uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Mounib Khanafer

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.A.Sc. (Electrical Engineering)

GRADE / DEGREE

School of Information Technology and Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Extending BGP Protocol to Achieve Inter-Domain Routing in Optical Networks

TITRE DE LA THÈSE / TITLE OF THESIS

H. Mouftah

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

A. El Saddik

F. R. Yu

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

Extending BGP Protocol to Achieve Inter-Domain Routing in Optical Networks

By
Mounib Khanafer

Thesis submitted to the Faculty of Graduate and Postdoctoral Studies in partial fulfillment of the requirements for the MASc degree in Electrical Engineering

Ottawa-Carleton Institute of Electrical and Computer Engineering
School of Information Technology and Engineering
University of Ottawa

June 2007

Your file Votre référence
ISBN: 978-0-494-32458-5
Our file Notre référence
ISBN: 978-0-494-32458-5

Abstract

The revolution of Internet and its popularity among millions of users has resulted in a significant increase in the data traffic carried by today's telecommunications networks. The evolution of optical networks, along with its ingenious technologies like Wavelength Division Multiplexing (WDM), has played a critical role in coping with that huge data traffic. However, the unusual growth in the number of users has been paralleled by a similar growth in the size of the networks. The latter has complicated the task of network management and mandated the need to segregate huge networks into several administrative units, each controlled by a unique organization, known as Autonomous Systems (ASes). For routers in different ASes to communicate with each other, reachability information should be available at each router for it to know the optimal path to reach its targeted router. The latter task is handled in IP-based electronic networks by the de facto inter-domain routing protocol known as the Border Gateway Protocol (BGP). In order to achieve efficient inter-domain routing in optical networks, we better benefit from the experience gained from the extensive deployment of BGP in electronic IP-based networks. In this thesis, we study the problem of achieving inter-domain routing in optical networks using BGP. We incorporate some modifications in standard BGP and introduce a new version of it, called the e**X**tended **B**GP for **O**ptical **N**etworks (xBON), which can be deployed in optical networks. We build a model for xBON and simulate it using OPNET Modeler platform. Finally, simulation results are presented and analyzed.

Acknowledgement

I would like to express my deepest thanks to my supervisor, Prof. Hussein Mouftah, whose valuable instructions, discussions, and comments made this thesis elegant and successful. Prof. Mouftah's suggestions and extensive revisions of this thesis significantly enhanced my skills in technical writing.

Special thanks to Dr. Jing Wu, of the Communications Research Centre Canada (CRC), for his dedicated efforts in guiding me through my thesis work. Dr. Wu spent long hours in discussing my design and highlighting the technical challenges that require more attention. Dr. Wu's support and encouragement made a vital contribution in resolving a lot of encountered problems during my thesis work.

I would also like to extend my gratitude to my colleagues Salim Said and Ahmad Abdo for the valuable time they spent in discussing and reviewing my system model and its generated results.

Dedication

To my parents and elder brother who encouraged me to continue my graduate studies and inspired me to love research.

To my wife and daughter who supported me with their patience during my work in this thesis. A lot of time should have been spent with you and without your sacrifice this work would not be possible.

Table of Contents

List of Figures

List of Tables

Acronyms

AS	Autonomous System
BGP	Border Gateway Protocol
BP	Blocking Probability
CAG	Client Access Group
CAP	Client Access Point
CDB	Centralized Database
CI	Confidence Interval
CR-LDP	Constraint-Based Routed Label Distribution Protocol
D-node	Destination Node
DVR	Distance-Vector Routing
E-BGP	External BGP
EGP	Exterior Gateway Protocol
FF	First Fit
GMPLS	Generalized Multiprotocol Label Switching
I-BGP	Internal BGP
IGP	Interior Gateway Protocol
IGRP	Interior Gateway Routing Protocol
IS-IS	Intermediate System to Intermediate System
LSA	Link-State Advertisement
LSR	Link-State Routing
MP-BGP	Multiprotocol extensions for BGP

OBGP	Optical BGP
ORBGP	Optical Routing BGP
OSPF	Open Shortest Path First
OXC	Optical Cross-Connect
RIB	Routing Information Base
RIP	Routing Information Protocol
RSVP	Resource Reservation Protocol
RSVP-TE	RSVP with traffic engineering extensions
SLA	Service-Level Agreement
S-node	Source Node
SxS	Simple xBON Scheme
TE	Traffic Engineering
TTS	Time Threshold Scheme
WA	Wavelength Assignment
WADB	Wavelength Availability Database
WDM	Wavelength Division Multiplexing
WTS	Wavelength Threshold Scheme
xBON	Extended BGP for Optical Networks

Chapter 1

Introduction

1.1 Background

The unprecedented growth in data traffic, fueled by the popularity of Internet and the tremendous increase in its users, mandated dramatic changes in the telecommunication industry in order to cope with the needs for high-capacity transport. In fact, with the emergence of Internet, data traffic continually overtook voice traffic to become the main and dominant type of information that is being transported by high-speed communication networks. This phenomenon is essentially driven by the advent of tens of ingenious applications such as the World Wide Web, e-mail, video conferencing, and E-commerce as well as the emergence of applications such as telemedicine, remote instrumentation, virtual reality gaming, and video-on-demand [CH06]. The trend can also be demonstrated by noticing that an average phone call may last for 3 minutes while an average Internet call, via dialup lines, typically lasts for 20 minutes. This means that an Internet call generates about six times the traffic generated by a voice call [RA02].

Optical networking became the strongest candidate that is capable of handling such a huge increase in traffic over communication networks. Since the mid-1990s [CA02], optical networks entered a new era with the deployment of Wavelength Division Multiplexing (WDM) technology. Given that a single fiber can carry up to 100 Tbps of data [CA02], WDM technology could exploit the bandwidth of a single fiber by carrying the 100 Tbps data over a single wavelength. Therefore, the capacity of a single fiber can be increased

significantly by carrying multiple wavelengths at the same time. In other words, a fiber carrying 4 wavelengths will be having a total bandwidth of about 400 Tbps, which is much more than the bandwidth needed to handle all the telephone traffic in the United States during an hour of peak usage [CA02]. With a host of advantages that includes, high capacity, 20-year lifespan of fiber, low deployment costs, highly secure data communication, centralized administration, and the ability to support new protocols and higher speeds without replacing the infrastructure, optical networks promise to be at the lead of future high-speed telecommunication networks.

The aforementioned facts about optical networking attracted the leading telecommunications providers (Telcos) to start deploying and investing in optical networks. Major telecom players, like AT&T, Sprint, and WorldCom as well as emerging carriers, like Quest Communications, Williams Communications and Broadwing, represent some examples of the long list of companies that went into the optical networking business in the last decade (see [CA02] for details).

As optical networks became popular at a high pace, critical challenges and problems occurred eventually. One of the main challenges is the management and control of optical networks. Managing a large and dynamic optical network is a difficult task and the existing network management infrastructure is inadequately tooled to handle it [BE04]. In fact, just provisioning an optical connection between two nodes requires cumbersome steps that need extensive manual interventions and may take weeks, or even months, to be completed. This shows that, with optical networking, the cost of running a network remains very high although a major drop in network equipment prices is achieved. Therefore, to best benefit from the intelligent technological breakthroughs, like WDM, achieved with optical

networking, efficient control and management systems (or planes) to run the network are needed. In the rest of this thesis we mainly focus on the control plane.

As defined in [BE04], the control plane refers to the infrastructure and distributed intelligence that controls the establishment and maintenance of connections in the network. The intelligence imposed by the control plane is typically accomplished by various communication protocols that are broadly classified into signaling and routing protocols. In other words, the control plane can be viewed as being composed of a signaling component and a routing component. The signaling component is responsible for establishing, maintaining and releasing lightpaths. The routing component, on the other hand, is responsible for neighbor discovery, topology discovery, and path selection (see [LI02a]-[SA03b]). The scope of this thesis concentrates on the routing component of the control plane.

1.2 Motivation and Objective

In traditional IP-based networks, routing protocols are responsible for advertising topology and reachability information throughout the network. By collecting that information, each node in the network learns how to reach all the other nodes constituting the network. Routing is an essential component of the communication network management system. The task of managing a communication network becomes complicated as the size of the network increases. To simplify the management burden, networks that are huge in size are divided into multiple administrative units each of which is known as an Autonomous Systems (AS). An AS is composed of routers and networks that are managed by a single organization [ST04]. Routing among the routers within an AS is handled by "intra-domain" routing protocols (like Open Shortest Path First (OSPF) protocol and Intermediate System-to-

Intermediate System (IS-IS) protocol) while routing among the ASes themselves is handled by "inter-domain" routing protocols (like Border Gateway Protocol (BGP)). (In Chapter 2, Inter-Domain in Optical Networks, we discuss intra- and inter-domain routing protocols in more depth).

As optical networking paves the way for the future of high-speed data communications, routing in optical networks emerges among the important aspects that should be addressed properly. Yet major challenges against achieving efficient routing in optical networks are still to be resolved.

The main objective of this thesis is to address the inter-domain routing problem in optical networks.

1.3 Thesis Contribution

This thesis achieves two contributions to the area of routing in optical networks:

1. We propose a new inter-domain routing protocol that extends the de facto inter-domain routing protocol of IP-based networks, namely, the BGP protocol, to be deployable in optical networks.
2. We develop a new testing tool in OPNET Modeler, the platform used for our simulations, to validate our proposed protocol. Although used to study the performance of an inter-domain routing protocol, our tool can be generally used, and easily extended, to study routing in general (including intra-domain routing), signaling, and wavelength assignment problems in optical networks. With the capabilities provided by OPNET Modeler, our tool can also provide a researcher with a collection of built-in performance analysis utilities.

1.4 Thesis Outline

The rest of this thesis is organized as follows. Chapter 2 gives a detailed description of the routing problem in both IP-based networks and optical networks with an emphasis on the inter-domain routing protocols in optical networks. Chapter 3 provides a detailed description of a new inter-domain routing protocol for optical networks and discusses the modeling of that protocol. Chapter 4 provides details about the simulations conducted on our system model, presents the generated results, and analyzes the performance of our new protocol. Finally, Chapter 5 concludes our work and states future research directions.

Chapter 2

Inter-Domain Routing in Optical Networks

2.1 Introduction

In this chapter we give an overview of routing protocols in telecommunication networks with an emphasis on the problem of inter-domain routing in optical networks. We provide a review of previous proposals available in the literature to achieve effective inter-domain routing in the optical networking area.

As mentioned earlier, routing protocols are needed to distribute reachability information among the nodes of a network. In other words, for a node to communicate with any other node in the network, it needs to learn about all the paths that lead to that targeted node. Routing protocols take the role of informing each node of those paths.

This chapter is organized as follows. Section 2.2 describes the types of routing protocols that are deployed in IP-based networks. Section 2.3 concentrates on the BGP protocol as the main focus of our thesis. Section 2.4 discusses the challenges faced by IP-based routing protocols when deployed in optical networks. Section 2.5 provides a survey of previous work that has been conducted in the area of inter-domain routing in optical networks. Finally, Section 2.6 provides a summary of this chapter.

2.2 Types of Routing Protocols

In general, routing protocols can be classified into two main types: Distance-Vector Routing (DVR) protocols and Link-State Routing (LSR) protocols. In the following sub-sections we detail both of these two categories.

2.2.1 Distance-Vector Routing

With Distance-Vector Routing (DVR) approach, each node in the network maintains a cost value associated with each of the other nodes in the network. This cost value represents a certain link characteristic like the length of the link. For a source node (S-node) to communicate with a certain destination node (D-node), the minimum cost associated with the links directly connected to the S-node is chosen to determine the first hop towards that D-node. Then, the second hop is again chosen depending on the minimum cost and so on until the D-node is reached. This means that the final path chosen by the S-node to reach the D-node is the path with the minimum cost. This path will be stored by the S-node in a local routing table. Each node periodically exchanges its local routing table with its neighbors. Nodes use the received tables to create new local routing tables that have more information about the overall network topology. Only the paths with minimum costs are kept in these routing tables. The exchange of routing tables advises all the nodes in the network of the complete paths to reach all the nodes. Any change in the cost of a link (as a result of link failure or link cost update) directly affects the total costs of all the paths that contain this link. Changes of a link cost are firstly detected by the nodes connected to the link endpoints. These changes are incorporated in the latter nodes routing tables and are spread out, to the other nodes, in the periodic process of exchanging routing tables.

Routing Information Protocol (RIP) and Internet Gateway Routing Protocol (IGRP) are two well-known examples of DVR protocols. Although DVR protocols are simple to understand and implement, they suffer from many drawbacks. Basically, the process of exchanging routing tables among the nodes causes two problems:

- Routing tables are exchanged periodically. This means that changes occurring in the costs of links are not reported directly to the other nodes. The negative consequence is that longer times are consumed to fully populate the other node's local routing table with the optimal paths. This behavior leads to routing loops as well as wrong communication decisions. Refer to Figure 2.1 for an illustration. Assume that B needs to send a packet to D. The path with minimum cost to reach D is B-C-D. Also, assume that while B is in the

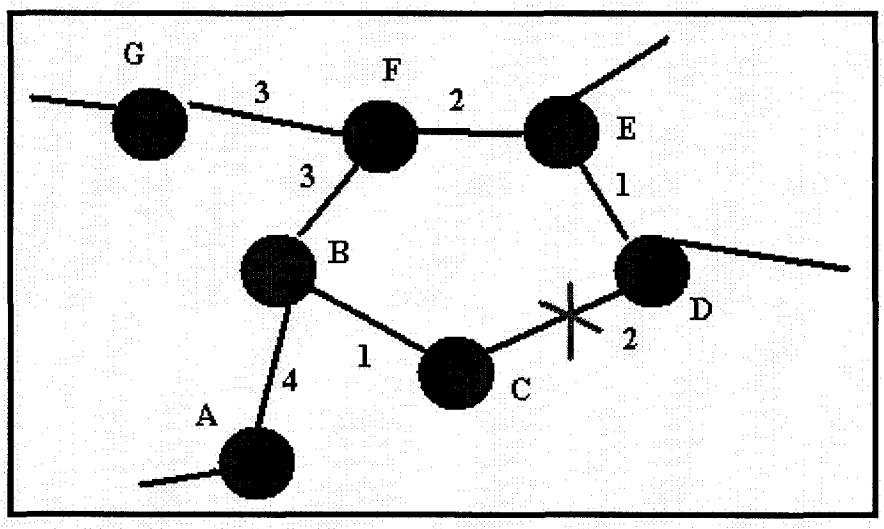

Figure 2.1 An example of the looping problem with RIP.

process of selecting the optimal path to D, the link C-D failed. C will not report this failure to B, and thus the invalidity of the path B-C-D, until its timer triggers it to send its routing table to the neighbors. Thus once B sends the packet to C, C will return the packet to B. Then, B will send the packet again to C until it is updated with the failure of the link C-D.

- DVR protocols are not able to scale well as the size of the network increases. This is because of the exchanging of the complete routing tables. These tables grow in size as

the network grows and it becomes impractical to continuously exchange such big tables among the nodes.

To overcome the aforementioned drawbacks, a variant version of DVR has been proposed and known as Path-Vector Routing (PVR). With PVR, nodes (instead of links) along the path to D-nodes are used to construct reachability information. That is, each node builds a local table that contains paths to reach all of the other nodes in the network, and these paths are nothing but a sequence of "next hops" to D-nodes. This is illustrated in Figure 2.2.

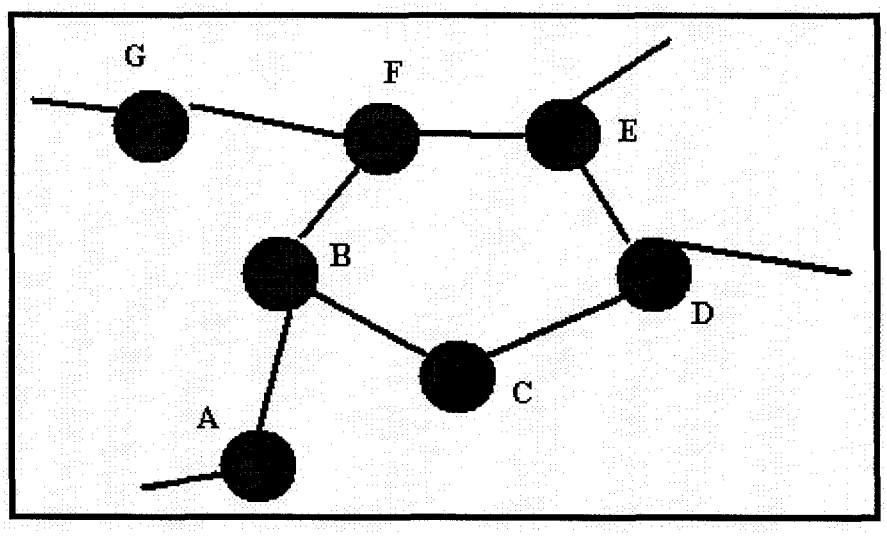

Figure 2.2 PVR advises node A, for example, that the path to reach node D is A-B-C-D or A-B-F-E-D.

The benefit is that routing loops are easily detected (by avoiding visiting the same node more than once along a certain path) and thus avoided. The Border Gateway Protocol (BGP) is a well known example of a PVR protocol. We discuss BGP in more details in Section 2.5.

2.2.2 Link State Routing

With Link State Routing (LSR) approach, each node constructs its own view of the network topology and link costs. Whenever a change in the link state occurs (link failure, link cost

change... etc), a message called the link-state advertisement (LSA) is generated and flooded throughout the network. Once a node receives an LSA, it re-computes the routes to all other nodes and updates its local routing table accordingly. It is important to note that nodes differ in their views of the states of the link due to the propagation delays experienced by routing updates to reach each node. The latter situation results in routing loops as the case is with DVR protocols. However, these loops eventually disappear as soon as all nodes receive the appropriate update about states of the links. The Open Shortest Path First (OSPF) and Intermediate System-to-Intermediate System (IS-IS) are two well-known examples of LSR protocols.

In contrast to DVR protocols, LSR protocols are simple to implement, more scalable, and less bandwidth-intensive. This is due to that updates are sent only in case of changes, and exchanging LSAs is much more practical, in terms of its size, than exchanging complete routing tables. On the other hand, LSR protocols are complex to implement, require intensive computations, and consume much of memory.

2.2.3 Routing and Autonomous Systems

As we have mentioned in Chapter 1, the global communications network is going through a dramatic growth in size day after day. One of the main challenges of having such a huge network is how to successfully and efficiently manage it in a way to guarantee high-quality services to the end-users. To cope with this problem, the concept of Autonomous Systems (ASes) has been introduced. Typically, an AS is a single administrative unit that is composed of routers (and even networks) and managed by a single organization.

Special attention should be paid for the "routing problem" when dealing with ASes. In fact, depending on whether the pair of routers talking to each other is contained within a single

AS or two different ASes, we can divide routing protocols, informing them of the reachability information, into two broad categories:

- Interior Gateway Protocols (IGPs) or Intra-Domain Routing protocols, which handle routing within an AS, and
- Exterior Gateway Protocols (EGPs) or Inter-Domain Routing Protocols, which handle routing among ASes themselves.

Both OSPF and IS-IS are two well known examples of IGPs while BGP is considered to be the de facto EGP that is extensively deployed in today's networks. The requirement of having different routing protocols to handle interior and exterior routing is driven by many reasons like (see [JE03] for more details):

- IGPs face a major scalability problem if deployed in large networks (that are spanning even a small number of ASes). This is due to the large amount of information that is continuously exchanged among routers. For instance, with RIP, complete routing tables are being exchanged. It is a big challenge to update all the routers, within all the ASes, with these routing tables. The case is the same with OSPF where LSAs are flooded upon detecting a change in any link. On the other hand, PVRs achieve a huge reduction in the amount of information it propagates through the network. This is attained by exchanging paths, rather than routing tables, among the border routers of the ASes, rather than all the routers.
- Routing protocols within a single AS, in addition to the primary role of carrying reachability information, may carry policy information to be implemented on the AS's routers. There is absolutely no need to propagate this policy information to other ASes as long as they are specific to the aforementioned local AS.

- The speed of routing convergence (that is, making all the routers aware of all the paths to reach each other within the AS) is essential for the performance of routing protocols within the AS. However, between the ASes themselves, routing convergence may not be of a critical issue. In other words, it is the main task of IGPs to distribute accurate and optimal paths to all the nodes and in a fast fashion while the main task of EGPs is to exchange reachability information across ASes so that data traffic can be communicated over the borders of ASes [BA03].
- It is essential for an AS, as an independent administrative unit, to preserve certain information about its network (like the local infrastructure for example) from being shared among different ASes. Using IGPs on the inter-domain links may leak such information to other ASes. EGPs, on the other hand, can force certain policies on the information crossing the borders of an AS as we show in Section 2.3.

2.3 Border Gateway Routing Protocol (BGP)

BGP is considered to be the standard EGP that is widely deployed in networks in the market. The main driver behind proposing BGP was to design a routing protocol that has the following features [JE03]:

a. Loop-free routing: this is attained by using a DVR algorithm that reveals the "paths" over which the information is to be sent.
b. Minimum exchange of routing information: this is achieved by adopting an "incremental updates" technique. The latter refers to that only "changes" of routing information, as they occur, are exchanged with neighbors (this is done, of course, after an initial phase to build complete routing tables at each node).

c. Depending on TCP as the reliable protocol needed to exchange the "incremental updates". This will avoid the burden of designing a new transport protocol from scratch.

d. Depending on "attributes" to encode special information about the paths that BGP distributes among neighbors.

Routers that are configured to run BGP are usually referred to as BGP-speakers. Only the edge routers of an AS, which are the ones connected to the edge routers of the neighboring ASes, are allowed to be configured as BGP-speakers. When two BGP-speakers attempt to exchange routing information between each other, they firstly open a "BGP session" together. In that case, these speakers are referred to as "BGP peers". To setup a BGP session, BGP peers should establish a TCP connection between themselves so that they guarantee a reliable delivery of data. Once a TCP connection is successfully established, BGP peers can start exchanging data according to the message formats mandated by BGP.

In the following sub-sections we give an overview of BGP and highlight some of its main advantages as an EGP or an inter-domain routing protocol.

2.3.1 BGP Messages

BGP uses four different messages: OPEN, UPDATE, NOTIFICATION, and KEEPALIVE. We describe briefly each of these messages in what follows.

- **OPEN Message**

This is the first message exchanged between the BGP peers after successfully setting up the TCP connection. Through this message, each peer introduces itself to the other and specifies the protocol parameters (like timers) that it will follow in its communication.

- **UPDATE Message**

This message is used to propagate routing information updates between routers. UPDATE messages handle both announcing a new path and withdrawing a previously announced path.

- **NOTIFICATION Message**

This message is used to inform BGP peers of an error that has occurred and that the present TCP connection will be closed (and thus the BGP session will be lost).

- **KEEPALIVE Message**

BGP peers exchange this message frequently to confirm that the connection between them is still alive.

A BGP-speaker stores all the routes it learns from its neighbors in a routing table that is referred to as the Routing Information Base (RIB).

2.3.2 Path Attributes

"Path attributes" are one of the most important features of BGP that contribute to its efficiency. Path attributes are part of BGP's UPDATE message and are used to indicate certain descriptions associated with the routes being advertised. The following are examples of the path attributes supported by BGP:

- **ORIGIN**: This attribute identifies the sources of the routes advertised by BGP. Basically, BGP learns the routes that it advertises from either IGPs or an EGP that BGP is replacing. The details of the latter fact are beyond the scope of this thesis.
- **AS-PATH**: This attribute specifies all the ASes over which the UPDATE message has traveled. These ASes form the paths from which BGP-speakers build their RIB.
- **NEXT-HOP**: This attribute specifies the IP address of the BGP-speaker (being the border of a certain AS) that should be used as the next hop to reach a destination node.

