

WDM CROSS-PATH SWITCHING FOR  
LARGE-SCALE ATM SWITCHES

BY

JIN MAI

A THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF PHILOSOPHY

DIVISION OF INFORMATION ENGINEERING

THE CHINESE UNIVERSITY OF HONG KONG

JUNE 1998



# Acknowledgement

I would like to express my deepest gratitude to my supervisor Professor Tony T. Lee for his continuous support, guidance and encouragement, which make this two years study a fruitful and rewarding experience. I would also like to thank Professor Raymond W. H. Yeung for his kindly support and advices.

Special thanks to Professor Philip To, Mr. S. Y. Liew, C. H. Lam, Oo Tang and Raymond Lin for their continuous discussions and varies kinds of help.

# Abstract

Large-scale ATM switch is critical for ATM technology. Recently, a quasi-static routing scheme has been proposed by Lee and Lam [34] which uses predetermined periodical connection pattern in the central stage of 3-stage Clos network to assign the required bandwidth from each input module to output module. It has been shown to be able to handle multirate and multimedia traffic effectively in the Clos network.

In this thesis, we modify the scheme proposed by the above authors and fix the frame size to be one. Using the wavelength division multiplexing technology, we propose a new scheme using passive star couplers in the middle stage to assign the internal virtual paths between the input modules and output modules. Based on the concept of effective bandwidth, the QoS of connections on the virtual path level can be guaranteed, while the cell level QoS can be controlled at each individual input and output modules.

The WDM technique has been extensively studied for optical cross-connects and broadcast-and-select networks, but some technological challenges are faced when we use WDM technique for large scale ATM switches, the tuning speed and tuning range of the optical components are critical ones. Compare to the most current large scale ATM switches using optical star couplers as the core or



middle stage in Clos network, the WDM system used in our proposed scheme is operated at path level, but not cell level, which has low requirements on tuning speed of tunable transmitters or receivers.

Shared buffer memory switch has lower total memory needed and the flexibility of the memory management schemes. It can reach 100% throughput due to the separate logical queuing at the central memory. We use the shared buffer memory switches for example in our switch model to illustrate the local routing control under our WDM cross-path switching scheme.

For different sets of switch parameters, the tradeoffs between the switch complexity and the performance (mainly the call-blocking rate at the virtual path level) are evaluated. At last, two multicasting schemes are proposed and their performances are compared under different fanout distributions.

## 摘要

异步傳輸模式(ATM)技術的其中一個關鍵是大規模ATM交換機.最近 Lee和Lam [34]提出了一種半固定的選路方案,這種方案在三級Clos網絡的中間級使用預定的周期性的連接模式來分配從任一輸入模塊到輸出模塊之間所需的帶寬.這種方案已被顯示能有效的在Clos網絡中處理多速率和多媒体的業務.

在這篇論文里,我們修改了上述作者提出的交換方案并固定幀的大小為1.利用波分复用(WDM)技術,我們提出了一種新的交換方案.它使用通過式的星形耦合器作為中間級來分配從輸入模塊到輸出模塊之間的虛通路.基於有效帶寬的概念,各個連接在虛通路上的業務質量(QoS)可以被保證,而各信元的QoS可以在各個輸入輸出模塊里被控制.

WDM技術已被廣泛的研究以用來實現光的交叉連接,廣播和選擇網絡等.但用WDM技術來實現大規模ATM交換時將面對一些技術上的困難,其中關鍵的問題是光器件的調波速度和範圍.和現今大多數使用光星形耦合器來作為核心或是Clos網絡的中間級的大規模ATM交換機相比較,在我們提出的交換方案中,WDM系統的操作是基於虛通路而不是基於信元的,所以對可調波發射器或接收器的調波速度只有很低的要求.

共享記憶體交換機有較低的記憶體需求和靈活的記憶體管理方案.因為在記憶體內分開的邏輯排隊,它可以達到100%的吞吐量.我們以共享記憶體交換機為例來說明WDM交叉路徑交換方案下局部的路由.

對於不同的交換機參數,我們評估了交換機的複雜度和性能(主要是虛通路上的呼叫阻塞率).最後我們提出兩種點到多點通信方案,并在幾種不同fanout分布下比較了它們的性能.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background and Motivation . . . . .	1
1.2	Organization of the thesis . . . . .	8
<b>2</b>	<b>Principles of WDM Cross-Path Switch</b>	<b>11</b>
2.1	Principles of path scheduling . . . . .	11
2.2	Call setup and path rearrangement . . . . .	15
2.3	ABR control . . . . .	17
<b>3</b>	<b>Star coupler and WDM path scheduling</b>	<b>20</b>
3.1	Star coupler and other WDM ATM switches . . . . .	20
3.2	Two schemes of implementation . . . . .	22
<b>4</b>	<b>input/output modules and local routing</b>	<b>26</b>
4.1	Shared buffer memory switch . . . . .	26
4.2	local routing at input/output modules . . . . .	29
<b>5</b>	<b>Multicasting</b>	<b>32</b>
5.1	Two multicasting schemes . . . . .	32

5.2	Call blocking . . . . .	36
<b>6</b>	<b>Performance</b>	<b>37</b>
6.1	Introduction . . . . .	37
6.2	Switch complexity . . . . .	38
6.3	Speed up . . . . .	40
6.4	Two multicasting schemes . . . . .	41
<b>7</b>	<b>Switch Model and Operation</b>	<b>47</b>
<b>8</b>	<b>Conclusions</b>	<b>50</b>
<b>A</b>	<b>Effective bandwidth and QoS guarantee</b>	<b>52</b>
A.1	ATM service categories and QoS parameters . . . . .	52
A.2	Effective bandwidth for single source . . . . .	53
A.2.1	Markovian on/off source approach . . . . .	54
A.2.2	Leaky bucket regulated source . . . . .	55
A.3	Effective bandwidth for multiplexed sources . . . . .	60
A.3.1	Gaussian model approach . . . . .	60
	<b>Bibliography</b>	<b>62</b>

# List of Tables

5.1	Output modules look up table at the input module for scheme 1	34
5.2	Routing table and head translation at the output module for scheme 1 . . . . .	35
5.3	Routing table and head translation at the input module for scheme 2 . . . . .	35
6.1	Number of star couplers for different pair of $n/k$ . . . . .	39
6.2	Number of star couplers for different speed up factors . . . . .	40



# List of Figures

1.1	Three stage output buffered switch . . . . .	4
1.2	Correspondence between middle-stage connection pattern in a Clos network and edge-coloring of a regular bipartite graph . . .	5
1.3	Illustration of path scheduling for cross-path switch . . . . .	7
2.1	A three-stage Clos network model . . . . .	12
2.2	The connection matrix . . . . .	14
2.3	Searching process for path rearrangement . . . . .	17
3.1	Star coupler . . . . .	20
3.2	Connection matrix and the corresponding tuning table of receivers	23
3.3	Unique wavelength lasers and tunable receivers . . . . .	24
3.4	Connection matrix and the corresponding tuning table of transmitters . . . . .	25
3.5	Tunable lasers and unique wavelength receivers . . . . .	25
4.1	Shared buffer memory switch . . . . .	27
4.2	Buffer management in a shared memory switch . . . . .	28
4.3	Connection matrix and local routing at the input modules . . .	30

5.1	Multicasting schemes for WDM cross-path switch . . . . .	33
6.1	Call blocking rate for different effective bandwidth . . . . .	42
6.2	Call blocking rate for different fanout distributions in two multicasting schemes . . . . .	43
6.3	Call blocking rate vs. number of star couplers for two multicasting schemes with loading rate 0.9 . . . . .	44
6.4	Call blocking rate vs. number of star couplers for two multicasting schemes with loading rate 0.8 . . . . .	46
7.1	Proposed switch architecture . . . . .	48
A.1	Leaky bucket for traffic shaping . . . . .	56
A.2	Leaky bucket regulated periodic on off source . . . . .	57
A.3	Maximum queue length for a greedy leaky bucket regulated traffic	58
A.4	Worst case delay probability for a leaky bucket regulated traffic	59



# Chapter 1

## Introduction

### 1.1 Background and Motivation

Computer and communication applications are on the rise with a variety of services, such as the World Wide Web, video conferencing, video on demand, and high-definition television (HDTV), which all require tremendous amounts of capacity and resources for high-quality transmissions. Several types of high-speed/high-bandwidth networks exist. Among them, asynchronous transfer mode (ATM) technology is proving to be one of the better technical and commercial solutions, and appears to be able to satisfy this increased demand for heterogeneous communications and dramatically increased bandwidth. Emerging high-speed networks are expected to integrate a wide variety of applications with different traffic characteristics and quality of service (QoS) requirements.

The traffics can be categorized by their QoS requirements briefly as follows:

1. the sources have low cell delay requirements and low cell loss ratio, like interactive video.

2. those are delay-sensitive but can tolerate a moderate cell loss, like telephone.
3. applications are not delay-sensitive but ask for accurate transfer, like interactive data and network computing.
4. the remaining sources which have very low delay requirement like file transfer.

With so many different characteristic and various data communications, multirate, higher bandwidth needed, it is a huge challenge for broadband network to integrate all these traffics.

Circuit switch using either time division switching or space division switching or the combination of two to setup an “circuit” between the end user and its destination. The delay is determined by the propagation delay on the transmission links and the processing time in the switches, which in theory is fixed and very small. The circuit switch is very inflexible, since the time slots giving to a specific connection is fixed, and the bandwidth is also fixed to be 64Kbps or its times.

Packet switch is wildly used in the computer networks. The packets entering the switch compete for switch resources (memory and processing time) equally (or with very limited priority control). Packet switch is very efficient for low speed data transfer, but QoS of individual traffics will be difficult to guarantee.

The Asynchronous Transfer Mode (ATM) uses relatively small fixed length (53 bytes) cells, and it is operated in a connection-oriented mode. ATM does not only allow multiplexed traffic for achieving high utilization of network resources, but also can promise satisfactory diverse QoS guarantee. When a call setup

request arrives to network node (ATM switch), the QoS parameters will be negotiated by the call and the network node and will be guaranteed in the further transmission. (See appendix A for various service categories and QoS parameters for ATM traffics).

One way to achieve the QoS guarantee is to reserve a certain amount of network resources to certain connections. Effective bandwidth of a connection (or a set of connections) is the sufficient bandwidth needed to guarantee the QoS requirements of each connection. The concept of effective bandwidth is proved to be adequate and has been extensively studied. There are many approaches, like using Markovian on/off model [3] (see appendix A.1.1), using fluid flow model for leaky bucket regulated traffics [18] (see appendix A.1.2), using large-deviations asymptotic analysis [25], [27]. The Gaussian model's approach is a simple but useful method [22] (see Appendix A.2.1).

The three-stage Clos network has been extensively studied as a framework for large-scale ATM switches. Most of them are output buffered switches [41, 45], as shown in the figure 1.1.

There are mainly two different schemes. One is dynamic routing [14, 9]. The first stage distributes arriving cells across all switches in the middle stage to balance the load. The second and the third stages route cells to the output, using an output port number inserted in the cell header by the input port processor. One issue with dynamic routing is that cells can get out of order as they pass through the system. This requires the addition of a re-sequencing buffer to each output port processor, to restore the correct cell ordering, which will introduce some added delay, increasing the minimum latency of the system. Another scheme is to use static routing [41, 43, 30]. In static routing networks, all cells



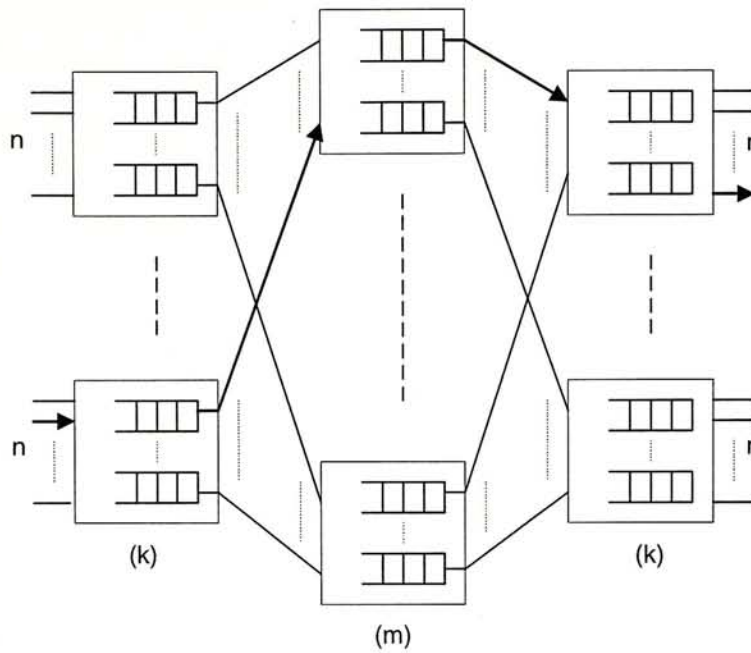


Figure 1.1: Three stage output buffered switch

in a given virtual circuit are constrained to follow the same path from the first stage to the third stage. When a new virtual circuit is to be added from some input port to some output port of a static routing network, the control processor managing the switching system must find some path through the network with sufficient unused bandwidth on each of its links to accommodate the new virtual circuit. For a three stage Clos network with module size  $n$ , minimum allowed virtual circuit bandwidth  $b$  and maximum allowed virtual circuit bandwidth  $B$ , the number of central modules needed to avoid call blocking satisfies the following inequality

$$m \geq 2 \max_{b \leq \omega \leq B} \left\lfloor \frac{n - \omega}{\max\{1 - \omega, b\}} \right\rfloor + 1$$

This number will go very large with larger  $B$  or smaller  $b$ . For example, with module size  $n=32$ , maximum virtual circuit bandwidth  $B = 0.3$  link speed, the number of central modules goes to three times of module size [41]. Virtual

circuit blocking analyses for this class of systems can be found in [44, 37, 38]

Path switching is a quasi-static routing scheme. It uses a predetermined periodical connection pattern in the middle stage, which provides the “virtual paths” between any pair of input and output modules. It is a compromise of the static scheme and the dynamic scheme, and since the routing at the central module is predetermined, the complexity of central modules can be much simplified.

The basic concepts of path switch is to provide the sufficient bandwidth (virtual paths) between the input and output modules to guarantee the QoS of individual connections, based on the concept of effective bandwidth. For a three stage Clos network, by considering the input and output modules as nodes, then the connection pattern in the middle stage of the Clos network can be represented by the edge-coloring of a bipartite graph as illustrated in Figure 1.2 [34].

