providing nice results for the average values, fails to take into account the randomness and variability around the mean. These can be of significant size and of special interest in problems like the ones dealt in this presentation. If the random variability is allowed, one enters into the world of continuous stochastic processes. The approach is called the diffusion approximation.

Diffusion approximation is based on the fact that many independent and identically distributed sources generate essentially a Gaussian type of aggregate behaviour due to the central limit theorem (see e.g. [43]) of the probability

theory. The reason to use the continuous approximation is thus obvious; we have a theoretical support for the results regardless the source distributions which otherwise would have to be modelled. A more detailed motivation can be found e.g. in [25]. Rigorous derivations are outside the scope of this presentation, but we shall now outline the general mathematical framework before making any modelling decisions. We shall start with the definition of the diffusion processes.

A diffusion process is a continuous time parameter stochastic process which possesses the (strong) Markov property and for which the sample paths $X(t)$ are continuous (with probability 1) functions of t [16].

For any such process there are two commonly defined infinitesimal parameters:

$$\mu(x,t) = \lim_{h\to 0} \frac{\mathsf{E}[X(t+h) - X(t)|X(t) = x]}{h}, \tag{4.41}$$

$$\sigma^2(x,t) = \lim_{h\to 0} \frac{\mathsf{E}[\{X(t+h) - X(t)\}^2|X(t) = x]}{h}, \tag{4.42}$$

which are called the drift parameter and the infinitesimal variance, respectively. For our purposes it is necessary to consider only the time homogenous processes for which the infinitesimal parameters depend only on the state x and not on t.

Following the procedure used in the discrete load models, we now set the state 0 and the predefined overflow limit c as absorbing barriers. The problem is to find the probability of absorption into c when starting from the position $X(0) = x$, $0 < x < c$. Denote this probability by $u(x)$ and select a small time duration h, so small that the absorption probability is negligible. Naturally one can set $u(0) = 0$ and $u(c) = 1$. Further, let $\delta X = X(h) - x$ and one can write

$$\begin{aligned} u(x) &= \mathsf{E}[u(X(h))|X(0)] + o(h) \\ &= \mathsf{E}[u(x+\delta X)|X(0)] + o(h) \\ &= u(x) + \mathsf{E}[\delta X|X(0)]u'(x) + \frac{1}{2}\mathsf{E}[(\delta X)^2|X(0)]u''(x) + o(h) \\ &= u(x) + \mu(x)hu'(x) + \frac{1}{2}\sigma^2(x)hu''(x) + o(h). \end{aligned} \tag{4.43}$$

From this, by subtracting $u(x)$, dividing by h, and taking the limit when h

tends to zero, we get a differential equation for the desired probability.

$$0 = \mu(x)\frac{du}{dx} + \frac{1}{2}\sigma^2(x)\frac{d^2u}{dx^2}, \qquad 0 < x < c,\ u(0) = 0,\ u(c) = 1. \tag{4.44}$$

The equation (4.44) can be solved elegantly using the scale and speed measures as represented in [16]. Assume that $\sigma^2(x) > 0$, let

$$s(x) = \exp\left\{-\int^x \frac{2\mu(\eta)}{\sigma^2(\eta)}d\eta\right\}, \tag{4.45}$$

and define the scale function of the process as

$$S(x) = \int^x s(\tau)d\tau = \int^x \exp\left\{-\int^x \frac{2\mu(\eta)}{\sigma^2(\eta)}d\eta\right\}d\tau. \tag{4.46}$$

The speed density of the process is

$$m(x) = \frac{1}{\sigma^2(x)s(x)}. \tag{4.47}$$

Now we introduce $1/s(x)$ as an integrating factor and separate the variables in (4.44) which leads to

$$\frac{1}{2}\frac{1}{m(x)}\frac{d}{dx}\left(\frac{1}{s(x)}\frac{du}{dx}\right) = 0. \tag{4.48}$$

Writing the scale and speed measure in the differential form, $dS = s(x)dx$ and $dM = m(x)dx$, we get the canonical representation of the problem,

$$\frac{1}{2}\frac{d}{dM}\left(\frac{du}{dS}\right) = 0, \tag{4.49}$$

which can be solved by two successive integrations leading to

$$u(x) = D + CS(x) \qquad u(0) = 0,\ u(c) = 1. \tag{4.50}$$

Solving the integration coefficients we get the final result

$$u(x) = \frac{S(x) - S(0)}{S(c) - S(0)} \qquad 0 \le x \le 1. \tag{4.51}$$

The result given here can be easily applied to provide results for all time ho-

mogenous diffusion processes. For example for the standard Brownian motion the parameters are $\mu = 0$ and $\sigma^2 = 1$ which gives the scale function $S(x) = x$. So the probability of absorption into barrier at c from the state x becomes simply

$$u(x) = \frac{x}{c} \qquad 0 \leq x \leq 1. \tag{4.52}$$

The situation above, however, seldom corresponds a real buffer behaviour and it is necessary to discuss the modelling aspects of diffusion approximation. For a more realistic model we follow the example of Harrison and Patel [12], and shed light on a GI/GI/1 queuing model with independent inter-arrival time A and independent service time B with means $\mathsf{E}[A] = 1/a$ and $\mathsf{E}[B] = 1/b$, respectively. The corresponding variances are c_a^2/a^2 and c_b^2/b^2, where the c_a and c_b are the corresponding variation coefficients.