- **LOCAL-PREFERENCE**: This attribute is used to mark a certain route within an AS to be of higher degree of preference than other routes.

For the interest of this thesis, we mainly focus on the AS-PATH attribute and ignore all the other attributes.

Although BGP carries all the AS paths that lead a router to another, a "routes selection process" is run at each router to make sure that the shortest AS paths are the only ones stored in the RIBs. The details of that routes selection process is beyond the scope of this thesis and can found in [BA03].

It is essential to mention that BGP-speakers within a certain AS exchange routing information (that they learn from other ASes) among each other. The latter mode of operation is referred to as the Internal BGP (I-BGP) mode. On the other hand, when BGP-speakers of different ASes exchange routing information, we call this mode of operation the External BGP (E-BGP) [TH01]. With I-BGP, each edge node peers with all the other edge nodes through "logical connections". That is, edge nodes do not need to be connected directly for I-BGP peering to take place. In other words, the logical connection may span some other internal nodes (which are non BGP-speakers) to set up the I-BGP peering. These logical connections are configured manually by the AS administrator. The latter fact makes miss-configuration of I-BGP likely to appear. This causes data packets to be lost or sent back and forth (a situation that is called "oscillation") without reaching the desired destination node (see [GR02], [BA02], and [MA02a]).

In this thesis, we aim at deploying BGP in optical networks to achieve inter-domain routing among ASes. Extending BGP to handle that role is preferred over designing an inter-domain routing protocol, specific to optical networks, from scratch. BGP has been deployed

extensively in industry and there is a full understanding of its benefits and drawbacks. Moreover, significant research efforts continue to improve the performance of BGP. We show in Chapter 3 how we benefit from BGP to achieve an efficient inter-domain routing protocol for optical networks. In the following section, however, we show the main facts that should be taken into consideration when migrating routing protocols designed for IP-based networks to optical networks.

2.4 Routing in WDM optical networks vs. routing in IP-based electronic networks

The authors in [BA00] and [ZE04] provide a detailed comparison between routing in IP-based electronic networks and WDM optical networks. They identify the following major differences between the two domains:

a. IP-based networks are connectionless networks in which both control and data planes functionalities are involved in the routing procedure. From one side, control plane distributes topology information throughout the network and uses that information to construct a forwarding table. On the other side, data plane handles the actual forwarding of IP packets. In optical networks, however, we are dealing with connection-oriented circuit-switching networks in which no involvement of the data plane is required at all. In other words, end-to-end connections are provisioned between an S-node and a D-node based on the network topology. Once a connection is provisioned, data is transferred over it without any intervention of the routing engine.

b. In IP-based networks, routing protocols are affected with the decisions taken by the data plane to forward packets to their destinations. Hence, the failure of routing protocols

adversely impacts the services offered to end users. Due to the separation between control and data planes, routing protocol failures in optical networks do not adversely impact existing connections.

c. Since a connection has to be established and appropriate resources (e.g., wavelengths) have to be reserved in advance of data transfer, routing in optical networks requires knowledge of the availability of different resources in the network.

d. In optical networks, the S-node is responsible for the computation of the entire path from source to destination. Thus, the information maintained by the S-node should be as much accurate as possible so that the best path can be computed to the D-node. On the other hand, in IP-based networks, routing is accomplished on a hop-by-hop basis. This means that each node along the path to the D-node decides its next hop independently.

Routing protocols have been extensively researched and deployed in IP-based networks. Therefore, there is a considerable expertise to design routing protocols for optical networks. In fact, different standard bodies like IETF, OIF and ITU-T have accomplished progress towards achieving optical intra-domain routing using OSPF and IS-IS (see [BA00] for comprehensive details). In this thesis, we focus on extending BGP to efficiently achieve inter-domain routing in the optical domain.

2.5 State-of-the-art proposals for inter-domain routing in optical networks

In this section we review the proposals introduced in the literature to address the inter-domain routing problem in optical networks.

2.5.1 Optical BGP (OBGP)

Optical BGP (OBGP) protocol has been introduced in [FR02]. OBGP proposes extensions to BGP to be applicable in optical networks. With OBGP, BGP handles the setup and control of lightpaths (the tasks of a signaling protocol) in addition to its routing functionality. This means that the end user is supported with a complete control methodology, to provision a lightpath through multiple ASes, that requires no additional communication channel for signaling messages. The support of signaling and routing in a single protocol is motivated by the useful path attributes supported by BGP that can leverage important information needed by the signaling protocol. For example, the AS_PATH attribute informs each BGP-speaker of the end-to-end path needed to for an S-D pair to communicate. With this attribute one can restrict the ability of reading certain BGP messages to those routers on the end-to-end path and reduce the amount of signaling traffic in the network. The result is that better distribution of BGP messages is achieved in the network. The design of OBGP makes sure that any intra-domain routing protocol can still be used within the available ASes. The basic design requirements of OBGP are the following:

a. A database at each OBGP-speaker to store the wavelength availability information, which is needed to know the available resources to successfully setup a lightpath.
b. The ability to communicate lightpath reservation requests and responses among OBGP-speaking devices.
c. The ability to propagate up-to-date information about the status of resources in the network.

OBGP introduces a new message type, called the OBGP message, to BGP. The purpose of adding this new message is to simplify the modifications to BGP and to isolate the changes

required to support lightpath establishment without affecting the usual operation of the other messages. OBGP uses the information provided by the AS_Path to establish a lightpath between two nodes. OBGP achieves the lightpath establishment by operating in either of two modes, namely, Two-phase setup mode and Four-phase setup mode. With the two-phase setup mode, an OBGP-speaker firstly discovers the available wavelengths on the link that connects it to the next OBGP-speaker as defined by the AS_Path (Discovery phase). Once an available wavelength is detected, the lightpath is directly setup with that next OBGP-speaker (Setup phase). The process is repeated until the destination node is reached. Since wavelength-conversion is not supported, in case many wavelengths are found available on a certain link then many setup failures will occur due to competition scenarios. With the four-phase setup mode, all resources are firstly reserved before the actual setup of the lightpath. Thus, discovering at least one available wavelength along the AS path (Discovery phase), that wavelength is directly reserved (Reservation phase), the lightpath is setup (Setup phase) and finally a confirmation message is sent from the D-node to the S-node to confirm the successful establishment of the lightpath (Confirmation phase). Figures 2.3 and 2.4 show the sequence diagrams of both modes of operation of OBGP protocol.

As discussed in [KH04], OBGP suffers from the following limitations:

a. Introducing a new fifth message in BGP may add unneeded complexity to BGP. As we show in our proposal in Chapter 3, the UPDATE message in BGP can effectively handle the functionality given to that new message.
b. OBGP gives signaling responsibilities to BGP. Depending on BGP to handle signaling tasks is risky. This is because I-BGP peering is configured manually and we may have situations of oscillation in the network, as discussed in Section 2.3.2.

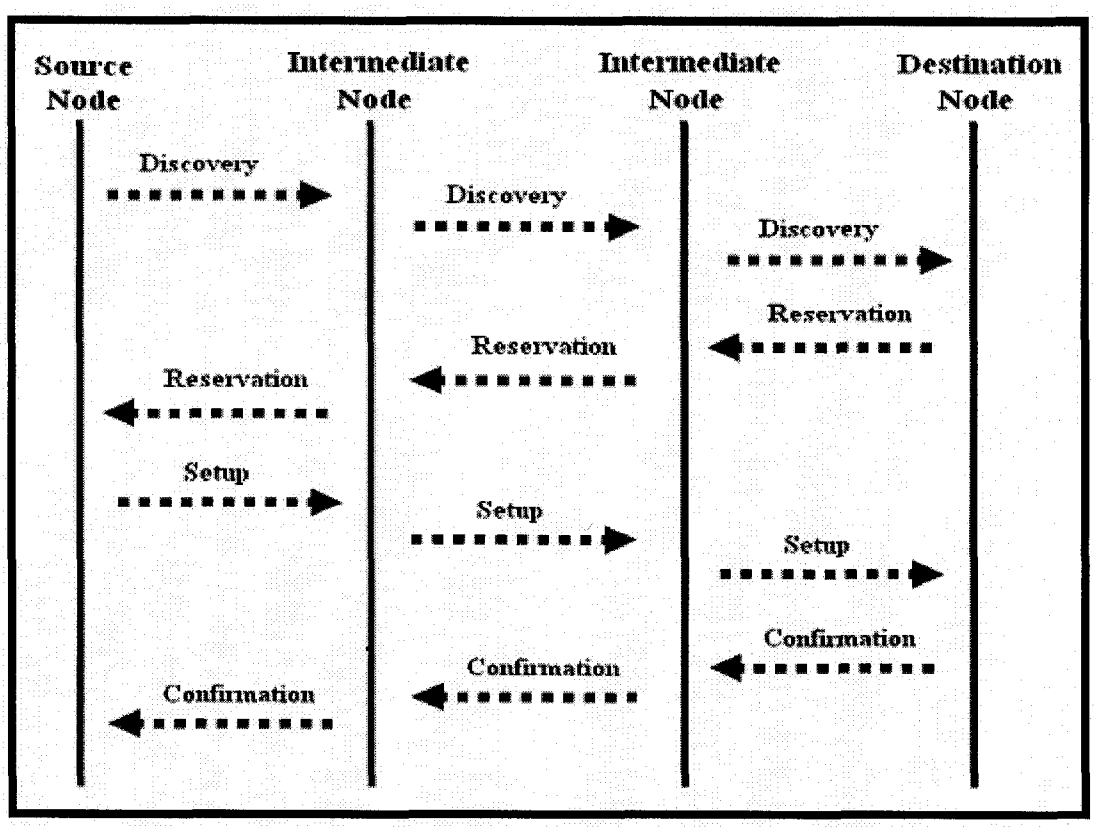

Figure 2.3 Four-Phase lightpath establishment in OBGP.

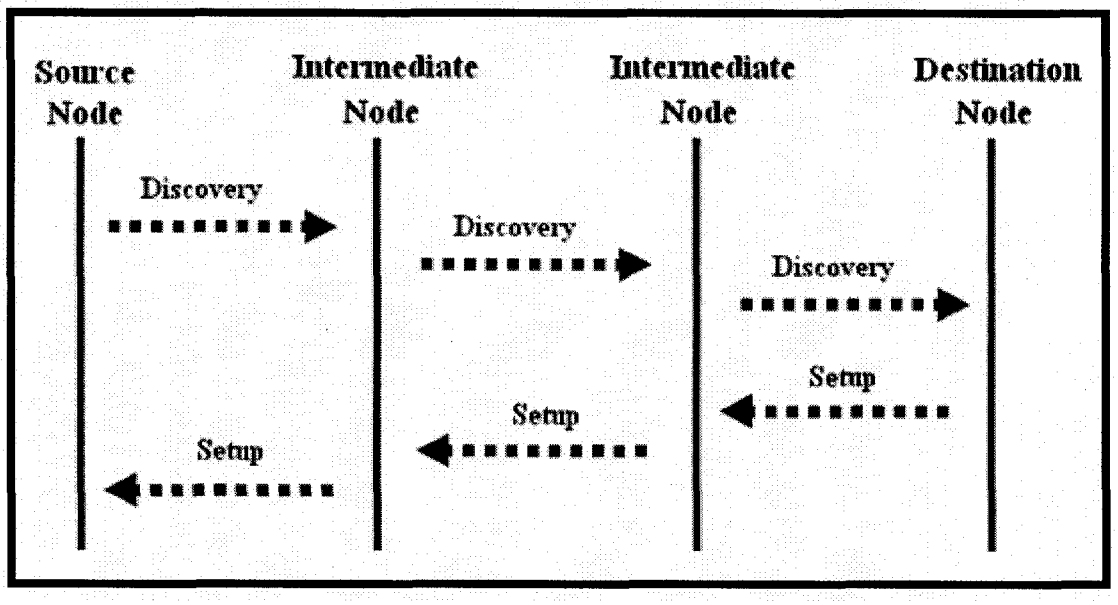

Figure 2.4 Two-Phase lightpath establishment in OBGP.

2.5.2 Optical BGP (OBGP): Inter-AS lightpath provisioning

In [BL01], the authors assume that the available ASes own their wavelengths and their OXCs. With this ownership, ASes can grant their customers virtual control over the optical resources of the ASes. This means that the customers can have the privilege of establishing and tearing down lightpaths based on their needs.

The protocol undergoes two phases to provision a lightpath from an S-node to a D-node (over multiple ASes):

a. **The lightpath reachability phase**: In this phase, each AS uses BGP to advertise information about the availability wavelengths on the direct links connected to it. The information is encoded using BGP multi-protocol extensions (BGP-MP) and extended community attribute (see [BA00] and [CH96b] for details about BGP-MP and the extended community attribute).

b. **The lightpath establishment phase**: The advertisements sent in phase 1 allow each AS in the network to build a database or "lightpath RIB" that is used to discover the available paths to reach any destination AS. In phase 2, a source AS uses its lightpath RIB to communicate with a certain destination AS and then sends a BGP update message to inform all the other ASes of the change in the wavelength availability in the network.

The main difference between [BL01] and [FR02] is that the wavelength availability information will be carried in BGP's UPDATE messages by benefiting from BGP multi-protocol extensions (BGP-MP). BGP-MP extensions allow BGP to include the necessary optical and routing information in the UPDATE Message. The latter information include the IP address of the source node, the IP addresses of the destination nodes, and a lightpath

identifier (which is needed during the reachability and establishment phases to know whether certain wavelengths are available or not).

In [JE02], the authors propose some modifications to OBGP explained above. Here, each AS incorporates a unit that combines an IP router and an OXC. The latter unit is referred to as a combined unit. The OXC in each combined unit is allowed to share its optical resources with neighboring ASes. For example, in Figure 2.5, router R2 in AS2 can use the optical resources of Unit1 in AS1. This means that OBGP now assumes a relationship of trust among AS1 and AS2. As a result, an AS can grant other ASes a virtual control over its combined unit(s) to provision a lightpath. Furthermore, each AS treats its combined unit as being composed of a normal BGP-speaker (the IP router) and a virtual BGP-speaker (the OXC). This means that the IP router advertises itself independently of the OXC. Then, each OXC is capable of advertising wavelengths availability information to all edge routers (within its AS and the other ASes). Moreover, both the normal and virtual BGP-speakers are treated as two separate ASes. The latter means that both the virtual and normal BGP-speakers will be advertising their information without the need to use I-BGP mode of operation (discussed in Section 2.3.2).

Both of the protocols in [BL01] and [JE02] suffer from the following drawbacks (see [KH04]):

a. They assume a complete trust relationship. That is, each AS allows its customers to have complete control over its nodes.
b. It does not clearly specify the type of intra-domain routing information to be advertised.

c. It depends on BGP UPDATE messages to handle the task of establishing lightpaths. In other words, UPDATE messages will play a signaling role and this should be avoided as discussed earlier.

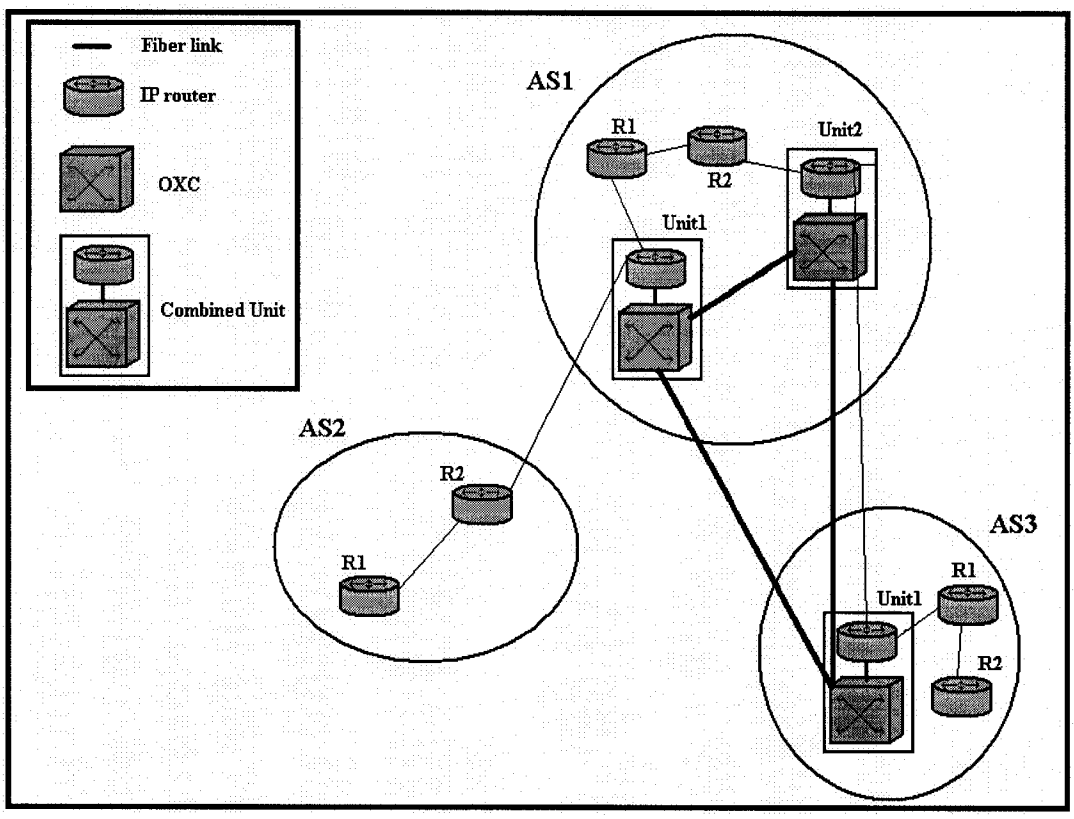

Figure 2.5 An example of a network operating under OBGP and using the concept of combined units.

Furthermore, the protocol in [JE02] suffers from the problem of requiring each AS to have a relatively large list of all the combined units in the network. This problem becomes worse when the ASes deploy more than one combined unit as the number of ASes will get bigger and bigger. Besides, having more ASes within an AS increases the complexity of coordinating and advertising resources within that AS.

2.5.3 Optical Routing BGP (ORBGP)

In [KH04] a routing protocol named Optical Routing BGP (ORBGP) is proposed. With this protocol, each node maintains a local routing table that contains information about all the available wavelengths to all destination nodes in the global network. More specifically, each ORBGP-speaker, which is basically located at the edge of an AS, has two types of information:

a. Information about the wavelengths available in its own AS. This information results from the interaction between intra-domain routing (using OSPF) and inter-domain routing (using ORBGP).
b. Information about the wavelengths available in neighboring ASes. This is attained by communication between an ORBGP-speaker and its ORBGP-peer located at the edge of a neighboring AS.

Depending on the information above, each ORBGP-speaker will be capable of computing paths to reach any destination node provided that wavelength-continuity is achieved. Those computed paths are then stored and advertised according to BGP mechanisms. It is important to note that the interaction between the ORBGP-peers does not reveal any information specific to each AS except two things:

a. The wavelength(s) that is (are) available to communicate with a certain destination node.
b. The edge node or ORBGP-speaker that is used as the next-hop on the path to the destination node.

This means that no information about the internal topology of an AS is made known or public to ORBGP-speakers that are contained in other ASes.

ORBGP takes care of the fact that the continuous changing nature of the availability of wavelengths will directly affect the scalability of BGP. To solve that, two advertisement schemes were introduced to overcome that limitation:

a. Advertise changes after fixed time periods: In this scheme, each ORBGP-speaker waits for a certain time period before informing its neighbors of the changes it detected.
b. Advertise changes after certain percentage changes: In this scheme, a counter is used at each ORBGP-speaker to count the number of wavelengths that had change in their status. By using different thresholds for that counter, the frequency of announcing changes can be controlled.

The main weakness in this protocol is that each ORBGP-speaker needs to store information about the availability of wavelengths on both intra-domain and inter-domain links. It is expected that the total number of links in the global network to be very big and thus updating all the ORBGP-speakers of a change happening in only one link within a certain AS will be affecting the scalability of ORBGP significantly.

2.5.4 BGP/GMPLS based inter-domain routing in optical networks

In [XU03], the authors solve the inter-domain routing problem in optical networks based on a BGP/GMPLS proposal (for more information about GMPLS see [PA06] and [KO02]). Basically, the BGP/GMPLS proposal provides a complete specification for the two components of the control plane as follows:

a. Routing component: handled by BGP with some extensions to support optical networks.
b. Signaling component: handled by GMPLS with some extensions to it to support the inter-domain problem.

We do not discuss the GMPLS component as it is beyond the scope of this thesis.

With the BGP/GMPLS proposal, the points at which a client network (like an IP network) is attached to a provider network (an optical network) are called Client Access Points (CAPs). A group of CAPs that share the same physical and/or logical attributes that are critical to set up a lightpath is called a Client Access Group (CAG). Therefore, BGP announces or disseminates only CAG route information through the network.

To support these CAGs, the BGP/GMPLS proposal introduces five new path attributes. Also, other extensions are made to make BGP capable of leaking abstracted topology information (according to Service Level Agreements (SLAs)) between the client network and the provider network. This topology information allows the client network to specify certain routing constraints for the lightpaths it establishes. These routing constraints control the selection of routes for any incoming call request. All of these extensions to BGP can achieve a strong support to traffic engineering (TE). However, the main drawback in the BGP/GMPLS proposal is that it does not address the problem of continuously reporting the status of the wavelengths to the routers in the network. This is a critical issue that significantly affects the scalability of BGP and should be addressed properly.

2.6 Summary

In this chapter, we have provided an overview of intra-domain and inter-domain routing protocols in IP-based networks. We have highlighted the main differences between routing in IP-based networks and optical networks. Special attention has been given to BGP as the de facto inter-domain routing protocol in IP-based networks, and we have specified it as the strong candidate to be migrated to optical networks. Finally, the chapter has presented a review of the optical inter-domain routing protocols proposed in the literature.

Chapter 3

The Extended BGP for Optical Networks

3.1 Introduction

In this chapter we provide a full description of a new protocol, called the eXtended BGP for Optical Networks (xBON), which achieves inter-domain routing in optical networks. The chapter is organized as follows. Section 3.2 introduces a high–level design of xBON. Section 3.3 qualitatively compares xBON to other optical inter-domain routing protocols. Section 3.4 explains the modeling of xBON in general and the reasons behind choosing OPNET Modeler for our system modeling. Section 3.5 describes in details our xBON model in OPNET Modeler. Finally, Section 3.6 provides a summary of this chapter.

3.2 xBON Design

As mentioned earlier, the main motivation behind developing BGP is the inability of intra-domain routing protocols to scale well as the size of the communication network grows. This is a key point that should be kept in mind when introducing any extensions or modifications to BGP (to be deployable in optical networks). We also mentioned that an advanced stage has been reached in designing optical intra-domain routing protocols. Therefore, we assume that the intra-domain routing problem is already handled and concentrate mainly on the inter-domain routing problem.

At the high-level, we view each AS in the network as a single node (or BGP-speaker). To illustrate this refer to Figure 3.1 and Figure 3.2. In Figure 3.1 we show a traditional view of

an optical network (please note that in the rest of this thesis, all of the networks drawn are assumed to be optical networks) that is segregated into three ASes. Each of these ASes uses a unique intra-domain routing protocol (IS-IS in AS1, RIP in AS2, and OSPF in AS3). BGP is used between the ASes to advertise AS paths. In Figure 3.2 we redraw the network in Figure 3.1 without the internal details of each AS. Therefore, the ASes are treated as single nodes and this helps in concentrating on BGP and the inter-domain links.

