TEKNILLINEN KORKEAKOULU
Teknillisen fysiikan ja matematiikan osasto
Teknillisen fysiikan koulutusohjelma

Eeva Nyberg

Calculation of Blocking Probabilities and Dimensioning of Multicast Networks

Diplomi-insinöörin tutkintoa varten tarkastettavaksi jätetty diplomityö

Työn valvoja professori Jorma Virtamo
Työn ohjaaja professori Jorma Virtamo

Espoo, October 18, 1999

HELSINKI UNIVERSITY OF TECHNOLOGY ABSTRACT OF THE MASTER'S THESIS

Author:	Eeva Nyberg
Department:	Department of Engineering Physics and Mathematics
Major subject:	Systems and Operations Research
Minor subject:	Telecommunications Technology
Title:	Calculation of Blocking Probabilities and Dimensioning of Multicast Networks
Title in Finnish :	Jakeluliikenneverkkojen eston laskenta ja mitoitus
Chair:	S-38 Telecommunications Technology
Supervisor:	Professor Jorma Virtamo
Instructor:	Professor Jorma Virtamo

This thesis is a study on mathematical models for calculating blocking probabilities and dimensioning of multicast networks. Multicasting is a bandwidth saving transmission technique designed for group applications such as video conferencing. The transmission reaches many different end-users without a separate connection required for each user. Due to this characteristic, conventional blocking probability analysis of loss networks has to be modified. The study derives a new algorithm for calculating blocking probabilities in dynamic multicast networks. The exact algorithm is based on the well-known algorithm for calculating blocking probabilities in hierarchical access networks. The algorithm is modified by introducing a new convolution method, the OR-convolution. The algorithm is applied to an existing infinite user population multicast model, for which until now only a single link solution has been given. The study derives a new finite user population model, which is in accordance with the infinite user population model, and applies the algorithm to this new network model. The algorithm is further extended by taking into account the effect of non-multicast traffic, background traffic, in the network.

The dimensioning of the network is studied using the finite population model. An optimum capacity allocation is found for an example network using Moe's principle.

Due to the complexity of the exact algorithm, the use of an approximate method, the Reduced Load Approximation (RLA), is studied.

The work assumes that the reader has a basic understanding of Probability theory and Stochastic processes. The theory behind Stochastic processes and network protocols are however briefly introduced.

Number of Pages:	84
Keywords:	Multicast, network, blocking probability, dimensioning, Moe's principle, RLA, Poisson process, multinomial distribution

Department fills

Approved: **Library:**

TEKNILLINEN KORKEAKOULU **DIPLOMITYÖN TIIVISTELMÄ**

Tekijä:	Eeva Nyberg
Osasto:	Teknillisen fysiikan ja matematiikan osasto
Pääaine:	Systeemi- ja operaatiotutkimus
Sivuaine:	Teletekniikka

Työn Nimi:	Jakeluliikenneverkkojen eston laskenta ja mitoitus
Title in English:	Calculation of Blocking Probabilities and Dimensioning of Multicast Networks
Professuuri:	S-38 Teletekniikka
Työn valvoja:	Professori Jorma Virtamo
Työn ohjaaja:	Professori Jorma Virtamo

Työssä tutkitaan estotodennäköisyyksien laskemiseen ja jakeluliikenneverkkojen mitoittamiseen tarkoitettuja matemaattisia malleja. Jakeluliikenneverkoissa lähetys saavuttaa ryhmän loppukäyttäjiä kapasiteettia säästäen, koska lähetys ei vaadi omaa kaistaa jokaiselle käyttäjälle, vaan kaista jaetaan yhteisillä reiteillä. Perinteiset estotodennäköisyyden laskentamallit eivät päde jakeluliikenteelle sellaisenaan. Työssä johdetaan uusi algoritmi estotodennäköisyyksien laskemiseksi dynaamisissa jakeluliikenneverkoissa. Algoritmi perustuu tunnettuun hierarkisen tilaajaverkon eston-laskentamalliin. Tätä algoritmia muokataan työssä korvaamalla perinteinen konvoluutio uudella OR-konvoluutiolla. Algoritmia sovelletaan jo aikaisemmin esitettyyn äärettömän populaation jakeluliikennemalliin, joka on toistaiseksi ratkaistu vain yhden linkin tapaukselle. Työssä johdetaan uusi äärellisen populaation jakeluliikennemalli, johon myös sovelletaan algoritmia. Mallia yleistetään edelleen ottamalla huomioon verkossa kulkevan muun liikenteen, ns. taustaliikenteen, vaikutus estotodennäköisyyteen.

Verkon mitoitusta tutkitaan äärellisellä populaatiomallilla. Esimerkkiverkolle lasketaan optimaalinen kapasiteettiallokaatio käyttäen Moen kriteeriä.

Tarkan algoritmin laskennallisesta vaikeudesta johtuen työssä tutkitaan vähennetyn kuorman approksimaatiomenetelmän käyttöä jakeluliikenneverkoissa.

Työn ymmärtäminen vaatii todennäköisyyslaskun ja stokastisten prosessien alkeiden tuntemista. Työssä tosin esitetään lyhyesti stokastisten prosessien ja verkkoprotokollien perusteita.

Sivumäärä:	84
Avainsanat:	Jakeluliikenne, multicast, estotodennäköisyys, mitoitus, Moen periaate, vähennetyn kuorman approksimaatio (RLA), Poisson-prosessi, multinomijakauma

Täytetään osastolla

Hyväksytty: **Kirjasto:**

Preface

The Master's thesis was carried out at the Laboratory of Telecommunications Technology at Helsinki University of Technology. The work was part of the COST 257 project funded by Nokia Networks, Sonera, and Tekes.

Professor Jorma Virtamo, who acted not only as my supervisor, but as my instructor as well, I thank for taking the time to engage himself with my project. During the last months of my thesis, I also had the pleasure of working with Ph.D. Samuli Aalto. I thank him for helping me translate my thoughts into mathematics.

The personnel at the laboratory I thank for the hilarious coffee breaks that spiced my days. At the last stages of my thesis, working late became a habit, not least due to the five o'clock tea gang, Arja and Kirsi.

The persons that have always been part of my life, my parents and sisters, I hug warm-heartedly. Olli I hug for his love, humor, and ironing.

Helsinki, October 18, 1999

Eeva Nyberg

Contents

Abbreviations

ATM	Asynchronous Transfer Mode
DVMRP	Distance Vector Multicast Routing Protocol
FDDI	Fiber Distributed Data Interface
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IHL	Internet Header Length
IP	Internet Protocol
LAN	Local Area Network
MBone	(Internet) Multicast Backbone
MOSPF	Multicast Open Shortest Path First
PIM	Protocol Independent Multicast
QoS	Quality of Service
RLA	Reduced Load Approximation
TCP	Transmission Control Protocol
TTL	Time to live
UBR	Unspecified Bit Rate
UDP	User Datagram Protocol
UNI	User Network Interface
VC	Virtual Circuit
VCI	Virtual Circuit Identifier
VPI	Virtual Path Identifier

Chapter 1

Introduction

The purpose of this thesis is to examine blocking probabilities and the dimensioning of multicast networks. A multicast transmission originates at a source and, opposed to a unicast transmission, is replicated at various network nodes to form a tree-and-branch structure. The transmission reaches many different end-users without a separate transmission required for each user. A multicast connection has therefore a bandwidth saving nature. Blocking occurs in a network when, due to limited capacity, at least one link on the route is not able to admit a new call. Traditional mathematical models to calculate blocking probabilities in tree-structured networks exist for unicast traffic. Due to different resource usage, these models cannot directly be used for multicast networks where requests from different users arrive dynamically. Only recently, have mathematical models to calculate blocking probabilities in multicast networks been studied. The model studied in this work is based on the model proposed by Karvo et al. in [10]. The model assumes an infinite user population and Poisson arrivals, but allows general holding times. The paper by Karvo et al. covers only the simplified case of all but one link in a network having infinite capacity. Extending the calculations to the whole network has been done only approximately in [11].

The present study extends the single link case discussed in [10] and [11] to a multicast network with any number of finite capacity links. The network model remains as a point-to-multipoint model, allowing the formulation of an exact algorithm, which in the case of multiple sources would be too complex. The exact algorithm is based on the well-known algorithm for calculating blocking probabilities in hierarchical multiservice access networks, where link occupancy

distributions are alternately convolved and truncated. The resource sharing of multicast connections requires modification of the algorithm by using a new type of convolution, the OR-convolution. However, as the number of multicast channels increases, calculations based on the exact algorithm present a problem. The use of the Reduced Load Approximation (RLA) to calculate blocking probabilities in a multicast network is therefore studied.

Based on the multicast traffic model for an infinite user population, a traffic model for a finite user population is derived. Blocking probabilities are studied for networks with finite user populations behind leaf links. In addition to studying blocking probabilities for simplified networks with all but one link having infinite capacity, the exact algorithm for calculating blocking probabilities in networks is used to obtain exact end-to-end channel blocking probabilities for example networks. Furthermore, dimensioning of a multicast network where subscriptions arrive from finite user populations is studied and an optimum capacity allocation is found by using Moe's principle.

The algorithm is further extended to include background traffic, allowing the analysis of networks carrying mixed traffic, e.g. multicast and unicast traffic. The background traffic algorithm is applied to both the infinite and finite user population models.

The thesis presents a mathematical study for a specific telecommunication transport technique. Hence, both the telecommunication framework and the mathematical theory will first be introduced. The study starts by introducing the basic theory and technology behind multicast and telecommunication networks. The third chapter reviews the theory of stochastic processes and Markov processes in particular. In the fourth chapter, basic teletraffic terminology in the theory of loss networks is presented. Previous work on multicast networks from the teletraffic point of view is discussed in chapter 5. Two multicast models are studied in depth. Comparisons are made between the multicast model by Karvo et al. [10] and the layered video-transmission model by Chan and Geraniotis [5]. In chapter 6, the multicast single link model is extended to the whole network. The blocking probabilities in a multicast network given by the new model and the use of RLA as an approximate method for calculating blocking probabilities in the multicast network, are studied and compared. The multicast traffic model for a finite user population is presented in chapter 7, together with the exact algorithm for calculating end-to-end channel blocking probabilities. In chapter 8, dimensioning of multicast networks

for finite user populations is studied. Chapter 9 introduces the revised algorithm, which generalizes the multicast network model by taking into account background traffic. The new algorithm is applied to both the infinite and finite user populations models. The thesis is concluded in chapter 10.

Chapter 2

Multicast Traffic and Networks

A unicast transmission is designed for point-to-point communication, where a source sends a message to only one receiver. Once the message is intended to be received by a group, using one of the point-to-multipoint transmissions, multicast or broadcast, is more effective. Broadcasting a message is transmitting it to all users on the network and may therefore require unnecessary bandwidth and/or limiting the number of recipients. A multicast transmission originating at a source is replicated at various network nodes to form a tree-and-branch structure. The transmission reaches the end-users requesting the transmissions without a separate transmission required for each user, as would be the case in a unicast transmission. A multicast connection has therefore a bandwidth saving nature (figure 2.1). A multicast transmission is sent to a

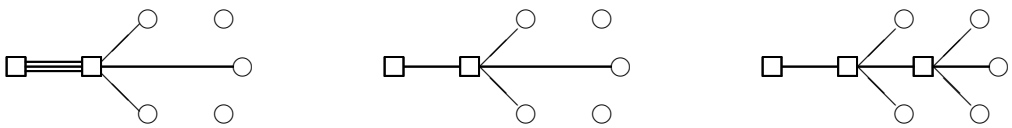

Figure 2.1: Difference between unicast, multicast, and broadcast.

multicast group, a group of users, requesting the transmission. The multicast groups are dynamic, receiver controlled groups, where a host can join or leave the group at any time. Traditionally, the use of multipoint connections has been limited to local area network (LAN) applications. Applications on the Internet relying on multicast transmission have increased in the past few years. Due to the Internet Multicast Backbone (MBone), IP-multicast has become a

widely used multicast protocol.

The first section reviews the principles of data networks. It provides a short introduction for readers less familiar with telecommunication networks. The second section introduces the main elements of the Internet Protocol (IP), IP-multicast, and the Internet Multicast Backbone (MBone). The use of Asynchronous Transfer Mode (ATM) technology is spreading, especially in the backbone networks. After an introduction to ATM, the use of ATM-multicast, and the integration of IP and ATM will be discussed.

2.1 Introduction to Networks

A network provides a platform for setting up connections between subscribers desiring to communicate with each other. A subscriber is typically represented by a terminal, e.g. a telephone or a computer. The network topology describes how the connections are formed (physically or logically).

2.1.1 Network Topologies

In a mesh topology, transmission lines, or links, exist between all subscribers in the network. Such a network topology becomes too expensive and inefficient once the number of subscribers and the distance between the subscribers grow. For most communication purposes, a star, bus, or ring topology is used. In these network topologies, the subscribers have to share transmission lines with each other. In a star topology, all the subscriber lines are point-to-point connected to the center of the network. This center usually consists of a switch. The switch permits the sharing of transmission lines between the subscribers, as it connects transmission lines. In the bus topology, the subscribers are connected to a shared transmission medium. The traffic transmitted on a bus can be picked up by any subscriber. A tree topology is a branching bus, the mutual transmission line branches out, and each branch itself has a bus topology. The ring topology is a closed bus, the transmission line between the subscribers is shared and forms a ring. The traffic (telephone call, data packet) can only travel in one direction. The topologies are presented in figures 2.2 and 2.3.

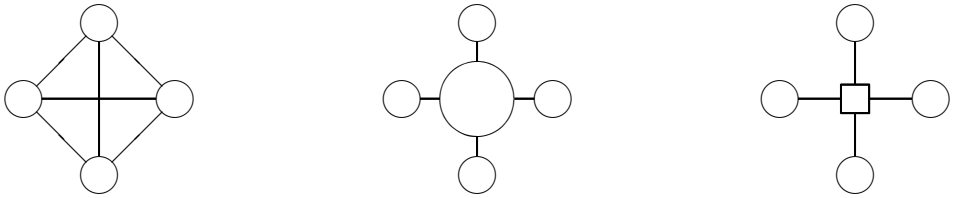

Figure 2.2: The mesh, ring, and star topologies.

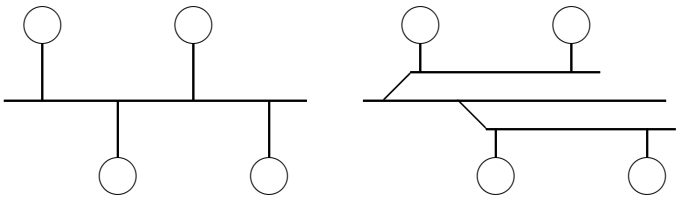

Figure 2.3: The bus and tree topologies.

2.1.2 LAN

The Local Area Network (LAN) connects users that are geographically close to each other. LANs may have a star, bus, tree, or ring topology. A LAN can be used when distances between terminals are from 100 m to 10 km. LANs are widely used to interconnect workstations, or data terminals, with processing resources close to each other. Due to the shared medium topologies used, LAN connections are point-to-multipoint and are capable of transmitting multicast traffic. An exception is the ring topology, which requires a multicast capable switch. Separate LANs can be interconnected using a high-speed backbone. The use of ATM as a LAN backbone is prevailing. Some of the technologies used in LANs are Ethernet, Token ring, Token bus, and Fiber Distributed Data Interface (FDDI). For more information, see [7], [18], and [20].

2.2 Interconnection Using Switches, Bridges, Routers, and Gateways

The star topology was defined as a set of point-to-point connections to a switch. A switch can also be used to connect networks with each other. An access

network links the switching nodes to the terminals, while a trunk network connects the switching nodes with each other (figure 2.4). In this way, the subscribers in the two access networks are connected to each other through two or more switches.

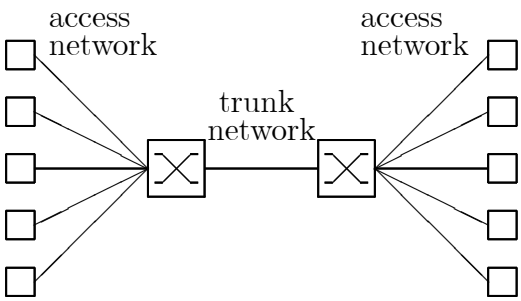

Figure 2.4: The use of a switch to connect two access networks via a trunk network.

A bridge can be used to interconnect two LANs of the same type. A bridge verifies that the recipient is on the other side of the bridge. A bridge does not control the packets, it only examines the forwarding LAN address. Because of this limited intelligence, bridges are only used to interconnect LANs inside an organization.

Routers also forward data packets between different networks irrespective of the underlying technology. A router chooses an appropriate route. The selection is made based on some predetermined criteria. The criteria can be e.g. the cost of the link or the time-to-live (TTL). The TTL tells how many routers a packet can traverse, measured in router hops, or how long, in real time, it can traverse in the network before being discarded. A router is more intelligent than a bridge, it controls, filters, and restricts data packets. A router can be used as a firewall to prevent intrusions to the network. A router forwards packets based on an IP-address. This means that it operates on top of the network level and can interconnect LANs. A router can connect LANs between different organizations, as a router can be given the capability of performing security surveillance.

A gateway is responsible for modifying the information traveling from one network to another, so that the receiving network is able to handle the packets. A router can also act as a gateway, and often the terms are used interchangeably.

2.2.1 Switching Technologies

The main difference between a digital telephone network and the Internet, a set of interconnected networks, is the switching technology used. Circuit switching technology is used for the telephone network while packet switching is the technology used in the Internet. When circuit switching is used, a whole physical line or channel is reserved for the call. The whole line is reserved until the call is ended, even if pauses occur during transmission. In packet switching, the physical transmission media are used more efficiently, as the data to be sent is divided into packets, and the packets are transmitted through the network. This allows data from different sources and with different destinations to share the physical links. Because each router handles the packets in a different way, the packets may be routed to different links, they may arrive in different orders, and delays between packets may occur.

Packet switching is appropriate when data is sent, as data can easily be constructed from the packets, as long as no packets have been lost. Packet switching is not appropriate in transmitting real-time voice or video, as the human ear and eyes cannot tolerate delays exceeding a certain limit. The ATM technology discussed in section 2.5 is a circuit switched technology where the information is divided into cells. A physical link can transport cells from different sources, as is done in packet switching. The circuit switching characteristics are preserved with the help of virtual circuits. A required amount of capacity is reserved for each virtual circuit and ATM cells from a source are all routed through the same virtual circuit thus retaining their order.

2.3 IP and IP-Multicast

The IP protocol is responsible for the addressing and routing of packets. It operates on the network layer. The networks are connected with gateways and routers. IP thus enables the transfer of packets between different networks, hence the term internetworking. The hosts use IP to communicate with other hosts in other networks. Each host in the Internet has a unique address, which is composed of a network address and a host address. The routers forward IP packets based on the destination network address.

2.3.1 IP Address

The IP datagram [24] shown in figure 2.5 consists of a set of fields belonging to the header and a data field. The 'Internet header length' (IHL) field contains

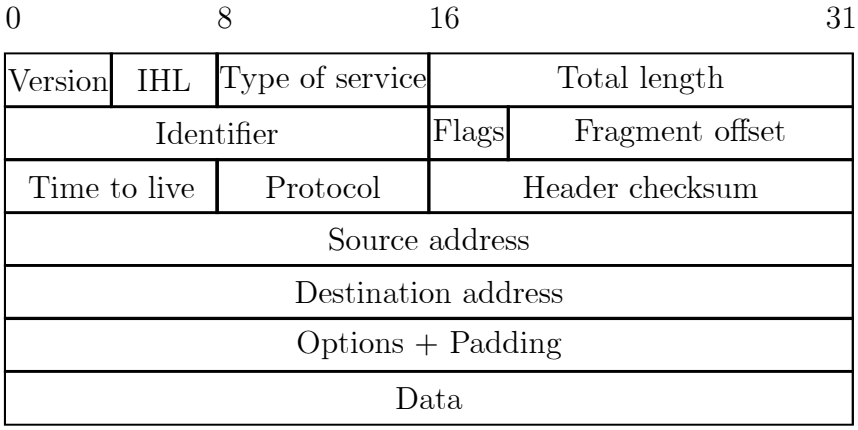

Figure 2.5: The IP datagram.

information on the actual length of the header. The 'Options and Padding' field has a variable length depending on the options used and the padding needed to ensure that the total length of the header is a 32-bit multiple. The 'Type of service' field specifies reliability, precedence, delay, and throughput parameters. The IP datagram may have to be split into smaller packets, as different networks have different packet sizes. The 'Identifier', 'Flag', and 'Fragment offset' are used to reassemble the packet. The 'Time to live field' is the number of router hops the packet can travel before it is discarded. The 'Protocol' field indicates the next level protocol that is to receive the data field at the destination. The 'Header checksum' is used for error detection on the header.

