MIT Document Services

Room 14-0551
77 Massachusetts Avenue
Cambridge, MA 02139
ph: 617/253-5668 | fx: 617/253-1690
email: docs@mit.edu
http://libraries.mit.edu/docs

DISCLAIMER OF QUALITY

Due to the condition of the original material, there are unavoidable flaws in this reproduction. We have made every effort to provide you with the best copy available. If you are dissatisfied with this product and find it unusable, please contact Document Services as soon as possible.

Thank you.

Wide Area Optical Backbone Networks

by

Philip Jin-Yi Lin

B.S., California Institute of Technology, 1989
S.M., Massachusetts Institute of Technology, 1991

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 1996

© Massachusetts Institute of Technology 1996. All rights reserved.

Author ....
Department of Electrical Engineering and Computer Science
January 31, 1996

Certified by..
Robert G. Gallager
Fujitsu Professor of EECS
Thesis Supervisor

Accepted by..
F. R. Morgenthaler
Chairman, Departmental Committee on Graduate Students

Wide Area Optical Backbone Networks

by

Philip Jin-Yi Lin

Submitted to the Department of Electrical Engineering and Computer Science
on January 31, 1996, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering

Abstract

The focus of this research is on architectures for all optical networks (AON's) using wavelength division multiplexing (WDM). AON's, by using fiber as the transport medium, can provide higher throughput than electronic networks. Unfortunately, it is currently economically unattractive to route traffic in units smaller than wavelength bands. Most of today's applications are not suitable for AON's because they are too small to fill up one wavelength channel. One solution to this problem is to aggregate the small users together using electronic local networks so the aggregated traffic can fill up wavelength channels. Therefore, we propose a hierarchical network structure where the local networks aggregate traffic from the end users, and the AON acts as a backbone interconnecting these local networks. We design the backbone architecture under various traffic assumptions, and analyze the network performance. In one case, we design the minimum cost network under the constraint of a performance measure. In the future, users with bit rates equivalent to one wavelength will emerge and aggregation of such users will not be required. We also propose two novel constructions in building networks that support these large rate users.

Thesis Supervisor: Robert G. Gallager
Title: Fujitsu Professor of EECS

Acknowledgments:

This thesis is dedicated to my friends, family, and my Lord Jesus Christ.

Thank you Prof. Robert Gallager for being my thesis supervisor. Your patient guidance and faith in my ability pulled me through the thesis process. I also appreciate your valuable career advice.

Thank you Prof. Jeff Shapiro, Dr. Steve Finn, and Dr. Rick Barry for being on the thesis committee. Your insights and editorial comments improved this thesis. I appreciate the time you spent in those long thesis meetings.

Thank you Prof. Alvin Drake for allowing me to TA 6.041 which provided the turning point of my graduate career. Thanks for your encouragements through difficult times.

Thank you group 67 of the MIT Lincoln Laboratory for the summer internships. It was there the ideas for this thesis were formulated. I like to especially thank Eric Swanson, Roe Hemenway, Al Tidd, and Jean Mead. Thank you Dr. Vincent Chan for taking an interest in my thesis as well as my personal well being.

Thank you Dr. Charles Rohrs and Dr. James Mills for being the sounding board of my thesis ideas. Your comments have been helpful.

Thank you Kathleen O'Sullivan for performing above and beyond your call of duty. Students like me would not have survived otherwise. Thank you Nancy Young-Wearly and Sheila Hegarty for your administrative helps.

Thank you LIDS friends for your technical support as well as your friendship: David Tse, David Lin, Li Shu, Cynara Wu, Aradhana Narula, Diane Ho, Marc Ibanez, James Sarvis, Muriel Medard, Yannis Paschalidis, and many others. In more ways than you realize, all of you have helped to make my graduate studies a fuller and more enjoyable experience.

Thank you my brothers and sisters in Christ and fellow sojourners at MIT: Marilyn Chen, Jeff Kuo, Tim Chow, Jerry Chen, and Tom Wang. Your dedications to work and to the Lord are truly inspiring. Thank you members of the Chinese Christian Fellowship, you have made CCF a place of refuge where I was able to receive rest and refreshment.

Thank you Pastor Steve, Tom Lee, Bob Lee, Eric Lew, Regan Wong, Mark Gonzales, Chris Lam, Josh Li, Steve Lee (both of you), Lorraine Ho, Sue Lim, Julie Gamponia, Tina Yuen, Walton Yuen, Clarice Law, Angela Lih, Peter Huang, the music worship team, and other members and alumni of the Boston Chinese Evangelical Church. Thank you for making BCEC a warm, loving, and secure spiritual home. I would not have finished without your support and prayers.

To my closest friends: Thank you Melissa Lee, for being a trusted comrade through thick and thin; Carl Lim, for your steadfast friendship; Hsi-Jung Wu, for your advise on personal matters; and Moses Lam, for being a true "blood".

And my best friend of all, Wendy, thank you for patiently waiting five long years, for taking great care of me, for believing in my abilities, for lifting me up when I'm down, for laughing at my corny jokes, for supporting me when I'm right, correcting me when I'm wrong, and loving me in spite of it all.

Thank you mom and dad for providing a worry-free environment so I can concentrate on my studies. Thank you Helen and Alan, my dear sister and brother-in-law, for urging me to press on. Thank you San-Yi, my talented younger sister, for the delicious home cooking and clean laundry. I am blessed to be part of this wonderful family.

Lastly and most importantly, thank you Lord Jesus Christ, for your love and salvation, without the context of which all my achievements would be nothing.

This research was supported by National Science Foundation Grant NCR-9206379, Advanced Research Projects Agency (ARPA MDA 972-92-J-1038) and the Army Research Office ARO ASSERT DAAH04-94-G-0221.

Contents

List of Figures

Chapter 1

Introduction

Advances in optical technology have made optical networks a reality. Optical networks have many attractive attributes. Optical fiber is an ideal transport medium because it has large bandwidth and low loss. Wavelength routers, along with tunable transmitters and receivers, provide the network a simple and convenient way of routing large bandwidth sessions. Multicasting and multiple connections are simple using the broadcast abilities of the optics. These qualities make optical networks ideal backbones for high volume traffic, and good competitors to electronic networks for high rate point to point connections.

As technology matures, traffic patterns will change depending both on the available bandwidth provided by the network and on newly emerging applications. A network architect must design a network structure that will accommodate these changes. The network should support not only a large number of users, but also a large variety of data rates.

There are two extreme cases in the user data rate: low rate and high rate. Electronic mail and conventional data transfer are examples of low rate users. Remote medical imaging and movies on demand are examples of high rate users. The division between the two categories is gray. Usually, a low rate user occupies a small fraction of a wavelength channel, while a high rate user occupies one or more channels.

This thesis considers an all optical network (AON) for a wide area network. AON's are networks such that the signal is all optical from the origin to the destination.

Electrically controlled optical switches are allowed. The states of the switches, wavelengths of the signals, and the topology of the network determine the path a signal takes in an AON. All paths in the network can be shared by different sessions using wavelength division multiplexing (WDM) or time division multiplexing (TDM). We focus on WDM, and TDM within WDM channels, although many of the results also apply to pure TDM. We do not allow time slot interchange, optical buffers, or optical packet switching in the backbone because these are areas of research still in their infant stages.

Smaller rate users are aggregated electronically before they are injected into the AON. In this case, the AON acts as a backbone providing bit pipes that transport the aggregated data to the desired destinations. The data is de-aggregated upon exiting the backbone.

The optical network sets up circuits for high rate point to point connections. A combination of transmitter/receiver tuning and switch reconfiguration allows build up and tear down of the circuits.

Present day applications are dominated by small users, and therefore most of this thesis concentrates on how an AON can be utilized as a backbone for aggregated small rate users. If the amount of aggregation is large enough, then the resulting backbone traffic is close to static, and the required AON can be passive. Furthermore, if the traffic is uniform, then the aggregated traffic can be approximated as uniform all to all traffic. We design and analyze a passive AON sufficient to support such a traffic pattern. We also discuss methods of dealing with static but non-uniform traffic, and slowly varying traffic.

The last chapter of the thesis deals with network design for point to point large rate users. In this case, the AON acts as a network providing circuits for large rate users. The AON builds up and tears down circuits as the traffic demand changes. This problem has been analyzed before, e.g., [Bar 93] [Pan 92]. We extend their analysis with some novel constructions.

Chapter 2

Network and Traffic Model for Aggregated Traffic

All optical networks (AON's) utilize routers, optical switches, and other optical devices that operate on wavelength channels. Unfortunately, the capacity of a wavelength channel is too large for most present day applications. One way to solve this problem is to simply assume the future emergence of large rate users and focus on user connectivity problems. We do this in chapter 4. Alternatively, one can view local area networks as single users with large volumes of traffic to each of the other local area networks. This thesis emphasizes the latter approach, where the AON acts as an optical backbone interconnecting local networks. The size of the local network is a design parameter, and the backbone traffic depends on the size of the local network.

This chapter motivates the readers why we are looking at aggregated traffic from small rate users. This chapter also develops the network and traffic models for the aggregated traffic. These traffic models are developed from the local networks' point of view. In later chapters, we design backbone topologies that can support each of the traffic models defined in this chapter. Then, we develop a more realistic traffic model from the end users' point of view, and calculate the blocking probability of the network. We also discuss the trade off between blocking probability and network cost. The results from these chapters will be beneficial for creating a hierarchical network structure, connecting a large number of users over a large geographical area.

The word *network* is used for both the optical backbone and the connecting local network. Where it is unclear in context, the word *backbone* will be used for the underlying optical network, and the phrase *local network* will be used for a local network utilizing the backbone.

2.1 Motivation for Aggregated Traffic

In order to utilize the fiber bandwidth, one must either find high data rate applications or aggregate a large number of low data rate applications. In most cases, even the aggregated data rate is small compared with the fiber capacity. A possible solution is wavelength division multiplexing (WDM) where the fiber capacity is divided into many wavelength channels. However, even under such a scheme, the channel rate for each wavelength channel is still large in comparison with many present day applications. For example, the AON consortium of AT&T, DEC, and MIT [Al+ 93] is building a WDM system with a wavelength channel rate of a few gigabits per second, while present day local area networks run at 100 megabits per second.

One can further divide a wavelength channel using time division multiplexing (TDM) or sub-band frequency division multiplexing (SFDM). However, it is currently impossible to independently route sub-band frequency divided channels in an AON. TDM channels can be routed independently, but timing issues become problematic for long distance mesh networks, (see discussion in section 2.2). Therefore, TDM can only be used in limited cases where timing problems can be solved, and in most cases of present, the practical channel size is one wavelength.

In the near future, comparatively low rate applications are expected to dominate the information space in terms of number and importance, if not cumulative rate. Therefore, aggregation is required to utilize the fiber bandwidth. Even when high rate applications become more popular, aggregation is still needed to accommodate commonplace low rate applications like small file transfer, email, and voice.

For the major part of the thesis, we envision local area networks attached to an AON backbone. Local data stays within the local area networks, while long distance data traverse through the backbone. The local area networks are responsible for aggregating long distance data in order to utilize the high capacity channels of the AON backbone.

2.1.1 Aggregation method

TDM and SFDM are valid methods for aggregating small rate users before their data bits are injected into the backbone. These methods require scheduling mechanisms assigning the sub-channels to each session. Another method is packet switching like Asynchronous Transfer Mode (ATM). ATM provides a flexible and standardized architecture in aggregating low rate connections.

Today, signals that are traditionally analog, like voice, are being digitized, and packet switching has become more popular in place of traditional circuit switching. Therefore, ATM switches will be emphasized as the method for aggregation. However, the backbone designed in this thesis can by applied to any aggregation method.

2.1.2 Advantage of aggregation

Aggregation not only allows the network to use the fibers more efficiently, but also simplifies the backbone because the resulting traffic appears static. The static nature is the result of statistical multiplexing. As we increase the amount of aggregation, the total bit rate of an aggregated group goes up linearly with the number of sessions while the standard deviation goes up as the square root of the number of sessions. Therefore, the percent deviation from the mean of the combined bit rate of an aggregated group decreases with increased aggregation. This implies that we can design a static backbone to support the traffic. The backbone will block incoming traffic only when the bit rate of an aggregated group deviates and increases beyond the allotted capacity. The probability of such blocking decreases with higher aggregation.

The probability of blocking can also be reduced by changing the static backbone to a more complicated flexible backbone that adjusts its state according to the backbone traffic. As the amount of aggregation increases, the backbone traffic appears more static, and a simpler backbone can be used for the same blocking probability. However, higher aggregation implies larger and more complicated local electronic networks. The trade off between the complexity of the backbone and the complexity of the local networks is an important and previously overlooked issue.

We can lower the probability of network blocking with a more costly overall network. At some point, when the probability of blocking is low enough, a more costly network is not desirable. For example, if the probability of encountering a busy end user is significantly higher than the network blocking probability, then it is not cost effective to further reduce the probability of network blocking. Also, when the probability of blocking is small enough, traffic modeling error becomes important. The precision of the failure probability depends on the accuracy of our traffic model.

The issues of how much to aggregate, the complexity of the backbone, and the trade off between the two, will be investigated as we consider blocking probabilities of these networks.

2.2 TDM in a long distance AON

As alluded to previously, TDM in an AON, where each time slot channel can be routed independently, is difficult. To achieve full TDM, the AON needs switching nodes that can switch at high speeds (small fractions of a time slot duration) and time slot alignment between connecting nodes. Time slot alignment is difficult because long distance fiber suffers length contractions and expansions from temperature variations, and optical buffers do not exist. Applying TDM on top of WDM is even more difficult because the effective length of a fiber is dependent on the wavelength (dispersion). Even if one is able to control the effective length of a fiber, there are also a variety of alignment types that may lead to different network connectivity. Appendix A investigates this issue in more detail.

However, limited TDM in an AON is possible, even on top of WDM. The timing requirement is greatly reduced if no slot switching is allowed in the intermediate nodes and the slots on different wavelengths do not interact with one another. In this case, slots join TDM streams, and all slots on the same stream are routed the same way. There is no interaction between the TDM streams except at the transmitters and the receivers. The only timing required is for the transmitters to transmit at the correct time so the data will arrive at the correct slots in the TDM streams, and for the receiver to listen for the desired slots. There might be slot conflicts at the transmitting and receiving station, i.e., a receiver may have to listen to more than one wavelength at a given instant. Multiple transmitters/receivers, one for each wavelength, can be used to solve this problem.

Limited optical TDM is best viewed as another level of aggregation done within the backbone in addition to external electronic aggregation. This concept will be exploited in section 3.4. The problem of time slot alignment for a general AON without optical memory is briefly discussed in Appendix A.

2.3 The network model

This chapter develops the network and traffic model used for the hierarchical network supporting aggregated small rate users. We envision an optical backbone that interconnects local networks (see figure 2-1). The traffic between two different local networks are called *inter-network traffic*. The local networks aggregate the long distance connections which make up the inter-network traffic, and transport those connections via the backbone.

We assume that the local networks are ATM networks though the backbone developed will be transparent to any local networks that have the proper input/output requirements, (see section 2.3.2). The optical backbone is an all optical network (AON), i.e., no optical-electrical conversions exist inside the backbone. In general the ATM networks may vary in size, and are typically large in order to achieve the type of aggregation required. *AON ports* are the interfaces between the ATM networks and the backbone. Each AON port is connected to one ATM network.

An ATM network connects to an AON port by connecting one or more ATM switches to the port using *access-links*. Access-links are logical connections. In prac-

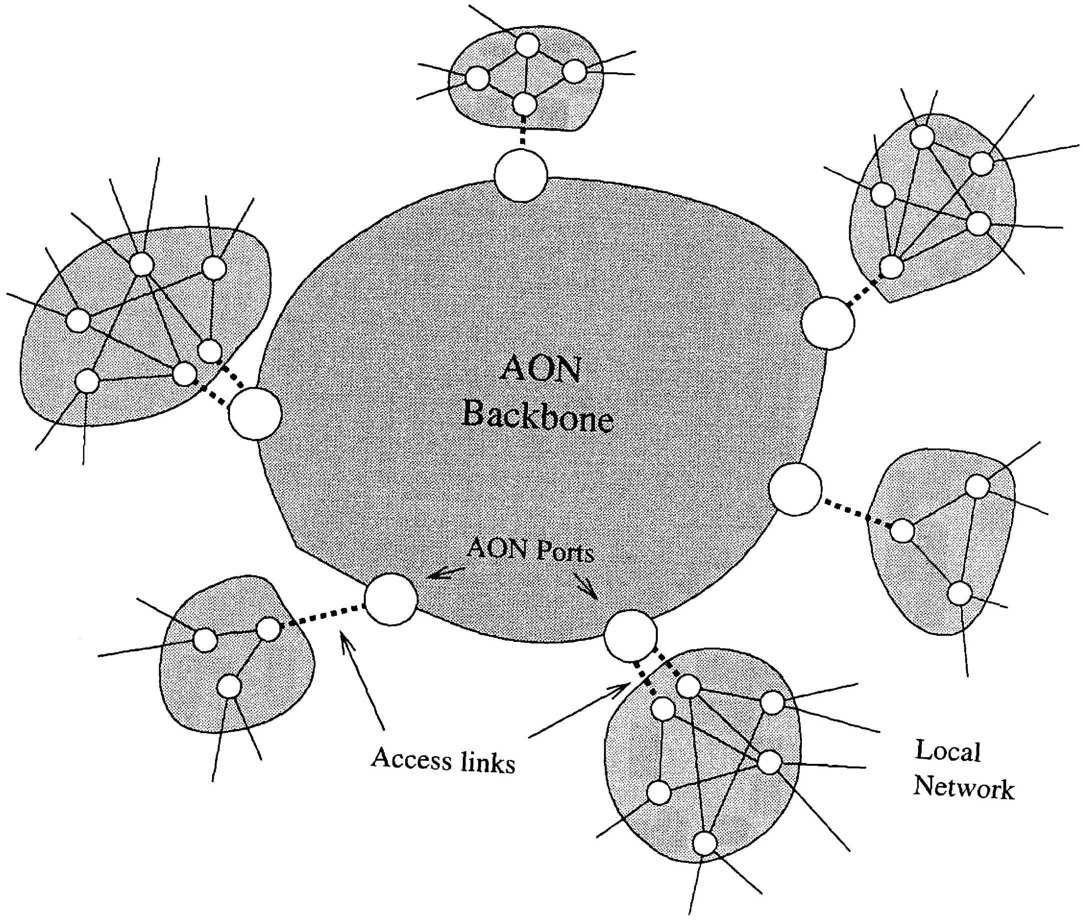


Figure 2-1: Optical backbone connecting ATM networks. Dotted lines denote multiple connection lines.

tice, multiple access-links can be multiplexed onto one physical wire or fiber. The important requirement is that the ports can easily identify and demultiplex the multiplexed access-links.

This thesis concentrate on the backbone design and network performance analysis, rather than the specifics of the local networks. Therefore, the local networks have the freedom to decide how to connect themselves as long as the conditions outlined in section 2.3.2 are satisfied.

In general, there are multiple fibers connecting each AON port to the inside of the AON. The number of fibers in each connection depend on the inter-network traffic requirements. Only the aggregated traffic from port to port will effect the backbone topology. This thesis considers various port to port traffic requirements as described in section 2.4.

2.3.1 The basic channel

The AON backbone sets up *bit pipes*, routes of a certain bandwidth, from one AON port to another. The local networks utilize these bit pipes to transport inter-network traffic. The capacity of a bit pipe is an integer multiple of a *basic channel.* A basic channel is the smallest unit of capacity that can be routed independently in the AON. Each basic channel can be routed differently from other basic channels, but all data within the same basic channel are routed the same way. If the backbone utilizes wavelength division multiplexing (WDM), then each wavelength is one basic channel. If time division multiplexing (TDM) is used on top of WDM, then each basic channel is one TDM slot in one wavelength.

