Multi-source multicasting in IP/MPLS over WDM networks

Sona Hassani

2010

Sona, H. (2010). Multi-source multicasting in IP/MPLS over WDM networks. Master' s thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/46801>

<https://doi.org/10.32657/10356/46801>

Nanyang Technological University

Downloaded on 26 Apr 2023 18:21:03 SGT

Multi-Source Multicasting in IP/MPLS over WDM Networks

Sona Hassani

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Master of Engineering

2010

Acknowledgment

Many people have inspired and helped me during my graduate studies in Nanyang Technological University. I would like to take this opportunity to thank them all.

First, I would like to thank my advisor Professor Zhong Wende. This work would not have been possible without his support, encouragement, guidance and insightful feedback. I would also like to thank Associate Professor Sanjay Kumar Bose for his resourceful comments and guidelines.

My sincere thanks go to Professor Shum Ping, director of Network Technology Research Centre (NTRC), for all his support and warmth. I would also like to thank Dr. Zhang Feng for all his help and comments during my study in NTU. I extend my thanks to all the graduate students, research staff and supporting technicians of NTRC Communications Lab IV and Communications Lab III for their continuous support and the wonderful time we worked together.

My deepest gratitude goes to my parents and my sister. I never felt alone in any step of my life since they always have been there for me with all their support and encouragement.

Acknowledgment

Finally, I would like to thank NTU and division of communication at school of Electrical and Electronic Engineering for being a great host for my research activities.

Abstract

This study addresses the performance evaluation of several approaches for multi-source multicasting in IP/MPLS over WDM network. Traffic demand in communication networks is growing fast as a result of many bandwidth intensive applications such as video streaming and file sharing. With the introduction of Multi Protocol Label Switching (MPLS) technology, future Internet Protocol (IP) based networks will be able to provide Quality of Service (QoS) and traffic engineering. The need of sending the same information from one node to a group of selected nodes has grown with the emergence of many applications such as video conferencing, distance learning and multi-player interactive games. Multicasting is the technique used to efficiently provide such services.

In this study we introduce several possible approaches to establish a multi-source multicast session in an IP/MPLS over WDM network. We consider the overlay model in our study since it is more practical. Three main approaches are proposed for the establishment of a multi-source multicast session. These include multiple unidirectional trees approach, multi root-node approach and one bidirectional tree approach. The Multiple Unidirectional Trees approach (MUT) is to establish multiple unidirectional multicast trees each rooted at one of the multicast group members. The multi root-node approach is to establish one unidirectional tree rooted at a specific node called root-node and multiple unidirectional paths each from the multicast group members to the root-node. The

Abstract

third approach is to establish one bidirectional multicast tree to connect the multicast group members. We may reserve the maximum required bandwidth on each link of the tree for the sake of the simplicity of bandwidth reservation. We refer to this approach as one Bidirectional Tree with Maximum reserved bandwidth (BTM). In order to enhance the efficiency of the BTM approach, we further propose an algorithm to find the excessively reserved bandwidth on each link and release it. The new proposed approach is named Bidirectional Tree with Just enough reserved bandwidth (BTJ). The performance of the aforementioned approaches is evaluated in terms of blocking probability, time complexity and the memory consumption in the control plane. Simulation results show that among all the approaches, the newly proposed approach, known as BTJ, achieves the best overall performance.

Table of Contents

Acknowledgment i

Abstract iii

Table of Contents v

List of Figures viii

List of Acronyms xii

Chapter 1. Introduction 1

 1.1 Motivation 3

 1.2 Objectives 5

 1.3 Major Contribution of the Thesis 7

 1.4 Organization of the Thesis 8

Chapter 2. Literature Review 10

 2.1 IP/MPLS over WDM Network 10

 2.2 Multicasting 19

 2.2.1 Optical Multicasting 20

 2.2.2 IP Multicasting 24

 2.2.3 MPLS Multicasting 28

 2.3 Traffic Grooming 30

 2.4 Summary 34

Table of Contents

Chapter 3. Approaches of Multi-Source Multicasting	36
3.1 Introduction	36
3.2 Multiple Unidirectional Trees Approach (MUT)	39
3.3 Multi Root-Node Approach	40
3.3.1 Fixed Root-Node Approach	42
3.3.2 Random Selection of the Root-Nodes (RRN)	45
3.4 One Bidirectional Tree Approach	46
3.5 Bidirectional Tree with Just Enough Reserved Bandwidth (BTJ)	48
3.6 Summary	51
Chapter 4. Evaluation of Blocking Probability and Time Complexity	52
4.1 Network Model	52
4.2 Traffic Models	55
4.3 Algorithms	60
4.3.1 Multicast Traffic Grooming and Multicast Tree Construction	60
4.3.2 Choosing a Root-Node in the Multi Root-Node Approach	63
4.3.3 Calculating the Required Bandwidth in the BTJ Approach	67
4.4 Simulation Experiments and Results	70
4.4.1 Assumptions	70
4.4.2 Network Blocking Probability	72
4.4.3 Time Complexity	85

Table of Contents

4.5	Summary	92
Chapter 5. Evaluation of Memory Consumption in Multi-Source Multicasting		93
5.1	IB Table Structure for Different M2M Multicast Approaches	93
5.1.1	Multiple Unidirectional Trees Approach (MUT)	94
5.1.2	Root-Node Approach (RRN and FRN).....	95
5.1.3	One Bidirectional Tree Approach (BTM and BTJ)	97
5.2	Simulation Experiments	98
5.2.1	Assumptions.....	98
5.2.2	Simulation Results	99
5.3	Summary	103
Chapter 6. Conclusions and Recommendations.....		105
6.1	Summary	105
6.2	Recommendation for Further Research.....	108
References.....		111
Author's Publications Related to This Thesis.....		116
Appendix I		117

List of Figures

Fig. 2-1 An example of a 32-bit label.....	13
Fig. 2-2 Development for IP/MPLS over WDM solutions.....	14
Fig. 2-3 Network architecture in IP/MPLS over WDM	15
Fig. 2-4 Node architecture in IP/MPLS over WDM.....	15
Fig. 2-5 A lightpath in the optical network.....	16
Fig. 2-6 The network architecture in the peer model.....	17
Fig. 2-7 The network architecture for overlay model.	18
Fig. 3-1 MUT: Multiple Unidirectional Trees for one multicast session	40
Fig. 3-2 FRN: Fixed Root-Nodes.....	44
Fig. 3-3 RRN: Random selection of Root-Nodes	46
Fig. 3-4 BTM: Bidirectional multicast Tree with Maximum reserved bandwidth.	47
Fig. 3-5 An example showing the traffic flows on a link of a bidirectional multicast tree.....	49
Fig. 3-6 BTJ : Bidirectional Tree with Just enough reserved bandwidth	50
Fig. 4-1 Four network topologies for a given network	53
Fig. 4-2 Multicast tree construction flowchart.....	63
Fig. 4-3 Choosing one of the root-nodes randomly in FRN	64
Fig. 4-4 Choosing the nearest root-node to sender in FRN	65
Fig. 4-5 The NSF network topology	71
Fig. 4-6 The Cost239 network topology.....	71

List of Figures

Fig. 4-7 Blocking probability vs. effective traffic load for the NSF network, root-node selection (avg. multicast group size = 4.5).....	73
Fig. 4-8 Blocking probability vs. effective traffic load for the NSF network (avg. multicast group size = 4.5).....	75
Fig. 4-9 Blocking probability vs. effective traffic load for the Cost239 network (avg. multicast group size = 4.4).....	76
Fig. 4-10 Blocking probability vs. effective traffic load for the NSF network where partial blocking is not allowed (avg. multicast group size = 4.5).....	78
Fig. 4-11 Blocking probability vs. effective traffic load for the Cost239 network where partial blocking is not allowed (avg. multicast group size = 4.4)	79
Fig. 4-12 Blocking probability vs. unicast percentage for the NSF network (arrival rate = 2.5, avg. multicast group size = 6).....	80
Fig. 4-13 Blocking probability vs. unicast percentage for the Cost239 network (arrival rate = 7, avg. multicast group size = 6).....	81
Fig. 4-14 Blocking probability vs. avg. multicast group size for the NSF network (avg. arrival rate = 2)	82
Fig. 4-15 Blocking probability vs. avg. multicast group size for the Cost239 network (avg. arrival rate = 6)	83
Fig. 4-16 Blocking probability vs. avg. multicast group size for the NSF network where partial blocking is not allowed (avg. arrival rate = 2)	84
Fig. 4-17 Blocking probability vs. avg. multicast group size for the Cost239 network where partial blocking is not allowed (avg. arrival rate = 6)	85

List of Figures

Fig. 4-18 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. effective traffic load for the NSF network (avg. multicast group size = 4.5)	87
Fig. 4-19 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. effective traffic load for the Cost239 network (avg. multicast group size = 4.4)	88
Fig. 4-20 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. avg. multicast group size for the NSF network (avg. arrival rate = 2) .	89
Fig. 4-21 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. avg. multicast group size for the Cost239 Network (avg. arrival rate = 6)	90
Fig. 5-1 Information Base (IB) table that is stored in all nodes for the MUT	94
Fig. 5-2 Information Base (IB) table that is stored in root-nodes for FRN and RRN	95
Fig. 5-3 Information Base (IB) table that is stored in all nodes for FRN and RRN	96
Fig. 5-4 Information Base table that is stored for unicast paths in FRN and RRN	97
Fig. 5-5 Information Base (IB) table that is stored in all nodes for BTM and BTJ	98
Fig. 5-6 Avg. number of records for one multicast session vs. effective traffic load for the NSF network (avg. multicast group size = 4.5).....	99
Fig. 5-7 Avg. number of records for one multicast session vs. effective traffic load for the Cost239 network (avg. multicast group size = 4.4).....	101
Fig. 5-8 Avg. number of records for one multicast session vs. multicast group size for the NSF network (avg. arrival rate = 2)	102

List of Figures

Fig. 5-9 Avg. number of records for one multicast session vs. multicast group size
for the Cost239 network (avg. arrival rate = 6) 103

Fig. I Avg. number of copies $E(x)$ or fanout vs. distribution parameter
 q118

List of Acronyms

ATM	Asynchronous Transfer Mode
BGP	Border Gateway Protocol
CBT	Core-Based Trees
CCM	Centralized Control and Management
DNS	Domain Name System
DVMRP	Distance Vector Multicast Routing protocol
FDDI	Fiber Distributed Data Interface
FTP	File Transfer Protocol
IB	Information Base
IGMP	Internet Group Management Protocol
IGP	Interior Gateway Protocol
IP	Internet Protocol
LIB	Label Information Base
LSR	Label Switch Router
MEMS	Micro Electro-Mechanical System
MIDB	Multicast Information Database
MOSPF	Multicast Open Shortest Path First
MSDP	Multicast Source Discovery Protocol
OADM	Optical Add/Drop Multiplexers
OSPF	Open Shortest Path First
OXC	Optical Cross-Connects
PIM-DM	Protocol Independent Multicast-Dense Mode
PIM-SM	Protocol Independent Multicast-Sparse Mode
QoS	Quality of Service
RP	Rendezvous Point
SDH	Synchronous Digital Hierarchy
SONET	Synchronous Optical Networking

List of Acronyms

UNI	User Network Interface
WDM	Wavelength Division Multiplexing
WRS	Wavelength Routing Switch
WWW	World Wide Web

Chapter 1. Introduction

Traffic volume in communication networks is growing fast as a result of many bandwidth demanding applications such as video streaming, file sharing, video conferencing and online high quality games. Recently, online virtual worlds have also attracted a large portion of the Internet users. Most of these applications are also delay sensitive rather than only being bandwidth intensive. They need fast data transmission and data delivery guarantee. Therefore empowering the networks with high speed and large capacity is an imperative need.

Optical fibers provide the required huge capacity for the network. Wavelength Division Multiplexing (WDM) is the dominant technology to exploit the huge capacity offered by optical fibers and other optical components. WDM is based on using multiple sources each operating at slightly different wavelength. Multiple wavelengths are used to transmit different data streams over a single strand of optical fiber. The development of WDM technology has increased the transmission capacity to several Tb/s over a single strand of optical fiber [52] .

Even by the introduction of WDM and increasing the capacity per wavelength channel, the network's bottleneck problem which is the result of the exponential traffic growth is not solved. WDM only shifts the bottleneck problem from the link to the electronic router. One possible approach to solve this problem

would be to transfer IP traffic directly over optical networks by employing optical cross-connects (OXC) and Optical Add/Drop Multiplexers (OADMs) [47]. In this approach, some of the switching and routing functions are integrated into the optical domain instead of being performed electrically. In order to avoid the optical-electrical-optical conversion, an all optical path is established between the sender and receiver through the intermediate OXC or OADM, which is called lightpath [7]. A lightpath is considered as a single hop path although it may pass several OXC/OADM nodes.

WDM reduces the cost by providing large bandwidth capacity and the IP layer is a revenue generating layer which offers great convergence. IP routing algorithms in general search for the shortest path without considering overall network utilization. This might cause to overloading the shorter links while keeping the longer ones underutilized. With the introduction of Multi Protocol Label Switching (MPLS) technology, future IP based networks will be able to provide Quality of Service (QoS) and traffic engineering. They will not only support the traditional best effort services such as File Transfer Protocol (FTP), electronic mail (email), and World Wide Web (WWW) but will also provide delay sensitive services such as video on demand, video conferencing, and interactive online games. If this approach is followed, IP/MPLS over WDM will be the technology dominating future Internet expansion.

1.1 Motivation

In many present bandwidth intensive applications, it is required to send the same information from one node to a group of selected nodes. Since a considerable part of the paths between the source and the destination nodes may be common especially in the area around the source node, it would be inefficient to establish multiple unicast connections between the source node and the destination nodes. This is due to the fact that by establishing multiple unicast connections, multiple copies of the same packet might pass through the same link, resulting in inefficient use of network resources. *Multicasting* is a technique to send only one copy of each packet on the same link. In two situations multicasting would be very useful: the traffic volume be sufficiently high, and there be plenty of receivers [27]. Applications such as video conferencing, distance learning and online interactive games need to use multicasting in nature, because there is usually more than one destination to receive the same data at the same time. Obviously these applications are not broadcast applications since always a selected set of destination nodes are interested to receive the data generated by the source node.

Offering Quality of Service (QoS) is a necessity for this type of multicasting applications. For these applications, we are concerned about the entire duration of the application rather than a single moment. There should be an assurance that there is a reliable connection for the application's duration and sufficient amount of bandwidth is available. There are different strategies to establish multicast

sessions. Bandwidth guaranteed connections should be provided for a multicast session in order to provide QoS while exploiting the network resources.

So far most research on multicasting assumes that the source node is fixed within the multicast group. It means that within a multicast group, the data is being generated by one node only, and the other members of the multicast group always receive the generated data. However, in many present and future applications there is not only one fixed source of data. For example, in a video conferencing session, there could be a conversation between multiple members of the session and multiple members may speak simultaneously. Multiple source nodes may exist in a multicast session and they may transmit data simultaneously.

Although the capacity of optical fibers can be exploited using WDM technology, there is still a huge gap between the bandwidth required by one application and the capacity of a wavelength channel. In order to significantly enhance the utilization of optical wavelength channels in future IP over optical networks, low-speed traffic streams need to be efficiently multiplexed onto the high-speed wavelength channels [5],[8],[9]. Traffic grooming is to collect low speed traffic flows of different applications and pass them through a common link in order to utilize the huge capacity of the links [8]. The virtual topology consists of the lightpaths established in the optical network. The traffic grooming problem is closely related to the problem of virtual topology design according to the traffic distribution in the IP/MPLS layer [8].

1.2 Objectives

Deering proposed an IP multicast model in 1988 to provide multicast services over the Internet [29]. Since then multiple IP multicasting protocols have been designed and they are being used nowadays such as Core Based Trees (CBT) [40], Protocol Independent Multicast-Sparse Mode (PIM-SM) [41], Protocol Independent Multicast-Dense Mode (PIM-DM) [51], Distance Vector Multicast Routing Protocol (DVMRP) [50] and Multicast Source Discovery Protocol (MSDP) [42]. The lack of QoS and traffic engineering has been tackled by the merging of MPLS and IP.