Each host is assigned a unique 32-bit address used in the source and destination fields. The IP address is divided into five classes shown in figure 2.6. The minimum length of the header is $5 * 32$ bits.

The size of the 'Data field' varies, but must be an 8-bit multiple. The total length of the IP-datagram restricts the maximum size of the Data field. The maximum of the total length is 64 kilobytes (65535 octets).

The five classes of the IP address shown in figure 2.6 are divided into three groups. Classes A through C are used for unicast messages and are composed of a network-address and a host-address. Class D is for multicast messages.

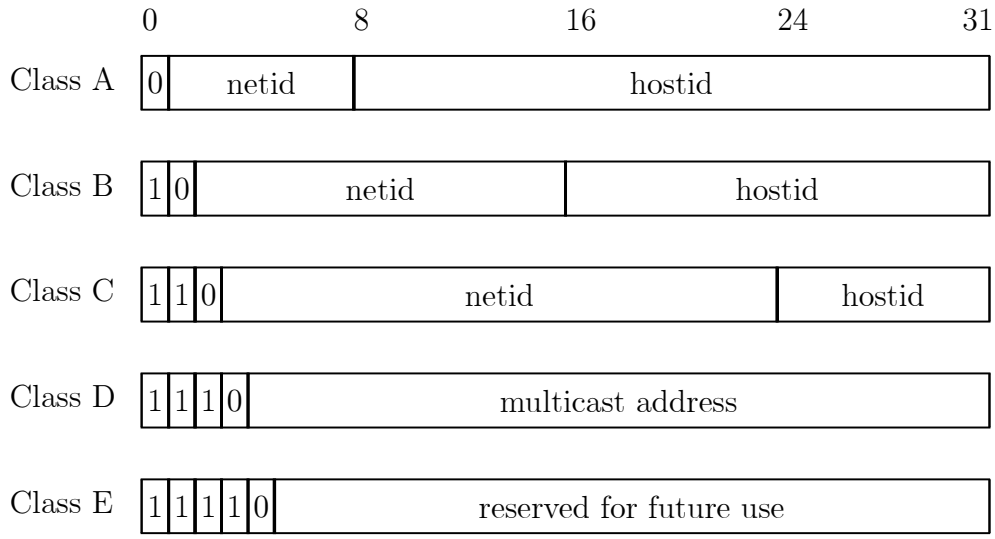

Figure 2.6: The IP address classes

The main difference between an IP unicast packet and an IP-multicast packet is the presence of the group address in the destination address field of the IP header (figure 2.5). One class D address is assigned to a multicast group. The multicast address is therefore logical, acting as a pointer to routing information. A unicast address on the other hand is global, possibly containing end-user information. Class E is reserved for future use.

2.3.2 TCP and UDP

The Internet is a connectionless network, where the routers forward packets without establishing a connection between the sender and the receiver. In a connectionless network, the packets can travel via different routes and can thus arrive at different times. Most applications need a connection-oriented protocol. Such a protocol is heavier to use, but can offer a larger variety of services than a connectionless protocol. The Transmission Control Protocol (TCP) is a connection oriented transport-level protocol that works on top of the IP-level. Multicast transmissions are, however, inherently connectionless, as the sender does not know the address of the receiver and therefore cannot establish a connection. The transport-level protocol used for IP-multicast datagrams is the User Datagram Protocol (UDP).

2.3.3 IP-Multicast Group

The IP offers a unique service class that is "best effort". Each IP packet is treated in the same way and has the same possibility of reaching its destination as all the other packets. Only in the case of congestion, or when the lifetime counter TTL is below a threshold value, is a packet discarded. The IP-multicast packet is delivered to the destination group members with the same "best effort" reliability. For a multicast transmission, this means that multicast packets may not reach all hosts or may arrive in a different order for different hosts.

The IP-multicast group is a dynamic group, where users can join and leave the group at any time. A host can be a member of one or more multicast groups and does not have to be a member of a group in order to send a multicast message to the group. In addition to being a dynamic group, a multicast group has no restriction on the number of members or their physical location. The protocol used to manage the dynamic group membership is the Internet Group Management Protocol (IGMP). IGMP is used by hosts and routers to share information about group membership in a physical network. IGMP gives the routers the necessary information on the multicast datagrams to forward. The IGMP runs between hosts and their contiguous routers. The mechanism allows a host to inform its local router of the group transmissions it wishes to receive. The routers also periodically monitor the active group members. In different versions of IGMP, the monitoring of active members, and the dynamics of the multicast group differ. For more detail, see [6] or [16].

The TTL field limits how far, i.e. over how many routers, a packet can traverse on the way to a destination. In IP-multicast, the destination is a group, and the TTL allows the application to monitor the position of the receivers relative to the sender.

A host joins a group by informing the IGMP of the groups it wants to join. The scaling properties of this receiver-initiated join process have two major advantages. The sender does not have to know the location and address of the receiver, sending the packet to the group address suffices. As the group grows in size, the probability that a new group member locates a nearby branch of the multicast distribution tree increases [16].

2.4 MBone and Transmission of a Multicast Datagram

IP-multicast is widely in use due to the Internet Multicast Backbone (MBone) founded in 1992. The MBone is a semipermanent testbed for IP-multicast. MBone is a virtual network layered on top of a part of the physical Internet. It is an interconnected set of subnetworks, e.g. multicast capable LANs, and routers, which support the delivery of IP-multicast. The networks that support multicast, called islands, are connected to each other by virtual links, tunnels. Multicast messages are forwarded through the tunnels in the non-multicast-capable parts of the network. IP-multicast packets are encapsulated as IP-over-IP, making them appear as normal unicast packets to the intervening non-multicast routers. The separate routing protocols used to forward multicast packets are Distance Vector Multicast Routing Protocol (DVMRP), Multicast Open Shortest Path First (MOSPF), and Protocol-Independent Multicast (PIM). The majority of MBone routers use the DVMRP for routing, the UDP for end-to-end transmission, and the IGMP for group management.

2.4.1 DVMRP

DVMRP forwards the first datagram to the entire internetwork. A router that forwards a packet to all leaf routers except the one it arrived from is said to flood the packet. If no group members exist behind the leaf router, the router sends prune messages back to the flooding router. The DVMRP thus creates a source-specific shortest path tree, with all leafs of the tree having group members. The flooding and sending of prune messages is repeated after a period, and a new routing tree evolves. The dynamic character of the tree is further enhanced by allowing the leaf routers to cancel a previously sent prune message by sending a graft message, when a new group member subscribes to the network. When more than one DVMRP router exists on a subnetwork, one router is elected the Dominant Router for the particular source subnetwork. For more information on DVMRP or the other routing protocols, see [6] and [16].

2.4.2 Transmission

The transmission and delivery of a multicast datagram can be divided into two cases. When the sender and the receiver are in the same subnetwork, e.g. LAN, the source station addresses the IP packet to the multicast group, the network interface maps the address to the Ethernet address, and the frame is sent. Receivers need to notify their IP layer that they want to join the multicast group.

The more complicated case occurs when the sender and receiver are located in different subnetworks. The routers must implement a multicast routing protocol that permits the handling of multicast packets between networks. The multicast router must construct delivery trees and forward the packets. Each multicast router must also implement a group membership protocol, i.e. IGMP. Based on the information conveyed by the IGMP, a router is able to determine which multicast packets need to be forwarded to which subnetworks. IP-multicast is supported by multicast routers using one of the multicast routing protocols (DVMRP, MOSPF, or PIM) and the information learned from IGMP.

2.5 ATM

The use of Asynchronous Transfer Mode (ATM) technology is spreading, as the demand for higher speed, flexibility, and quality of service (QoS) support is increasing. Many audio and video multicast applications, such as video conferencing and video-on-demand, need the high bandwidth and versatile on-line communication technology offered by ATM. The major drawback is that the current ATM does not support multipoint-to-multipoint connections. ATM is connection oriented and is therefore not suitable for the transportation of multicast packets in a dynamic group environment. The development of ATM to support multicast connections and the integration of IP and ATM to efficiently use the networks has therefore been an important research topic and has been discussed in [3] and [8]. The basic concepts of ATM are introduced in this section. The next section discusses the demands that multicast traffic has introduced on ATM and the solutions that implement IP-multicasting on ATM.

ATM is designed for both data transfer and audio and video applications. ATM is a circuit switched technology that relies on virtual circuits (VCs). ATM is therefore connection oriented. The data to be transported is divided into cells, similar to packets used in packet switching. The cell switching technology used in ATM is actually a combination of the two switching techniques. An ATM cell has 53 bytes ($53 * 8$ bits). Opposed to circuit switching, no physical circuit connection is established. Data cells from different sources may be transmitted via the same physical link. The difference to packet switching is that cells originating from a source are transported through the same route, thus retaining their order, called the virtual circuit.

The ATM cell consists of 48 bytes of data and a 5-byte address header. The fields defined in the header are 'Virtual Channel Identifier' (VCI) and 'Virtual Path Identifier' (VPI). The VCI in the header of the cell identifies to what virtual circuit the cell belongs in order for the router to retain the established virtual circuit. A virtual path is a collection of virtual circuits. A virtual path is used to simplify the switching of cells. The VPI identifies the virtual path that the connection belongs to, and the ATM switch needs only to know the VPI in the cell header. The routing of the data takes place only once for a given virtual connection. The first cell is routed, and the subsequent cells are passed on the same route. ATM is therefore a technology that promises high bandwidth, and is suitable for transmissions of audio and video signals. It is however unlikely that all users are connected to an ATM network. Therefore, the use of a more common network protocol, e.g. IP, is needed. The idea in integrating these two networks is the "Classical IP over ATM" discussed in the next section.

2.6 IP over ATM

The classical models of IP over ATM exploit ATM characteristics, but leave IP unchanged. The ATM characteristics being used are high speed at the user terminal and router interfaces, VC switching capabilities, and the Unspecified Bit Rate (UBR) service class support. VC connections are allowed within a single subnetwork, and their dynamic setup and release are used to simplify network management. VCs are set up when IP packets need to be transferred between terminals without connection. The UBR service class supports delay-tolerant applications. It is intended to support connectionless data traffic that

does not require QoS guarantees. The ATM UBR service class is well suited for IP "best effort".

The IP-multicast introduced in the previous section is a multipoint-to-multipoint service. ATM is not a shared medium technology and is not capable of setting up multipoint-to-multipoint connections. ATM defined in User-Network Interface (UNI) version 3.0 [23] is able to support point-to-multipoint connections. However, these connections are sender initiated. The sending host adds new receivers to point-to-point connections. Although a point-to-multipoint connection seems to support multicasting, this is not generally the case. Only a single host, the one who set up the point-to-multipoint connection, can send information to the formed group. Other hosts must set up new point-to-multipoint connections, if they want to interact with the group.

Methods of achieving multipoint-to-multipoint connections in ATM include setting up N point-to-multipoint connections in order to completely connect N hosts in a mesh topology. This method becomes infeasible, as the number of hosts grows large. An alternative way is to have one host, usually the server, acting as the root of the multicast tree and setting up point-to-multipoint connections with the N hosts in the network. These hosts in turn form point-to-point connections with the root. The server is then responsible for controlling all the connections, and congestion would occur at the server, as it needs to receive and transmit each multicast packet.

Adapting IP-multicast to ATM point-to-multipoint capabilities has been defined in an IETF standard [2]. This standard is based on the UNI 3.0/3.1. The UNI allows clusters of native ATM hosts to implement IP-multicast service using the UBR service class by means of either direct ATM point-to-multipoint connections or ATM-multicast servers. In shorter terms, separate ATM clusters communicate through multicast routers.

Some IP over ATM research aspects are considered in the paper by Guarene et al. [8]. The two integrated multicasts discussed are MBone over ATM and High-Performance Multicasting, where an IP-ATM-hybrid multicasting node is used. In this hybrid node, the flow merging is performed at the IP level and the flow replication at the ATM level.

Chapter 3

Markov Processes

A stochastic process is a collection of random variables X_t, which are indexed by the time of occurrence t: $\{X_t, t \in \mathcal{T}\}$, where $\mathcal{T} \subset \Re$ is the parameter space. In teletraffic theory, the random variable X_t may denote the number of arrived calls or the number of calls in progress at time t.

The distribution of a stochastic process is governed by the probabilities

$$P(X_{t_1} \leq x_1, ..., X_{t_n} \leq x_n),$$

where $(t_1, ..., t_n) \in \mathcal{T}^n$, $t_1 \leq t_2 \leq ... \leq t_n$ and $(x_1, ..., x_n) \in \Re^n$. The trivial stochastic process consists of independent random variables. The simplest non-trivial stochastic process, where the random variables are dependent only on the previous instant, is the Markov-process. A stochastic process is a Markov-process, if it has the Markov property:

$$P(X_{t_{n+1}} = x_{n+1} \mid X_{t_1} = x_1, ..., X_{t_n} = x_n) = P(X_{t_{n+1}} = x_{n+1} \mid X_{t_n} = x_n).$$

The distribution of the Markov process is therefore

$$\begin{aligned} P(X_{t_1} \leq x_1, ..., X_{t_n} \leq x_n) &= P(X_{t_1} \leq x_1) P(X_{t_2} \leq x_2 \mid X_{t_1} \leq x_1) \cdot \ldots \\ &\quad \cdot P(X_{t_n} \leq x_n \mid X_{t_{n-1}} \leq x_{n-1}). \end{aligned}$$

The Markov property allows the study of processes that depend on the previous instant of time, without considering how, through which states, the current state was reached. This implies that in order to study the process from time t onwards, the only information needed is the state of the process at time t.

A stochastic process can be either discrete or continuous. Processes in teletraffic are usually modeled with continuous Markov processes. A continuous

Markov process is time homogeneous, if the transition probability only depends on the time t elapsed

$$P(X_{s+t} = j \mid X_s = i) = P(X_t = j \mid X_0 = i) = p_{i,j}(t).$$

The transition probability indicates the probability that a transition from state i to state j will occur during time t. For continuous homogenous Markov processes, the transition rate $q_{i,j}$, which is independent of time, is defined as

$$\begin{aligned} q_{i,j} &= \lim_{\Delta t \to 0} \frac{p_{i,j}(\Delta t)}{\Delta t}, \text{ for } i \neq j \\ q_{i,i} &= -\sum_{j \neq i} q_{i,j}. \end{aligned}$$

The Memoryless Property The exponential distribution is often used to model the distribution of time intervals between events in the continuous time stochastic processes. This is due to the memoryless property. The memoryless property means that the distribution of the time before an event takes place is independent of the instant t when the previous event took place. More explicitly,

$$\begin{aligned} P(X < t + h \mid X > t) &= \frac{P(t < X < t + h)}{P(X > t)} \\ &= \frac{1 - e^{-\lambda h} e^{-\lambda t} - 1 + e^{-\lambda t}}{e^{-\lambda t}} \\ &= 1 - e^{-\lambda h} \\ &= P(X < h), \end{aligned}$$

where the time between events is exponentially distributed with parameter λ. Markov processes are memoryless in the sense that the intervals between transitions, e.g. arrivals or departures, of the process are exponentially distributed.

In teletraffic models, the time between events can be the time between arrivals to a system or the time between departures from a system. If the stochastic process is the number of arrivals to a system A_t, a new arrival increases the state by one unit. Similarly, if the stochastic process is the number of calls in the system N_t, then an arrival increases the state by one, while a departure decreases the state by one unit. Processes, where transitions are only allowed between neighboring states, are called birth-death-processes. Birth-death-processes are used to characterize most teletraffic systems. The models used in this work are based on this class of Markov processes, discussed in the next section.

3.1 Birth-Death-Processes

Birth-death-processes are continuous time Markov processes where transitions are only allowed between neighboring states, with an appropriate enumeration of the states. A transition to the next state is called a birth, while a transition to the previous state is called a death. As an example consider a system, where calls arrive with exponentially distributed time intervals, with intensity λ_i, and have exponentially distributed service times, with intensity μ_i. These intensities can depend on the state of the system, e.g. the number of calls in the system $N_t = i$, at time t. When a new call arrives to a system, N_t increases by one. When a call leaves the system N_t decreases by one. If no calls arrive N_t remains unchanged. This system can be characterized by a birth-death-process, with transition intensities

$$\begin{cases} q_{i,i-1} &= \mu_i \geq 0 \quad , i > 0, \\ q_{i,i+1} &= \lambda_i \geq 0 \quad , i > 0, \\ q_{i,j} &= 0 \qquad\quad , \text{if } |i-j| > 1. \end{cases}$$

These intensities are derived from the exponential distribution. If the process is a pure birth process, $\mu_i = 0$, $\forall i$. When time intervals between successive arrivals are independent and exponentially distributed, with parameter λ, the birth process, i.e. the arrival process A_t, is a Poisson process with probability distribution

$$P(A_t = i) = \frac{(\lambda t)^i}{i!} e^{-\lambda t}.$$

Steady state probabilities Stationary state probabilities, $\pi_i, i = 1, 2, ...$ can be calculated from the detailed balance equations for the birth death process,

$$\begin{aligned} \pi_{i+1} q_{i+1,i} &= \pi_i q_{i,i+1}, \text{ for } i \geq 0, \\ \pi_{i+1} \mu_{i+1} &= \pi_i \lambda_i, \text{ for } i \geq 0. \end{aligned}$$

When the process is stable, the transitions between states are balanced. Solving the detailed balance equations gives the stationary probability

$$\pi_i = \pi_0 \prod_{i=1}^{\infty} \frac{\lambda_i}{\mu_i}, \text{ i=1,2,....}$$

A sufficient condition for existence is that $\sum_{i=0}^{\infty} \pi_i = 1$ and $\pi_0 > 0$.

Reversibility Birth-death processes are reversible processes. A stochastic process X_t is reversible, if $(X_{t_1}, X_{t_2}, ..., X_{t_n})$ has the same distribution as $(X_{\tau-t_1}, X_{\tau-t_2}, ..., X_{\tau-t_n})$ for all $t_1, t_2, ..., t_n$, $\tau \in \mathcal{T}$. A Markov process is reversible, if and only if there exist stationary probabilities $\pi_i, \forall i$ that satisfy the detailed balance equations

$$\pi_i q_{i,j} = \pi_j q_{j,i}, \forall i, j.$$

For proofs, see [12] chapter 1. In the case of birth-death-processes, the detailed balance equations are satisfied and the process is always reversible.

Before calculating stationary state probabilities for some queuing models used in this study, Kendall's notation is presented. The notation gives a good general classification of queuing models.

3.1.1 Kendall's Notation

Kendall's notation classifies stochastic queuing models according to five parameters:

$A/B/n/s/k$		
A	=	inter arrival time distribution
B	=	holding time distribution
n	=	number of service stations
s	=	number of places in the system
	=	n + number of waiting places
k	=	user population

The inter arrival and holding time distributions, A and B, are usually assumed to be identically and independently distributed. Three basic distributions are used:

M	=	exponentially distributed inter arrival (or holding) times
D	=	deterministic times
G	=	generally distributed times

Teletraffic models can be divided into two main types of models: queuing and loss models. If the system does not have a waiting room, i.e. $n = s$, then the model is a pure loss system. In a pure loss model, calls that arrive when the system is full, i.e. not able to admit new calls, are lost. If n is finite, but the system has infinite capacity, that is $s = \infty$ the system is a pure queuing system. In a pure queuing system, there is always a possibility to wait for the server to be emptied. The default value for parameters s and k is ∞. They are usually included in the notation only when they have finite values.

In the sections to follow, stationary probabilities for queuing and loss models are given. The models can be divided into two main groups according to the value of s. The transition diagram of an infinite system is in figure 3.1 and the transition diagram of a finite system in figure 3.2.

Both chains satisfy the detailed balance equations, as the transitions between a pair of states are balanced, namely

$$\pi_i q_{i,j} = \pi_j q_{j,i}, \forall i, j,$$

and are therefore reversible processes.

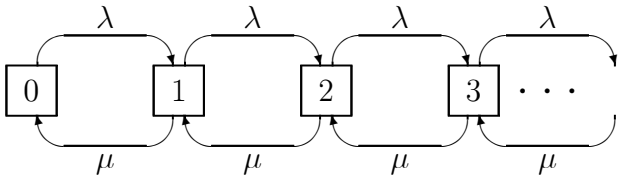

Figure 3.1: Markov chain with infinite state space.

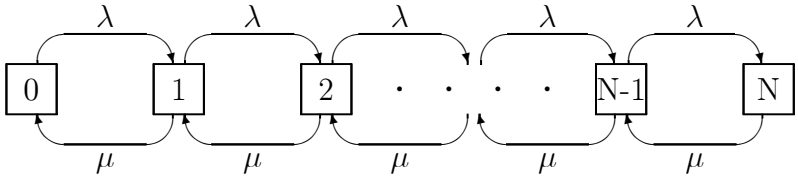

Figure 3.2: Truncated Markov chain.

