Fair Treatment of Multicast Sessions and Their Receivers

– Incentives for more efficient bandwidth utilization

Patrik Österberg

Department of Information Technology and Media
Mid Sweden University

Doctoral Thesis No. 38
Sundsvall, Sweden
2007

Mittuniversitetet
Informationsteknologi och medier
SE-851 70 Sundsvall
SWEDEN

ISBN 978-91-85317-74-5
ISSN 1652-893X

Akademisk avhandling som med tillstånd av Mittuniversitetet framlägges till offentlig granskning för avläggande av teknologie doktorsexamen onsdagen den 5 december 2007 i sal L111, Mittuniversitetet, Holmgatan 10, Sundsvall.

Tryck: Tryckeriet Mittuniversitetet

To Angelica and Lisa

Abstract

Media-streaming services are rapidly gaining in popularity, and new ones are knocking on the door. Standard-definition *Internet protocol television* (IPTV) has already entered many living rooms, and high-definition IPTV will become common property in the not too distant future. Then even more advanced and resource-demanding services, such as three-dimensional and free-view TV, are next in line. Video streaming is by nature extremely bandwidth intensive, and this development will put the existing network infrastructure to the test.

In scenarios where many receivers are simultaneously interested in the same data, which is the case with popular live content, multicast transmission is more bandwidth efficient than unicast. The reason is that the receivers of a multicast session share the resources through a common transmission tree where data are only transmitted once along any branch. The use of multicast transmission can therefore yield huge bandwidth savings. There are however no really strong incentives for the *Internet service providers* (ISPs) to support multicast transmission, and the deployment has consequently been slow.

We propose that more bandwidth is allocated to multicast flows in the case of network congestion. The ratio is based upon the number of receivers and the bitrate that they are able to obtain, since this is what determines the degree of resource sharing. We believe that it is fair to take this into account, and accordingly call the proposed allocation *multicast-favorable max-min fair*. Further, we present two bandwidth-allocation policies that utilize different amount of feedback to perform allocations that are reasonable close to be multicast-favorable max-min fair.

We also propose two cost-allocation mechanisms that build upon the assumption that the cost for data transmission should be covered by the receivers. The mechanisms charge the receivers based on their share of the resources usage, which in general is favorable to multicast receivers. The two cost-allocation mechanisms differ in that one strives for optimum fair cost allocations, whereas the other might give discounts to some receivers. The discounts facilitate larger groups of receivers, which can provide cheaper services for the non-discounted receivers as well.

The proposals make multicast transmission more attractive to the users of media-streaming services. If the proposals were implemented in multicast-enabled networks, the rest of the ISPs would be forced to support multicast, to stay competitive.

Sammanfattning

Tjänster för strömmad media stiger kraftigt i popularitet, samtidigt som utbudet av denna typ av tjänster ökar. *Internet protocol television* (IPTV) med standardupplösning levereras redan till många hem, och högupplöst IPTV kommer att bli vanligt inom en relativt snar framtid. Mer avancerade tjänster, som tredimensionell TV och TV med fritt valbara vyer, står sedan på tur. Strömmad video är av naturen väldigt bandbreddskrävande, och denna utveckling kommer därför att sätta den befintliga nätverksinfrastrukturen på prov.

Multicast är mer bandbreddseffektivt än unicast för scenarion där många mottagare samtidigt är intresserade av samma data, vilket är fallet med populärt direktsänt material. Anledningen är att mottagarna av multicast-sessioner delar på resurserna via ett gemensamt transmissionsträd, där ingen data sänds mer än en gång över någon gren. Användningen av multicast kan därför generera stora besparingar av bandbredd. Internetleverantörerna har dock inga riktigt starka skäl för att stödja multicast, vilket medfört att spridningen varit långsam.

Vi föreslår att multicast-sessioner tilldelas mer bandbredd när det uppstår trafikstockningar i näten. Fördelningen baseras på antalet mottagare och datatakten som de erhåller, eftersom det är det som avgör graden av resursdelning. Vi anser att det är rättvist att ta hänsyn till detta, och kallar därför den föreslagna bandbreddsfördelningen *multicast-favorable max-min fair*. Vidare så presenteras två bandbreddstilldelningspolicyer som använder sig av olika mängd återkoppling för att uppnå fördelningar som ligger förhållandevis nära den föreslagna.

Vi föreslår även två mekanismer för kostnadsallokering, vilka bygger på antagandet att kostnaden för dataöverföring ska täckas av mottagarna. De föreslagna mekanismerna fördelar kostnaderna mellan mottagarna baserat på deras andel av resursutnyttjandet, vilket generellt är fördelaktigt för multicast-mottagare. De två mekanismerna för kostnadsallokering skiljer sig åt genom att den ena eftersträvar optimalt rättvis fördelning av kostnaderna, medan den andra kan ge rabatt till vissa mottagare. Rabatten möjliggör större grupper med mottagare, vilket även kan reducera kostnaderna för icke rabatterade mottagare.

Förslagen gör multicast mer attraktivt för användarna av strömmad media. Om förslagen implementerades i nätverk med multicast-stöd så skulle övriga Internetleverantörer bli tvungna att stödja multicast för att vara konkurrenskraftiga.

Acknowledgments

Firstly I would like to thank Professor Youshi Xu and Docent Tingting Zhang, who have given me the opportunity to perform my Ph.D. studies at Mid-Sweden University. I also thank Professor Björn Pehrson, at the Royal Institute of Technology, who was the main supervisor for the first part of my Ph.D. studies. A second thanks to Docent Tingting Zhang, for her excellent supervision and fruitful cooperation, especially during the later part of the studies. I would also like to acknowledge Dr. Stefan Pettersson for leading the fund raising, which enabled me to finish my Ph.D. studies, and Professor Theo Kanter and Dr. Mikael Gidlund for giving me constructive feedback on the thesis.

A warm thanks to all the present and previous co-workers at the Information and Communication Systems Division of the Department of Information Technology and Media at Mid-Sweden University. There are too many of you to mention all, but rest assured, you are not forgotten. However, I would particularly like to thank Ulf Jennehag and Daniel Forsgren, with whom I shared a lot of fruitful discussions and a number of interesting conference trips.

Further, I acknowledge Acreo AB, for allowing me to carry out parts of my studies within the project "Managing Dynamic Testbed", and Dr. Mikael Gidlund for presenting me with this opportunity. I also thank the friendly staff at Acreo, who made this such a joyful experience.

But most of all I would like to thank my family for all their love and support. My parents Lena and Christer, for all the ground service during my undergraduate studies and nowadays babysitting with short notice, just to mention a few of the things. My little sister Sofia, who now is pursuing a Ph.D. of her own, for being the best sister a brother can have. My fiancée Angelica, who has taken over a great share of the ground service and given me a wonderful daughter. Angelica and Lisa, I love you and dedicate this thesis to you.

Sundsvall, November 2007

Patrik Österberg

Contents

List of Papers

This thesis is mainly based on the following papers, which are appended to the thesis and herein referred to by their Roman numerals:

I P. Österberg and T. Zhang. A Bandwidth-Allocation Policy Taking Layered Video Multicast into Consideration. In *Proceedings of 8th IASTED International Conference Internet and Multimedia Systems and Applications (IMSA), Kauai, Hawaii, USA*, pages 347–352, August 2004.

II P. Österberg and T. Zhang. Fair Allocation of Link Capacity through Feedback of Bottleneck Information. In *Proceedings of 1st IEEE International Conference on Digital Telecommunications (ICDT), Cap Esterel, Côte d'Azur, France*, August 2006.

III P. Österberg and T. Zhang. Fairer Allocation of Link Capacity through Information Feedback. In *Proceedings of 5th IASTED International Conference on Communication Systems and Networks (CSN), Palma de Mallorca, Spain*, pages 143–148, August 2006.

IV P. Österberg and T. Zhang. Revised Definition of Multicast-Favorable Max-min Fairness. In *Proceedings of 3rd IASTED International Conference on Communications and Computer Networks (CCN), Lima, Peru*, pages 63–68, October 2006.

V P. Österberg and T. Zhang. Multicast-Favourable Max-min Fairness – The Definition and how to Comply. Submitted to *IASTED International Journal of Computers and Applications*, 2007.

VI P. Österberg and T. Zhang. Fair Cost Sharing Among Multicast Receivers. In *Proceedings of 2nd IEEE International Conference on Digital Telecommunications (ICDT), San Jose, California, USA*, July 2007.

VII P. Österberg and T. Zhang. Bid-Based Cost Sharing Among Multicast Receivers. In *Proceedings of 4th ACM International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), Vancouver, British Columbia, Canada*, August 2007.

Papers by the author, but not included in the thesis:

1 D. Forsgren, U. Jennehag, and P. Österberg. Objective End-to-End QoS Gain from Packet Prioritization and Layering in MPEG-2 Streaming. In *Proceedings of 12th International Packetvideo Workshop (PV), Pittsburgh, Pennsylvania, USA*, April 2002.

2 P. Österberg, D. Forsgren, and T. Zhang. Receiver-Controlled Joint Source/Channel Coding on the Application Level, for Video Streaming over WLANs. In *Proceedings of 57th IEEE Semiannual Vehicular Technology Conference (VTC), Jeju, Korea*, volume 3, pages 1558–1561, April 2003.

3 T. Zhang, P Österberg, and Y. Xu. Multicast-Favorable Max-Min Fairness – A General Definition of Multicast Fairness. In *Proceedings of 1st IEEE International Conference on Distributed Frameworks for Multimedia Applications (DFMA), Besançon, France*, pages 239–244, February 2005.

4 P. Österberg, T. Zhang, and M. Gidlund. Bandwidth Allocation in Broadband Access Networks. In *Proceedings of 12th European Conference on Networks & Optical Communications (NOC), Stockholm, Sweden*, pages 525–532, June 2007.

List of Figures

List of Tables

Terminology

Abbreviations and Acronyms

AAP	multicast Address Allocation Protocol
ACK	ACKnowledgment
AFX	Animation Framework eXtension
ALM	Application-Layer Multicast
ARP	Address Resolution Protocol
ARQ	Automatic Repeat reQuest
AS	Autonomous System
ATM	Asynchronous Transfer Mode
BB LSD	Bid-Based Link Split Downstream
BFRD	Bottleneck-Feedback and Receiver Dependent
BGMP	Border Gateway Multicast Protocol
BU	Bitrate Unit
CBT	Core Based Trees
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CU	Cost Unit
DCT	Discrete Cosine Transform
DiffServ	Differentiated Services
DM	Dense Mode
DS	Differentiated Service
DTM	Dynamic synchronous Transfer Mode
DVB	Digital Video Broadcasting
DVMRP	Distance Vector Multicast Routing Protocol
ELSD	Equal Link Split Downstream
ETS	Equal Tree Split
EXPRESS	EXPlicitly REquested Single-Source
FEC	Forward Error Correction
FFRD	Full-Feedback and Receiver Dependent

FGS	Fine Granularity Scalability
FTP	File Transfer Protocol
FVV	Free-Viewpoint Video
GOP	Group Of Pictures
GW	GateWay
HDTV	High-Definition TeleVision
HTTP	Hypertext Transfer Protocol
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IGMPv0	Internet Group Management Protocol version 0
IGMPv1	Internet Group Management Protocol version 1
IGMPv2	Internet Group Management Protocol version 2
IGMPv3	Internet Group Management Protocol version 3
IHL	Internet Header Length
IntServ	Integrated Services
IP	Internet Protocol
IPTV	Internet Protocol TeleVision
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISO	International Organization for Standardization
ISP	Internet Service Provider
ITU-T	International Telecommunication Union – Telecommunication standardization sector
IU	Information Unit
IX	Internet eXchange point
LAN	Local Area Networks
LCD	Liquid Crystal Display
LogRD	Logarithmic Receiver Dpendent
MAAS	Multicast Address Allocation Server
MAC	Medium Access Control
MADCAP	Multicast Address Dynamic Client Allocation Protocol
MAGMA	Multicast & Anycast Group Membership
MALLOC	Multicast Address Allocation Architecture
MASC	Multicast Address-Set Claim
MBGP	Multiprotocol extensions for Border Gateway Protocol 4
MBone	Multicast Backbone
MC	Marginal Cost
MFMF	Multicast-Favorable Max-min Fairness
MOSPF	Multicast extensions to Open Shortest Path First
MPEG	Moving Picture Experts Group

MPLS	MultiProtocol Label Switching
MSDP	Multicast Source Discovery Protocol
MSEC	Multicast SECurity
MTU	Maximum Transmission Unit
MU	Monetary Unit
MV	Motion Vector
MVC	Multiview Video Coding
NACK	Negative ACKnowledgment
NAT	Network Address Translation
PAL	Phase Alternating Line
PAT	Program Association Table
PCR	Program Clock Reference
PDAM	Proposed Draft AMendment
PID	Packet IDentifier
PIM-DM	Protocol Independent Multicast – Dense Mode
PIM-SM	Protocol Independent Multicast – Sparse Mode
PMT	Program Map Table
PSI	Program Specific Information
QoS	Quality of Service
QoS-D ETS	QoS-Dependent Equal Tree Split
QoS-D LSD	QoS-Dependent Link Split Downstream
RI	Receiver Independent
RLM	Receiver-driven Layered Multicast
RMS	Root Mean Square
RMT	Reliable Multicast Transport
RPF	Reverse Path Forwarding
RSVP	resource ReSerVation Protocol
RTCP	RTP Control Protocol
RTP	Real-time Transport Protocol
SDTV	Standard-Definition TeleVision
SH	SHapely value
SM	Sparse Mode
SNR	Signal-to-Noise Ratio
SRTP	Secure RTP
SSM	Source-Specific Multicast
SVC	Scalable Video Coding
TCP	Transmission Control Protocol
TDM	Time-Division Multiplexing
TFMCC	TCP-Friendly Multicast Congestion Control
TTL	Time To Live
TU	Time Unit
UDP	User Datagram Protocol
VLAN	Virtual Local Area Network

VoD	Video on Demand
2D	Two Dimensional
3D	Three Dimensional
3DVO	Three Dimensional Video Objects

Notations

Notations related to bandwidth allocation

$\mathbf{B}$	a bandwidth allocation to the sessions in $\mathbf{N}$
$\mathbf{B}_{i,j}$	a vector containing the bandwidths allocated to the receivers in $\mathbf{R}_{i,j}$, sorted in ascending order together with a zero element
$\mathbf{b}_{i,k}$	the rate at which $r_{i,k}$ receives data
$\mathbf{C}$	the set of J link capacities in $\mathbf{N}$
c_j	the capacity of link l_j
$\mathbf{L}$	the set of J links in $\mathbf{N}$
l_j	the j^{th} link in $\mathbf{N}$
$\mathrm{M}()$	the multicast-favorable function
$\mathbf{N}$	network with a set of sessions $\mathbf{S}$, links $\mathbf{L}$, and link capacities $\mathbf{C}$
$\mathbf{R}_i$	the set of receivers of session s_i
$\mathbf{R}_{i,j}$	the receivers in $\mathbf{R}_i$ that are located downstream link l_j
$r_{i,k}$	the k^{th} receiver of session s_i
$\mathbf{S}$	the set of I sessions in $\mathbf{N}$
s_i	the i^{th} session in $\mathbf{N}$
$\mathrm{U}_i()$	the utility function of session s_i
$\mathrm{V}(\mathbf{B}_{i,j})$	the multicast-favorable value of $\mathbf{B}_{i,j}$

Notations related to cost allocation

$C_d^q()$	the total cost allocated to downstream receivers of QoS^q
$C_u^q()$	the total cost allocated to upstream receivers of QoS^q
$\mathbf{c}$	the set of additional costs c^q, for all QoS levels
c^q	the additional cost for providing QoS^q, when compared to that of QoS^{q-1}
$f_d^q()$	the share of c^q, that is allocated to downstream receivers
$f_u^q()$	the share of c^q, that is allocated to upstream receivers
n_d^q	the number of downstream receivers of QoS^q
n_u^q	the number of upstream receivers of QoS^q
QoS^q	the q^{th} QoS level
$\mathbf{z}_d$	the set of receiver numbers z_d^q, for all QoS levels
$\mathbf{z}_u$	the set of receiver numbers z_u^q, for all QoS levels

z_d^q	the number of downstream receivers that utilize the information of QoS^q
z_u^q	the number of upstream receivers that utilize the information of QoS^q

Chapter 1

Introduction

Media-streaming services are gaining in popularity, spanning from websites like YouTube[1] where end users can upload their own personally made material onto streaming servers, to *Internet protocol television* (IPTV) distribution of proprietary high-quality content. The latter category obviously has high bandwidth requirements, but the wide employment of the former also results in a substantial network load.

At the same time, bandwidth is a limited resource even in today's high-speed fiber-optical communication networks. The ever-growing interest in video-streaming services, which offer higher and higher quality, will certainly test the network capacity. Multicast transmission is one technique that can mitigate these effects, however it is not deployed to its full extent. The work presented within this thesis is therefore produced with the intention to increase the deployment of multicast.

1.1 Background and Problem Motivation

The ever-growing interest in high-quality media-streaming services is bound to push the network capacity to its limit. The video-data payload may be reduced by efficient compression techniques, such as those implemented in H.264 [4]. But if the visual quality is maintained at an acceptable level, the remaining data will still give rise to a considerable bitrate. Figure 1.1 gives a prediction of the increased demand of bandwidth due to transmission of video, when compared to that of data and telephony [5].

IPTV is the main current high-quality video service, which is typically performed by the *Internet service providers* (ISPs) within their own closed networks where they

[1] YouTube [1] is the leading video-streaming site and streamed 100 million videos per day in July 2006, which corresponded to 60 percent of all videos watched online, and thereby held 29 percent of the U.S. multimedia entertainment market [2]. YouTube still had a 60 percent share of the U.S. online video market in June 2007 [3].

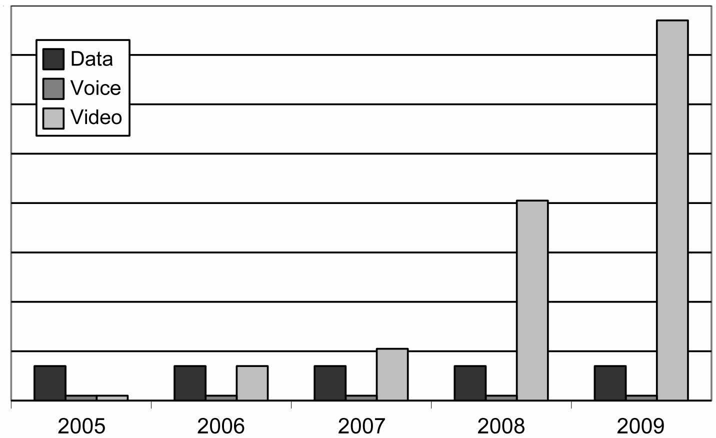


Figure 1.1: The expected demand of bandwidth due to transmission of video, when compared to that of other data and telephony.

are in control of the entire infrastructure [6]. In the future, service providers probably want to increase the number of potential customers by delivering TV services over Internet. Joost [7] is already trying to achieve this. Further, such techniques could enable the video-content producers to circumvent the traditional distributors, such as TV companies, IPTV providers, or even video stores, and instead stream the content themselves directly from a single place somewhere on the globe.

In the not too distant future, the employment of *high-definition television* (HDTV) will increase the bandwidth required for a video stream by a factor of three or more, when compared to *standard-definition television* (SDTV) [8]. If that is not enough, three-dimensional (3D) and *free viewpoint video* (FVV) are on the horizon with even higher requirements on the bandwidth [9]. According to [10], LG were already producing 3D displays in January 2007 and furthermore had plans to introduce 3D IPTV in the near future.

Multicast is a technique that can mitigate the increased demands for bandwidth, since it is more bandwidth efficient than unicast in scenarios where many receivers are simultaneously interested in the same data. This is certainly the case for popular live content, e.g. big sport events such as the FIFA World Cup and the Olympic Games, and music galas such as the Eurovision Song Contest and the MTV Video Music Awards. The reason why multicast is bandwidth efficient is that the receivers of a multicast session share the resources through a common transmission tree and data are only transmitted once along each branch. The use of multicast transmission can therefore yield huge savings in bandwidth on the network links corresponding to the more popular parts of the tree. Nevertheless, multicast transmission is not fully deployed at present.

This thesis focuses on *Internet protocol* (IP) Multicast, the implementation of the multicast concept on the IP routing level, although there exist overlay implementations targeting networks that are not IP multicast enabled. Unless otherwise is specified, multicast refers to IP multicast in the reminder of this thesis. IP multicast was already being discussed during the first half of the 80's [11], and the first version of the protocol standards was proposed during the second half, beginning with [12].

Nevertheless, the deployment of multicast has been very slow, especially when compared to the development of the Internet. In year 2000, Diot et al. [13] pointed out that "Multicast is included with the standard set of protocols shipped with most commercial routers, but most IP carriers have not yet enabled the service in their networks". They believed that the main reason was the poor support for commercial requirements such as group management, security, and billing. Some other reasons were the lack of established standards and the complex architectural design that required additional intelligence in core and edge routers, according to [14].

Since then, the main standards have become more established and many of the commercial requirements have been addressed in proposals made by the Internet Engineering Task Force (IETF) Reliable Multicast Transport (RMT) , Multicast Security (MSEC) , and Multicast & Anycast Group Membership (MAGMA) working groups. However, the architectural complexity remains. The study [15], presented in 2003, showed that the multicast infra structure was actually gradually shrinking. New networks and address spaces were being connected, but the number of old addresses no longer being advertised was larger. On the bright side, the remaining infrastructure was becoming more and more stable.

The slow deployment of multicast is still a burning issue. According to the *Multicast Status Web Page* [16], less than 3 percent of the *autonomous systems* (ASs), which constitute the Internet, were multicast enabled in the autumn of 2007[2]. To "push commodity ISP adoption of multicast routing", is for example one of Internet2[3] Multicast Working Group's goals for 2007 [17].

1.2 Overall Aim

The overall aim of this thesis is a more efficient usage of the bandwidth in IP networks. The scenario addressed is real-time video streaming, for which IP multicast is the most bandwidth-efficient transmission method. However, IP multicast it is not deployed to full extent, and the more specific aim of the thesis is therefore the proposal of policies and mechanisms that introduce incentives for a faster deployment of IP multicast.

[2]The percentage is calculated from the number of ASs with routing, respectively multicast routing, from the AmericaFree.TV AS. This may not cover the entire Internet, but is likely to give a representable view of the situation.

[3]Internet2 is a networking consortium, led by the research and education community since 1996. It provides its members with leading-edge network capabilities and partnership opportunities, which together facilitate the development and deployment of new Internet technologies.

1.3 Scope

This thesis focuses on multicast transmission with a single source, which is the most likely scenario for multicast distribution of high-quality multimedia content. The quality requirements for multiple-source multicast, typically used for applications such as video conferencing, are generally lower. The results will however be valid for any kind of multicast, independently of the number of sources. We further consider the data to be hierarchically encoded and the router nodes to be capable of priority dropping.

A certain minimum amount of data is typically required to produce a useful representation of some information. In hierarchically encoded information, the first and most important part of the data corresponds to such a coarse representation of the original information. The more data that are added, the more detailed this representation becomes. Hierarchical encoding can be performed on audio, image, and video information with a relatively low decrease in compression efficiency.

Priority dropping is a technique where the least important information is discarded first, in case of network congestion. This of course implies that the data are marked according to the importance. In combination with hierarchical encoding, priority dropping results in a very robust transmission scheme that facilitates graceful degradation of the received information. Another consequence is that data can potentially be transmitted to receivers with heterogeneous bitrate demands and/or capabilities, through a single multicast group.

1.4 Concrete Goals

The concrete goals of this thesis are proposals of policies and mechanisms that introduce incentives for the employment of multicast transmission. Such incentives might be:

- allocation of more resources to multicast sessions
- cheaper services for multicast receivers

This would make multicast transmission more attractive to the users, and consequently increase the demand for ISPs with multicast-enabled networks, thereby forcing the ISPs to deploy multicast functionality in their networks to keep up with the competition.

1.5 Outline

Chapter 2 consists of the background theory that this thesis is built upon. Section 2.1 covers video coding, whereas Section 2.2 deals with the distribution of multimedia content. In Section 2.3, packet-switched computer networks are briefly described and Section 2.4 contains a survey of the mechanisms and protocols involved in multicast transmission. Section 2.5 outlines different views of fair bandwidth allocation, in particular with regard to multicast traffic. Section 2.6 deals with existing cost-allocation mechanisms, also bearing multicast in mind, and Section 2.7 provides some basic notions of game theory. Game theory is applicable to a joint resource- and cost-allocation process.

In Chapter 3, the results within the area of fair bandwidth allocation to multicast sessions are outlined. Section 3.1 describes our definition of fair allocations and how these can be used to measure the fairness of other allocations. Section 3.2 contains the proposal of two bandwidth-allocation policies, which perform allocations close to those of the definition. In Section 3.3, the possible problem arising from what we have chosen to call *fragmented sessions*, are discussed.

Chapter 4 deals with the cost allocation for multicast transmission. In Section 4.1, previous work in the area is evaluated. A terminology for cost sharing among multicast receivers is then presented in Section 4.2. This terminology is used when two cost-allocation strategies are proposed in Section 4.3.

Chapter 5 contains the summary of the thesis, including discussions of possible future research topics and some concluding remarks.

1.6 Contributions

A definition of fair allocation of bandwidth, which favors multicast sessions, is presented in subsection 3.1.3. The contribution of this definition is that it supports scenarios involving multicast receivers with heterogeneous bit-rate demands or capabilities. Based on the definition, a measure of the fairness of alternative bandwidth allocations is proposed in 3.1.4. These results are presented in papers II and V.

Two bandwidth-allocation policies, which produce allocations reasonably close to those of the proposed definition of fair bandwidth allocation, are presented in subsections 3.2.1 and 3.2.2. They require differing amounts of information feedback, as described in 3.2.4, and perform accordingly as shown by an example in 3.2.3. The bandwidth-allocation policies are published in papers III and IV, and their feedback requirements are evaluated in paper V.

Different scenarios with fragmented sessions are discussed. They constitute a problem since what was originally a single session might end up by being considered to be multiple sessions, resulting in unfair bandwidth allocations. Some possible solutions to this problem are proposed in subsection 3.3.3. This topic is also addressed in paper I.

The allocation of costs related to the resources used by multicast transmission trees, is studied. Notations for the description of cost-allocation mechanisms, which split the cost of multi-rate multicast trees among the receivers, are introduced in Section 4.2. The notations were originally proposed in paper VI.

Following on from this, two such cost-allocation mechanisms are proposed. The first mechanism, which is presented in subsection 4.3.1, is optimally fair in the sense that it splits the costs of multicast trees between the receivers based strictly on their share of the resource usage. The second mechanism, presented in 4.3.2, allows discounts for poor and/or greedy receivers, motivated by the fact that more receivers can be served while the costs of the rest of the receivers, i.e. the non-poor and non-greedy, can simultaneously be reduced, as shown in subsection 4.3.3. The bandwidth-allocation policies are published in papers VI and VII.

The author is responsible for all the aforementioned results, although the definition of fair bandwidth allocations that favor multicast sessions builds on a previous version that was mainly designed by T. Zhang. The original version of the definition was presented in the paper listed as number 3 among the papers that are by the author, but not included in the thesis. As for the rest, T. Zhang has served as a partner for fruitful discussions.

Chapter 2

Background Theory

In this chapter, background theory within areas related to the topic of this thesis is outlined. Listed in their order of appearance, the sections of this chapter and their contents are:

1. **Digital video.** This section is intended to provide an understanding of the bandwidth requirements of video content. Some other aspects that are important to this thesis are also introduced, such as hierarchical and layered coding.
2. **Multimedia distribution.** The functionality necessary for real-time distribution of multiple media streams that belong together, such as audio and video. The section also outlines some existing protocols intended for multimedia distribution.
3. **The Internet network model.** A brief introduction to the basic components and handling of computer networks.
4. **Multicast.** A survey of the operations involved in IP multicasting and a short overview of overlay-multicast approaches.
5. **Fairness.** Outlines a number of criteria regarding what might be considered fair, both generally and in a multicast context.
6. **Cost allocation.** Describes mechanisms that can be used to distribute costs among customers. The emphasis is particularly on multicast receivers.
7. **Game theory.** Outlines some basic notions of game theory, which might be applicable to a joint resource- and cost-allocation process.