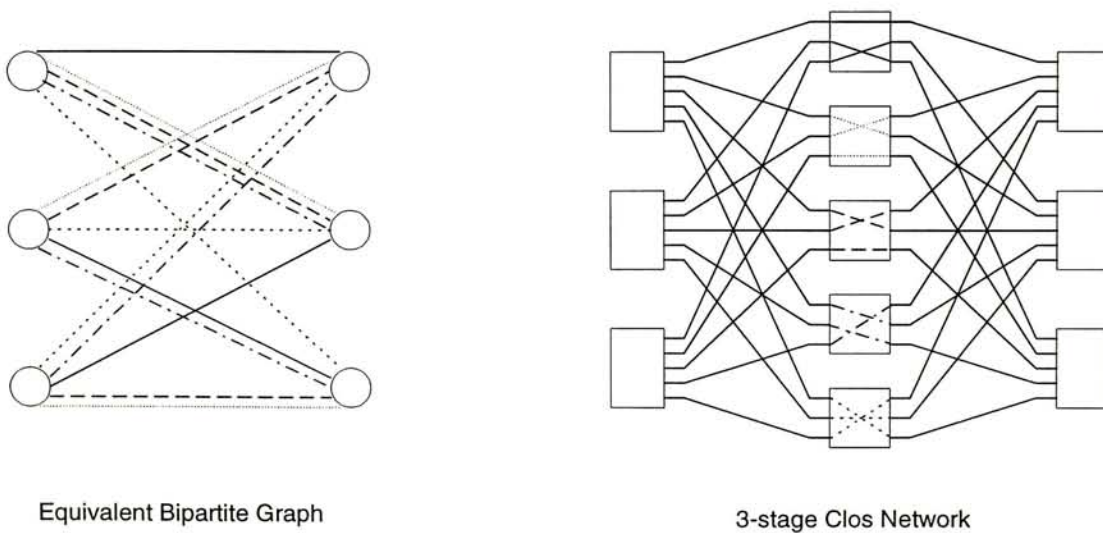


Figure 1.2: Correspondence between middle-stage connection pattern in a Clos network and edge-coloring of a regular bipartite graph

The number of paths linking the left side node  $i$  and right side node  $j$  denotes the number of cells which can be transferred from the input module  $i$  to the output module  $j$  at one time slot, which is realized by the connection pattern in the central module of the corresponding 3-stage Clos network.

In circuit switch, a “circuit” is setup either by time division or by space division and reserved exclusively for that connection during the whole call sustaining time. In dynamic routing, the routing for every cell will be calculated independently, i.e. connection pattern will be calculated slot by slot, without using the information of the characteristics of each virtual circuits we know at the call setup time. We using finite number of different connection patterns in the middle stage repeatedly, as a compromise of above two extreme schemes. Let  $\Lambda_{ij}$  is the traffic loading matrix, where the entries  $\lambda_{ij}$  is the total traffic from the input module  $i$  to the output module  $j$ . Then for any given  $\Lambda_{ij}$ , if  $\sum_i \Lambda_{ij} < n \leq m$ , and  $\sum_j \Lambda_{ij} < n \leq m$ , we can always find a finite number  $f$  of regular bipartite multigraphs such that

$$\frac{\sum_{t=1}^f e_{ij}(t)}{f} > \Lambda_{ij}, \quad (1.1)$$

where  $e_{ij}(t)$  is the number of edges from input module  $i$  to output module  $j$  of the corresponding bipartite multigraph in the  $t$ -th time slot.

The process consists of two steps, the capacity allocation and the route assignment. The capacity allocation is to find the capacities  $C_{ij} \leq \lambda_{ij}$  by optimizing some objective function subject to  $\sum_i C_{ij} = \sum_j C_{ij} = m$ . The choice of objective function depends on which traffic characteristic is emphasized, like cell delay or buffer overflow probability. Next we choose a large enough integer  $f$  such that  $C_{ij}f$  is an integer for all  $i, j$ . the new capacity matrix  $fC_{ij}$  represents



a regular bipartite graph with degree  $fm$ , it can be edge colored by  $fm$  different colors. Assume  $a \in \{0, 1, \dots, fm - 1\}$  is the color assignment of one edge connecting input module  $i$  and output module  $j$ . Divided it by  $f$ , we have

$$a = r \cdot f + t$$

where  $r \in \{0, 1, \dots, m - 1\}$  and  $t \in \{0, 1, \dots, f - 1\}$ . We know that for any  $a$ , the pair  $(r, t)$  is unique determined and conversely, any pair  $(r, t)$  corresponding the unique  $a \in \{0, 1, \dots, fm - 1\}$ . In other words, there is a one to one corresponds for color assignment  $a$  and any pair  $(r, t)$ . We can then use pair  $(r, t)$  to represent a color assignment, which we can interpret as the central module  $r$  in the time slot  $t$ . Therefore, we have edge-colorings with degree  $m$  for  $f$  regular bipartite graphs. This process is called the time-space interleaving [34].

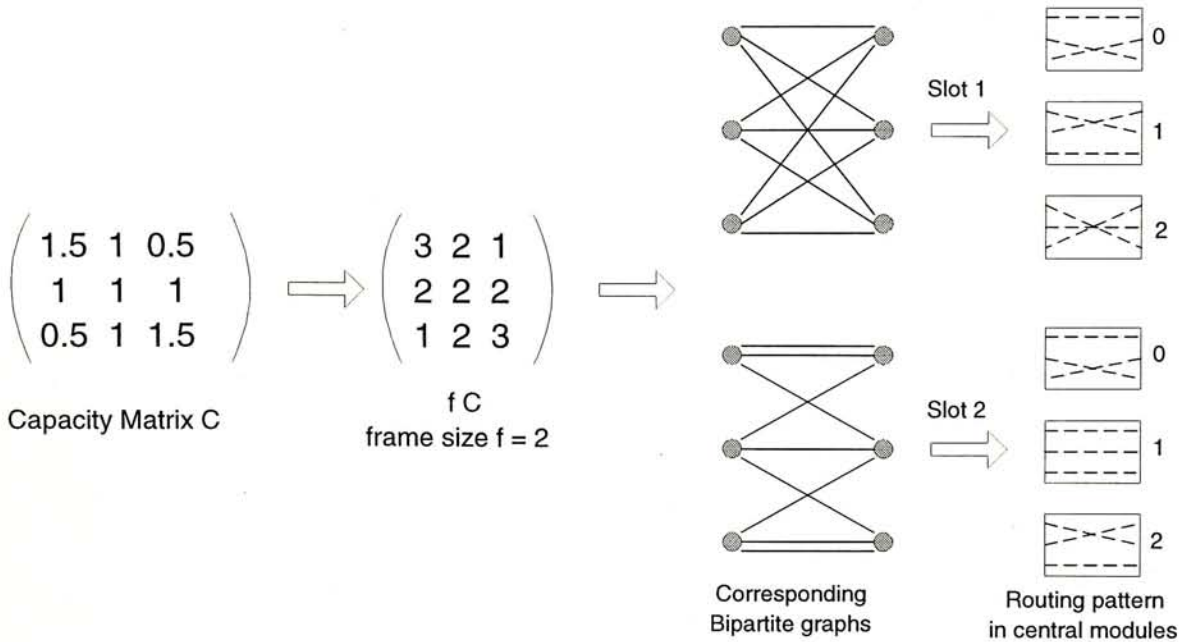


Figure 1.3: Illustration of path scheduling for cross-path switch



The capacity requirement 1.1 can be satisfied if the system provides connections repeatedly according to the coloring of these  $f$  bipartite multigraphs, and these finite amount of routing information can be stored in the local memory of each input module to avoid the slot-by-slot computation of route assignments.

Wavelength division multiplexing (WDM) technique has been extensively studied for optical cross-connects [8, 12, 20]. It can provide a simple cross-connect structure to implement a broadband cross-connect system. Some technological challenges are faced when we use WDM technique for large-scale ATM switches [10, 26, 12], the tuning speed and tuning range of the optical components are critical ones. In the cross-path switching introduced above, a predetermined cross-connect pattern between the first and third stage of a Clos network is needed for a large-scale ATM switch. Since the connection pattern is changed slot by slot, this requires high tuning speed of optical components for a high speed ATM switch (OC3 link speed yields  $2.7\mu s$  for a cell time slot).

By fixing the frame size to be 1 at the above path scheduling scheme, the connection pattern need not to be changed every time slot, and the same connection pattern remains unchanged until the call setup time and the traffic loading is changed significantly. These properties make it suitable to be implemented by the wavelength division multiplexing technology.

## 1.2 Organization of the thesis

In this thesis, we will propose a new switch model which uses shared buffer memory switches as input and output modules and use passive star couplers interconnecting them to provide the necessary capacities of virtual paths between

them. The remaining content of this paper is organized as follows.

Chapter 2 gives the principles of WDM cross-path scheduling, call setup conditions and path rearrangement after the traffic characteristic has been changed significantly and the old virtual path assignment can not accommodate the total traffic anymore. At last of chapter 2, ABR flow control is briefly discussed for efficient access the spare virtual path capacities.

In chapter 3, we introduced the model of star-couplers, other large-scale ATM switches using WDM systems are briefly discussed. Then two possible implementations using star-couplers for our switch model are proposed and the corresponding tuning tables at the transmitters or receivers are given.

In chapter 4, the pros and cons of the shared buffer memory switches are discussed, for which we have chosen as our input and output module switches. The local routings at the input and output modules are also discussed.

Chapter 5 introduced two multicasting schemes for WDM cross-path switching. Cell manipulation at the input and output modules are also discussed for two multicasting schemes respectively. And we found that path level call blocking is possible for multicasting connections.

In chapter 6, we did some performance evaluations for our switch model. First, the central stage switch complexity is compared for different pair of  $n/k$  in our switch model. Then we introduced the speedup at the central WDM system and different speed up factor are compared. At last, we compared the two multicasting schemes with different effective bandwidths of calls, fanout distributions and under different loading rates.

Chapter 7 is a discussion of the proposed switch model and its operation with the switch parameters we have chosen at the chapter 6. Chapter 8 will

give conclusions. An Appendix is given for varies QoS requirements for ATM service categories and the calculations of effective bandwidths using different traffic models.

## Chapter 2

# Principles of WDM Cross-Path Switch

### 2.1 Principles of path scheduling

As shown in the figure 2.1, is a three-stage Clos network model. The first stage consists  $k$  input modules, each has  $n$  inlets and  $m$  outlets, the middle stage is  $m$  central modules size  $k$  by  $k$ , the last stage is composed of  $k$  output modules, each has  $m$  inlets and  $n$  outlets. The total size of the switch is  $N$  by  $N$ , where  $N = n \times k$ . In a three-stage Clos network, each input and output module has one and only one link to each central module, and the number of possible different paths between each pair of input and output module is the number of central modules connecting them.

The three-stage Clos network has been extensively studied as a suitable model for large scale ATM switches [43, 41, 45]. The main concern is how



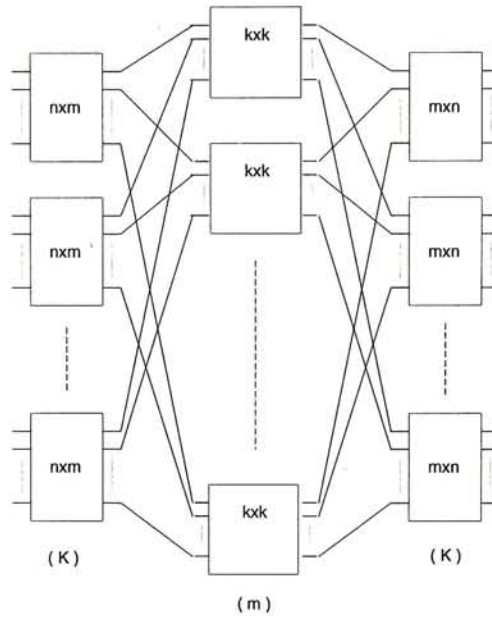


Figure 2.1: A three-stage Clos network model

to assign central modules for each pair of input and output modules, in other words, the inter-connection pattern. For input module  $i$  and output module  $j$ , let  $\lambda_{ij}$  be the effective bandwidth of the aggregate traffic between them (traffic arriving at input module  $i$  and destined to output module  $j$ ). We call matrix  $\Lambda = (\lambda_{ij})_{k \times k}$  the traffic loading matrix of the switch [34], then the main problem is how to assign the capacity to each virtual path between pair of input output modules.

Before we deduce the requirements and conditions on the traffic loading matrix  $\Lambda$ , we define  $\alpha_{uv}$  be the effective bandwidth of aggregate traffic arriving at inlet  $u$  and destined to outlet  $v$ . Let the link capacity be 1, then we have the following link capacity constraints:

1. input link capacity constraints:

$$\sum_u \alpha_{uv} \leq 1, \forall 0 \leq v \leq N - 1; \quad (2.1)$$

2. output link capacity constraints:

$$\sum_v \alpha_{uv} \leq 1, \forall 0 \leq u \leq N - 1. \quad (2.2)$$

Since the total traffic leaving from each input module can not exceed  $n$ , and similarly, the total traffic coming to each output module will be at most  $n$ , here we consider the unicast connections only, then we have the following

1. input module capacity constraints:

$$\sum_j (\lambda_{ij}) < n, \forall 0 \leq i \leq k - 1; \quad (2.3)$$

2. output module capacity constraints:

$$\sum_i (\lambda_{ij}) < n, \forall 0 \leq j \leq k - 1; \quad (2.4)$$

If we fix the frame size [34] to be 1, for any traffic  $\lambda_{ij}$ , we need  $\lceil \lambda_{ij} \rceil$  paths to transmit them from input module  $i$  to output module  $j$ . We call the matrix  $P$  with entries  $P_{ij} = \lceil \lambda_{ij} \rceil$  the virtual path assignment matrix. Since each  $\lceil \lambda_{ij} \rceil < \lambda_{ij} + 1$ , after round up, we need

$$\sum_j P_{ij} = \sum_j \lceil \lambda_{ij} \rceil \leq n + k - 1$$

virtual paths leaving from any input module  $i$ . At the same time, we need

$$\sum_i P_{ij} = \sum_i \lceil \lambda_{ij} \rceil \leq n + k - 1$$

virtual paths going to each output module  $j$ . Let  $m = n + k - 1$ , then the virtual paths connecting input and output modules can be represented by a bipartite

graph with the degree at each vertex less than or equal to  $m$ . Using edge coloring of regular bipartite graph, we can always color all required paths using  $m$  or less colors [34]. Let  $c_0, c_1, \dots, c_{m-1}$  be  $m$  different colors (central modules). Then we get the connection matrix  $C$ , with each entry  $C_{ij}$  associated with the colors used to color the paths between input module  $i$  and output module  $j$ .