3.1.2 Unrestricted Systems

$M/M/1$ System

This is a queue with exponentially distributed state independent inter arrival and departure times, one service station, and an infinite queue. The state transition diagram for an $M/M/1$ queue is presented in figure 3.1. The detailed balance equations are

$$\pi_{i+1}\mu = \pi_i\lambda, \text{ for } i = 0, 1, ...$$

Solving the set of equations gives

$$\pi_i = \pi_0\Big(\frac{\lambda}{\mu}\Big)^i = \pi_0\rho^i = (1-\rho)\rho^i,$$

where the ratio λ/μ is denoted by ρ, and $0 < \rho < 1$.

$M/M/n$ System

This system is a generalization of the $M/M/1$ system. The number of service stations is $n \geq 1$. Therefore, the system has a state independent arrival process and a state dependent departure process

$$\begin{aligned} \mu_i &= i\mu, \text{ for } i < n, \\ \mu_i &= n\mu, \text{ for } i \geq n. \end{aligned}$$

The detailed balance equations are

$$\begin{aligned} \pi_{i+1}(i+1)\mu &= \pi_i\lambda, \text{ for } i \leq n, \\ \pi_{i+1}n\mu &= \pi_i\lambda, \text{ for } i > n. \end{aligned}$$

Solving the set of equations gives

$$\begin{aligned} \pi_i &= \pi_0\frac{\rho^i}{i!}, \text{ for } i \leq n, \\ \pi_i &= \pi_0\frac{\rho^i}{n^{i-n}n!}, \text{ for } i > n, \\ \pi_0 &= \Big[\sum_{i=0}^{n}\frac{\rho^i}{i!} + \sum_{i=n+1}^{\infty}\frac{\rho^i}{n^{i-n}n!}\Big]^{-1}. \end{aligned}$$

As $n \to \infty$ the stationary state probabilities of the $M/M/\infty$ system are

$$\begin{aligned} \pi_i &= \pi_0\frac{\rho^i}{i!} \text{ for } i = 1, 2, ..., \\ \pi_0 &= \Big[\sum_{i=0}^{\infty}\frac{\rho^i}{i!}\Big]^{-1} = e^{-\rho}. \end{aligned}$$

3.1.3 Truncated Systems

Finite systems are also called truncated systems. The state space is truncated, as due to the capacity restriction some states are not allowed in the state space. The stationary state probabilities of a truncated Markov chain, which satisfy the detailed balance equations, differ from the stationary state probabilities of

an unrestricted Markov chain only by the normalization constant. The resulting Markov chain in the truncated space again satisfies the detailed balance equations. The proof can be found in [9] and [12]. The idea is that a removal of a pair of states that satisfy the detailed balance equations does not affect the stationary state probabilities, as the pair is balanced. Therefore, the truncation only affects the normalization constant, as the probability of the removed states is set to zero.

$M/M/1/n$ System

This queue is a truncated $M/M/1$ queue, where the queuing capacity is finite. It is therefore a mixture of a queuing and loss model. The difference between the $M/M/1$ queue and the $M/M/1/n$ queue can be seen by inspecting the state transition diagram in figure 3.2. The detailed balance equations are only defined for states with $i \leq n$. For these states, the stationary probabilities are identical to the $M/M/1$ queue. The normalization constant and thus the probability of the queue being empty are different, and the stationary state probability is

$$\pi_i = \pi_0 \Big(\frac{\lambda}{\mu}\Big)^i = \pi_0 \rho^i = \frac{1-\rho}{1-\rho^{n+1}} \rho^i, \text{ for } i = 0, 1, ..., n.$$

$M/M/n/n$ System

This is a pure loss model. There are n service stations in the system, with no queuing possibility. The departure intensity is the same as in the previous case, but for $i > n$ the intensity is zero. This system is called the Erlang system and the stationary state probability is

$$\begin{aligned} \pi_i &= \pi_0 \frac{\rho^i}{i!}, \text{ for } i \leq n, \\ \pi_0 &= \Big[\sum_{i=0}^{n} \frac{\rho^i}{i!}\Big]^{-1}. \end{aligned}$$

$M/M/n/n/k$ System

The last system introduced is a loss system with a finite user population $k > n$. In this system, both the arrival process and departure process intensities

depend on the state of the process.

$$\begin{aligned}\mu_i &= i\mu, \text{ for } i \leq n,\\ \mu_i &= n\mu, \text{ for } i > n,\\ \lambda_i &= (k-i)\lambda, \text{ for } k-i \geq 0.\end{aligned}$$

This system is called the Engset system, and the stationary probability is

$$\begin{aligned}\pi_i &= \pi_0 \binom{n}{i} \rho^i, \text{ for } i \leq n,\\ \pi_0 &= \Big[\sum_{i=0}^{n} \binom{n}{i} \rho^i\Big]^{-1}.\end{aligned}$$

Erlang B-Formula. In 1917, Erlang devised a formula, called the Erlang B-formula for calculating the blocking probability for the simplest loss system, the $M/M/n/n$ system. For a link (system) with capacity C (number of stations) and one traffic class (capacity requirement $c = 1$) with offered traffic intensity $a = \lambda/\mu$, the call blocking probability is

$$B^c = E(a, C) = \frac{a^C/C!}{\sum_{i=0}^{C} a^i/i!}. \tag{3.1}$$

The Insensitivity Property. The Erlang formula was later proven to hold for generally independent and identically distributed holding times with mean $1/\mu$. For the proof of this so-called insensitivity property, see [12]. An outline of the proof, as given in [15], is as follows.

The insensitivity property can be proven, by assuming that the holding time distribution is a mixture of finite convolutions of exponential distributions, resulting in gamma distributions. This new state process is a Markov process and satisfies the required partial balance equations. The state probabilities are therefore insensitive to the holding time distribution.

An arbitrary distribution $F(\cdot)$ can be expressed as the limit of a sequence of distributions, where each distribution in the sequence has mean $1/\mu$ and is a mixture of finite convolutions of exponential distributions. As the insensitivity property holds for each distribution in the sequence, it holds for the limit of the sequence, the arbitrary distribution.

3.2 Multidimensional Birth-Death-Processes

The theory presented in the previous section can be extended to multidimensional birth-death-processes. Figure 3.3 shows the transition diagram for a two-dimensional birth-death-process with infinite capacity. The stationary

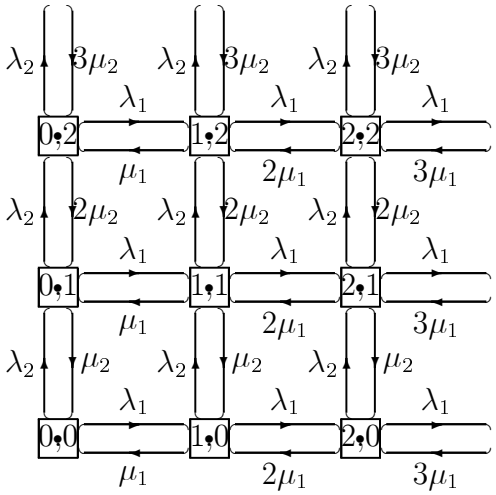

Figure 3.3: Transition diagram for a two dimensional Markov chain.

state probabilities of this chain have a product form

$$\pi(x_1, x_2) = G^{-1} \cdot \frac{{\rho_1}^{x_1}}{x_1!} \cdot \frac{{\rho_2}^{x_2}}{x_2!}.$$

Generally, for an N dimensional chain, the stationary state probabilities are

$$\pi(\mathbf{x}) = G^{-1} \prod_{i=1}^{N} \frac{{\rho_i}^{x_i}}{x_i!}.$$

The insensitivity and truncation theorems discussed in the previous section apply also to the multidimensional Markov chains. For proof, see [9].

Chapter 4

Loss Networks

4.1 Blocking in a Pure Loss System

Blocking occurs when a call is rejected, i.e., not able to enter the system. Clearly, blocking occurs only if the number of waiting places is finite. Blocking is measured by the probability that blocking occurs. Three different blocking probabilities can be considered: call, time, and traffic blocking. Call blocking is the probability that a call arrives when the system is full. Time blocking is the probability that the system is full at an arbitrary instant. Traffic blocking is the proportion of total traffic that is lost due to blocking.

The probabilities of call, time, and traffic blocking are not necessarily the same, but they depend on each other. Time blocking is usually easier to calculate than call blocking, but call blocking is often of more interest. If arrivals constitute a Poisson process, call blocking is equal to time blocking. If the user population is finite with exponentially distributed inter arrival times, as is the case of an Engset system, the call blocking can be expressed with the help of the time blocking probability.

Loss systems are usually used to model voice and video traffic, and queuing systems are used to model data traffic. This is due to the different requirements and technologies used in the connection and transfer of these types of traffic. Real-time voice and video traffic require reserved bandwidth, as they can tolerate only small delay variations. A connection carrying voice or video traffic can be established only if enough capacity is available in the network. Otherwise, the connection is refused or blocked. Data traffic can be sent in

packets using packet transmission when delay due to packets queuing in a buffer is permitted. Multicast techniques are used for a variety of applications, and this classification is somewhat coarse. Calls, or transfer of voice, can also wait in a queue for a connection to be established. Transferring video is one important application, and therefore the use of loss networks in modeling a multicast network is appropriate. Generally, video and voice traffic cannot tolerate delays and data traffic cannot tolerate loss of packets. The use of IP and ATM technology was discussed in the previous chapter. The use of ATM to transfer multicast traffic is tempting, as it is a circuit switched technology and therefore well suited for voice and video traffic.

In Kendall's notation, n is the number of service stations. A general definition for n is the capacity of the station or link. Traffic is transported in links, and the capacity of the link defines the amount of traffic that can be transported/held at the same time. When considering only one type of traffic, i.e. one traffic class, the capacity is a multiple of the capacity requirement of the traffic. For example, if a call requires two units of capacity, and the link capacity is four (or five), the system is able to carry two telephone calls, i.e. there are two service stations. In this case, it is appropriate to express the link capacity as two, instead of four. In the case of many different traffic classes, with each having different capacity requirements, the notation is not so simple. Generally, each capacity requirement is a multiple of a basic unit also called Basic Bandwidth Unit (BBU). If a system has many traffic classes each having the same capacity requirement, it is called a singleservice system. If the capacity requirement varies along different traffic classes, the system is a multiservice system.

Two parameters affect the system and thus the blocking in the system: the capacity of the system and the offered traffic intensity. Capacity was defined above. It indicates how many calls can be admitted to the system at the same time. If calls arrive with a rate λ and have mean holding time $1/\mu$, then the traffic intensity is the product of the two, $a = \lambda/\mu$. The unit is defined as one Erlang. The traffic intensity is also defined as the average number of calls in progress if the traffic were offered to an infinite system. To calculate blocking for a system, all that needs to be known are the arrival and departure processes of the calls and how many calls are allowed in the system at the same time.

4.2 Blocking Probability in a Link

The Erlang formula given in equation (3.1) cannot be used for loss systems carrying multiservice traffic. The formula is based on calculating the proportion of time, i.e. probability, that the system is full. Equation (3.1) is actually the time blocking probability, which for the case of Poisson arrivals is the same as the call blocking probability.

The blocking probability for a loss system with more than one traffic class needs to be defined using a more general definition of blocking states. Different types of traffic are divided into different traffic classes, class-k, $k = 1, ..., K$. A traffic class is characterized by its capacity requirement. The number of class-k calls in the system is x_k. The capacity of the link is C. Information on the capacity requirement of class-k calls, for $k = 1, ..., K$, is stored in a vector $\mathbf{d}$ of length K. Let $\mathcal{S}$ denote the set of all states that satisfy the capacity restrictions of the system

$$\mathcal{S} = \{\mathbf{x} \geq 0 \mid \mathbf{x} \cdot \mathbf{d} \leq C\}, \tag{4.1}$$

where $\mathbf{x} = (x_1, ..., x_K)$.

Blocking occurs in the states that are not able to admit a new call due to the capacity restriction. Let $\mathcal{S}_k$ denote the set of all states in which a new class-k call is not blocked.

$$\mathcal{S}_k = \{\mathbf{x} \in \mathcal{S} \mid (\mathbf{x} + \mathbf{e}_\mathrm{k}) \cdot \mathbf{d} \leq C\},$$

where $\mathbf{e}_k$ is a unit vector in the direction of k. Then $\mathcal{S}_k$ is the set of states which are able to admit a new class-k call with respect to the capacity restrictions of the network. The set of blocking states is therefore $\mathcal{S}_k^B = \mathcal{S} \setminus \mathcal{S}_k$. Finally, the time blocking probability is the probability that a state belongs to the set $\mathcal{S}_k^B$, i.e., the probability of the set $\mathcal{S}_k^B$. The probability is calculated by summing over the individual state probabilities that satisfy the condition.

The general definition for the call blocking probability for class-k traffic is

$$B_k^t = \frac{G(\mathcal{S}_k^B)}{G(\mathcal{S})} = 1 - \frac{G(\mathcal{S}_k)}{G(\mathcal{S})}, \tag{4.2}$$

where $G(\mathcal{S})$ denotes the state sum over all the unnormalized state probabilities $\tilde{\pi}(\mathbf{x})$ belonging to the set $\mathcal{S}$,

$$G(\mathcal{S}) = \sum_{\mathbf{x} \in \mathcal{S}} \tilde{\pi}(\mathbf{x}). \tag{4.3}$$

The blocking states of a link carrying two traffic classes are shown in figure 4.1. Traffic class one requires two units of capacity, while traffic class two requires one unit of capacity. The total capacity allowed on the link is six, which is illustrated by the linear constraint of the state space. The blocking states of the first traffic class are denoted by a small circle and the blocking states of the second traffic class by a large circle.

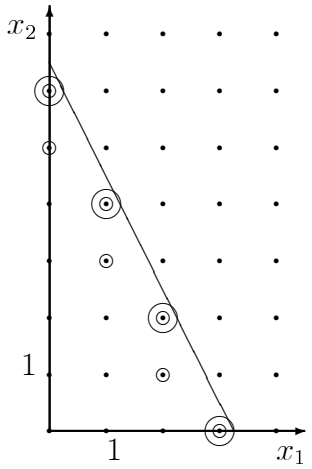

Figure 4.1: Blocking states in a link with two traffic classes.

4.2.1 Poisson Arrivals

The Erlang B-formula in equation (3.1) was derived assuming that the arrivals to the system are generated by a Poisson process. This assumption is very often made when studying teletraffic models, as many analytical calculations are feasible only when the arrivals constitute a Poisson process. The temptation to use Poisson arrival processes is due to the memoryless property of the exponential distribution. For a Poisson process, the times between two successive arrivals are exponentially distributed and independent. The memoryless property means that the distribution of the time before an event takes place is independent of the instant t, when the previous event took place. However, as discussed in section 3.1.3, due to the insensitivity property, the holding time or duration of service distribution does not have to be exponential. The results presented hold for any holding time distribution with finite mean $1/\mu_k$.

When the arrival process is Poisson the state probabilities $\pi(\mathbf{x})$ have a product form (section 3.2). Let a_k denote the intensity of class-k traffic,

$$a_k = \lambda_k/\mu_k,$$

where $1/\mu_k$ is the average holding time.

The state probability for the link is

$$\pi(\mathbf{x}) = G^{-1} \prod_{k=1}^{K} \frac{{a_k}^{x_k}}{x_k!} = G^{-1} \prod_{k=1}^{K} \tilde{\pi}_k(x_k) = G^{-1}\tilde{\pi}(\mathbf{x}), \tag{4.4}$$

where $\tilde{\pi}_k(x_k)$ denotes the unnormalized state probabilities and G^{-1} is the normalization constant, which is determined by the normalization condition

$$\sum_{\mathbf{x}\in\mathcal{S}} \pi(\mathbf{x}) = 1.$$

Equation (4.2) in terms of equations (4.3) and (4.4) reads

$$B_k^t = 1 - \frac{\sum\limits_{\mathbf{x}\in\mathcal{S}_k} \tilde{\pi}(\mathbf{x})}{\sum\limits_{\mathbf{x}\in\mathcal{S}} \tilde{\pi}(\mathbf{x})}. \tag{4.5}$$

4.3 Aggregate States Using Link Occupancy

As the number of traffic classes increases, the computation of the blocking probabilities becomes more time consuming due to the multidimensional state space. In the one link case, the multidimensional state space can be changed to a one dimensional aggregate state space. The new random state variable Z is the link occupancy. The state probabilities are calculated via convolution, which is equivalent to calculating the new probability generating function

$$P(Z = z) = \sum_{j=0}^{\infty} q(j) z^j = \prod_{k=1}^{K} \sum_{x_k=0}^{\infty} \pi_k(x_k) z^{x_k d_k},$$

where $\pi_k(x_k) = G^{-1} \frac{{a_k}^{x_k}}{x_k!}$ is the stationary probability for class-k traffic.

For a network, the dimensionality of the link occupancy state space is equal to the number of links in the network. The transition from the traffic class state space to the link occupancy state space is therefore generally useful only if the number K of traffic classes exceeds the number J of links.

The blocking probability of equation (4.5) expressed with the help of the link occupancy distribution is

$$B_k^t = \frac{\sum\limits_{j=C-d_k+1}^{C} q(j)}{\sum\limits_{j=0}^{C} q(j)}.$$

4.4 Blocking Probability in a Network

In the previous section, blocking probabilities were calculated for a single link with finite capacity. Using the non-blocking and admissible sets of states $\mathcal{S}_k$ and $\mathcal{S}$ respectively, the blocking probabilities in a network with two or more finite links can be calculated. The traffic classes k have the same properties as in the one link case, but they are further divided into classes that use the same route in the network. A route is a defined sequence of links. For a user u the links on its route belong to the set $\mathcal{R}_u$. Link j in the network does not necessarily carry all sets of traffic classes k. The capacity requirement d_k of class k is replaced by an J by K matrix $\mathbf{D}$. Its element $d_{j,k}$ is equal to the capacity required by class k on link j. If class-k traffic does not use link j, then $d_{j,k} = 0$. The set of admissible states $\mathcal{S}$ is analogous to equation (4.1)

$$\mathcal{S} = \{\mathbf{x} \geq 0 \mid \mathbf{Dx} \leq \mathbf{C}\},$$

where $\mathbf{C} = (C_1, ..., C_J)$ is the capacity restriction vector for the links. Similarly, the set $\mathcal{S}_{(k,u)}$ is

$$\mathcal{S}_{(k,u)} = \{\mathbf{x} \in \mathcal{S} \mid \mathbf{d}_j(\mathbf{x} + \mathbf{e}_k 1_{j \in \mathcal{R}_u}) \leq C_j, \forall j\},$$

where $\mathbf{d}_j = (d_{j,k})_{k=1,...,K}$ and $1_{j \in \mathcal{R}_u}$ is the indicator function equal to one if $j \in \mathcal{R}_u$ and zero otherwise.

Clearly, blocking occurs on the route, if it occurs on at least one link on the route. Conversely, a class-k call of user u is admitted to the network only if after the admission of the call, the capacity restrictions of all the links on the route $\mathcal{R}_u$ are satisfied.

The blocking states of a network with two users are shown in figure 4.2. There is only one traffic class requiring one unit of capacity. The network is a tree type network with two links with three units of capacity combining into one link with four units of capacity. Thus, the state space has three linear constraints. The blocking states for the first user class are denoted by a small circle and the blocking states for the second user by a large circle.

4.5 Access Network as an Example

In a tree network, the state probabilities of the common link are joint probabilities of the individual state probabilities at the leaf links and are calculated

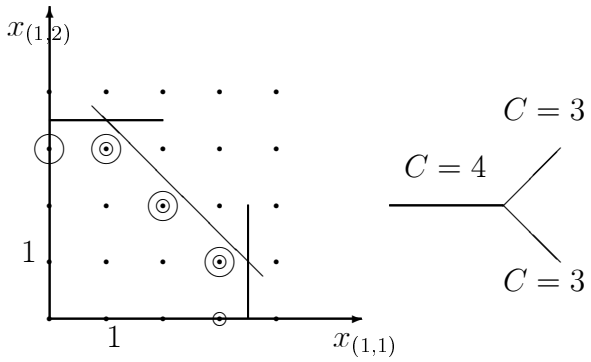

Figure 4.2: Blocking states for two traffic classes in the network on the left.

using convolution. The access network is used as an example of a tree-type network to show how the joint state probabilities are calculated for the network and how the capacity restrictions of the links in the network are taken into account. The access network is introduced here in order to lay the foundation for the study of tree-structured multicast networks.

Assuming a Poisson arrival process, the state probabilities for the leaf links are given by equation (4.4). The convolution and truncation are therefore done for an access network with product form state probabilities.

The example network is a hierarchical network with three levels, from right to left. In the second level, traffic from the first level links are combined and offered to the second level links. These links are combined to form the last link in the third level, called the common link. See figure 4.3. The capacity restrictions of each link are taken into account by truncating the state probabilities at each stage. As explained, the joint state probabilities at a stage are calculated by convolving the state probabilities of the previous stage. The truncated joint probability at the common link is then the overall joint state probability of the access network.