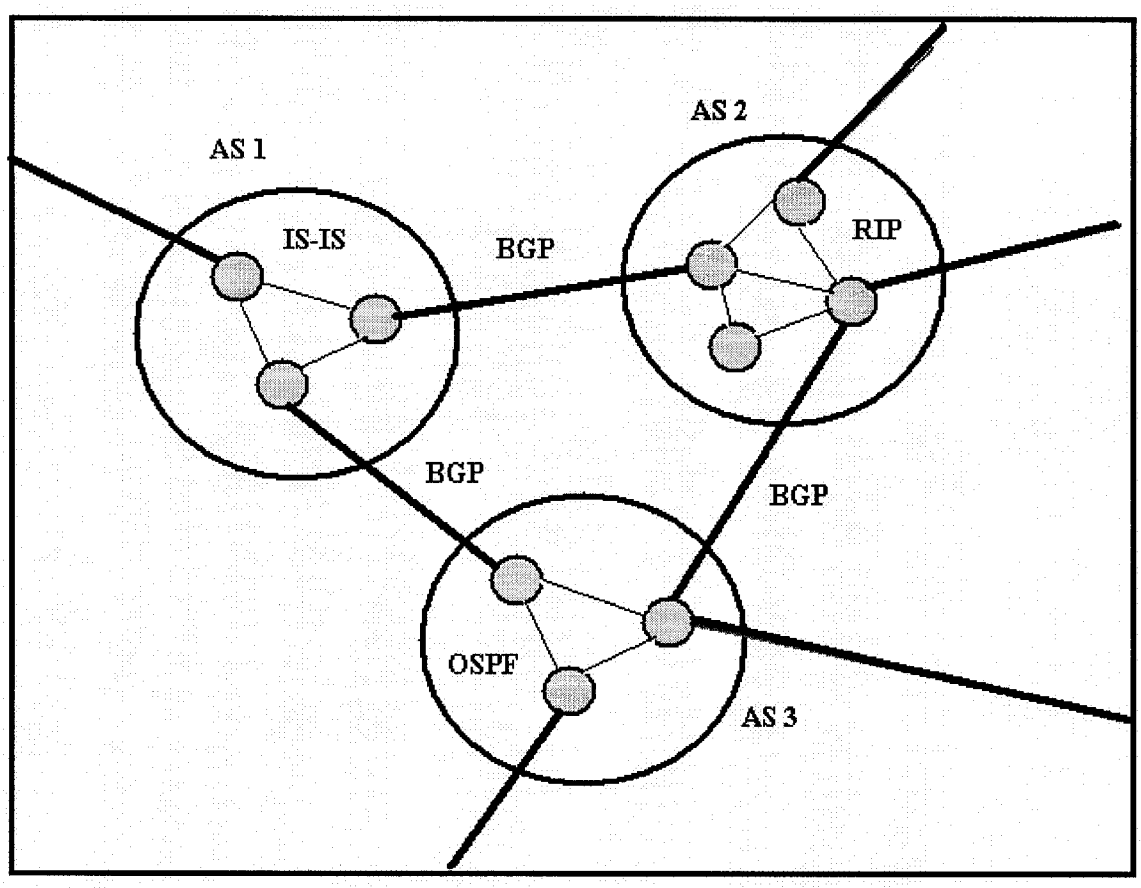

Figure 3.1 A network consisting of 3 ASes. The thin links are intra-domain links while the thick links are inter-domain ones. Each AS is using a certain intra-domain routing protocol (OSPF, IS-IS, and RIP) while BGP is used between the ASes.

In the rest of this thesis we use the term "link" and "inter-domain link" interchangeably. We also assume that all of the links in the network have the same capacity (in terms of the maximum number of wavelengths carried on each link). Furthermore, we assume that each

node has a local database that contains information about the available and used wavelengths on all the links of the network. We call this database the **W**avelength **A**vailability **D**ata**B**ase (WADB). The WADB of a node is updated continuously once a change in a wavelength status is detected by the node itself or upon receiving an UPDATE message from a neighboring node.

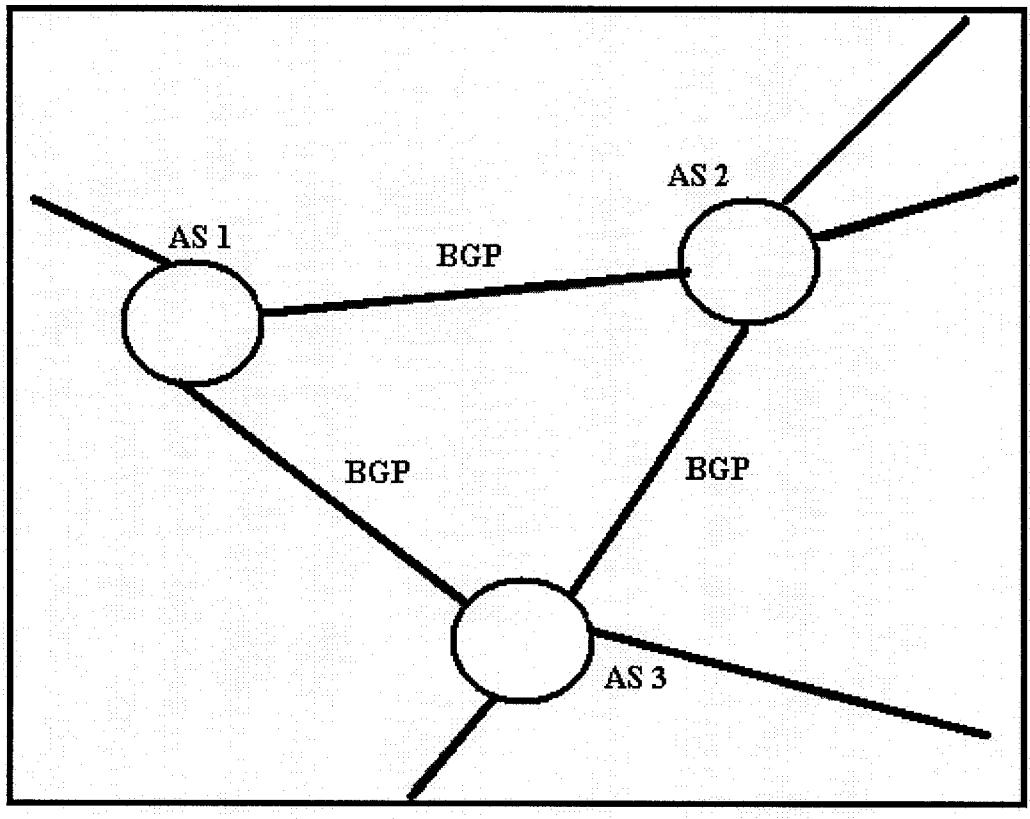

Figure 3.2 The network in Figure 3.1 is re-drawn without the internal details of each AS. Thus, each AS is now treated as a single node or BGP-speaker.

With the standard BGP, each node sends an UPDATE message to inform its neighbors about any new AS path it learns of. As mentioned in Chapter 2, these AS paths are well-described to the receiving nodes through the use of path attributes. We introduce a new path attribute type that is used to advise the nodes of the wavelengths, on the inter-domain links, which encountered a change in their status (whether they have become available or used). We call

this new path attribute the WAVELENGTH_STATUS attribute. The value of this attribute is set to the number corresponding to the index of the wavelength changed. For example, if the links in the network have a capacity of 8 wavelengths, then the wavelengths are numbered as follows: λ_0, λ_1, λ_2 ... λ_7. Thus, the value of the WAVELENGTH_STATUS attribute can be 0, 1, 2, ..., or 7. Once a node receives an UPDATE message, it examines both the AS_PATH attribute and the WAVELENGTH_STATUS attribute. From the AS_PATH, the node can identify the links that are affected by the change, and from the WAVELENGTH_STATUS attribute the node can identify the wavelength that had its status changed. As a result, the node updates its WADB to include the new status of the wavelength. Consider Figure 3.3 and Table 3.1 for an illustration. In Figure 3.3 we give an example of an optical network that is composed of 6 ASes. All the links in this network are assumed to have a capacity of 6 wavelengths. In Table 3.1 we show the WADB of AS3. The first column of the WADB lists all the links of the network. This list is populated using the AS_PATH attribute carried by the UPDATE message. In other words, the AS_PATH informs the node of the paths it can use to communicate with all the other nodes in the network and this helps in identifying all the links in the network. The rest of the columns in the WADB contain information about the status of the wavelengths on the links specified in the first column. We use a "0" to note that a wavelength is already used by a lightpath a "1" to note that the wavelength is available for use. Thus, for the link AS3-AS1, the WADB specifies that only λ_0, λ_4, and λ_5 are available for use while for the link AS3-AS5 we have all the wavelengths available for use. We call this new version of BGP that carries the WAVELENGTH_STATUS attribute the e**X**tended **B**GP for **O**ptical **N**etworks (xBON).

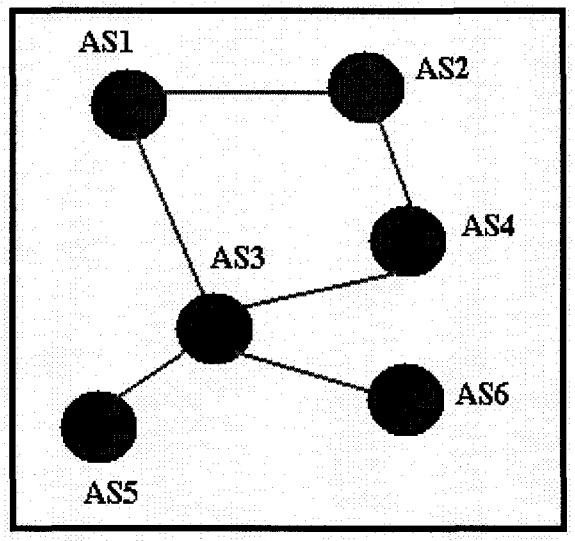

Figure 3.3 An example of an optical network.

Thus, the main task of xBON, in addition to handling the primary functionality of BGP, is to advertise information about the wavelength availability on the inter-domain links. xBON incorporates the new WAVELENGTH_STATUS attribute into BGP's UPDATE message. The latter attribute informs the nodes of the status of the wavelengths on the inter-domain links. xBON operates in three different modes:

Link	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5
AS1-AS2	0	1	1	0	1	0
AS1-AS3	0	0	1	0	0	1
AS2-AS4	0	0	0	0	0	0
AS3-AS4	0	0	1	0	1	1
AS3-AS5	1	1	1	1	1	1
AS3-AS6	0	1	1	1	1	1

Table 3.1 The WADB of AS3.

a. Simple xBON Scheme (SxS): in this scheme, xBON operates exactly as BGP except that an UPDATE message is not only sent upon learning new AS paths, but also upon detecting a change in the availability of any wavelength on the inter-domain links.
b. Wavelength Threshold Scheme (WTS): with this scheme, UPDATE messages are sent once a certain percentage of the wavelengths (that is, a wavelength threshold), on a certain link, undergoes a change in the status. For example, if the capacity of the links is 10 wavelengths and we set the wavelength threshold to 20%, then an UPDATE message will be sent whenever 2 wavelengths are changed in status.
c. Time Threshold Scheme (TTS): with this scheme, UPDATE messages are sent once a certain time period (that is, a time threshold) passes. For example, if we set the time threshold to 10 ms, then an UPDATE message will be sent by every node in the network every 10 ms (whether there was a change in wavelengths' status or not).

It is essential to mention that xBON is a routing protocol and we still need a signaling protocol for the process of establishing a lightpath to be complete. In this thesis we focus on the routing component of the control plane and we do not specify a certain signaling protocol to handle the messaging between the different nodes. However, the main tasks that should be handled by that signaling protocol are the following:

a. Confirming that the same wavelength is available on all the links, along the AS path, before establishing the lightpath. The latter condition is referred to as the wavelength-continuity constraint.
b. Establishing the lightpath between the S-D pair of nodes.
c. Tearing down the lightpath.
d. Informing the S-node that the requested wavelength is not available for use.

Therefore, in our description of the lightpath establishment process, we assume that the four above tasks are accomplished through the following messages respectively: CONFIRM, ESTABLISH, TEARDOWN, and REJECT.

To illustrate how xBON handles the routing part in the process of establishing lightpaths, we describe the steps followed under the Simple xBON Scheme (SxS) to provision a connection between AS4 and AS5 in Figure 3.3. We will highlight the stages where the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS) behave different than the Simple xBON Scheme (SxS). Assume that the WADBs of AS4 and AS5 are as shown in Tables 3.2 and 3.3 respectively. Note that although the WADBs of all the nodes list all the links of the network, the nodes have different views of the status of the wavelengths. For example, AS3 views the link AS4-AS2 as having all of its wavelengths available except λ_0. On the other hand, AS4 views the same link as having all of its wavelengths available. These different views are resulting from the fact that it takes the UPDATE message some time delay to reach all the nodes. In other words, if AS4 detected that λ_0 is released, it will update its WADB accordingly. However, it will take some time, equal to the propagation delay on the link AS4-AS3, for the UPDATE message to reach AS3 and inform it of this new status of λ_0. This is an important fact that contributes mainly to the blocking in the network as will be discussed later in this chapter.

Once AS4 receives a call request to communicate with AS5, it directly retrieves the AS path to reach AS5 which is AS3-AS5 (recall that with BGP the shortest paths are kept in the RIBs of the BGP-speakers). After that, AS4 consults its WADB to identify the wavelengths available for use on the first link of the AS path (that is, link AS4-AS3).

Link	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5
AS1-AS2	0	1	1	1	1	0
AS1-AS3	0	0	1	0	0	1
AS2-AS4	1	0	0	0	0	1
AS3-AS4	0	0	1	0	1	1
AS3-AS5	1	1	1	1	1	1
AS3-AS6	0	1	1	1	1	0

Table 3.2 The WADB of AS4.

From Table 3.2, we can see that λ_2, λ_4, and λ_5 are available for use. AS4 then applies the First-Fit (FF) wavelength assignment (WA) scheme to choose the wavelength over which the lightpath will be setup. According to the FF scheme, the available wavelengths should be ordered according to their indexes in an ascending manner and the wavelength with the least index will be chosen. In our case, AS4 chooses λ_2 and marks it as unavailable, by showing the value "0" instead of "1" for λ_2 in the WADB. It is important to mark λ_2 as unavailable, even though the establishment of the lightpath is not started yet, to prevent any new arriving call requests from choosing it while AS4 is still processing the older call request. After selecting λ_2, AS4 sends a CONFIRM message to inform AS3 that it needs to use λ_2 to establish a lightpath to AS5. AS3 consults its RIB to find the AS path to AS5 and then checks its WADB to see if λ_2 is available on the links AS3-AS4 and AS3-AS5. As shown in Table 3.1, λ_2 is available on both links. Next, AS3 marks λ_2 as unavailable and sends a CONFIRM message to AS5 to inform it that it needs to establish a lightpath using λ_2 on the

link AS3-AS5. AS5 will repeat the process of checking the RIB and the WADB to find that λ_2 is available for use.

Link	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5
AS1-AS2	1	1	1	1	1	0
AS1-AS3	0	0	1	0	0	1
AS2-AS4	1	0	1	0	0	1
AS3-AS4	0	0	1	0	1	1
AS3-AS5	1	1	1	1	1	1
AS3-AS6	0	0	1	1	1	0

Table 3.3 The WADB of AS5.

At that moment, AS5 marks λ_2 as unavailable and sends AS3 an ESTABLISH message to show its approval of the CONFIRM message, and AS3 directly sends another ESTABLISH message to AS4 which starts the setup of the lightpath. The nodes AS1, AS2, and AS6, which are not involved in this lightpath setup, should be informed of λ_2 status change. Therefore, UPDATE messages with the WAVELENGTH_STATUS attributes set to the value of 2 are sent (in the case of the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS), UPDATE messages are sent once the appropriate thresholds are crossed) by AS4, and AS3 to their neighbors (AS5 has only AS3 as a neighbor and thus it does not send an UPDATE message).

In case that λ_2 was seen as unavailable by either AS3 or AS5, a REJECT message should be sent back to AS4 which, in turn rejects the call request.

Once AS4 needs to teardown the lightpath connecting it to AS5, it sends a TEARDOWN message to AS3 and marks λ_2 as available. Then, AS3 sends a TEARDOWN message to AS5 and releases λ_2, while AS5 releases λ_2 upon receiving the teardown message. Finally, UPDATE messages are sent again (in the case of the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS), UPDATE messages are sent once the appropriate thresholds are crossed) to inform the rest of the nodes of λ_2 being released.

By sending UPDATE messages upon having a change in the status of a certain wavelength, the Simple xBON Scheme (SxS) keeps the WADBs as accurate as possible. However, the Simple xBON Scheme (SxS) suffers from a main drawback. Given the continuous changing behavior in the wavelengths' status, it is not practical to keep sending UPDATE messages to the nodes whenever a change occurs. This behavior harms the scalability of BGP significantly. On the other hand, the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS) achieve better support for the scalability of BGP. This support, however, sacrifices the accuracy of information stored in the WADBs. We show in Chapter 4 how the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS) can provide a balance between the scalability of BGP and the accuracy of information in the WADBs so that we achieve a well-performing routing protocol for optical networks.

Figures 3.4, 3.5, and 3.6 show the flow charts of the Simple xBON Scheme (SxS), the Wavelength Threshold Scheme (WTS), and the Time Threshold Scheme (TTS), respectively.

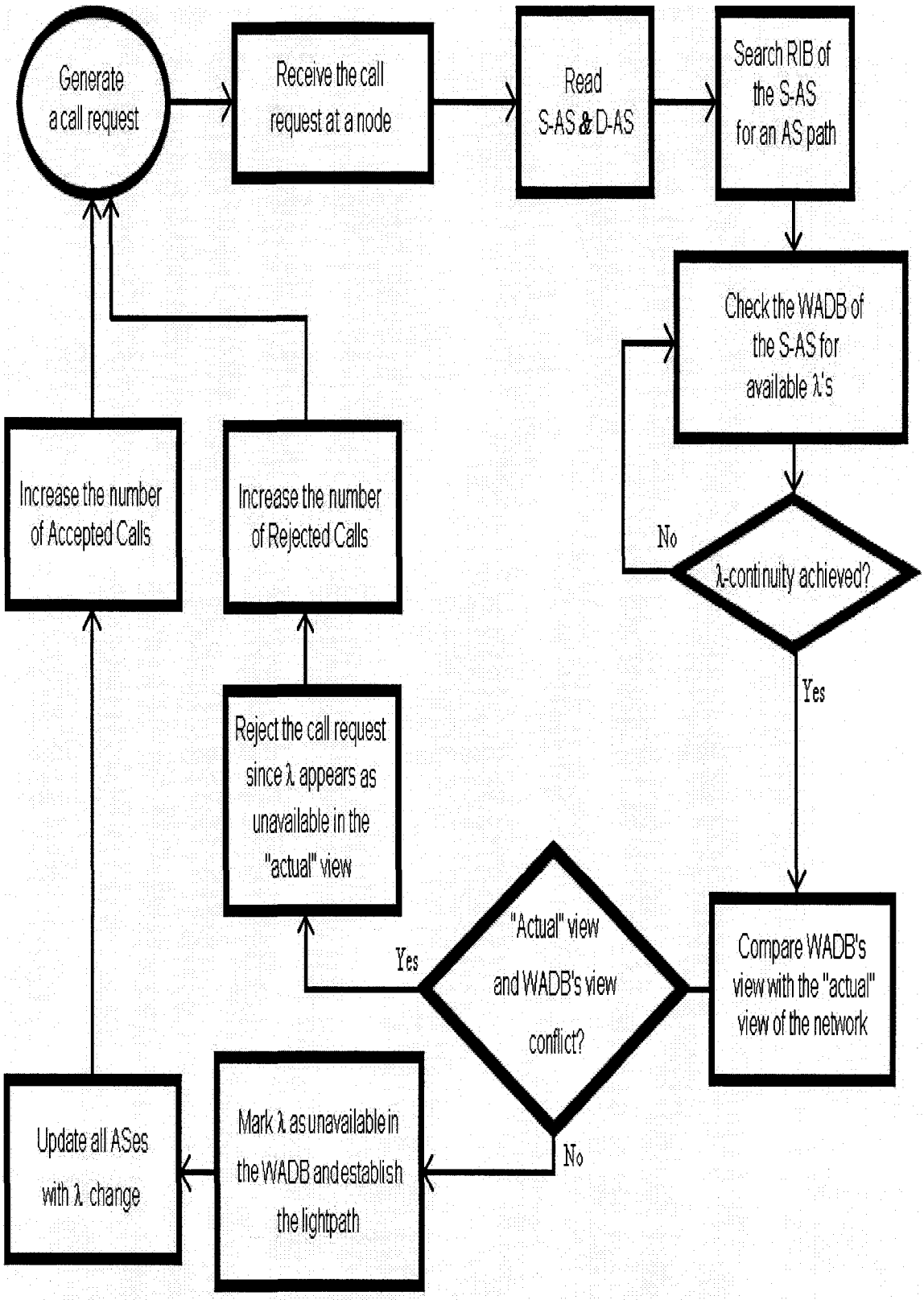

Figure 3.4 Flow chart of the Simple xBON Scheme (SxS). In this chart, S-AS refers to the source node, D-AS refers to the destination node, and "λ"refers to a wavelength.

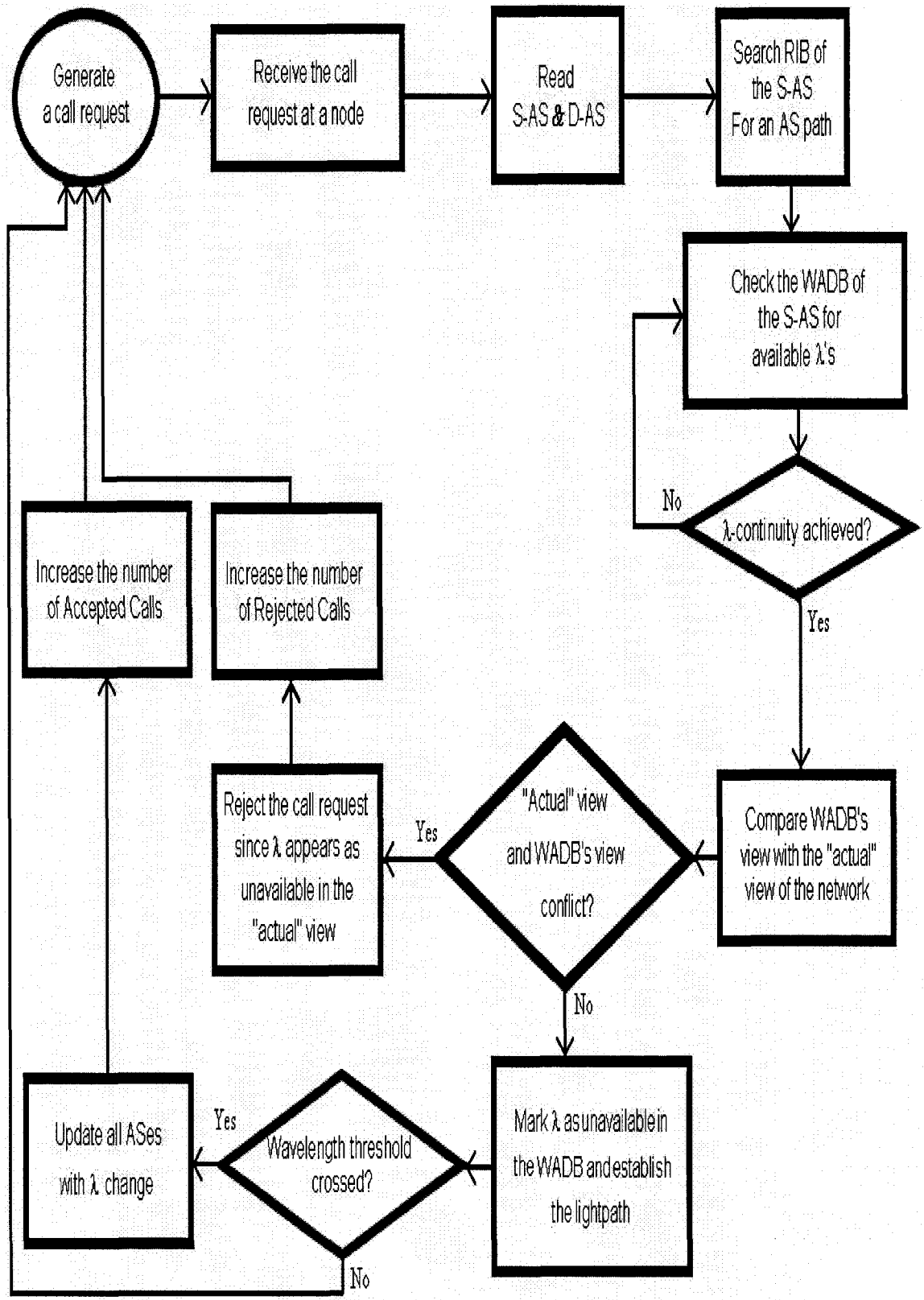

Figure 3.5 Flow chart of the Wavelength Threshold Scheme (WTS). In this chart, S-AS refers to the source node, D-AS refers to the destination node, and "λ"refers to a wavelength.