Those who are familiar with some of the above mentioned topics may find it possible to miss out the corresponding sections without an reduction in the understanding of subsequent chapters. For those facing time constraints, a more superficial reading of the chapter may still enable the majority of the thesis to be understood. In general, the first four sections comprise the basic background material, whereas the last three are more specific to and of greater importance for the thesis.

2.1 Digital Video

Video is merely a number of consecutive pictures. When the difference between adjacent pictures is sufficiently small, the switching between the pictures becomes invisible to the human eye, and the video is experienced as running in a fluent manner.

When video is to be transmitted over the Internet, it has to be digitized. Each picture is then divided into a number of small elements which are represented by a single color nuance and brightness. These elements are traditionally squared in shape and called pixels.

A video sequence typically corresponds to 25-30 pictures per second, and a large number of pixels per picture. The resolution of the current European SDTV standard *phase alternating line* (PAL) is for example 768×576 pixels, whereas HDTV implies 1280×720 or 1920×1080 pixels. This consequently results in a huge amount of raw video data, which in turn necessitates compression when the data are to be stored or transmitted. The compression can be performed in the spatial and/or temporal domain, as will be briefly described in subsections 2.1.1 and 2.1.2.

Subsection 2.1.3 covers the concept of scalable video coding, which is performed in such a way so that parts of the encoded data can be decoded into lower-quality representations of the original video. The compression and resultant video quality can thus be dynamically adjusted by means of the amount of encoded data that is used.

Subsections 2.1.4 and 2.1.5 cover 3D video and free-viewpoint video, which are not yet commercial services, but may be the next significant move for television after HDTV. According to [10], LG is already planning to introduce 3D IPTV.

Subsection 2.1.6 briefly outlines the evolution into the currently used video coding standards.

2.1.1 Spatial Video Coding

Spatial video coding, also called *intra coding*, refers to separate coding of the individual pictures in the video sequence. The simplest solution is to reduce the resolution, i.e. the number of pixels that are used to represent the pictures. However, there are other techniques, such as the *discrete cosine transform* (DCT), wavelet transform, or fractal coding, which can provide higher compression at lower degradation of the image quality [18].

The common denominator is that all these techniques utilize information-theoretical properties of the pictures. For example, closely spaced pixels are correlated, and most of the energy of a picture therefore lies in the low spatial frequencies [19]. Some color nuances appear more frequently than others. The incapabilities of the human visual system might also be exploited. One such limitation is that the contrast sensitivity of the human visual system falls off at high frequencies [20].

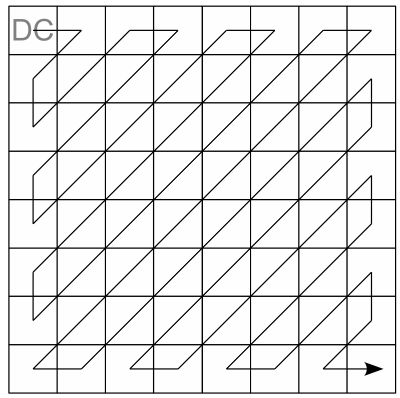


Figure 2.1: The zigzag scan order of MPEG-2.

DCT-Based Compression

Taking the MPEG-2 video coding standard [21] as an example, the intra coding is based on the *two-dimensional* (2D) DCT as described in [22]. The 2D DCT transforms the input matrix, i.e. each 8-by-8 pixel block of the picture, into the frequency domain. One property of the 2D DCT is that the average of the input pixels, i.e. the DC component, is located in a corner of the output matrix, say the upper left. Further, the lowest horizontal, vertical and diagonal frequencies are represented by the neighboring elements. The further away from the DC component, the higher are the frequencies that the matrix elements represent.

As mentioned previously, most of the energy of a picture lies in the low spatial frequencies, and the contrast sensitivity of the human visual system decreases at high frequencies. The elements in the output matrix of the 2D DCT are therefore scanned in a diagonal zigzag order[1], see Figure 2.1, such that the elements that correspond to the lower and more important frequencies are to be found at the beginning of the vector. These elements are then quantized with smaller quantization steps than the elements that correspond to higher frequency components. This procedure typically produces a large number of zeros at the end of the vector, and they can be eliminated by means of run-length coding. In addition to the compression ratio achieved by the run-length encoding, entropy coding is used for the representation of the remaining coefficients. [18]

2.1.2 Temporal Video Coding

In video with a decent frame rate, temporally neighboring pixels are highly correlated [23]. The technique that utilizes this property is called *temporal video coding* or

[1] There is also a choice of an alternate scanning order that is more adapted to interlaced video, where the vertical correlation is reduced since every second transmitted picture frame only contains the odd or even lines of pixels of the original picture [18].

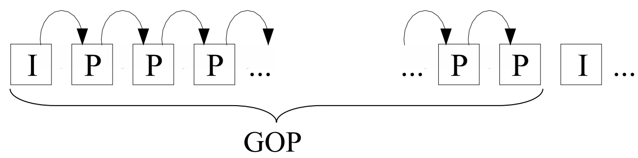


Figure 2.2: A group of pictures, consisting of one I and a number of P pictures, with the picture dependencies represented by arrows.

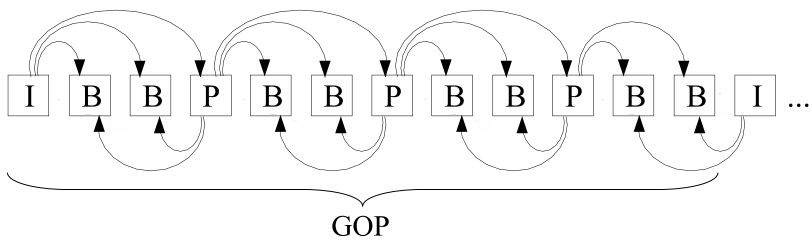


Figure 2.3: A common MPEG-2 GOP structure, consisting of one I, three P, and eight B pictures, with the picture dependencies represented by arrows.

inter coding. Inter coding is achieved by initially transmitting the entire first picture, which is preferably intra coded. Then the second picture is predicted from the first picture and only the difference between them is transmitted. Instead of the third picture, the difference between it and the second picture can be transmitted, and so on. These predicted pictures are denoted as P pictures, while the intra-coded pictures are denoted as I pictures.

The I pictures are typically transmitted periodically, see Figure 2.2, to limit the propagation of errors that may arise if parts of the video data are lost or corrupted, e.g. during transmission. The I pictures also provide entry points to the video stream. An I picture and the P pictures predicted from it are called a *group of pictures* (GOP).

To gain higher compression ratios, it is sufficient to transmit the difference between each group of four blocks, called a *macro block*, in a picture and the most similar macro block in the preceding picture, within a specified search radius. This is called motion compensation and the higher compression ratio is attained from the increased correlation brought by this method. To be able to reference different macro blocks of the preceding picture, overhead in the form of *motion vectors* (MV) must be transmitted. A MV is simply the difference between the spatial positions of the two macro blocks.

A method that is able to increase the compression ratio even further involves the use of bidirectionally predicted pictures, called B pictures. In contrast to the P pictures, which are only forward predicted from the preceding picture, these pictures are predicted both from the preceding I or P picture and the succeeding I or P picture. A common MPEG-2 GOP structure, including B pictures, is shown in Figure 2.3.

2.1.3 Scalable Video Coding

Scalable video coding means that the data rate of the video can be adjusted after encoding. If the video is *hierarchically encoded*, some information is more important than other. The output will for example contain base information that corresponds to a minimum-quality representation at a lowest possible data rate. This base information has to be obtained in order to decode the video. More information may then be added to enhance the video quality, but only in the correct hierarchical order. Hierarchically encoded video is often divided into several layers, and referred to as *layered video*.

The main field of application for layered video is multicast transmission to users with heterogeneous requirements, capabilities (e.g. rendering capacity), and/or network conditions [24]. Layered video can be beneficially accompanied by packet prioritization together with priority dropping in router nodes [25], and differentiated error protection [26]. Scalable video coding can be performed in the temporal and/or spatial domain, or in terms of the *signal-to-noise ratio* (SNR), as described briefly in the remainder of this subsection.

Temporal Scalability

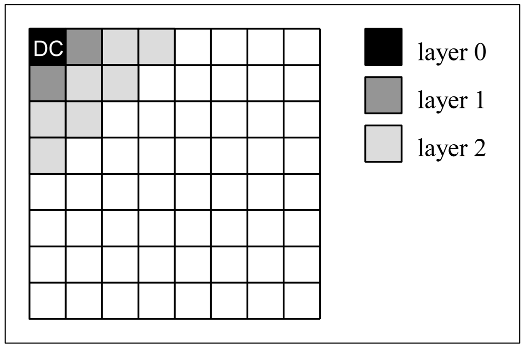
Temporal scalability implies that there is the ability to vary the picture rate. In the case of the MPEG-2 GOP structure shown in Figure 2.3, three different picture rates can easily be achieved. Transmission of all the pictures provides the full picture rate, whereas transmission of only the I and P pictures offers a third of the full picture rate and transmission of only the I pictures results in a twelfth of the full picture rate. The rate may be reduced further by only transmitting every second or third I picture, etc. Other picture rates may also be achieved for this GOP structure, but not at a constant picture rate.

Spatial Scalability

Spatial scalability refers to a coarse resolution video, whose resolution can be enhanced through additional information. It might prove difficult to implement this efficiently, and it is mainly useful when the users have heterogeneous display capabilities. See [27] for more information with regards to spatial scalability.

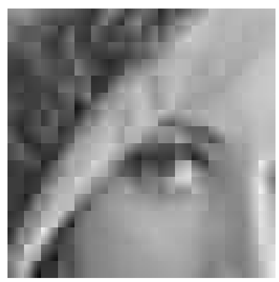
SNR Scalability

SNR scalability is also typically performed in the spatial domain, but it does not necessarily concern the resolution. Any method that can produce separate quality levels at different data rates could be utilized. When using DCT compression, one solution is to start with coarse quantization of the coefficients and then provide different levels of refinement [28]. Another alternative is to partition the DCT coefficients into different groups, according to their importance [29]. See Figure 2.4 for an example



(a) Partitioning of the DCT coefficients of a block into three layers.

(b) Picture quality provided by layer 0.



(c) Picture quality provided by layer 0 and 1.

(d) Picture quality provided by layer 0, 1, and 2.

Figure 2.4: SNR scalability through partitioning of DCT coefficients, and its effect on the picture quality.

of such coefficient partitioning into three layers[2], and the effect it has on the picture quality.

2.1.4 3D Video

In addition to the experience provided by ordinary 2D video, 3D video offers some added sense of depth in the scenes, i.e. the distance to separate objects appears to differ. There are a number of visual depth cues that provide assistance in assessing the distance to an object [18].

- **accommodation** – the change in focal length of the eye lens
- **convergence** – the relative difference in direction of the eyes
- **binocular disparity** – the dissimilarity in the views of the eyes
- **motion parallax** – the change in views due to motion

[2]The base layer must also contain the rest of the necessary information, such as MVs and overhead.

- **size** – the actual size of many objects are known.
- **linear perspective** – the change in size of an object due to distance
- **aerial perspective** – distant objects appear hazy and bluish[3]
- **shading** – objects farther from a light source appear darker
- **shadowing** – gives hints regarding the relative positions of objects
- **color** – bright indicates closer than dark
- **texture gradient** – gives indications of distance and relative positions

The first four cues cannot be fulfilled by ordinary 2D video, whereas the later can. There are a number of methods that have been proposed in order to stimulate more of these cues. They are quite thoroughly outlined in [18], and some of them will now be briefly described.

Stereoscopic Methods

Stereoscopic methods require the use of special viewing devices such as glasses. These methods are used to produce binocular disparity.

One alternative is *color separation*, where images intended for one eye are tinted red, whereas images intended for the other eye are tinted green (or blue). Glasses with a red filter on one eye and a green (or blue) filter on the other are then used to filter out the images intended for the other eye.

Another alternative is *passive separation*, where images intended for one eye are polarized one way and the others are polarized perpendicularly. This of course requires special display equipment. Glasses with differently polarized lenses are then used to block images intended for the other eye.

A third alternative is *active separation*, which uses active glasses that block the eyes interchangeably, synchronized with the display update frequency. It can therefore be performed on an ordinary viewing system, as long as the material is correctly encoded for the purpose.

Autostereoscopic Methods

Autostereoscopic methods require no visual aids. Two such methods are presented here, both of which produce binocular disparity, but are also able, to some extent, to achieve motion parallax.

Parallax-based systems can use a plate with vertical slits in front of a display that shows alternating vertical strips for the left and right eyes. The slits in the plate

[3]This is due to scattering of light from the sky (which is mainly blue), by molecules and larger particles, into the line of sight of the viewer.

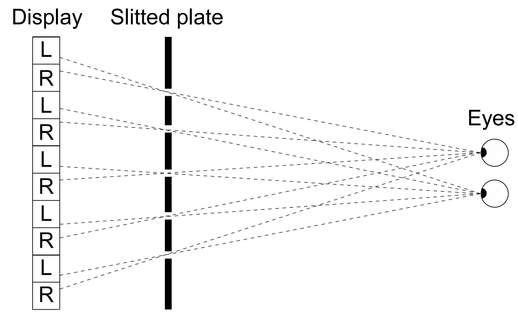


Figure 2.5: A parallax-based system with a slitted plate in front of the display.

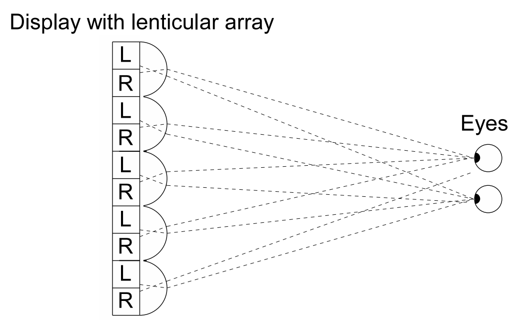


Figure 2.6: A lenticular system with half-cylinder shaped lenses in front of the display.

force the left eye to see only its intended strips and vice versa. The principle is visualized in Figure 2.5. Another parallax-based solution is to use thin vertical light sources behind a *liquid crystal display* (LCD) that shows alternating strips in the same manner as before. Both alternatives require a fixed distance between the viewer and the display, and they have a problem with low light intensity, owing to the blocking of the light and limited light source.

Lenticular systems typically use an array with vertically positioned long and narrow half-cylinder-shaped lenses in front of a display that shows alternating strips in a manner similar to that for the parallax-based systems. The lenticular array refracts the light from each strip to the intended eye. See Figure 2.6 for a principal sketch.

The viewing distance of lenticular systems is not as critical as that for parallax-based systems. Both parallax-based and lenticular systems can be designed to provide more than two views to achieve motion parallax. HD displays with lenticular arrays providing 9 respectively 25 horizontal views are currently produced by Philips [30] and LG [31], although they are not to be considered as consumer products at the present time. Further, lenticular arrays can be made of half-spherical lenses, to also provide vertical motion parallax.

3D-Video Encoding

Intuitively, a larger amount of data is required for the representation of 3D as compared to 2D video. For raw data the ratio is directly proportional to the number of views. However, adjacent views are highly correlated and specific *multiview video coding* (MVC) algorithms are therefore significantly more efficient than separate compression of individual views.

Further, MVC schemes must also record the camera parameters, such as position and direction.

2.1.5 Free-Viewpoint Video

FVV provides a selection of viewpoints and viewing directions within a certain range [9]. These parameters are typically specified at the time of rendering, and not at the time of recording [32].

FVV can be represented in terms of the shape and textures of *3D video objects* (3DVO), as is performed in computer graphics models. Another alternative is *point cloud* representation, where objects are made up from a number of points with 3D coordinates. [9]

FVV is related to 3D video in the sense that both techniques involve multiple views of one scene. FVV does however necessitate much larger spacing between cameras, and the correlation between adjacent views is therefore much smaller. Hence, the MVC techniques mentioned in subsection 2.1.4 can be used for both FVV and 3D video, but FVV data can generally not be compressed to the same extent as 3D video.

2.1.6 Video Coding Standards

So forth, the frequently used MPEG-2 standard has served as an example for showing different aspects of video coding. However, there exist other competing video-coding standards. The MPEG and H.26X standards will now be outlined briefly. The H.26X standards only deal with video coding, whereas the MPEG standards typically include support for related issues, such as audio coding, content synchronization, and transmission. However, this section purely focuses on the video coding.

ISO/IEC MPEG

MPEG is a working group of the *International Organization for Standardization*[4] (ISO) and *International Electrotechnical Commission*[5] (IEC) organizations. The standards are named MPEG-X, where X is a number. However, all values of X do not correspond

[4] ISO is a network of the national standards institutes of 157 countries.

[5] IEC is a global organization that prepares and publishes international standards for all electrical, electronic and related technologies.

to video coding standards and some values are missed out or saved for later use. E.g. MPEG-1 through MPEG-4 are video coding standards, whereas MPEG-7 describes a *multimedia content description interface* and MPEG-21 defines a *rights expression language*.

MPEG-1 [33] is a coding standard for progressive video that is based on H.261 and achieves the compression through the use of DCT in conjunction with I, P, and B pictures. Progressive video is rendered picture by picture, whereas in interlaced video the even and odd numbered lines are rendered alternately. MPEG-1 was originally aimed at resolutions of around 350×250 at 25-30 frames per second, where it produces bitrates of approximately 1.2 Mbps, although it supports higher resolutions and bitrates. MPEG-1 video coding is used for Video CD (VCD) and for some DVDs. The MPEG-1 audio layer 3 is the popular audio format that is also known as MP3.

MPEG-2 [21] is the basis of most digital television and DVD formats. It is backward compatible with MPEG-1, but also provides support for interlaced video. It is not optimized for bit-rates below 1 Mbps, but outperforms MPEG-1 at 3 Mbps and above. The systems part of MPEG-2, will be described further in subsection 2.2.2.

MPEG-3 was aimed at HDTV, but was abandoned in favor of performance-enhancing extensions to MPEG-2.

MPEG-4 [34] consists of a number of parts that describe different aspects of the system. There are two parts that particularly concern the video coding. Part 2 is called *Visual* and contains the *advanced simple profile* (ASP), which for example is adopted by the DivX and Xvid codecs. Part 10 was developed to support twice the compression of part 2 and describes a number of profiles that are collected under the name *advanced video coding* (AVC). Applications that support MPEG-4/AVC include Blu-ray Disc and HD DVD. A great deal of effort has been put into the scalability of MPEG-4. In 2000 there was a *proposed draft amendment* (PDAM) of *fine grain scalability* (FGS) [35] for the ASP. FGS has been described and further developed since then in a vast number of publications, but the draft amendment was never approved. However, according to [36] a *scalable video coding* SVC standard has now been approved as an amendment 3 of AVC. The compression efficiency of SVC is superior to that of FGS, and is said to be "...very high and hardly distinguishable from "single-layer" AVC codecs in most operation modes". Further, [36] also mentions an AVC-based PDAM for MVC. Part 16 of MPEG-4, the *animation framework extension* (AFX), already supported both 3DVO and point-cloud representation [37].

ITU-T H.26X

The *International Telecommunication Union*[6] *– Telecommunication Standardization Sector* (ITU-T) is a body that among other things specifies video coding standards or recommendations. These are numbered H.26X, where the X is a consecutive number starting at 1.

[6]United Nations agency for information and communication technology.

H.261 [38] was the first practical digital video coding standard when it was presented in 1988. Subsequent international video coding standards such as H.262, H.263, H.264, MPEG-1, MPEG-2 and MPEG-4 have been based on its design with motion compensated inter-picture prediction and spatial transform coding with scalar quantization, zig-zag scanning and entropy coding.

H.262 was developed from H.261 and it is equal to the video part of MPEG-2.

H.263 [39] is a further development of H.261 and H.262, which has resemblances with MPEG-4/ASP. It was designed as a low-bitrate compressed format standard for video conferencing. Most Flash Video[7] content is encoded by a variant of H.263.

The H.264 standard was developed in collaboration with ISO/IEC and it equals MPEG-4/AVC.

Summary

Most digital television systems currently provide MPEG-2 video with SDTV resolution at bitrates in the order of 4 Mbps. Future systems will likely be based on MPEG-4/AVC, which can encode SDTV with the corresponding quality at approximately half the bitrate of MPEG-2. Existing HDTV services are typically already based upon MPEG-4/AVC, which requires approximately 6.5 Mbps for interlaced video at 1920×1080 resolution. [8]

Additionally, MPEG-4/AVC provides efficient support for scalable video coding, which is crucial for the ideas presented in Chapter 3. The standard also covers techniques for 3D video and FVV, which might be the next big digital-video services to be introduced, following the deployment of HDTV.

2.2 Multimedia Streaming

Multimedia streaming refers to real-time delivery of primarily video and audio. The real-time concept is somewhat diffuse in this context. Receivers might have considerable buffer space to compensate for fluctuations in the arrival rate. A reasonable minimum requirement could however be that the media should start to be experienced before the transmission ends, and that the average transmission rate is close to the actual data rate of the media. A certain kind of media-streaming services are called IPTV, and this concept is addressed in subsection 2.2.1.

There are a number of mechanisms that are required for multimedia distribution over computer networks. As indicated by the name, multimedia consists of different media types such as audio, video, and data. Additionally, there might also exist several alternative instances of a particular media type, e.g. multiple audio streams may associated with the same video stream, providing different languages or audio formats. Each such instance is called an elementary stream. The elementary streams

[7] Flash Video content is distributed by sites such as Google Video, MySpace, Yahoo! Video, and YouTube, and decoded with the proprietary Adobe/Macromedia Flash Player.

that belong together have to be multiplexed somehow when they are to be transmitted through a single channel.

The two most popular methods to achieve this are *time-division multiplexing* (TDM) and *packet multiplexing* [18]. In time-division multiplexing, time slots are periodically allocated to the different streams. Packet multiplexing implies that each packet belongs to a single media stream, and the different packet types are interleaved at transmission. Either way, there is a need for synchronization of the different streams at the receiver.

In addition, the information might be encrypted to provide conditional access, i.e. to prevent unauthorized eavesdropping or perhaps, more likely, illegal file sharing. A multimedia transmission scheme might also include error-detection and/or correction capabilities.

The MPEG-2 *transport stream* (TS) is the most widely used multimedia transmission protocol, and it is described in subsection 2.2.2. However, there appears to be a movement in favor of a migration to the native *real-time transport protocol* (RTP), which would reduce the overhead [40]. Native RTP is outlined in subsection 2.2.3, and it is compared with MPEG-2 TS in 2.2.4.

2.2.1 IPTV

IPTV services are forecasted to have 300 million subscribers in 2016, to be compared with only a few million in 2002 [41]. However, the IPTV concept is not distinctly defined, hence which services are to be considered as IPTV? According to the name *Internet protocol television*, the basic requirement for a service to be considered as IPTV would be that it concerns video streaming over IP networks.

IPTV is however generally seen as a competitor to, or a replacement for, traditional television distributed over cable, terrestrial or satellite connections. This limits the scope to high-quality video, i.e. with SDTV resolution or higher, which is hard to distribute reliably over the Internet. Consequently, these services are typically offered on closed distribution networks where the service providers manage the entire infrastructure [6]. Video of lower quality, delivered over Internet by service providers such as Joost [7], is therefore generally not considered as IPTV, but rather *Internet video*, and is not predicted to represent a major threat to IPTV for several years [42].

Judging by the marketing efforts of IPTV providers such as Telia [43] and Bredbandsbolaget [44], the main IPTV application currently seems to be *live TV*, which is similar to traditional TV in the sense that programs start and end at specific point in time. The programs are also bundled into channels, which the users subscribe to. Each TV channel can therefore be transmitted on an individual multicast group, comprising the subscribing users.

However, in contrast to traditional TV, simultaneous IPTV programs do not have to be transmitted on separate physical channels. IPTV can therefore be delivered as *video on demand* (VoD), where individual programs are delivered upon request.

This scenario requires more network resources than live TV, since the number of simultaneous receivers typically will be low, which will reduce the usefulness of multicast transmission. VoD programs may however be offered at periodical starting times to create groups of users that can share multicast groups, and the periodicity could be a function of the popularity of the program.

The concept of letting customers subscribe to individual programs starting at specific times is known as *pay-per-view* and was introduced as a service by a cable TV operator as early as in 1974 [45]. However, all programs were still delivered to all customers, hence the total number of simultaneous programs was limited by the number of physical channels and the access to content was controlled by means of scrambling.

2.2.2 MPEG-2 TS

The system part (part 1) of MPEG-2 [21], defines two container formats. The one of interest for this thesis is the TS, which is designed to carry digital video and audio over somewhat unreliable media. The other container format is the *program stream* (PS), which is designed for more reliable media[8]. The MPEG-2 TS is commonly used in broadcast applications, such as *digital video broadcasting* (DVB) and ATSC. The TS is quite complex and only the basic functionality is described here. All other details can be found in the standard.

The MPEG-2 TS utilizes packet multiplexing of packets that are 188 bytes long. Each packet has a TS header that includes a 13-bit *packet identifier* (PID), which specifies the different streams.

Packets with a PID value of 0 contain the *program association table* (PAT), which defines the program number of each program within the TS and the PIDs that are used by packets that carry the *program map tables* (PMT) for those programs. The PMTs specify the PIDs of the elementary streams that are associated with each program. The PMTs also indicate the PID of the packets that carry the *program clock reference* (PCR) for each program. The PCR is primarily used for rate control at the transmitter.

Conditional access to programs encoded in the TS is supported by the MPEG-2 standard, although no actual conditional access mechanisms are specified. Programs and elementary streams, or parts thereof, may be scrambled for conditional access. However, *program specific information* (PSI) such as the PAT and PMTs should not be scrambled. The PID value 1 is used by packets that carry the *conditional access table* (CAT), which defines the type of scrambling used and the PID values of streams that contain information regarding the conditional access management and authorization levels of specific decoders.

The MPEG-2 TS supports error detection in terms of a 32 bit cyclic redundancy check (CRC) of the PAT, PMTs, and CATs.

[8]The MPEG-2 program stream is for example used in the SVCD and DVD standards.

2.2.3 Native RTP

In native RTP [46], each elementary stream is transmitted in a separate session that is identified by its destination IP address and port number. The sessions are then packet multiplexed. Synchronization of, for example, audio and video is achieved using timing information carried in the *RTP control protocol* (RTCP) packets for both sessions.

RTCP performs a number of functions, of which the primary one is to provide feedback on the quality of the data transmission. This is achieved through parameters such as the fraction and number of lost packets and inter-arrival jitter, which can be calculated thanks to the sequence number in the RTP headers. RTCP also carries a transport-level identifier for an RTP source, called the canonical name, which keeps track of each participant. The canonical name might also be used to associate multiple data streams from a given participant in a set of related RTP sessions, e.g. audio and video streams that belong together.

A profile of RTP also exists aimed at conditional access. It is called *secure RTP* (SRTP) [47] and provides confidentiality, message authentication, and replay protection to the RTP and RTCP traffic.

2.2.4 Comparison between MPEG-2 TS and Native RTP

The native RTP requires less header information, and thereby produces approximately 29% lower overhead when compared to MPEG-2 TS. If MPEG-2 TS is transmitted using RTP, the difference is approximately 41%. [40]

There is also a greater option for intelligent mapping of video information directly to RTP packets than if the MPEG-2 TS is used. This can be utilized to minimize the negative effects from packets that are lost during transmission. Further, audio frames might be interleave in RTP packets to trade one long period of silence for a number of short ones, in the case a packet is lost. This is not possible with the MPEG-2 TS. [40]

Another drawback of MPEG-2 TS is the inflexibility to meet demands for different sets of elementary streams. Users might for example request different audio streams in conjunction with the same video stream. With the MPEG-2 TS there are basically two alternatives, either all streams are embedded in a single TS, which means that all receivers obtain all elementary streams, or one TS has to be compiled for each combination of streams to be offered [48].

