

University of London  
Imperial College of Science, Technology and Medicine  
Department of Computing

# Quantitative Models for the Design of Cellular Networks

Sérgio Coury

*August 1993*

Thesis submitted for the degree of *Doctor of Philosophy* of the University of London,  
and for the Diploma of the Imperial College of Science Technology and Medicine  
(D.I.C.).



## Abstract

Cellular Networks represent the extension of the packet radio technology into a global environment. They are a result of a natural evolution of packet radio networks, enabling them to become an integrated part of a wider packet exchange network. Their topology extends over a large geographical area formed by restricted regions (called *cells*), each having one central station functioning as part of a high speed network, and several mobile stations communicating with it through radio links.

This investigation introduces a new conceptual cellular network which enables multiple packet transmission within the same area by using slotted multi-channel radio links. Its operation is described by the specification of its medium access protocols, buffer administration techniques and channel assignment algorithms.

Analytical models are introduced in order to predict the behaviour of some critical performance measures. It is shown that the central station can be represented by a multi-queue model with a single server traveling in cyclic order. We derive the equilibrium distribution of the length of each queue and the channel throughput by solving the underlying Markov chain in steady state.

More generally, we introduce a novel way of modelling polling systems of this type by representing the inter-visit time probability distribution as phase-type. We derive equations for the time delay distribution, which previously has been considered only in continuous time models. This technique is of greater generality and can be extended to several other polling strategies.

A global view of the network is also undertaken using decomposition methods. We approximate the corresponding queuing network as being of product-form and obtain the flow equivalent server (FES) of a given cell. However, this approximation leads to a prohibitive computational effort, since we are dealing with networks with large population and number of nodes. A novel technique for estimating asymptotic properties of closed queuing networks is introduced, based on a partial fraction ex-

pansion of the generating function of the normalising constant. Again, this technique has a much wider applicability, not being limited to the model analysed here.

We apply these techniques numerically and show how they can be used by the network designer to compare different medium access protocols and to estimate performance quantities under various workloads. We validate our phase-type model against simulation and use it to choose an adequate number of radio channels to operate in each cell. Finally we discuss the problem of avoiding congestion at the base station by analysing the closed queuing network model and its asymptotic properties. We show that our method is asymptotically exact and compares favourably with the existing ones in large-medium networks.

## Acknowledgements

First and foremost, I would like to thank my supervisor Dr.P.G.Harrison, who gave me encouragement and sound advice during difficult periods of this research and made this thesis possible. I am also very grateful to all members of the Advanced Languages and Architectures Group at Imperial College for providing a stimulating environment to work in. In particular, I owe special thanks to Dr. A. J. Field, to Mr. A. de C. Pinto for our thought-provoking (and sometimes exhausting) technical meetings and to Ms. E. Pitel for her careful reading of some chapters of this thesis.

Many thanks are also due to Sarah, for her editing skills and being so supportive and caring.

I dedicate this thesis to my parents – *Muito Obrigado!*

Finally, I would like to thank *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES) – a Brazilian government-funded education body – for giving me the financial support I needed it.

# Contents

|          |                                                                |           |
|----------|----------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                                            | <b>9</b>  |
| 1.1      | Cellular and packet radio networks . . . . .                   | 9         |
| 1.2      | Issues in the design of cellular networks . . . . .            | 11        |
| 1.3      | Objectives of this thesis . . . . .                            | 14        |
| <b>2</b> | <b>Background and Related Work</b>                             | <b>18</b> |
| 2.1      | Introduction . . . . .                                         | 18        |
| 2.2      | The evolution of packet radio networks . . . . .               | 20        |
| 2.2.1    | Centralised networks . . . . .                                 | 21        |
| 2.2.2    | Distributed networks . . . . .                                 | 23        |
| 2.2.3    | Integrated networks . . . . .                                  | 27        |
| 2.2.4    | Multichannel networks . . . . .                                | 30        |
| 2.2.5    | Cellular Networks . . . . .                                    | 32        |
| 2.3      | Modelling packet radio networks . . . . .                      | 36        |
| 2.3.1    | Modelling single-hop networks . . . . .                        | 38        |
| 2.3.2    | Modelling centrally controlled packet radio networks . . . . . | 39        |
| 2.3.3    | Modelling multi-channel packet radio networks . . . . .        | 40        |
| 2.3.4    | Modelling cellular networks . . . . .                          | 41        |
| 2.4      | A proposed multi-channel cellular network . . . . .            | 43        |
| <b>3</b> | <b>Queue Length Distribution in the Base Station</b>           | <b>49</b> |

|          |                                                                       |           |
|----------|-----------------------------------------------------------------------|-----------|
| 3.1      | Introduction . . . . .                                                | 49        |
| 3.2      | Base Station Operation . . . . .                                      | 50        |
| 3.3      | Model description . . . . .                                           | 51        |
| 3.4      | Steady state solution . . . . .                                       | 52        |
| 3.5      | Performance measures . . . . .                                        | 57        |
| 3.6      | Numerical Example . . . . .                                           | 59        |
| 3.7      | Conclusion . . . . .                                                  | 62        |
| <b>4</b> | <b>Time Delay Distribution in the Base Station</b>                    | <b>63</b> |
| 4.1      | Introduction . . . . .                                                | 63        |
| 4.2      | Model description . . . . .                                           | 65        |
| 4.3      | Probability distributions of phase-type . . . . .                     | 67        |
| 4.4      | Analysis of the model . . . . .                                       | 70        |
| 4.4.1    | The Inter-Visit Time Distribution . . . . .                           | 71        |
| 4.4.2    | The Reference Queue Operation . . . . .                               | 73        |
| 4.5      | The algorithmic solution . . . . .                                    | 82        |
| 4.6      | Numerical results . . . . .                                           | 84        |
| 4.6.1    | Model validation . . . . .                                            | 86        |
| 4.7      | Conclusion . . . . .                                                  | 90        |
| <b>5</b> | <b>Asymptotic Analysis and Cell Modelling</b>                         | <b>91</b> |
| 5.1      | Introduction . . . . .                                                | 91        |
| 5.2      | Product-form queuing networks . . . . .                               | 93        |
| 5.2.1    | The visitation rate . . . . .                                         | 95        |
| 5.2.2    | The convolution theorem . . . . .                                     | 95        |
| 5.3      | Expressions for the normalising constant in product-form networks . . | 98        |
| 5.3.1    | Single pole networks with constant service rates . . . . .            | 99        |
| 5.3.2    | Homogeneous networks with constant service rates . . . . .            | 100       |

|       |                                                              |            |
|-------|--------------------------------------------------------------|------------|
| 5.3.3 | Multiple pole networks with constant service rates . . . . . | 101        |
| 5.3.4 | Infinite server networks . . . . .                           | 105        |
| 5.3.5 | Combined networks . . . . .                                  | 106        |
| 5.3.6 | More general queuing networks . . . . .                      | 108        |
| 5.4   | Asymptotic properties . . . . .                              | 109        |
| 5.4.1 | Single pole networks . . . . .                               | 109        |
| 5.4.2 | Homogeneous networks . . . . .                               | 110        |
| 5.4.3 | Multiple pole networks . . . . .                             | 111        |
| 5.4.4 | Infinite server networks . . . . .                           | 113        |
| 5.4.5 | Combined networks . . . . .                                  | 113        |
| 5.5   | Cell Modelling . . . . .                                     | 116        |
| 5.5.1 | Cell Architecture . . . . .                                  | 116        |
| 5.5.2 | Queuing network model . . . . .                              | 118        |
| 5.5.3 | The flow equivalent server of the cell . . . . .             | 119        |
| 5.6   | Conclusion . . . . .                                         | 123        |
| 6     | <b>Network Design and Comparative Analysis</b>               | <b>124</b> |
| 6.1   | Introduction . . . . .                                       | 124        |
| 6.2   | Designing a buffer administration technique . . . . .        | 126        |
| 6.2.1 | Modelling the buffer administration techniques . . . . .     | 128        |
| 6.2.2 | Comparative Analysis . . . . .                               | 131        |
| 6.2.3 | Conclusion . . . . .                                         | 135        |
| 6.3   | Choosing the numbers of channels and stations . . . . .      | 136        |
| 6.3.1 | The service time distribution . . . . .                      | 137        |
| 6.3.2 | Numerical Results . . . . .                                  | 140        |
| 6.3.3 | Conclusion . . . . .                                         | 141        |
| 6.4   | Congestion control at the base station . . . . .             | 142        |
| 6.4.1 | Asymptotic bounds for the cell throughput . . . . .          | 143        |

|       |                                                          |     |
|-------|----------------------------------------------------------|-----|
| 6.4.2 | Numerical Results . . . . .                              | 149 |
| 6.4.3 | Conclusion . . . . .                                     | 155 |
| 7     | Conclusion and Future Work                               | 156 |
| 7.1   | Future direction for cellular networks . . . . .         | 159 |
| A     | Exact Analysis of a Two-Queue System                     | 161 |
| A.1   | State Space . . . . .                                    | 161 |
| A.2   | State Transition Probabilities . . . . .                 | 162 |
| B     | Irreducible Representations of a Phase-Type Distribution | 166 |
| C     | Asymptotic Expansions of the Normalising Constant        | 168 |

# Chapter 1

## Introduction

### 1.1 Cellular and packet radio networks

This thesis is concerned with cellular networks. These were developed in recent years as an extension of packet radio technology. As packet radio networks have evolved, they have become an integrated part of a wider packet exchange network. The topology of cellular networks extends over a large geographical area which is formed by restricted regions (called *cells*). Each of these cells has one *central station* (also referred to as a *base station*) which also belongs to a high speed network. They also comprise several *mobile stations* which communicate with the base station through radio links. A typical mobile station consists of a portable computer connected to a radio transmitter and receiver. A typical transaction would be a portable computer talking to a central database, or sending electronic mail to a remote user.

Packet radio networks are the result of packet switching technology extending into the field of mobile radio communication systems. Their purpose is to provide access for users located in areas where wired connections between source and destination users are either impractical or not cost-effective. A packet radio network comprises a number of mobile stations which use radio channels to integrate into a network.

Central to the notion of wireless links is the concept of broadcasting. Using a broadcast system, packets which are transmitted by one station can be received by all others. This system necessitates shared communication channels between stations. Consequently, it is essential to have a channel access policy in order to prevent multiple, simultaneous transmissions. Otherwise there might be packet loss due to collision.

In radio system design there is a tendency to use radio links in the entire end-to-end path. This approach requires relatively high transmitter power to cover even modest distances, and also forces an unnecessary increase in the size and weight of the radio hardware and its battery. Covering a large area also reduces the spectrum utilisation since a fixed frequency cannot be re-used in different regions.

A better approach is to configure the radio system to use a radio connection between a mobile user and a packet switched distribution network. In this way equipment of moderate power can be used throughout a large coverage area. Also the portable stations can have lighter and smaller transmitters. Moreover, an efficient use of the radio spectrum could be ensured by re-using frequencies throughout different regions.

In order to re-use frequencies, a large area must be divided into smaller regions with one base station in the centre. Radio channels on the same carrier frequency need to cover regions separated by a great enough distance to make the effect of radio interference negligible.

Cellular networks appear to be the natural choice when applying frequency re-use in packet radio networks. A cell is defined by an area inside which a particular channel set allows communication between the mobile and base station. Through frequency re-use, a cellular system can handle a far larger number of simultaneous packet exchanges than that achieved using only the allocated channel frequencies.

The end-to-end connectivity is ensured by the central station, which is respon-

sible for forwarding packets to their final destination. This may be another mobile station (located in the same or another cell) or a central computer to which these mobiles have access. An individual mobile station can only exchange packets with its nearest central station. When it crosses the cell boundary, it immediately switches to the new central station.

## 1.2 Issues in the design of cellular networks

Designing cellular networks involves specifying the parameters that will achieve the desired performance. Many design choices must be made before the networks can be developed. There is usually no single correct choice, and decisions are dependent on the particular environment in which the network will operate.

There are three areas of design which need to be identified. The first area covers the physical aspects of the network, specifically the radio connectivity, channel sharing and data recovery. The second area deals with the automated management of the network, in particular link management, routing and congestion control. The third area is connected to network maintenance.

Radio connectivity issues are related to the establishment of a digital link between stations in the network so that information may be exchanged from the traffic source to its destination. Given the number and location of the stations, and their traffic generation and mobility needs, the design problems at this level can be formulated as follows:

- How to select a group of radio frequencies which can be used during communication and how to allocate them to the various cells.
- How to evaluate the communication channels' throughputs and how to predict their behaviour as the offered traffic grows.
- How to adopt a cell topology and to devise its corresponding communication

protocol.

- How to evaluate the effect of this protocol on the network's performance and to compare different approaches.

Channel sharing mechanisms allow different stations access to the same group of channels. These prevent contention. This can be done in two ways. Firstly by providing a dynamic channel access algorithm which defines when users are permitted to transmit packets. This would be based on channel conditions and traffic demands. Alternatively, a central station can control the channel's access and give time-slots and a priority to each station. If a centralised topology is adopted it becomes necessary to find an associated buffer administration technique so that packets can be stored inside and then forwarded from the central station. Also essential is a channel assignment algorithm to allocate stored packets to different forwarding transmission channels. The main design problems at this level are:

- How to choose a buffer administration technique and its corresponding channel assignment algorithm.
- How to evaluate performance results, such as queuing delays and channel utilisation, and how these results are dependent on the technique adopted.

Data control and recovery are areas where careful design issues need to be considered. As in any packet communication system where some degree of reliability across the links is needed, acknowledgement mechanisms are necessary to notify the sending station that it has succeeded in transmitting a packet. The primary issue for data control and recovery is how to set up transmission parameters such as time-outs and retransmission windows that achieve an adequate level of performance.

The next design step concerns link management, routing and congestion control.

The link management issues refer to the network connectivity at end-to-end level. They involve both inter-cell communication through wired links and local commu-

nication through radio links. As the stations are mobile, a connectivity map of the network should be drawn up to show the station address with its corresponding location. This map is dynamically controlled and maintained by the central stations. It reveals the eventual changes of the mobile location. Link management techniques rely on the ability to update this map as quickly as possible. However the updating time, due to transient conditions, should be of sufficient length to ensure that the data traffic is not frequently overloaded with unnecessary control information. Therefore a trade-off between the updating time and the traffic overload is required.

Routing provides a means of selecting a communication path between stations and a way of giving full connectivity within a certain coverage area. Apart from issues related to the node's connectivity, routing strategies pose problems connected to the network topology. If the topology inside the cell is not centralised a packet forwarding technique is necessary. Packet forwarding is necessary whenever station *A* communicates with stations *B* and *C*, and where stations *B* and *C* are situated in such a way that direct communication is impossible between them. There are several types of packet forwarding mechanisms such as simple and directional flooding. All types introduce additional traffic by duplicating packets. This causes problems with the efficiency of the communication protocols.

Congestion control ensures that the traffic in the network does not rise above a certain level. In essence, a packet radio network is a network of queues. At each node, there is a queue of packets for each outgoing channel. If the rate of arrivals at these queues exceeds the rate at which packets are transmitted, the queue size grows out of control. The object of the congestion control is therefore to limit the network traffic so as to avoid response time collapse.

The third design step for cellular networks concerns network maintenance and so its behaviour should be studied from a global viewpoint. One issue connected with maintenance is that of the temporary change in the network topology due to

occasional failures. The main problems at this level are how to determine the cost of these failures in terms of network performance, and how to achieve an overall network reliability by trading off redundancy with hardware costs.

### 1.3 Objectives of this thesis

The main objective of this thesis is to provide quantitative models for the design of cellular networks.

We initially describe the evolution of packet radio and cellular networks. They were first developed in the 1960's and have become widespread in a great number of applications. As regards the needs and challenges of the operating environments, it is clear that these can be met only by enhancing the network architecture and its protocols.

We then introduce a new conceptual cellular network which enables multiple packet transmission within the same area by using slotted, multi-channel radio links. We describe its operation by showing its medium access protocols, buffer administration techniques and channel assignment algorithms.

Analytical models are introduced in order to predict the behaviour of some performance measures. It is shown that the central station can be represented by a multi-queue system with a single server traveling in cyclic order at discrete time intervals.

We carry out a closer investigation of the base station management schemes. We represent the multi-queue system by a Markov chain whose states correspond to the number of packets in each queue and the position of the server at the beginning of each time interval. We consider a new situation in which no more than one customer from each visited queue is served. Furthermore, in order to avoid overlapping, a given queue is not visited more than once at each discrete time interval. A set of linear equations is produced from the Markov chain's balance equations. The queue

length distribution and the channel throughput follows from the solution of these equations.

The main performance measure of interest in multi-queue systems is the time spent by the messages in the transmission queues. Although some important results concerning waiting times in multi-queue systems are available in the literature (e.g. [Taka 86], [RuBa 90], [KoKo 77]), analytical results are available only in a few cases (see for example [TaTT 88]). We introduce a new (approximate) method to evaluate the sojourn time distribution of packets in each queue by introducing a new way of looking at the server. Our approach is based on the fact that the inter-visit time is a random variable represented by transitions in a discrete Markov chain. Each state of this Markov chain represents the position of the server at each discrete time interval. We show how to construct its transition probability matrix and then derive a recursive matrix equation that yields the inter-visit time probability mass function. Based on this equation we obtain an equivalent phase-type system that produces the same distribution, following the modelling approach of M. F. Neuts (see [Neut 81], [Neut 77], [Neut 79]). Although the inter-visit time distribution is of phase-type, the waiting time distribution is not, in the discrete case. Therefore, we use Neuts' results wherever possible, but also derive new theoretical results for the discrete versions of certain theorems. We then use these results to produce equations for the residual service time distribution, the waiting time distribution and formulas for their respective first and second moments. This model is validated against simulation. We show that it is very accurate when the incoming traffic is low or moderate. We observe that this technique is of greater generality and can be extended to several other polling strategies.

A global view of the network is also undertaken using decomposition methods. We introduce a closed queuing network model to analyse asymptotic properties of cellular networks and evaluate the flow equivalent server (FES) of a cell. A

method for determining performance values for closed queuing networks is given in [ChHW 75]. Performance measures of interest are calculated from ratios of the normalising constant at different populations. However, evaluating this normalising constant directly from the definition frequently leads to prohibitive computational effort, particularly when dealing with networks of large populations and number of nodes. In these cases it is more convenient to derive asymptotic expansions. One well-known technique for deriving this expansion was introduced by J. McKenna and D. Mitra (see [McMi 82],[MiMc 86] and [MiMc 84]). Although this technique simplifies the computation of the normalising constant, it is restricted to certain types of node and can be applied only in networks with think times considerably larger than processing times. We introduce a new technique for estimating asymptotic properties of closed queuing networks, based on a partial fraction expansion of the generating function of the normalising constant. The method is also adopted to the case of medium-sized populations. Again, this technique has a much wider applicability, not being limited to the model analysed here, and compares favourably with [McMi 82].

We show how these techniques can help the system designer to analyse and compare different solutions and to identify trouble spots in the network protocols. We apply the new techniques introduced and give an interpretation for the results obtained. This is done by addressing some of the design issues described here. Initially, we compare three different buffer administration techniques. They vary in the way in which the channels are allocated between the transmission queues. We show the strengths and weaknesses of these techniques by comparing their performance. Subsequently, we show how to select a suitable number of channels in a cell based on the traffic requirements of the area where the base station is located. We use the phase-type model to produce a load curve which reflects the base station's amount of work. The number of channels and the number of stations supported can be de-

terminated from the analysis of this curve. Finally, we discuss the problem of avoiding congestion in the base station. We use our asymptotic method to calculate the cell throughput and give its bounds when the number of packets in the cell becomes large.

The remainder of this thesis is organised as follows. Chapter 2 presents an evolutionary view of packet radio and cellular networks. We discuss their limitations and introduce a new network which operates using multi-channel radio links. Chapter 3 presents a Markov chain model for the multi-queue system in the base station. We derive the queue length distribution of the packets in each buffer. We also evaluate the channel throughput considering a multi-queue system with finite buffer. Chapter 4 calculates the sojourn time distribution of discrete time multi-queue systems. Chapter 5 is dedicated to the study of asymptotic properties of queuing networks and its particular application to cellular networks. Chapter 6 gives examples of how these models can be used by the system designer and gives some recommendations based on the analytical results obtained. Our conclusion and suggestions for future work are laid out in the final chapter.

# Chapter 2

## Background and Related Work

### 2.1 Introduction

Wireless communications technology and its services have expanded rapidly in recent years, resulting in the introduction of new systems and applications. This expansion has been made possible by using digital technology in mobile and portable communications. Wireless techniques have long been used in a large range of applications including mobile radio-telephones, vehicular services and cordless telephones. As interest in portable communications grew, so research and development programmes were initiated to apply packet radio network technology to cellular environments.

This chapter summarises the results of these programmes and formulates an evolutionary view of packet radio and cellular networks. We also describe how these have been modelled and how performance quantities were derived from their corresponding models. Finally, we introduce a new digital data cellular network that may possibly be regarded as the next step in this evolution.

This chapter is organised as follows:

Section 2.2 describes the evolution of packet radio networks. We start by describing the two pioneer packet radio networks in operation: ALOHANET (see [Abra 70] and [AbKu 73]) and DARPA PR-NET (see [JuTo 87]). The first was developed by

the University of Hawaii in 1970 to allow scattered stations to access a central computer system. The second was a result of a research programme in 1972 by the Defense Advanced Research Projects Agency (DARPA) to develop a packet radio network. The aim of this work was to provide a means of sharing the broadcast radio channel while at the same time allowing for the incomplete and changing connectivity map of the network environment.

Although some stations of ALOHANET and DARPA PR-NET can also be connected to a main packet switched network, there are some limitations to be considered. ALOHANET does not perform well with moderate/high traffic. DARPA PR-NET stations need to be located within line-of-sight of each other. This is difficult to ensure over large geographical areas. Furthermore, using radio to provide end-to-end connection over a large area requires a considerable amount of signal power and does not utilise the spectrum efficiently. This is because frequencies are not re-used. Consequently simultaneous communication is impossible.

A more sophisticated topology was then introduced by D.C.Cox (see [Cox 87]) in which a portable radio system used a radio link only to connect to the public telephone network. We describe how it can be implemented by showing its architecture, and some of its operational aspects such as radio spectrum utilisation, operating frequencies and medium access protocols.

Next we describe a way to improve the network throughput by providing multiple communication channels. This was proposed by N. Shacham and J. Westcott (see [ShWe 87]).

We conclude this section by showing that cellular networks are the result of a natural evolution of packet radio networks. It is an attractive solution if a greater coverage area is desired since it supports more stations without saturating the radio channels (achieved by re-using frequency bands), and without imposing restrictions on the portables' locations (achieved with a central base station in each cell).

Section 2.3 reviews modelling techniques for packet radio and cellular networks. We show why standard techniques for performance evaluation of point-to-point networks can not be applied in broadcast networks. We describe how packet radio networks differ from wired networks, and how these differences can be incorporated in the analytical models.

Section 2.4 proposes a new digital data cellular network in which the cell throughput can be improved by using multi-channel transmission in the radio communication channels. We describe the architecture, the medium access protocol and the buffer administration technique of this network.

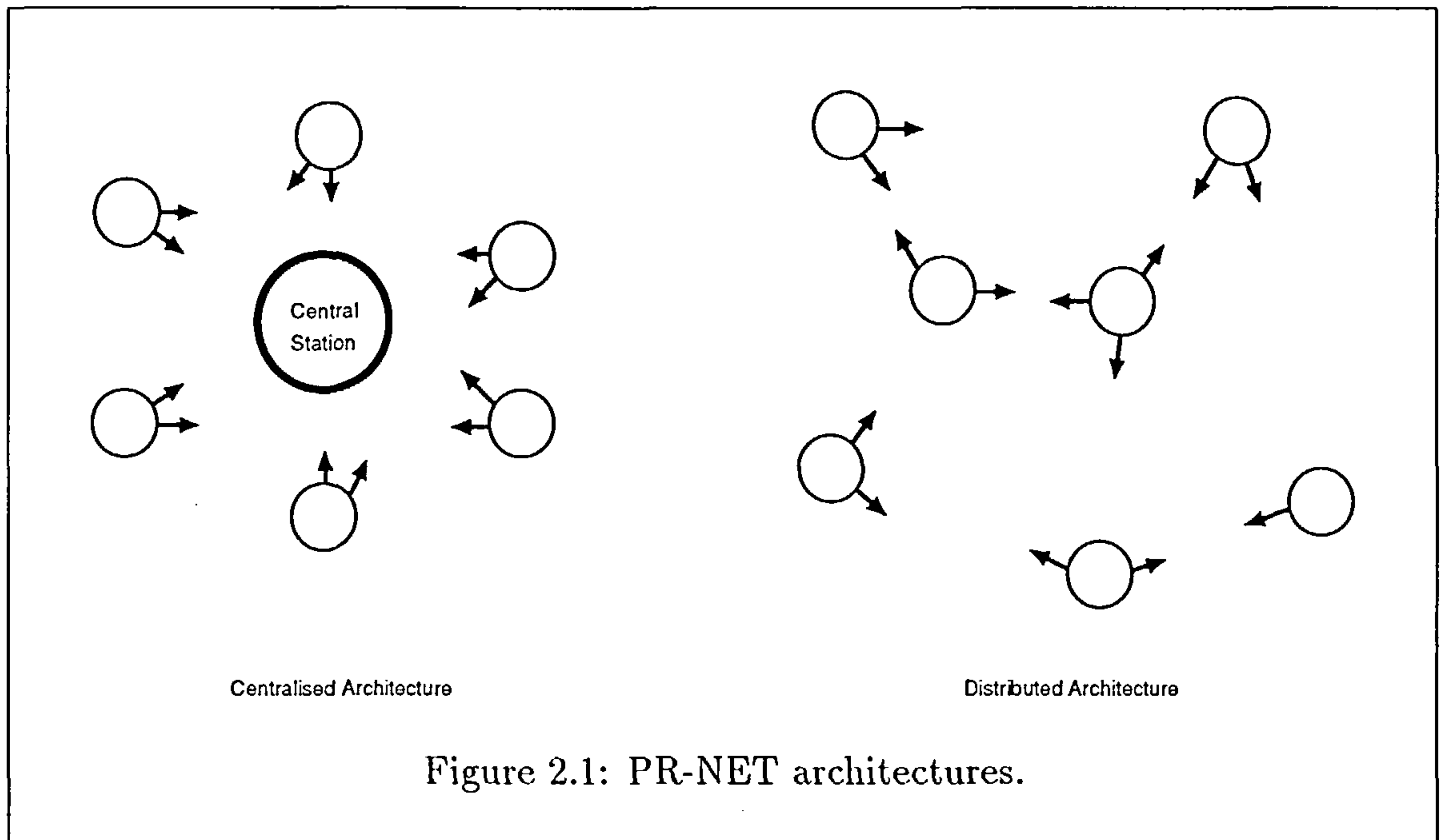
The modelling techniques and the analysis discussed throughout this thesis are tailored to this network. However, the models are also appropriate for networks with different medium access protocols and buffer administration techniques.

## 2.2 The evolution of packet radio networks

The packet radio network was conceived as an extension of the packet switching technology to broadcast networks. Packet switching technology is based on partitioning long messages into small and independent packets and multiplexing them throughout the network. Broadcast networks were designed to take advantage of the nature of the communication medium which is readily available, easily accessible and suitable for communication among mobile users.

Its architecture can be centralised or distributed (see Figure 2.1). In a centralised architecture, the communication can be established only from a remote to a central station. A link between two remote stations is indirect since it is mediated by the central station. In a distributed architecture a direct packet exchange between two stations is possible through a common channel access protocol.

According to the connectivity of each station, a packet radio network can be classified as *single-hop* when every packet transmission is received by all destination



nodes (i.e., the network is fully connected), or *multiple-hop* when not all stations are within line-of-sight of each other (therefore packet retransmission is necessary to provide connectivity to remote located stations).

### 2.2.1 Centralised networks

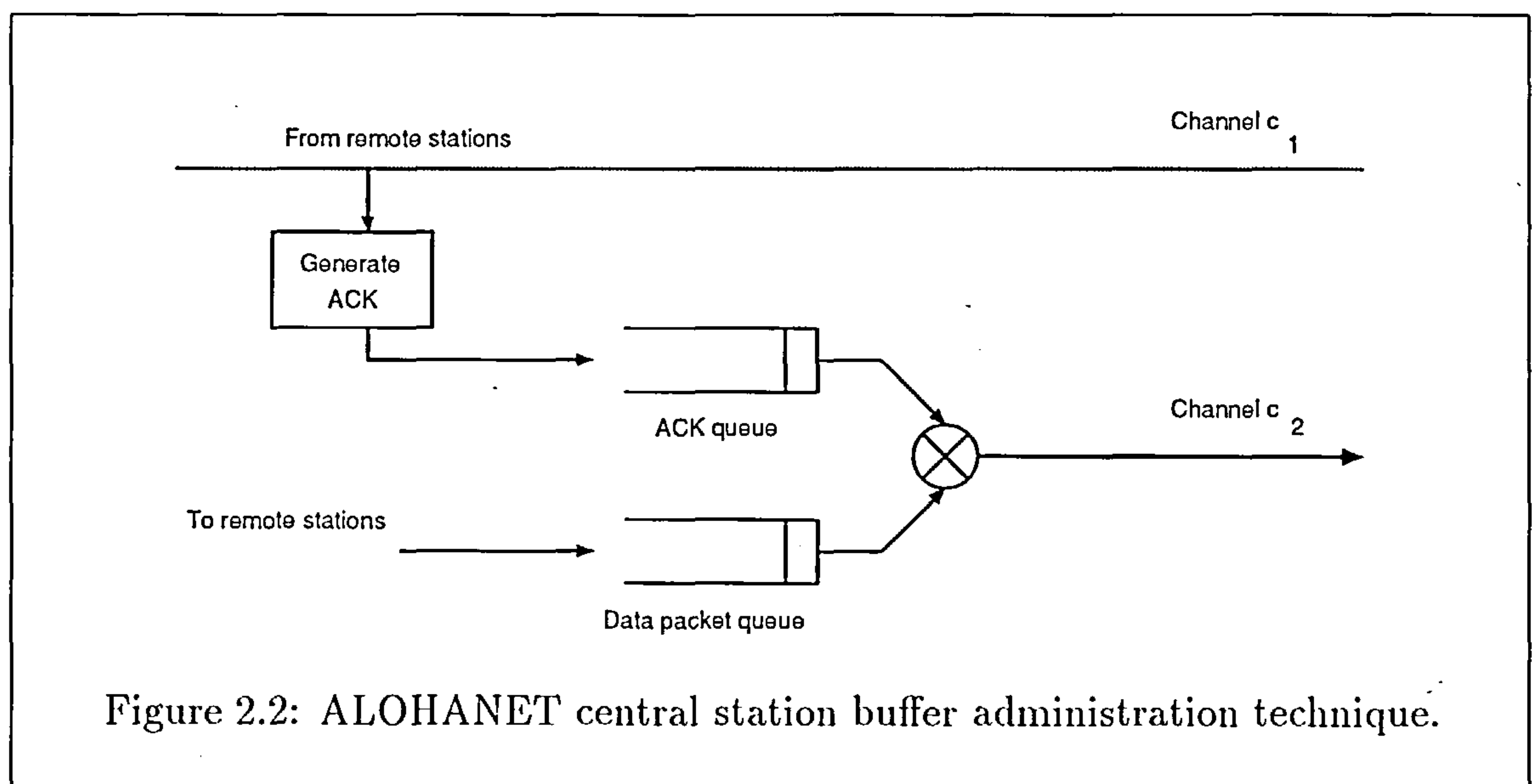
One example of centralised networks is ALOHANET, developed by the University of Hawaii, which was the first packet radio network to become operational. Its objective was to integrate terminals located at different points in the campus and provide access to the university computer system.

Traffic was primarily from remote to central station and vice-versa. Messages from remote to remote stations were routed via the central station. Two channels were used to handle this traffic. The first channel  $c_1$  operates at frequency 407.35 MHz and was dedicated to the *uplink* packet flow, i.e., packets from remote to central station. The second channel  $c_2$  operates at a frequency of 413.475 MHz and was dedicated to the *downlink* packet flow, i.e., packets from central to remote station. Both channels have a bandwidth of 100 KHz, using Phase-shift keying

(PSK) modulation and data rate of 9600 bps. Packets were approximately 800 bits long and could carry a maximum of 80 characters of data.

The uplink channel  $c_1$  was shared amongst all remote stations. Consequently, a medium access protocol was necessary. The protocol used was ALOHA (see [Toba 80] and [AbKu 73]), in which every remote station transmits a packet whenever it has data to send and expects to receive an acknowledgement from the central station. The sender retransmits the same packet if the acknowledgement is not received after a timeout interval. In this case, it is assumed that a collision occurred, and to avoid a second collision the packet is retransmitted after a random period uniformly distributed between 0.2s and 1.5s.

The downlink channel  $c_2$  is dedicated to the central station transmission packets. Two types of packets were transmitted in this channel: acknowledgement (ACK) and data packets. The former have higher priority than the latter since the timing of ACKs is critical. Therefore two queues of packets waiting to be transmitted are maintained by the central station, one for ACKs and one for data packets (see figure 2.2).



In the original implementation central station packets were not acknowledged

by the remote stations, since there is no contention in the downlink channel, consequently there was a higher probability of successful transmission. Moreover, remote station ACKs would increase the traffic in the uplink channel.

Other aspects of ALOHNET, such as routing and flow control can be found in [Stal 88] and [Abra 70].

### 2.2.2 Distributed networks

One example of distributed networks is DARPA PRNET, developed by the Defense Advanced Projects Agency, which was designed to provide reliable computer communication in a mobile environment. Broadcast radio is used as a communication medium, which brings the benefits of supporting mobility and requiring easy installation and configuration procedures.

The network area is larger than the radio coverage area of each individual station, therefore each station is connected only to a certain number of stations. Furthermore, this connectivity configuration is altered from time to time, since stations can change their locations. However, each station must operate within the line-of-sight of at least one other station. Once one link is established, the system discovers the new connectivity map of the network and organises routing strategies dynamically.

#### Network architecture

Each station comprises a Low-cost Packet Radio (LPR) and a collection of digital devices such as host computers, gateways, and Network Interface Units (NIUs), each attached to a LPR through a wire high-level data link control (HDLC) interface.

The LPR consists of both digital and radio subsystems. The radio subsystem is capable of omnidirectional, half-duplex transmission/reception at 400 and 100 kbit/s data rates. The digital subsystem controls the routing and flow of packets between stations and exchange packets through the HDLC interface. Every packet

received by the LPR via radio link is either retransmitted to another station that is one hop closer to the final destination or transmitted to the attached device via wire interface.

In order to send data across a PRNET, a device (such as a host computer) must be connected to a LPR via a HDLC wire interface. This station must run DARPA internetwork, transport and application level protocols. If a user wishes to send data across the PRNET from a terminal that does not run the required protocols, a NIU is required between the device and the LPR. The PRNET can also be connected to other networks via a gateway. (see figure 2.3).

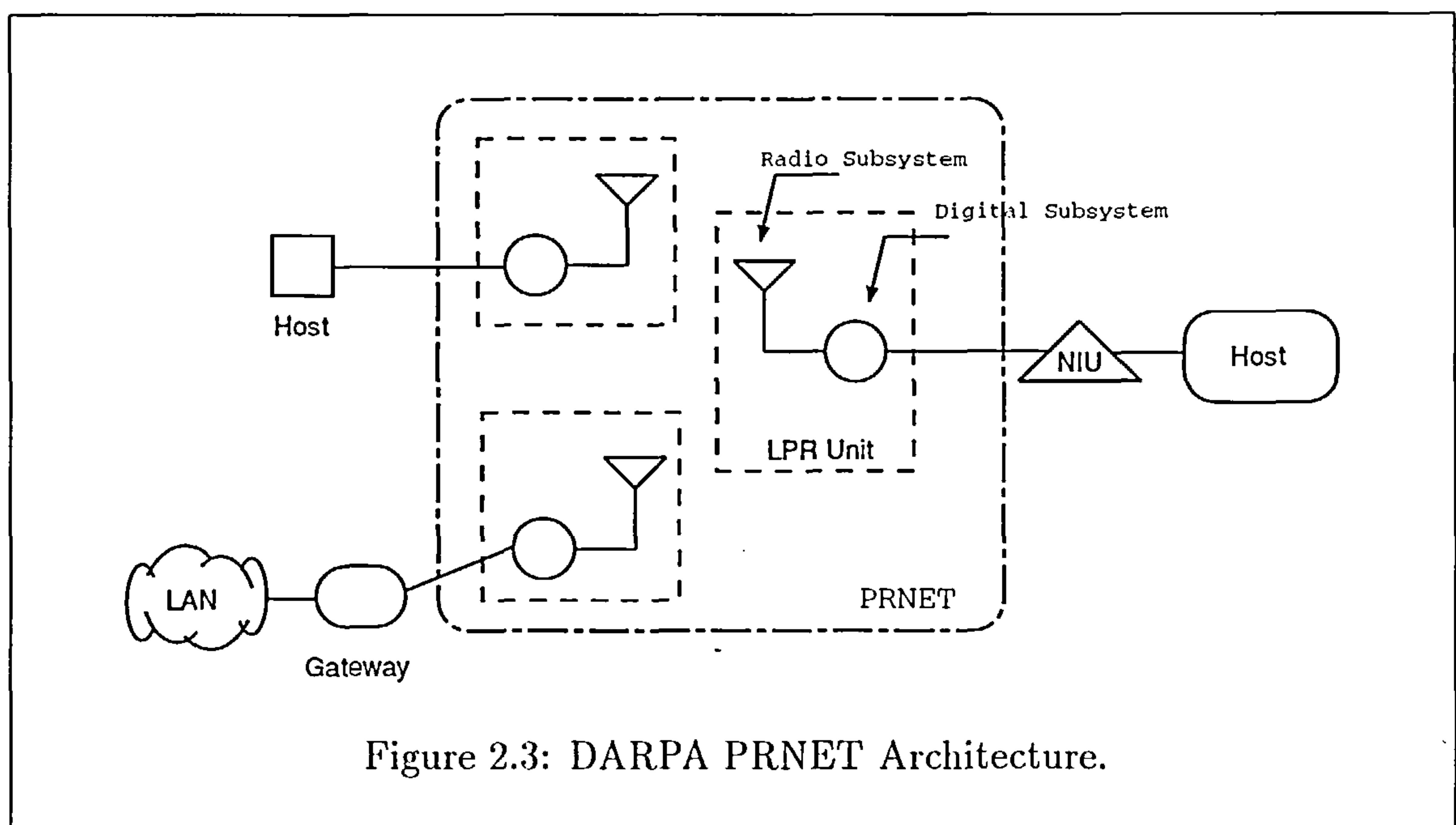


Figure 2.3: DARPA PRNET Architecture.

### Network size

The current network protocols are designed to support up to 138 stations, each of which can be formed by any combination of a LPR and attached devices. As the number of stations increases, the number of neighbours (those stations in direct line-of-sight) is likely to increase. The current protocols allow each station to support up to 16 neighbours. If, however, the stations are more densely located, the congested

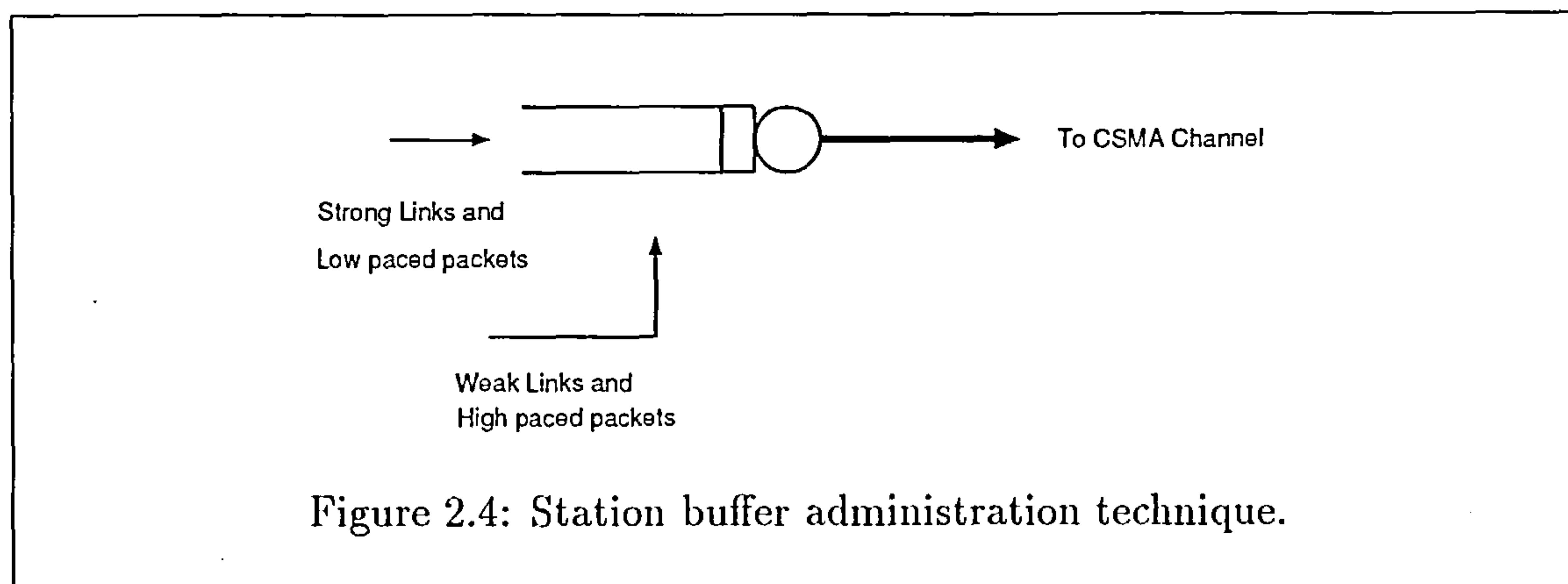
stations employ an algorithm to determine a logical neighbourhood of 16 stations.

### **Routing and packet forwarding**

Each station stores and maintains enough information about the network topology so that it can make independent decisions about how to route data through the network to any destination. The network topology is stored in tables which are established upon initialisation and updated as the topology changes. Stations broadcast a Packet Radio Organisation Packet (PROP) every 7.5 s to announce its existence and to inform the network topology from its perspective. Its neighbours use this packet to update their tables. Similarly a given station will receive periodic PROPs from its neighbours and maintain its own table. Based on this information a route can be established, since a station knows when a received packet should be forwarded (see [JuTo 87] for more details).

### **Buffer administration technique**

As a general rule, packets to be retransmitted in each station are stored in a First-Come-First-Served (FIFO) queue. However there are two exceptions to this rule. The first is a result of a *pacing procedure* (see [JuTo 87]), that can slow or speed up the rate of packets to be forwarded. This is intended to provide a fair use of the radio channel, since the retransmission rate of the packets can be equalised according to their destination. In this way, high paced packets have higher priority in the queue. The second exception of the FIFO rule is to give low priority to packets from HDLC interface or packets from a strong (with low error rate) radio link. The purpose of this exception is again to provide a fair use of the radio channel, since a stream from a practically error-free source would monopolise the channel and the correct packets from a weak radio link would not get a fair share (see figure 2.4).



### Medium access protocol

The medium access protocol used in DARPA PRNET is Carrier Sense Multiple Access (CSMA), which assumes that transmissions can not overlap in time, and prescribes a random delay before transmitting a packet whenever the sending station finds the channel busy.

### Limitations

DARPA PRNET was designed to support a relatively small number of stations. A typical network may be composed of 50 stations. Although research is being carried out to allow simultaneous operation with other packet radio networks without a gateway, the use of only one radio channel limits the number of stations in a given neighbourhood. Furthermore, the use of forwarding techniques imposes an additional traffic which may saturate the channel - even with a relatively low number of stations.

Finally, due to the dynamic network topology there is a possibility that a given station becomes disconnected even if it remains static. Therefore, an eventual break in the radio link imposes limitations in the network reliability and performance.

### 2.2.3 Integrated networks

As we have seen in sections 2.2.1 and 2.2.2, early packet radio networks offered a relatively limited digital transmission capacity since only one or two channels are shared among the stations. Consequently they support a not very large number of stations. Furthermore, their operation could compete with other radio services such as mobile radio telephone, radio paging or vehicular services for the use of the spectrum.

For this reason, a methodology to provide a planned use of the radio spectrum and a more universal portable network, which supports most of the radio services has been proposed (see [Cox 87]). This approach accomplishes an integrated portable radio communication which supports data and voice radio networks as part of the telephone exchange network.

Fixed radio ports functioning as part of the telephone distribution network are placed throughout service regions. Efficient use of the spectrum is ensured by frequency reuse, as we will see in the sections below.

Integrated networks operate in such a way that a portable system uses radio connection only to connect to the nearest fixed radio port. The end-to-end connection is guaranteed by the coordinated operation of both radio and telephone links.

#### Network architecture

Integrated Networks use radio links only for the last thousand or so feet of individual phone lines (see figure 2.5). The ends of the telephone exchange network, called *Network Radio Ports* or simply *ports*, are connected to portable terminals either directly via radio links or indirectly via telephone base terminals. The telephone base terminals are interfaces between ports and the portable terminals. The implementation of ports using only radio links is easier since it eliminates the need for wiring to and within individual buildings. A portable terminal can be either a

cordless telephone, or a packet radio station.

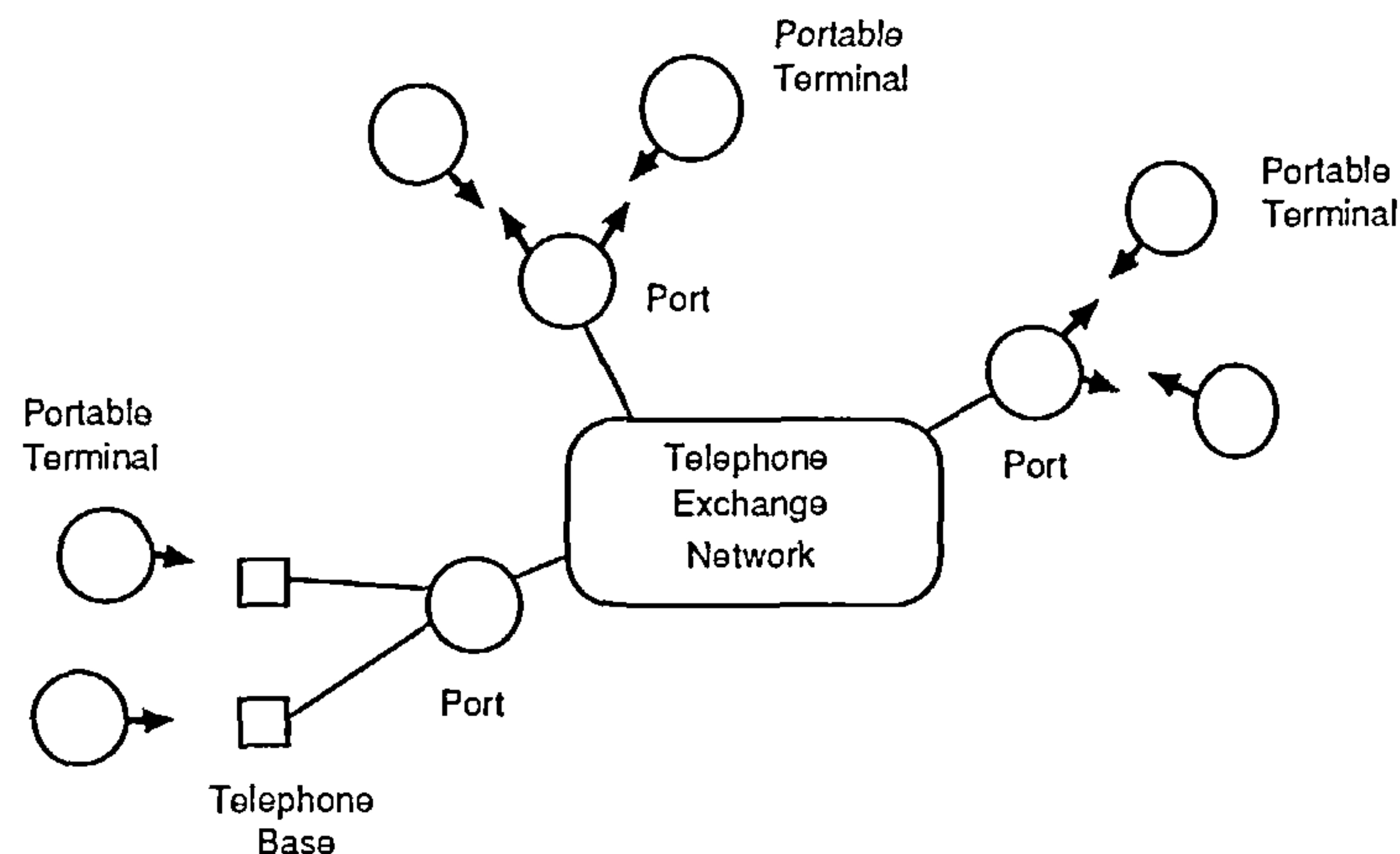


Figure 2.5: Network Architecture.

### Spectrum utilisation

In order to cope with the undesirable effects of the radio attenuation in and around large buildings, network radio ports must be distributed throughout the network coverage area. A large building, for example, requires the use of several ports.

Distributing ports in large areas permits frequency reuse, which can significantly increase the user density operating in a limited bandwidth. Frequency reuse is a strategy in which a given area, which operates at a frequency  $f$  is surrounded by areas operating in non-interfering frequencies. The frequency  $f$  can be reused in a remote area located so that its interference is negligible in the first area.