In an IP/MPLS over WDM network a huge bandwidth is provided by the WDM layer. In order to exploit the bandwidth given, it is very important to know when and where a new lightpath should be established and also how to assign the bandwidth of the lightpaths to the multicast groups. There are multiple approaches for establishing multicast sessions, but most of them consider single source multicasting. Very little work has been carried out for multi-source multicasting. This study will address the multi-source multicasting problem. We consider that once a multicast session is established, every member of that multicast group is able to generate data and send it to the multicast group. Multicast group members may send data simultaneously.

The main objectives in this study are:

- To study and evaluate the pros and cons of different strategies of establishing multi-source multicast sessions. We will compare the performance of these different approaches from the aspect of network blocking probability.
- To investigate and propose a novel and practical method for establishing multi-source multicast sessions in IP/MPLS over WDM networks that provides better performance than the existing ones.
- We also aim to study and compare the time complexity of different multi-source multicasting approaches. In order to find an appropriate approach to establish a multicast session, only knowing the resource consumption of the approach is not enough. The time that is required to establish a multicast session is also an important factor to be considered.
- The information about multicast trees is kept in Information Base (IB) tables in the routers for routing and control decisions. We want to compare the volume of these tables for different approaches of establishing multi-source multicast sessions.
- To study an appropriate dynamic multicast traffic grooming policy to exploit the network's resources for multicasting traffic. Grooming low

bandwidth multicast traffic into wavelength channels will be important for efficiently using the future Internet.

1.3 Major Contribution of the Thesis

The contributions of this thesis are as follows:

- Introduction of three basic approaches to establish a multi-source multicasting session. In order to establish a multicast session in an IP/MPLS over WDM network, at least one multicast tree must be established and sufficient bandwidth must be reserved on the links of the tree to guarantee the availability of the required bandwidth for the entire application's running time. The three introduced approaches are
 - Multiple Unidirectional Trees (MUT) approach: establishing multiple unidirectional trees for one multicast session.
 - Root-node approach: establishing one unidirectional tree and multiple unicast paths for one multicast session (FRN and RRN).
 - One Bidirectional Tree with Maximum reserved bandwidth (BTM): establishing one bidirectional multicast tree for one multicast session.
- A new approach is proposed in order to establish a multi-source multicast session, namely one Bidirectional Tree with Just enough reserved bandwidth (BTJ). In this approach we propose an algorithm to calculate the excess reserved bandwidth in the BTM approach and release it. Based on the simulation results, the BTJ approach achieves the best overall performance.

- The information of multicast trees is stored in Information Base (IB) tables in the routers for routing and forwarding decisions. The IB table scheme is proposed for the introduced approaches of establishing a multi-source multicast session. The volume of the IB tables is evaluated and compared for different multi-source multicasting approaches.

1.4 Organization of the Thesis

The rest of this thesis is organized as follows. Chapter 2 provides an overview of IP/MPLS over WDM networks and the development path from current network's protocol stack to one with fewer layers. The concept of multicasting is then introduced. Some existing work on optical multicasting, IP multicasting, MPLS multicasting and traffic grooming are also discussed in this chapter.

Chapter 3 gives an introduction to several approaches of multi-source multicast session establishment. The characteristics of the multicast trees in different approaches are described in this chapter.

In Chapter 4 we explain the algorithms of establishing multicast trees and traffic grooming of the introduced multicasting approaches. The methods of choosing a root-node for the FRN approach and bandwidth calculation for the BTJ approach are also presented in this chapter. Simulation results are reported for

network blocking probability and the time complexity of different multi-source multicasting approaches.

In Chapter 5 the Information Base (IB) table is introduced and some examples of IB tables for different approaches of multi-source multicasting are given. The simulation results are reported for the volume of the IB tables in the control plane of different approaches.

Finally the thesis is concluded in Chapter 6 and some recommendations are offered for further researches.

Chapter 2. Literature Review

In this chapter, an overview of multicasting in IP over WDM networks is provided. First the IP/MPLS over WDM network is described as a promising future backbone network. Then the concept of multicasting is explained and related work on optical and IP multicasting are reviewed. Next the merging of MPLS and multicasting as two complementary techniques is discussed. And finally a brief review on the concept of traffic grooming is provided.

2.1 IP/MPLS over WDM Network

The legacy networks are classified into three generations based on the underlying technology in the physical layer [43],[44].

- First generation networks: “copper-based” or microwave technology is used in the physical layer. Both transmission and switching of the data are performed in the “electronic domain”.
- Second generation networks: Optical fiber is used in the physical layer and data transmission is performed in the optical domain. However switching is still done in the electronic domain. Therefore the second generation networks cannot fully exploit the huge capacity provided by the optical fiber and just use a small portion of the available bandwidth.

- Third generation networks: The capacity of the optical fiber is divided into multiple non-overlapping channels and hence a much larger fraction of the optical fiber bandwidth is utilized compared to the second generation networks. Here, still switching is performed in the electronic domain, but performance is increased by each channel operating at peak electronic speed (e.g. WDM networks) [12].

If only one data stream is transmitted through one optical fiber, the huge capacity provided by the optical fiber is not utilized. Wavelength Division Multiplexing (WDM) is the dominant technology to exploit the huge capacity provided by optical fibers and other optical components. WDM is based on using multiple sources each operating at a slightly different wavelength. Multiple wavelengths are used to transmit different data streams in a single fiber. Each channel has the bandwidth of the peak electronic data rate. The development of WDM technology has vastly increased the transmission capacity of optical systems to several Tb/s [1].

Optical cross-connects (OXC) are switching systems that are used to switch high speed optical signals in a fiber optic network. There are two ways to implement an OXC. Optical-Electrical-Optical (OEO) OXCs perform the switching process in electrical domain. The signals are converted from optical domain to electrical domain before switching and from electrical domain to optical

domain after switching. This kind of OXCs is also called opaque OXC. The second approach is to switch the signals in the optical domain. The OXC that uses this approach is usually named as transparent OXC. An opaque OXC has a limitation on the maximum bandwidth of signals. However it is easy to monitor the signals' quality in such an opaque OXC. On the other hand a transparent OXC has a significantly higher speed than an opaque OXC but it does not allow easy optical signal quality monitoring [56].

In general, an optical network consists of multiple layers, such as IP, ATM, SONET/SDH, optical and fiber layer. Each layer provides flexibility to the network by its advantages and on the other hand it may have some limitations too. The network should be utilized in order to benefit from the advantages provided by a particular layer in a specific part of the network. The multiplicity of the layers in today's network could have occurred in response to the need of being backward compatible. If the protocol layers were going to be designed from scratch today, many of the existing protocol layers (e.g., ATM.SONET/SDH) would not be there [1]. We will probably move from a complex IP/ATM/SDH/WDM multilayered network, towards a simpler MPLS-empowered-IP directly over WDM network [3].

The next generation IP over WDM network is defined as one where IP is directly supported by a WDM network. WDM reduces the cost by providing large bandwidth capacity. In other words WDM will be able to support the rich features

provided by the current SONET/SDH network. The IP layer is a revenue-generating layer which offers great convergence. In order to omit the layers in between the IP layer and the WDM layer, the IP layer should also be able to provide quality of service. This would be possible by empowering the IP layer with Multi Protocol Label Switching (MPLS) [1].

MPLS is a data-carrying mechanism that belongs to the family of packet-switched networks. It is an advanced forwarding scheme that enhances routing with packet forwarding and path controlling. Packets are classified easily at the domain entry and rerouted faster in the case of link failures. Using this mechanism, the shortest path would not always be the best choice for routing. Packets may follow the explicit constructed routes based on some traffic controlling policies [31]. MPLS can be used to carry many different kinds of traffic, including IP packets, as well as native ATM, SONET, and Ethernet frames. In the MPLS domain, Label Switching Routers (LSRs) use a short, fixed-length label inserted in the packet header between the IP packet header and the layer two's header to forward packets. An example of 32-bit label is shown in Fig. 2-1 [6].

Fig. 2-1 An example of a 32-bit label.

Each label stack entry contains four fields:

- A 20-bit label value.
- A 3-bit field for QoS priority.
- A 1-bit bottom of stack flag.
- An 8-bit TTL field.

Fig. 2-2 shows the evolutionary development for IP/MPLS over WDM networks protocol stack.

Fig. 2-2 Development for IP/MPLS over WDM solutions

A typical IP/MPLS over WDM network and node architecture is shown in Fig. 2-3 and Fig. 2-4 [58]. An OXC is connected to a Label Switching Router (LSR) through a User Network Interface (UNI). The LSR is equipped with several transceivers (transmitter/receiver). The transmitter in the LSR sends out the traffic as an optical signal on a wavelength. On the other hand the receiver in LSR converts the received traffic from an optical signal to electronic data. The paths between the ingress LSRs (at the domain entry) and egress LSRs (at the domain exit) are called Label Switched Paths (LSPs).

Fig. 2-3 Network architecture in IP/MPLS over WDM

Fig. 2-4 Node architecture in IP/MPLS over WDM

MPLS uses signaling protocols such as Resource Reservation Protocol (RSVP) and Label Distribution Protocol (LDP) to set up LSPs [31]. The LSRs provide the electrical switching for routing and traffic grooming. An LSR processes all packet traffic passing it and ensures the availability of the bandwidth for the bandwidth demands of the LSP requests on the outgoing lightpaths. Both

tunable transceivers and fixed transceivers may be used in a WDM network. In WDM networks, a lightpath is a single hop path which is established by configuring the transceivers (transmitter, receiver) and some intermediate optical cross-connects (OXC) along the path [7]. The client LSR in the IP/MPLS layer requests for establishing a lightpath in the optical layer through the UNI. A lightpath is shown in Fig. 2-5.

Fig. 2-5 A lightpath in the optical network

The architecture of IP/MPLS over optical networks is categorized into three models known as the overlay model, the augmented model and the peer model [25].

- The peer model has a single integrated control plane which controls both the optical layer and IP/MPLS layer. Both layers have information about

the status of the other layer and a common addressing scheme used for these two layers. [25]. Fig. 2-6 shows the network architecture in the peer model.

Fig. 2-6 The network architecture in the peer model

- In the overlay model, the optical networks and IP/MPLS networks have independent separate control planes. There are independent routing, topology distribution and signaling protocols for these two layers. IP/MPLS client devices are only aware of some restricted information about the optical network. IP/MPLS layer does not have knowledge about optical topology or hardware resources. Therefore separate instances of control plane for routing and signaling protocols would have to be deployed in IP/MPLS domain and the optical domain. The overlay model has several benefits compared to the peer model. First there is a limited interaction between the two control planes through the User Network

Interface (UNI), for example requesting for a connection setup. Having separate management policies for the IP/MPLS domain and the optical domain makes independency for each layer. Changes in each layer could be performed easier without affecting the other layer. For example the optical layer could be enhanced with the existing technologies without forcing change to the upper layer. Moreover the privacy of the information in each layer is ensured. In the short term, the overlay model seems to be more practical and has been experimented in laboratories and research projects [16]. In this study we consider the overlay model because of its simplicity and ease of implementation. The architecture of the network in the overlay model is shown in Fig. 2-7.

Fig. 2-7 The network architecture for overlay model.

- The peer model has a good potential to offer cost effective network solutions. The limit in this model is that LSRs and OXCs may be managed by different vendors running proprietary protocols and hence the tight coupling scenario is less practical. The other issue is that to maintain a global network status, a huge amount of control messages should be flooded across entities. The overlay model does not need to exchange any information across layers and is easy to manage. But, it may cause network under utilization. Another approach is the augmented model [59]. The augmented model provides compromise between the two extreme models by allowing the exchange of some network information between the layers. The exchanged information includes reachability and summary of link state information (e.g. residual capacity) depending on a specific agreement between the two layers [60].

There are two traffic engineering components for this two-layer network, the LSPs in the IP/MPLS layer and the lightpaths in the optical layer. Multiple policies have been introduced for LSP establishment in IP/MPLS over optical networks in [5].

2.2 Multicasting

In many applications it is required to send the same data from a source to multiple destinations simultaneously. Multicasting is the efficient technique to perform this task. There are two situations that multicast would be very important for an application: the traffic volume is sufficiently high, and there are plenty of

receivers. Multicasting prevents sending multiple copies of the same data over the same link. It significantly saves the bandwidth. Some instances for these applications would be video conferencing, online multiplayer games with high-quality graphics and web sites pushing data such as camera output or a TV show. The other use of multicast would be to find other members of some sort of services when their unicast address is not known in advance. For instance, it might be useful to find other Border Gateway Protocol (BGP) routers within your domain, or to find a Domain Name System (DNS) server [27].

The optical cross-connects with multicasting capability provide dynamic connections between an input port and multiple output ports. In an IP over optical network, the packets experience two layers of multicasting, which are optical multicasting and IP multicasting.

2.2.1 Optical Multicasting

A lightpath is an all optical single hop path which is established between two nodes in the network by configuring the transceivers in the end nodes and the intermediate optical cross-connects (OXC) [7]. By setting up a lightpath between two nodes, they become virtual neighbors although they may not be directly connected in the optical layer.

Wavelength conversion is the capability of shifting an optical signal on a wavelength to a different wavelength. This feature allows for local reconfiguration

rather than global reconfiguration therefore the recovery from link failure or node failure would be much easier. By using wavelength converters the networks become more scalable and the capacity would be utilized in a better way [4].

Typically the availability of the needed bandwidth is required for the application's duration in order to provide QoS. One approach to ensure the availability of the required bandwidth for the multicasting application is to establish light trees [12]. A light tree is a point to multipoint all-optical channel which may span multiple fiber links and enables single-hop communication between a source node and a set of destination nodes; thus, a light tree based virtual topology significantly reduces the hop distance, thereby the network delay performance would be improved for multicast traffic. If wavelength conversion is not supported by the network, all of the links in the multicast tree must be set up on the same wavelength. Since we consider dynamic randomly generated traffic, it may be very difficult to find an available wavelength to set up all of the multicast tree's links in that. However if at least some nodes in the network are capable of doing wavelength conversion, different parts of the multicast tree could be established on different wavelengths [12].

In order to construct a light tree, the branching switch nodes must be able to split the incoming wavelength channel to multiple copies and switch each copy to a different output wavelength channel. This capability is called light splitting functionality, or optical layer multicasting capability [26]. A Wavelength Routing Switch (WRS) can have both wavelength converters and light splitters. If a light

splitting capable (multicast capable) WRS has no wavelength converter, the output wavelength channels that are split from the same incoming wavelength channel should occupy the same wavelength; otherwise, they can use different wavelengths from the incoming wavelength. Wavelength converters and light splitters are expensive components. Sparse wavelength conversion and sparse light splitting is the approach to reduce the cost which is to equip only a part of the switch nodes with wavelength conversion and/or light splitting functionality. Additionally, light splitters have limited splitting factor, which is defined as the maximum splitting fan-outs of light splitting at the given switch node.

One approach is to establish one aggregated multicast tree for multiple multicast groups [26]. Since all of these multicast groups may not have the same set of nodes, some of the leaves in this tree may belong to one or some of the multicast groups which this tree supports but not all of them. These nodes may receive the multicast data that does not belong to their multicast group. In other words by using an aggregated tree the data leaks from the source to some unrelated nodes which causes to the bandwidth wastage. This method could be efficient when there are multicast groups that have the exact same or very close set of members. Especially, when there is no wavelength conversion capability in the network, this approach would help to occupy fewer number of wavelengths [28].

In [13], authors proposed to construct multiple multicast light trees for one multicast group when there is no wavelength conversion capability. This is due to the fact that the requested network resources may be occupied by earlier requests,

therefore it is impossible to establish one light tree for the whole multicast group using a single wavelength. In this case multiple multicast trees with different wavelengths could be established. Using this method will cause a lower blocking probability because it is more flexible than just establishing one tree using a single wavelength. However, it increases the overall traffic load because there is more duplication of data in the network. By having multiple trees for one multicast session higher amount of information should be stored for routing and packet forwarding policies.

The quality of optical signal at each destination node is seriously affected by multicast routing scenario. In order to ensure that different destinations in a multicast group can receive alike optical signal quality, it is desirable to deliver the multicast traffic over a loss-balanced light-forest. A loss-balanced light-forest is composed of a set of light-trees with each light-tree having both bounded destination size and bounded source-destination distance. It means that the partial multicast tree expansion process would be stopped when it reaches a threshold for source-destination distance and the number of the destination nodes it includes and another tree is established for the remaining nodes of the multicast group [26].