2.3 The Internet Network Model

This section contains a basic introduction to the construction and handling of packet-switched computer networks. The intention behind the section is primarily to introduce some of the concepts used in the next section, which outlines multicast trans-

Table 2.1: The five-layer TCP/IP model.

layer	name	function
5	application	network access to applications
4	transport	end-to-end connectivity and reliability
3	network	logical addressing and routing
2	data link	physical addressing and medium access
1	physical	medium, signal and binary transmission

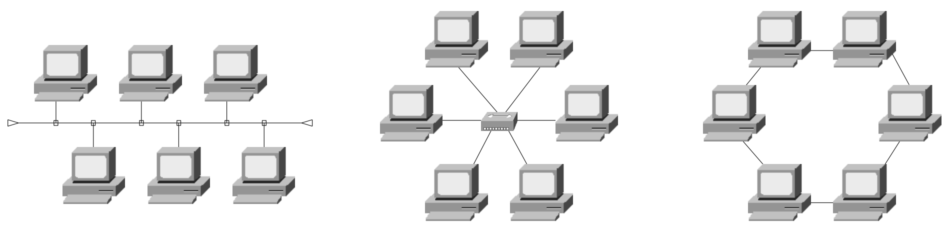


Figure 2.7: Example of three network topologies: a bus, star and ring network.

mission.

In packet-switched networks, the data are divided into packets that independently are transmitted to the receivers. The alternative to packet-switched networks is circuit-switched networks, which function as traditional telecommunication networks where a dedicated channel is allocated end-to-end. However, they are not deployed for data communication to any greater extent, although there exists some standards for circuit-switched data communication such as the *dynamic synchronous transfer mode* (DTM) [49]. There also exists schemes where virtual circuit switching is performed over packet-switched networks to provide service guarantees, e.g. *asynchronous transfer mode* (ATM) [50] and *multiprotocol label switching* (MPLS) [51].

The five-layer TCP/IP model divides the functionality of a computer network into five layers, each of which is responsible for a particular part of the communication. The layers and their functionality are listed in Table 2.1, with the most basic physical layer at the bottom, and the most abstract application layer at the top. The model is a useful tool to describe the handling of a computer network, and will be referenced on several occasions in the following subsections.

2.3.1 Computer Networks

A computer network is basically a number of computers that are physically interconnected to allow them to communicate with one another. The computers can be interconnected with cables in a number of different ways, as visualized in Figure 2.7, or they could be connected wirelessly. In either case, in order to facilitate one-to-one communication, there must be a common addressing architecture.

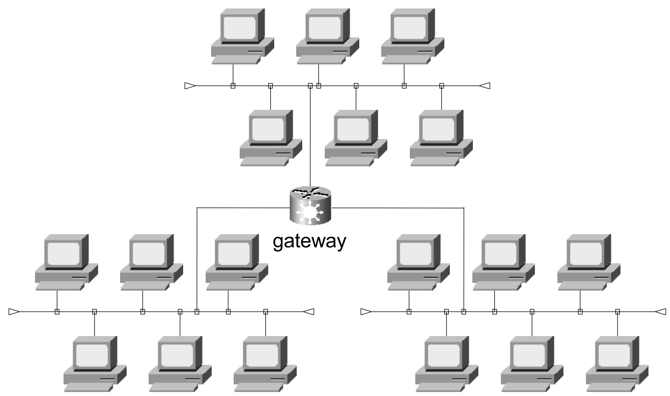


Figure 2.8: Three computer networks interconnected by a gateway.

In the network layer, end-to-end addressing is performed by the IP. *IP version 4* (IPv4) [52] is the most commonly used version, but its addresses are limited to 32 bits, and the address space is therefore becoming exhausted. This problem can be relieved somewhat by *network address translation* (NAT), which facilitates reuse of certain parts of the IPv4 address space [53]. Addresses that belong to these parts are called private and are only allowed for used within local networks. When communication is to be performed across the network boarders, these private IP addresses are translated into a few public addresses that are unique within Internet. The alternative to IPv4 is the *IP version 6* (IPv6) [54], which provides longer addresses and several other features. IPv6 was standardized in 1995, but the deployment has been rather slow. Unless otherwise stated, IP therefore refers to IPv4 in the remainder of this thesis.

The IP addresses are built up in a hierarchical way, such that the first part of the address specifies the network class[9], the second part specifies the network, and the remaining data identify the actual computer. However, IP addresses are unknown to the data link layer. Each device connected to the network has a physical *medium access control* (MAC) address. These are the addresses that the devices listen for on the network, to determine which packets are intended for them. Consequently, there has to be a mapping between IP and MAC addresses, and this is maintained by the *address resolution protocol* (ARP) table.

The packets can be transmitted directly to the destination if they are on the same local network as the source and the IP address exists in the source's ARP table. If the IP address does not exist in the table, the source must first send an ARP request broadcast message on the local subnet. The host with the given IP address sends an ARP reply in response, whereupon the source's ARP table can be updated and the packets can be transmitted directly to the destination.

[9]There are three classes of networks, A, B and C, of which the class A networks are the largest and class C the smallest.

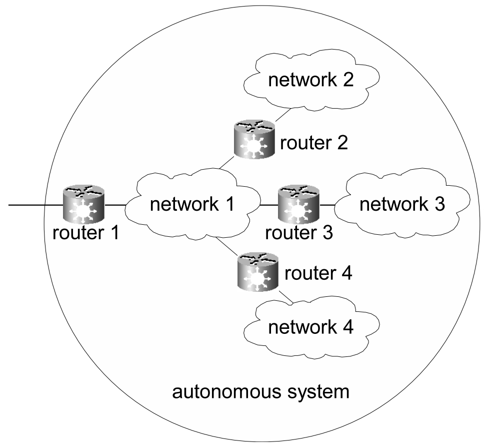


Figure 2.9: An autonomous system comprising four networks.

Interconnected Networks

A number of computer networks may be connected together by a *gateway* (GW), as shown in Figure 2.8. If the destination is on another network, the packets have to be transmitted through the GW. Such packets are therefore transmitted with the IP address of the final destination but the MAC address of the GW. The essential functionality of the GW is routing, although it can support mechanisms such as NAT and firewalling[10]. Routing of packets is assisted by a routing table that specifies the next hop router on the transmission paths to the destinations.

2.3.2 Autonomous Systems

An administrative domain is a part of the Internet that is controlled by a single network operator, typically an ISP or some organization. The concept of an administrative domain is closely related to that of an *autonomous system* (AS), which is defined as "...a connected group of one or more IP prefixes run by one or more network operators which has a SINGLE and CLEARLY DEFINED routing policy." [55].

The mentioned routing policy refers to the exterior routing that is performed between ASes. Exterior routing is, somewhat misleading, also called interdomain routing, and this is also the case in this thesis. The routing that is performed within an AS is analogously called interior or intra domain routing. In the AS in Figure 2.9, router 1 performs both interior and exterior routing, whereas router 2 through 4 only performs interior routing.

Each AS is allocated an *AS number* by the *Internet assigned numbers authority* (IANA). These numbers uniquely identify each network on the Internet and are used

[10] A firewall is a filter which restricts the access to the local network, thereby limiting the risk of intrusion.

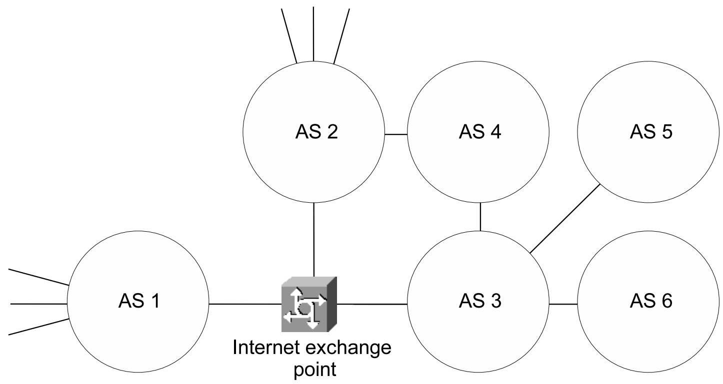


Figure 2.10: Six interconnected autonomous systems.

during exterior routing.

Interconnected Autonomous Systems

The Internet consists of a large number[11] of interconnected ASs. The ASs can be grouped into three categories, depending on their connections and operation:

- A *stub* AS is an AS that is only connected to one other AS.
- A *multihomed* AS is an AS that has connections to more than one other AS. This allows the AS to remain connected to the Internet in the event of a complete failure of one of their connections. However, this type of AS does not allow transit traffic, i.e. traffic between two other ASs, to pass through.
- A *transit* AS is an AS that provides connections through itself to separate networks. ISPs are always transit ASs, because they provide connections from one network to another.

In Figure 2.10, ASs 1 through 3 are transit ASs, AS 4 is a multihomed AS, whereas ASs 5 and 6 are stub ASs.

ISPs are connected together by *Internet exchange points* (IXs), where they exchange Internet traffic between their networks by means of mutual peering agreements. The primary purpose of an IX is to allow networks to interconnect directly, via the exchange point, rather than through one or more third party network. The main advantages of the direct interconnection are more bandwidth and lower latency, at a reduced cost.

[11] In the autumn of 2007 there were approximately 26,000 ASs with BGP routing [16].

<table>
<tr><td>version</td><td>IHL</td><td>DS</td><td colspan="2">total length</td></tr>
<tr><td colspan="3">identification</td><td>flags</td><td>fragment offset</td></tr>
<tr><td colspan="2">time to live</td><td>Protocol</td><td colspan="2">header checksum</td></tr>
<tr><td colspan="5">source address</td></tr>
<tr><td colspan="5">destination address</td></tr>
<tr><td colspan="5">...</td></tr>
</table>

Figure 2.11: Mandatory fields of the IPv4 header.

2.3.3 Network Protocols

There are a number of protocols that are involved in communication over computer networks. Some of the more common and important network protocols are now described, listed from the bottom of the TCP/IP model and upwards.

Data Link Layer Protocols

The data link layer protocols provide both physical addressing within the network and MAC mechanisms. The MAC mechanisms are used to detect or avoid collisions between packets sent over the network. In the case where collisions are detected, it is the responsibility of the protocol to perform retransmission. Two examples of data link layer protocols are Ethernet, the dominating *local area network* (LAN) technology that is standardized as IEEE 802.3 and uses *carrier sense multiple access with collision detection* (CSMA/CD), and the wireless standard IEEE 802.11 that uses *carrier sense multiple access with collision avoidance* (CSMA/CA).

Network Layer Protocols

The network layer protocols are encapsulated within the data link layer protocols. The IP is the network layer protocol that handles end-to-end addressing of packets over the Internet. As previously stated, the following description is restricted to IPv4, although the functionality can be considered as a subset to that of IPv6.

In Figure 2.11, the header transmitted at the beginning of all IPv4 packets is visualized. In addition to the self-explanatory host and destination addresses, the IPv4 header contains a number of other fields, of which a couple are explained here. The *Internet header length* (IHL) and *total length* fields contain information regarding the length of the header, respectively the entire packet. The *differentiated service* (DS) field may be used for requests of certain constraints in terms of delay, throughput and reliability [56]. The *time to live* (TTL) field specifies the maximum number of hops, i.e. the number of network links, that the packet may traverse. There is also a *protocol* field that specifies the higher level protocol encapsulated within the IP packet, and a *header checksum* field for detection of bit errors in the header.

Two other network layer protocols are the *Internet control message protocol* (ICMP) that is mainly used for feedback regarding delivery problems, e.g. unreachable destinations or timed out packets, and the already mentioned ARP. The *Internet group management protocol* (IGMP), which will be discussed further in section 2.4.1, also belongs to the network layer.

Transport Layer Protocols

The *transmission control protocol* (TCP) [57] and the *user datagram protocol* (UDP) [58] are the two most common transport layer protocols. The TCP offers reliable connections through congestion control and retransmissions in the case of packet loss. The UDP, on the other hand, is connectionless, which means that packets are sent without any guarantees of them actually reaching the receiver. Both protocols have fields for the specification of port numbers in their headers. The application layer protocols have well-known ports that are assigned by the IANA, to facilitate application-to-application communication.

Application Layer Protocols

Application layer protocols interact directly with the application processes. Some common protocols are the *file transfer protocol* (FTP), the *hypertext transfer protocol* (HTTP) used to transfer web-page information, and RTP, which has already been described in subsection 2.2.3.

2.3.4 Quality of Service Provisioning

Quality of service (QoS) denotes the quality associated with transmission over a network, e.g. in terms of minimum bandwidth, maximum delay and maximum error rate. In packet-switched networks, certain levels of QoS can be requested and performed by two alternative network-layer frameworks, namely *integrated services* (IntServ) and *differentiated services* (DiffServ). For Ethernet networks, QoS provisioning can also be performed on the data link layer through prioritization of *virtual LANs* (VLAN).

IntServ [59] utilizes the transport layer *resource reservation protocol* (RSVP) [60] to, if possible, set up a path to the receiver with a guaranteed service level before the start of transmission. These resources are then allocated for the entire duration of the transmission.

With DiffServ [61], the first six bits of the DS field in the IP header is used to request a desired QoS level. This level is then provided in a best effort manner through the queuing management in routers and switches.

A VLAN is a group of hosts that communicate as if they were attached to the same physical LAN, although they do not have to be. The VLAN standard IEEE 802.1q specifies a tag that is added to the Ethernet frames. This tag identifies the

VLAN and the priority of the frame. There are eight priority levels, and their handling is described in the IEEE 802.1p standard.

2.4 Multicast

IP multicast has been developed for simultaneous transmission from one source to many receivers. The current standard for IP multicasting is described in [12], by Deering. In brief, multicast works according to the following basics:

- Sources can send multicast packets at any time without the necessity for registration.
- Sources only need to know the multicast address, not the individual receivers.
- Receivers can join and leave multicast groups at any time.
- Multicast packets are delivered with UDP, according to best-effort principles.

This section will supply information regarding IP multicast group management, addressing, intra- and inter-domain routing. Some more specific implementations are also outlined, such as layered multicast and multicast with error control. Finally, overlay multicast is described. It refers to alternative techniques that might be employed when IP multicast is not supported.

2.4.1 Multicast Group Management

Hosts desiring to start or stop receiving multicast traffic destined for a particular multicast group, must communicate their wishes to any neighboring multicast routers. This is the main task of the *Internet group management protocol* (IGMP), which has appeared in four incarnations.

When the first version of IGMP was designed, which was later referred to as *IGMP version 0* (IGMPv0) [62], the creation of multicast groups and the maintenance of group membership information were the responsibilities of the multicast routers[12]. From IGMPv1 [63] and onwards, these responsibilities have been transfered to the hosts. A brief description will be given here to the IGMPv1 and the functionality added by the succeeding versions.

IGMPv1

There are only two types of IGMPv1 [63] messages of concern to the hosts, the *host membership query* and the *host membership report*. Multicast routers send host membership queries to the *all-systems multicast group* with IP address 224.0.0.1 to discover

[12]The multicast routers were called *multicast agents* in IGMPv0.

which multicast groups have members on their local network. Hosts respond with host membership reports that provide information regarding the multicast groups they belong to. Hosts also send host membership reports immediately when they join a new group, in case they are the first member of that group within the network. When a host wants to leave a multicast group, it simply stops reporting that group.

When a host receives a host membership query, rather than immediately sending a report, it starts a timer set to a randomly-chosen value, for each of its group memberships on the network interface of the incoming query. When a timer expires, a report is generated for the corresponding multicast group and the reports are therefore spread out over time.

A host membership report is sent with an IP destination address equal to the multicast group address being reported, and with an IP TTL of 1. Therefore, all the other members of the same multicast group on that network will also receive the report. When a host hears a report for a group to which it belongs, the host stops its timer for that group and does not generate a report. Thus, generally only one report will be generated for each group present on the network.

IGMPv2

In addition to the functionality of IGMPv1, IGMPv2 [64] facilitates the reporting of group membership termination through *leave group* messages. This is most useful for high-bandwidth multicast groups and/or subnets with highly volatile group membership.

Additionally, the host membership queries are divided into *general queries*, concerning all multicast groups, and *group-specific queries*. When a router has received a leave group message regarding a multicast group, a group-specific query is transmitted in order to discover whether there are any other remaining hosts of that particular multicast group. In contrast to general queries, which are sent to the all-systems multicast group, group-specific queries are sent with an IP destination address equal to the multicast address of interest.

IGMPv3

In IGMPv3 [65], support for source filtering is added. Hosts can thereby specify in their membership reports which source addresses they are interested in receiving traffic from, or alternatively choose to receive traffic from all but the specified addresses. This procedure is called *source-specific multicast* (SSM) and is further described in [66]. The information is also useful for multicast routing protocols to avoid delivering multicast packets from specific sources to networks containing no interested receivers.

Furthermore, a *group-and-source-specific query* message type is added. These queries are, for example, sent by the routers when a host leaves a multicast group, to verify whether there are any remaining members of that particular group, who would like to receive traffic from a given set of sources. As is the case with group-specific queries, group-and-source-specific queries are sent to the multicast group being queried.

2.4.2 Multicast Addressing

To simplify the task of routers, multicast IP addresses must be separated from the ordinary unicast addresses. They are therefore located in a separate address space, which is outlined in the first subsection. The multicast address-allocation process is described in the second subsection.

Multicast Address Space

As specified in [67], the IPv4 addresses in the range from 224.0.0.0 through 239.255.255.255 are reserved as multicast addresses. These addresses, numbering approximately 270 million, are divided into smaller address ranges, which are devoted to different purposes as specified in [68]. For example, the 232/8 address space is reserved for SSM, whereas 233/8 and 239/8 are reserved for GLOP addressing and administratively scoped multicast respectively, which will be described in the following section.

Multicast Address Allocation

The *multicast address allocation architecture* (MALLOC), presented in [69], is a high-level description regarding how the address-allocation process should be structured. The MALLOC was never adopted as a standard, for reasons that will be given later in this section. Nevertheless, it is outlined here to illustrate some interesting aspects of multicast address allocation. Following on from this, the techniques that have actually been deployed are described.

Multicast addresses were, in the MALLOC, proposed to be allocated statically, scope relative, or dynamically. Static addresses would be used for protocols requiring well-known addresses in order to work. Scope-relative addresses were for infrastructure protocols, which require an address in every administrative scope. Since address space is limited and both static and scope-relative addresses generally had a permanent lifetime, these addresses would only be assigned after examination and approval of a request.

Dynamic addresses would be provided on demand, have a limited lifetime, and be aimed at the rest of the applications. According to [69], the dynamic address allocation process should be divided into three layers. Layer-one protocols[13] enable hosts to request multicast addresses from a *multicast address allocation server* (MAAS). Layer two consists of intradomain protocols[14]which should be used by the MAASs

[13]The *multicast address dynamic client allocation protocol* (MADCAP) [70] belongs to layer one.
[14]The *multicast address allocation protocol* [71] is a proposed layer-two protocol.

to coordinate allocations so that no addresses are allocated twice. The MAASs would also use layer-two protocols to acquire ranges of multicast addresses from *prefix coordinators*. Interdomain allocation of multicast address ranges to prefix coordinators would be the task of layer-three protocols[15]. However, according to [73], dynamic multicast address allocation protocols are complex and were abandoned since they provided no benefit over GLOP and unicast-prefix-based allocation. These techniques will therefore be outlined in the next section.

GLOP addresses have global scope and arelimited to the 233/8 address space. The AS number is mapped into the two middle octets of the address, in accordance with [74]. Consequently, each AS receives its own /24 GLOP address space.

Unicast-prefix-based multicast addressing was first proposed, and is most useful, for IPv6 systems [75], although there also exists a proposal for IPv4 [76]. The mechanism is similar to that for GLOP addressing, but instead of the AS number, the unicast prefix is used. The number of multicast addresses assigned to an AS would therefore depend on the size of its unicast address space. According to [73], this technique has not added sufficient value for it to be adopted for IPv4.

Another type of addresses that have been deployed are the administratively scoped IP multicast addresses [77], which are for local use within a domain. Consequently, packets addressed to administratively scoped multicast addresses are not forwarded across domain boarders. This means that the same address can be reused within each domain, but this has the drawback that the addresses are not valid outside their domain scope. A parallel might be drawn between administratively scoped IP multicast addresses and private IP addresses.

2.4.3 Intradomain Multicast Routing

Intradomain multicast routing protocols are responsible for the routing of multicast traffic within administrative domains, and can be divided into two basic categories. *Dense-mode* (DM) protocols initially broadcast multicast traffic and then allows routers, which are not interested in the information, to prune the multicast tree. *Sparse-mode* (SM) protocols, on the other hand, use explicit join messages to build the tree. SM protocols are therefore more efficient when the receivers are widely spaced, the break even lies in the area of 10-20% active nodes [78].

The *protocol independent multicast* (PIM) protocol is a common intradomain multicast routing protocol that has gained its name from the fact that it has no routing table of its own, but instead uses whatever underlying unicast routing table is available. PIM exists in both DM and SM versions, and it will therefore be used to briefly describe the basic mechanisms of DM and SM protocols. Other examples of DM and SM protocols are *multicast extensions to open shortest path first* (MOSPF) respectively *core based trees* (CBT).

[15]The *multicast address-set claim* (MASC) [72] protocol is an example of a layer-three protocol.

Dense-Mode Protocols

Protocol independent multicast – dense mode (PIM-DM) [79] is generally used for individual small domains. It uses source-based trees, i.e. a separate multicast distribution tree is built for each source sending data to a multicast group.

When a multicast source starts to send data, each router on the same LAN receives the data and forwards it to all its PIM neighbors and to all links with directly attached receivers. Each router receiving a forwarded packet on its upstream interface will forward it in a likewise manner. The upstream interface is the interface that according to the routing table would be used for transmission back to the source. The procedure of determining whether an interface is the upstream interface is known as a *reverse path forwarding* (RPF) check. Packets arriving on other interfaces are dropped, to prevent forwarding loops. In this way, the data are flooded to all parts of the network.

Routers that do not have any interested downstream receivers send PIM prune messages to remove themselves from the tree. After a certain amount of time, the prune state at each router will time out. Data will once again be transmitted to these parts of the network and the prune process has to be repeated.

Sparse-Mode Protocols

Protocol independent multicast – sparse mode (PIM-SM) [80] is more common than PIM-DM. With PIM-SM, routers must explicitly send join messages to their upstream neighbors concerning their interest in particular groups, whereas prune messages are used to leave multicast distribution trees. As is this case with PIM-DM, a time out occurs after a period of time, and join messages are therefore retransmitted periodically.

PIM-SM by default uses shared trees, which are multicast distribution trees rooted at some selected node, which in PIM is called the *rendezvous point* (RP). To send data to the multicast group, it must be encapsulated in PIM control messages and sent by unicast to the RP. This is performed by the source's *designated router* (DR), which is a specific PIM router on the source's local network. PIM-SM also supports the use of source-based trees, which might be used to optimize the data path.

Bi-directional protocol independent multicast (BIDIR-PIM) [81] is a proposal of another SM protocol, which is based on PIM-SM. The main difference, as compared to PIM-SM, is in the method used to send data from a source to the RP. In PIM-SM, data are sent using either encapsulation or a source-based tree, whereas in BIDIR-PIM the data flow to the RP along the shared bi-directional tree. BIDIR-PIM scales better when there are many sources for each group, but the lack of source-based trees means that traffic is forced to remain on the possibly inefficient shared tree.

2.4.4 Interdomain Multicast Routing

The interdomain multicast routing protocols are responsible for the routing between administrative domains. The historical evolution of interdomain multicast routing, and the motivation behind it, is now described briefly in accordance with [14].

In the early and mid 1990's, multicast was implemented as a flat virtual topology. Islands of multicast enabled networks were connected via tunnels, transmitting unicast-encapsulated multicast packets. The routing was performed by the *distance vector multicast routing protocol*[16] (DVMRP). This virtual multicast network was called the *multicast backbone* (MBone). At a later stage, native multicast capability was incorporated into the MBone, i.e. the routers became able to handle multicast packets. However, as the MBone grew, the flat topology became a problem. The routers had to retain a great deal of state information and the system was very sensitive to misconfigurations.

There are two major proposals regarding how hierarchical interdomain multicast routing should be implemented and these are described below. The first solution is more widely employed, but has been described as a short term solution, because it also does not scale particularly well. The second alternative does provide better scaling, but is dependent upon a strict addressing scheme.

MBGP/PIM-SM/MSDP

The *multiprotocol extensions for border gateway protocol-4* (MBGP) [83] are used to carry multicast routing information between administrative domains. The information does not specify multicast groups, but rather whether or not the multicast sources of the networks can be reached through the router. For reachable networks, the next-hop router is also specified. This information is used when a join message is sent from a receiver or a RP towards the source.

However, MBGP does not provide functions for the multicast-tree construction. One solution is to also use PIM-SM to establish multicast trees between administrative domains.

Another issue also requires a solution. If PIM-SM is used for intradomain routing, the implication of multiple domains is that there may be multiple RPs, i.e. one RP per administrative domain. However, if there is no mechanism for communication between RPs, a RP in one domain will not be aware of sources in other domains. This is the purpose of the *multicast source discovery protocol* (MSDP) [84], which runs on the same routers as the RPs and announces the active multicast sources to other domains.

[16] DVMRP is a DM protocol, similar to PIM-DM, although it maintains its own routing tables. See [82] for more information.

BGMP

The *border gateway multicast protocol* (BGMP) [85] utilizes bidirectional trees with a single root, which are shared between the administrative domains. The root is placed in the domain that owns the multicast address in question. BGMP therefore requires an address allocation scheme with distinct ownership of each multicast address. This could for example be provided by protocols such as MASC, AAP and MADCAP, which were mentioned in subsection 2.4.2.

2.4.5 Layered Multicast

Layered multicast, where multiple multicast groups are used for transmission of related content, was first proposed by Deering [86]. However, McCanne et al. were the first to study the complete *receiver-driven layered multicast* (RLM) scheme in detail, including adaptation algorithms, as presented in [24]. The receivers search the optimum number of layers to subscribe to by adding layers until congestion occurs and then drop layers until all packets are received. RLM was evaluated through simulations and showed reasonable loss and convergence rates under several scaling scenarios.

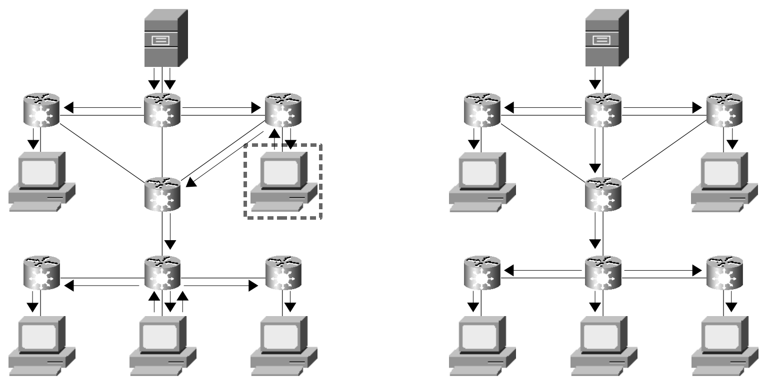
Layered multicast can advantageously be combined with layered video coding, whereby different video layers are transmitted on separate multicast groups. The receivers can then adjust the video quality to suit their rendering and processing capacity, by joining and leaving multicast groups. The receivers can also choose to tailor their bandwidth usage in accordance with the prevailing network conditions.

Multicast transmission schemes with dynamic layering, to better match the demands and limitations of the receivers, have been proposed in [87] and [88]. In [89], Rimac et al. investigated the benefits of the sender adaptation and showed that the user satisfaction can be improved significantly, but that the increased complexity in some cases makes it unsuitable.