### Operating frequencies

At the present date, there are no radio frequencies allocated for integrated networks. Frequency allocations are limited for cordless telephones, mobile radio and radio paging. However from the technical point of view, the frequency range from 500 MHz to about 1 GHz is acceptable. This range is evaluated in [Cox 87] considering

factors like efficiency of antennas and radio circuits.

### Modulation and data rates

Considering that integrated networks support both portable and voice communications, it is assumed that the radio link uses digital modulation.

Data rates are not specified, but it is desirable for the radio link to be able to support Integrated Services Digital Networks (ISDN) transmission rates. This implies a data rate of at least 144 kbits/sec. Furthermore the radio link architecture should be able to accommodate different data transmission rates, due to the heterogeneous nature of the user requirements.

### Medium access methods

A two-frequency Time Division Multiple Access (TDMA) is proposed as a medium access protocol for integrated networks (see figure 2.6). One frequency  $f_1$  is used for port-to-portable (downlink) transmission and another frequency  $f_2$  is used for portable-to-port transmission.

The downlink data stream is divided into frames. Each frame is divided into blocks, so that up to one block is reserved for a portable in the frame. Every block contains in its header control information such as source and synchronisation sequence, destination address, and error-detection bits.

The uplink frequency  $f_2$  is paired with  $f_1$  and is used for transmission from portables to ports. The portable transmission blocks are synchronised with the downlink stream so that a time slot is given to every portable for its own transmission. Considering that a portable has one slot in every frame, it can place or not its own block if there is data to be sent.

A single frequency TDMA is also proposed: the reader should refer to [Cox 87] for more details.

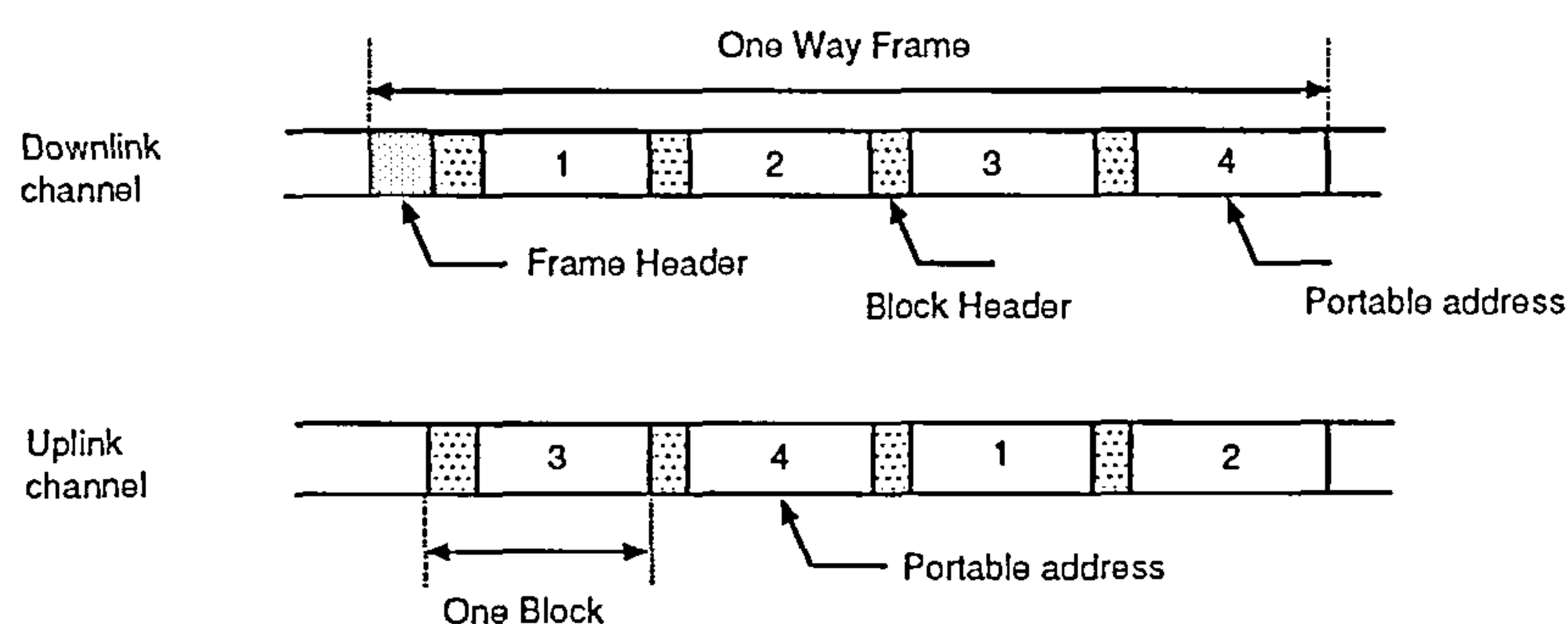


Figure 2.6: Dual frequency TDMA.

### Limitations

Integrated networks represent an advance in the current packet radio networks architectures and a challenge for the future, since they require a considerable effort to be implemented. However using telephone networks for data networks is adequate only for applications that do not require an extensive use of computational resources, since telephone lines are more appropriate for sporadic data exchange.

#### 2.2.4 Multichannel networks

The networks discussed so far comprise stations that share one or two channels. A typical station employs a transmitter/receiver (also called *transceiver*) that can operate in one or two frequencies. However, in the packet radio domain, a transceiver can change its channel in milliseconds (see [ShWe 87]). A multichannel PRNET permits the use of more than two channels within the same geographical area. Consequently a multiple transmission can be carried out if the transmission channels are orthogonal, i.e the crossover channel interference is negligible.

Multichannel operation is based on splitting a fixed transmission band usually fixed by the system designer (or governmental bodies as seen in section 2.2.3) into several frequency sub-channels. This is advantageous when the allocated bandwidth

exceeds the bandwidth required by the transceiver. In this case several parallel channels can be offered and a higher level of traffic can be supported by distributing them among different sources.

A distributed multihop PRNET with multichannel operation has been proposed (see [ShWe 87]). Two basic architectural options are possible, one with a single transceiver per node and another with several transceivers per node, which allows a node to make several transmissions simultaneously.

These channels can be either in the frequency domain, using Frequency Division Multiple Access method (FDMA), or in the code domain using Coding Division Multiple Access Method (CDMA). Code division refers to transmission schemes which are modulated using *spread spectrum* techniques (see section 2.3). The signal modulation is associated with a certain code. In this way, signals with different codes may overlap in time with little or no effect in each other.

One important constraint for both schemes is that the channels must be orthogonal, implying little crossover channel interference. The access method in each channel is the slotted ALOHA (this protocol is explained in briefly in section 2.3.1 and in more details in [Toba 80], [Stal 88] and [Tane 89]).

## Architecture

In a single transceiver architecture, a station selects a *quiescent* channel to listen on when idle. A transmission takes place in the quiescent channel if it is idle, otherwise another channel is selected. In this configuration each node must know which quiescent channel its neighbours are using. This can be done by allocating channels to stations at network initialisation and updating it whenever there is a change in the network topology.

A multiple transceiver architecture supports simultaneous transceiver operations within a station. This is achieved through increase in hardware cost and software

complexity.

### Limitations

A better utilisation of the spectrum can be achieved by having a multichannel operation in a distributed PRNET. However the use of a contention based medium access protocol imposes complex operation algorithms. Furthermore, the efficiency of this protocol is dependent on the propagation delay. We expect a decrease in channel efficiency when large areas are used.

#### 2.2.5 Cellular Networks

Cellular Networks represent the extension of the packet radio technology into the cellular environment. Its topology extends over a large geographical area formed by restricted regions referred to as cells. The idea of using cellular topology for radio systems is not new, in fact it has been used in Cellular Mobile Phone Service (CMPS) networks to provide telephone communication for portable users.

CMPS is a circuit-switched network used primarily for voice. Each mobile telephone (MT) is attached to an antenna that broadcasts in the 800 to 900 MHz range (in the U.S.). The cell size corresponds to the range of the mobile transmitters, so that the signal can be heard throughout its own cell. The cell shape is usually chosen to be hexagonal (see [Macd 79]). Within each cell a set of frequencies is available for communication. All the adjoining cells employ a different set of frequencies, so there is no interference. Assigning a different set of frequencies to a given number of cells is a problem addressed in [Macd 79]. Figure 2.7 illustrates the CMPS geometry.

Each cell has a base station (BS) at its center which can communicate with all internal MTs. When a MT is switched on it searches among the BS frequencies to find the strongest one and it sends a message announcing its number. The BS then tells the MT which cell it is in, and gives it a frequency to use. During the call

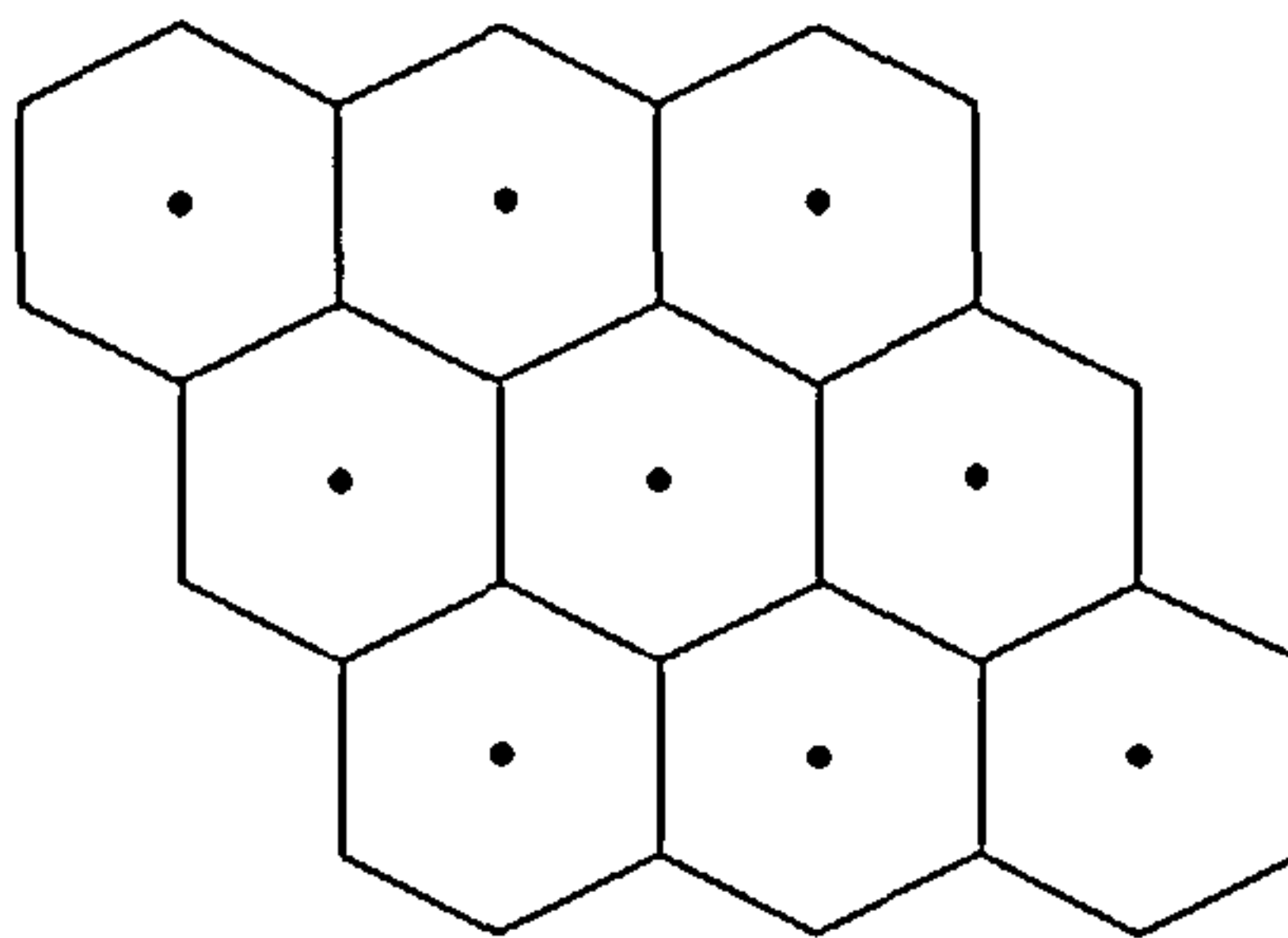


Figure 2.7: CMPS geometry.

set up procedure the MT sends a message asking for an available frequency. While the call takes place the MT keeps monitoring the signals of all BSs to see which one is the strongest. Whenever it discovers that a different BS becomes stronger (presumably because it has moved to another cell), the MT tells its BS which then hands it over to the new base. The new BS instructs the MT to switch its frequency. This automatic switching control function is referred to as a *handoff*. The location of each MT is monitored continuously by a computer to which all BS have access.

The Advanced Mobile Phone Service (AMPS), of the United States and Canada and the Groupe Speciale Mobile in Europe are two examples of CMPS in operation.

The AMPS (see [Lee 82] and [Macd 79]) operates in a 40 MHz bandwidth in the 800 to 900 MHz range. This bandwidth is divided into 666 two-way channels. Each two-way channel consists of two 30 KHz subchannels used for uplink and downlink communication. These channels are grouped in sets and distributed amongst the cells so as to avoid adjoining cells having interfering sets. Each set contains channels with the largest possible frequency separation to avoid a large guard band in each channel, i.e. the channels are grouped in such a way to avoid adjacent-channel interference.

The GSM (see [WaMH 91] and [PaGa 89]) is the first digital CMPS to become operational in some European countries. It has the advantage of following a standard

set of services, protocols and interfaces, and supporting interconnection to public telecommunication networks (ISDN and data). It operates in the region of 900 MHz. In total, 124 two-way channels are defined, each one occupying a bandwidth of 200 kHz. Voice and data are transmitted at a rate of 271 kbits/sec.

### Digital Data Cellular Networks

Although CMPS networks support, to a limited extent, data terminals they are not appropriate for a great number of applications, since they introduce long delays during the call set-up process. These delays become unacceptable particularly when the data traffic comes in largely spaced bursts of data blocks. The existing call set-up delays in telephone networks are appropriate for communication of speech or data which have a call holding time of at least tens of seconds.

Certain commercial applications, for example, require short data transactions between remote terminals and a central computer. In these applications the use of a CMPS network is not attractive due to its high cost and low performance. One solution would be to share a dedicated group of radio communication channels and have access to a wired packet switched network.

Digital data cellular networks meet the requirements of such applications. An example of such a network is the one opened for public service by Paknet, a joint venture company between Racal Telecom and Mercury Communications (see [DaSm 91]).

The overall structure of this network is shown in Figure 2.8. It consists of portable devices, and base stations connected to a packet switched exchange network. Each base station provides service to a large number of users within its geographical coverage area. Typically, this range extends to about 10 Km from the base stations, which are located in a cellular pattern to provide a continuous coverage area.

The radio link operates in the range of 159–164 MHz. This band is divided in

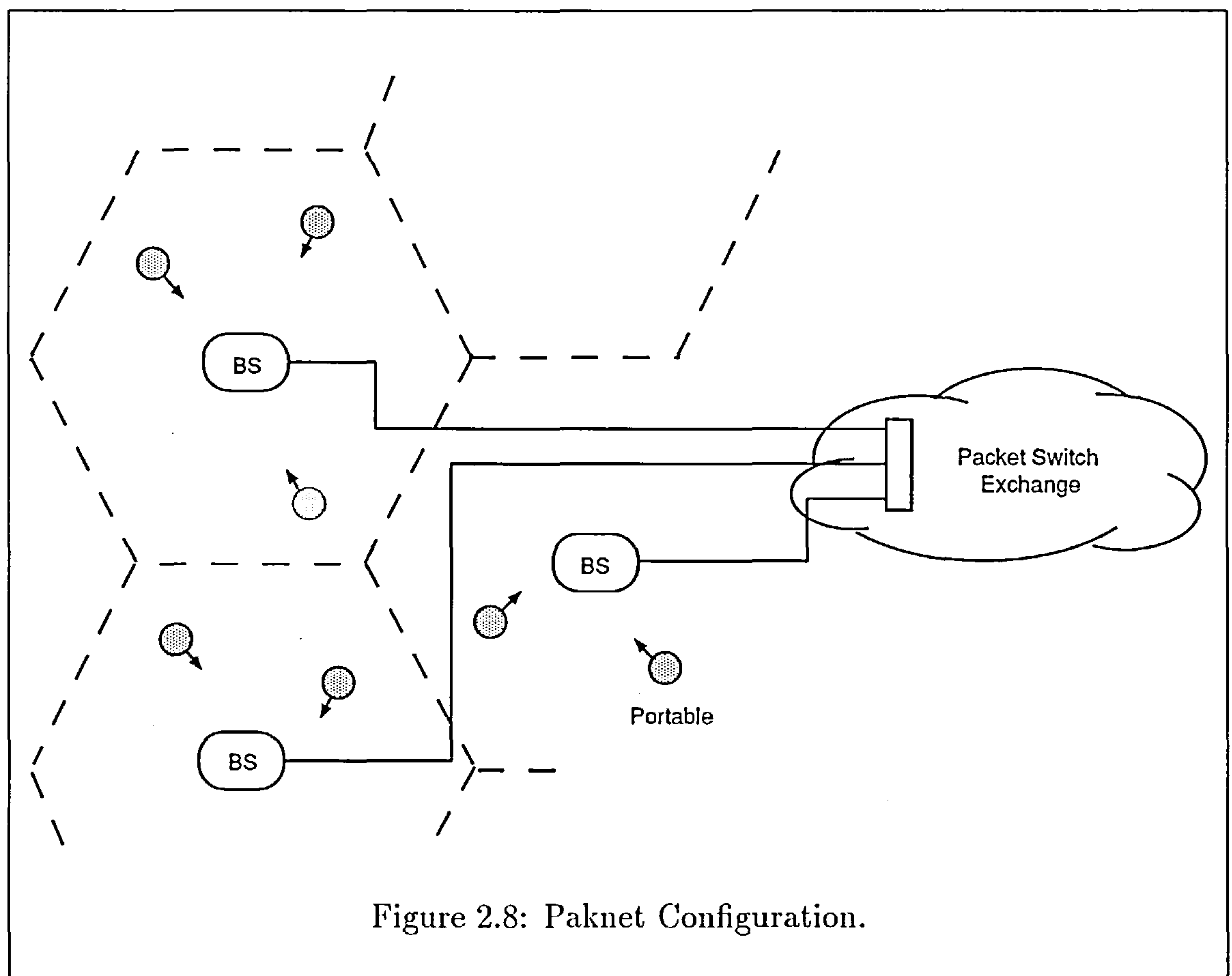


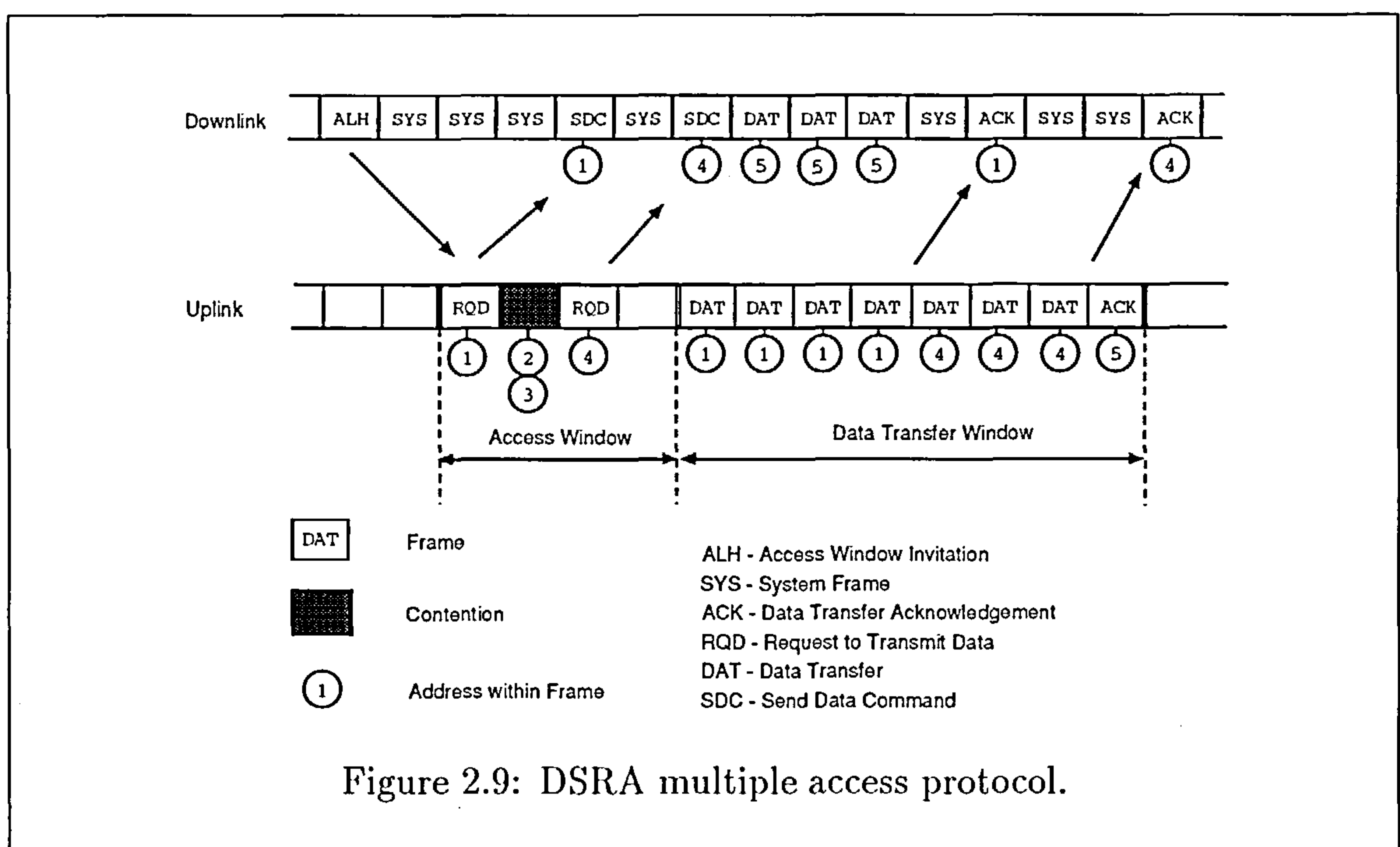
Figure 2.8: Paknet Configuration.

two-way channels for uplink and downlink communication respectively. The channel bandwidth is 12.5 kHz which supports a data rate up to 8 kbits/sec. The modulation scheme adopted is narrow-band binary frequency shift keying (FSK), in which the digital signal is modulated so that there are two different frequencies near the carrier frequency to represent the two binary values.

The medium access protocol adopted is referred to as DSRA (Dynamic Slotted Reservation ALOHA), which can be classified as a centrally controlled demand assignment protocol (see Fig. 2.9). Transmission in both uplink and downlink channels takes place in successive fixed-size time slots, or frames. Frames in the uplink channel are grouped in two windows of variable length: the access window and the data transfer window. The portable terminal acquires synchronisation by monitoring the downlink channel. When the base station decides to offer an access window

it broadcasts an *access window invitation* (ALH) frame, which also contains the next access window size. If a portable terminal has data to send it picks a random slot from the access window and transmits a *request to transmit data* (RQD) frame, which also specifies the number of slots necessary to transmit its data. If a RQD is successfully received, the central controller then transmits a *send data command* (SDC) frame, which specifies when the terminal is allowed to transmit its frames.

This protocol is dynamic because both the frequency and the length of the windows are adjusted dynamically to suit traffic requirements.



## 2.3 Modelling packet radio networks

The design of a PR-NET system involves a large number of issues as described above and in chapter 1. Clearly, a single model which incorporates all relevant performance measures can not be formulated effectively. However, performance analysis through mathematical modelling and simulation is an important part of the design process.

Unfortunately the modelling of packet radio networks has proven to be a complex

task. Techniques devised for point-to-point networks are not directly applicable since there are some aspects that makes the PR-NET operation different from fixed networks.

The first aspect is the effect of using a group of shared channels between several stations. Sharing resources among several users implies establishing priorities and resolving eventual conflicts, usually by queuing users when the resource is not available. Packet radio network models must encompass a medium access protocol and its associated buffer administration technique. The medium access protocol regulates the shared use of the group of channels by giving priorities and by controlling the channel availability. The buffer administration technique regulates the queuing process of the competing users and their corresponding servicing policy.

The second aspect is the signaling method used: *narrow band* or *spread spectrum*. These two methods differ by their corresponding capture effects. Capture refers to the ability of the station transceiver to receive correctly a packet despite the presence of another time overlapping packet. Narrow band networks operate in a zero capture mode, i. e., the overlap of two or more packets at the same receiver results in the destruction of all. Spread spectrum signaling supports non-zero capture by increasing the signal bandwidth and by providing a way to acquire bit synchronisation so that interfering signals are rejected if they are not synchronous with the received packet. This is done by adding a preamble in each packet. A preamble is a known code placed before the actual data. Having successfully processed the preamble, the receiver *locks onto* the data portion of the packet until its reception is completed. This can be done by using the fact that the preamble is strongly autocorrelated. In this way, any other simultaneous transmission appear as a background noise.

Performance models are described in terms of a medium access protocol, an associated buffer administration technique, a capture method, and aspects related to the network architecture (centralised or distributed), connectivity (single-hop

or multiple-hop) and the number of shared channels (single-channel or multiple channel).

The measures of interest are i) the packet delay, ii) the distribution of the population throughout the network, iii) the throughput/offered traffic characteristic, iv) the network capacity, i.e., the maximum throughput supported for a given required traffic, and v) stability conditions.

### 2.3.1 Modelling single-hop networks

Extensive analysis of single-hop networks operating under a variety of medium access protocols and signaling methods are available in the literature. A survey of medium access protocols can be found in [Toba 80], [Toba 87], and [RuBa 90].

The simplest medium access protocol is pure ALOHA, which is used in ALOHANET as described in section 2.2. This network was modelled by Abrahamson ([AbKu 73] and [Abra 70]). The network throughput was evaluated considering homogeneous packet generation generated at exponentially distributed time intervals. The maximum channel efficiency is 18 per cent.

A slotted version of ALOHA has the advantage of better efficiency, since the transmission starts only at discrete time intervals. Models of slotted ALOHA can be found in [Klei 75] and [Abra 77].

A PR-NET using disciplined ALOHA and spread-spectrum signaling was modeled by Storey and Tobagi (see [StTo 89]). In disciplined ALOHA, a station will start its transmission only if it is not receiving a packet from another station. According to their model, the network is represented by a continuous time Markov chain, whose state at time  $t$  (denoted as  $Z(t)$ ) corresponds to the number of stations receiving packets and the number of stations transmitting a packet. Several approximations were made to allow numerical results to be derived. The maximum throughput of this network was given as a function of the offered traffic and number

of stations. A similar approach was used by McKenney (see [McBa 91]) to provide a model which accounts for the effects of interference, acknowledgement traffic, and retransmission policy.

Carrier Sense Multiple Access (CSMA) and its variations is widely used in packet radio networks (in DARPA PR-NET for example). A packet radio network using CSMA is modelled and analysed by Kleinrock and Tobagi (see [KlTo 75]). The model assumes an infinite number of stations generating packets at exponentially distributed time intervals of mean  $1/\lambda$ . Clearly, this assumption is violated in these protocols, however simulation results show that it is a good approximation for aggregate measures. The throughput  $S$  is solved in terms of the offered traffic rate  $G$ , taking into account the probability of a successful transmission for a given packet knowing that it takes  $T$  seconds to be transmitted.

### 2.3.2 Modelling centrally controlled packet radio networks

Centrally controlled networks can be considered as a particular case of centralised networks in which remote stations require explicit information regarding the right to access the shared resource. In such networks, a central scheduler is responsible for assigning the shared resources (i.e. channels) to different users (i.e. stations). Observe that some centralised networks, such as ALOHANET, have a medium access protocol which is not centrally controlled.

The centrally controlled technique most used for packet oriented systems is polling. The central station sends a polling packet to each remote station in sequential order asking if there is a message to transmit. The polled station sends a message if there is one in its buffer or an absence reply otherwise.

A packet radio network using a polling technique as medium access control is analysed by Tobagi and Kleinrock (see [ToKl 76]). The network consists of  $M$  identical stations being polled by a central station. Packets arrive at a station and

are queued in its buffer. Stations are polled by a central station at which time their buffer is completely emptied. Their analysis is based on a previous work presented by Konheim and Meister (see [KoMe 72]) who analysed such a system deriving stationary distributions for queue lengths and waiting times.

In [ZhPa 90] an integrated voice/data system for mobile radio networks using polling is proposed. This system consists of a number of voice and data terminals, and a central station which coordinates the transmissions. For a two way communication, the radio channel is divided into two frequency bands. One band is for downlink communication in which Time Division Multiplexing (TDM) is used, and the other band is for the uplink communication in which framed polling is used.

TDM is a transmission scheme in which different packet sources are assigned to different time slots within a block of information (called *frame*). In this case, the central station can send packets to voice and data users in the same channel by allocating a specific time slot to them in each downlink frame.

The uplink channel operates in a similar way, however the right to access a slot is coordinated by the central station which polls the remote stations and gives permission to transmit their packets.

The mean queuing delay was obtained by representing the remote station buffer as a M/D/1 queue. An more accurate method was also presented using a M/G/1 queuing model. Results were shown to be very accurate compared with the ones obtained by simulations.

### 2.3.3 Modelling multi-channel packet radio networks

A single hop multi-channel packet radio network using slotted ALOHA as a channel access protocol is analysed in [Purs 89]. The network has a fixed number of channels available for packet transmission. Three transmission schemes are considered. In the first one each station is assigned to a specific channel, and all of its transmissions

take place on this channel only. The second scheme allows each station to make an independent random selection of a channel for each time slot. In the third scheme each transmission hops over the set of channels during the transmission of a packet. Delay and throughput are obtained for all three methods using simulation. Results show that fixed assignments provide better throughput for a given delay, but if the propagation and processing delays are large, channel hopping provides higher throughput.

A multi-channel CSMA/CD (Carrier Sense Multiple Access with Collision Detection) packet radio network is modelled and analysed in [ChLy 91]. The system consists of  $N$  nodes interconnected through  $M$  sub-channels, each of which is susceptible to noise. A terminal that has a packet to send selects a channel among the idle ones and attempts transmission on it following the non-persistent CSMA/CD protocol. Each station can be in one of three states: idle, transmitting or backlogged (i.e., it has failed its transmission due to collision or corruption by noise). The system is represented by a two-dimensional Markov chain where each state is given by two random variables which describe the number of transmitting and backlogged stations respectively. The average delay from the packet generation to its successful reception is derived.

#### 2.3.4 Modelling cellular networks

Cellular Mobile Phone Service networks have received considerable attention in the last few years and consequently a great number of results concerning their performance are available. In CMPS networks the main performance parameter of interest is the blocking probability of a given call. The evaluation of this parameter depends mainly on the type of channel assignment policy adopted. Channel assignment policy refers to the way that the available channels are distributed among the cells.

Basically, two different types of channel assignment schemes have been suggested

in the literature: the Fixed Channel Assignment (FCA) and the Dynamic Channel Assignment (DCA).

In the fixed-frequency method the total number of frequencies available is divided into  $m$  separate groups, each with  $j$  frequency channels. A cell is assigned to one of the  $m$  groups. Each base station can communicate with as many as  $j$  mobile units in its cell simultaneously. The number  $m$  of groups is chosen as the smallest number that satisfies the interference requirements. A method to evaluate the blocking probabilities in FCA networks is found in [Schi 70].

The dynamic channel assignment policy assumes that a new call will be blocked only if there is no possible reallocation of channels to calls, including reallocating the calls in progress. A mixed method using analytical results and simulation to evaluate the blocking probabilities can be found in [EvDa 89]. In this analysis, the state of the system of  $m$  cells is represented by a vector of random variables  $\mathbf{N} = [N_1, N_2, \dots, N_m]$ , where  $N_i$  represents the number of calls in progress in cell  $i$ . It is shown that this system has a product form solution. However it is difficult to calculate the blocking probabilities directly from the analytic solution due to the large number of states and the unusual shape of the boundary of the state space. A Monte-Carlo acceptance/rejection technique was found to be the most efficient method of evaluating the blocking probabilities from the product-form solution. According to this method, a state formed by  $m$  independent Poisson random variables is generated. This state is accepted if it belongs to the state space, otherwise it is rejected.

### Modelling Digital Data Cellular Networks

Very few results concerning Digital data cellular networks are available in the literature. However the performance of a multiple-channel cellular network using CSMA is presented in [WiRa 85]. The normalised mean delay against the system bandwidth

utilisation is derived under the CSMA protocol ([KlTo 75]).

## 2.4 A proposed multi-channel cellular network

In this section, we propose a new digital data cellular network (referred to as *MCN*). This will enable multiple packet transmission within the same area by using slotted, multi-channel radio links. We describe this network in terms of its architecture, its medium access protocol and its buffer administration technique.

The quantitative models introduced in this thesis are tailored to the configuration presented here, although they are flexible enough to be adapted to networks of different protocols and buffer administration techniques.

### MCN architecture

The network of stations considered here consists of a group of Base Stations (BS) and of Portable Stations (PS). A typical portable station can be either a laptop computer or an electronic point-of-sale. The base stations are devices which are connected to a wired packet exchange network. They can communicate with the portable stations via multi-channel radio links. A typical transaction would be a laptop having access to a remote host computer or a point-of-sale consulting a central data-base.

Base stations are located in a cellular pattern to provide continuous coverage over a wide area. The topology inside each cell is centralised, i.e., the portables communicate only with the base station and vice-versa. The base station is responsible for forwarding their messages to their final destinations. No direct link between portables is available, however their communication is possible using the base station as an intermediate node.

The cellular technique of spatial frequency reuse is employed, which implies that the total bandwidth is divided into groups of frequencies and these groups are

assigned to the cells. This problem of selecting and allocating frequencies to the cells is a design issue as described in chapter 1 and discussed in more detail in chapter 6.

To illustrate how this technique could be implemented we give an example of a seven-cell reuse pattern (this number is adopted by the AMPS cellular network, as showed in [Macd 79]). Suppose that the total bandwidth supports 35 frequencies. We divide them in seven groups, namely  $A, B, C, D, E, F$  and  $G$ , each of them containing 5 frequencies. Supposing uniform traffic, the base station in each cell would have 5 frequencies for data and control packet exchange in both directions. The resulting cellular lay-out is given in Fig. 2.10. We observe that the whole area is covered by clusters of seven cells repeated in a regular pattern.

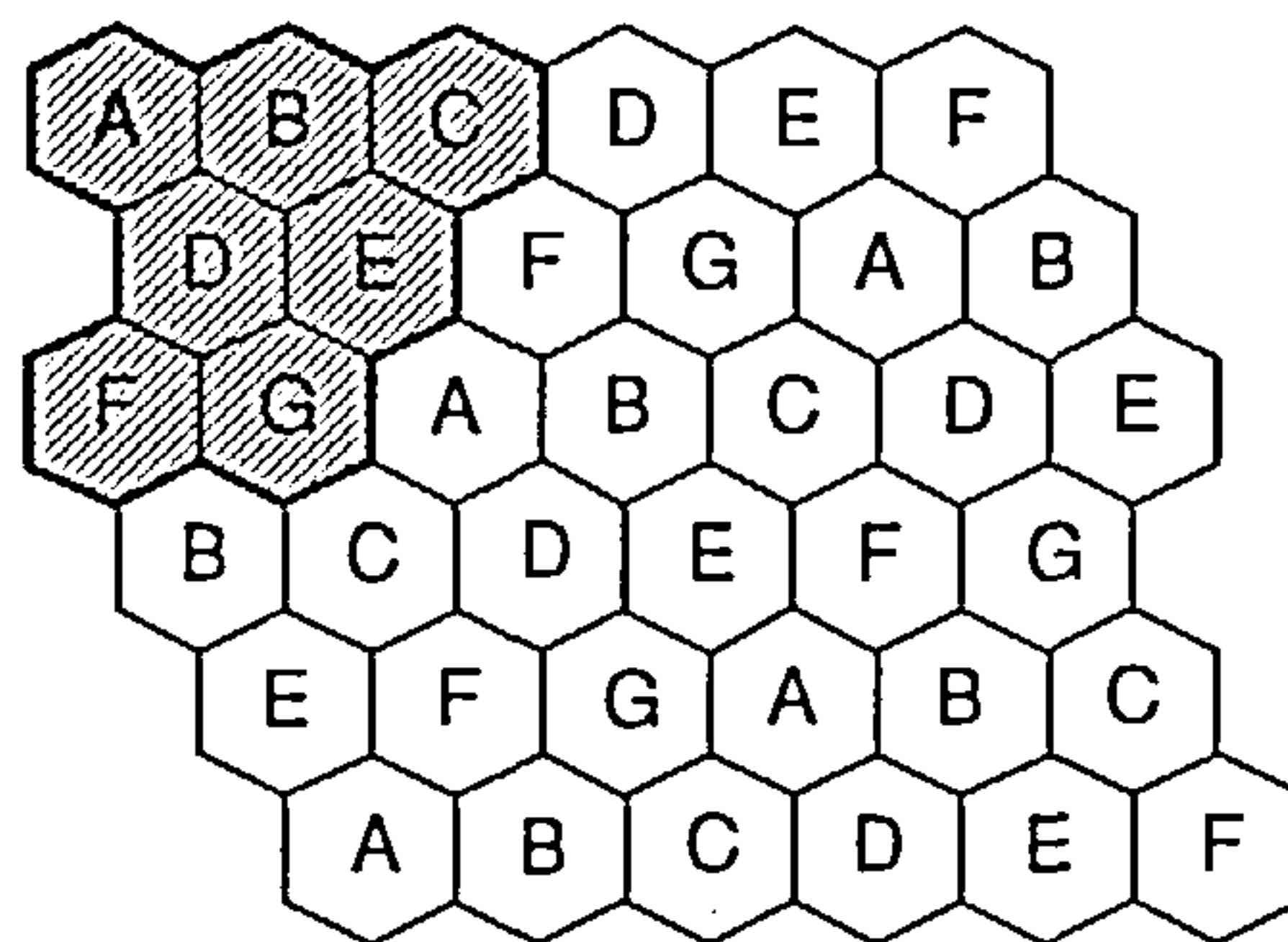


Figure 2.10: The seven cell reuse lay-out.

In order to avoid adjacent channel interference, each group contains the largest possible frequency separation. Suppose that in the above example the channels are numbered sequentially from  $c_1$  to  $c_{35}$ . In this case, group  $A$  contains the frequencies  $c_1, c_8, \dots, c_{29}$ . The other groups are formed in a similar way.

### MCN Medium Access Protocol

The novel medium access protocol described here is a combination of a multi-channel TDMA and polling. It can be regarded as an extension of the DSRA protocol described in section 2.2.5. It is a centrally controlled protocol, since the right to

access each channel is controlled by the base station.

The allocated group of channels is divided into three sub-groups: control, uplink data transfer and downlink data transfer channels. In all channels, packets are transmitted at discrete time intervals referred to as *time slots*.

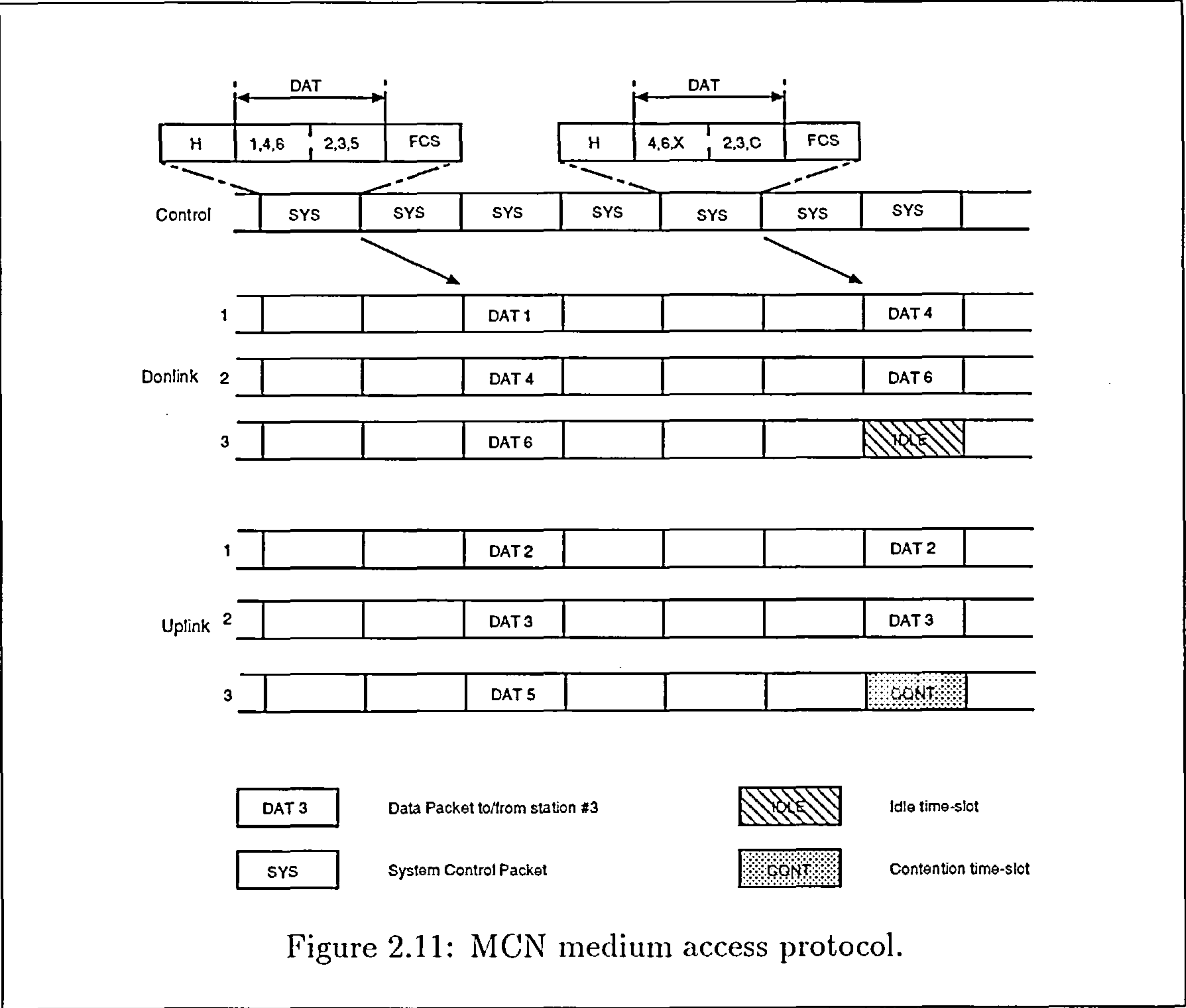
The control sub-group contains one broadcast channel used to transmit system packets (SYS) from the base station to portables. The SYS packet contains a Header (H), the Data transfer Assignment Table (DAT) and a Frame Checking Sequence (FCS).

The Header contains a synchronisation preamble and the base station identification. The synchronisation preamble allows the portable and the base station to operate with the same timing. The base station identification informs the portable location, and gives the information regarding the number of available channels in the cell and its corresponding frequencies.

The Data transfer Assignment Table is divided into two parts. The first part informs which portables are addressed in the downlink transmission during the next time-slot. This allows the selected portables to tune in the corresponding frequencies and receive the packet. The second part informs which stations can access the uplink channels in the next time-slot (it is the equivalent of a poll command). If the portable has a packet to transmit it tunes to the channel selected and transmits its message; otherwise a null reply is sent.

The base station reserves a *contention time-slot* in certain periods. This is used when a new portable has moved into a cell and it wants to inform of its existence. Since more than one station can transmit in the same contention slot, the sending station waits for an acknowledgement. After receiving a correct acknowledgement, the portable can wait for a time-slot dedicated to it in the downlink channels.

Figure 2.11 gives an example of a cell containing 3 channels for uplink, 3 channels for downlink data transfer and 6 portable stations.



We can see that the contents of the first DAT shown in fig. 2.11 indicate that the mobile stations number 1,4 and 6 are addressed during the next time slot in the downlink channels number 1,2 and 3 respectively. The DAT also indicates that the mobile stations number 2,3 and 5 are allowed to transmit a packet during the next time slot. The selected mobile stations (i.e. the stations number 2,3, and 5) start their transmission in the uplink channels number 1,2 and 3 respectively.

If there is no packet to be transmitted to a mobile station the DAT contain an X. In the same way, if no mobile station is selected to transmit a packet the DAT will contain the code C, which indicates a contention time slot.

This protocol has been designed to provide a good channel utilisation, since the base station can minimise the number of idle time slots by allocating different packets

to the channels. The base station can also control how frequently the stations are polled, providing in this way a dynamic traffic control.

### Buffer Administration Technique

Finally, we describe the downlink buffer administration technique, which tells how the base station incoming packets are stored and which channels are allocated for their forwarding.

According to the medium access protocol adopted, the base station controls the transmission in both uplink and downlink sub-groups. We assume that the portable stations have only one transceiver which can be tuned into more than one frequency, both for transmitting and receiving a packet. Consequently, a station can not receive and transmit a packet simultaneously. Moreover, the base station can not use more than one channel at the same time to transmit packets to the same destination, since the portable can not tune into more than one channel at the same time.

In order to prevent address repetition in a given time slot, the incoming packets in the base station are separated in different queues. There is one queue for each destination address (the mobile stations). The base station contains a *channel controller* which is responsible for these queues' administration.

The channel controller operates at discrete time intervals. At the beginning of each time interval a group of transmitters (associated with the downlink channels) are available for the downlink operation. The channel controller starts visiting the queues in cyclic order. If the visited queue is empty, then the channel controller immediately visits the next queue in sequence, otherwise the packet is removed and allocated to the first transmitter available.

This operation is repeated until either all the transmitters have been filled or all queues have been visited. Once one of these conditions is met, the channel controller stops (pointing to the last queue visited) and the transmitters are activated.

This allocation policy prevents two packets from the same queue being transmitted at the same time. Fig 2.12 illustrates the channel controller operation.

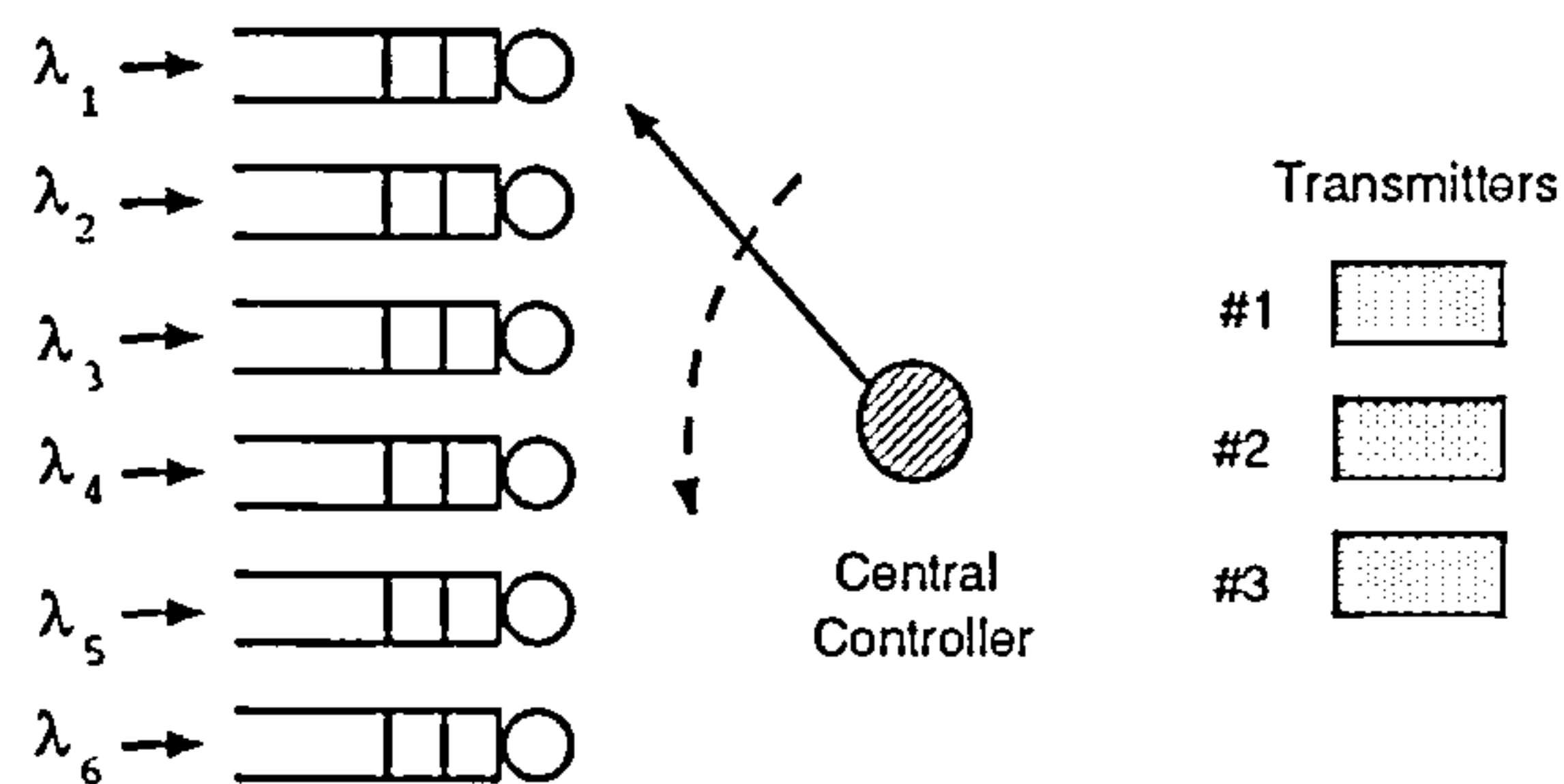


Figure 2.12: MCN buffer administration technique.