Now we can approximate the queue state by a continuous path process $X(t)$ with parameters which are actually constant in steady state. Using the notation above

$$\mu = \mu(x) = \lim_{t\to\infty} \mu(x,t) = a - b, \tag{4.53}$$

$$\sigma^2 = \sigma^2(x) = \lim_{t\to\infty} \sigma^2(x,t) = ac_a^2 + bc_b^2. \tag{4.54}$$

Thus, for a GI/GI/1 queue under the diffusion approximation we get

$$s(x) = \exp\left(-\frac{2\mu x}{\sigma^2}\right), \tag{4.55}$$

$$S(x) = C\exp\left(-\frac{2\mu x}{\sigma^2}\right) + D \qquad \text{(with constants } C \text{ and } D\text{)}. \tag{4.56}$$

and further the probability of overflow from state x becomes

$$u(x) = \frac{e^{-2\mu x/\sigma^2} - 1}{e^{-2\mu c/\sigma^2} - 1}, \tag{4.57}$$

where the barrier c is $K + 1$ in the queuing model.

Note that the diffusion approximation requires independent and identically distributed inter-arrival times and service times. Moreover the mutual independence of the arrival and departure processes is required, which is not exactly true on large scale; there cannot be more departures than arrivals at any time and so the departure process is limited by the arrivals. However, during a busy period the inter-departure times are just the service times and hence

independent from the arrival process. This is enough to justify the results obtained.

Further generalization of the mathematical models requires abandoning the Markovian property. That is, the traffic is assumed to depend on the history rather than just on the current state. As the mathematical properties become essentially more complicated there are only a few traffic models suggested where correlations are involved. One of the most prominent of these is the fractional Brownian motion which will be discussed next.

4.4 Fractional Brownian motion

Until now we have assumed that the traffic is independent of its past. This is a rather natural assumption if we are looking into short periods of time as in cases we have examined here. Assuming that the round trip times are significantly longer than busy periods, it seems logical that the arrival process has the Markovian property. However, various traffic measurements, starting from the Bellcore LAN measurements [9], have shown that network traffic is very bursty at all time-scales, a feature which the Markovian models cannot explain. The traces showed self-similar or fractal-like behaviour which can be modelled with long-range depended processes. On the other hand, due to the reasons mentioned in discussing the diffusion processes, Gaussian models are desirable as many independent sources result in essentially Gaussian process. Combining this with the long-range dependence we arrive at fractional Brownian motion (fBm).

Next we shall present the storage model with fBm input following the presentation of Norros [31]. A normalised fractional Brownian motion $Z(t)$ with the Hurst parameter H is a Gaussian process which has continuous paths, stationary increments, mean $\mathsf{E}[Z(t)] = 0$ and variance $\mathsf{E}[Z(t)^2] = |t|^{2H}$ for all t.

Using the Reich formula we describe the system occupancy or the amount of work in the buffer with the leaky bucket model

$$V(t) = \sup_{s \leq t} \left(A(t) - A(s) - C(t-s) \right), \tag{4.58}$$

where $A(t)$ is the amount of work arrived before t and $A(0) \equiv 0$. C stands for the service rate.

Now we can define the long-range dependent arrival process as

$$A(t) = \mu t + \sqrt{\sigma^2 \mu} Z(t). \tag{4.59}$$

It should be noted that although the parameters of the process (μ,σ^2 and H) can be chosen so that negative arrivals are unlikely, they cannot be fully avoided and thus the model is rather non-physical at small time scales.

Now the interesting question is, given the process path during the busy period, what is the conditional probability to reach overflow before the end of the busy period? Although this problem is well defined, it seems daunting indeed. Hence, we take an alternative approach and sketch an approximation to determine a lower bound for the probability.

Suppose that when a busy period starts, measurements on the work residing in the system are made at short constant intervals, δ. When a packet is leaving the system (the marking moment) at the time T after the start of the busy period the buffer will have an amount $V(T)$ of work left. That means that the busy period will last at least the time $V(T)/C$ from the time T on. This requires naturally the assumption that $A(t)$ is practically increasing with all $t \in (-\infty, \infty)$.