2.4.6 Multicast with Error Control

It is complicated to combine multicast transmission with channel coding in heterogeneous network environments. It is impossible to simultaneously satisfy the error rates of all receivers. In [90], the maximum-regret criterion is minimized in order to find a good ratio between source and *forward error correction* (FEC) data. However, it still results in a high number of receivers receiving unnecessary redundancy, while others receive source data that they cannot use due to an insufficient amount of redundancy.

Automatic repeat request (ARQ) schemes also face problems with multicast transmission. An *acknowledgment* (ACK)-based multicast system is exposed to feedback implosion, as described in [91]. A *negative acknowledgment* (NACK)-based scheme, as presented in [92], limits the feedback-implosion problem, but still results in an excessive amount of redundancy data being transmitted to receivers experiencing low



(a) An end-system multicast tree. (b) An IP multicast tree.

Figure 2.12: An end-system multicast tree together with a corresponding IP multicast tree.

packet-loss rates.

The IETF RMT working group has laterly proposed both FEC-based and NACK-based protocols, e.g. in [93] and [94].

Layered Multicast with Error Control

Layered multicast provides the means for an attractive solution to the problem associated with error control for multicast transmission. Redundancy information, which can be used for re-creation of lost packets, may be sent on separate multicast groups. For maximum efficiency, each such multicast group with FEC data should only correspond to a single layer of source data. Several FEC multicast groups might however correspond to the same source layer, facilitating several degrees of error correction.

A number of transmission schemes with dynamic FEC for video, which all build on RLM, have been proposed. The work presented in [26] and [95] contain proposals for the basic scheme, whereas [96] outlines a more advanced version where the sender has an active role in the adaptation.

2.4.7 Overlay Multicast

Overlay multicast, also referred to as *application-layer multicast* (ALM), was developed as a response to the slow deployment of IP multicast [97]. In contrast to IP multicast, overlay multicast only requires limited infrastructural support.

End-System Multicast

One approach to overlay multicast is called *end-system multicast* because the group management, routing, and data replication are handled at the application layer of the end systems [98]. End-system multicast therefore has few implementation obstacles to face when it comes to employment. However, there are a couple of other drawbacks, when compared to IP multicast. Data replication in end systems implies less efficient bandwidth utilization than replication in router nodes, since data will have to be transmitted multiple times across some of the network links [99]. An example of a end-system multicast tree and a corresponding IP multicast tree is provided in Figure 2.12.

The multicast tree also becomes vulnerable to receivers that leave the multicast group, or simply breaks. If they were replicating the information to other receivers, these receivers will also lose their connection to the multicast tree and typically have to repeat the join procedure [98]. If the receiver that is encircled by a dashed line in Figure 2.12(a) leaves the multicast group or breaks, the three receivers at the bottom will also lose their connection.

There exists a wast number of proposals for end-system multicast, targeting both reliability and performance. The work presented in [98] uses a control mesh that helps the receivers to faster find a new source of the multicast session if an upstream node breaks. In [100], centralized performance monitoring is used to progressively and adaptively evolve the overlay multicast topology over time to achieve high end-to-end throughput. The compatibility with IP multicast has also been addressed. The solution presented in [99] offers an IP multicast interface, through the use of a middleware on the end systems, and is thereby transparent to IP multicast enabled applications.

In [101] a hybrid multicast technique is presented, which actually utilizes IP multicast where possible. These IP multicast enabled island are connected in an overlay tree structure, and in each island one receiver is responsible for communication with other islands through unicast and the local receivers through IP multicast.

Proxy-Based Multicast

There also exist overlay multicast approaches that utilize some additional support in the network and thereby not classify as end-system multicast. In the work presented in [102] and [103], the content is sent to agents, or proxy servers, that are placed at strategic positions in the network and forward the content to the receivers in their surrounding. These solutions scale better than end-system multicast when the number of receivers increase.

2.5 Fairness

There are many opinions with regards to what might be considered as fair, and the data-transmission area is no exception. Some argue that all users should be entitled to obtain the same bandwidth or perceive the same satisfaction, others that the accumulated bandwidth or satisfaction of all users should be maximized, and yet others that the users should be entitled to use the same amount of resources within the network. Multicast transmission makes it even more difficult to define fairness.

The research on multicast fairness is quite diverse. The standpoint of the IETF is that multicast flows should be treated as unicast flows. In 1998, IETF therefore proposed that a TCP-friendly congestion-control scheme should be developed for multicast traffic [104]. Handley and Floyd responded to the call and presented a strawman specification of *TCP-friendly multicast congestion control* (TFMCC) the same year [105]. TFMCC is a single-rate scheme which adapts the transmission rate to the receiver that experiences the worst network conditions. The latest version of TFMCC, a protocol specification of experimental status, was published in 2006 [106].

However, in [107] Chiu studied a TCP-friendly scheme together with max-min and proportional fairness, described in [108] and [109] respectively, and drew the conclusion that none were satisfactorily fair. Chiu was of the opinion that the cost-saving sharing of bandwidth, between receivers of the same multicast flow, should be rewarded. From a fairness perspective, it would be reasonable to favor multicast flows when performing the bandwidth allocation.

Since the favoring of multicast traffic is motivated by resource sharing, multicast flows should optimally be prioritized based on the number of receivers. An intuitive solution is to allocate the bandwidth linearly proportional to the number of receivers of each flow. However, this has the negative side effect that unicast flows run the risk of being starved in scenarios involving many multicast flows with many receivers. In [110], Legout et al. therefore proposed a bandwidth-allocation policy where the bandwidth was allocated logarithmically proportional to the number of receivers of each flow. The policy was called *logarithmic receiver dependent* (LogRD) and proved to be a good compromise between prioritizing multicast flows linearly proportional to their number of receivers and not starving unicast flows.

Other researchers have attempted to make the in research frequently employed RLM transmission scheme, outlined in subsection 2.4.5, fairer, since it was proved to be unfair in [111] and [112]. Examples of such work are [113], [114], [115], and [116]. Other multicast-fairness policies that have been proposed include [117], [118], [119], and [120]. The most complete work on a general definition is probably *multicast utility max-min fairness*, presented by Rubenstein et al. in [113], but they did not consider prioritization of multicast flows.

The concept of *utility functions* is explained in the first subsection. Utility functions are utilized by some of the fairness definitions that are described in subsection 2.5.2 through to 2.5.5, which represent the evolution from *max-min fairness* towards multicast utility max-min fairness. Subsection 2.5.6 outlines the LogRD bandwidth-allocation policy.

2.5.1 Utility Functions

When von Neumann and Morgenstern wrote in 1944 the classical work *Theory of games and economic behavior* [121], the utility concept was already "well known". They used the terms utility and satisfaction interchangeably and argued for their assumption that utilities could be treated as numerically measurable quantities. In [122],

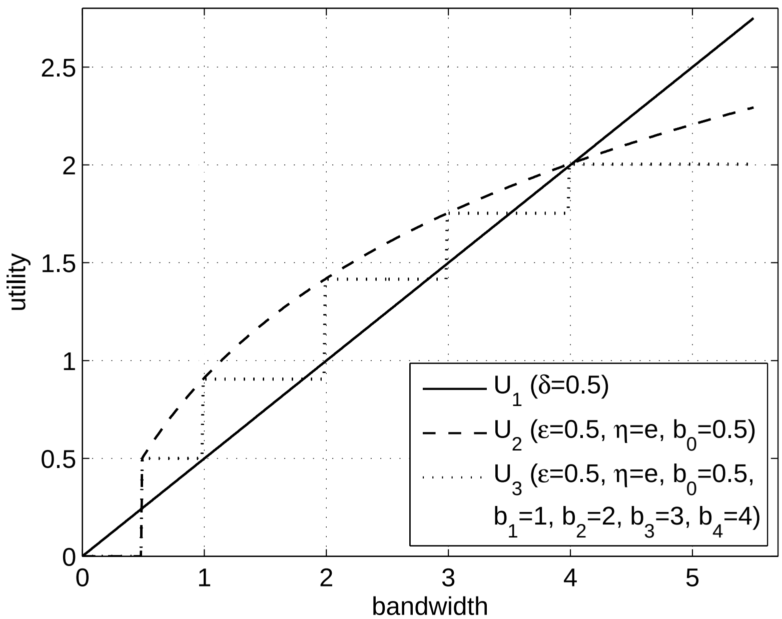


Figure 2.13: Three possible utility functions.

Nash introduced the concept of anticipation, which is closely related to utility. He denoted the assignment of a real number to each anticipation of an individual to be a *utility function*.

In [123], Shenker applied utility functions to computer networks and allowed them describe the mapping from delay, throughput, and packet drops, etc. to the performance of the receiving application. Somewhat simplified, they can be regarded as the mapping from the obtained bitrate to the satisfaction of the receiver. Since an increase in bitrate b never leads to a decrease of receiver satisfaction, the simplified utility function U must fulfill the criterion $b > b' \Rightarrow \mathrm{U}(b) \geq \mathrm{U}(b')$.

Now three examples of possible utility functions are provided. The utility functions are also visualized in Figure 2.13.

1. File download may be represented by a linear utility function,

$$\mathrm{U}_1(b) = \delta \cdot b,$$

   where $\delta > 0$ and represents the urgency of the file.

2. Video encoded for rate b, with minimum feasible bitrate b_0 and corresponding utility ε, could have a utility function similar to

$$\mathrm{U}_2(b) = \begin{cases} 0 & \text{for } b < b_0 \\ \varepsilon + \log_\eta(1 + b - b_0) & \text{for } b \geq b_0, \end{cases}$$

where $b_0 > 0$, $\varepsilon > 0$, and $\eta > 0$ is a parameter that represents the quality improvements that an increase in bandwidth brings.

3. The utility function of video encoded into k layers, where b_0 is the bitrate of layer 0, $b_1 - b_0$ is the bitrate of layer 1, and so on, would then become

$$\mathrm{U}_3(b) = \begin{cases} 0 & \text{for } 0 \leq b < b_0 \\ \mathrm{U}_2(b_0) & \text{for } b_0 \leq b < b_1 \\ \mathrm{U}_2(b_1) & \text{for } b_1 \leq b < b_2 \\ \vdots & \\ \mathrm{U}_2(b_{k-1}) & \text{for } b_{k-1} \leq b. \end{cases}$$

2.5.2 Max-Min Fairness

The definition of max-min fairness, which is well described in [108], maximizes the resource allocation to the receivers that are worst off. The definition reads as follows.

Let $\mathbf{B} = \{b_1, b_2, ..., b_K\}$ be a feasible bandwidth allocation to K receivers, where b_k is the bandwidth allocated to receiver r_k. Further, let $\mathbf{B}'$ be an alternative feasible allocation such that there exists a receiver r_k for which $b'_k > b_k$. Then $\mathbf{B}$ is *max-min fair* if for every alternative allocation $\mathbf{B}'$ there exists another receiver $r_{k'}$, such that $b_k \geq b_{k'} > b'_{k'}$.

In [124], Mo and Walrand discovered that the max-min fair bandwidth allocation is just one of the allocations that can be described by a more general equation. Let $\mathbf{B}$ and $\mathbf{B}'$ be defined in the same manner as earlier. Then the equation

$$\sum_{k=1}^{K} \frac{b'_k - b_k}{b_k^{\alpha}} \leq 0, \ \forall\, \mathbf{B}'$$

can be used to describe max-min fairness when α approaches infinity, whereas maximum throughput, proportional fairness [109], and minimum potential delay [125] are represented when α equals zero, one and two, respectively. A larger α correspond to a fairer allocation policy, which generally has been considered to be at the expense of the system throughput [126]. However, according to Tang et al. [127], a fair allocation does not necessarily have to be inefficient and the outcome depends upon the network topology and traffic pattern.

2.5.3 Multicast Max-Min Fairness

In [128], Tzeng and Siu modified the max-min fairness definition to make it valid for scenarios with single-rate multicast traffic. In single-rate multicast, all receivers in a multicast group receive data at the same rate. To describe the modified definition, the concept of sessions must firstly be introduced. In a unicast scenario, a session simply denotes the flow from the transmitter to the receiver, whereas in a multicast context it comprises the flows to all the receivers in the multicast group.

Translated into the notations that are listed in the Terminology chapter of this thesis, the definition reads as follows. For a network $\mathbf{N}$, let $\mathbf{B}$ be an allocation vector of receiver rates and let $\mathbf{B}'$ be an alternative feasible allocation, where $b'_{i,k} > b_{i,k}$ for some receiver $r_{i,k}$ of session s_i. Then $\mathbf{B}$ is *multicast max-min fair* if for every feasible allocation $\mathbf{B}'$ there exists a receiver of another session $s_{i'}$, such that $b_{i,k} \geq b_{i',k'} > b'_{i',k'}$.

Tzeng and Siu also defined a *bottleneck link* of a session as a link where all other sessions traversing that link are allocated equal or smaller amounts of bandwidth. They then proved that a feasible bandwidth allocation $\mathbf{B}$ is max-min fair if, and only if, every session has a bottleneck link.

2.5.4 Utility Max-Min Fairness

In [129], Cao and Zegura targeted a connection-oriented service and combined the concept of max-min fairness with the consideration of utility instead of bandwidth. As described in subsection 2.5.1, the utility represents the satisfaction a service brings to its end users. The resultant definition consequently maximized the minimum satisfaction of any end user.

Cao and Zegura defined a *utility max-min allocation* as the feasible bandwidth-allocation vector that results in the largest ordered vector of utilities. However, it could just as well have been defined in the same manner as for that of ordinary max-min fairness, with the bandwidths replaced by utilities.

Further, Cao and Zegura defined a *utility-bottleneck link* of a connection, with respect to a specific allocation, as a link whose capacity is fully allocated and for which the utility of all other connections are smaller than, or equal to, the utility of the connection in question. Finally, they proved that an allocation is *utility max-min fair* if, and only if, it is feasible and all connections have a utility-bottleneck link.

2.5.5 Multicast Utility Max-Min Fairness

Rubenstein et al. presented a definition of fairness in [113], which took both multi-rate multicast and utilities into consideration. Translated into the notations of the Terminology chapter, the definition reads as follows. For a network $\mathbf{N}$, let $\mathbf{B}$ be an allocation vector of receiver rates and let $\mathbf{B}'$ be an alternative feasible allocation, where $U_i(b'_{i,k}) > U_i(b_{i,k})$ for some receiver $r_{i,k}$ of session s_i. Then $\mathbf{B}$ is *multicast util-max-min fair* if for every feasible allocation $\mathbf{B}'$ there exists a receiver of another session $s_{i'}$, such that $\mathrm{U}_i(b_{i,k}) \geq \mathrm{U}_{i'}(b_{i',k'}) > \mathrm{U}_{i'}(b'_{i',k'})$.

Rubenstein et al. then observed two desirable properties of unicast utility max-min fairness. Besides the obvious *max-min fairness* property, they defined *same-path receiver fairness* as: if two sessions have identical routes, then either they have the same utility or at least the session with the lowest utility has reached its maximum. Rubenstein et al. then expanded these fairness properties into four new properties that were better fitted to multicast traffic: *fully-utilized receiver fairness*, *same-path re-*

ceiver fairness, *per-receiver link fairness*, and *per-session link fairness*, which are all outlined in [113]. Finally, they proved that multi-rate multicast util-max-min fair allocations satisfy all four fairness properties.

2.5.6 The LogRD Bandwidth-Allocation Policy

Employing the LogRD bandwidth-allocation policy presented in [110], the share of bandwidth $d_{i,j}$ of a link l_j allocated to a particular session s_i depends logarithmically on the number of receivers $|\mathbf{R}_{i,j}|$ that are downstream of link l_j. The fair share of bandwidth is calculated according to

$$d_{i,j} = \frac{1 + \ln |\mathbf{R}_{i,j}|}{\sum_{i'=1}^{I_j} (1 + \ln |\mathbf{R}_{i',j}|)}, \tag{2.1}$$

where c_j is the capacity of link l_j, and I_j is the number of sessions traversing link l_j.

Legout et al. evaluated the LogRD policy in terms of receiver satisfaction and fairness. The satisfaction was measured as the mean obtained bitrate of all the receivers, and the fairness was measured as the inverse of the standard deviation of the obtained bitrates of all the receivers.

Simulations were performed with the *receiver independent* (RI) and *linear receiver dependent* (LinRD) bandwidth-allocation policies as references. The RI policy divides the bandwidth equally between the sessions, independently of their number of receivers, whereas LinRD allocates the bandwidth linearly in proportion to the number of receivers of each session. The LogRD and LinRD policies both lead to a significant increase in average satisfaction for the multicast receivers when compared to RI, but LogRD does not sacrifice the average satisfaction of the unicast receivers to the same extent as LinRD.

2.6 Cost Allocation

In this section, a number of cost-allocation mechanisms for cost sharing among multicast receivers are outlined. These are a selection of existing mechanisms, other proposals for example include [130] and [131]. However, some of the terminology associated with cost sharing among multicast receivers is firstly introduced.

2.6.1 Terminology for Multicast Cost Sharing

This section outlines the notations for cost sharing among multicast receivers, originally introduced in [132].

The number of receivers upstream and downstream respectively for a particular link are denoted by n_u and n_d. The receivers downstream of a link are those re-

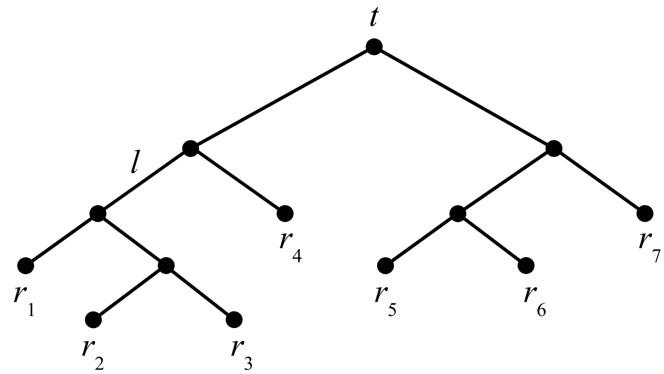


Figure 2.14: A multicast transmission tree with seven receivers.

ceivers whose transmission paths from the source traverse that link. The receivers upstream of a link are somewhat less intuitively defined as the receivers who are not located downstream of that link. In the multicast tree of Figure 2.14, where t is the transmitter, receivers r_1, r_2 and r_3 are located downstream of link l, whereas receivers r_4 through r_7 are upstream of link l. The part of the cost of the link allocated to the upstream receivers is described by the function $\mathrm{F}_u(n_u, n_d)$, whereas $\mathrm{F}_d(n_u, n_d)$ represents the part of the cost that is allocated to the downstream receivers.

Multicast sessions that support multiple QoS levels are also covered in [132]. The shares of the total cost allocated to the upstream and downstream receivers requesting QoS level i, are denoted by $\mathrm{F}^i_u(z_u, z_d)$ and $\mathrm{F}^i_d(z_u, z_d)$ respectively. However, the terms z_u and z_d are not defined.

2.6.2 The Edge-Pricing Paradigm

Pricing and cost allocation in computer networks are treated extensively by Shenker et al. in [133]. They initiate their discussion with pricing based on estimated congestion conditions. The reason being the high complexity associated with the computation of the actual prevailing congestion conditions and the consequence is basically QoS-sensitive time-of-day pricing. They then claim that differentiated pricing based on estimated congestion conditions can be exchanged for differentially priced QoS classes. When the estimated congestion probability is low, even cheaper QoS classes will perform well. Users can therefore adapt their costs by monitoring and changing QoS classes.

Shenker et al. further propose that the pricing, aside from the QoS class, should only depend on the locations of the source and destination. The costs of the actual transmission path are approximated using the costs of the expected path. Consequently, the prices are based upon the estimated congestion conditions along the expected transmission path from the source to the destination. If information about congestion conditions is gathered at the edges of an ISP's network, it should be possible to determine the price of a session at the access point. For connections that traverse the borders between different ISPs, these must purchase the service from each other in the same manner that regular users purchase service. This solution is called the *edge-pricing paradigm*.

Multicast traffic causes a challenge for the edge-pricing paradigm, because a multicast destination address is merely a logical name and does not identify the individual receivers of the multicast group. The only information about multicast sessions that is present in a router node is regarding the next hop(s). It is therefore impossible to estimate the multicast tree at the access points. Shenker et al. propose control messages to be sent when new receivers join a multicast group. These messages should be forwarded along the reverse multicast tree to the access point of the source, where the cost of the tree may be approximated. The ISPs would process the control messages at the edges of their network and thereby extract adequate information. An alternative solution is to record the cost of each link within the control messages.

Shenker et al. also have a general discussion relating to cost sharing among multicast receivers. However, they do not propose any cost-allocation mechanism.

2.6.3 Single QoS Cost Allocation

In [132], Herzog et al. present an extensive work regarding how the costs of multicast trees should be split among the receivers. They present a number of cost-allocation mechanisms, of which the *equal tree split* (ETS) and *equal link split downstream* (ELSD) mechanisms are given the most attention.

The ELSD cost-allocation mechanism splits the cost of each link in the tree evenly between the downstream receivers. Using the notations introduced in subsection 2.6.1, the part of the cost of the link allocated to the upstream receivers can be described as

$$\mathrm{F}_u(n_u, n_d) = 0, \tag{2.2}$$

whereas the part of the cost allocated to each downstream receivers becomes

$$\mathrm{F}_d(n_u, n_d) = \frac{1}{n_d}. \tag{2.3}$$

The ETS cost-allocation mechanism splits the cost of the entire transmission tree uniformly amongst all the receivers. Using the same notations, we obtain

$$\mathrm{F}_u(n_u, n_d) = \mathrm{F}_d(n_u, n_d) = \frac{1}{n_u + n_d}. \tag{2.4}$$

2.6.4 QoS-Based Cost Allocation

If the transmitted data are hierarchically encoded and marked and the router nodes employ priority dropping, users may choose to subscribe to a service although they cannot utilize the entire data rate transmitted by the source. The most obvious reason associated with such limitations are network connections with low capacity. When the transmitted content is real-time video, another limiting factor might be the rendering capacity of the receiving device. In either case, these users do not utilize the entire bandwidth allocated to a multicast session, at least not on all of the links along their transmission path.

In [132], Herzog et al. observe that this should affect the cost allocation of multicast sessions, but they do not propose any specific cost-allocation mechanism for these scenarios. Using the terminology of subsection 2.6.1, they do however point out that if the cost-allocation functions fulfill the following condition,

$$\sum_{i=1}^{I} \left(z_u^i \cdot \mathrm{F}_u^i(z_u, z_d) + z_d^i \cdot \mathrm{F}_d^i(z_u, z_d) \right) = 1, \tag{2.5}$$

the costs associated with the link in question are fully allocated among the receivers.

Liu et al. study usage-based pricing and cost sharing of multicast traffic in [134]. They propose a cost-allocation mechanism, whose cost sharing they state "is proportional to individual members resource requirements, should a unicast service be used". The receivers are divided into categories depending on their requested QoS level. The costs associated with a particular category are then aggregated over the entire multicast tree, but only split among receivers obtaining that QoS level or higher, in an ETS fashion. Henceforth, this cost-allocation mechanism is therefore referred to as *QoS-dependent ETS* (QoS-D ETS).

2.7 Game Theory

Many researchers have considered the bandwidth-allocation and pricing process from a game-theoretic perspective. Somewhat simplified, this implies that potential users place bids which reflect what the service is worth to them. The *Internet service provider* (ISP) then allocates the resources according to these bids. Some basic notions of game theory that found in [135] are outlined in 2.7.1, followed by two game-theoretic cost-allocation mechanisms. Other works on the same subject are [136] and [137].

2.7.1 Game-Theoretic Notions

A cost-allocation mechanism in which the costs allocated to the users exactly match the cost of the service, is called *budget balanced*. A user's *welfare* can be described as the satisfaction after obtaining a service for a certain cost. An *efficient* cost-allocation mechanism chooses to serve the set of users that maximizes the aggregated welfare of all the users.

Assume that a user is part of a user set that is a subset of a larger set of users. Then a cost-allocation mechanism is *cross-monotonic* if for all such user sets, the cost allocated to the user when the larger set is served, is lower or equal in comparison to when the smaller set is served.

It is reasonable to assume that users are selfish and place bids that maximize their probable welfare. A cost-sharing mechanism is *strategyproof* if users maximize their welfare by placing bids that truthfully correspond to how much the service is worth to them. *Group strategyproof* is a harder criterion that requires the cost-allocation mechanism to be resistant against groups of users who jointly place their bids in an attempt to increase their welfares.

Another contribution of [135], is the establishing of the following three basic requirements:

- **no positive transfers** – no user is paid to obtain a service
- **voluntary participation** – no user is forced to obtain a service
- **consumer sovereignty** – no user is refused a service if their bid is sufficiently high

According to [138], there are two cost-allocation mechanisms that are naturally strategyproof and adhere to these basic requirements, the *marginal-cost* (MC) and *Shapley-value* (SH) mechanisms. Further, it is stated that these are the two most appropriate mechanisms for cost sharing among multicast receivers.

2.7.2 The Shapley-Value Mechanism

The SH cost-allocation mechanism is the game-theoretical equivalent to ELSD. It splits the cost of a network link equally between all receivers that are located downstream [138]. The SH mechanism is group strategyproof and budget balanced. However, it is not efficient but has the smallest maximum loss of welfare among the budget-balanced mechanisms.

2.7.3 The Marginal-Cost Mechanism

As described in [135], the MC mechanism essentially charges the marginal cost to the users, that is the cost of providing the service to all users minus the cost of providing the service to all but the user in question. It therefore has the characteristic that it treats equals equally, that is if two receivers give rise to the same marginal cost and place identical bids, they are allocated the same amount of resources and are charged the same cost. Further, the MC mechanism is efficient but not budget balanced nor group strategyproof.

In [139], the MC mechanism is applied to multicast sessions that support multiple rates. The *split session* and *layered* paradigms are studied, but only the layered paradigm is somewhat relevant here, since a split session basically implies separate transmissions of different QoS levels, i.e. the problem associated with multiple QoS levels is divided into a number of problems, each with a single QoS level.

As described in subsection 2.4.5, the layered paradigm utilize hierarchically encoded data, which is divided into QoS layers that are transmitted to individual multicast groups. The receivers consequently join multicast groups with QoS layers that can be combined into the desired QoS level. The layered paradigm therefore inherently implies that costs are separated according to QoS requirements.

2.7.4 Comparison between the SH and MC Mechanisms

In [140], both the SH and MC cost-allocation mechanisms are implemented and experiments are carried out. The MC is shown to generate a smaller revenue, which is not surprising since it is not budget balanced. On the other hand, the MC mechanism is faster than the SH mechanism.

In [138], it is observed that the MC mechanism only requires two messages per link in the multicast tree, whereas the number of messages required for the SH mechanism is of the order of the square of the number of links.

Chapter 3

Fair Bandwidth Allocation for Multicast Traffic

When many Internet users are simultaneously interested in the same content, multicast transmission offers considerable resource savings compared to transmission of separate unicast flows to every user. Nevertheless, deployment of multicast proceeded at a rather slow rate. According to [16], less than 3 percent of the ASs were multicast enabled in the autumn of 2007.

If multicast sessions were prioritized during the bandwidth-allocation process, it would create a significant incentive for the employment of multicast transmission. As mentioned in Section 2.5, it might also be considered as being unfair if a unicast flow, with a single receiver, obtains the same amount of bandwidth as a multicast session, which possibly has several thousand receivers.

Consequently, in Section 3.1, a new definition is presented regarding how the bandwidth should be distributed between unicast and multicast sessions. The definition favors multicast sessions to produce fair bandwidth allocations and to promote the deployment of multicast transmission. Further, in Section 3.2, two bandwidth-allocation policies which attempt to produce allocations as close to the definition as possible are proposed. Finally, in Section 3.3, some transmission techniques that might pose problems for this kind of bandwidth-allocation policies are discussed.

3.1 Multicast-Favorable Max-Min Fairness

This section includes a discussion regarding fair allocation of bandwidth to multicast sessions and a definition of multicast fairness called *multicast-favorable max-min fairness* (MFMF) is presented. A fairness measure, which is based on the definition, is also proposed. However, the concept of multicast-favorable functions is firstly introduced.