We also propose in chapter 6 two alternative techniques called *channel split* and *reservation* respectively. They also prevent address repetition by separating the incoming packets in different queues and by assigning their packets to different channels. However, they differ by the way that these queues are organised and the way the packets are assigned to the channels. We compare their performance and highlight their advantages and disadvantages.

### Inter-cell communication

A mobile station can also communicate with another station outside its own cell. In this case, this mobile station specifies the remote destination in the data packet and the base station forward it to another cell.

# Chapter 3

## Queue Length Distribution in the Base Station

### 3.1 Introduction

As we have seen in chapter 1, the base station in centralised networks is responsible for local traffic control and channel management. Performance measures such as channel throughput and queue length distribution are closely related to how the packets are stored and forwarded, and how the channels are allocated.

In this chapter we carry out a closer investigation of the base station management schemes by introducing a discrete-time Markov chain model for its multichannel communication in a centralised cellular network. The method employed assumes a group of parallel queues in which the inter-arrival time in each queue is exponentially distributed. These queues are visited in a cyclic order at discrete time instants by a channel controller. The channel controller is responsible for assigning packets to the transmit buffers associated with the radio channels.

We consider a new situation in which no more than one customer from each visited queue is served. Furthermore, in order to avoid overlapping, a given queue is not visited more than once at each discrete time interval. Overlapping is undesirable,

as discussed in the previous chapter, since the queues are organised according to final address destination. Assigning more than one packet from the same queue to different transmit buffers would result in a simultaneous transmission to the same destination, which has only one data receiver.

We represent this system by a Markov chain in which the states correspond to the number of packets in each queue and the position of the channel controller at the beginning of each time interval. A set of linear equations is produced from the corresponding balance equations which are solved numerically. The queue length distribution and the channel throughput immediately follow from the solution of these equations.

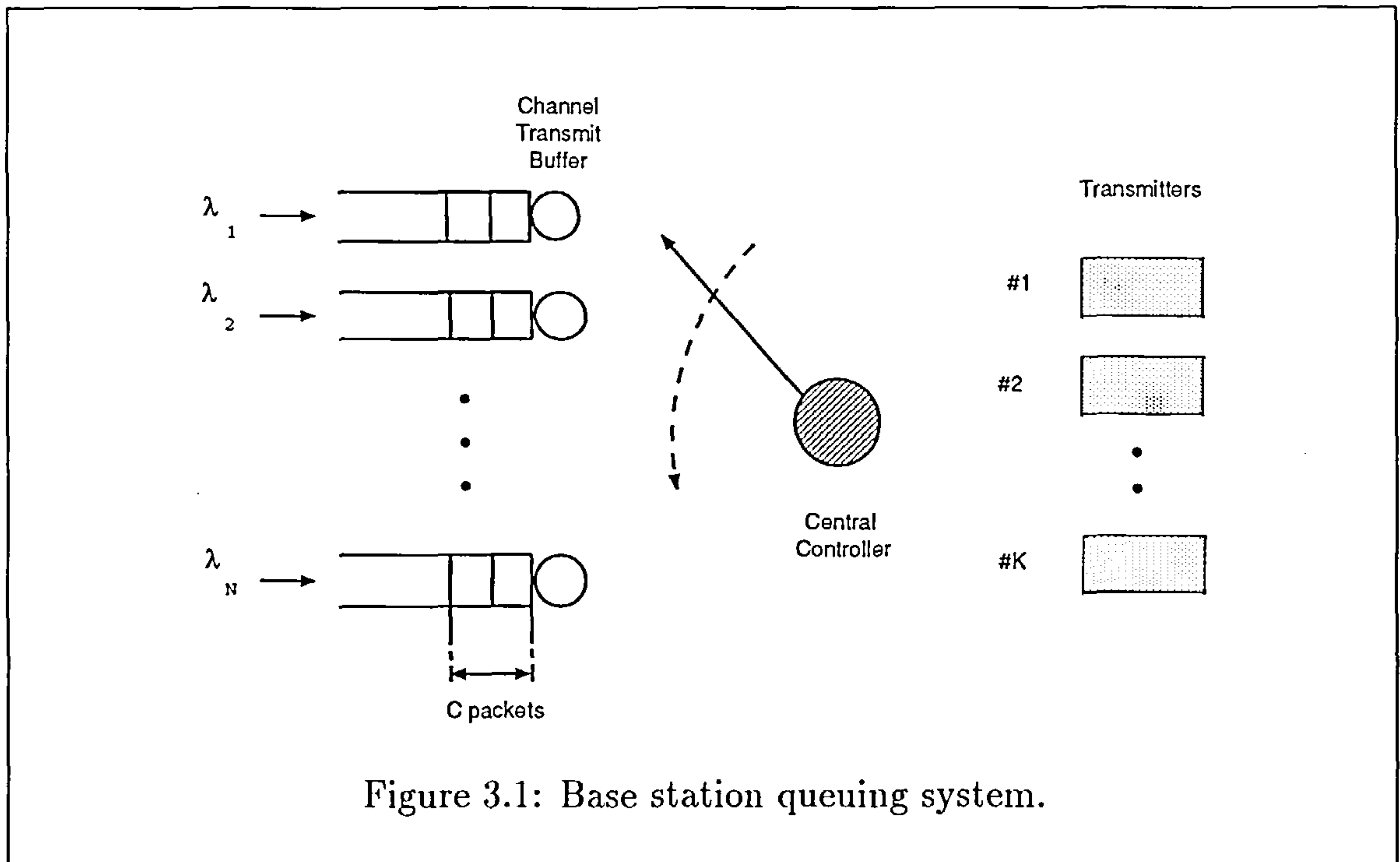
We compare the analytical results with the results obtained using simulation. We observe that validation of this model with respect to simulation indicates a high degree of accuracy.

## 3.2 Base Station Operation

We consider a queuing system with  $N$  parallel queues  $Q_1, \dots, Q_N$  (see figure 3.1). Each queue can store up to  $C$  packets (its customers). Packets are discarded if the queue is full. We assume that the effects of packet loss due to queue overflow are negligible.

The packets in each queue wait to be placed in a *channel transmit buffer*, which effectively is the head of the queue. These queues are served on a first-come-first-served basis. This system is regarded as  $N$  independent external Poisson streams, generating packets to the parallel queues, each one with arrival rate  $\lambda_i$ , ( $1 \leq i \leq N$ ).

A *channel controller* operates at discrete time intervals  $\tau_x$  ( $x = 0, 1, 2, \dots$ ) called *time-slots*. It is assumed that the time-slot duration  $\theta$  is greater than the transmission time of each packet. At each time  $\tau_x$  a group of  $K$  transmitters are available for simultaneous operation. The channel controller starts visiting the queues in cyclic



order. If the channel transmit buffer of the visited queue is empty, then the channel controller immediately visits the next queue in sequence, otherwise the first packet in the queue is removed and allocated to the first transmitter available.

The channel controller repeats this operation until either all the transmitters have been filled or all queues have been visited. Once one of these conditions is met, the channel controller stops (pointing to the last queue visited) and the transmitters are activated. At time  $\tau_{x+1}$  the first queue to be visited is the next in sequence (this queue is referred to as the *starting point*). We suppose that the time spent during the queue scanning operation is negligible compared with the time-slot duration.

### 3.3 Model description

This system can be represented by a discrete-time Markov chain, whose state is  $(i, \mathbf{n})$ , where  $i$  ( $1 \leq i \leq N$ ) is the starting point queue, and  $\mathbf{n} \in S = \{n_i | 0 \leq n_i \leq C, 1 \leq i \leq N\}$  is a vector (vectors are shown in bold type) containing the number of customers in each queue observed immediately before the channel controller

operation.

Let  $p_i(\mathbf{n})$  be the steady state probability that the Markov chain is in the state  $(i, \mathbf{n})$  and  $t_{ji}(\mathbf{m}, \mathbf{n})$  be the transition probability from the state  $(j, \mathbf{m})$  to the state  $(i, \mathbf{n})$ . Clearly, the balance equations of this Markov chain are:

$$p_i(\mathbf{n}) = \sum_{j=1}^N \sum_{\mathbf{m} \in S} t_{ji}(\mathbf{m}, \mathbf{n}) p_j(\mathbf{m}) \quad (3.1)$$

We observe that the probabilities  $p_i(\mathbf{n})$  are related to each queue length at the time immediately prior to the channel controller operation.

Our approach assumes that transitions in the Markov chain correspond to an alteration of the system state between two discrete time intervals  $\tau_x$  and  $\tau_{x+1}$  ( $x = 0, 1, \dots$ ). The system state changes as an effect of the channel controller operation, which allocate packets to the transmitters, and as an effect of arrivals in the system.

In figure 3.2 we show an example of the Markov chain model of a system comprising 4 queues (i.e.  $N = 4$ ) of capacity  $C = 10$  and 2 transmitters (i.e.  $K = 2$ ). State  $S_1$  indicates that queue 2 is the starting point, and queues 1 to 4 have 4, 3, 2 and 1 packets respectively. A transition from state  $S_1$  to state  $S_2$  occurs whenever there are no arrivals in the system (this probability is given by  $A(0, 0, 0, 0)$ ). Observe that in this case packets from queues 2 and 3 were allocated to the transmitters.

Notice that there is no transition between  $S_1$  and  $S_3$  and vice-versa since they have incompatible starting points.

## 3.4 Steady state solution

We are interested in finding the steady state solution of the Markov chain described in section 3.3 (which we assume to have matrix generator  $P$ ), but we shall first investigate the existence of such solutions. The following proposition points out in which situations these solutions are obtainable.

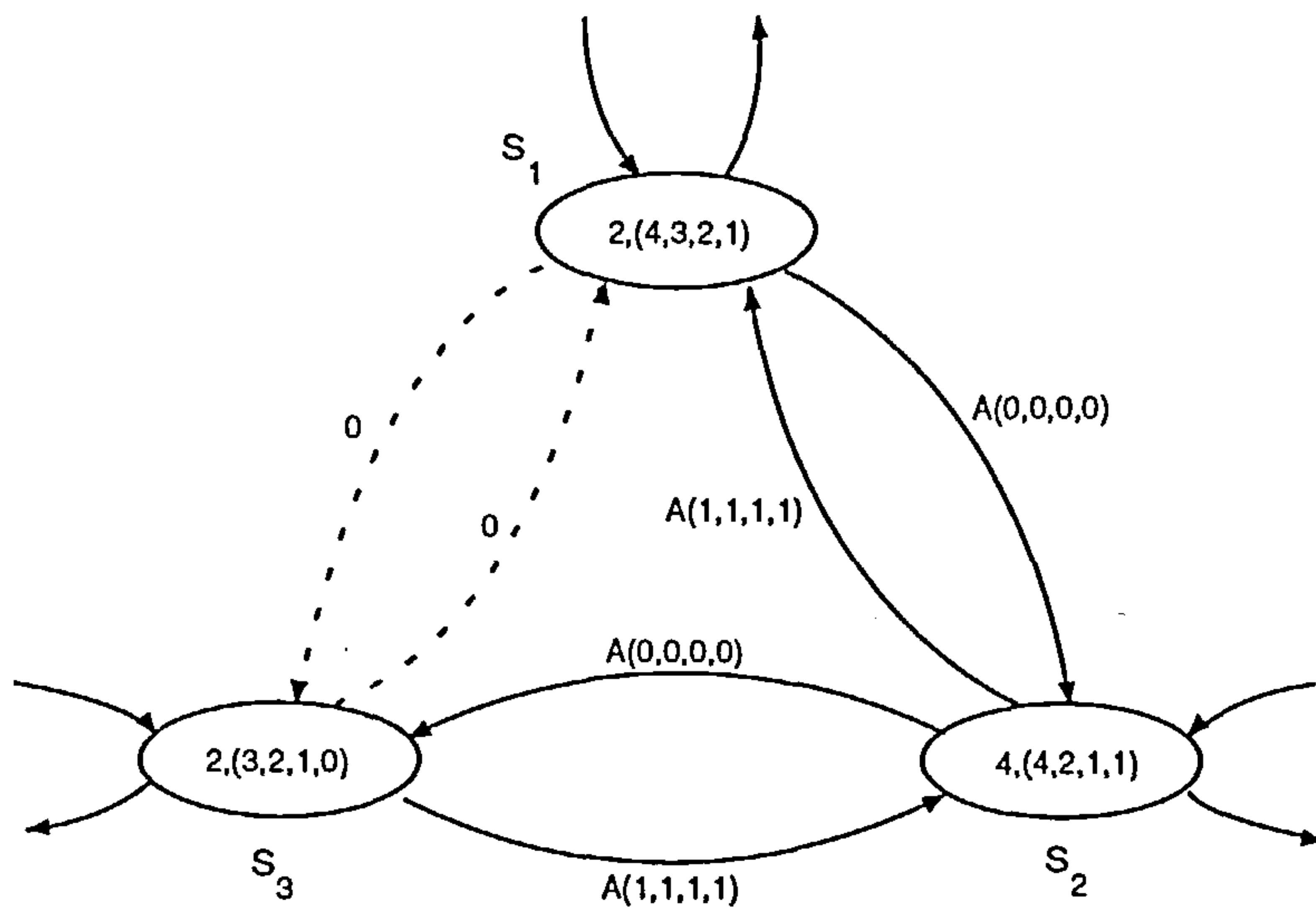


Figure 3.2: Markov chain model.

**Proposition 1** *The Markov chain for the queuing system with  $N$  queues and  $K$  transmitters is irreducible when  $N > K$ .*

**Proof**

Initially, we observe that a state  $(i, \mathbf{0})$  ( $i = 1, \dots, N$ ), where  $\mathbf{0} = (0, \dots, 0)$ , is accessible from any state  $(j, \mathbf{n})$  ( $\mathbf{n} \in S, j = 1, \dots, N$ ) via  $\mathbf{0}$  arrivals in sufficiently many time slots. In the same way, a given state  $(j, \mathbf{m})$  ( $\mathbf{m} \in S$ ) is accessible from any state  $(j, \mathbf{0})$ , via  $\mathbf{m}$  arrivals in a time-slot. Therefore, it suffices to prove that state  $(j, \mathbf{0})$  is accessible from any state  $(i, \mathbf{0})$ .

If  $N > K$  then state  $(i-1, \mathbf{0})$  is accessible from state  $(i, \mathbf{0})$ . There is a transition from  $(i, \mathbf{0})$  to  $(i-1, \mathbf{0})$ , for example, when there is one arrival at queues  $i, i+1, \dots, i+K-1$  and one arrival in queue  $i-2$  (for clarity no special notation is used to denote arithmetic mod  $N$ ).

Therefore state  $(j, \mathbf{0})$  is accessible from  $(i, \mathbf{0})$ , via  $(i-1, \mathbf{0}), (i-2, \mathbf{0}), \dots, (j, \mathbf{0})$ .  
□.

**Remarks**

- This Markov chain is aperiodic since a transition probability from a state  $(i, n)$  to itself is greater than 0 (there is a transition from  $(i, n)$  to itself when there is one arrival in each queue that was found not empty by the channel controller).

### Stability Conditions

We assume that the multi-queue system is stable, so that in particular there is an integer  $C$  such that the probability of having a given queue  $i$  with  $C$  customers is arbitrarily small. The stability conditions must be satisfied both for the channel queuing system and for each queue individually. Hence we require:

1. The average number of arrivals in the system is less than the maximum number of departures during a time-slot, i.e.  $\lambda_0\theta < K$ , where  $\lambda_0 = \sum_{j=1}^N \lambda_j$ .
2. The average number of arrivals in given queue is less than the maximum number of departures in the same queue during a time-slot, i.e. one. Therefore the arrival rates in each queue must satisfy the condition  $\max\{\lambda_i\theta\} < 1$ .

We observe that we consider a closed system with finite state space, therefore there is always a steady state solution. The assumption of stability is to guarantee that the losses will be small for a sufficiently large  $C$ .

### Transition probabilities

We proceed by evaluating the transition probabilities  $t_{ji}(m, n)$ . These probabilities are related to the number of arrivals in the corresponding queuing system between two discrete time intervals  $\tau_x$  and  $\tau_{x+1}$  ( $x = 0, 1, \dots$ ). They are also related to the number of non-empty queues between  $i$  and  $j$  as explained below.

Let the function  $e_{ji}(n)$  represent the number of packets removed from each queue due to the channel controller operation when it finds  $n$  packets in the system and jumps from queue  $j$  to  $i$ .

Let the function  $\Delta_{ji}(\mathbf{m})$  be equal to 1 whenever there is a transition from a state  $(j, \mathbf{m})$  to any state  $(i, \mathbf{n})$  ( $\mathbf{n} \in S$ ), otherwise  $\Delta_{ji}(\mathbf{m})$  is equal to 0.

When there is no packet loss a transition from state  $(j, \mathbf{m})$  to a state  $(i, \mathbf{n})$  (when it is possible) corresponds to  $\mathbf{n} - \mathbf{m} + \mathbf{e}_{ji}(\mathbf{n})$  arrivals in the system. Observe that there is no transition between states with incompatible starting points (i.e. when  $\Delta_{ji}(\mathbf{m})$  is equal to 0).

Since no more than one packet is removed from each queue, the quantities  $\mathbf{e}_{ij}(\mathbf{m}) = \{e_k^{(ij)}(m_k) | 1 \leq k \leq N\}$  are given by:

$$e_k^{(ij)}(\mathbf{m}) = \begin{cases} \delta(m_k) & \text{if } i \leq k < j \\ 0 & \text{otherwise} \end{cases} \quad (3.2)$$

where  $\delta(y)$  is the delta function (i.e.  $\delta(y) = 1$  if  $y = 0$  and  $\delta(y) = 0$  if  $y > 0$ ).

The function  $\Delta_{ji}(\mathbf{m})$  is defined as follows:

- When the number of queues between  $j$  and  $i$  is less than  $K$ ,  $\Delta_{ji}(\mathbf{m}) = 0$ , since the channel controller visits at least  $K$  queues.
- When the number of queues between  $j$  and  $i$  is greater than or equal to  $K$  but  $i \neq j$ , a transition between states with starting points  $i$  and  $j$  is possible only if in the corresponding queuing system there are  $K - 1$  non-empty queues between  $j$  and  $i - 2$  and queue  $i - 1$  is not empty. In this case all  $K$  transmitters have been filled and the channel controller stops pointing to queue  $i$  (next in sequence). Thus, we have:

$$\Delta_{ji}(\mathbf{m}) = \begin{cases} 1 & \text{if } \sum_{k=j}^{i-1} \delta(m_k) = K \\ 0 & \text{otherwise} \end{cases} \quad (3.3)$$

- When  $i$  is equal to  $j$ , a transition between states with same starting point is possible only in two situations: either less than  $K$  queues were found not empty (then the channel controller stops its operation to avoid overlapping)

or exactly  $K$  queues were found not empty and one of them is queue  $i - 1$ .

Thus, we have:

$$\Delta_{ii}(\mathbf{m}) = \begin{cases} 1 & \text{if } \sum_k \delta(m_k) < K \text{ or} \\ & \sum_{k=i}^{i-2} \delta(m_k) = K - 1 \text{ and } \delta(m_{i-1}) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (3.4)$$

Let the arrival function  $A(\mathbf{x})$  describe the joint probability that in one time-slot exactly  $x_i$  packets arrive at queue  $i$  ( $1 \leq i \leq N$ ). Considering that the number of arrivals in a given queue is an independent Poisson distributed random variable, the function  $A(\mathbf{x})$  is given by:

$$A(\mathbf{x}) = \exp(-\lambda_0 \theta) \prod_{j=1}^N \frac{(\lambda_j \theta)^{x_j}}{x_j!} \quad (3.5)$$

Once we have defined these functions we can proceed to obtain the balance equations of the Markov chain model. Thus we have:

$$p_i(\mathbf{n}) = \sum_{j=1}^N \sum_{\mathbf{m}=0}^{\mathbf{u}_{ij}(\mathbf{n})} \sum_{\mathbf{r}=0}^{\mathbf{v}(\mathbf{n})} p_j(\mathbf{m}) A(\mathbf{r} + \mathbf{n} - \mathbf{m} + \mathbf{e}_{ji}(\mathbf{m})) \Delta_{ji}(\mathbf{m}) \quad (3.6)$$

where:

- the elements of the vector  $\mathbf{u}_{ij}(\mathbf{n}) = \{u_k^{(ij)}(n_k) | 1 \leq k \leq N\}$  are defined as:

$$u_k^{(ij)}(\mathbf{n}) = \begin{cases} n_k + \bar{\delta}(n_k - C) & \text{if } j \leq k < i \\ n_k & \text{otherwise} \end{cases} \quad (3.7)$$

where  $\bar{\delta}(i) = 1 - \delta(i)$  ( $i = 0, \dots$ ).

- the quantities  $\mathbf{v}(\mathbf{n}) = \{v_k(n_k) | 1 \leq k \leq N\}$  are given by:

$$v_k(\mathbf{n}) = \begin{cases} \infty & \text{if } n_k = C \\ 0 & \text{otherwise} \end{cases} \quad (3.8)$$

Observe that the vector  $\mathbf{r}$  in equation 3.6 contains the number of packets which are lost due to buffer overflow.

We also have the normalisation equation:

$$\sum_{i=1}^N \sum_{\mathbf{n}=0}^{\mathbf{C}} p_i(\mathbf{n}) = 1 \quad (3.9)$$

where  $\mathbf{C}$  is a vector containing the maximum queue size in all elements.

### 3.5 Performance measures

Once we have obtained  $p_i(\mathbf{n})$  for all values of  $(i, \mathbf{n})$ , using the above equations 3.6 and 3.9, we can proceed to obtain other performance measures such as channel throughput as follows.

We define *offered traffic*  $G$  as being the mean number of arrivals in the system per time-slot, i.e.,  $G = \lambda_0 \theta$ . The *channel throughput*  $S$  is defined as being the expected number of transmitted packets per time-slot.

The number of transmitted packets per time-slot depends on the corresponding state of the Markov chain  $(i, \mathbf{n})$ . Thus, we define the function  $g(\mathbf{n})$  which describes this number when there are  $\mathbf{n}$  customers in the system. Clearly, the number of transmitted packets is equal to the number of non-empty queues (not exceeding the number of transmitters). Hence  $g(\mathbf{n})$  is given by:

$$g(\mathbf{n}) = \min\left\{K, \sum_{i=1}^N \delta(n_i)\right\} \quad (3.10)$$

The channel throughput  $S$  is then the expected value of  $g(\mathbf{n})$ , given by:

$$S = \sum_{i=1}^N \sum_{\mathbf{n}=0}^{\mathbf{C}} p_i(\mathbf{n}) g(\mathbf{n}) \quad (3.11)$$

The queue length distribution, its moments and the average delay are evaluated as follows. First we evaluate the probability that there are  $j$  packets in queue  $i$ , observed *at departure time*, i.e. after channel assignment (which we denote

$\pi_i(j)$  ( $0 \leq j \leq C-1$ ). Notice that  $\pi_i(C) = 0$ , since a packet can not leave  $C$  packets behind. We then evaluate the steady state probability that there are  $j$  packets in queue  $i$  observed at a *random time* (which we denote  $q_i(j)$ ).

Let  $\sigma_i$  be the average number of transmitted packets from queue  $i$  ( $1 \leq i \leq N$ ) per time-slot (i.e. the queue  $i$  throughput). This quantity can be evaluated by summing all the steady state probabilities  $p_j(\mathbf{n})$  in which a packet from queue  $i$  is transmitted:

Let the function  $\varepsilon_{ji}(\mathbf{n})$  be equal to 1 in every state  $(j, \mathbf{n})$  in which a packet from queue  $i$  is removed by the channel controller and allocated to a transmitter (otherwise it is equal to 0). The function  $\varepsilon_{ji}(\mathbf{n})$  is equal to 1 whenever the number of non-empty queues between  $j$  and  $i$  is less or equal to the number of transmitters (i.e.  $K$ ) and queue  $i$  is not empty. Thus we have (for brevity we assume that the sum below takes into account that the queues are numbered circularly):

$$\varepsilon_{ji}(\mathbf{n}) = \begin{cases} 1 & \text{if } \sum_{x=j}^{i-1} \delta(n_x) < K \text{ and } \delta(n_i) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (3.12)$$

The average number of transmitted packets from queue  $i$  (i.e. the probability that a packet is removed from queue  $i$ ) is then given by:

$$\sigma_i = \sum_{j=1}^N \sum_{\mathbf{n}=0}^{\mathbf{C}} p_j(\mathbf{n}) \varepsilon_{ji}(\mathbf{n}) \quad (3.13)$$

The probability that a packet which is removed from queue  $i$ , leaves behind  $j$  packets (which we denote  $\pi_i(j)$ ) is the sum of all states in which there are  $j+1$  packets in queue  $i$  and the packet is effectively transmitted. Thus for  $0 \leq j \leq C-1$  we have:

$$\pi_i(j) = \frac{1}{\sigma_i} \sum_{\mathbf{n}_1=0}^C \dots \sum_{\mathbf{n}_{i-1}=0}^C \sum_{\mathbf{n}_{i+1}=0}^C \dots \sum_{\mathbf{n}_N=0}^C p_i(\mathbf{n}^*) \varepsilon_{ji}(\mathbf{n}^*) \quad (3.14)$$

where  $\mathbf{n}^* = (n_1, \dots, n_{i-1}, j+1, \dots, n_N)$ . The steady state queue length distribution at queue  $i$  observed at random time  $q_i(j)$  can be obtained from the probability

$\pi_i(j)$ . This is done by using a (mild) approximation that the loss probability is state independent (given by  $q_i(C)$ ). Conditioning on only those customers who join the queue, we can apply the results on the  $M/G/1$  queue (see [GrHa 85], [Coop 90]) which states that the distribution observed at departure time is equal to the distribution at arrival time. Thus we have:

$$q_i(j) = \begin{cases} 1 - \frac{\sigma_i}{\lambda_i \theta} & \text{if } j = C \\ \frac{\sigma_i}{\lambda_i \theta} \pi_i(j) & \text{if } 0 \leq j \leq C - 1 \end{cases} \quad (3.15)$$

The mean queue length and waiting time immediately follow from equation 3.15.

### 3.6 Numerical Example

In this section we obtain exact results for a system comprising one transmitter and two queues with arrival rates  $\alpha\lambda_0$  and  $(1 - \alpha)\lambda_0$  respectively ( $0 < \alpha < 1$ ). We assume buffer capacity  $C = 2$ . The corresponding Markov chain has 18 states  $(i, (n_1 n_2))$   $\{i = 0, 1; 0 \leq n_i \leq C\}$ . The transition probability matrix can be found in Appendix A.

We obtain the values of  $p_i(n)$  by solving equations 3.6 and 3.9 numerically. Figs. 3.3 and 3.4 show graphics for the throughput and mean queue length in two cases: when the incoming traffic is *balanced* (i.e., when  $\alpha = 0.5$ ) which we show in continuous line, and when the incoming traffic is *unbalanced* ( $\alpha = 0.1$  in this example) which we show in a dashed line.

Fig. 3.3 represents the absolute and normalised throughput  $S$  versus offered traffic  $G = \lambda_0 \theta$ . The absolute throughput is given by equation 3.11 and the normalised throughput is the ratio  $S/G$ .

We observe that the throughput in both balanced and unbalanced systems increases with the offered traffic. However, the balanced system has a higher throughput since it provides a homogeneous distribution of packets amongst the queues and hence a better use of the system resources.

We also observe that the normalised throughput decreases with the offered traffic. The normalised throughput reflects the percentage of the incoming traffic which is effectively transmitted. When the incoming traffic is high (i.e.  $G$  close to 1) there is more packet loss due to buffer overflow, thus the normalised throughput is lower.

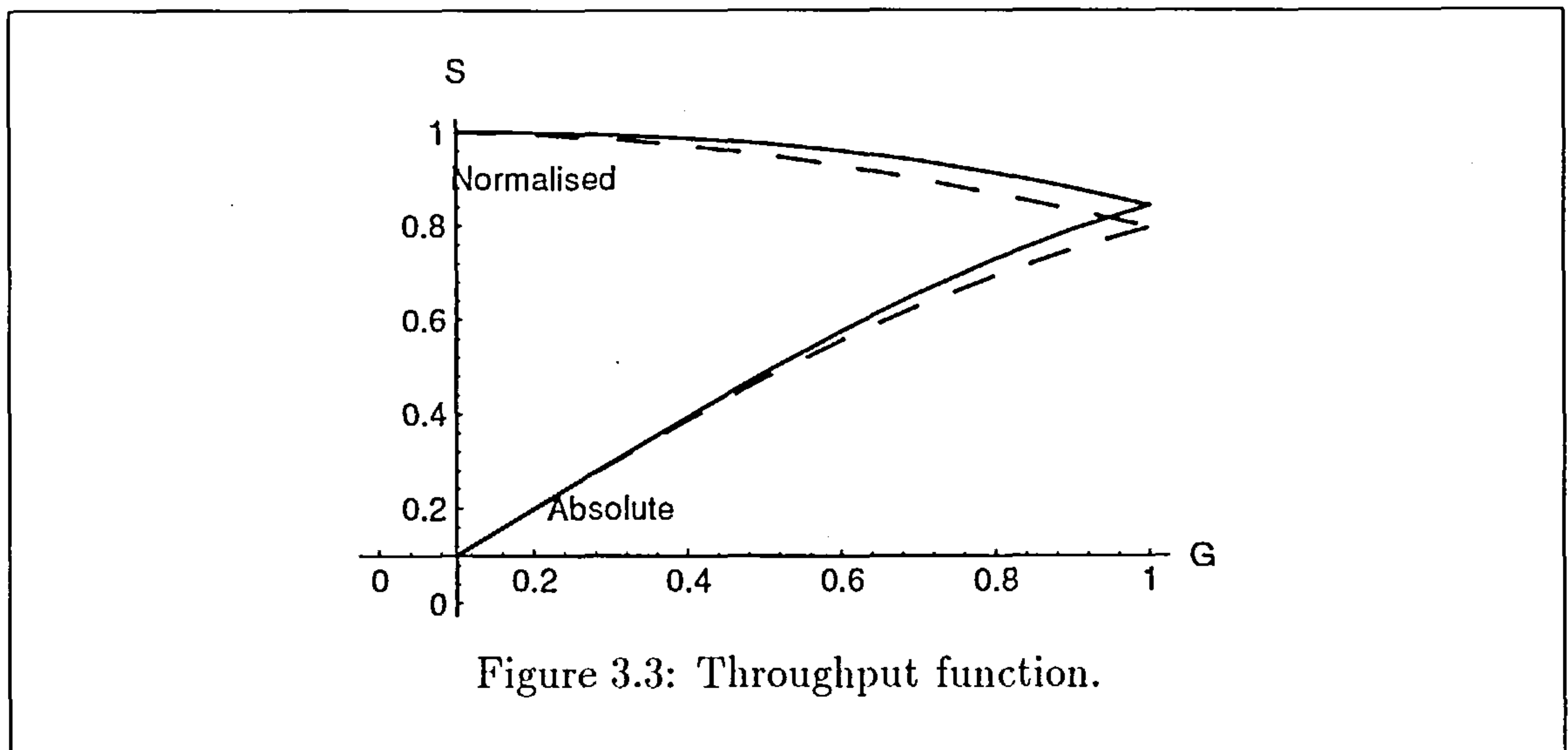


Figure 3.3: Throughput function.

Fig. 3.4 shows the mean queue length versus the offered traffic  $G = \lambda_0 \theta$ . The continuous line represents the results for  $\alpha = 0.5$  (in this case both queues have the same mean). The dashed lines represent the mean queue length of both queues in an unbalanced system with  $\alpha = 0.1$ . We also show the results obtained by simulation (in the balanced system only). Results obtained by simulation agree with the analytical ones for all cases with accuracy of 0.99. Clearly, this is as expected since this is an exact model (except for equation 3.15). In fact we have demonstrated the mutual validation of two models and the simulation program serves to support the correctness of the implementation of the analytical method proposed.

### Numerical Solution of the Markov Chain

The number of states of the Markov chain described in section 3.3 is  $N(C + 1)^N$ , which may be quite large in certain circumstances. A numerical evaluation of its corresponding steady state solution can be carried out by aggregating equivalent

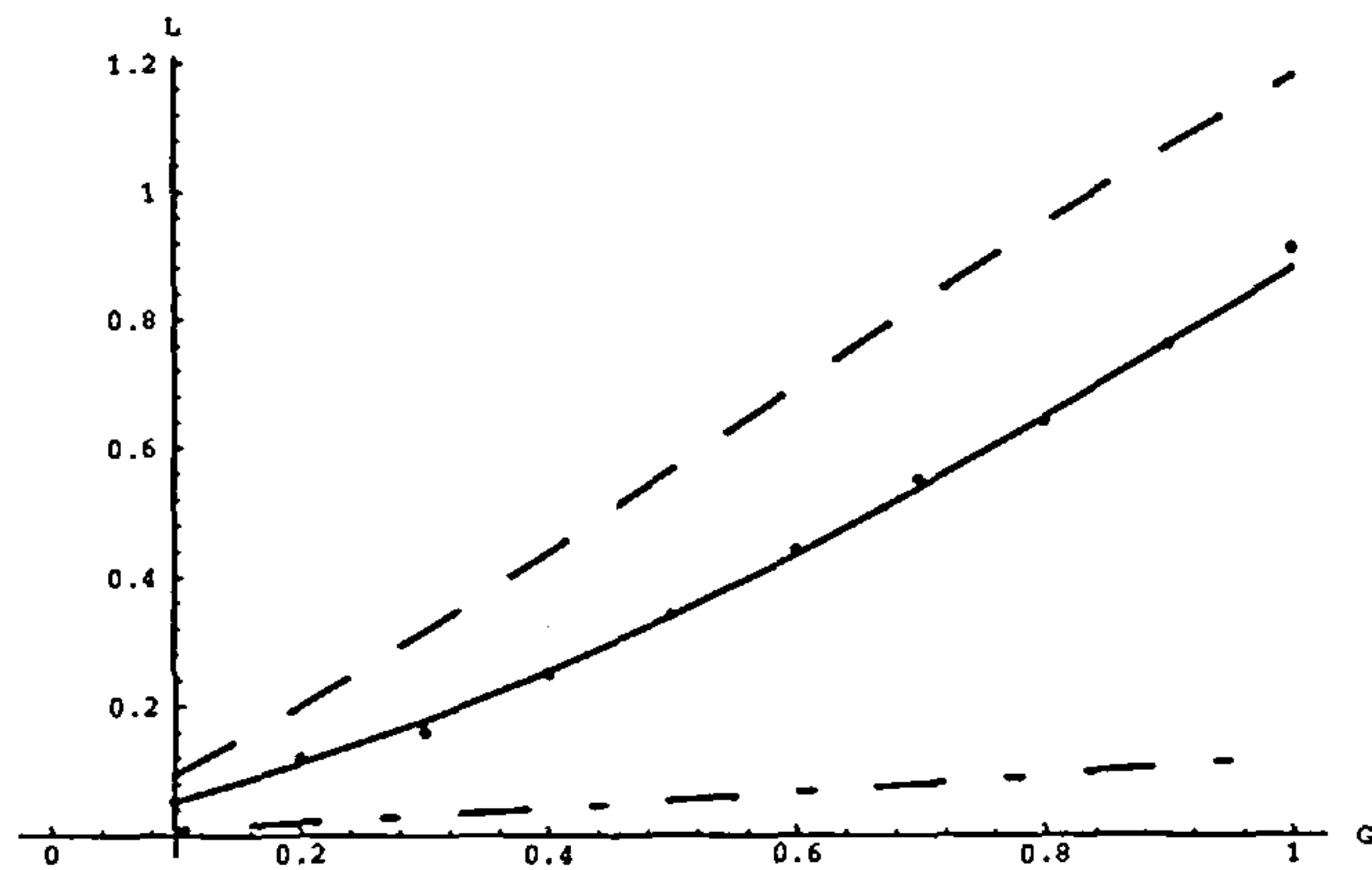


Figure 3.4: Mean Queue Length.

states.

This method is particularly appealing in balanced systems, where the starting point position  $i$  in each state can be eliminated. In fact, by symmetry, we can easily verify that states  $(i, shr(n, i-1))$   $\{1 \leq i \leq N\}$  are equivalent (the function  $shr(n, j)$  shifts circularly the vector  $n$  by  $j$  positions to the right). Therefore we can aggregate the equivalent states, reducing the size of the state transition matrix by a factor of  $N$ .

A further reduction of the transition matrix of both balanced and unbalanced systems is possible by using generalised aggregation-disaggregation methods (see [Stew 91] for a survey). In general, an aggregation of any sub-set of states is possible and the problem of solving a large set of linear equations can be reduced to solving a set of iterative equations of smaller size. The advantage of using these methods is that more efficient algorithms for numerical evaluation can be produced.

We also observe that the state transition matrix is *sparse*, i.e. most of the elements are zero. An efficient algorithm to solve large Markov chains with sparse transition matrices is given in [SaCh 81].

## 3.7 Conclusion

We have presented a Markov chain model of a multi-queue system with a single server (the channel controller) travelling in cyclic order. This system differs from an usual multi-queue system since the service time in each queue depends on the global state of the system. Furthermore, we considered a situation in which the number of the channel controller's visits in a time-slot is determined by the number of non-empty queues and the number of transmitters available. We then obtained the distribution of each queue length and the system throughput by solving the balance equations of this Markov chain. In order to test the analytical solution, we compared its results with the ones obtained by simulation. Simulation results indicate a high degree of accuracy.

# Chapter 4

## Time Delay Distribution in the Base Station

### 4.1 Introduction

A common strategy used in communication networks for the management and access control of one or more transmission channels is polling. These systems usually have analytical models based on several queues visited by a single server in a cyclic arbitration order. The main performance measure of interest in such systems is the sojourn time for the messages at the transmission queues.

Medium access control systems which have channel activity controlled by time-slotted policies, have a discrete inter-visit time distribution. Although some important results concerning waiting and sojourn times in discrete systems are available in the literature (e.g. [Taka 86], [RuBa 90], [KoKo 77]), analytical results are available only in some cases. Some exact results, involving average delays were derived using Kleinrock's *work conservation principle* [Klei 75]. Under this principle the amount of work (i.e., amount of time the server needs to clear all queues) is constant regardless of the service order (whether cyclic or not). Although the waiting time distribution depends on the nature of the arbitration mechanism, the total population and av-

erage waiting time remain invariant. The subsequent *pseudo conservation law* was a generalisation of this principle using a stochastic decomposition of the system's unfinished work. Exact expressions for weighted sums of mean waiting times were derived using this method (see [BoGr 88]). Several other methods have been developed for computing the mean delay, the mean queue length for each queue, the amount of work of the server and the cycle time. A survey of these methods is given by Levy and Sidi (see [LeSi 90]).

This chapter tackles the problem of finding an approximate method of evaluating the waiting time distribution for multi-queue systems with discrete service time. It also considers the sojourn time distribution of systems with exceptional first service times. A direct application for such systems can be found in time-slotted medium access protocols. In these systems several traffic sources compete to use a group of transmission channels. These transmitters are available only at the beginning of fixed length time-slots. At each discrete interval a single server starts visiting queues in cyclic order and assigning packets to the transmitters.

In order to evaluate these distributions, we introduce a new way of looking at the server. Our approach is based on the fact that the server inter-visit time can be regarded as a random variable represented by transitions in a discrete-time Markov chain. Each state of this Markov chain represents the server position at the beginning of a time-slot. We show how to construct the transition probability matrix of this Markov chain and then derive a recursive matrix equation that yields the probability mass function of the inter-visit time. Based on this equation we obtain an equivalent phase-type system that produces the same distribution, following the modelling approach presented by M. F. Neuts (see [Neut 81], [Neut 77], [Neut 79]). This approach is particularly suitable for our problem, since it produces a compact matrix formula for the Laplace transform of the sojourn time distribution, from which its moments and the distribution itself follow. Furthermore, these equations

are easily converted into iterative procedures for numerical evaluation. These procedures can be written down straight from the equations, producing results of great accuracy since they are based on the expansion of rapidly convergent series.

However, although the inter-visit time distribution is of phase-type, the waiting (or queueing) and sojourn time distributions are not in the discrete case. Therefore, we use Neuts' results wherever possible, but also derive new theoretical results for the discrete versions of certain theorems. We then use these results to produce equations for the residual service time distribution, the waiting and sojourn time distribution and formulas for their respective moments. We show how these equations can be translated into algorithms for numerical evaluation and apply them in some examples.

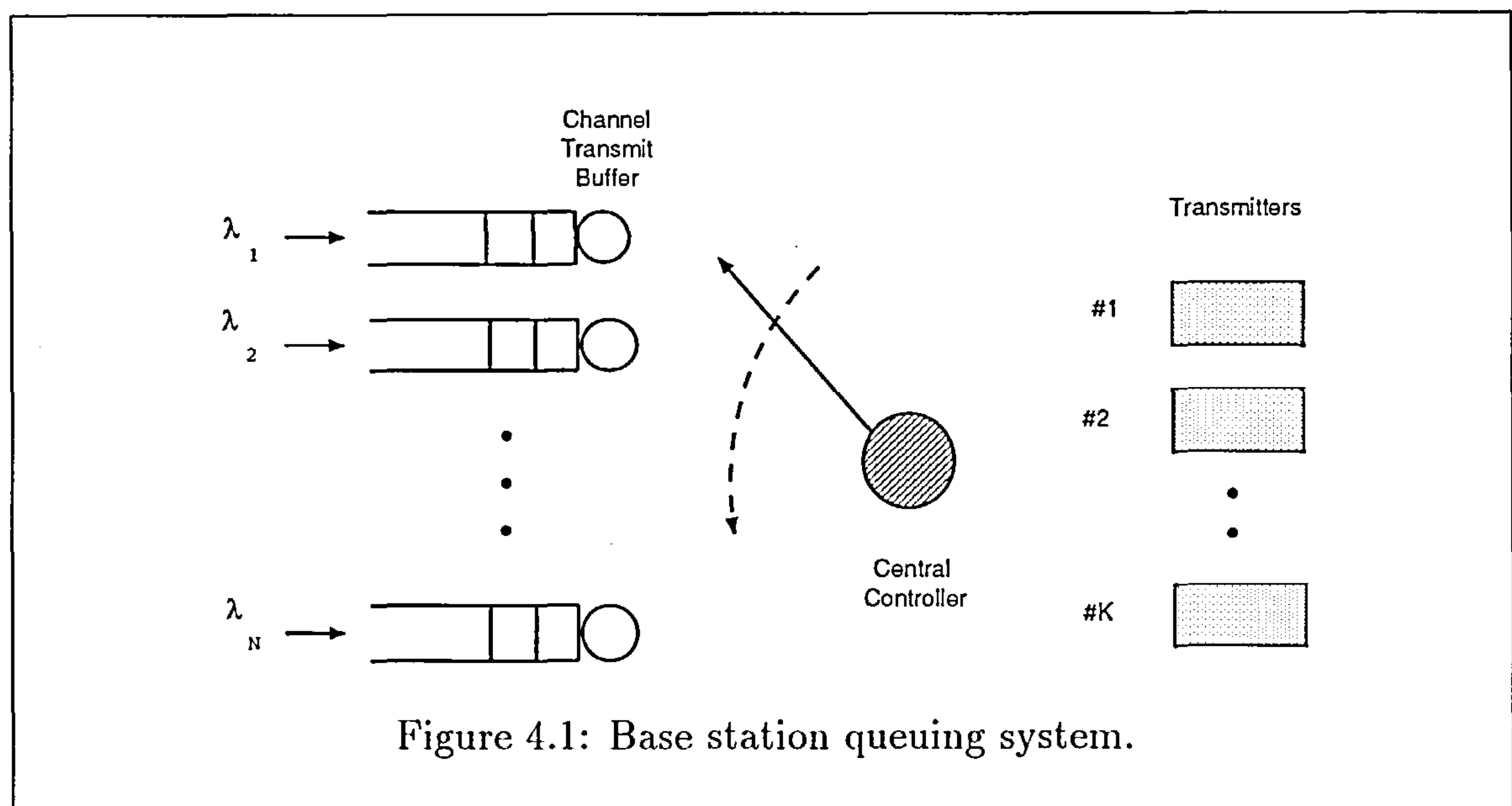
The chapter is organised as follows: section 4.2 describes the model to be analysed and the assumptions to be considered; section 4.3 reviews basic properties of the phase-type renewal process, and introduces two new theorems that are used in our analysis; section 4.4 presents a model of the system considered in this thesis, based on a discrete Markov chain. This section also shows how this chain can be mapped into a phase-type probability distribution. Two new theorems are introduced. They extend the results presented in [Neut 81] for the residual service time distribution and waiting time distribution. The same solution is presented in algorithmic form in section 4.5 with numerical results being given in section 4.6. Finally, in section 4.7 we give our conclusion.

## 4.2 Model description

We consider a queuing system with  $N$  parallel queues  $(Q_1, \dots, Q_N)$ , each one with infinite buffer capacity to store packets, i.e. its customers (see fig 4.1). Clearly, if the buffer capacity is big enough in the actual system, i.e. the probability of packet loss is small, this is a good approximation.

The packets in each queue wait to be placed in a *channel transmit buffer* (which effectively is the head of their queue), administered by a *channel controller* on a first-come-first-served basis. This system can be regarded as  $N$  independent external Poisson streams, generating packets to the parallel queues, each one with an arrival rate of  $\lambda_i$ , ( $1 \leq i \leq N$ ).

The channel controller operates at discrete time intervals  $\tau_n$  ( $n = 0, 1, 2, \dots$ ) called *time-slots*. As in the previous chapter, it is assumed that the interval between time-slots is greater than the transmission time of each packet. At each time  $\tau_n$ , a group of  $K$  transmitters are available for simultaneous operation. The channel controller starts visiting the queues in cyclic order. If the channel transmit buffer of the visited queue is empty, then the channel controller immediately visits the next queue, otherwise the packet is removed and allocated to the first transmitter available.



As described in chapter 3, the channel controller repeats this operation until either all the transmitters have been filled or all queues have been visited. At time  $\tau_{n+1}$  the first queue to be visited is the next in sequence. We suppose that the time spent during the queue scanning operation is negligible compared with the time-slot

duration.

This allocation policy prevents two packets from the same queue being transmitted at the same time, as required in our model. We have seen that packets from the same queue generally go to the same destination (the mobile station). This destination is usually unable to receive more than one packet per time-slot. However, the model is easy to adapt to other protocols, for example where the transmitters are allocated until they or all waiting packets are exhausted.

We consider *sojourn time* as being the time spent by a packet in the queue (including the time spent in the channel transmit buffer), together with *service time* to be the time spent in the channel transmit buffer.

### 4.3 Probability distributions of phase-type

The class of probability distributions of phase-type and some of its properties are summarised in this section. An extensive and comprehensive treatment of this subject can be found in [Neut 81]. We also introduce two new theorems, that will be used in section 4.4.

Consider a discrete Markov chain with  $(m + 1)$  states (called *phases*) and transition probability matrix given by:

$$P = \begin{pmatrix} S & \mathbf{S}^\circ \\ \mathbf{0}' & 1 \end{pmatrix} \quad (4.1)$$

where  $S$  is a  $(m \times m)$  sub-stochastic matrix, such that  $I - S$  is non-singular,  $\mathbf{S}^\circ$  is a column vector such that  $P$  is stochastic and  $\mathbf{0}'$  is the null vector. We denote column vectors in bold type and row vectors also in bold type with a prime operator. Obviously the  $(m+1)$  state is absorbing. We suppose that once an absorption occurs, the process is immediately re-started in state  $i$  ( $1 \leq i \leq m + 1$ ) with probability  $\beta_i$ . Thus, the initial probability vector is  $(\beta', \beta_{m+1})$ , where  $\beta' = (\beta_1 \dots \beta_m)$ .

The probability distribution  $\{p_k\}$  (called *phase-type* or *PH*) of the number of steps  $\mathcal{N}$  until absorption has representation  $(\beta', S)$  and is given by:

$$p_k = \begin{cases} \beta_{m+1} & k = 0 \\ \beta' S^{k-1} \mathbf{S}^\circ & k = 1, 2, \dots \end{cases} \quad (4.2)$$

Note that  $\mathbf{S}^\circ$  is uniquely determined by  $S$ . In the analysis presented here we will assume  $\beta_{m+1} = 0$ . For this case, the probability generating function, Laplace transform (i.e. the expectation  $E[e^{-s\mathcal{N}}]$ ) and the factorial moments of the probability distribution  $\{p_k\}$  with representation  $(\beta', S)$  are given by:

$$\mathcal{P}(z) = z\beta'(I - zS)^{-1}\mathbf{S}^\circ \quad (4.3)$$

$$f(s) = \mathcal{P}(e^{-s}) = e^{-s}\beta'(I - e^{-s}S)^{-1}\mathbf{S}^\circ \quad (4.4)$$

$$\mathcal{P}^{(k)}(1) = k!\beta'S^{k-1}(I - S)^{-k}\mathbf{e} \quad (4.5)$$

where  $\mathbf{e}' = (1, 1, \dots, 1)$

We further consider an equivalent version of the Markov chain in which the absorbing state  $m + 1$  is removed; it is replaced by an “invisible” instantaneous state. We obtain a Markov chain with  $m$  states and transition probability matrix given by

$$S^* = S + S^\circ B^\circ \quad (4.6)$$

where  $S^\circ$  is a  $(m \times m)$  matrix with identical columns  $\mathbf{S}^\circ$  and  $B^\circ = \text{diag}(\beta_1, \dots, \beta_m)$ . The representation  $(\beta', S)$  is called *irreducible* if and only if the matrix  $S^*$  is non-singular. Clearly, matrix  $S^*$  is stochastic and the stationary solution of the equations  $\theta'S^* = \theta'$  and  $\theta'\mathbf{e} = 1$  is given by:

$$\theta' = \mu\beta'(I - S)^{-1} \quad (4.7)$$

where  $\bar{m} = \mu^{-1}$  is the mean of the distribution with representation  $(\beta', S)$ , which can be obtained using equation 4.5. The following two theorems are the discrete case versions of the theorems presented in [Neut 81].