2.3.2 Local Network Specification

The local network is responsible for routing local traffic and preparing long distance traffic to be routed by the AON. On the transmitting end, the local network aggregates long distance packets into bit streams according to the destination, and sends the aggregated streams to the AON backbone. On the receiving end, it accepts aggregated streams from the AON backbone, deaggregates the bit stream into packets, and routes the packets to the appropriate destinations.

The specifics of the local network are not crucial as long as the following requirements are satisfied: 1) On the transmitting end, the local network provides bit streams of size no greater than one basic channel, and sends each bit stream to the port via an access-link. 2) On the receiving end, the local network must have the ability to receive bit streams, via the access-links, of constant rate up to one basic channel. 3) The aggregation is done such that all data on the same access-link are destined to the same receiving port, (although several access-links may have the same destination). 4) There is some control communication between the local network and the backbone such that the local network can request a long distance connection, and the backbone can instruct the local network when and over which access-link to send the data.

For example, if the local network is an ATM network, and the basic channel rate is the same as an OC-3 line in a standard SONET (Synchronous Optical Network) link, then a possible implementation is depicted in figure 2-2.

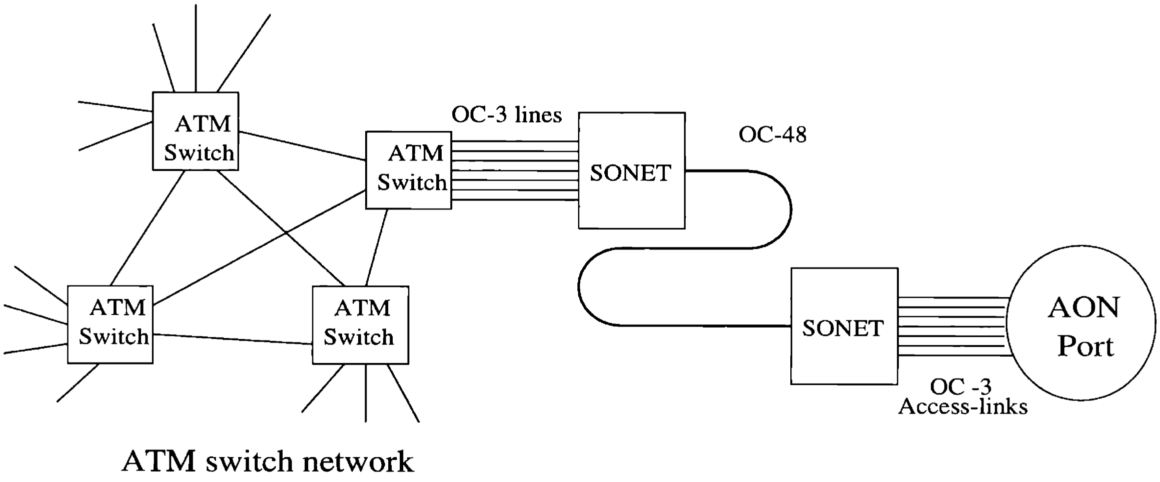


Figure 2-2: Example of a local network consist of ATM switches. The basic channel rate is the same as an OC-3 rate.

The ATM network connects to the port via OC-3 lines which act as access-links. For a given backbone configuration, the data destination is determined by the access-link used. The destination for a given access-link is changed only when the backbone is reconfigured. (Backbone reconfiguration includes tuning of transmitters and receivers within the ports.)

By routing the ATM packets, the local network aggregates packets for the same destination on the same OC-3 lines. For ease of transport, in case the AON backbone port is relatively far from the local network, sixteen OC-3 lines are multiplexed together into one OC-48. At the end of the OC-48 line, a SONET switch demultiplexes the data back to OC-3 streams and puts each stream on the appropriate access-link.

If the rate of the data streams on the access-link is smaller than one basic channel, then capacity is wasted because the AON backbone cannot route a partial basic channel. In that case, the AON treats the incoming data stream as one full basic channel. To avoid waste, the system should be designed such that the rate of the data streams matches that of a basic channel rate.

2.3.3 AON Ports

This section describes the AON port design. For simplicity, we assume that the basic channel is one wavelength, i.e., the backbone utilizes WDM, but not TDM. AON port design for a backbone utilizing TDM is more complicated, and it will be described in section 3.4.3.

The AON ports are the interfaces attaching ATM networks to the AON backbone. For convenience, separate the transmitting part of a port from the receiving part. The transmitting part will be called a *t-port* and the receiving part will be called an *r-port.* The words "port" will be used to denote an entity that contains one t-port and one r-port. Denote N as the number of ports, N_t as the number of t-ports, and N_r as the number of r-ports. In most cases, $N_t = N_r = N$, though we will not restrict our study to this case. Figure 2-3 shows one possible physical implementation of a t-port and an r-port.

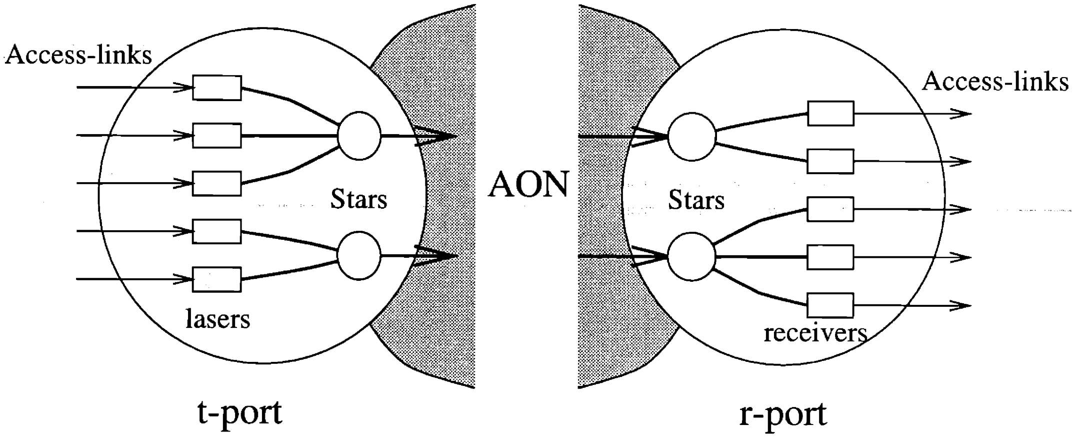


Figure 2-3: Physical implementation of a t-port (left) and an r-port (right). In the t-port, data from the access-links modulate the lasers. The laser output is transported through the AON backbone to an r-port. In the r-port, the receiver receives data from the backbone and converts it to data streams ready to be accepted by the ATM network.

Local ATM networks group long distance packets into bit streams according to the destination and route the bit streams to the appropriate access-links on the t-ports. The size of each bit stream is at most one basic channel. If the amount of traffic for the same destination is more than one basic channel, then multiple access-links are used for that same destination. Each t-port contains many laser transmitters. (There are a variety of lasers commercially available including distributed feedback (DFB) lasers, and distributed bragg reflector (DBR) lasers. A good reference on optical device technology is [ST 91].)

The laser is modulated by the data on the attached access-link. Depending on the modulation method and whether the bit-streams are electrical or optical, optical to electronic conversion and data format conversion may be needed before the bit streams can modulate the lasers. In any case, the laser output will carry the data information contained on the access-link.

The backbone configuration, including the transmitter and receiver tuning state, sets up bit pipes and determines the destination r-port for each optical bit stream. The r-ports provide a receiver for each optical bit stream received, and convert the bit streams to the appropriate format compatible to the local network. After conversion, the bit streams are sent via the access-links to the local network. The receiving local ATM networks then sort the received data and route them to their final destinations.

In essence, the AON backbone creates bit-pipes of size equal to one or more basic channels between t-ports and r-ports. Cooperating with the AON backbone, the local networks utilize the bit-pipes to transport their long distance traffic. For example, to set up a long distance connection from local network LAN-t to local network LAN-r, first, LAN-t sends out a connection request specifying the destination and the size of the connection (in multiples of one basic channel). Then, assuming the LAN-r agrees to the connection, the AON finds (or creates) a bit-pipe of the appropriate size connecting from the t-port of LAN-t to the r-port of LAN-r. Finally, the AON informs the local networks which access-links to use to receive the data.

The AON backbone limits the size of the bit stream on each access-link to be at most one channel. Therefore, if the instantaneous long distance traffic wanting to use the same access-link exceeds the size of one basic channel, then those packets should be queued up within the local network. (This is assuming no other access-link that goes to the same destination is available.) At no time should the traffic on an access-link exceed one basic channel rate. Similarly, the receiving local network must have the capability of receiving bit streams of size equal to the channel rate.

A *star coupler* is an s x s device such that each output simultaneously receives all the inputs with the power reduced by a factor of s. A star coupler acts as a s x 1 wavelength multiplexer when only one of the outputs is used, and all inputs operate on different wavelengths. We utilize stars as wavelength multiplexers in the t-ports. Star couplers multiplex the transmitter outputs of different wavelengths onto the same fiber so the fiber capacity can be fully utilized.

R-ports use star couplers in conjunction with coherent receivers as wavelength demultiplexers. Bit streams arrive from the backbone to the stars in the r-ports. Only one input of each star is used. The stars split the active input evenly among the outputs, and each tunable receiver listens to the appropriate wavelength. The receiver then converts the data to the appropriate form. The converted data streams are then passed to the ATM network, which deaggregates the streams into packets and routes the packets to the appropriate final destinations.

The transmitters and receivers within the ports are generally tunable, although in most cases they can be fixed. This is because aggregated traffic appears static and the inter-network traffic does not change. This means that bit pipes, which are set up according to inter-network traffic, need not be changed. If a particular bit stream is directed to a particular destination, then this bit stream will simply be routed by the local network to the access-link that is connected to a transmitter transmitting to the correct destination r-port.

The number of stars used in an AON port depends on the throughput of the AON port and the topology of the backbone. The most efficient network has its stars filled to capacity.

In summary, AON ports are interfaces between the backbone and the local ATM networks. The t-ports receive data streams from the ATM networks and put them on the appropriate bit-pipe to be transported by the backbone. Similarly, the r-ports demultiplex data received from the backbone into different streams and send them to be further processed by the destination ATM networks.

2.4 The traffic model

The traffic model developed here describes the data intensity between t-ports and r-ports. Since the traffic at a t-port is the aggregation of many small users, it is reasonable to assume that each t-port requires connections to multiple r-ports. Define this type of traffic to be *multiple connection* traffic. Note that this is different than multi-casting. In multi-casting, the same data is sent to a multiple number of destinations. In multiple connection traffic, there are independent connections originating from the same place. All inter-network traffic patterns considered in this thesis are of this type. Most network studies assume either point to point or broadcast/multi-cast traffic. Multiple connection traffic has not been previously analyzed.

Let $tr(i,j)$ be the traffic from AON port i to AON port j, i.e., the data rate for connection (i,j). The collection of $tr(i,j)$ $\forall$ i,j defines the traffic matrix.

Another way of representing traffic, besides using a traffic matrix, is by a directed graph $G(V,E)$. V is the set of vertices representing the AON ports. E is the set of directed edges representing the traffic between the ports; the weight of the edge from i to j is simply $tr(i,j)$. A zero weight edge is equivalent to no edge at all.

Each $tr(i,j)$ actually contains the aggregation of many end-to-end sessions, but for the backbone design, only the values of $tr(i,j)$ are important. We are concerned with three different types of traffic matrix: *uniform all-to-all traffic*, *fixed multiple connection traffic*, and *variable multiple connection traffic*.

2.4.1 The uniform all-to-all (UATA) traffic model

In uniform all-to-all (UATA) traffic, $tr(i,j) = c$, where c is constant for all i and j. Every t-port communicates with every r-port, and each connection contains the same amount of traffic. Let N_t be the number of t-ports and N_r be the number of r-ports. There are $N_r N_t$ connections in all. In the graph representation, UATA traffic is a complete graph with edge weight of c.

This traffic model assumes that the local networks have aggregated enough traffic from the end users such that the traffic flow from one AON port to another is constant.

This is reasonable because one would expect the traffic intensity between two major cities like New York and Los Angeles to vary slowly. The aggregation is assumed to be uniform so $tr(i, j)$ is the same for any pair of AON ports. (i.e., small cities aggregated together so the resulting traffic profile matches that of a big city like New York.)

For simplicity, our study assumes $tr(i, j)$ is the same for $i = j$ as for $i \neq j$. This is not particularly realistic, but is unimportant when the number of ports is large.

Since the UATA traffic does not change over time, the backbone required can be passive. The proposed backbone contains passive components that take advantage of the uniformity of the traffic. The backbone simply provide a static bit pipe between every port pair.

Even if the UATA model does not reflect the real traffic, it is still beneficial to study this model. A network built to support UATA traffic with $tr(i, j) = c$ can be used to provide connections between port to port, but only with a guaranteed rate of c. In this case, sessions that cause $tr(i, j)$ to increase beyond c will be blocked or delayed. If $tr(i, j) < c$, then bandwidth is wasted on link (i, j). We would like to increase c to reduce the probability of blocking, but increasing c will cause the bandwidth of many links to be wasted. This type of trade off will be investigated in chapter 4, where we develop a traffic model from the end users' point of view. In that model the expected inter-network traffic is UATA, but the actual traffic has a finite probability of overflowing the bit-pipes in the AON backbone.

Another reason for studying the UATA traffic model is its simplicity. UATA traffic provides us the first step of research before more general models are studied. The next two sections describe successive generalizations of this model.

2.4.2 The fixed multiple connection (FMC) traffic model

The first generalization of the traffic model is the Fixed multiple connection (FMC) traffic. FMC traffic allows $tr(i, j)$ to depend on i and j. However, as the name suggests, $tr(i, j)$ for a given i and j is constant over time. Note that UATA is simply a special case of FMC.

Since the FMC traffic is constant over time, a passive backbone can be used. The backbone sets up fixed bit-pipes between each port pair. Each bit-pipe has fixed capacity large enough to accommodate the traffic between the connected port pair. Unfortunately, unlike the UATA network, there is no nice symmetry in the traffic. The size of each bit-pipe depends on the values of $tr(i,j)$ of the given FMC traffic.

A FMC network can be practical even if the traffic between two ports is not constant in reality. In that case, $tr(i,j)\ \forall\ i,j$ are the assumed design parameters, and the network is built accordingly. The sessions that cause the traffic on connection (i,j) to exceed that of the designed $tr(i,j)$ will be blocked or delayed. One would like to choose $tr(i,j)$ to be large in order to reduce the amount of blocking, but small enough to not waste network resources.

2.4.3 The variable multiple connection traffic model

Variable multiple connection traffic (VMC) is a further generalization of the traffic mixes described so far. Instead of restricting $tr(i,j)$ to be a fixed constant, it allows some variations. However, since $tr(i,j)$ is the aggregation of a large amount of traffic, one would expect that the rate of change is slow. It might be sufficient to update the traffic matrix on an hourly basis. The AON has ample time to reconfigure itself to support the changing traffic demand. The problem in this case is to design a backbone network that has flexible port to port connections. The bit-pipes in the VMC network can be switched between different port pairs according the the changing traffic.

Chapter 3

The Uniform All-to-All (UATA) Network

This chapter describes how a uniform all-to-all (UATA) backbone network can be built to support UATA traffic. The network is efficient and scalable. The only device used in the UATA network is the Latin Router which is described next.

3.1 The Latin Router

The *Latin Router (LR)* is a wavelength routing device that routes the input to a particular output according to the wavelength. LR's can be implemented physically on integrated silicon as described in [Dr+ 89], and its technology continues to improve. We are interested in the functionality of LR's, and assume perfect implementation. The functionality of an LR is fully described in [Bar 93]. We will briefly describe it here.

Let F be the number of wavelengths used in the WDM backbone. Let k be a divisor of F. Define k as the coarseness of the LR. A LR of coarseness k has F/k inputs and F/k outputs. The inputs of the LR are numbered from 0 to $F/k - 1$ beginning with the uppermost input and the outputs are numbered in the same fashion. Wavelength are numbered from 1 to F in the order of increasing wavelength. Wavelength w on input i will be routed to output j if and only if $j = i+w-1 \bmod F/k$.

Figure 3-1 shows an example of a LR with $F = 4$ and $k = 1$.

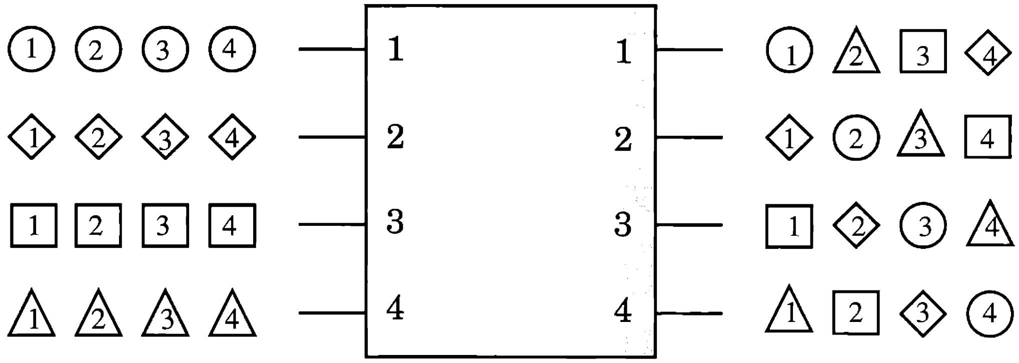


Figure 3-1: A 4 x 4 Latin Router. The number indicates the wavelength of the data, and the shape indicates the input position.

Consider a LR with $k = 1$. Effectively, the LR provides a bit pipe of size equal to one wavelength channel between each possible pairing of an input and an output. Therefore, up to F^2 simultaneous connections can be supported without blocking. If R is the bit rate of each wavelength channel, then the capacity of the LR is F^2R.

If $k \geq 1$, then the LR is a F/k x F/k device providing a bit pipe between every pairing of an input and an output. The size of a bit pipe is k wavelength channels. In this case, the LR capacity is $\left(\frac{F}{k}\right)^2 kR = \frac{F^2R}{k}$.

The spectral separation between two wavelengths depends on R. Larger R implies a larger separation. Since the useful capacity of a fiber is limited, this leads to a smaller F. Therefore, for a constant k LR, R can be increased only if the number of inputs/outputs decreases. This trade off is useful in adjusting the size of the bit-pipes. This trade off can also be accomplished by keeping R constant and changing k. This method is especially useful if the size of a wavelength is restricted by the electronic modulators. In this case, each input is connected to each output by a bit pipe of size equal to k wavelength channels. Increasing k increases the size of the bit pipe but decreases the number of inputs/outputs, which is F/k.

For the backbone, we will be looking at the bit-pipe connections. The LR provides bit-pipe connections between all input-output pairs. This bit-pipe can be viewed as one wavelength with rate kR, or k wavelengths with rate R. There is no logical difference between the two views when considering the bit-pipe connections within the LR. However, $k > 1$, gives the network the added ability of splitting the bit-pipes into smaller units. This added ability is not necessary in the UATA network because all bit-pipes are of the same size. Therefore, when considering the topology of the UATA backbone, $k = 1$ will be assumed.

Also, for $k > 1$, the port design is different. In this case each bit-pipe can only be utilized fully if k lasers are operating simultaneously. Consequently, k access-links should be connected to the same bit-pipe, one for each laser. Furthermore, each local network puts out k bit streams for the same destination r-port.