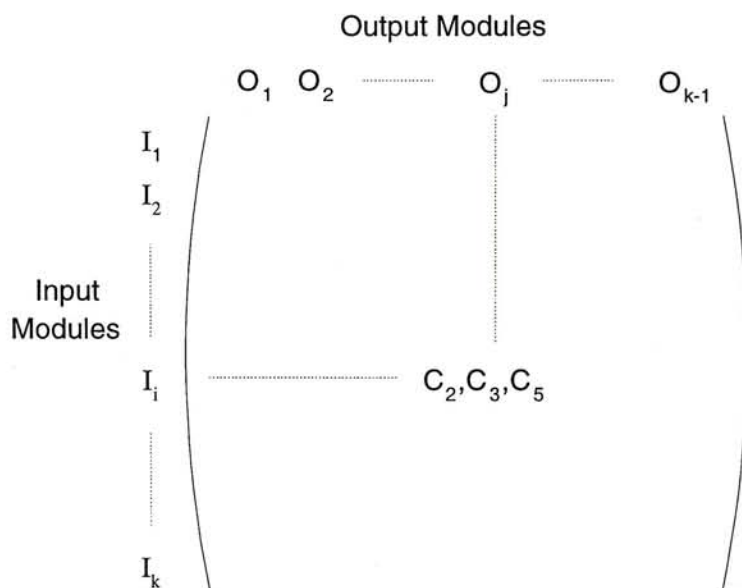


Figure 2.2: The connection matrix

As shown in figure 2.2, the connection matrix is size  $k$  by  $k$ , the entry at the  $i$ -th row and  $j$ -th column is the central modules connecting the input module  $i$  and the output module  $j$ , there are three such central modules  $c_1, c_2, c_4$  as shown in the figure. That also means the virtual path between the input module  $i$  and output module  $j$  has total capacity 3.

The number  $m = n + k - 1$  can be further reduced if we allow a small call blocking rate, as we will discuss later.



## 2.2 Call setup and path rearrangement

Based on the calculation of effective bandwidth of each virtual path  $\lambda_{ij}$ , we conclude that the path level QoS can be guaranteed by providing the effective bandwidth to each virtual path (see Appendix A). As for cell level QoS requirements, since the cell queuing and buffer overflow only happens at the input and output modules respectively, the cell level QoS will be controlled by each involved input and output modules and many call admission schemes [19, 7, 40] and service schemes [39, 42] can be adopted. On path level, the following capacity constraints should be satisfied before a set of calls can be setup:

1. input/output link capacity constraint: the bandwidth required to accommodate the aggregate traffic on each input and each output link cannot exceed the link capacity 2.1 and 2.2.
2. Virtual path capacity constraint : the total virtual paths needed to interconnect the input and output modules can not exceed the number of central modules  $m$

$$\begin{cases} \sum_j P_{ij} \leq m, & \forall 0 \leq i \leq k-1; \\ \sum_i P_{ij} \leq m, & \forall 0 \leq j \leq k-1. \end{cases} \quad (2.5)$$

For switches accepting only unicast connections, assuming that the calls are not external blocked, i.e., the input/output link capacity constraints are satisfied. By summing the total effective bandwidth of traffics leaving each input module, we have

$$\sum_j \lambda_{ij} = \sum_{u \in I_i, v} \alpha_{uv} < n$$

for any input module  $I_i$ . similarly, we have

$$\sum_i \lambda_{ij} = \sum_{u,v \in O_j} \alpha_{uv} < n$$

for any output module  $O_j$ . Therefore, the calls will not be internal blocked if the number of central modules  $m = n + k - 1$ . For multicasting case, these statements will no longer be true, hence the internal call blocking is possible, which will be discussed later.

Now, when a new call arrives, assume its effective bandwidth is  $\delta$ , travels from input module  $I_i$  to output module  $O_j$ . The input and output link capacity constraints are first checked to ensure it is acceptable (this can be done distributively at each input and output module). The new effective bandwidth of the aggregate traffic from the  $I_i$  to  $O_j$ ,  $\lambda'_{ij}$ , should be recalculated to accommodate the new call. Then the virtual path capacity constraints 2.5 should be satisfied before the call can be setup. Otherwise, the connection is rejected. If the connection is accepted, then there are two possibilities: if  $\lceil \lambda'_{ij} \rceil = \lceil \lambda_{ij} \rceil$ , this means the current virtual paths between input module  $i$  and output module  $j$  can accommodate the new call, hence, no path rearrangement is needed; if  $\lceil \lambda'_{ij} \rceil = \lceil \lambda_{ij} \rceil + 1$ , then a new path between input module  $i$  and output module  $j$  is required to be setup before accepting the new call.

Let  $P' = (P'_{ij})$  be the new path assignment matrix after the new call arrived. Then either it is not changed or one new path should be setup. On later case, we know it is just a path rearrangement problem in the three-stage Clos network for circuit switching, where module size  $m$  (since  $\sum_i P_{ij} \leq m$ ), number of input/output modules  $k$ . With number of central modules  $m$ , we know that it is rearrangeably non-blocking. Suppose one new path need to be setup between the

input module  $i$  to the output module  $j$ , then the standard path rearrangement algorithm in circuit switching can be used to setup the new path, as illustrated in the figure 2.3. That is: first, find a pair of central modules that do not occur in the  $i$ -th row and  $j$ -th column of our connection matrix respectively, say,  $c_1$  and  $c_2$ . Then replace  $c_1$  with  $c_2$ , and search for  $c_2$  in  $i$ -th row, and place it with  $c_1$ , then search for  $c_1$  in that column and so on. At last, the rearrangement chain will stop and we find a set of rearrangements needed.

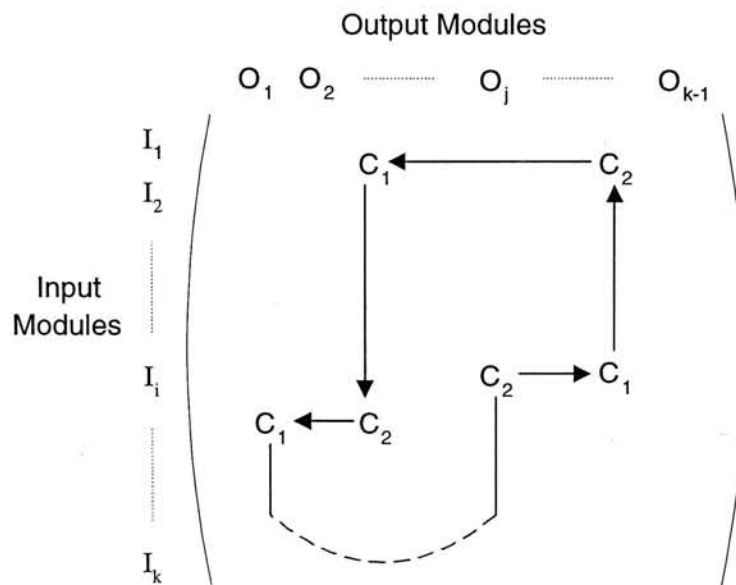


Figure 2.3: Searching process for path rearrangement

## 2.3 ABR control

Available Bit Rate (ABR) is a service category for which the transfer characteristics provided by the network may change subsequent to connection establishment. The ABR service does not require bounding the delay or the delay variation experienced by a given connection. On the establishment of an ABR



connection, the end-system shall specify to the network both a maximum and the minimum usable bandwidth, which are called as peak cell rate (PCR), and the minimum cell rate (MCR), respectively. The MCR may be specified as zero. The bandwidth available from the network may vary, but shall not become less than MCR.

A flow control mechanism is specified [4] for ABR traffics. The feedback of the changing of the network traffic characteristics is conveyed to the source through the forward and backward Resource Management Cells (RM-cells). Through RM-cells, the network nodes will fairly inform the end-system of ABR connections about the congestion and to increase by a certain amount of cell rate, called rate increase factor (RIF) or to decrease by a certain amount of cell rate, called rate decrease factor (RDF), which are agreed by the end-system and the network node at the call setup time. The goal of the flow control for ABR traffics is to provide rapid access to unused network bandwidth at up to PCR, whenever the network bandwidth is available.

In our switch model, since MCR of the ABR traffic has to be guaranteed by the switch, it should be accommodated in the total traffic on the corresponding virtual path. While the spare capacities  $s_{ij}$  for virtual path from input module  $i$  to output module  $j$  satisfies

$$\begin{cases} \sum_j s_{ij} < m - \sum_j \lambda_{ij}, & \forall 0 \leq i \leq k - 1; \\ \sum_i s_{ij} < m - \sum_i \lambda_{ij}, & \forall 0 \leq j \leq k - 1. \end{cases} \quad (2.6)$$

While the control system of the switch should find a matrix  $S = (s_{ij})$  satisfies 2.6 to fairly distribute the spare virtual path capacities to the ABR traffics between any pair of input output modules, with consideration of their MBR,

current cell rate, PCR, and rate increase factor and rate decrease factor. These will be left for further study.

# Chapter 3

## Star coupler and WDM path scheduling

### 3.1 Star coupler and other WDM ATM switches

In our switch model, we use star couplers as the central modules to interconnect input modules and output modules and provide the necessary virtual paths between them.

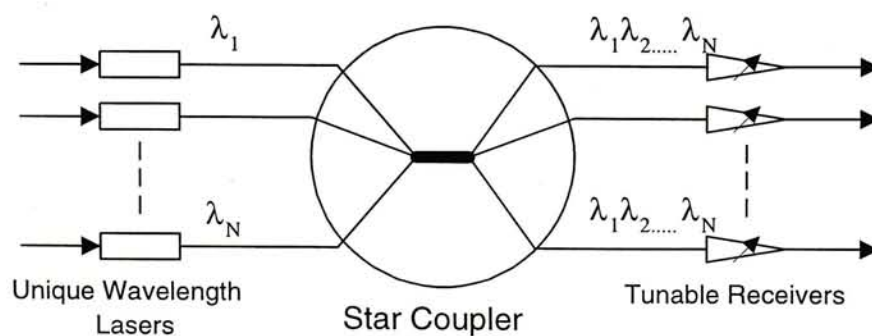


Figure 3.1: Star coupler

As shown in figure 3.1, the passive star-coupler is a broadcast-and-select network. In this network, all inputs (with different wavelength) are combined in a star coupler and broadcast to all outputs. There are some different ways to setup the interconnections.

1. Tunable input lasers and fixed wavelength receivers. A connection is setup by tuning the input laser to the wavelength of the receiver it is destined to. This is basically a space-division switch in function. Output port contention exists for this network, so that a contention resolution scheme should be provided when operating.
2. Fixed wavelength input lasers and tunable receivers. By tuning the receivers to the wavelength of the input channel it want to receive, interconnections can be setup. This network supports multicasting since more than one output receivers can be tuned to receive one input signal at the same time.
3. Both the transmitters and receivers are made tunable. The number of wavelength needed to setup the interconnection pattern can be reduced in this network if not all  $N$  inputs are destined to different one of  $N$  outputs respectively, but a more complex algorithm needed to use this property.

WDM technique and star-couplers have been extensively studied for optical cross-connects and broadcast-and-select networks [8, 20, 17]. Star-couplers are also used in some packet switch models [2, 33], where tunable lasers or tunable receivers are necessary for setting up specific inter-connection patterns. There are also many large scale ATM switch models using star-couplers as core or



interconnecting [10, 12, 26]. Some technological challenges are faced when we use WDM technique for large scale ATM switches, the tuning speed and tuning range of the optical components are critical ones. OC3, for example, has one cell time about  $2.7\mu\text{s}$ . fast tuning times of lasers, on the order of a few nanoseconds having been measured, but the tuning range is limited to 10nm or so [13]. On the other hand, wide tuning range is available but the tuning speed is on the order of microseconds, which makes it not possible to transmit high speed links on the cell level.

## 3.2 Two schemes of implementation

Our switch model requires  $m$   $k$  by  $k$  star couplers. Corresponding to the connection matrix  $C$  we got at chapter 2, there are two possible implementations using star couplers: tunable laser at input modules and fixed wavelength receivers at output modules or conversely, fixed wavelength transmitter at input modules and tunable receiver at output modules. For the first scheme, the receivers at the output module  $j$  is fixed at  $\lambda_j$  and the transmitters at the input module  $i$  is tuned according to the  $i$ th row of the connection matrix.

As shown in the figure 3.2 and 3.3, the entry at the  $i$ -th row and  $j$ -th column at the connection matrix is the central modules through which the virtual path should be setup between the input module  $i$  and the output module  $j$ . Hence, the receivers at the output module  $j$  connected to those star couplers should be tuned to the wavelength  $\lambda_i$ , which is the wavelength used for output signals in all the transmitters at the input module  $i$ . Such that an optical path between these pair of input output modules could be setup. For example, the entry

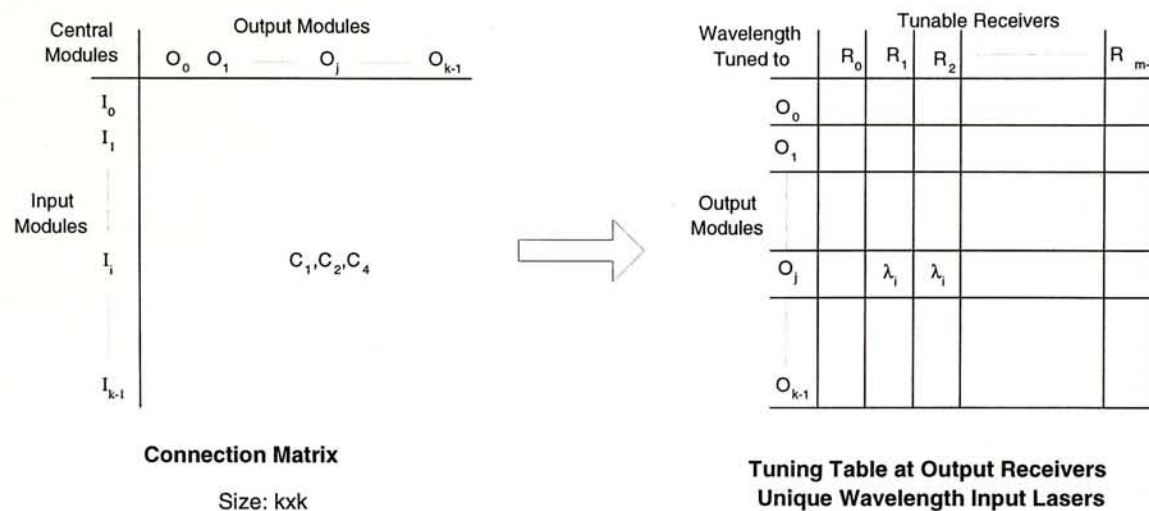


Figure 3.2: Connection matrix and the corresponding tuning table of receivers

$C_{ij} = \{c_1, c_2, c_4\}$  in the connection matrix  $C$ , then the receivers  $R_1, R_2, R_4$  should be tuned to  $\lambda_i$  as in the tuning table at the output module  $O_j$ .