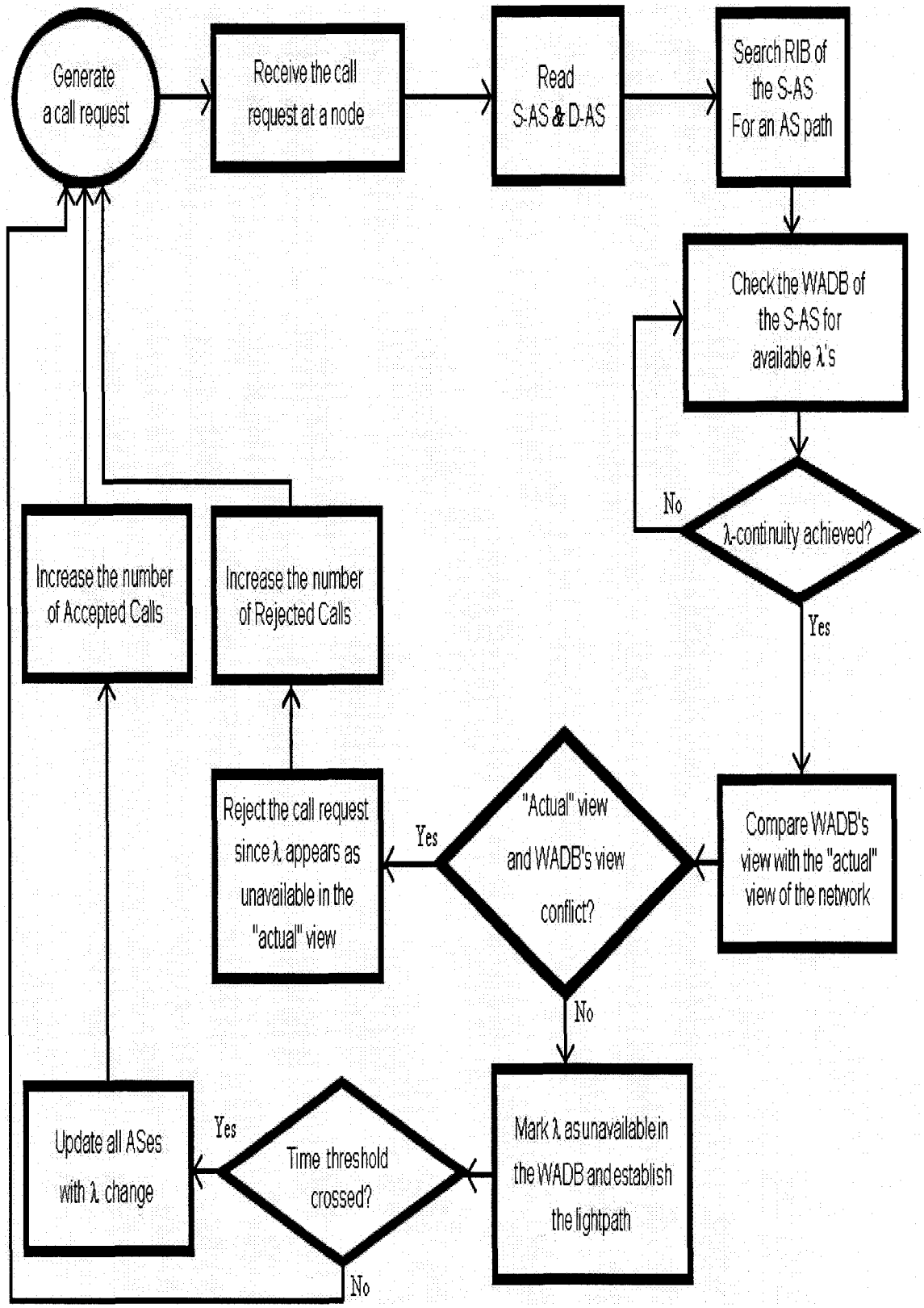

Figure 3.6 Flow chart of the Time Threshold Scheme (TTS). In this chart, S-AS refers to the source node, D-AS refers to the destination node, and "λ"refers to a wavelength.

3.3 Comparison between xBON and other optical inter-domain routing protocols

In this section we provide a comparison between xBON and the routing protocols that we have reviewed in Chapter 2.

As discussed in Section 2.5.1, OBGP in [FR02] modifies BGP by introducing a new signaling message to it, however, with some problems associated with it. With xBON, we overcome the complications associated with OBGP's modifications by exploiting the capabilities of BGP rather than modifying them. In other words, BGP is a routing protocol in its nature and modifying that by assigning signaling responsibilities to it is not recommended as we have discussed earlier. Thus, xBON depends on using existing signaling protocols, such as CR-LDP and RSVP-TE to complete the lightpath provisioning operation. On the other hand, xBON exploits the capabilities of BGP's UPDATE message by introducing a new path attribute type. Path attributes are used to describe certain features associated with the AS path being advertised. Thus, introducing a new path attribute type to an already deployed BGP message is more convenient and practical than introducing a new message that may result in unexpected complications to the functionality of BGP.

In Section 2.5.2 we have discussed the drawbacks associated with the proposals in [BL01] and [JE02]. Basically, both of these protocols assume a complete trust relationship between ASes. Moreover, they do not specify the type of intra-domain routing information to be advertised, and they depend on BGP for signaling. The latter problem is not an issue with xBON as we have discussed in the previous section. On the other hand, the only modification that xBON makes to BGP is the advertisement of the information related to the

inter-domain links. Thus, xBON does not suffer from the problems associated with assuming trust relationships between ASes and leaking intra-domain routing information.

We have also shown that the protocol in [JE02] further suffers from deploying combined units and requiring a large list of all the combined units in a network. The burden of using such large lists and requiring complicated coordination among the ASes is completely avoided in xBON. The only overhead xBON adds to each AS is the need to maintain a wavelength availability database (WADB). The process of populating and updating each WADB is very simple as it depends on the information embedded in the UPDATE message.

In Section 2.5.3 we have mentioned that the main drawback of ORBGP in [KH04] is that it requires each ORBGP-speaker to maintain information about the availability of wavelengths on both intra-domain and inter-domain links. As we have highlighted in Section 2.5.3, the latter behavior of ORBGP is expected to negatively affect the scalability of BGP. However, xBON supports the scalability of BGP by advertising the availability of wavelengths only on the inter-domain links. Usually, the number of intra-domain links is much more than the number of inter-domain links. For instance, two neighboring ASes may be interconnected by one inter-domain link while each of them may contain large networks composed of tens of intra-domain links. Therefore, xBON achieves more support to scalability than the case is with ORBGP.

In Section 2.5.4 we have explained that the BGP/GMPLS proposal in [XU03] does not address the problem of continuously reporting the status of the wavelengths to the routers in the network. The latter is a critical issue as it affects the scalability of BGP. With xBON, we have proposed the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS) to support the scalability of BGP.

It has to be noted that we do not provide a quantitative comparison between xBON and all of the protocols discussed above due to many reasons. Firstly, OBGP protocol, mentioned in Sections 3.2.1 and 3.2.2, is both a routing and a signaling protocol while xBON is a pure routing protocol. Thus, we have no common ground to quantitatively compare the performance of xBON with the performance of OBGP. Secondly, no quantitative study is available for the BGP/GMPLS proposal in [XU03] itself. Finally, we have highlighted that the main drawback of ORBGP is that it affects the scalability of BGP. Thus, in order to compare xBON to ORBGP in terms of scalability, we need to use significantly large networks in our simulations. However, as we mention in Chapter 4, with our limited experience with the simulation tool we have used (that is, OPNET Modeler), we faced many problems in our simulations as we increase the size of the network. For example, we have experienced tedious debugging efforts to solve many problems (especially memory overflows).

3.4 xBON Modeling

By careful consideration of the flowcharts shown in Figures 3.4, 3.5, and 3.6, we can see that modeling xBON requires the development of a system that achieves the following tasks:

a. Packet or call request generation.

b. Database management (that is, writing data to and reading data from databases).

c. Statistics collection.

Basically, tasks a and c are not specific to xBON and generally needed in simulating the performance of any communication protocol. Thus, task b is the one that distinguishes the performance of xBON and our modeling efforts should concentrate on this task. Towards that end, it is quite clear that tasks a, b, and c can be accomplished by writing our own

simulator using any programming language (C, C++, Java ...etc). However, to simplify the modeling process, we use OPNET Modeler to model xBON. In fact, OPNET Modeler provides a development environment to model systems in C programming language. Furthermore, OPNET Modeler has rich built-in C libraries that facilitate the development process. These libraries already implement the functionalities required to achieve tasks a and c. Therefore, we can focus on implementing task b in OPNET Modeler while benefiting from the latter's capabilities to implement tasks a and c. This strategy preserves the generality of our model as the libraries of OPNET Modeler that we use can be easily written by any developer. In particular, OPNET Modeler provides C libraries that manipulate and process packets, effectively manage linked lists, process interrupts, collect statistics, implement queues and much other functionality. All of the latter libraries are in fact extensions to C functions that can be written by any developer.

In this section we describe the modeling alternatives according to which we can build our xBON system in OPNET Modeler. Since these modeling alternatives are mainly dependent on OPNET Modeler's model of BGP, we firstly give a brief introduction to OPNET Modeler before delving into the details of these alternatives.

3.4.1 OPNET Modeler

OPNET Modeler is a general-purpose simulation tool we use in this thesis to simulate the performance of a new inter-domain routing protocol for optical networks. OPNET Modeler (in the rest of this thesis we use the name "OPNET" instead of "OPNET" for simplicity) supports researchers with an environment that is rich of capabilities to design and build communication networks and protocols and study their performance.

OPNET presents the network as a hierarchy of three levels or editors:

a. Project Editor: This level views the actual network under study. This editor helps the designer in constructing a complete network by choosing the appropriate nodes (routers, switches, hubs...etc) from a list of well-known vendors (Cisco, Nortel, Motorola, Juniper Networks ... etc) and also selecting the appropriate technologies (ATM, FDDI, Ethernet ... etc) to connect these nodes. The Project Editor can be also referred to as the "workspace".

b. Node Editor: This level views the internals of each node in the network. This editor enables the designer to specify the stack of protocols supported by each node in the network. For example, if a designer requires a node to function as a BGP-speaker, then he/she should use the Node Editor to select the built-in BGP model and save it in that node (note that some other built-in models, like OSPF model, should also be saved in the same node for the configuration to be complete). The main communications protocols are ready as built-in models. For example, IPv4, IPv6, IS-IS, RIP, RSVP, UDP, MAC...etc.

c. Process Editor: This level views the finite-state machines according to which the node models operate. At this level the designer is capable of customizing any built-in model or building his/her new models. The language of programming used in the Process Editor is C.

For important definitions that will be used in the rest of this chapter, and for more information about OPNET, refer to Appendix A.

3.4.2 Modeling Alternatives

As we explained in the Chapter 2, BGP is the de facto inter-domain routing protocol that is extensively studied and deployed in today's telecommunications networks. Implementations

of BGP are widely available on-line using different programming languages and several platforms (See for example Vyatta at http://www.vyatta.com/, Quagga at http://www.quagga.net/, GNU Zebra at http://www.zebra.org/, and OpenBGPD at http://www.openbgpd.org/). In our thesis we depend on the implementation of OPNET v11.0. With OPNET BGP operates in five different scenarios each of which leverage certain features of BGP. We are, however, not interested in most of these capabilities and we need to simplify our implementation of xBON as much as possible. With this in mind, we have three modeling alternatives that can be used to model our system in OPNET:

A. Use the already implemented BGP in OPNET and modify it so that the changes proposed in xBON are taken into consideration.
B. Implement BGP from scratch, to include only the capabilities we need, and then apply the desired changes imposed by xBON, or
C. Benefit from OPNET's implementation of BGP, by concentrating on the basic functionality of BGP that we need, and build a user-defined prototype that deploys the desired changes imposed by xBON. In other words, create a hybrid model that benefits from both modeling alternatives A and B.

In the following sections we discuss the advantages and disadvantages of each modeling alternative and then choose the alternative that will be implemented and studied.

3.4.2.1 Modeling Alternative A: Use OPNET BGP Model

A snapshot of BGP's network in OPNET v11.0 is shown in Figure 3.7.

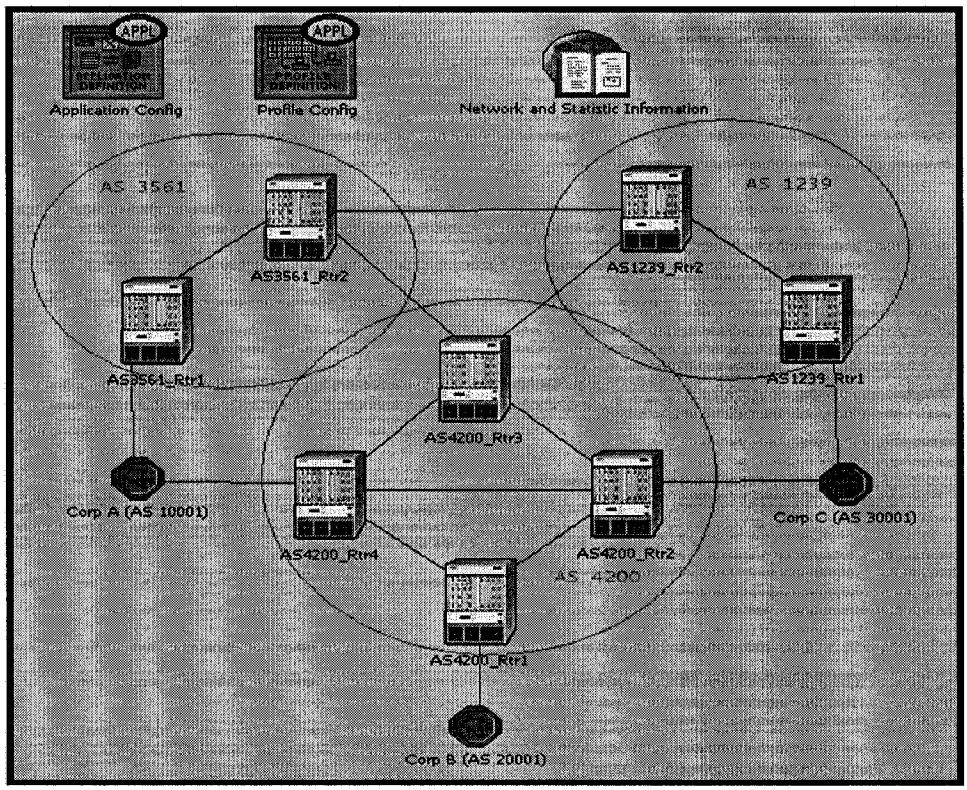

Figure 3.7 BGP's network in OPNET v11.0.

With this modeling alternative we can avoid the burden of implementing BGP from scratch and depend on OPNET's implementation of BGP. This allows us to invest our design efforts mainly in deploying our proposed changes to BGP. In addition, we can benefit from the already configured statistics in OPNET for BGP and compare the impact of xBON on the network before and after deploying it. However, the following drawbacks make this modeling alternative less attractive:

i. Extensive efforts are required to fully understand the whole system. This is because OPNET employs, in addition to BGP, several other communication protocols like OSPF, IS-IS, RIP, EIGRP, RSVP, ATM...etc, in the same workspace of the BGP model. Each of these protocols has its own module that connects directly or indirectly to the BGP module and removing any of these modules prevents the system from functioning

properly. This means that additional burden is added to understand each of these protocols and how it interacts with BGP before deploying xBON. A partial snapshot of the node model of a BGP-speaking router, showing the modules of the aforementioned communication protocols, is shown in Figure 3.8.

ii. OPNET implements many BGP path attributes (like MED, Local Preference...etc) and BGP capabilities (like filters) that are not needed in the system we aim to build. Again, such unneeded features of BGP may consume significant efforts to understand. These efforts should be fully directed to the construction of xBON.

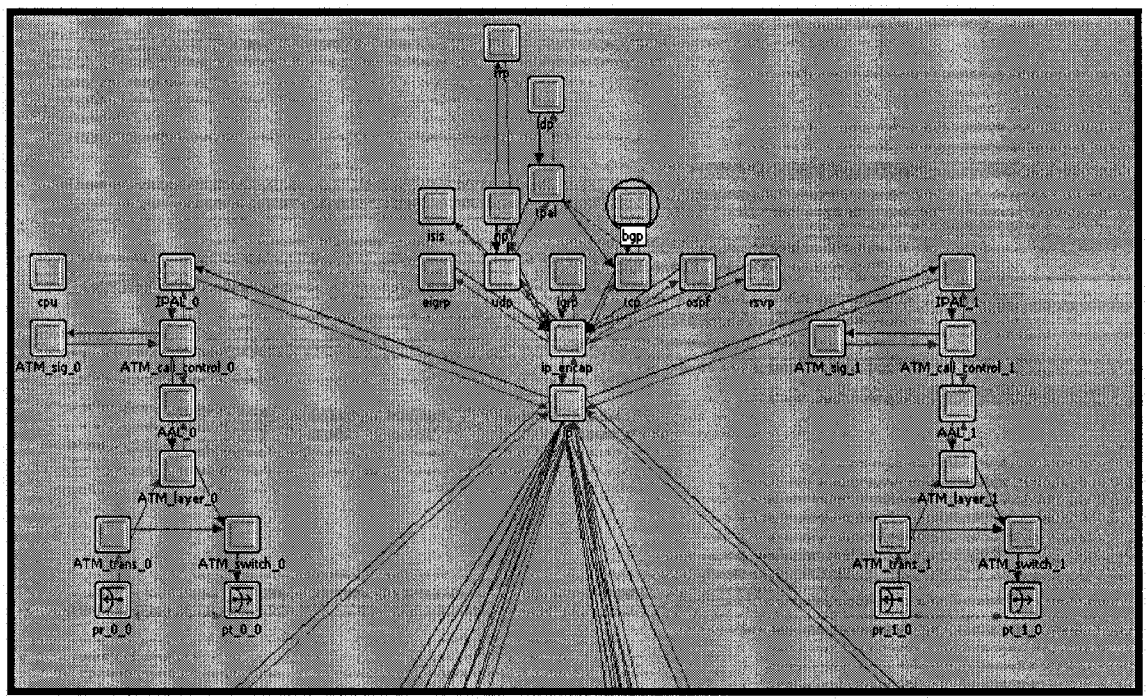

Figure 3.8 The node model of a BGP-speaker in OPNET v11.0.

3.4.2.2 Modeling Alternative B: Model The Whole System From Scratch

With this modeling alternative we ignore OPNET's BGP model and create a new project to implement BGP, along with our modifications to it, from scratch. This approach gives us the flexibility of employing only the capabilities of BGP that are of use in our final system with

xBON. By this we overcome the drawbacks associated with modeling alternative 1. However, this approach suffers from the following drawbacks:

i. The early phases of the system development will be focusing on the functionality of BGP itself and thus extensive tests will be needed to ensure that the system is functioning as intended. This is before adding the new functionality associated with xBON.

ii. Depending on trusted implementations of BGP, like OPNET's implementation, is preferred over using user-defined systems. This is because OPNET's BGP has been in use for a long period of time and its performance is verified and validated thoroughly.

3.4.2.3 Modeling Alternative C: Hybrid Model

The hybrid model depends on the experience we gain from having a general understanding of the BGP model in OPNET and then choosing the functionality of BGP that best suits our needs. After that, we build a new system in OPNET that deploys the chosen functionality as well as the functionality proposed in xBON. The main advantage of this modeling alternative is that with it we invest the whole development process in building capabilities and functionalities that are required in xBON. In other words, this approach eliminates the unused BGP features and all the other communication protocols that are beyond the scope of our study. The main drawback of this approach, however, is the tedious work needed to identify the portions of the BGP model, as implemented in OPNET, that are of use to our study and the portions that should be ignored.

From the above description of the three modeling alternatives, we choose Model alternative 3 as it suffers from fewer drawbacks and saves a significant amount of work to implement the overall system of xBON.

3.5 xBON Modeling in OPNET

In this section we describe the system model we built in OPNET v11.0 to implement xBON protocol. We build the network shown in Figure 3.9 in OPNET to deploy xBON. A node that is configured with xBON is referred to as an xBON-speaker.

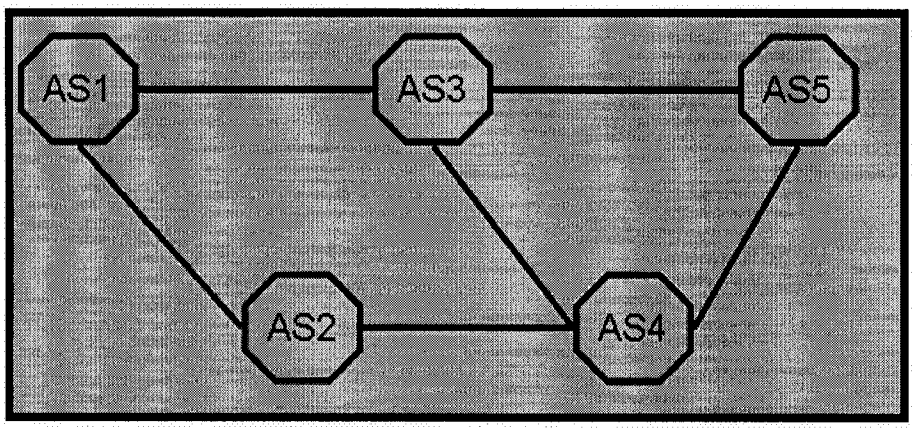

Figure 3.9 An example network to deploy xBON.

Figure 3.9 shows the top view of our network as it exactly appears in the workspace of OPNET. As discussed earlier in section 3.2, we are focusing on the inter-domain routing problem and thus we deal with each AS as if it contains only one xBON-speaker. Therefore, in the rest of this thesis, whenever we refer to a node or an xBON-speaker we are in fact talking about the AS as well.

Going further down in OPNET's hierarchy, Figure 3.10 shows the node model of each of the nodes shown in Figure 3.9. The ASx in this Figure refers to AS1, AS2, AS3, AS4, or AS5.