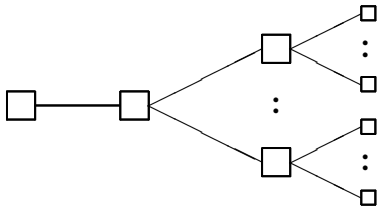

Figure 4.3: Hierarchical access network.

The state probabilities for each first level link $j \in \mathcal{J}_1 = \{1, ..., J_1\}$ are calculated using the unnormalized state probabilities in equation (4.4). The allowed states on the link are determined by the capacity restriction of the link. Changing state spaces from the K-dimensional space to the one-dimensional link occupancy state and defining the convolution operator $\otimes$ gives

$$Q_j^{(1)}(c) = \sum_{\mathbf{x}\cdot\mathbf{d}_j=c} \prod_{k=1}^{K} \tilde{\pi}_k(x_k) = [\bigotimes_{k=1}^{K} \tilde{\pi}_k](c),$$

where $\mathbf{d}_j$ is the capacity requirement vector for the first stage link $j \in \mathcal{J}_1$ and $Q_j^{(1)}$ is the link occupancy probability for link $j \in \mathcal{J}_1$.

Taking the capacity restriction of the first level links into account gives

$$Q_j^{(1)}(c) = \begin{cases} [\bigotimes_{k=1}^{K} \tilde{\pi}_k](c) & \text{, if } c \leq C_j, \\ 0 & \text{, otherwise.} \end{cases} \tag{4.6}$$

The state probabilities offered to the second level link $j \in \mathcal{J}_2$ are a combination of the state probabilities of the first level links $i \in \mathcal{I}_j$ terminating at link j and are calculated with the same convolution operator. The state probability on a second level link $j \in \mathcal{J}_2$

$$Q_j^{(2)}(c) = \begin{cases} [\bigotimes_{i\in\mathcal{I}_j} Q_i^{(1)}](c) & \text{, if } c \leq C_j, \\ 0 & \text{, otherwise.} \end{cases} \tag{4.7}$$

Finally, the allowed state probabilities in the common link J are

$$Q_J^{(3)}(c) = \begin{cases} [\bigotimes_{i\in\mathcal{I}_J} Q_i^{(2)}](c) & \text{, if } c \leq C_J, \\ 0 & \text{, otherwise.} \end{cases} \tag{4.8}$$

When calculating the blocking probability of class-k traffic for user u, the succession of convolutions and capacity restrictions have to be repeated in order to find the state sum of the states that are not blocked. The steps have to be repeated starting from equation (4.6). The capacity restriction, that is the truncation step, is tighter on the route $\mathcal{R}_u$, as there must be enough available capacity to accept a new class-k call. The capacity restriction to be satisfied has the form $c \leq C_j - d_{j,k}$, for $j \in \mathcal{R}_u$ and $j = 1, ..., J_i$, for $i = 1, 2$ or $j = J$. Otherwise, the steps are identical to the ones in equations (4.6) through (4.8).

The state probabilities $\hat{Q}_j^{(i)}(c)$ for $j = 1, ..., J_i$ at each stage $i = 1, 2, 3$ for the states that are not blocked in terms of the aggregate state space are

$$\hat{Q}_j^{(1)}(c) = \begin{cases} [\bigotimes_{k=1}^{K} \tilde{\pi}_k](c) & \text{, if } c \leq C_j - d_{j,k} \text{ and } j \in \mathcal{R}_u, \\ Q_j^{(1)}(c) & \text{, if } j \notin \mathcal{R}_u, \\ 0 & \text{, otherwise,} \end{cases}$$

$$\hat{Q}_j^{(2)}(c) = \begin{cases} [\bigotimes_{i \in \mathcal{I}_j} \hat{Q}_i^{(1)}](c) & \text{, if } c \leq C_j - d_{j,k} \text{ and } j \in \mathcal{R}_u, \\ Q_j^{(2)}(c) & \text{, if } j \notin \mathcal{R}_u, \\ 0 & \text{, otherwise,} \end{cases}$$

$$\hat{Q}_J^{(3)}(c) = \begin{cases} [\bigotimes_{i \in \mathcal{I}_J} \hat{Q}_i^{(2)}](c) & \text{, if } c \leq C_J - d_{J,k}, \\ 0 & \text{, otherwise.} \end{cases}$$

The truncated state probabilities at the common link for the blocking and admissible states, $\hat{Q}_J^{(3)}(c)$ and $Q_J^{(3)}(c)$ respectively, are the joint state probabilities of all links in the network, with the capacity restrictions taken into account. The state probabilities are therefore the state probabilities of the network and inserting them into equation (4.2) gives the end-to-end blocking probability for class-k traffic and user u in the access network,

$$B_{(k,u)}^t = 1 - \frac{\sum_{c=0}^{C_J - d_{J,k}} \hat{Q}_J^{(3)}(c)}{\sum_{c=0}^{C_J} Q_J^{(3)}(c)} = \frac{\sum_{c=C_J - d_{J,k}+1}^{C_J} \hat{Q}_J^{(3)}(c)}{\sum_{c=0}^{C_J} Q_J^{(3)}(c)}.$$

In the previous example, the capacity needed to transport class-k traffic depended on the number of calls in the network. The capacity required to transport two class-k calls was double the capacity required to transport one call of the same traffic class, as is indicated by the constraint in e.g. equation (4.1). For multicast traffic, the problem and calculation of blocking probabilities are different, since there are only two possible states, idle or active.

4.6 The Reduced Load Approximation (RLA)

The RLA is an approximate method, where the blocking probabilities are calculated by assuming that blocking probabilities in each network link is independent. The traffic offered to the link in question is thinned by the probability

that the traffic is blocked on the other links of the route. Under the assumption of independence, the probability that blocking does not occur in the network is simply the product of the individual probabilities for each link. The RLA gives rise to a set of fixed-point equations $T(L) = L$, whose solution L is the approximate blocking probability. The mathematics of the RLA-algorithm will be presented for a singleservice network with Poisson arrivals using the Erlang formula (equation 3.1). The derivations can also be found in [15].

4.6.1 Singleservice Network

The blocking probability of class-k traffic in a singleservice link j with capacity C_j and traffic intensity a_k is

$$E(\sum_{k=1}^{K} a_k, C_j), \tag{4.9}$$

where $E(\cdot, \cdot)$ is the Erlang formula defined in equation (3.1). The set of links on the route of traffic class k is $\mathcal{R}_k$. When blocking occurs, the traffic intensity offered to the link j, $\sum_{k=1}^{K} a_k$, is thinned due to blocking

$$L_j = E(\sum_{k=1}^{K} a_k t_k(j), C_j), \tag{4.10}$$

where $t_k(j)$ denotes the probability that the traffic is not blocked in the other links on the route: links belonging to $\mathcal{R}_k - \{j\}$. Under the independence assumption, this is

$$t_k(j) = \prod_{i \in \mathcal{R}_k - \{j\}} (1 - L_i). \tag{4.11}$$

Combining equations (4.10) and (4.11) gives the fixed-point equation

$$L_j = E(\sum_{k=1}^{K} a_k \prod_{i \in \mathcal{R}_k - \{j\}} (1 - L_i), C_j), \quad j = 1, ..., J, \tag{4.12}$$

whose solution is the approximate blocking on all the links j of the network.

Under the independence assumption, the total blocking probability of class-k traffic is

$$B_k = 1 - \prod_{i \in \mathcal{R}_k} (1 - L_i). \tag{4.13}$$

4.6.2 Multiservice Network

For a multiservice network, the blocking probabilities are calculated under the same assumptions. However, the Erlang formula cannot be used, as the capacity requirements of the traffic classes differ. See [15] for the derivation of the RLA fixed-point equation for a multiservice network.

Chapter 5

Multicast Traffic Models

The advantage of using multicast is its bandwidth saving nature. The total bandwidth needed for transmission is constant and does not depend on the number of users requesting the transmission. After reviewing some related work on multicast models in section 5.1, two multicast models [5] and [10] are studied in depth. The model by Chan and Geraniotis [5] presented in section 5.2 is a multipoint-to-multipoint model for a network with subscriptions arriving from single users. Chan and Geraniotis give a closed form expression for the time blocking probability in the network, but use RLA for numerical calculations. The model by Karvo et al. [10] reviewed in section 5.3 is a point-to-multipoint model for a network, with subscriptions arriving from an infinite user population. Karvo et al. give exact solutions for blocking probabilities for the special case of all but one link in the network having infinite capacity. For the network with more than one link having finite capacity, they use RLA as well [11]. In chapter 6 the model by Karvo et al. is extended and an exact algorithm for calculating call and channel blocking probabilities in a network with an arbitrary number of finite links is given.

5.1 Related Work on Multicast Routing, Admission Control, and Blocking Probability

Multicast networks and their modeling are topics that have drawn much attention in the past few years. Most of the research seems to tackle multicast routing questions and the adaptation of ATM to multicasting. Diot et al. give a good overview of multicast and multipoint communications in [6]. Semeria and Maufer have written the "Introduction to IP Multicast Routing located on the Internet [16]. The paper deals with routing algorithms and their implementation in IP. The paper by Bagwell, McDearman, and Marlow [3] compares ATM-multicast to IP-multicast and serves as a good introduction to some problems that faced ATM-multicast a few years ago. Guarene et al. [8] propose two mechanisms to overcome these difficulties. These articles serve as a good introduction to IP and ATM-multicast. New multicast routing protocols are also being developed constantly, and a number of articles can be found on the subject.

Blocking probabilities and admission control problems have also been investigated. However, a variety of different multicast models and blocking probability models are used. The papers can be divided into a few main categories. Mainly, the research has been focused on blocking probabilities in multicast capable switches. Kim [13] studies blocking probabilities in a multirate multicast switch. Three stage switches are studied by Yang and Wang [22] and Listanti and Veltri [14]. Stasiak and Zwierzykowski [19] study blocking in an ATM node with multicast switching nodes carrying different multi-rate traffic, unicast and multicast, using Kaufman-Roberts recursion and Reduced-Load Approximation. Admission control algorithms are studied in [17].

The paper by Almeroth and Ammar [1] investigates multicast group behavior in the MBone. From this data, they conclude that interarrival times are exponentially distributed while group membership duration times are exponentially distributed for small networks and Zipf distributed for larger networks. The study of intersession data suggests that simultaneous sessions, where a user subscribes to more than one channel, occur, but not frequently.

The multicast networks and models in this paper originate from the telecommunication environment. Multicast is also used in parallel computing appli-

cations, such as the parallel algorithm for the Fast Fourier Transform (FFT) and write update/invalidate in directory based cache coherence protocols. The paper by Yang [21] discusses multicast for parallel computing applications.

Two models deserve a closer look into. The core of this study is based on the model by Karvo et al. [10] presented in section 5.3. A similar, but more general model has been given by Chan and Geraniotis [5] and is presented in the following section.

5.2 Layered Video Transmission Model for a Multicast Network

Chan and Geraniotis [5] explore the tradeoff between blocking and dropping in multicast networks. The model is based on two main characteristics of video transmission in a multicast network: receivers share part of the connections and a source may transmit video signals at the same time to a group of receivers with different receiving capabilities and/or requirements. The model is therefore based on subband coding. Subband coding is used to encode a signal into several layers each containing a part of the information. The lowest layer contains essential information for transmitting a low quality version of the video signal. Higher layers add information to the signal. As long as all the lower level signals are received, a higher layer adds to the resolution of the signal.

Each node in the network may be a source and/or a user, resulting in a multipoint-to-multipoint model. The connection between a user node and a source node is called a physical path $p \in \mathcal{P}$. The users are divided into classes $t \in \mathcal{T}$ according to the level of video signal they request. Class 1 is the highest class of service and therefore for class $t < \tau$, t is said to be a superset of τ. B_t is the bandwidth required by a class t video signal and $s \in \mathcal{S}$ is the video source. The triplet physical path, class, and video source define the logical path (p, s, t). In a multicast connection, the state of the logical path is $n_{pst} \in \{0, 1\}$. The user model is a finite population model, where each user, equivalent to a logical path, is modeled by a two state Markov model, with transition rate a_{pst} from the off-state to the on-state and rate b_{pst} from the on-state to the off-state. The activity factor defined as the probability

$P(n_{pst} = 1)$ that the logical path is active is

$$\rho_{pst} = \frac{a_{pst}}{a_{pst} + b_{pst}}.$$

The states of the logical path are expressed with the help of three vectors $\mathbf{n}_{ps} = (n_{pst}, t \in \mathcal{T})$, $\mathbf{n}_s = (n_{ps}, p \in \mathcal{P})$, and $\mathbf{n} = (n_s, s \in \mathcal{S})$. The set of network links is denoted by $\mathcal{L}$.

Chan and Geraniotis also define the state i_s of the source that determines the rate at which the video signal is transmitted. The model of the source is a $(M+1)$-state continuous-time Markov chain, given in an earlier paper [4] by Chan and Geraniotis. This state is however used in the calculation of dropping probabilities and is not needed for the formulation of blocking probabilities.

The steady state probability is given by

$$\begin{aligned}
P(\mathbf{n}) &= \frac{1}{G^{\mathbf{n}}(\mathbf{c})} \prod_{s=1}^{|\mathcal{S}|} \prod_{p \in \mathcal{P}} \prod_{t=1}^{|\mathcal{T}|} \Big(\frac{a_{pst}}{b_{pst}}\Big)^{n_{pst}}, \\
G^{\mathbf{n}}(\mathbf{c}) &= \sum_{\Omega^n(\mathbf{c})} \prod_{\zeta=1}^{|\mathcal{S}|} \prod_{q \in \mathcal{P}} \prod_{\tau=1}^{|\mathcal{T}|} \Big(\frac{a_{q\zeta\tau}}{b_{q\zeta\tau}}\Big)^{n_{q\zeta\tau}}, \\
\Omega^n(\mathbf{c}) &= \Big\{\mathbf{n} \mid 0 \le \sum_{\zeta=1}^{|\mathcal{S}|} \max_{\substack{q \in \mathcal{P}_l \\ \tau \in \mathcal{T}}} \{B_{q\zeta\tau} \cdot I(n_{q\zeta\tau} = 1)\} \le c_l, \forall l \in \mathcal{L}\Big\}.
\end{aligned}$$

A request to set up a logical path (p^*, s^*, t^*) is rejected if the lower levels are off and there is not enough capacity on the link to turn the logical path on. In other words,

$$\begin{cases}
n_{p^* s^* t} = 0 & , \forall t < t^* \\
\sum_{s=1}^{|\mathcal{S}|} \max_{\substack{p \in \mathcal{P}_l \\ t \in \mathcal{T}}} \{B_{pst} \cdot I(n'_{pst} = 1)\} > c_l & , \text{for some } l \in \mathcal{L},
\end{cases}$$

where

$$n'_{pst} = \begin{cases} 1 & , \text{if } s = s^*, p = p^*, t = t^* \\ n_{pst} & , \text{otherwise.} \end{cases}$$

All $\mathbf{n}$ satisfying the above conditions belong to the set $\Omega^*(\mathbf{c})$. The time blocking probability of the logical path (p^*, s^*, t^*) is given by

$$PB_{p^* s^* t^*} = \sum_{\Omega^*(\mathbf{c})} P(\mathbf{n}).$$

The authors do not use the model derived to calculate the blocking probabilities of the multicast network, as prohibitive computational effects would be required. Instead, they use the Reduced Load Approximation.

5.2.1 The RLA Algorithm Used by Chan and Geraniotis

Let X denote the bandwidth used by source $s \in \mathcal{S}_l$ on link l, then the probability that B_t units of bandwidth are allocated is

$$P(X = B_t) = \Big\{ \prod_{q \in \mathcal{P}_l} \prod_{\tau < t} (1 - \rho_{qst}) \Big\} \cdot \Big\{ 1 - \prod_{q \in \mathcal{P}_l} (1 - \rho_{qst}) \Big\}. \tag{5.1}$$

The bandwidth is allocated, if all logical paths requiring a higher class of service are idle and at least one logical path on the physical path $q \in \mathcal{P}_l$ with the same class of service is active. The bandwidth along link l, used by all other sources excluding source s, is defined in a similar fashion and is denoted by Y. Then the joint probability, by assumption of independency, is $P(X, Y) = P(X)P(Y)$.

The blocking probability is then

$$LB_{ls}^{(t)} = \frac{\sum_{X=0}^{B_t-1} \sum_{Y=c_l-B_t+1}^{c_l-X} P(X,Y)}{\sum_{X=0}^{B_1} \sum_{Y=0}^{c_l-X} P(X,Y)}.$$

The reduced load is the probability that the logical path is active and not blocked,

$$\rho'_{pst} = \rho_{pst} \cdot \prod_{j \in p, j \neq l} (1 - LB_{js}^{(t)}).$$

In the numerator the sum $\sum_{X=0}^{B_t-1}$ is due to the requirement that the class of service t is off. This corresponds to other sources requiring bandwidth more than $c_l - B_t$, hence the sum $\sum_{Y=c_l-B_t-1}^{c_l-X}$. In the denominator all possible bandwidth allocations in respect to the capacity c_l available on the link are considered. The resulting blocking probability is the time blocking probability.

5.3 A Multicast Traffic Model for a Single Link

In this section, we review the point-to-multipoint model for a dynamic multicast network with an infinite user population presented by Karvo et al. in [10].

A single source offers a variety of channels, belonging to the set $\mathcal{I}$. Subscriptions to channel $i \in \mathcal{I} = \{1, ..., I\}$ arrive according to a Poisson process with intensity λ_i. The channel i is chosen independently according to a preference distribution α_i,

$$\alpha_i = \frac{p(1-p)^{i-1}}{1-(1-p)^I}. \tag{5.2}$$

The offered traffic intensity for a multicast channel i is then $a_i = \lambda_i/\mu_i = \alpha_i\lambda/\mu_i$, where $1/\mu_i$ is the average holding time of channel i and is generally distributed. It is shown in [10] that in a multicast network with all links having infinite capacity, the distribution of the number of users simultaneously connected to channel i in a link is the queue length distribution of a $M/G/\infty$ queue with offered traffic intensity a_i. The probability of having channel i on is therefore the probability that at least one user subscribes to channel i,

$$p_i = 1 - e^{-a_i} = \frac{T_{i,\text{on}}^{(\infty)}}{T_{i,\text{on}}^{(\infty)} + T_{i,\text{off}}^{(\infty)}}. \tag{5.3}$$

The on and off times of a channel are distributed as the busy and idle periods, respectively, of a M/G/∞ queue, with mean

$$T_{i,\text{on}}^{(\infty)} = \frac{e^{a_i} - 1}{\lambda_i},$$

$$T_{i,\text{off}}^{(\infty)} = {\lambda_i}^{-1}.$$

Multicast traffic is characterized by its on/off nature. All calls that arrive when the channel is turned on, are accepted and transferred with no increase in the required capacity. A call that arrives when the channel is off in an infinite link, turns the channel on and increases the occupied capacity of the link.

When the link capacity is restricted, blocking occurs. To calculate these blocking probabilities, Karvo et al. [10] have considered the one link case, where other links in the network have infinite capacity. The blocking probability is divided into three types: channel blocking B_i^c, call blocking b_i^c and time blocking b_i^t. The channel blocking probability is the probability that an attempt to turn channel i on fails due to the capacity restriction of the link. Channel blocking can therefore occur only, if the channel is in the off state. Call blocking occurs when a user is not able to subscribe to channel i. Call and channel blocking are different as a subscription to a channel is always accepted if the channel is active. The time blocking probability B_i^t was originally, in the paper by Karvo et al., defined as the probability that at least $C - c_i + 1$ capacity

units of the link are occupied at an arbitrary instant. The probability that at least $C - c_i + 1$ capacity units are occupied when the channel is off is the actual time blocking probability in a multicast link and is denoted by b_i^t. The capacity requirement of channel i is denoted by c_i.