3.2 All-to-all network design

Let the number of t-ports and r-ports be the same; $N_t = N_r = N$. In the UATA network, the traffic between every pair of AON ports is constant; $tr(i, j) = c$. We can assume $c = R$ because R can be adjusted. A design for c as a fraction of R is described in section 3.4. Chapter 4 will treat the case where $tr(i, j)$ is stochastic and varies around its mean.

For $N \leq F$ the network design is simple. Since the LR provides a bit pipe of rate R between each possible pairing of input and output, by placing the t-ports on the inputs of the LR and the r-ports on the outputs of the LR, the LR provides a dedicated bit pipe for each connection desired. Therefore, a single LR provides UATA connections for up to F AON ports. In the case that $N = F$, the number of connections demanded by the traffic is $N^2 = F^2$, which equals the maximum number of connections provided by one LR. The LR is used to its capacity for $N = F$.

For $N > F$, group the t-ports in groups of F and call each group a t-group. Similarly, form r-groups of size F with the r-ports. There are $\lceil \frac{N}{F} \rceil$ t-groups and $\lceil \frac{N}{F} \rceil$ r-groups. Select any t-group and any r-group. The connection required between these two groups forms an F x F UATA traffic matrix. This is efficiently provided by a single LR. One LR is required to connect each possible pairing of a t-group and a r-group, resulting in a total of $\lceil \frac{N}{F} \rceil^2$ LR's. Figure 3-2 shows the resulting network for $F = 4$ and $N = 8$.

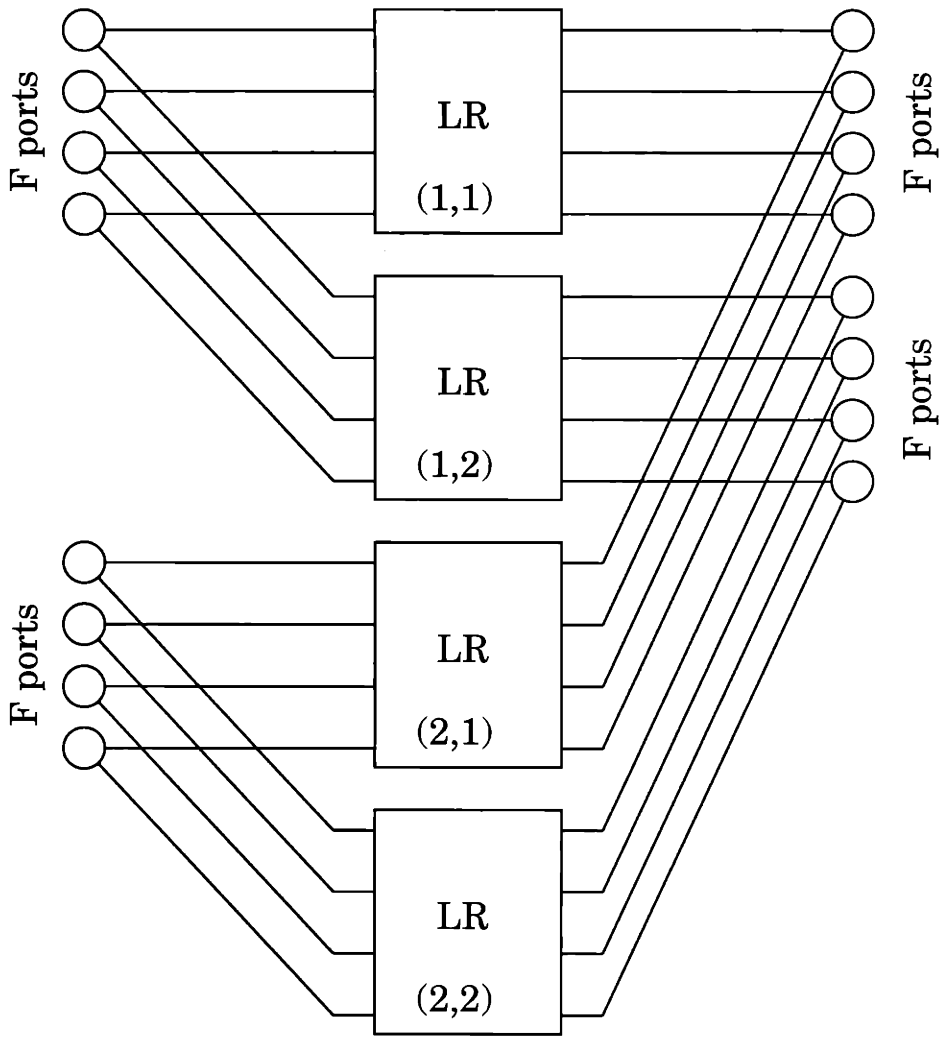


Figure 3-2: The all-to-all network. The AON ports are grouped into groups of size F. Each t-group is connected with each r-group via a LR.

The above design can be extended to the case where $N_t \neq N_r$. In this case, the number of t-groups is $\lceil \frac{N_t}{F} \rceil$ and the number of r-groups is $\lceil \frac{N_r}{F} \rceil$. $\lceil \frac{N_t}{F} \rceil \lceil \frac{N_r}{F} \rceil$ LR's are needed, one for each possible pairing of a t-group and an r-group.

If $\frac{N}{F}$ is not an integer, then there is one t-group and one r-group each of which contains less than F ports. This will cause inefficiency because the LR's connecting to the incomplete groups are not filled to capacity. This effect is discussed in section 3.5.

There are multiple fibers connecting each AON port to the LR's. This is required because each AON port is connected to a multiple number of LR's. Specifically, each t-port is connected to $\lceil \frac{N_r}{F} \rceil$ LR's, and each r-port is connected to $\lceil \frac{N_t}{F} \rceil$ LR's. The design of the AON port allows multiple fibers. The general port design is described in section 2.3.3, and the physical implementation is illustrated in figure 2-3. For the UATA traffic, the design is simpler. Since every frequency is used simultaneously, the transmitters/receivers are not required to be tunable. Figure 3-3 shows the implementation of a t-port and a r-port for $F = 4$ and $\lceil \frac{N_r}{F} \rceil = 2$. Here, groups of $F = 4$ transmitters, each operating on a fixed but different frequency, are combined together by a multiplexer, which is implemented by a star coupler (see section 2.3.3). The number of stars, or equivalently the number of out going fibers per port, is $\lceil \frac{N}{F} \rceil = 2$. If $N_r \neq N_t$, then the number of stars in a t-port is $\lceil \frac{N_r}{F} \rceil$ and the number of stars in an r-port is $\lceil \frac{N_t}{F} \rceil$

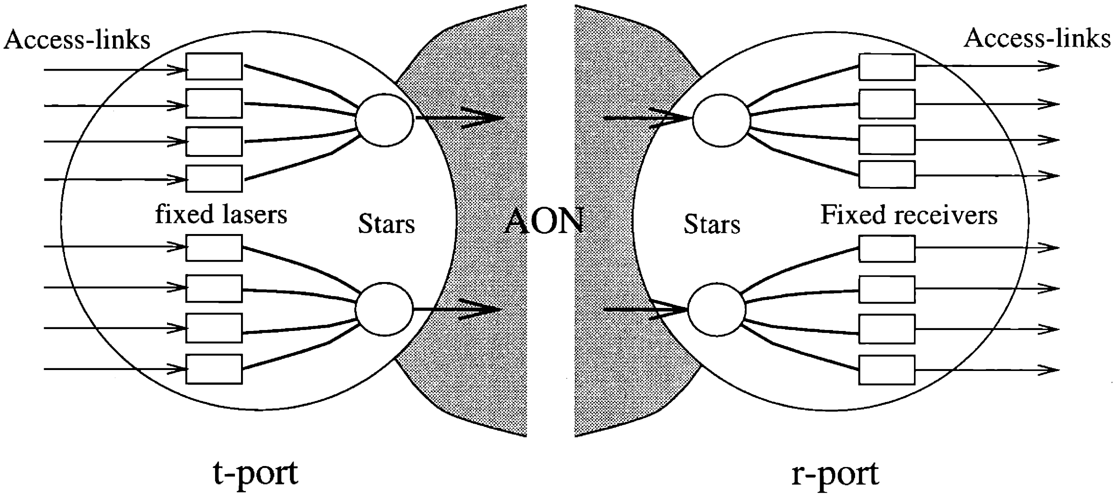


Figure 3-3: Example of AON ports for the UATA network. Note that the transmitters and receivers do not need to be tunable. The attached ATM is responsible in choosing the correct input to the port. There are F transmitters/receivers attached to each star.

It is the responsibility of the attached ATM network to route the packet to the correct access-link, on a t-port, that is connected to the desired fixed tuned transmitter. The optical backbone is a passive device, providing a bit pipe connection between every pair of AON ports. Therefore, the selected transmitter determines the destination port of a packet. The receivers are fixed tuned. Each access-link on an r-port contains data from the same fixed t-port origin.

3.2.1 UATA network with very large k

We have designed a UATA network where there is a bit-pipe connecting each t-port to each r-port, and each bit-pipe contains k wavelengths. The coarseness k, can be any integer fraction of F, where F is the total number of available wavelengths on a fiber. There may be cases where $k > F$ is desirable. In those cases, the UATA simply connects each port pair by $\lceil k/F \rceil$ fibers. The conservation of fibers using LR's no longer exists. We are more interested in cases where $k < F$. Also, most applications today are small enough that small k will lead to enough aggregation for our purpose.

3.3 Scalability of the UATA design

This section demonstrates the scalability of the UATA network designed in the previous section. Here, scalability is defined as the ability to grow the network by adding components and not tearing down any existing hardware. The UATA network is scalable, and the number of ports is increased at increments of F.

Figure 3-4 demonstrates how an N port UATA network can be built from an $N - F$ port UATA network. Increasing the number of ports by F causes the addition of one t-group and one r-group. Because of the added groups, the network needs additional LR's to connect all the additional pairings of the groups. The resulting N port UATA network contains the original $N - 1$ port UATA network. No existing connection nor component is deleted in the process. Therefore, the UATA network is scalable.

Note that asymmetric scaling is possible. One can increase the number of t-ports by pF and, at the same time, increase the number of r-ports by qF where p and q are non-negative integers not necessarily equal to each other. There are p additional t-groups and q additional r-groups. The network requires additional LR's to connect the added pairings of the groups.

The hardware in the original AON ports need not be changed because scaling does not delete any old connection but only adds new ones. Therefore, if the number of r-ports is increased by qF then the original t-ports will have to add qF access-links

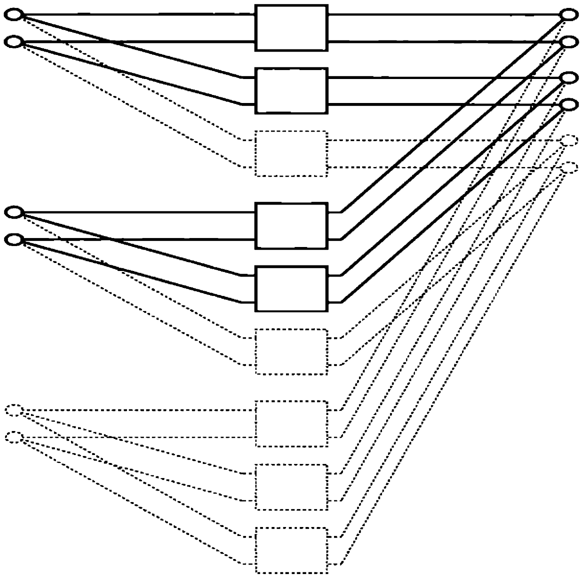


Figure 3-4: Scalability of the UATA network. The UATA network of N ports contains an $N - F$ port UATA network.

and lasers, and q stars. Similarly, if the number of t-ports is increased by pF, then the original r-ports will have to add pF access-links and receivers, and p stars.

3.4 Time sharing All-to-all Backbone

As mentioned in the introduction to Part I, it is difficult to independently route TDM channels in long distance networks. However, we can still employ TDM if all the TDM channels on one fiber of the same wavelength are combined locally and kept together throughout the network until they reach the destination. This way, we avoid the timing problem associated with independent routing of TDM channels. Effectively, we are using TDM as another level of traffic aggregation. This TDM aggregation is done in the optical domain.

In this section, each wavelength channel is further divided into T time slots. The basic channel is a TDM slot on a wavelength, and has average rate R/T. As will be clear later, independent routing of TDM channels is not required and the construction is feasible. The AON port design and scheduling is more complicated because the transmitters/receivers need to be aware of the assigned time slot. Section 3.4.3

describes the port design in more details.

The main idea in time sharing is to take a UATA network, and apply TDM on it. Specifically, take a UATA network with 1 wavelength as the port to port bit-pipe size, and apply TDM so multiple ports can share the same bit pipe. This sharing is implemented by a device called a *multiple access star cluster* (MASC). We now describe specifications for a MASC.

3.4.1 Multiple access star cluster (MASC)

MASC's are simple devices made of fibers and stars. Examples are shown in figure 3-5. MASC's come in two flavors: combining MASC's and broadcasting MASC's.

A combining MASC has Mb input fibers and b output fibers. Each star of the combining MASC multiplexes M input fibers onto one output fiber. One can view a combining MASC as b optical multiplexers in parallel.

Similarly, a broadcasting MASC has M input fibers and Mb output fibers. Each star broadcasts an input to M outputs.

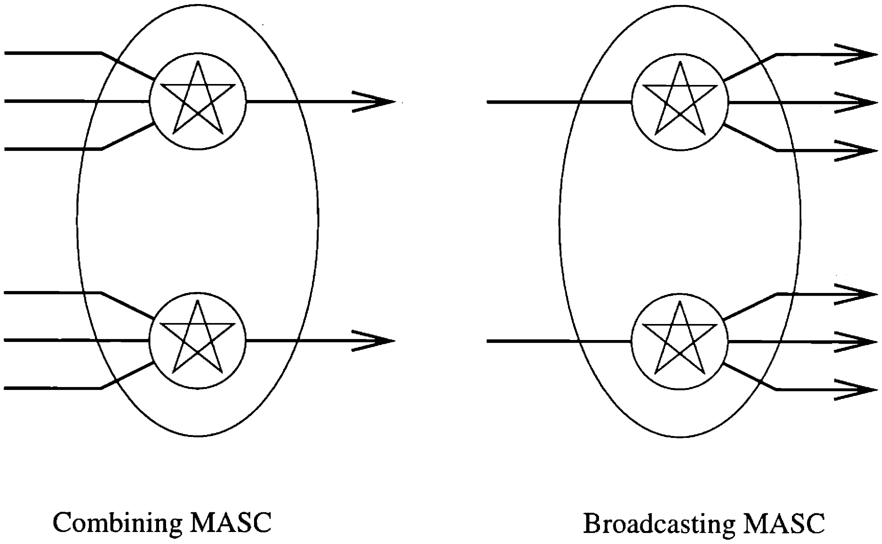


Figure 3-5: Multiple access star cluster, MASC's. The left one is a combining MASC, the right one is a broadcasting MASC. For both MASC's, $M = 3$ and $b = 2$.

The purpose of the MASC is to allow multiple ports to time share the same wavelength channel. In the UATA backbone constructed so far, each bit-pipe consists of a wavelength channel connected between a t-port and an r-port. By replacing the ports with MASC's and allowing multiple ports to connect to the MASC, the wavelength channel can now be time shared among different pairings of the ports. TDM breaks up the wavelength channel into time slotted pieces. This is the basic idea in the TDM UATA backbone design.

3.4.2 TDM UATA backbone design

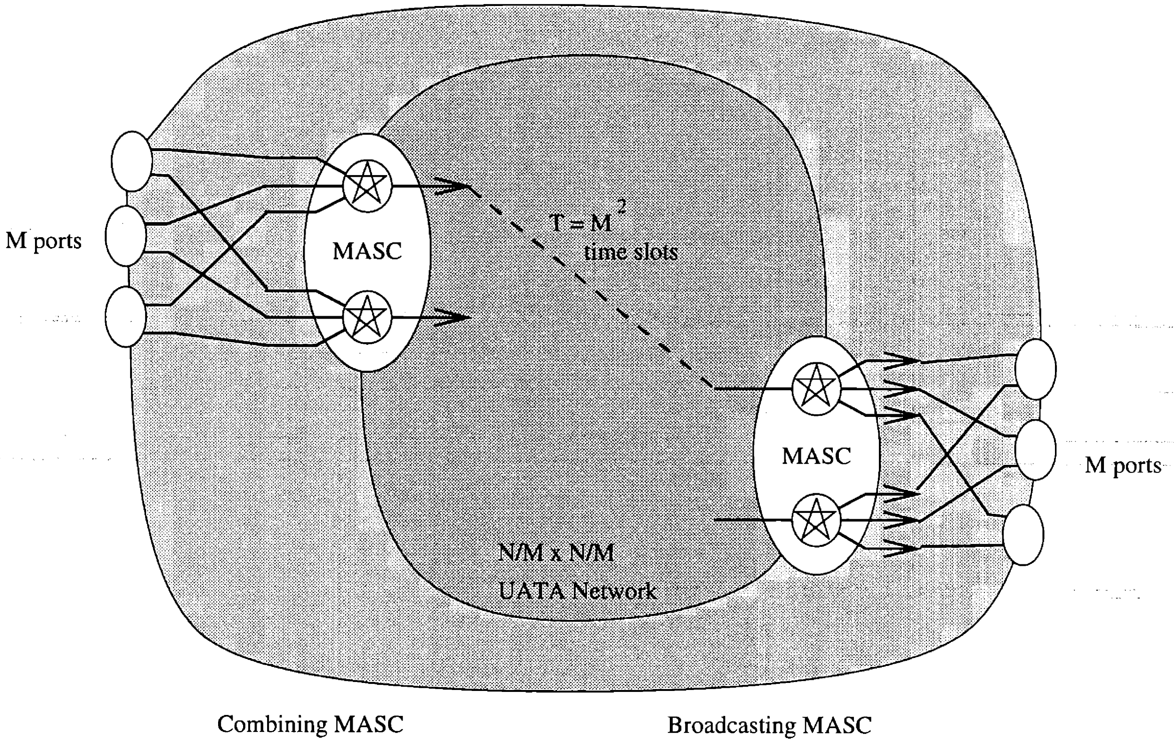


Figure 3-6: Time shared uniform-all-to-all network. Clusters of M transmitters share the same bit pipes. Only one t-cluster and one r-cluster is shown.

Let $M = \sqrt{T}$ where T is the number of time slots. To build a TDM UATA backbone, first, take an $\left\lceil \frac{N_t}{M} \right\rceil$ x $\left\lceil \frac{N_r}{M} \right\rceil$ regular UATA network. Next, replace all the original t-ports with combining MASC's and the original r-ports with broadcasting MASC's. In order to do the replacement, b, for the MASC's, must match the number of connecting fibers in the original ports. This means for the combining MASC's, $b = \left\lceil \left\lceil \frac{N_r}{M} \right\rceil / F \right\rceil$, and for the broadcasting MASC's, $b = \left\lceil \left\lceil \frac{N_t}{M} \right\rceil / F \right\rceil$. Finally, connect M new t-ports to each of the combining MASC's and M new r-ports to each of the broadcasting MASC's. The construction is shown in figure 3-6.

By virtue of the $\left\lceil \frac{N_t}{M} \right\rceil$ x $\left\lceil \frac{N_r}{M} \right\rceil$ UATA network, each combining MASC has a whole wavelength bit-pipe connection to any broadcasting MASC. This wavelength is now broken up into smaller units, which are the time slots. Each of these time slots is assigned to a different pairing of the connecting ports. This implies the existence of a TDM channel from each t-port to each r-port. This is true for all pairs of MASC's. Therefore, the network supports UATA.