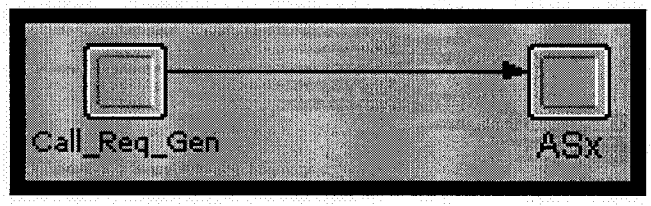

Figure 3.10 The node model of each xBON-speaker in the network shown in Figure 3.9.

For implementation purposes, we can see that replicating the node model shown in Figure 3.10 and including it in each node in Figure 3.9 may not be a practical approach (especially if the network grows in size). Instead, we can have a single node in the workspace that contains the functionality common to all the nodes in the network. The RIBs and the WADBs of the nodes can all be implemented as several databases within this single node. These databases can be easily accessed to process any incoming call request at any node. The topology of the network can be easily learnt from the RIB database. In other words, each node can depend on the AS paths to discover its neighbors and location within the network.

The above mentioned single node is named "BGP Optical Network" and will be appearing in the workspace as shown in Figure 3.11. At the node model level, the ASx module (see Figure 3.10) will be renamed as "Resources_Checker". The latter name comes from the fact that the main role of that single node is the management of the resources (typically, the wavelength) available in the network. This management covers the processes of accessing, reading, and updating all the RIBs and WADBs available in the network. The final node model is shown in Figure 3.12. The module named "Call_Req_Gen" (which stands for **Call Request Gen**erator) is used to generate call requests. This module is a copy of an OPNET's built-in module called "simple_source" which is responsible for generating packets according to any known distribution function. We modify the "simple_source" process model by removing some functionality that is of no use to our system. Figure 3.13 shows the finite state machine as it appears in the process model of our "Call_Rq_Gen" module. The "Init" state in this figure is used to initialize all the variables and statistics used in the process model.

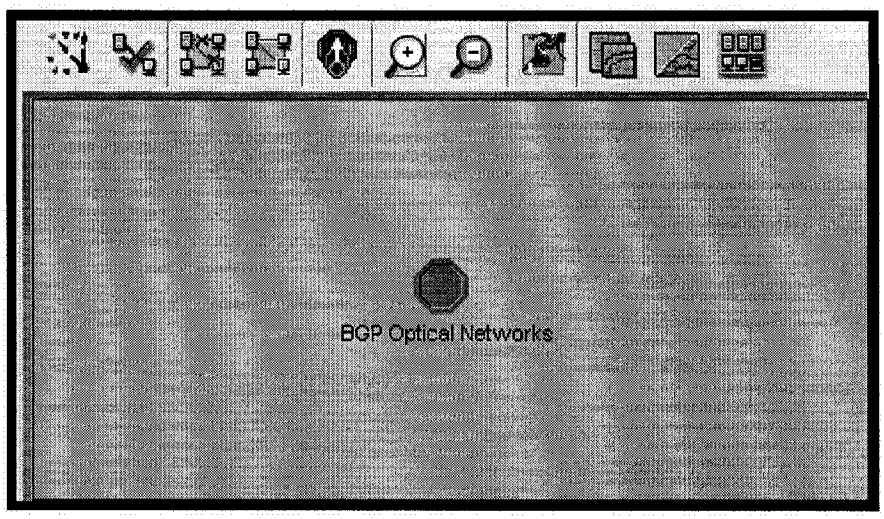

Figure 3.11 The workspace of the final BGP Optical Network project.

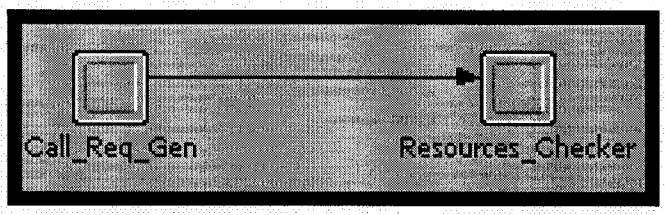

Figure 3.12 The node model of the single node of the BGP Optical Network.

The transition from the "Init" state to either the "generate" or the "stop" state is controlled by the conditions available on the transitions to these states. In particular, the transition from the "Init" state to the "generate" state requires that the condition START to be TRUE. Otherwise, the "stop" state will be entered. The statement (START)/ss_packet_generate, appearing on the transition from the "Init" state to the "generate" state, means that once START is evaluated to be TRUE, the C function ss_packet_generate (the "ss" stands for simple_source)will be called before entering the "generate" state. Basically, the ss_packet_generate function is the one responsible for generating the packets. We customized this function so that we guarantee that the generation of packets is following a

uniform distribution. That is, each node will be having an equal opportunity of receiving a packet (that is, a call request) at any instant. The "generate" state is responsible for scheduling the time at which a packet is generated, and the time at which the generated packet is sent to the Resources_Checker module.

The transition with the condition PACKET_GENERATE originates from and ends at the "generate" state itself. This means that we will remain in the latter state as long as the PACKET_GENERATE condition is TRUE. As shown in Figure 3.13, the transition from and to the "generate" state, while TRUE, guarantees that a packet is generated (by ss_packet_generate) whenever we go back to the Enter Executives.

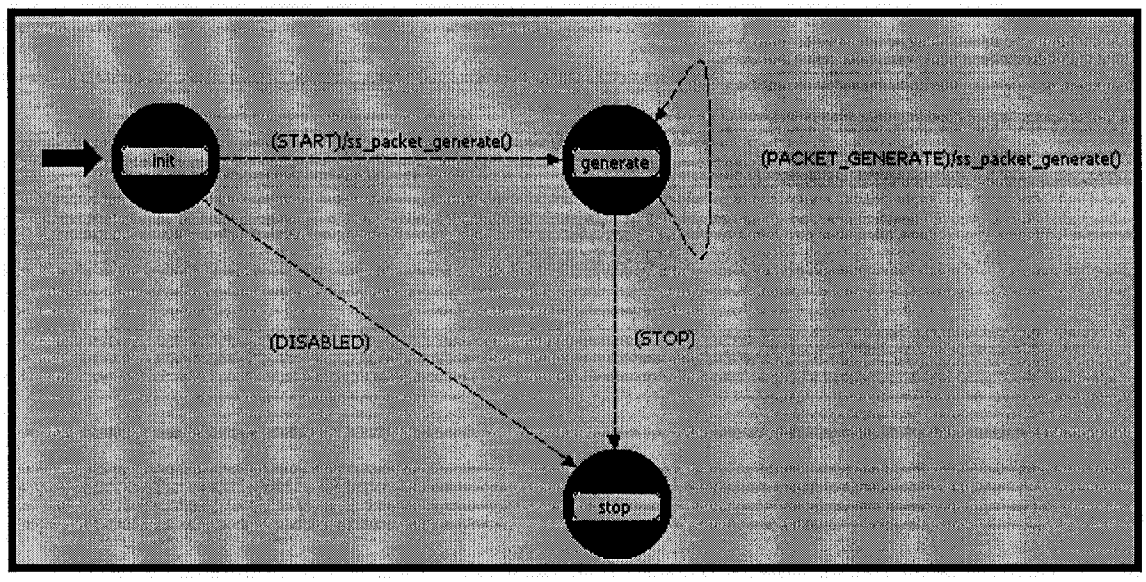

Figure 3.13 The process model of the "Call_Rq_Gen" module.

The "stop" is entered if either the DISABLED condition or the STOP condition is set to TRUE. We never use this state and thus the latter two conditions are always set to FALSE.

OPNET provides a capability of formatting and customizing packets according to the designer's needs. For the purpose of our simulation, we use the following simple format for the call request packets:

a. "Src_AS" field: used to identify the source AS.

b. "Dest_AS" field: used to identify the destination AS, and

c. "Holding_Time" field: used to specify the holding time period for the requested optical connection.

In Figure 3.14 we show the format of our packets. The length of all the fields in this packet format, as shown in the figure, is chosen to be 8 bits. The latter choice is arbitrary and should have no effect on our simulations. The generation of the packets, or call requests, can follow any known probability distribution (normal, uniform, exponential...etc).

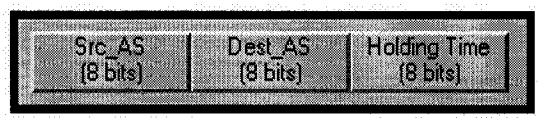

Figure 3.14 The format of the packets generated by the "Call_Rq_Gen" module.

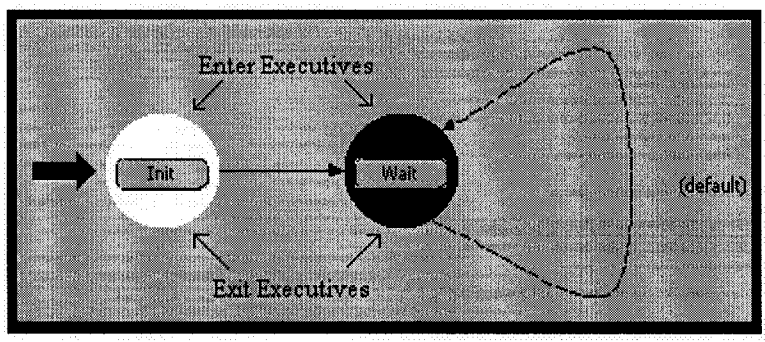

Figure 3.15 The process model of the Resources_Checker module.

The process model of the Resources_Checker module is shown in Figure 3.15. The Enter Executives of the "Init" state shown in this figure is used to initialize the static variables, the arrays, the linked lists, and the statistics that will be used during the course of simulation. In the Exit Executives of the "Init" state we locate the RIB and the WADB databases of all the

nodes in the network. In order to simplify our implementation, we assume that the RIBs are already constructed and available at the beginning of the simulation. In other words, we do not go through the phase of exchanging UPDATE messages among the BGP peers to populate their RIBs (which is the case in the normal operation of BGP). This assumption is valid because all the wavelengths should be available while the RIBs are being built and the actual exchange of the WAVELENGTH_STATUS attribute will not take place until the call requests start to arrive at the nodes.

The "Init" state is a forced state because, other than initializing the variables, statistics, and databases, no critical functions or operations are handled in this state and thus no need to remain in this state. Therefore, we transition to the "Wait" state directly after the initialization process is done in the "Init" state.

The "Wait" state contains the essential functionality of our system. This functionality covers the following operations:

a. Processing incoming call requests.
b. Accessing the RIBs and the WADB to check whether the incoming call requests can be granted optical connections or not.
c. Managing RIBs and WADBs.
d. Generating statistics.

The Exit Executives of the "Wait" state handles OPNET interrupts. Basically, we use two types of interrupts in the "Wait" state:

a. Stream interrupts: these are generated once a packet arrives at the Resources_Checker module.
b. Self interrupts: these are generated upon reserving or releasing wavelengths.

The Enter Executives of the "Wait" state is left empty. This is because of that, as mentioned in Appendix A, each unforced state firstly applies the C code available in the Enter Executives and then waits until certain conditions occur before applying the C code in the Exit Executives. In our case we need the system, after finishing the initializations of the "Init" state, to wait until any interrupt (stream or self) to occur. Thus, we leave the Enter Executives empty and keep the system waiting for the interrupts. Upon the occurrence of any interrupt, the Exit Executives is entered and the code there is applied. As shown in Figure 3.15, the "Wait" state, once the Exit Executives is exited, transitions back to the Enter Executives and waits for future interrupts. This is indicated by the "default" flag that occurs on the transition. The "default" refers to that no condition is required for the "Wait" state to transition from the Exit Executives to the Enter Executives (in other words, the condition is always TRUE).

In the rest of this chapter we cover in details the main functionality handled by the "Wait" state. Note that the description to follow applies to the Simple xBON Scheme (SxS), the Wavelength Threshold Scheme (WTS), and the Time Threshold Scheme (TTS). Thus, we assume that the Simple xBON Scheme (SxS) is the scheme applied and we highlight the differences related to the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS) whenever needed.

3.5.1 Processing Incoming Call Requests

During the course of simulation, the Resources_Checker module receives packets from the Call_Req_Gen module. Upon the arrival of a packet, the Src_AS and Dest_AS values are read from the packet fields. The Src_AS field helps in identifying the RIB and the WADB of the source AS. The Dest_AS field is used to select the correct AS path from the RIB. The

Resources_Checker then consults the WADB of the source AS to see if at least one wavelength is available and achieving the wavelength-continuity constraint on all the links constituting the selected AS path. A call request is rejected if the wavelength-continuity constraint cannot be satisfied.

By considering the wavelength-continuity constraint we are omitting the option of using wavelength converters. Wavelength converters can simplify our implementation significantly but are very expensive to deploy and thus avoided.

3.5.2 Accessing Databases

We have mentioned earlier in Section 3.2 that this thesis is not specifying a certain signaling protocol for a successful lightpath establishment. For the purpose of our implementation, we use a **C**entralized **D**ata**B**ase (CDB) that keeps up-to-date information about the wavelength availability on all the links in the network. By keeping such information, CDB can accurately find out whether a lightpath can be established over the As path or not. Therefore, CDB can be viewed as a "virtual" signaling protocol with which no actual signaling is taking place while the functionality of the signaling protocol is achieved. Therefore, no need to have an actual implementation of the CONFIRM, ESTABLISH, REJECT, and TEARDOWN messages that constitute the signaling protocol required to work with xBON. The CDB is used as follows. Once the source AS confirms that a certain wavelength (say, λ_x) is available (according to the view reflected by the WADB) to establish a lightpath to the destination node, it accesses the CDB. The CDB provides the source AS with the exact information about λ_x. Therefore, the source AS starts comparing the WADB's view of λ_x with CDB's view. The comparison can result in one of the following:

a. λ_x appears to be available for use in both CDB and WADB on all the links along the AS path. In this case, the setup of the lightpath will be successful using λ_x.

b. λ_x appears to be available for use in CDB but not available in WADB. In this case, the source AS depends on its WADB's view and ignores λ_x; and according to the FF scheme, the source AS examines the status of λ_{x+1} and repeats the databases comparison.

c. λ_x appears to be available for use in WADB but not available in CDB. In this case, the source AS should reject the call request since its information about λ_x is out-of-date.

d. λx appears to be reserved in both CDB and WADB. In this case, the source AS ignores λ_x; and according to the FF scheme, the source AS examines the status of λ_{x+1} and repeats the databases comparison.

3.5.3 Managing Databases

By "managing" we mean the process of retrieving information from the available databases (RIBs and WADBs) and updating them. As mentioned in the previous section, the CDB and the WADB of the source AS are accessed to find the wavelength over which the lightpath will be established. Once a wavelength (say, λ_x) is confirmed to satisfy the wavelength-continuity constraint, both of the CDB and the WADB are be updated to indicate that λ_x is now used and the lightpath is successfully established. The source AS should then send an UPDATE message to its neighbors for them to update their WADBs with the new status of λ_x (in the case of the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS), the UPDATE message is sent after crossing the wavelength threshold and the time threshold, respectively). These neighbors should also update their neighbors about the new change. The updating process continues until all the nodes in the network are

informed. Once the lightpath is torn down and λ_x is released, our system directly updates the CDB and the WADB of the source AS. The source AS then initiates the updating process of all the nodes in the network.

In order to simplify our system, we avoid the generation of actual UPDATE "packets" and depend instead on software interrupts. From an implementation perspective, using interrupts should have exactly the same functionality expected from the UPDATE packets. In particular, UPDATE packets experience a propagation delay while traveling over the links and once they reach an AS they advise it of certain updates to the WADB. Interrupts, on the other hand, can be scheduled to occur after a time period equal to the propagation delay (assuming that the propagation delay is the same for all the links in the network). Once an interrupt occurs at a node, the system directs the node to check the updates accompanying the interrupt so that the node can update its WADB accordingly. Since our whole system is implemented in a single node (see Figures 3.11 and 3.12), the type of interrupt used here, to implement the impact of UPDATE messages, is "self-interrupt". Interrupts in our system are received and processed in the Exit Executives of the "Wait" state.

In the rest of this thesis, once we mention the operation of "updating" or "notifying" nodes of any changes in the wavelengths availability, we implicitly mean that an interrupt is generated by a certain node.

3.5.4 Generating Statistics

The total number of generated call requests and the number of accepted (or rejected) call requests are collected in the Exit Executives of the "Wait" state. These data are used to determine the "blocking" performance of xBON. The statistics generated in our system are presented and discussed in depth in Chapter 4.

3.6 Summary

In this chapter we have introduced the OPNET tool used in the implementation and simulation of our system. After that, we have introduced and described xBON, a new inter-domain routing protocol for optical networks. Finally, we have given a detailed description of the system used in OPNET to implement and study xBON and its performance.

Chapter 4

Simulation and Results

4.1 Introduction

In this chapter we provide a complete description of the simulations we conduct on OPNET to study the performance of xBON. This chapter is organized as follows. Section 4.2 presents the system assumptions along with simulation parameters. Section 4.3 describes the different schemes of xBON implemented in OPNET along with the simulation runs. Finally, Section 4.4 provides the results generated and discusses them.

4.2 System Assumptions

The network we use in our simulation has been presented in Section 3.5 and is redrawn in Figure 4.1 for the reader's convenience.

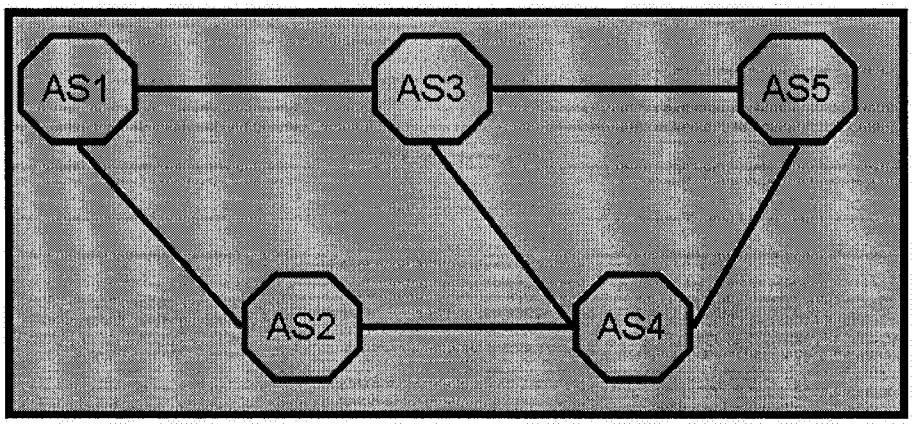

Figure 4.1 Network under study.

As mentioned in Chapter 3, each node in our network will have its RIB ready at the beginning of the simulation. In other words, we do not give some time for the RIBs to be built from zero.

This network is chosen randomly. However, we make sure that it is both simple and in harmony with our research objectives. The simplicity is needed due to some OPNET complexity in managing large networks, especially user-defined ones like ours. Although OPNET is a powerful simulation tool, a developer building a completely new and customized component may suffer from significant debugging work needed to fix even minor problems. Fortunately, our simple network can still prove our objectives in studying xBON. In particular, as has been discussed in Sections 3.2 and 3.5.3, propagation delay plays a significant role in making the WADBs out-of-sync with the CDB. To clearly see the impact of the propagation delay in our simulations, we make sure that we have at least one AS path consisting of 2 links. As an example, to setup a connection between AS1 and AS5, the shortest path to utilize is AS1-AS3-AS5.

In our simulations we set the following assumptions:

a. Each link in the network carries a total of 32 wavelengths.
b. The holding time of a lightpath is exponentially distributed with a mean value of 10 ms.
c. The time consumed in processing UPDATE packets at any node is neglected since the dominating time is the propagation delay on the inter-domain links.
d. Three different seeds are used in generating all the data. This means that for a given traffic load, we run the simulation three times and the values collected from these runs are averaged and plotted.

e. A confidence interval of 95% is used in the graphs to verify our results. We use MS Excel to compute the confidence interval (CI) for the generated results (see Appendix B for more information about the computation of CIs). We do not show the confidence intervals on the graphs given in Section 4.3 because they are very small. For example, for the graph shown in Figure 4.2, the computed confidence intervals for the curve corresponding to a propagation delay of 0.005 s are: 0.272217 ± 0.000421 (at a load of 0.5 Erlang), 0.359192 ± 0.00162 (at a load of 1 Erlang), 0.579368 ± 0.000264 (at a load of 5 Erlangs), 0.687671 ± 0.000116 (at a load of 10 Erlangs), and 0.743174 ± 9.78E-05 (at a load of 15 Erlangs). The above data clearly show how small the confidence intervals are.

f. The establishment and the release of lightpaths are both source initiated. That is, the source node starts the establishment of a lightpath, after detecting enough resources to do so, and once the holding time of the lightpath expires, the source node releases it.

g. The traffic of call requests is uniformly distributed among the nodes of the network. That is, each node has an equal opportunity of receiving a call request at any instant.

h. Call requests arrive according to a Poisson process. In other words, the inter-arrival time of call requests is exponentially distributed. The mean value (μ) of the latter distribution controls the load arriving at each node in the network according to the formula:

$$\textbf{Load (Erlang(s))} = \sigma\lambda$$

Where σ is the average holding time (10 ms) and λ (= $1/\mu$) is the call arrival rate (call/sec).

The objective of our simulations is to study the blocking performance of xBON. The blocking probability is defined as the probability of rejecting a call request arriving at a

node. We compute the blocking probability and study its behavior against the incoming traffic load of call requests.

We can identify two main reasons behind blocking a call request at the source node:

First reason: If both the node's WADB and the CDB show that no wavelengths are available for use or no available wavelengths can achieve the wavelength-continuity constraint. We refer to this reason as the **No_Wavelengths_Available** reason. The probability of blocking due to this reason is denoted by P_1 and computed as follows:

$$P_1 = \frac{\text{Number of rejected calls due to No\_Wavelengths\_Available}}{\text{Total number of call requests}}$$

Second reason: If the node's WADB shows that a certain wavelength is available for use and achieves the wavelength-continuity requirement to reach the destination node while the CDB shows the wavelength as unavailable. The blocking occurs in this case because the node will depend on its view and attempts to establish the lightpath over that wavelength. The attempt will fail and the call will be rejected. We refer to this reason of blocking as the **Out_of_Date_Information** reason. In this case, we denote the blocking by P_2 and compute it as follows:

$$P_2 = \frac{\text{Number of rejected calls due to Out\_of\_Date\_Information}}{\text{Total number of call requests}}$$

The total blocking probability (BP) in the network will simply equal the sum of P_1 and P_2.

P_1 results from the limited capacity of the links (that is, the number of wavelengths the links provide) as well as the traffic load (as we increase the traffic, the consumption of wavelengths increases and thus P_1 increases). This means that reducing P_1 may be difficult since the traffic in real telecommunication networks tends to increase significantly, and

using links with higher capacities will not have a considerable impact. On the other hand, P_2 results from the out-of-sync between the WADBs and the CDB. As we decrease this out-of-sync, the value of P_2 drops. This means that we should focus on implementing schemes that reduce the out-of-date information in the WADBs in order to reduce P_2. Thus, we have a strong opportunity of reducing the total blocking probability (BP) by concentrating on reducing P_2. In Chapter 3 we have introduced the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS) to reduce the impact of xBON on the scalability of BGP. Since the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS) do not provide instant update about the status of the wavelengths, they tend to increase P_2 compared to the Simple xBON Scheme (SxS). However, we show in the next sections that these schemes achieve a compromise between saving the scalability of BGP while providing reasonable blocking performance.

4.3 Schemes and Simulations

4.3.1 Simple xBON Scheme (SxS)

This scheme allows BGP to function normally by sending UPDATE messages, which hold the WAVELENGTH_STATUS attribute, whenever a change is detected in the wavelengths' availability. Although this scheme is impractical, due to its negative impact on the scalability of BGP, we need to study it first as it helps in understanding and evaluating the performance of the other two schemes.