Now we can predict the distribution of values of corresponding normalised process (denote by $\mathbf{z_1}$) we would measure during this time (denote the set of the observation times by $T_{\mathbf{z_1}}$) on condition of all the previous measurements (denote by $\mathbf{z_2}$) during the interval $t \in [0, T)$. If an overflow is to happen during this time it will certainly be before the end of the current busy period. Evaluating the overflow probability in these discrete points gives a lower bound for the actual overflow probability. After the time $T+V(T)/C$ we will not have certainty whether the busy period continues and it will be computationally too heavy to calculate all possible conditional outcomes. Thus, we have the

following limit:

$$\mathrm{P}(V(t) \geq x|\ \mathbf{z_2},\ V(s) > 0\ \forall\ s < t) \tag{4.60}$$

$$\geq\ \mathrm{P}(V(t) \geq x|\ \mathbf{z_2},\ t < V(T)/C) \tag{4.61}$$

$$\geq\ \max_t \mathrm{P}(\mu t + \sqrt{\sigma^2 \mu} Z(t) - Ct \geq x|\ \mathbf{z_2}, t \in T_{\mathbf{z_1}}) \tag{4.62}$$

$$=\ \max_t \mathrm{P}\left(Z(t) \geq \frac{x + (c-\mu)t}{\sqrt{\sigma^2 \mu}} |\ \mathbf{z_2},\ t \in T_{\mathbf{z_1}} \right). \tag{4.63}$$

It makes sense to observe the process in these discrete steps since it simplifies the problem tremendously as well as defines clearly when the level is crossed. The time steps should be slightly shorter than the average packet service time. In order to compute the conditional values of $Z(t)$, we need to determine the covariances and conditional distributions in FBM context following [33].

First, the covariance of $Z(t)$ and $Z(s)$ is determined by the formula

$$\mathrm{Cov}[Z(t), Z(s)] = \Gamma(t,s) = \frac{1}{2}\left(t^{2H} + s^{2H} - |t-s|^{2H}\right). \tag{4.64}$$

Assume that a $k \times 1$ vector $\mathbf{z}$ consists of two parts (of length k_1 and k_2 respectively) $\mathbf{z} = [{\mathbf{z_1}}^T, {\mathbf{z_2}}^T]^T$ and we want to know the distribution of $\mathbf{z_1}$ when $\mathbf{z_2}$ is known. Let $\mathbf{\Gamma}$ be the corresponding covariance matrix $\mathrm{E}[\mathbf{z}\mathbf{z}^T]$ and $\mathbf{A} = \mathbf{\Gamma^{-1}}$. Let $\mathbf{A}$ be partitioned as follows

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix}, \tag{4.65}$$

where $\mathbf{A}_{11}$ is a square matrix of size $k_1 \times k_1$ and $\mathbf{A}_{22}$ a square matrix of size $k_2 \times k_2$ and $\mathbf{A}_{12} = \mathbf{A}_{21}^T$ is a rectangular $k_1 \times k_2$ matrix. Now the conditional distribution of $\mathbf{z_1}$ is Gaussian with the mean and variance:

$$\begin{aligned} \mathsf{E}[\mathbf{z_1}|\mathbf{z_2}] &= -{\mathbf{A}_{11}}^{-1}\mathbf{A}_{12}\mathbf{z_2}, \\ \mathsf{E}[\mathbf{z_1}{\mathbf{z_1}}^{\mathbf{T}}|\mathbf{z_2}] &= {\mathbf{A}_{11}}^{-1}. \end{aligned} \tag{4.66}$$

In summary, we are able to determine a rough lower bound for the conditional

overflow probability which can be expected to be more accurate when the buffer is quite full. However, the inaccuracy and complexity of the model and the need to estimate parameters such as the Hurst parameter make this method virtually useless without further improvements in the probability calculation. Such improvements could maybe be found using the path space approach and large deviations approximation along the lines of [32] but we leave this for further study at this point.

Chapter 5

Analysis

In this chapter we compare the mathematical methods presented previously and examine how closely they can approximate the ideal SPSP marking procedure. The emphasis is put on the robustness, the applicability and the behaviour of the methods themselves *within* mathematical models and the discussion on the actual model selection to describe traffic is mostly omitted. The reason for this is simply that the model selection is too large an issue to be handled here. The question whether some model is able to capture the relevant features of some particular stream has been one of the central research topics in traffic theory since the earliest stages of this branch of science.

5.1 Comparison of different methods

This section consists of comparisons between the marking probabilities obtained with different methods and a discussion on the differences. Differences can occur due to the used approximation, such as diffusion approximation, or from the fact that the actual traffic does not follow the used model. This should shed some light on the robustness of the models.

5.1.1 Effects of the process

First we shall study the simple M/M/1/K model. From the general form of the overflow probability we instantly have two results on the limit behaviour:

$$\lim_{\rho\to 0} p_n = \lim_{\rho\to 0} \frac{\rho^{K+1} - \rho^{K+1-n}}{\rho^{K+1} - 1} = \begin{cases} 0 & n < K+1, \\ 1 & n = K+1. \end{cases} \tag{5.1}$$

And correspondingly using the l'Hospital's rule

$$\lim_{\rho\to 1} p_n = \lim_{\rho\to 1} \frac{1-\rho^{-n}}{1-\rho^{-(K+1)}} = \lim_{\rho\to 1} \frac{n\rho^{1-n}}{(K+1)\rho^{-K}} = \frac{n}{K+1}. \tag{5.2}$$

That is, the marking probability/price behaves approximately as a barrier at $K+1$ when ρ is small and approaches a linear form under heavy traffic. Naturally it is possible to determine the prices for $\rho > 1$ and in this domain the marking probability is a concave function, approaching value 1 for all positive states when the intensity grows without bounds. However, in a properly functioning network, implementing congestion pricing these situations should be very rare as it is the goal of the whole scheme to avoid dropping packets. For more complicated models, the dependence is generally on both mean and variance, but what is significant, the *form of the marking probability remains roughly the same.*