On the other hand, we can also fix the all transmitters at the input module  $i$  to  $\lambda_i$ , and tune the receivers at the output module  $j$  according to the  $j$ th column of the connection matrix to setup the needed internal connection paths, see figure 3.4 and 3.5.

As shown in the figures, the entry at the  $i$ -th row and  $j$ -th column in the connection matrix  $C_{ij} = \{c_1, c_2, c_4\}$  same as in the above example. Thus the tunable transmitters  $T_1, T_2, T_4$  at the input module  $I_i$  should be tuned to  $\lambda_j$ , which is the fixed wavelength used at the receivers at the output module  $O_j$ .

As shown in the last chapter, our path assignment scheme is to provide an integer number of sufficient virtual paths between any pair of input and output modules and the connection pattern should be renewed only when a path rearrangement is needed, which only happens at the call setup time, and the traffic characteristic is changed significantly. That means the tuning (either

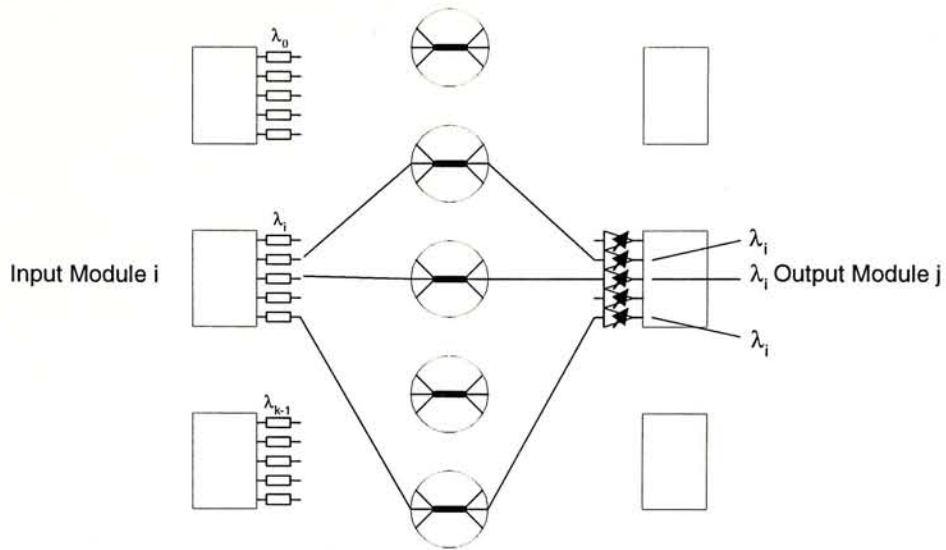


Figure 3.3: Unique wavelength lasers and tunable receivers

at the receivers in the output modules or the transmitters in the input modules) only happens at the call setup time, hence the magnitude of microsecond of tuning speed of these optical components will just satisfy the operation of our switch.

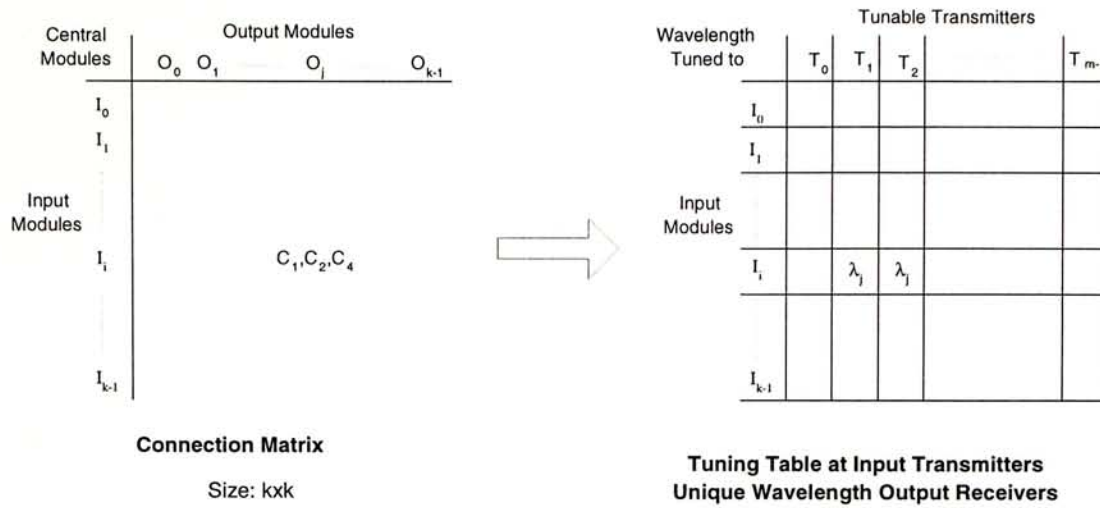


Figure 3.4: Connection matrix and the corresponding tuning table of transmitters

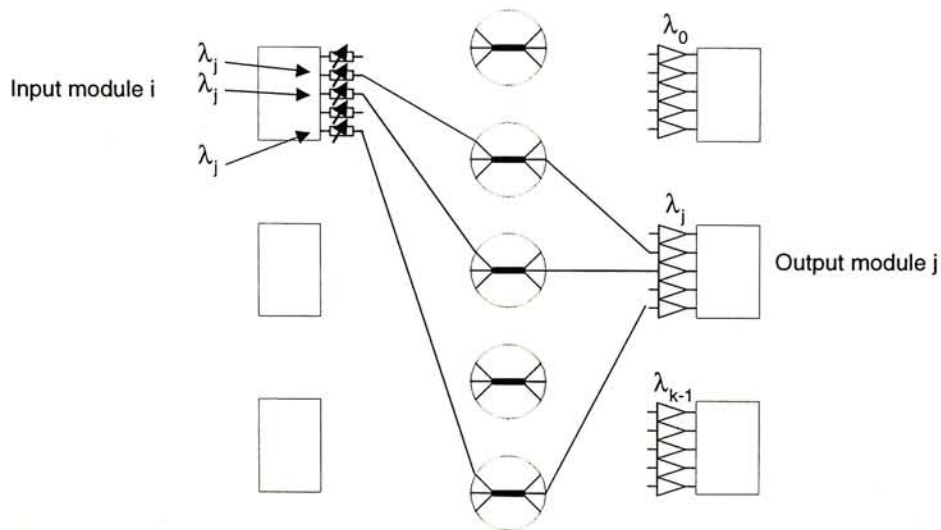


Figure 3.5: Tunable lasers and unique wavelength receivers



# Chapter 4

## input/output modules and local routing

### 4.1 Shared buffer memory switch

We use shared buffer memory switch as the input module in our switch model. Figure 4.1 illustrates the basic structure of a shared buffer memory switch. It consists of a single memory shared and accessed by all input and output links and managed by a central controller. In every time slot, the cells arriving on all input links are converted from serial to parallel form, and written sequentially to a dual port Random Access Memory. The cells inside the memory are logically organized as separate queues, one for each output link (some implementations maintain four logical queues, CBR and rt-VBR, nrt-VBR, ABR and UBR respectively, for each output port to satisfy different QoS requirements, or furthermore one logical queue per each virtual connection). Outgoing cells are



de-multiplexed at the same time slot, from the logical queues in the memory. An output stream of cells is formed by retrieving cells sequentially according to the output links (some service schema, e.g., round-robin, or priority control should be used when there are more than one logical queues for one output port to provide QoS), and is de-multiplexed to the outputs and converted from parallel to serial form.

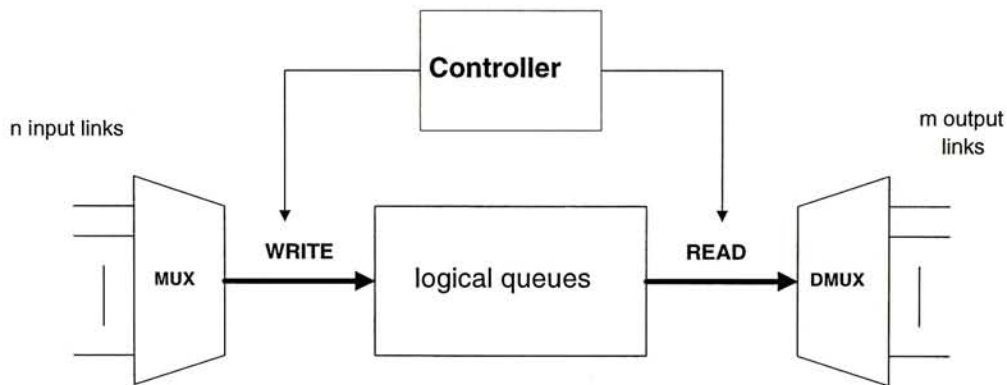


Figure 4.1: Shared buffer memory switch

The shared memory in the switch is controlled by a central controller. Each logical queue of cells is pointed by a chain of address pointers, as shown in figure 4.2. The chain is ended by a pointer to an empty cell, which will be the address to write the next input cell of this logical queue. When the first cell in the logical queue is read out to its output link according to the service scheme, The starting address of this logical queue will be moved to the address of the next cell of this queue. The shared buffer memory will largely decrease the memory needed for buffering cells in switches [30].

CNET's Prelude switch [16] was one of the earliest prototypes of this technique. Hitachi's shared buffer switch [30, 29] has been frequently referred to as an example of this class of switches in the literature.

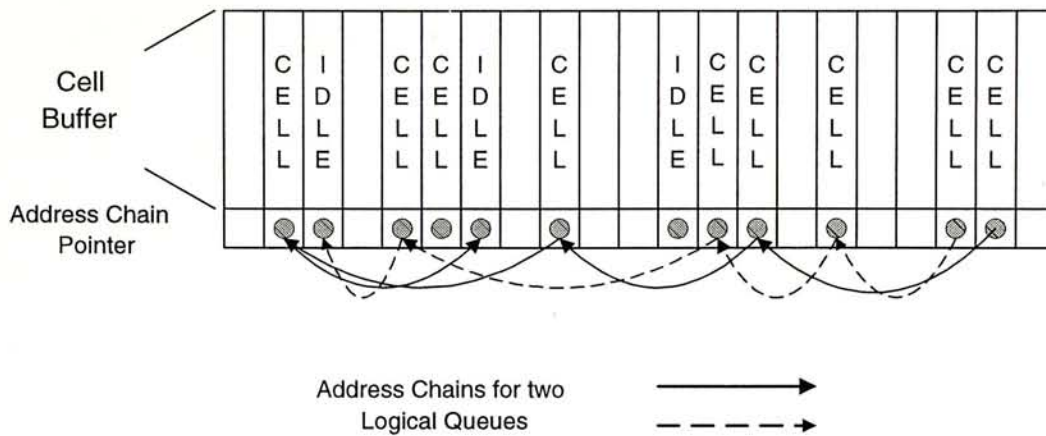


Figure 4.2: Buffer management in a shared memory switch

Compare to the space-division switch, the shared buffer memory switch has the privilege of reducing the total memory needed to buffer the cells and the flexibility of the memory management schemes. Various service schemes (Round-Robin, Deficit Round-Robin [42], Generalized Processor Sharing (GPS) [39]) can also be flexibly chosen to guarantee the QoS of individual connections. Since cells destined to different output port are read out sequentially. No contention will occur at the output ports, the switch can reach 100% throughput.

The main restriction faced by the shared buffer memory switch is the speed of the fast memory access. Since  $n$  cells should be written to the shared buffer and  $m$  cells should be read out to their outlets in one slot time, which requires  $n + m$  time speed up then the external link speed. One way to increase the switch size is to use parallel memory access technology. For 32x32 switch with link speed OC3, if we use 32-bit parallel memory access, that requires 6.4ns for one write and one read access. If we use 64-bit parallel memory access, the same memory access speed will be needed for a 64x64 OC3 shared memory switch. If we go to extreme, to read and write the whole cell in parallel, then 128x128 with OC3 can be achieved by memory access time around 11ns [12].

## 4.2 local routing at input/output modules

Due to the path connecting the input and output modules in our switch model are "virtual", the actual physical path will change following the change of the path assignment matrix and the connection matrix. That means even cells belonging to the same logical queue in the input module may be routed to different output links, hence different central modules from time to time. Queuing at the central module rather than at individual output port make it suitable for dynamically routing cells to different links. At the same time, flexible service schemes can be adopted for a shared buffer memory switch to guarantee the QoS of each virtual circuit. Therefore, we choose the shared buffer memory switch to implement the input modules in our switch model.

The whole switch architecture of the shared buffer memory switch [30] can be adopted except that the control scheme of the switch should be a little modified to make it be used in our ATM path switch model. We take the input module  $i$  as an example to show the control scheme. Cells are read from input ports and queued in the logical queues according to their VPI/VCI as same as usual shared buffer memory switches. Then, according to the local routing table which is given by the  $i$ th row of the connection matrix, cells are write to output links  $c_0, c_1, \dots, c_{m-1}$  with one cells per link. As show in fig 4.3, since the  $(i, j)$ -th entry of the connection matrix is the central modules  $C_2, C_3, C_5$ , which means the output links  $c_2, c_3, c_5$  at the input module  $i$  are connected to output module  $j$ , then at most 3 cells can be write to the output links  $c_2, c_3, c_5$  and transmitted to output module  $j$ . Hence, 3 cells are chosen from the logical queues which are destined to the output module  $j$  according to the specific service scheme



adopted, empty cells can be added if all these queues are empty.