In studying the Simple xBON Scheme (SxS), we aim to see the impact of propagation delays on the blocking probability in the network. We vary the value of the propagation delay on the links as follows: 1ms, 3 ms, 5 ms, 7 ms, and 9 ms. For each of these values, we use a traffic load in the range from 0 to 15 Erlangs.

Figure 4.2 shows the performance of our network under the Simple xBON Scheme (SxS). As the legend of the graph shows, each curve in the graph corresponds to a certain propagation delay. As expected, the graph clearly shows that BP increases significantly as the propagation delay on the inter-domain increases. For example, at traffic load of 5 Erlangs, BP reaches the value 25.55% at a propagation delay of 1msec while it increases to 69.29% at a propagation delay of 9ms. In other words, a reduction of almost 63% in the value of BP is achieved by reducing the propagation delay on the inter-domain links from 9 ms to 1ms. This can be understood from the fact that as we reduce the propagation delay we are faster in updating the nodes with any changes occurring in the availability of wavelengths.

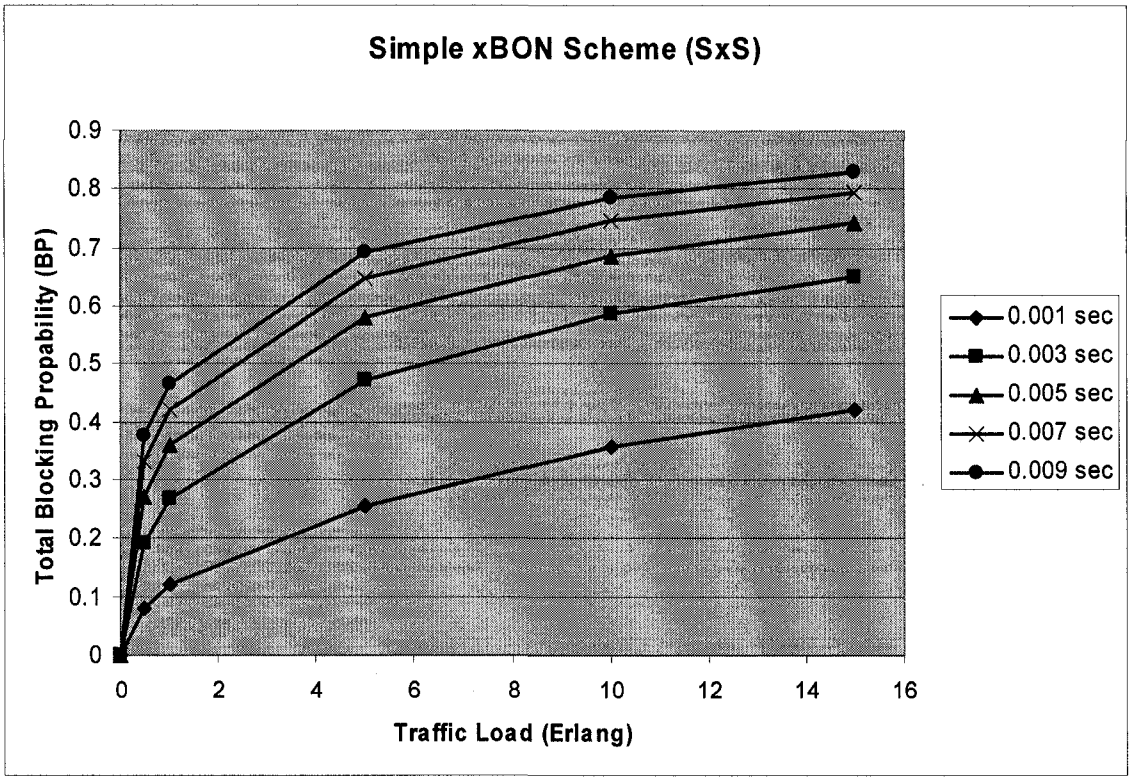

Figure 4.2 The blocking performance of the network under SxS.

Thus, we have reduced the possibility of having out-of –date information in the WADBs (with respect to the CDB) and thus have reduced the blocking due to **Out_of_Date_Information** problem.

Moreover, the graph shows that BP increases significantly as the traffic load increases. Obviously, this is a direct result of consuming the wavelengths quickly as the number of incoming calls increases.

To find out whether the **No_Wavelengths_Available** reason or the **Out_of_Date_Information** reason is contributing more to the blocking in the network, we show P_1 and P_2 in Figures 4.3 and 4.4, respectively. By carefully examining these two graphs, we can see that the blocking due to the **Out_of_Date_Information** reason is the main contributor to the total probability of blocking (BP).

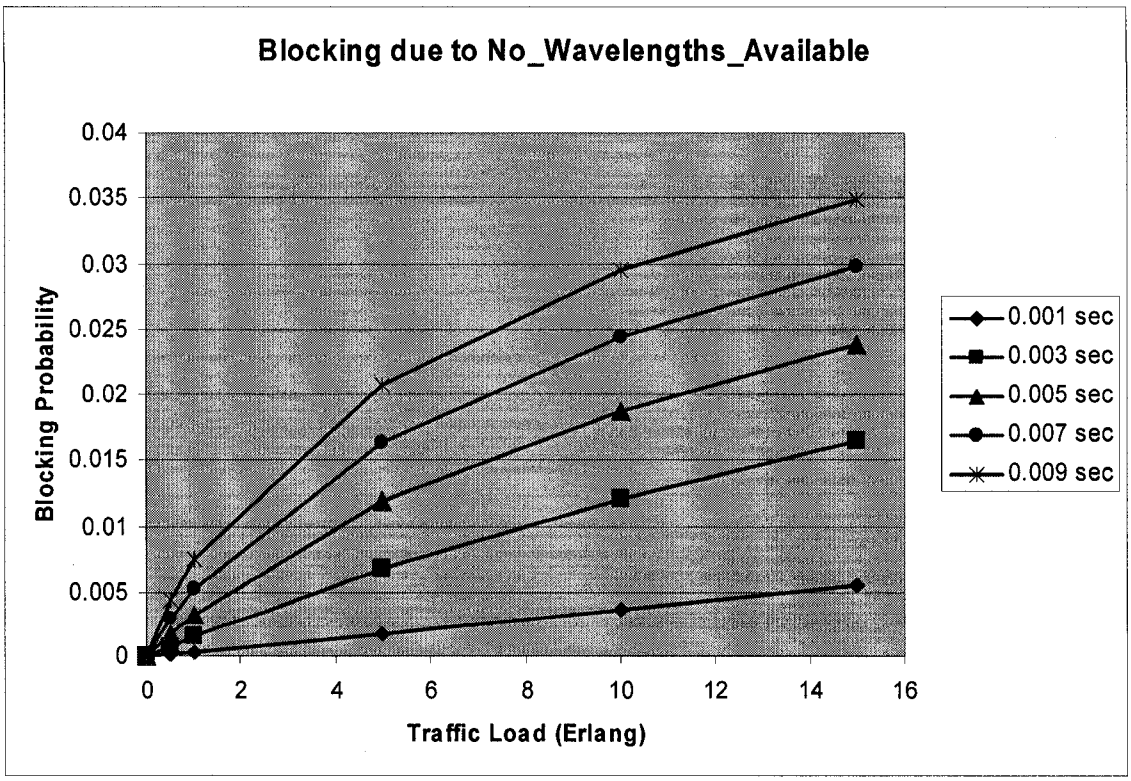

Figure 4.3 Blocking in the network, under SxS, due to No_Wavelengths_Available.

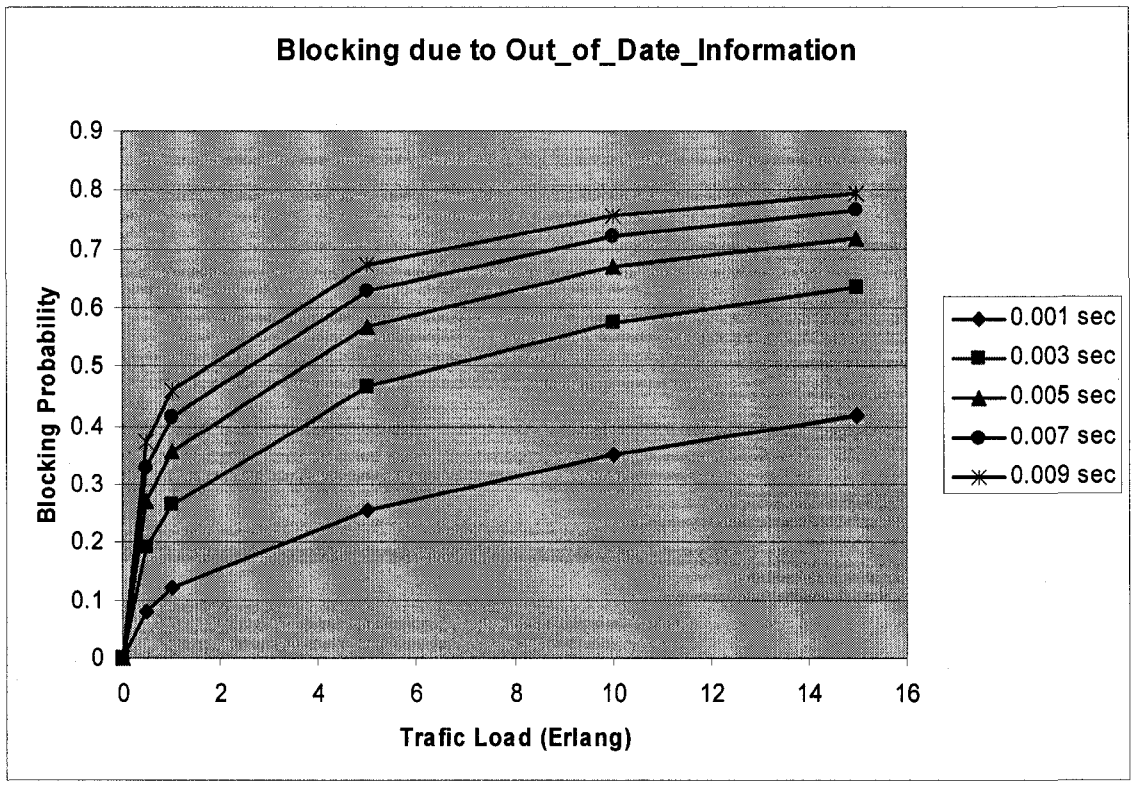

Figure 4.4 Blocking in the network, under SxS, due to Out_of_Date_Information.

This can be seen from both the general shape of the graph in Figure 4.3, which resembles the shape of the graph in Figure 4.2, as well as the values of the blocking probability in both of these two graphs. For example, at traffic load of 10 Erlangs and a propagation delay of 5 ms, P2 is 66.9% and BP is 68.77%, and at the same traffic load with a propagation delay of 3 ms, P2 is 57.55% and BP is 58.8%.

By examining Figure 4.3, we can see that P1 is 1.2% at a propagation delay of 3 ms and 1.87% at a propagation delay of 5 ms. The latter values show the minor contribution of P1 to BP.

All of the observations above prove that we should focus on introducing methodologies that reduce P2 in order to get the total probability of blocking (BP) reduced significantly.

It should be mentioned that although the values of BP and P2 look high, we should expect such high values. This is because our network is relatively small in size and the capacity of each inter-domain link is small while the traffic we inject into the network is relatively high. In other words, if AS1 receives traffic load of 5 Erlangs with a lightpath holding time not less than 0.01 second, then AS1 will be receiving 500 calls/sec. The maximum that AS1 can accept at a time is only 64 calls. This is because of that it has only two links connected to it and each of them can carry a maximum of 32 lightpaths at a time. This leads to a BP of around 87 % which is very high.

The main drawback of the Simple xBON Scheme (SxS), that makes it impractical, is that it requires the nodes to update their neighbors once a change occurs in the status of any wavelength. Given that the changes in the availability of wavelengths occur very frequently, a huge amount of UPDATE messages will be communicated among the nodes. As the size of the network grows, it is impractical to send exchange such a huge amount of messages and the scalability of BGP will be dramatically harmed.

4.3.2 Wavelength Threshold Scheme (WTS)

With the Wavelength Threshold Scheme (WTS) we do not send an UPDATE message as soon as a change occurs in the wavelengths availability. Instead, we wait until a certain percentage of the total capacity changes before sending the UPDATE message. In other words, with a total capacity of 32 wavelengths per link, we set "wavelength thresholds" that trigger the sending of UPDATE messages. For the purpose of our simulations, we use the following wavelength thresholds: 2, 4, 8, and 16 wavelengths which correspond to 6.25%, 12.5%, 25%, and 50% of the total capacity available on a link. The results are shown in Figures 4.5, 4.6, and 4.7. In generating the graph in Figure 4.5, we have used a propagation

delay of 3 ms on all the inter-domain links. Figure 4.5 clearly shows that the blocking probability increases as the traffic load increases and this is expected as has been discussed with the Simple xBON Scheme (SxS).

Figure 4.5 shows that the different wavelength thresholds perform almost the same in terms of the blocking probabilities achieved. Furthermore, when we compare Figure 4.5 with the curve generated for the Simple xBON Scheme (SxS) with a propagation delay of 3 ms, we can see that the curves are almost identical. For instance, at traffic load of 1 Erlang and a propagation delay of 3 ms on the inter-domain links, the Simple xBON Scheme (SxS) achieves a BP of 26.79% and the Wavelength Threshold Scheme (WTS) achieves a BP of almost 26% (this using any of the thresholds we previously set).

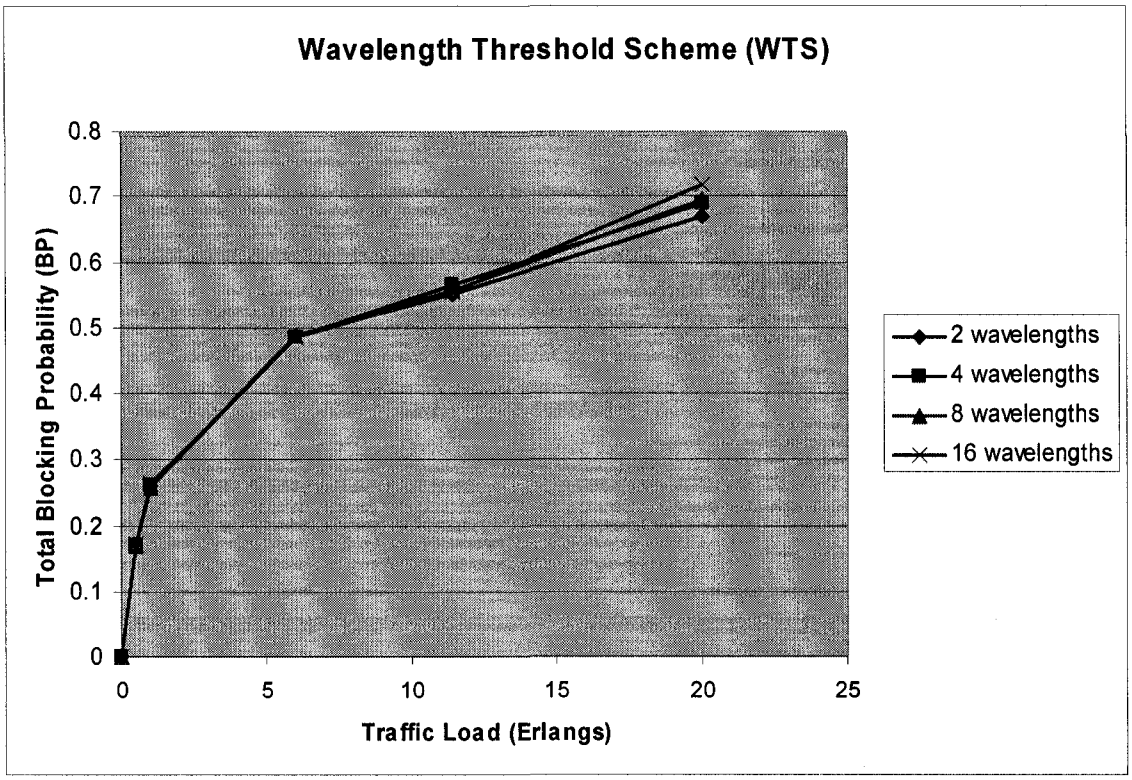

Figure 4.5 The blocking performance of the network under WTS.

The reason behind having almost the same BP with all the wavelength thresholds can be understood by examining Figures 4.6 and 4.7. In Figure 4.6, the overall behavior of the curve tends to increase as we increase the traffic load and this is expected. However, we can see that as we increase the wavelength threshold, the blocking due to **No_Wavelengths_Available** is decreased. This can be understood from observing that as we reduce the wavelength thresholds the information in the WADBs will be much more accurate than the case is with high wavelength thresholds. Thus, with small wavelength thresholds (2 or 4 for example), as we consume the resources available for a node (especially at high traffic loads), the possibility of rejecting a call request due to the **No_Wavelengths_Available** reason gets higher. On the other hand, in Figure 4.7, the value of P2 increases with the increase of the wavelength threshold. This is expected since as we increase the value of the wavelength threshold, the WADBs will wait for longer times before getting updated with the actual status of the wavelengths. Thus, with higher wavelength thresholds the WADBs will be reflecting the wrong view of the network and the blocking gets higher. An interesting observation in the behavior of P2 can be seen. At lower loads, the curves in this Figure tend to increase until a maximum value is reached and then, as the load gets higher, the curves start decreasing before stabilizing at a certain value. This is because at lower loads, the changes in the availability of wavelengths are less frequent and thus updating the nodes with the changes encountered will be very slow (since it takes longer time to cross the specified wavelength thresholds). This behavior causes the WADBs of the neighboring nodes to be containing out-of-date information and thus higher blocking probabilities result. As the incoming traffic load increases, wavelength thresholds will be crossed much faster and the updating process will get faster too. As a result, more accurate

information will be stored in the WADBs and this contributes to lower blocking probabilities (than the case with lower traffic loads). From the description above, we can see that P1 is low at lower traffic loads and increases gradually as the load increases while P2 has almost the opposite behavior (that is, high values at lower traffic loads and vice versa). The latter fact explains why BP under the Wavelength Threshold Scheme (WTS) behaves the same under the different wavelengths thresholds. In other words, the summation of P1 and P2 (which is basically BP) will not be affected by changing the wavelength threshold since they behave oppositely under lower and higher traffic loads. Furthermore, the discussion above explains the significant increase in P1 (and decrease in P2) under the Wavelength Threshold Scheme (WTS) when compared with P1 and P2 under the Simple xBON Scheme (SxS). Specifically, with the Simple xBON Scheme (SxS), at traffic load of 10 Erlangs and a propagation delay of 3ms on the links, we achieved a P1 of 1.2% while with the Wavelength Threshold Scheme (WTS) P1 is not less than 14% with a threshold of 16 wavelengths. That is, an increase by a factor of 11.67 in the value of P1 resulted with the Wavelength Threshold Scheme (WTS). On the other hand, at the same latter load and propagation delay, P2 under the Simple xBON Scheme (SxS) is 57.55% while with the Wavelength Threshold Scheme (WTS) P2 can be as low as 25.7% when we use a threshold of 2 wavelengths. This means a reduction of more than 55% in P2 could be achieved with the Wavelength Threshold Scheme (WTS). By now it should be very clear that the Wavelength Threshold Scheme (WTS) could significantly reduce the impact of **Out_of_Date_Information** on the overall blocking probability in the network. However, the price for that reduction was experiencing higher blocking due to the **No_Wavelengths_Available** reason.

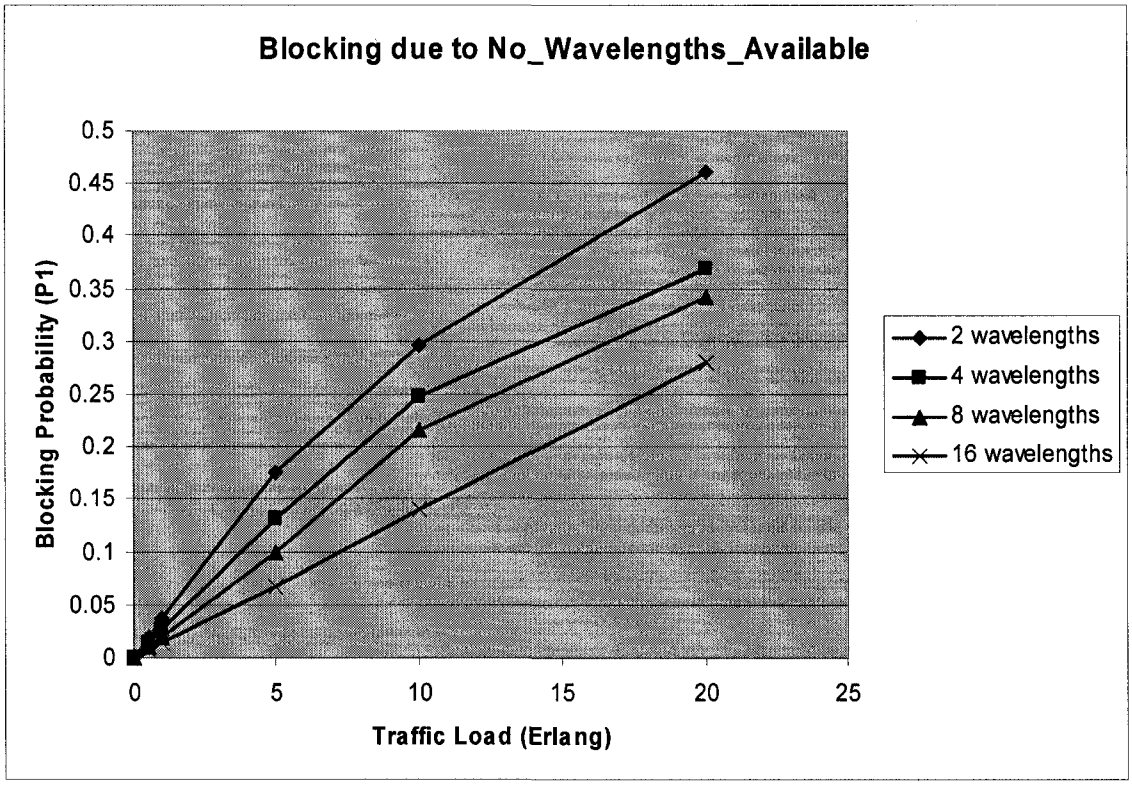

Figure 4.6 Blocking in the network, under WTS, due to No_Wavelengths_Available.