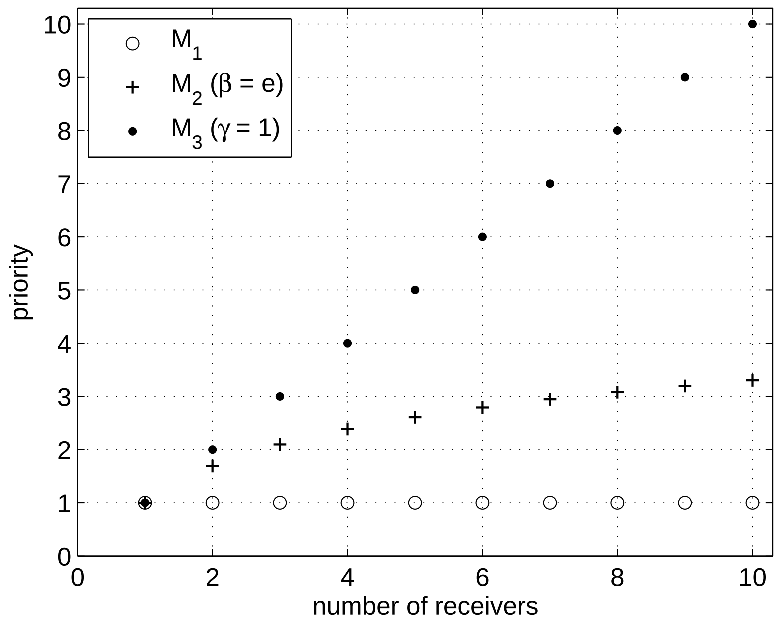


Figure 3.1: Three potential multicast-favorable functions.

3.1.1 Multicast-Favorable Functions

A multicast-favorable function M is a function to describe the prioritization of multicast sessions based on the number of receivers. Such a function must intuitively fulfill the following three criteria:

- for one receiver it should equal unicast, $\mathrm{M}(1) = 1$
- an increase in the number of receivers should never lead to a decrease in priority, $n > n' \Rightarrow \mathrm{M}(n) \geq \mathrm{M}(n')$
- a multicast session should never obtain a higher priority than the sum of the corresponding unicast sessions, $\mathrm{M}(n) \leq n$

There now follows three examples of possible multicast-favorable functions, which are also plotted in Figure 3.1:

1. no prioritization of multicast,

$$\mathrm{M}_1(n) = 1$$

2. logarithmic prioritization of multicast,

$$\mathrm{M}_2(n) = 1 + \log_\beta(n), \text{ where } \beta > 1$$

3. linear prioritization of multicast,

$$\mathrm{M}_3(n) = 1 + \gamma(n-1), \text{ where } 0 < \gamma \leq 1$$

As Legout et al. observed in [110], a multicast-favorable function based on the natural logarithm appears to be a good trade off between no prioritization and linear prioritization. Multicast sessions with many receivers are prioritized, but the increase in priority is reduced for larger numbers of receivers, thereby limiting the risk of starvation of unicast sessions. They chose

$$\mathrm{M}(n) = 1 + \ln(n) \tag{3.1}$$

as the multicast-favorable function, but never claimed this to be the optimum multicast-favorable function. As the choice of multicast-favorable function is subjective, it is possible that the optimum function may differ between networks, depending on their application.

3.1.2 Multicast-Fairness Discussion

There is much to be won if a definition of multicast fairness was be able to take the number of receivers of multicast sessions into account. Multicast sessions with many receivers could then be prioritized in order to encourage the use of multicast transmission when possible. This would preserve bandwidth, and at the same time facilitate an increase in the average obtained bitrate of the receivers. In spite of this, no unicast session should be badly starved, all in line with the work of Legout et al. presented in [110]. in general terms, the definition of multicast fairness should support multicast-favorable functions, but leave the actual choice of function open.

As the target might be maximum satisfaction for the receivers, the definition should be able to consider the utility instead of the obtained bitrate, as was proposed in [123] and performed in [129]. In this thesis, utility functions are considered to be a direct mapping from bitrate to utility. However, it should be observed that the consideration of utility instead of bitrate results in an unfortunate drawback. The penalty in terms of reduced receiver satisfaction, associated with inefficient bandwidth utilization due to poor source coding, is drastically reduced. Utility functions should therefore be used with care.

Further, to allocate more bandwidth to a session on a link than can be obtained on a later (bottleneck) link, is a waste of resources and should therefore obviously be avoided. This is complicated to achieve and requires, in practice, additional feedback regarding the bandwidth available at the bottleneck link [141]. However, it does not constitute a real problem since the main target here is a definition of fairness, not a distributed bandwidth-allocation policy.

To achieve the best results, the adequate bandwidth-allocation policy should be implemented in large parts of the Internet. However, flows that traverse AS borders would require ISPs to exchange corresponding bandwidth-information, which might constitute an obstacle. Implementation within ASs would therefore be a good start.

3.1.3 The Definition of MFMF

To simplify the definition description, a term called *multicast-favorable value* is introduced. The multicast-favorable value of session s_i on link l_j, for an allocation $\mathbf{B}$, is defined as

$$\mathrm{V}\left(\mathbf{B}_{i,j}\right) = \sum_{m=1}^{|\mathbf{R}_{i,j}|} \frac{\mathrm{U}_i(\mathbf{B}_{i,j}[m]) - \mathrm{U}_i(\mathbf{B}_{i,j}[m-1])}{\mathrm{M}(|\mathbf{R}_{i,j}|) - m + 1}. \tag{3.2}$$

where $\mathbf{B}_{i,j}$ is a vector whose first element $\mathbf{B}_{i,j}[0]$ is 0 and the following elements are the receiving rates of the receivers of session s_i in allocation $\mathbf{B}$, whose routes traverse link l_j, sorted in ascending order. According to equation (3.2), each part of the utility[1] is only favored based on the number of receivers actually obtaining at least that utility. The definition of MFMF is now presented using the multicast-favorable value.

Let $\mathbf{B}$ be an allocation of receiver rates and let $\mathbf{B}'$ be an alternative allocation, where $\mathrm{U}_i(b'_{i,k}) > \mathrm{U}_i(b_{i,k})$ for some receiver $r_{i,k}$ of session s_i. Then $\mathbf{B}$ is defined to be *multicast-favorable max-min fair* if for every feasible alternative allocation $\mathbf{B}'$ there exists some other session $s_{i'}$, such that

$$\mathrm{V}(\mathbf{B}_{i,j}) \geq \mathrm{V}(\mathbf{B}_{i',j}) > \mathrm{V}(\mathbf{B}'_{i',j}), \tag{3.3}$$

for some link $l_j | r_{i,k} \in \mathbf{R}_{i,j}$.

As mentioned in subsection 3.1.2, the consideration of utility instead of bitrate reduces the incentive for efficient source coding. The use of utility functions in equation (3.2) is therefore optional. In situations where a fair allocation of the resources is more relevant than a fair distribution of the satisfaction among the receivers, the utilities should equal the obtained bitrates.

3.1.4 An Unfairness Measure

In this subsection, a measure of the unfairness of an arbitrary bandwidth allocation $\mathbf{B}$ is proposed. The solution is the root mean square (RMS) of the negative differences in obtained utility for all receivers, when compared to the MFMF allocation $\mathbf{B}^{\mathrm{MFMF}}$. Only the negative differences are considered, since only they measure a reduction in the treatment of the receivers, and thereby contribute to the unfairness of the allocation.

The unfairness can therefore be expressed as

$$\sqrt{\frac{\sum_{i=1}^{I} \sum_{k=1}^{|\mathbf{R}_i|} \min^2\left(0, \mathrm{U}_i(b_{i,k}) - \mathrm{U}_i\left(b_{i,k}^{\mathrm{MFMF}}\right)\right)}{\sum_{i=1}^{I} |\mathbf{R}_i|}}, \tag{3.4}$$

[1]The utility is herein considered to be a direct mapping from the bandwidth.

where I is the number of sessions, $\mathbf{R}_i$ is the set of receivers of session s_i, $\mathrm{U}_i()$ is the utility function of session s_i, and $b_{i,k}$ is the bitrate obtained by the k^{th} receiver of session s_i. As with the MFMF definition, utility functions are optional and the utilities should equal the received data rates when a fair allocation of the resources is more relevant than a fair distribution of the satisfaction among the receivers. Some examples of unfairness calculations are included in subsection 3.2.3.

3.2 Bandwidth-Allocation Policies

In this section, two bandwidth-allocations policies that produce allocations close to those of the MFMF definition, are proposed. Perfect MFMF bandwidth allocations are very difficult to achieve, since there is no global information about fair shares of bandwidth on links available at the individual nodes. Both of the proposed policies utilize feedback regarding the bottleneck links of downstream receivers to compensate for this lack of information. The details of the feedback procedure are not included within the scope of this thesis, and the actual bandwidth that the feedback would consume is consequently not accounted for.

The two bandwidth-allocation policies are described in subsections 3.2.1 and 3.2.2, respectively. Then, in subsection 3.2.3, the policies are evaluated through the use of an example and the unfairness measure presented in subsection 3.1.4, whereas subsection 3.2.4 deals with the feedback of the policies.

3.2.1 The Bottleneck-Feedback and Receiver Dependent Policy

The *bottleneck-feedback and receiver dependent* (BFRD) policy is based on the same bandwidth-allocation equation as the LogRD policy (2.1). The difference between the two policies is that the BFRD policy utilizes information feedback regarding the largest bottleneck link of all downstream receivers. Thanks to the feedback, the BFRD policy can avoid the waste of resources through allocation of more bandwidth to a session on a link than can be utilized by at least one of the downstream receivers. The BFRD policy allocates bandwidth to the sessions according to equation (2.1), with the exception that it never exceeds the bottleneck-information feedback. The behavior of the BFRD policy will now be described in greater detail.

The BFRD Policy in a Unicast Scenario

Nodes along a transmission path send feedback concerning their downstream bottleneck link, which may or may not equal the bottleneck link of the path. Therefore, at each intermediate node, the information that is passed further back along the transmission path is the bandwidth of the smallest received bottleneck information and the fair share on the adjacent downstream link. If a node is the receiver of a session, the feedback information should preferably correspond to the maximum bitrate it is able to process. In Figure 3.2, an example of the fair share of bandwidth, the

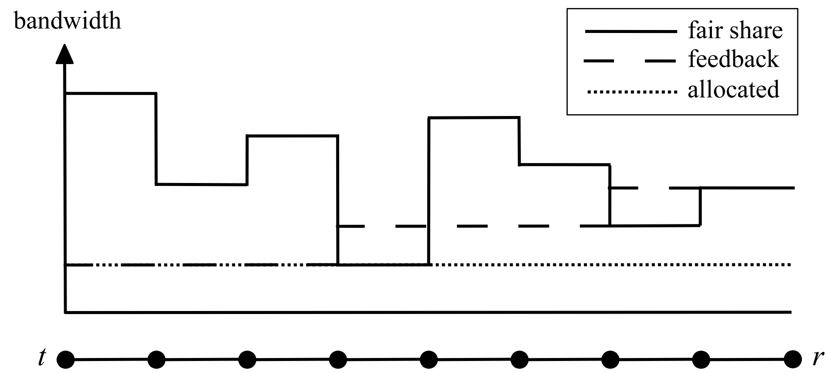


Figure 3.2: The fair share, bottleneck-information feedback, and allocated bandwidth along a transmission path.

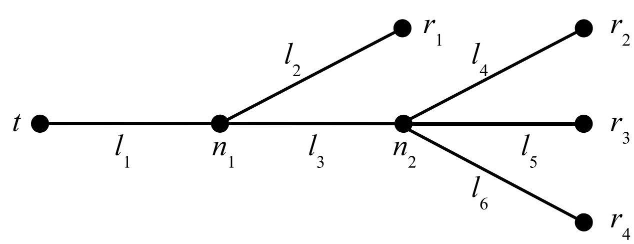


Figure 3.3: A multicast tree.

bottleneck-information feedback, and the actual allocated bandwidth along a transmission path consisting of eight links, are visualized.

When a new session is to be initialized, there is no bottleneck information present at the sender. Consequently, the sender has to start transmitting at the maximum rate that the sender application and the fair share of bandwidth on the outgoing link can manage. The transmission rate is adjusted as soon as the bottleneck-information feedback starts arriving. Most sessions are unlikely to reach their fair share of bandwidth on all links, due to other bottleneck links or limitations in the transmission rate of the senders. In these cases, the unused bandwidth is distributed fairly among the sessions for which the link in question is the bottleneck.

The BFRD Policy in a Multicast Scenario

To transmit feedback of information concerning the largest downstream bottleneck link in a multicast scenario, the information must be processed by each node where the multicast session forks. The process is exemplified below.

In Figure 3.3, a multicast tree with one transmitter, t, and four receivers, r_1 through r_4, is depicted. The multicast tree also includes two intermediate nodes,

n_1 and n_2. Node n_2 processes the bottleneck information arriving on links l_4, l_5, l_6, and the fair share of bandwidth on those links, and only transmits information about the biggest bottleneck back to node n_1. Node n_1 processes this information arriving on link l_3 together with that arriving on link l_2, and the fair share of bandwidth on those links. The result is sent further back to the transmitter, which is then able to determine the adequate transmission rate.

3.2.2 The Full-Feedback and Receiver Dependent Policy

The aim of the design for the *full-feedback and receiver dependent* (FFRD) bandwidth-allocation policy is to be fairer than any other existing policy. Since we regard the MFMF definition as the best description of a fair bandwidth allocation, the goal was a bandwidth-allocation policy that produces allocations as close to the MFMF definition as possible. Nevertheless, the bandwidth efficiency is also important and was therefore also considered.

Of the previously proposed bandwidth-allocation policies, the BFRD policy performed the bandwidth allocations closest to those of the MFMF definition. However, the BFRD policy generally does not produce perfect MFMF allocations. The main reason is that, according to the MFMF definition, the bandwidth allocated to a particular session on a link depends on the bitrate actually obtained by each of the receivers of every session, as can be seen from equations (3.2) and (3.3). The BFRD policy, on the other hand, only adapts the allocation to the largest bottleneck link of the downstream receivers. This makes no difference for unicast sessions, where the bandwidth allocated on the bottleneck link equals the bitrate obtained by the receiver. However, for multicast sessions, all receivers but one may obtain lower bitrates than that indicated by the bottleneck-information feedback.

To reduce this disparity, the FFRD bandwidth-allocation policy therefore extends the amount of feedback to deal with information regarding the bottleneck links of all downstream receivers. To produce bandwidth allocations as close to the MFMF definition as possible, the FFRD policy utilizes equations (3.2) and (3.3), locally on each router node. The multicast-favorable function is based on the natural logarithm (3.1). However, as the utility functions are generally unknown to the nodes, the utilities are replaced by the obtained bitrates. Further, the allocated bandwidths are limited by the bitrates at which the sessions arrive at each node and the received feedback information.

The implication of this is that for a bandwidth allocation $\mathbf{B}$ to be compliant with the FFRD policy, it has to fulfill the following criterion. Let $\mathbf{B}'$ be an alternative allocation, where $b'_{i,k} > b_{i,k}$ for some receiver $r_{i,k}$ of session s_i. Let $\mathbf{B}_{i,j}$ be a vector whose first element $\mathbf{B}_{i,j}[0]$ is 0 and the following elements are the receiving rates of the receivers of session s_i in allocation $\mathbf{B}$, whose routes traverse link l_j, sorted in ascending order. On each link l_j the subsequent equation (3.5) must be satisfied for every feasible alternative allocation $\mathbf{B}'$, and some session $s_{i'}$, given that the computed receiving rate of any receiver may not exceed the received bottleneck-information feedback regarding the receiver in question, nor the bitrate at which the session arrives at the preceding node.

$$\begin{aligned} &\sum_{m=1}^{|\mathbf{R}_{i,j}|} \frac{\mathbf{B}_{i,j}[m] - \mathbf{B}_{i,j}[m-1]}{1 + \ln(|\mathbf{R}_{i,j}| - m + 1)} \\ \geq &\sum_{m=1}^{|\mathbf{R}_{i',j}|} \frac{\mathbf{B}_{i',j}[m] - \mathbf{B}_{i',j}[m-1]}{1 + \ln(|\mathbf{R}_{i',j}| - m + 1)} \\ > &\sum_{m=1}^{|\mathbf{R}_{i',j}|} \frac{\mathbf{B}'_{i',j}[m] - \mathbf{B}'_{i',j}[m-1]}{1 + \ln(|\mathbf{R}_{i',j}| - m + 1)} \end{aligned} \tag{3.5}$$

For comparison, if the BFRD policy was to be described in the same manner as the FFRD policy, the equation corresponding to (3.5) would be

$$\frac{\mathbf{B}_{i,j}[|\mathbf{R}_{i,j}|]}{1 + \ln(|\mathbf{R}_{i,j}|)} \geq \frac{\mathbf{B}_{i',j}[|\mathbf{R}_{i',j}|]}{1 + \ln(|\mathbf{R}_{i',j}|)} > \frac{\mathbf{B}'_{i',j}[|\mathbf{R}_{i',j}|]}{1 + \ln(|\mathbf{R}_{i',j}|)}. \tag{3.6}$$

3.2.3 Evaluation of the Fairness

As mentioned earlier, the BFRD policy typically does not produce perfect MFMF bandwidth allocations, as this generally requires global information. Consequently, the FFRD bandwidth-allocation policy will also not always produce optimum allocations. The intuitive way to evaluate the BFRD and FFRD policies would have been simulations where the results were compared with MFMF bandwidth allocations. However, to our knowledge no efficient method of finding the MFMF bandwidth allocations exists for large networks with high traffic intensities[2] and an evaluation through simulations is therefore infeasible. Instead an example is used to shed some light on the fairness and receiver satisfaction, in terms of the average obtained bitrate, of the two policies. For comparison, the corresponding LogRD and RI allocations are also calculated.

The Example

The scenario depictured in Figure 3.4 is now studied, where t_i is the transmitter and $r_{i,k}$ is a receiver of session s_i. The maximum transmission rate of the transmitters and the capacities c_j of all links l_j are limited to 10 Mbps. For simplicity, all sessions are considered to have linear utility functions that equal the obtained bitrates.

Additionally, the MFMF definition is used together with the multicast-favorable function of (3.1), which is based on the natural logarithm. These are reasonable choices that produce representative results. Furthermore, that is the same multicast-favorable function as is used by the FFRD, BFRD and LogRD policies.

[2]This problem might be NP hard, although this has not yet been proven.

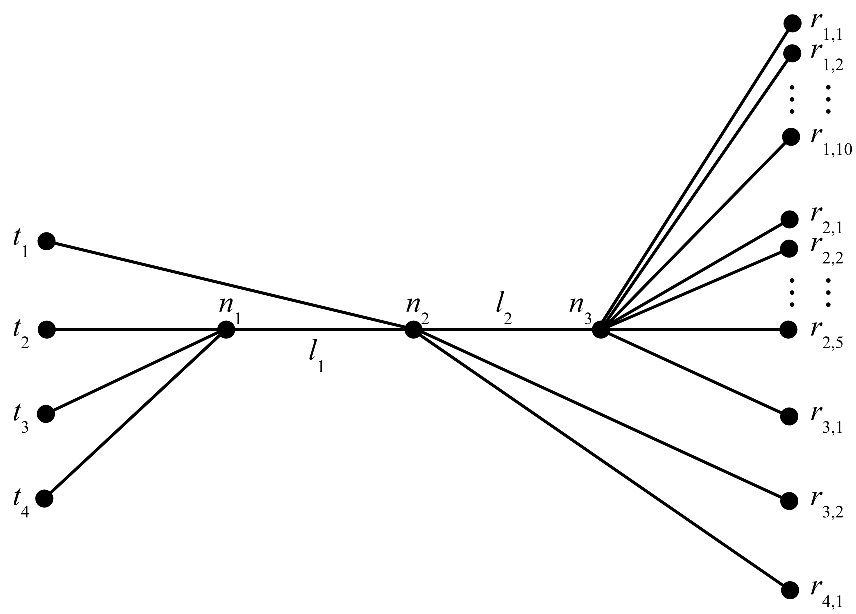


Figure 3.4: The traffic pattern of the bandwidth-allocation example.

The MFMF Bandwidth Allocation

For such simple network topologies and traffic patterns, a MFMF bandwidth allocation can be obtained by evenly increasing the utilities of each session until all sessions have reached their fair share of bandwidth on some bottleneck links. The fair share of bandwidth of a session on a link is calculated, with the aim being that the multicast-favorable value (3.2) should be the same, and maximal, for all sessions with receivers downstream that link. Receivers located downstream of bottleneck links, which have become fully allocated during this process, will have reached their MFMF bitrate. As long as there are receivers which have not yet reached their MFMF bitrate, the whole procedure must be repeated for the links along the transmission paths to these receivers.

In this example, l_1 and l_2 are the only two links with more than one session competing for bandwidth. They are therefore the only links that require a closer study, on all other links the single session present will be entitled to a fair share equaling the entire link capacity of 10 Mbps. In an effort to minimize the jungle of indices, we simply let b_i represent the fair share of bandwidth of session s_i in any given allocation step.

On link l_1, the three present sessions are s_2 with five receivers, s_3 with two receivers, and s_4 with one receiver. Assuming that the sessions are not restricted by any other link, equations (3.2) and (3.3) give rise to the following fair shares of bandwidth

$$\begin{cases} c_1 = 10.00 \text{ Mbps} \\ b_2 + b_3 + b_4 = c_1 \\ \dfrac{b_2}{1+\ln(5)} = \dfrac{b_3}{1+\ln(2)} = \dfrac{b_4}{1+\ln(1)} \end{cases} \Rightarrow \begin{cases} b_2 \approx 4.92 \text{ Mbps} \\ b_3 \approx 3.19 \text{ Mbps} \\ b_4 \approx 1.89 \text{ Mbps.} \end{cases}$$

On link l_2, the three traversing sessions are s_1 with ten receivers, s_2 with five receivers, and s_3 with one receiver. Once again assuming that the sessions are not restricted by any other link, equations (3.2) and (3.3) yield the following fair shares of bandwidth on link l_2

$$\begin{cases} c_2 = 10.00 \text{ Mbps} \\ b_1 + b_2 + b_3 = c_2 \\ \dfrac{b_1}{1+\ln(10)} = \dfrac{b_2}{1+\ln(5)} = \dfrac{b_3}{1+\ln(1)} \end{cases} \Rightarrow \begin{cases} b_1 \approx 4.78 \text{ Mbps} \\ b_2 \approx 3.78 \text{ Mbps} \\ b_3 \approx 1.45 \text{ Mbps.} \end{cases}$$

In the first allocation step, the sessions are supposed to obtain bandwidth allocated according to the fair share on their bottleneck link. The resultant bandwidth allocation is therefore as follows

$$\begin{cases} b_1 \approx 4.78 \text{ Mbps} \\ b_2 \approx \min(4.92, 3.78) \approx 3.78 \text{ Mbps} \\ b_3 \approx \min(1.89, 1.45) \approx 1.45 \text{ Mbps} \\ b_4 \approx 1.89 \text{ Mbps.} \end{cases}$$

After this has been carried out, link l_2 is fully allocated. Since all receivers of session s_1 and s_2 are located downstream of link l_2, they have reached their MFMF bitrates and the sessions may therefore be excluded from the rest of the computation. This is also applicable to receiver $r_{3,1}$.

However, link l_1 still has some free capacity. Sessions s_3 and s_4 are now the only ones traversing link l_1 and have downstream receivers who are not restricted by any fully allocated link, i.e. $r_{3,2}$ and $r_{4,1}$. The allocation to these receivers on link l_1 may consequently be increased according to equations (3.2) and (3.3). That is,

$$\begin{cases} c_1 = 10.00 \text{ Mbps} \\ b_2 + b_{3,2} + b_{4,1} = c_1 \\ \dfrac{b_3}{1+\ln(2)} + \dfrac{b_{3,2} - b_3}{1+\ln(1)} = \dfrac{b_{4,1}}{1+\ln(1)} \end{cases} \Rightarrow \begin{cases} b_{3,2} \approx 3.41 \text{Mbps} \\ b_{4,1} \approx 2.81 \text{Mbps.} \end{cases}$$

The total capacity of both links l_1 and l_2 is now allocated and all receivers have reached their MFMF bitrates since they are restricted by one of these links.

The MFMF definition is consequently fulfilled when the receivers of sessions s_1, s_2, and s_4 obtain 4.78, 3.78, and 2.81 Mbps, whereas the two receivers of session s_3 obtain 1.45 and 3.41 Mbps, respectively.

The FFRD Policy

We assume that the sessions arrive in the order of their indices, which means that session s_1 is the first. Since session s_1 is alone in the network, the FFRD policy allocates the entire capacity of 10 Mbps to it on all the links that it traverses.

When session s_2 arrives, node n_2 splits the capacity of link l_2 according to equation (3.5). The ten receivers of s_1 and the five receivers of s_2 therefore obtain

$$\begin{cases} c_2 = 10.00 \text{ Mbps} \\ b_1 + b_2 = c_2 \\ \dfrac{b_1}{1+\ln(10)} = \dfrac{b_2}{1+\ln(5)} \end{cases} \Rightarrow \begin{cases} b_1 \approx 5.59 \text{ Mbps} \\ b_2 \approx 4.41 \text{ Mbps}, \end{cases}$$

respectively, where

$$\begin{cases} b_{1,k} = b_1 & \text{for } k = 1, 2, 3, \cdots, 10 \\ b_{2,k} = b_2 & \text{for } k = 1, 2, 3, 4, 5. \end{cases}$$

Now session s_3 arrives and the capacity on link l_2 has to be split in three. The receivers of sessions s_1 and s_2, and the first receiver of s_3 therefore obtain the following fair shares of the bandwidth

$$\begin{cases} c_2 = 10.00 \text{ Mbps} \\ b_1 + b_2 + b_{3,1} = c_2 \\ \dfrac{b_1}{1+\ln(10)} = \dfrac{b_2}{1+\ln(5)} = \dfrac{b_{3,1}}{1+\ln(1)} \end{cases} \Rightarrow \begin{cases} b_1 & \approx 4.78 \text{ Mbps} \\ b_2 & \approx 3.78 \text{ Mbps} \\ b_{3,1} & \approx 1.45 \text{ Mbps}, \end{cases}$$

However, now there are also two sessions competing for the capacity on link l_1. These sessions are s_2 whose five receivers are limited to obtain at most 3.78 Mbps by link l_2, and s_3 with two receivers, of which $r_{3,1}$ is limited to 1.45 Mbps. If we assume that these receivers will obtain bitrates according to their fair shares on link l_2, this means that for the bandwidth allocated on l_1,

$$\begin{cases} c_1 = 10.00 \text{ Mbps} \\ b_2 + b_{3,2} = c_1 \text{ Mbps} \\ b_2 \approx 3.78 \text{ Mbps} \end{cases} \Rightarrow b_{3,2} = 6.22 \text{ Mbps},$$

and consequently equation (3.5) evaluates into

$$\frac{b_2}{1+\ln(5)} < \frac{b_{3,1}}{1+\ln(2)} + \frac{b_{3,2} - b_{3,1}}{1+\ln(1)}.$$

It is therefore safe to say that the receivers of session s_2 will reach their fair share of bandwidth on link l_2 and that the assumptions were accurate.