Next we compare the M/M/1/K model with two other models: deterministic and uniformly distributed service times. Exact solutions to these models can be calculated using the jump chain analysis described in Section 4.3.1. First, assume that the arrival intensity is $\lambda = 3$ and the traffic intensity is $\rho = 0.75$ in all cases with the service times

Model	$\mathsf{E}[X]$	$\mathrm{Var}[X]$
Exp(4)	1/4	1/16
D=1/4	1/4	0
U(0,0.5)	1/4	1/48

For comparison, consider the same models at $\rho = 0.9$, $\lambda = 9$ with the service times

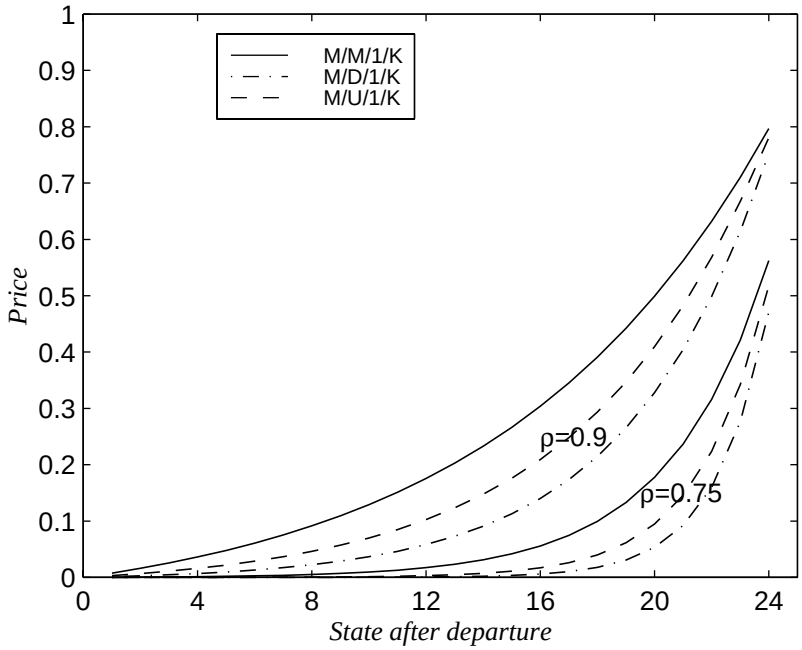

Figure 5.1: M/M/1/K, M/D/1/K and M/U(0,0.5)/1.

Model	$\mathsf{E}[X]$	$\mathrm{Var}[X]$
Exp(10)	1/10	1/100
D=1/10	1/10	0
U(0,0.2)	1/10	1/300

The exact marking probabilities for $K = 25$ are plotted in Figure 5.1. It is obvious that smaller variance results in smaller overflow probability and thus into lower prices. These examples were given here only to provide insight on the form of the marking function. The differences between the presented models are not that large, but still large enough so that we cannot play down the importance of model selection. Especially when the utilization is high but overflows relatively rare (i.e. the traffic has low variability), the model seems to play a very central role in the price determination. Hence, it will be too limiting to assume a certain model for the traffic without proper justification. The model should be very general with easily estimable parameters, such as the diffusion processes, which will be discussed next.

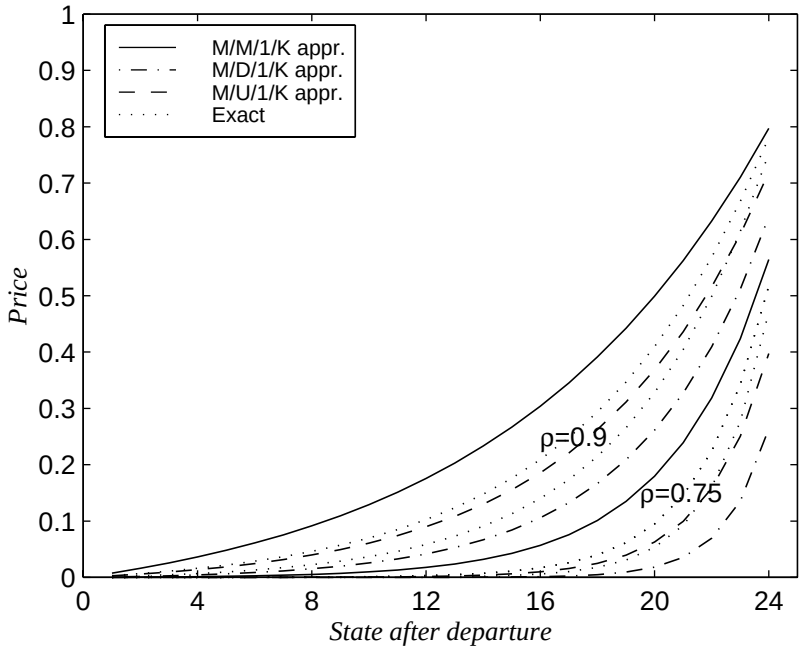

Figure 5.2: Diffusion approximation.