**Theorem 1** *If  $p_k$  is a discrete distribution of phase-type with (irreducible) representation  $(\beta', S)$  and generating function  $\mathcal{P}(z)$ , then*

$$\mathcal{Q}(z) = \mu z \frac{1 - \mathcal{P}(z)}{1 - z} \quad (4.8)$$

*is the generating function of the discrete PH distribution with representation  $(\theta', S)$ , where  $\theta'$  is the stationary solution of the equations  $\theta' S^* = \theta'$  and  $\theta' \mathbf{e} = 1$ .*

**Proof**

*Let  $\{q_k\}$  be a discrete probability distribution with generating function  $\mathcal{Q}(z)$ . We know that  $q_k$  is the coefficient of  $z^k$  in the series:*

$$\mathcal{Q}(z) = \mu z \sum_{i=0}^{\infty} z^i \left[ 1 - \sum_{j=0}^{\infty} p_j z^j \right] \quad (4.9)$$

*Hence we have:*

$$q_k = \begin{cases} 0 & k = 0 \\ \mu \sum_{i=k}^{\infty} p_i & k \geq 1 \end{cases} \quad (4.10)$$

*Now define the vectors  $\mathbf{p}'_k = \beta' S^{k-1}$  ( $k \geq 1$ ),  $\mathbf{p}'_0 = \mathbf{0}'$ , and similarly  $\mathbf{q}'_k = \mu \sum_{i=k}^{\infty} \mathbf{p}'_i$ . Obviously  $q_k = \mathbf{q}'_k \mathbf{S}^0$  by equation 4.2. Multiplying the sequence  $\mathbf{q}'_k$  by  $S$  and subtracting from  $\mathbf{q}'_k$  we obtain:*

$$\mathbf{q}'_k (I - S) = \mu \beta' S^{k-1} \quad (4.11)$$

*The product of the matrices  $S$  and  $(I - S)^{-1}$  is commutative (this can be proved using the fact that the product of the matrices  $S$  and  $(I - S)$  is commutative, therefore  $S = (I - S)^{-1} \cdot S \cdot (I - S)$ ). Using this property and equation 4.7 we obtain:*

$$\mathbf{q}'_k = \mu \beta' (I - S)^{-1} S^{k-1} = \theta' S^{k-1} \quad (4.12)$$

*Therefore  $\{q_k\}$  has representation  $(\theta', S)$ .*

**Theorem 2** *If  $p_k$  is a discrete PH distribution with (irreducible) representation  $(\beta', S)$ , then the convolution of  $p_k$  with itself is also a discrete PH-distribution with representation  $(\gamma', L)$  given by:*

$$\gamma' = (\beta', \mathbf{0}') \quad (4.13)$$

$$L = \begin{pmatrix} S & S^\circ B^\circ \\ 0 & S \end{pmatrix} \quad (4.14)$$

**Proof**

*This theorem is a particular case of the theorem presented in [Neut 81].*

## 4.4 Analysis of the model

We now concentrate on the analysis of one queue in our packet-switching system, let us say Queue 1 (called the *reference queue*). This queue can be regarded as having a single server represented by an  $N$ -state Markov chain, where each state represents the queue to which the channel controller is pointing at time  $\tau_n$ , after the transmitter allocation. This Markov chain has a  $(N \times N)$  transition probability matrix  $T = [t_{ij}]$  where  $t_{ij}$  is the probability of the server pointing to queue  $j$  at  $\tau_{n+1}$  given that it was pointing to queue  $i$  at  $\tau_n$ .

Assuming the system to be in a steady state, let  $\sigma_i$  be the probability of the channel controller finding the channel transmit buffer of queue  $i$  non-empty ( $1 \leq i \leq N$ ). We make an approximating assumption that the probabilities  $\sigma_i$  are independent. The validity of this approximation is considered in section 4.6. At each discrete time interval the channel controller visits queues  $i+1, i+2, \dots, i+j$  (for simplicity no special notation will be used to denote arithmetic  $\text{mod}(N)$ ). The probabilities  $t_{ij}$  can be evaluated in three different cases:

(a) When  $i+1 \leq j \leq i+K-1$ , we have  $t_{ij} = 0$ , since at least  $K$  queues are visited every time-slot.

(b) When  $i + K \leq j \leq i - 1$  we have a situation in which the  $K$  transmitters were filled, therefore queue  $j - 1$  was non-empty. Thus we have:

$$t_{ij} = \sigma_{j-1} P[K - 1 \text{ non-empty queues between } i \text{ and } j - 2] \quad (4.15)$$

Let  $\psi$  be a set of  $K - 1$  queues between  $i$  and  $j - 2$ , and  $\Psi$  the set of all possible  $\psi$ . Then, under the independence assumption above, the probability  $t_{ij}$  is given by:

$$t_{ij} = \sigma_{j-1} \sum_{\psi \in \Psi} \prod_{m \notin \psi} (1 - \sigma_m) \prod_{n \in \psi} \sigma_n \quad (4.16)$$

(c) When  $i = j$  we have a situation in which not necessarily all transmitters were filled. Let  $\psi_x$  be a set of any  $x$  queues and  $\Psi_x$  the set of all possible  $\psi_x$ . The probability  $t_{ii}$  is given by:

$$\begin{aligned} t_{ii} = & \sum_{x=0}^{K-1} \sum_{\psi_x \in \Psi_x} \prod_{m \notin \psi_x} (1 - \sigma_m) \prod_{n \in \psi_x} \sigma_n \\ & + \sigma_{i-1} \sum_{\psi_{K-1} \in \Psi_{K-1}} \prod_{m \notin \psi_{K-1}} (1 - \sigma_m) \prod_{n \in \psi_{K-1}} \sigma_n \end{aligned} \quad (4.17)$$

The first term of equation 4.17 evaluates  $t_{ii}$  when there are less than  $K$  non-empty channel transmit buffers. The second term evaluates  $t_{ii}$  when there are exactly  $K$  non-empty channel transmit buffers. This occurs only when the channel controller stops after queue  $i - 1$  (which is not empty).

#### 4.4.1 The Inter-Visit Time Distribution

We now consider a discrete random variable  $Y$  which describes the number of time-slots between consecutive visits of the server to the reference queue. Let  $r_k = P[Y = k]$ . We can easily show that there is a steady state solution for the system of equations  $\pi' T = \pi'$  and  $\pi' \mathbf{e} = 1$  (the Markov chain defined by  $T$  is clearly irreducible, aperiodic and has finite state space). We first evaluate  $r_1$  and  $r_2$  and then generalise by giving an expression for  $r_k$ , which is easily verified by induction.

We know that the reference queue is visited whenever there is a jump in the corresponding Markov chain from a state  $i$  to a state  $j \leq i$ . Thus, we have:

$$r_1 = \pi_1(t_{11}) + \pi_2(t_{21} + t_{22}) + \dots + \pi_N \sum_{j=1}^N t_{Nj} = \sum_{i=1}^N \sum_{j \leq i} \pi_i t_{ij} \quad (4.18)$$

This uses the fact that an equilibrium state exists just after a polling cycle with a visit to queue 1, since this queue was chosen arbitrarily. Using matrix notation for equation 4.18 yields:

$$r_1 = \pi' R e \quad (4.19)$$

where  $R$  is the lower triangular part of the matrix  $T$ , defined by  $r_{ij} = t_{ij}$  if  $i \geq j$ ;  $r_{ij} = 0$  otherwise. The probability  $r_2$  is calculated using the fact that there are two jumps in the corresponding Markov chain: the first goes from a state  $i$  to a state  $j > i$  and the second goes from a state  $j$  to a state  $k \leq j$ . Thus we have:

$$r_2 = \sum_{i=1}^N \sum_{j>i} \sum_{k \leq j} \pi_i t_{ij} t_{jk} \quad (4.20)$$

Using the matrix form yields:

$$r_2 = \pi' (T - R) R e \quad (4.21)$$

The probability  $r_k$  is calculated in the same way, i.e., by considering that there are  $k-1$  intermediate jumps in the corresponding Markov chain to states numbered successively higher than the initial state, and one jump to a lower numbered state. Thus the expression for  $r_k$  can be written:

$$r_k = \pi' (T - R)^{k-1} R e \quad (4.22)$$

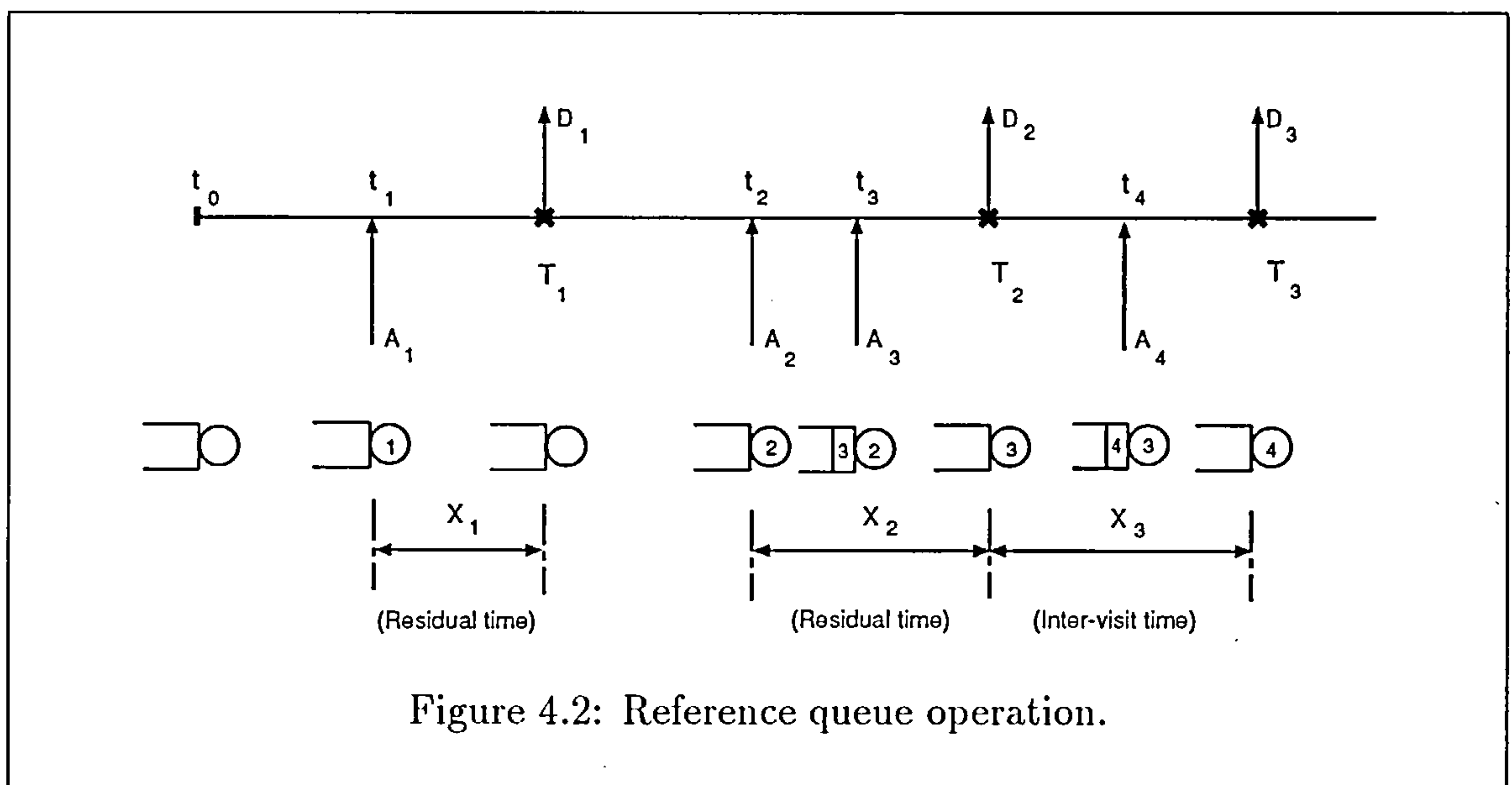
We observe that this probability distribution is equivalent to a phase-type distribution with representation  $(\pi', (T - R))$ , since it produces the same probabilities  $\{r_k\}$ . This representation is reducible, since  $(T - R)^*$  is singular (all elements of its two last rows are identical). An irreducible representation of these probabilities can

be obtained by defining a reduced matrix  $S$  and solving a system of linear equations (this method is discussed in more detail in Appendix B). We therefore assume  $(\beta', S)$  to be an irreducible representation of the probabilities  $\{r_k\}$ .

The system so formed can be regarded as a  $M/G/1$  queue visited by a single server at discrete times. The visit times can be considered as a renewal process of phase-type with representation  $(\beta', S)$ .

#### 4.4.2 The Reference Queue Operation

A typical reference queue operation is depicted in Fig. 4.2. We can see that instants  $t_i$  correspond to the arrival of the customer  $i$  (denoted  $A_i$ ). The instants  $T_i$  correspond to a visit of the channel controller and a departure of a customer (if any).



In the example shown we observe that customer 1 found the queue empty, therefore its sojourn time is given by the time until the next visit of the server. We call this time the *residual time* of the inter-visit renewal process. We have seen that the probability mass function  $\{r_k\}$  of the renewal period of this process is a discrete phase-type with (irreducible) representation  $(\beta', S)$ .

From the same example we also observe that the sojourn time of customer 3 is the residual time plus the inter-visit time, since it found one customer already in the queue when it arrived.

This example shows us that the reference queue operates as a queue with *exceptional first service times* (see [HaPa 92] and [King 90]), where the first service is the time until the next renewal, and ordinary service is the inter-visit time.

We proceed by considering the sojourn time of the reference queue. Let  $\theta'$  be the stationary solution of the system of equations  $\theta' S^* = \theta'$  and  $\theta' \mathbf{e} = 1$ . Let  $\{p_k\}$  be the probability mass function of the PH-distribution with representation  $(\theta', S)$  and the vector  $\mathbf{q} = \{q_k = p_{k+1} - p_k | k = 0, 1, \dots\}$ .

Let  $\bar{m}$  be the mean inter-visit time and  $\rho = \lambda \bar{m}$ . The following definitions will be used in sequence:

**Definition 1** Let the matrix  $Q$  contain the convolutions of the vector  $\mathbf{q}$  with itself, i.e.  $Q = [q_{ij}]$  and  $q_{ij}$  is the  $i$ -th element of the  $j$ -th convolution of  $\mathbf{q}$  with itself ( $i, j = 0, 1, \dots$ ). We also denote  $\mathbf{q}_i^*$  as the  $i$ -th row of  $Q$ . We note that the 0-th column of  $Q$  is the vector  $\mathbf{q}$  itself (i.e.  $q_{k0} = q_k$ ).

**Definition 2** Let the matrix  $E(t)$  contain the elements of the serial expansion of the functions  $\exp(\rho(t - i))$  ( $i = 0, 1, \dots$ ), i.e.  $E = [e_{ij}(t)]$  and  $e_{ij}(t) = \frac{\rho(t-j)^i}{i!}$ . We also denote  $\mathbf{e}_j(t)$  as the  $j$ -th column of  $E$ .

**Definition 3** Let the operation  $d$  of a square matrix  $A$  (denoted to as  $A^d$ ) produce a vector which contains the elements of the diagonal of  $A$ .

The following two theorems in sequence evaluate the residual time and the waiting time distributions of a queue with service time of representation  $(\theta', S)$ . They are useful for evaluating the sojourn time distribution of the reference queue (whose first and ordinary service time have distributions  $\hat{r}(t)$  and  $\{r_k\}$  respectively).

**Theorem 3** *For a renewal process with discrete renewal period distribution of phase type with representation  $(\beta', S)$  and mass function  $\{r_k\}$ , the stationary probability density function of the residual life is given by:*

$$\hat{r}(t) = \mathbf{q}' \mathbf{H}(t) \quad (4.23)$$

Where  $\mathbf{H}(t) = (H(t), H(t-1), \dots)'$  and  $H(t)$  is the unit-step function ( $H(t) = 1$  if  $t > 0$ ,  $H(t) = 0$ , otherwise).

**Proof**

Knowing that the Laplace transform of the probability mass function  $\{r_k\}$  with representation  $(\beta', S)$  is  $f(s)$ , the transform for the residual life  $\hat{r}^*(s)$  is given by:

$$\hat{r}^*(s) = \frac{1 - f(s)}{\bar{m}s} \quad (4.24)$$

Using Theorem 1 (with  $p_k$  replaced by  $r_k$  and  $q_k$  by  $p_k$ ), we obtain the transform of the probability mass function  $\{p_k\}$  with representation  $(\theta', S)$  as:

$$g(s) = \frac{1 - f(s)}{\bar{m}(e^s - 1)} \quad (4.25)$$

Substituting 4.25 in 4.24 we obtain:

$$\hat{r}^*(s) = \frac{g(s)(e^s - 1)}{s} \quad (4.26)$$

The anti-transform of  $g(s)$  is given by  $\sum_{k=1}^{\infty} p_k \delta(t - k)$  (recall that  $p_0 = 0$ ), and the anti-transform of  $g(s)(e^s - 1)$  is  $\sum_{k=0}^{\infty} q_k \delta(t - k)$ , where the function  $\delta(\cdot)$  is the Dirac delta function. To anti-transform 4.26 we obtain:

$$\hat{r}(t) = \int_{-\infty}^t \sum_{k=0}^{\infty} q_k \delta(x - k) dx = \mathbf{q}' \mathbf{H}(t) \quad (4.27)$$

**Remarks**

1. Notice that time is normalised with the time-slot duration  $\tau_{n+1} - \tau_n$ .

2. The first and second moments of the residual time distribution are given by:

$$\bar{m}_r = -\frac{d}{ds} \hat{r}^*(s) \big|_{s=0} = \theta'(I - S)^{-1} \mathbf{e} - \frac{1}{2} \quad (4.28)$$

$$\bar{\bar{m}}_r = \frac{d^2}{ds^2} \hat{r}^*(s) \big|_{s=0} = \frac{1}{3} + 2\theta'S(I - S)^{-2} \mathbf{e} \quad (4.29)$$

Equation 4.28 can be proved by calculating the derivative of  $\hat{r}^*(s)$  from equation 4.26. We obtain:

$$\frac{d}{ds}\hat{r}^*(s) = \frac{g(s)(se^s - e^s + 1)}{s^2} + \frac{(e^s - 1)g'(s)}{s} \quad (4.30)$$

Knowing that:

$$\lim_{s \rightarrow 0} \frac{e^s(s - 1) + 1}{s^2} = \frac{1}{2} \quad (4.31)$$

and that  $g(0) = 1$ ,  $g'(0) = -\theta'(I - S)^{-1}\mathbf{e}$  the result immediately follows.

We can prove equation 4.29 in the same way, since  $\overline{\overline{m}}_r = g''(0) + g'(0) + \frac{1}{3}g(0)$  and  $g''(0) = \theta'(I - S)^{-1}\mathbf{e} + 2\theta'S(I - S)^{-2}\mathbf{e}$ .

**Theorem 4** *In the stable M/G/1 queue with service time distribution  $\{r_k\}$  of phase-type with representation  $(\beta', S)$ , the stationary probability density function of the waiting (queuing) time is given by:*

$$w(t) = (1 - \rho) [\delta(t) + \rho (QE(t))^d \mathbf{H}(t)] \quad (4.32)$$

where  $\rho = \lambda \overline{m}$  and  $\overline{m}$  is the mean service time.

**Proof**

The Laplace transform  $w^*(s)$  of  $w(t)$  is given by the Pollachek-Khinchine formula:

$$w^*(s) = \frac{1 - \rho}{1 - \rho \hat{r}^*(s)} = (1 - \rho) \sum_{i=0}^{\infty} [\rho \hat{r}^*(s)]^i \quad (4.33)$$

We proceed by inverting equation 4.33 term-by-term. Let  $w_i^*(s)$  be defined as  $w_i^*(s) = [\rho \hat{r}^*(s)]^i$  and  $w_i(t)$  be the function obtained by inverting  $w_i^*(s)$  (i.e. the  $i$ -th convolution of  $\hat{r}(t)$  with itself multiplied by  $\rho^i$ ). Thus:

$$1. w_0(t) = \delta(t).$$

2.  $w_1(t)$  is obtained by using theorem 3. We observe that (see definition 1) the  $k$ -th element of the vector  $\mathbf{q}$  is equal to  $q_{k0}$ . Thus we have:

$$w_1(t) = \rho \mathbf{q}' \mathbf{H}(t) = \rho \sum_k q_{k0} H(t - k) \quad (4.34)$$

For simplicity, we omitted the limits of the summation above. It is then understood that the summation is from 0 to  $\infty$ .

3.  $w_2(t)$  is calculated as follows:

$$\begin{aligned} w_2(t) &= \rho^2 \int_{-\infty}^t \sum_m q_{m0} H(t - x - m) \sum_n q_{n0} H(x - n) dx \\ &= \rho^2 \sum_m \sum_n q_{m0} q_{n0} \int_{-\infty}^t H(t - x - m) H(x - n) dx \end{aligned} \quad (4.35)$$

Integrating the equation above we obtain:

$$w_2(t) = \rho^2 \sum_m \sum_n (t - m - n) q_{m0} q_{n0} H(t - m - n) \quad (4.36)$$

Rearranging the equation above and using definition 1 we obtain:

$$w_2(t) = \rho \sum_k q_{k1} [\rho(t - k)] H(t - k) \quad (4.37)$$

4. Using the same procedure the function  $w_i(t)$  is given by (observe that we used definition 2):

$$\begin{aligned} w_i(t) &= \rho \sum_k q_{ki-1} \frac{[\rho(t - k)]^{i-1}}{(i-1)!} H(t - k) \\ &= \rho \sum_k q_{ki-1} e_{i-1,k}(t) H(t - k) \end{aligned} \quad (4.38)$$

The function  $w(t)$  is then given by:

$$\begin{aligned} w(t) &= (1 - \rho) [\delta(t) + \sum_i \sum_k q_{ik} e_{ki}(t) H(t - i)] \\ &= (1 - \rho) [\delta(t) + \sum_i \mathbf{q}_i^* \mathbf{e}_i(t) H(t - i)] \end{aligned} \quad (4.39)$$

Using definition 3 we obtain equation 4.32.  $\square$

### Remarks

1. Notice again that time is normalised with the time-slot duration.

2. For  $0 \leq t \leq 1$   $w(t)$  can be explicitly derived as:

$$w(t) = (1 - \rho) [\delta(t) + \rho \exp(\theta' S^\circ \rho t)] \quad (4.40)$$

3. The first and second moments of the waiting time distribution are given by:

$$\bar{m}_w = -\frac{d}{ds} w^*(s) \big|_{s=0} = \frac{\rho \bar{m}_r}{(1 - \rho)} \quad (4.41)$$

$$\bar{\bar{m}}_w = \frac{d^2}{ds^2} w^*(s) \big|_{s=0} = \frac{\rho [2\rho(\bar{m}_r)^2 + \bar{m}_r(1 - \rho)]}{(\rho - 1)^2} \quad (4.42)$$

The following theorem considers the sojourn time of a packet in the reference queue. We have seen that it operates with exceptional first service time, where the distribution of the ordinary service time is discrete phase-type of representation  $(\beta', S)$  and the distribution of the first service time is  $\hat{r}(t)$ .

Let the matrices  $\dot{E}(t) = [\dot{e}_{ij}(t)]$  and  $\ddot{E}(t) = [\ddot{e}_{ij}(t)]$  be defined as follows:

$$\begin{aligned} \dot{e}_{ij}(t) &= e_{i+1j}(t) = \frac{[\rho(t-j)]^{i+1}}{(i+1)!} \\ \ddot{e}_{ij}(t) &= (i+1)e_{i+2j}(t) = (i+1) \frac{[\rho(t-j)]^{i+2}}{(i+2)!} \quad (i, j = 0, 1, \dots) \end{aligned} \quad (4.43)$$

As in definition 2, the  $j$ -th columns of  $\dot{E}(t)$  and  $\ddot{E}(t)$  are referred to as  $\dot{e}_j(t)$  and  $\ddot{e}_j(t)$  respectively.

Let the matrix  $\Gamma(t) = [\gamma_{ij}(t)]$  be defined as follows:

$$\gamma_{ij}(t) = q_i [\varphi - (t - i - j) \mathbf{q}_j^{*'} \dot{e}_{i+j}(t) + \frac{1}{\rho} \mathbf{q}_j^{*'} \ddot{e}_{i+j}(t)] H(t - i - j) \quad (4.44)$$

where  $\rho = \lambda \bar{m}$ ,  $\bar{m}$  is the mean ordinary service time and  $\varphi = \frac{1}{\lambda} - \bar{m}$ .

Let the vector  $\mathbf{u}(t) = \{u_i(t) | i = 0, 1, \dots\}$  be defined as follows:

$$u_i(t) = [\varphi - (t - i)] H(t - i) \quad (4.45)$$

**Theorem 5** *In a stable  $M/G/1$  queue in which the distribution of the first service time is  $\hat{r}(t)$  and the distribution of the ordinary service time is of phase-type with representation  $(\beta', S)$ , the sojourn time is distributed according to the function  $W(t)$  given by:*

$$W(t) = \pi_0 \lambda [(1 + \mathbf{q}' \mathbf{u}(t)) H(t) + (Q \dot{E}(t))^d \mathbf{H}(t) + \mathbf{e} \Gamma(t) \mathbf{e}'] \quad (4.46)$$

where  $\pi_0$  is the steady state probability of having 0 customers in the reference queue (observed at a random instant), i.e. :

$$\pi_0 = \frac{1 - \lambda \bar{m}}{1 + \lambda(\bar{m}_\tau - \bar{m})} \quad (4.47)$$

**Proof**

The Laplace transform of the sojourn time distribution (denoted  $W^*(s)$ ) of a  $M/G/1$  queue with exceptional first time service is given by (see [HaPa 92] and [King 90]):

$$W^*(s) = \frac{\pi_0 (\lambda f(s) - (\lambda - s) \hat{r}^*(s))}{s - \lambda(1 - f(s))} \quad (4.48)$$

where  $f(s)$  and  $\hat{r}^*(s)$  are the Laplace transforms of the ordinary and first service respectively. Using equation 4.24, and re-arranging equation 4.48 we obtain:

$$W^*(s) = W_1^*(s) W_2^*(s) \quad (4.49)$$

where  $W_1^*(s)$  and  $W_2^*(s)$  are given by:

$$W_1^*(s) = \frac{\pi_0 \lambda}{1 - \rho \hat{r}^*(s)} \quad (4.50)$$

$$W_2^*(s) = \frac{1}{s} + \left(\varphi - \frac{1}{s}\right) \hat{r}^*(s) \quad (4.51)$$

The anti-transform of  $W_1^*(s)$  can be easily obtained by using theorem 4 (multiplying equation 4.32 by  $\frac{\pi_0 \lambda}{(1-\rho)}$ ). We then obtain:

$$W_1(t) = \pi_0 \lambda [\delta(t) + \rho (Q E(t))^d \mathbf{H}(t)] \quad (4.52)$$

It is easy to verify that the anti-transform of  $W_2^*(s)$  is:

$$W_2(t) = H(t) + \mathbf{q}' \mathbf{u}(t) \quad (4.53)$$

The sojourn time distribution is the convolution of  $W_1(t)$  and  $W_2(t)$ . Thus we have:

$$W(t) = \pi_0 \lambda \left[ \int_{-\infty}^t H(t-x) \delta(x) dx + \int_{-\infty}^t H(t-x) \rho(QE(t))^d \mathbf{H}(x) dx + \int_{-\infty}^t \mathbf{q}' \mathbf{u}(t-x) \delta(x) dx + \int_{-\infty}^t \mathbf{q}' \mathbf{u}(t-x) \rho(QE(t))^d \mathbf{H}(x) dx \right] \quad (4.54)$$

We proceed by calculating the four integrals of equation 4.54 (denoted to as  $I_1$ ,  $I_2$ ,  $I_3$  and  $I_4$  respectively).

1. The integral  $I_1$  yields:

$$I_1 = \int_{-\infty}^t H(t-x) \delta(x) dx = H(t) \quad (4.55)$$

Observe that  $H(t-x) = 1$  in the interval  $(-\infty, t]$ .

2. The integral  $I_2$  is evaluated as follows (note that  $\sum_i = \sum_{i=0}^{\infty}$ ):

$$\begin{aligned} I_2 &= \rho \int_{-\infty}^t H(t-x) \sum_i \mathbf{q}_i^{*'} \mathbf{e}_i(x) H(x-i) dx \\ &= \rho \sum_i \int_{-\infty}^t \mathbf{q}_i^{*'} \mathbf{e}_i(x) H(t-i) dx \end{aligned} \quad (4.56)$$

In sequence, we use the following property: if  $f(x)$  is a real function integrable over  $(-\infty, t]$  and  $a$  is a real constant then:

$$\int_{-\infty}^t f(x) H(t-a) dx = H(t-a) \int_a^t f(x) dx.$$

Thus we have:

$$\begin{aligned} I_2 &= \rho \sum_i H(t-i) \int_i^t \mathbf{q}_i^{*'} \mathbf{e}_i(x) dx \\ &= \rho \sum_i H(t-i) \sum_j q_{ij} \int_i^t e_{ji}(x) dx \end{aligned} \quad (4.57)$$

Using definition 2 and integrating the equation above we obtain:

$$\begin{aligned} I_2 &= \sum_i H(t-i) \sum_j q_{ij} \frac{[\rho(t-i)]^{j+1}}{(j+1)!} \\ &= \sum_i H(t-i) \mathbf{q}_i^{*'} \dot{\mathbf{e}}_i(t) = (Q \dot{E}(t))^d \mathbf{H}(t) \end{aligned} \quad (4.58)$$

3. The integral  $I_3$  is clearly given by:

$$I_3 = \mathbf{q}' \mathbf{u}(t) H(t) \quad (4.59)$$

4. The integral  $I_4$  is evaluated as follows:

$$\begin{aligned} I_4 &= \rho \int_{-\infty}^t \sum_i q_i [\varphi - (t-x-i)] H(t-x-i) \\ &\quad \times \sum_j \mathbf{q}_j^{*'} \mathbf{e}_j(x) H(x-j) dx \end{aligned} \quad (4.60)$$

As in the previous step, we use the fact that if a function  $f(x)$  is integrable in  $(-\infty, t]$  and  $a$  and  $b$  are real numbers then

$$\int_{-\infty}^t f(x) H(t-x-a) H(x-b) dx = H(t-a-b) \int_b^{t-a} f(x) dx.$$

Thus, rearranging equation 4.60 we obtain:

$$I_4 = \rho \sum_i \sum_j H(t-i-j) \int_j^{t-i} q_i [\varphi - (t-x-i)] \mathbf{q}_j^{*'} \mathbf{e}_j(x) dx \quad (4.61)$$

The equation above can be re-written as:

$$\begin{aligned} I_4 &= \rho \sum_i \sum_j H(t-i-j) q_i \left\{ \int_j^{t-i} [\varphi - (t-i-j)] \mathbf{q}_j^{*'} \mathbf{e}_j(x) dx \right. \\ &\quad \left. + \int_j^{t-i} (x-j) \mathbf{q}_j^{*'} \mathbf{e}_j(x) dx \right\} \end{aligned} \quad (4.62)$$

Integrating equation above we obtain:

$$\begin{aligned} I_4 &= \sum_i \sum_j H(t-i-j) q_i \left\{ [\varphi - (t-i-j)] \mathbf{q}_j^{*'} \dot{\mathbf{e}}_{i+j}(t) + \frac{1}{\rho} \mathbf{q}_j^{*'} \ddot{\mathbf{e}}_{i+j}(t) \right\} \\ &= \sum_i \sum_j \gamma_{ij}(t) = \mathbf{e} \Gamma(t) \mathbf{e}' \end{aligned} \quad (4.63)$$

By substituting equations 4.55, 4.58, 4.59 and 4.63 in equation 4.54 we obtain equation 4.46  $\square$ .

### Remarks

1. Time is normalised with the time-slot duration.
2. The first moment of the sojourn time distribution is given by:

$$\bar{m}_s = \frac{\pi_0 \lambda}{(1 - \rho)^2} \left[ (1 - \rho) \left( \frac{\bar{m}_r}{2} + \varphi \bar{m}_r \right) + \rho \bar{m}_r (\varphi + \bar{m}_r) \right] \quad (4.64)$$

## 4.5 The algorithmic solution

The algorithm presented here is based on successive convolutions of the sequence formed by the vector  $\mathbf{q}$ . It is assumed that the probability mass function  $\{r_k\}$  of the inter-visit time distribution with representation  $(\beta', S)$  is given. The objective of the algorithm is to produce the value of  $W(t)$  for a given  $t$ .

The algorithm starts reading  $(\beta', S)$ ,  $\sigma$  and  $t$ . From this, it constructs the probability mass function  $\{p_k\}$  with representation  $(\theta', S)$ . The vector  $\mathbf{q}$  is calculated using the iterative formula  $q_k = p_{k+1} - p_k$ , ( $k = 0, 1, \dots$ ). It is useful to store the successive convolutions  $q_k^{(m)}$  in memory. These convolutions can be evaluated either directly from the definition or in two steps: (1) Calculating  $p_k^{(m)}$  using theorem 2 and (2) Calculating the convolution  $p_k^{(m)}$  with  $d_k^{(m)}$ , where  $d_k^{(m)}$  is the  $m$ -th convolution of the anti-transform of  $z/1 - z$  with itself.

Knowing that  $W(t)$  is a sum of convergent series, a precision parameter  $\varepsilon$  can also be given and tested every iteration.

The algorithm can be summarised in the following steps:

### Algorithm 1

1. Read  $(\beta', S)$ ,  $\lambda$  and  $t$ .

2. Calculate the mean of the probability mass  $\{r_k\}$  using the expression

$$\bar{m} = \frac{1}{\mu} = \beta'(I - S)^{-1} \mathbf{e} \quad (4.65)$$

(The parameters  $\rho = \lambda \bar{m}$  and  $\varphi = \lambda^{-1} + \bar{m}$  immediately follows).

3. Calculate  $\theta'$  using the expression

$$\theta' = \mu \beta'(I - S)^{-1} \quad (4.66)$$

4. Calculate probabilities  $\{p_k\}$  of representation  $(\theta', S)$  using equation 4.2.

5. Calculate sequence  $\mathbf{q}$  using the expression

$$q_k = p_{k+1} - p_k \quad (4.67)$$

6. Calculate the matrix  $Q$  by evaluating the convolutions of  $\mathbf{q}$  with itself.

7. Create the functions  $\dot{e}_{ij}(t)$  and  $\ddot{e}_{ij}(t)$  given by:

$$\begin{aligned} \dot{e}_{ij}(t) &= \frac{[\rho(t-j)]^{(i+1)}}{(i+1)!} \\ \ddot{e}_{ij}(t) &= (i+1) \dot{e}_{i+1j}(t) \end{aligned} \quad (4.68)$$

8. Create the function  $\gamma_{ij}(t)$  given by:

$$\gamma_{ij}(t) = q_i \left[ \sum_{k=0}^{\max} q_{jk} \dot{e}_{k+i+j}(t) + \rho^{-1} \sum_{k=0}^{\max} q_{jk} \ddot{e}_{k+i+j}(t) \right] \quad (4.69)$$

where  $Q = [q_{ij}]$  and  $\mathbf{q} = \{q_i\}$  ( $i, j = 0, 1, \dots$ ).

9. Evaluate  $s_1(t)$ ,  $s_2(t)$  and  $s_3(t)$  given by:

$$\begin{aligned} s_1(t) &= 1 + \sum_{i=0}^{\lfloor t \rfloor} q_i (\varphi - t - i) \\ s_2(t) &= \sum_{i=0}^{\lfloor t \rfloor} \sum_{j=0}^{\max} q_{ij} \dot{e}_{ji}(t) \\ s_3(t) &= \sum_{i=0}^{\lfloor t \rfloor} \sum_{j=0}^{\lfloor t \rfloor - i} \gamma_{ij}(t) \end{aligned} \quad (4.70)$$

10. Evaluate  $W(t)$  using the expression

$$W(t) = \pi_0 \lambda (s_1(t) + s_2(t) + s_3(t)) \quad (4.71)$$

where  $\pi_0$  is calculated by using equation 4.47.

### Remarks

1. Observe that the parameter  $max$  is chosen so that  $q_{i_{max}} \dot{e}_{i_{max}} < \varepsilon$ , where  $\varepsilon$  is the pre-defined precision parameter.
2. The matrix  $S$  of representation  $(\beta', S)$  is given in terms of the probabilities  $\sigma_i$ . These probabilities can be evaluated from the arrival rates  $\lambda_i$ . We have seen that  $1 - \sigma_i$  is the probability of the channel controller finding queue  $i$  empty. This happens whenever queue  $i$  is empty at a random instant (this probability is given by  $\pi_{0i}$ ) and there is no arrival until the next visit of the channel controller (we referred to this interval as residual time). Therefore,  $\sigma_i$  is calculated by finding the real roots of the following system of equations:

$$1 - \sigma_i = \pi_{0i} \int_0^\infty e^{-\lambda x} \hat{r}(x) dx = \pi_{0i} \hat{r}^*(\lambda_i) \quad (i = 0, 1, \dots, N) \quad (4.72)$$

## 4.6 Numerical results

We examine the case where  $\sigma_i = \sigma$ , ( $i = 1, 2, \dots, N$ ). Firstly we consider the case where  $N = 4$  and  $K = 2$ . The matrix  $T$  can be constructed using the equations 4.16 and 4.17. Thus we have:

$$T = \begin{pmatrix} 1 - 3\sigma^2 + 2\sigma^3 & 0 & \sigma^2 & 2\sigma^2(1 - \sigma) \\ 2\sigma^2(1 - \sigma) & 1 - 3\sigma^2 + 2\sigma^3 & 0 & \sigma^2 \\ \sigma^2 & 2\sigma^2(1 - \sigma) & 1 - 3\sigma^2 + 2\sigma^3 & 0 \\ 0 & \sigma^2 & 2\sigma^2(1 - \sigma) & 1 - 3\sigma^2 + 2\sigma^3 \end{pmatrix} \quad (4.73)$$

Obviously, the steady state solution for the discrete Markov chain with state transition matrix  $T$  is  $\pi' = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$ . In section 4.4.1 we defined  $r_k$  as being the probability of having  $k$  time-slots between consecutive visits to the reference queue. Using equation 4.22 we have:

$$r_k = \begin{cases} 0 & k = 0 \\ 1 - \sigma^2(1 - \frac{\sigma}{2}) & k = 1 \\ \frac{\sigma^2}{2}(2 - \sigma) & k = 2 \\ 0 & k > 2 \end{cases} \quad (4.74)$$

We observe that an irreducible representation of the probabilities  $\{r_k\}$  is  $(\beta', S)$  given by:

$$\beta' = (\frac{1}{2}, \frac{1}{2}) \quad (4.75)$$

$$S = \begin{pmatrix} 0 & \sigma^2(2 - \sigma) \\ 0 & 0 \end{pmatrix} \quad (4.76)$$

Figure 4.3 shows  $W(t)$  for  $\lambda = 0.15$ ,  $\lambda = 0.25$  and  $\lambda = 0.45$ . Notice that explicit equations for its first moment are given by equation 4.64.

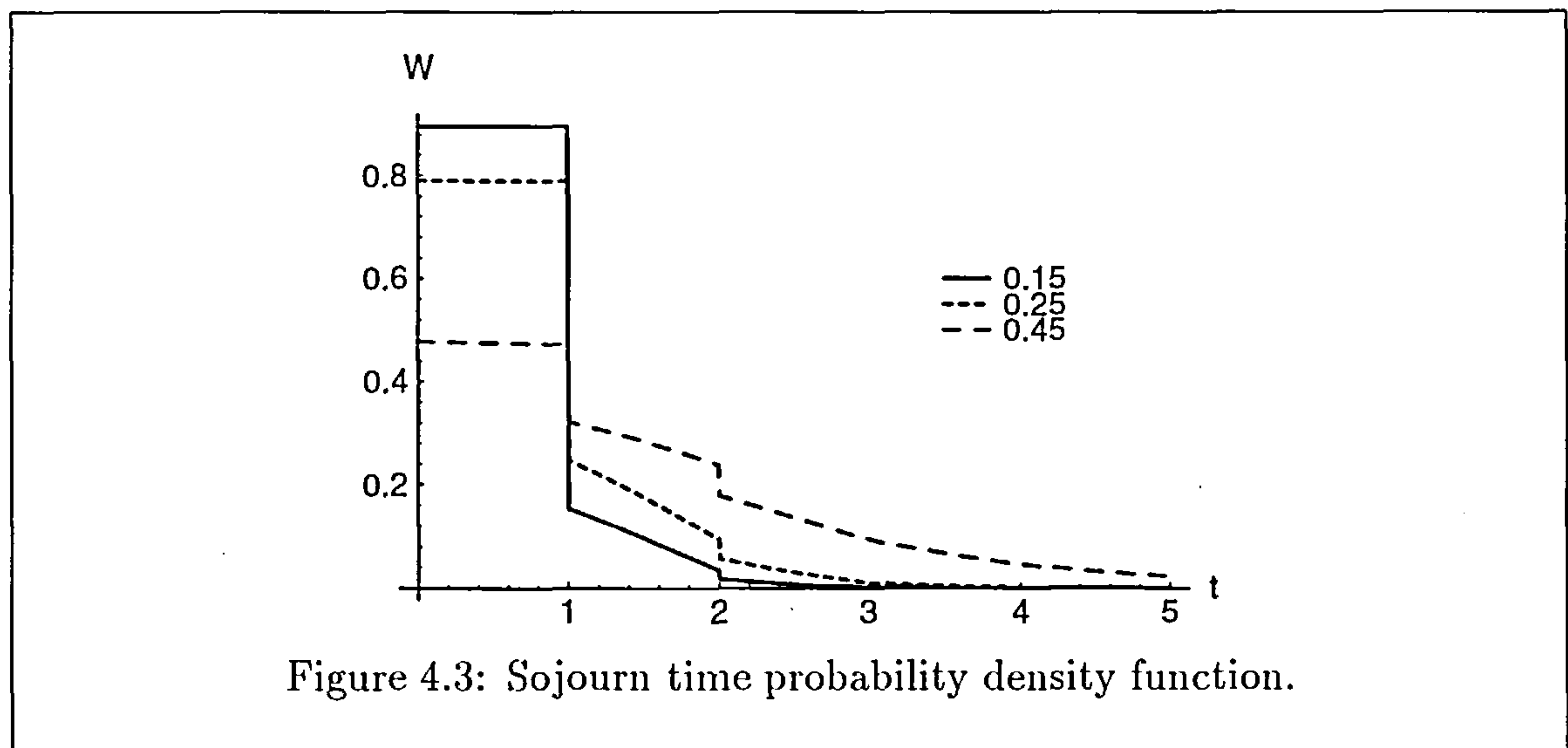
Expressions for the first and second moments of the residual time can be calculated using equations 4.28 and 4.29. Thus we have:

$$\overline{m}_r = \frac{2(\sigma^3 - 2\sigma^2 - 1)}{\sigma^3 - 2\sigma^2 - 2} - \frac{1}{2} \quad (4.77)$$

$$\overline{\overline{m}}_r = \frac{-7\sigma^3 + 14\sigma^2 + 2}{-6\sigma^3 + 12\sigma^2 + 12} \quad (4.78)$$

From the equations above the first moment  $\overline{m}_s$  immediately follows.

The sojourn time distribution has points of discontinuity when  $t = 1$  and  $t = 2$ . These points correspond to the points of discontinuity of  $\hat{r}(t)$ . We also observe that the mean sojourn time tends to infinity when at higher arrival rates, which is as expected, since the system then becomes unstable.



### 4.6.1 Model validation

In this section we validate our analytical model by comparing its results with the results obtained by simulation. We carried out a discrete-event simulation of the system described in section 4.2. A Sun/Sparc workstation was used and the program was written in *C* programming language.

Initially we estimated the mean sojourn time for balanced systems comprising four queues and two transmitters. The results are shown in Fig. 4.4 (observe that the dashed line represents the simulation results). We also present the same results in table 4.1. The confidence intervals were estimated using the batch means approach. Each sample ran 300,000 units of time (time-slots) and 6 samples were taken.

As can be seen from the table below, the analytical model is very accurate when the arrival rate is less or equal than 0.4 packets/time unit. The relative errors are more noticeable at higher arrival rates where our analytical model underestimates the mean sojourn time. We therefore examine our initial assumption of independence of the probabilities  $\sigma_i$ .

Let  $Z_i$  be a random variable which is equal to 0 when the channel controller finds queue  $i$  empty, otherwise it is equal to 1 (obviously  $\sigma_i = P(Z_i = 1)$ ). We use our simulation model to estimate the covariance and correlation between  $Z_i$  and

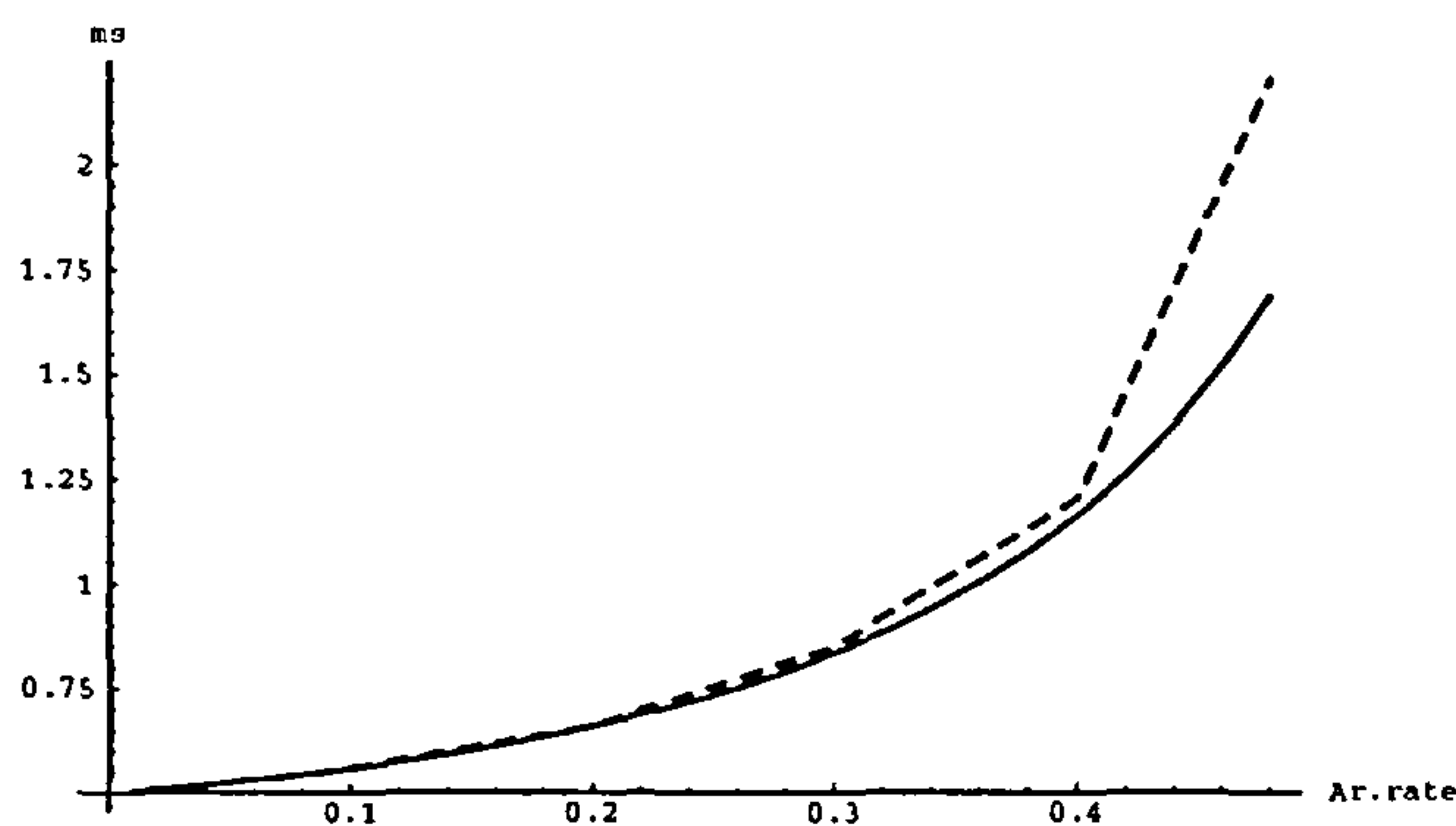


Figure 4.4: Mean sojourn time.

| $\lambda$ | Anal. Model | Simulation | 0.95 Conf.Interval | Relative Error |
|-----------|-------------|------------|--------------------|----------------|
| 0.10      | 0.558       | 0.560      | (0.540,0.580)      | -0.36          |
| 0.20      | 0.659       | 0.660      | (0.643,0.677)      | -0.15          |
| 0.30      | 0.834       | 0.849      | (0.821,0.877)      | -1.80          |
| 0.40      | 1.160       | 1.204      | (1.152,1.256)      | -3.65          |
| 0.48      | 1.684       | 2.101      | (2.079,2.123)      | -19.8          |

Table 4.1: Mean Sojourn Time Results.