It is desired to establish one multicast tree for a multicast group that connects the multicast member nodes together and provides the required bandwidth for the application. Establishing one multicast tree per multicast group eliminates the leaking problem that was mentioned earlier in the approach of establishing a common multicast tree for multiple multicast groups. It also minimizes the

unnecessary data replication problem in the method of establishing multiple multicast trees for one multicast group. Two well-known types of multicast trees are shared tree and source tree [3]. Shared tree is a shortest path tree that is rooted at a router called Rendezvous Point and the multicast member nodes are destination nodes in this tree. Source tree is a shortest path tree rooted at one of the multicast group member nodes and connects it to other members of that multicast group. By using shared tree, multiple nodes within a multicast group are able to send multicast data to the group. But in source tree approach for each source one separate tree must be established. The shared tree scheme has the benefit of storing less state information in the routers, but congestion might occur at Rendezvous Point when the multicast data transmission rate is high. Therefore it is applied to the networks which have more multicast sources transmitting data at a lower rate. Source tree balances the network traffic and has a shorter end-to-end delay. Therefore it is suitable for real time applications which have heavy traffic and are affected by the end-to-end delay.

2.2.2 IP Multicasting

Deering proposed an IP multicast model in 1988 to provide the multicast service over the Internet [29]. In this model, neither the senders nor the receivers need to know the location of each other, and the nodes could join or leave a multicast session dynamically. Multicast routing protocols are responsible to keep track of the membership information of a multicast group, and to establish multicast trees. Multicast routing protocol is the main component of this model

[30]. As in RFC 3171, IP addresses from 224.0.0.0 to 239.255.255.255 are dedicated for multicasting. It is also called class D [57].

The original IP multicast model (RFC 1112) [38] is an extension of the Ethernet multicast model. A globally unique address G must be obtained in order to create a group. Anyone in the Internet can ask to receive any packet which is sent to group address G , and anyone in the Internet can send a packet to the group address G . Internet is responsible for delivering the multicast packets efficiently.

Two main categories of multicasting are dense multicasting and sparse multicasting [27]. Dense multicasting is when many of the network nodes are interested in receiving the multicast data. The technique to find the multicast tree for this type is *flood and prune* technique. These protocols are known as *reverse-path multicast* algorithms. First a sender starts sending and the traffic is flooded out through the network. If a router receives the traffic on more than one interface, it would choose one and rejects the others. And then it would send a copy of the received packet on all its interfaces excluding the received one. Every link in the network would be traversed once and every node in the network would receive the data. Now if some nodes are not interested in receiving the multicast data, they should send a prune message. The tree would be pruned back to the minimal tree that connects the nodes that are interested in receiving the multicast data. Obviously flood and prune technique has scaling problems. It is not practical to flood traffic for all groups everywhere knowing that some nodes with a large distance to the source might be interested in this data, and also it is infeasible to

keep track of all the group addresses that each node does not want to receive the data from them. Some examples of IP multicasting protocols for dense multicasting are Distance Vector Multicast Routing Protocol (DVMRP) [50], Protocol Independent Multicast-Dense Mode (PIM-DM) [51], Multicast Source Discovery Protocol (MSDP) [42] and Multicast Open Shortest Path First (MOSPF) [39]. Dense-mode protocols require the maintenance of a large amount of state information about all of the routers in the network regardless of the existence of receivers.

The other multicasting category would be sparse mode multicasting in which the receivers of the multicast data are sparsely distributed within the network. In this type receivers are supposed to join or leave a multicast group explicitly by sending “join” or “leave” messages. Two instances of this approach are Core Based Trees (CBT) [40] and Protocol Independent Multicast-Sparse Mode (PIM-SM) [41]. In CBT there is a root for the bidirectional multicast tree and the receivers have to explicitly send a “join” message towards the root of the tree. PIM-SM creates a unidirectional tree rooted at Rendezvous Point.

The control and management in traditional multicasting framework suffers from the problems of efficiency, scalability, and incremental deployment [33],[34].

- Problem in efficiency: In traditional IP multicast, the routers have to forward the data and also perform the control operations. The multicast

capable router must setup and maintain the tree and this consumes a lot of control overhead messages.

- Problem in scalability: All multicast capable routers are required to maintain forwarding states for all multicast group member nodes. The routing and forwarding mechanisms in IP routing protocols are too complicated, therefore all multicast routers have to be involved in multicast branching, flow control and load balancing.
- Problem in deployment: The multicast service should be provided without frequent changes in the configurations at all routers for dynamic joining and leaving of the members. The membership information which is exchanged between the multicast domains need to be collected and updated for deploying the wide-area and sparse-mode multicast networks.

Generally IP routing algorithms search for the shortest path without paying attention to the overall network utilization. This might cause to overloading the shorter links while keeping the longer ones underutilized. By adding MPLS to the IP, the routing algorithms could be enhanced by traffic engineering policies and QoS provisioning. In the next part we will review some related works about MPLS based multicasting.

2.2.3 MPLS Multicasting

Multicast and MPLS are two complementary technologies. Merging these two technologies and building multicast trees on top of MPLS networks enhances the network performance and offers an efficient solution for the problems of multicast scalability and control overhead. Multicast reduces the network's bandwidth usage, while MPLS attempts to provide users with the required bandwidth. Multicast reduces the load of servers and network equipment but it suffers from scalability problem. On the other hand MPLS provides flexibility, speed and scalability [31].

MPLS supports both shared tree and source tree [3]. In [14] authors propose an extension to the existing multicasting protocols in order to add the MPLS benefits to them. They use one node called Rendezvous Point (RP) in the network and make all the senders to first unicast their information to the RP. They measure the volume of label tables in the nodes and the RP and conclude that the memory consumption of control messages and the labels are comparable to that of the IP multicasting. The proposed structure is for establishing the unidirectional multicast shared trees.

It is proposed in [32] to switch from source tree to shared tree and vice versa when it is appropriate. The parameters R-max and R-min stand for the maximum and minimum threshold rates of RP. Whenever the arriving rate at RP exceeds R-max, some multicast sources switch to source trees in order to avoid congestion at

RP and if it is under R-min, RP triggers the switching back operation. There are S-min and S-max parameters, standing for the minimum and maximum threshold of multicast source. Whenever the multicast source's rate exceeds S-max, multicast tree will be switched to source tree and it can switch back to shared tree when the rate becomes under S-min.

The information about the multicast tree is kept in tables in the routers for routing and control decisions. There is one entry in the Multicast Information Database (MIDB) for each multicast group. For shared trees this MIDB is stored in the Rendezvous Points (RP) and the roots of the multicast trees. RP is in charge of forwarding the received data from a source to the related multicast group. On the other hand each router should keep one entry per outgoing interface for each multicast group in the Label Information Base (LIB) [36].

A main reason of the multicasting scalability problem is the fast growth in the size of multicast forwarding table. In multicast routing protocols, each multicast router has to maintain a multicast forwarding table entry for every group whose multicast tree passes through the router. When the number of groups in the network increases, the related multicast forwarding table will also be very large, which causes high router cost and low forwarding performance [30].

A solution is proposed in [30] in order to reduce the size of multicast forwarding tables for sparse multicasting groups. When a multicast group is sparse, multiple nodes in the multicast tree might have only one immediate downstream

and therefore there is no replication of data at them. The data could be sent through unicast channels in this case. After dynamic tunnels are established, only the root node, branching nodes and leaves of the multicast tree need to maintain the state information. The nodes that have one immediate downstream and do not have to know about the group since the multicast packets are forwarded in a unicast mode between the end points of the tunnel.

In [37] a scalable MPLS sparse-mode multicast algorithm is designed which merges the tunneling technology and the branching nodes technology. Using tunneling technology to implement M2M shared tree multicast can solve the M2M label distribution problem, and branching nodes technology reduces multicast forwarding state in routers and enhances multicast scalability. As discussed earlier, shared trees consume much fewer labels than source trees and achieve better scalability.

2.3 Traffic Grooming

There is a huge gap between the bandwidth requirements of the LSPs and the capacity provided in wavelength channels. Traffic grooming is to collect low speed traffic flows of different applications and pass them through a common link in order to utilize the huge capacity of the links [8].

For the mesh WDM network, most of the current literature has focused on traffic grooming problem when the node-to-node traffic demands do not vary

significantly over time and could be well represented by a static matrix of traffic demands [47]. In this case, the virtual topology design problem can be formulated as an optimization problem, using principles from physical routing of lightpaths and traffic flow on the virtual topology. In general, the optimal virtual topology problem has been conjectured to be NP-hard [9], which means that the problem cannot be solved optimally for large problem sizes, unless one resorts to some form of exhaustive search.

In [20] the problem of traffic grooming is divided into four sub problems which are not essentially independent. These four sub problems can be summarized as follows:

- Make a virtual topology and determine the number of lightpaths between any node pair.
- Route the lightpaths over the physical topology.
- Assign wavelengths to the lightpaths in an optimal way.
- Route the low-speed requests on the virtual topology in a way which exploit the resources.

In [61], the authors consider routing traffic demands and routing and assigning wavelengths to lightpaths, as a combined optimization problem and present a general linear formulation for that. For small networks linear formulation can be solved in a reasonable time. The authors relaxed the integer constraints in order to solve the linear formulation for large networks.

The virtual topology in a WDM mesh network would be a set of lightpaths established to provide all-optical connectivity between nodes for a given traffic demand. Traffic grooming in a WDM mesh network is very close to the concept of the virtual topology design and reconfiguration. The scalability and economic issues make it impossible to establish a lightpath for every node pair. A portion of the traffic may need to be switched electronically from one lightpath to another at intermediate nodes until it reaches its destination. The design of the virtual topology is the problem of optimizing the use of network's resources and network's throughput, for a given traffic demand [8],[9]. When the traffic changes dynamically, a virtual topology which is optimized for a given traffic demand may not be optimal for a different traffic demand. This implies that the virtual topology should also be changed to match the current traffic patterns. The dynamic structure of the optical cross-connects, i.e., the ability of switching wavelength channels from any input fiber to any output fiber dynamically, allows this modification in the optical layer. This is known as reconfiguration of the virtual topology or *dynamic traffic grooming* [47].

In [62] a novel graph model is introduced for the peer model. The edges in this graph represent a certain equipment or functionality at a node. Different objective functions under different grooming policies are represented by weighting the edges based on the physical constraints. This model can be used for both static and dynamic traffic grooming.

Dynamic traffic grooming in the IP/MPLS over WDM network is a two-layer routing problem regarding the constraints in the IP/MPLS and the optical layers and the relationship between them. In the IP/MPLS layer, MPLS traffic engineering tries to exploit network resources and to minimize congestion by trying to distribute the traffic evenly [63]. In the optical layer, wavelength routing needs to consider the wavelength conversion capability in the network and availability of the add/drop ports [64].

The peer model can facilitate the network optimization and the overlay model is more practical and offers ease of implementation. In [5], the *IP-Layer-First* (ILF) and *Optical-Layer-First* (OLF) traffic grooming policies are proposed under the overlay model for the establishment of LSPs. With the ILF policy, networks first try to route a new LSP request in the IP/MPLS layer. When there is no enough available bandwidth in the IP/MPLS layer, the LSP request is transferred to the optical layer for setting up a new direct lightpath between the source and destination. In contrast, with the OLF policy, the network first attempts to set up a lightpath in the optical layer for a new LSP request.

In [65] the *One-Hop-First* (OHF) traffic grooming policy is proposed. In this policy a lightpath is set up whenever a feasible one-electrical-hop path cannot be found for a new LSP request in the existing IP topology. The OLF and OHF policies cause less blocking probability than the ILF policy, as they try to set up new lightpaths before blocking happens in the IP layer. In contrast, when the ILF

policy tries to set up a new lightpath, the IP layer is already congested with a lot of traffic flowing on long routes.

2.4 Summary

The IP over WDM network is introduced in this chapter and the evolution from the traditional multilayered network to the next-generation two-layer IP/MPLS over WDM network is described. Multicasting is introduced as a technique that makes it possible to send only one copy of data on the same link when there are multiple destination nodes for the same packet. The existing research work in the areas of optical multicasting, IP multicasting and MPLS multicasting are discussed. Finally a brief review on traffic grooming in IP/MPLS over WDM networks is reported.

So far most research work considers single source multicasting. In this study our focus is on multi-source multicasting in which there may be multiple sources each of which may be able to send data to the rest of the members of the multicast group simultaneously. Multicasting applications such as video conferencing and online multiplayer interactive games demand huge bandwidth, therefore one of the main issues is how to allocate the network resources to the multicast groups so that the overall network throughput would be enhanced. As will be discussed in the subsequent chapters, we aim to find a strategy to establish multi-source multicast sessions with reasonable blocking probability. The information about the multicast trees is stored in Information Base (IB) tables in the routers for routing and control

decisions. The size of the IB tables grows fast with the increase in the number of multicast groups in the network and size of each multicast group. This issue also should be studied in order to find an efficient multi-source multicasting approach.

Chapter 3. Approaches of Multi-Source Multicasting

3.1 Introduction

As discussed earlier multicasting is the technique to send the same data from the source to some selected destination nodes simultaneously. Using multicasting technique, the source node would generate only one copy of the data and the data packet will be replicated wherever the path from source to destinations is split. Therefore the nodes at the branches must be able to replicate the received data and send it out on different branches.

As described in [48], multicasting applications are divided into two main categories: One-to-Many multicasting and Many-to-Many multicasting.

1. *One-to-Many* (O2M) multicasting: This type of multicasting is most widely studied. In this case, a multicast group has a single source node and all the other nodes of the multicast group are destination nodes and receive the data simultaneously. Some examples of this type are:

- Scheduled audio/video (a/v) distribution: Lectures, presentations, meetings.
- Push media: News headlines, weather updates, sports scores.
- File distribution and caching: Web site content, executable binaries, and other file based updates sent to distributed end user or replication/caching sites.
- Announcements: Network time, multicast session schedules, random numbers, keys, configuration updates.

- Monitoring: Stock prices, sensor equipment, security systems, manufacturing or other types of real time information.

2. *Many-to-Many* (M2M) multicasting: This type of multicasting has rarely been studied. In this case, there is more than one sender in a multicast group. Some member nodes in the multicast group may receive data from multiple senders simultaneously while they also send data to the multicast group at the same time. As a result, M2M applications often present complex coordination and management challenges. Some examples for this type are as follows:

- Multimedia conferencing.
- Synchronized resources: Shared distributed databases of any type like schedules and directories.
- Concurrent processing: Distributed parallel processing.
- Collaboration: Shared document editing.
- Distance Learning.
- Chat Groups.
- On-line multi-player games.

In this study we focus on M2M multicasting applications. Nodes that are either source or destination within a multicast group are called *multicast member nodes*. A multicast tree should connect the multicast member nodes and there may be other nodes included in the tree in order to make a connected tree, while they are not a member of the multicast group. It is assumed that once a multicast session is established, all of the multicast member nodes can generate data and

send it to the group and also receive the multicast data generated by other members at the same time. In this study all of the multicast group member nodes can generate the data and send it to the group simultaneously, therefore all of the multicast group members are given equal priority. In this chapter we describe several approaches for implementing M2M multicasting in IP/MPLS over WDM networks. In Chapter 4 we examine and compare their performances in terms of blocking probability and time complexity. In Chapter 5 memory consumption in the control plane is evaluated for the introduced multi-source multicasting approaches.

Every node in the network that wants to start a multicast session must send a multicast request which specifies the set of multicast member nodes and the amount of the bandwidth needed for this multicast connection. Upon a multicast request's arrival, a multicast tree must be established in order to connect the multicast group member nodes together. QoS is required for these applications and it must be guaranteed that there is sufficient available bandwidth for the member nodes to send their data to the group, any time during the application's running time. Therefore the requested bandwidth is reserved on every link of the multicast tree. In order to reserve the needed bandwidth, a Label Switch Path (LSP) is established for each link of this multicast tree which guarantees the availability of the requested bandwidth.

Assume that N is the number of the nodes in the network or in the other words the network size. We use $M_i(D_i, b_i)$ to denote the i^{th} multicast request in the

network, where D_i is the set of the member nodes and b_i is the bandwidth required to transmit the data generated by any one of the member nodes. We denote n_i as the number of member nodes in the i^{th} multicast request, which is also referred to as the size of the i^{th} multicast group.