5.1.2 Diffusion approximation

Diffusion approximation has the advantage of being free of the traffic models. When there are many independent sources contributing to the load at a resource we feel confident that this model would perform well. However, this is just another assumption, the significance of which we intend to determine. This section attempts to study the approximation under essentially non-Gaussian environment using the Markovian models from the previous section. For the sake of robustness, the selected pricing scheme should behave well under all circumstances and a comparison with these models will help us to determine the applicability of the approximation in this context.

Should the behaviour be well approximated, this would be the most promising method to calculate the prices when the traffic poses little or no long-range dependence. Figure 5.2 shows the same models for which the exact prices were computed in Figure 5.1, only now computed using diffusion approximation (4.57) with the corresponding expectations and variances.

With Poissonian traffic (M/M/1/K) we see almost identical behaviour to the original one for large ρ and the two other models, which are by nature far from

Gaussian, are approximated surprisingly well. The slight underestimation of the price is a rather encouraging result but, unfortunately, it may not be accurate enough. It is important that the pricing works well for heavy traffic and high states as there the difference is largest and the loss prevention actually takes place. For light traffic there is less need to control it and thus the marking does not play so significant role. It should be noted also that under heavy traffic in a resource designed to serve a significant number of users, the heavy traffic means usually more independent users (rather than larger flows for single users) and thus more Gaussian traffic. This too seems encouraging, but we may not rely on this assumption.

The convergence to the diffusion approximation depends essentially on the skewness of the counting process $N(t)$. If the arrival and service processes are similar in form (which is the case in the M/M/1/K system when ρ grows) the convergence is fast and excellent results are obtained. Otherwise, the price will be over or under the estimated, depending on which tail of the counting process is larger. Naturally one cannot assume that the traffic would behave so that the model will work (even if it did) and we are forced to look for solutions elsewhere.

A profound observation is that the *form* of the marking probability can be very well approximated with the diffusion process absorption formula regardless of the model. Hence, we could include a single real valued correction parameter $\delta \in (-1, 2)$ to the calculations reflecting the bias caused by the skewness of the counting process distribution. That is, if the system is left to state n we calculate the price as if the system was in the state $n + \delta$. This minor adjustment would provide accurate approximations as can be seen from Figure 5.3. Estimation of this correction term is an interesting open question (one option would be to estimate the overflow probability from the state $K - 1$, set the result equal to $u(K - 1 + \delta)$, and solve the term from 4.57), in this case we had the exact answers available and used the least squares method for the models:

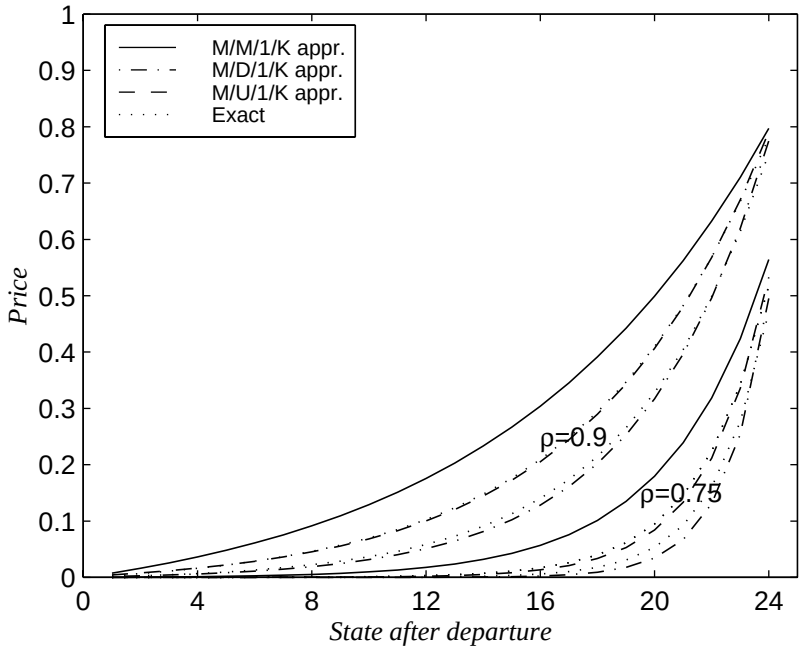

Figure 5.3: Adjusted diffusion approximation.

Model	λ	μ	δ
M/D/1/25	3	4	0.9461
M/D/1/25	9	10	0.8581
M/U(0,0.5)/1/25	3	4	0.6358
M/U(0,0.2)/1/25	9	10	0.5791

The diffusion parameters, expectation and variance are straightforward to estimate from the incoming traffic. The estimation procedure, however has its own pitfalls, basically dealing with the trade-off between accuracy and responsivity to change, but these implementation aspects are, although interesting, outside the more general scope of this thesis and will not be discussed further here.