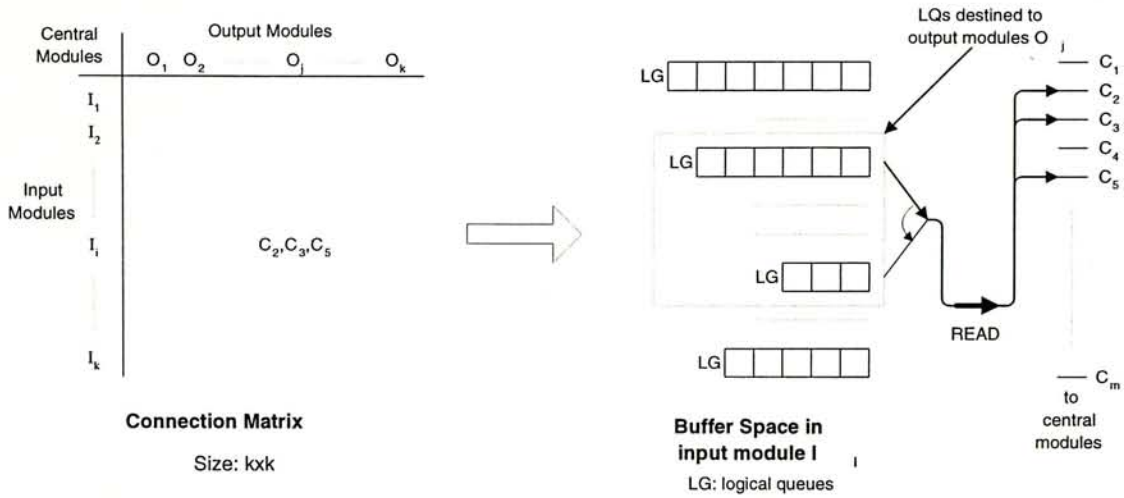


Figure 4.3: Connection matrix and local routing at the input modules

Since there are possibility more than one virtual path interconnecting one specific pair of input output modules, it is possible for more than one cells be transferred to the output module simultaneously. In other words, the out of sequence problem may occur. There are some different ways to avoid this problem. One way is to keep the sequence at both the transmitting and receiving. That means, if we sent out cells at the input modules from the first output link to the last output link and four cells sequentially per link, and receive at the output modules by the same order, that is, from the first input link to the last input link and keep the sequence of four cells per link.

Another way is by restricting reading more than one cells from each virtual queue at one time slot. Out of sequence can be avoided if we follow the two rules:

1. The integrated effective bandwidth of each virtual queue is no more than 1 link capacity.

2. No more than one cell will be read to the central star couplers from each virtual queue at any one time slot.

The rule 1 can be adopted since any call of the switch will not have effective bandwidth larger than the link speed. And when the queue is not empty, one cell per time slot of service rate can handle the queue of effective bandwidth no more than one link speed. The rule 2 avoids more than one cell from one connection being transferred to the output module at one time slot.

Another way is put no control and restriction on the input modules, but do a re-sequencing at the output modules. Since there are at most  $m$  cells being transferred from any input module, the size of the buffer at the output module to temporally contain the cells from each input module will be sufficient to be  $m$ . And re-sequencing those  $m$  cells will actually re-sequence the cells of each virtual circuit.

When a cell arrived to its destined output module, it will be further routed to its destined output port. Any kind of switch will work as output module in our switch model. We will still use shared buffer memory switch for its 100% throughput and flexible buffer management and service schemes. No out of sequence problem will occur at this stage if we still use logical queuing at the central shared memory of each switches.



# Chapter 5

## Multicasting

### 5.1 Two multicasting schemes

Multicasting is a function that new generation of ATM switches should provide for some applications such like video telephony, video on demand. Since the input and output modules of our switch model is implemented by shared buffer memory switch, multicasting can be easily provided by writing the concerned cell several times each to one of its destined output links. The virtual paths between the input and output modules should also accordingly provide the necessary bandwidth for those connections. According to the places the replication is performed, there are two possible schemes:

1. Scheme 1: Cells are replicated at both input and output modules.

In this scheme, a multicasting cell is first replicated in the input module and routed to the output modules it is destined to. The number of replication is equal to the number of destined output modules. When each copy

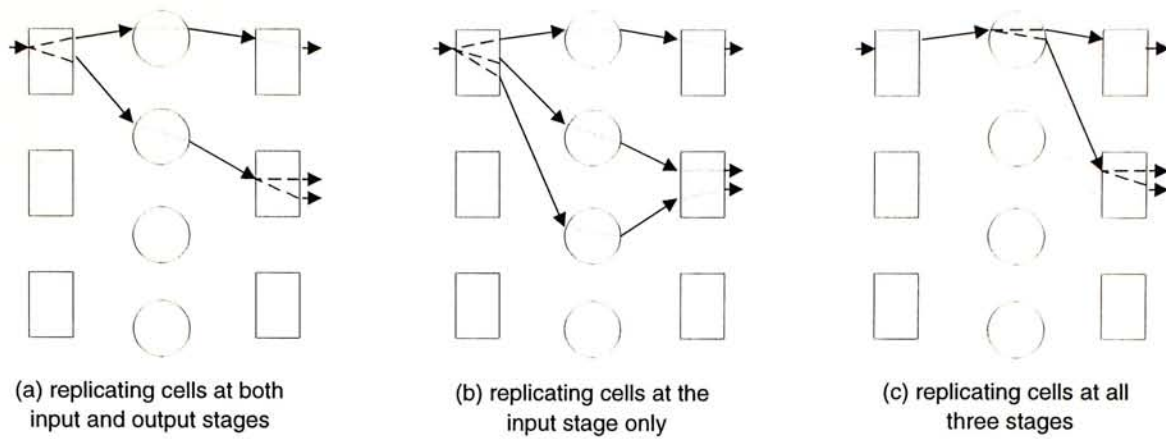


Figure 5.1: Multicasting schemes for WDM cross-path switch

arrives to its output module, it is further replicated to the output ports it is destined to within that output module. As shown in figure 5.1 (a).

2. Scheme 2: Cells are replicated at the input modules only.

In this scheme, the number of replication at the input module is equal to the total number of output ports it is destined to. Each copy will be routed to its output module through the virtual paths and no replication needed at any output module. As shown in figure 5.1 (b).

Due to the broadcasting property of the pass-through star-couplers, the incoming signal at the star-coupler can be received by more than one receiver at the output modules (tunable receivers shall be used). Thus the multicasting can also be implemented at the middle stage, as shown in figure 5.1 (c), but this scheme is useful only when one multicasting connection is using large amount of bandwidth (close to the link speed) or there is a set of multicasting connections destined to the same set of output modules. If not, the path scheduling will be very complex and difficult. We leave this possible scheme for further study (like add an extra path scheduling algorithm when the case really happens such that

using the broadcasting property of star-couplers at the central modules can save the internal virtual path capacity tremendously).

The routing at the input and output modules due to the two different multicasting schemes are different. For scheme 1, the switch controller at each input module only needs to know which output modules the cells of a specific call are destined to. They will be sent to those output modules for further delivery. Due to the further multiplication at the output modules, the input link, VPI, VCI of the cells shall be passed to the output modules rather than do any header translation, see table 5.1 as an illustration.

Input link	VPI	VCI	Output modules
01011	3	5	00110, 01001, 10011
01001	5	5	00110, 01101
...	...	...	...

Table 5.1: Output modules look up table at the input module for scheme 1

The input module controller will choose each logical queue for service according to some service scheme. The cell tagged with the input module number and the input link number to be routed to the output module. Therefore, the output module can recognize which VC it belongs to. It can then be further routed to its destined output links.

As illustrated in the table 5.2, for an incoming cell, the output module controller can look up the table according to its input module and input link which were tagged by the input module and the cell's VPI and VCI, find out which output links with that output module the cell is destined to and their new output VPI and VCI.

Input Module	Input link	VPI	VCI	Output Link	Output VPI	Output VCI
00101	01011	3	5	00110	4	4
				00111	3	2
				01000	3	3
00101	00100	5	5	00001	3	3
				00101	5	2
...	...	...	...	...	...	...

Table 5.2: Routing table and head translation at the output module for scheme 1

Input Link	VPI	VCI	Output Module	Output Link	Output VPI	Output VCI
01011	3	5	00110	00110	4	4
			00110	00111	3	2
			00110	01000	3	3
			01001	...	...	...
			10011	...	...	...
...	...	...	...	...	...	...

Table 5.3: Routing table and head translation at the input module for scheme 2

For multicasting scheme 2, the cells will be replicated at the input module only. Hence it is adequate to do all header translations at the input module.

As shown in the table 5.1, the cells are replicated and tagged with output link, output VPI, output VCI and transferred to each destined output modules. The output module controller will then further route each cell to its outlet according to its header, no routing table lookup or header translation is needed.



## 5.2 Call blocking

From the two schemes, we can see that the virtual paths needed to provide the interconnection of input and output modules for scheme 2 is more than scheme 1, and they are both more than that for the case of unicast. Let us see the scenario of the call setup of a multicasting connection at scheme 1. When such a call arrived at, say, input module  $i$ , with effective bandwidth  $\delta$ . According to the output links its destined to, we find out the output modules its destined to, say,  $j_1, j_2, \dots, j_r$ . That means  $r$  of bandwidth  $\delta$  should be added to the entries  $\lambda_{ij_1}, \lambda_{ij_2}, \dots, \lambda_{ij_r}$  of the traffic loading matrix  $\Lambda$ .

Due to the replication at the input module, the input module capacity constraint 2.3 will not be always hold. Hence the success coloring of the corresponding bipartite graph will not be guaranteed and therefore the internal call blocking at the virtual path level may happen for multicasting connections. Assume the number of star couplers connecting input and output modules is  $m$ , then the virtual path capacity constraints concerned by this new call is as follows

$$\begin{cases} \sum_j P_{ij} \leq m, \\ \sum_i P_{ij_s} \leq m, \quad \forall s = 1, 2, \dots, r. \end{cases}$$

For scheme 2, the scenario is a little different. For the output modules  $j_1, j_2, \dots, j_r$ , we assume there are  $l_1, l_2, \dots, l_r$  number of output links it is destined to respectively. Then each entry  $\lambda_{ij_s}$  in the new traffic loading matrix should be recalculated to accommodate these  $l_s$  times of new capacities for any  $s = 1, 2, \dots, r$ . The virtual path capacity constraints are same as which in scheme 1.



# Chapter 6

## Performance

### 6.1 Introduction

From the discussion of unicast and multicast of WDM cross-path switch in the above chapters, we know that for the same switch size but different pair of  $n$ ,  $k$ , the number of central star couplers needed to provide necessary internal virtual paths, hence the complexity of the switch is varied. For multicasting switch, due to the duplication at the first stage module, the call blocking is unavoidable, so what is the sufficient number of central star couplers to make the call blocking rate acceptable small. Also, two different multicasting schemes will introduce different switch complexity, we need to numerically compare these two schemes.

First, let us fix the size of our switch in consideration as  $1024 \times 1024$  with link speed OC3. Since the throughput of the input and output modules is 100% if we use the shared buffer memory switches, and the cell level loss and delay can be controlled by the specific service scheme in the individual input and output

modules, the main consideration of the performance of our switch model is the path level call blocking rate.

## 6.2 Switch complexity

In simulation, the distribution of effective bandwidth of the calls, which will arrive the switch node in a network, is too varies to define. Therefore, we fix the effective bandwidth of each call to simplify the simulation. The effects of different fixed effective bandwidth will be compared later in this chapter. We fix the effective bandwidth of each call to be 0.25 link speed, which is about 39Mbps for OC3 link, which is large enough for varies connections (calls with higher effective bandwidth will suffer larger blocking rate, which is natural and will be shown later).

We assume each call is randomly arrived to input links of our switch and equally likely destined to each output link. The loading rate of our switch is assumed to be 0.9. At this stage, we consider unicast case only. For each pair of  $n, k$ ,  $n \times k = 8 \times 128 = 16 \times 64 = 32 \times 32 = 64 \times 16 = 128 \times 8$ , according to the virtual path capacity constraints 2.5, we get the smallest number of central star couplers  $m$  such that the path level call blocking rate is less than  $10^{-4}$  [15], as shown in the table 6.1.

For fixed switch size  $N = nk$ , larger  $n$  means a small value of  $k$ . From the table 6.1, we conclude with larger module size  $n$ , less the central stage complexity and smaller the expansion factor. In the other words, higher ratio of the total throughput of the central star couplers to the total capacity of input and output modules can be achieved. It is natural since if  $n$  go to extreme be equal to the

module size $n$	no. of modules $k$	no. of star couplers $m$	complexity $m \times k \times 2$	expansion factor $m/n$
8	128	32	8192	4
16	64	48	6144	3
32	32	50	3200	1.6
64	16	72	2304	1.125
128	8	132	2112	1.03

Table 6.1: Number of star couplers for different pair of  $n/k$ 

switch size  $N$ , then only one input module, no central modules needed, and the complexity of individual input output modules are not considered in the table.

At the same time, the traffic stream on each virtual path would be much smoother, resulting better statistical multiplexing gain. This is because the superposition of  $n$  point processes will approach to a Poisson process for large  $n$ . Also, the capacity assignment and route assignment will be simpler for a smaller value of  $k$ , since the path assignment matrix and path rearrangement algorithm and routing tables are all  $k$  by  $k$  matrices.

On the other hand, the larger  $n$  also introduces larger complexity of input and output modules. While the limitation of the size of each individual switch is just the reason for us to consider the three-stage Clos network model. Particularly, for shared buffer memory switches, the limitation of the current technology we discussed in chapter 4 gives us a reasonable switch size of 64 by 64 with OC3 link speed. Therefore, we choose  $n=64$  and hence  $k=16$  for our switch model.



speed up factor $s$	no. of star couplers $m$	complexity $m \times k \times 2$	expansion factor $m \times s/n$
1	72	2304	1.125
2	44	1408	1.375
4	26	832	1.625
8	16	512	2

Table 6.2: Number of star couplers for different speed up factors

### 6.3 Speed up

The internal wavelength division multiplexing system usually operating at much higher speed than the external link speed (OC3 as we assumed). By speed up the internal link speed of the middle stage, we can reduce the number of central modules needed. Let the speed up factor be  $s$ , then the virtual path needed for the aggregate traffic  $\lambda_{ij}$  becomes

$$P_{ij} = \lceil \lambda_{ij}/s \rceil, \forall 0 \leq i \leq k-1, 0 \leq j \leq k-1.$$

Since  $P_{ij} < \lambda_{ij}/s + 1$ , by summing up according to all input modules and all output modules, we have

$$\begin{cases} \sum_j P_{ij} \leq n/s + k - 1, & \forall 0 \leq i \leq k-1; \\ \sum_i P_{ij} \leq n/s + k - 1, & \forall 0 \leq j \leq k-1. \end{cases} \quad (6.1)$$

Hence the corresponding bipartite graph has maximum vertex degree  $m = n/s + k - 1$ .