Using the above definitions the call blocking probability of channel i is

$$b_i^c = \frac{\lambda_i T_{i,\text{off}} - 1}{\lambda_i T_{i,\text{on}} + \lambda_i T_{i,\text{off}}}. \tag{5.4}$$

The mean number of failed attempts to subscribe to the channel during an on/off-cycle is $\lambda_i T_{i,\text{off}} - 1$, as the last call arriving during the off-period is accepted. The time the channel is in the off state increases, as blocking occurs. The frequency of accepted calls in the off state is $\lambda_i(1 - B_i^c)$, B_i^c denoting the channel blocking probability of channel i. The off-period of channel i is thus

$$T_{i,\text{off}} = \frac{1}{\lambda_i(1 - B_i^c)}. \tag{5.5}$$

Once the channel is turned on, all calls are accepted and no blocking occurs. The average time the channel is on is therefore the same as the average on-period for an infinite link

$$T_{i,\text{on}} = T_{i,\text{on}}^{(\infty)} = \frac{e^{a_i} - 1}{\lambda_i}. \tag{5.6}$$

Equation (5.6) can also be expressed as a function of p_i equation (5.3). The probability that channel i is active in an infinite capacity system is

$$T_{i,\text{on}} = \frac{p_i}{(1 - p_i)\lambda_i}. \tag{5.7}$$

By combining equations (5.4), (5.5), and (5.7), the call blocking probability can be written as,

$$b_i^c = \frac{B_i^c}{(1 - B_c^i)(e^{a_i} - 1) + 1} = \frac{(1 - p_i)B_i^c}{1 - p_i B_i^c}. \tag{5.8}$$

The equation confirms that channel blocking probability equals call blocking probability upon condition that the channel is off, as in the finite system the probability that the channel is off is

$$\frac{T_{i,\text{off}}}{T_{i,\text{on}} + T_{i,\text{off}}} = \frac{\frac{1}{\lambda_i(1-B_i^c)}}{\frac{p_i}{(1-p_i)\lambda_i} + \frac{1}{\lambda_i(1-B_i^c)}} = \frac{1 - p_i}{1 - p_i B_i^c}. \tag{5.9}$$

Karvo et al. [10] proceed in deriving an expression for the channel blocking probability B_i^c of channel i. They observe that channel blocking in the one

finite link case is equal to the call blocking in a certain generalized Engset system, that is a $M/G/C/C/I$ queue. Thus the channel blocking probability B_i^c is equal to the time blocking probability $B_i^{t(i)}$ in a system where channel i is removed. The channel blocking probability is

$$B_i^c = B_i^{t(i)} = \frac{\sum_{j=C-c_i+1}^{C} {\pi_j}^{(i)}}{\sum_{j=0}^{C} {\pi_j}^{(i)}}, \tag{5.10}$$

where $\pi_j^{(i)}$ is the link occupancy distribution of a system with channel i removed.

An alternative way of calculating the call blocking probability is calculating the time blocking probability of the whole system,

$$b_i^t = \frac{\sum_{j=C-c_i+1}^{C} {\pi_j}^{(x_i=0)}}{\sum_{j=0}^{C} \pi_j}, \tag{5.11}$$

where π_j is the link occupancy distribution for an infinite system and $\pi_j^{(x_i=0)}$ is the link occupancy distribution restricted to the states with channel i off,

$$\pi_j^{(x_i=0)} = \sum_{\mathbf{x}\cdot\mathbf{c}=j, x_i=0} \pi(\mathbf{x}) = (1-p_i)\pi_j^{(i)}, \tag{5.12}$$

where $\mathbf{c} = (c_i, i \in \mathcal{I})$. The last expression stems from the product form of the link occupancy distribution π_j. Similarly, the link occupancy distribution π_j in terms of the link occupancy distribution $\pi_j^{(i)}$ is

$$\pi_j = (1-p_i)\pi_j^{(i)} + p_i \pi_{j-c_i}^{(i)}. \tag{5.13}$$

Because of Poisson arrivals, the time blocking and the call blocking probabilities for the whole system are equal. This can also be verified by investigating equations (5.8) through (5.11) and rewriting the expressions for $\pi_j^{(x_i=0)}$ and π_j with the help of equations (5.12) and (5.13),

$$\begin{aligned}
b_i^t &= \frac{\sum_{j=C-c_i+1}^{C} {\pi_j}^{(x_i=0)}}{\sum_{j=0}^{C} \pi_j} \\
&= \frac{(1-p_i)\sum_{j=C-c_i+1}^{C} {\pi_j}^{(i)}}{(1-p_i)\sum_{j=0}^{C} \pi_j^{(i)} + p_i \sum_{j=0}^{C-c_i} \pi_j^{(i)}} \\
&= \frac{(1-p_i)\sum_{j=C-c_i+1}^{C} {\pi_j}^{(i)}}{(1-p_i)\sum_{j=0}^{C} \pi_j^{(i)} + p_i(\sum_{j=0}^{C} \pi_j^{(i)} - \sum_{j=C-c_i-1}^{C} \pi_j^{(i)})} \\
&= \frac{(1-p_i)B_i^c}{1-p_i+p_i(1-B_i^c)} = \frac{(1-p_i)B_i^c}{1-p_iB_i^c} = b_i^c.
\end{aligned}$$

Equation (5.11) is computationally more efficient when calculating blocking probabilities for different traffic classes, as the denominator does not have to be calculated anew for each traffic class.

5.3.1 The RLA Algorithm Used by Karvo et al.

The calculation of the end-to-end blocking probabilities for a multicast network was done in [11] using the RLA-algorithm. Instead of the Erlang formula, equation (5.8) is used for the calculation of the blocking probabilities. The fixed point equation is now

$$L_j^i = b_i^c(\mathbf{s}_j, \mathbf{c}, C_j) = \frac{(1 - p_i(\mathbf{s}_j))B_i^c(\mathbf{s}_j, \mathbf{c}, C_j)}{1 - p_i(\mathbf{s}_j)B_i^c(\mathbf{s}_j, \mathbf{c}, C_j)}, \text{ for } i = 1, ..., I \text{ and } j = 1, ..., J, \tag{5.14}$$

where $\mathbf{c} = (c_i, ..., c_I)$ is the capacity requirement vector, C_j the capacity of link j, and $\mathbf{s}_j$ is the thinned traffic intensity vector $\mathbf{s}_j = (s_{j,1}, ..., .s_{j,I})$,

$$s_{j,i} = \sum_{u \in \mathcal{U}_j} a_{u,i} \prod_{k \in \mathcal{R}_u - \{j\}} (1 - L_k^i). \tag{5.15}$$

Here $\mathcal{U}_j$ is the set of users downstream of link j, $a_{u,i}$ is the traffic intensity offered by user population u for channel i, and $\mathcal{R}_u$ is the set of links on the route from the user u to the root.

Writing equation (5.14) in matrix form gives a fixed point equation $T(\mathbf{L}) = \mathbf{L}$ for $\mathbf{L} = (L_j^i)$. For a multiservice network, with different capacity requirements for different traffic classes, the solution is not always unique see [15]. Solving the fixed point equation gives the call blocking probability for each individual link in the network. Under the independence assumption, the blocking of the traffic class denoted by the user u and channel i is

$$b_{u,i}^c = 1 - \prod_{k \in \mathcal{R}_u} (1 - L_k^i).$$

Chapter 6

A New Multicast Network Model

The papers presented in the previous chapter gave approximate algorithms for calculating blocking probabilities in a network. Although Chan and Geraniotis were able to give a closed form expression for time blocking probabilities in a multipoint-to-multipoint network, prohibitive computational effects required the use of RLA.

In this chapter, the point-to-multipoint model by Karvo et al., with infinite user populations subscribing to the network, is used to formulate an exact algorithm for calculating blocking probabilities in a network with more than one link having finite capacity. Including only one source in the model, allows for the formulation of the algorithm. The algorithm is based on the well-known algorithm for calculating blocking probabilities in hierarchical multiservice access networks presented in section 4.5. The resource sharing of multicast connections requires the modification of the algorithm by using a new type of convolution, the OR-convolution. The algorithm applies to tree-type networks with one source node offering multiservice multicast traffic.

6.1 Notation

The notation used throughout the rest of this study is as follows. The set of all links is denoted by $\mathcal{J}$. Let $\mathcal{U} \subset \mathcal{J}$ denote the set of leaf links. The leaf link and user population behind the leaf link is denoted by $u \in \mathcal{U} = \{1, ..., U\}$. The set

of links on the route from leaf link u to the source is denoted by $\mathcal{R}_u$. The set of links downstream link $j \in \mathcal{J}$ including link j is denoted by $\mathcal{M}_j$, while the set of downstream links terminating at link $j \in \mathcal{J}$ are denoted by $\mathcal{N}_j$. The set of user populations downstream link j is denoted by $\mathcal{U}_j$. The set of channels offered by the source is $\mathcal{I}$, with channel $i = 1, .., I$. Let $\mathbf{d} = \{d_i; i \in \mathcal{I}\}$, where d_i is the capacity requirement of channel i. Here it is assumed that the capacity requirements depend only on the channel, but link dependencies could also be included into the model. The capacity of the link j is denoted by C_j. The different sets are shown in figure 6.1.

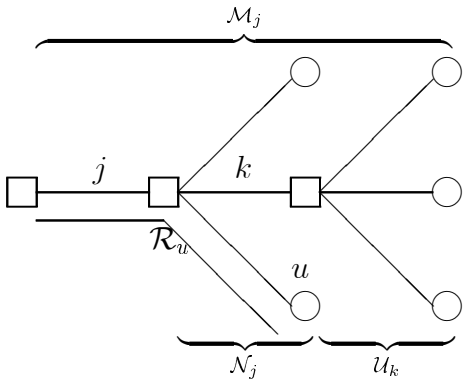

Figure 6.1: An example network to show the notation used.

6.2 Network with Infinite Link Capacities

Consider a network with all links having infinite capacity. Subscriptions to channel i behind leaf link u arrive from an infinite user population as from a Poisson process with intensity $\lambda_{u,i} = \alpha_i \lambda_u$, where α_i is generated from a preference distribution for channel i and λ_u is the arrival intensity for user population u. The channel holding time is assumed to be generally distributed with mean $1/\mu_i$. The traffic intensity denoted by $a_{u,i}$ is then, $a_{u,i} = \alpha_i \lambda_u / \mu_i$. Let the pair $(u, i) \in U \times I$ denote a traffic class also called a connection. The connection state, which may be off or on, is denoted by $X_{u,i} \in \{0, 1\}$. The state probability for a connection, according to the $M/G/\infty$ model, is

$$\pi_{u,i}(x_{u,i}) = P(X_{u,i} = x_{u,i}) = (p_{u,i})^{x_{u,i}}(1 - p_{u,i})^{1-x_{u,i}},$$

where $p_{u,i} = 1 - e^{-a_{u,i}}$.

In the infinite link capacity case, all connections are independent of each other.

For leaf link u, the state probability has a product form and is

$$\pi_u(\mathbf{x}_u) = P(\mathbf{X}_u = \mathbf{x}_u) = \prod_{i \in \mathcal{I}} \pi_{u,i}(x_{u,i}), \tag{6.1}$$

where $\mathbf{X}_u = (X_{u,i}; i \in \mathcal{I}) \in \mathcal{S}$ is the state vector for the leaf link, and $\mathcal{S} = \{0,1\}^I$ denotes the link state space.

The leaf link states jointly define the network state $\mathbf{X}$,

$$\mathbf{X} = (\mathbf{X}_u; u \in \mathcal{U}) = (X_{u,i}; u \in \mathcal{U}, i \in \mathcal{I}) \in \Omega, \tag{6.2}$$

where $\Omega = \{0,1\}^{U \times I}$ denotes the network state space. For the whole network, the state probability is

$$\pi(\mathbf{x}) = P(\mathbf{X} = \mathbf{x}) = \prod_{u \in \mathcal{U}} \pi_u(\mathbf{x}_u),$$

as each user population is independent of each other.

6.3 OR-Convolution

The leaf link state distributions jointly define the network state distribution, as was shown in the previous section. In order to calculate the link state distributions in a tree-structured network a convolution operation is needed. The resource sharing characteristic of multicast traffic requires a new type of convolution, the OR-convolution. Consider two downstream links $s, t \in \mathcal{N}_v$ terminating at link v, where $s, t, v \in \mathcal{J}$. Channel i is idle in link v if it is idle in both links s and t and active in all other cases, which is equivalent to the binary OR-operation. In other words, for $\mathbf{y}_s, \mathbf{y}_t \in \mathcal{S}$

$$\mathbf{y}_v = \mathbf{y}_s \oplus \mathbf{y}_t \in \mathcal{S}, \tag{6.3}$$

where the vector operator $\oplus$ denotes the OR-operation taken componentwise. The OR-convolution, denoted by $\otimes$, is then the operation,

$$[f_s \otimes f_t](\mathbf{y}_v) = \sum_{\mathbf{y}_s \oplus \mathbf{y}_t = \mathbf{y}_v} f_s(\mathbf{y}_s) f_t(\mathbf{y}_t)$$

defined for any distributions f_s and f_t.

In a multicast link, the link state depends on the user states downstream the link. If a channel is idle in all links downstream link j it is off in link j and in

all other cases the channel is active. The OR-operation on the network state gives the link state $\mathbf{Y}_j = (Y_{j,i}; i \in \mathcal{I}) \in \mathcal{S}, j \in \mathcal{J}$ as a function of the network state,

$$\mathbf{Y}_j = \mathbf{g}_j(\mathbf{X}) = \bigoplus_{k \in \mathcal{U}_j} \mathbf{X}_k.$$

Similarly, the OR-convolution on the network state distribution gives the link state distribution. Thus, the state probability, denoted by $\sigma_j(\mathbf{y}_j)$, for $\mathbf{y}_j \in \mathcal{S}$, is equal to

$$\sigma_j(\mathbf{y}_j) = P(\mathbf{Y}_j = \mathbf{y}_j) = [\bigotimes_{k \in \mathcal{U}_j} \pi_k](\mathbf{y}_j) = \begin{cases} \pi_j(\mathbf{y}_j) & , \text{ if } j \in \mathcal{U} \\ [\bigotimes_{k \in \mathcal{N}_j} \sigma_k](\mathbf{y}_j) & , \text{ otherwise .} \end{cases}$$

When $\mathbf{X} = \mathbf{x}$ the occupied capacity on the link j is $\mathbf{d} \cdot \mathbf{g}_j(\mathbf{x})$.

6.4 Blocking Probabilities in a Network with Finite Link Capacities

When the capacities of one or more links in the network are finite, the state spaces defined above are truncated according to the capacity restrictions. The network state $\mathbf{X}$ defined in equation (6.2) is replaced by the truncated network state $\tilde{\mathbf{X}} \in \tilde{\Omega}$, where $\tilde{\Omega}$ denotes the truncated state space

$$\tilde{\Omega} = \{\mathbf{x} \in \Omega \mid \mathbf{d} \cdot \mathbf{g}_j(\mathbf{x}) \leq C_j, \forall j \in \mathcal{J}\}.$$

The insensitivity [12] and truncation principles [9] apply for this product form network, and for the truncated system the state probabilities of the network differ only by the normalization constant $G(\tilde{\Omega}) = \sum_{\mathbf{x} \in \tilde{\Omega}} \pi(\mathbf{x})$. The state probabilities of the truncated system are therefore

$$\tilde{\pi}(\mathbf{x}) = P(\tilde{\mathbf{X}} = \mathbf{x}) = \frac{\pi(\mathbf{x})}{G(\tilde{\Omega})}, \text{ for } \mathbf{x} \in \tilde{\Omega}.$$

When the capacity on the links is finite, blocking occurs. Due to Poisson arrivals, the call blocking probability is equal to the time blocking probability of the system. A call in traffic class (u, i) is blocked if there is not enough capacity in the network to set up the connection. Note that, once the channel is active on all links belonging to the route $\mathcal{R}_u$ of user population u, no extra

capacity is required for a new connection. Another truncated set $\tilde{\Omega}_{u,i} \subset \tilde{\Omega}$ with a tighter capacity restriction for links with channel i idle is defined as,

$$\tilde{\Omega}_{u,i} = \{\mathbf{x} \in \Omega \mid \mathbf{d} \cdot \Big(\mathbf{g}_j(\mathbf{x}) \oplus (\mathbf{e}_i 1_{j \in \mathcal{R}_u})\Big) \leq C_j, \forall j \in \mathcal{J}\},$$

where $\mathbf{e}_i$ is the I-dimensional vector consisting of only zeroes except for a one in the ith component and $1_{j \in \mathcal{R}_u}$ is the indicator function equal to one for $j \in \mathcal{R}_u$ and zero otherwise. This set defines the states where blocking does not occur when user u requests a connection to channel i. The call blocking probability b_i^c for traffic class (u, i) is thus,

$$b_{u,i}^c = 1 - P(\tilde{\mathbf{X}} \in \tilde{\Omega}_{u,i}) = 1 - \frac{G(\tilde{\Omega}_{u,i})}{G(\tilde{\Omega})}. \tag{6.4}$$

This approach requires calculating two sets of state probabilities: the set of non-blocking states appearing in the numerator and the set of allowed states appearing in the denominator of equation (6.4).

A multicast network is a tree-type network, and much of the theory in calculating blocking probabilities in hierarchical multiservice access networks [15] can be used to formulate the end-to-end blocking probability in a multicast network as well.

6.5 The Algorithm

As in the case of access networks, the blocking probability can be calculated by recursively convolving the state probabilities of individual links from the leaf links to the origin link. At each step, the state probabilities are truncated according to the capacity restriction of the link.

In order to calculate the denominator of equation (6.4), a new subset $\tilde{\mathcal{S}}_j$ of set $\mathcal{S}$ is defined,

$$\tilde{\mathcal{S}}_j = \{\mathbf{y} \in \mathcal{S} \mid \mathbf{d} \cdot \mathbf{y} \leq C_j\}, \text{ for } j \in \mathcal{J}.$$

The corresponding truncation operator acting on any distribution f is

$$T_j f(\mathbf{y}) = \begin{cases} f(\mathbf{y}) & , \text{ if } \mathbf{y} \in \tilde{\mathcal{S}}_j \\ 0 & , \text{ otherwise.} \end{cases} \tag{6.5}$$

Let

$$Q_j(\mathbf{y}_j) = P(\mathbf{Y}_j = \mathbf{y}_j; \mathbf{Y}_k \in \tilde{\mathcal{S}}_k, \forall k \in \mathcal{M}_j), \text{ for } \mathbf{y}_j \in \mathcal{S}. \tag{6.6}$$

It follows that the $Q_j(\mathbf{y})$'s can be calculated recursively,

$$Q_j(\mathbf{y}) = \begin{cases} T_j \pi_j(\mathbf{y}) & , \text{ if } j \in \mathcal{U} \\ T_j[\bigotimes_{k \in \mathcal{N}_j} Q_k](\mathbf{y}) & , \text{ otherwise.} \end{cases}$$

Note that, if the capacity constraint of link $j \in \mathcal{M}_j$ is relaxed, then the branches terminating at link j are independent, and the jointly requested channel state can be obtained by the OR-convolution. The effect of the finite capacity C_j of link j is then just the truncation of the distribution to the states for which the requested capacity is no more than C_j.

The state sum $G(\tilde{\Omega})$ needed to calculate the blocking probability in equation (6.4) is equal to

$$G(\tilde{\Omega}) = \sum_{\mathbf{y} \in \mathcal{S}} Q_J(\mathbf{y}),$$

where Q_J is the probability (6.6) related to the common link $j = J$.

Similarly for the numerator of equation (6.4), let $\tilde{\mathcal{S}}_j^{u,i} \subset \tilde{\mathcal{S}}_j$ be defined as the set of states on link j that do not prevent user u from connecting to multicast channel i. In other words

$$\tilde{\mathcal{S}}_j^{u,i} = \{\mathbf{y} \in \mathcal{S} \mid \mathbf{d} \cdot \left(\mathbf{y} \oplus (\mathbf{e}_i 1_{j \in \mathcal{R}_u})\right) \leq C_j\}, \text{ for } j \in \mathcal{J}, i \in \mathcal{I}.$$

The truncation operator is then

$$T_j^{u,i} f(\mathbf{y}) = \begin{cases} f(\mathbf{y}) & , \text{ if } \mathbf{y} \in \tilde{\mathcal{S}}_j^{u,i} \\ 0 & , \text{ otherwise.} \end{cases} \tag{6.7}$$

The non-blocking probability of link j is

$$Q_j^{u,i}(\mathbf{y}_j) = P(\mathbf{Y}_j = \mathbf{y}_j; \mathbf{Y}_k \in \tilde{\mathcal{S}}_j^{u,i}, \forall k \in \mathcal{M}_j), \text{ for } \mathbf{y}_j \in \mathcal{S}. \tag{6.8}$$

Similarly, as above, it follows that

$$Q_j^{u,i}(\mathbf{y}) = \begin{cases} T_j^{u,i} \pi_j(\mathbf{y}) & , \text{ if } j \in \mathcal{U} \\ T_j^{u,i}[\bigotimes_{k \in \mathcal{N}_j} Q_k^{u,i}](\mathbf{y}) & , \text{ otherwise.} \end{cases}$$

The state sum in the numerator of equation (6.4) is then

$$G(\tilde{\Omega}_{u,i}) = \sum_{\mathbf{y} \in \mathcal{S}} Q_J^{u,i}(\mathbf{y}),$$

where $Q_J^{u,i}$ is the probability (6.8) related to the common link $j = J$.

The blocking probability in equation (6.4) is therefore

$$b_{u,i}^c = 1 - \frac{\sum_{\mathbf{y}\in\mathcal{S}} Q_J^{u,i}(\mathbf{y})}{\sum_{\mathbf{y}\in\mathcal{S}} Q_J(\mathbf{y})}.$$

The single link approach by Karvo et al. is a special case of the network formulation presented, hence the same results can be obtained using the network algorithm. Note also that the algorithm calculates time blocking probabilities and is therefore applicable to systems where call or channel blocking probabilities can be expressed in terms of time blocking.