5.2 Simulation experiment

5.2.1 Differences with SPSP

The experiment consists of simulating the buffer occupancy of an M/M/1/K resource, where the exact pricing formula (4.10) is used in parallel with the ideal SPSP scheme. We generated arrival and service times for a number of packets and served them in a queue according to the first-in-first-out principle. As a result we obtained traces of the queue occupancy where we marked the packets using the two schemes. In this simulation we were able, unlike in the reality, to mark all the packets during the critical congestion intervals in the SPSP scheme. In the approximative scheme, packets were marked using the rules from Section 4.2.3, that is all the packets in the buffer receive a mark at overflow and unmarked packets are marked according to the state they leave the system in (using the pricing formula (4.10)). This gives us a good idea how close it is possible to get to the ideal scheme by the approximation. In both schemes we did not distinguish the packets which are dropped from those which were merely marked.

Figure 5.4 shows the difference between calculated prices when $\rho = 0.9$ and $K = 25$. The statistics were obtained using 10000 packets with $\lambda = 9$ and $\mu = 10$.

The results show the distribution of the differences. The expectation of the distribution is practically zero, as it should be. Only a small portion (about 7% here) of SPSP-marked packets escape the system while most of them receive almost correct price (more 60 % equal to or no more than 0.1 larger than the correct price). We can conclude that *if* we are able to determine the risk of overflow correctly, we are able approximate the SPSP scheme very accurately.

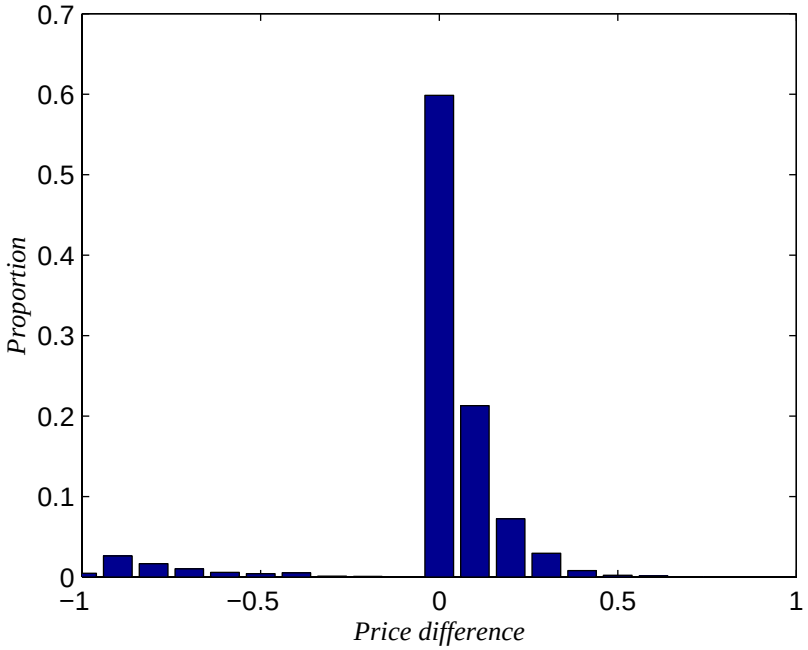

Figure 5.4: Approximative prices minus ideal prices.

Chapter 6

Conclusions

As the Internet is expanding at exponential rate we are facing constant difficulties with evolving demands and constraints. Development of new networks and protocols and the evolution of the existing ones are unavoidable. In this thesis we have discussed one promising approach for the model of future Internet.

Congestion pricing is able to provide an elegantly simple method for the network congestion control. Although it is rather controversial even on the conceptual level, mainly because of the difficulties related to fairness of pricing and distribution of wealth, it is one of the few methods capable of providing service differentiation and almost only one to address the issue of fair resource allocation in a communications network.

Proportionally Fair Pricing is an stylish adaptation of the congestion pricing concept into the current world of the Internet, combining the benefits of the economics point of view with the mechanisms already implemented in the present Internet technology. The network is able to share resources fairly without explicitly knowing the users' utilities. Congestion, an extrenality to users, causes a social cost that is divided among those responsible for it. This cost sharing is implemented through an ECN-type marking procedure.

Marking would be ideally performed by setting the congestion bit to one for all the packets arriving within the critical congestion interval, between an arrival to an empty system and the last overflow within the busy period. However, this marking scheme, known as Sample Path Shadow Pricing, cannot be implemented as packets may have escaped from the system before any overflow is

detected. Even in the ideal case SPSP is not perfect as it requires an overflow to happen and therefore probably causes bursts of marks to the users, whose reactions are likely to lead into oscillatory behaviour. A marking scheme with an early warning method would be better. This led us to look into alternative marking possibilities.

We showed that within a mathematical model for a resource it is possible to replace SPSP by marking packets probabilistically. A packet's price at a given time equals the probability of overflow from the state of the system at that time. In a more general model the probability depends also on the history of the current busy period. Naturally the best possible imitation of SPSP is achieved with the following rules. First, every packet is marked at the resource at the overflow and then all the unmarked packets are marked as they leave the system by using the probabilistic marking scheme. This scheme, called predictive marking, is able to provide the desired early warning of congestion. In addition to imitating SPSP it can be seen as a stand-alone marking method with the interpretation that the users are no longer paying only for the lost packets of other users, but for the *risk* of congestion.

For the most robust and implementable model we suggest the diffusion approximation. It has the theoretical support for uncorrelated traffic and its parameters are easily estimable.