Finally, session s_4 arrives and this firstly affects the allocation on link l_1. If we assume that the receivers of session s_2 will still reach their limit on link l_2, s_3 and s_4 will share the remaining 6.22 Mbps capacity on link l_1. We further assume that also receiver $r_{3,1}$ of session s_3 will reach its limit on link l_2. If the assumptions are incorrect, this will be visible in the subsequent calculations. According to equation (3.5), the bitrates that sessions s_3 and s_4 will obtain on link l_1 are

$$\begin{cases} c_1 = 10.00 \text{ Mbps} \\ b_2 \approx 3.78 \text{ Mbps} \\ b_{3,1} \approx 1.45 \text{ Mbps} \\ b_2 + b_{3,2} + b_4 = c_1 \\ \dfrac{b_{3,1}}{1+\ln(2)} + \dfrac{b_{3,2} - b_{3,1}}{1+\ln(1)} = \dfrac{b_4}{1+\ln(1)} \end{cases} \Rightarrow \begin{cases} b_{3,2} \approx 3.41 \text{ Mbps} \\ b_4 \approx 2.81 \text{ Mbps}, \end{cases}$$

respectively. This gives

$$\frac{b_2}{1+\ln(5)} < \frac{b_{3,1}}{1+\ln(2)} + \frac{b_{3,2}-b_{3,1}}{1+\ln(1)} = \frac{b_4}{1+\ln(1)},$$

and consequently, the assumptions proved to be correct.

The receivers of sessions s_1, s_2 and s_4 therefore obtain bitrates of 4.78, 3.78, and 2.81 Mbps, respectively. The receivers of session s_3 obtain different bitrates, $r_{3,1}$ obtains 1.45 Mbps, while $r_{3,2}$ obtains 3.41 Mbps. This bandwidth allocation is actually identical to that of the MFMF definition and consequently the FFRD policy is optimally fair in this example. The average obtained bitrate of the receivers becomes

$$\frac{10\cdot 4.78 + 5\cdot 3.78 + (1.45 + 3.41) + 2.81}{10+5+2+1} \approx 4.13 \text{ Mbps.}$$

The BFRD Policy

Once again assuming that the sessions arrive in index order, the BFRD policy results in the same allocations as the FFRD policy until session s_4 arrives. Further assume, as we did for the FFRD policy, that $r_{3,1}$ and the receivers of session s_2 will still obtain bitrates according to their fair shares on link l_2 when session s_4 arrives. According to equation (3.6), the bitrates that sessions s_3 and s_4 will then obtain on link l_1 are

$$\begin{cases} c_1 = 10.00 \text{ Mbps} \\ b_2 \approx 3.78 \text{ Mbps} \\ b_2 + b_{3,2} + b_4 = c_1 \\ \dfrac{b_{3,2}}{1+\ln(2)} = \dfrac{b_4}{1+\ln(1)} \end{cases} \Rightarrow \begin{cases} b_{3,2} \approx 3.91 \text{ Mbps} \\ b_4 \approx 2.31 \text{ Mbps.} \end{cases}$$

This gives that for the bandwidth allocated on link l_1,

$$\frac{b_2}{1+\ln(5)} < \frac{b_{3,2}}{1+\ln(2)} = \frac{b_{4,1}}{1+\ln(1)},$$

and thus the assumption was correct.

The result is that the bandwidth received by $r_{3,2}$ is increased from 3.41 to 3.91 Mbps, while the bandwidth allocated to $r_{4,1}$ is reduced from 2.81 to 2.31 Mbps, when compared to the FFRD allocation. All other receivers are unaffected. Using equation (3.4), this corresponds to an unfairness of

$$\sqrt{\frac{10\cdot 0^2 + 5\cdot 0^2 + 2\cdot 0^2 + (2.31 - 2.81)^2}{10+5+2+1}} \approx 0.119 \text{ Mbps.}$$

and an average obtained bitrate of

$$\frac{10\cdot 4.78 + 5\cdot 3.78 + (1.45 + 3.91) + 2.31}{10+5+2+1} \approx 4.13 \text{ Mbps.}$$

The LogRD Policy

The LogRD bandwidth-allocation policy does not employ bottleneck-information feedback. Node n_1 is therefore unaware that the bitrate obtained by all the receivers of session s_2 is limited to 3.78 Mbps by link l_2. Consequently, node n_1 according to equation (2.1) still attempts to allocate the whole fair share of

$$\frac{1+\ln(5)}{(1+\ln(5))+(1+\ln(2))+(1+\ln(1))} \cdot c_1 \approx 4.92 \text{ Mbps}$$

to session s_2 on link l_1. The result of this is that the bitrates obtained by receiver $r_{3,2}$ and $r_{4,1}$ are reduced to

$$\frac{1+\ln(2)}{(1+\ln(5))+(1+\ln(2))+(1+\ln(1))} \cdot c_1 \approx 3.19 \text{ Mbps}$$

and

$$\frac{1+\ln(1)}{(1+\ln(5))+(1+\ln(2))+(1+\ln(1))} \cdot c_1 \approx 1.89 \text{ Mbps},$$

respectively, when compared to the BFRD allocation. The allocation on link l_2 is not affected.

The unfairness of the LogRD allocation therefore becomes

$$\sqrt{\frac{10\cdot 0^2 + 5\cdot 0^2 + (0^2 + (3.19-3.41)^2) + (1.89-2.81)^2}{10+5+2+1}} \approx 0.225 \text{ Mbps},$$

whereas the average obtained bitrate adds up to

$$\frac{10\cdot 4.78 + 5\cdot 3.78 + (1.45+3.19) + 1.89}{10+5+2+1} \approx 4.07 \text{ Mbps}.$$

The RI Policy

The RI bandwidth-allocation policy simply splits the bandwidth evenly between all the sessions on every link, independently of the number of receivers. In this example, all links have a capacity of 10 Mbps. Further, all sessions traverse at least one link together with two other sessions on the paths to each of their receivers, and they traverse no link with more than two other sessions. All receivers will therefore obtain a bitrate of

$$\frac{10.0}{3} \approx 3.33 \text{ Mbps},$$

which is consequently also equal to the average obtained bitrate. The unfairness is calculated as

$$\sqrt{\frac{\left(\ 10(3.33-4.78)^2 + 5(3.33-3.78)^2 + (0^2 + (3.33-3.41)^2) + 0^2\ \right)}{10+5+2+1}} \approx 1.10 \text{ Mbps}.$$

Table 3.1: Unfairness and average obtained bitrate of the bandwidth-allocation policies in the example.

policy	unfairness (Mbps)	average obtained bitrate (Mbps)
MFMF	-	4.13
FFRD	0.000	4.13
BFRD	0.119	4.13
LogRD	0.225	4.07
RI	1.10	3.33

Comparison of the Results

The results, in terms of unfairness and average obtained bitrate, are compiled into Table 3.1. The conclusion is that for the traffic scenario of this example, the FFRD bandwidth-allocation policy is optimally fair, the BFRD policy is the second fairest, and they both produce the same average obtained bitrate. However, the BFRD policy does not always manage to match the average obtained bitrate of the FFRD policy. This becomes apparent if for instance a second receiver of session s_4 is attached to node n_2.

In the example, the LogRD policy is almost twice as unfair as the BFRD policy and it also produces a lower average obtained bitrate. The reason is that it does not utilize any kind of bottleneck-information feedback. The RI policy is clearly the worst performer, both in terms of fairness and the average obtained bitrate, since it does not prioritize multicast sessions.

For larger networks with complex traffic scenarios including large multicast groups, the differences between the policies would most probably be larger. Further, in such a scenario the FFRD policy would be unlikely to perform a completely MFMF bandwidth allocation, because of the lack in global information. However, it is not feasible to use a larger network as an example.

3.2.4 Evaluation of the Feedback Overhead

Although the details of the feedback procedure are beyond the scope of this thesis, the amount of feedback that the FFRD and BFRD bandwidth-allocation policies would require might still be compared. As for the LogRD policy, described in subsection 2.5.6, both policies must send information towards the source, with regards to the number of receivers downstream each link. The FFRD policy must also transmit information concerning the obtained bit rates of all downstream receivers, whereas the BFRD only requires information about the largest bottleneck link of the downstream receivers.

We start by pointing out that the FFRD policy might aggregate the information into a single packet at each node. This reasoning is valid for the assumption that the packet size[3] is sufficiently large to contain all the information, otherwise multiple packets are required. In either case, the overhead associated with an IP packet is typically small compared to the payload[4], and the actual number of packets that have to be transmitted is therefore disregarded in the following calculations. Further, with the FFRD policy, information regarding the number of downstream receivers might be deduced from the information concerning the obtained bit rates.

Without delving in any greater depth into this argument, let us assume that the representation of the number of downstream receivers occupies the same amount of data as the information about a bottleneck bandwidth or one received bit rate. We consider this to be one *information unit* (IU). For simplicity, let us also make the somewhat unlikely assumption that the multicast tree is an m-ary tree[5] with n levels.

For each feedback occasion, the number of transmitted packets then equals the number of branches. At level x, there are m^x branches, which means that the total is

$$\sum_{x=1}^{n} m^x$$

packets transmitted per feedback occasion. If the assumption is made that $m > 1$, i.e. that each internal node has more than one child so that the tree is not just a path to a single receiver, this geometric series evaluates to

$$\frac{m^{n+1} - m}{m - 1}. \tag{3.7}$$

This assumption is maintained throughout this section, since the opposite is of no significant interest.

The LogRD bandwidth-allocation policy requires one IU per feedback packet. According to equation (3.7),

$$I_{\text{LogRD}} = \frac{m^{n+1} - m}{m - 1} \text{ IU}$$

therefore has to be transmitted at each feedback occasion for the LogRD policy.

The BFRD bandwidth-allocation policy requires twice the number of IUs, when compared to the LogRD policy, since all feedback packets must contain information regarding both the number of downstream receivers and the largest bottleneck link. Consequently,

$$I_{\text{BFRD}} = 2\frac{m^{n+1} - m}{m - 1} \text{ IU}$$

per feedback occasion are required for the BFRD policy.

[3] The largest allowed size for IP packets sent over Ethernet, i.e. the *maximum transmission unit* (MTU), is 1500 bytes [142]. If UDP/IP is used, the payload would be sufficient to transmit information regarding the obtained bandwidth of 370 receivers in single-precision (32-bit) floating-point format.

[4] For UDP/IP the overhead is 20 bytes [58], i.e. 1.3% of the Ethernet MTU.

[5] An m-ary tree is a tree where every node, except for the leaf nodes, have m children.

Table 3.2: The amount of feedback information, measured in information units, required by the LogRD, BFRD, and FFRD bandwidth-allocation policies, and the complexity of these expressions.

policy	required amount of feedback	order/complexity as $m, n \to \infty$
LogRD	$\frac{m^{n+1} - m}{m - 1}$	$\mathcal{O}(m^n)$
BFRD	$2\frac{m^{n+1} - m}{m - 1}$	$\mathcal{O}(m^n)$
FFRD	nm^n or $nm^n + \frac{m^{n+1} - m}{m - 1}$	$\mathcal{O}(nm^n)$

At level x of the tree, the FFRD bandwidth-allocation policy has to transmit information regarding the obtained bandwidths of m^{n-x} receivers. For the entire tree, this adds up to

$$I_{\mathrm{FFRD}} = \sum_{x=1}^{n} m^x m^{n-x} = nm^n \text{ IU}.$$

If the number of receivers is deduced from the information about obtained bandwidths, this is sufficient. Otherwise, the amount of feedback information required by the FFRD policy is increased to

$$I_{\mathrm{FFRD}} = \sum_{x=1}^{n} m^x (m^{n-x} + 1) = nm^n + \frac{m^{n+1} - m}{m - 1} \text{ IU}.$$

If we study the complexity of the quantity of feedback information, we find that both

$$I_{\mathrm{LogRD}} \in \mathcal{O}(m^n) \quad \text{as } m, n \to \infty$$

and

$$I_{\mathrm{BFRD}} \in \mathcal{O}(m^n) \quad \text{as } m, n \to \infty,$$

whereas

$$I_{\mathrm{FFRD}} \in \mathcal{O}(nm^n) \quad \text{as } m, n \to \infty.$$

The number of information units, required for information feedback by each of the bandwidth-allocation policies, is compiled into Table 3.2, together with the complexity of these expressions. The height of the tree, n, i.e. the length of the transmission paths, is shown to have an approximately linear effect on the relation of the amount of feedback information transmitted by the FFRD policy, when compared to the LogRD and BFRD policies. This is because the FFRD policy aggregates the information on its way towards the root, whereas the other policies transmit a fixed amount of data per link. However, the width of the tree, m, does not affect the relationship to any greater extent.

3.3 Fragmented Sessions

The term fragmented session is herein used to describe a scenario where the transmission of some data, that belong together, is divided into a number of sessions with different source and/or destination addresses. These sessions are called subsessions and it should be noted that they may cause a problem for bandwidth-allocation policies. The reason for this is that they may be treated as individual sessions and thereby, when combined, obtain a larger amount of the bandwidth than they are supposed to.

3.3.1 Fragmented Unicast

Some of the peer-to-peer schemes that are used for file sharing, such as Swarmcast and the popular BitTorrent protocol [143], constitute a source of fragmented sessions. They utilize a file-sharing technique, where files are fragmented into a number of segments. When a user wants to download a file, its segments may be obtained from different servers. Numerous sources of the same session are therefore common, and each of them contributes with a subsession. However, the traffic consists solely of unicast transmissions, which means that each subsession will obtain a minimum share of the bandwidth.

3.3.2 Fragmented Multicast

Layered multicast, which was described in subsection 2.4.5, is another transmission scheme that causes fragmented sessions. The reason is that the main session is divided into a number of subsessions transmitted on separate multicast groups, each one contributing with quality enhancements in an hierarchically additive fashion. The consequence is that a layered multicast session will obtain as many “shares” of the bandwidth as it has layers.

3.3.3 Solutions

Since fragmented multicast is identified as the main problem, two solution to that scenario are presented, followed by a more general solution which would produce somewhat different results.

One solution is to check the source and destination addresses of each session and treat matching sessions jointly when the bandwidth is allocated. The implication from this is that information of the identities of the receivers of multicast traffic must be available in the router nodes. Such information must consequently be forwarded along the reverse multicast tree every time a receiver joins or leaves a multicast group. Another prerequisite is that all layers are transmitted by the same source, which is highly likely, but not necessarily true.

A different approach is to add information to the data packets, concerning the belonging of multicast layers. The advantage is that the receiver identities no longer have to be available in the router nodes. On the negative side, multiple sources of the multicast layers may still cause a problem, and the realization is completely left to the good will of the sources.

The most general solution to the problem of fragmented sessions would be to look exclusively at the identities of the receivers. If a receiver takes part in more than one session, traversing any particular link, its impact factor could be divided amongst these sessions during the bandwidth-allocation process. This would however affect all sessions that the receiver takes part in, not only the fragmented ones.

Chapter 4

Cost Allocation for Multicast Receivers

If the costs for multicast transmission were favorable when compared to those of unicast, this would create another incentive for the employment of multicast. In this chapter, we therefore study how the costs of multicast sessions should be allocated to achieve this goal.

To begin with, we adopt the fundamental assumption made by Herzog et al. in [132], that costs of multicast trees should be assigned to the receivers and not to the source. The reason is that multicast transmission is receiver initiated and that the service primarily is of use to the receivers, since the sources typically are streaming servers. The three basic requirements listed in subsection 2.7.1, no positive transfers, voluntary participation, and consumer sovereignty, are also sustained.

Further, we believe that fair cost allocation should be based on resource usage. This is likely to make the resource utilization more effective. With a flat-rate policy, there are no incentives for limiting the resource usage, as long as it is maintained within the postulated limit.

As an example, in everyday life, the expectation is that a train ticket will cost less than an air ticket. In addition, domestic flights are expected to cost less than international flights. Furthermore, a shared cab is cheaper per capita than a private one. The higher costs involved in more exclusive services together with a limited budget, probably accounts for the most common reason why people do not travel more, further, and faster, etc. A season ticket or the like, i.e. a flat rate policy, negates this incentive. However, other motives such as environmental awareness do exist.

For data transmission over computer networks, the two major resource-related factors, which might differ between receivers, also relate to distance and quality. Namely the transmission path and the QoS requirements. This is also true for multicast receivers, where the "shared-cab" aspect also comes into play. As an example, picking a server that is geographically close, or already has many receivers located in fairly close, will save resources and should therefore be cheaper.

In this chapter we firstly study existing cost-allocation mechanisms for multicast traffic, and the finding is that none take all of the aforementioned factors into consideration. A terminology for cost-allocation mechanisms that targets multi-rate multicast sessions is then introduced, whereupon two new cost-allocation mechanisms are proposed.

4.1 Evaluation of Existing Mechanisms

In this section, the cost-allocation mechanisms outlined in Section 2.6 and 2.7 are evaluated based on their attractiveness to the receivers. Important parameters are the magnitude of the costs and how fairly the costs are distributed.

4.1.1 The Edge-Pricing Paradigm

The edge-pricing paradigm [133], briefly described in subsection 2.6.2, possesses some attractive properties, and it appears to be based upon sound approximations. However, the authors do not specify the pricing policy to be used. This decision is left to the individual ISPs. There are two main classes of pricing policies; usage-based policies where users are charged based on their actual usage, and capacity-based or flat-rate policies, where the users pay for the desired capacity. The choice, in this case, was to focus on usage-based pricing policies, since they are more favorable to multicast sessions and also might be considered to be fairer.

4.1.2 Single QoS Cost Allocation

For usage-based pricing policies, the cost of a multicast session should be divided among the receivers. The receivers in a multicast group have unique transmission paths per definition, otherwise they would have been positioned at the same location. As outlined in subsection 2.6.3, Herzog et al. propose a couple of cost-allocation mechanisms that are based upon the individual receivers' transmission paths [132]. However, there is a second factor that might affect the amount of resources that are utilized by the individual receivers, namely the QoS requirements.

4.1.3 QoS-Based Cost Allocation

As stated in subsection 2.6.4, users may choose to subscribe to a service although they cannot utilize the entire data rate transmitted by the source. These users do not use the entire bandwidth allocated to a multicast session, and should therefore, from a usage-based pricing perspective, be allocated a smaller share of the costs.

Although the work of Herzog et al. presented in [132] is extensive, the case involving individual receivers of a multicast group requesting different levels of QoS is covered on less than half a page. The discussion is very general and no specific cost-allocation mechanism is proposed for these scenarios.

The QoS-D ETS cost-allocation mechanism described by Liu et al. in [134] does however represent this approach. The costs corresponding to each QoS level are aggregated over the entire multicast tree, and divided uniformly among the receivers obtaining that level or higher. Thus, the lengths of the individual transmission paths are not taken into consideration. The statement in [134] concerning the cost sharing being proportional to the individual receivers' resource requirements, if unicast had been used, is therefore not strictly true.

4.1.4 Game-Theoretic Approaches

In game-theoretic approaches, the bandwidth allocation is incorporated with the pricing procedure. However, we primarily aim at a cost-allocation mechanism that can fairly distribute the costs of an existing bandwidth allocation, e.g. produced by one of the BFRD and FFRD policies. The game-theoretic mechanisms are therefore ruled out.

4.1.5 Section Summary

To sum up this section, the game-theoretic approaches do not support cost-allocation of arbitrary bandwidth allocations, and none of the pure cost-allocation mechanisms takes both the transmission path and the QoS requirements into consideration. The mechanisms do not fully reflect the resource usage, and consequently there is room for improvements.

4.2 Terminology for Multicast Cost Sharing

As mentioned in subsection 2.6.4, the notations for cost-allocation functions targeting multicast sessions with differentiated QoS levels, introduced by Herzog et al. in [132] and outlined in subsection 2.6.1, are not well defined. Thus, the decision was made to interpret and extend the terminology, in order to better suit multicast sessions that provide multiple QoS levels. This will prove to be useful in the following section, where two new cost-allocation mechanisms are proposed.

We define n_u^q and n_d^q to be the number of upstream and downstream receivers of the q^{th} QoS level (QoS^q), and let z_u^q and z_d^q denote the total number of upstream and downstream receivers utilizing the information corresponding to QoS^q. That is,

$$z_u^q = \sum_{x=q}^{Q} n_u^x$$

and

$$z_d^q = \sum_{x=q}^{Q} n_d^x,$$

given that there are Q available QoS levels. We also define the vectors

$$\mathbf{z}_u = \left\{z_u^1, z_u^2, \ldots, z_u^Q\right\}$$

and

$$\mathbf{z}_d = \left\{z_d^1, z_d^2, \ldots, z_d^Q\right\}.$$

Further, Herzog et al. not only allow the cost-allocation functions to control the division of the costs between receivers requesting the same QoS level, but also the distribution of the total cost among the different QoS levels. On the contrary, our opinion is that the cost-allocation functions should be general and not influence the distribution of the cost among the QoS levels. This distribution should instead fully reflect the resource requirements of each QoS level and the corresponding pricing made by the ISP in question.

Consequently, the cost vector

$$\mathbf{c} = \left\{c^1, c^2, \ldots, c^Q\right\}$$

is introduced, where the additional costs for supporting QoS^q on a particular link during a specific period of time, when compared to those of QoS^{q-1}, are denoted by c^q. These costs should reasonably be split among the receivers requiring QoS^q or higher, and two cost-allocation subfunctions, $\mathrm{f}_u^q(z_u^q, z_d^q)$ and $\mathrm{f}_d^q(z_u^q, z_d^q)$, are introduced for this purpose. These subfunctions describe the shares of the additional costs, for supporting QoS^q level, that should be allocated to the receivers of QoS^q or higher, both upstream and downstream of the link in question. The total cost that is to be allocated to the upstream and downstream receivers of QoS^q may now be written as

$$\mathrm{C}_u^q(\mathbf{z}_u, \mathbf{z}_d, \mathbf{c}) = \sum_{x=1}^{q} \mathrm{f}_u^x(z_u^x, z_d^x)c^x \tag{4.1}$$

and

$$\mathrm{C}_d^q(\mathbf{z}_u, \mathbf{z}_d, \mathbf{c}) = \sum_{x=1}^{q} \mathrm{f}_d^x(z_u^x, z_d^x)c^x, \tag{4.2}$$

respectively.

The two cost-allocation functions $\mathrm{C}_u^q(\mathbf{z}_u, \mathbf{z}_d, \mathbf{c})$ and $\mathrm{C}_d^q(\mathbf{z}_u, \mathbf{z}_d, \mathbf{c})$ represent the actual cost, whereas the original cost-allocation functions $\mathrm{F}_u(z_u, z_d)$ and $\mathrm{F}_d(z_u, z_d)$ described the fraction of the total cost to be allocated to the users. The condition (2.5), regarding full cost allocation, is therefore no longer valid. Instead, for the costs corresponding to each QoS level to be fully allocated, the following equation

$$z_u^q \cdot \mathrm{f}_u^q(z_u^q, z_d^q) + z_d^q \cdot \mathrm{f}_d^q(z_u^q, z_d^q) \geq 1, \tag{4.3}$$

must be fulfilled for all integers q between one and Q, where Q is the highest QoS level with a receiver downstream of the link in question.

If equation (4.3) is an equality for all integers q between one and Q, this guarantees that the sum of all allocated costs equals the sum of the costs,

$$
\begin{aligned}
&\sum_{q=1}^{Q}\big(n_u^q\cdot \mathrm{C}_u^q(\mathbf{z}_u,\mathbf{z}_d,\mathbf{c}) + n_d^q\cdot \mathrm{C}_d^q(\mathbf{z}_u,\mathbf{z}_d,\mathbf{c})\big)\\
&= \sum_{q=1}^{Q}\left(n_u^q\cdot \sum_{x=1}^{q}\mathrm{f}_u^x(z_u^x,z_d^x)c^x + n_d^q\cdot \sum_{x=1}^{q}\mathrm{f}_d^x(z_u^x,z_d^x)c^x\right)\\
&= \Big(n_u^1\cdot \mathrm{f}_u^1(z_u^1,z_d^1)c^1 + n_d^1\cdot \mathrm{f}_d^1(z_u^1,z_d^1)c^1\Big)\\
&+\Big(n_u^2\cdot \big(\mathrm{f}_u^1(z_u^1,z_d^1)c^1 + \mathrm{f}_u^2(z_u^2,z_d^2)c^2\big) + n_d^2\cdot \big(\mathrm{f}_d^1(z_u^1,z_d^1)c^1 + \mathrm{f}_d^2(z_u^2,z_d^2)c^2\big)\Big) + \cdots\\
&\cdots + \Big(n_u^Q\cdot \big(\mathrm{f}_u^1(z_u^1,z_d^1)c^1 + \mathrm{f}_u^2(z_u^2,z_d^2)c^2 + \cdots + \mathrm{f}_u^Q(z_u^Q,z_d^Q)c^Q\big)\\
&+ n_d^Q\cdot \big(\mathrm{f}_d^1(z_u^1,z_d^1)c^1 + \mathrm{f}_d^2(z_u^2,z_d^2)c^2 + \cdots + \mathrm{f}_d^Q(z_u^Q,z_d^Q)c^Q\big)\Big)\\
&= \Big(\mathrm{f}_u^1(z_u^1,z_d^1)c^1\cdot \big(n_u^1 + n_u^2 + \cdots + n_u^Q\big) + \mathrm{f}_d^1(z_u^1,z_d^1)c^1\cdot \big(n_d^1 + n_d^2 + \cdots + n_d^Q\big)\Big)\\
&+ \Big(\mathrm{f}_u^2(z_u^2,z_d^2)c^2\cdot \big(n_u^2 + n_u^3 + \cdots + n_u^Q\big) + \mathrm{f}_d^2(z_u^2,z_d^2)c^2\cdot \big(n_d^2 + n_d^3 + \cdots + n_d^Q\big)\Big) + \cdots\\
&\cdots + \Big(\mathrm{f}_u^Q(z_u^Q,z_d^Q)c^Q\cdot n_u^Q + \mathrm{f}_d^Q(z_u^Q,z_d^Q)c^Q\cdot n_d^Q\Big)\\
&= \sum_{q=1}^{Q}\left(\mathrm{f}_u^q(z_u^q,z_d^q)c^q\cdot \sum_{x=q}^{Q} n_u^x + \mathrm{f}_d^q(z_u^q,z_d^q)c^q\cdot \sum_{x=q}^{Q} n_d^x\right)\\
&= \sum_{q=1}^{Q} c^q\cdot \big(z_u^q\cdot \mathrm{f}_u^q(z_u^q,z_d^q) + z_d^q\cdot \mathrm{f}_d^q(z_u^q,z_d^q)\big) = \sum_{q=1}^{Q} c^q,
\end{aligned}
\tag{4.4}
$$

which means that the cost-allocation mechanism is budget balanced.

As an example, consider the QoS-D ETS cost-allocation mechanism described in subsection 2.6.4. Using the terminology introduced in this section, it is be represented by cost-allocation subfunctions corresponding to the cost-allocation functions of the ETS mechanism (2.4)

$$
\mathrm{f}_u^q(z_u^q,z_d^q) = \mathrm{f}_d^q(z_u^q,z_d^q) = \frac{1}{z_u^q + z_d^q}.
$$

Consequently

$$
z_u^q\cdot \mathrm{f}_u^q(z_u^q,z_d^q) + z_d^q\cdot \mathrm{f}_d^q(z_u^q,z_d^q) = z_u^q\frac{1}{z_u^q + z_d^q} + z_d^q\frac{1}{z_u^q + z_d^q} = \frac{z_u^q + z_d^q}{z_u^q + z_d^q} = 1,
$$

and the QoS-D ETS mechanism is therefore budget balanced according to equation (4.4).

4.3 Fair Cost-Allocation Strategies

The evaluation of existing cost-allocation mechanisms in Section 4.1 was concluded with the realization that none were satisfactorily fair. The reason was that, at most, they consider one of the two main factors affecting the resource usage, i.e. the transmission path and the QoS requirements. Using the terminology introduced in Section 4.2, a new cost-allocation mechanism, which takes both these factors into consideration, is proposed in subsection 4.3.1.

Although the aim of this mechanism is to achieve optimum fairness, it might have one, possibly severe, shortcoming: Optimum fairness may not be the primary interest of the receivers, if it occurs at the expense of higher costs. If poor and greedy receivers get a discount on the service, it may actually become cheaper for the rest of the receivers. An alternative mechanism is therefore proposed in subsection 4.3.2.

4.3.1 QoS-Differentiated Link Split Downstream

The first proposal is designed to perform perfectly fair cost allocations, taking into consideration both the transmission path and the QoS requirements. It builds on the ELSD cost-allocation mechanism, presented by Herzog et al. in [132], but is enhanced to support differentiated QoS levels.