The complexity of the algorithm increases exponentially with the number of channels, as the number of states in the distributions to be convolved is 2^I. Therefore the use of RLA as a computationally simpler method is studied.

6.6 The Appropriateness of Using RLA in Multicast Networks

As the number of channels in a multicast network increases, the exact computation of the blocking probabilities becomes impossible. The RLA-algorithm is a method that can be applied to a multicast network with dynamic multicast connections. The RLA assumes that the traffic, which is thinned due to blocking in the links of the network, is independent and from a Poisson process. This assumption is not exactly satisfied, as the blocking depends on the traffic intensity offered to the link. Because at this moment no other approximation method exists, it is worthwhile to study the performance of RLA. In [11], the accuracy of the RLA-algorithm was studied by comparing it to simulation results with $I = 30$. In this section, the RLA-algorithm presented in section 5.3.1 will be compared to exact results, but only for a network that offers $I = 8$ channels.

6.6.1 Results

Comparisons were made between the exact solution and the RLA-algorithm. The network used is shown in figure 6.2. The number of channels offered is

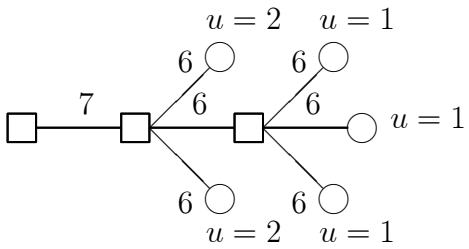

Figure 6.2: The example network used to compare the exact result with the result given by the RLA-algorithm.

eight. Each channel requires one unit of capacity. The common link in the network has a capacity of seven units. All other links have a capacity of six units. The blocking probabilities are calculated for the least used channel using a truncated geometric distribution for the channel preference, as defined in equation (5.2)

$$\alpha_i = \frac{p(1-p)^{i-1}}{1-(1-p)^I},$$

with parameter $p = 0.2$. The mean holding time is the same for all channels, $1/\mu = 1$. In addition, the arrival intensity is the same for both user populations, $\lambda_u = \lambda$ and consequently, the traffic intensity $a = \lambda/\mu$ is the same for both user populations.

The results are given in table 6.1. The comparison was also done for multiservice traffic, where the capacity requirement is one for odd channels and two for even channel numbers. The capacity of the common link was eleven units and those of the other links were nine units. The results are given in table 6.2.

The results confirm the comparisons made in [11]. The RLA-algorithm yields blocking probabilities of the same magnitude as the exact method. As a rule, RLA gives larger blocking values for both routes. For route 2, RLA gives very good results. This is because the route is very short, and the assumption of independence between the links is not violated severely.

Table 6.1: Call blocking probabilities for the network shown in figure 6.2.

	Route1 ($u = 1$)			Route2 ($u = 2$)		
a	Exact	RLA	error	Exact	RLA	error
1.0	0.0056	0.0064	14 %	0.0027	0.0028	4 %
1.1	0.0084	0.0098	17 %	0.0041	0.0044	7 %
1.2	0.0121	0.0141	17 %	0.0060	0.0064	8 %
1.3	0.0166	0.0195	17 %	0.0083	0.0090	8 %
1.4	0.0220	0.0260	18 %	0.0112	0.0121	8 %
1.5	0.0282	0.0336	19 %	0.0146	0.0157	8 %
2.0	0.0715	0.0868	21 %	0.0382	0.0416	9 %

Table 6.2: Call blocking probabilities for the network depicted in figure 6.2, with channel requirements $c_{odd} = 1$ and $c_{even} = 2$.

		Route1 ($u = 1$)			Route2 ($u = 2$)		
Channel	a	Exact	RLA	error	Exact	RLA	error
7	1.0	0.0051	0.0058	14 %	0.0019	0.0022	16 %
8	1.0	0.0127	0.0139	9 %	0.0028	0.0029	4 %
7	1.1	0.0074	0.0086	16 %	0.0029	0.0034	17 %
8	1.1	0.0179	0.0198	11 %	0.0042	0.0045	7 %
7	1.2	0.0103	0.0120	17 %	0.0042	0.0049	17 %
8	1.2	0.0243	0.0270	11 %	0.0062	0.0066	6 %
7	1.3	0.0138	0.0162	17 %	0.0058	0.0068	17 %
8	1.3	0.0318	0.0355	12 %	0.0086	0.0092	7 %
7	1.4	0.0179	0.0211	18 %	0.0077	0.0091	18 %
8	1.4	0.0403	0.0454	13 %	0.0116	0.0124	7 %
7	1.5	0.0226	0.0268	19 %	0.0100	0.0118	18 %
8	1.5	0.0499	0.0566	13 %	0.0151	0.0162	7 %
7	2.0	0.0536	0.0645	20 %	0.0255	0.0299	17 %
8	2.0	0.1101	0.1276	16 %	0.0400	0.0431	8 %

Chapter 7

A Finite User Population Model

In this chapter, a new finite user population model is formulated. First, it will be shown that the conditional distribution of mutually independent Poisson distributed variables X_i, with parameters $a_i = Np_i$, for $i = 0, ..., I$, given that $x_0 + ... + x_I = N$ is distributed as variables from a multinomial distribution with parameters p_i. This known property is used in order for the finite user population model and the infinite user population model to relate. After introducing the single user model in section 7.2, it will be shown in section 7.3 how this model can be generalized to a finite user population of size N. Furthermore, as N goes to infinity, the leaf link distributions for the finite user population of size N converge to those obtained by the infinite user population model presented in chapter 6. Numerical results on end-to-end channel blocking probabilities are presented in section 7.4.

7.1 Multinomial Distribution

The multicast traffic model derived in [10] and discussed in chapter 6 assumes an infinite user population. Under this assumption, the channel requests arrive according to independent Poisson processes with parameters $a_i, i \in \mathcal{I}$ defined in chapter 5. For mutually independent Poisson distributed variables, the state probabilities $P(\mathbf{X} = \mathbf{x})$ have a product form,

$$P(X_0 = x_0, X_1 = x_1, ..., X_I = x_I) = \prod_{i=0}^{I} \frac{a_i^{x_i}}{x_i!} e^{-a_i}. \tag{7.1}$$

When considering multicast traffic, the channel has only two possible states,

$X_i = 0$ or $X_i \geq 1$. The latter condition is taken into account by summing over all states with $X_i \geq 1$ resulting in equation (6.1). The state probabilities for a finite user population will be derived using equation (7.1) and the multicast traffic characteristics will be taken into account later.

When the population size is finite, the total number of users is restricted to $X_0+X_1+...+X_I = N$, where X_0 is the random variable for the number of users not requesting any channel. If the Poisson parameters are $a_i = Np_i$, where (in the case of a finite population) p_i is the probability that a user requests channel i and Np_i is the expected value of N users requesting channel i, equation (7.1) changes to

$$P(X_0 = x_0, ..., X_I = x_I \mid X_0 + ... + X_I = N) =$$
$$= \begin{cases} \dfrac{\prod_{i=0}^{I} \dfrac{(Np_i)^{x_i}}{x_i!} e^{-Np_i}}{P(X_0 + ... + X_I = N)}, & \text{if } x_0 + ... + x_I = N, \\ 0, & \text{otherwise,} \end{cases} \tag{7.2}$$

where the probability of choosing a channel is $p_1 + ... + p_I = 1 - p_0$. The variable $X_0 + ... + X_I$ is distributed as a Poisson variable with parameter $Np_0 + ... + Np_i = N$. Thus the probability that the population size is N is

$$P(X_0 + ... + X_I = N) = \frac{N^N e^{-N}}{N!}.$$

Equation (7.2) is thus

$$P(X_0 = x_0, ..., X_I = x_I \mid X_0 + ... + X_I = N) =$$
$$= \begin{cases} \dfrac{N^N e^{-N} \prod_{i=0}^{I} \dfrac{p_i^{x_i}}{x_i!}}{\dfrac{N^N}{N!} e^{-N}} = N! \prod_{i=0}^{I} \dfrac{p_i^{x_i}}{x_i!} & , \text{ if } x_0 + ... + x_I = N, \\ 0 & , \text{ otherwise.} \end{cases} \tag{7.3}$$

This is the state probability of variables having a multinomial distribution with parameters N and $p_0, ..., p_I$. The multinomial distribution is a generalization of the binomial distribution. The sum of the probabilities $p_0, ..., p_I$ is equal to one. The state probability $P(\mathbf{X} = \mathbf{x})$ gives the probability that of the N users x_0 have not chosen any channel, x_1 have chosen channel 1 etc. The sum of the number of users requesting the different channels $x_0 + x_1 + ... + x_I$ has to be equal to N.

7.2 Model for a Single User

In the paper by Chan and Geraniotis [5], for each logical state (p, s, t) there was only one user that was either active or idle. This is not a very practical model, as often the user is presented with a selection of sources or channels to choose from and is idle when none of these channels is chosen. The finite model proposed here is a Markov chain, with $I + 1$ states. All transitions by user u are made via the idle state, corresponding to state 0. The transition rate from state 0 to any state $i \in I$ is denoted by $\lambda_{u,i} = \alpha_i \hat{\lambda}_u$, where α_i is the preference distribution used earlier. The transition rate from any state i to state 0 is denoted by μ_i. The Markov chain is shown in figure 7.1. The steady state probabilities solved from the detailed balance equations give

$$\begin{aligned} \pi_{u,i} &= \rho_{u,i}\pi_{u,0} \\ \pi_{u,0} &= \Big[1 + \sum_{i=1}^{I} \rho_{u,i}\Big]^{-1}, \end{aligned}$$

where $\rho_{u,i} = \frac{\alpha_i \hat{\lambda}_u}{\mu_i}$.

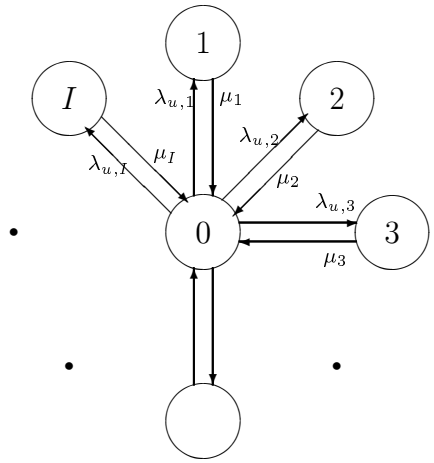

Figure 7.1: The Markov chain used to model user behavior.

Because the detailed balance equations

$$\pi_{u,0}\lambda_{u,i} = \pi_{u,i}\mu_i, \forall i \in \mathcal{I}$$

are satisfied, the process is reversible and the state space can be truncated. Furthermore, it can be shown that the insensitivity property discussed in sec-

tion 3.1.3 applies and the service times can be generally distributed with mean $1/\mu_i$.

The probability P_u that user u connects to the multicast network is by definition

$$P_u = \frac{\sum_{k\in\mathcal{I}} \rho_{u,k}}{1+\sum_{k\in\mathcal{I}} \rho_{u,k}}, \forall u \in \mathcal{U}, \tag{7.4}$$

from which it follows that

$$1 - P_u = \frac{1}{1+\sum_{k\in\mathcal{I}} \rho_{u,k}} = \pi_{u,0}, \forall u \in \mathcal{U}.$$

In addition, the parameter $\hat{\alpha}_i$ is defined as the conditional probability of being in state i given that the user connects to the multicast network,

$$\hat{\alpha}_i = \frac{\rho_{u,i}}{\sum_{k\in\mathcal{I}} \rho_{u,k}} = \frac{\alpha_i/\mu_i}{\sum_{k\in\mathcal{I}} \alpha_k/\mu_k}, \forall i \in \mathcal{I}. \tag{7.5}$$

It follows that $\pi_{u,i}$ in terms of P_u and $\hat{\alpha}_i$ is

$$\pi_{u,i} = P_u \hat{\alpha}_i, \forall i \in \mathcal{I}, \forall u \in \mathcal{U},$$

and the steady state probabilities $\pi_u(\mathbf{x}_u)$ for user u have the following form,

$$\pi_u(\mathbf{x}_u) = \begin{cases} P_u\hat{\alpha}_i & , \text{ if } \mathbf{x}_u = \mathbf{e}_i, i \in I, \\ 1 - P_u & , \text{ if } \mathbf{x}_u = \mathbf{0}, \\ 0 & , \text{otherwise.} \end{cases} \tag{7.6}$$

7.3 A Network with Infinite Link Capacities

Consider a network with all links having infinite capacity. Behind leaf link k there is one user that subscribes to the network with probability P_k, with the steady state probabilities at link k given by equation (7.6).

Similarly to section 6.3, the OR-convolution gives the link state distribution. Thus, the state probability, denoted by $\sigma_j(\mathbf{y}_j)$, for $\mathbf{y}_j \in \mathcal{S}$, is equal to

$$\sigma_j(\mathbf{y}_j) = P(\mathbf{Y}_j = \mathbf{y}_j) = [\bigotimes_{k\in\mathcal{U}_j} \pi_k](\mathbf{y}_j).$$

When the leaf link state probabilities are defined as in equation (7.6), the link state probabilities obtained by convolving all leaf link distributions downstream link j gives the same result as calculating the state probabilities of

$|\mathcal{U}_j| = N$ users using a multinomial distribution with parameters $p_i = P_u\hat{\alpha}_i$, for $i \in \mathcal{I}$ and $p_0 = 1 - P_u$ and then summing the state probabilities to take into account the multicast conditions. Here it is assumed that all leaf links $k \in \mathcal{U}_j$ have $P_k = P_u$, that is they form a new user population u with N users.

As the number of users N in the finite population model tends to infinity, the population model converges to the infinite population model presented by Karvo et al. This is easily seen, as the multinomial distribution with parameters $p_i = P_u\hat{\alpha}_i$ and expected value $NP_u\hat{\alpha}_i$ converges to a distribution of independent Poisson distributed variables with parameter $a_{u,i} = \frac{\lambda_u}{\mu_i}\alpha_i$. Writing the expected value with the help of equations (7.4) and (7.5) gives,

$$NP_u\hat{\alpha}_i = N\frac{\hat{\lambda}_u \sum_{k\in\mathcal{I}} \alpha_k/\mu_k}{1 + \hat{\lambda}_u \sum_{k\in\mathcal{I}} \alpha_k/\mu_k}\frac{\alpha_i/\mu_i}{\sum_{k\in\mathcal{I}} \alpha_k/\mu_k}$$

The limit of the expected value $NP_u\hat{\alpha}_i$ is then,

$$\lim_{N\to\infty}(NP_u\hat{\alpha}_i) = \lim_{N\to\infty}(\frac{\alpha_i}{\mu_i}\frac{N\hat{\lambda}_u}{1 + \hat{\lambda}_u \sum_{k\in\mathcal{I}} \alpha_k/\mu_k}) \to \frac{\alpha_i}{\mu_i}\lambda_u = a_{u,i}, \forall i \in \mathcal{I}$$

It can be seen that the finite user population model converges to the infinite user population model defined in chapter 6, when $\lim_{N\to\infty} N\hat{\lambda}_u \to \lambda_u$.

It is worthwhile to study the link occupancy distributions in a network in terms of the varying size of the user population.

7.3.1 The Link Occupancy Distribution for Varying Population Size

The link occupancy distribution is studied for a single link with infinite capacity. This same approach has been taken for the infinite population size in the earlier paper by Karvo et al. [10]. Here, the link occupancies will be studied for a finite user population. The link occupancy distribution gives a rough estimation of the capacity needed to ensure that blocking probability stays below the critical level (e.g. 0.01). Clearly, links with one user subscribed to them need to have a capacity of $\max_i d_i$, and as the number of users using a link increases, so that each channel surely has at least one user, the required capacity approaches the maximum capacity $\sum_{i\in\mathcal{I}} d_i$.

The link occupancy distributions are calculated for varying population sizes. The example shown in figure 7.2 is a multicast network with leaf links having two users. The next level of the network has four links with four users connected to the link combined to form eight links with a user population of 16 users. At the first level, two links with a user population of 64 combine to form the common link with a user population of 128 users. Each link in the network has infinite capacity and the link occupancy distributions are calculated using the OR-convolution as described in the previous section. The link occupancy

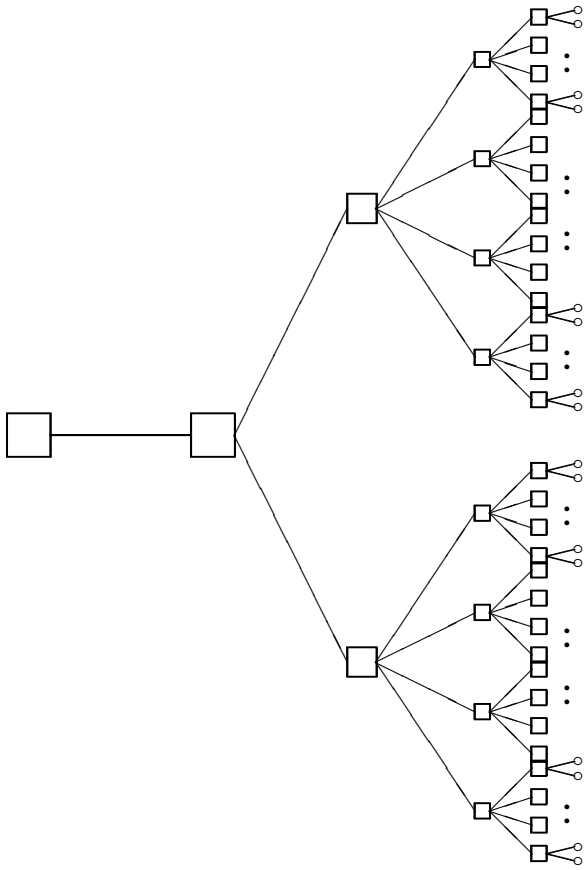

Figure 7.2: The example network.

distributions are calculated for each single link in the example network. They are state probabilities on a single link with infinite capacity defined by the downstream user population of size N, not joint probabilities of the network.

Figure 7.3 shows how the link occupancy distribution changes as the number of users increases. The channel preference distribution (α_i) is calculated using the truncated geometric distribution (equation 5.2). The number of channels is $I = 8$, and the probability that a user subscribes to the link is $P = 0.5$. The user populations are 2, 4, 16, 64, and 128. The probability to subscribe to

the network is quite large, and therefore the link occupancy probabilities are large for capacity requirements equal to the user population size. The capacity required on the link is equal to the population size N, when $N \leq I$, and equal to the number of offered channels I, when the population size is larger than the number of offered channels. This of course applies to cases when the capacity requirements of the channels are the same. For singleservice traffic the capacity needed on the link is simply $\min(N, I)$, when the probability of subscribing to the network is large.

In figure 7.4, the probability of subscribing to the network is $P = 0.1$. Here, the capacity required is less than the number of channels offered while $N < 64$. Figure 7.5 shows the link occupancy distribution for a multiservice network, with a capacity requirement of one unit for odd channels and two units for even channels. The maximum required capacity of the network is 12.

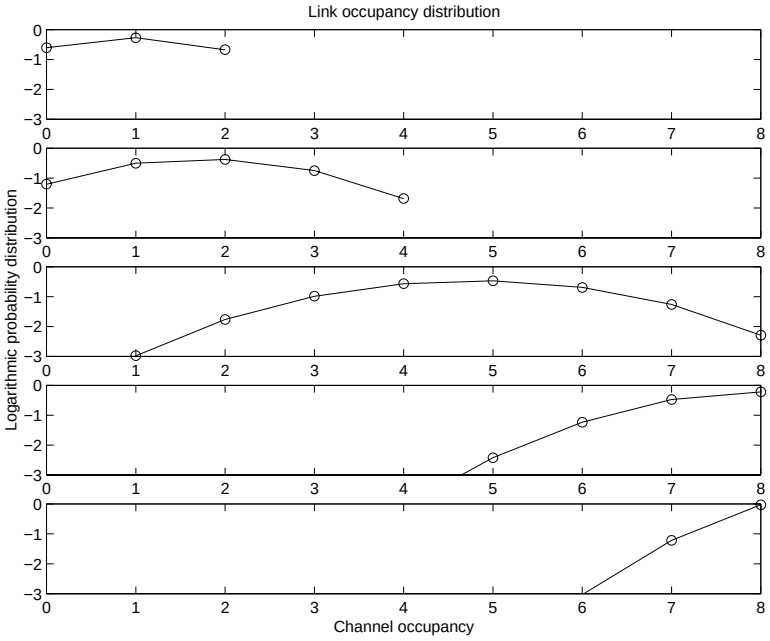

Figure 7.3: Link occupancy distribution (logarithmic) for 2, 4, 16, 64, and 128 users and 8 channels, with $P = 0.5$.