3.2 Multiple Unidirectional Trees Approach (MUT)

A straight forward approach for setting up an M2M multicast session is to establish multiple unidirectional multicast trees, each rooted at one of the member nodes. For the i^{th} multicast request, the reserved bandwidth in each link of these unidirectional multicast trees is equal to b_i where each tree is unidirectional and carries the data generated by one node. This method repeats the multicast tree establishment procedure n_i times for the i^{th} multicast request and the control plane must keep information of all the n_i unidirectional multicast trees needed for the multicast session. Apart from its implementation complexity, a large volume of control messages would also need to be exchanged between the nodes. Later on in this study we will refer to this approach as Multiple Unidirectional Trees approach (MUT).

Fig. 3-1 MUT: Multiple Unidirectional Trees for one multicast session

Fig. 3-1 shows a multicast group with three nodes A, B and C ($n=3$). The network size is 8 in this example ($N=8$). There are three separate unidirectional trees established each rooted at one of the multicast member nodes. Since each tree carries the data generated in one node, bandwidth of b_i is reserved on each link.

3.3 Multi Root-Node Approach

In the MUT approach the number of unidirectional trees to be set up increases with the multicast group size. That is, the complexity of MUT increases with the multicast group size. In order to tackle the complexity problem of the MUT approach, we here consider a multi root-node approach, whereby some specific nodes in the network are selected as *root-nodes*. In general, root-nodes should have a high node degree since every multicast session would include one of the root-nodes and the links adjacent to these would have high probability of being

used. These root-nodes must be distributed as widely as possible in the network in order to balance the traffic load. This multi root-node approach makes multi-source multicasting effectively as simple as the fixed single source multicasting. For the i^{th} multicast request, a unidirectional tree including one root-node and the n_i member nodes would be established rooted at the selected root-node. One advantage of this approach compared to the MUT approach would be lower memory consumption for keeping track of the session information. In addition multiple trees protection problem now becomes single tree protection. And also when a node wants to join to or leave the multicast session, if there are multiple multicast trees for a single multicast session, the processes of leaving and joining would be multiplied by the number of the multicast trees that belong to that session. For one multicast tree, there is only one joining or leaving operation per node.

LSPs are unidirectional [49]. If we want to have a tree with bidirectional links, two LSPs in opposite directions must be established on each link of the tree. And it must be checked that the link has sufficient bandwidth in both directions. Since the multicast tree is unidirectional in this approach, only one LSP is established in one direction on each link of the multicast tree.

The selected root-node would be the single source and all of the member nodes would be the destination nodes in this unidirectional tree. In this approach, every node of the multicast group which wants to send data to the multicast group must send it to the selected root-node through unicasting. Today, IP multicast uses

a similar approach to establish multicast sessions. Two IP protocols which use the similar approach are Protocol Independent Multicast-Sparse Mode (PIM-SM) [41] and Core Based Trees (CBT) [40].

We want to establish multicast sessions in which all of the multicast member nodes are able to send data to the group simultaneously. The reserved bandwidth on the links of the multicast tree must be sufficient to transmit all the traffic even in the worst case which happens when all member nodes of the multicast session send multicast data simultaneously. In the worst case the selected root-node receives the data from all of the multicast member nodes at the same time and needs to multicast them all together. In order to establish a tree which is able to transmit all the traffic in the worst case, the amount of nb_i bandwidth needs to be reserved on each link of the tree. Since the multicast tree is unidirectional, one LSP in one direction is established on each link of the multicast tree. Bandwidth guaranteed unicast paths are also required to transmit the unicast traffic from each member node to the root-node. To ensure that sufficient bandwidth is always available for unicast data from multicast member nodes to the root-node, the bandwidth of b_i is reserved on each unicast path from the member nodes to the root-node.

3.3.1 Fixed Root-Node Approach

In this approach, a few specific nodes in the network are pre-selected as the root-nodes. Upon the arrival of a multicast request, a most appropriate root-node will be chosen from the pre-defined root-nodes, and a unidirectional tree rooted at

the root-node should be established to connect the multicast group member nodes. In order to transmit the data generated in multicast member nodes to the root-node, n_i unicasting paths must be established with sufficient bandwidth reserved on them.

This Fixed Root-Node approach (FRN) is suitable when a network is divided into several separated domains. In this situation if a multicast group is spread over multiple domains, multiple local trees could be established rooted at the root-node of each domain and then these root-nodes could communicate with an inter domain communication protocol. The other application of this approach is when not all of network nodes have equal capabilities, either hardware abilities or the protocols they support. When only a few nodes have the potential to be the root of the unidirectional multicast trees, we must select one of them for each multicast request (session).

There are three important issues for this approach.

- *Number of the root-nodes:*

If one of the fixed root-nodes must be chosen for every multicast request, based on the frequency and bandwidth demand of the multicast requests and also the network size, there must be enough number of root-nodes to satisfy the needs of the network. The network behavior and traffic history should be studied in depth in order to decide how many root-nodes are needed for a given network.

- *Distribution of the root-nodes:*

The location of the root-nodes is highly dependent on the distribution of the multicast requests and the traffic density of the applications in different parts of the network. The root-nodes must be widely distributed in order to satisfy the needs of incoming multicast requests. If there is not enough number of root-nodes in one part of the network where many multicast groups are gathered, then they have to choose a far root-node to establish their trees and this would lead to unnecessary resource consumption and higher blocking probability.

- *Choosing a root-node for a multicast group:*

The root-node must be selected efficiently for each multicast group. The objective is to minimize the size of the multicast trees in order to consume less network resources and achieve a lower blocking probability.

Fig. 3-2 represents a multicast session in the FRN approach.

Fig. 3-2 FRN: Fixed Root-Nodes

In Fig. 3-2 there are three fixed root-nodes in the presented network which are RN, D and F. RN has been chosen to be the root-node for the current multicast request with three nodes A, B and C. A unidirectional multicast tree is established rooted at the RN to connect the multicast member nodes together. The bandwidth of $3 \times b_i$ is reserved on each link of this unidirectional tree. Three unicast paths are also established from the member nodes to the root-node. The bandwidth of b_i is reserved on each unicast path to transmit the data generated by each node of the multicast group.

3.3.2 Random Selection of the Root-Nodes (RRN)

For a network where all nodes have equal capabilities and support the same protocols and the size of the network is no longer a problem, there is no need of pre-defining some specific nodes as root-nodes. Every node of the network can be selected as the root-node for multicast trees. In such network, we may randomly choose one of the member nodes of a multicast request as the root-node. By doing so, there is no need to include a non-member node as a root-node and hence the multicast tree to be established would be smaller. This is expected to enhance the efficiency of the network. For each coming multicast request a root-node is chosen among the multicast group nodes and a unidirectional multicast tree rooted at the root-node is established for this multicast group.

Fig. 3-3 RRN: Random selection of Root-Nodes

Fig. 3-3 shows a multicast group with three nodes A, B and C ($n=3$). In this case the root-node is chosen among the multicast member nodes. Here node A has been selected to be the root-node for this multicast group. A unidirectional tree is established rooted at node A to connect the multicast member nodes. The bandwidth of $3 \times b_i$ is reserved on each link of the multicast tree. Two unicast paths are established from B and C to the root-node A. The bandwidth of b_i is reserved on the unicast paths.

3.4 One Bidirectional Tree Approach

Another way to tackle the excessive complexity of MUT approach is to establish one bidirectional tree instead of multiple unidirectional trees for each multicast request. Sufficient bandwidth should be reserved in both directions on every link of this tree so that each node can send the data on outgoing LSPs and receive data from incoming LSPs. As we mentioned earlier, LSPs are

unidirectional [49]. Therefore two separate LSPs should be established in opposite directions on each link. Each LSP must have sufficient bandwidth to transmit the multicast traffic at any time. The required bandwidth in each direction of each link is different because the number of the multicast group member nodes on each side of a link is different. For the i^{th} multicast request, the required bandwidth in one direction of a link of the tree varies in the range of b_i to $(n_i-1)b_i$. Because the minimum number of the multicast group member nodes at one side of a multicast tree link is one and the maximum number of the multicast group member nodes would be (n_i-1) . To simplify the bandwidth reservation, a simple scheme is to reserve the maximum required bandwidth $(n_i-1)b_i$ in each direction of each link of the tree. We refer to this scheme as one Bidirectional Tree with Maximum reserved bandwidth (BTM). Fig. 3-4 illustrates an example of this approach.

Fig. 3-4 BTM: Bidirectional multicast Tree with Maximum reserved bandwidth

In this example we have a network with 8 nodes ($N=8$). A bidirectional tree is established for the multicast group (A, B and C). The bandwidth of $2 \times b_i$ is reserved in each direction of the links of this tree.

In this approach there is only one multicast tree for each multicast session. The information of only one tree should be kept in the control memory. BTM with one bidirectional multicast tree and no unicast path is expected to have lower overheads than MUT, FRN and RRN approaches. However, the bandwidth reserved in this approach is much larger than the actual required bandwidth, which would lead to a higher blocking probability. In the next section, a more efficient approach is proposed.

3.5 Bidirectional Tree with Just Enough Reserved Bandwidth (BTJ)

In the BTM scheme, the bandwidth is over-reserved in each direction of each link of a bidirectional tree. To deal with this problem, we here propose an efficient scheme whereby only just enough bandwidth is reserved in each direction of each link of a bidirectional tree. We refer to this scheme as one Bidirectional Tree with Just enough reserved bandwidth (BTJ). We next analyze the bandwidth required to be reserved in each direction of a link of a bidirectional tree.

Each link of the i^{th} bidirectional tree separates the tree into two sub trees having m and $n_i - m$ of multicast member nodes respectively where ($1 \leq m < n_i$). The worst case is when all the member nodes need to send their data simultaneously,

which would require the reserved bandwidth to be mb_i in one direction and $(n_i-m)b_i$ in the other direction on each link of the tree. Hence, a total bandwidth of n_ib_i would be needed on each link. Fig. 3-5 shows the multicast tree for a multicast session with 6 nodes. Link AB connects two sub graphs SA and SB together. As can be seen, there are 2 nodes in SA and 4 nodes in SB. If all of these 6 nodes be the multicast members and send data simultaneously, on the link AB, there is a bandwidth of $2 \times b_i$ needed from A to B and a bandwidth of $4 \times b_i$ from B to A. The total bandwidth of $6 \times b_i$ would be required on this link.

Fig. 3-5 An example showing the traffic flows on a link of a bidirectional multicast tree.

Since the value of m varies from 1 to $n_i - 1$ on each link of the i^{th} multicast tree, the amount of the bandwidth that should be reserved in each direction of each link is different from the other links of the same multicast tree. The important point is that two LSPs on each link consume a total bandwidth of n_ib_i . This amount is much less than $2 \times (n_i - 1) \times b_i$, which is the total reserved bandwidth for two LSPs on each link of the i^{th} multicast tree in the BTM scheme. Therefore, it is expected

that BTJ is much more efficient than BTM. However the former is more complicated in reserving the bandwidth on each link. The process of calculating the exact amount of required bandwidth in each direction of each link of the multicast tree is described in Chapter 4. Fig. 3-6 shows an example of this approach (BTJ).

Fig. 3-6 BTJ : Bidirectional Tree with Just enough reserved bandwidth

In this example we have a network with 8 nodes ($N=8$). A bidirectional tree is established for the multicast group (A, B and C). On each link of this tree two LSPs are established. The total reserved bandwidth on each link of the tree is $3 \times b_i$.

The overhead of the BTJ approach in the control plane would be less than that of other approaches including MUT, FRN and RRN, since there is only one tree to be maintained for each multicast group. The reserved bandwidth for each multicast tree is minimized in this approach therefore the blocking probability is expected to be less than that of the BTM approach.

3.6 Summary

This chapter introduces several approaches for establishing a multi-source multicast session. In this study, it is assumed that once a multicast session is established, all of the multicast member nodes are able to generate multicast data and send to the group simultaneously while they are able to receive the multicast data generated by other member nodes of the same group at the same time. The first proposed approach is named Multiple Unidirectional Trees approach (MUT). In this method, for each multicast session multiple unidirectional multicast trees must be established, each of which is rooted at one of the multicast member nodes. The second proposed approach is the multi root-node approach. In this method we have only one unidirectional multicast tree for each multicast session. The root of this tree is called the root-node which could be one of the fixed pre-defined root-nodes in the network (FRN) or be selected randomly among the multicast member nodes (RRN) based on the application. The last proposed approach is to have one bidirectional multicast tree for a multicast session. Two different methods are discussed for this approach. They are the Bidirectional Tree with Maximum reserved bandwidth (BTM) and the Bidirectional Tree with Just enough reserved bandwidth (BTJ).

Chapter 4. Evaluation of Blocking Probability and Time Complexity

In Chapter 3 we introduced and described several approaches to establish multi-source multicast sessions. In this chapter, firstly the network model and the traffic model that are used in our performance evaluation are described. We then explain the methods and algorithms that are used for the simulation experiments. Finally the simulation results for different approaches of multi-source multicasting from the aspects of the blocking probability and the time complexity are reported.

4.1 Network Model

In this section we describe the network model used in this study. As shown in Fig. 4-1, we consider four topologies for a given network [9],[47],[53]. As discussed in Section 2.1, each network node consists of an OXC and an LSR, which are interconnected through a User Network Interface (UNI) as shown in Fig. 2-3. The optical layer of a network consists of the optical fiber links and the OXCs. This is referred to as the physical topology of the network. The dynamic topology of a network consists of the OXCs and those fiber links which have at least one free wavelength. The virtual topology consists of the LSRs and the existing lightpaths. The feasible virtual topology is comprised of the LSRs and those existing lightpaths whose available bandwidth is at least as much as the bandwidth

needed for establishing a multicast tree for a new multicast request. Upon the arrival of a new multicast request, the network attempts to establish a new multicast tree based on the feasible virtual topology. Fig. 4-1 shows the aforementioned four topologies.

Fig. 4-1 Four network topologies for a given network

It is noted that the links in the feasible virtual topology are a subset of the links in the virtual topology and the links in the dynamic topology are a subset of the links in the physical topology.

Let $G(V, E_p)$ represent the physical topology where V is the set of the N nodes in the network and E_p is the set of the optical links in the physical layer. The electrical switching/routing needed for multicast routing and traffic grooming is provided by the LSRs. Without loss of generality, we assume that each optical link

consists of two optical fibers transmitting in opposite directions. All the optical links are considered to be of unit length, each of which supports W wavelengths in each direction. Each wavelength has a capacity (or bandwidth) of C . We also assume that all of the OXCs are capable of wavelength conversion. Therefore, as explained in Chapter 2, a new lightpath could be established with a different wavelength on a different link.

Let $A = [a_{ij}]$ be an $N \times N$ matrix, representing the dynamic topology of the network, where the elements of the matrix are as follows:

$$a_{ij} = \begin{cases} 1 & \text{for } 1 \leq w_{ij} \leq W, i \neq j \\ \infty & \text{otherwise} \end{cases} \quad (4.1)$$

where N is the size of the network, W is the number of wavelengths in each optical fiber and w^{ij} is the number of free wavelengths in the optical fiber from node i to node j . In order to set up a new lightpath, Dijkstra's shortest path algorithm is applied to this matrix to find the shortest path between two nodes.

Let $F = [f_{ij}]$ be a $N \times N$ matrix, representing the feasible virtual topology where the elements of the matrix are as follows

$$f_{ij} = \begin{cases} \min_{1 \leq q \leq k} H_q^{i,j} & k > 0 \\ \infty & k = 0 \end{cases} \quad (4.2)$$

where k is the number of feasible lightpaths from node i to node j whose available bandwidth is at least as much as b_i at the current time and $H_q^{i,j}$ is the length of the q^{th} lightpath between i and j (i.e, number of optical fiber hops it passes through). Among the q feasible lightpaths from node i to node j , the one with the shortest length is selected. In order to find the shortest path in the virtual topology we apply the Dijkstra's shortest path algorithm to the feasible virtual topology.

4.2 Traffic Models

We use $M_i(D_i, b_i)$ to denote the i^{th} multicast request in the network, where D_i is the set of the member nodes and b_i is the bandwidth required to transmit the data generated by any one of the member nodes. We denote n_i to be the number of member nodes in the i^{th} multicast request, which is also referred to as the size of the i^{th} multicast group. A node that belongs to D_i is known as the member node of the i^{th} multicast request (M_i). As mentioned earlier, the capacity of each wavelength and therefore the capacity of each established lightpath is C . The required bandwidth b_i , which is the bandwidth needed to transmit the traffic generated by one of the multicast group member nodes is a random variable in $[x_1C, x_2C]$ following a uniform distribution, where $0 < x_1 \leq x_2 < 1$.