The cost-allocation subfunctions therefore correspond to equations (2.2) and (2.3), and become

$$\mathrm{f}_u^q(z_u^q, z_d^q) = 0 \tag{4.5}$$

and

$$\mathrm{f}_d^q(z_u^q, z_d^q) = \frac{1}{z_d^q}, \tag{4.6}$$

respectively. This gives that

$$z_u^q \cdot \mathrm{f}_u^q(z_u^q, z_d^q) + z_d^q \cdot \mathrm{f}_d^q(z_u^q, z_d^q) = z_u^q \cdot 0 + z_d^q \frac{1}{z_d^q} = \frac{z_d^q}{z_d^q} = 1,$$

and the cost-allocation mechanism is consequently budget balanced according to equation (4.4).

Substituting equations (4.5) and (4.6) into (4.1) and (4.2), the main cost-allocation functions for receivers of QoS^q become

$$\mathrm{C}_u^q(\mathbf{z}_u, \mathbf{z}_d, \mathbf{c}) = 0 \tag{4.7}$$

and

$$\mathrm{C}_d^q(\mathbf{z}_u, \mathbf{z}_d, \mathbf{c}) = \sum_{x=1}^{q} \frac{c^x}{z_d^x}. \tag{4.8}$$

We call the cost-allocation mechanism described by equations (4.7) and (4.8), the *QoS-differentiated link split downstream* (QoS-D LSD) mechanism.

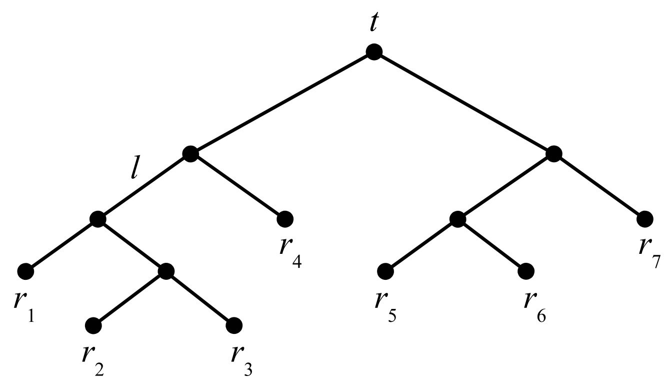


Figure 4.1: A multicast transmission tree with seven receivers.

Bandwidth-Differentiated Link Split Downstream

As observed in [132], in the extreme case, each receiver will have a QoS level of its own. This can be taken one step further, by assuming the bandwidth to be the predominant cost factor and considering the bandwidth consumption as a direct function of the QoS level. Let us also assume that the bandwidth is uniformly priced and costs c *monetary units* (MU) per *bitrate unit* (BU) and *time unit* (TU).

Let $\mathbf{b}$ be a vector whose first element $\mathbf{b}[0]$ is 0 and the n_d following elements are the receiving rates of the receivers downstream of the link in question, sorted in ascending order. The total cost per TU, allocated to the downstream receiver obtaining the q^{th} smallest bandwidth, may now be rewritten as

$$\mathrm{C}_d^q(n_d, \mathbf{b}) = c \sum_{x=1}^{q} \frac{\mathbf{b}[x] - \mathbf{b}[x-1]}{n_d - x + 1}. \tag{4.9}$$

The *bandwidth-differentiated link split downstream* cost allocation performed by equation (4.9) is only a special case of the QoS-D LSD mechanism.

A Cost-Allocation Example

As a small example of the QoS-D LSD mechanism, let us study how equation (4.9) allocates the cost of link l in Figure 4.1, where t is the transmitter and r_1 through r_7 are the receivers. For simplicity, we assume that receiver r_i obtains i BU for one TU, and that the bandwidth on link l costs one MU per BU and TU. Now we have

$$\begin{aligned} c &= 1 \\ n_d &= 3 \\ \mathbf{b} &= \{0, 1, 2, 3\}, \end{aligned}$$

Table 4.1: The obtained bitrates in BUs of the seven receivers in the example, together with the costs in MUs, allocated by the ETS, ELSD, QoS-D ETS, and QoS-D LSD cost-allocation mechanisms.

receiver	bitrate	ETS	ELSD	QoS-D ETS	QoS-D LSD
r_1	1	7.29	3.00	1.71	1.58
r_2	2	7.29	5.50	3.55	4.42
r_3	3	7.29	6.50	5.55	7.92
r_4	4	7.29	5.00	7.30	6.08
r_5	5	7.29	10.3	8.96	9.17
r_6	6	7.29	11.3	11.0	11.7
r_7	7	7.29	9.33	13.0	10.2

which when substituted into equation (4.9) give the cost of link l being allocated to receiver r_1, r_2, and r_3 as follows,

$$\mathrm{C}_d^1(n_d, \mathbf{b}) = \sum_{x=1}^{1} \frac{\mathbf{b}[x] - \mathbf{b}[x-1]}{4-x} = \frac{1}{3} \text{ MU},$$
$$\mathrm{C}_d^2(n_d, \mathbf{b}) = \sum_{x=1}^{2} \frac{\mathbf{b}[x] - \mathbf{b}[x-1]}{4-x} = \frac{1}{3} + \frac{1}{2} = \frac{5}{6} \text{ MU},$$

and

$$\mathrm{C}_d^3(n_d, \mathbf{b}) = \sum_{x=1}^{3} \frac{\mathbf{b}[x] - \mathbf{b}[x-1]}{4-x} = \frac{1}{3} + \frac{1}{2} + \frac{1}{1} = \frac{11}{6} \text{ MU}.$$

If we, similarly, calculate the total costs allocated to receiver r_1, r_2, and r_3, link by link from the source, they become

$$\left(\frac{1}{4}\right) + \left(\frac{1}{3}\right) + \left(\frac{1}{1}\right) = \frac{19}{12} \text{ MU},$$
$$\left(\frac{1}{4} + \frac{1}{3}\right) + \left(\frac{1}{3} + \frac{1}{2}\right) + \left(\frac{2}{2}\right) + \left(\frac{2}{1}\right) = \frac{53}{12} \text{ MU},$$

and

$$\left(\frac{1}{4} + \frac{1}{3} + \frac{1}{2}\right) + \left(\frac{1}{3} + \frac{1}{2} + \frac{1}{1}\right) + \left(\frac{2}{2} + \frac{1}{1}\right) + \left(\frac{3}{1}\right) = \frac{95}{12} \text{ MU},$$

respectively. To make the calculations easier to follow, the costs are presented for every bandwidth interval, and costs arising from the same link are grouped together by parentheses.

The costs allocated to all the seven receivers in the multicast tree are presented in Table 4.1, together with the corresponding costs produced by the ETS, ELSD, and QoS-D ETS cost-allocation mechanisms.

The ETS and ELSD mechanisms were not designed with differentiated QoS demands in mind. Both these mechanisms will therefore generally allocate disproportionately large parts of the cost to receivers with low QoS demands. The ETS mechanism simply splits the aggregated cost of the entire multicast tree equally among all the receivers, and is therefore also unfair towards receivers with short transmission paths. The ELSD mechanism only splits the link costs among downstream receivers, and the receivers that are treated most unfairly are consequently those with low QoS demands, compared to the receivers with whom they share the links. Examples of such mistreated receivers are consequently r_1, r_2, and r_5.

The QoS-D ETS mechanism performs differently, as it is now the receivers with short transmission paths, such as r_4 and r_7, that are treated unfairly. The situation is worst for r_7, which obtains the highest QoS level, and therefore has to share the costs of the entire multicast tree.

4.3.2 Bid-Based Link Split Downstream

As mentioned previously, the proposed QoS-D LSD cost-allocation mechanism attempts to achieve optimum fairness, but it has one possibly severe shortcoming: Optimum fairness may not be the primary interest of the receivers if it is at the expense of higher costs. If poor and greedy receivers get a discount on the service, it may actually become cheaper for the rest of the receivers. Here we further investigate this issue and propose an alternative cost-allocation mechanism that solves the shortcoming.

We start by drawing a parallel to an everyday situation. Children and/or retired people often receive a discount on the entrance fee to sport events, festivals, and museums etc. Most people are willing to accept this since it typically does not negatively affect their fees. As long as the events are not sold out, the economy of the organizers might actually benefit from this, and thereby allow them to also lower the standard fees[1].

However, if the scenario was the opposite and the attendance of discounted groups had a negative influence on standard fees, i.e. forcing the regular visitors to subsidize those on discounted rates, few would be happy about accepting such a system. Consumer goods are seldom discounted in this manner, since they are associated with specific material and production costs.

If we look at the game-theoretic approaches of Section 2.7, the SV mechanism allocates the costs in a LSD manner, and therefore shares the aforementioned shortcoming. The MC mechanism on the other hand does not require the receivers to cover more than their marginal cost. It is consequently not budget balanced, and may thereby produce a financial deficit for the ISPs.

We propose a bid-based cost-allocation mechanism, where fair cost allocation according to the QoS-D LSD mechanism is retained as the target. However, bids that do not cover the receivers' fair shares of the costs, but do cover at least the *additional*

[1] If any organizers actually do this in reality is a completely different question.

Table 4.2: Possible outcomes of a placed bid, with a certain maximum cost, for the BB LSD cost-allocation mechanism.

relative size of the maximum cost of the bid	served	allocated cost
maximum cost < additional cost	no	–
additional cost ≤ maximum cost < fair share	yes	maximum cost
fair share ≤ maximum cost	yes	fair share

cost associated with receivers' requests, are also accepted. That is, the additional cost for providing the receiver with the requested service, compared to the cost of providing the service to the existing set of receivers.

The main difference between marginal cost and additional cost is that the latter is dependent upon the order of the arrival of the bids which, in turn, guarantees that the proposed mechanism is budget balanced. However, although an expansion of the user set never causes increased costs for users within the original set, the mechanism is not cross monotonic, since it only applies to ordered sets of users.

A placed bid consequently leads to one of the outcomes described in Table 4.2. The fair share is the cost calculated according to the principles of the QoS-D LSD mechanism, with the addendum that if some poor receivers are discounted, these costs have to be carried by the wealthier receivers. The costs not covered by a receiver are distributed between the affected links and QoS levels of the existing transmission tree, proportional to that receiver's fair cost shares, and are split among the higher-bidding receivers utilizing these resources. The proposed mechanism is called *bid-based link split downstream* (BB LSD).

Bid Structure

There are a number of parameters that a bid must contain, namely:

- the maximum acceptable cost of the transmission
- the requested duration of the transmission
- the requested QoS level of the transmission
- the TTL of the bid

It is insufficient to replace the maximum cost and requested duration with a maximum cost per TU. This would prevent the calculation of other receivers' maximum costs, since these are affected by receivers who leave the service prematurely. There is also a possibility of non-recurrent costs associated with setting up the service. The bid TTL is required since most users are only interested in a particular service if it can be started within a given amount of time.

A receiver may request a service at a particular price, but be willing to settle for a poorer QoS level at a lower price if the main bid cannot be accepted. The main bid

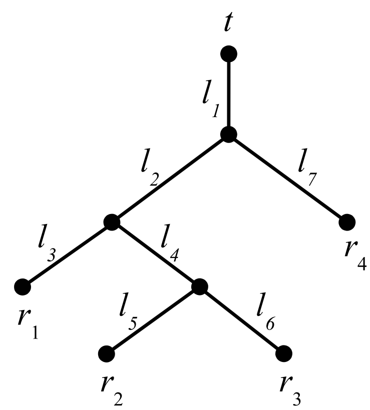


Figure 4.2: The multicast transmission tree of the example in subsection 4.3.3.

Table 4.3: The bids of the receivers in the example in subsection 4.3.3. The maximum cost is measured in MU.

receiver	requested QoS	maximum cost
r_1	QoS^1	25
r_2	QoS^2	25
r_3	QoS^3	80
r_4	QoS^4	100

could then possibly remain effective during its TTL, in case the costs associated with it were to be reduced. We observe that there may be as many subbids as there are QoS levels, but do not discuss these composite bids any further.

Strategyproofness

The BB LSD mechanism is not strategy proof. There is an obvious risk that users place dishonestly low bids, i.e. bids that do not correspond to their estimated value of the service, in an attempt to find the minimum cost of the service. To avoid this destructive behavior for the system, we propose an upper limit on the bid frequency of any particular receiver. This might not make the mechanism strategy proof, but it should make users more honest, since a lower bid equals a higher risk of missing out on the service for a particular amount of time.

The problem of finding a sufficient maximum bid frequency is a weighing of the honesty of the bids against the adaptability of the mechanism. It is possible that the economic prerequisites of a receiver change for the better after a low bid has been placed. An alternative to a fixed maximum bid frequency, is to exponentially increase the period of time until a new bid might be placed or considered.

4.3.3 A Cost-Allocation Example

The transmission tree in Figure 4.2 is used as an example in order to shed some light on the possible advantages of the BB LSD cost-allocation mechanism. The requested QoS levels are outlined in Table 4.3, together with the maximum total cost that the receivers are willing to pay for the service. For simplicity, assume that all requests concern the same duration, say 10 TU, and that the bandwidth on all links cost one MU per BU and TU. Further assume that the bitrate is the predominant cost factor and that QoS^q constantly requires q BU. The incremental cost of transmitting QoS^q, when compared to that of QoS^{q-1}, is consequently one MU/TU per link.

In the two first subsections, the QoS-D LSD and MC cost-allocation mechanisms are utilized to allocate the bandwidth and costs, and in the third subsection, these parameters are calculated according to the proposed BB LSD mechanism. For the latter mechanism, the order of arrival of the bids is essential. For simplicity, we base the order on the receiver numbers, and assume the arrivals of the bids to be sufficiently closely spaced in time for the requested transmissions to be considered simultaneously from a cost-sharing perspective. The results of the cost-allocation mechanisms are compared in the last subsection.

Allocation According to QoS-D LSD

We start by studying how the QoS-D LSD mechanism would allocate the cost of link l_2, under the assumption that all receivers are able to obtain the requested service at prices not exceeding their maximum costs. According to equation (4.8), receiver r_1 will be charged

$$\frac{10}{3} \approx 3.33 \text{ MU}$$

for receiving QoS^1, since there are three receivers utilizing this information. In the same manner, the cost of link l_2 allocated to receivers r_2 and r_3, which are requesting QoS^2 respectively QoS^3, become

$$\frac{10}{3} + \frac{10}{2} \approx 8.33 \text{ MU}$$

and

$$\frac{10}{3} + \frac{10}{2} + \frac{10}{1} \approx 18.33 \text{ MU}.$$

The cost of each receiver can be calculated link by link from the source. The total costs of receivers r_1 through r_4 then become

$$\left(\frac{10}{4}\right) + \left(\frac{10}{3}\right) + \left(\frac{10}{1}\right) \approx 15.83 \text{ MU},$$

$$\left(\frac{10}{4} + \frac{10}{3}\right) + \left(\frac{10}{3} + \frac{10}{2}\right) + \left(\frac{10}{2} + \frac{10}{2}\right) + \left(\frac{10}{1} + \frac{10}{1}\right) \approx 44.17 \text{ MU}, \qquad (4.10)$$

$$\left(\frac{10}{4}+\frac{10}{3}+\frac{10}{2}\right)+\left(\frac{10}{3}+\frac{10}{2}+\frac{10}{1}\right)+ \\ \left(\frac{10}{2}+\frac{10}{2}+\frac{10}{1}\right)+\left(\frac{10}{1}+\frac{10}{1}+\frac{10}{1}\right)\approx 79.17\ \mathrm{MU}, \tag{4.11}$$

and

$$\left(\frac{10}{4}+\frac{10}{3}+\frac{10}{2}+\frac{10}{1}\right)+\left(\frac{10}{1}+\frac{10}{1}+\frac{10}{1}+\frac{10}{1}\right)=60.83\ \mathrm{MU},$$

respectively. To make the calculations easier to follow, the costs arising from the same link are grouped by parentheses.

Apparently, the assumption that all receivers are able to obtain the service, at a cost not exceeding their maximum limits, was false. Receiver r_2 is only willing to pay 25 MU, but would be charged over 44 MU. It will therefore not obtain the service, and the rest of the receivers will consequently have to cover a larger part of the costs on the shared links. Receivers r_1, r_3, and r_4 will now be charged

$$\left(\frac{10}{3}\right)+\left(\frac{10}{2}\right)+\left(\frac{10}{1}\right)\approx 18.33\ \mathrm{MU},$$

$$\left(\frac{10}{3}+\frac{10}{2}+\frac{10}{2}\right)+\left(\frac{10}{2}+\frac{10}{1}+\frac{10}{1}\right)+ \\ \left(\frac{10}{1}+\frac{10}{1}+\frac{10}{1}\right)+\left(\frac{10}{1}+\frac{10}{1}+\frac{10}{1}\right)\approx 103.33\ \mathrm{MU},$$

respectively

$$\left(\frac{10}{3}+\frac{10}{2}+\frac{10}{2}+\frac{10}{1}\right)+\left(\frac{10}{1}+\frac{10}{1}+\frac{10}{1}+\frac{10}{1}\right)\approx 63.33\ \mathrm{MU}.$$

Consequently, the cost allocated to receiver r_3 exceeds its bid of 80 MU, and it will also fail to obtain the requested service. The costs of receivers r_1 and r_4 are increased accordingly to

$$\left(\frac{10}{2}\right)+\left(\frac{10}{1}\right)+\left(\frac{10}{1}\right)=25.00\ \mathrm{MU}$$

and

$$\left(\frac{10}{2}+\frac{10}{1}+\frac{10}{1}+\frac{10}{1}\right)+\left(\frac{10}{1}+\frac{10}{1}+\frac{10}{1}+\frac{10}{1}\right)=75.00\ \mathrm{MU},$$

respectively. These costs are covered by the receivers' bids.

Allocation According to MC

The MC cost-allocation mechanism has received its name because it allocates the marginal cost to each user. The marginal cost of a user is the additional cost of providing the service to that user, when compared to the cost of providing the service to the remaining set of users.

In this example the marginal cost of receiver r_1 corresponds to that of QoS^1 on link l_3, i.e. 10 MU, since r_2 and r_3 also utilize QoS^1 on the rest of the transmission path from the source to r_1. On link l_1, QoS^1 is also utilized by receiver r_4.

In the same manner, the marginal cost of receiver r_2 is derived from the provision of QoS^2 on link l_5, that is 20 MU. On the rest of the transmission path from the source to r_2, QoS^2 is shared by receiver r_3.

Receiver r_3 is allocated the total cost for QoS^3 on its last hop link l_6, which corresponds to 30 MU. Further, on links l_2 and l_4, r_3 is the only receiver that utilizes QoS^3. It therefore has to cover the additional cost of QoS^3, when compared to that of QoS^2, on these links. This implies a cost of 10 MU per link. However, r_3 does not have to contribute to the costs of l_1, since QoS^3 is shared with receiver r_4 on that link. The aggregated cost allocated to receiver r_3 is consequently 50 MU.

Finally, receiver r_4 is charged with the total cost of QoS^4 on link l_7 and the additional cost of QoS^4 on link l_1. This adds up to a total of 50 MU, and all receivers will therefore be served since the maximum costs of their bids cover the allocated costs.

Allocation According to the BB LSD Mechanism

Now the proposed BB LSD cost-allocation mechanism is applied to the same example.

When the bid of receiver r_1 is placed, its maximum cost of 25 MU is insufficient to cover the cost of the requested QoS^1, which is calculated as being 30 MU over the three-link transmission path from the source. The bid is therefore not accepted, but remains effective, pending other bids that may share the costs.

When the bid of receiver r_2 arrives, the costs associated with its request for QoS^2 is 80 MU. The bid is on 25 MU, and can therefore also not be accepted, not even when considered jointly with the bid of r_1.

Then the bid of receiver r_3 is placed. It concerns QoS^3 and is worth 80 MU, whereas the cost for offering the service is 120 MU. The total cost for serving r_1, r_2, and r_3 would be 150 MU, whereas their joint means are calculated as being 130 MU. Separately considering r_1 and r_3, or r_2 and r_3, does not make the situation more favorable.

Finally, the bid of receiver r_4 is placed. The costs for the requested transmission to r_4 is 80 MU, and the bid on 100 MU can therefore be accepted on its own. However, to decide what costs will actually be allocated to r_4, the bids of the other receivers must first be reconsidered.

Let us start by considering receiver r_2. The costs of the resources that r_2 must cover in total, i.e. those of link l_5, are 20 MU according to the last parenthesis of equation (4.10). It therefore has 5 MU left to contribute to the cost sharing on the upstream links. These 5 MU will be split uniformly according to r_2's fair shares of the costs on these links, which corresponds to the remaining first three parenthesis of (4.10). This results in

$$5 \cdot \frac{10/4}{(10/4 + 10/3) + (10/3 + 10/2) + (10/2 + 10/2)} \approx 0.52 \text{ MU}$$

for QoS^1 on l_1, and in the same manner approximately 0.69 MU for QoS^2 on l_1 and QoS^1 on l_2, and 1.03 MU for QoS^2 on l_2, and QoS^1 and QoS^2 on l_4.

Receiver r_3 must cover the entire 30 MU for link l_6 and the remaining costs on l_4. Further, it also has to cover the additional cost of QoS^3 on l_2 together with the remaining cost for QoS^2. Consequently, there are approximately

$$80 - 30 - (30 - 2 \cdot 1.03) - (20 - 1.03) \approx 3.09 \text{ MU}$$

left on the bid of r_3. Split uniformly according to r_3's remaining costs shares, which can be found in equation (4.11), this yields

$$3.09 \cdot \frac{10/4}{(10/4 + 10/3 + 10/2) + (10/3)} \approx 0.55 \text{ MU}$$

for QoS^1 on link l_1, and in the same manner approximately 0.73 MU for QoS^2 on l_1, 1.09 MU for QoS^3 on l_1, and 0.73 MU for QoS^1 on l_2.

Consequently, receiver r_1 that only requested QoS^1, has to cover 10 MU on link l_3 and the remaining costs on l_2, which is approximately

$$10 - 0.69 - 0.73 = 8.58 \text{ MU}.$$

On link l_1, r_1 will be charged its own fair share of the costs, plus its share of the costs for QoS^1 that are not covered by r_2 and r_3. This adds up to

$$\frac{10}{4} + \frac{10/4 - 0.52}{2} + \frac{10/4 - 0.55}{2} \approx 4.47 \text{ MU}.$$

The total cost allocated to r_1 thereby aggregates into approximately

$$10 + 8.58 + 4.47 = 23.05 \text{ MU}.$$

The remaining costs, which are allocated to receiver r_4, are calculated as being 40 MU for link l_7, and approximately

$$(10 - 4.47 - 0.52 - 0.52) + (10 - 0.69 - 0.73) + (10 - 1.09) + 10 = 31.98 \text{ MU}$$

for link l_1, where each QoS level is accounted for separately. This gives a total cost for receiver r_4 of approximately 71.98 MU.

Comparison of Results

In Table 4.4, the outcomes for the receivers with the proposed BB LSD cost-allocation mechanism are presented together with them of MC and QoS-D LSD.

Table 4.4: The outcomes for the receivers with the QoS-D LSD, MC and BB LSD cost-allocation mechanisms. The costs are measured in MU.

	QoS-D LSD		MC		BB LSD	
receiver	served	cost	served	cost	served	cost
r_1	yes	25.0	yes	10.0	yes	23.0
r_2	no	–	yes	20.0	yes	25.0
r_3	no	–	yes	50.0	yes	80.0
r_4	yes	75.0	yes	50.0	yes	72.0

Table 4.5: The announced costs of the provided services and the generated incomes, both measured in MU, with the QoS-D LSD, MC and BB LSD cost-allocation mechanisms.

	QoS-D LSD	MC	BB LSD
announced service costs	100	200	200
generated incomes	100	130	200

The most obvious difference between the BB LSD and QoS-D LSD mechanisms is that receivers r_2 and r_3 are served by BB LSD but not by QoS-D LSD, since they cannot fully cover their fair shares of the costs. As a consequence, the costs allocated to receivers r_1 and r_4 are somewhat lower for the BB LSD mechanism, where receiver r_3 contributes to the cost sharing on links l_1 and l_2. Another, more significant effect, which is apparent in Table 4.5, is that the income of the ISP is doubled through the use of the BB LSD mechanism.

The BB LSD and MC mechanisms serve the same user sets. However, all the receivers are allocated lower costs by using the MC mechanism, since it only charges the marginal costs. As can be seen in Table 4.5, the result is, if not a financial deficit, at least a 70 MU reduction of the ISP's revenue, when compared to the budget-balanced BB LSD mechanism.

Chapter 5

Summary

Video-streaming services are rapidly gaining in popularity, and the quality of these services is also increasing. Internet video already has attracted a large crowd, but the quality leaves more to wish for. IPTV is being deployed on a wider extent and the transition to HDTV resolution is ongoing. In the longer run, 3D video and FVV services will also be offered. This development produces challenges for computer networks of all sizes, from small LANs to the whole Internet.

The employment of multicast transmission can reduce the resource demands of services where some content is simultaneously transmitted to a number of users. The reason is that the receivers of a multicast session share the resources through a common transmission tree, where data are only transmitted once along each branch. Nevertheless, multicast transmission is not deployed to its full extent.

This thesis has aimed at more efficient usage of bandwidth in IP networks. The area that has been targeted is the slow deployment of multicast transmission. Two proposals that produce incentives for the employment of multicast have therefore been presented.

The first proposal concerns the bandwidth-allocation process. The core idea is that multicast sessions with many receivers should be favored and obtain a larger portion of the bandwidth. This would also lead to an increase in average user satisfaction. The details were presented in Chapter 3 and the corresponding summary is outlined in Section 5.1.

The second proposal was to reduce the costs for users of multicast sessions. The cost reduction is brought about by the resource savings offered by the bandwidth sharing. The work was described in Chapter 4, and the corresponding summary is to be found in Section 5.2.

5.1 Bandwidth Allocation

A definition of fair allocation of bandwidth, which favors multicast sessions, has been presented. The definition is named as *multicast-favorable max-min fairness* (MFMF) and bases the favoring upon the bitrates that the individual receivers are able to obtain. We state that the MFMF definition can be used to represent a fair distribution of the bandwidth for any particular network, given that correct choices of multicast-favorable function and utility functions are made.

It has also been shown how the MFMF definition can be used for the evaluation of the fairness of other bandwidth allocations. However, a problem that remains unsolved is how to find the MFMF allocation, given any network topology and traffic pattern. This problem becomes very complex for large networks with high traffic intensities. It might even be NP hard, although this has not as yet been proved.

Further, two bandwidth-allocation policies that favor multicast sessions have been presented. They employ feedback regarding the bottleneck links of downstream receivers to produce allocations that according to the fairness measure are relatively close to those of the MFMF definition. The *full-feedback and receiver dependent* (FFRD) policy produces the fairest bandwidth allocations, but at the same time requires more overhead than the *bottleneck-feedback and receiver dependent* (BFRD) policy, in terms of feedback.

The problem with fragmented sessions has also been addressed. Fragmented sessions can have a negative effect on the fairness of bandwidth-allocation policies, since what is in fact a single session may be treated as multiple sessions and thereby cause the receipt of a larger amount of the bandwidth than should be the case. Some possible solutions were presented, which built on the use of a specific session identifier or information about the actual receivers of multicast sessions.

5.1.1 Future Work

Future work in the bandwidth-allocation area could focus on proving that MFMF bandwidth allocations are, or are not, NP hard to calculate. If they are not NP hard, a theoretical model or algorithm for calculating MFMF allocations could be developed.

Future work may also include the implementation of the BFRD and FFRD bandwidth-allocation policies in real networks and experiments with different multicast-favorable and utility functions.

5.2 Cost Allocation

Fair cost sharing among multicast receivers has been addressed. This would favor the multicast receivers under the assumption that fair cost sharing should be based upon resource usage. Two major resource-related factors were observed; the transmission path and the bandwidth or QoS requirements. Existing cost-allocation mechanisms for multicast were evaluated, but none took both these parameters into consideration. The *quality-of-service-dependent link split downstream* (QoS-D LSD) cost-allocation mechanism was therefore proposed. It considers both the transmission path and the QoS requirements, in order to achieve optimum fairness.