For different speedup factors  $s = 1, 2, 4, 8$ , we compared the number of central modules needed to keep the call blocking rate less than  $10^{-4}$ , using the same assumptions as in the last section for simulation.



As shown in the table 6.3, the number of central modules is reduced tremendously with higher speedup factor. On the other hand, due to the larger overhead when rounding up the virtual paths, larger expansion factor is faced when speeding up. The decision of which speedup factor shall be chosen largely depends on the technology available and the prices. We choose the speed up factor 4 here, which will give a reasonable usage, and reasonable number of total optical components needed.

## 6.4 Two multicasting schemes

As we fixed the effective bandwidth for simulation of call blocking rate at the above sections, here we first compare the effect of different effective bandwidth to the performance of the switch model. Where we have chosen the input output module size  $n = 64$  and number of input output module  $k = 16$  such that the total switch size is  $1024 \times 1024$ . The speed up factor is 4 for central WDM systems compared to the link speed OC3. The other simulation conditions are same. The loading rate is also set to 0.9. We choose the effective bandwidths are 0.125, 0.25, 0.5, and 1 link speed, that is approximately 19, 39, 78 and 155Mbps respectively.

As shown in the figure 6.1, we can see that larger the effective bandwidth of the calls, larger the probability of the call getting blocked at the virtual path level, as we expected.

In order to compare the performance of two multicasting schemes proposed in chapter 5, we consider three different fanout distributions:

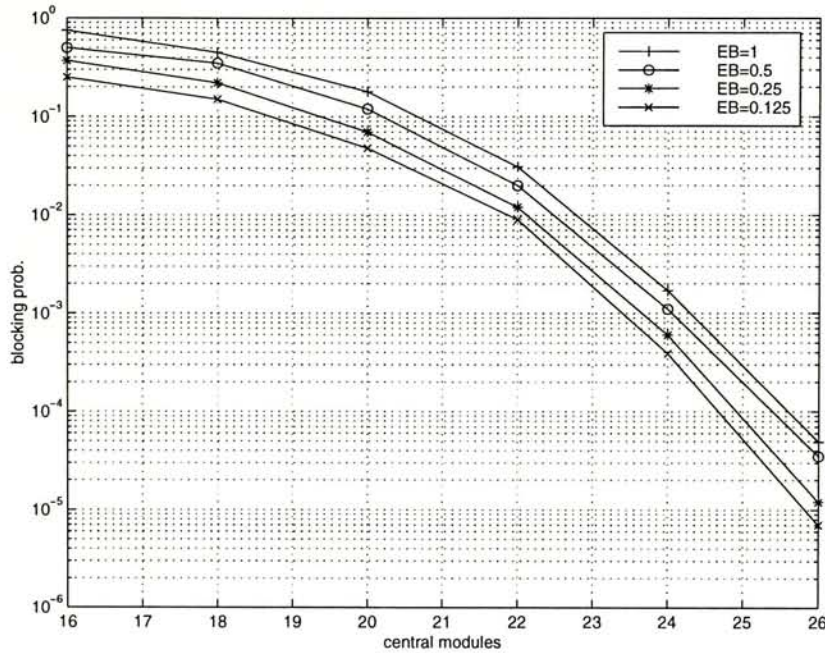


Figure 6.1: Call blocking rate for different effective bandwidth

## 1. Constant distribution

$$Pr\{Y = M\} = 1$$

Mean  $E[Y] = M$ , and variance  $Var[Y] = 0$ .

## 2. Uniform distribution

Suppose the requested fanout is uniformly distributed from 1 to  $M$ . In other words,

$$Pr\{Y = y\} = 1/M, 1 \leq y \leq M.$$

Thus  $E[Y] = \frac{M+1}{2}$  and  $Var[Y] = \frac{M(M+1)}{6}$ .

## 3. Truncated geometric distribution

$$Pr\{Y = y\} = \frac{(1-q)q^{y-1}}{1-q^M}, 1 \leq y \leq M. \quad (6.2)$$

The mean is given by

$$E[Y] = \frac{1}{1-q} - \frac{Mq^M}{1-q^M}.$$

This distribution is often used in the literature for modeling the fanout distributions [32, 25]. By fixing  $M$  equal to the switch size  $N = 1024$ , the parameter  $q$  will determine the mean fanout number  $E[Y]$ . We will use this assumption when we use this fanout distribution in our simulation.

Under the same switch parameters and assumptions for simulations in the above sections, with speed up factor 4 we have chosen in section 3, we find the call blocking rates for different value of expansion factors.

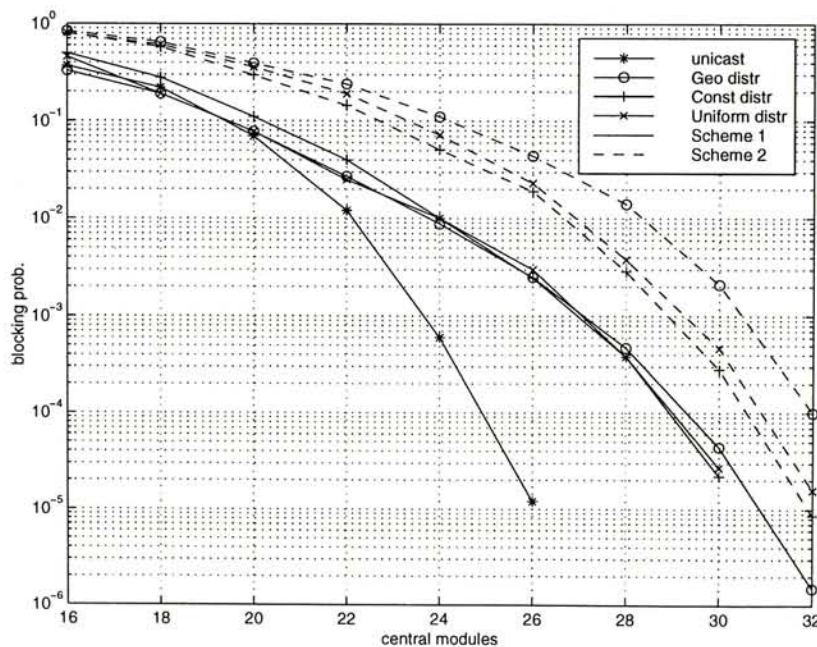


Figure 6.2: Call blocking rate for different fanout distributions in two multicasting schemes

As shown in the figure 6.2, since the truncated geometric distribution has the largest variance, it suffers the largest call blocking rate. The next large variance is the uniform distribution and the smallest one is the constant distribution,



which also corresponds to the order of the call blocking rate of these two distributions, as shown by the curves of the multicasting scheme 2. As for scheme 1, due to the manipulation at the last stage, the loading to the central stage, which causes the call blocking on the path level, is reduced. Therefore the effect is not clear.

At last, we compare the two multicasting schemes for different mean fanout numbers. We use the truncated geometric distribution for illustration. The four numbers for the parameter  $p$  in the distribution 6.2 are 0, 0.5, 0.75 and 0.875. There are corresponding to the mean fanout number 1 (for unicast), 2, 4, and 8 respectively. The other simulation assumptions are unchanged.

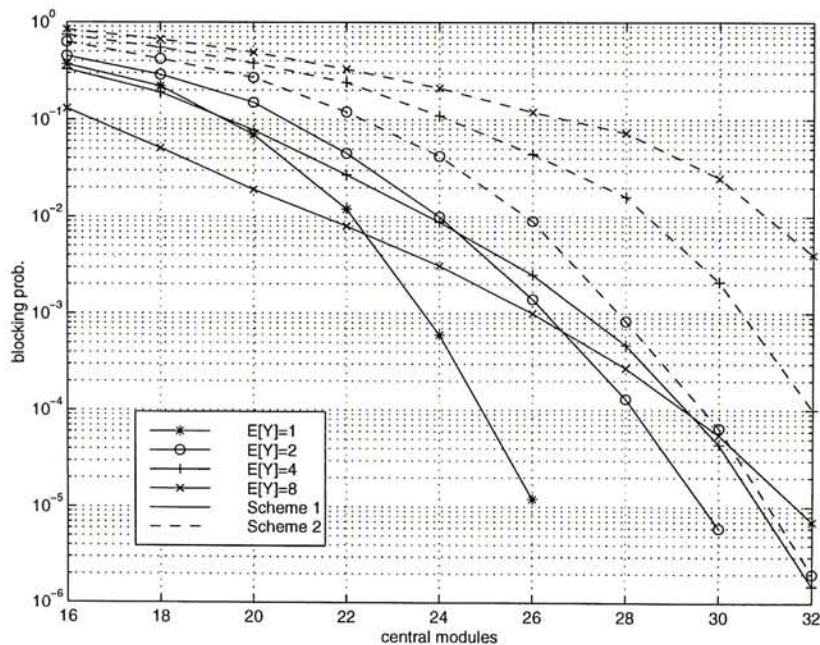


Figure 6.3: Call blocking rate vs. number of star couplers for two multicasting schemes with loading rate 0.9

We did simulations under two loading rates 0.9 and 0.8. From the figure 6.3 and 6.4, we can see that larger the mean fanout number, the larger call blocking rate approximately. It is natural since the multicasting call with larger number



of fanout will has larger probability that no enough virtual paths connecting to any of its destined output module. The effect is clear for multicasting scheme 2. As for multicasting scheme 1, the effect is influenced by the cell replication at the output module. For small number of central stage, since the multicasting at the last stage, the loading to the central stage is reduced, and the larger mean fanout number the reduction will be larger, i.e., the loading to the central stage will be smaller. Hence the call blocking on the virtual paths are actually smaller for higher mean fanout number. When the number of central modules get larger, this effect is weakened compare to the higher probability for connections with larger fanout number.

For any distribution with different mean fanout number, we can see that the multicasting scheme 1 suffers less call-blocking rate than the scheme2, as we expected. And we find that for scheme 1, when  $m$  is 32, the call blocking rate for all three fanout distributions is less than  $10^{-4}$ , which is acceptable.

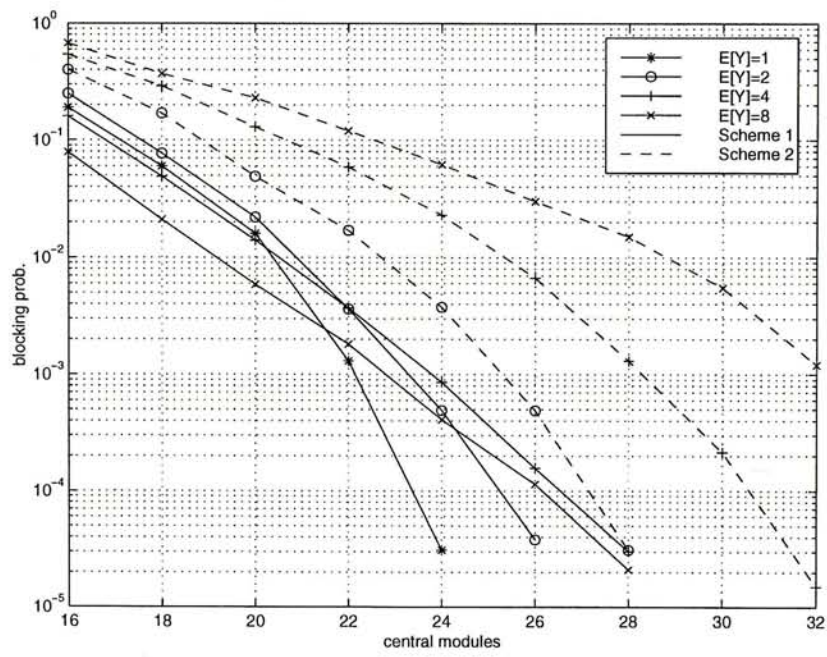


Figure 6.4: Call blocking rate vs. number of star couplers for two multicasting schemes with loading rate 0.8

## Chapter 7

# Switch Model and Operation

The switch model we proposed is an  $1024 \times 1024$  ATM switch with link speed *OC3*. As shown in the figure 7.1, the switch model is based on the three-stage Clos network with  $k = 16$  input output modules and the size of each input output module is  $n = 64$ . The central stage is total  $m = 32$  star couplers each size 16 by 16. The WDM system is operating at four times of the link speed, that is *OC12*.

The control of switch is composed of path level and cell level. As described in the chapter 2, the effective bandwidth of the aggregate traffics from each pair of input output modules has been calculated and the number of virtual paths is decided. The connection pattern of the central modules is an edge coloring of the bipartite graph. When the traffic characteristic has been changed significantly, the connection matrix has been changed according to the path rearrangement algorithm. The cell level control is distributed to each individual input and output modules.

We use the shared buffer memory switch as example for input and output

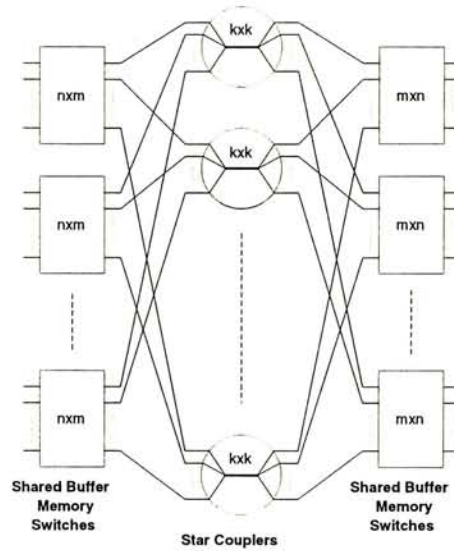


Figure 7.1: Proposed switch architecture

modules to see how the switch works. Each input or output module has total throughput of 19.9Gbps. The number of inlets, outlets  $n = 64$  is not essential since all arriving cells are multiplexed and write to the central shared memory, and de-multiplexed to output ports, the switch is inherently modularized. We can also take module size  $n = 16$  with link speed *OC12*, or any combinations with the same total throughput. Logical queues are maintained at the shared buffer memory according to the cell's VPI/VCI. Cells are write out to the output links according to the routing table 4.3, which is changed following the change of the connection matrix and virtual path rearrangement. Due to the speed up factor 4, at most 4 cells can be write to each output link and be transferred to the star couplers at the optical transmitter. The cells are chosen from the logical queues at the central memory according to specific service scheme being adapted to guarantee the cell level QoS.