Since the backbone contains no switching, the only timing required is when the time slots merge in the combining MASC's. No time slot alignment is required between the different wavelengths. No slot collision is possible once the slots are correctly merged in the combining MASC's. In the r-ports, a receiver listens only on the assigned time slot. Each wavelength on each output of the star is connected to a dedicated receiver. Therefore, the different wavelengths on different fibers can be timed independently. Next, we discuss the necessary port modifications for using TDM.

3.4.3 Port modification for UATA backbone with TDM

In a TDM based UATA network, the basic channel is a TDM slot in a wavelength. The average rate of the basic channel is $\frac{R}{T}$, but the instantaneous rate is either R or 0. These bursty basic channels are the connections between a t-port and a r-port. This creates a problem for the local networks because the local network must know the timing of the slot and transmit the data accordingly. This problem can be avoided if buffers are installed on the access-links of the AON port, (see figure 3-7).

The role of the buffer is to allow the local networks to transmit at a steady rate of $\frac{R}{T}$ while the laser is transmitting at the average rate of $\frac{R}{T}$. Similarly, r-ports should have buffers in the access-links to store the bursty traffic from the backbone. The buffers will be emptied by the local network at a constant rate of $\frac{R}{T}$. With the buffers installed, the backbone will be more transparent to the local network. The local network will only need to know the average operating data rate, not the timing of the slots.

Furthermore, access-links that use the same wavelength can share the same laser.

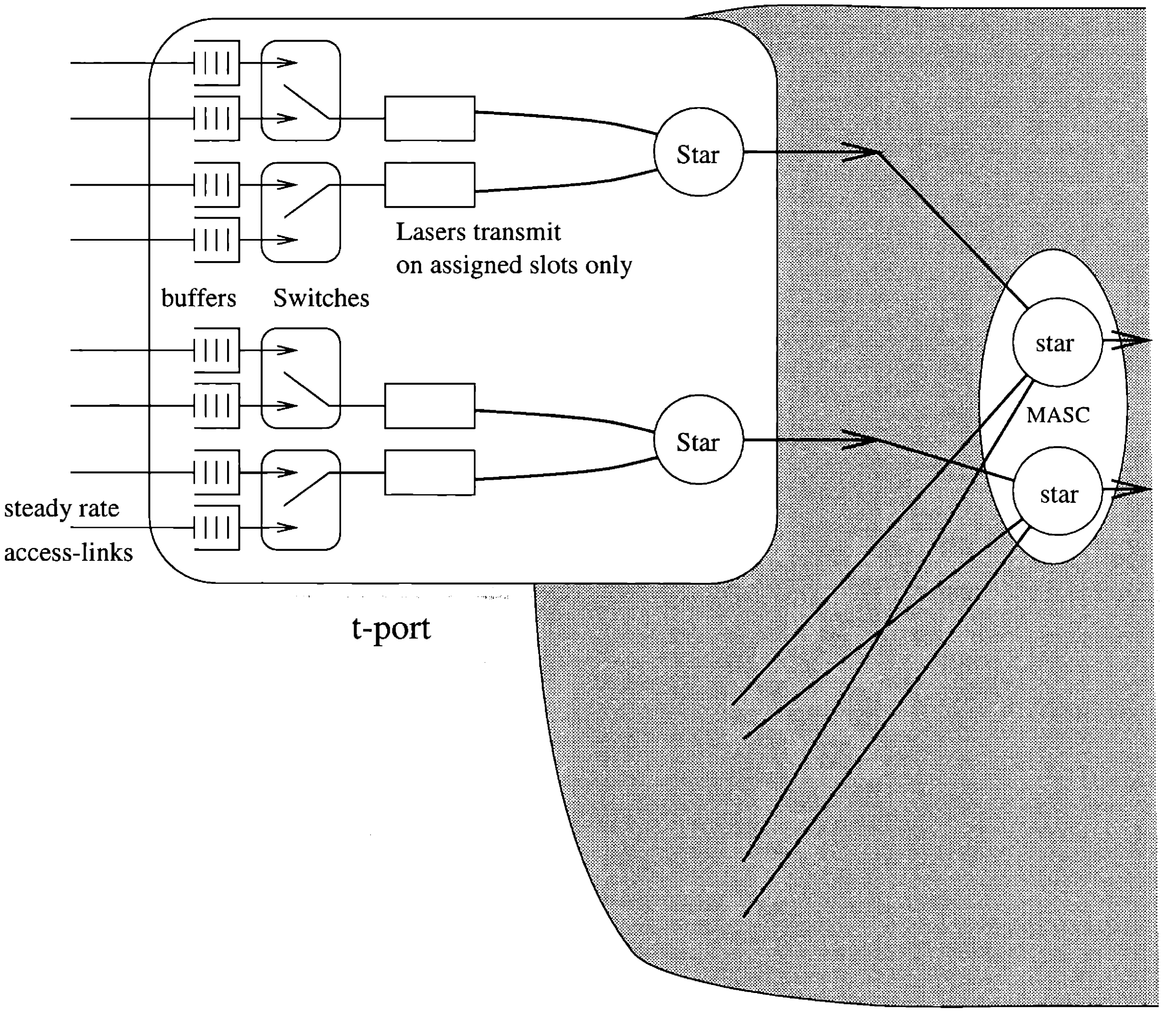


Figure 3-7: Port design of UATA network with TDM. Buffers are installed in the port-links to convert between steady and bursty rates. Switches are used to share lasers operating on the same frequency. Only one t-port is shown, r-ports are analogous. In this example, $F = 2$, $M = 2$, and $T = 4$.

This is done by using the M x 1 switch, (see figure 3-7). In each t-port, there are M access-links that are assigned to the same destination MASC because each destination MASC is connected to M r-ports. Therefore, these access-links use the same wavelength, and one laser can be shared among them. Similarly, access-links in the r-ports that receive data from the same MASC can share the same optical receiver because they use the same wavelength on the same fiber.

The total number of lasers required in a t-port is the number of destination MASC's, which is N/M. The total number of receivers required in a r-port is also N/M. Note that even with sharing of lasers, each laser is only active in M of the M^2 time slots. This is also true with the optical receivers. Therefore, further sharing of the transmitters or receivers is possible. This can be accomplished by allowing idle lasers/receivers to be used by other wavelengths. However, this adds complexity to the network because the network now needs tunable lasers/receivers, coordination between the ports, larger switches in the ports, and most critically, synchronization between the wavelengths. Therefore, this sharing between wavelength does not appear practical with current technology, and will not be done in our backbone.

3.4.4 Advantage of TDM

TDM increases the number of channels per fiber by reducing the bit rate per channel. This gives more flexibility to the network, and the advantages are as follows:

First, TDM allows a better matching between the basic channel rate to the port to port data rate c, and hence, better efficiency in the backbone. If c is much smaller than R, where R is the bit rate of one wavelength channel, then the design without TDM is very wasteful because the wavelength channels are not filled to its capacity. TDM divides each wavelength channel into TDM slots, and reduces the basic channel from one wavelength, R to one TDM slot, R/T. R/T can be adjusted, by changing T, to match that of c. A more efficient use of the basic channels results in a more efficient network, and a smaller number of LR's and fibers is required to support the same number of AON ports. In other words, efficient allocation of the available bandwidth can reduce the backbone size.

Alternatively, instead of reducing the backbone size, one can increase the number of ports supported. If we keep the number of LR's, i.e., the maximum throughput of the backbone, constant, then TDM creates more basic channels in our backbone, and this translates into an increase in the number of AON ports supported by the backbone. The number of ports is increased by a factor of M at the expense of reducing the port to port rate by a factor of M^2. At first, this may seem to be an unfair trade off, with the number of AON ports increasing linearly and the rate decrease quadratically. However, one must realize that a linear increase in the number of AON ports corresponds to a quadratic increase in the total number of sessions in a UATA traffic. Since we have not added any extra resources in the network, the port to port bandwidth must decrease quadratically.

Furthermore, TDM provides the opportunity for ports connecting to the same MASC to share their time slots. The UATA network without TDM is very rigid and unused bandwidth is simply wasted. With TDM, time slots not being used by one port can be used by other ports connecting to the same MASC. In fact, one can view TDM as another level of aggregation, and this aggregation is done in the optical domain. Of course, this advantage only exists if the time slot assignment is flexible. We will calculate the advantage of flexible slot assignment in terms of congestion probabilities in section 4.3.3.

Having another level of aggregation in the optical domain, as oppose to aggregating solely in the local networks, provides network architects more options in designing the network. It may be too costly to use local networks to aggregate to the nominal wavelength channel rate because large fast switches are required. Using TDM to aggregate in the optical domain may be more cost effective. Also, aggregating in the optical domain allows the AON ports to be physically far apart.

3.4.5 Slot re-use in UATA-TDM?

Figure 3-8 represents the logical connections in the UATA network using TDM. (The connections are logical because the wavelength pipes are drawn separately even though they may exist on the same fiber.) Each wavelength bit-pipe gathers the data from M t-ports, and then distributes the data to M r-ports. In this setting, M^2 time slots are needed on each wavelength pipe, one for each possible t-port r-port pairing. An interesting question that arises is whether or not one can save the number of time slots by staggering the inputs and outputs on the bit-pipes.

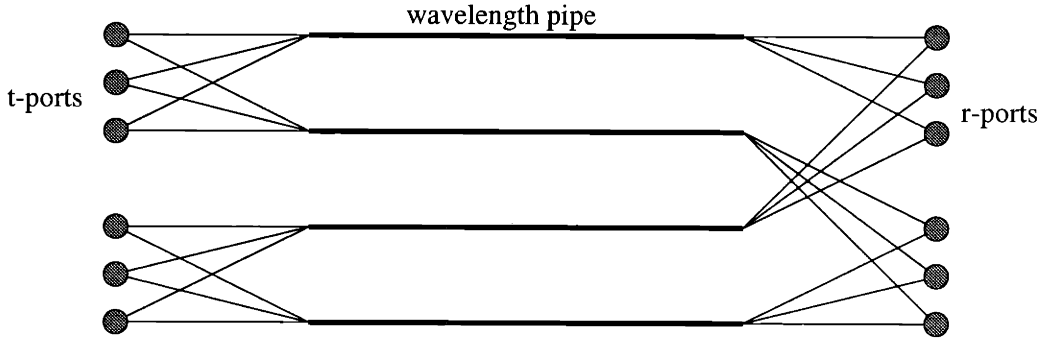


Figure 3-8: A logical UATA-TDM network. Each wavelength bit-pipe aggregates the traffic from M t-ports and transport the traffic to M r-ports.

Figure 3-9 shows a network such that the inputs and the outputs to the wavelength pipes are staggered. Conceivably, if an r-port extracts data from the early part of the pipe and empties one time slot, then the empty slot can be re-used by another connection further down the pipe. We want to investigate the amount of saving by strategically staggering the t-port and r-ports along the wavelength pipes. We will show that at most a factor of 2 saving in the number of time slots is possible. Therefore, the saving does not appear to warrant the greatly added complexity of the network, and our UATA-TDM network is reasonable.

For a fair comparison, the out degree of a t-port, (and the in degree of an r-port), must be the same between the staggered and the non-staggered scheme. Let d be the out degree of a t-port and the in degree of an r-port. Let T be the number of time slots required. For the non-staggered UATA-TDM network,

$$T = \left(\frac{N}{d}\right)^2. \tag{3.1}$$

This is because each wavelength pipe is connected to $\frac{N}{d}$ t-ports and $\frac{N}{d}$ r-ports. (For simplicity, we have assumed $\frac{N}{d}$ to be an integer.)

In the staggered scheme, a t-port is connected to the wavelength pipe via a coupler,

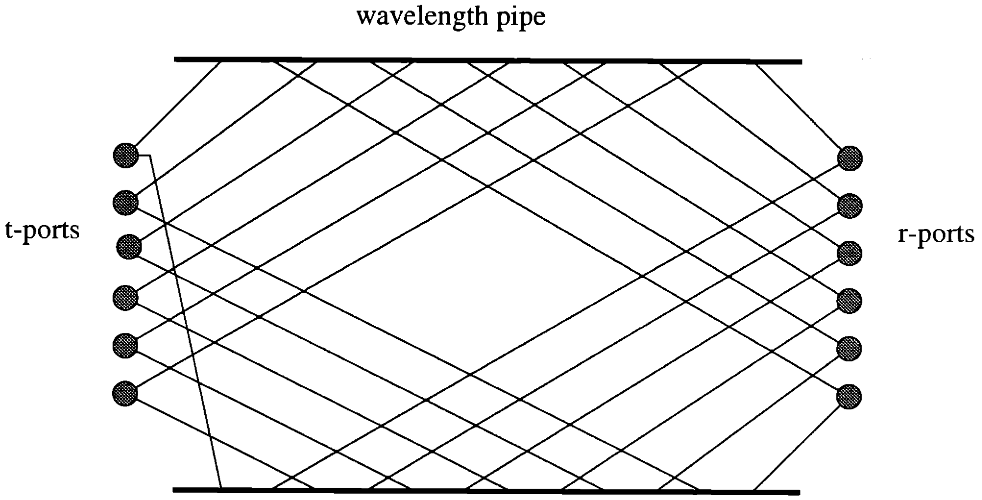


Figure 3-9: In a staggered network, the traffic enters and exists the network at various points of the wavelength pipe. Time slots used at some sections of the pipe may be re-used by other sections of the pipe.

(see figure 3-10). The data on the pipe is later extracted by an r-port, using an optical switch. Data are taken out of the pipe only when they are switched out by the optical switch. At one switch, some connections are being switched out to the r-port, while others are being passed down to the later part of the pipe. A port to port connection is *visiting* a switch if it either passes through a switch or is switched out by the switch. Each port to port connection visiting the same switch must use a unique time slot. Therefore, the number of time slots required is the maximum number of port to port connections visiting a switch where the maximization is taken over all switches. We now calculate this maximum.

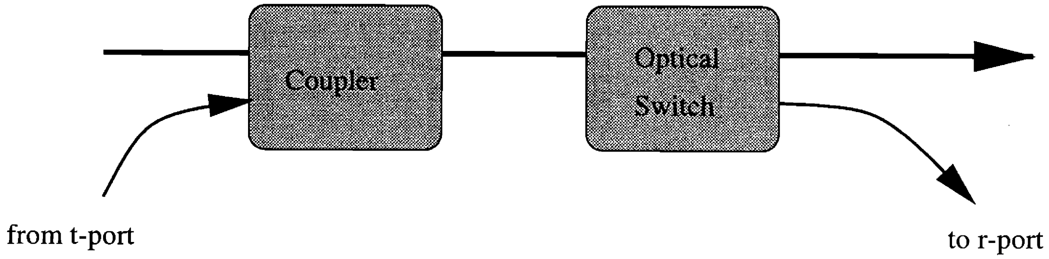


Figure 3-10: Traffic enters the pipe via optical couplers. Traffic exits the pipe by the optical switch.

Each t-port can reach d wavelength pipes. Hence, each t-port can reach at most d r-ports by visiting only one switch. At most dN out of N^2 of the UATA port to port connections can be established using just one switch visit in each connection. At most dN more port to port connections can be established using two switch visits, etc. The total number of switch visits over the N^2 connections is minimized if exactly dN connections use one visit, exactly dN connections use two visits, and so forth up to the final dN connections using $\frac{N}{d}$ visits. Therefore, the total number of switch visits of all N^2 port to port connections is

$$\geq dN + 2dN + 3dN + \ldots + \frac{N}{d}dN. \tag{3.2}$$

(Again, for simplicity, we assume $\frac{N}{d}$ is an integer.) The total number of switches is dN, which is the number of r-ports multiplied by the in degree of an r-port. Therefore,

$$\frac{dN + 2dN + 3dN + \ldots + \frac{N}{d}dN}{dN} = \frac{N}{2d}\left(\frac{N}{d} + 1\right) \tag{3.3}$$

is the average number of visits per switch. This means there is one switch where the number of visits on that switch is at least $\frac{N}{2d}\left(\frac{N}{d}+1\right)$. Therefore, the network requires at least

$$T \geq \frac{N}{2d}\left(\frac{N}{d} + 1\right) \tag{3.4}$$

number of time slots. This is at most a factor of 2 improvement over the non-staggered scheme.

Obviously, the staggered scheme is much more complicated to build and control than the non-staggered scheme. Since the added complexity can achieve at most a factor of 2 improvement, it does not appear to warrant the effort. Therefore, our UATA-TDM scheme is reasonable.

It may or may not be possible to achieve the lower bound in expression 3.4 using the staggered scheme. The t-ports and the r-ports must be ordered in very specific ways. The strategy required to achieve the upper bound is interesting although it is tangential to the main topic of the thesis. We have included a discussion on a related problem in appendix B for the interested readers.

3.5 Efficiency of the UATA design

This section shows that the UATA design is efficient. Since LR's are the only backbone components, other than the AON ports, which act as interfaces, the network is efficient if all the LR's are used to their capacity.

Each LR has capacity F^2R. The maximum throughput allowed in the UATA network is F^2R times the number of LR's. We will show that the network is efficient if $\frac{N_t}{FM}$ and $\frac{N_r}{FM}$ are integers and the port to port traffic $tr(i,j) = \frac{R}{M^2}$.

If $T = M^2 = 1$, i.e., no TDM, then the number of t-groups equals N_t/F. For general $T = M^2 > 1$, the number of t-groups equals $\frac{N_t}{FM}$. Similarly the number of r-groups equals $\frac{N_r}{FM}$. One LR is used for each pairing of an r-group with a t-group. Therefore, the network uses $\frac{N_t N_r}{F^2 M^2}$ LR's. The maximum throughput supported by the network is $\frac{N_t N_r}{F^2 M^2} F^2 R = N_t N_r \frac{R}{M^2}$. This equals the total traffic rate for the UATA traffic with N_t t-ports, N_r r-ports, and traffic rate of $\frac{R}{M^2}$ per connection. Therefore, all the LR's are filled to capacity, and the network is efficient.

Note also that since the LR's are filled to capacity, the input and output fibers for the LR's are also filled to capacity. If the TDM time sharing is done by groups of ports that are physically close to each other, then most of the long distance fibers are filled to capacity, and no bandwidth is wasted.

In the case in which $\frac{N_t}{FM}$ and $\frac{N_r}{FM}$ are not integers, then some inefficiency develops, though very little. In this case, the maximum throughput supported by the backbone is $\left\lceil \frac{N_t}{F} \right\rceil \left\lceil \frac{N_r}{F} \right\rceil F^2 \frac{R}{M^2}$, and the total traffic rate is $N_t N_r \frac{R}{M^2}$. The difference between the two is $\left(\left\lceil \frac{N_t}{F} \right\rceil \left\lceil \frac{N_r}{F} \right\rceil - \frac{N_t}{F}\frac{N_r}{F} \right) F^2 \frac{R}{M^2}$. The percentage difference is small if N/F is large. For example, if $N_t = N_r \approx 100F$, then the inefficiency is about two percent.

Chapter 4

Congestion Probability of the UATA Backbone

The UATA backbone design supports the UATA traffic exactly. Unfortunately, UATA traffic often does not accurately represent the actual port to port traffic. However, the UATA backbone design can still be useful if the port to port traffic approximates UATA. In this chapter, we define a more realistic traffic model. The resulting port to port traffic approximates the UATA traffic with small variations.

If the UATA backbone is applied to this new traffic, then it is possible that the actual aggregated traffic on a bit-pipe exceeds the amount provided by the backbone. A bit-pipe is congested if it is full and cannot accommodate any new sessions. The probability of congestion is analyzed in this chapter.