The drawbacks of the predictive marking are obvious. How well does the assumed model describe the actual traffic? In this case we are especially concerned with the assumptions on the Gaussian and Markovian properties of the buffer occupancy. Although non-Gaussian nature of the process can, in many cases, be corrected simply with a small deviation in argument, the possible internal correlations of the traffic may cause some inaccuracy. Furthermore, the estimation of the parameters of the model is required. This leads to a tradeoff between responsiveness and accuracy as many of the estimators are based on calculating some kind of average value. If one takes more packets (and time) into consideration in the estimation process, one will get smaller variance for the estimator. However, if the underlying properties are subject to change, the expectation of the estimator becomes more biased at the same time. Although this method is far more reactive than any of the threshold methods based directly on the average queue size (e.g. RED), it may not be

enough. One should be able to answer the question what is the probability of an overflow *right now*, rather than on average in this kind of situation.

It should be noted, however, that the minor biases in the packet level marking are not that significant. On the flow level they will be compensated by the vast amount of packets and ultimately by the congestion events as only a portion of packets are marked using a model and the rest by congestion events. Under very light or very heavy traffic all the models also agree in prices.

In summary, we have found a rather general pricing method the parameters of which can be estimated from the traffic. The possibility of its direct application may still be slightly far-fetched, but it gives some important insight into how the pricing should work.

6.1 Further work

Further work should be related to the interface between mathematical models and reality. The concepts should be tested in an appropriate environment, but how can we ever be sure that there will not be any types of traffic that cause the model to choke? The most acute problem of the diffusion approximation is the estimation of parameters. Fast, simple and accurate estimation of mean, variance and possibly the correction term is essential. Are these parameters enough to implement a robust pricing? If not, would the fBm model perform well enough? Could the risk of overflow be measured empirically?

More fundamental issue is the averaging. When using the (impossible) ideal SPSP scheme we do not need to use any kind of averaging at the resource. However, as mentioned earlier it always requires a loss to occur before any marks are generated and it is thus not a suitable early warning of congestion. If any kind of predictive method is used to approximate SPSP, we face the averaging at the resource.

Consider then a single resource with a number of users with different round-trip times. If the packets are labeled corresponding to the buffer occupancy only, the users *themselves* can do the averaging to estimate the buffer behaviour from their point of view and act correspondingly. The prices could, for example, be generated by sending marks for each overflowing packet back to the user

or by extracting the price from the flow of marks. That is, marks and prices are separate or their relation is more complex than linear. Users could then predict the prices using the marks. This framework would further simplify the core network but naturally becomes complicated when there are more than one resource. The bottom line is that the marking should be very simple, fair and transparent so that all the complexity could be left to the end-users.

Finally it should be noted that in all these calculations we have implicitly assumed that the contribution of a single user is small (for the diffusion approximation) and that one user cannot change the behaviour of the flow significantly. This is a rather natural assumption for a large scale network but if, for some reason, this cannot be accepted, we face interesting game theoretical problems as the users anticipate the effects of their own behaviour to the price.

Bibliography

[1] Mark Allman. Collection of TCP/IP research papers. http://tcpsat.lerc.nasa.gov/tcpsat/papers.html.

[2] Sanjeewa Athuraliya, Victor H. Li, Steven H. Low, and Qinghe Yin. REM: active queue management. Submitted for publication, http://www.ee.mu.oz.au/staff/slow/research/projects.html, October 2000.

[3] Sanjeewa Athuraliya and Steven H. Low. Optimization flow control, II: Random exponential marking. Submitted for publication, http://www.ee.mu.oz.au/staff/slow/research/projects.html, May 2000.

[4] Mokhtar S. Bazaraa, Hanif D. Sherali, and C. M. Shetty. *Nonlinear programming: Theory and algorithms.* John Wiley and Sons, New York, 2nd edition, 1993.

[5] Pierre Brémaud. *Markov chains: Gibbs fields, Monte Carlo simulation and queues.* Springer-Verlag, New York, 1999.

[6] Sally Floyd. TCP and Explicit Congestion Notification. *ACM Computer Communication Review*, 24(5):10–23, 1994.

[7] Sally Floyd and Kevin Fall. Promoting the use of end-to-end congestion control. *IEEE/ACM Transactions on Networking*, 7(4):458–472, 1999.

[8] Sally Floyd and Van Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, August 1993.

[9] Henry J. Fowler and Will E. Leland. Local area network traffic characteristics, with implications for broadband network congestion manage-

ment. *IEEE Journal of Selected Areas in Communications*, 9(7):1139–1149, 1991.

[10] Richard J. Gibbens. Control and pricing for communication networks. Technical Report 1999-10, Statistical Laboratory, University of Cambridge, 1999. Available at http://www.statslab.cam.ac.uk/Reports/.

[11] Richard J. Gibbens and Frank P. Kelly. Resource pricing and the evolution of congestion control. *Automatica*, 35:1969–1985, 1999.

[12] Peter G. Harrison and Naresh M. Patel. *Performance Modelling of Communication Networks and Computer Architectures*. Addison-Wesley, Wokingham, 1992.