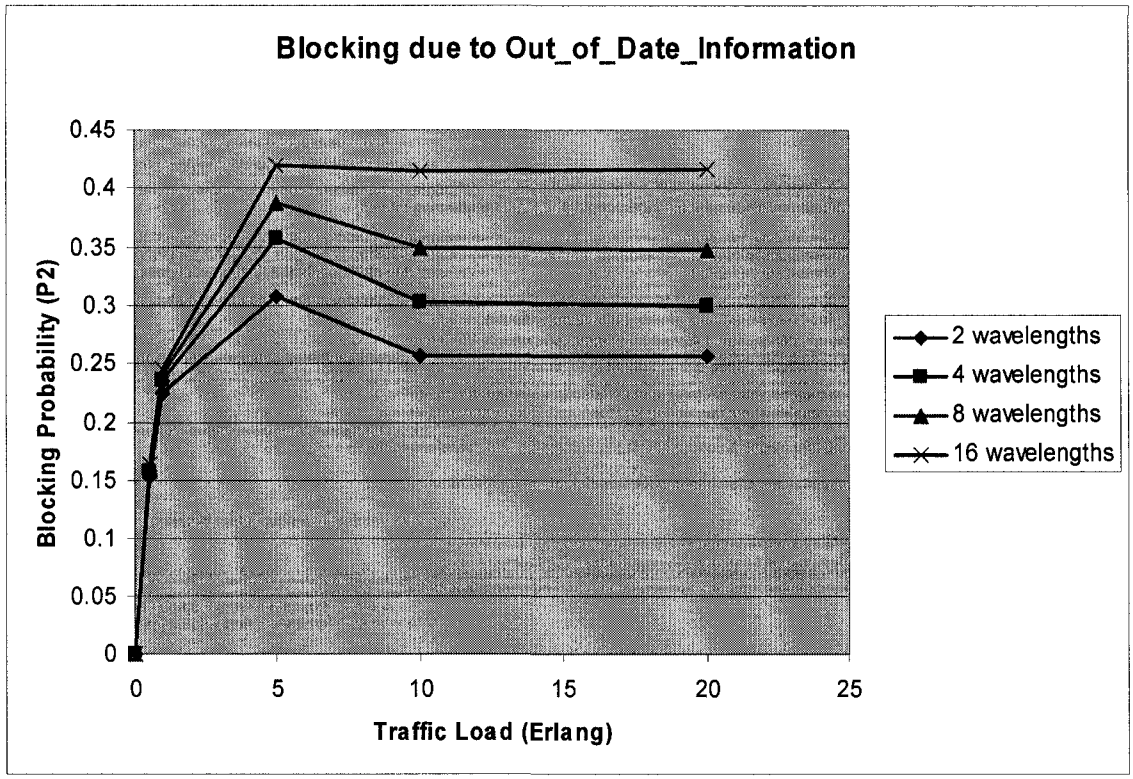

Figure 4.7 Blocking in the network, under WTS, due to Out_of_Date_Information.

4.3.3 Time Threshold Scheme (TTS)

With the Time Threshold Scheme (TTS), each node uses a timer that is set to certain "time thresholds" after which an UPDATE is automatically sent to the node's neighbors. In our simulations, we use the following time thresholds: 20msec, 40msec, 60msec, 80msec, and 100msec. The propagation delay used on the links is 3msec as the case was with the Wavelength Threshold Scheme (WTS). The resulting graphs are shown in Figures 4.8, 4.9, and 4.10. Figure 4.8 shows that the general behavior of BP tends to increase as we increase the traffic load (as the case was with both the Simple xBON Scheme (SxS) and the Wavelength Threshold Scheme (WTS)). Furthermore, with traffic loads less than 7.5 Erlangs, we can see that the blocking increases as we increase the value of the time threshold. This is because as we increase the time threshold value we are keeping the WADBs out-of-date for longer periods of time and thus the inaccuracy of the information stored there will result in higher probability of blocking. However, as can be seen from Figure 4.8, the latter behavior changes once we cross the 7.5 Erlangs and then the blocking starts to get higher as we lower the value of the time threshold. We can reason this due to the fact that the network is relatively small and the capacity of each inter-domain link is limited to 32 wavelengths. Thus, as we inject more traffic load into the network, we reach a point where we are getting out of resources while a huge amount of incoming calls continue to be received. The result is that the network becomes over-flooded, and as we update more frequently (that is, as we use lower time thresholds), it is likely by the time a node receives an update the information contained in that update is already out-of-date itself (due to the huge amount of incoming calls). However, if we give longer periods of time before updating (that is, as we use higher time thresholds) we will be reducing the amount of updates sent

and thus minimize the amount of error that can be in these updates. It is also worth mentioning that over-flooding the network with incoming traffic loads while having limited resources, may lead the network to reach a state where changing the time thresholds will have no effect on the overall blocking. This can be seen by observing that in Figure 4.8 the value of BP at traffic loads higher than 7.5 Erlangs and at the thresholds of 60msec, 80msec, and 100msec is almost the same. Such a behavior was not noticed with the Wavelength Threshold Scheme (WTS), although high traffic loads are also injected into the network, because of that in the Wavelength Threshold Scheme (WTS) the UPDATE message will always contain information about the changed wavelengths while with the Time Threshold Scheme (TTS) an UPDATE message is sent at the time thresholds whether there has been a change in the status of the wavelengths or not. Figure 4.8 clearly shows that the Time Threshold Scheme (TTS) achieves an overall blocking in the network that is less than the blocking achieved under the Simple xBON Scheme (SxS) and the Wavelength Threshold Scheme (WTS). For example, at traffic load of 15 Erlangs and a propagation delay on the links of 3msec, the Time Threshold Scheme (TTS) can achieve a BP of a value as low as 50.26% (with a time threshold of 20msec) while values of 65.13% and 60% are achieved with the Simple xBON Scheme (SxS) (with 22.83% reduction) and the Wavelength Threshold Scheme (WTS) (with 16.23% reduction), respectively. Let us now examine Figures 4.9 and 4.10 to more understand the performance of the network under the Time Threshold Scheme (TTS). In Figure 4.9 we can see that P1 increases as we decrease the time threshold from 100msec to 20msec. However, in Figure 4.10 we can see that P2 increases as the time threshold value increases from 20msec to 100 msec.

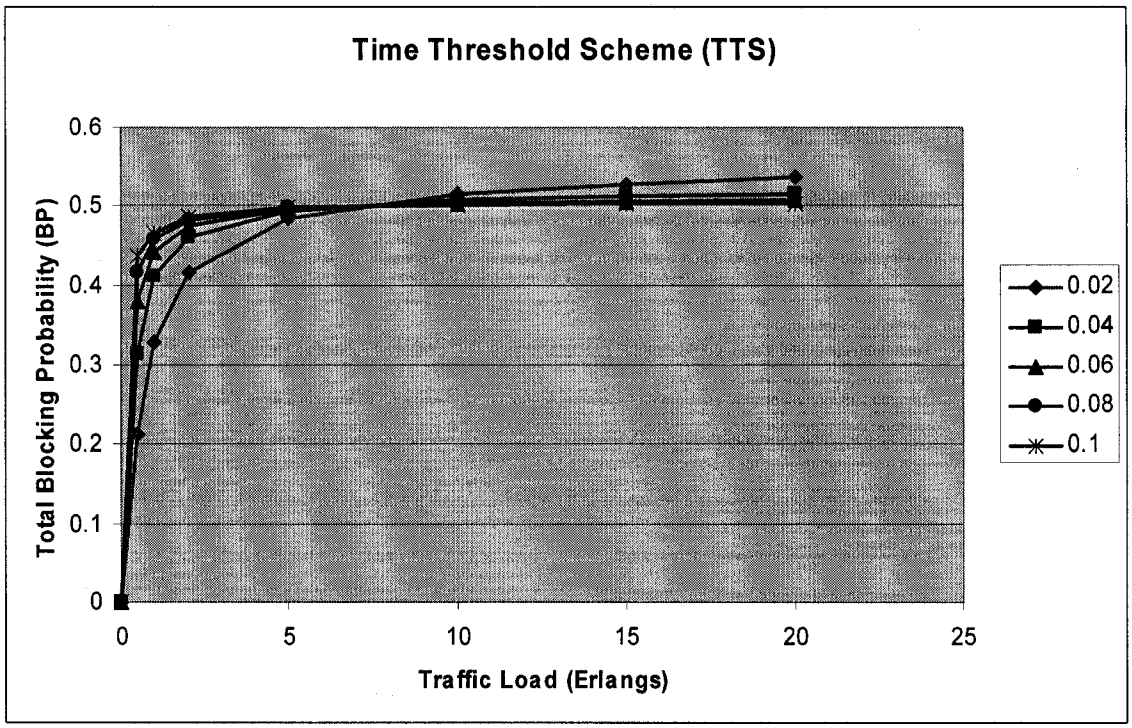

Figure 4.8 The blocking performance of the network under TTS.

Furthermore, we can see that the maximum value reached by P1 is 9.13%, which is achieved at traffic load 1 Erlang and time threshold 20msec. Meanwhile, P2 increases to reach values as high as 44.74% at 1 Erlang and continues to grow gradually, with the increase of the traffic load, until it stabilizes at around 49%. This means that P2 is the main contributor to BP under the Time Threshold Scheme (TTS). Thus, the blocking is taking place mainly because of suffering from inaccurate WADBs. This can be understood from the fact that as we increase the value of time threshold then the WADBs will wait for longer times before getting refreshed with the latest status of the wavelengths. Therefore, the incoming call requests are more likely to be blocked due to the wrong view of the network reflected by the WADBs (and this increases P2 dramatically). On the other hand, we can see in Figure 4.9 that P1 tends to increase at lower traffic loads to reach a maximum value, at a load of 0.5 Erlang (except for the case with time threshold of 20msec in which P2 reaches its maximum

at 1 Erlang). Then P1 starts to decrease its value before stabilizing. To understand this behavior let us consider the case of having a time threshold of 20msec and 100msec. At a 20msec-threshold, the refreshment of the information in the WADBs will be more frequent while with a 100msec-htreshold, the refreshment will be slow. Thus, with the latter time threshold, the accuracy of the stored data will be much less than the accuracy with a 20msec-threshold. This will result in that the incoming calls rejected due to the lack of resources (that is, due to the **No_Wavelengths_Available** reason) with a 20msec-threshold will be much more than the calls rejected with a 100msec-threshold. This means that with 100msec-threshold, the contribution of P2 to BP will be much more than the case is with a 20msec-threshold.

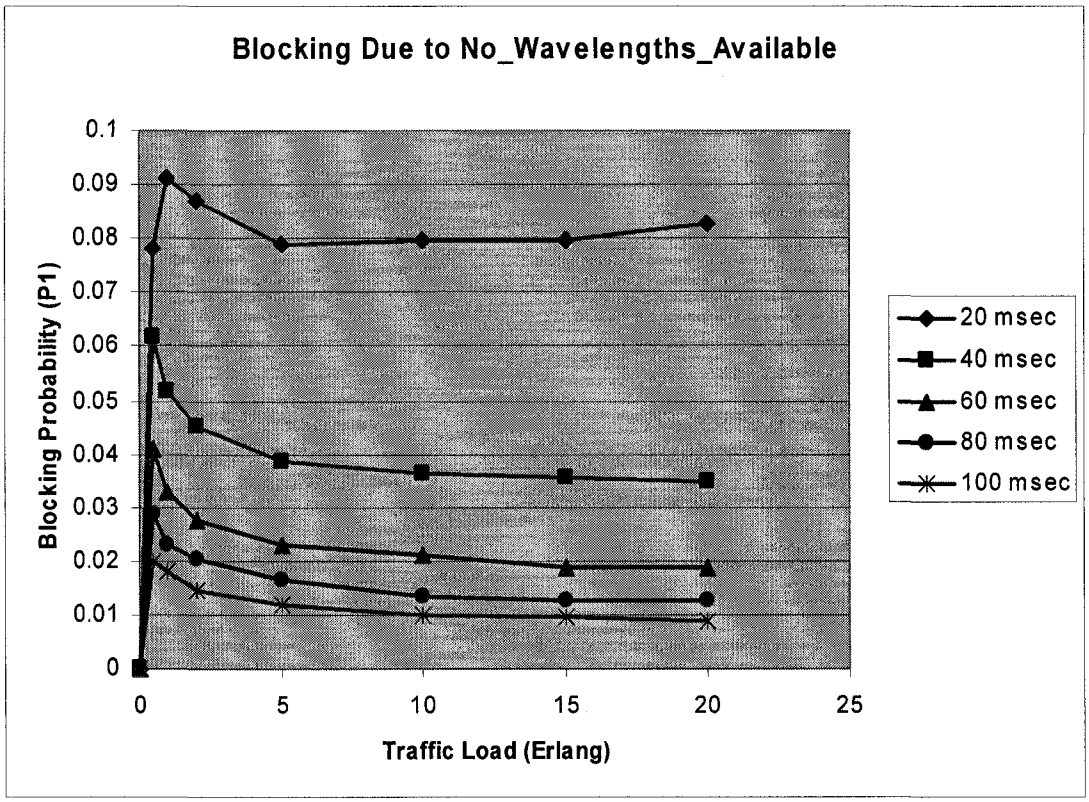

Figure 4.9 Blocking in the network, under TTS, due to No_Wavelengths_Available.

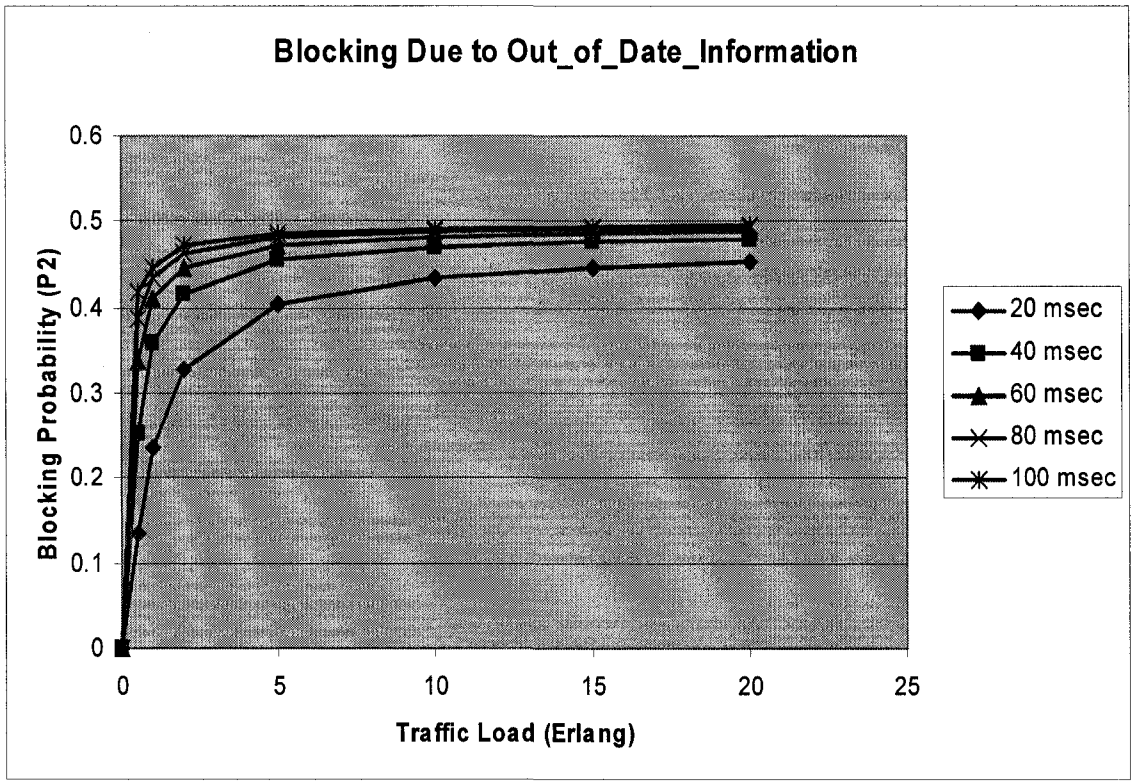

Figure 4.10 Blocking in the network, under TTS, due to Out_of_Date_Information.

4.4 Summary

In this chapter we have explained the methodology we have followed to simulate the performance of xBON. We have described the parameters and assumptions of our simulations as well as the schemes under which xBON has been studied. Finally, we have shown the data we generate using OPNET and have discussed our observations from these data.

Chapter 5

Conclusion

5.1 Summary and Concluding Remarks

In this thesis we have addressed the problem of achieving inter-domain routing in optical networks. We have described the routing problem in telecommunication networks in general and have focused on the protocols proposed to achieve inter-domain routing in these networks. Our focus has been mainly directed to the BGP protocol, the de facto inter-domain routing protocol that is traditionally and extensively deployed in today's IP-based networks. We have proposed the xBON protocol which extends BGP so that it can be successfully deployed in the optical domain. For BGP to carry information about the wavelengths reserved/released in the network, xBON uses a new path attribute named the WAVELENGTH_STATUS attribute and adds it to the UPDATE message of BGP. This attribute contains information about the status of the wavelengths on the inter-domain links only. We have not addressed the intra-domain routing problem in the optical domain. Therefore, any intra-domain protocol proposed for optical networks in the literature can be used to fill this gap.

xBON can operate in three modes or schemes:

i. Simple xBON Scheme (SxS): BGP works as usual and an UPDATE message is sent whenever a change in the status of any wavelength takes place.

iii. Wavelength Threshold Scheme (WTS): UPDATE messages are sent once pre-defined wavelength-thresholds are crossed.

iv. Time Threshold Scheme (TTS): UPDATE messages are sent once pre-defined time-thresholds are crossed.

We have qualitatively compared xBON to other optical inter-domain routing protocols. In particular, we have shown that xBON avoids the problems arising from as assigning signaling responsibilities to BGP (as the case is with OBGP in [FR02], [BL01] and [JE02]). We have also shown that xBON does not depend on trust relationships between ASes and does not advertise any intra-domain routing information (as the case is with OBGP in [BL01] and [JE02]). Moreover, we have discussed that xBON can operate in two modes, namely the Wavelength Threshold Scheme (WTS) and the Time Threshold Scheme (TTS), to efficiently report the continuous change in the availability of wavelengths (a feature that is missing in the BGP/GMPLS proposal in [XU03]). Finally, we have explained how xBON avoids advertising any intra-domain information about the wavelength availability (as the case is with ORBGP in [KH04]) as this can affect the scalability of BGP negatively.

The three schemes of xBON have been implemented in OPNET platform and extensive simulations were run to study the performance of these schemes.

In conclusion we have the following findings:

a. Simple xBON Scheme (SxS) is not a practical scheme to be deployed since sending UPDATE messages continuously, given the continuous changing nature of the wavelengths, will dramatically harm the scalability of BGP, which is the main motivation behind designing BGP itself. We have noticed that blocking due to lack of accurate information about the resources available for use (blocking due to the **Out_of_Date_Information** reason) is mostly contributing to the overall blocking in the network. The latter resulted from having a network over-flooded with incoming call

requests, having limited links' capacity, and keeping very accurate information about the status of the wavelengths in the network (see section 5.3.1 for more details).

b. The Wavelength Threshold Scheme (WTS) is more practical than the Simple xBON Scheme (SxS) as we send an UPDATE message after having considerable amount of changes in the status of the wavelengths. Thus, compared to the Simple xBON Scheme, the scalability of BGP is less affected with the Wavelength Threshold Scheme (WTS). Moreover, the Wavelength Threshold Scheme (WTS) could significantly reduce the blocking due **Out_of_Date_Information**. Although the blocking due to the lack of resources (blocking due to **No_Wavelengths_Available**) was significantly increased when compared to the Simple xBON Scheme (SxS) case, this problem is mainly related to the deployed capacity in the links of the network (see Section 4.3.2 for more details).

c. The Time Threshold Scheme (TTS) is more practical than the Simple xBON Scheme (SxS) as we send UPDATE messages after waiting for certain pre-defined periods of time. Again, this methodology helps in supporting the scalability of BGP while achieving a very good performance. In terms of overall blocking in the network, the Time Threshold Scheme (TTS) performs better than the Simple xBON Scheme (SxS) and the Wavelength Threshold Scheme (WTS) as it shows more stability under heavy traffic loads. Also, in terms of sources of blocking in the network, we can see that the Time Threshold Scheme (TTS) achieves more stability in the behavior of the blocking due to **Out_of_Date_Information** while still achieve very low values for the blocking due to **No_Wavelengths_Available** (see section 4.3.3 for more details).

d. We could see that the Time Threshold Scheme (TTS) is capable of providing better performance under heavy loads while giving a performance that almost resembles the

Simple xBON Scheme (SxS) and the Wavelength Threshold Scheme (WTS) under light loads. Also, in terms of handling both sources of blocking in the network, we can see that the Time Threshold Scheme (TTS) can provide significantly low values for the blocking due to the **No_Wavelengths_Available** reason when compared with the Wavelength Threshold Scheme (WTS). Although the Simple xBON Scheme (SxS) is achieving lower values for the blocking due to the **No_Wavelengths_Available** reason, the Simple xBON Scheme (SxS) as a whole is not viable practically. For the blocking due to the **Out_of_Date_Information** reason, while the Time Threshold Scheme (TTS) and the Wavelength Threshold Scheme (WTS) tend to stabilize its value at under heavy loads, we saw that the Simple xBON Scheme (SxS) keeps increasing with the increase of the load and this is another disadvantage of the Simple xBON Scheme (SxS). (See Section 4.3.3 for more details).

e. The overall conclusion is that the performance of xBON under the Time Threshold Scheme (TTS) makes it practically more preferred over the Wavelength Threshold Scheme (WTS) especially under heavy loads.

5.2 Future Research

a. It is important to study the scalability of xBON by deploying it in large-size networks like the NSF network. Also, the impact of the network type (start networks, ring networks ... etc) is an important area to focus on.

b. An interesting study to be focused on in the future is the interaction between xBON and any intra-domain routing protocol proposed for optical networks.

c. Wavelength conversion is an option to be considered in future studies to see its impact on the blocking due to **No_Wavelengths_Available**.

d. An interesting scheme that can more improve the performance of xBON is to define both wavelength and time thresholds at which an UPDATE message is sent.

Bibliography

[BA00a] J. Banks, J. S. Carson II, and B. L. Nelson, "Discrete-Event System Simulation", ISBN 0-13-088702-1, Prentice Hall PTR, Inc., 2000.

[BA00b] T. Bates, Chandra, Katz, and Y. Rekhter, "Multiprotocol Extensions for BGP4", RFC 2858, Jun. 2000.

[BA01] A. Banerjee, J. Drake, J.P. Lang , B. Turner, K. Kompella, and Y. Rekhter,"Generalized multiprotocol label switching: an overview of routing and management enhancements", IEEE Communications Magazine, vol. 39, no. 1. pp. 144 –150, Jan. 2001.

[BA02] A. Basu, C.-H. L. Ong, and A. Rasala, Gordon Wilfong, "Route Oscillation in I-BGP with Route Reflection", Proceedings of ACM SIGCOMM, USA, pp. 235 247, Aug. 2002.

[BA03] A. Banerjee, J. Drake, and J. Lang, "Generalized Multi-protocol Label Switching: An Overview of Signaling Enhancements and Recovery Techniques", IEEE Communications Magazine, pp. 144 -151, Jan. 2003.

[BE02] G. M. Bernstein, V. Sharma, and L. Ong., "Inter-domain optical routing", Journal of Optical Networking , vol. 1, no. 2, pp. 80 - 92, Feb. 2002.

[BE04] G. Bernstein, B. Rajagopalan, and D. Saha, "Optical Network Control: Architecture, Protocols, and Standards", Addison-Wisley, ISBN 0-201-75301-4, 2004.

[BL01] M. Blanchet, " Optical BGP (OBGP): InterAS lightpath Provisioning", draftparent-obgp-01.txt, Internet Draft, expired, Aug. 2001.

[CA02] D. Cameron, "Optical Networking", ISBN 0-471-44368-9, John Wiley & Sons, Inc., 2002.

[CH96a] I. Chlamtac, and T. Zhang, "Lightpath (wavelength) Routing in Large WDM Networks", IEEE Journal on Selected Areas in Communication, vol. 14, no. 5, pp. 909 - 913, Jun. 1996.

[CH96b] R. Chandra, P. Traina, "BGP Communities Attribute", RFC 1997, Aug. 1996.

[CH03] X. Chu, B. Li, and J. Liu, "Wavelength Converter Placement under a Dynamic RWA Algorithm in Wavelength-Routed All-Optical Networks", IEEE Transactions on Communications, vol. 51, no. 4, pp. 607 - 617, Apr. 2003.

[CH06] G.-K. Chang, J. Yu, Y.-K. Yeo, A. Chowdhury, and Z. Jia, "Enabling technologies for next-generation optical packet-switching networks", Proceedings of the IEEE, vol. 94, no. 5, pp. 892-910, May 2006.

[CO02] D. Colle, S. De Maesschalck, M. Pickavet, P. Demeester, M. Jaeger, and A. Gladisch, "Developing control plane models for optical networks", Proceedings of Optical Fiber Communication Conference and Exhibit (OFC'02), USA, pp. 757 – 759, Mar. 2002.