The star couplers in the middle stage are used to provide the necessary number of paths to connect input modules and output modules. Either fixed



wavelength transmitter at input modules and tunable receiver at output modules or conversely, tunable transmitters at input modules and fixed wavelength receivers at output modules can be used. A path is setup by tuning the receiver or transmitter correspondingly to the specific wavelength, as shown in the tuning tables 3.2 and 3.4 respectively.

When a cell arrives its destined output module, it can be further routed to its destined output links. We will still use the shared buffer memory switch as output modules for its 100% throughput and flexibility of choosing adequate service schemes for varies QoS requirements.

Multicasting is easy to implement in a shared buffer memory switch [30], two multicasting schemes are compared and we choose the scheme 1 for its lower call blocking rate. The routing table and header transfer for multicasting at input and output modules are given in 5.1, 5.2 and 5.1.

# Chapter 8

## Conclusions

In this thesis, we have proposed a switching scheme for large scale ATM switches which uses shared buffer memory switches as input and output modules and optical star couplers interconnecting them to provide the virtual paths between any pair of input and output modules. The capacities of the virtual paths are calculated by the effective bandwidth of the aggregate traffics between the corresponding pair of input and output modules. Hence the path level QoS is guaranteed. The control of the cell level QoS is at the individual input and output modules by adopting service schemes for the logical queues of virtual circuits.

When we use the WDM technique for large scale ATM switches, some technological challenges are faced, the critical ones are the tuning speed and tuning range of the optical components. Compare to the most current large scale ATM switches using star couplers as the core or middle stage in Clos network, the WDM system used in our proposed scheme is operated at path level, but not cell level, which has low requirements on tuning speed of tunable transmitters

or receivers.

Two multicasting schemes have also been studied. The routing at the input and output modules and the tuning table for the WDM systems are also given. By simulating the call blocks at the path level, we found that scheme 1 suffers less call blocking rate.

For a specific switch size  $1024 \times 1024$  with link speed 155Mbps, we have proposed a switch model which is composed of 16  $64 \times 32$  shared buffer memory switches as input modules, 16  $32 \times 64$  shared buffer memory switches as output modules, 32  $16 \times 16$  passive star couplers and 512 optical transmitters and receivers operating at 622Mbps, either the transmitters or the receivers are made tunable. The switch is non-blocking on path level for unicasting calls, and low call blocking rate (less than  $10^{-4}$ ) for multicasting.

# Appendix A

## Effective bandwidth and QoS guarantee

### A.1 ATM service categories and QoS parameters

As specified by ATM Forum [4], the QoS parameters used by the ATM service categories are:

1. Peak-to-peak Cell Delay Variation (peak-to-peak CDV)
2. Maximum Cell Transfer Delay (maxCTD)
3. Cell Loss Ratio (CLR)

The architecture of services provided by the ATM layer consists of the following five service categories:



**CBR** Constant Bit Rate

specified QoS parameters: peak-to-peak CDV, maxCTD, CLR.

**rt-VBR** Real-Time Variable Bit Rate

specified QoS parameters: peak-to-peak CDV, maxCTD, CLR.

**nrt-VBR** Non-Real-Time Variable Bit Rate

specified QoS parameters: CLR.

**UBR** Unspecified Bit Rate

no QoS parameter is specified.

**ABR** Available Bit Rate

CLR can be specified for network specific.

## **A.2 Effective bandwidth for single source**

The basis of call level capacity allocation is the QoS requirement at the cell level, where the QoS requirement of each kind of service scheme is discussed at the above section. Those QoS parameters of each call are negotiated by the end user and the network nodes at the call setup time. The effective bandwidth of a call is the minimum bandwidth required to satisfy the prerequisite QoS of each call. The most important two QoS parameters being discussed are the delay and the loss. The delay of a cell on its transmission path from the source to the destination is the sum of the propagation delay, which is fixed (for fixed routing) and the varying queuing delay. The delay constraint is defined statistically at each switch node:

$$-\log_{10} Pr\{W > \tau\} > \delta_D \quad (\text{A.1})$$

Where  $Pr\{W > \tau\}$  is the probability of the cell waiting time exceeds a give delay bound  $\tau$ .

On its way of transmission, a cell may be discarded at some switch node due to the limited resources such as switch bandwidth or the buffer size. The cell loss probability is also defined in a statistical way at each switch node:

$$-\log_{10} Pr\{X > B|seenbyarrivals\} > \delta_L \quad (\text{A.2})$$

where  $X$  is the random number of the queuing length and  $B$  is the required buffer size. Thus the required delay and loss probability constraint can be satisfied by the effective bandwidth  $\mu$  and the buffer size  $B$ .

### A.2.1 Markovian on/off source approach

The calculation of the effective bandwidth of a single Markovian on/off source is given in [3]. The arrival process of a Markovian on/off source has two states: the source sends cells at a certain rate at the 'on' state and keeps idle at the 'off' state. The traffic can be fully characterized by the following three parameters [3]:

1. the mean arrival rate:  $m$ ;
2. the peak arrival rate:  $P$ ;
3. the average 'on' period  $T_{on}$ .

If the source arrives a queue with constant service rate  $C$ , the queuing length distribution is

$$Pr\{X > x\} = \frac{m}{\mu} e^{-\frac{P(\mu-m)x}{\mu T_{on}(P-\mu)(P-m)}} \quad (\text{A.3})$$

where  $\mu$  is the service rate.

The queuing length distribution with condition seen by arrivals is

$$Pr\{X > x|seenbyarrivals\} = e^{-\frac{P(\mu-m)x}{\mu T_{on}(P-\mu)(P-m)}} \quad (\text{A.4})$$

Since the cell delay  $\tau$  and the queuing length  $x$  satisfies the equation  $x = \mu\tau$ , the cell delay distribution is given by equation A.4 as follows

$$Pr\{W > \tau\} = e^{-\frac{P(\mu-m)\tau}{T_{on}(P-\mu)(P-m)}} \quad (\text{A.5})$$

Substitute the distribution A.5 to the delay constraint A.1, we can calculate the effective bandwidth

$$C = m + \frac{T_{on}(\delta_D + \log_{10} e)(P - m)^2}{P\tau + T_{on}(\delta_D + \log_{10} e)(P - m)} \quad (\text{A.6})$$

Similarly, the buffer size requirement can be calculated by the cell loss constraint A.2 and the equations A.4 and A.6, it is given by

$$B = \frac{CT_{on}(P - C)(P - m)(\delta_L + \log_{10} e)}{P(C - m)} \quad (\text{A.7})$$

### A.2.2 Leaky bucket regulated source

Leaky bucket regulator [44] is a simple algorithm to regulate the cell rate of source traffics, it has been chosen by the ATM Forum as the traffic shaper for

VBR sources. It is characterized by the following three parameters:

1. token generating rate  $r$ ,
2. token buffer size  $B_T$ ,
3. peak rate  $P$ .

As shown in the fig A.1, the traffic source entering the leaky bucket buffer, it can departure the leaky bucket regulator only if there is unused token in the token buffer and the departure rate is limited by the service rate  $P$ , that is the maximum cell rate the end-user can sent to the network.

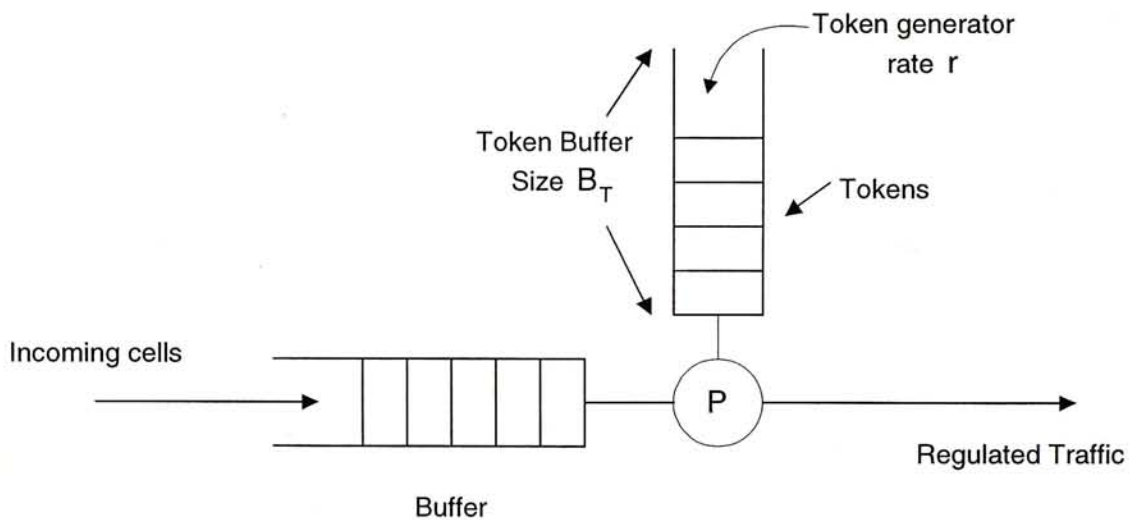


Figure A.1: Leaky bucket for traffic shaping

It has been shown [18] that the most bursty traffic after the leaky bucket regulator is the periodic on off traffic. For such a greedy source, the on period will exhaust all tokens in the token buffer  $B_T$  in peak departure rate  $P$ . Then the source will keep idle until the token buffer is fulfilled again. As shown in the figure A.2.



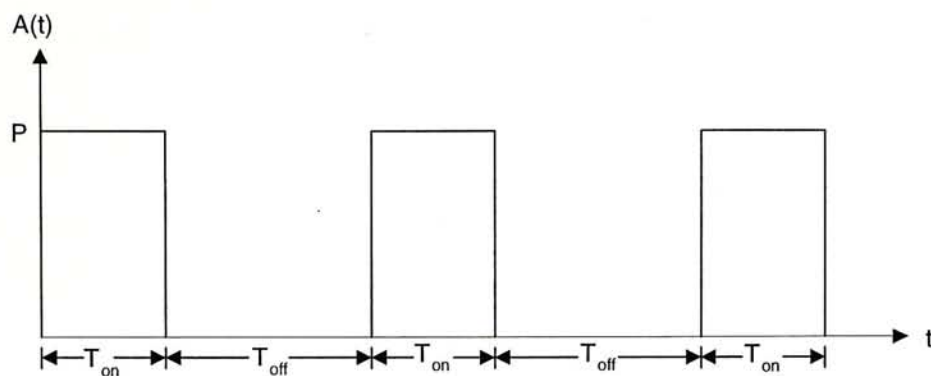


Figure A.2: Leaky bucket regulated periodic on off source

Since  $P \times T_{on} = B_T + r \times T_{on}$ , we have

$$T_{on} = \frac{B_T}{P - r}$$

In the "off" period, the source is idle until the token buffer is fulfilled. Therefore,  $r \times T_{off} = B_T$ , hence we have

$$T_{off} = \frac{B_T}{r}$$

When such a bursty traffic arrives a network node with service rate  $C$ , here we assume the capacity  $C$  is reserved exclusively to this source and all the remaining bandwidth not dedicated to the call at each node is exhausted by other calls during the whole duration of the call. This model gives the worst case bound and thus we use this relatively conservative model for calculating the required bandwidth.

We assume  $C > r$ , the stability condition and the buffer size is infinite. We know that the queue length will be bounded at this model and the delay can be controlled in a certain range. And further, if a sufficient large service rate is given, the delay can be bounded to  $\tau$  in the delay requirement A.1. In other words, the probability  $\delta_D$  will tend to zero.

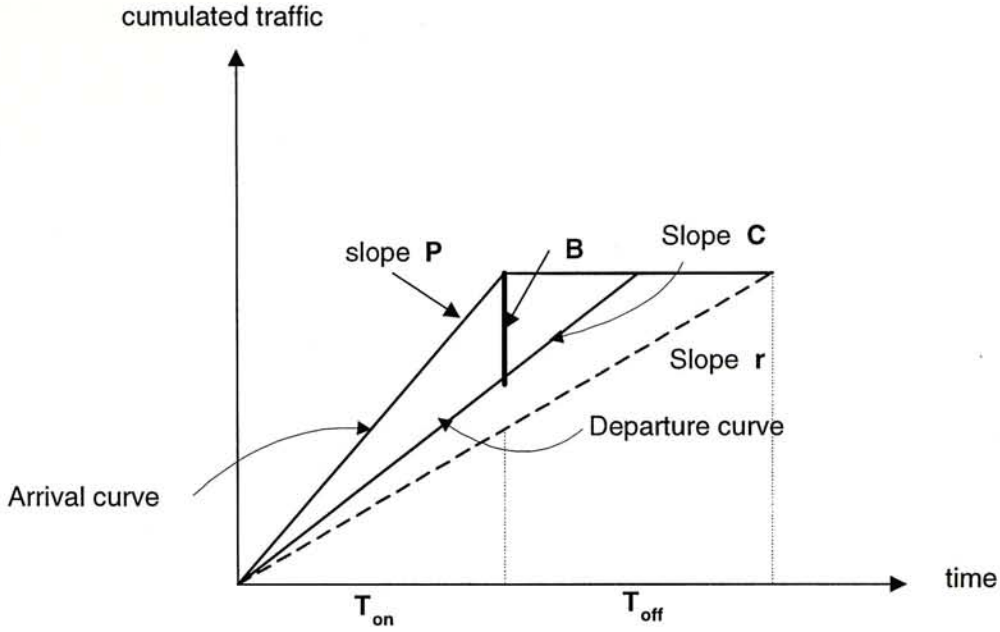


Figure A.3: Maximum queue length for a greedy leaky bucket regulated traffic

As shown in figure A.3, the maximum queue length happens at the end of the "on" period of the traffic. Let the maximum queue length be  $B$ , we have  $B = T_{on}(P - C)$ , therefore,

$$B = \frac{B_T}{P - r}(P - C)$$

Hence the maximum delay  $W_{max}$  is just a scaled version of the maximum queue length

$$W_{max} = \frac{B}{C} = \frac{B_T(P - C)}{(P - r)C}$$

If we have the delay requirement  $W_{max} \leq \tau$ , then we can calculate the capacity requirement

$$C \geq \frac{B_T P}{B_T + \tau(P - r)} \quad (\text{A.8})$$

When the capacity requirement A.8 is not satisfied, then the delay requirement  $W_{max} \leq \tau$  can not be 100% guaranteed.