However, optimum fairness might not be in the best interest of the users, when it is at the expense of higher costs. An alternative cost-allocation mechanism, called *bid-based link split downstream* (BB LSD), was therefore proposed. The BB LSD mechanism enabled the users to place bids for a requested service, revealing their maximum acceptable cost. A bid that does not cover the user's fair share of the costs for the requested service is accepted if it does cover at least the additional cost associated with the request. This guarantees that the BB LSD mechanism is budget balanced. The result is not only a possible reduction in the costs for the rest of the users, but also an increase in revenue for the ISPs, which are able to serve more users.

Unfortunately, the BB LSD mechanism is not strategy proof. To avoid users seeking the minimum cost by placing dishonestly low bids, an upper limit on the bid frequency of any particular receiver was therefore proposed. Another alternative would be an exponentially growing time out in the case of a rejected bid. This should make the users more honest, i.e. to bid closer to what the service is worth to them, since a lower bid equals a higher risk of missing out on the service.

5.2.1 Future Work

Future research about cost-allocation mechanisms may involve the problem of finding a sufficient maximum bid frequency, or other procedures to mitigate the fact that the BB LSD mechanism is not strategy proof. Another alternative might be the search for a completely new mechanism that is naturally strategy proof and still possesses as many of the BB LSD mechanism's attractive properties as possible.

Further research topics are the implementation of the QoS-D LSD and BB LSD cost-allocation mechanisms, and the process of actually charging the receivers with the allocated costs.

5.3 Concluding Remark

The proposed bandwidth-allocation policies and cost-allocation mechanisms make multicast transmission more attractive to the users. If these proposals were to be implemented in multicast-enabled computer networks, the demand for ISPs that support multicast would increase. This would force the ISPs not only to deploy multicast functionality in their networks, but also to exchange multicast traffic with each other, to stay competitive.

The main consequence of an increased deployment of multicast would be a reduced traffic load on the IP networks. This would be of benefit to the Internet community as a whole, from ISPs and other service providers to the end users.

Bibliography

[1] YouTube, http://www.youtube.com.

[2] YouTube serves up 100 million videos a day online. USA Today, July 16, 2006. Retrieved June 28, 2007. [Online]. Available: http://www.usatoday.com/tech/news/2006-07-16-youtube-views_x.htm

[3] YouTube visits larger than rivals combined: survey. Reuters, June 28, 2007. Retrieved June 28, 2007. [Online]. Available: http://www.reuters.com/article/technologyNews/idUSN2742598120070628

[4] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.

[5] "IPTV bandwidth study," Bell Labs Research, March 2006.

[6] "Preparing for the IPTV future," Micronas, Whitepaper, June 2006.

[7] Joost, http://www.joost.com.

[8] K. Bodecek and V. Novotny, "From standard definition to high definition migration in current digital video broadcasting," in *Proceedings of IEEE International Multi-Conference on Computing in the Global Information Technology (ICCGI)*, Guadeloupe, French Caribbean, March 2007.

[9] A. Smolic, K. Mueller, P. Merkle, C. Fehn, P. Kauff, P. Eisert, and T. Wiegand, "3D video and free viewpoint video - technologies, applications and MPEG standards," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, Toronto, ON, Canada, July 2006, pp. 2161–2164.

[10] LG unveils 3D FLATRON monitor. LG press release, January 17, 2007. Retrieved July 19, 2007. [Online]. Available: http://hk.lge.com/ir/html/ABboards.do?action=read&group_code=AB&list_code=PRE_MENU&seq=5277&page=1&target=pressreleases_read.jsp

[11] L. Aguilar, "Datagram routing for Internet multicasting," *ACM SIGCOMM Computer Communication Review*, vol. 14, no. 2, pp. 58–63, June 1984.

[12] S. E. Deering, "Host extensions for IP multicasting," Network Working Group, RFC 1112, August 1989.

[13] C. Diot, B. N. Levine, B. Lyles, H. Kassem, and D. Balensiefen, "Deployment issues for the IP multicast service and architecture," *IEEE Network*, vol. 14, no. 1, pp. 78–88, January/February 2000.

[14] K. C. Almeroth, "The evolution of multicast: From the mbone to interdomain multicast to Internet2 deployment," *IEEE Network*, vol. 14, no. 1, pp. 10–20, January/February 2000.

[15] P. Rajvaidya and K. C. Almeroth, "Analysis of routing characteristics in the multicast infrastructure," in *Proceedings of 22nd IEEE International Conference INFOCOM*, vol. 2, San Francisco, CA, USA, March 2003, pp. 1532–1542.

[16] Multicast Status Web Page. Retrieved August 28, 2007. [Online]. Available: http://www.multicasttech.com/status/index.html

[17] Internet2 multicast working group goals for 2007. Internet2 Multicast Working Group. Retrieved February 7, 2007. [Online]. Available: http://multicast.internet2.edu/wg-multicast-goals-for-2007.shtml

[18] B. G. Haskell, A. Puri, and A. N. Netravali, *Digital Video: An Introduction to MPEG-2*, ser. Digital Multimedia Standards. New York, NY, USA: Chapman & Hall, 1997.

[19] M.-T. Sun and A. R. Reibman, Eds., *Compressed Video over Networks*, ser. Signal Processing and Communications. New York, NY, USA: Marcel Decker, 2001.

[20] J. Watkinson, *MPEG-2*. Oxford, England: Focal Press, 1999.

[21] "Information technology – Generic coding of moving pictures and associated audio information," ISO/IEC 13818, 1995.

[22] R. C. Gonzalez and R. E. Woods, *Digital image processing*. Reading, MA, USA: Addison Wesley, 1993.

[23] Y. Wang, J. Ostermann, and Y.-Q. Zhang, *Video processing and communications*, ser. Signal Processing. Upper Saddle River, NJ, USA: Prentice Hall, 2002.

[24] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 4, pp. 117–130, October 1996.

[25] S. Bajaj, L. Breslau, and S. Shenker, "Uniform versus priority dropping for layered video," *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 4, pp. 131–143, October 1998.

[26] P. A. Chou, A. E. Mohr, A. Wang, and S. Mehrotra, "Error control for receiver-driven layered multicast of audio and video," *IEEE Transactions on Multimedia*, vol. 3, no. 1, pp. 108–122, March 2001.

[27] A. Puri and A. H. Wong, "Spatial domain resolution scalable video coding," in *SPIE Visual Communications and Image Processing*, vol. 2094, Boston, MA, USA, October 1993.

[28] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 301–317, March 2001.

[29] P. Österberg, D. Forsgren, and T. Zhang, "Receiver-controlled joint source/channel coding on the application level, for video streaming over wlans," in *Proceedings of* 57^{th} *IEEE Semiannual IEEE Semiannual Vehicular Technology Conference (VTC)*, vol. 3, Jeju, Korea, April 2003, pp. 1558–1561.

[30] Stunning 3D viewing, spectacle without spectacles. Philips leaflet, May 2, 2007. Retrieved July 19, 2007. [Online]. Available: http://www.business-sites.philips.com/assets/Downloadablefile//2006-05-30-42inch-3D-display-leaflet-13552.pdf

[31] LG först i världen med att leverera massproducerad äkta 3D-upplevelse. LG press release, December 21, 2006. Retrieved July 19, 2007. [Online]. Available: http://nordic.lge.com/sv/ir/html/ABboards.do?action=read&group_code=AB&list_code=PRE_MENU&seq=5447&page=1&target=pressreleases_read.jsp

[32] J. Kilner, J. Starck, and A. Hilton, "A comparative study of free-viewpoint video techniques for sports events," in *Proceedings* 3^{rd} *IET European Conference on Visual Media Production (CVMP)*, London, Great Britain, November 2006.

[33] "Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5mbit/s," ISO/IEC 11172, 1993.

[34] "Information technology – Coding of audio-visual objects," ISO/IEC 14496, 1998.

[35] "Information technology – Coding of audio-visual objects – Part 2: Visual, Amendment 4: Streaming video profile," ISO/IEC 14496-2:1999/PDAM 4, March 2000.

[36] MPEG debutes scalable video coding. MPEG press release, July 6, 2007. Retrieved July 23, 2007. [Online]. Available: http://www.chiariglione.org/mpeg/meetings/lausanne07/lausanne_pr.htm

[37] "Information technology – Coding of audio-visual objects – Part 16: Animation framework extension (AFX)," ISO/IEC 14496-16:2006, December 2006.

[38] "H.261: Video codec for audiovisual services at p x 384 kbit/s," ITU-T Recommendation H.261 (11/88), November 1988.

[39] "H.263: Video coding for low bit rate communication," ITU-T Recommendation H.263 (03/96), March 1996.

[40] A. MacAulay, B. Felts, and Y. Fisher, "IP streaming of MPEG-4: native RTP vs MPEG-2 transport stream," Envivio, Whitepaper, October 2005.

[41] G. Andersson. IPTV – teknikskifte i klass med mobiltelefonin. Elektroniktidningen, May 21, 2007. Retrieved October 17, 2007. [Online]. Available: http://www.etn.se/index.php?option=com_content&task=view&id=22034&Itemid=87

[42] "Internet video for IPTV service providers," Multimedia Research Group, Quarterly Technology & Content Report, September 2007.

[43] Tv. Telia. Retrieved October 19, 2007. [Online]. Available: http://www.telia.se/privat/produkter_tjanster/tv?sl=teliase_entre_tv

[44] Digital-tv. Bredbandsbolaget. Retrieved October 19, 2007. [Online]. Available: http://www.bredband.com/wps/portal/privat/digitaltv?page=new&WCM_GLOBAL_CONTEXT=/wps/wcm/connect/b2/privat/digitaltv

[45] R. Garay. Pay-per-view. The Museum of Broadcast Communication. Retrieved October 17, 2007. [Online]. Available: http://www.museum.tv/archives/etv/P/htmlP/payperview/payperview.htm

[46] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," Network Working Group, RFC 3550, July 2003.

[47] M. Baugher, D. McGrew, M. Naslund, E. Carrara, and K. Norrman, "The secure real-time transport protocol (SRTP)," Network Working Group, RFC 3711, March 2004.

[48] H. Fuchs, "IPTV standardization and architecture," presented at the Television Reborn Workshop at Mid Sweden University, Sundsvall, Sweden, April 2007.

[49] Dynamic synchronous transfer mode (DTM) fundamentals and network solutions. International Engineering Consortium, Turorial. Retrieved October 1, 2007. [Online]. Available: http://www.iec.org/online/tutorials/dtm_fund/

[50] Asynchronous transfer mode (ATM) fundamentals. International Engineering Consortium, Turorial. Retrieved October 1, 2007. [Online]. Available: http://www.iec.org/online/tutorials/atm_fund/

[51] E. C. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," Network Working Group, RFC 3031, January 2001.

[52] J. Postel, "Internet protocol, darpa internet program protocol specification," Information Sciences Institute, University of Southern California, RFC 791, September 1995.

[53] Y. Rekhter, R. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address allocation for private internets," Network Working Group, RFC 1918, February 1996.

[54] S. Deering and R. Hinden, "Internet protocol, version 6 (IPv6) specification," Network Working Group, RFC 1883, December 1995.

[55] J. Hawkinson and T. Bates, "Guidelines for creation, selection, and registration of an autonomous system (AS)," Network Working Group, RFC 1930, March 1996.

[56] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers," Network Working Group, RFC 2474, December 1998.

[57] J. Postel, "DOD standard, transmission control protocol," Information Sciences Institute, University of Southern California, RFC 761, January 1980.

[58] ——, "User datagram protocol," Information Sciences Institute, University of Southern California, RFC 768, August 1980.

[59] R. Braden, D. Clark, and S. Shenker, "Integrated services in the Internet architecture: an overview," Network Working Group, RFC 1633, June 1994.

[60] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource reservation protocol (RSVP) – version 1 functional specification," Network Working Group, RFC 2205, September 1997.

[61] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," Network Working Group, RFC 2475, December 1998.

[62] S. E. Deering, "Host extensions for IP multicasting," Network Working Group, RFC 988, July 1986.

[63] ——, "Host extensions for IP multicasting," Network Working Group, RFC 1054, May 1988.

[64] W. Fenner, "Internet group management protocol, version 2," Network Working Group, RFC 2236, November 1997.

[65] B. Cain, S. E. Deering, I. Kouvelas, W. C. Fenner, and A. Thyagarajan, "Internet group management protocol, version 3," Network Working Group, RFC 3376, October 2002.

[66] H. Holbrook and B. Cain, "Source-specific multicast for IP," Network Working Group, RFC 4607, August 2006.

[67] S. Bradner and V. Paxson, "IANA allocation guidelines for values in the Internet protocol and related headers," Network Working Group, RFC 2780, March 2000.

[68] Z. Albanna, K. Almeroth, D. Meyer, and M. Schipper, "IANA guidelines for IPv4 multicast address assignments," Network Working Group, RFC 3171, August 2001.

[69] D. Thaler, M. Handley, and D. Estrin, "The Internet multicast address allocation architecture," Network Working Group, RFC 2908, September 2000.

[70] S. R. Hanna, B. V. Patel, and M. Shah, "Multicast address dynamic client allocation protocol (MADCAP)," Network Working Group, RFC 2730, December 1999.

[71] M. Handley, "Multicast address allocation protocol (AAP)," IETF, Internet Draft, December 1997, expired.

[72] P. Radoslavov, D. Estrin, R. Govindan, M. Handley, S. Kumar, and D. Thaler, "The multicast address-set claim (MASC) protocol," Network Working Group, RFC 2909, September 2000.

[73] P. Savola, "Overview of the Internet multicast addressing architecture," IETF, Internet Draft, October 2006, work in progress.

[74] D. Meyer and P. Lothberg, "Glop addressing in 233/8," Network Working Group, RFC 3180, September 2001.

[75] B. Haberman and D. Thaler, "Unicast-prefix-based ipv6 multicast addresses," Network Working Group, RFC 3306, August 2002.

[76] D. Thaler, "Unicast-prefix-based IPv4 multicast addresses," Network Working Group, Internet Draft, October 2004, expired.

[77] D. Meyer, "Administratively scoped IP multicast," Network Working Group, RFC 2365, July 1998.

[78] W. Liming and D. Estrin, "Multicast routing in dense and sparse modes: simulation study of tradeoffs and dynamics," in *Proceedings of 4th IEEE International Conference on Computer Communications and Networks (ICCCN)*, September 1995, pp. 150–157.

[79] A. Adams, J. Nicholas, and W. Siadak, "Protocol independent multicast – dense mode (PIM-DM): Protocol specification (revised)," Network Working Group, RFC 3973, January 2005.

[80] W. Fenner, M. Handley, H. Holbrook, and I. Kouvelas, "Protocol independent multicast – sparse mode (PIM-SM): Protocol specification (revised)," Network Working Group, RFC 4601, August 2006.

[81] M. Handley, I. Kouvelas, T. Speakman, and L. Vicisano, "Bi-directional protocol independent multicast (BIDIR-PIM)," IETF, Internet Draft, February 2007, work in progress.

[82] D. Waitzman, C. Partridge, and S. Deering, "Distance vector multicast routing protocol," Network Working Group, RFC 1075, November 1988.

[83] T. Bates, R. Chandra, D. Katz, and Y. Rekhter, "Multiprotocol extensions for BGP-4," Network Working Group, RFC 2283, February 1998.

[84] W. Fenner and D. Meyer, "Multicast source discovery protocol (MSDP)," Network Working Group, RFC 3618, October 2003.

[85] D. Thaler, "Border gateway multicast protocol (BGMP):protocol specification," Network Working Group, RFC 3913, September 2004.

[86] S. E. Deering, "Internet multicast routing: State of the art and open research issues," presented at the Multimedia Integrated Conferencing for Europe (MICE) Seminar at the Swedish Institute of Computer Science, Stockholm, Sweden, October 1993.

[87] Y. R. Yang, M. S. Kim, and S. S. Lam, "Optimal partitioning of multicast receivers," in *Proceedings of 8th IEEE International Conference on Network Protocols (ICNP)*, Osaka, Japan, November 2000, pp. 129–140.

[88] J. Liu, B. Li, and Y.-Q. Zhang, "A hybrid adaptation protocol for TCP-friendly layered multicast and its optimal rate allocation," in *Proceedings of 21th IEEE International Conference INFOCOM*, vol. 3, New York, NY, USA, June 2002, pp. 1520–1529.

[89] I. Rimac, J. Schmittt, and R. Steinmetz, "Is dynamic multi-rate multicast worthwhile the effort?" in *Proceedings of 28th IEEE Conference EUROMICRO*, Dortmund, Germany, September 2002, pp. 233–239.

[90] A. Majumdar, D. G. Sachs, I. V. Kozintsev, K. Ramchandran, and M. M. Yeung, "Multicast and unicast real-time video streaming over wireless LANs," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 524–534, June 2002.

[91] B. Rajagopalan, "Reliability and scaling issues in multicast communication," *ACM SIGCOMM Computer Communication Review*, vol. 22, no. 4, pp. 188 – 198, October 1992.

[92] Y. Xu and T. Zhang, "An adaptive redundancy technique for wireless indoor multicasting," in *Proceedings of 5th IEEE Symposium on Computers and Communications (ISCC)*, Antibes, France, July 2000, pp. 607–614.

[93] M. Luby, L. Vicisano, J. Gemmell, L. Rizzo, M. Handley, and J. Crowcroft, "Forward error correction (FEC) building block," Network Working Group, RFC 3453, December 2002.

[94] B. Adamson, C. Bormann, M. Handley, and J. Macker, "Negative-acknowledgment (NACK)-oriented reliable multicast (NORM) protocol," Network Working Group, RFC 3940, November 2004.

[95] W. tian Tan and A. Zakhor, "Video multicast using layered FEC and scalable compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 373–386, March 2001.

[96] Q. Zhang, Q. Guo, Q. Ni, W. Zhu, and Y.-Q. Zhang, "Sender-adaptive and receiver-driven layered multicast for scalable video over the Internet," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 4, pp. 482–495, April 2005.

[97] Y. hua Chu, S. G. Rao, and H. Zhang, "A case for end system multicast," *ACM SIGMETRICS Performance Evaluation Review*, vol. 28, no. 1, pp. 1–12, June 2000.

[98] Z. Li and P. Mohapatra, "HostCast: A new overlay multicasting protocol," in *Proceedings of 38th IEEE International Conference on Communications (ICC)*, vol. 1, Anchorage, AK, USA, May 2003, pp. 702–706.

[99] D. Milic, M. Brogle, and T. Braun, "Video broadcasting using overlay multicast," in *Proceedings of 7th IEEE International Symposium on Multimedia (ISM)*, Irvine, CA, USA, December 2005.

[100] Y. Zhu, J. Guo, and B. Li, "oEvolve: Toward evolutionary overlay topologies for high-bandwidth data dissemination," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 7, pp. 1237–1251, September 2004.

[101] B. Zhang, S. Jamin, and L. Zhang, "Host multicast: a framework for delivering multicast to end users," in *Proceedings of 21st IEEE International Conference INFOCOM*, vol. 3, New York, NY, USA, June 2002, pp. 1366–1375.

[102] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. James W. O'Toole, "Overcast: Reliable multicasting with on overlay network," in *Proceedings of 4th Conference on Symposium on Operating System Design & Implementation (OSDI)*, vol. 4, San Diego, CA, USA, October 2000.

[103] Y. Chawathe, "Scattercast: An adaptable broadcast distribution framework," *Springer Journal on Multimedia Systems*, vol. 9, no. 1, pp. 104–118, July 2003.

[104] A. Mankin, A. Romanow, S. Bradner, and V. Paxton, "IETF criteria for evaluating reliable multicast transport and application protocols," Network Working Group, RFC 2357, June 1998.

[105] M. Handley and S. Floyd, "Strawman specification for TCP friendly (reliable) multicast congestion control (TFMCC)," Reliable Multicast Research Group, Meeting, December 1998.

[106] J. Widmer and M. Handley, "TCP friendly multicast congestion control (TFMCC): Protocol specification," Network Working Group, RFC 4654, August 2006.

[107] D. M. Chiu, "Some observations on fairness of bandwidth sharing," in *Proceedings of 5th IEEE Symposium on Computers and Communications (ISCC)*, Antibes, France, July 2000, pp. 125–131.

[108] D. Bertsekas and R. Gallager, *Data Networks*. Upper Saddle River, NJ, USA: Prentice-Hall, 1987.

[109] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.

[110] A. Legout, J. Nonnenmacher, and E. W. Biersack, "Bandwidth-allocation policies for unicast and multicast flows," *IEEE/ACM Transactions on Networking*, vol. 9, no. 4, pp. 464–478, August 2001.

[111] R. Gopalakrishnan, J. Griffioen, G. Hjálmtýsson, and C. J. Sreenan, "Stability and fairness issues in layered multicast," in *Proceedings of the 9th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Basking Ridge, NJ, USA, June 1999.

[112] A. Matrawy, I. Lambadaris, and C. Huang, "On layered video fairness on IP networks," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 4, San Antonio, TX, USA, November 2001, pp. 2627–2633.

[113] D. Rubenstein, J. Kurose, and D. Towsley, "The impact of multicast layering on network fairness," *IEEE/ACM Transactions on Networking*, vol. 10, no. 2, pp. 169–182, April 2002.

[114] W. K. Lai and C. Y. Pan, "Achieving inter-session fairness for layered video multicast," *IEEE Transactions on Broadcasting*, vol. 48, no. 3, pp. 215–222, September 2002.

[115] L. Vicisano and J. C. L. Rizzo, "TCP-like congestion control for layered multicast data transfer," in *Proceedings of 17th IEEE International Conference INFOCOM*, vol. 3, San Francisco, CA, USA, March 1998, pp. 996–1003.

[116] J. W. Byers, G. Horn, M. Luby, M. Mitzenmacher, and W. Shaver, "FLID-DL: congestion control for layered multicast," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1558–1570, October 2002.

[117] T. Jiang, M. H. Ammar, and E. W. Zegura, "Inter-receiver fairness: a novel performance measure for multicast ABR sessions," *ACM SIGMETRICS Performance Evaluation Review*, vol. 26, no. 1, pp. 202–211, June 1998.

[118] S. Sarkar and L. Tassiulas, "Fair allocation of utilities in multirate multicast networks: a framework for unifying diverse fairness objectives," *IEEE Transactions on Automatic Control*, vol. 47, no. 6, pp. 931–944, June 2002.

[119] ——, "Fair bandwidth allocation for multicasting in networks with discrete feasible set," *IEEE Transactions on Computers*, vol. 53, no. 7, pp. 785–797, July 2004.

[120] ——, "Fair distributed congestion control in multirate multicast networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 1, pp. 121–133, February 2005.

[121] J. von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton, NY, USA: Princeton University Press, 1944.

[122] J. F. Nash, "The bargaining problem," *Econometrica*, vol. 18, no. 2, pp. 155–162, April 1950.

[123] S. Shenker, "Fundamental design issues for the future Internet," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 7, pp. 1176–1188, September 1995.

[124] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Transactions on Networking*, vol. 8, no. 5, pp. 556–567, October 2000.

[125] L. Massoulié and J. Roberts, "Bandwidth sharing: objectives and algorithms," *IEEE/ACM Transactions on Networking*, vol. 10, no. 3, pp. 320–328, June 2002.

[126] H. Luo, S. Lu, V. Bharghavan, J. Cheng, and G. Zhong, "A packet scheduling approach to QoS support in multihop wireless networks," *Kluwer Journal on Mobile Networks and Applications*, vol. 9, no. 3, pp. 193–206, June 2004.

[127] A. Tang, J. Wang, and S. H. Low, "Is fair allocation always inefficient," in *Proceedings of 23rd IEEE International Conference INFOCOM*, vol. 1, Hong Kong, March 2004, pp. 35–45.

[128] H.-Y. Tzeng and K.-Y. Siu, "On max-min fair congestion control for multicast ABR service in ATM," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 545–556, April 1997.

[129] Z. Cao and E. W. Zegura, "Utility max-min: An application-oriented bandwidth allocation scheme," in *Proceedings of 18th IEEE International Conference INFOCOM*, vol. 2, New York, NY, USA, March 1999, pp. 793–801.

[130] H. J. Einsiedler, P. Hurley, B. Stiller, and T. Braun, "Charging multicast communications based on a tree metric," in *Proceedings of GI Multicast Workshop*, Braunschweig, Germany, May 1999.

[131] A. Bueno, P. Vila, and R. Fabregat, "Multicast extension of unicast charging for QoS services," in *Proceedings of 4th IEEE European Conference on Universal Multiservice Networks (ECUMN)*, Toulouse, France, February 2007, pp. 119–126.

[132] S. Herzog, S. Shenker, and D. Estrin, "Sharing the "cost" of multicast trees: An axiomatic analysis," *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 847–860, December 1997.

[133] S. Shenker, D. Clark, D. Estrin, and S. Herzog, "Pricing in computer networks: Reshaping the research agenda," *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 2, pp. 19–43, April 1996.

[134] C.-C. Liu, S.-C. Chang, and H.-H. Cheng, "Pricing and fee sharing for point to multipoint and quality guaranteed multicast services," in *Proceedings of 7th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, Iwate, Japan, July 2000, pp. 255–260.

[135] H. Moulin and S. Shenker, "Strategyproof sharing of submodular access costs: Budget balance versus efficiency," *Springer Journal on Economic Theory*, vol. 18, no. 3, pp. 511–533, 2001.

[136] S. Chawla, D. Kitchin, U. Rajan, R. Ravi, and A. Sinha, "Profit guaranteeing mechanisms for multicast networks," in *Proceedings of 4^{th} ACM Conference on Electronic Commerce (EC)*, San Diego, CA, USA, June 2003, pp. 190–191.

[137] M. Bläser, "Approximate budget balanced mechanisms with low communication costs for the multicast cost-sharing problem," in *Proceedings of 15^{th} ACM-SIAM Symposium on Discrete Algorithms (SODA)*, New Orleans, LA, USA, January 2004, pp. 625–195.

[138] J. Feigenbaum, C. H. Papadimitriou, and S. Shenker, "Sharing the cost of multicast transmissions," *Elsevier Journal of Computer and System Sciences*, vol. 63, no. 1, pp. 21–41, August 2001.

[139] M. Adler and D. Rubenstein, "Pricing multicasting in more flexible network models," *ACM Transactions on Algorithms*, vol. 1, no. 1, pp. 48–73, July 2005.

[140] N. Garg and D. Grosu, "Performance evaluation of multicast cost sharing mechanisms," in *Proceedings of 21^{st} IEEE International Conference on Advanced Networking and Applications (AINA)*, Niagara Falls, Canada, May 2007, pp. 901–908.

[141] H.-J. Byun and J.-T. Lim, "Max-min fair rate allocation scheme in multirate multicast networks using supervisory control," *IEICE Transactions on Communications*, vol. E89-B, no. 5, pp. 1678–1681, May 2006.

[142] C. Hornig, "A standard for the transmission of IP datagrams over ethernet networks," Network Working Group, RFC 894, April 1984.

[143] BitTorrent protocol specifications v1.0. BitTorrent.org. Retrieved January 19, 2007. [Online]. Available: http://www.bittorrent.org/protocol.html

Biography

Patrik Österberg was born on the 21st of February 1975 in Sundsvall, Sweden. He received his Master of Science in Electrical Engineering from Mid Sweden University, Sweden, in 2000, and the degree of Licentiate of Technology in Teleinformatics from the Royal Institute of Technology (KTH), Sweden, in 2005. In 2007, he picked up a best paper award for the paper "Fair Cost Sharing among Multicast Receivers", which was presented at the International Conference on Digital Telecommunications in San Jose, CA, USA.

During approximately half the year of 2007, Österberg has been employed as a development engineer at Acreo AB, a research institute specialized in microelectronics, optics, and communication technology.

Included Papers