$Z_j$  ( $j \neq i$ ). We also estimate the autocorrelation function of  $Z_i$  regarded as a time series and the results are presented below.

### Estimation of the covariance and correlation

We used our simulation model to collect 2000 observations of the random variables  $Z_i$  and  $Z_j$ . Although we show here the results for  $i = 1$  and  $j = 2$ , we can say that similar results were obtained for the other cases. We estimated the means  $\bar{z}_1$  and  $\bar{z}_2$ , the covariance  $cv(Z_1, Z_2)$  and the correlation  $cr(Z_1, Z_2)$ . Again, we adopted the batch means approach, where 6 samples were taken. Each sample run 300,000 units of time giving a total simulation time of 1,800,000 units.

We simulated systems comprising  $N = 4$  queues and  $K = 2$  transmitters (see table 4.2). We also simulated systems with  $N = 8$  queues and  $K = 4$  transmitters

| <i>Balanced</i>   |             |             |                |                       |                |
|-------------------|-------------|-------------|----------------|-----------------------|----------------|
| $\lambda_0$       | $\bar{z}_1$ | $\bar{z}_2$ | $cv(Z_1, Z_2)$ | 95% conf. interval    | $cr(Z_1, Z_2)$ |
| 0.5               | 0.127       | 0.130       | -0.00133       | (-0.00134 – -0.00130) | -0.011         |
| 1.0               | 0.263       | 0.264       | 0.00318        | (0.00315 – 0.00321)   | 0.016          |
| 1.4               | 0.428       | 0.422       | 0.01130        | (0.01116 – 0.01145)   | 0.046          |
| 1.8               | 0.640       | 0.652       | 0.02110        | (0.02102 – 0.02117)   | 0.092          |
| 1.95              | 0.860       | 0.861       | 0.00910        | (0.00907 – 0.00928)   | 0.082          |
| <i>Unbalanced</i> |             |             |                |                       |                |
| $\lambda_0$       | $\bar{z}_1$ | $\bar{z}_2$ | $cv(Z_1, Z_2)$ | 95% conf.interval     | $cr(Z_1, Z_2)$ |
| 0.5               | 0.352       | 0.075       | -0.00265       | (-0.00270 – -0.00260) | -0.015         |
| 0.7               | 0.508       | 0.116       | 0.00612        | (0.00601 – 0.00623)   | 0.038          |
| 0.9               | 0.657       | 0.148       | 0.00343        | (0.00339 – 0.00347)   | 0.022          |
| 1.1               | 0.785       | 0.178       | 0.00209        | (0.00200 – 0.00218)   | 0.015          |
| 1.35              | 0.964       | 0.212       | -0.00021       | (-0.00021 – -0.00020) | -0.002         |

Table 4.2: Estimation of the covariance and correlation ( $N = 4, K = 2$ ).

| <i>Balanced</i>   |             |             |                |                       |                |
|-------------------|-------------|-------------|----------------|-----------------------|----------------|
| $\lambda_0$       | $\bar{z}_1$ | $\bar{z}_2$ | $cv(Z_1, Z_2)$ | 95% conf. interval    | $cr(Z_1, Z_2)$ |
| 0.5               | 0.054       | 0.063       | 0.00198        | (0.00190 – 0.00206)   | 0.019          |
| 1.5               | 0.205       | 0.180       | 0.00365        | (0.00355 – 0.00375)   | 0.022          |
| 3.0               | 0.404       | 0.408       | 0.01347        | (0.01339 – 0.01355)   | 0.055          |
| 3.8               | 0.662       | 0.669       | 0.01861        | (0.01855 – 0.01858)   | 0.084          |
| <i>Unbalanced</i> |             |             |                |                       |                |
| $\lambda_0$       | $\bar{z}_1$ | $\bar{z}_2$ | $cv(Z_1, Z_2)$ | 95% conf. interval    | $cr(Z_1, Z_2)$ |
| 0.4               | 0.304       | 0.064       | -0.00101       | (-0.00110 – -0.00092) | -0.014         |
| 0.8               | 0.573       | 0.115       | 0.00027        | (0.00030 – 0.00024)   | 0.001          |
| 1.0               | 0.721       | 0.168       | -0.00502       | (-0.00510 – -0.00490) | -0.033         |
| 1.3               | 0.915       | 0.193       | -0.00018       | (-0.00020 – -0.00017) | 0.001          |

Table 4.3: Estimation of the covariance and correlation ( $N = 8, K = 4$ ).

(see table 4.3). The above mentioned parameters were calculated for various total incoming traffic (referred to as  $\lambda_0 = \sum \lambda_i$ ). We considered both balanced systems (where  $\lambda_i = \lambda_0/N$ ) and unbalanced systems (where we assumed  $\lambda_1 = 0.70\lambda_0$ ,  $\lambda_2 = 0.15\lambda_0$ ,  $\lambda_3 = 0.10\lambda_0$  and  $\lambda_i = 0.05 \frac{\lambda_0}{(N-3)}$  ( $i = 4, \dots, N$ )).

By observing tables 4.2 and 4.3 we can say that there is no evidence that variables  $Z_1$  and  $Z_2$  are correlated. We observe, for example, in table 4.2 that when  $\lambda_0 = 1.95$  (note that the maximum total arrival rate supported is  $\lambda_0 = 2.0$ ), we have  $cv(Z_1, Z_2) = 0.009$ . This is a quite small number compared with  $\bar{z}_1\bar{z}_2 = 0.78$ . Since

| <i>Balanced Systems</i>   |            |      |       |      |      |       |     |
|---------------------------|------------|------|-------|------|------|-------|-----|
| $\lambda_0$               | <i>lag</i> |      |       |      |      |       | $q$ |
|                           | 1          | 2    | 3     | 4    | 5    | 6     |     |
| 0.5                       | 0.07       | 0.01 | -0.01 | 0.02 | 0.01 | -0.00 | 0   |
| 1.0                       | 0.24       | 0.11 | 0.04  | 0.03 | 0.05 | 0.04  | 1   |
| 1.4                       | 0.25       | 0.14 | 0.07  | 0.05 | 0.04 | -0.04 | 2   |
| 1.8                       | 0.42       | 0.30 | 0.23  | 0.20 | 0.17 | 0.04  | 5   |
| <i>Unbalanced Systems</i> |            |      |       |      |      |       |     |
| $\lambda_0$               | <i>lag</i> |      |       |      |      |       | $q$ |
|                           | 1          | 2    | 3     | 4    | 5    | 6     |     |
| 0.5                       | 0.16       | 0.09 | 0.06  | 0.04 | 0.03 | 0.00  | 1   |
| 0.7                       | 0.20       | 0.10 | 0.05  | 0.06 | 0.03 | 0.01  | 1   |
| 0.9                       | 0.28       | 0.18 | 0.13  | 0.07 | 0.07 | 0.05  | 3   |
| 1.1                       | 0.33       | 0.22 | 0.15  | 0.08 | 0.06 | 0.04  | 3   |

Table 4.4: Estimation of the autocorrelation function ( $N = 4, K = 2$ ).

$cv(Z_1, Z_2) = E(Z_1 Z_2) - \bar{z}_1 \bar{z}_2$ , we conclude that it is reasonable to assume that the variables are independent.

### Estimation of the autocorrelation function

From our simulation model, we collected 2000 observations of the variable  $Z_i$  and estimated its autocorrelation function. This function gives the correlation between  $Z_i$  observed at time  $t$  with  $Z_i$  observed at time  $t + k$ . The interval  $k$  between these observations is called *lag*. Clearly, successive observations of  $Z_i$  form a stationary stochastic process. We therefore used *Bartlett's formula* (see [BoJe 71]) to estimate the smallest lag  $q$  beyond which the autocorrelation function is zero.

We show in table 4.4 the auto-correlation function for both balanced and unbalanced systems comprising four queues and two transmitters. We estimated the autocorrelation function for queue 1 (i.e. we collected observations of the variable  $Z_1$ ) and we show the results for several total incoming traffic  $\lambda_0$ .

The simulation results indicate that successive values of the value  $Z_1$  are auto-correlated, but only slightly at low-medium traffic, and increasing at higher traffic. This agrees with what would be expected in this case, since successive cycles in a

polling systems are interdependent.

As a consequence of this, our model underestimates the probability  $\sigma_i$ . This is so because it does not fully capture the effect of the interdependence of successive values of  $Z_i$ . This is the “price we pay” by using a simplified Markov chain to represent the operation of the channel controller.

A more accurate model at higher traffic can be obtained by considering a phase-type process which includes the state of every queue in the multi-queue system, as we have done in chapter 3. In this way, the probabilities  $t_{ij}$  (see section 4.4) will reflect the state of the system, rather than only the position of the channel controller. We consider this as a suggestion for further work.

Based on the numerical results obtained in this section we can say that our model is very accurate for  $\lambda \leq 0.4$  (for systems with  $N = 4$  and  $K = 2$ ). Considering that the maximum arrival rate supported in such a system is  $\lambda = 0.5$ , we assume that the analytical model can be used for multi-queue systems in which the arrival rate does not exceed 80% of its total capacity.

## 4.7 Conclusion

A solution for the sojourn time distribution in multi-queue systems served in cyclic order in discrete time intervals has been presented. The analytical model is based on the inter-visit time renewal process, which we considered as being of phase-type. We have seen that its operation can be represented by a Markov chain, the state of which represents the position of the server. The advantage of using this method is that it is simpler than the method described in chapter 3 and that it produces compact matrix equations and efficient recursive algorithms for numerical evaluation. This methodology can be extended to several other polling strategies, which is a suggestion for future research.

# Chapter 5

## Asymptotic Analysis and Cell Modelling

### 5.1 Introduction

In this chapter we introduce a closed queuing network model in order to analyse the asymptotic properties of cellular networks. Queuing network (QN) models are used widely in the analysis of computer systems and communication networks. The model presented here has some nodes with non-exponential service time distribution. A method for determining approximate performance values for closed QN's with first-come-first-served (FCFS) disciplines and general service times is given in [ChHW 75]. Using this method, we can derive quantities such as network throughput, processor utilisation and mean response time. This is achieved using ratios of the normalising constant  $g_m(k)$  expressed in terms of the network population  $k$  and the number of nodes  $m$ .

However, evaluating the normalising constant directly from the definition frequently leads to prohibitive computational effort, particularly when dealing with networks with large populations. In such cases, it is more convenient to generate asymptotic expansions of the normalising constant. In this way, the task of solv-

ing numerically a large recursion equation is reduced to the calculation of a few expansion coefficients.

One well-known technique for deriving this expansion was introduced by J. McKenna and D. Mitra (see [McMi 82], [MiMc 86] and [MiMc 84]). The approach used was based on an integral representation of the normalising constant, obtained by using Euler's integral formula for the factorial function. This integral representation contained a parameter, which reflected the size of the network. Expansions of these integrals were expressed in terms of inverse powers of this parameter.

Although this method simplified the computation of the normalising constant for networks of various sizes, it has some limitations. Firstly, equations in the asymptotic coefficients are given in terms of the normalising constant of what is called *pseudo-networks*. These pseudo-networks contain all the nodes of the original network with the exception of the infinite-server (IS) nodes. Consequently, this computation is still complex in networks with a large number of non-IS nodes. Secondly, the analysis is restricted to certain limits of what is called *network usage*. This is a parameter related to the node utilisation.

This chapter introduces a new method for evaluating asymptotic properties of large closed queuing networks, considering nodes with both constant service rates and certain cases of load dependent service rates. This method is based on the convolution theorem which expresses the generating function of the normalising constant as a product of simpler node functions. We also observe that in cases where the nodes are load independent, the generating function  $G_m(z)$  is expressed as a product of ratios of polynomials in  $z$ . These polynomials are also in the formula of the generating function of a great number of cases, including networks with load dependent nodes. It is therefore convenient to transform this rational function into a sum of simpler terms. A term-by-term inversion of the resulting sum is carried out, obtaining in this way a series whose  $k$ -th element is the generating function at

population  $k$ . An exact closed formula which can be used directly in a great number of cases will be obtained. However, when the network population is large it is more convenient to transform this series into an equivalent continuous function expressed in an integral form. From this formula the asymptotic expansion can be carried out by using existing methods.

There are several advantages in using this method. It gives an exact closed formula for the normalising constant in a great number of cases. There are no restrictions on the network usage and it produces the required asymptotic behaviour for the normalising constant.

In the second part of this chapter we present a queuing network model for cellular networks, based on the model given by [DaSm 91]. We obtain the flow equivalent server for the sub-network comprising the stations inside a cell.

This chapter is organised as follows: section 5.2 provides a brief summary of the relevant parts of the product-form theory for queuing networks; section 5.3 gives formulas for the normalising constant for various types of queuing networks; section 5.4 gives asymptotic results of the normalising constant; section 5.5 applies these results in cellular networks to determine the flow equivalent server of a cell. Finally, in section 5.6 we give our conclusions.

## 5.2 Product-form queuing networks

In this section we recapitulate some of the well known results concerning product-form stochastic networks. Let  $m$  be the number of nodes of a closed Gordon-Newell queuing network with population  $k$ . Let  $N_i$  be the random variable which describes the number of customers at node  $i$  in steady state, and the vector  $\mathbf{N}$  given by  $\mathbf{N} = (N_1, \dots, N_m)$ . Let  $\mathbf{n} = (n_1, \dots, n_m)$  denote a vector whose elements are non-negative integers. Considering networks of fixed population  $k$ , the state space of  $\mathbf{N}$

is:

$$S = \{\mathbf{n} | n_i \geq 0; \sum_i n_i = k; 1 \leq i \leq m\}. \quad (5.1)$$

The steady state joint queue length probability distribution is given by:

$$\pi(\mathbf{n}) = \frac{1}{g_m(k)} \prod_{i=1}^m r_i(n_i), \quad \mathbf{n} \in S \quad (5.2)$$

where  $g_m(k)$  is the normalising constant of a network of  $m$  nodes and population  $k$  and  $r_i(n_i)$  is given by:

$$r_i(n_i) = \frac{e_i^{n_i}}{\prod_{j=1}^{n_i} \mu_i(j)} \quad (5.3)$$

where  $e_i$  is the visitation rate of node  $i$  and  $\mu_i(j)$  is the service rate of node  $i$  when its queue length is  $j$ . The normalising constant can be obtained by summing the product of all quantities  $r_i(n_i)$  over the entire state. We obtain:

$$g_m(k) = \sum_{\mathbf{n} \in S} \prod_{i=1}^m r_i(n_i) \quad (5.4)$$

Note that the number of terms in the summation above (i.e. the number of elements in the set  $S$ ) can be determined by simple combinational arguments. It is given by the number of ways of putting  $k$  balls into  $m$  bags, i.e.

$$|S| = \binom{k+m-1}{m-1} = \frac{(k+m-1)!}{k!(m-1)!} \quad (5.5)$$

Evaluating the normalising constant is important in its own right. Performance measures such as network throughput, node utilisation and mean queue length can be obtained from ratios of the normalising constant at different populations. However, direct computation using equation 5.4 is inefficient particularly when the population or the number of nodes is large in view of the combinatorial size of the state space. Moreover, this method can be inaccurate due to numerical errors introduced by the representation of these numbers.

### 5.2.1 The visitation rate

The visitation rate vector  $\mathbf{e} = (e_1, \dots, e_m)$  is the solution of the following system of linear equations:

$$(P - I)\mathbf{e} = \mathbf{0} \quad (5.6)$$

where  $P = [p_{ij}]$  is a  $m \times m$  routing matrix, i.e. a matrix whose element  $p_{ij}$  is the constant probability that a customer goes to node  $j$  after leaving node  $i$ . This system of equations has infinitely many solutions which differ by a multiplicative factor, therefore:

$$\mathbf{e} = v\mathbf{e} \quad (5.7)$$

where  $e$  is an arbitrary constant and  $\mathbf{v}$  is the solution of the system of equations 5.6 with  $v_1 = 1$ .

### 5.2.2 The convolution theorem

The approach used in this analysis is based on the convolution theorem (see [HaPa 92]) which considers the normalising constant  $g_m(k)$  as an element of a sequence  $\{g_i(k)\}$  ( $i = 1, \dots, m$ ). This theorem shows that its corresponding generating function  $G_m(z)$  can be expressed in terms of a product of simpler node functions.

We define the marginal load  $\rho_i(j)$  as being:

$$\rho_i(j) = \begin{cases} 1 & j = 0 \\ \frac{e_i}{\mu_i(j)} & j \geq 1 \end{cases} \quad (5.8)$$

Substituting equation 5.8 in equation 5.3 we obtain:

$$r_i(k) = \prod_{j=0}^k \rho_i(j) \quad (5.9)$$

The generating function  $R_i(z) = \sum_{k=0}^{\infty} r_i(k)z^k$  of the sequence  $\{r_i(j)\}$  is called the *node  $i$  generating function*. This generating function is improper since the series

$\sum r_i(k)$  does not necessarily converge. Therefore  $R_i(z)$  is defined only inside the unit circle  $|\rho_i z| < 1$ .

The convolution theorem states that the generating function of the normalising constant is given by the product of all node generating functions. So we have:

$$G_m(z) = \sum_{k=0}^{\infty} g_m(k) z^k = \prod_{i=1}^m R_i(z) \quad (5.10)$$

We evaluate  $R_i(z)$  for some cases:

- Nodes with constant service rate

Let  $i$  be a node with service rate  $\mu_i(j)$  independent of the number of customers  $j$  in the queue. Therefore we have  $\mu_i(j) = \mu_i$ ,  $r_i(k) = \rho_i^k$  and  $\rho_i = \frac{e_i}{\mu_i}$ . Thus:

$$R_i(z) = \frac{1}{1 - \rho_i z} \quad (5.11)$$

- Infinite server node

The infinite server node (IS) is a queue where the number of servers is always greater than the number of customers (so there is no queuing). In this case we have  $\mu_i(j) = j\mu_i$ ;  $j = 1, 2, \dots$ . The  $k$ -th element of the sequence  $\{r_i(j)\}$  can be written as:

$$r_i(k) = \prod_{j=1}^k \frac{e_i}{j\mu_i} = \frac{\rho_i^k}{k!}, \quad k = 1, 2, \dots \quad (5.12)$$

where  $\rho_i = \frac{e_i}{\mu_i}$ . Assuming  $r_i(0) = 1$  the node generating function is given by:

$$R_i(z) = \exp(\rho_i z) \quad (5.13)$$

We observe that the mean service time in a infinite server node (i.e.  $\mu_i^{-1}$ ) is referred to as *think time*, since IS server nodes are often used to represent the time a user takes to generate tasks (i.e. customers) in processor-sharing systems.

- Multiple parallel servers

The flow equivalent server of a set of  $m$  parallel queues each one with the same service rate  $\mu_i$  is given by:

$$\mu_i(j) = \frac{j \cdot m}{m + j - 1} \mu_i \quad (5.14)$$

The ratio  $\rho_i(j)$  is given by:

$$\rho_i(j) = \begin{cases} 1 & j = 0 \\ \frac{m + j - 1}{j \cdot m} \rho_i & j \geq 1 \end{cases} \quad (5.15)$$

where  $\rho_i = \frac{e_i}{\mu_i}$ . Thus we have:

$$r_i(k) = \prod_{j=1}^k \frac{m + j - 1}{j \cdot m} \rho_i = \left(\frac{\rho_i}{m}\right)^k \frac{(m + k - 1)!}{k!(m - 1)!} \quad (5.16)$$

The generating function of this sequence is therefore given by:

$$R_i(z) = \frac{1}{(1 - \frac{\rho_i}{m} z)^m} \quad (5.17)$$

- Load Dependence due to Heffes

Heffes (see [Hef 82]) generalised the load dependence given in the previous example and considered the following expression for  $\mu_i(j)$ :

$$\mu_i(j) = \frac{j}{a + j} \mu_i \quad (5.18)$$

where  $\mu_i$  and  $a$  are positive real constants. In this case, the ratio  $\rho_i(j) = \frac{e_i}{\mu_i(j)}$  is given by:

$$\rho_i(j) = \left(1 + \frac{a}{j}\right) \rho_i \quad (5.19)$$

where  $\rho_i = \frac{e_i}{\mu_i}$ . Thus from equation 5.9 we have:

$$r_i(k) = \rho_i(k) r_i(k - 1) \quad (5.20)$$

Substituting equation 5.19 in 5.20 multiplying by  $z^k$  and summing we obtain the following first order differential equation:

$$(1 - \rho_i z)R'_i(z) - \rho_i(1 + a)R_i(z) = 0 \quad (5.21)$$

Solving equation 5.21 with the boundary condition  $R_i(0) = 1$  we obtain:

$$R_i(z) = \frac{1}{(1 - \rho_i z)^{a+1}} \quad (5.22)$$

- Linear load dependence

The same method can be used to evaluate  $R_i(z)$  when  $\mu_i(j) = (j - a)\mu_i$ . We observe that IS servers represent a particular case of linear load dependence ( $a = 0$ ). This produces the following differential equation:

$$zR'_i(z) - (a + \rho_i z)R_i(z) = 0 \quad (5.23)$$

where  $\rho_i = \frac{\epsilon_i}{\mu_i}$ . Solving this equation with boundary condition  $R_i(0) = 1$  we obtain:

$$R_i(z) = z^a \exp(\rho_i z) \quad (5.24)$$

## 5.3 Expressions for the normalising constant in product-form networks

As we have seen in section 5.2, there are various nodes in which the generating function  $R_i(z)$  can be expressed as a rational term and/or an exponential. Consequently, the generating function of the normalising constant of a queuing network is a product of rational polynomials multiplied by an exponential term. In these cases, it is convenient to transform the rational functions of  $z$  as a sum of simple

terms. Since the sum of transforms equals the transform of the sum we can invert the resulting expression term by term.

In the subsequent section we apply this technique to various queuing networks and obtain a closed expression for their normalising constant.

We can classify a QN according to the generating function of its normalising constant. We define a *single pole* as a network in which  $G_m(z)$  is rational and in which its denominator is a polynomial whose roots have a multiplicity of one. In the same way, a *multiple pole* is therefore a network with at least one root which has a multiplicity greater than one.

We are also interested in the behaviour of networks whose number of nodes approaches infinity. In this case, it is often reasonable (e.g. if the network is load-balanced) to assume the same constant load,  $\rho$ , for all nodes. These are defined as *homogeneous networks*. Finally, we define *combined networks* as networks which contain both load independent and IS load dependent nodes. The generating function of their normalising constant is expressed as an exponential term divided by a polynomial.

### 5.3.1 Single pole networks with constant service rates

We consider a network with  $m$  nodes and constant service rates  $\mu_i$ , ( $1 \leq i \leq m$ ). This network is a single pole when  $\rho_i \neq \rho_j$  if  $i \neq j$ , ( $1 \leq i \leq m, 1 \leq j \leq m$ ). In this case the expression for the generating function of the normalising constant is given by:

$$G_m(z) = \prod_{i=1}^m \frac{1}{(1 - \rho_i z)} \quad (5.25)$$

We notice that the denominator of 5.25 is a polynomial in  $z$  expressed in its factored form. Clearly, this polynomial has  $m$  roots of multiplicity one located at

$z = \frac{1}{\rho_i}$ . If  $G_m(z)$  is in this form then it is possible to express it as follows:

$$G_m(z) = \sum_{i=1}^m \frac{A_i}{(1 - \rho_i z)} \quad (5.26)$$

Let  $\alpha_{ij}$  be the ratio  $\alpha_{ij} = \frac{\rho_j}{\rho_i}$ . The coefficients  $A_i$  are described as a function of  $\alpha_{ij}$ .

$$A_i = \prod_{j \neq i} \frac{1}{(1 - \alpha_{ij})} \quad (5.27)$$

The expression for the normalising constant  $g_m(k)$  is obtained by anti-transforming 5.26. Thus we have:

$$g_m(k) = \sum_{i=1}^m A_i \rho_i^k \quad (5.28)$$

Observe that equation 5.28 is remarkably simpler than equation 5.4 since it is given in its closed form. We also observe that the terms  $A_i \rho_i^k$  are negligible for large  $k$ , if  $\rho_i < 1$ . The node with the largest  $\rho$  is referred as the network *bottleneck*. We shall discuss this in more detail in section 5.4.

### 5.3.2 Homogeneous networks with constant service rates

A homogeneous QN has  $m$  nodes with identical load  $\rho$ . This case is particularly important when we consider networks of large numbers of nodes, i.e, when  $m$  is large. Its analysis provides an upper bound on the performance of perfectly balanced systems.

Knowing that each node generating function  $R_i(z)$  is given by equation 5.11 and using the convolution theorem, we have the following expression for the generating function of the normalising constant:

$$G_m(z) = \frac{1}{(1 - \rho z)^m} \quad (5.29)$$

Anti-transforming 5.29 we obtain:

$$g_m(k) = \frac{(k + m - 1)!}{k!(m - 1)!} \rho^k \quad (5.30)$$

Notice that we are usually interested in evaluating the ratio  $\frac{g_m(k-1)}{g_m(k)}$ . Therefore we do not need to calculate the factorials of equation 5.30. We note that the case of multiple parallel servers is an instance of a homogeneous network (see equation 5.17).

### 5.3.3 Multiple pole networks with constant service rates

In this section we consider a queuing network with constant service rates where there are more than one node with a given load  $\rho_i$ . Let  $m$  be the number of nodes of such network and  $n$  be the number of different loads  $\rho_i$  ( $1 \leq i \leq n$ ;  $n < m$ ). Let  $M_i$  be the set of nodes with the same load  $\rho_i$  and  $m_i$  be the number of elements of  $M_i$  ( $1 \leq i \leq n$ ). We say that the load  $\rho_i$  has multiplicity  $m_i$ . Clearly we have  $m_1 + m_2 + \dots + m_n = m$ . The generating function of the normalising constant of this network is given by:

$$G_m(z) = \frac{1}{\prod_{i=1}^n (1 - \rho_i z)^{m_i}} \quad (5.31)$$

Comparing this expression with equation 5.25 we notice that we also have in this case a ratio expression, in which the denominator is a polynomial in  $z$  expressed in its factored form. This polynomial has  $n$  roots, each one of multiplicity  $m_i$  ( $1 \leq i \leq n$ ). Clearly, these roots are located at  $z = \frac{1}{\rho_i}$ .

We can transform equation 5.31 in its sum form, in the same way we have done in section 5.3.1. But in this case we obtain  $m_i$  terms for each load  $\rho_i$ , so the resulting equation is given by:

$$\begin{aligned} G_m(z) = & \frac{A_{11}}{(1 - \rho_1 z)^{m_1}} + \frac{A_{12}}{(1 - \rho_1 z)^{m_1-1}} + \dots + \frac{A_{1m_1}}{(1 - \rho_1 z)} \\ & + \frac{A_{21}}{(1 - \rho_2 z)^{m_2}} + \frac{A_{22}}{(1 - \rho_2 z)^{m_2-1}} + \dots + \frac{A_{2m_2}}{(1 - \rho_2 z)} \\ & + \dots \\ & + \frac{A_{n1}}{(1 - \rho_n z)^{m_n}} + \frac{A_{n2}}{(1 - \rho_n z)^{m_n-1}} + \dots + \frac{A_{nm_n}}{(1 - \rho_n z)} \end{aligned} \quad (5.32)$$

The coefficients  $A_{ij}$  can be obtained iteratively by operations between simple sequences. We initially introduce some auxiliary variables and sequences for each row  $i$  of  $A_{ij}$ . Then, we use a recursive equation to evaluate a sequence  $\{h_i(k)\}$ , whose elements are related to columns  $j$  of  $A_{ij}$ . This method is described as follows.

With each load  $\rho_j$  ( $j \neq i$ ) we associate the ratios  $\alpha_j$  and  $\beta_j$  defined as (for a fixed  $i$ ):

$$\alpha_j = \frac{\rho_j}{\rho_i} \quad (5.33)$$

$$\beta_j = \frac{\rho_j}{\rho_i - \rho_j} \quad (5.34)$$

To each ratio  $\beta_j$  we associate one sequence  $\{s_j(k)\}$  defined as:

$$s_j(k) = \frac{(m_j + k - 1)!}{(m_j - 1)!k!} (-\beta_j)^k \quad (5.35)$$

Observe that the generating function of this sequence is given by:

$$S_j(z) = \frac{1}{(1 + \beta_j z)^{m_j}} \quad (5.36)$$

Let the sequence  $\{h_i(k)\}$  be the convolution of the sequences  $\{s_j(k)\}$  ( $1 \leq j \leq n, j \neq i$ ) associated with  $R_i$ , i.e.

$$h_i(k) = \sum_{k_1 + \dots + k_m = k} \prod_{j \neq i} s_j(k_j).$$

We call this sequence the  $i$ -th characteristic sequence of  $G_m(z)$ .

The following proposition relates the elements of the sequence  $\{h_i(k)\}$  to the coefficients  $A_{ij}$ :

**Proposition 2** *Let  $\{h_i(k)\}$  be the  $i$ -th characteristic sequence of the generating function  $G_m(z)$ . Then, the coefficients  $A_{ij}$  are given by:*

$$A_{ij} = H_i^0 h_i(j - 1) \quad j = 1, 2, \dots, m_i \quad (5.37)$$

where  $H_i^0$  is defined by:

$$H_i^0 = \prod_{k=1; k \neq i}^n \frac{1}{(1 - \alpha_k)^{m_k}} \quad (5.38)$$

**Proof**

The general expression for the coefficients  $A_{ij}$  is given by (see, e.g. [Klei 75]):

$$A_{ij} = \frac{1}{(j-1)!} \left( -\frac{1}{\rho_i} \right)^{j-1} \frac{d^{j-1}}{dz^{j-1}} H_i(z) \Big|_{z=\frac{1}{\rho_i}} \quad (5.39)$$

where the function  $H_i(z)$  is defined as follows:

$$H_i(z) = (1 - \rho_i z)^{m_i} G_m(z) = \frac{1}{\prod_{j=1; j \neq i}^n (1 - \rho_j z)^{m_j}} \quad (5.40)$$

We define the function  $H_i^*(x)$  by letting  $x = 1 - \rho_i z$ , and substituting in equation 5.40. We obtain:

$$H_i^*(x) = H_i^0 \prod_{j=1; j \neq i}^n \frac{1}{(1 + \beta_j x)^{m_j}} \quad (5.41)$$

Knowing that  $(1/\rho)^{j-1} (d/dz)^{j-1} H_i(z) \Big|_{z=1/\rho_i} = (d/dx)^{j-1} H_i^*(x) \Big|_{x=0}$  and using the intermediate value theorem (i.e.  $1/n! (d/dz)^n F(z) \Big|_{z=0} = f_n$  where  $F(z) = \sum f_n z^n$ ) the result immediately follows  $\square$ .

**Remarks**

- The elements of the sequence  $\{s_j(k)\}$  can be obtained iteratively as follows:

$$s_j(k) = \begin{cases} 1 & k = 0 \\ -\frac{m_j + k - 1}{k} \beta_j s_j(k-1) & k = 1, 2, \dots \end{cases} \quad (5.42)$$

- The elements of the sequence  $\{h_i(k)\}$  can be obtained using a recurrence formula to compute a convolution of sequences. We use auxiliary sequences  $\{x_j(k)\}$  to accumulate convolutions of  $\{s_j(k)\}$ . Without loss of generality

we assume  $i = n$  to simplify our notation. Then, we obtain the following equations:

$$\begin{aligned} x_1(k) &= s_1(k) \\ x_j(k) &= \sum_{l=0}^k x_{j-1}(l) s_j(k-l) \quad j = 2, 3, \dots, n-1 \end{aligned} \quad (5.43)$$

The resulting element of the sequence  $\{h_i(k)\}$  will be:

$$h_i(k) = x_{n-1}(k) \quad (5.44)$$

This completes the description of our method. Its advantage is that it produces an efficient algorithm for numerical evaluation of the coefficients  $A_{ij}$ .

We have seen so far that the generating function of the normalising constant of a multiple pole network with constant service rates can be expressed as a sum of terms where each of them looks like a homogeneous QN of the sort we studied in section 5.3.2. To obtain the expression for the sequence  $\{g_m(k)\}$  we invert term by term the equation 5.32 using equation 5.30. This produces the following expression:

$$g_m(k) = \sum_{i=1}^n \sum_{j=1}^{m_i} A_{ij} \frac{(m_i + k - j)!}{(m_i - j)! k!} \rho_i^k \quad (5.45)$$

The equation above produces a very efficient algorithm for the evaluation of  $g_m(k)$  since it consists of a summation of only  $\sum_i m_i$  terms. To illustrate how the evaluation of the normalising constant is dramatically simplified we give an example of a QN comprising 20 nodes with load  $\rho_1$  and 30 nodes with load  $\rho_2$ . If we want to evaluate  $g_{50}(1000)$  we will need to carry out a summation of only 50 terms. If equation 5.4 were used we would need to perform a summation of  $\binom{1049}{49} = 1.09 \times 10^{86}$  terms (see equation 5.5). Another advantage of using this equation is that the number of terms in the summation is independent of  $k$ , therefore the algorithm is equally efficient for moderate or high populations.

However, in networks with large number of nodes the summation above involves also a considerably larger number of terms. In this situation it is more convenient to obtain an integral representation of equation 5.45. In this way the computational task of summing  $m$  terms is reduced to the numerical evaluation of one integral. A method to obtain this integral representation will be presented in section 5.4.

### 5.3.4 Infinite server networks

We now evaluate the normalising constant of an IS queuing network, i.e. a queuing network where there is no service degradation as the number of customers increases.

Suppose that  $m$  is the number of nodes of the network. The service rate of the  $i$ -th node ( $1 \leq i \leq m$ ) is proportional to the number  $j$  of customers in the queue. So we have:

$$\mu_i(j) = j\mu_i \quad (5.46)$$

To obtain the generating function of the normalising constant we use the convolution theorem given in section 5.2.2. We know that the node generating function of node  $i$  is given by equation 5.13. The expression for  $G_m(z)$  is the product of the node generating function of each node. Thus we have:

$$G_m(z) = \exp(\rho_0 z) \quad (5.47)$$

where  $\rho_0 = \sum_{i=1}^m \rho_i$ . It is noteworthy but not surprising that the parameters  $\rho_i$  for the IS centers appear lumped together. We observe that a similar result is also derived by J. McKenna and D. Mitra (see [McMi 82]).

The expression for the normalising constant is the anti-transform of equation 5.47. It is given by:

$$g_m(k) = \frac{(\rho_0)^k}{k!}, \quad k = 0, 1, \dots \quad (5.48)$$

### 5.3.5 Combined networks

In this section we combine the results obtained in sections 5.3.3 and 5.3.4 to analyze a more general queuing network where part of it is IS load-dependent. Let  $m$  be the number of nodes of such a network. We assign loads  $\rho_1, \dots, \rho_n$  to nodes with constant service rate of multiplicity  $m_i$  ( $1 \leq i \leq n$ ), and loads  $\rho_{n+1}, \dots, \rho_q$  to IS nodes with service rates given by equation 5.46.

Using the convolution theorem, the generating function of the normalising constant can be derived from equations 5.31 and 5.47. This yields:

$$G_m(z) = \exp(\rho_0 z) \prod_{i=1}^n \frac{1}{(1 - \rho_i z)^{m_i}} \quad (5.49)$$

Observe that in this case the load  $\rho_0$  is given by:

$$\rho_0 = \sum_{i=n+1}^q \rho_i = \sum_{i=n+1}^q \frac{e_i}{\mu_i} \quad (5.50)$$

Comparing equation 5.49 with equation 5.31 we notice that we have a rational expression multiplied by an exponential term. The denominator of the rational expression is also a polynomial in  $z$  expressed in its factored form. Using the same method described in section 5.3.3, we transform the rational part of  $G_m(z)$  in a sum of terms. Thus we obtain:

$$G_m(z) = \exp(\rho_0 z) \sum_{i=1}^n \sum_{j=1}^{m_i} \frac{A_{ij}}{(1 - \rho_i z)^{m_i - j + 1}} \quad (5.51)$$

Equation 5.50 shows us that load  $\rho_0$  can be expressed in terms of the visitation rates of nodes  $n + 1$  to  $q$ . Looking at equation 5.7 we observe that these visitation rates are proportional to an arbitrary constant  $e$ . Then we can choose a convenient value for this constant  $e$  such that  $\rho_0 = 1$ . This value is given by:

$$e = \left( \sum_{i=n+1}^q \frac{v_i}{\mu_i} \right)^{-1} \quad (5.52)$$

Therefore, without loss of generality we assume  $\rho_0 = 1$ .

The following propositions are used to obtain the anti-transform of equation 5.49:

**Proposition 3** *Let  $\rho$  be a real constant. Then, the elements of the sequence  $\{r(k)\}$  whose generating function is  $R(z) = \frac{\exp(z)}{(1-\rho z)}$  are given by:*

$$r(k) = [\exp(\rho^{-1}) - e(k+1)]\rho^k, \quad k = 0, 1, \dots \quad (5.53)$$

where  $e(k) = \sum_{i=k}^{\infty} \frac{\rho^{-i}}{i!}$ .

**Proof**

The elements of the sequence  $\{r(k)\}$  are the coefficients of  $z^k$  in  $R(z)$ . Hence:

$$r(k) = \sum_{i=0}^k \frac{1}{(k-i)!} \rho^i \quad (5.54)$$

Dividing both sides of 5.54 by  $\rho^k$  and changing the summation variable we obtain:

$$\frac{r(k)}{\rho^k} = \sum_{i=0}^k \frac{\rho^{-i}}{i!} \quad (5.55)$$

We know that the serial expansion of  $\exp(\rho^{-1})$  is given by:

$$\exp(\rho^{-1}) = \sum_{i=0}^k \frac{\rho^{-i}}{i!} + e(k+1) \quad (5.56)$$

Substituting equation 5.55 in 5.56 we obtain 5.53.  $\square$

**Remarks**

- If  $k$  is sufficiently large the elements of the sequence  $\{r(k)\}$  can be approximated by:

$$r(k) \sim \rho^k \exp(\rho^{-1}) \quad (5.57)$$

An upper bound  $\varepsilon(k+1)$  on the error of this approximation is given by:

$$\varepsilon(k+1) = \frac{\rho^{-(k+1)} \exp(\rho^{-1})}{(k+1)!} \quad (5.58)$$

- The error function  $\varepsilon(k)$  decreases strictly when  $k > \rho^{-1}$  (since  $\frac{\varepsilon(k)}{\varepsilon(k+1)} = \frac{k+1}{\rho^{-1}}$ ) and goes asymptotically to 0 as  $k$  goes to infinity.

**Proposition 4** *Let  $\rho$  be a real constant and  $j$  a non negative integer. Then, the elements of the sequence  $\{r(k)\}$  whose generating function is  $R(z) = \frac{\exp(z)}{(1-\rho z)^{j+1}}$  are given by:*

$$r(k) = \sum_{i=0}^k \frac{(j+i)!}{j! i!} \frac{\rho^i}{(k-i)!} \quad (5.59)$$

**Proof**

*By using the same procedure as for proposition 3.  $\square$*

The exact formula for the normalising constant can be obtained by anti-transforming equation 5.51 using equations 5.53 and 5.59. Thus we obtain:

$$g_m(k) = \sum_{i=1}^n \sum_{j=1}^{m_i} A_{ij} S_{i, m_i-j}(k) \quad (5.60)$$

where  $S_{ij}(k)$  is given by:

$$S_{ij}(k) = \sum_{l=0}^k \frac{(j+l)!}{j! l!} \frac{\rho_i^l}{(j-l)!} \quad (j = 0, \dots, m_i - 1; i = 1, \dots, n) \quad (5.61)$$

Although the number of terms of the equation above is still dependent on  $m$  and  $k$ , the computation of the normalising constant is considerably simplified since we have a linear dependence against a combinatorial dependence given by equation 5.4 (we compare our method with some of the existing ones in chapter 6).

As we have seen in section 5.4.3, in large networks it is more convenient to obtain an integral representation of the normalising constant. This will be introduced in the next section.

### 5.3.6 More general queuing networks

In this section we derive expressions for the normalising constant for networks with IS and load independent nodes. However, our method can handle many cases of load dependence (as we have seen in section 5.2) any whose generating function can be expressed as a ratio of polynomials.

Another interesting application of our method is in networks with *multi-server nodes*. A multi-server node  $i$  is a node which contains  $c$  servers. Its node load is therefore  $\rho_i(j) = \rho_i/j$  if  $j \leq c$ , and  $\rho_i(j) = \rho_i/c$  otherwise. This produces the following node sequence  $r_i(k)$ :

$$r_i(k) = \prod_{j=0}^k \rho_i(j) = \begin{cases} \frac{\rho_i^k}{k!} & k \leq c \\ P\left(\frac{\rho_i}{c}\right)^{k-c-1} & k > c \end{cases} \quad (5.62)$$

where  $P = \frac{\rho_i^{c+1}}{(c c!)}$ . The node  $i$  generating function  $R_i(z)$  is therefore:

$$R_i(z) = \sum_{n=0}^c \frac{(\rho_i z)^n}{n!} + \frac{P z^{c+1}}{(1 - \frac{\rho_i}{c} z)} \quad (5.63)$$

We observe that equation 5.63 can still be expressed as  $\frac{N(z)}{D(z)}$ , where  $N(z)$  and  $D(z)$  are polynomials in  $z$ . Therefore it is possible to apply our method. However, the algorithm proposed in section 5.3.3 needs to be extended to consider  $N(z)$  in the calculation of the coefficients  $A_{ij}$ . This is a suggestion for further research.

## 5.4 Asymptotic properties

In this section we study the behaviour of the normalising constant in large networks, i.e., networks with large populations. We observe that in some simple cases an exact closed formula for the ratio  $g_m(k-1)/g_m(k)$  can be calculated. However, for the cases in which these formulas can not be obtained, we introduce an integral representation of  $g_m(k)$ . Numerical evaluation of these equations can be carried out by the use of standard methods of numerical integration.

### 5.4.1 Single pole networks

We consider a QN of  $m$  nodes whose corresponding node loads  $\rho_i (1 \leq i \leq m)$  have multiplicity 1. Let  $\rho_1$  be the largest node load. From equation 5.28 we have:

$$g_m(k) = \sum_{i=1}^m A_i \rho_i^k = A_1 \rho_1^k \left(1 + \sum_{i=2}^m \frac{A_i}{A_1} \alpha_i^k\right) \quad (5.64)$$

where  $\alpha_i = \rho_i/\rho_1$  ( $i = 2, \dots, m$ ).

We observe that as  $k \rightarrow \infty$  we have  $\alpha_i^k \rightarrow 0$ , since  $\alpha_i < 1$ . Then for sufficiently large  $k$  the following approximation can be used:

$$g_m(k) \sim A_1 \rho_1^k \quad (5.65)$$

Substituting equation 5.65 in equation 5.2 we have the following expression for the joint queue length probability distribution:

$$\pi(\mathbf{n}) \sim \frac{1}{A_1} \prod_{i=2}^m \alpha_i^{n_i} \quad (5.66)$$

Knowing that  $\alpha_i < 1$  we expect the joint probabilities  $\pi(\mathbf{n})$  to be greater in states in which  $n_i$  is low ( $2 \leq i \leq m$ ). In this case, for a large  $k$  we have greater joint probabilities when  $n_1$  is large, since  $n_1 = k - \sum_i n_i$ . This implies that a great number of customers will form in node one, referred as *bottleneck* of the network. This qualitative behaviour of QN has been previously discussed in [Klei 75] for example.

### 5.4.2 Homogeneous networks

We consider in this section a homogeneous network in which the number of queues  $m$  and the population  $k$  go to infinity at constant ratio  $C = \frac{m}{k}$ . The node load  $\rho$  is constant throughout the network.

Let  $U(m, k)$  be the ratio of the normalising constant at populations  $k-1$  and  $k$  respectively. We are interested in the limit  $L = \lim_{m, k \rightarrow \infty} U(m, k)$  when  $\frac{m}{k} = C$ . Using equation 5.30 we obtain:

$$L = \lim_{k \rightarrow \infty} \frac{\rho^{-1}}{C + 1 - \frac{1}{k}} = \frac{\rho^{-1}}{C + 1} \quad (5.67)$$

This limit can be used to evaluate performance parameters such as throughput, mean queue length, etc.

### 5.4.3 Multiple pole networks

In this section we analyse a queuing network as defined in section 5.3.3. We concentrate our analysis on the particular case where there are a large number of nodes of same node load  $\rho_1$  of multiplicity  $m_1$ , and one node with load  $\rho_2$  and multiplicity 1 so that  $\rho_1 < \rho_2$ . The generating function of the normalising constant is then given by:

$$G_m(z) = \frac{1}{(1 - \rho_1 z)^{m_1} (1 - \rho_2 z)} = \sum_{j=1}^{m_1} \frac{A_{1j}}{(1 - \rho_1 z)^{m_1+j-1}} + \frac{A_{21}}{(1 - \rho_2 z)} \quad (5.68)$$

The exact expression for  $g_m(k)$  can be obtained by using the method of section 5.3.3. In order to evaluate its asymptotic value when the number of nodes is large we propose a method based on the integral approximation of finite series.

Initially we point out that the coefficients  $A_{1j}$  and  $A_{21}$  can be calculated using the method proposed in section 5.3.3. We can easily verify that  $A_{1j} = H_1^0 (-\beta)^{j-1}$  and  $A_{21} = H_2^0$ , where  $H_1^0$ ,  $H_2^0$  and  $\beta$  are given by:

$$H_1^0 = \frac{\rho_1}{\rho_1 - \rho_2}, \quad H_2^0 = \left( \frac{\rho_2}{\rho_2 - \rho_1} \right)^{m_1}, \quad \beta = \frac{\rho_2}{\rho_1 - \rho_2} \quad (5.69)$$

Substituting  $H_1^0$ ,  $H_2^0$  and  $\beta$  in equation 5.45 and changing the summation variable (i.e. making  $i = m_1 - j$  and substituting in equation 5.45) we obtain:

$$g_m(k) = H_1^0 \rho_1^k (-\beta)^{m_1-1} \sum_{i=0}^{m_1-1} u_k(i) + H_2^0 \rho_2^k \quad (5.70)$$

where  $u_k(i) = \frac{(k+i)!}{k! i!} (-\beta)^{-i}$ . The series  $\sum u_k(i)$  is positive and convergent since  $0 < (-\beta)^{-1} < 1$ . It can be easily verified that this series increases and then decreases monotonically as  $i$  increases. Let  $\gamma$  be the real number such that  $u_k(\gamma)$  increases for every integer  $k < \gamma$ . This number can be calculated by solving the inequality  $u_k(i)/u_k(i-1) > 1$ . Thus we obtain:

$$\gamma = \frac{-k}{\beta + 1} \quad (5.71)$$

We now propose a method to obtain the partial sum  $\sum_{i=0}^{m_1-1} u_k(i)$  by integrating an equivalent continuous function. Let  $u_k^*(x)$  be a continuous function with the property that  $u_k^*(i) = u_k(i)$  at an integer  $i$ . A function  $u_k^*(x)$  can be obtained by recalling Stirling's formula for the factorial function:

$$n! \sim \sqrt{2\pi n} \left(\frac{n}{e}\right)^n \quad (5.72)$$

Thus, the function  $u_k^*(x)$  is given by:

$$u_k^*(x) = \frac{\sqrt{x^{-1} + k^{-1}}}{\sqrt{2\pi}} \left(\frac{x}{k} + 1\right)^k \left(\frac{k}{x} + 1\right)^x (-\beta)^{-x} \quad (5.73)$$

Let  $a$  and  $b$  be real numbers such that  $b > a$ . Our approximation is based on the fact that  $\int_a^b u_k^*(x) dx$  can be used to obtain a good approximation of  $\sum_{i=a}^b u_k(i)$  if  $u_k^*(x)$  either increases or decreases monotonically between  $a$  and  $b$ . Therefore, we need to distinguish the case where  $m_1 < \gamma$  (where  $u_k^*(x)$  increases) from the case where  $m_1 > \gamma$  (where  $u_k^*(x)$  decreases).

- If  $m_1 \leq \gamma$  we propose the following approximation:

$$\sum_{i=0}^{m_1-1} u_k(i) \sim \int_0^{m_1-1} u_k^*(x) dx + \frac{1}{2} u_k(m_1 - 1) \quad (5.74)$$

The maximum error  $\varepsilon$  of this approximation (neglecting the error introduced by using equation 5.72) is given by:

$$\varepsilon = \frac{1}{2} u_k(m_1 - 1) \quad (5.75)$$

- If  $m_1 > \gamma$  then it is necessary to calculate the total sum  $\sum_{i=0}^{\infty} u_k(i)$ . This sum is clearly  $(1 + \beta^{-1})^{-k-1}$ . This can be verified by obtaining the generating function  $U_k(z) = \sum u_k(i) z^i = (1 + \beta z)^{-k-1}$ . The total sum is given by  $U_k(1)$ .