We assume that every node in the network is able to generate multicast requests. Multicast requests arrive to the network following a Poisson process with an average arrival rate λ and they have an exponentially distributed holding time

with a mean value μ . A multicast request involves at least three nodes in the network. The member nodes of a multicast request are selected randomly among all nodes in the network. We assume that the size of a multicast group (x) follows a truncated geometric distribution [54]. The probability of multicast request with a group size k is given by

$$P(x = k) = \frac{(1-q)q^{k-2}}{q - q^{N-1}} \quad 3 \leq k \leq N \quad (4.3)$$

The average size of the multicast group, $\bar{n}$, can be calculated by

$$E(x) = \frac{3q - 2q^2 - (N+1)q^{N-1} + Nq^N}{(1-q)(q - q^{N-1})} \quad (4.4)$$

where N is the number of nodes in the network and q is a parameter that determines the average size of multicast groups. The proof is given in Appendix I.

Let R_m be the ratio of the number of multicast requests to the total number of unicast and multicast requests generated in the network. Every member node in a multi-source multicast session may send data simultaneously to the other member nodes of the multicast session. A unicast session involves two nodes only and each node can send data to the other node simultaneously. Therefore the effective average arrival rate is $\bar{n} R_m \lambda$ for the multicast requests, and $2(1-R_m)\lambda$ for the unicast requests. The effective unicast traffic load ρ_U , the effective multicast traffic

load ρ_M and the effective overall traffic load ρ , normalized to the capacity of the wavelength channel C , will then be given by

$$\rho_U = \frac{(x_1 + x_2)}{2} (1 - R_m) 2\lambda\mu \quad (4.5)$$

$$\rho_M = \frac{(x_1 + x_2)}{2} (n - 1) \bar{n} R_m \lambda\mu \quad (4.6)$$

$$\rho = \rho_U + \rho_M = \frac{(x_1 + x_2)}{2} \{2(1 - R_m) + \bar{n}(\bar{n} - 1)R_m\} \lambda\mu \quad (4.7)$$

In this study, once a multicast session is established, all of the multicast member nodes are able to generate traffic and send to the multicast group members. We must reserve sufficient bandwidth on the tree links so even if all of the multicast member nodes generate data simultaneously there would be enough bandwidth to transmit the traffic. The network may not be able to connect all of the member nodes of a multicast request. Those nodes of a multicast request that cannot be connected to a multicast tree will be blocked. We assume that the network allows to block some member nodes of a multicast request, while establishing a partial tree connecting the rest of the member nodes. We use the blocking probability as a parameter to evaluate the performance of the multi-source multicasting approaches described in Chapter 3. The blocking probability is defined by

$$P_b = \frac{\sum_i (\text{blocked bandwidth for } i^{\text{th}} \text{ multicast request})}{\sum_i (\text{total required bandwidth for } i^{\text{th}} \text{ multicast request})} \quad (4.8)$$

If only one multicast tree is set up for a multi-source multicast session, the blocking probability can be calculated by

$$P_b = \frac{\sum_i n'_i (n_i b_i)}{\sum_i n_i (n_i b_i)} \quad (4.9)$$

where n'_i is the number of blocked member nodes in the i^{th} multicast request, and $n_i b_i$ is the bandwidth required by one member node of the i^{th} multicast request (which includes the bandwidth of sending data, b_i , and the bandwidth of receiving data, $(n_i - 1)b_i$ from the rest of the member nodes). For FRN, RRN, BTM and BTJ the blocking probability formula is the same as equation (4.9) since there is only one multicast tree for a multicast session. But in MUT we have n_i unidirectional multicast trees. The blocked node for each single unidirectional tree of this approach may be different from the blocked nodes for another unidirectional tree of the same multicast session. Therefore we need to calculate the blocked bandwidth for each tree separately and then add them together. The multicasting blocking probability for the MUT is given by

$$P_b = \frac{\sum_i \sum_j^{n_i} (n'_{ij} b_i)}{\sum_i n_i (n_i b_i)} \quad (4.10)$$

where n'_{ij} is the number of the blocked nodes in the j^{th} unidirectional tree for the i^{th} multicast request. Since we have n_i unidirectional trees for the i^{th} multicast request, the denominator of this fraction would be $\sum_i n_i(n_i b_i)$ which is the same as the denominator of the blocking probability fraction in equation (4.9).

For some types of multicasting applications, it is strongly required that all of the multicast group members could be present in the multicast session and if even one of them cannot be included, the multicast session should not be established. We implemented a few simulation experiments to study the blocking probability in the network when partial blocking of the multicast group nodes is not allowed. In this case even if one of the multicast group nodes could not be included in the multicast tree, the multicast request will be blocked. For this situation we define blocking probability as follows:

$$P_b = \frac{\sum_j (\text{total required bandwidth for } j^{\text{th}} \text{ blocked multicast request})}{\sum_i (\text{total required bandwidth for } i^{\text{th}} \text{ multicast request})} \quad (4.11)$$

It can be calculated by

$$P_b = \frac{\sum_j n_j(n_j b_j)}{\sum_i n_i(n_i b_i)} \quad (4.12)$$

where n_j is the number of member nodes in the j^{th} blocked multicast request, $n_j b_j$ is the bandwidth required by one member node of the j^{th} blocked multicast request, n_i is the number of member nodes in the i^{th} multicast request, and $n_i b_i$ is the

bandwidth required by one member node of the i^{th} multicast request (which includes the bandwidth of sending data, b_i and the bandwidth of receiving data, $(n_i - 1)b_i$ from the rest of the member nodes).

4.3 Algorithms

In this section we describe several important algorithms that are used in this study. These include the algorithms for traffic grooming and multicast tree construction, the algorithm for the root-node selection in the multi root-node approach with fixed root-nodes, and the algorithm for calculating the required bandwidth in the BTJ approach.

4.3.1 Multicast Traffic Grooming and Multicast Tree Construction

We assume to have a central network manager which controls the overall network operation. This central network manager processes the multicast requests one at a time. If two or more multicast requests arrive at the same time, the central network manager picks one of them randomly and establishes a multicast session for that multicast request before proceeding to process the next one in a random manner.

As described earlier, the i^{th} multicast request is modeled as $M_i(D_i, b_i)$, where D_i is the set of the member nodes and b_i is the bandwidth required to transmit the data generated by any one of the member nodes. It should be mentioned that although one node requests for a multicast session for a known set of nodes (D_i),

after the establishment of the multicast session, all of the nodes in D_i are able to generate traffic and send it to the multicast group simultaneously.

Finding an optimum multicast tree is a Steiner tree problem which is NP hard. Following the technique suggested by Takahashi and Matsuyama [17], we here consider a shortest path heuristic (SPH) algorithm to find a multicast tree in the network. The main idea is that at each step we connect a member node with a minimum cost path to the existing partial multicast tree.

For a multicast request, if only one bidirectional tree is established, we can select a member node either by a random manner or by a certain algorithm to initiate a partial tree. Let M represent the set of the multicast group member nodes, let T' denote the partial tree and let M' ($M' \subseteq M$) be the set of multicast member nodes which are not connected to the partial tree yet. If a node of set M' is the closest to the established partial tree T' , we refer to that node in M' and the corresponding node in the partial tree as *neighbor nodes* [55]. Since we use the overlay model, these neighbor nodes could be found either in the IP/MPLS layer or the optical layer. After initiating the partial tree with a node, at each step we find a neighbor node to the tree and expand the tree until it covers all of the multicast member nodes or there is no possible way to connect any of the remaining multicast member nodes to the partial multicast tree.

Both topologies in the IP/MPLS layer and the optical layer have the same set of nodes but the graph edges are different in them. Let $D(I)$ denote the distance

between the neighbor nodes in the IP/MPLS layer. This would be actually done by calculating all the shortest paths between nodes in M' and the nodes of the (current) partial multicast tree and selecting the shortest one. If a node from M' is found with the shortest path from T' in the IP/MPLS feasible topology, it would be added to the partial tree and removed from M' . If there is no possible path between the nodes in M' and the partial tree in the IP/MPLS feasible virtual topology, we set $D(I) = \infty$. In a similar fashion, the distance between neighbor nodes in the optical layer is denoted by $D(O)$. When there is no possible path between the nodes in M' and T' in the IP/MPLS feasible topology, we calculate all the shortest paths between nodes in M' and the nodes of the (current) partial multicast tree in the dynamic topology and select the shortest one. A new lightpath would be established for the found shortest path. The new established lightpath is added to the virtual topology and we try to expand the partial tree in the new feasible virtual topology. If there is no path between nodes of M' and T' in the optical layer, then we set $D(O) = \infty$ and the remaining nodes in M' would be blocked. If there are more than one node in M' with the minimum distance from T' , we choose one of them randomly and expand the tree through that node. The whole idea of this algorithm is to keep the overall cost of the tree minimum by choosing the nearest multicast group member node to the partial tree in each step. This approach is described in depth as *algorithm1* in [55]. We use this algorithm to find and establish a multicast tree in all the multi-source multicasting approaches introduced in Chapter 3. Fig. 4-2 shows the procedure of establishing a multicast tree.

Fig. 4-2 Multicast tree construction flowchart.

4.3.2 Choosing a Root-Node in the Multi Root-Node Approach

As described earlier, in the multi root-node structure one approach would be to have multiple pre-defined and fixed root-nodes in the network and to choose a

most appropriate one for each coming multicast request. This is referred to as the Fixed Root-Node approach (FRN). In order to choose the most suitable root-node for the current multicast request, different parameters should be considered. Since the member nodes of a multicast group are chosen randomly among the nodes in the network, the locations of the multicast group member nodes are unpredictable. Therefore in order to choose a root-node for a multicast request we should first locate all the multicast group member nodes.

Assume that we choose one of the pre-defined root-nodes randomly without considering the locations of the multicast group member nodes. If the root-node has a large distance from a considerable number of the multicast group member nodes, a huge amount of bandwidth would be wasted to connect the root-node to them and reserve the needed bandwidth on those links. But if we could discover that there is a nearer root-node to most of the multicast group member nodes, then definitely we would make a better choice. Therefore choosing one of the root-nodes randomly is not a good choice.

Fig. 4-3 Choosing one of the root-nodes randomly in FRN

Fig. 4-3 shows a network with 8 nodes. R1 and R2 are two pre-defined root-nodes in this network. $(\{A, G, D\}, b)$ is a multicast request. Suppose that R1 is chosen randomly for this multicast request. It is shown in Fig. 4-3 that the total distance (which is defined as total number of link hops) of the minimum cost tree connecting R1, A, G and D is 4. However, the total distance of the minimum cost tree connecting R2, A, G and D is 3. In larger networks, choosing one of the root-nodes randomly could cause to a significant waste of network resources.

If we do not want to choose one of the root-nodes randomly, the easiest way to choose a root-node for this multicast request is to choose the nearest root-node to the sender of the multicast request. But if the sender has a long distance to a considerable number of the multicast group member nodes, this choice would be as bad as choosing a root-node randomly.

Fig. 4-4 Choosing the nearest root-node to sender in FRN

Fig. 4-4 shows a network with 8 nodes. R1 and R2 are two pre-defined root-nodes in this network. $(\{A, G, F\}, b)$ is a multicast request. Assume that node A has sent the multicast request. The distance between R1 and A is smaller than the

distance between R2 and A. But as shown in the picture, the size of the minimum cost tree including R1, A, G and F is 4 and the size of the minimum cost tree including R2, A, G and F is 3. Therefore choosing the nearest root-node to the sender of multicast request is not a good choice. This example is a small network. But in larger networks, choosing the nearest root-node to the sender of the multicast request to be the root-node could cause to a huge waste of network resources.

For each multicast request, in order to choose the most appropriate root-node from the pre-defined fixed root-nodes, we here consider to calculate the shortest path from each root-node to every member node in the multicast group. Then for each root-node we sum the lengths of the shortest path between the root-node and each member of the multicast group. The root node that has the smallest distance sum would be chosen as the root-node for the multicast request. Since the dynamic topology and the feasible virtual topology change frequently, for the sake of simplicity, we choose the physical topology to determine the shortest paths in this algorithm. The following pseudo code shows the proposed algorithm for choosing one of the predefined fixed root-nodes for the multicast request $M(D, b)$.

1. Multicast Request (D, b) arrives.
2. $M = D = \{m_1, m_2, \dots, m_n\}$ and $R = \{r_1, r_2, \dots, r_k\}$
// M is the set of multicast group member nodes and R is the set of root-nodes.
3. $i = 1, j = 1, \text{Distance} = \infty, \text{Root} = r_1, \text{RD} = 0$.
4. If $i > k$ goto 10.
// "k" is the number of the pre-defined fixed root-nodes.
5. If $j > n$ goto 8.
// "n" is the multicast group size (number of the member nodes).

6. $RD = (RD + \text{ShortestPathLength}(r_i, m_j))$
7. $j = j + 1$ goto 5.
8. If $RD < \text{Distance}$ then ($\text{Distance} = RD$, $\text{Root} = r_i$)
9. $i = i + 1$, $RD = 0$, goto 4.
10. return Root.

4.3.3 Calculating the Required Bandwidth in the BTJ Approach

In Section 3.5 we introduced the BTJ approach in order to establish one bidirectional multicast tree with just enough bandwidth reserved on the links of a multicast tree. As discussed in Chapter 3, the total needed bandwidth on each link of the tree is $n_i b_i$. However, the required bandwidth in each direction of each link is different and hence it is necessary to calculate the bandwidth for each direction of each link. The main idea is that since each link divides a tree into two smaller disjoint trees, we need to count the number of the multicast group member nodes on each side of the link in order to be able to calculate the exact amount of the needed bandwidth for each direction. And since the result of this calculation is strongly dependant on the location of each link in the multicast tree, we first must have a multicast tree and then apply this method to that. Therefore the proposed approach is to first find a bidirectional multicast tree for a new multicast request using the BTM approach, whereby a bandwidth of $(n-1)b_i$ is reserved on each link in each direction. After the tree is set up, the exact amount of bandwidth needed in each direction is calculated and the excess of the over-reserved bandwidth is released.

Here we choose the Depth First Search (DFS) algorithm [45] to count the number of member nodes on each side of each link of the multicast tree. DFS is a method to traverse all of the nodes of a tree. In DFS we take a branch in the tree and go as deep as possible without backtracking. When the node that we are in that has no more unvisited child, we backtrack to the most recent node that still has some unvisited children. When this algorithm ends, we have visited all of the nodes in the tree.

In order to calculate the exact amount of the bandwidth to be reserved in each direction of a multicast tree's link, we divide the multicast tree into two smaller trees by cutting the mentioned link. Then we apply the DFS algorithm to each of those smaller trees to count the number of the nodes in that tree. It should be mentioned that it is not necessary that every node in a multicast tree be a multicast member node. Some nodes may be used to make a connected tree to connect the multicast group member nodes. And for calculating the amount of needed bandwidth in each direction of each link, we actually need to count the number of multicast member nodes at each side of the link. Only member nodes may generate new data and the other nodes in the tree just forward the received data to the neighbors. Therefore at each step of DFS, when a node of the tree is detected, we check whether it is a multicast member node or not.

When we count the number of the multicast group member nodes at each side of a multicast tree's link, the only thing that we have to do is to release the excess reserved bandwidth in each direction and return it to the network resources

to be used by later requests. The following pseudo code shows the steps of this approach.

1. Multicast request (D, b) arrives.
2. $n = |D|$, $B = (n - 1) \times b$, $M' = D$, $T' = \{\text{random } m_i \in M'\}$, $M' = M' - \{m_i\}$
3. If $M' = \Phi$ goto 12.
4. Make the feasible topology for B .
5. Calculate $D(I)$.
6. If $D(I) = \infty$ goto 8.
7. Connect the neighbor nodes with bidirectional links in virtual layer. Establish LSPs in opposite directions on each link with B reserved for them. Update M' and T' . go to 3.
8. Calculate $D(O)$.
9. If $D(O) = \infty$ go to 11.
10. Connect the neighbor nodes in optical layer by establishing two lightpaths in opposite directions. Go to 3.
11. Block the remaining nodes in M' .
12. $m = 0$;
13. **For** any link $(i, j) \in T'$,
 - Apply DFS to the tree rooted at j without letting the DFS traverse the link (i, j) . Calculate the number of the multicast group member nodes that we visit by DFS and set m .
 - $B_{ij} = (n - m) \times b$
 - Release $(B - B_{ij})$ from the bandwidth reserved for the link (i, j) .
 - $B_{ji} = m \times b$
 - Release $(B - B_{ji})$ from the bandwidth reserved for the link (j, i) .
14. Establish T' .