It has been proved [31] that when the arriving curve is the one as shown in the fig A.4, the probability of delay exceeds  $\tau$  take the largest number. Where the slope of the arrival curve before  $t_0$  is  $\frac{y_1}{t_1} = \frac{y_0}{t_0} = P$  and the slope between  $t_0$  and  $t_2$  is  $\frac{y_2 - y_0}{t_2 - t_0} = r$ .

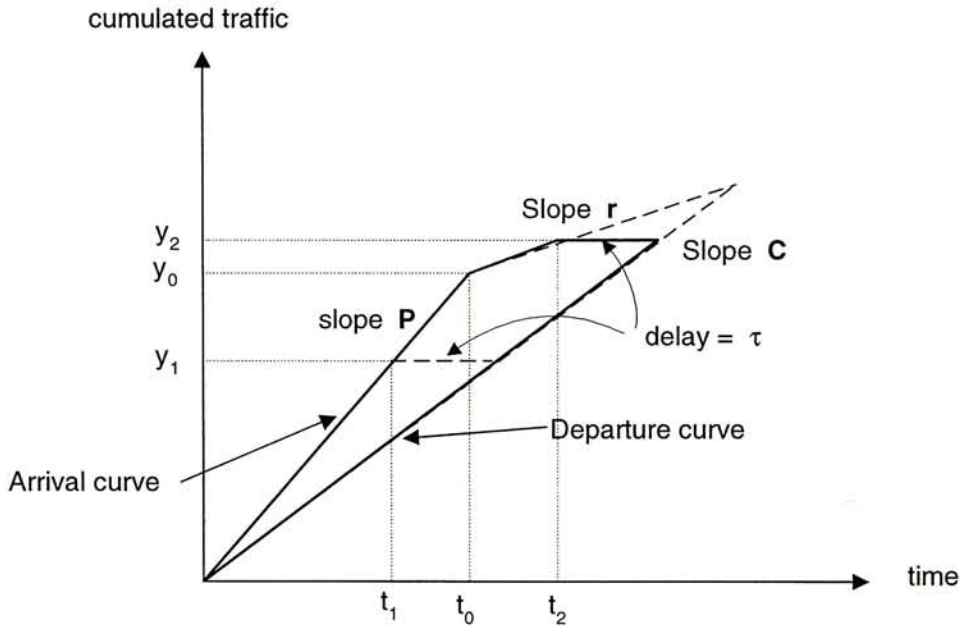


Figure A.4: Worst case delay probability for a leaky bucket regulated traffic

Let  $W$  be the random variable of waiting time. It can be calculated that

$$Pr\{W > \tau\} = 1 - \frac{y_1}{y_2} = 1 - \frac{P(C - r)\tau}{(P - C)(B_T - r\tau)}$$

Hence, if the delay constraint is  $Pr\{W > \tau\} < \delta$ , we can calculate the capacity required to guarantee the delay constraint

$$C > \frac{Pr\tau + P(B_T - r\tau)(1 - \delta)}{P\tau + (B_T - r\tau)(1 - \delta)}$$

## **A.3 Effective bandwidth for multiplexed sources**

When several connections are multiplexed to a single queue with certain buffer size, the queuing analysis for the bandwidth needed to guarantee the QoS of each connections quickly becomes infeasible as the number of multiplexed arrival streams increases. One approach is to use asymptotic analysis based on the large buffer size and small tail probability [25, 27, 21]. Another method is to use service separation by dividing the overall traffic flows into classes, homogeneous in terms of QoS requirements and statistical characteristics, which share the bandwidth of a link according to some specified policy [7, 40]. Many other researches did in the literature [3, 11, 28, 46].

### **A.3.1 Gaussian model approach**

One simple technique for evaluating the effective bandwidth of a set of multiplexed traffics is using Gaussian approximation [21]. First, each connection is approximately characterized by its mean cell rate  $m$  and standard deviation of its arrival distribution  $\sigma$ . For Markovian on/off source in Appendix A.2.1, the mean cell rate is  $m$  and the variance is  $\sigma^2 = m(P - m)$ . For leaky bucket regulated source in Appendix A.2.2, the mean bit rate is  $r$  and the variance is

$$\frac{r(P - r)^2 + (P - r)r^2}{P}$$

Assume there is a set of traffics  $j = 1, 2, \dots, N$ , with mean cell rates  $m_j$  and variance  $\sigma_j^2$ . The Gaussian approximation is based on the simplifying assumption that if the number of sources being multiplexed,  $N$ , is large, total traffic arriving to the network node behaves like a Gaussian process with mean arrival rate



$$m = \sum_{j=1}^N m_j$$

and variance

$$\sigma^2 = \sum_{j=1}^N \sigma_j^2.$$

Suppose the effective bandwidth calculated for each individual connection is  $c_j$  for  $j = 1, 2, \dots, N$ . Then the total bandwidth of the set of multiplexed traffics can be estimated by the following formula

$$C = \min\{m + \alpha\sigma, \sum_{j=1}^N c_j\}$$

where

$$\alpha \simeq \sqrt{2 \ln \frac{1}{\epsilon} - \ln 2\pi}.$$

Here  $\epsilon$  is the desired buffer overflow probability for the multiplexed traffic.

The first term ( $m + \alpha\sigma$ ) relies on the gaussian approximation for the aggregate traffic, it provides a good estimate of the required bandwidth when many connections with long burst periods and relatively low utilization are multiplexed on the same network link.

The second term, the linear summation of the effective bandwidths, results in substantial lower total capacity compared to the bandwidth requirements obtain in the first term, for connections with small burst periods [21].

# Bibliography

- [1] A. C. Acampora, *An Introduction to Broadband Networks*, Plenum Press, 1994.
- [2] E. Arthurs, M. S. Goodman, H. Kobrinski, M. P. Vecchi, *HYPASS: An optoelectronic hybrid packet-switching system*, IEEE JSAC, Vol 6, pp. 1500-1510, 1988.
- [3] D. Anick, D. Mitra and M. M. Sondhi, *Stochastic Theory of a Data-handling System with Multiples Sources*, Bell System Tech. J., 61, pp. 1871-1894, 1982.
- [4] The ATM Forum, *Traffic Management Specification*, Ver. 4.0, April 1996.
- [5] R. Bellman, *Introduction to Matrix Analysis, 2nd edition*, McGraw-Hill, New York, 1970.
- [6] K. E. Batcher, *Sorting networks and their applications*, Proc. 1968 Spring Joint Comput. Conf.
- [7] R. Bolla, F. Davoli, and M. Marchese, *Bandwidth Allocation and Admission Control in ATM Networks with Service Separation*, IEEE Commun. Magazine. May 1997.

- [8] Charles A. Brackett, *Dense Wavelength Division Multiplexing Networks: Principles and Applications*, IEEE JSAC Vol. 8, No. 6, August 1990.
- [9] Tom Chaney, J. Andrew Fingerhut, Margaret Flucke, J. S. Turner, *Design of a Gigabit ATM Switch*, Infocom 1997.
- [10] Youngbok Choi, Hideki Tode, Hiromi Okada, and Hiromasa Ikeda, *A Large Capacity Photonic ATM Switch Based on Wavelength Division Multiplexing Technology*, IEICE Trans. Commun., Vol. E79-B, No. 4, April 1996.
- [11] Gagan L. Choudhury, David M. Lucantoni and Ward Whitt, *Squeezing the Most Out of ATM*, IEEE Trans. on Commun. vol. 44, no. 2, Feb. 1996.
- [12] Arturo Cisneros and Charles A. Brackett, *A Large ATM Switch Based on Memory Switches and Optical Star Couplers*, IEEE JSAC Vol. 9, No. 8, October 1991.
- [13] J. Cooper, J. Dixon, M. S. Goodman, H. Kobrinski, M. P. Vecchi, E. Arthurs, S. G. Menocal, M. Tur, and S. Tsuji, *Nanosecond wavelength switching with a double-section distributed feedback laser*, Conf. Proc. CLEO'88, Anaheim, CA. 1988.
- [14] M. De Prycker and M. De Somer, *Performance of a Service Independent Switching Network with Distributed Control*, IEEE J. on Selected Areas in Commun., vol. 5, no. 8, pp.1293-1301, Oct. 1987.
- [15] Martin De Prycker, *Asynchronous Transfer Mode, Solution for Broadband ISDN*, Second Edition, Ellis Horwood, 1993.

- [16] M. Devault, J. Cochenec and M. Serval, *The Prelude ATD experiment: assignments and future prospects*, IEEE JSAC, Vol. 6, No. 9, December 1988.
- [17] Nicholas R. Dono, Paul E. Green, Jr., Karen Liu, Rajiv Ramaswami and Franklin Fuk-Kay Tong, *A Wavelength Division Multiple Access Network for Computer Communication*, IEEE JSAC Vol. 8, No. 6, August 1990.
- [18] A. Elwalid, D. Mitra, and R. H. Wentworth, *A New Approach for Allocating Buffers and Bandwidth to Heterogeneous, Regulated Traffic in an ATM Node*, IEEE JSAC, vol. 13, no. 6, August 1995.
- [19] Erol Gelenbe, Xiaowen Mang, Raif Önvural, *Bandwidth Allocation and Call Admission Control in High-Speed Networks*, IEEE Commun. Mag., May 1997.
- [20] Matthew S. Goodman, Haim Kobrinski, Mario P. Vecchi, Ray M. Bulley and James L. Ginlett, *The LAMBDANET Multiwavelength Network: Architecture, Applications, and Demonstrations*, IEEE JSAC Vol 8, No. 6, August 1990.
- [21] Roch Guerin, Hamid Ahmadi and Mahmoud Naghshineh, *Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks*, IEEE JSAC, vol. 9, no. 7, Sept. 1991.
- [22] Roch Guerin and Levent Gün, *A Unified Approach to Bandwidth Allocation and Access Control in Fast Packet-Switched Networks*, INFOCOM '92, 1A.1.1-1A.1.12, 1992.



- [23] Eric Hall, Jeff Kravitz, Rajiv Ramaswami, Marty Halvorson, Stephen Tenbrink, and Richard Thomsen, *The Rainbow-II Gigabit Optical Network*, IEEE JSAC, vol. 14, no. 5, June 1996.
- [24] Joseph Y. Hui and Thomas Renner, *Queueing Analysis for Multicast Packet Switching*, IEEE Trans. Commun., Vol. 42, No. 2/3/4, Feb./Mar./Apr. 1994, pp. 723-731.
- [25] J. Y. Hui, *Resource allocation for broadband networks*, IEEE J. Select. Areas Commun. vol. SAC-6, pp. 1598-1608, 1988.
- [26] Yongdong Jin and Mohsen Kavehrad, *An Optical Cross-Connect System as a High-Speed Switching Core and Its Performance Analysis*, Journal of Lightwave Technology, Vol 14, No. 6, June 1996.
- [27] F. P. Kelly, *Effective bandwidths at multi-class queues*, Queueing Syst., vol. 9, pp 5-16, 1991.
- [28] G. Kesidis, J. Warland, and C. S. Chang, *Effective Bandwidths for Multi-class Markov Fluids and Other ATM Sources*, IEEE/ACM Trans. on Networking, vol. 1, no. 4, August 1993.
- [29] N. Endo, T. Kozaki, T. Ohuchi, H. Kuwahara and S. Gohara, *Shared Buffer Memory Switch for an ATM Exchange*, IEEE Trans. Commun., Vol. 41, No. 1, pp. 237-245, Jan. 1993.
- [30] H. Kuwahara, N. Endo, M. Ogino and T. Kozaki, *Shared Buffer Memory Switch for an ATM Exchange*, in Proc. Int. Conf. on Communications, Boston, MA, June 1989, pp. 4.4.1-4.4.5.

- [31] C. H. Lam, *Virtual Path Traffic Management of Cross-Path Switch*, Ph.D. thesis, The Chinese University of Hong Kong, 1997.
- [32] Tony T. Lee, *Nonblocking Copy Networks for Multicast Packet Switching*, IEEE JSAC, Vol. 6, No. 9, pp. 1455-1467, Dec. 1988.
- [33] T. T. Lee, M. S. Goodman, and E. Arthurs, *A broadband optical multicast switch*, ISS'90, 1990.
- [34] Tony T. Lee and Cheuk H. Lam, *Path Switching - A Quasi-static Routing Scheme for Large-Scale ATM Packet Switches*, IEEE JSAC, pp. 914-924, vol. 15, June 1997.
- [35] Raymond H. Lin, Cheuk H. Lam and Tony T. Lee, *Performance and Complexity of Multicast Cross-Path ATM Switches*, INFOCOM'97.
- [36] Riccardo Melen and Jonathan S. Turner. *Nonblocking Networks for Fast Packet Switching*. IEEE INFOCOM'89, pp. 548-557, April 1989.
- [37] Riccardo Melen and Jonathan Turner. *Nonblocking Multirate Networks*, SIAM Journal on Computing, March 1989.
- [38] Riccardo Melen, Jonathan Turner. *Nonblocking Multirate Distribution Networks*, IEEE Trans. on Commun., Vol. 41, no. 2, pp. 362-269, Feb. 1993.
- [39] Abhay K. Parekh, and Robert G. Gallager, *A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case*, IEEE/ACM Trans. on Networking, vol. 1, no. 3, June 1993.

- [40] G. Ramamurthy and Qiang Ren, *Multi-class Connection Admission Control Policy for High Speed ATM Switches*, Technical Report, NEC C&C Research Laboratories, 1996.
- [41] Yoshito Sakurai, Nobuhiko Ido, Shinobu Gohara, Noboru Endo, *Large-Scale ATM Multistage Switching Network with Shared Buffer Memory Switches*, IEEE Communications Magazine, Jan. 1991.
- [42] M. Shreedhar and George Varghese, *Efficient Fair Queuing Using Deficit Round-Robin*, IEEE/ACM Trans. on Networking, vol. 4, no. 3, June 1996.
- [43] H. Suzuki, et al., *Output-Buffer Switch Architecture for Asynchronous Transfer Mode*, Proceedings of the International Communications Conference, pp 4.1.1-4.1.5., June 1989.
- [44] J. Turner, *New Directions in Communications, or Which Way to the Information Age?*, IEEE Commun. Mag. vol. 24, pp. 8-15, 1986.
- [45] J. Turner and N. Yamanaka, *Architectural Choices in Large Scale ATM Switches*, WUCS 97-21, Department of Computer Science, Washington University, 1997.
- [46] G. de Veciana, G. Kesidis, J. Walrand, *Resource Management in Wide-Area ATM Networks Using Effective Bandwidth*, IEEE JSAC vol. 13, no. 6, Aug. 1995.





CUHK Libraries



003704429