[CO03] D. Colle, J. Cheyns, C. Develder, E. Van Breusegem, A. Ackaert, M. Pickavet, and P. Demeester,"GMPLS extensions for supporting advanced optical networking technologies", Proceedings of 2003 5th International Conference on Transparent Optical Networks, Poland, vol 1, pp. 170-173, Jun.-Jul. 2003.

[FR02] M.J. Francisco, L. Pezoulas, C. Huang, and I. Lambadaris, "End-to-end signaling and routing for optical IP networks", Proceedings of IEEE International Conference on Communications (ICC'02), USA, vol.5, pp. 2870 – 2875, Apr.-May 2002.

[GR02] T. G. Griffin, and G. Wilfong, "On the Correctness of IBGP configuration", Proceedings of ACM SIGCOMM, USA, pp. 17 - 29, Aug. 2002.

[HA05] A. Hafid, A. Maach, M. G. Khair, and J. Drissi, "Optical routing border gateway protocol-based advance lightpath setup", Proceedings of Systems Communications, Canada, pp. 223-228, Aug. 2005.

[JE02] S. Jeong, and C.-H. Youn, "Instability Analysis for OBGP Routing Convergence in Optical Networks", Proceedings of ICOIN-16, Korea, Jan. 2002.

[KA02] D. Katz, D. Yeung, and K. Kompella, "Traffic engineering extensions to OSPF version 2," Internet Draft, Work in Progress. Available online: draft-katz-yeung-ospf-traffic-09.txt, Oct. 2002.

[KH04] M. G. Khair, "An implementation approach for an inter-domain routing protocol for DWDM", Master's thesis, University of Ottawa, Mar. 2004.

[KO02] K. Kompella and Y. Rekhter, Eds., "OSPF extensions in support of generalized MPLS," Internet Draft, Work in Progress. Available online: draft-ietf-ccamp-ospf-gmpls-extensions-09.txt, Dec. 2002.

[KO05] K. Kompella, Y. Rekhter, L. Berger, "Link Bundling in MPLS Traffic Engineering (TE)", RFC 4201, Oct. 2005.

[LA98] C. Labovitz, G. R. Malan, and F. Jahanian, "Internet routing instability", IEEE/ACM Transactions on Networking, vol. 6, no. 5, pp. 515 – 528, Oct. 1998.

[LA02] N. Larkin, "ASON and GMPLS-the battle of the optical control plane", available online from www.dataconnection.com, 2002.

[LE-GA01] A. Leon-Garcia, and I. Widjaja, "Communication Networks: Fundamental Concepts and Key Architectures", ISBN 0-07-250353-X, McGraw-Hill Inc., 2001.

[LI00] K. H. Liu, C. Lui, and J. Y. Wei, "Overlay vs. Traffic Engineering for IP/WDM Networks", IEEE GLOBECOM, vol. 2, pp. 1293 -1297, Nov. 2000.

[LI02a] G. Li, J. Yates, D. Wang, and C. Kalmanek, "Control plane design for reliable optical networks", IEEE Communications Magazine, vol. 40, no. 2, pp. 90-96, Feb. 2002.

[LI02b] K. H. Liu, "IP over WDM", ISBN 0-470-84417-5, John Wiley & Sons Ltd, 2002.

[LO05] K. Loja, J. Szigeti, and T. Cinkler, "Inter-domain routing in multiprovider optical networks: game theory and simulations", Next Generation Internet Networks, pp. 157 – 164, Apr. 2005.

[MA02a] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP configuration", Proceedings of ACM SIGCOMM, USA, pp. 3 - 16, Aug. 2002.

[MA02b] O. Maennel, and A. Feldmann, "Realistic BGP Traffic for Test Labs", Proceedings of ACM SIGCOMM, USA, pp. 235 - 247, Aug. 2002.

[MO03] H. T. Mouftah, and P.-H. Ho, "OPTICAL NETWORKS Architecture and survivability", ISBN 1-4020-7196-5, Kluwer Academic Publishers, Inc., 2003.

[MO98] J. T. Moy, "OSPF: Anatomy of an Internet Routing Protocol", ISBN 0-201-63472-4, Addison Wesley Longman, Inc., 1998.

[PA06] D. Papadimitriou, Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Extensions for G.709 Optical Transport Networks Control", RFC 4328, Jan. 2006.

[PI05] C. Pinart and G.J. Geralt, "On managing optical services in future control-plane-enabled IP/WDM networks", Journal of Lightwave Technology, vol. 23, no. 10, pp. 2868-2876, Oct. 2005.

[**RA02**] R. Ramaswami, K. N. Sivarajan, "optical networks: a practical perspective", ISBN 1-55860-655-6, Morgan Kaufmann Publishers, 2002.

[SA03a] G. H. Sasaki, and S. Ching-Fong, "The interface between IP and WDM and its effect on the cost of survivability", IEEE Communications Magazine, vol. 41, pp. 74 - 79, Jan. 2003.

[SA03b] D. Saha, B. Rajagopalan, G. Bernstein, "The optical network control plane: state of the standards and deployment", IEEE Communications Magazine, vol. 41, no. 8, S29 - S34, Aug. 2003.

[SA04] T. Saad, and H.T. Mouftah, "Inter-domain wavelength routing in optical WDM networks", Proceedings of 2004 11th International Telecommunications Network Strategy and Planning Symposium, Austria, pp. 391-396, Jun. 2004.

[SO-PA03] J. Sole-Pareta, X. Masip-Bruin, S. Sanchez-Lopez, S. Spadaro, and D. Careglio, "Some open issues in the optical networks control plane", Proceedings of 2003 5th International Conference on Transparent Optical Networks, Poland, vol. 1, pp. 76 – 81, Jun.-Jul. 2003.

[ST00] W. Stallings, "Data and Computer Communication". ISBN 0-13-084370-9, Prentice Hall PTR Inc., 2000.

[ST99] J. W. Stewart, "BGP-4: Inter-Domain Routing in the Internet", ISBN 0-201-37951-1, Addison Wesley Longman Inc., 1999.

[ST04] W. Stallings, "Computer Networking with Internet Protocols and Technology", ISBN 0-13-141098-9, Prentice Hall, 2004.

[SZ04] J. Szigeti, J. Tapolcai, T. Cinkler, T. Henk, and G. Sallai, "Stalled information based routing in multidomain multilayer networks", Proceedings of 2004 11th International Telecommunications Network Strategy and Planning Symposium, Austria, pp. 297 – 302, Jun. 2004.

[TH01] S. A. Thomas, "IP Switching and Routing Essentials: Understanding RIP, OSPF, BGP, MPLS, CR-LDP, and RSVP-TE", ISBN 0-471-03466-5, WILEY Inc., 2001.

[TO01] P. Tomus, and C. Schmutzer, "Next Generation Optical Networks", SBN 0-13-028226-x, Prentice Hall PTR Inc., 2001.

[XI05] Y. Xi., and B. Ramamurthy, "Dynamic routing in translucent WDM optical networks: the intradomain case", Journal of Lightwave Technology, vol. 23, no. 3, pp. 955 – 971, Mar. 2005.

[XU03] Y. Xu, A. Basu, and Y. Xue, "A BGP/GMPSL Solution for Inter-domain Optical Networking'. draft-xu-bgp-gmpls-03, IETF Internet Draft, expired, Sept. 2003.

[WE98] J.Y. Wei, S. Chien-Chung, B. J. Wilson, M. J. Post, and Y. Tsai, "Connection management for multiwavelength optical networking", IEEE Journal on Selected Areas in Communications, vol. 16, no. 7, pp. 1097 – 1108, Sept. 1998.

[WH05] R. White, D. McPherson, and S. Srihari, "Practical BGP", ISBN 0-321-12700-5, Addison-Wisley, 2005.

[WU02a] J. Wu, J.M. Savoie, A. Vukovic, and H. Hua, "Control of multi-domain lightpaths in peer-to-peer optical networking", All-Optical Networking: Existing and Emerging Architecture and Applications/Dynamic Enablers of Next-Generation Optical Communications Systems/Fast Optical Processing in Optical Transmission/VCSEL and Microcavity Lasers. IEEE/LEOS Summer Topical Meetings, pp. TuM3-39 - TuM3-40, Jul. 2002.

[WU02b] J. Wu, J.M. Savoie, and B. St. Arnaud, "Functional Requirements of Peer-to-Peer Optical Networking", Proceedings of 2002 28th European Conference on Optical Communication (ECOC'02), Denmark, vol. 2, pp. 01 – 02, Sept. 2002.

[WR03] B. Wright, "Inter-area routing, path selection and traffic engineering", available online from www.dataconnection.com, 2003.

[ZA00] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks", SPIE/Baltzer Optical Network Magazine, vol. 1, no. 1, pp. 47–60, Jan. 2000.

[ZH01] Z. Zhang, J. Fu, D. Guo, and L. Zhang, "Lightpath Routing for Intelliaent Optical Networks", IEEE Network, Jul.-Aug. 2001.

[ZH02] J. Zheng and H. T. Mouftah, "Routing and wavelength assignment for advance reservation in wavelength-routed WDM optical networks", Proceedings of IEEE International Conference on Communications (ICC'02), USA, vol. 5, pp. 2722 – 2726, Apr.-May 2002.

Appendix A

Introduction to OPNET Modeler

OPtimized **N**etwork **E**ngineering **T**ool (OPNET) Modeler is general-purpose simulation platform that provides a comprehensive development environment to help a designer to model communication networks and distributed systems*. Figure A.1 shows the main GUI of OPNET Modeler v11 that appears after launching it.

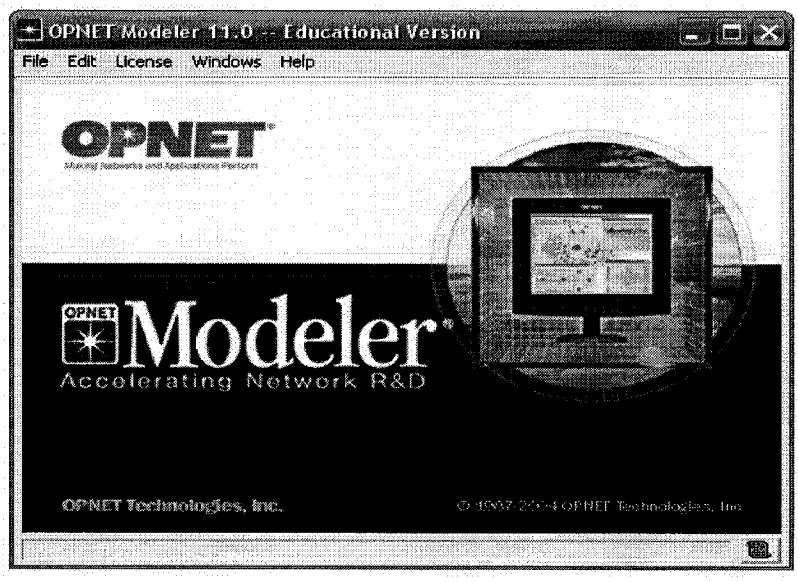

Figure A.1 Main GUI of OPNET Modeler v11.

The environment provided by OPNET Modeler supports a researcher with tools for all phases of a study, including model design, simulation, data collection, and data analysis. Among those tools are the "editors" which are used to simulate the characteristics and

* In writing this section we depend mainly on OPNTE Modeler Product Documentation for Release 11.0.

performance of a modeled system's behavior. The main editors that we work with for the purpose of this thesis are the following:

- **Project Editor**: This editor provides a workspace in which networks are developed. Network models are built from both subnets and node models. This editor also contains the basic simulation and analysis options and capabilities.
- **Node Editor**: This editor provides capabilities to develop node models. Node models are built from modules with process models. Node models appear as objects in the network model.
- **Process Editor**: This editor provides capabilities to develop process models. A process model is nothing but a finite state machine that is programmed in C language to perform the desired behavior of a node in the network.
- **Packet Format Editor:** This editor provides capabilities to develop packet formats models. Packet formats dictate the structure and order of information stored in a packet.

Project, Node, and Process editors are organized in a hierarchical fashion as shown in Figure A.2 (In this figure, we are showing a subnet view in order to be general; however, a project editor may just contain nodes without any subnets). This hierarchical organization means that the model specifications applied in the Project Editor rely on the work designed in the Node Editor. Moreover, when working in the Node Editor, the designer depends on models defined in the Process Editor.

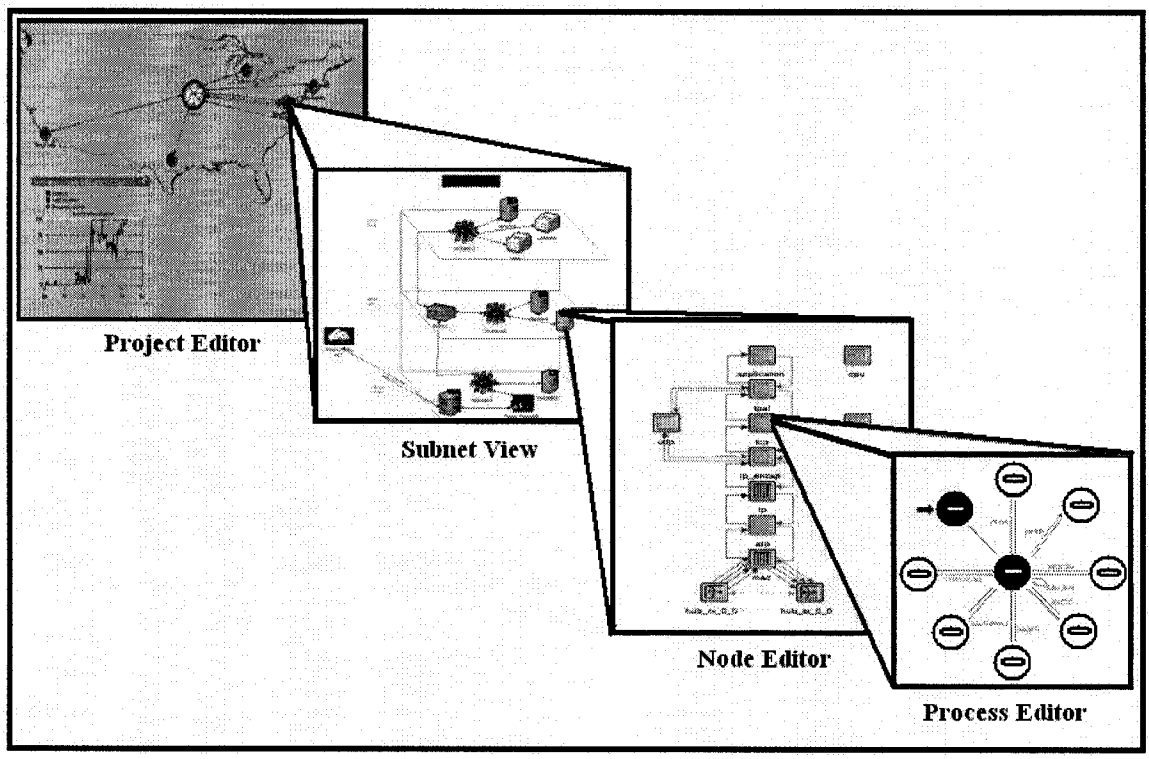

Figure 2 Each project in OPNET Modeler is organized in a hierarchical manner.

As mentioned above, the process editor depends on finite state machines to implement the desired behavior of a node in the network. As shown in Figure A.2 above, the process editor contains black states and white states. It is within these states where the C code is written. This is illustrated in Figure A.3. Each state is split into two sections, namely, Enter Executives (Execs) and an Exit Executives. Enter executives are executed when a process enters the state while exit executives are executed when the process leaves the state to follow one of the outgoing transitions (that lead the state to other states or keep the process in the same state). It is within these executives where the C code is written (as shown in Figure A.3). In OPNET Modeler two types of states are supported, namely, forced states and unforced states. In the process editor, forced states are shown in white while unforced states are shown in black (In OPNET Modeler, forced states are green-colored while unforced

states are red-colored. We present here different colors for printing purposes). With forced states the process will directly execute the Exit Execs once the execution of the Enter Execs is completed. However, with unforced states the process pauses between the Enter Execs and the Exit Execs. In other words, once the Enter execs are executed the control is directed back to the context that invoked the unforced state. After that, the process waits for a new invocation to progress into the Exit Execs of the unforced state.

With the description just mentioned, we can see that unforced states are best used to implement true states or modes of a system that persist for certain durations. On the other hand, forced states are useful in controlling flow decisions that are usually common to several unforced states.

In addition to C language, OPNET Modeler provides a rich built-in library that supports a designer with a lot of C functions that facilitate the design and modeling of networks. These C functions are referred to as Kernel Procedures (KPs). KPs are categorized according to the component of the system they deal with. That is, there are KPs for dealing packets, network topology, links, statistics, probability distributions, nodes, interrupts and many other sets. Such categorization facilitates modeling and provides a user-friendly environment where the developer can easily identify what KP is required to implement a certain process.

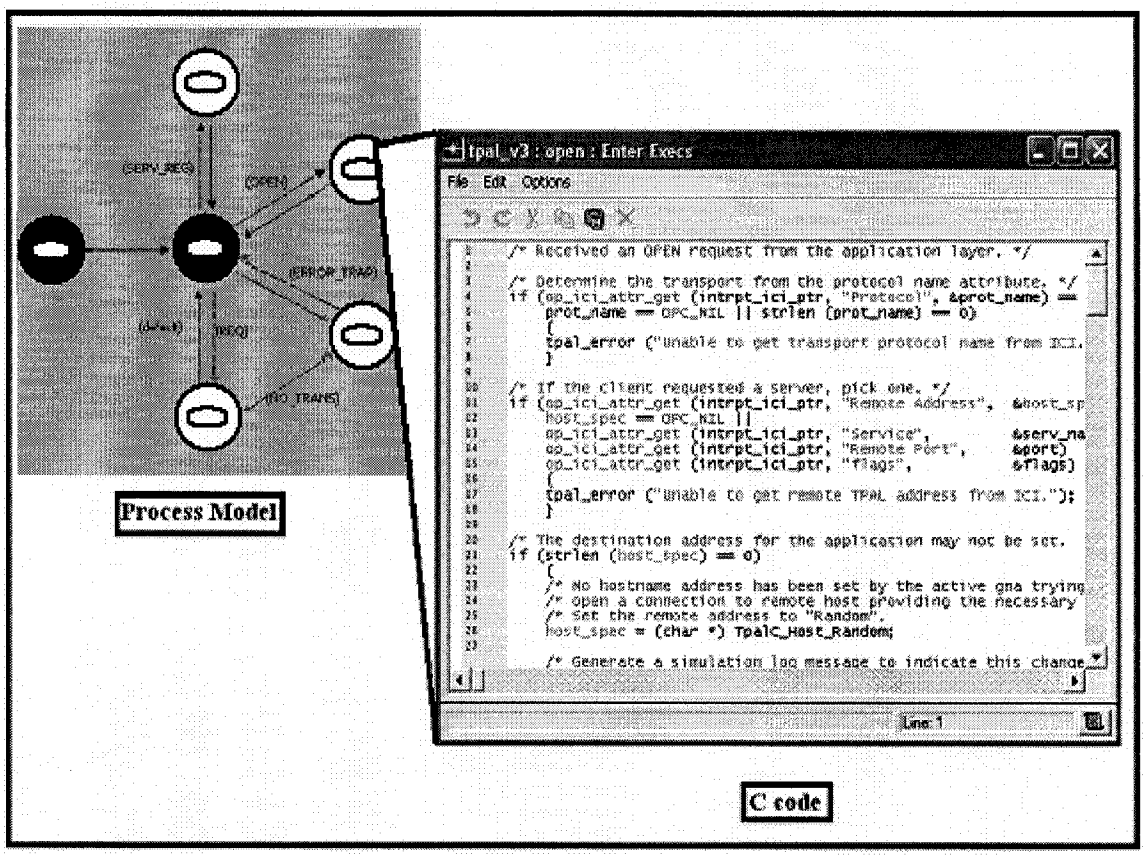

Figure A.3 An example of a process model and a C code that is located in the Enter Execs of one of the states.

Appendix B

Confidence Interval Computation

A confidence interval is used to quantify the uncertainty in any collected sample of data. It is defined as the estimated range of values within which a generated data lies with a specified probability. This probability is usually defined as 95%, which means that we have a confidence of 95% that the collected data lay in a certain interval (that is, the confidence interval). The end points of the confidence interval are known as the confidence limits. The confidence limits, and thus the confidence interval, for a sample data n that are normally distributed are computed as follows:

$$\mu \pm z\frac{\sigma}{\sqrt{n}} \qquad (1)$$

Where, μ is the mean value of n, σ is the standard deviation of n, and z is referred to as the significance level (described shortly). Equation 1 states that the confidence interval is centered at the mean value of the collected sample data.

The significance level z is used to specify the area under the normal distribution curve that corresponds to the desired confidence level. For example, refer to Figure B.1. To find the 95% confidence interval for the shown normal distribution, we need to exclude 5% of the total area under the curve from our computation. This means that we need to exclude 2.5% of the area on both sides of the mean μ. Then, we need to find the area that corresponds to 95% of the sample of data n. This area can be found using the z-table that is populated with the values of z (or areas) that correspond to the desired confidence level. A partial snapshot of the z-table is shown in Table B.1. To read z from this table, we firstly specify the

percentage of the sample data needed to achieve a 95% of confidence. This corresponds to 1-0.025 = 0.975. Thus, from the table we can see that z = 1.96 (once we locate the 0.975 in the table, we read the first two digits of z from the leftmost column. Then we get the third digit from the first row of the column where 0.975 is located).

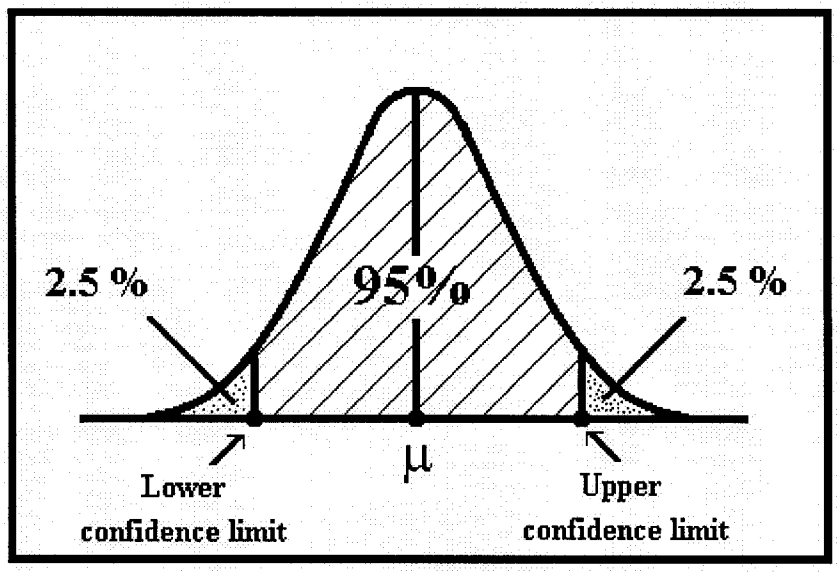

Figure B.1 An illustration of the normal distribution function of the sample data *n*. Only 95% of the area under the curve is considered to compute a confidence interval of 95% centered at the mean value μ.

z	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07
1.8	0.96407	0.96485	0.96562	0.96638	0.96712	0.96784	0.96856	0.96926
1.9	0.97128	0.97193	0.97257	0.97320	0.97381	0.97441	0.97500	0.97558
2	0.97725	0.97778	0.97831	0.97882	0.97932	0.97982	0.98030	0.98077

Table B.1 *z*-table