The described approach is the straight forward method. In this study we used an optimal way that calls the DFS algorithm only once for each multicast tree.

4.4 Simulation Experiments and Results

We have conducted simulations to examine the multi-source multicasting approaches introduced in Chapter 3 and compare their performances in terms of the network blocking probability and the time needed for each approach to establish a multicast session.

4.4.1 Assumptions

We here use the NSF network in Fig. 4-5 and the Cost239 network in Fig. 4-6 as the test networks to examine the performances of different approaches of multi-source multicasting on the two test networks. The NSF network has 14 nodes and 21 links. The Cost239 network has 11 nodes and 26 links. The degree of a node in a graph is the number of edges joined to it. The average node degree is 3 in the NSF network and is 4.72 in the Cost239 network. The average hop distances between nodes in the NSF network and the Cost239 network are 2.14 and 1.56 respectively. The Cost239 network has higher average node degree and shorter average node distance than the NSF network. Hence, the NSF network is sparser than the Cost239 network [47].

Fig. 4-5 The NSF network topology

Fig. 4-6 The Cost239 network topology

Nodes UT, IL, NY and TX are chosen to be the fixed pre-defined root-nodes in the NSF network for the FRN method. In the Cost239 network, the Copenhagen, Brussels, Prague and Milan are chosen to be the fixed pre-defined root nodes.

Each optical fiber is supposed to support 8 wavelengths. In our simulation we used $x_1 = 0.01$ and $x_2 = 0.08$ in $[x_1C, x_2C]$ which is the range of the required bandwidth for the traffic generated by one of the multicast group member nodes while C is the bandwidth provided by one lightpath. We repeated each simulation experiment 20 times and the average value is calculated for results. In each simulation experiment each node in the network generates 1000 multicast requests. These requests come randomly from different parts of the network. Some of the multicast requests may come at the same time. We assume to have a central manager to receive these multicast requests and establish multicast sessions for them one by one. At each time only one multicast session can be established. If the central network manager is busy when a new request arrives, the new multicast request must wait in a queue for its turn. If two multicast requests arrive at the exact same time, one of them is picked by central manager randomly to be served first. The service rate is 0.22 multicast request per time unit. In our simulation experiments we produced the time units for arrival intervals and service rate. This time is not related to the computer's speed that runs the program. We obtained 95% confidence intervals for our results and show them in Fig. 4-8. However, they are so narrow that we omit them from other figures in order to improve readability.

4.4.2 Network Blocking Probability

First, we examine three different methods of selecting a root-node for a multicast request in the FRN approach. We want to test the performance of our

proposed algorithm in Section 4.3.2. There are two other methods. One approach is to choose one of the pre-defined root-nodes in the network randomly regardless of the location and distribution of the multicast member nodes and the current state of the network. The other tested method is to choose the root-node which is the nearest to the sender of the multicast request. Fig. 4-7 shows the blocking probability versus the traffic load per node in the NSF network.

Fig. 4-7 Blocking probability vs. effective traffic load for the NSF network, root-node selection (avg. multicast group size = 4.5)

As can be seen in Fig. 4-7, FRN with our proposed root-node selection algorithm shows a slightly better performance in terms of blocking probability compared to the other two methods. Since the root-node selection algorithm proposed in Section 4.3.2 shows a better performance compared to the other two

methods, in the subsequent simulation experiments we use this method in the FRN approach.

It is noted that the performance of the FRN approach strongly depends on the number and location of the pre-defined fixed root-nodes in the network. As discussed in Section 3.3.1, based on the frequency and bandwidth demand of the multicast requests and also the network size, there must be enough number of root-nodes to satisfy the needs of the network. The root-nodes also must be widely distributed in order to satisfy the needs of incoming multicast requests. If there is not enough number of root-nodes in one part of the network where many multicast groups are gathered, then they have to choose a far root-node to establish their trees and this would lead to unnecessary resource consumption and higher blocking probability.

The network blocking probability vs. traffic load per node for the multi-source multicasting approaches introduced in Chapter 3 is shown in Fig. 4-8 for the NSF network. 95% confidence intervals are shown in this figure.

Fig. 4-8 Blocking probability vs. effective traffic load for the NSF network (avg. multicast group size = 4.5)

It is observed that the MUT scheme achieves the lowest average blocking probability. The reasons are that, in the MUT scheme, (i) each unidirectional tree is rooted at each member node and optimized with respect to that member node; (ii) the needed bandwidth for the links of each unidirectional tree is b_i and hence a small fraction of free bandwidth on the existing lightpaths could be utilized. The RRN approach shows better performance than FRN because when we force the multicast request to choose one of the fixed root-nodes, there is the possibility that the selected root-node is a member node of the multicast group. Therefore, the multicast tree would be larger in this method. Larger multicast trees would consume more bandwidth resources, resulting in higher blocking probability. BTM has the highest blocking probability because the bandwidth is over-reserved on each link of a bidirectional tree. BTJ shows significant improvement in terms of

blocking performance compared to BTM since BTJ reserves just enough bandwidth on each link of a bidirectional tree. In BTJ by calculating the exact needed amount of bandwidth for each direction on each link of the multicast tree, we reduce the total reserved bandwidth on each link from $2(n-1)b_i$ to nb_i .

The network blocking probability vs. the traffic load per node for the Cost239 network is shown in Fig. 4-9.

Fig. 4-9 Blocking probability vs. effective traffic load for the Cost239 network (avg. multicast group size = 4.4)

Results for the Cost 239 network are similar to those for the NSF network. In the Cost239 network there is a significant difference between the blocking probabilities of FRN and RRN. The reason is that the Cost 239 network is a denser network compared to the NSF network and there is a higher possibility to find a multicast tree that only includes the multicast member nodes without using other

nodes to make a connected tree. Therefore by using FRN, if the selected root-node is not a member of the multicast group, the multicast tree will be larger than what is found in the RRN approach for the same multicast group. In the Cost239 network, FRN has a high blocking probability close to that of BTM. BTJ shows significant improvement in terms of blocking performance compared to BTM since BTJ reserves just enough bandwidth on each link of a bidirectional tree.

In Section 4.2, it was mentioned that for some multicasting application it is required that all of the multicast group member nodes be included in the multicast tree, and if some of the multicast group member nodes could not be connected to the multicast tree the multicast request will be blocked. Fig. 4-10 shows the network blocking probability vs. traffic load per node for the NSF network when there is no partial blocking. In this simulation experiment a multicast session is established for a multicast request only if all of the multicast group member nodes are included in the multicast tree.

Fig. 4-10 Blocking probability vs. effective traffic load for the NSF network where partial blocking is not allowed (avg. multicast group size = 4.5)

It is observed that the MUT scheme achieves the lowest average blocking probability. The RRN approach shows better performance than FRN. The BTM approach has the highest blocking probability because the bandwidth is over-reserved on each link of a bidirectional tree. BTJ shows significant improvement in terms of blocking performance compared to BTM since BTJ reserves just enough bandwidth on each link of a bidirectional tree.

The network blocking probability vs. the traffic load per node for the Cost239 network is shown in Fig. 4-11. In this simulation experiment a multicast session is established for a multicast request only if all of the multicast group member nodes are included in the multicast tree.

Fig. 4-11 Blocking probability vs. effective traffic load for the Cost239 network where partial blocking is not allowed (avg. multicast group size = 4.4)

Similar results were observed for the Cost239 network compared to the results for NSF network.

Fig. 4-12 shows the average multicasting blocking probability versus the ratio of unicast request to the total unicast and multicast requests in the NSF network.

Fig. 4-12 Blocking probability vs. unicast percentage for the NSF network (arrival rate = 2.5, avg. multicast group size = 6)

It is observed that the overall multicasting blocking probability decreases by increasing the ratio of unicast requests to the total unicast and multicast requests. The reason is that the multicast traffic load per node decreases by increasing the ratio of unicast traffic to the total multicast and unicast traffic (R_m decreases).

The average multicasting blocking probability versus the ratio of unicast request to the total unicast and multicast requests for Cost239 network is shown in Fig. 4-13.

Fig. 4-13 Blocking probability vs. unicast percentage for the Cost239 network (arrival rate = 7, avg. multicast group size = 6)

Similar results were observed for the Cost239 network compared to the NSF network.

Fig. 4-14 represents the average multicasting blocking probability versus average multicast group size in the NSF network where all the coming requests are multicasting requests.

Fig. 4-14 Blocking probability vs. avg. multicast group size for the NSF network (avg. arrival rate = 2)

In this simulation experiment the unicast percentage is fixed to 0, which means that all the requests are multicast requests. MUT achieves the best blocking performance. BTJ has the blocking performance very close to that of MUT, which shows that BTJ is very effective since BTJ is less complex than MUT and only one bidirectional tree is setup for a multi-source multicast session. As the multicast group size grows, the blocking probabilities of BTM, FRN and RRN become very close to each other.

The average multicasting blocking probability versus average multicast group size for Cost239 network is shown in Fig. 4-15.

Fig. 4-15 Blocking probability vs. avg. multicast group size for the Cost239 network (avg. arrival rate = 6)

Similar results were observed for the Cost239 network compared to the NSF network. For small multicast groups, BTM shows lower blocking probability than FRN. Also MUT and BTJ have blocking probabilities very close to each other for small multicast groups.

Fig. 4-16 represents the average multicasting blocking probability versus average multicast group size in the NSF network where all the coming requests are multicasting requests. In this simulation experiment partial blocking is not allowed. A multicast session is established for a multicast request only if all of the multicast group member nodes are included in the multicast tree.

Fig. 4-16 Blocking probability vs. avg. multicast group size for the NSF network where partial blocking is not allowed (avg. arrival rate = 2)

In this simulation experiment the unicast percentage is fixed to 0, which means that all the requests are multicast requests. MUT achieves the best blocking performance. BTJ has the blocking performance very close to that of MUT, which shows that BTJ is very effective since BTJ is less complex than MUT and only one bidirectional tree is setup for a multi-source multicast session. As the multicast group size grows, the blocking probabilities of BTM, FRN and RRN become very close to each other.

The average multicasting blocking probability versus average multicast group size for Cost239 network is shown in Fig. 4-17. In this simulation

experiment a multicast session is established for a multicast request only if all of the multicast group member nodes are included in the multicast tree.

Fig. 4-17 Blocking probability vs. avg. multicast group size for the Cost239 network where partial blocking is not allowed (avg. arrival rate = 6)

Similar results were observed for the Cost239 network compared to the NSF network.

4.4.3 Time Complexity

So far we have compared the performance of different approaches of establishing the multi-source multicasting sessions from the aspect of network blocking probability. But there are other important factors that should also be considered in order to decide which approach to use. For example MUT with multiple unidirectional trees shows the least blocking probability, however this approach repeats the process of finding a multicast tree n_i times ($n_i = i^{th}$ multicast

group size) and this could lead to a higher delay compared to other approaches. We could measure the time that is taken for each approach to establish a multicast session but since it depends on the processing device and its current status, this factor is not reliable.

The search algorithm that is used to find the shortest paths is the Dijkstra's Shortest Path (SP) algorithm [46]. We assume that the time for other functions like variable assignments and comparisons is negligible. We counted the average number of the times that the Dijkstra's SP algorithm is used to establish a multicast session for one multicast request in different approaches. Based on our observations, this could be a good representation of the relative time complexity of different approaches and their relative required time to establish a multicast session.

4.4.3.1 Simulation Results

Fig. 4-18 shows the average number of times that the Dijkstra's SP algorithm is used for one multicast request versus the traffic load per node in the NSF network.

Fig. 4-18 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. effective traffic load for the NSF network (avg. multicast group size = 4.5)

MUT uses the search algorithm much more times than the other approaches, because this approach needs to repeat the process of finding a multicast tree n_i times for the i^{th} multicast request. FRN and RRN use the search algorithm slightly more times than BTM and BTJ. Although both of FRN and RRN require only one multicast tree for a multicast session, n_i unicast paths must be found between the multicast group members and the root-node, using the search algorithm. The average number of calls to the Dijkstra's SP algorithm for each multicast request depends on the size of the multicast group (multicast tree) and does not depend on the arrival rate of the requests. The reason that this value decreases slightly by increasing the arrival rate is that increase in the arrival rate causes to higher blocking probability and leads to smaller multicast trees.

Fig. 4-19 shows the average number of times that the Dijkstra’s SP algorithm is used for one multicast request versus the traffic load per node in the Cost239 network as a dense network.

Fig. 4-19 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. effective traffic load for the Cost239 network (avg. multicast group size = 4.4)

Similar results were observed for the Cost239 network compared to the NSF network. MUT has a great distance from the other approaches in terms of search algorithm usage and therefore it takes a very longer time to establish the multicast sessions.

Fig. 4-20 represents the average number of times Dijkstra's algorithm is used for each multicast request versus the multicast group size in the NSF network.

Fig. 4-20 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. avg. multicast group size for the NSF network (avg. arrival rate = 2)

It is observed that when the size of the multicast group increases, more calls to the search algorithm are made because larger multicast trees should be established. In the MUT method, the number of calls to the search algorithm increases very rapidly in response to the increase in the average multicast group size, while this increases slowly in other approaches.

Fig. 4-21 represents the number of calls made to Dijkstra's SP algorithm for one multicast request versus the multicast group size for the Cost 239 network.

Fig. 4-21 Avg. number of times Dijkstra's algorithm is used for each multicast request vs. avg. multicast group size for the Cost239 Network (avg. arrival rate = 6)

We observed the similar results for the Cost239 network compared to the NSF network.

4.4.3.2 Order of the Time Complexity

Let E be the number of the links, and R be the number of the root-nodes in the network. The time required for establishing a multicast session with n member nodes for each of the approaches described in Chapter 3 is of the order given below:

$$\text{MUT: } \frac{n^2(n+1)}{2} O(E \log(N)) \quad (4.13)$$

$$\text{FRN: } (R + \frac{n(n+1)}{2} + n)O(E \log(N)) \quad (4.14)$$

$$\text{RRN: } (\frac{n(n+1)}{2} + n)O(E \log(N)) \quad (4.15)$$

$$\text{BTM and BTJ: } \frac{n(n+1)}{2}O(E \log(N)) \quad (4.16)$$

The term $O(E \log(N))$ in (4.11-4.14) represents the order of complexity for the Dijkstra's SP algorithm [46]. The Dijkstra's SP algorithm finds the shortest path between one node and all other network nodes in one run. Therefore at each step of the multicast tree establishment we have to call the Dijkstra's SP algorithm for every node in M' . Since the multicast group size is n , the number of the calls to the Dijkstra's SP algorithm for one multicast request would be $(n+(n-1)+(n-2)+\dots+1)$ which is equal to $\frac{n(n+1)}{2}$. The time needed for MUT is n times that of BTM and BTJ, since n unidirectional multicast trees are needed to be set up for MUT whereas only one bidirectional multicast tree is required for BTM and BTJ. Apart from what is required in BTM and BTJ, the FRN approach also needs to make R calls to Dijkstra's SP algorithm for the root-node selection procedure. There are additional n calls to Dijkstra's SP algorithm for n unicast path establishments in FRN and RRN. The reason that we have to make n calls to the Dijkstra's SP algorithm for finding the n unicast paths instead of just one call from the root-node is that we reserve the required bandwidth for each selected path

therefore the unicast paths cannot share the same path (the same portion of a lightpath capacity). The feasible virtual topology changes by reserving a path, therefore this procedure must be performed separately for n unicast path using the current feasible virtual topology.

4.5 Summary

In this chapter we explained the algorithms that are used for selecting one of the fixed root-nodes in FRN, establishing the multicast tree, the multicast traffic grooming and calculating the exact amount of bandwidth in each direction in BTJ. We also described the network and traffic models and the performance parameters.

The simulation results for the blocking probability and time complexity of the different approaches including MUT, FRN, RRN, BTM and BTJ are reported in this chapter. It is observed from the simulation results that MUT has the lowest blocking probability. However, it requires a much longer time to establish a multicast session compared to other approaches. Although BTM is the fastest approach to establish a multicast session, it has the highest blocking probability and uses network resources more than other approaches. BTJ shows a significant improvement over BTM in network's efficiency. It also requires a much reduced time to establish a multicast session compared to MUT.