Thus we propose the following approximation:

$$\sum_{i=0}^{m_1-1} u_k(i) \sim \frac{1}{(1 + \beta^{-1})^{k+1}} - \int_{m_1}^{\infty} u_k^*(x) dx + \frac{1}{2} u_k(m_1 - 1) \quad (5.76)$$

The maximum error  $\varepsilon$  of this approximation (neglecting the error introduced by using equation 5.72) is given by:

$$\varepsilon = \frac{1}{2}u_k(m_1 - 1) \quad (5.77)$$

We observe that by using the integral representation of  $\sum_{i=0}^{m_1-1} u_k(i)$  we avoid doing a summation over  $m_1 - 1$  elements when this quantity is large. Their numerical evaluation can be carried out by using existing methods of numerical integration.

#### 5.4.4 Infinite server networks

We consider a large infinite server queuing network where the number of nodes  $m$  and the population  $k$  go to infinity at constant ratio  $C = \frac{m}{k}$ . Using equation 5.48 we obtain the ratio of the normalising constants at populations  $k - 1$  and  $k$ :

$$U(m, k) = \frac{g_m(k-1)}{g_m(k)} = \frac{\rho^{-1}}{C} \quad (5.78)$$

where  $\rho = \rho_0/m$  is the load in the IS nodes.

#### 5.4.5 Combined networks

We have seen that the exact formula for the normalising constant of a combined queuing network is given by equation 5.60. We notice that this equation contains the coefficients  $A_{ij}$ , which were obtained from the expansion of equation 5.49 into a sum of simpler terms. This equation also contains the finite sums  $S_{ij}(k)$  ( $i = 1, \dots, n$ ) ( $j = 0, \dots, m_i - 1$ ). The finite sum  $S_{ij}(k)$  is the result of a convolution of the exponential sequence with each node generating function  $R_i(z)$ .

In this section we introduce an integral representation of the sums  $S_{ij}(k)$ . This representation is based on Stirling's formula for the factorial function (used in  $S_{i0}(k)$ )

and Euler's integral (used in  $S_{ij}(k)$   $j \geq 1$ ). The numerical evaluation of this expression can be carried out by using standard methods of numerical integration. Furthermore, the advantage of having these integral expressions is that they can be expanded into a more rapidly convergent series. However, further work is required to derive their expansion coefficients.

### The integral representation of $S_{i0}(k)$

Let  $R_i(z) = (1 - \rho_i z)^{-1}$  be the node  $i$  generating function of a combined network and  $r_i(k)$  be the anti-transform of  $R_i(z)$ . We have in section 5.4.3 that the convolution of  $r_i(k)$  with the exponential sequence (i.e.  $s(k) = 1/k!$ ) is given by:

$$S_{i0}(k) = \rho_i^k \sum_{l=0}^k \frac{\rho_i^{-l}}{l!} \quad (5.79)$$

Let  $v_i(l) = \rho_i^{-l}/l!$ . Obviously the series  $\sum v_i(l)$  converges to  $\exp(\rho_i^{-1})$ . Let  $v_i^*(x)$  be a continuous function with the property that  $v_i^*(l) = v_i(l)$  ( $l = 0, 1, \dots$ ). Using Stirling's formula we obtain:

$$v_i^*(x) = \frac{(x\rho_i)^{-x}}{\sqrt{2\pi x}} \exp(x) \quad (5.80)$$

The function  $v_i^*(x)$  increases monotonically for  $0 \leq x \leq \gamma = \rho_i^{-1}$  and decreases monotonically for  $x > \gamma$ . Using the integral approximation, we obtain the following expression for the series  $\sum v_i(l)$ :

$$\sum_{l=0}^k v_i(l) \sim \begin{cases} \int_0^k v_i^*(x) dx & k < \gamma \\ \exp(\rho_i^{-1}) - \int_k^\infty v_i^*(x) dx & k \geq \gamma \end{cases} \quad (5.81)$$

The maximum error of this approximation  $\varepsilon$  is given by:

$$\varepsilon = \frac{\rho_i^{-k}}{k!} \quad (5.82)$$

### Remarks

- Clearly for populations  $k \gg \rho_i^{-1}$  the following approximation can be used:

$$S_{i0}(k) \sim \rho_i^k \exp(\rho_i^{-1}).$$

The integral representation of  $S_{ij}(k)$  ( $j \geq 1$ )

Let  $R_i(z) = (1 - \rho_i z)^{-(j+1)}$  ( $i = 1, \dots, n; j = 1, \dots, m_i$ ) be the node  $i$  generating function of a combined network and  $r_i(k)$  the anti-transform of  $R_i(z)$ . The convolution of  $r_i(k)$  with the exponential sequence (i.e.  $s(k) = 1/k!$ ) is a series  $S_{ij}(k)$  given by:

$$S_{ij}(k) = \sum_{i_1+i_2=k} \frac{(j+i_1)!}{j! i_1!} \frac{\rho_i^{i_1}}{i_2!} \quad (5.83)$$

We recall Euler's integral for the factorial function:

$$n! = \int_0^\infty \exp(-u) u^n du \quad (5.84)$$

Returning to equation 5.83, we use this representation to write:

$$(j+i_1)! = \int_0^\infty \exp(-u) u^{j+i_1} du \quad (5.85)$$

Substitution in equation 5.83 yields:

$$S_{ij}(k) = \frac{1}{j!} \int_0^\infty \exp(-u) u^j \sum_{i_1+i_2=k} \frac{(u\rho_i)^{i_1}}{i_1! i_2!} du \quad (5.86)$$

Using the binomial theorem we obtain:

$$S_{ij}(k) = \frac{1}{k! j!} \int_0^\infty \exp(-u) u^j (\rho_i u + 1)^k du \quad (5.87)$$

Note that equation 5.87 gives an alternative method to calculating  $S_{i0}(k)$  by taking  $j = 0$ .

The integral representation of  $g_m(k)$

Substituting equations 5.87 and 5.81 in equation 5.60 we obtain the following integral formula for  $g_m(k)$ :

$$\begin{aligned} g_m(k) = & \sum_{i=1}^n \sum_{j=1}^{m_i-1} A_{ij} \frac{1}{(m_i-j)! k!} \int_0^\infty \exp(-u) u^{m_i-j} (\rho_i u + 1)^k du \\ & + \sum_{i=1}^n A_{i m_i} \rho_i^k S_{i0}(k) \end{aligned} \quad (5.88)$$

The main advantage of adopting equation 5.88 as an alternative method for obtaining  $g_m(k)$  is that we avoid the cumbersome task (at least in computational terms) of summing over  $j$ , when the value assumed by  $j$  is considerably large. We end up with a more concise formula, which can be evaluated by standard methods of numerical integration.

## 5.5 Cell Modelling

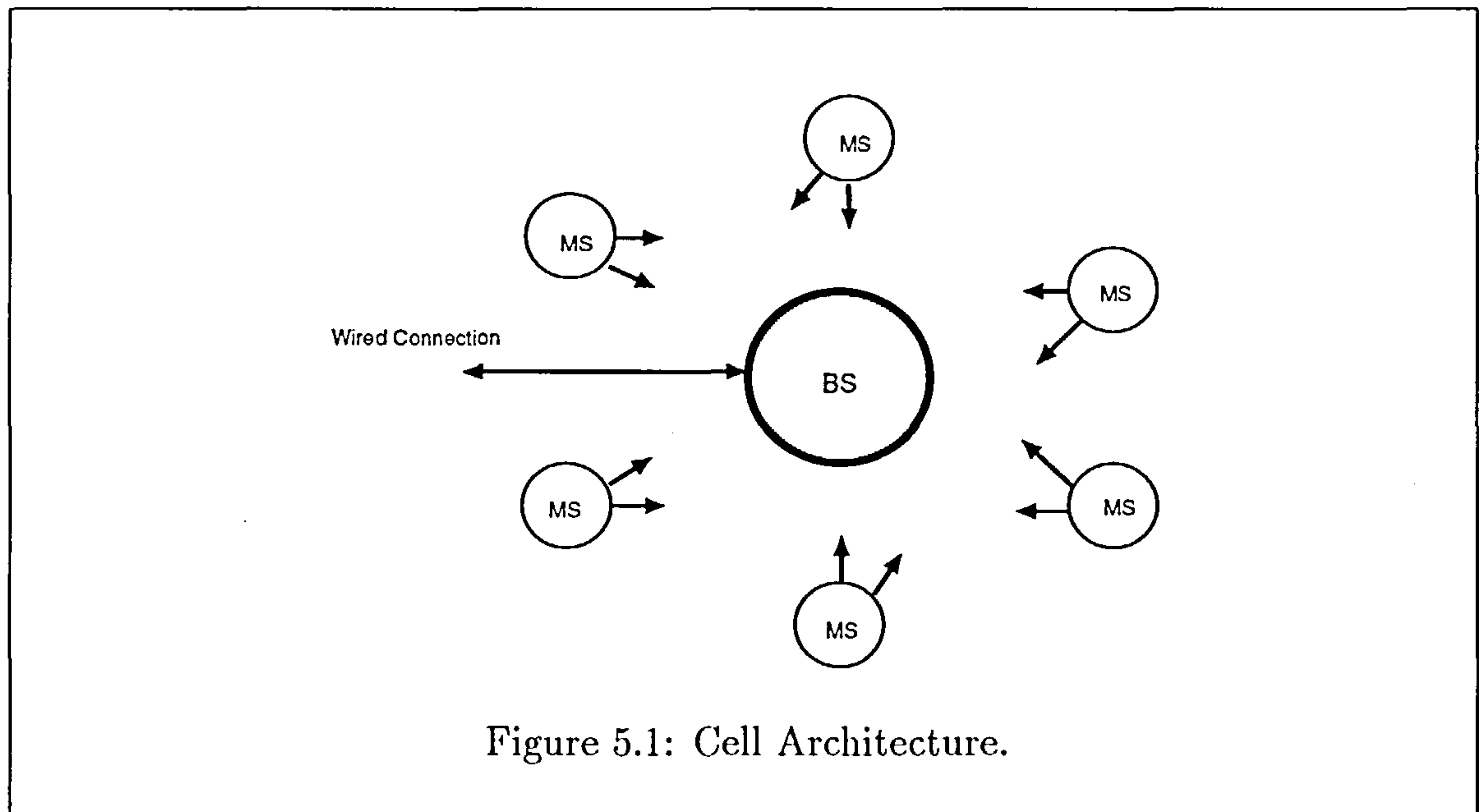
This section develops a closed queuing network model for cellular networks. The cellular network analysed in this section has a corresponding closed queuing network model with non-exponential service time distributions. A method for determining performance values for closed QN with FCFS discipline and general service times is given in [ChHW 75]. However this method utilises ratios of the normalising constant at different populations.

The aim of this section is to apply the results obtained in sections 5.3 and 5.4 to cellular networks and obtain the flow equivalent server (FES) of a given cell. The queuing network model presented in this section is based on the model presented in [DaSm 91] and tailored to the architecture proposed in chapter 2.

### 5.5.1 Cell Architecture

As we have seen in chapter 2, the architecture of the sub-network formed by the stations inside a cell is centralised (see Fig. 5.1). The base station (BS) is responsible for: (i) transmitting data packets to the portable stations inside the cell through multi-channel radio links, (ii) polling the local stations and receiving their data packets (if any), (iii) forwarding packets either to local stations or other base stations.

We also have seen that the group of radio channels allocated for each cell is



divided into three sub-groups: control, uplink and downlink channels. The control group contains a one-way channel used by the base station to inform the downlink and uplink stations addressed in the next time-slot. A portable station senses the control channel and if it is selected for the next downlink time-slot, it tunes in its receiver in the corresponding frequency. In the same way, if a portable station wishes to transmit a packet it waits until it has been selected in the next uplink time-slot and then it tunes its transmitter in the corresponding frequency and transmits the first packet of its buffer.

We suppose that there are  $m$  portable stations inside a cell, and that the base station separates the incoming packets into  $m$  different queues (queues  $Q_1$  to  $Q_m$  in Fig. 5.2). There is one queue for each mobile station. The system so formed can be analysed using a multi-queue model served by a channel controller in cyclic order as we have seen in chapters 3 and 4. There we evaluated performance measures by looking at the base station in isolation. Here, we use a different approach which considers all the stations in a cell. We make the following approximations and simplifying assumptions:

- We consider a balanced system (i.e. queues  $Q_1$  to  $Q_m$  have the same traffic flow).
- We consider an independent server for each queue  $Q_i$  ( $i = 1, \dots, m$ ). The service time represents the time that the head of the queue spends waiting for the channel controller, and its transmission time.

We suppose that each mobile station (MS) contains two separate buffers: the *RX buffer* and the *TX buffer* (queues  $Q_{m+1}$  to  $Q_{3m}$  in Fig. 5.2). The RX buffer contains the packets received in the downlink channel. The packets stored in the RX buffer are delivered to the final user and the packets in the TX buffer are transmitted in the uplink channels. Observe that only one packet is transmitted at a time. The rate in which the packets are transmitted in the uplink channel is controlled by the base station.

### 5.5.2 Queuing network model

We associate each buffer described in section 5.5.1 with a FCFS queue and a packet with a customer. The queuing model so formed is shown in Fig. 5.2. We can see a *think center*, which is responsible for the packet consumption/generation. This think center operates as an infinite server (IS) queue, i.e. every time a packet arrives it stays there for a random period of time. After this time the packet goes to the mobile station to be transmitted in the uplink channel.

Random delays are also used in our model to represent the effect of packet loss due to error. A transmitted packet can be lost in the downlink and uplink channels with probabilities  $p_u$  and  $p_d$  respectively. Every time a packet is lost, it is recovered after a random delay (with rates  $\mu_u$  and  $\mu_d$ ). Hence, the generation of recovered packets is also represented by infinite servers (IS).

### 5.5.3 The flow equivalent server of the cell

In order to evaluate the flow equivalent server of the cell we short-circuit points  $A$  and  $B$  of figure 5.2, assuming in this way that there is no external traffic. We observe that the corresponding queuing network is combined, since part of it is IS load dependent.

Let the parameter  $\rho$  denote the load of each queue and the indexes 1, 2, 3 and 4 denote the base station TX queues, mobile RX queues, mobile TX queues and base station RX queue respectively.

Let the indexes 5 denote think center, 6 base station error-recovery generator and 7 mobile error-recovery generator. Using equation 5.49 we obtain the following expression for the generating function of the normalising constant:

$$G(z) = \frac{1}{(1 - \rho_1)^m (1 - \rho_2)^m (1 - \rho_3)^m (1 - \rho_4)} \exp(\rho_0 z) \quad (5.89)$$

where the load  $\rho_0$  is given by:

$$\rho_0 = m(\rho_6 + \rho_7) + \rho_5 \quad (5.90)$$

Observe that, as we have done in section 5.3.5, we can choose the visitation rates in such a way that  $\rho_0 = 1$ . Then, we can apply directly the method described in this section and obtain the generating function of the closed cell QN by anti-transforming equation 5.89. This leads to the following equation:

$$g(k) = \sum_{i=1}^3 \sum_{j=1}^m \rho_i^k S_{i,m-j}(k) + A_{41} \rho_4^k S_{40}(k) \quad (5.91)$$

where the scalars  $A_{ij}$  contain the coefficients of the expansion of equation 5.89 in a sum of terms.

We now follow the method described in section 5.3.3 to obtain the coefficients of the expansion of equation 5.89. For simplicity we show here how to obtain the coefficients  $A_{1j}$ . The other coefficients can be calculated in a similar way.

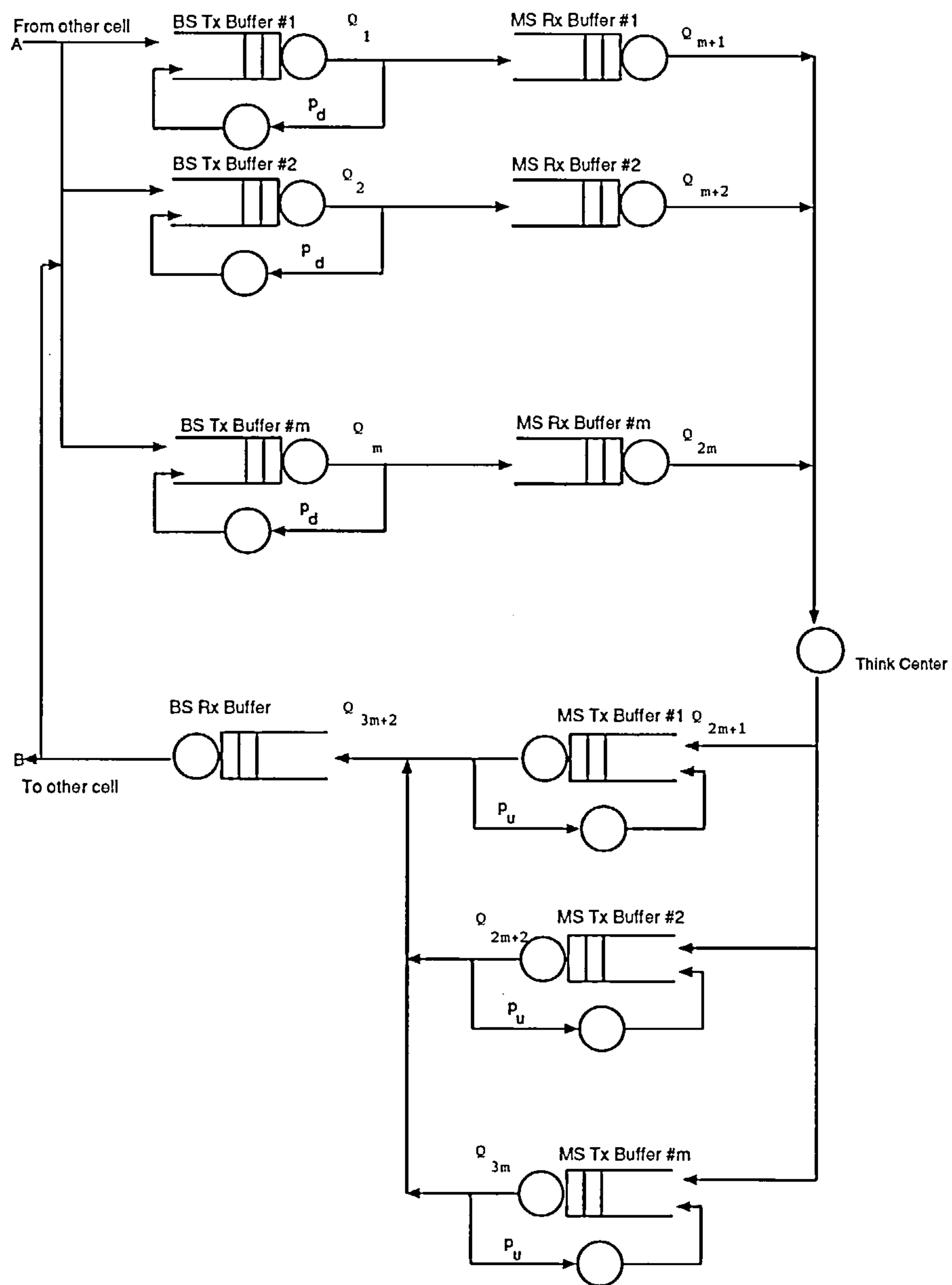


Figure 5.2: Queuing Network Model.

- The set of node loads is:

$$R = \{\rho_1, \rho_2, \rho_3, \rho_4\} \quad (5.92)$$

- We define the set  $R_1 = R - \{\rho_1\}$ , i.e.:

$$R_1 = \{\rho_2, \rho_3, \rho_4\} \quad (5.93)$$

- With each element of  $R_1$  we associate the following ratios:

$$\begin{aligned}\alpha_1 &= \frac{\rho_2}{\rho_1} & \beta_1 &= \frac{\rho_2}{\rho_1 - \rho_2} \\ \alpha_2 &= \frac{\rho_3}{\rho_1} & \beta_2 &= \frac{\rho_3}{\rho_1 - \rho_3} \\ \alpha_3 &= \frac{\rho_4}{\rho_1} & \beta_3 &= \frac{\rho_4}{\rho_1 - \rho_4}\end{aligned}\tag{5.94}$$

- For each ratio  $\beta_j$  ( $j = 1, 2, 3$ ) we associate the sequences  $\{s_j(k)\}$ . Equation 5.89 show us that the loads  $\rho_2$  and  $\rho_3$  have multiplicity  $m$  and load  $\rho_4$  has multiplicity 1. Thus the sequences  $\{s_1(k)\}$  and  $\{s_2(k)\}$  are obtained by replacing  $m_j$  by  $m$  in equation 5.35. The elements of the sequence  $\{s_3(k)\}$  are simply  $s_3(k) = (-\beta_3)^k$ .
- The elements of the sequence  $\{h_1(k)\}$  can either be obtained by the recurrence formulas given by equations 5.43 and 5.44 or directly by the equation below:

$$h_1(k) = \frac{1}{[(m-1)!]^2} \sum_{k_1+k_2+k_3=k} \frac{(m+k_1-1)!(m+k_2-1)!}{k_1!k_2!} (-\beta_1)^{k_1} (-\beta_2)^{k_2} (-\beta_3)^{k_3}\tag{5.95}$$

- Using proposition 2 we obtain the coefficients  $A_{1j}$ :

$$A_{1j} = H_1^0 h_1(j-1) \quad j = 1, 2, \dots, m\tag{5.96}$$

where  $H_1^0$  is given by:

$$H_1^0 = \frac{1}{(1-\alpha_1)^m (1-\alpha_2)^m (1-\alpha_3)}\tag{5.97}$$

We can now apply the flow equivalent server method to the cell sub-network. According to this method we evaluate the throughput function of the closed short-circuited network and use it as the service rate of an equivalent queue in the complement network. The complement network is formed by the rest of the cellular

network and it is not considered in this chapter. Therefore we refer to  $e_c$  as being the visitation rate of the FES node in the FES-transformed network.

According to [HaPa 92] the throughput along the short-circuited arc of the FES sub-network with population  $k$  is given by:

$$T(k) = e_c \frac{g(k-1)}{g(k)} \quad (5.98)$$

where  $g(k)$  is the normalising constant of the closed sub-network given by equation 5.91.

#### Remarks

- Note that the total number of nodes of the cell is given by  $m_t = 5m + 2$ .
- The normalising constant can also be given by using the approximation we suggested in section 5.4.5.
- We discuss numerical results and their accuracy in chapter 6.

#### Evaluating performance parameters of the whole network

We have applied our method to evaluate the flow equivalent server of a cell. We have seen that this method reduces considerably the complexity of computing the cell throughput function, since the number of terms in the summations involved depends linearly on the number of nodes and on the population.

Clearly this method can also be used to evaluate the throughput function of the whole network. However when the number of nodes and the population is large our method becomes less efficient, since the summations also become large. In chapter 6 we will present an integral formula for the normalising constant which does not depend on the number of nodes nor the population. Further work is required to obtain the expansion coefficients of such formula.

## 5.6 Conclusion

We have presented a method to evaluate asymptotic properties of queuing networks and applied it to a study of cellular networks. This method utilises a formula for the normalising constant obtained from an inversion of its generating function using the convolution theorem.

The advantage of this method is that it is of greater generality compared with the existing ones and gives an exact closed formula in a great number of cases and a good approximation in others. In the next chapter we will apply our method numerically, and compare it with the existing ones in terms of both accuracy and efficiency of the algorithm.

# Chapter 6

## Network Design and Comparative analysis

### 6.1 Introduction

In previous chapters we have described some of the design problems in cellular networks and introduced three different analytical models for the performance evaluation of various aspects. We show in this chapter how these techniques can help the system designer to analyse and compare different solutions and to identify trouble spots in the network protocols. We then apply these techniques numerically and give an interpretation for the results obtained in order to give quantitative support for a design decision.

Section 6.2 discusses what is perhaps one of the most important aspects of multi-access systems, which is the *resource sharing* mechanisms (see [Klei 75]). We present three different buffer administration techniques for the base station, which differ by the way that the transmitters (i.e., the resources) are allocated to the base station packets (i.e., the competitive demands). The first two techniques are modelled using the approach described in chapter 3. The mathematical modelling of the third technique is presented here and is based on a modification of the same technique. We

obtain the channel throughput, mean queue length and packet delay. We present numerical results and compare them in terms of system performance.

Section 6.3 addresses the issue of choosing the number of radio channels to be used in each cell. This number is related to the traffic requirements where the base station is located. A small number of channels operating in a cell may result in congestion at the base station or unacceptable performance in terms of packet delay. On the other hand a large number of channels result in an efficient use of the spectrum. We consider the multi-queue system described in chapter 4 which comprises  $N$  queues (representing the forwarding buffers to the mobile stations) and  $K$  transmitters (associated with the radio channels). We apply the same methodology to evaluate the effective service time distribution considering that the queues are served in cyclic order by a channel controller (their server). From this distribution, we obtain a work  $\times$  traffic curve which relates the utilisation factor  $\rho_0$  with the arrival rate  $\lambda$ . By using this curve we can define a *work region* which reflects the load conditions the base station operates. The maximum number of channels and stations supported can be determined from analysis of this curve.

Section 6.4 discusses the problem of avoiding congestion at the base station. Such congestion is undesirable since its performance affects all other stations inside the cell. The object of the congestion control is to limit the network throughput when the number of packets (or nodes) in the network becomes high. We then use the approach of chapter 5 to obtain the cell throughput function (which is a ratio of its normalising constant at different populations). We show how this throughput is related to the network population and number of nodes. We give asymptotic bounds when these parameters approach infinity. We also discuss qualitatively the behaviour of the network normalising constant by representing through a product of simpler node generating functions. We then apply the method described in chapter 5 numerically and discuss its accuracy. We conclude this section by giving a summary

of the results obtained.

## 6.2 Designing a buffer administration technique

This section addresses one of the design issues discussed in chapter 1, which is the design of a buffer administration technique for the base station and its associated channel assignment algorithm. We have seen that a buffer administration technique is a set of norms which tells how the incoming packets are organised inside the base station and how they are forwarded to their final destinations (which are the mobile stations inside the cell or other cells via wired links). The channel assignment algorithm is responsible for assigning the stored packets to the transmitters. The transmitters are associated with the different communication channels, i.e. we assume there is one transmitter for each radio frequency. We assume that the mobile stations have only one receiver which is tuned into one of these communication channels. Consequently, the base station can not transmit packets to the same destination on more than one channel at the same time.

The three techniques described here prevent address repetition by separating the incoming packets in different queues and assigning their packets to different transmitters. However, they differ by the way that these queues are organised and the way their customers are assigned to the transmitters. They are referred to as *Cyclic Polling*, *Channel split* and *Reservation* buffer administration techniques and described as follows.

### Cyclic polling technique

In this case, we assume that there are  $N$  parallel queues of capacity  $C$  and there are  $K$  transmitters. There is one queue for each destination address. A *channel controller* operates at discrete time intervals. Its operation consists of visiting the queues in cyclic order, and allocating its packets (customers) to the transmitters.

The allocation is described in chapter 3 and can be summarised as follows. If a visited queue is empty the channel controller immediately jumps to the next queue in sequence. If it is not empty its first packet (customer) is allocated to the first transmitter available. The channel controller stops visiting the queues either when all  $K$  transmitters have been filled or all  $N$  queues have been visited (in order to avoid overlapping). In both situations the channel controller stops pointing to the next queue to be visited in the next discrete time interval, when it re-starts its operation.

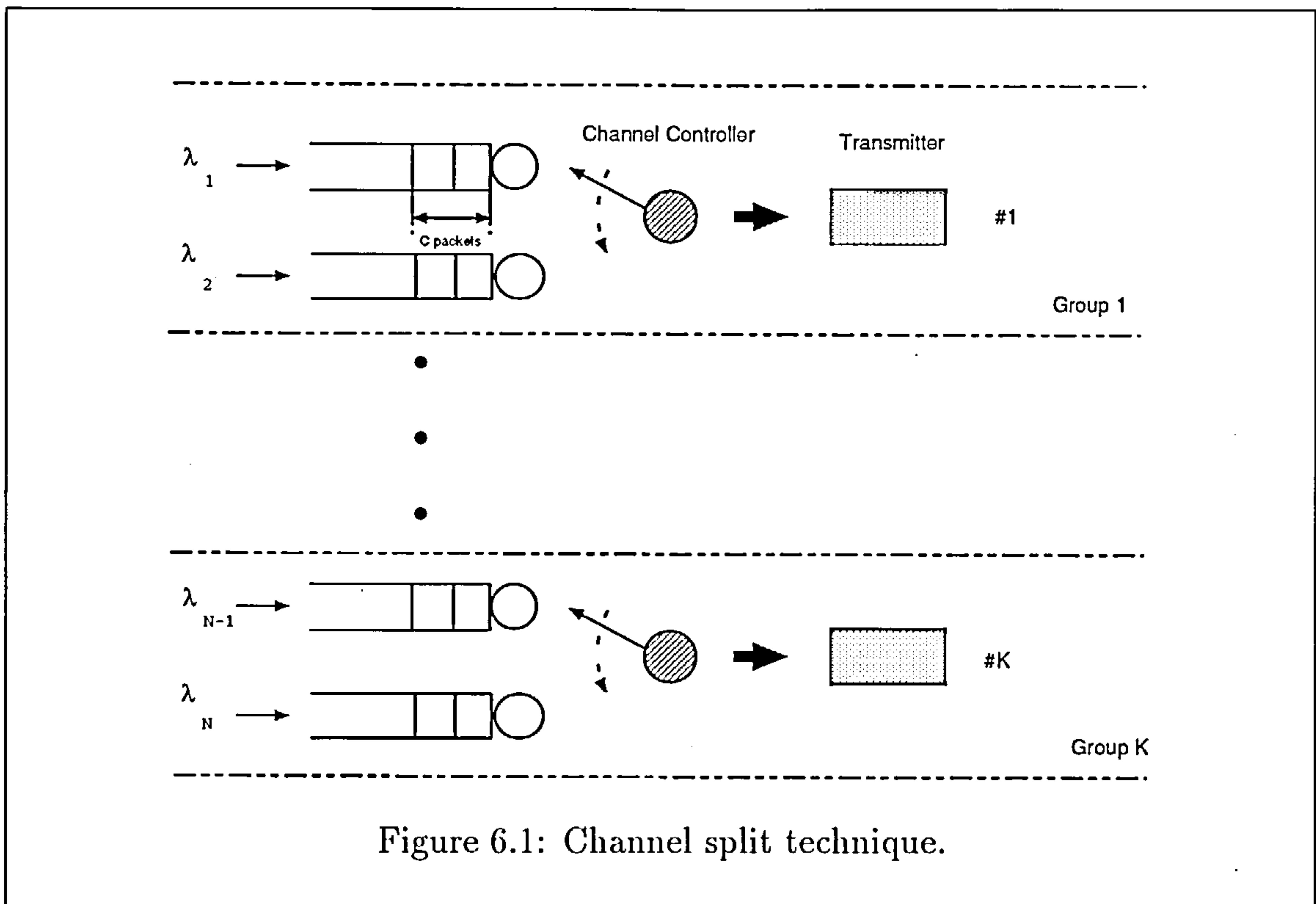
### Channel split technique

As in the cyclic polling technique, we assume that there are  $N$  parallel queues of capacity  $C$  and that there are  $K$  transmitters. However, in this case we assume that the queues are divided into  $K$  groups. The number of queues in each group can be arbitrary, but a given queue belongs to just one group. To each group we assign one transmitter. Fig. 6.1 shows an example where there are two queues in each group.

An independent channel controller operates in each group at discrete time intervals. There is only one transmitter available for each group. The channel controller operation is similar to the cyclic polling technique, however it visits only queues of its own group (i.e. it searches for the first queue with a packet to transmit).

### Reservation technique

As in the channel split technique, we assume that the  $N$  incoming packet streams are divided into  $K$  independent groups. However, in this case packets from the same group are merged into just one queue. The first packet of each queue is allocated to the transmitter at the beginning of each discrete time interval. Fig. 6.2 shows an example where there are two packet streams merged into one queue. The capacity of each queue is  $g_i C$ , where  $g_i$  is the number of packet streams merged in group

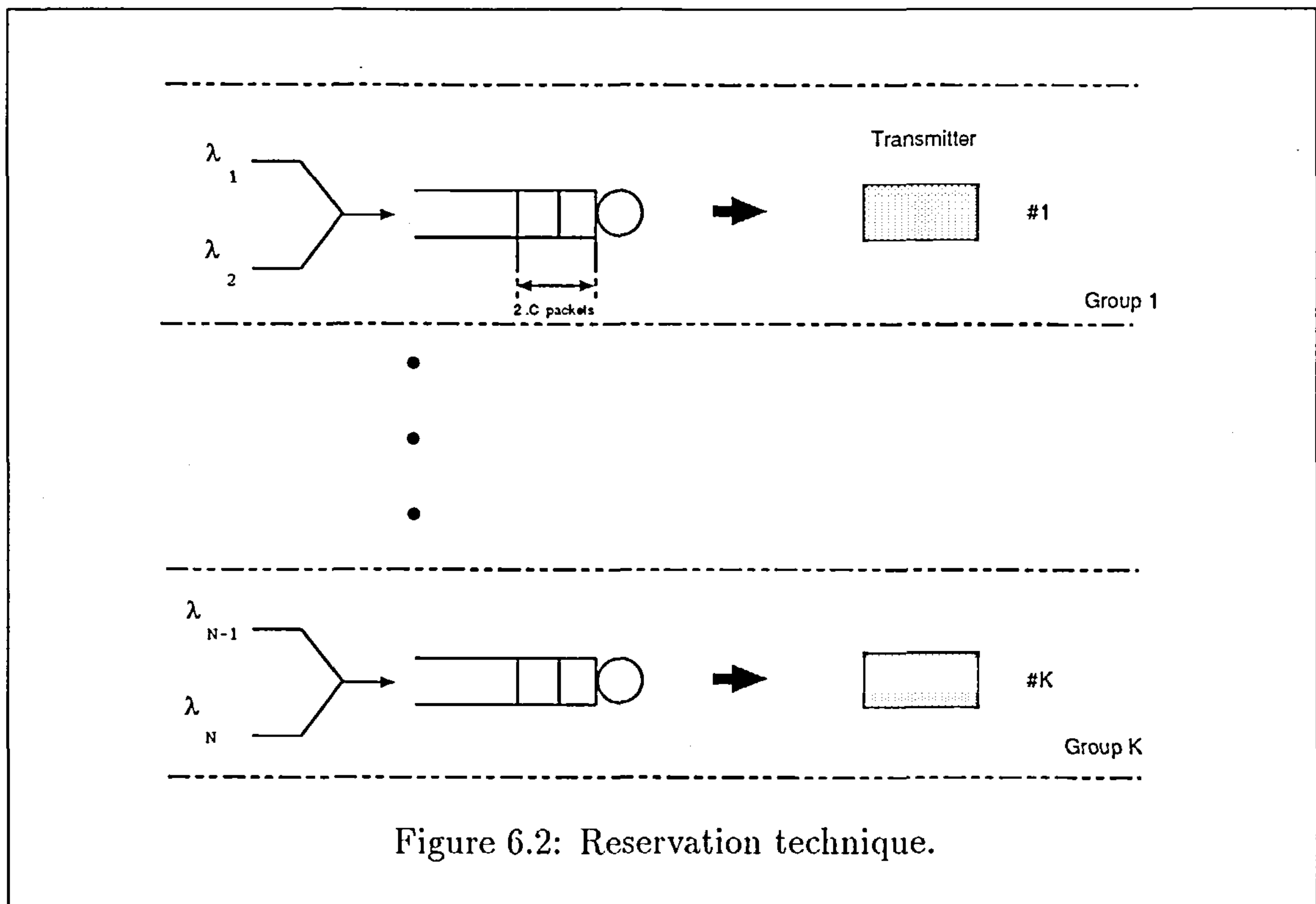


$i$  ( $1 \leq i \leq K$ ).

We observe that both reservation and channel split technique have the advantage of transmitting packets from a given queue in the same channel, which can not be guaranteed in the cyclic polling technique. This can be useful in situations in which the mobile stations can not change the receiving channel dynamically, as described in chapter 2.

### 6.2.1 Modelling the buffer administration techniques

The cyclic polling technique can be modelled using the approach presented in chapter 3. The system is represented by a Markov chain where a state corresponds to the number of packets in each queue and the position of the channel controller at the beginning of each discrete time interval. The channel throughput and the steady state queue length distribution is obtained from the solution of the balance equations.



The channel split technique can also be modelled using the same method considering each group as an independent multi-queue system where the number of queues is  $n_g$  and the number of channels is 1 ( $1 \leq g \leq K$ ).

The reservation technique is modelled as follows.

We consider group  $g$  ( $1 \leq g \leq K$ ) separately. Let  $\lambda_g$  be the total arrival rate of this group, i.e.,  $\lambda_g = \sum_{i=1}^{n_g} \lambda_i$ , where  $n_g$  is the number of Poisson streams merged. Let  $M$  be a random variable which describes the number of packets (customers) at queue  $g$  at each discrete time instant, and  $m$  be an integer which describes the value of  $M$  observed *prior* to the transmitter allocation. The transmitter is filled at each discrete time point in which queue  $g$  is not empty.

Let  $C^*$  be the capacity of queue  $g$  and  $a_i$  the probability of  $i$  arrivals between consecutive discrete time points of length  $\theta$ . Assuming that interarrival times are exponentially distributed, we have:

$$a_i = \frac{(\lambda_g \theta)^i}{i!} \exp(-\lambda_g \theta) \quad (6.1)$$

This system can be represented by a Markov chain with state space:

$$S = \{m | 0 \leq m \leq C^*\} \quad (6.2)$$

The  $(C^* + 1) \times (C^* + 1)$  transition probability matrix  $T$  is given by:

$$T = \begin{bmatrix} a_0 & a_1 & a_2 & \dots & 1 - \sum_{i=0}^{C^*-1} a_i \\ a_0 & a_1 & a_2 & \dots & 1 - \sum_{i=0}^{C^*-1} a_i \\ 0 & a_0 & a_1 & \dots & 1 - \sum_{i=0}^{C^*-2} a_i \\ & & & \vdots & \\ 0 & 0 & 0 & \dots & 1 - a_0 \end{bmatrix} \quad (6.3)$$

Let  $p_i$  be the steady state probability of having the Markov chain in state  $i \in S$ .

The balance equations and the normalising condition are then:

$$p_i = \begin{cases} p_0 a_i + \sum_{j=1}^{i+1} p_j a_{i-j+1} & i = 0, 1, \dots, C^* - 1 \\ 1 - \sum_{j=0}^{C^*-1} p_j & i = C^* \end{cases} \quad (6.4)$$

These  $C^* + 1$  (consistent) equations in  $C^* + 1$  unknowns can then be solved for all probabilities  $p_i$ . Let  $\pi_i$  be the probability of having  $i$  packets, observed immediately *after* a departure. This is given by:

$$\pi_i = \frac{1}{\sigma} p_{i+1} \quad i = 0, 1, \dots, C^* - 1 \quad (6.5)$$

where  $\sigma$  is the queue throughput given by  $\sigma = 1 - p_0$ . Let  $q_i$  be the probability of the queue having  $i$  packets, observed by a *random* observer. We then have (see [GrHa 85], [Coop 90]):

$$q_i = \begin{cases} 1 - \frac{\sigma}{\lambda_g} & i = C^* \\ \frac{\sigma}{\lambda_g} \pi_i & i = 0, 1, \dots, C^* - 1 \end{cases} \quad (6.6)$$

### 6.2.2 Comparative Analysis

We are interested in comparing the performance of the three techniques described above through a numerical example. We consider a base station which has a Poisson incoming traffic of rate  $\lambda_0$ . There are two transmitters available (i.e.,  $K = 2$ ). Transmissions take place at the beginning of a time-slot, which for simplicity, we assume to have length  $\theta = 1$  time unit (t.u.). The incoming traffic is divided into four streams of rates  $\alpha_i \lambda_0$  ( $1 \leq i \leq 4$ ) and  $\sum_i \alpha_i = 1$ . We consider two cases: the *balanced flow*, where  $\alpha_i = 0.25$ , and the *unbalanced flow* where  $\alpha_1 = 0.70$ ,  $\alpha_2 = 0.15$ ,  $\alpha_3 = 0.10$  and  $\alpha_4 = 0.05$ .

In the cyclic polling technique, the incoming packets are stored in four different queues of capacity  $C = 2$ . There is just one channel controller which visits these queues at the beginning of each time slot. It allocates packets to the two transmitters to avoid overlapping (as explained in 6.2).

In the channel split technique, the four streams are divided into two groups of two queues each. Group one contains the streams of arrival rate  $\alpha_1 \lambda_0$  and  $\alpha_2 \lambda_0$ . The four queues also have capacity  $C = 2$ , and they are served by two channel controllers (one for each group), which operates in a similar way as in the cyclic polling technique.

In the reservation technique, the same group division is adopted. However, the two streams of each group are merged, and its packets are placed in one queue. Thus, we have two queues of capacity  $C^* = 4$  and arrival rates  $(\alpha_1 + \alpha_2) \lambda_0$  and  $(\alpha_3 + \alpha_4) \lambda_0$  respectively. At each discrete time interval, the head of each queue (if there is one) is allocated to the transmitter.

#### Comparing the maximum arrival rate supported

We are interested in finding the maximum arrival rate  $\lambda_0$  supported in order to have the base station operating in a steady state (see chapter 3).

We observe that balanced systems support the same maximum arrival rate  $\lambda_0 = 2 \text{ packets/t.u.}$ . However, in unbalanced systems, the cyclic polling technique supports a higher arrival rate. In this case, the base station operates in steady state when the two conditions below are met:

$$\begin{aligned}\lambda_0\theta &< 2 \\ \alpha_1\lambda_0\theta &< 1\end{aligned}\tag{6.7}$$

In the example above, the cyclic polling technique supports a maximum total arrival rate of  $\lambda_0 = 1.428 \text{ packets/t.u.}$ . Both channel split and reservation techniques operate in steady state when  $\lambda_0\theta < 2$  and  $(\alpha_1 + \alpha_2)\lambda_0\theta < 1$ , which means a maximum total arrival rate of  $\lambda_0 = 1.176 \text{ packets/t.u.}$

### Comparing the performance at moderate incoming traffic

Using the methods described in chapter 3 and section 6.2, for each queue  $i$  ( $1 \leq i \leq 4$ ), we derived the mean queue length (denoted as  $\bar{q}_i$ ), the throughput (denoted as  $\sigma_i$ ), and the mean waiting time, which were obtained using Little's result (i.e.,  $\bar{w}_i = \frac{\bar{q}_i}{\sigma_i}$ ). Table 6.1 shows these results when the total arrival rate is  $\lambda_0 = 0.8 \text{ packets/t.u.}$

We observe that the cyclic polling technique performs better than the channel split technique in both balanced and unbalanced systems, since it has a higher throughput and a lower mean waiting time. In fact, the channel split technique does not perform well in situations where there are packets in both queues of the same group and no packets in other groups (for example 1 packet in queues 1 and 2, and no packets in queues 3 and 4). In this situation, the transmitters allocated to groups with empty queues are wasted.

We also observe that the reservation technique provides a higher throughput, compared to the channel split technique and cyclic polling. The queue throughput ( $\sigma_i$ ) is a quantity related to the packet loss due to queue overflow. As  $\sigma_i$  gets to  $\lambda_i\theta$ ,

|                  | <i>Balanced</i> |           |           | <i>Unbalanced</i> |           |           |
|------------------|-----------------|-----------|-----------|-------------------|-----------|-----------|
|                  | Cyc. Poll.      | Ch. Split | Reservat. | Cyc. Pol.         | Ch. Split | Reservat. |
| $\bar{q}_1$      | 0.1311          | 0.1549    | 0.3294    | 0.4353            | 0.5206    | 0.8697    |
| $\bar{q}_2$      | 0.1311          | 0.1549    | -         | 0.0692            | 0.0896    | -         |
| $\bar{q}_3$      | 0.1311          | 0.1549    | 0.3294    | 0.0444            | 0.0458    | 0.0682    |
| $\bar{q}_4$      | 0.1311          | 0.1549    | -         | 0.0213            | 0.0221    | -         |
| $\sum \bar{q}_i$ | 0.5243          | 0.6196    | 0.6588    | 0.5722            | 0.6781    | 0.9379    |
| $\sigma_1$       | 0.1981          | 0.1973    | 0.3994    | 0.5185            | 0.5026    | 0.6657    |
| $\sigma_2$       | 0.1981          | 0.1973    | -         | 0.1197            | 0.1194    | -         |
| $\sigma_3$       | 0.1981          | 0.1973    | 0.3994    | 0.0799            | 0.0799    | 0.1200    |
| $\sigma_4$       | 0.1981          | 0.1973    | -         | 0.0400            | 0.0400    | -         |
| $\sum \sigma_i$  | 0.7923          | 0.7891    | 0.7989    | 0.7581            | 0.7418    | 0.7857    |
| $\bar{w}_1$      | 0.6618          | 0.7851    | 0.8247    | 0.8395            | 1.0358    | 1.3064    |
| $\bar{w}_2$      | 0.6618          | 0.7851    | -         | 0.5781            | 0.7504    | -         |
| $\bar{w}_3$      | 0.6618          | 0.7851    | 0.8247    | 0.5557            | 0.5732    | 0.5683    |
| $\bar{w}_4$      | 0.6618          | 0.7851    | -         | 0.5325            | 0.5525    | -         |

Table 6.1: Performance measures for moderate incoming traffic ( $\lambda_0 = 0.8$ ).

so the number of packets lost is reduced.

The mean queue length and delay in the reservation technique is also higher. This is due to its higher capacity  $C^*$  and its higher throughput. A larger number of packets join the queue, hence its higher mean queue length and delay.

### Comparing the performance at high incoming traffic

Table 6.2 shows the mean queue length, the queue throughput and the steady state probability of queue 1 being empty (denoted as  $q_1(0)$ ), and full (denoted as  $q_i(max)$ ). These performance values were obtained considering a total arrival rate  $\lambda_0 = 1.9$  packets/t.u. (balanced systems) and  $\lambda_0 = 1.1$  packet/t.u. (unbalanced systems).

We observe that when the total incoming traffic is high, the system performance is impaired by packet loss due to queue overflow. This can be verified by comparing the total throughput ( $S = \sum_i \sigma_i$ ) with the average number of arrivals per time slot (1.9 for balance systems and 1.1 for unbalanced systems). This loss is particularly critical in unbalanced systems, where the probability of queue 1 being full is 0.2018

|                  | Balanced ( $\lambda_0 = 1.9$ ) |           |           | Unbalanced ( $\lambda_0 = 1.1$ ) |           |         |
|------------------|--------------------------------|-----------|-----------|----------------------------------|-----------|---------|
|                  | Cyc. Pol.                      | Ch. Split | Reservat. | Cyc. Pol.                        | Ch. Split | Reserv. |
| $\bar{q}_1$      | 0.5377                         | 0.6214    | 1.6826    | 0.6540                           | 0.7940    | 1.6349  |
| $\bar{q}_2$      | 0.5377                         | 0.6214    | -         | 0.1014                           | 0.1441    | -       |
| $\bar{q}_3$      | 0.5377                         | 0.6214    | 1.6826    | 0.0643                           | 0.0663    | 0.0988  |
| $\bar{q}_4$      | 0.5377                         | 0.6214    | -         | 0.0305                           | 0.0315    | -       |
| $\sum \bar{q}_i$ | 2.1508                         | 2.4856    | 3.3652    | 0.8503                           | 1.0359    | 1.7337  |
| $\sigma_1$       | 0.4227                         | 0.4094    | 0.8597    | 0.6600                           | 0.6146    | 0.8516  |
| $\sigma_2$       | 0.4227                         | 0.4094    | -         | 0.1640                           | 0.1629    | -       |
| $\sigma_3$       | 0.4227                         | 0.4094    | 0.8597    | 0.1097                           | 0.1097    | 0.1650  |
| $\sigma_4$       | 0.4227                         | 0.4094    | -         | 0.0550                           | 0.0550    | -       |
| $\sum \sigma_i$  | 1.6910                         | 1.6376    | 1.7198    | 0.9887                           | 0.9422    | 1.0166  |
| $q_1(0)$         | 0.5723                         | 0.5166    | 0.2341    | 0.4889                           | 0.4078    | 0.2456  |
| $q_1(max)$       | 0.1100                         | 0.1381    | 0.0950    | 0.1429                           | 0.2018    | 0.0892  |

Table 6.2: Performance measures for high incoming traffic.

(in the channel split technique).

We also observe that the cyclic polling technique still outperforms the channel split technique. However, the reservation technique provides a higher throughput (see fig. 6.3 and 6.4 for the throughput/offered traffic graphic).