There are many parameters in designing a network. These parameters can be varied in order to achieve a given probability of congestion. Naturally, one can trade off between inter-dependent parameters.

If, in addition, cost functions are assigned to both the local network and the optical backbone, then we can calculate the parameter values which will minimize the cost and satisfy the congestion criterion. For example, given a set of end users and their statistics, one can solve for a minimum cost network that satisfy a congestion probability constraint. Often, when designing a hierarchical network like the ones proposed here, it is difficult to decide the amount of aggregation required, i.e., the optimal size of the local network before they are connected together by the backbone layer. This chapter explores this problem.

4.1 The backbone model

The UATA backbone connects every t-port to every r-port with a bit-pipe. All the bit-pipes have the same capacity, and one can design the network so the capacity of a bit-pipe is an integer multiple of a wavelength (using LR of coarseness $k \geq 1$) or a fraction of one wavelength (using TDM). However, once the backbone is built, the bit-pipe capacity is then fixed. Note that the backbone is totally defined if the coarseness, k, and the number of ports, N, are known.

If we allow k to take on fractional values ≤ 1, then we can use k as a parameter indicating the size of the bit-pipe. The capacity of a bit-pipe equals kR, where R is the capacity of one wavelength. The general backbone can be denoted by UATA-k. If $k > 1$, then the backbone uses LR's of coarseness k. IF $k < 1$, then the backbone uses TDM with $1/k$ time-slots. In either case, the capacity of a bit-pipe equals kR.

A backbone using TDM is considerably different than a backbone without TDM. Most notably, the added TDM functionality requires a more complex backbone. For example, TDM requires synchronization of transmitters and receivers so they can transmit and receive at the appropriate time. Furthermore, the lasers and the receivers are not used fully since they only operate on the assigned time slots. Because of differences like those mentioned, the cost function for the two types of backbone will be different. When optimizing the cost, the cost function, C_t, will be split into two regimes: for $k \geq 1$ (no TDM), and for $0 < k < 1$ (TDM).

In either case, the backbone provides a bit-pipe between a t-port and an r-port. In general, the bit-pipes can be utilized in any fashion by the connecting local networks. However, for simplicity, we assume that local networks split the bit-pipes into many small equal sized circuits. Each circuit supports one end to end session at the rate of r bits per second. Since the bit-pipes have fixed capacity, each bit-pipe can only handle a fixed number of end to end sessions. Let c_s be the maximum number of sessions allowed in a bit-pipe. Then, $c_s = k \lceil R/r \rceil$. Usually, $r << R$. For simplicity, assume that R/r is an integer so the ceiling function can be ignored. One can view c_s as the capacity of a bit-pipe in number of end user sessions. In this chapter, we explore how the network performs with changing k.

Since each bit-pipe allows a maximum of c_s simultaneous sessions, the transmitting local network must restrict the number of sessions entering the bit-pipe. There is no blocking at the receiving r-port because the backbone does not send more traffic than can be handled by the r-port. The receiving end user may be busy, but we ignore this because we are interested in blocking due to backbone congestion.

The general UATA backbone is shown in figure 4-1. In this figure, ports, local networks, and end users are split into their transmitting and receiving parts. This is done for clarity. In practice, the transmitting and receiving components often occupy the same physical space.

Let E be the total number of end users in the overall network and assume uniformity in the network. Then the number of end users attached to a port (via the local network) is E/N, where N is the number of ports in the backbone. (We also assume symmetry, i.e., the number t-ports equals the number of r-ports). We are interested in the type of problems where E is given, and N, the size of the local network, is a variable to be optimized.

We will focus on the probability of congestion on a given bit-pipe. Bit-pipe congestion probability is the probability that, at any given instant of time, the maximum number of sessions, c_s, are active on the bit-pipe. Denote this probability by p_c. If a session request arrives on a full bit-pipe, it will be rejected. Note that the backbone has no queues. Hence, a request is either accepted or rejected by the backbone. Bit-pipe congestion is more relevant to end users than overall backbone blocking probability because a user is only affected by the traffic on the bit-pipe being used and there is one and only one bit pipe assigned to each origin and destination port pair.

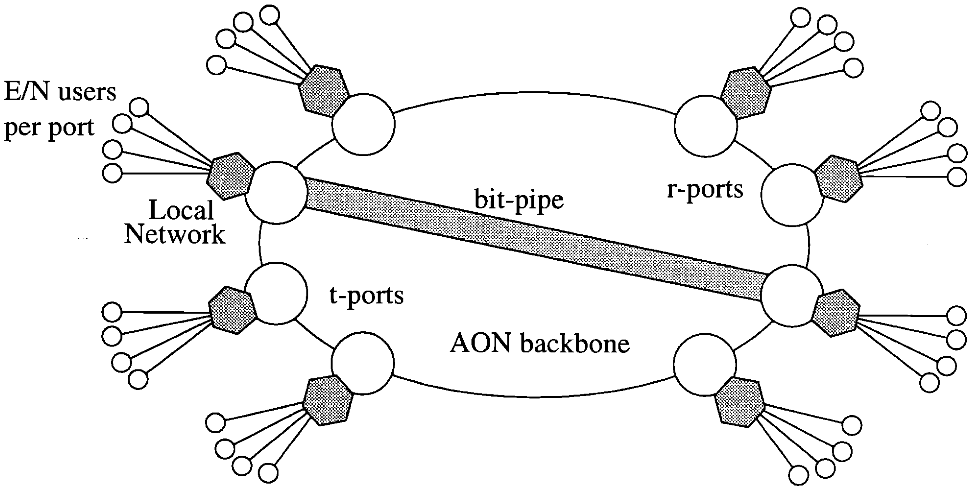


Figure 4-1: The general UATA backbone. There are N r-ports and N t-ports. Each port is connected to E/N users via the local network. A bit-pipe of capacity kR connects each t-port r-port pair.

4.2 The traffic model and probability of congestion

In this section, we define the traffic model and calculate the congestion probability. Since we are concerned with the bit-pipe congestion probability, the relevant statistics are the probabilistic descriptions of the aggregated traffic on a bit-pipe.

We consider two different models: the Bernoulli model and the Poisson model. In the Bernoulli model, we take an instant of time, and model the probability that transmitting end users are active as independent Bernoulli trials. In the Poisson model, we assume that the aggregated requests arrive on a bit-pipe as a Poisson process. These two models impose slightly different assumptions on the traffic. However, the differences are not critical. In fact, they are negligible in the limit of high aggregation and small congestion probability. As it turns out, both models give rise to the same bound for the probability of congestion.

Both models assumes uniformity. The end users statistics, i.e, the arrival rate and destination, are independent and identically distributed.

4.2.1 The Bernoulli model

Consider a transmitting end user. This end user can be in one of three states, the *transmitting state*, the *waiting state*, or the *idle state*. When the end user has data to transmit, it requests a connection to the corresponding bit-pipe. If the bit-pipe is not congested, then the end user immediately goes into the transmitting state and starts the transmission. If the bit-pipe is congested, then the end user goes into the waiting state and waits until either the bit-pipe is no longer congested, or times out and gives up on the connection. An end user returns to the idle state when it times out from a waiting state, or finishes transmitting in the transmitting state. We call the end user *engaged* if it is not idle, i.e, it is in either the transmitting or the waiting state. If the end user is not engaged, then it is idle. Figure 4-2 shows how an end user typically transitions between the different states.

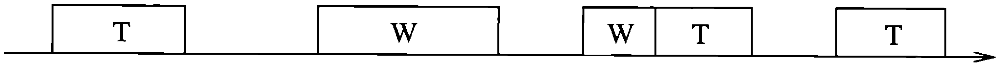


Figure 4-2: Example of an end user activity. The letter 'T' denotes the transmitting state, and 'W' denotes the waiting state. Otherwise, the user is idle.

The engaged period is the contiguous block of time when the end user is engaged, starting at the instant that the end user changes from the idle state to the engaged state, and ending at the first instant thereafter that the user changes back to the idle state. Allow the system to run for a long time, say t seconds, and assume that the system will eventually reach statistical equilibrium. Then, the number of engaged periods divided by the elapsed time approaches a limit as t approaches infinity. This value is the time average number of engaged periods per second. Call this the arrival rate per end user and denote it by l.

If we take the total length of all the engaged periods within time t, and divide by the total number of engaged periods, then we have the time average length of an engaged period. Again, in the limit of $t \to \infty$, this average approaches a limit. Define $1/\mu$ as the average length, in number of seconds, of an engaged period. (It will be clear later, when considering the Poisson model, why we use $1/\mu$ here.)

In the Bernoulli model, we assume that the system is ergodic, so these time averages equal statistical averages. Therefore, the expected number of engaged periods per unit time is l, and the expected length of an engaged period is $1/\mu$. We observe the system at a snapshot of time. In that snapshot, a transmitting end user is engaged with probability

$$\phi = \frac{l}{\mu}. \tag{4.1}$$

This can be viewed as an application of Little's Law where ϕ is the expectation of an end user being in the system, $\frac{1}{\mu}$ is the mean time spent in the system, and l is the arrival rate.

An engaged transmitting end user chooses among the N possible r-port destinations with equal probability. Hence, if we observe a particular bit-pipe, each connecting transmitting end user has probability ϕ/N of being engaged on that bit-pipe. The transmitting end users are independent of each other. Therefore, they can be considered as independent Bernoulli trials, i.e., they are either engaged or idle, and the probability of being engaged is ϕ/N. Clearly, the bit-pipe is congested if more than c_s transmitters are engaged on that bit-pipe at any instant of time, where c_s is the maximum number of sessions allowed on a bit-pipe.

There are E/N end users connected to each t-port. Therefore, at a snapshot of time, the aggregated traffic on a bit pipe looks like the result of E/N independent Bernoulli trials. The probability of engagement is ϕ/N, and the probability of congestion is equal to the probability that there are c_s or more engagements in these E/N independent Bernoulli trials.

Let $p(i)$ be the probability that there are exactly i engagements. Then,

$$p(i) = \binom{\frac{E}{N}}{i} \left(\frac{\phi}{N}\right)^{i} (1 - \left(\frac{\phi}{N}\right))^{\frac{E}{N}-i}. \tag{4.2}$$

This is the binomial distribution.

The probability of congestion, p_c, is the probability that $i \geq c_s$, that is, the sum of the tail of the binomial distribution, i.e.,

$$p_c = \sum_{i=c_s}^{E/N} p(i). \tag{4.3}$$

The tail of the binomial can be bounded using the Chernoff bound. Using the result from [Bar 93, pg. 166], we have,

$$\ln p_c \leq c_s \ln \frac{\phi E}{N^2 c_s} + (c_s - \frac{\phi E}{N^2}). \tag{4.4}$$

The above equation is valid for $c_s \geq \frac{\phi E}{N^2}$, we will see that this is always the desired case.

Define, the *utilization*, γ, as the expected fraction of the bit-pipe being used. Hence, γ is the ratio of the average number of successes to the maximum number of sessions allowed. The average number of successes is the probability of success of each trial, which is ϕ/N, multiplied by the number of trials, which is E/N. Therefore,

$$\gamma = \frac{E\phi}{N^2 c_s} = \frac{El}{\mu N^2 c_s}. \tag{4.5}$$

Generally, small p_c is desired, and small p_c is achieved when $\gamma < 1$. Therefore, we always want to operate in the regime where $c_s \geq \frac{\phi E}{N^2}$

Rewriting our bound for $\ln p_c$, we have,

$$\ln p_c \leq c_s \left[\ln \gamma + (1 - \gamma)\right]. \tag{4.6}$$

Or,

$$p_c \leq \left(\gamma e^{(1-\gamma)}\right)^{c_s}. \tag{4.7}$$

4.2.2 The Poisson model

In the Poisson model we look at all session requests onto a bit-pipe as a whole, and make two simplifying assumptions. First, the aggregate session arrivals form a Poisson process. Second, the aggregated session arrival rate is λ, which is independent of the number of sessions that are active on the bit-pipe. The assumptions are reasonable if the arrivals to each bit-pipe are the result of aggregation of a large number of end users, and if the number of end users aggregated is much larger than the average number of sessions on a bit-pipe.

Since λ is the aggregated request arrival rate, λ changes as the amount of aggregation changes. To reflect this dependence, we write λ in terms of the individual end user statistics. From the previous section, l is the average arrival rate per end user. Since there are E/N transmitting end users connected to each t-port, the aggregate session request arrival rate on a t-port is $\frac{lE}{N}$. Each of these requests chooses among the N possible destinations with equal probability. Therefore, the aggregate session request rate on a given bit pipe can be written as:

$$\lambda = \frac{lE}{N^2}. \tag{4.8}$$

We view the bit-pipe as a system of c_s servers. The bit-pipe always satisfies a request if there are empty servers. If all c_s servers are busy, then a request arrival will be denied. We assume that denied requests are not re-tried, i.e., the bit-pipe is a loss system. Note that this is a different assumption then the one made in the Bernoulli model. However, this is not an additional assumption since we've already made the assumption that λ is independent of the system state (the number of sessions active on the bit-pipe). If we were to include the continuous retries of a waiting transmitting end user, then we should also include the effect that an end user cannot request a connection while it is already transmitting.

Assume each of the c_s servers serves at the rate of μ sessions per second with arbitrary service distribution. Therefore, the expected length of each session is $1/\mu$ seconds. This parallels the Bernoulli model where $1/\mu$ the expected length of an engaged period. Since each session is operating at r bits per second, $S = r/\mu$ is the expected session size.

To recapitulate, we have modeled the bit-pipe as a c_s-server loss system. The bit-pipe simultaneously serves up to c_s sessions at rate $\mu = r/S$. When all c_s servers are busy, the system denies and drops any further requests until a server is no longer busy. The arrival process to the system is Poisson with rate λ. The session duration is arbitrarily distributed with an average duration of $1/\mu$.

This model assumes large aggregations and uniform traffic. Uniform traffic implies that all end users have the same statistics. Large aggregation implies that E/N is much greater than c_s.

This system is normally denoted as an $M/G/c_s/c_s$, c_s-server loss system: with M indicating memoryless Poisson arrivals, G denoting general distribution for the service time, the first c_s representing the maximum number of simultaneous sessions that can be served, and the last c_s indicating that arrivals with c_s busy servers will be dropped.

In this model, the utilization is the average fraction of the bit-pipe being used when the probability of congestion is negligible. If all requests were admitted, then the average number of sessions in the system is λ/μ. The maximum number of sessions allowed on the bit-pipe is c_s. Therefore, the utilization is:

$$\gamma = \frac{\lambda}{\mu c_s} = \frac{lE}{N^2 \mu c_s}. \tag{4.9}$$

Note that equation 4.9 is consistent with equation 4.5.

We are now ready to calculate the probability of congestion using the Poisson model. The probability of congestion is the probability that c_s sessions are active in the $M/G/c_s/c_s$ system. This probability is given by the *Erlang loss formula*, (See, [Ros 83, pg. 170]):

$$p_c = \frac{\frac{1}{c_s!}\left(\frac{\lambda}{\mu}\right)^{c_s}}{\sum_{i=0}^{c_s} \frac{1}{i!}\left(\frac{\lambda}{\mu}\right)^{i}}. \tag{4.10}$$

From equation 4.9, $\frac{\lambda}{\mu} = \gamma c_s$. Substituting this into equation 4.10, we have,

$$p_c = \frac{\frac{1}{c_s!}\left(\gamma c_s\right)^{c_s}}{\sum_{i=0}^{c_s} \frac{1}{i!}\left(\gamma c_s\right)^{i}}. \tag{4.11}$$

The denominator almost equals $e^{\gamma c_s}$. Therefore, we can re-write p_c as,

$$p_c = \frac{\frac{1}{c_s!}\left(\gamma c_s\right)^{c_s}}{e^{\gamma c_s}}\left[\frac{1}{\sum_{i=0}^{c_s}\frac{1}{i!}\left(\gamma c_s\right)^{i}e^{-\gamma c_s}}\right]. \tag{4.12}$$

The denominator in the bracketed term is simply the partial sum of a Poisson PMF. The upper limit of the sum is c_s, and the mean of the Poisson PMF is γc_s. Therefore, the bracketed term is increasing in γ, and thus, maximizes over $\gamma \leq 1$ by $\gamma = 1$. Another way to see this is by taking the derivative of the bracketed term with respect to γ and show that the derivative is always positive. Hence, we can upper bound p_c by:

$$p_c \leq \frac{\frac{1}{c_s!}\left(\gamma c_s\right)^{c_s}}{e^{\gamma c_s}}\left[\frac{e^{c_s}}{\sum_{i=0}^{c_s}\frac{1}{i!}{c_s}^{i}}\right]. \tag{4.13}$$

This can be further bounded by taking only one term of the summation in the denominator. The largest term in the sum is when $i = c_s$. Ignoring other terms, we have,

$$p_c \leq \frac{\frac{1}{c_s!}\left(\gamma c_s\right)^{c_s}}{e^{\gamma c_s}}\left[\frac{e^{c_s}}{\frac{1}{c_s!}c_s^{c_s}}\right]. \tag{4.14}$$

This reduces to,

$$p_c \leq \left(\gamma e^{(1-\gamma)}\right)^{c_s}. \tag{4.15}$$

This is the same bound as equation 4.7.

We can also lower bound p_c by letting the bracketed term in equation 4.12 equal one. In this case,

$$\frac{\frac{1}{c_s!}\left(\gamma c_s\right)^{c_s}}{e^{\gamma c_s}} \leq p_c. \tag{4.16}$$

The famous Sterling formula is:

$$c_s! \leq \sqrt{2\pi c_s}\left(\frac{c_s}{e}\right)e^{\frac{1}{12c_s}}. \tag{4.17}$$

Therefore, we have,

$$\frac{e^{\frac{-1}{12c_s}}}{\sqrt{2\pi c_s}}\left(\gamma e^{(1-\gamma)}\right)^{c_s} \leq p_c. \tag{4.18}$$

This differs from the upper bound by a factor of $\frac{1}{\sqrt{c_s}}$. Therefore, our upper bound is reasonably tight.

4.2.3 List of variables and relationships

For ease of reference in what follows, here is a list of variables defined in this chapter:

- UATA backbone parameters:

  k bit-pipe size in number of wavelengths. If $k > 1$, then the backbone uses LR's of coarseness k. $k < 1$, then backbone uses TDM with $1/k$ time slots.

  E total number of end users.

  N number of ports.

  $\frac{E}{N}$ number of end users per port.

  R bit-rate of one wavelength.

  r bit rate of each session.

  S expected session size in bits.

  kR capacity of a bit-pipe.

  $c_s = \frac{kR}{r}$ maximum number of sessions allowed on a bit-pipe.

- Bernoulli model:

  l arrival rate per end user.

  $\frac{1}{\mu} = \frac{S}{r}$ average engaged period.

  $\phi = \frac{l}{\mu}$ end user probability of engagement.

  $\frac{\phi}{N}$ end user probability of engagement on a given bit-pipe.

  $\gamma = \frac{\phi E}{N^2 c_s} = \frac{lES}{N^2 kR}$

- Poisson model:

  $\lambda = \frac{lE}{N^2}$ aggregate Poisson arrival rate to a bit-pipe.

  $\gamma = \frac{\lambda}{\mu c_s}$ bit-pipe utilization.

- Probability of Congestion:

$$p_c \leq \left(\gamma e^{(1-\gamma)}\right)^{c_s} \tag{4.19}$$

4.3 Trade off analysis

With the upper bound derived for the probability of congestion, a network architect can design a network to achieve any value of p_c. This is done by choosing appropriate values for the different network parameters. In this section, we discuss how some of these inter-dependent parameters can be traded off. We also give an example of how the p_c upper bound can be used to calculate the required network parameters.