Chapter 5. Evaluation of Memory Consumption in Multi-Source Multicasting

Multicast and MPLS are two complementary technologies. Merging these two technologies and building multicast trees on top of MPLS networks enhances the network's performance and offers an efficient solution for the problems of multicast scalability and control overhead. Multicast reduces the network's bandwidth usage, while MPLS attempts to provide users with the required bandwidth. Multicast reduces the loads on servers and network equipment but it suffers from scalability problem. On the other hand MPLS provides flexibility, speed and scalability [34].

The information about the multicast trees is stored in Information Base (IB) tables in the routers for routing and control decisions. In this chapter first the IB table's structure for different multi-source multicasting approaches introduced in Chapter 3 is described. Then the memory consumption of the aforementioned tables is compared for different multi-source multicasting approaches.

5.1 IB Table Structure for Different M2M Multicast Approaches

In this section the structure of the IB tables we used for different approaches of multi-source multicasting is described.

5.1.1 Multiple Unidirectional Trees Approach (MUT)

A unique label is assigned to each multicast group. There are n_i unidirectional trees for each multicast request and each tree is rooted at one of the multicast request member nodes. Therefore both the multicast group label and the original sender of the packet are needed to identify the multicast tree that a packet must be sent to. The router in each node receives a packet on an incoming interface, finds the appropriate row in the information base table by checking its group label and its source. Then it should replicate it to the number of the outgoing interfaces and forward the replicated packets onto them. If there is only one outgoing interface for a multicast packet, no replication is needed. An example of the information base table is shown in Fig. 5-1.

Group and Label	Source	Incoming interface	Outgoing interface
G_1, L_{G1}	S_{1G1}	I_{G1S1}	O_{1G1S1}, O_{2G1S1}
G_2, L_{G2}	S_{3G2}	I_{G2S3}	$O_{1G2S3}, O_{2G2S3}, O_{3G2S3}$

Fig. 5-1 Information Base (IB) table that is stored in all nodes for the MUT

In Fig. 5-1, L_{G1} represents the label used for multicast group G_1 ; S_{1G1} shows the source of this unidirectional tree; I_{G1S1} is the incoming interface at this node (since here we have unidirectional trees with one source for each, in each node we have at most one incoming interface for a particular tree in a particular multicast session); and O_{1G1S1} shows the first outgoing interface for the multicast packet that belongs to the unidirectional tree originated at S_1 in multicast group G_1 . The first and second rows show that in this particular node there are two outgoing interfaces

for multicast data from S_1 of G_1 and three outgoing interfaces for multicast data from S_3 of G_2 , respectively.

5.1.2 Root-Node Approach (RRN and FRN)

Here for each multicast session we have one unidirectional multicast tree originated at one root-node and n unicast paths between the multicast group member nodes and the root-node. In this approach three different tables need to be kept in the control plane for one multicast session. First we have a table in the root-node to forward the incoming data from the member nodes of a multicast group onto the appropriate outgoing interfaces. Here the incoming packet must be replicated to the number of the outgoing interfaces. In this table the original sender of the packet is not important. All the packets that have the same multicast group label must be forwarded onto the same set of outgoing interfaces. An example is shown in Fig. 5-2.

Group and Label	Incoming Interface	Outgoing Interface
G_1, L_{G1}	$I_{G1S1}, I_{G1S2}, \dots, I_{G1Sn}$	$O_{1G1}, O_{2G1}, \dots, O_{mG1}$
G_2, L_{G2}	$I_{G2S1}, I_{G2S2}, \dots, I_{G2Sn}$	$O_{1G2}, O_{2G2}, \dots, O_{mG2}$

Fig. 5-2 Information Base (IB) table that is stored in root-nodes for FRN and RRN

In Fig. 5-2, L_{G1} is the label used for multicast group G_1 . The incoming interfaces in the root-node for multicast group G_1 are shown by $I_{G1S1}, I_{G1S2}, \dots, I_{G1Sn}$. These are the interfaces of the unicast paths from the member nodes of multicast group G_1 to this root-node. Root-node must replicate the multicast group G_1 data

received from these interfaces and send them onto outgoing interfaces $O_{1G1}, O_{2G1}, \dots, O_{mG1}$.

The second table that should be kept in all nodes to facilitate forwarding of the received multicast packets to appropriate outgoing interfaces is shown in Fig. 5-3. This table is similar to the table in Fig. 5-1. The only difference is that in the table shown in Fig. 5-1 the source of a packet is not required because there is only one multicast tree for one multicast group. The router checks the multicast group label of an arriving packet on an incoming interface and finds the appropriate row in the table. Then it must replicate the packet to the number of outgoing interfaces and forward them onto the outgoing interfaces.

Group and Label	Incoming Interface	Outgoing Interface
G_1, L_{G1}	I_{G1}	$O_{1G1}, O_{2G1}, \dots, O_{mG1}$
G_2, L_{G2}	I_{G2}	$O_{1G2}, O_{2G2}, \dots, O_{mG2}$

Fig. 5-3 Information Base (IB) table that is stored in all nodes for FRN and RRN

The multicast data with label L_{G1} received on I_{G1} will be forwarded onto $O_{1G1}, O_{2G1}, \dots, O_{mG1}$. Replication must be done if there is more than one outgoing interface.

As mentioned earlier, in this approach, we reserve n unicast paths from the member nodes to the root-node. The 3rd table needs to be kept in the routers that are on the way of unicasting paths from the source nodes to the root-node. For the

unicast packets from the source nodes to the root-node, the source of the data is needed to be known because the unicasting paths from the source nodes to the root-node might pass some common lightpaths on their way. Routers must check the source and the multicast group label of the arriving packet and forward it onto the appropriate outgoing interface. An example is given in Fig. 5-4.

Group and Label	source	Incoming Interface	Outgoing Interface
G ₁ , L _{G1}	S ₁	I _{G1}	O _{1G1}
G ₂ , L _{G2}	S ₄	I _{G2}	O _{1G2}

Fig. 5-4 Information Base table that is stored for unicast paths in FRN and RRN

Since this table is for unicast paths, at each router, there is one incoming interface and one outgoing interface for the data coming from a particular source in a particular multicast group.

5.1.3 One Bidirectional Tree Approach (BTM and BTJ)

In this approach since a multicast tree is bidirectional, the data flow is in both directions on any link of the tree. In the forwarding table for the bidirectional tree approach, the source of the data is not important since the trees are not unidirectional. At each node, the relevant interfaces for each multicast group label must be stored. There is an equal outgoing interface for each incoming interface in each node of the tree since there are two unidirectional LSPs on each bidirectional link of the multicast tree. Upon receiving a multicast packet at each node, the router checks the multicast group label of the packet and the incoming interface

and forwards it onto the relevant outgoing interfaces of the multicast tree excluding the one whose parallel incoming interface receives the packet. Obviously when there is more than one outgoing interface to forward the packet, replication of the packet is performed by the router. An example for the forwarding table in both BTM and BTJ is shown in Fig. 5-5.

Group and Label	Incoming Interfaces	Outgoing Interfaces
G_1, L_{G1}	$I_{1G1}, I_{2G1}, \dots, I_{mG1}$	$O_{1G1}, O_{2G1}, \dots, O_{mG1}$
G_2, L_{G2}	$I_{1G2}, I_{2G2}, \dots, I_{kG2}$	$O_{1G2}, O_{2G2}, \dots, O_{kG2}$

Fig. 5-5 Information Base (IB) table that is stored in all nodes for BTM and BTJ

The received data with label L_{G1} will be replicated and forwarded on $O_{1G1}, O_{2G1}, \dots, O_{mG1}$ excluding the one whose parallel incoming interface receives the packet.

5.2 Simulation Experiments

We have carried out simulation to examine the memory consumption of the IB tables required in the control plane for each of MUT, FRN, RRN, BTM and BTJ approaches to compare their performances.

5.2.1 Assumptions

The same assumptions made in Section 4.4.1 are used for the simulation experiments reported in this chapter. The IB tables introduced in Section 5.1 are considered for MUT, FRN, RRN, BTM and BTJ approaches. We measure the total number of the records in all IB tables in the network and then we divide it to the

number of multicast requests in the network. Therefore the result of the simulation experiments in this chapter is the average number of the records related to one multicast group in different IB tables in the network. This could represent the memory consumption of one multicast group in a specific multi-source multicasting approach.

5.2.2 Simulation Results

Fig. 5-6 represents the average number of the records (rows) for one multicast session in IB tables in different nodes of the network versus traffic load per node for the NSF network.

Fig. 5-6 Avg. number of records for one multicast session vs. effective traffic load for the NSF network (avg. multicast group size = 4.5)

Here all the coming requests are multicast requests ($R_m=1$). It is observed from the simulation results that MUT has the highest average number of the

records in the control plane for keeping the information of the multicast sessions and multicast trees. This is because in MUT the information of n multicast trees must be stored in the control plane. The average number of the records for FRN and RRN is significantly less than that of MUT. The BTM and BTJ have the least number of records in the control plane for one multicast session. Although there is only one multicast tree in all FRN, RRN, BTM and BTJ, but FRN and RRN must keep track of the unicast paths from the multicast member nodes to the root-nodes too. The number of the records in the control plane depends on the multicast group size and the number and size of paths and trees that are established for that multicast group. It is not related to the arrival rate. The reason that the number of the records is decreased slightly by increasing the traffic load is that as the arrival rate increases, the blocking probability also grows. Therefore the size of the established trees would be smaller which leads less information that needs to be kept in the control plane.

Fig. 5-7 shows the average number of the records (rows) for one multicast session in Information Base tables in different nodes of the network versus traffic load per node for the Cost239 network (a relatively dense network).

Fig. 5-7 Avg. number of records for one multicast session vs. effective traffic load for the Cost239 network (avg. multicast group size = 4.4)

The results for the Cost239 network are similar to the results for the NSF network.

Fig. 5-8 shows the average number of records in the control plane for one multicast session versus the average size of the multicast group for the NSF network.

Fig. 5-8 Avg. number of records for one multicast session vs. multicast group size for the NSF network (avg. arrival rate = 2)

It is observed from the simulation results that by increasing the average multicast group size, the average number of the records in the control plane increases. This is due to the fact that bigger multicast groups need larger multicast trees and more reserved paths, which would involve more network nodes and hence lead to higher number of records in the control plane. FRN, RRN, BTM and BTJ show a slow growth in the average number of the records for one multicast session as the average multicast group size increases, while MUT shows a rapid increase in information volume in the control plane as the multicast group size increases. This is because in the MUT approach, the number of unidirectional multicast trees to be set up is proportionally increased with the multicast group size.

Fig. 5-9 shows the average number of records in the control plane for one multicast session versus the average size of the multicast group for the Cost239 network.

Fig. 5-9 Avg. number of records for one multicast session vs. multicast group size for the Cost239 network (avg. arrival rate = 6)

The results for the Cost239 network are similar to the results for the NSF network.

5.3 Summary

The information about the established multicast sessions must be kept in Information Base (IB) tables in the control plane for forwarding and control decisions. This information includes the multicast group labels, the incoming and outgoing interfaces in each router related to a particular multicast request and the sources of the multicast data. In this chapter we reported the simulation

experiments results about the number of the records in different IB tables in the network related to one multicast session.

It is observed from the simulation results that BTM and BTJ have the lowest volume of information in the control plane. The number of the records in control plane for one multicast session for FRN and RRN approaches is higher than that of BTM and BTJ but significantly lower than that of MUT. MUT has the highest volume of information in control plane. The number of the records in IB tables for MUT increases very rapidly in response to the increase in the average multicast group size, while this increases slowly in other approaches.

Chapter 6. Conclusions and Recommendations

This thesis studied the multi-source multicasting in IP/MPLS over WDM networks. We discussed the requirements of establishing the multi-source multicasting sessions and presented three different main approaches. We carried out simulations to evaluate the performance of these approaches in terms of network blocking probability, time complexity and memory consumption in the control plane. Some enhancements were proposed in order to improve the performance of multi-source multicasting.

6.1 Summary

In many present and emerging bandwidth intensive applications, it is required to send the same information from one node to a group of selected nodes. Multicasting is the technique to send only one copy of each packet on the same link. Applications such as video conferencing, distance learning and online interactive multi-player games use multicasting in nature, since there is usually more than one destination to receive the same data simultaneously. There are different strategies for establishing the multicast sessions. We consider multi-source multicast sessions in which all of the multicast group member nodes can send data to the multicast group simultaneously.

In Chapter 3 we introduced three different approaches of establishing a multi-source multicasting session. They are as follows:

- **Multiple Unidirectional Trees approach (MUT):** In this method, multiple unidirectional multicast trees are established for one multicast session, each rooted at one of the multicast group member nodes. Each tree transmits the data originating at one of the member nodes to other members of the same multicast group.
- **Multi root-node approach:** In this method only one unidirectional multicast tree is established for each multicast session. The root of this tree is called root-node. All of the multicast group member nodes send their data to the root-node through unicasting and then the root-node multicasts it to the multicast group. The root of the multicast tree could be one of the pre-defined Fixed Root-Nodes in the network (FRN) or be selected randomly among the multicast group member nodes (RRN).
- **One bidirectional tree approach:** In this method only one bidirectional multicast tree connects the multicast group member nodes for each multicast session. There is sufficient bandwidth reserved on each link of the tree so that each node could send data to the group and receive the data generated by other members of the same multicast group simultaneously. Two methods were considered for reserving bandwidth on each link of the tree. The first proposal was to reserve the maximum needed bandwidth in both directions on every link of the multicast tree. This approach is referred to as one Bidirectional Tree

with Maximum reserved bandwidth (BTM). As BTM over-reserves bandwidth on each link of the tree, it is not an efficient method. To tackle this problem, we proposed a method to calculate the just enough bandwidth to be reserved on each link of the bidirectional tree and release the excess reserved bandwidth in BTM. This approach is referred to as one Bidirectional Tree with Just enough reserved bandwidth (BTJ).

In Chapter 4 we described the algorithms that are used for selecting one of the fixed root-nodes in FRN, establishing the multicast tree, the multicast traffic grooming and calculating the exact amount of bandwidth in each direction for the BTJ. We also described the network and traffic models and the performance parameters. The simulation results for the blocking probability and time complexity of MUT, FRN, RRN, BTM and BTJ approaches were reported in this chapter.

It is observed from the simulation results that MUT has the lowest blocking probability while it requires a much longer time to establish a multicast session compared to other approaches. Although BTM is the fastest approach to establish a multicast session, it has the highest blocking probability therefore it uses network resources more than other approaches. The BTJ approach shows significant improvement in network efficiency compared to FRN, RRN and BTM. It also shows much reduced time in establishing a multicast session compared to MUT.

The information about the established multicast sessions must be kept in Information Base (IB) tables in the control plane for forwarding and control decisions. This information includes the multicast group labels, the incoming and outgoing interfaces in each router related to a particular multicast group and the source of the multicast data. In Chapter 5 we reported simulation results for the number of the records stored in different IB tables related to one multicast session for different multi-source multicasting approaches.

It is observed from the simulation results that BTM and BTJ have the lowest volume of information in the control plane. The number of the records in control plane for one multicast session for the FRN and RRN approaches is higher than that of the BTM and BTJ approaches but significantly lower than that of MUT. MUT has the highest volume of information in control plane.

6.2 Recommendation for Further Research

So far most of the research work in this area considers single source multicasting. With the advent of the new bandwidth intensive interactive applications, more research is desired for bandwidth intensive multi-source multicasting. The threads are open either in the multi-source multicasting protocol design or the algorithms of routing and resource allocation. In the following, we summarize some possible future research topics.

- As reported in Chapter 3, in the Fixed Root-Node approach (FRN) an important issue is the selection of some nodes in the network as the fixed root-nodes. The number and the location of these specific nodes have a great impact on resource allocation to the multicast sessions. This could be decided more efficiently if an efficient procedure can be developed to study the traffic behaviors and network current status.
- Further studies are required in order to find more efficient traffic grooming policies for the introduced multi-source multicasting approaches. By applying more efficient traffic grooming policies to the multi-source multicasting, the blocking probability could be decreased and hence the network would be utilized in a better way.
- Further research is required in order to study the protection of the multicast sessions for the introduced multi-source multicasting approaches in the case of link failure or node failure.
- In this study we assumed that for each coming multicast request, all the senders and receivers are known and the size of an existing multicast group does not change during the application's running time. However in practice many nodes may join a current multicast session or leave it. Further research is needed to study the resource allocation, the message passing

and memory consumption in the control plane for joining and leaving a multicast session.