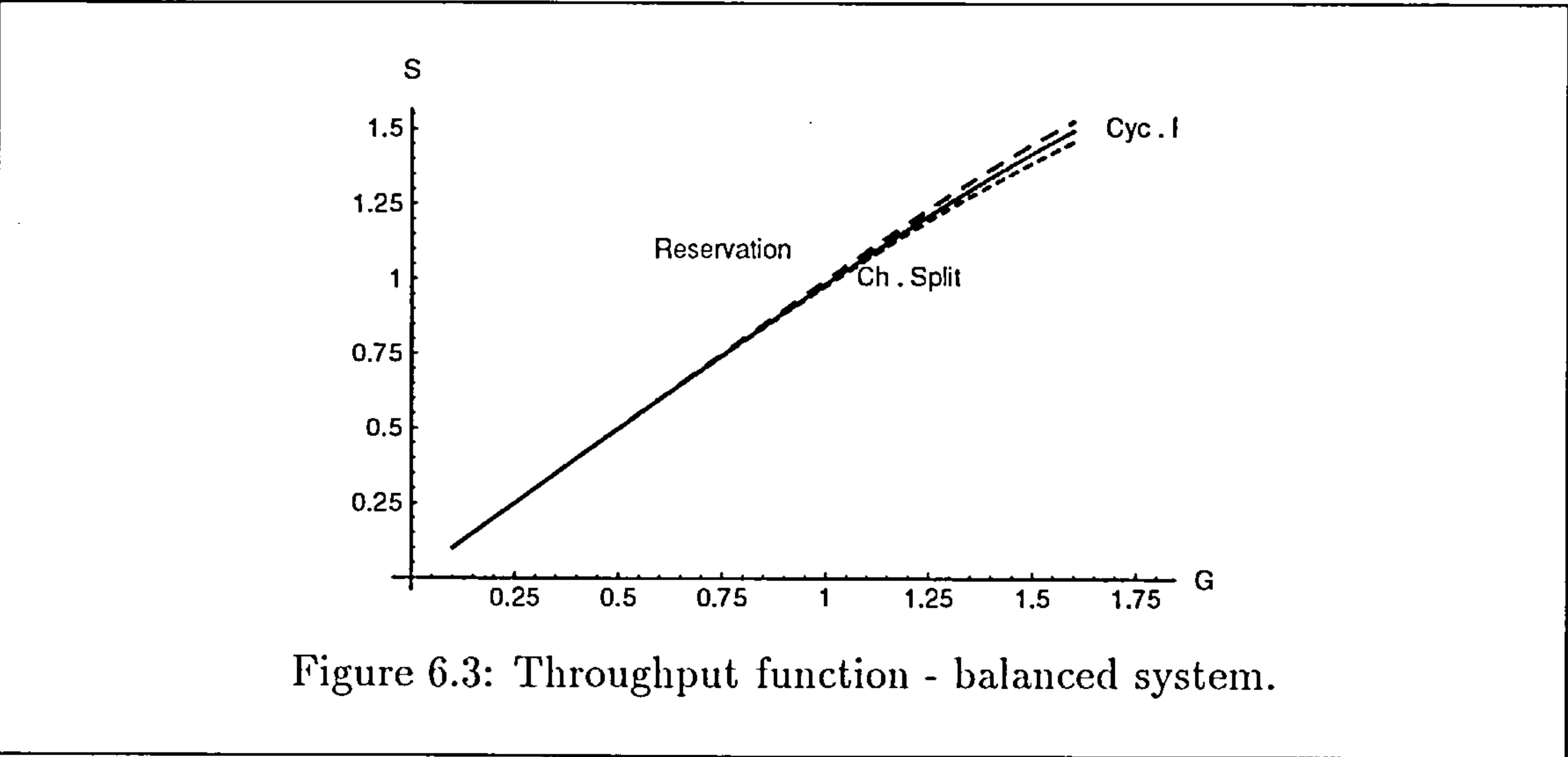


Figure 6.3: Throughput function - balanced system.

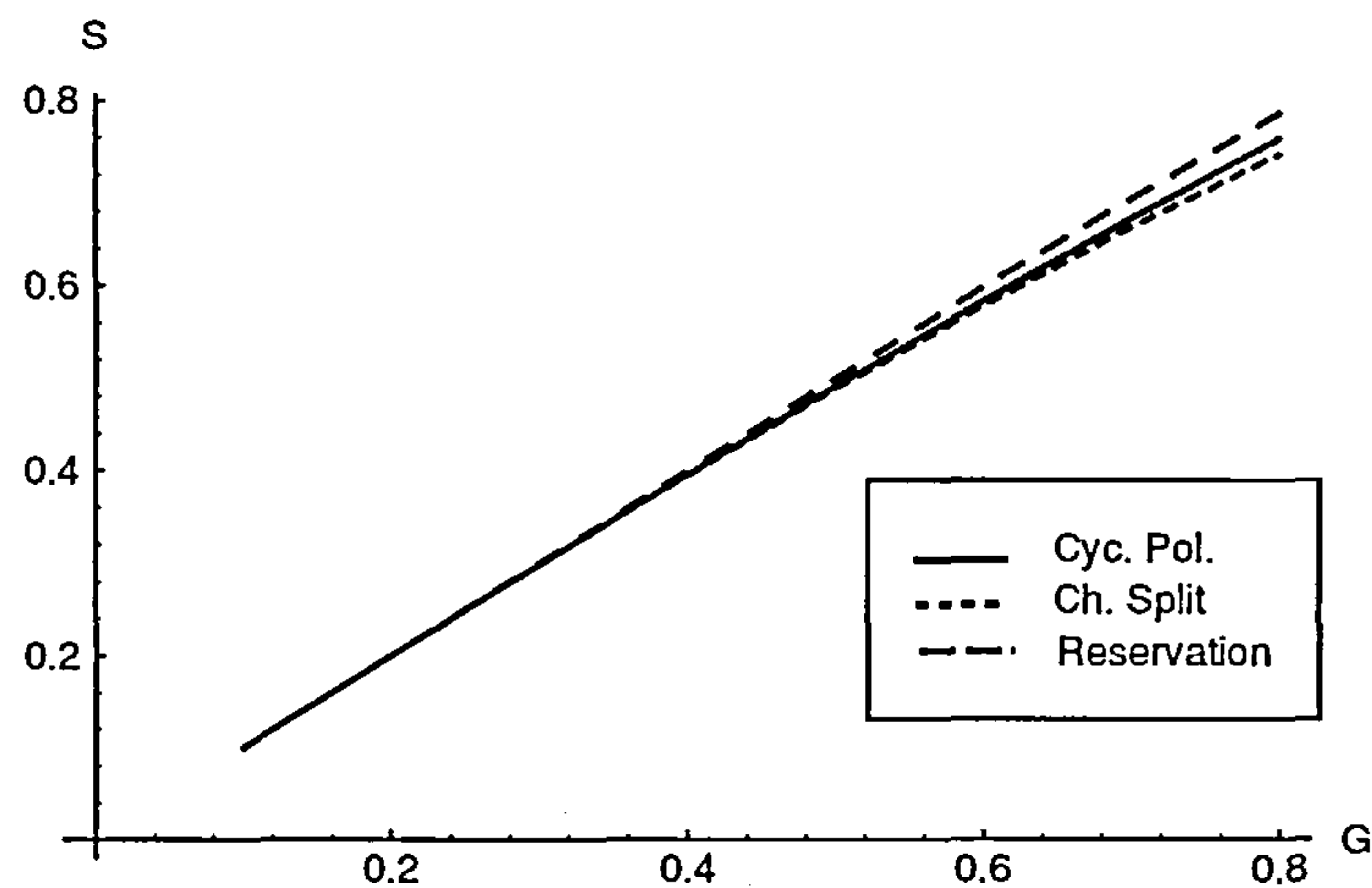


Figure 6.4: Throughput function - unbalanced system.

### 6.2.3 Conclusion

We have compared three different methods to store and forward packets in the base station. We have assumed that they have the same number of transmitters, traffic demands and total queue capacity. However, they differ by the way that queues are organised, and the way that the packets are assigned to the transmitters.

We evaluated their performance through an numerical example using the methods presented in section 6.2 and chapter 3. Results presented here suggest the following conclusions:

- Balanced systems can operate at higher loads, the cyclic polling technique, in particular, supports the highest total arrival rate.
- The cyclic polling technique outperforms the channel split technique in both

balanced and unbalanced systems. This is due to the fact that the cyclic polling technique uses the transmitters more efficiently.

- The reservation technique provides a highest throughput. This is due to the fact that it has the highest queue capacity.

Although the ranking of the strategies agrees with the expected, the value of quantitative modelling is to estimate “how much” one method is better than another, enabling cost/performance decisions to be made.

## 6.3 Choosing the numbers of channels and stations

One important design issue for cellular networks is the number of radio channels needed for each cell. This number is related to the traffic requirements in the particular region where the base station is located. Selecting a small number of channels may result in base station congestion or in unacceptable performance in terms of packet delay. Congestion may, to a limited extent, be avoided by controlling the mobile station polling rate (see section 6.4). On the other hand, a large number of channels results in an inefficient use of the spectrum, since the total channel capacity is under-utilised.

Therefore, the system designer must determine the appropriate number of channels by taking into consideration the *load region* in which the base station operates (load region will be defined later in this section). It is also important to predict the maximum number of mobile stations supported once the number of channels has been assigned. Obviously, the number of stations in a cell can change dynamically and we can only determine an upper bound.

In this section we apply the multi-queue system described in chapter 4. There we evaluated the time delay distribution of a packet (customer) arriving in a queue

(denoted to as the *reference queue*) which was served in cyclic order by a channel controller. Here, we use the same approach to obtain the base station work  $\times$  traffic curve. This represents the reference queue aggregate utilisation factor  $\rho_0$ , as a function of its arrival rate  $\lambda$ . This utilisation factor is the effective workload of the reference queue and can be interpreted as the utilisation factor seen by an arriving customer in isolation.

Initially, we find the reference queue service time distribution by observing the renewal process formed by the server visiting times. We have seen that its probability mass function can be regarded as being of phase-type. The work  $\times$  traffic curve is obtained by equating the mean service time ( $\rho_0/\lambda$ ) with the result obtained from the distribution. We use fixed-point methods to obtain the utilisation factor  $\rho_0$  as a function of the arrival rate  $\lambda$ .

By using the work  $\times$  traffic curve we can define a work region and then see how many channels are necessary to allow the the base station to operate within it. The maximum number of stations can also be calculated by the analysis of the same curve with different parameters.

### 6.3.1 The service time distribution

We consider a multi-queue system comprising  $K$  transmitters and  $N$  queues served in cyclic order by a channel controller as described in chapter 4. We concentrate our analysis on balanced systems, i. e. we assume that the arrival rate in all queues is  $\lambda$ . Let  $\sigma$  represent the steady-state probability of the reference queue not being empty at each discrete point  $\tau_n$  ( $n = 1, 2, \dots$ ).

We have seen that the random variable  $Y$ , which describes the number of intervals between consecutive visits, has a probability mass function of phase-type with representation  $(\pi', (T - R))$ . The elements  $t_{ij}$  of the  $N \times N$  stochastic matrix  $T$  represent the probability of the channel controller pointing at queue  $j$  at  $\tau_{n+1}$  given

that it was pointing to queue  $i$  at  $\tau_n$ , and the elements  $r_{ij}$  of the matrix  $R$  are given by  $r_{ij} = t_{ij}$  if  $j \leq i$  and  $r_{ij} = 0$  otherwise.

We proceed by evaluating the probabilities  $t_{ij}$  in balanced systems. Notice that only the elements  $t_{ij}$  ( $i > j$ ) are of interest, since they are part of the matrix  $(T - R)$ . In this case, a jump from queue  $i$  to  $j$  occurs whenever queue  $j$  is not empty and there are  $K - 1$  non-empty queues between  $i$  and  $j - 1$ . We assume that under moderate traffic the variable  $\sigma$  is independent. Clearly it is the same for all queues under balanced traffic. Then, the probabilities  $t_{ij}$  are given by:

$$t_{ij} = \begin{cases} 0 & \text{if } j < i + K \\ \binom{j-i-1}{K-1} \sigma^{K-1} (1-\sigma)^{j-i-K} & \text{otherwise} \end{cases} \quad (6.8)$$

Obviously, for each  $N$  and  $K$  there is one matrix  $(T - R)$ . We give an example, when  $N = 6$  and  $K = 2$ .

$$(T - R) = \begin{bmatrix} 0 & 0 & \sigma^2 & 2\sigma^2(1-\sigma) & 3\sigma^2(1-\sigma)^2 & 4\sigma^2(1-\sigma)^3 \\ 0 & 0 & 0 & \sigma^2 & 2\sigma^2(1-\sigma) & 3\sigma^2(1-\sigma)^2 \\ 0 & 0 & 0 & 0 & \sigma^2 & 2\sigma^2(1-\sigma) \\ 0 & 0 & 0 & 0 & 0 & \sigma^2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (6.9)$$

From this example we observe that the representation  $(\pi', (T - R))$  is *reducible* since rows 5 and 6 of  $(T - R)$  are identical. Let  $(\beta', S)$  be an irreducible representation of  $(\pi', (T - R))$ . Observe that  $(\beta', S)$  is not unique (a method of obtaining *one* irreducible representation is presented in Appendix B). Considering the example above, one possible irreducible representation is:

$$S = \begin{bmatrix} 0 & 3\sigma^4 - 2\sigma^5 & 6\sigma^4 - 15\sigma^3 + 10\sigma^2 - 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad \beta' = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} \quad (6.10)$$

Let  $\bar{m}$  be the mean inter-visit time. From the analysis presented in chapter 4 we have:

$$\bar{m} = \beta'(I - S)^{-1}\mathbf{e} \quad (6.11)$$

where  $I$  is the identity matrix, and  $\mathbf{e}$  is the unit vector. The Laplace transform of the probability mass function  $\{r_k\}$  of representation  $(\beta', S)$  is given by:

$$f(s) = e^{-s}\beta'(I - e^{-s}S)^{-1}\mathbf{S}^\circ \quad (6.12)$$

where the  $i$ -th element of  $\mathbf{S}^\circ$  is given by  $1 - \sum_j s_{ij}$  ( $s_{ij}$  are the elements of the matrix  $S$ ).

### The service time in the reference queue

We described the operation of the reference queue in chapter 4. We have seen that the service time of a given customer is either the time until the next visit of the channel controller (which we called residual time), or the time between consecutive visits (which we called inter-visit time). The former occurs when the customer finds the reference queue empty and the latter when it is not empty.

Let  $\hat{r}^*(s)$  and  $b(s)$  be the Laplace transform of the residual time and the service time distribution respectively. The following equation must hold:

$$b(s) = \pi_0 \hat{r}^*(s) + (1 - \pi_0)f(s) \quad (6.13)$$

where  $\pi_0$  is the steady state probability the reference queue is not empty upon arrival. Let  $\bar{x}$  be the mean service time. From the equation 6.11 and the mean residual time calculated in chapter 4 we obtain the following equation:

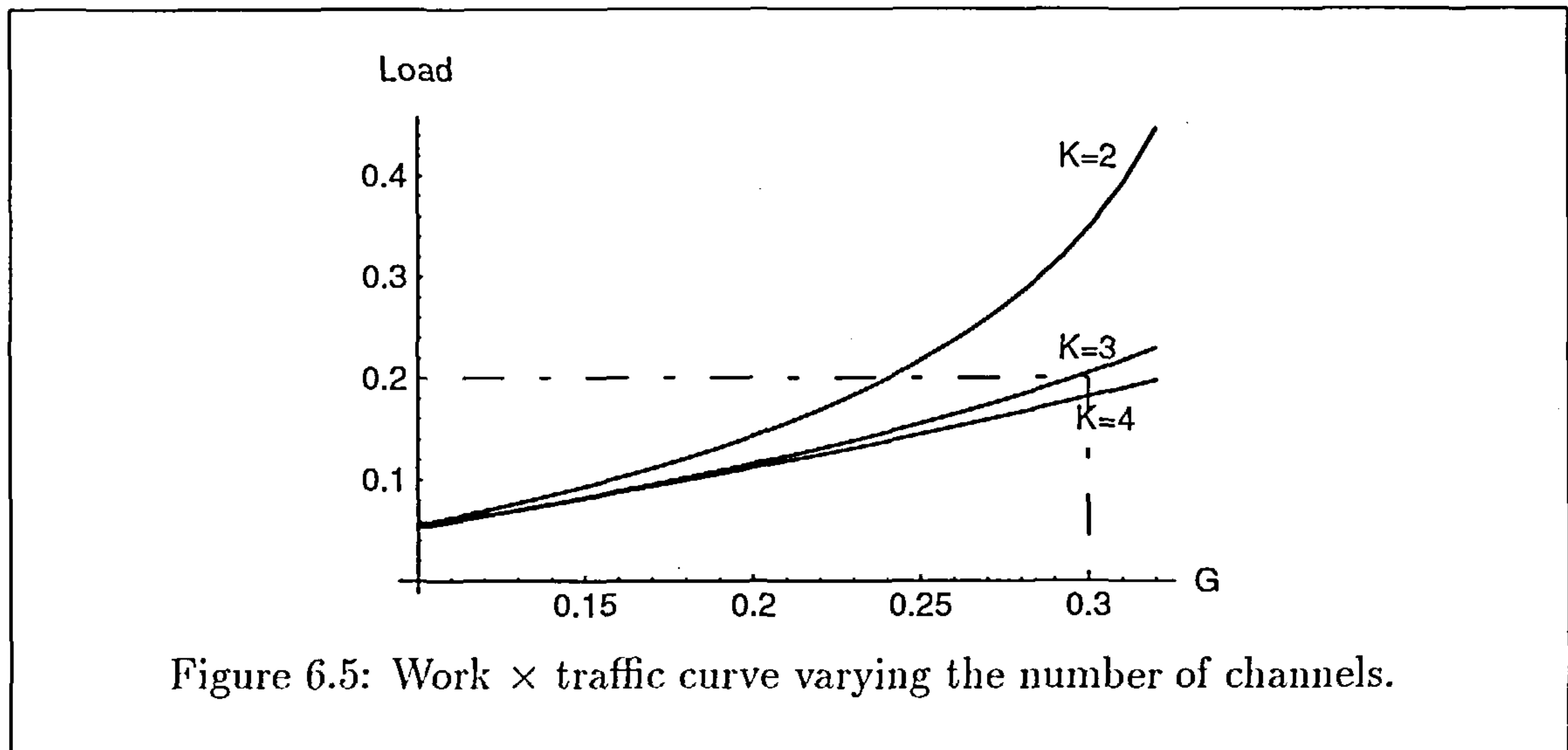
$$\bar{x} = \frac{\rho_0}{\lambda} = \pi_0[\theta'(I - S)^{-1}\mathbf{e} - \frac{1}{2}] + (1 - \pi_0)\beta'(I - S)^{-1}\mathbf{e} \quad (6.14)$$

where the vector  $\theta'$  is the stationary solution of the Markov chain with transition probability matrix given by  $S^* = S + S^\circ B^\circ$ . The matrix  $S^\circ$  contains identical columns  $\mathbf{S}^\circ$  and the matrix  $B^\circ = \text{diag}(\beta')$ .

### 6.3.2 Numerical Results

In order to obtain the parameter  $\sigma$  we solve the equation  $1 - \sigma = \pi_0 \hat{r}^*(\lambda)$  using fixed-point methods (see chapter 4). The load  $\rho_0$  immediately follows, by using equation 6.14. Then, we can calculate a work  $\times$  traffic curve which relates the traffic ( $G = \lambda$ ) with the load  $\rho_0$ . We plotted this curve for several values of  $N$  and  $K$ .

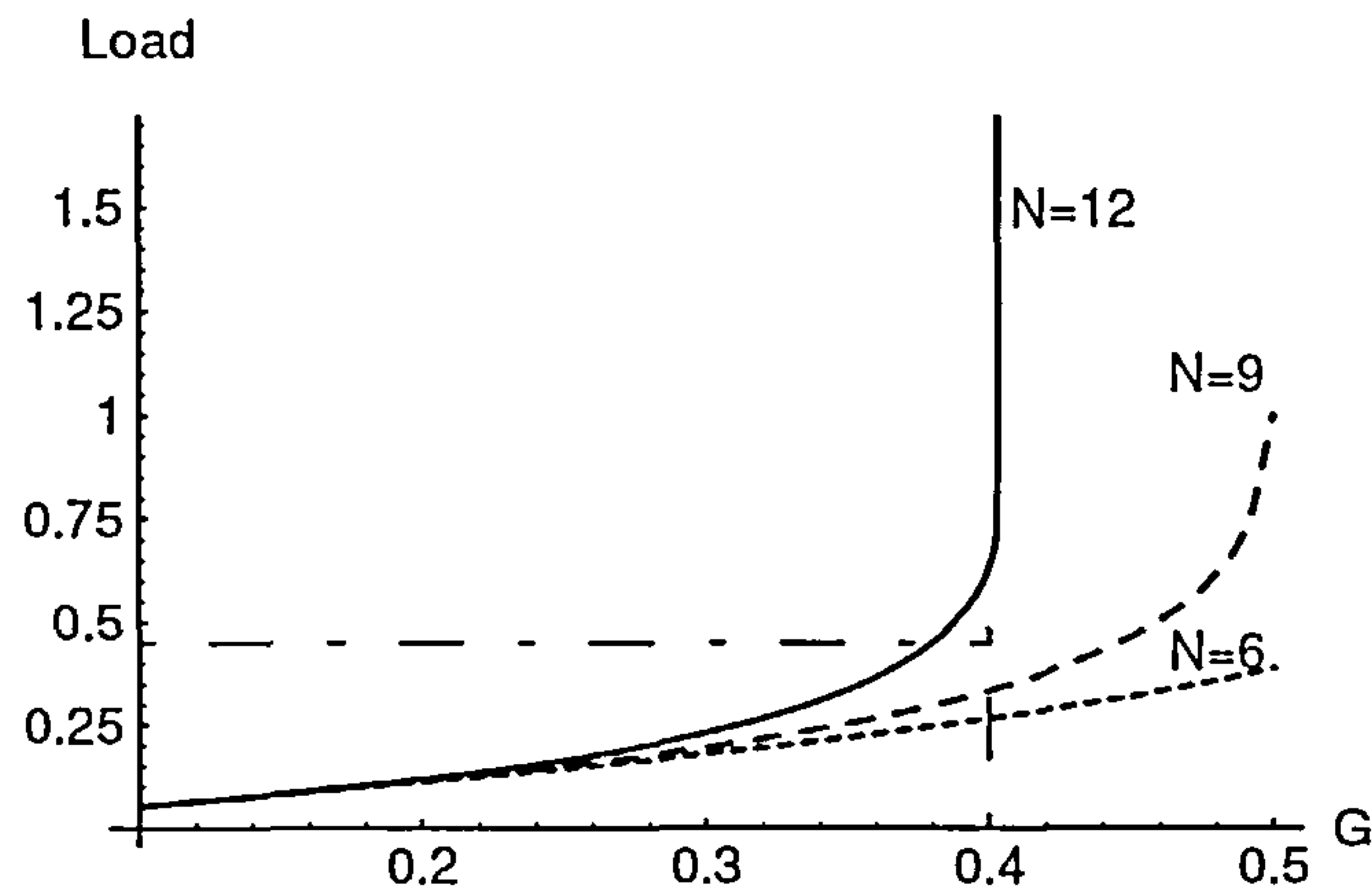
In fig. 6.5 we show three curves for  $N = 10$  and  $K = 2, 3$  and 4.



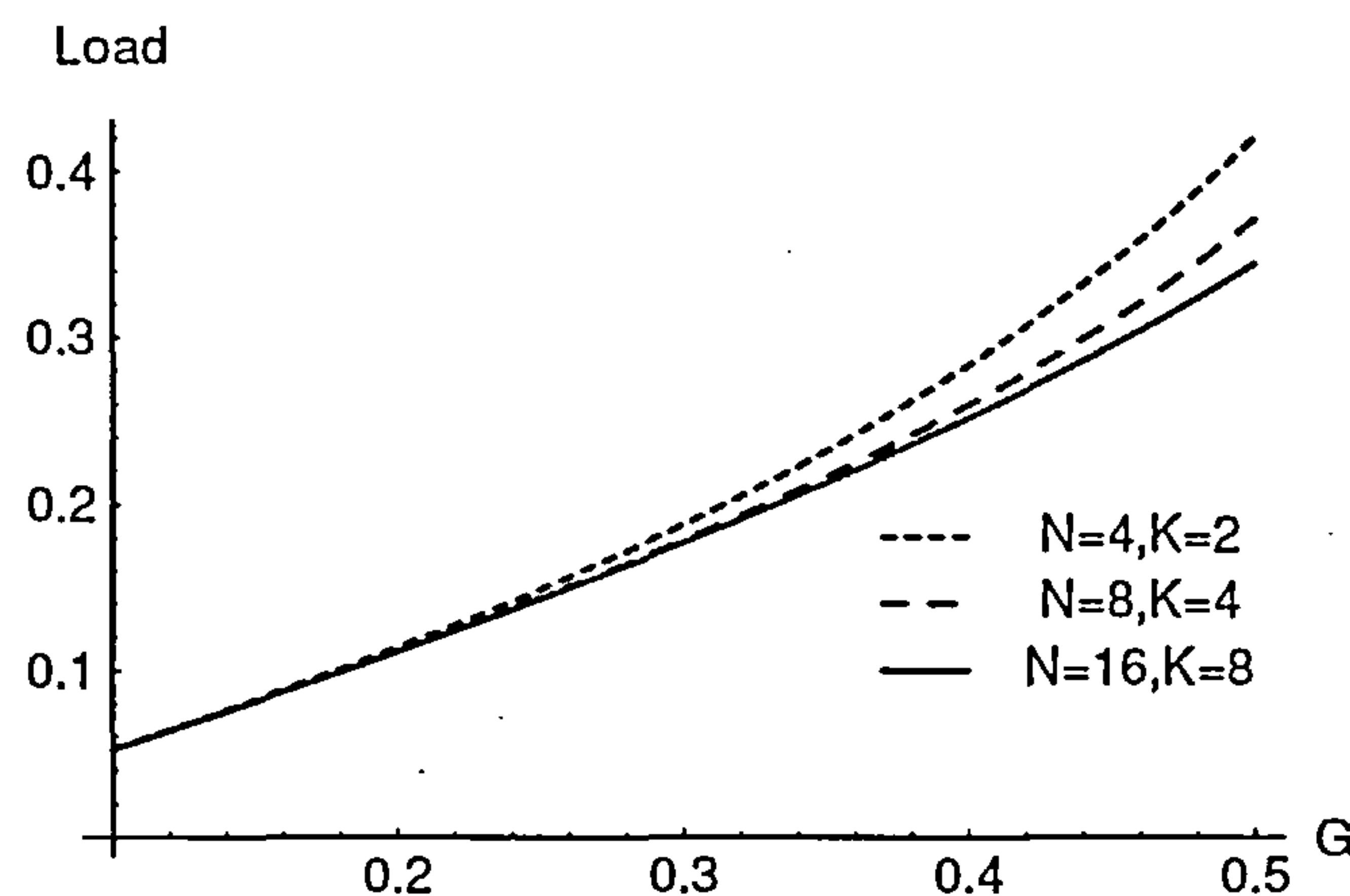
We define a *work region* as being the set of points  $(\lambda, \rho_0)$  such that  $\lambda < \lambda^*$  and  $\rho_0 < \rho_0^*$ , where  $\lambda^*$  and  $\rho_0^*$  are maximum pre-defined values.

This work region must be specified by the system designer. The choice of  $\lambda^*$  and  $\rho_0^*$  must be based on the traffic requirements of the area where the base station is located. In the example shown in fig. 6.5 we adopted  $\lambda^* = 0.3 \text{ packets/t.u.}$  and  $\rho_0^* = 0.2$  as we require a low mean service time. We can see that the most appropriate number of channels for this situation is  $k = 3$ , since the load curve for  $k = 2$  is out of the work region when  $\lambda > 0.25$ . We can also see that the choice of  $K = 4$  is not the best, since the total channel capacity would be under-utilised.

We give another example in fig. 6.6, where we have chosen the number of channels to be  $K = 3$ . We can see that for a work region in which  $\lambda_{max} = 0.4$  and  $\rho_{max} = 0.25$  the maximum number of stations supported is  $N = 9$ .

Figure 6.6: Work  $\times$  traffic curve varying the number of queues.

In fig. 6.7 we kept the ratio  $N/K$  constant and plotted the work  $\times$  traffic curve for the pairs  $(N = 4, K = 2)$ ,  $(N = 8, K = 4)$ , and  $(N = 16, K = 8)$ . We can see that at higher traffic, systems with larger numbers of channels and stations provide a better performance since their working load is smaller.

Figure 6.7: Work  $\times$  traffic curve with constant ratio.

### 6.3.3 Conclusion

We have presented and applied a methodology to evaluate the service time distribution of a multi-queue system served in cyclic order by a channel controller. By

equating the mean service time with the one obtained from the distribution we obtained a work  $\times$  traffic curve which reflects the load conditions at which the base station operates. From the analysis of this curve we defined a work region and obtained an adequate number of channels to be allocated for each cell. However, there should be a mechanism that ensures that base station operates in the work region defined. This topic will be discussed in the next section.

## 6.4 Congestion control at the base station

We have seen in previous chapters that the cell architecture is centralised. The base station is then responsible for local traffic control. The traffic in both downlink and uplink channels is regulated by a congestion control mechanism. The congestion control ensures that the cell throughput does not rise above a certain level. This is to avoid overloading. The cell throughput depends on the network population. Congestion in the base station is undesirable, since its performance affects all other stations in the cell.

In chapter 5 we showed that the cell can be represented by a closed queuing network model. We then calculated the generating function of its normalising constant and proposed a method to invert it. In this section we use this expression to study the cell throughput function, which is the ratio of the normalising constant at different populations.

We have seen that when the base station polls the mobile stations inside the cell it gives them permission to transmit packets. Therefore the base station can control the service time in their transmit buffers (and consequently the cell throughput) by choosing a suitable polling time. The polling and packet transmission times are represented in our model by the service time in the *mobile transmit buffer* queue.

In the first part of this section we show how the cell throughput function is related to the network population and number of nodes. We give asymptotic bounds when

they approach infinity. We also explain the behaviour of the normalising constant by representing it as a convolution of simpler node generating functions. The second part is dedicated to the numerical analysis of some examples. We compare our method with the existing ones, show its limitations and make suggestions for future study. We conclude this section by giving a summary of the results obtained.

### 6.4.1 Asymptotic bounds for the cell throughput

In chapter 5 we described a queuing network model for cellular networks. We consider that there are  $m$  mobiles inside a cell, and that indexes 1,2,3 and 4 denote the following queues: base station transmit queues, mobile reception queues, mobile transmit queues and base station reception queues. We then showed that the generating function of the normalising constant is given by:

$$G_m(z) = \prod_{i=0}^4 R_i(z) \quad (6.15)$$

where the functions  $R_i(z)$  ( $i = 0, \dots, 4$ ) are given by:

$$\begin{aligned} R_0(z) &= \exp(z) \\ R_i(z) &= \frac{1}{(1 - \rho_i z)^m}, \quad i = 1, 2, 3 \\ R_4(z) &= \frac{1}{(1 - \rho_4 z)} \end{aligned} \quad (6.16)$$

Let  $r_i(k)$  be the elements of the sequence formed by anti-transforming the functions  $R_i(z)$ . We have seen that:

$$\begin{aligned} r_0(k) &= \frac{1}{k!} \\ r_i(k) &= \frac{(k + m - 1)!}{k!(m - 1)!} \rho_i^k, \quad i = 1, 2, 3 \\ r_4(k) &= \rho_4^k \end{aligned} \quad (6.17)$$

We now develop our analysis by assuming that  $\rho_i < 1$ . We observe that this does not necessarily imply a loss of generality, since we could make a change of variables

in equation 6.15 so as to obtain  $\rho_i < 1$  (we would then have  $\rho_0 \neq 1$ , however this does not imply in a radical change in our method only that  $R_0(z)$  would become  $\exp(\rho_0 z)$ ). The problem of choosing adequate values for the parameters  $\rho_i$  is discussed in more detail in [King 90].

We now observe the following property:

**Property 1** *The normalising constant is the convolution of five sequences only (i.e.  $r_0(k), \dots, r_4(k)$ ) no matter how many mobile stations there are in the cell (this number is represented by the variable  $m$ ).*

This property is particularly useful for obtaining an exact value for the normalising constant with lower populations. In this case, the following simple recurrence formula can be used:

$$g_i(k) = \sum_{j=0}^k g_{i-1}(j) r_i(k-j) \quad i = 1, 2, 3, 4. \quad (6.18)$$

Initialising  $g_0(k) = \frac{1}{k!}$ , the exact value of  $g_i(k)$  can be obtained within a few operations. However, this formula is not appropriate for medium-sized populations, since the number of operations can increase dramatically as the population grows to a moderate size.

Consider now the behaviour of the normalising constant  $g_m(k)$  as the population  $k$  grows to infinity. We start by looking at each function  $r_i(k)$  individually. Fig. 6.8 shows a typical example of these sequences.

In fig. 6.8, curve number 1 represents  $r_i(k) = \frac{(k+m-1)!}{(m-1)!k!} \rho_i^k$ , considering  $m = 11$  and  $\rho_i = 0.5$  (for clarity, we show it as a continuous function). We observe that  $r_i(k)$  increases and then decreases. Let us call the real number  $\gamma_i$  at which this sequence stops increasing the *mode* of  $r_i(k)$  (i.e.  $r_i(k)$  increases for every  $k < \gamma_i$ ). This number can be calculated by solving the inequality of  $r_i(k+1)/r_i(k) > 1$ . Thus we have:

$$\gamma_i = \frac{\rho_i}{1 - \rho_i} (m - 1) \quad i = 1, 2, 3. \quad (6.19)$$

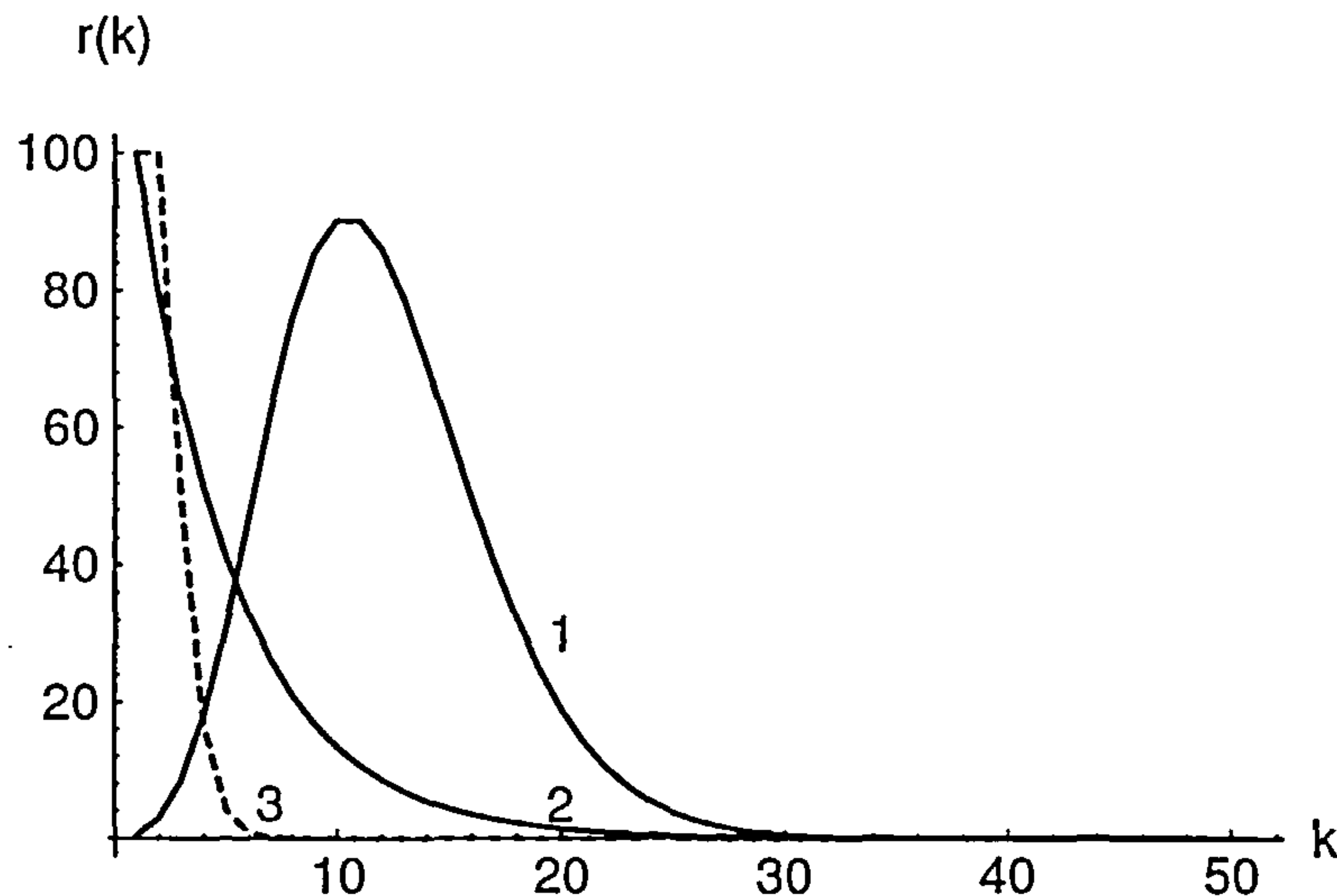


Figure 6.8: Node sequences  $r_i(k)$  (NB: curves 2 and 3 are not to scale).

Curves number 2 and 3 in fig. 6.8 represent the sequences  $r_4(k) = \rho_4^k$  and  $r_0(k) = \frac{1}{k!}$  respectively. We can see that these sequences always decrease as  $k$  increases, therefore we can say that their modes  $\gamma_4$  and  $\gamma_0$  are equal to 0.

Having calculated the mode of all  $r_i(k)$  sequences, the following property describes the behaviour of  $g_m(k)$ :

**Property 2** *The generating function  $g_m(k)$  increases and then decreases as  $k$  grows to infinity with mode  $\gamma_g$ .*

One problem which is still unsolved is how to derive an exact expression for the mode of  $g_m(k)$ . However, we can find a heuristic solution based on the fact that the product of the sequences  $r_i(n_i)$  is larger when  $n_i$  is an integer close to  $\gamma_i$  ( $i = 0, \dots, 4$ ). We then give the following approximate solution:

$$\gamma_g \sim \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4^* \quad (6.20)$$

where  $\gamma_4^* = \frac{\rho_4}{1-\rho_4}$ . The expression for  $\gamma_4$  is empirical, and was based on the observation that  $\rho_4$  shifts  $\gamma_g$  to the right. The sequence  $r_4(k)$  decreases strictly as  $k$

increases, therefore for small values  $\rho_4$  the convolution  $\sum_{n_1+\dots+n_4=k} \prod r_i(k)$  is maximum when  $n_4 = 0$ . However, the series  $\sum r_4(n_4)$  diverges as  $\rho_4$  approaches 1 and consequently the mode  $\gamma_g$  goes to infinity.

By calculating the normalising constant, we can evaluate the probability of each individual state of the network. However, we are interested in performance values which are expressed in terms of the cell throughput function ( $T_m(k)$ ) defined by (if we take the base station's visitation rate equal to one):

$$T_m(k) = \frac{g_m(k-1)}{g_m(k)} \quad k = 1, 2, \dots \quad (6.21)$$

Considering that the series  $\sum g_m(k)$  converges (to  $G_m(1)$ ), the following property applies to the throughput function:

**Property 3** *The cell throughput increases monotonically for  $k \geq 1$  and:*

- *Its first element  $T_m(1)$  is given by  $T_m(1) = (1 + m(\rho_1 + \rho_2 + \rho_3) + \rho_4)^{-1} < 1$ .*
- *The elements  $T_m(k)$  are such that  $T_m(k) < 1$  if  $k < \gamma_g$  and  $T_m(k) \geq 1$  if  $k \geq \gamma_g$  (since  $g_m(k)$  increases for  $k < \gamma_g$  and decreases for  $k > \gamma_g$ ).*
- *The element  $T_m(k)$  goes asymptotically to  $\rho_{max}^{-1}$  as  $k$  approaches infinity, where  $\rho_{max} = \max\{\rho_i\}$  ( $i = 1, \dots, 4$ ) (this can be verified by observing that the behaviour of  $T_m(k)$  is dictated by the network's bottleneck as  $k$  approaches infinity).*

In fig. 6.9 we plot the generating function  $g_m(k)$  considering  $\rho_1 = 0.5$ ,  $\rho_2 = 0.6$ ,  $\rho_3 = 0.7$  and  $\rho_4 = 0.4$ . We consider two cases:  $m = 4$  and  $m = 6$ . We can see that the effect of increasing the number of nodes is a shift to the right in the mode of  $g_m(k)$  (this can be easily verified by using property 2). We also observe that the maximum absolute value of  $g_m(k)$  also increases.

The cell throughput function is plotted in fig. 6.10 using the same network. We observe that the asymptotic value of  $T_m(k)$  is always determined by the maximum

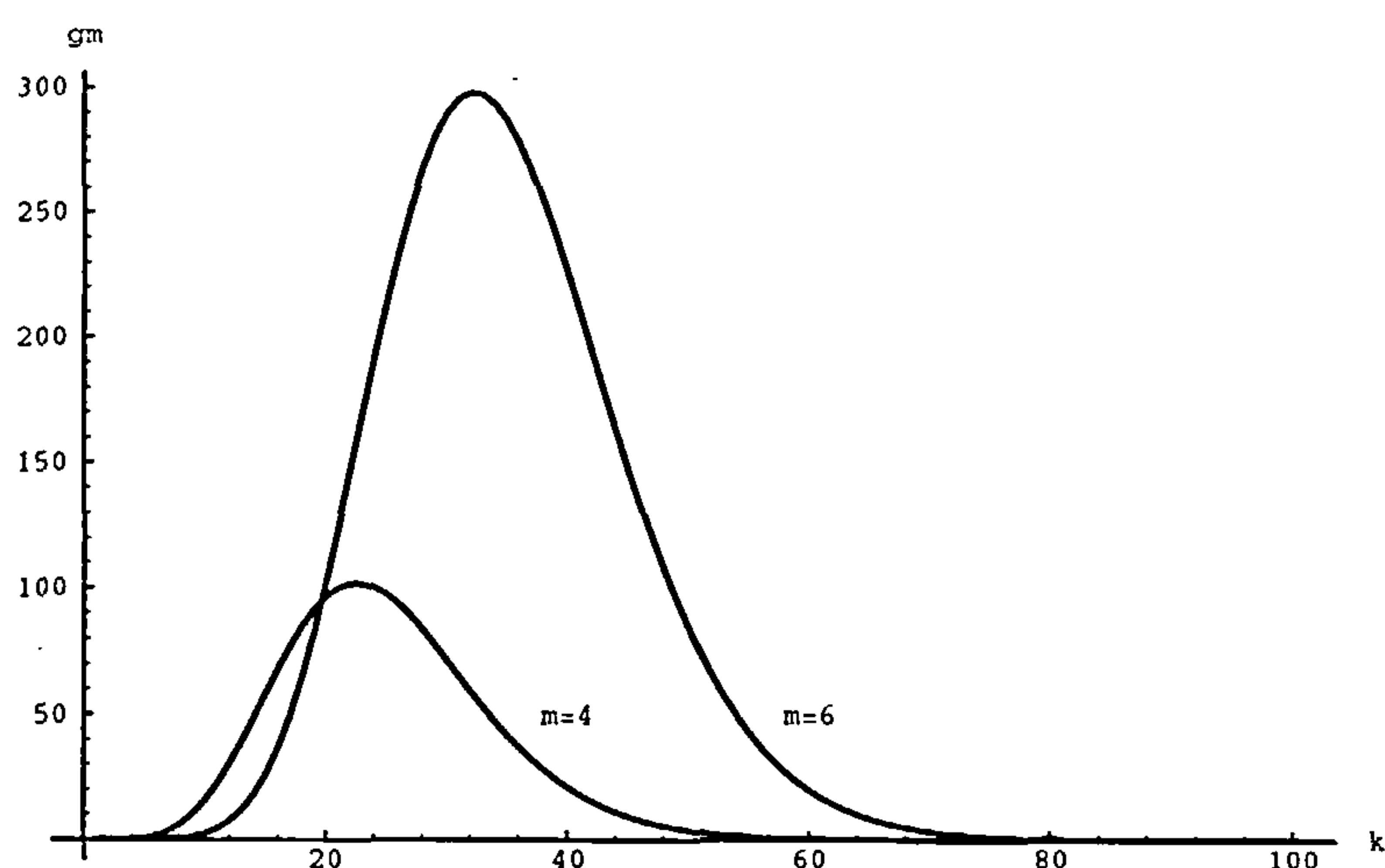


Figure 6.9: Generating function  $g_m(k)$  (NB: higher peak is not to scale).

parameter  $\rho$  (which is  $\rho_3$  in this example). However, we observe that the effect of increasing the number of stations is a reduction in the cell throughput at the same population. This is a consequence of the *shift to the right* of the normalising constant's mode  $\gamma_g$ .

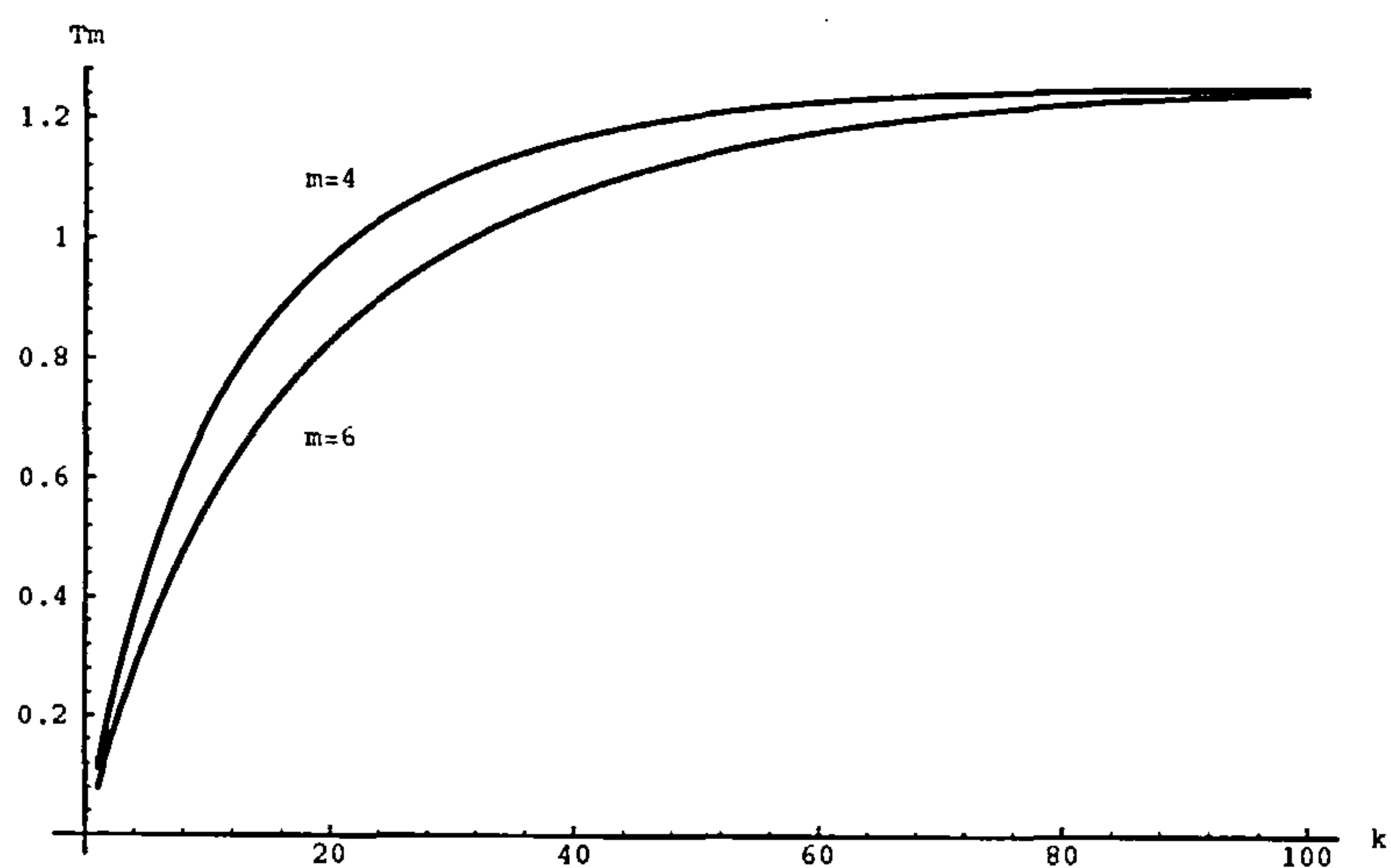


Figure 6.10: Cell throughput function.

We have seen in chapter 5 that the parameter  $\rho_3$  is related to the mobile transmit queues. The packets in these queues are transmitted on a FCFS basis through

one of the uplink channels. We have seen in chapter 2 that a packet can only be transmitted if the mobile station has been selected (polled) by the base station. When the mobile station is selected one packet will be transmitted during the next time-slot. Therefore, the service time in the mobile station transmit queues can be set up so that the parameter  $\rho_3 > \rho_i$  ( $i = 1, 2, 4$ ) (this is done by controlling the mobile station polling time). Another important consequence of property 2 is that the cell throughput is bounded by  $\rho_3^{-1}$  as the population approaches infinity.

Fig. 6.11 shows the effect of changing the parameter  $\rho_3$ , considering a fixed number of stations ( $m = 10$  in this example) and  $\rho_1 = 0.05$ ,  $\rho_2 = 0.06$  and  $\rho_3 = 0.07$ . We can see that as  $\rho_3$  increases the throughput decreases. We can also observe the *shift to the right* in the normalising constant's mode by looking at the point where  $T_m(k) = 1$ . When the population is large the throughput becomes saturated. This corresponds to the situation in which the mobile transmit buffers form a *bottleneck* in the network. When the population approaches infinity their node utilisation is given by their service rate (this agrees with what we observed in property 3).