4.3.1 Some Trade off examples

Observe that $\gamma e^{(1-\gamma)}$ monotonically increases with γ for $\gamma \leq 0 \leq 1$. Therefore, if c_s is constant, the probability of congestion decreases with decreasing γ. Also, $\gamma e^{(1-\gamma)} \leq 1$ for all $\gamma < 1$. Hence, for any given $\gamma < 1$ one can always achieve the desired probability of congestion with sufficiently large c_s. So our goal is to either decrease γ, or increase the exponent in the p_c bound, or both.

Rewriting the p_c bound, we have,

$$p_c \leq \left(\frac{lES}{N^2kR}\, e^{(1-\frac{lES}{N^2kR})}\right)^{\frac{kR}{r}} . \tag{4.20}$$

Increasing k and keeping l, E, S, N, R and r fixed will keep all aspects of the network fixed except for the port to port bandwidth, kR, which is increased. Naturally, this operation results in decreased p_c, as can be verified from the above equation, (i.e., the exponent is increased and γ is decreased). Note that this examples shows how p_c can be decreased at the expense of a larger backbone, and hence, less efficient use of the backbone resources, (i.e., smaller γ).

Increasing N and keeping l, E, S, k, R and r fixed will also decrease p_c. This operation increases the total number of bit-pipes in the backbone while keeping the bit-pipe size constant. Again, we are using a larger backbone, and therefore, the backbone is used less efficiently (smaller γ).

If backbone resources are expensive, then it may be undesirable to increase the total capacity of the backbone. Nevertheless, p_c can still be decreased without changing the backbone capacity. One way to do this is by decreasing r and keeping l, E, S, N, k, and R fixed. We have not changed the number of bit-pipes nor the capacity per bit-pipe, and therefore, the backbone capacity has not changed. Here, p_c is decreased solely by the increase in c_s, which is a result of decreased r. In other words, although each bit-pipe has the same capacity as before, it is serving more users simultaneously. This is at the expense of slower service for each end user. A more finely divided bit-pipe is more flexible, and hence, produces smaller p_c.

Another way to keep the backbone capacity constant while decreasing p_c is by increasing the amount of aggregation. Here, we fix γ and the total offered load, ϕE, and decrease N. (Note that in this case, N can be decreased only if c_s increase.) Because

$$p_c \leq \left(\gamma e^{(1-\gamma)}\right)^{\frac{\phi E}{N^2 \gamma}}, \tag{4.21}$$

p_c is decreased with fixed γ and decreasing N. The consequence of larger aggregation is larger and more expensive local networks.

Observations about scalability can be made by varying E against other parameters. For example, assume that we want to keep both γ and c_s constant, and hence p_c constant. Since $\gamma c_s = \frac{\phi E}{N^2}$, p_c can be constant if E increases with N^2. In other words, a linear increase in the number of ports will allow a quadratic increase in number of users supported by the network. This is not surprising because the same conclusion was made during the design of the UATA backbone.

In summary, we have shown that p_c can be decreased using the following methods: 1) increase the backbone capacity by increasing the size of the bit-pipes, or adding more bit-pipes of the same size, or both. This leads to a less efficient use of the backbone. 2) increase the amount of aggregation by decreasing the service rate per end user session. This causes longer service times for the end users. 3) increase the amount of aggregation while keeping the backbone capacity constant. This requires larger and more complex local networks. We have also shown that for the same p_c, linear increase in the number of ports can support a quadratic increase in the number of end users.

Next, we give an example of how to use the p_c bound to calculate required network parameters. In this example, we keep everything fixed except for the number of ports, and the capacity per port.

4.3.2 Example: The optimal N and the minimum required port capacity

Suppose we are given the problem of designing a backbone so that p_c is small. Further suppose that the end user statistics, E and ϕ, are given, and r, the bit rate per session, is also given. Of course, we can achieve any value of p_c by simply choosing larger and larger backbone capacity. Therefore, the backbone capacity needs to be somehow constrained. We can certainly keep the backbone capacity fixed, but a more interesting constraint is to keep the port capacity fixed. As will be shown later, this constraint leads to some interesting results.

Let P be the capacity per port in number of sessions. Then, $P = c_s N$ because there are c_s sessions from port to port, and each t-port is connected to N r-ports. Also, the utilization, γ equals $\frac{\phi E}{N^2 c_s}$ which equals $\frac{\phi E}{PN}$. Hence, the probability of congestion bound can be written as:

$$p_c \leq \left(\frac{\phi E}{PN} \exp\left(1 - \frac{\phi E}{PN}\right)\right)^{\frac{P}{N}}, \tag{4.22}$$

with $\frac{\phi E}{PN} < 1$. Our problem is: with fixed ϕ, E, and P, find the optimal N that will produce the smallest p_c. Although N represents the number of ports, it also represents the amount of aggregation because E/N is the number of end users per port. Large N indicates low aggregation, small N indicates high aggregation.

At first glance, one is tempted to conclude that larger N will always give us better performance. After all, the total capacity of the backbone is NP, which increases linearly with increasing N. However, larger N implies smaller aggregation, and the network loses its multiplexing advantage. We now find the optimal N for fixed P.

Taking the natural log of equation 4.22, we have:

$$\ln(p_c) \leq \frac{P}{N}\left[\left(\ln\left(\frac{\phi E}{NP}\right)\right) + \left(1 - \frac{\phi E}{NP}\right)\right], \tag{4.23}$$

where $N > \frac{\phi E}{N}$. Standard methods of finding the minimum can be applied here, (i.e., take the first derivative with respect to N, set it to zero, and check the second derivatives as well as the boundaries). Omitting the algebra, we find that the minimum is achieved when

$$\left(\ln\left(\frac{\phi E}{NP}\right)\right) + 2\left(1 - \frac{\phi E}{NP}\right) = 0. \tag{4.24}$$

Solving this numerically, we find that p_c is minimized when $\frac{\phi E}{NP} = 0.2032$. This means the optimal N is:

$$N = \frac{1}{0.2032}\frac{\phi E}{P}. \tag{4.25}$$

There is an interesting observation here. Remember that $\gamma = \frac{\phi E}{NP}$ which is the utility. The optimal N occurs when $\gamma = 0.2032$, which is independent of P. Therefore, regardless of the capacity of the ports, the minimum upper bound for p_c is achieved when the utilization is about 20%.

This does not mean that one would design the backbone such that the utility is 20%. Although the optimal γ is independent of P, the actual minimum p_c upper bound achieved depends on P. This is better explained with a numerical example.

Figure 4-3, plots the p_c bound versus N for various values of P. In this example, $\phi E = 10,000$. For each given P, there is an optimal N where the p_c bound is minimized. This N is such that the utility, $\gamma = \frac{a}{N}$, is 0.2032. Note that the actual minimum achieved decreases as P increases.

Often, the optimal N is not required because the probability of congestion is already very small. Therefore, one can trade off larger port capacity for smaller number of ports (or equivalently, higher backbone utilization).

For any given P, there is a minimum value that can be achieved by the p_c upper

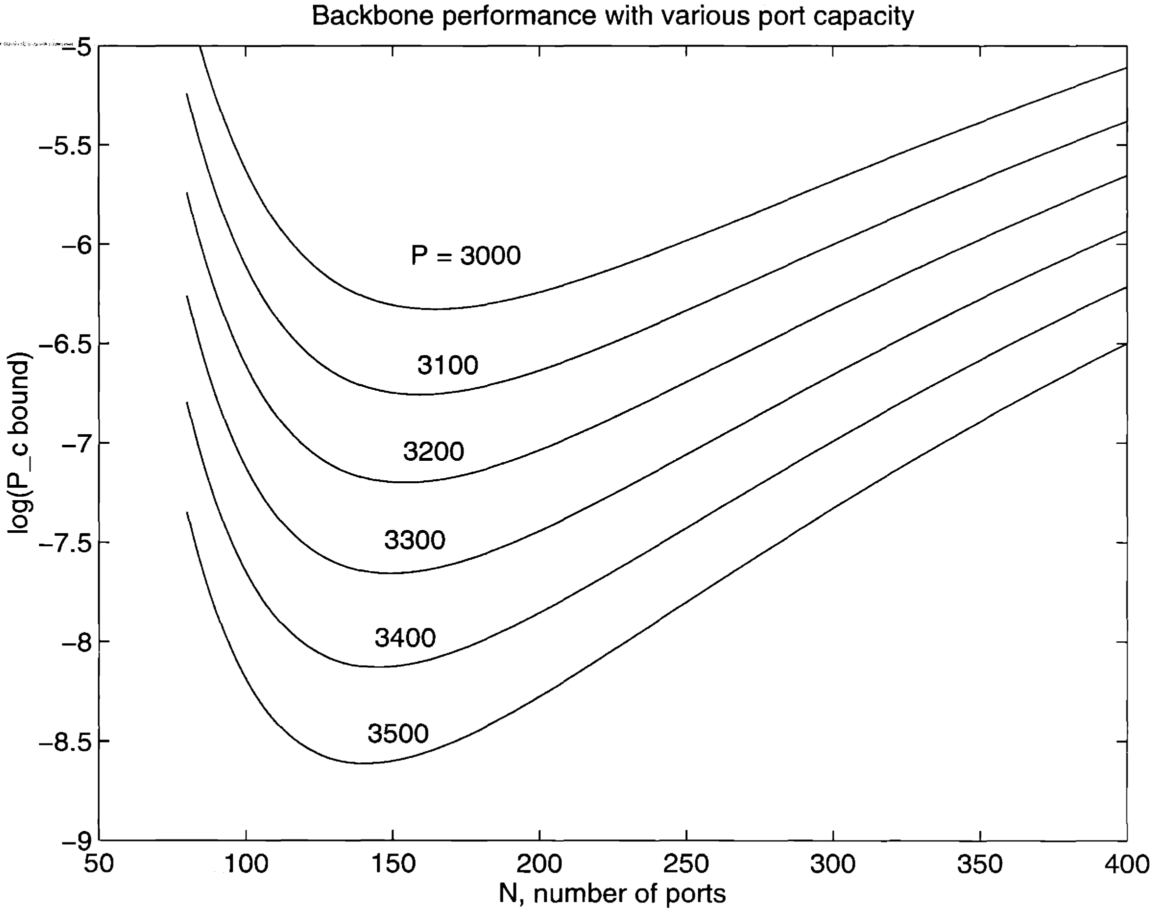


Figure 4-3: Plot of p_c bound vs. N for given port capacity.

bound. Next, we calculate the minimum port capacity required to achieve a certain value for the p_c upper bound.

Substituting the optimal N into equation 4.22, we have:

$$p_c \leq (0.4508)^{\frac{0.2032P^2}{\phi E}}. \tag{4.26}$$

This equation represents the minimum achievable upper bound for p_c with a given P.

We can use this to calculate the minimum port capacity required in order to attain a given value for the p_c upper bound. For example, assume that we desire to design a UATA network with guaranteed probability of congestion below ϵ. This requires

$$p_c \leq (0.4508)^{\frac{0.2032P^2}{\phi E}} \leq \epsilon, \tag{4.27}$$

Which simplifies to:

$$P \geq \sqrt{-6.177 \ln(\epsilon)\ \phi E} \tag{4.28}$$

This is the minimum port capacity required, in number of sessions, in order for the probability of congestion bound to be less than ϵ.

For a numerical example, let $\phi E = 10,000$ and $\epsilon = 10^{-8}$. Then we find that the minimum P required is 3373, which is achieved with $N = 146$.

If we allow P to be larger than the minimum required, then we can use smaller values of N to achieve the same p_c bound. This allows higher utilization to be achieved. The following table gives some example values for $\phi E = 10,000$ and $\epsilon = 10^{-8}$.

γ	P	N
0.2032	3,373	146
0.5000	4,367	46
0.7000	6,814	21

The higher utilization is gained at the expense of larger port capacity and larger local networks. Of course, this is beneficial if large utilization is important. If, on the other hand, utilization is unimportant and smaller local networks are desired, one can also trade larger P for larger N. Trade offs like this can be more accurately described if we assign cost to the various components of the network. We will do this in section 4.4.

4.3.3 Advantage of TDM with flexible time slot assignment

Section 3.4.4 mentioned various advantages for employing TDM in the UATA backbone. One of the advantages is that if the time slot assignments are flexible, then there will be opportunities for ports connecting to the same MASC to share their common time-slots. We now can quantify this advantage in term of the probability of congestion.

In the UATA backbone with TDM, the basic channel is one time slot with rate kR, where R is the rate of one wavelength channel, and $1/k$ is the number of time slots. If the time slot assignment is rigid, then each time slot can be viewed as a bit-pipe connecting two ports. The maximum number of sessions supported in one basic channel is c_s which is equal to $\frac{kR}{r}$. The probability of congestion can be calculated if $\frac{\lambda}{\mu}$ is known.

If the time slot assignment is flexible, i.e., unused time slots can be used by other sessions traversing the same route, then the probability of congestion must decrease due to the added flexibility. To confirm this, note that the MASC's act as another level of aggregation and the probability of congestion can be viewed from the larger wavelength-pipes connecting the MASC's. These wavelength-pipes have capacity R. Therefore, the new c_s used to calculate the p_c bound is increased by a factor of $1/k$. The utilization, γ must be the same for both rigid and flexible TDM because, γ is, assuming no blocking, the average throughput divided by the maximum throughput. Since the probability of congestion is upper bounded by $\left(\gamma\ e^{(1-\gamma)}\right)^{c_s}$, and γ is the same and < 1 in both cases, larger c_s will produce smaller probability of congestion. Therefore, flexible TDM performs better than rigid TDM. This simply reflects the statistical multiplexing advantage. Of course, it may not be worthwhile to have smaller congestion probability if the cost for the extra flexibility is high.

4.3.4 Probability of congestion and trade off conclusions

We have calculated the probability of congestion using two different traffic models. Both models give rise to the same upper bound. To achieve a given probability of congestion, parameters can be traded off. Specifically, the probability of congestion can be decreased in many ways, including: increasing backbone capacity, allowing more simultaneous though smaller sessions on a pipe, increasing aggregation, and allowing a TDM backbone to flexibly share the unused time slots.

If the end user statistics are fixed, and the port capacity is also fixed, then there is an optimal N where the p_c upper bound is minimized. However, the optimal N is not always desirable if the minimum p_c bound achieved is very small. We have calculated the minimum port capacity required to achieve a certain value for the p_c upper bound. We also have illustrated how larger port capacities can be traded for better backbone utilization.

There are many practical consideration when changing the parameters. First of all, λ, the aggregation rate, cannot be too small or our Poisson assumption will be inaccurate. On the other hand, larger λ implies a larger and more costly local network, (although there will be fewer of them in the overall network). Also, smaller utilization means that the backbone resources are wasted, and this may not be desirable if the cost of the backbone is high. In the next section, we define cost functions to reflect some of these considerations.

4.4 Minimum cost UATA networks

The probability of congestion is a reasonable network performance parameter, and the desire is to make the probability of congestion small. However, small congestion often implies a larger and more costly network. In this section, we define a cost function with respect to the parameters listed in section 4.2.3. This cost function incorporates both the local network cost and the backbone cost. The problem of interest is the minimization of cost with the constraint that p_c must be bounded by some given value.

The cost function is heavily dependent on the actual application and the available device technology. We give a plausible cost function as an example here, but emphasize that the methodology is more important than the actual result.

4.4.1 The cost function

There are two major components that contribute to the cost function: the device count in the backbone, and the switching complexity in the local network. The backbone cost is mostly from the number of devices because it is a passive network. The local network cost is mostly from the work it has to perform to do the aggregation and the deaggregation. We will not take into account the link cost which is the cost of connecting the nodes in the network. This is because the link cost depends heavily on the physical situation of the problem. Often, the problem is not where to lay the fibers; rather, the problem is the delineation between the local network and the backbone layer in an existing physically connected network.

Networks using TDM are different from networks without TDM. In TDM, the lasers and receivers are not used 100% of the time, and additional control cost is needed for the extra complexity. A separate cost function is defined for backbones using TDM.

Cost function without TDM

Consider the case where the amount of traffic between two ports are greater than or equal to one wavelength channel. In this case, $k > 1$ and TDM is not used.

First, consider the backbone device count. In the backbone, the dominant cost will be the laser transmitters, the optical receivers, and the LR's. The number of wavelengths connecting a t-port to an r-port is k. This is the coarseness of the LR used in the UATA backbone.

One transmitter/receiver is required for each wavelength used. Since the number of transmitters and receivers are the same, we will only calculate the number of transmitters needed and the resulting number will be proportional to the cost for both.

The number of transmitters in one t-port is Nk because there are N destinations and each requires k wavelengths. There are N t-ports. The total number of transmitters is:

$$\begin{aligned}\text{number of transmitters} &= N^2k \\ &= \left(\frac{\phi E r}{R}\right)\frac{1}{\gamma} \qquad (4.29)\end{aligned}$$

The second equality comes from the fact that $\gamma = \frac{\phi E}{N^2 c_s} = \frac{\phi E r}{N^2 k R}$ which is from equation 4.5. This cost function also could be used to represent the number of fibers used in the system. Within each t-port, the laser outputs are aggregated together using star couplers. The number of stars is equal to the number of lasers divided by the number of wavelengths available per fiber. Therefore, the number of fibers connecting to the ports is proportional to the number of laser transmitters.

The number of LR's required is $(Nk/F)^2$ where F is the number of wavelengths available on a fiber. This means,

$$\text{number of LR's} = \left(\frac{\phi E r}{F^2 R}\right)\frac{k}{\gamma}. \qquad (4.30)$$

We will assume that F is fixed and that LR's of different coarseness have the same cost, although this may or may not be the case in practice. A definitive method of how to weight the cost according to the coarseness does not exist since the development of LR's is still in its infant stage.

In the local network, switches are needed to set up virtual circuits connecting the end users to the ports. To calculate the amount of switching required, assume each local network is a $\frac{E}{N}$ x $\frac{E}{N}$ non-blocking network supporting permutation routing. Then the switching complexity of one local network is on the order of $\frac{E}{N}\log\left(\frac{E}{N}\right)$. (For example, an $\frac{E}{N}$ x $\frac{E}{N}$ Beneš network has roughly $4\frac{E}{N}\log\left(\frac{E}{N}\right)$ cross-points, see [Hui 90, pg. 72].) The base of the logarithm is 2. There are N local networks. So the switch complexity is,

$$\begin{aligned} \text{switching complexity} &= E \log\left(\frac{E}{N}\right) \\ &= \frac{E}{2}\log\left(\frac{ER}{\phi r}\right) + \frac{E}{2}\log(\gamma k) \end{aligned} \tag{4.31}$$

Cost function with TDM

For a backbone with TDM, the cost function is different in terms of the backbone device count and the added complexity due to TDM. Remember that a backbone with TDM is built upon a backbone without TDM. M ports are combined together using a MASC, and the MASC's are connected by a UATA network. There are $M^2 = \frac{1}{k}$ time slots in each wavelength. Each time slot is used by a pair of ports. The available bit rate from port to port is kR.

For the laser transmitter count, assume that local sharing is possible, and one laser is used for all traffic to the same destination MASC, then each t-port requires N/M transmitters. The total number of transmitters for N ports is,

$$\begin{aligned} \text{number of transmitters} &= \frac{N^2}{M} \\ &= \left(\frac{\phi Er}{R}\right)\frac{1}{\gamma\sqrt{k}} \end{aligned} \tag{4.32}$$

By symmetry the number of receivers is the same as the number of transmitters, and the cost for receivers is absorbed into the cost of the transmitters. Assume that the cost of the additional switches and buffers within the ports are negligible.