References

- [1] S. Dixit, *IP over WDM: Building the Next-Generation Optical Internet*, Chap 1, WILEY INTERSCIENCE, 2003.
- [2] P. De Dobbelaere, K. Falta, L. Fan, S. Gloeckner, and S. Patra, "Digital MEMS for optical switching," *IEEE Communications Magazine*, vol. 40, pp. 88-95, Mar. 2002.
- [3] Q. Zheng and G. Mohan, "Protection approaches for dynamic traffic in IP/MPLS-over-WDM networks," *IEEE Communications Magazine*, vol. 41, pp. 24-29, May 2003.
- [4] K. E. Stubkjaer, et al, "Wavelength converter technology," *IEICE Transactions on Communications*, vol. E82 C, no. 2, pp. 338-347, Feb 1999.
- [5] X. Niu, W. D. Zhong, G. Shen, and T. H. Cheng, "Connection establishment of label switched paths in IP/MPLS over optical networks," *Photon. Netw. Commun.*, vol. 6, no. 1, pp. 33-41, Jul. 2003.
- [6] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," <http://www.faqs.org/rfcs/rfc3031.html>, IETF RFC 3031, 2001.
- [7] K. Zhu and B. Mukherjee, "A review of traffic grooming in WDM optical network: architectures and challenges," *Optical Networks Mag.*, vol. 4, no. 2, pp. 55-64, Mar. 2003.
- [8] K. Zhu, H. Zang, and B. Mukerjee, "A comprehensive study on next-generation optical grooming switches," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 7, pp. 1173-1186, Sep. 2003.
- [9] K. Y. Zhu and B. Mukherjee, "Traffic grooming in an optical WDM mesh network," *IEEE Journal on Selected Areas in Communications*, vol. 20, pp. 122-133, Jan. 2002.
- [10] D. Banerjee and B. Mukherjee, "Wavelength-routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study," *IEEE/ACM Transactions on Networking*, vol. 8, pp. 598-607, Oct. 2000.
- [11] W. Sun, et al, "Prototype demonstration of IP multicasting over optical networks with dynamic point-to-multipoint configuration," *Proc. OFC 2005*, vol. 3, pp. 133-135, Mar. 2005.
- [12] L. H. Sahasrabudde and B. Mukherjee, "Light-trees: optical multicasting for improved performance in wavelength-routed networks," *IEEE Commun.Mag.*, vol. 37, pp. 67-73, Feb. 1999.
- [13] W. Liang, "Constructing Multiple Light Multicast Trees in WDM Optical Networks," *Proceedings. 7th International Symposium on Parallel Architectures, Algorithms and Networks*, pp.482-488, May 2004.

References

-
- [14] A. Matrawy, W. Yi, I. Lambadaris, and C. H. Lung, "MPLS-based multicast shared trees," *Proceedings of the 4th Annual Communication Networks and Services Research Conference*, pp. 227-234, May 2006.
 - [15] E. Modiano and P. J. Lin, "Traffic grooming in WDM networks," *IEEE Communications Magazine*, vol. 39, pp. 124-129, Jul. 2001.
 - [16] C. Cavazzoni, et al, "The IP/MPLS over ASON/GMPLS test bed of the IST project LION," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, pp. 2791-2803, Nov. 2003.
 - [17] H. Takahashi and A. Matsuyama, "An approximate solution for the steiner problem in graphs," *Mathematica Japonica*, vol. 24, no. 6, pp. 573-577, 1980.
 - [18] S. Bigo, et al, "Transmission of 125 WDM channels at 42.7 Gbit/s (5 Tbit/s capacity) over 12*100 km of TeraLight TM Ultra fibre," in *Proc. Proceedings 27th European Conference on Optical Communication, Amsterdam, Netherlands*, Sep. 2001.
 - [19] P. J. Wan, G. Calinescu, L. W. Liu, and O. Frieder, "Grooming of arbitrary traffic in SONET/WDM BLSRs," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1995-2003, Oct. 2000.
 - [20] B. Mukherjee, D. Banerjee, S. Ramamurthy, and A. Mukherjee, "Some principles for designing a wide-area WDM optical network," *IEEE/ACM Transactions on Networking*, vol. 4, pp. 684-696, Oct. 1996.
 - [21] R. Ramaswami and K. N. Sivarajan, "Design of logical topologies for wavelength-routed optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, pp. 840-851, Jun. 1996.
 - [22] A. E. Gencata and B. Mukherjee, "Virtual-topology adaptation for WDM mesh networks under dynamic traffic," *IEEE/ACM Transactions on Networking*, vol. 11, pp. 236-247, Apr. 2003.
 - [23] J. F. P. Labourdette and A. S. Acampora, "logically rearrangeable multihop lightwave networks," *IEEE Transactions on Communications*, vol. 39, pp. 1223-1230, Aug. 1991.
 - [24] D. Papadimitriou, J. Drake, J. Ash, A. Farrel, L. Ong, "Requirements for Generalized MPLS (GMPLS) Signaling Usage and Extensions for Automatically Switched Optical Network (ASON)," <http://www.faqs.org/rfcs/rfc4139.html>, IETF RFC 4139, Jul. 2005.
 - [25] B. Rajagopalan, J. Luciani, and D. Awduche, "IP over Optical Networks: A Framework," <http://www.faqs.org/rfcs/rfc3717.html>, IETF RFC 3717, Mar. 2004.
 - [26] A. R. B. Billah, B. Wang, and A. A. S. Awwal, "Multicast traffic grooming in WDM optical mesh networks," *GLOBECOM '03. IEEE*, vol. 5, pp. 2755-2760, 2003.
 - [27] R. Perlman, "Models for IP multicast," *12th IEEE International Conference on networks*, vol. 2, pp. 678-682, 2004.

References

-
- [28] J. H. Cui, J. Kim, D. Maggiorini, K. Boussetta, and M. Gerla, "Aggregated Multicast — A Comparative Study," *Cluster Computing-The Journal of Networks Software Tools and Applications*, vol. 8, no. 1, pp. 15-26, Jan. 2005.
 - [29] S. E. Deering. "Multicast Routing in a Datagram Internetwork" *PhD thesis*, Stanford University, Dec. 1991.
 - [30] J. Tian and G. Neufeld, "Forwarding State Reduction for Sparse Mode Multicast Communication," *INFOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies.*, vol. 2, pp. 711-719, Apr. 1998.
 - [31] A. Boudani and B. Cousin, "Multicast tree in MPLS network," *IEEE Global Telecommunications Conference (GLOBECOM 05)*, vol. 2, pp. 1017-1022, 28 Nov.-2 Dec. 2005.
 - [32] L. Xia, J. Y. Zhang, J. Li, and Y. Lan, "MTDS: A Multicast Tree Dynamic Switching Approach Based on Rate," *International Conference on Wireless Communications, Networking and Mobile Computing WiCom 2007*, pp. 2936 - 2939, Sep. 2007.
 - [33] M. Ramalho, "Intra- and Inter-Domain Multicast Routing Protocols: A Survey and Taxonomy," *IEEE Communications*, vol. 3, no. 1, pp. 2-25, Mar. 2000.
 - [34] S. Deering, et al, "The PIM Architecture for Wide-Area Multicast Routing," *IEEE/ACM Transactions on Networking*, vol. 4, no. 2, pp. 153-162, Apr. 1996.
 - [35] C. C. Wen, C. S. Wu, and K. J. Chen, "Centralized Control and Management Architecture Design for PIM-SM Based IP/MPLS Multicast Networks," *Global Telecommunications Conference GLOBECOM IEEE*, pp.443- 447, Nov. 2007.
 - [36] J. S. Pedersen and N. Chilamkurti, "A new approach to Multicasting and QOS within an MPLS backbone," *International Conference on Wireless and Optical Communications Networks*, pp.-5, 2006.
 - [37] X. Y. Li and X. L. Gui, "Merging Source and Shared Trees Multicast in MPLS Networks," *Seventh International Conference on Parallel and Distributed Computing, Applications and Technologies, PDCAT*, pp.23- 28, Dec. 2006.
 - [38] S. Deering, "Host extensions for IP Multicasting", <http://www.faqs.org/rfcs/rfc1112.html>, IETF RFC 1112. Aug. 1989.
 - [39] J. Moy, "Multicast Extensions to OSPF", <http://www.faqs.org/rfcs/rfc1584.html>, IETF RFC 1584, Mar. 1994.
 - [40] A. Ballardie. "Core Based Trees (CBT version 2) Multicast Routing", <http://www.faqs.org/rfcs/rfc2189.html>, IETF RFC2189, Sep. 1997.
 - [41] P. WG, B. Fenner, M. Handley, R. Kermode and D. Thaler, "Bootsmp Router (BSR) Mechanism for PIM Sparse Mode." <http://tools.ietf.org/html/draft-ietf-pim-sm-bsr-03>, PIM Draft. Feb. 2003.

References

-
- [42] B.Fenner and D.Meyei, "Multicast Source Discovery Protocol (MSDP)", <http://www.rfc-editor.org/rfc/rfc3618.txt>, IETF RFC3618, Apr. 2003.
 - [43] J. P. E Green, *Fiber Optic Networks*, Englewood Cliffs, NJ: Prentice Hall, 1993.
 - [44] B. Mukherjee, *Optical Communication Networks*, New York: McGraw-Hill, 1997.
 - [45] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*, Second Edition. MIT Press and McGraw-Hill, 2001 Section 22.3: Depth-first search, pp.540–549.
 - [46] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*, Second Edition. MIT Press and McGraw-Hill, Section 24.3: Dijkstra's algorithm, pp.595–601.
 - [47] B. Chen, "Dynamic Traffic Grooming in IP/MPLS over WDM networks", *PhD thesis*, Nanyang Technological University, 2006.
 - [48] A. K. Salkintzis, *Mobile Internet: Enabling Technologies and Services*, Chap 7, CRC Press, 2004.
 - [49] N. K. Tan, *MPLS for Metropolitan Area Networks*, CRC Press, 2004.
 - [50] D. Waitzman, C. Partridge, and S. Deering, "Distance Vector Multicast Routing Protocol", <http://tools.ietf.org/html/rfc1075>, IETF RFC 1075, Nov. 1988.
 - [51] A. Adams, J. Nicholas, W. Siadak, "Protocol Independent Multicast - Dense Mode (PIM-DM)", <http://tools.ietf.org/html/rfc3973>, IETF RFC3973, Jan. 2005.
 - [52] K. Kurokawa, T. Yamamoto, K. Tajima, A. Aratake, K. Suzuki, and T. Kurashima, "High Capacity WDM Transmission in 1.0 μm Band over Low Loss PCF Using Supercontinuum Source," *Optical Fiber Communication/National Fiber Optic Engineers Conference, OFC/NFOEC 2008*, pp. 1-3, San Diego, CA, Feb. 2008.
 - [53] B. Chen, S. K. Bose, and W. D. Zhong, "Applying saturated cut method in dynamic multicast traffic grooming in IP/MPLS over WDM mesh networks", *Proc., International Conf. on Information, Communications and Signal Processing (ICICS)*, pp.1207-1211, Bangkok, Thailand, Dec. 2005.
 - [54] W. D. Zhong, Y. Onozato, and J. Kaniyil, "A copy network with shared buffers for large scale multicast ATM switching", *IEEE/ACM Trans. on Networking*, vol.1, no.2, pp.157-165, Apr. 1993.
 - [55] B. Chen, W. D. Zhong, S. K. Bose, and Y. Jin, "Dynamic multicast traffic grooming in IP/MPLS over WDM mesh networks", *Proc. COIN 2005*, pp.184-189, Chongqing, China, May 2005.
 - [56] C. Mas and I. Tomkos, "Comparison of transparent versus opaque interconnecting nodes of OADM rings with respect to failure location", *Proceedings of 7th International Conference on Transparent Optical Networks*, vol. 1, pp. 67-70, Jul. 2005.

References

-
- [57] Z. Albanna, K. Almeroth, D. Meyer, and M. Schipper, "IANA Guidelines for IPv4 Multicast Address Assignments", <http://tools.ietf.org/html/rfc3171>, IETF RFC3171, Aug. 2001.
 - [58] B. Chen, W. D. Zhong and S. K. Bose, "A path inflation control strategy for dynamic traffic grooming in IP/MPLS over WDM network", *Communications Letters, IEEE* vol. 8, no. 11, pp. 680-682, Nov. 2004.
 - [59] S. Balasubramanian and A.K. Somani, "On Traffic Grooming Choices for IP over WDM networks", *Proceedings of 3rd international conference on Broadband Communications, Networks and Systems 2006*, pp. 1-10, Oct, 2006.
 - [60] S. Koo, G. Sahin, and S. Subramanian, "Dynamic LSP provisioning in overlay, augmented, and peer architectures for IP-MPLS over WDM networks", *Proceeding of INFOCOM 2004*, vol. 1, pp. 514-523, Mar. 2004.
 - [61] R. M. Krishnaswamy and K. N. Sivarajan, "Design of logical topologies: A linear formulation of wavelength-routed optical networks with no wavelength changers," *IEEE/ACM Transactions on Networking*, vol. 9, pp. 186-198, Apr. 2001.
 - [62] H. Y. Zhu, H. Zang, K. Y. Zhu, and B. Mukherjee, "A novel generic graph model for traffic grooming in heterogeneous WDM mesh networks," *IEEE/ACM Transactions on Networking*, vol. 11, pp. 285-299, Apr. 2003.
 - [63] D. O. Awduche, "MPLS and traffic engineering in IP networks," *IEEE Communications Magazine*, vol. 37, pp. 42-47, Dec. 1999.
 - [64] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Optical Network Magazine*, vol. 1, pp. 47-60, Jan. 2000.
 - [65] C. Assi, A. Shami, M. A. Ali, Y. H. Ye, and S. Dixit, "Integrated routing algorithms for provisioning "sub-wavelength" connections in IP-over-WDM networks," *Photonic Network Communications*, vol. 4, pp. 377-390, July 2002.

Author's Publications Related to This Thesis

Journal Papers:

Sona Hassani and Wen-De Zhong, "Multi-Source Multicasting in IP/MPLS over WDM Networks," accepted for publication in Photonic Network Communications.

Conference Papers:

Sona Hassani, Wen-De Zhong and Sanjay Kumar Bose, "Multi root node structure for dynamic multi-source multicasting in IP/MPLS over WDM networks", Joint conference of the Opto-Electronics and Communications Conference, 2008 and the 2008 Australian Conference on Optical Fibre Technology, OECC/ACOFT, pp. 1-2, Jul. 2008.

Appendix I

$$P(X = k) = \frac{(1-q)q^{k-2}}{q - q^{N-1}} \quad 3 \leq k \leq N$$

Proof:

$$P(x = k) = \frac{(1-q)q^{k-1}}{X} \quad 3 \leq k \leq N$$

$$\sum_{k=3}^N P(x = k) = 1 \Leftrightarrow \sum_{k=3}^N \frac{(1-q)q^{k-1}}{X} = 1$$

$$\Leftrightarrow \frac{(1-q)}{X} \sum_{k=3}^N (q^{k-1}) = 1 \Leftrightarrow \frac{(1-q)}{X} \frac{q^2(1 - q^{N-2})}{1 - q} = 1$$

$$\Rightarrow X = q^2(1 - q^{N-2})$$

$$\Rightarrow P(x = k) = \frac{(1-q)q^{k-1}}{q^2(1 - q^{N-2})} = \frac{(1-q)q^{k-2}}{q - q^{N-1}} \quad 3 \leq k \leq N$$

And the average size of the multicast group is given by

$$E(x) = \frac{3q - 2q^2 - (N+1)q^{N-1} + Nq^N}{(1-q)(q - q^{N-1})}$$

Proof:

$$\begin{aligned} E(x) &= \sum_{k=3}^N P(x = k) \times k = \sum_{k=3}^N \frac{(1-q)q^{k-1}}{q^2 - q^N} \times k = \frac{(1-q)}{q^2 - q^N} \sum_{k=3}^N q^{k-1} \times k \\ &= \frac{(1-q)}{q^2 - q^N} \sum_{k=3}^N (q^k)' = \frac{(1-q)}{q^2 - q^N} \left(\sum_{k=3}^N q^k \right)' = \frac{3q - 2q^2 - (N+1)q^{N-1} + Nq^N}{(1-q)(q - q^{N-1})} \end{aligned}$$

Fig. I Avg. number of copies $E(x)$ or fanout vs. distribution parameter q