[13] Van Jacobson. Congestion avoidance and control. In *Proc. ACM SIGCOMM '88*, pages 314–329, August 1988.

[14] Van Jacobson. Modified TCP congestion avoidance algorithm. Available at ftp://ftp.ee.lbl.gov/email/vanj.90apr30.txt, April 1990.

[15] Samuel Karlin and Howard M. Taylor. *A first course in stochastic processes*. Academic Press, New York, 1975.

[16] Samuel Karlin and Howard M. Taylor. *A second course in stochastic processes*. Academic Press, San Diego, 1981.

[17] Frank P. Kelly. Charging and rate control for elastic traffic. http://www.statslab.cam.ac.uk/∼frank/elastic.html, 1997. Corrected version.

[18] Frank P. Kelly. Models for a self-managed Internet. *Philosophical Transactions of the Royal Society*, A358:2335–2348, 2000.

[19] Frank P. Kelly. Mathematical modelling of the Internet. In B. Engquist and W. Schmid, editors, *Mathematics Unlimited - 2001 and Beyond*, pages 685–702. Springer-Verlag, Berlin, 2001.

[20] Frank P. Kelly, Peter B. Key, and Stan Zachary. Distributed admission control. *IEEE Journal on Selected Areas in Communications*, 18(12):2617–2628, December 2000.

[21] Frank P. Kelly, Amam K. Maulloo, and David K.H. Tan. Rate control in communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49:237–252, 1998.

[22] Peter B. Key and Laurent Massoulièe. User policies in a network implementing congestion pricing. Workshop on Internet Service Quality Economics (MIT), Available at http://www.research.microsoft.com/users/pbk/, December 1999.

[23] Peter B. Key and Derek R. McAuley. Differential QoS and pricing in networks: where flow-control meets game theory. *IEE Proceedings Software*, 142(2):39–43, March 1999.

[24] Peter B. Key, Derek R. McAuley, Paul Barham, and Koenraad Lavens. Congestion pricing for congestion avoidance. Technical Report MSR-TR-99-15, Microsoft Research, February 1999.

[25] Leonard Kleinrock. *Queuing systems, volume II: Computer applications*. John Wiley and Sons, New York, 1976.

[26] David E. Lapsley and Steven H. Low. An optimization approach to ABR control. In *Proc. IEEE International Conference on Communications*, pages 523–527, June 1998.

[27] Steven H. Low and David E. Lapsley. Optimization flow control, I: Basic algorithm and convergence. *IEEE/ACM Transactions on Networking*, 7(6):861–874, December 1999.

[28] Jeffrey K. MacKie-Mason and Hal R. Varian. Pricing the Internet. In B. Kahin and J. Keller, editors, *Public Access to the Internet*. Prentice-Hall, New Jersey, 1994.

[29] Lee W. McKnight and Joseph P. Bailey, editors. *Internet Economics*. The MIT Press, Cambridge, 1997.

[30] Marcel F. Neuts. *Matrix-geometric Solutions in Stochastic Models*. The Johns Hopkins University Press, Baltimore, 1981.

[31] Ilkka Norros. A storage model with self-similar input. *Queueing Systems*, 16:387–396, 1994.

[32] Ilkka Norros. Busy periods of fractional brownian storage: a large deviations approach. *Advances in Performance Analysis*, 2(1):1–20, 1999.

[33] Ilkka Norros and Jorma Virtamo. Handook of FBM formulae. *COST257TD(96)*, 1996.

[34] Andrew M. Odlyzko. Paris Metro pricing for the Internet. In *Proc. ACM Conference on Electronic Commerce (EC'99)*, pages 140–147, 1999.

[35] K. K. Ramakrishnan and Sally Floyd. A proposal to add explicit congestion notification (ECN) to IP. *RFC 2481*, 1999.

[36] K. K. Ramakrishnan, Sally Floyd, and D. Black. The addition of explicit congestion notification (ECN) to IP. http://www.aciri.org/floyd/papers/draft-ietf-tsvwg-ecn-00.txt, 2000.

[37] K. K. Ramakrishnan and Raj Jain. A binary feedback scheme for congestion avoidance in computer networks. *ACM Transactions on Computer Systems*, 8(2):158–181, May 1990.

[38] Lennart Råde and Bertil Westergren. *Mathematics Handbook for science and engineering*. Studentlitteratur, Lund, 3rd edition, 1995.

[39] Scott Shenker. Fundamental design issues for the future Internet. *IEEE Journal on Selected Areas in Communications*, 13(7):1176–1188, 1995.

[40] William H. Stallings. *Data and computer communications*. Prentice Hall International, New Jersey, 1997.

[41] William H. Stallings. *High-speed networks : TCP/IP and ATM design principles*. Prentice Hall International, Upper Saddle River, 1998.

[42] David K.H. Tan. Rate control and user behaviour in communication networks. 4th INFORMS Telecommunications conference, Boca Raton, Florida. Available at http://http://www.statslab.cam.ac.uk/∼dkht2/conf.ps, March 1998.

[43] David Williams. *Probability with Martingales*. Cambridge University Press, 1991.

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