The number of LR's required is $\left(\frac{N}{FM}\right)^2$ because the backbone is a UATA network with $\frac{N}{M}$ MASC's. Therefore,

$$\begin{aligned} \text{total number of LR} &= \left(\frac{N}{FM}\right)^2 \\ &= \left(\frac{\phi Er}{F^2R}\right)\frac{1}{\gamma}. \end{aligned} \tag{4.33}$$

The added TDM capability implies added control complexity to the network. For example, the transmitters and receivers must be synchronized to the appropriate slot boundaries. If we assume that the cost is proportional to the number of ports per MASC, M, then,

$$\text{cost of added complexity} = \frac{1}{\sqrt{k}}. \tag{4.34}$$

Of course, even with TDM, the cost associated with the local network switching (equation 4.31) is still applicable.

4.4.2 Cost minimization with constraint on p_c

So far, we have calculated the cost of the network with respect to various parameters of the network. An interesting problem to be solved is as follows: Given a set of end users and its statistics, build a network that minimizes the overall cost and satisfies $p_c \leq \epsilon$ where ϵ is some given constant value.

Since the end user statistics are given, then E, the total number of end users, and ϕ the probability of engagement per user are fixed constants.

Assume that r, R, and F are constants as well. It is desirable for r to be constant because this way, the network provides a constant bit rate for each session. The bit-rate per wavelength, R, depends heavily on the technology of the Latin Router, the speed of the lasers, and the speed of the receivers. Therefore, R may not be easily changeable, so it is assumed to be constant. Finally, F, the total number of wavelength per fiber, depends on the capacity of fiber which is fixed.

With all the constant listed above, all the cost defined in the previous section can be written as a function of two variables: k and γ. The goal is to find the values for k and γ such that the cost is minimized and the criterion $p_c < \epsilon$ is met. The two variables, k and γ uniquely define the backbone architecture. For example, the number of ports N, is determined by $N = \sqrt{\frac{\phi E r}{\gamma k R}}$. Figure 4-4 shows the plot of constant N on a k versus γ plot.

The total cost without TDM can be written as:

$$\text{total cost} = w_1 \frac{1}{\gamma} + w_2 \frac{k}{\gamma} + w_3 \log(\gamma k) + w_4 \tag{4.35}$$

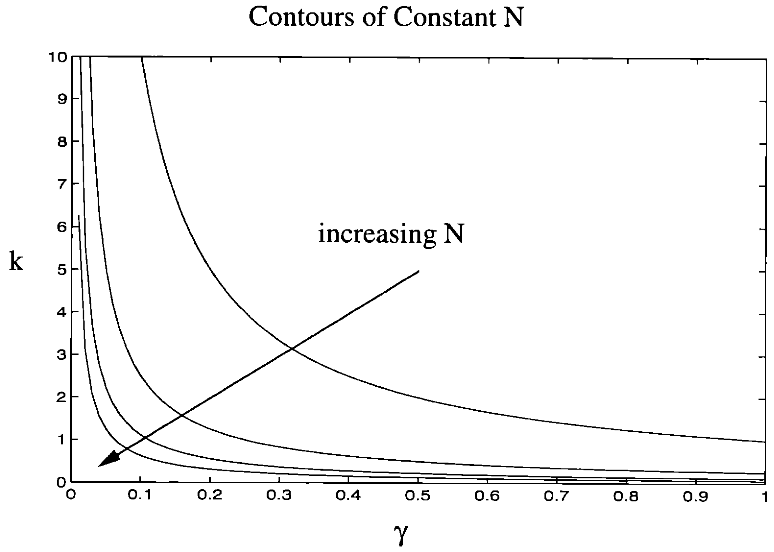


Figure 4-4: Contours of constant N on the k versus γ plot.

Where W_i are the appropriate weighting functions reflecting both the fixed constants and also the relative weight between the different costs: w_1 is associated with the transmitter/receiver cost, w_2 is associated with the LR cost, and w_3 is associated with the switch complexity cost. The constant term arises from the switch complexity cost, $\frac{E}{2}\log\left(\frac{ER}{\phi r}\right)$; we ignored this in the following because an additive constant is irrelevant when optimizing cost.

For simplification, normalize the cost by dividing the equation by w_3. Minimizing the actual cost is equivalent to minimizing the normalized cost. The normalized cost function is:

$$C_{\text{-WDM}} = W_1\frac{1}{\gamma} + W_2\frac{k}{\gamma} + \log(\gamma k). \tag{4.36}$$

Similarly, the normalized cost for the network with TDM is:

$$C_{\text{-TDM}} = W_1\frac{1}{\gamma\sqrt{k}} + W_2\frac{1}{\gamma} + \log(\gamma k) + W_3\frac{1}{\sqrt{k}} \tag{4.37}$$

Here, for consistency, we have used the same weighting constants as those for $C_{\text{-WDM}}$. The extra term in $C_{\text{-TDM}}$ is from the added control cost of TDM.

If we let $k = 1$, we see that $C_{\text{-WDM}}$ equals $C_{\text{-TDM}}$ except for a constant term. The constant term corresponds to the added complexity due to the network's potential in having TDM.

The goal is to minimize the cost. Cost function $C_{\text{-WDM}}$ is used for the WDM backbone without TDM, which is when $k \geq 1$. Otherwise, TDM is used, and the cost function is $C_{\text{-TDM}}$. Note that for $k \geq 1$, k must be an integer, and for $k \leq 1$, $M = \frac{1}{\sqrt{k}}$ must be an integer. We ignore the burden of these integer constraints in the following analysis. This way, we can better focus on the minimization.

The cost minimization is under the constraint that $p_c \leq \epsilon$. The constraint can be re-written as:

$$k \geq \frac{\frac{r}{R}\ln(\epsilon)}{\ln(\gamma) + (1-\gamma)} \tag{4.38}$$

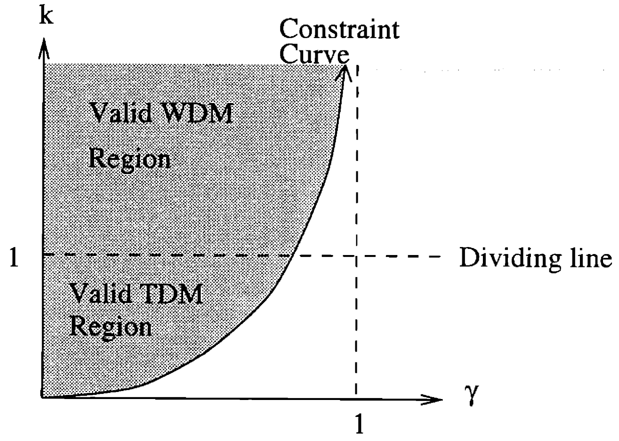


Figure 4-5: Valid values for k and γ are the regions above the constraint curve. The region above $k = 1$ is for WDM, and the region below is for TDM.

The above expression, with the inequality replaced with equality, is called the *constraint curve.* If we plot k versus γ, then the region above the constraint curve contains all valid values of k and γ (subject to the integer constraints which we are ignoring). Call this region the *valid region.* The valid region is further divided into two regions by the line $k = 1$, which we call the *dividing line.* The area above the dividing line is the *valid WDM region.* It is the valid operating region for the UATA backbone without TDM. The area below the dividing line is the *valid TDM region.* It is the valid operating region for backbones using TDM. Figure 4-5 depicts the typical valid regions.

We will focus our attention on the k versus γ plot. Therefore, unless stated otherwise, whenever the terms, points, lines, curves, or regions, are used, they refer to the respective items on the k versus γ plot.

4.4.3 Solution for the minimum cost problem

The problem of finding the minimum cost network can now be solved. This problem is a two variable cost-constraint problem with two regions. In the valid WDM region, $C_{\text{-WDM}}$ is used for the cost function. In the valid TDM region, $C_{\text{-TDM}}$ is used for the cost function. The strategy is to solve for the minimum cost for both regions, compare the two costs, and take the smaller of the two solutions. Note that each solution can be on the boundary of the appropriate region, and in fact, this is often the case.

Solution for WDM

We first state some properties of $C_{\text{-WDM}}$, and then take advantage of these properties to solve for the minimum cost network for WDM.

Note that $C_{\text{-WDM}}$ increases with k. Therefore, we always want to use the smallest k possible. This means the solution must lie on the constraint curve or on the dividing line. Furthermore, any valid point below another valid point will have a smaller value of k, and hence produce a smaller cost. Therefore, if we take any arbitrary point in the valid region, then we can obtain a smaller cost network by traversing downwards on the k versus γ plot.

If we take the partial derivative of $C_{\text{-WDM}}$ with respect to γ, and set it to zero, then we have only one solution which is,

$$\gamma = W_2 k + W_1. \tag{4.39}$$

The second derivative of $C_{\text{-WDM}}$ with respect to γ shows that $\gamma = W_2 k + W_1$ is a minimum. This is a straight line in the k versus γ plot. Call this equation *line A*. Line A represents the γ which will produce the smallest network cost for any fixed k. Also, since the partial derivative has only one solution, and the cost function is smooth in γ, one can always obtain smaller cost by taking a point closer to line A in the horizontal direction. Therefore, to minimize cost, it is desirable to traverse horizontally toward line A.

Next, let $k = v\gamma$ for some parameter v. We are looking at radial lines described by the radial lines $k = v\gamma$ for all v. Then our cost function becomes:

$$C_{\text{-WDM}} = \frac{W_1}{\gamma} + W_2 v + \log(v\gamma^2) \tag{4.40}$$

If we again take the partial derivative with respect to γ, and check to make sure we have a global minimum, we get,

$$\gamma = \frac{W_1}{2}. \tag{4.41}$$

Call this vertical line *line B*. Line B is independent of v. What this means is that on any given radial line radiating from the origin, the minimum cost is obtained by letting $\gamma = \frac{W_1}{2}$. Therefore, starting at a point with $\gamma > \frac{W_1}{2}$, one can always obtain a smaller cost network by traversing toward the origin. On the other hand, if one started at a point with $\gamma < \frac{W_1}{2}$, one would traverse radially away from the origin to obtain a smaller cost.

We have three useful methods which will provide us with lower cost network:

M1a Always traverse downwards.

M2a Always traverse horizontally toward line A.

M3a Always traverse radially, with respect to the origin, toward line B.

Note that regardless of the weights, line A always crosses the dividing line to the right of line B. (To see this, observe that line A crosses $k = 0$, at $\gamma = W_1$, while line B is a vertical line at $\gamma = W_1/2$.)

The strategy is to start at any point in the valid WDM region, and find a path that will lead us to the minimum cost network. The path is such that each segment of the path leads to a smaller cost network. The solution has three cases:

Case I: The constraint curve crosses the dividing line to the right of line A. In this case, we start at any point in the valid region, and use M3a to traverse radially toward

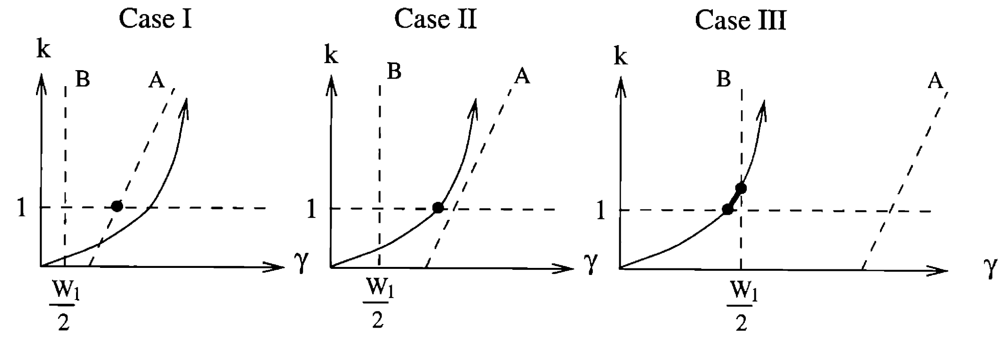


Figure 4-6: Three cases for the WDM network. In each case, the constraint curve crosses the dividing line at different points.

line B. This will bring us to either line B, or the dividing line. If at this point, we are at line B, then use property M1a to traverse downwards to the dividing line. Lastly, traverse horizontally to line A using M2a. Therefore, the cost is minimized at the intersection of line A and the dividing line. Solving for γ, we have,

$$\gamma = W_2 + W_1. \tag{4.42}$$

Typically, the constraint curve approaches $\gamma = 1$ very quickly if ϵ is small. Therefore, Case I occurs when $W_2 + W_1 < 1 - \delta$, where δ is a small value which depends on ϵ, R, and r. Therefore, if W_1 and W_2 are small, meaning that the cost of aggregation is more than the cost of the backbone, then Case I results. This makes sense because if the cost of aggregation is large, then one should use the smallest k. Similarly, because the cost of the backbone is relatively small, one can sacrifice the utilization by using a smaller γ.

Case II: The constraint curve crosses the dividing line in between line A and line B. Start again at any place in the valid WDM region. Using M3a, traverse radially toward line B. This will bring us to either line B or the dividing line. Next, traverse downwards, using M1a, until the dividing line is reached. Lastly, use M2a and traverse to the right until we meet the constraint curve. Therefore, the solution is at the intersection of the constraint curve and the dividing line. One can solve numerically for γ using the equation for the constraint curve.

This case occurs when the cost of backbone is beginning to be significant such that the backbone must have high utilization. However, aggregation cost is still large so the smallest k is used.

Case III: The constraint curve crosses the dividing line on the left of line B. In this case, the solution is not as elegant. Since traversing to the right toward line A always leads to a smaller cost network, the solution lies somewhere on the constraint curve. Also, if the constraint curve crosses line B (note that this does not have to be the case), then one can traverse radially to line B first, and then down to the constraint curve. Therefore, the solution lies somewhere on the constraint curve between the dividing line and line B. (If the constraint curve does not cross line B, then the solution is somewhere on the constraint curve above the dividing line.) To find the solution, substitute the constraint equation into the cost function. The result is a minimization problem in one variable, γ. Unfortunately, the cost function is a transcendental function which can only be solved numerically.

Case III occurs when W_1 is large. This means the cost for the backbone is large compared with the aggregation cost. Therefore, high utilization, which can be obtained with high aggregation, is desired. Our solution reflects this notion.

Solution for TDM

Now we find the solution in the TDM region. Remember that the cost function is:

$$C_{\text{-TDM}} = W_1 \frac{1}{\gamma\sqrt{k}} + W_2 \frac{1}{\gamma} + \log(\gamma k) + W_3 \frac{1}{\sqrt{k}} \tag{4.43}$$

First, take the partial derivative of $C_{\text{-TDM}}$ with respect to k, and equate it to zero. We get,

$$\sqrt{k} = \frac{W_1}{2\gamma} + \frac{W_3}{2}. \tag{4.44}$$

Call this equation *curve C.* A check on the second derivative shows that curve C represents the minimum cost network for any given γ. Therefore, traversing vertically toward curve C will always produce a smaller cost.

Next, take the partial derivative of $C_{\text{-TDM}}$ with respect to γ, and equate it to zero. We get,

$$\sqrt{k} = \frac{W_1}{\gamma - W_2}. \tag{4.45}$$

Call this equation *curve D*. Again, a check the second derivative shows that this is a minimum. Curve D represents the minimum cost points for constant c_s. Therefore, traversing horizontally toward curve D will always produce a smaller cost. Note that curve D only have real solutions for $\gamma \geq \frac{W_2 R}{r}$

Hence, we have two methods to reduce cost:

- M1b Always traverse vertically toward curve C.
- M2b Always traverse horizontally toward curve D.

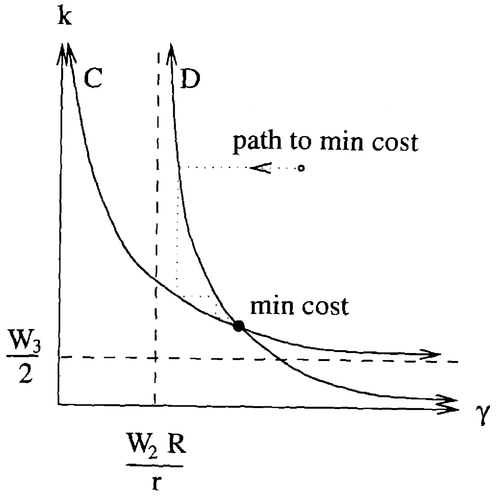


Figure 4-7: Method for finding the minimum cost of a TDM network. Always move horizontally to curve D, and vertically to curve C. If valid region is ignored, the path leads to the intersection point between curve D and curve C.

Figure 4-7 shows a typical picture of curve C and curve D. It is straight forward to verify that the two curves intersect at exactly one point. Call this point p_m. If we ignore the valid region, then the minimum cost network can be found by alternatively performing M1b and M2b movements. One such path is illustrated in the figure. All such paths done this way leads to p_m. Therefore, p_m is the global minimum (without constraint).

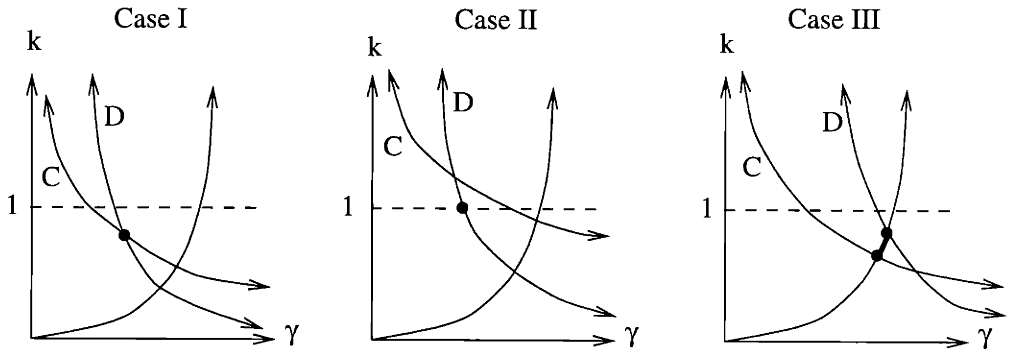


Figure 4-8: Three solution cases for the TDM network. In each case, p_m lies in a different region.

If we impose the constraint, then the answer has three cases:

Case I: The global minimum, p_m, is within the valid TDM region. In this case, the minimum cost TDM network occurs at p_m. This case occurs when W_3, the additional control cost due to TDM, is small, and W_2, the cost of the LR's is small compared with the cost of aggregation. The small TDM control cost allows the network to use TDM. The large cost of aggregation forces the network to have small k.

Case II: The global minimum, p_m, is above the dividing line. In this case, alternately performing M1b and M2b will produce a path that leads to the dividing line. This means the minimum occurs when $k = 1$. Since $C_{\text{\_WDM}} < C_{\text{\_TDM}}$ for $k = 1$, the minimum cost network does not use TDM. This case occurs when W_3 is large. Therefore, the additional control cost of TDM makes TDM prohibitively expensive.

Case III:

The global minimum, p_m is below the dividing line, and outside the valid TDM region. In this case, the path drawn by performing M1b and M2b alternately, will end up on the constraint line. Therefore, the minimum cost network occurs somewhere on the constraint line. To solve for the actual value, substitute the constraint curve into $C_{\text{\_TDM}}$. The resulting cost function has only one variable. The minimum can then be found numerically.

This case results in the highest γ in comparison to the other two TDM cases. This is because, in this case, W_2, the cost of LR's, is high. Therefore, the network must use the backbone more efficiently.