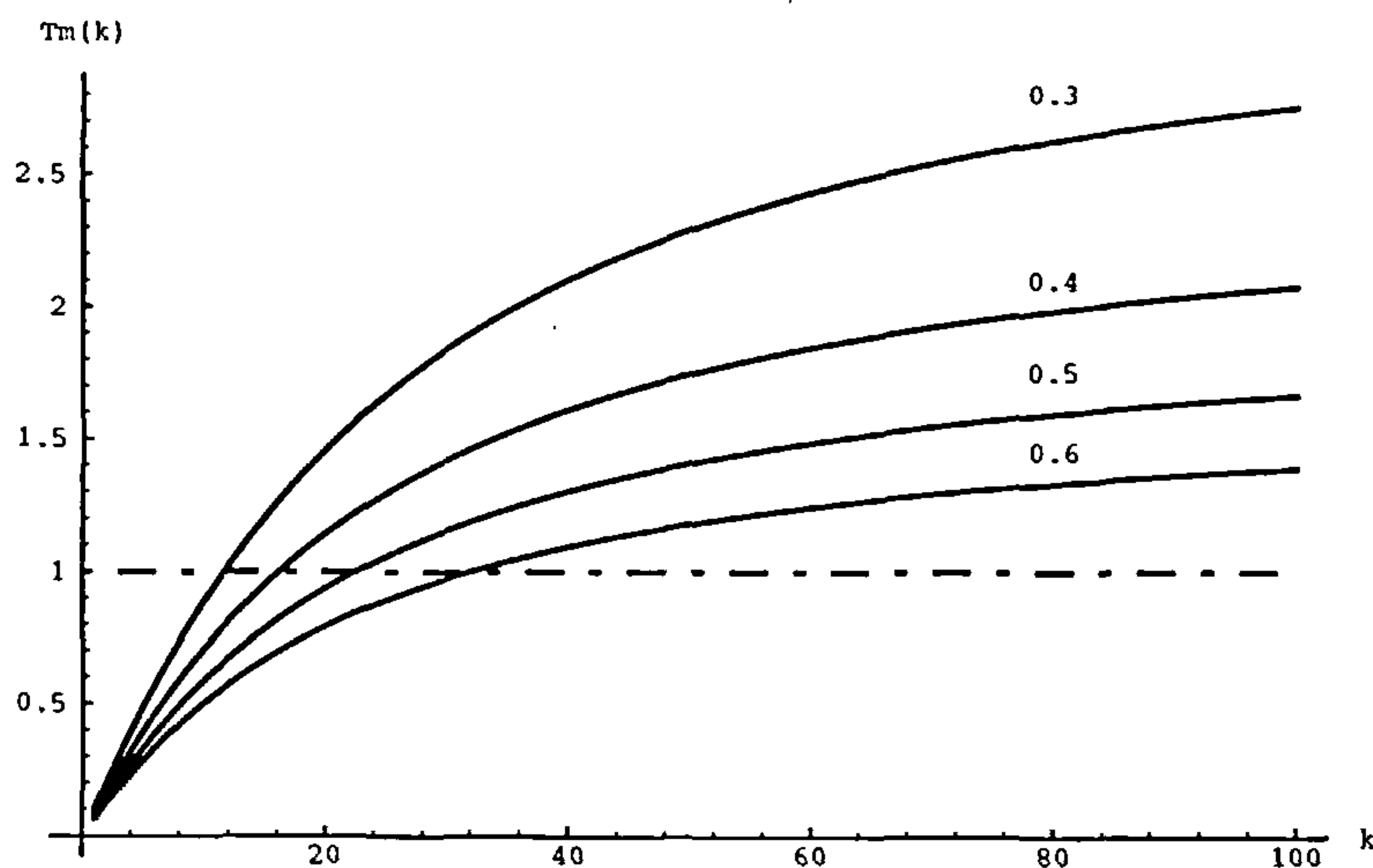


Figure 6.11: Cell throughput function.

### 6.4.2 Numerical Results

Using the method proposed in chapter 5, the exact formula for the normalising constant is given by:

$$g_m(k) = \sum_{i=1}^3 \sum_{j=1}^m A_{ij} \sum_{l=0}^k \frac{(l+m-j)!}{(m-j)!l!} \frac{\rho_i^l}{(k-l)!} + A_{41} \sum_{l=0}^k \frac{\rho_4^l}{(k-l)!} \quad (6.22)$$

We observe that our method produces a very efficient algorithm for numerical evaluation since it involves a summation of  $(3m+1)(k+1)$  terms.

#### The integral representation of the normalising constant

We have seen in chapter 5 that the normalising constant can also be evaluated using standard methods of numerical integration which is particularly effective at medium/large populations. We obtained the following equation (considering  $k > \max\{\rho_i^{-1}\}$ ):

$$g_m(k) = \sum_{i=1}^3 \sum_{j=1}^{m-1} A_{ij} \frac{1}{(m-j)!k!} \int_0^\infty \exp(-u) u^{m-j} (\rho_i u + 1)^k du + \sum_{i=1}^4 A_{i m_i} \rho_i^k \left( \exp(\rho_i^{-1}) - \int_k^\infty v_i^*(u) du + \frac{\rho_i^{-k}}{2k!} \right) \quad (6.23)$$

where  $m_i = m$  if  $i = 1, 2, 3$  and  $m_i = 1$  if  $m = 4$ . The function  $v_i^*(u)$  is obtained using Stirling's formula for the factorial. So we have:

$$v_i^*(u) = \frac{(u\rho_i)^{-u}}{\sqrt{2\pi u}} \exp(u) \quad (6.24)$$

Equation 6.23 is more convenient for networks with large populations since it contains the sum of only  $3m+1$  terms.

#### Networks with large number of nodes

Equations 6.22 and 6.23 are effective for networks with moderate number of nodes. However, as the number of nodes increases, the number of coefficients also increases and consequently the algorithm for numerical evaluation becomes less effective.

We can still handle a large number of nodes at low/moderate populations by using property 1 of section 6.4.1. In this way, we have:

$$g_m(k) = \sum_{\mathbf{n}=k} \prod_{i=0}^4 r_i(n_i) \quad (6.25)$$

where  $\mathbf{n} = \{n_i | i = 0, \dots, 4\}$ . Observe that the number of terms of this summation is greatly reduced, since this it is a convolution of five sequences only (the number of terms is given by  $\binom{k+4}{4}$ , and it is independent of  $m$ ). This equation handles well populations up to 200 and it is possible to obtain results for populations up to 1000.

For even larger populations we need to obtain the integral form of equation 6.25. This can be done by using Euler's integral (see chapter 5). We obtain:

$$g_m(k) = [(m-1)!]^{-3} k!^{-1} \int_{Q^+} \exp(-u_0) (u_1 u_2 u_3)^{m-1} \left( \sum_{i=1}^4 \rho_i u_i + 1 \right)^k d\mathbf{u} \quad (6.26)$$

where  $\mathbf{u} = (u_1, \dots, u_4)$ ,  $u_0 = \sum_{i=1}^4 u_i$  and  $Q^+ = \{\mathbf{u} | \mathbf{u} \geq 0\}$ .

Although equation 6.26 eliminates the summations over  $m$  and  $k$ , it is difficult to implement numerically, since it involves the complex task of dealing with a multiple integral. Further research is necessary to produce an asymptotic expansion of this equation.

### Comparison of this method with previous methods

We proceed by comparing the evaluation of our method with the existing methods. Firstly, as we have seen in chapter 5, the queuing network under examination comprises  $m_t$  queues, where  $m_t = m_p + m_q$ . The first term  $m_p = 3m + 1$  is the number of non-IS queues (these represent the base station, mobile station transmission and reception queues). The second term  $m_q = 2m + 1$  is the number of IS queues (these represent the delay due to error and the think center).

We now observe that:

- Using the original formula the evaluation of the normalising constant would require a summation of the following number of terms:

$$\binom{m_t + k - 1}{m_t - 1} \quad (6.27)$$

Clearly, evaluating the normalising constant in this way would lead to a prohibitive computational cost even for small networks.

- Using Buzen's Algorithm (see [BrBa 80]) the computation would involve a sum of at least  $m_t k$  terms (e.g.  $(5m + 2)k$  terms). Clearly, our method is more efficient, since it uses the fact that there are several nodes  $i$  with the same parameter  $\rho_i$ .
- The method proposed by J. McKenna and D. Mitra (see [McMi 82]) produces an integral representation and an asymptotic expansion expressed as the sum of the normalising constants of smaller *pseudonetworks*. By observing that these pseudonetworks contain all the non-IS nodes of the original network, we can say that the calculation of these coefficients is still complex in large networks. For example, let us consider a network with  $k = 1000$ ,  $m_p = 31$  and  $m_q = 20$ . In this case, using our method the calculation of the normalising constant requires the summation of  $1001 \times 31 = 31031$  terms. Using the above mentioned method the calculation of the normalising constant is carried out by evaluating the normalising constant of smaller networks (without the IS nodes and with smaller populations - typically 6). Therefore, each pseudonetwork requires a summation of  $\binom{m_p + 5}{m_p - 1} = 1.62 \cdot 10^6$  terms.

Furthermore, the assumption of *normal usage* (see Appendix C) restricts the evaluation of  $g_m(k)$ . This assumption applies both in absence of bottlenecks and in considering networks whose think times are much greater than processing times (i.e.  $\rho_0 \gg \rho_i$ ). Clearly, this assumption is not adequate for the

|     | $m = 2$ |        |       | $m = 10$ |        |        | $m = 20$ |        |       |
|-----|---------|--------|-------|----------|--------|--------|----------|--------|-------|
| $k$ | asympt  | exact  | error | asympt   | exact  | error  | asympt   | exact  | error |
| 20  | 19.67   | 19.67  | 0     | 18.75    | 18.75  | 0      | 17.71    | 17.71  | 0     |
| 100 | 97.83   | 97.83  | 0     | 92.33    | 92.33  | 0      | 85.99    | 86.444 | -0.5% |
| 200 | 190.84  | 192.44 | -0.8% | 138.83   | 178.49 | -28.5% | 129.94   | 164.93 | -27 % |
| 300 | -       | 249.67 | -     | -        | 242.84 | -      | -        | 227.52 | -     |

Table 6.3: Cell throughput using McKenna's method.

|     | $m = 2$ |        |       | $m = 10$ |        |       | $m = 20$ |        |       |
|-----|---------|--------|-------|----------|--------|-------|----------|--------|-------|
| $k$ | app.    | exact  | error | app.     | exact  | error | app.     | exact  | error |
| 20  | -       | 19.67  | -     | -        | 18.75  | -     | -        | 17.71  | -     |
| 100 | 97.83   | 97.83  | 0     | 87.37    | 92.33  | -5 %  | -        | 86.444 | -     |
| 200 | 192.44  | 192.44 | 0     | 178.26   | 178.49 | -0.1% | 148.78   | 164.93 | -10 % |
| 300 | 249.67  | 249.67 | 0     | 242.84   | 242.84 | 0     | 234.68   | 227.52 | 3%    |

Table 6.4: Cell throughput using our integral approximation.

study of congestion control in computer communication networks, since we need to evaluate the cell throughput at higher populations.

We also verified the accuracy of this method by evaluating the cell throughput of networks of various sizes and populations (see table 6.3). We adopted  $\rho_0 = 1, \rho_1 = 0.001, \rho_2 = 0.002, \rho_3 = 0.003$  and  $\rho_4 = 0.004$  and calculated the asymptotic expansions of  $g_m(k)$  by following the method proposed in [McMi 82]. In order to have a good accuracy we evaluated eight coefficients (although only five are suggested by the authors) and present these results in Appendix C. These coefficients are expressed as a ratio  $A_i/N^i$  where  $A_i$  depends on the normalising constant of the pseudonetworks and  $N$  is a large parameter designed to reflect the network size.

We can see that this method is very accurate for networks with small number of nodes and various populations. However, it does not handle well networks with moderate/large number of nodes. This is not unexpected since the normalising constant of the pseudonetworks grows considerably as the number of nodes increases. Therefore the ratio  $A_i/N^i$  also grows (since  $N$  remains invariant)

and more terms are needed to achieve a good accuracy. Results for  $k = 300$  could not be obtained due to the assumption of normal usage (this method applies for  $k < \max\{\rho_i^{-1}\}$ , i.e.  $k < 250$ ).

We obtained the exact values of  $g_m(k)$  in table 6.3 and also the normalising constant of the pseudonetworks by using our own method (represented by equations 6.22 and 6.25).

We also compare the exact results with our approximate results (from the integral representation of  $g_m(k)$ ). This is presented in table 6.4. We can see that our method becomes more accurate when  $k$  is large, and we could not obtain reasonable results at lower populations. This is because we have used Stirling's formula which is not accurate at lower numbers. Furthermore, the integral using Stirling formula is multiplied by the coefficients  $A_{im_i}$  which becomes large as  $m$  increases. Therefore, our method is accurate only for sufficiently large  $k$  (limitations of our method will be discussed in the next subsection) .

In summary, McKenna's method is accurate at low/moderate populations, but it does not handle well networks with moderate/large number of nodes. Conversely, our approximate method is not applicable at lower populations (where we can use an efficient and exact formula), but it is accurate at higher populations even at moderate number of nodes. Furthermore, we can still handle large number of nodes at low/moderate populations by using the exact convolution formula as discussed before.

### Limitations of our method and numerical example

We have calculated the normalising constant and the cell throughput for networks of various sizes and populations. In cases where the population is low/moderate (i.e.  $k \leq 100$ ) it is more convenient to use equation 6.25 (which is exact). This produces

an algorithm whose efficiency does not depend on the number of nodes.

In networks with moderate number of nodes and moderate populations ( $m \leq 30$  and  $100 < k \leq 300$ ) it is more convenient to use equation 6.22 since it is exact and the algorithm is efficient.

We used equation 6.23 for networks with moderate number of nodes and large populations ( $m \leq 30$ ). To evaluate the integrals we used the numerical integration package in *Mathematica* V.2.0, running in a Sun/Sparc workstation.

Finally, we used equation 6.25 for networks with both large number of nodes and population, although the algorithm is less efficient.

As we have seen, one limitation of our approximate method is that it becomes inaccurate if the population is small and the number of nodes is large. Typically our method applies for  $k > 100\gamma_g$ .

Another limitation is that if the population is too large compared with the normalising constant's mode the integrals of interest becomes too small and underflow occurs. Typically, our method applies for  $k < 500\gamma_g$ .

In table 6.5 we analysed the effect of increasing the number of nodes and the population at constant ratio  $C = m/k = 0.1$ . We consider a network with loads  $\rho_1 = 0.2$ ,  $\rho_2 = 0.3$ ,  $\rho_3 = 0.5$ , and  $\rho_4 = 0.1$  (rows 1 to 4). We also consider a network with loads  $\rho_1 = 0.002$ ,  $\rho_2 = 0.003$ ,  $\rho_3 = 0.005$ , and  $\rho_4 = 0.001$  (rows 5 to 8). We obtained the exact results by using equations 6.22 and 6.25. The approximate results were obtained (when possible) by using equation 6.23.

We observe that in the first case the cell throughput remains practically at the same point as  $m$  and  $k$  grow. This is so because the mode  $\gamma_g$  shifts to the right as  $m$  and  $k$  increases. Therefore the network becomes less saturated. In the second case, the mode  $\gamma_g$  is small, and practically does not change as  $m$  and  $k$  increases. Therefore the cell throughput increases, since the network becomes saturated.

We can also observe that the approximate results are very accurate. The reason

| $m$ | $k$  | $g_m(k)$               |                        | $T_m(k)$ |         | $\gamma_g$ |         |
|-----|------|------------------------|------------------------|----------|---------|------------|---------|
|     |      | exact                  | approx.                | exact    | approx. | exact      | approx. |
| 2   | 20   | $2.2 \cdot 10^{-3}$    | -                      | 1.86     | -       | 3          | 1.78    |
| 20  | 200  | $2.51 \cdot 10^{-22}$  | -                      | 1.7915   | -       | 32.0       | 33.0    |
| 80  | 800  | $5.23 \cdot 10^{-84}$  | -                      | 1.7864   | -       | 132.0      | 134     |
| 100 | 1000 | $1.62 \cdot 10^{-104}$ | -                      | 1.7861   | -       | 166        | 168     |
| 10  | 100  | $1.19 \cdot 10^{-153}$ | $1.19 \cdot 10^{-153}$ | 86.74    | 86.74   | 1          | 0.09    |
| 15  | 150  | $2.44 \cdot 10^{-151}$ | $2.41 \cdot 10^{-151}$ | 116.65   | 116.88  | 1          | 0.14    |
| 20  | 200  | $9.74 \cdot 10^{-353}$ | $9.73 \cdot 10^{-353}$ | 136.48   | 136.87  | 1          | 0.19    |
| 30  | 300  | $7.65 \cdot 10^{-560}$ | $7.67 \cdot 10^{-560}$ | 156.20   | 156.19  | 1          | 0.29    |

Table 6.5: Cell throughput and normalising constant.

for this is that only the second integral in equation 6.23 is approximate. We used Stirling's formula for the factorial which is very accurate at higher populations. A high degree of accuracy is also observed in the normalising constant's mode.

### 6.4.3 Conclusion

We have studied the behaviour of medium/large-sized networks with the objective of finding their asymptotic behaviour. By analysing the generating function of the normalising constant, we have seen that although the network's total number of nodes is large, the generating function can be obtained by convoluting only five sequences. We then observed that the normalising constant increases and then decreases as the population grows to infinity. Its mode is approximately the sum of each individual node functions' mode plus a *shift to the right* due to the node of multiplicity 1. The cell throughput function becomes saturated when the population grows to infinity. We observed that the saturation point depends only on the maximum parameter  $\rho_i$ , and the effect of increasing the number of stations is a reduction in the cell throughput. These qualitative results are well-known but we also used the exact and approximate equations derived in chapter 5 to produce quantitative conclusion too.

# Chapter 7

## Conclusion and Future Work

In our highly mobile society, we face a major technological challenge, how to provide data communications for people frequently away from their wireline, i.e. static computers. That laptop, palmtop and portable computers have proved so popular, clearly demonstrates the vast potential number of applications which may appear later this decade.

Packet radio and cellular network technology has now evolved to the point where its protocols can support multi-channel operation within an integrated environment.

Throughout this thesis we have outlined a number of design issues relating to cellular networks. The intent was to give an overview of the problems faced by the system designer when developing these networks and also to show how quantitative models could help solve these problems.

In chapter 2 we presented an evolutionary view of packet radio and cellular networks. We saw how digital cellular networks emerged from packet radio technology and how they have been moved into the field of mobile communication systems. We proposed a new cellular network that operated using multi-channel radio links. It was necessary, therefore, to introduce analytical performance models to help design an efficient system.

In chapter 3 we undertook a study of buffer administration techniques in the base

station. We showed that a multi-channel operation required a multi-queue model, in which a single server travelled in cyclic order at discrete time intervals. We represented this multi-queue system by a Markov chain whose states corresponded to the number of packets in each queue, and to the position of the server at the beginning of each time interval. We then obtained the steady-state solution of this Markov chain. This gave us the queue length distribution and the channel throughput. The above method allowed the computation of these distributions when using most reasonably sized systems. However, from a practical point of view, its main limitation was that it produced a large number of linear equations that need to be solved numerically. We then suggested two potential improvements for our method: one was to take advantage of the highly symmetrical structure of the transition probability matrix by using aggregation-disaggregation methods (see [Stew 91]). The alternative was to make use of the sparsity of the same matrix by using a method such as that given by [SaCh 81].

In chapter 4 we introduced a new method for evaluating the sojourn time distribution in multi-queue systems. This was achieved by looking at its server in a new way. Taking this approach, we considered the inter-visit time as a random variable represented by transitions in a discrete Markov chain. We then obtained an equivalent phase-type system that produced the inter-visit time probability mass function. Afterwards we used these results to produce equations for the residual service time distribution, the waiting time distribution in discrete phase-type queues, and finally the sojourn time distribution. We observed that this could help us greatly to assess sojourn-time distribution and its moments. Furthermore, this technique was of greater generality and could be extended to several other polling strategies. However, the result of using a simplified Markov chain to represent the behaviour of the various queues was that the model produced less accurate results when traffic was very heavy. We then suggested a further improved method. However this

required greater computational effort due to its complexity. This method considered the complete Markov chain (similar to that in chapter 4) to describe the server operation.

The method presented in chapter 3 was very accurate, however its main limitation was the large size of the state space of the Markov chain. The method of chapter 4 produced more efficient algorithms and could also be used to calculate the queue length distribution, although it was less accurate at higher loads.

In chapter 5 we introduced a closed queuing network model to analyse asymptotic properties of cellular networks. Our approach was based on a partial fraction expansion of the generating function of the normalising constant. We obtained an exact closed formula which could be used in a great number of cases. We also obtained the integral representation of these equations. These formulas were more suitable where the network population was large, since they produced a very compact formula and an efficient algorithm for numerical evaluation. We noted that this method produced algorithms more efficient than those already existing and was of greater generality. However, a further improvement would be to find the asymptotic expansion of the normalising constant's integral representation. In this way, we could avoid using numerical integration techniques when calculating the normalising constant.

In chapter 6 we demonstrated how these techniques could help the system designer to analyse and compare different solutions and to identify trouble spots in the network protocols. We put this into practice by addressing some of the design questions described in chapter 2. We compared three different buffer administration techniques. These varied in the allocation of radio channels between messages. We showed the strengths and weaknesses of these techniques by examining their performance. Subsequently, we described how to select a suitable number of channels for a cell. This was achieved by considering the traffic requirements of the base station's area. We used the phase-type model to produce load curves which reflected the

base station's workload. The number of channels and stations supported could be determined by analysing this curve. Finally, we discussed the problem of congestion in the base station and how this could be avoided. We used our asymptotic method to calculate the cell throughput and to give its bounds when the number of packets in the cell approached infinity.

## 7.1 Future direction for cellular networks

As new cellular systems and architectures emerge, researchers are turning their attention to the future by formulating the next generation of wireless information networks. They will need to devise networks which can carry all types of information so that future demands can be met. As a result, traffic in both radio and wired links should be able to support data, voice and image packets. Future cellular systems must therefore operate in harmony with broadband integrated services digital networks. Within the models presented in chapters 3 and 4 the time between arrivals in the multi-queue system was assumed to be distributed exponentially. This assumption becomes invalid when the traffic includes voice, since it then becomes bursty. Our analysis could therefore be extended to consider non-Markovian traffic.

Furthermore, as urban areas become more densely populated, new techniques will become essential to ensure better use of the spectrum. With this population increase traffic demand in a cell will eventually reach its full capacity. A further growth in traffic within the cell would require either: a revision of the cell boundaries or a dynamic system of re-allocating channels. The former technique is called *cell splitting* and involves creating smaller cells so that frequencies can be re-used more often. The latter technique is called *dynamic channel assignment* and consists of re-arranging the set group of channels according to changing demands. These techniques have been employed in cellular mobile phone service (CMPS) networks. If we wished to predict the performance of these networks, it would be necessary

to incorporate a variable number of channels (depending on the current state) and queues in our model. This would be another suggestion for further research.

# Appendix A

## Exact Analysis of a Two-Queue System

We present here an exact analysis of a two-queue system served in cyclic order by a channel controller operating according to the policy described in chapter 3. We assume that each queue has buffer length  $C = 2$  and one transmitter is available (i.e.,  $K = 1$ ). The arrival rates in each queue are  $\lambda_1$  and  $\lambda_2$  respectively.

We use the method described in chapter 3 (with some simplifications in the notation) and obtain the transition probability matrix of the underlying Markov chain. We show how the steady state solution can be numerically evaluated. We also show how the number of states can be reduced by aggregating equivalent states.

### A.1 State Space

We define the state for the two-queue system by the following set:

$$S = \{(i, (n_1 n_2)) | i = 1, 2; n_i = 0, 1, 2\} \quad (\text{A.1})$$

States  $(1, (00)), (1, (01)), (1, (02)), \dots, (1, (22))$  are numbered 1 to 9, and states  $(2, (00)), (2, (01)), (2, (02)), \dots, (2, (22))$  are numbered 10 to 18 respectively.

## A.2 State Transition Probabilities

Let  $a_0$  and  $a_1$  be the probability of 0 and 1 arrivals in queue 1, and  $b_0$  and  $b_1$  be the probability of 0 and 1 arrivals in queue 2 during a time slot of length  $\theta$ . Considering the number of arrivals in each queue to be independent Poisson distributed random variables, we have:

$$\begin{aligned} a_0 &= \exp(-\lambda_1\theta) & b_0 &= \exp(-\lambda_2\theta) \\ a_1 &= \lambda_1\theta a_0 & b_1 &= \lambda_2\theta b_0 \end{aligned} \tag{A.2}$$

Let  $a_1^+$  and  $a_2^+$  be the probability of at least 1 and 2 arrivals in queue 1 and  $b_1^+$  and  $b_2^+$  be the probability of at least 1 and 2 arrivals in queue 2. Clearly these values are given by:

$$\begin{aligned} a_1^+ &= 1 - a_0 & b_1^+ &= 1 - b_0 \\ a_2^+ &= a_1^+ - a_1 & b_2^+ &= b_1^+ - b_1 \end{aligned} \tag{A.3}$$

Let  $T = [t_{ij}]$  be the transition probability matrix of the Markov chain with state space  $S$ . In order to evaluate its elements  $t_{ij}$  ( $1 \leq i, j \leq 18$ ) we give an example. State  $(1, (2 \ 1))$  indicates that the channel controller is pointing to queue 1, and there are 2 and 1 customers in queues 1 and 2 respectively at the time immediately before its operation. The first packet of queue 1 is then removed from the queue and the channel controller stops its operation because the transmitter has been filled. The system will then go to state  $(2, (1 \ 1))$  if there are no arrivals during the next time slot (this probability is given by  $a_0 b_0$ ). If there are one or more arrivals in queue 2 and no arrivals at queue 1, the next state will be  $(2, (1 \ 2))$  (in this case there will be packet loss if the number of arrivals at queue 2 is greater than 1). Notice that there is no transition from state  $(1, (2 \ 1))$  to any state  $(1, (n_1 \ n_2))$  since the channel controller stops pointing to queue 2 whenever it finds queue 1 not empty.

The state transition matrix can then be obtained, using the procedure described in chapter 3. Thus we have:

$$T = \begin{bmatrix} T_1 & T_2 \\ T_3 & T_4 \end{bmatrix} \quad (\text{A.4})$$

where the matrices  $T_1, T_2, T_3$  and  $T_4$  are given by:

$$T_1 = \begin{bmatrix} a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1 b_0 & a_1 b_1 & a_1 b_2^+ & a_2^+ b_0 & a_2^+ b_1 & a_2^+ b_2^+ \\ a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1 b_0 & a_1 b_1 & a_1 b_2^+ & a_2^+ b_0 & a_2^+ b_1 & a_2^+ b_2^+ \\ 0 & a_0 b_0 & a_0 b_1^+ & 0 & a_1 b_0 & a_1 b_1^+ & 0 & a_2^+ b_0 & a_2^+ b_1^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{A.5})$$

$$T_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1 b_0 & a_1 b_1 & a_1 b_2^+ & a_2^+ b_0 & a_2^+ b_1 & a_2^+ b_2^+ \\ 0 & a_0 b_0 & a_0 b_1^+ & 0 & a_1 b_0 & a_1 b_1^+ & 0 & a_2^+ b_0 & a_2^+ b_1^+ \\ 0 & 0 & a_0 & 0 & 0 & a_1 & 0 & 0 & a_2^+ \\ 0 & 0 & 0 & a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1^+ b_0 & a_1^+ b_1 & a_1^+ b_2^+ \\ 0 & 0 & 0 & 0 & a_0 b_0 & a_0 b_1^+ & 0 & a_1^+ b_0 & a_1^+ b_1^+ \\ 0 & 0 & 0 & 0 & 0 & a_0 & 0 & 0 & a_1^+ \end{bmatrix} \quad (\text{A.6})$$

$$T_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1 b_0 & a_1 b_1 & a_1 b_2^+ & a_2^+ b_0 & a_2^+ b_1 & a_2^+ b_2^+ \\ 0 & a_0 b_0 & a_0 b_1^+ & 0 & a_1 b_0 & a_1 b_1^+ & 0 & a_2^+ b_0 & a_2^+ b_1^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1^+ b_0 & a_1^+ b_1 & a_1^+ b_2^+ \\ 0 & 0 & 0 & 0 & a_0 b_0 & a_0 b_1^+ & 0 & a_1^+ b_0 & a_1^+ b_1^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & b_0 & b_1 & b_2^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_0 & b_1^+ \end{bmatrix} \quad (\text{A.7})$$

$$T_4 = \begin{bmatrix} a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1 b_0 & a_1 b_1 & a_1 b_2^+ & a_2^+ b_0 & a_2^+ b_1 & a_2^+ b_2^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1 b_0 & a_1 b_1 & a_1 b_2^+ & a_2^+ b_0 & a_2^+ b_1 & a_2^+ b_2^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_0 b_0 & a_0 b_1 & a_0 b_2^+ & a_1^+ b_0 & a_1^+ b_1 & a_1^+ b_2^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{A.8})$$

### Steady state probabilities

Let  $\pi_i$  be the steady state probability that the Markov chain with transition probability matrix  $T$  is in state  $i$  ( $1 \leq i \leq 18$ ). It can be easily verified (see [Neut 81] for example) that the vector  $\pi$  of probabilities  $\pi_i$  is given by:

$$\pi = \mathbf{u}[(I - T)S + W]^{-1} \quad (\text{A.9})$$

where  $I$  is the identity matrix,  $S$  is equal to the identity matrix except for the last element of the main diagonal which is equal to 0,  $W$  is a matrix in which all

elements of the last column are equal to 1 and the others are equal to 0, and the vector  $\mathbf{u}$  is equal to the first row of  $W$ .

### Aggregating equivalent states

In balanced systems (e.g. when  $\lambda_1 = \lambda_2$ ), by symmetry we can verify, for example that states  $(1, (n_1 \ n_2))$  and  $(2, (n_2 \ n_1))$  ( $0 \leq n_1, n_2 \leq C$ ) are equivalent. Then the state transition matrix can be reduced to:

$$T_{red} = \begin{bmatrix} a_0^2 & a_0 a_1 & a_0 a_2^+ & a_0 a_1 & a_1^2 & a_1 a_2^+ & a_0 a_2^+ & a_1 a_2^+ & (a_2^+)^2 \\ a_0^2 & a_0 a_1 & a_0 a_2^+ & a_0 a_1 & a_1^2 & a_1 a_2^+ & a_0 a_2^+ & a_1 a_2^+ & (a_2^+)^2 \\ 0 & a_0^2 & a_0 a_1^+ & 0 & a_0 a_1 & a_1 a_1^+ & 0 & a_0 a_2^+ & a_1 a_2^+ \\ a_0^2 & a_0 a_1 & a_0 a_2^+ & a_0 a_1 & a_1^2 & a_1 a_2^+ & a_0 a_2^+ & a_1 a_2^+ & (a_2^+)^2 \\ 0 & 0 & 0 & a_0^2 & a_0 a_1 & a_0 a_2^+ & a_0 a_1^+ & a_1 a_1^+ & a_1^+ a_2^+ \\ 0 & 0 & 0 & 0 & 0 & 0 & a_0 & a_1 & a_2^+ \\ 0 & a_0^2 & a_0 a_1^+ & 0 & a_0 a_1 & a_1 a_1^+ & 0 & a_0 a_2^+ & a_1^+ a_2^+ \\ 0 & 0 & 0 & 0 & a_0^2 & a_0 a_1^+ & 0 & a_0 a_1^+ & (a_1^+)^2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_0 & a_1^+ \end{bmatrix} \quad (\text{A.10})$$

The steady state probabilities,  $\pi_{red}$  say, are then calculated by using equation A.9.

## Appendix B

# Irreducible Representations of a Phase-Type Distribution

We presented in chapter 4 a phase-type model of a multiqueue system comprising  $N$  queues. We assumed that a server visits these queues circularly at discrete intervals  $\tau_n$  ( $n = 0, 1, \dots$ ). This server stops its operation when  $K$  transmitters have been filled or all queues have been visited.

We have seen that the inter-visit time probability mass function  $\{r_k\}$  can be regarded as being of phase-type of representation  $(\pi', (T - R))$  where  $T$  is the transition probability matrix of the Markov chain which represents the server's operation,  $\pi'$  is the steady state solution of the same Markov chain and  $R$  is the lower triangular part of  $T$ .

We have seen that this representation is reducible. A difficult problem still unsolved is that of characterising the *minimal* representation for a given phase type distribution (see [Neut 81]). In those, the order of the reduced matrix is as small as possible. However, we can obtain *one* irreducible representation by solving a system of linear equations. This is described as follows.

Let  $k^*$  be the maximum integer such that  $r_k \neq 0$ . It can be easily verified that  $k^* = \lceil N/K \rceil$ , since the maximum number of time intervals between visits occurs

when all queues have been found not empty between  $\tau_n$  and  $\tau_{n+k^*}$ .

Let  $(\beta', S)$  be a irreducible representation of  $(\pi', (T-R))$ . It can be easily verified that if  $S$  is a  $k^* \times k^*$  matrix such that only the elements  $s_{ij}$  ( $i > j$ ) are not zero, then  $(\beta', S)$  is irreducible.

Therefore we propose the following algorithm to obtain  $(\beta', S)$ :

### Algorithm 2

1. Initialise  $\beta = \{1/k^*, \dots, 1/k^*\}$
2. Initialise  $s_{ij} = 0$  if  $i \leq j$  and  $s_{ij} = 1$  otherwise.
3. Insert  $k^*$  variables  $x_i$  ( $i = 1, \dots, k^*$ ) in the upper triangular part of  $S$ .
4. Solve the system of equations

$$\beta' S^{j-1} \mathbf{S}^\circ = \pi' X^{j-1} \mathbf{X}^\circ \quad j = 1, \dots, k^*$$

where  $X = T - R$ ,  $\mathbf{X}^\circ$  and  $\mathbf{S}^\circ$  are column vectors so that the matrices

$$\begin{pmatrix} S & \mathbf{S}^\circ \\ \mathbf{0}' & 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} X & \mathbf{X}^\circ \\ \mathbf{0}' & 1 \end{pmatrix}$$

are stochastic (note that  $\mathbf{0}'$  is the null-vector).

## Appendix C

# Asymptotic Expansions of the Normalising Constant

In chapter 6 we evaluated the normalising constant of a closed queuing network model using a method proposed by J. Mc Kenna and D. Mitra (see [McMi 82]). The network under consideration comprises a total of  $m_t = m_p + m_q$  nodes where  $m_p$  and  $m_q$  denote the number of non-IS and IS nodes respectively.

We considered a network with constant loads  $\rho_1, \rho_2, \rho_3$ , and  $\rho_4$  of multiplicity  $m, m, m$  and 1 respectively (clearly  $m_p = 3m + 1$ ). The total IS load  $\rho_0$  is assumed to be equal to 1. We denoted the normalising constant of such network as  $g_m(k)$ , where  $k$  is the network population.

J. Mc Kenna and D. Mitra introduced a new approach to solve the problem of evaluating  $g_m(k)$  when  $k$  is considerably large. They obtained an integral representation of the normalising constant and expanded this equation in inverse powers of a large parameter  $N$ . Using their method, this asymptotic expansion was proved to be:

$$g_m(k) \sim (k! \prod_{i=1}^{m_p} \alpha_i)^{-1} \sum_i \frac{A_i}{N^i} \quad (\text{C.1})$$

where  $\alpha_i$  is given by:

$$\alpha_i = 1 - N\beta\rho_i \quad i = 1, \dots, m_p \quad (\text{C.2})$$

and  $\beta$  is:

$$\beta = \frac{k}{N} \quad (\text{C.3})$$

The normal usage assumption requires  $\alpha_i > 0$  (i.e.  $k < \max\{\rho_i^{-1}\}$ ). The asymptotic expansions are in powers of  $(1/N)$  where  $N$  is a parameter designed to reflect the network size. This parameter was chosen to be:

$$N = \max\left\{\frac{1}{\rho_i}\right\} \quad (\text{C.4})$$

As a consequence of their method, the coefficients  $A_i$  of the asymptotic expansion are related to the normalising constant of a hypothetical network called a *pseudonetwork*. This pseudonetwork comprises  $m_p$  nodes (therefore it lacks IS centers). In the same way, their loads  $\rho_1^*, \rho_2^*, \rho_3^*$  and  $\rho_4^*$  have multiplicity  $m, m, m$  and 1 respectively. These loads are related to the loads of the original network. They can be calculated by using the equation below:

$$\rho_i^* = \frac{N\rho_i}{\alpha_i} = \frac{N\rho_i}{1 - N\beta\rho_i} \quad i = 1, \dots, m_p \quad (\text{C.5})$$

Let  $g(j)$  be the normalising constant of this pseudonetwork when its population is  $j$ . Note that although five coefficients are considered by the authors to be adequate for both small and large networks, we calculated  $g_m(k)$  by using eight coefficients since we were interested to check its accuracy when  $m$  is large. Using the method described in [McMi 82], we found that these coefficients are given by:

$$A_0 = 1$$

$$A_1 = -\beta g(2)$$

$$A_2 = 2\beta g(3) + 3\beta^2 g(4)$$

$$\begin{aligned}
A_3 &= -6\beta g(4) - 20\beta^2 g(5) - 15\beta^3 g(6) \\
A_4 &= 24\beta g(5) + 130\beta^2 g(6) + 210\beta^3 g(7) + 105\beta^4 g(8) \\
A_5 &= -120\beta g(6) - 924\beta^2 g(7) - 2380\beta^3 g(8) - 2520\beta^4 g(9) - 945\beta^5 g(10) \\
A_6 &= 720\beta g(7) + 7308\beta^2 g(8) + 26432\beta^3 g(9) + 44100\beta^4 g(10) + \\
&\quad 34650\beta^5 g(11) + 10395\beta^6 g(12) \\
A_7 &= -5040\beta g(8) - 64224\beta^2 g(9) - 303660\beta^3 g(10) - 705320\beta^4 g(11) - \\
&\quad 866250\beta^5 g(12) - 540540\beta^6 g(13) - 135135\beta^7 g(14) \\
A_8 &= 40320\beta g(9) + 623376\beta^2 g(10) + 3678840\beta^3 g(11) + 11098780\beta^4 g(12) + \\
&\quad 18858840\beta^5 g(13) + 18288270\beta^6 g(14) + 9459450\beta^7 g(15) + \\
&\quad 2027025\beta^8 g(16)
\end{aligned} \tag{C.6}$$

We observe that the first four coefficients were explicitly derived in [McMi 82].

The numerical results in chapter 6 were derived using these expressions.

# Bibliography

- [Abra 70] N. Abramson, "The ALOHA System – Another Alternative for Computer Communications", *Proc. Fall Joint Computer Conference*, 1970.
- [AbKu 73] N. Abramson and F. Kuo, *Computer Communication Networks*, Prentice-Hall, 1973.
- [Abra 77] N. Abramson, "The throughput of packet broadcast channels", *IEEE Trans. Comm.*, **25**, no 1, pp. 117–128, Jan. 1977.
- [BrBa 80] S. C. Bruell and G. Balbo, *Computational Algorithms for Closed Queueing Networks*, North-Holland, 1980.
- [BoJe 71] G. E. P. Box and G. M. Jenkins, *Time Series Analysis*, Holden-day, 1971.
- [BoGr 88] O. J. Boxma and W. P. Groenendijk, "Waiting Times in Discrete-Time Cyclic-Service Systems", *IEEE Trans. Comm.*, **36**, no 2, pp. 164–170, Feb. 1988.
- [Cinl 75] E. Çinlar, *Introduction to Stochastic Processes*, Prentice-Hall, 1975.
- [ChHW 75] K.M.Chandy U.Herzog and L.Woo, "Approximate Analysis of General Queueing Networks", *IBM J. Res. Develop.*, **19**, no 36, pp. 43–49, Jan. 1975.

- [ChLy 91] K. C. Chua and K. M. Lye, "Performance of Multichannel CSMA/CD network with channels susceptible to noise", *Electronic Letters*, **27**, no 1, Jan. 1991.
- [Coop 90] R. B. Cooper, *Introduction to Queuing Theory*, third edition, CEEPress Books, 1990.
- [CoMu 69] R. B. Cooper and G. Murray, "Queues Served in Cyclic Order", *B.S.T.J.*, pp. 675–689, Mar. 1969.
- [Cox 87] D.C.Cox, "Universal Portable Radio Communications", *Proceedings of the IEEE*, **75**, no 4, pp. 436-477, Apr. 1987.
- [DaSm 91] M.C.Davie and J.B.Smith, "A cellular packet radio data network", *Elect. and Comm. Eng. Journal*, pp. 137–143, Jun. 1991.
- [EvDa 89] D. Everitt and D. Manfield, "Performance Analysis of Cellular Mobile Communication Systems with Dynamic Channel Assignment", *IEEE Journal on Selec. Areas in Comm.*, **7**, no 8, pp. 1172–1179, Oct. 1989.
- [GrHa 85] D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*, second edition, John Willey, 1985.
- [HaPa 92] P.G. Harrison and N.M.Patel, *Performance Modelling of Communication Networks and Computer Architectures*, Addison-Wesley Publishing Company, 1992.
- [HaOh 72] O. Hashida and K. Ohara, "Line Accommodation Capacity of a Communication Control Unit", *Review of the Electr. Comm. Laboratories, Nippon Telegraph and Teleph. Public Corp.*, **20**, no 3–4, pp. 231–239, Mar–Apr 1972.

- [Hay 84] J. F. Hayes, *Modeling and Analysis of Computer Communications Networks*, Plenum Press, 1984.
- [Hef 82] M. Heffes, "Moment Formulae For a Class of Mixed Multi-Job-Type Queuing Networks", *The Bell Systems Technical Journal*, **61**, no 5, pp. 709–745, May–Jun. 1982.
- [JuTo 87] J. Jubin and J. D. Tornow, "The DARPA Packet Radio Network Protocols", *Proceedings of the IEEE*, **75**, no 1, pp. 21–32, Jan 1987.
- [King 90] P. J. B. King, *Computer and Communication Systems Performance Modelling*, Prentice-Hall, 1990.
- [Klei 75] L. Kleinrock, *Queuing Systems - Volume 1: Theory*, John Wiley & Sons, 1975.
- [Klei 76] L. Kleinrock, *Queuing Systems - Volume 2: Computer Applications*, John Wiley & Sons, 1976.
- [KlTo 75] L. Kleinrock and F. A. Tobagi, "Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Models and Their Throughput-Delay Characteristics", *IEEE Trans. Comm.*, **12**, no 12, pp. 1400–1416, Dec. 1975.
- [KoKo 77] H. Kobayashi and A. G. Konheim, "Queueing Models for Computer Communications System Analysis", *IEEE Trans. Comm.*, **25**, no 1, pp. 2–28, Jan. 1977.
- [Koba 78] H. Kobayashi, *Modeling and Analysis: An Introduction to System Performance Evaluation Methodology*, Addison-Wesley Publishing Company, 1978.

- [KoBi 91] Y. Kogan and A. Birman, "Asymptotic analysis of closed queueing networks with bottlenecks", *Proc. of the IFIP WG 7.3 International Conference on the Performance of Distributed Systems and Integrated Communication Networks*, Kyoto, Japan, ed. T. Hasegawa, H. Takagi and Y. Takahashi, North-Holland, pp. 265–280, 1992.
- [KoMe 72] A. G. Konheim and B. Meister, "Service in a loop system", *J. Ass. Comput. Mach.*, **19**, pp. 92–108, Jan. 1972.
- [Kueh 78] P. J. Kuehn, "Multiqueue Systems with Nonexhaustive Cyclic Service", *The Bell Systems Technical Journal*, **58**, no 3, pp. 671–698, Mar. 1979.
- [Lee 82] W. C. Y. Lee, *Mobile Communications Engineering*, Mc Graw Hill Book Company, 1982.
- [Leib 61] M. A. Leibowitz, "An approximate Method for Treating a Class of Multiqueue Problems", *IBM Journal* **5**, pp. 204–209, Jul. 1961.
- [LeSi 90] H. Levy and M. Sidi, "Polling Systems: Applications, Modeling, and Optimization", *IEEE Trans. Comm.*, **38**, no 10, pp. 1750–1760, Oct. 1990.
- [Macd 79] V. H. Mac Donald, "The Cellular Concept", *The Bell Systems Technical Journal*, **58**, no 1, pp. 15–41, Jan. 1979.
- [McMi 82] J. McKenna and D. Mitra, "Integral Representations and Asymptotic Expansions for Closed Markovian Queueing Networks: Normal Usage", *The Bell Systems Technical Journal*, **61**, no 5, pp. 661–683, May–Jun. 1982.
- [McBa 91] P. E. McKenney and P. E. Bausbacher, "Physical and Link Layer Modeling of Packet Radio Network Performance", *IEEE Journal on Selected Areas in Comm.*, **9**, no 1, pp. 59–64, Jan. 1991.

- [MiMc 86] D. Mitra and J. McKenna, "Asymptotic Expansions for Closed Markovian Networks with State Dependent Service Rates", *Journal of the ACM*, **33**, no 3, pp. 568–592, Jul. 1986.
- [MiMc 84] D. Mitra and J. McKenna, "Asymptotic Expansions and Integral Representations of Moments of Queue Lengths in Closed Markovian Networks", *Journal of the ACM*, **31**, no 2, pp. 346–360, Apr. 84.
- [Mitr 87] I. Mitrani, *Modelling of Computer and Communication Systems*, Cambridge University Press, 1987.
- [Neut 77] M. F. Neuts, "Algorithms for the waiting time distributions under various queue disciplines in the  $M/G/1$  queue with service time distributions of phase-type", *Algorithmic Methods in Probability*, North-Holland Publishing Company, 1977.
- [Neut 79] M. F. Neuts, "A Versatile Markovian Point Process", *J. Appl. Prob.*, **16**, pp. 764–779, 1979.
- [Neut 81] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models - an algorithmic approach*, Johns Hopkins, 1981.
- [Olve 74] F.W.J.Olver, *Introduction to Asymptotics and Special Functions*, Academic Press, New York, 1974.
- [PaGa 89] J.D. Parsons and J.G. Gardiner, *Mobile Communication Systems*, Halsted Press, 1989.
- [Pitt 79] B. Pittel, "Closed exponential networks of queues with saturation: the Jackson-type stationary distributions and its asymptotic analysis", *Math. Oper. Res.*, **6**, pp. 357–378, 1979.

- [Purs 89] M. B. Pursley, "Delay and Throughput for Three Transmission Schemes in Packet Radio Networks", *IEEE Trans. Comm.*, **37**, no 12, pp. 1264–1274, Dec.1989.
- [RuBa 90] I. Rubin and J. E. Baker, "Media Access Control for High-Speed Local Area and Metropolitan Area Communications Networks", *Proceedings of the IEEE*, **78**, no 1, pp. 168–203, Jan. 1990.
- [SaCh 81] C. H. Sauer and K. M. Chandy, *Computer Systems Performance Modeling*, Prentice-Hall,1981.
- [ShWe 87] N. Shacham and J. Westcott, "Future Directions in Packet Radio Architectures and Protocols", *Proceedings of the IEEE*, **75**, no 1, pp. 83 – 99, Jan. 1987.
- [Schi 70] L. Schiff, "Traffic Capacity of Three Types of Common-User Mobile Radio Communication Systems", *IEEE Trans. Comm.*, **18**, no 1, pp. 12–21, Feb. 1970.
- [Stal 88] W. Stallings, *Data and Computer Communications*, Collier Macmillan, second edition, 1988.
- [Stew 91] W. J. Stewart, *Numerical Solution of Markov Chains*, Marcel Dekker, 1991.
- [StTo 89] J. S. Storey and F. A. Tobagi, "Throughput Performance of an Unslotted Direct-Sequence SSMA Packet Radio Network", *IEEE Trans. Comm.*, **37**, no 8, pp. 814 – 823, Aug. 1989.
- [Tane 89] A. S. Tanenbaum, *Computer Networks*, Prentice-Hall, 2nd. Edition, 1989.
- [Taka 86] H. Takagi, *Analysis of Polling Systems*, M.I.T. Press, 1986.

- [TaTT 88] T. Takine, Y. Takahashi, and T. Hasegawa, "Exact Analysis of Asymmetric Polling Systems with Single Buffers", *IEEE Trans. Comm.*, **36**, no 10, pp. 1119 – 1127, Oct. 1988.
- [ToKl 76] F. A. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part III – Polling and (dynamic) split channel reservation multiple access", *IEEE Trans. Comm.*, **24**, pp. 832 – 845, Aug. 1976.
- [Toba 80] F. A. Tobagi, "Multiaccess Protocols In Packet Communication Systems", *IEEE Trans. Comm.*, **28**, no 4, pp. 468 – 488, Apr. 1980.
- [Toba 87] F. A. Tobagi, "Modeling and Performance Analysis of Multihop Packet Radio Networks", *Proceedings of the IEEE*, **75**, no 1, pp. 135 – 155, Jan. 1987.
- [WaMH 91] B. Walke, W. Mende, G. Hatziliadis, "CELLPAC: A Packet Radio Protocol Applied to the Cellular GSM Mobile Radio Network", *41th IEEE Vehicular Technology Conference*, St. Louis U.S.A., pp. 408–413, May 1991.
- [Walr 88] J. Walrand, *An Introduction to Queueing Networks*, Prentice-Hall, 1988.
- [WiRa 85] N.D. Wilson and S.S. Rappaport, "Cellular Mobile Packet Radio Using Multiple-Channel CSMA", **132**, no 6, pp. 517 – 526, Oct. 1985.
- [ZhPa 90] K. Zhang and K. Pahlavan, "An Integrated Voice/Data System for Mobile Indoor Radio Networks Using Multiple Transmission Rate", *IEEE Trans. Vehic. Technol.*, **39**, no 1, pp. 75–82, Feb. 1990.