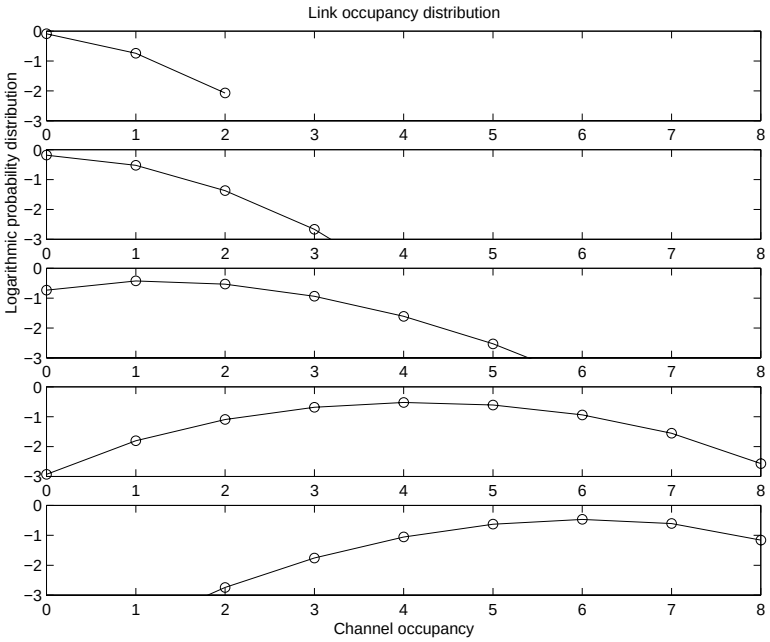

Figure 7.4: Link occupancy distribution (logarithmic) for 2, 4, 16, 64, and 128 users and 8 channels with $P = 0.1$.

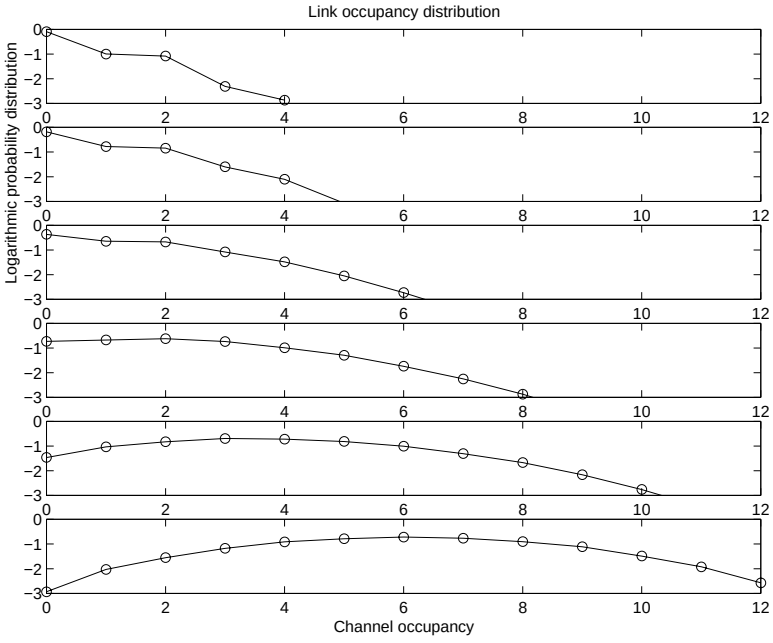

Figure 7.5: Link occupancy distribution (logarithmic) for 2, 4, 8, 16, 32, and 64 users, 8 channels with $P = 0.1$, and capacity requirement of one unit for odd and two units for even channels.

7.4 End-to-End Channel Blocking Probability

The algorithm devised in section 6.5 can be used to calculate end-to-end channel blocking probabilities in multicast networks with finite user populations. A network with a single user behind each leaf link is considered. This is a general model as larger user populations may be obtained by OR-convolving the state probabilities of single users in infinite links.

The Markov chain model that describes the behavior of a user was presented in section 7.2. The model assumes that each user is subscribed to one channel at a time and a request for a new channel can occur only through the idle state. A user's channel request is therefore blocked only if there is not enough capacity to turn the channel on. For this reason, channel blocking probabilities are studied. For the finite population model, channel blocking for user u is equal to time blocking in a network where user u is removed, as a user in the idle state that is blocked remains in the idle state. The original leaf link state probabilities are

$$\pi_u(\mathbf{x}_u) = \begin{cases} P_u\hat{\alpha}_i & , \text{ if } \mathbf{x}_u = \mathbf{e}_i, i \in I, \\ 1 - P_u & , \text{ if } \mathbf{x}_u = \mathbf{0}, \\ 0 & , \text{otherwise.} \end{cases} \tag{7.7}$$

Removing user u from the network is equivalent to setting user u in state 0,

$$\pi_u(\mathbf{x}_u) = \begin{cases} 1 & , \text{ if } \mathbf{x}_u = \mathbf{0}, \\ 0 & , \text{otherwise.} \end{cases}$$

For all other $u^* \in \mathcal{U}$ the state probabilities are given by equation (7.7).

As described in section 6.2 the leaf link distributions define the network distribution and by using the state probabilities defined above, the algorithms presented in section 6.5 can be used. The resulting end-to-end channel blocking probability is,

$$B^c_{u,i} = 1 - \frac{\sum_{\mathbf{y}\in\mathcal{S}} Q_J^{u,i}(\mathbf{y})}{\sum_{\mathbf{y}\in\mathcal{S}} Q_J(\mathbf{y})},$$

where

$$Q_j(\mathbf{y}) = \begin{cases} 1_{\mathbf{y}=\mathbf{0}} & , \text{ if } j = u, j \in \mathcal{U}, \\ T_j\pi_j(\mathbf{y}) & , \text{ if } j \notin u, j \in \mathcal{U}, \\ T_j[\bigotimes_{k\in\mathcal{N}_j} Q_k^{u,i}](\mathbf{y}) & , \text{otherwise,} \end{cases}$$

and

$$Q_j^{u,i}(\mathbf{y}) = \begin{cases} 1_{\mathbf{y}=\mathbf{0}} & , \text{ if } j = u, j \in \mathcal{U}, \\ T_j^{u,i}\pi_j(\mathbf{y}) & , \text{ if } j \notin u, j \in \mathcal{U}, \\ T_j^{u,i}[\bigotimes_{k \in \mathcal{N}_j} Q_k^{u,i}](\mathbf{y}) & , \text{otherwise.} \end{cases}$$

The end-to-end channel blocking was calculated using the network in section 7.3.1, with user population size on the ascending levels being 2, 4, 16, 64, and 128 users. The blocking probability was calculated for different network capacities with identical number of channels ($I = 8$), channel capacity requirements, and channel preference distributions (P=0.1). The leaf links with capacity set to one and one user behind them, are not shown in the figure for the example network, as no blocking occurs in these links. The blocking probabilities are represented table 7.1.

The end-to-end channel blocking probability was also calculated for the network in figure 8.5 with population size in ascending levels being 1, 4, 16, 64, and 256. The same conditions hold for this network as for the previous network, only the topology is different. The results are shown in table 7.2.

The capacity allocations of table 7.1 show that an increase in capacity at any level decreases the blocking probability, as should be the case for singleservice traffic. It seems that an increase in capacity at the common link decreases the blocking probability the most. This is especially true, when the capacity allocations at the previous levels are reasonable. In the next section, the optimum capacity allocation will be studied using Moe's principle.

Table 7.1: Channel blocking probabilities for the network in figure 7.2.

Link capacities					
C_5	C_4	C_3	C_2	C_1	$B_{i=8}^c$
1	3	4	6	5	0.1080
1	3	4	5	6	0.0963
1	3	4	6	7	0.0950
2	3	4	5	5	0.0267
2	4	5	5	5	0.0267
2	3	5	5	6	0.0041
2	3	5	6	6	0.0033
2	3	6	6	7	0.0002

Table 7.2: Channel blocking probabilities for the network in figure 8.5.

Link capacities				
C_4	C_3	C_2	C_1	$B^c_{i=8}$
3	4	6	6	0.5666
2	3	6	7	0.3244
3	3	7	7	0.3167
3	4	7	7	0.2915
3	4	7	8	0.0201
3	4	8	8	0.0147
3	5	8	8	0.0017

Chapter 8

Dimensioning a Multicast Network With a Finite User Population

A network has to be able to carry the call requests that it receives. This requires sufficient capacity. On the other hand, having too much idle capacity in a link or in the whole network is inefficient. Dimensioning the network means deciding how much capacity is needed in order to maintain a good quality of service. Blocking probabilities are used as one principle. Ideally, no blocking should occur. A more realistic goal is to allow some blocking, e.g. of magnitude 10^{-2}. The parameters that affect the blocking probabilities, besides the topology of the network, are the capacities of the links and the offered traffic intensity. The network is dimensioned by calculating the required capacities according to the known or forecasted traffic intensity and the greatest allowed blocking probability.

The dimensioning of multicast networks will be studied in section 8.1 for the simplified case of a network with all but one link having infinite capacity and for a network with more than one link having finite capacity in section 8.2.

8.1 Blocking Probabilities for a Single Link Having Finite Capacity

Blocking probabilities can be calculated for a single finite capacity link with a finite user population. Using the link occupancy distribution of the previous section, an approximate time blocking probability can be calculated. This is done in section 8.1.1. Time blocking probabilities for a finite population subscribing to a network with all but one link in the network having infinite capacity are calculated in section 8.1.2.

8.1.1 An Approximation for the Blocking Probability in a Single Link

The link occupancy distribution discussed in the previous section can also be expressed as blocking probabilities. However, the blocking probability used in this section will be the proportion of time that the traffic on the link occupies full capacity. This saturation probability is not the time blocking probability, as subscriptions to channels may still be accepted to the link. Even if a link is full, subscriptions are accepted, as long as the channel is turned on. Whereas, if the channel is idle blocking occurs. For singleservice traffic, the saturation probability of the link gives an upper bound for the time blocking probability. The approximation gives an indication to how a link should be dimensioned and illustrates the relationship between link capacity and population size. Figures 8.1 and 8.2 show how the required capacity depends on the population size for two values ($B = 0.01$ and $B = 0.001$) of the targeted blocking probability. In figure 8.1, all channels require one unit of capacity and in figure 8.2, odd channels require one unit of capacity and even channels require two units of capacity. The preference distribution is still the geometric distribution with parameter $p = 0.2$. Figures 8.1 and 8.2 show that, as the blocking probability constraint drops from 0.001 to 0.01, the number of users allowed until full capacity is needed on the link almost doubles.

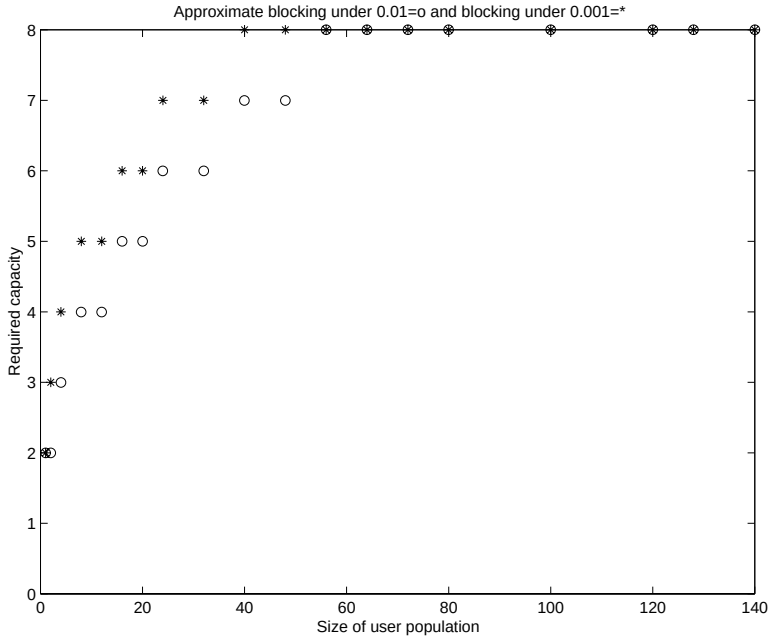

Figure 8.1: Required capacity vs. user population size for link occupancy blocking probability $B = 0.01$ and $B = 0.001$, 8 channels with $P = 0.1$, and one unit of capacity required.

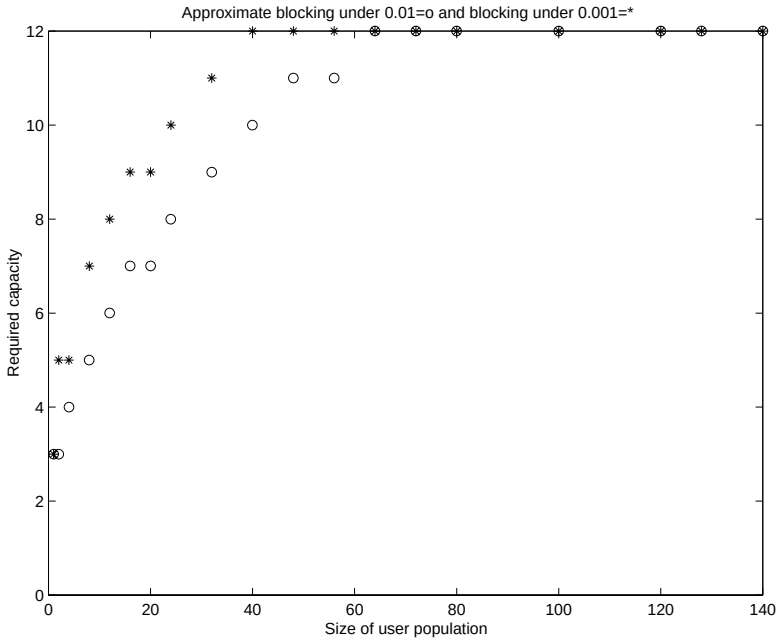

Figure 8.2: Required capacity vs. user population size for link occupancy blocking probability $B = 0.01$ and $B = 0.001$, 8 channels with $P = 0.1$, and capacity requirement of one unit for odd and two units for even channels.

8.1.2 Time Blocking Probability in a Single Link

The figures shown in the previous sections give a good approximation to the capacity required on a multicast link. The exact time blocking probability can be found using equation (5.11),

$$B_i^t = \frac{\sum_{j=C-c_i+1}^{C} \pi_j{}^{(x_i=0)}}{\sum_{j=0}^{C} \pi_j},$$

here $\pi_j^{(x_i=0)}$ is the aggregate state probability of the link occupancy including all states with channel i off,

$$\pi_j^{(x_i=0)} = \sum_{\mathbf{y}\cdot\mathbf{d}=j, y_{j,i}=0} \sigma(\mathbf{y}).$$

The state probability $\sigma(\mathbf{y})$ for $N = |\mathcal{U}_j|$ users is obtained by OR-convolving the state probability for one user $\pi_u(\mathbf{x}_u)$ given in equation (7.6) $N-1$ times as explained in section 7.3.

Figure 8.3 shows the capacity needed for a single link carrying singleservice traffic in order for the blocking probability to be below the given upper bounds. In figure 8.4 the traffic classes have different capacity requirements. The probability that the user population subscribes to the channel is $P = 0.1$ for all sizes of the user population. Comparing these figures to the ones obtained by the approximate blocking probability, figures 8.1 and 8.2 it can be deduced that the approximate blocking probability gives rather good results for singleservice traffic. For some numerical values too much capacity is allocated, as the approximate blocking probability is an upper bound for the time blocking probability. For multiservice traffic, the saturation probability does not always give an upper bound for the time blocking probability, as the actual blocking states include states where the capacities occupied on the link are less than the saturation level.

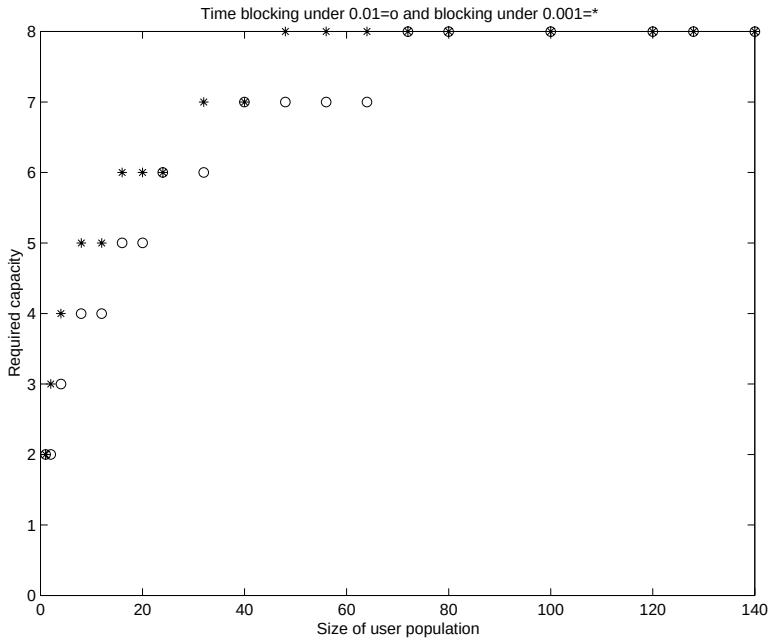

Figure 8.3: Required capacity vs. user population size for link occupancy time blocking probability $B = 0.01$ and $B = 0.001$, 8 channels with $P = 0.1$.

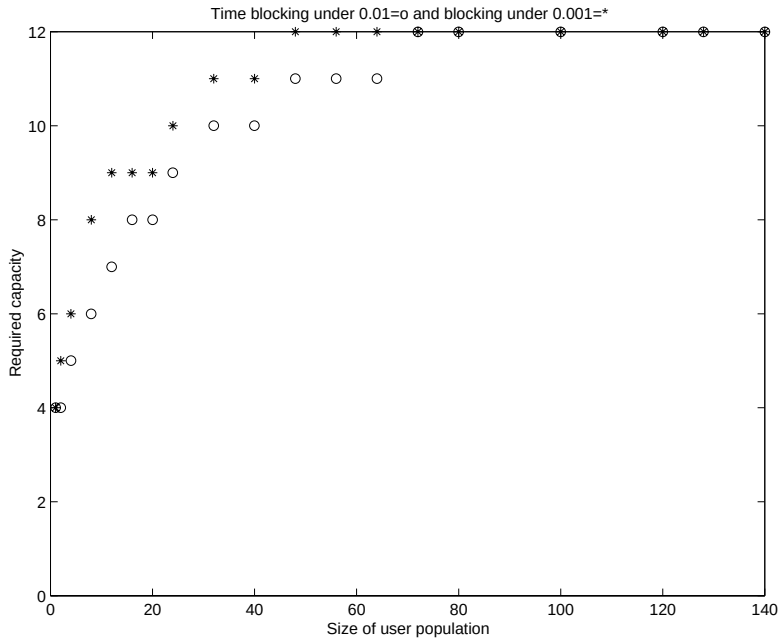

Figure 8.4: Required capacity vs. user population size for link occupancy time blocking probability $B = 0.01$ and $B = 0.001$, 8 channels with $P = 0.1$, and capacity requirement of one unit for odd and two units for even channels.

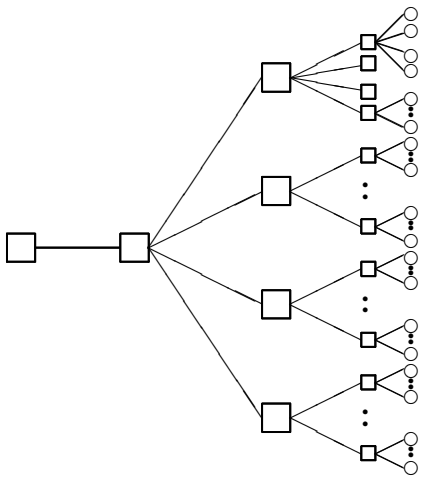

Figure 8.5: Network used to test Moe's criterion.

8.2 Dimensioning the network using Moe's principle.

The blocking probability of the network depends on the capacities allocated to the links of the network. The dimensioning of a network is a tradeoff between the cost of allocating capacity to the links and the lost revenue due to blocked customers. The capacity should therefore be allocated to those links that affect the blocking probability the most in relation to the cost of the extra capacity. As was shown in the previous section, it is not obvious which capacity allocation is the optimum. Moe's principle compares the effect different capacity allocations have on the blocking probability, and in addition takes into account the cost of the allocation. Moe's principle can be used for a symmetric network, where the fan out of the network is equal at each level. The capacity is allocated to the level that decreases the blocking probability the most in relation to the cost of the capacity. The cost of the capacity allocation is assumed equal to the total capacity allocated to each link.

The network used as an example has four levels. The common link has 256 users connected to it. The network fans out into 4 links, with each having 64 users. These links fan out to four links with 16 users and finally each link with 16 users fans out to four links with 4 users as shown in figure 8.5. The allocation of one unit of capacity to the common link only costs one unit, while the allocation to the fourth level costs 64 units, as there are 64 links on the

fourth level of the network. Define l as the stage of the network so that the leaf links have $l = L = 5$ and the common link has $l = 1$. The number of links belonging to stage l is denoted by M_l. For the example network shown in figure 8.5 $M_1 = 1$ and $M_4 = 64$.

The iteration is as follows

1. Choose an initial capacity allocation $\mathbf{C}_0$.
2. Calculate $H_l = \frac{B(\mathbf{C}_0) - B(\mathbf{C}_0 + e_l)}{M_l}$, where B is the channel blocking probability and M_l is the number of links at stage l.
3. Allocate the capacity to all the links j belonging to stage l, with the maximum H_l.
4. Update the capacity allocation of the network $\mathbf{C}_1 = \mathbf{C}_0 + \mathbf{e}_l$.
5. Repeat 2-5, until the end-to-end blocking probability is less than the target level (e.g. $B < 0.01$).

The network is dimensioned using the exact end-to-end channel blocking probability for the least used channel ($i = 8$). Usually the end-to-end blocking probability used in Moe's principle is the sum of the blocking probabilities in individual links. This is approximately true when the blocking probabilities are small.

The initial guess can be deduced using the saturation probabilities or the blocking probabilities calculated in the previous section. The number of users in the example network is [4 16 64 256]. The capacity needed to ensure a blocking probability less than 0.01 in each link can be used as an initial guess. Figure 8.1 in section 8.1 shows that a suitable initial guess would be [3 4 6 8]. In order to study the effect of the initial capacity allocation guess to the solution, Moe's principle is used for different initial guesses [2 3 5 7], [3 4 6 7], [4 4 5 7], and [3 5 7 7]. They are in tables 8.1 through 8.4. The importance of the initial guess can be seen from the results.

For the initial guesses [2 3 5 7], [3 4 6 7], and [3 5 7 7], the optimal capacity allocation is the same [3 5 8 8] for blocking probability below 0.01 or [3 6 8 8] for a blocking probability below 0.001, though the steps to reach the optimum vary. The initial guess [4 4 5 7] is too large, and the optimum capacity allocation [4 4 6 8] uses too much capacity. From this example, it can be deduced that

a network with 256 users that are subscribed to the network one tenth of the time, needs full capacity on the common link as well as on the preceding level, with 64 users.

Table 8.1: The iteration steps for the network in figure 8.5, with the initial guess [2 3 5 7].

Link capacities				
C_4	C_3	C_2	C_1	$B^c_{i=8}$
2	3	5	7	0.3583
2	3	5	8	0.2089
2	3	6	8	0.1176
2	3	7	8	0.0939
2	4	7	8	0.0352
2	4	8	8	0.0304
2	5	8	8	0.0212
3	5	8	8	0.0017
3	6	8	8	0.0005

Table 8.2: The iteration steps for the network in figure 8.5, with the initial guess [3 4 6 7].

Link capacities				
C_4	C_3	C_2	C_1	$B^c_{i=8}$
3	4	6	7	0.3002
3	4	6	8	0.0625
3	4	7	8	0.0201
3	4	8	8	0.0147
3	5	8	8	0.0017
3	6	8	8	0.0005

Table 8.3: The iteration steps for the network in figure 8.5, with the initial guess [4 4 5 7].

Link capacities				
C_4	C_3	C_2	C_1	$B^c_{i=8}$
4	4	5	7	0.3447
4	4	5	8	0.1837
4	4	6	8	0.0623
4	4	7	8	0.0199
4	4	8	8	0.0144
4	5	8	8	0.0013
4	6	8	8	0.0005

Table 8.4: The iteration steps for the network in figure 8.5, with the initial guess [3 5 7 7].

Link capacities				
C_4	C_3	C_2	C_1	$B^c_{i=8}$
3	5	7	7	0.2878
3	5	7	8	0.0086
3	5	8	8	0.0017
3	6	8	8	0.0005

Chapter 9

End-to-end Blocking Probabilities in a Network with Background Traffic

The networks considered until now were assumed to transfer only multicast traffic. The model can, however, be extended to cover networks with mixed traffic. In this case, the network transfers, in addition to multicast traffic, non-multicast traffic that is assumed independent on each link. The distribution does not depend on the multicast traffic in the link or on the traffic in the other links. The non-multicast traffic in link j is assumed to be Poisson with a traffic intensity A_j. The capacity requirement is equal to one unit of capacity. The link occupancy distribution of the non-multicast traffic in a link with infinite capacity is thus

$$q_j(z) = \frac{(A_j)^z}{z!} e^{-A_j}. \tag{9.1}$$

9.1 The Refined Algorithm

The inclusion of non-multicast traffic affects only the truncation step of the algorithm presented in section 6.5. The state probabilities are defined as in section 6.4. The state probabilities of the link states that require more capacity than available on the link are set to zero as before. However, the state probabilities of the states that satisfy the capacity restriction of the link are altered, as the available capacity on the link depends on the amount of non-multicast

traffic on the link. Another way of describing the relationship between the two different types of traffic, is to consider them as two traffic classes in a two dimensional infinite link occupancy state space as is shown in figure 9.1. The traffic classes are independent of each other. The capacity of the link is a linear constraint of this state space. Notice that the marginal distribution of the capacity occupancy of the multicast traffic is weighted by the sums over the columns of the occupancy probabilities of the background traffic. If the multicast traffic occupies $c = \mathbf{d} \cdot \mathbf{y}_j$ units of capacity, and the link capacity is C_j, then possible non-multicast traffic states on the link are those with $z \leq C_j - c$, where z is the number of non-multicast calls.

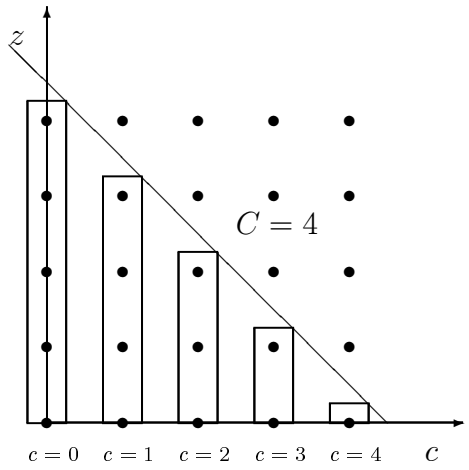

Figure 9.1: Shaping of the marginal distribution of the capacity occupancy when background traffic is included in the model.

Therefore, the truncation operators presented in equations (6.5) and (6.7) must be replaced by the operators

$$\hat{T}_j f(\mathbf{y}) = \begin{cases} \sum_{z=0}^{C_j - \mathbf{d}\cdot\mathbf{y}} q_j(z) f(\mathbf{y}) & , \text{ if } \mathbf{y} \in \tilde{\mathcal{S}}_j \\ 0 & , \text{ otherwise} \end{cases}$$

$$\hat{T}_j^{u,i} f(\mathbf{y}) = \begin{cases} \sum_{z=0}^{C_j - \mathbf{d}\cdot(\mathbf{y} \oplus (\mathbf{e}_i 1_{j \in \mathcal{R}_u}))} q_j(z) f(\mathbf{y}) & , \text{ if } \mathbf{y} \in \tilde{\mathcal{S}}_j^{u,i} \\ 0 & , \text{ otherwise.} \end{cases}$$

The algorithm differs therefore only by the truncation operator used,

$$\hat{Q}_j(\mathbf{y}) = \begin{cases} \hat{T}_j \pi_j(\mathbf{y}) & , \text{ if } j \in \mathcal{U} \\ \hat{T}_j[\bigotimes_{k \in \mathcal{N}_j} \hat{Q}_k](\mathbf{y}) & , \text{ otherwise.} \end{cases}$$

Similarly,

$$\hat{Q}_j^{u,i}(\mathbf{y}) = \begin{cases} \hat{T}_j^{u,i} \pi_j(\mathbf{y}) & , \text{ if } j \in \mathcal{U} \\ \hat{T}_j^{u,i}[\bigotimes_{k \in \mathcal{N}_j} \hat{Q}_k^{u,i}](\mathbf{y}) & , \text{ otherwise.} \end{cases}$$

The blocking probability in equation (6.4) is again obtained by two series of convolutions and truncations from the leaf links to the common link J. The end-to-end time blocking probability of the network is

$$\hat{b}_{u,i}^c = 1 - \frac{\sum_{\mathbf{y} \in \mathcal{S}} \hat{Q}_J^{u,i}(\mathbf{y})}{\sum_{\mathbf{y} \in \mathcal{S}} \hat{Q}_J(\mathbf{y})}.$$

9.2 Numerical Results for End-to-End Call Blocking Probabilities for an Infinite User Population

The leaf link probability

$$\pi_u(\mathbf{x}_u) = \prod_{i \in \mathcal{I}} (p_{u,i})^{x_{u,i}} (1 - p_{u,i})^{1 - x_{u,i}},$$

and the algorithm presented in the previous section give the end-to-end call blocking probability for a network with infinite user population and background traffic. The end-to-end call blocking probability was calculated using the same network as in section 6.6, figure 6.2. The intensity of the non-multicast traffic was set to $A_j = 0.1$ for all links. Table 9.1 shows the end-to-end call blocking probability for a network with only multicast traffic and for a network transferring multicast and non-multicast traffic. Table 9.2 shows the end-to-end call blocking probabilities when the multicast traffic requires twice the capacity of the non-multicast traffic.

The intensity of non-multicast traffic stays the same, as the intensity of the multicast traffic increases. Clearly, the blocking probabilities are affected less, as the intensity of the multicast traffic increases. This can also be seen by studying the relative change in blocking probabilities shown in tables 9.1 and

9.2. The effect of the non-multicast traffic to the blocking probability is of the same magnitude on both routes. From table 9.1 we see that an inclusion of unicast traffic with one tenth the intensity $a = 1.0$ of the multicast traffic almost doubles the blocking probability. From table 6.1 the blocking probability increases by a factor of 1.5, when the traffic intensity a is increased from 1.0 to 1.1. These two cases are not equivalent as the background traffic is assumed independent of the multicast traffic, but give a good reference to the effect that the background traffic has on end-to-end blocking probabilities.

Table 9.1: Blocking probabilities for the network in figure 6.2 with background traffic and multicast traffic.

	Route1 ($u = 1$)			Route2 ($u = 2$)		
a	Multicast	Background	Rel. change	Multicast	Background	Rel. change
1.0	0.0056	0.0109	1.95	0.0027	0.0053	1.96
1.2	0.0121	0.0206	1.70	0.0060	0.0105	1.75
1.4	0.0220	0.0341	1.55	0.0112	0.0177	1.58
2.0	0.0715	0.0927	1.30	0.0382	0.0501	1.31

Table 9.2: Blocking probabilities for the network in figure 6.2 with background traffic requiring one unit and multicast traffic requiring two units of capacity.

	Route1 ($u = 1$)			Route2 ($u = 2$)		
a	Multicast	Background	Rel. change	Multicast	Background	Rel. change
1.0	0.0056	0.01	1.79	0.0027	0.0049	1.81
1.2	0.0121	0.0195	1.61	0.0060	0.0099	1.65
1.4	0.0220	0.0328	1.49	0.0112	0.0171	1.53
2.0	0.0715	0.0914	1.28	0.0382	0.0495	1.30

9.3 Numerical Results for End-to-End Channel Blocking Probabilities for a Finite User Population

The idea is the same as in the preceding chapter, as only the truncation is affected when non-multicast traffic is included into the multicast network model. Using the leaf link distribution

$$\pi_u(\mathbf{x}_u) = \begin{cases} 1 & , \text{ if } \mathbf{x}_u = \mathbf{0}, \\ 0 & , \text{otherwise} \end{cases}$$

for user u, and for all other users $u^* \in \mathcal{U}$ the leaf link distribution

$$\pi_{u^*}(\mathbf{x}_{u^*}) = \begin{cases} P_{u^*}\hat{\alpha}_i & , \text{ if } \mathbf{x}_{u^*} = \mathbf{e}_i, i \in I, \\ 1 - P_{u^*} & , \text{ if } \mathbf{x}_{u^*} = \mathbf{0}, \\ 0 & , \text{otherwise} \end{cases}$$

to the algorithm presented in section 9.1 gives the end-to-end channel blocking probability for a network with a finite user population.

However, the choice of the intensity A_j is not necessarily as straightforward. When each leaf link has an infinite user population, it is reasonable to assume that the intensity of non-multicast traffic is equal in all links in the network. For a finite user population, the number of users connected to a link depends on the stage of the network that the link is located at. One could assume that a link with a user population of four has less non-multicast traffic than a link with 256 users. Furthermore, the assumption that the non-multicast traffic is from a Poisson process implies that the user population of the non-multicast traffic is infinite. This assumption applies to large networks with few multicast users.

The example network is the same as in section 8.2. The multicast traffic is offered by a finite user group, while the background traffic is offered by an infinite user group and independently to each link. The first network has the same non-multicast traffic intensity for all links, while the second has a background traffic intensity depending on the number of multicast users subscribed to the link. Neither network has non-multicast traffic in the leaf links that have one user. The results depicted in table 9.3 show that the effect of having the same amount of non-multicast traffic in all the links does not differ much from the case of having non-multicast traffic proportional to the number of users in the link. Only when the blocking probabilities are small, is the effect of including background traffic large. For small blocking probabilities, the difference in the two mixed traffic models is also notable.

Table 9.3: End-to-end channel blocking probabilities for the network in figure 8.5 with mixed traffic and multicast traffic.

Link capacities				Multicast	Background $A_j = 0.1$		Background $A_j = 0.1/N$	
C_4	C_3	C_2	C_1	$B^c_{i=8}$	$B^c_{i=8}$	Rel. change	$B^c_{i=8}$	Rel. change
2	3	5	7	0.3583	0.3686	1.0287	0.3830	1.0689
2	3	6	8	0.1176	0.1340	1.1397	0.1660	1.4116
2	4	7	8	0.0352	0.0520	1.4771	0.0800	2.2715
3	5	8	8	0.0017	0.0179	10.5530	0.0228	13.4380
3	6	8	8	0.0005	0.0167	33.4241	0.0205	40.9162
4	4	6	8	0.0623	0.0796	1.2783	0.0923	1.4816
4	4	8	8	0.0144	0.0307	2.1306	0.0382	2.6512
4	5	8	8	0.0013	0.0175	13.4657	0.0198	15.2395

Chapter 10

Conclusions

The thesis has explored new methods for calculating blocking probabilities in multicast networks. The study originates from the work by Karvo et al. [10] and [11]. The main result of the thesis is a new exact algorithm for calculating end-to-end blocking probabilities in tree-structured multicast networks.

Based on the infinite user population model given by Karvo et al., and the well-known algorithm for calculating blocking probabilities in hierarchical multiservice access networks, a new exact algorithm for calculating blocking probabilities in tree-structured multicast networks was presented. A new user model was then presented and defined so that a finite user population model of arbitrary size N could be constructed. Furthermore, the new finite user population model was defined in such a way that as the population size grows, the model can be replaced by the infinite user population model.

The different resource usage of multicast traffic called for a new convolution technique, the OR-convolution. Calculating the exact solution for the end-to-end blocking probability, however, becomes infeasible as the number of channels increases. In contrast to ordinary access networks, the one dimensional aggregate link occupancy description is not sufficient, since in a multicast network it is essential to do all calculations in the link state space, with 2^I states. This is due to the resource sharing property of multicast traffic; namely, the capacity in use on a link increases only if a channel that is not already being carried on the link is requested.

In addition to extending the single link approach by Karvo et al. to include the whole multicast network, the study showed that the new algorithm does

not require that the network leaf links have an infinite user population. The proposed finite population model assumes that a single user changes states, i.e. channels, through an idle state. It is defined in this manner, in order to preserve the detailed balance and truncation principle. Thus the algorithm, where link occupancy distributions are alternately convolved and truncated, was applied to networks with finite user populations. End-to-end channel blocking probabilities were calculated for finite user populations using the algorithm, and the dimensioning of the network was studied using Moe's principle.

Most networks do not carry solely multicast traffic. The thesis also generalized the algorithm to include background traffic, e.g. unicast traffic. Blocking probabilities for both, the infinite and finite user population models were studied in networks with background traffic.

The use of one approximation method, the Reduced Load Approximation (RLA), was studied, as the complexity of the RLA-algorithm does not depend critically on the number of channels transmitted in the network. The results showed that the RLA gives blocking probabilities of the same magnitude as the exact result, but even in a small network, with three stages, the error is around 20%.

As a rule, RLA gives larger blocking probabilities than the exact algorithm. It is therefore clear that new approximation methods need to be developed. In addition, efficient computational methods for the calculation of OR-convolutions are also needed in order to apply the exact model presented in this study to point-to-multipoint multicast networks offering a vast selection of programs.

The work presented in the thesis lays the foundation for the study of multicast networks. New exact methods to calculate blocking probabilities and dimension networks are presented, derived, and analyzed. However, the study is more of a theoretical kind, and further research must be done in developing the theory for implementation in the design of large networks, including networks with more than one source.

Bibliography

[1] Almeroth K.C. and Ammar M.H., Multicast Group Behavior in the Internet's Multicast Backbone (MBone), IEEE Communications Magazine 35(6), June 1997, pp. 124 -129.

[2] Armitage G., Support for Multicast over UNI 3.0/3.1 Based ATM networks, Internet RFC 2022, Nov. 1996.

[3] Bagwell R.T., McDearman J.R., Marlow D.T., A Comparison of Native ATM-Multicast to IP-multicast with Emphasis on Mapping Between the Two, Proceedings of the Twenty-Seventh Southeastern Symposium on System Theory, 1995, pp. 189 -193.

[4] Chan W.C., Geraniotis E., Limiting the Access Bandwidth of a Video Source: Model and Performance Analysis, Fourth International Conference on Computer Communications and Networks, Proceedings, 1995, pp. 546 -553.

[5] Chan W.C., Geraniotis E., Tradeoff Between Blocking and Dropping in Multicasting Networks, IEEE International Conference on Communications, Conference Record, Converging Technologies for Tomorrow's Applications pp. 1030 - 1034, 1996.

[6] Diot C. , Dabbous W., Crowcroft J., Multipoint Communication: A Survey of Protocols, functions, and Mechanisms, IEEE Journal on Selected Areas in Communications 15(3), 1997, pp. 277-290.

[7] Freeman R. L., "Telecommunications System Engineering", 3. ed., John Wiley & Sons, 1996.

[8] Guarene E., Fasano P., Vercellone V., IP and ATM Integration Perspectives, IEEE Communications Magazine 36 (1), pp. 74 -80, Jan. 1998.

[9] Hui J., "Switching and traffic theory for integrated broadband networks", Kluwer Academic Publishers, Boston, 1990.

[10] Karvo J., Virtamo J., Aalto S., Martikainen O., Blocking of dynamic multicast connections in a single link, Proc. of Broadband Communications '98, ed. P. Kühn, R. Ulrich, IFIP, 1998.

[11] Karvo J., Virtamo J., Aalto S., Martikainen O., Blocking of Dynamic Multicast Connections, In 4th INFORMS Telecommunications Conference, Boca Raton, Florida, March 1998. (To appear in Telecommunications Systems)

[12] Kelly F.P., "Reversibility and Stochastic Networks", John Wiley & Sons, 1979.

[13] Kim C.K., Blocking Probability of Heterogeneous Traffic in a Multirate Multicast Switch, IEEE Journal on Selected Areas in Communications 14(2), 1996, pp. 374-385.

[14] Listanti M., Veltri L., Blocking Probability of Three-Stage Multicast Switches, IEEE International Conference on Communications, Conference Record, 1998, pp. 623 -629.

[15] Ross K. W., "Multiservice Loss Models for Broadband Telecommunication Networks", Springer-Verlag, London, 1995.

[16] Semeria C., Maufer T., Introduction to IP-multicast Routing, *http://www.3com.com/nsc/501303.html*.

[17] Shacham N., Yokota H., Admission Control Algorithms for Multicast Sessions with Multiple Streams IEEE Journal on Selected Areas in Communications 15(3), 1997, pp. 557-566.

[18] Stallings W., "Data and Computer Communications", Macmillan Publishing Company, 1994.

[19] Stasiak M. and Zwierzykowski P., Analytical Model of ATM Node with Multicast Switching, Mediterranean Electrotechnical Conference, 1998, pp. 683 -687.

[20] Voipio K., Uusitupa S., "Tietoliikenneaapinen", Otatieto, 1998.

[21] Yang Y., A Class of Interconnection Networks for Multicasting, The 10th International Parallel Processing Symposium, Proceedings of IPPS '96, 1996, pp. 796 -802.

[22] Yang Y., On Blocking Probability of Multicast Networks, IEEE Transactions on Communications 46 (7), pp. 957 -968, July 1998.

[23] ATM User-Network Interfaces Specification V3.0, Prentice-Hall, Inc.,1993

[24] Internet Protocol RFC 791, DDN Network Information center, SRI International, Menlo Park, CA, September 1981.