4.4.4 Minimum cost conclusions

We have formulated a reasonable cost function for the UATA network, and found the minimum cost network subject to the constraint that $p_c \leq \epsilon$. This minimum network is defined by two variables, k, the coarseness of the bit-pipes, and γ, the bit-pipe utilization. With k and γ, the rest of the network can then be determined. The solution to the minimization makes intuitive sense: Costly local network forces smaller aggregation, costly backbone forces higher utilization, and costly TDM control prohibits TDM.

Chapter 5

Overflow Shared Resources for the UATA Network

The UATA network contains a fixed capacity bit-pipe between every pair of ports. When a bit-pipe is full, end to end sessions desiring to transmit on that bit-pipe will have to wait or be dropped. In this chapter, we augment the UATA network with a shared resource. Sessions that overflow the dedicated bit-pipe in the UATA network may use this shared resource. The shared resource is essentially a smaller UATA network where each bit-pipe is being shared among different port pairs. The probability that a shared bit-pipe is also congested depends on the size and coarseness of the shared resource.

5.1 Network and Traffic model

Figure 5-1 shows the network block diagram. The lower block is a UATA network with fixed bit-pipes. Each bit-pipe is dedicated for the traffic between the two connecting ports. The capacity of a bit-pipe is the guaranteed bandwidth between each port pair. We augment the UATA network with a shared resource. This shared resource also contains bit-pipes, but each bit-pipe is shared among many different port pairs. Call these pipes the *shared resource bit-pipes (SRB's)*. If a dedicated bit-pipe is congested, then new session arrivals on the bit-pipe will overflow it. The overflow traffic may try to use the shared resource. Note that only the dedicated bit-pipe capacity is guaranteed for each port pair. The shared resource allows overflow traffic to get through, but many port pairs share this extra bandwidth. Allocation of the shared bandwidth is not guaranteed.

Since the dedicated portion of the network can be made such that the probability of congestion is small, we assume that overflow events are infrequent. Hence, the shared resource can be relatively small compared with the dedicated portion. Further, we restrict the number of fiber connections from each port to the shared resource. Let b be the number of fibers connecting each port to the shared resource. Hence, b restricts the amount of traffic flow between the shared network and one port.

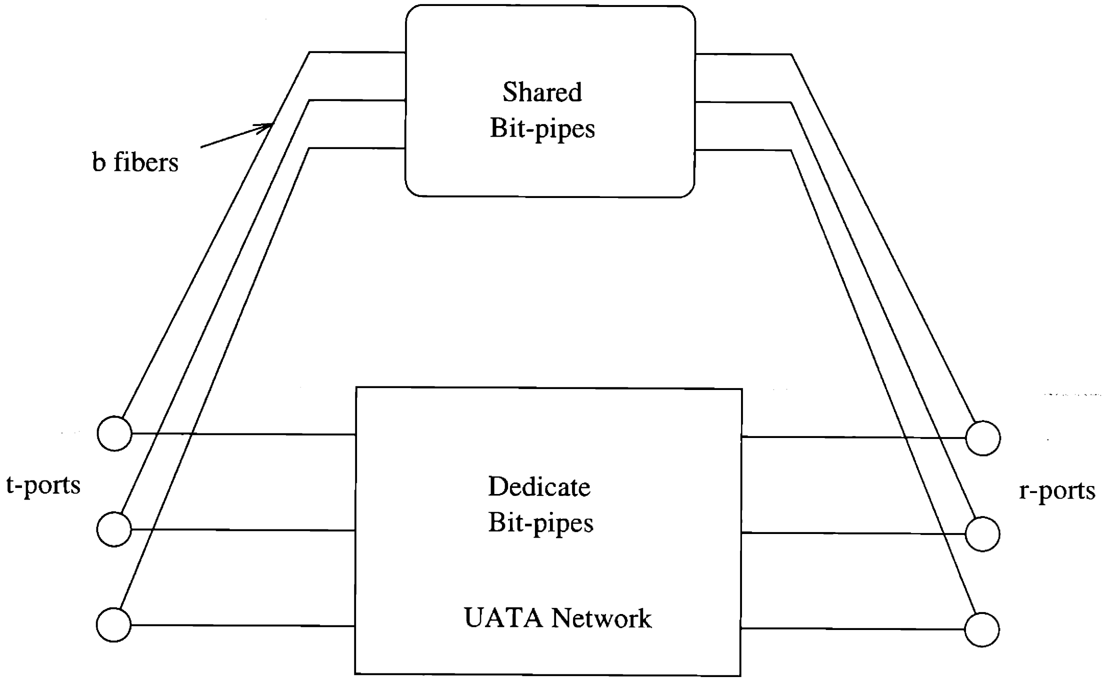


Figure 5-1: UATA network with shared resource. The UATA network is augmented by adding a shared resource. Sessiôns that are not able to use the dedicated bit-pipes may be able to use the shared resource. Each port is allowed b fiber connections to the shared resource.

Upon receiving a session request, the network first tries to accommodate the session on the dedicated portion of the network. However, if the required bit-pipe is congested, then the network will try to set up the session in a SRB. We assume the same end user statistics as in chapter 4. Specifically, we model the activity of each transmitting end user as an independent Bernoulli trial. (See section 4.2.1 for details.) The probability that a transmitting end user is engaged is ϕ. The goal is to create a shared resource such that the probability of congestion on a given SRB is small.

5.2 Shared resource structure

Now, we focus our attention on the shared resource. The shared resource contains SRB's that can be used by different port pairs whenever overflow occurs in the dedicated bit-pipes. There is no reason to favor one port pair over another. The implementation uses ideas developed for the UATA network.

The shared resource is a smaller UATA network using b^2 LR's of coarseness h, and MASC's as its inputs and outputs. Each MASC contains b star couplers. This is similar to the structure of UATA-TDM, (see figure 3-6), except here each bit-pipe connecting the MASC's contains h wavelengths. There are $\frac{bF}{h}$ combining MASC's and $\frac{bF}{h}$ broadcasting MASC's. Divide the ports evenly into $\frac{bF}{h}$ groups, and connect each group to one MASC. Therefore, there are $M = \frac{Nh}{bF}$ ports connected to each MASC. A port connects to a MASC using b fibers. Each t-port uses bF transmitters to transmit to the shared resource, and similarly each r-port uses bF receivers for overflow traffic.

Let R be the bit rate of a wavelength, and T be the number of time slots in the network. The SRB bandwidth is broken up into pieces each of size $\frac{R}{T}$. Subject to availability, a port pair may use any number of these pieces. For simplicity, we assume that $\frac{R}{T} = r$, where r is the bit rate per end user session. That way, the SRB can be shared on a per end user session basis.

The UATA network using LR's of coarseness h allows UATA connections between the MASC's. A bit-pipe of size h wavelengths is connected between every MASC pair. The objective of small SRB congestion leads one to believe that the largest h should be used. This intuition is in general incorrect. Indeed, larger h will lead to larger SRB's, which have capacity $= \frac{hR}{r}$ sessions. However, larger h will also lead to

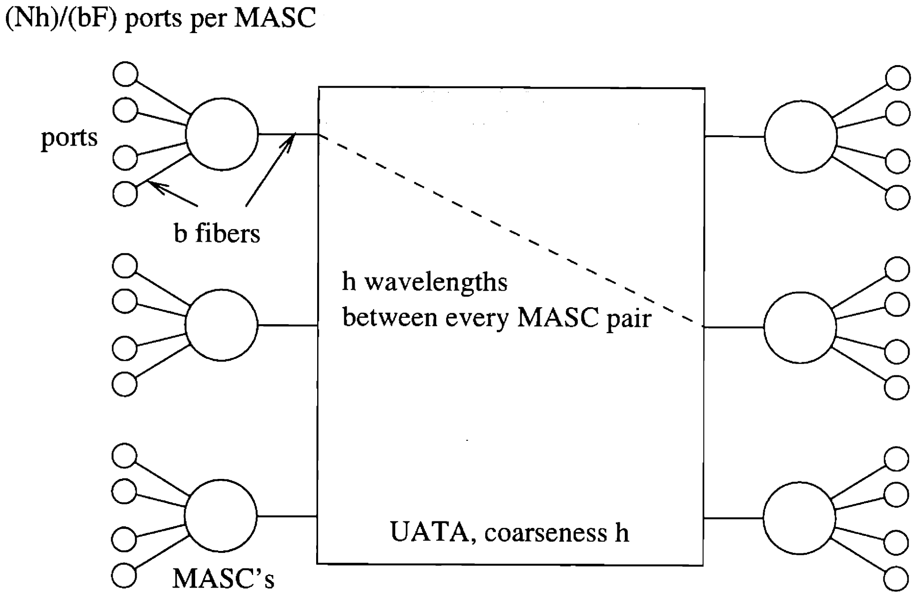


Figure 5-2: Shared resource for overflow traffic. There are $\frac{bF}{h}$ MASC's and $\frac{Nh}{bF}$ ports connected to each MASC. Each MASC pair is connected by a bit-pipe containing h wavelengths.

a larger number of ports connected to an SRB. The number of port pairs that can use a given SRB is $\left(\frac{Nh}{bF}\right)^2$ which goes up as the square of h. From another point of view, the total capacity of the shared resource gets smaller as h increases. This is because the total capacity of the shared resource is the number of MASC pairs multiplied by the capacity per MASC pair, which is $\left(\frac{bF}{h}\right)^2 \frac{hR}{r}$. In summary, because larger h leads to a smaller overall shared resource, it does not necessarily leads to smaller SRB congestion.

5.3 Shared resource bit-pipe (SRB) congestion probability

This section analyzes the probability that a given SRB is congested. When a SRB is congested, future overflow traffic to that SRB will be blocked. Let p_{sc} be the probability that a given SRB is congested. We find the optimal h that will make p_{sc} as small as possible.

The capacity of an SRB is hR bits per second. Each end user session occupies r

bits per second. Therefore, an SRB can hold up to $\frac{hR}{r}$ simultaneous sessions. The probability of SRB congestion is then the probability that more than $\frac{hR}{r}$ sessions are engaged on the given SRB.

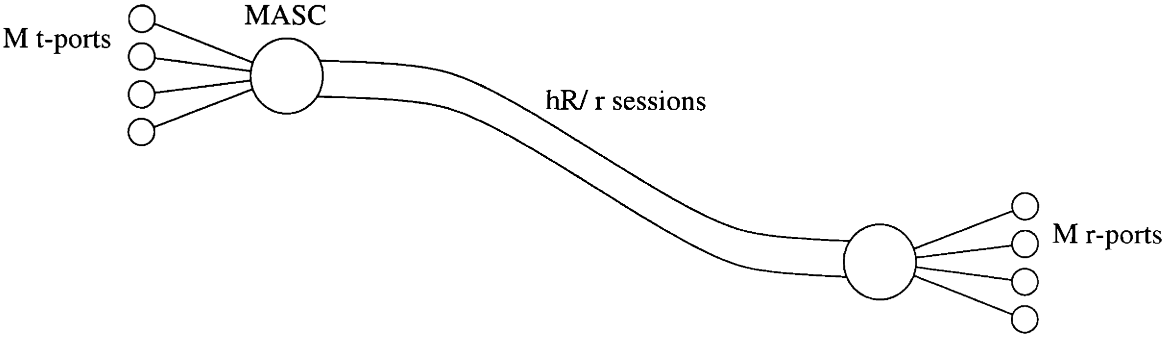


Figure 5-3: A shared resource bit-pipe (SRB) can hold up to $\frac{hR}{r}$ simultaneous end user session.

There are numerous ways an SRB can be congested. An SRB can be congested because exactly 1 port pair (pp) is using it, or exactly 2 pp's are using it, or exactly i pp's are using it. Therefore,

$$p_{sc} = \sum_{i=1}^{M^2} Pr\,(\text{exactly } i \text{ pp's use SRB, SRB congested})\,. \tag{5.1}$$

There are $\binom{M^2}{i}$ ways of choosing the i pp's. Therefore, writing the probability in terms of i specific pp's,

$$p_{sc} = \sum_{i=1}^{M^2} \binom{M^2}{i} Pr\,(\text{given set of } i \text{ pp's use SRB, SRB congested})\,. \tag{5.2}$$

For simplicity, assume the traffic on each pp is independent of any other pp. Since each of the given set of i pp's uses the SRB, each pp must have traffic amount greater than c_s, the session capacity of the dedicated bit-pipes. Furthermore, the number of sessions exceeding c_s, summing over all i pp's, must be greater than $\frac{hR}{r}$. Hence, if i pp's use the SRB and the SRB is congested, then it implies the total traffic within those i pp's must be $\geq ic_s + \frac{hR}{r} = c_s(i + h/k)$, where $k = \frac{c_s r}{R}$ is the coarseness of the dedicated bit-pipes. Therefore, we can upper bound p_{sc} by,

$$p_{sc} \leq \sum_{i=1}^{M^2} \binom{M^2}{i} Pr\left\{\text{total traffic of } i \text{ pp's} \geq c_s(i + h/k)\right\} (1 - p_c)^{M^2 - i}, \tag{5.3}$$

where p_c is the probability of congestion of a dedicated bit-pipe. This is an upper bound because there are cases where less than i pp's are using the SRB, even though the sum of the traffic from i pp's is $\geq c_s(i + h/k)$. Define A as the event that the total traffic in a given set of i pp's is $\geq c_s(i + h/k)$.

There are $\frac{E}{N}$ end users connected to a given port, where E is the total number of end users and N is the total number of ports. The probability of engagement within a pp is $\frac{\phi}{N}$. Simplifying by viewing the traffic within the i pp's as $i\frac{E}{N}$ independent Bernoulli trials, we can upper bound $Pr(A)$ using the same method as in section 4.2.1. The result is:

$$Pr(A) \leq \left[\gamma' \exp(1 - \gamma')\right]^{c_s(i+h/k)}, \tag{5.4}$$

where,

$$\gamma' = \frac{i\phi E}{N^2\left(c_s(i + h/k)\right)}. \tag{5.5}$$

But

$$\gamma' = \frac{i\phi E}{N^2\left(c_s(i + h/k)\right)} \leq \frac{\phi E}{N^2 c_s} = \gamma. \tag{5.6}$$

Therefore,

$$Pr(A) \leq \left[\gamma \ \exp(1 - \gamma)\right]^{c_s(i+h/k)}, \tag{5.7}$$

Let $U = [\gamma \ \exp(1 - \gamma)]^{c_s}$, be the upper bound for the probability of congestion of the dedicated bit-pipe, i.e., $p_c \leq U$ (see section 4.2). Note that the dedicated portion of the network is designed such that $U < 1$. Therefore,

$$p_{sc} \leq \sum_{i=1}^{M^2} \binom{M^2}{i} U^{(i+h/k)} (1 - U)^{M^2 - i}. \tag{5.8}$$

This is simply the sum of a binomial with the $i = 0$ term missing. Re-writing the bound, we have:

$$p_{sc} \leq U^{(h/k)} \left(1 - (1-U)^{M^2}\right). \tag{5.9}$$

Upper bound this by the first term in the expansion of $(1-U)^{M^2}$:

$$p_{sc} \leq M^2 \, U^{(1+h/k)}, \tag{5.10}$$

or equivalently,

$$p_{sc} \leq M^2 \left[\gamma \ \exp(1-\gamma)\right]^{\left(c_s + \frac{hR}{r}\right)}. \tag{5.11}$$

This is a reasonable upper bound for p_{sc} if $U \ll 1$ because the right hand side is approximately the union bound of all events such that exactly one pp overflows the dedicated bit-pipe by $\frac{hR}{r}$ sessions.

Substituting $\frac{Nh}{bF}$ for M, we have,

$$p_{sc} \leq \left(\frac{Nh}{bF}\right)^2 U^{(1+h/k)} \tag{5.12}$$

For small h, the term $\left(\frac{Nh}{bF}\right)^2$ dominates the upperbound. For large h, the term, $U^{(1+h/k)}$ dominates. The upperbound increases until it reaches a maximum, and then decreases with increasing h. The maximum is achieved at

$$h = \left(\frac{-2k}{\ln(U)}\right) \tag{5.13}$$

The valid range for h is between 1 and bF. Therefore, if $\left(\frac{-2k}{\ln(U)}\right) < 1$, then the upper bound always decrease with increasing h.

The goal is to build a shared resource such that the SRB congestion probability is as small as possible. Therefore, if $\left(\frac{N}{bF}\right)^2 U^{(1+1/k)} < (N)^2 \, U^{(1+bF/k)}$ then use $h = 1$, otherwise, use $h = bF$.

When $h = bF$, the shared network is simply a giant star. Or more precisely, the network has b fibers connecting the combining MASC and the broadcasting MASC. All t-ports connect to the single combining MASC, and all r-ports connect to the single broadcasting MASC. The b fibers are shared by all port pairs. Note that this case happens when U is small and b is large. Small U implies that overflow events are rare. Although large h leads to a smaller overall shared resource, it does not hurt the network in this case because overflow events are rare. The more important aspect is that the resource is being shared as widely as possible. Also, larger b provides a larger overall shared resource, and hence, encourages more sharing of the resource.

On the other hand, $h = 1$ should be used when U is relatively large. This is because large U implies many overflow events, and the smallest h provides the largest overall shared resource.

5.4 Shared resource conclusion

The UATA network can be augmented by adding a shared resource. This shared resource takes care of the traffic that overflows the dedicated bit-pipes in the original UATA network. The shared resource consists of bit-pipes, each carrying h wavelengths. Each bit-pipe is shared among many port pairs. To make the probability of congestion on the shared bit-pipes as small as possible, one either uses $h = 1$, or $h = bF$ depending on the parameters of the original UATA network and the amount of traffic allowed into the shared resource. When overflow events are rare, one uses $h = bF$. Although this leads to a smaller shared resource, the network allows maximum sharing of the shared resource. However, when overflow events are frequent, one uses $h = 1$ to maximize the size of the shared resource.

Chapter 6

The Fixed Multiple Connection (FMC) Network

The fixed multiple connection (FMC) traffic model is similar to the uniform all to all traffic (UATA) model in that the port to port traffic, $tr(i, j)$, is constant. Because the port to port traffic is fixed, the backbone is a passive network that contains physical bit-pipes connecting the ports.

For the UATA network, we take advantage of the fact that the traffic is uniform, and use the properties of the Latin Router to create a tightly packed efficient network. However, the FMC traffic is not uniform, specifically, $tr(i, j)$ can be different for all pairs (i, j). Therefore, elegant constructions of the FMC network cannot be found in general.

The most straight forward method to build a FMC network is simply to lay down physical bit-pipes between pairs of ports wherever necessary. However, a more cost effective way may be to combine some ports together so the resulting port to port traffic is more uniform. This way, the FMC problem is transformed to the UATA problem, and the construction for the UATA network can be applied. Unfortunately, for FMC traffic that are skewed, it is often not possible to combine the ports in a reasonable way.

This chapter deals with the FMC traffic. We will not discuss the straight forward method. Instead, we describe the method of combining ports. We then show an example why the FMC problem cannot always be transformed to the UATA problem. Next, we investigate a more general problem of combining the ports in such a way that the resulting port to port traffic can be supported from an existing network infrastructure. We find that this general problem is NP-complete.

6.1 Port combination using MASC's

Let $tr(i,j)$ denote the traffic from t-port i, to r-port j. For the FMC traffic, $tr(i,j)$ is a fixed constant, although the constant depends on the (i,j) pair. The traffic does not have to be symmetric, i.e., $tr(i,j)$ is not necessarily the same as $tr(j,i)$.

It may be desirable to combine groups of ports together so the traffic is more uniform. The ports are combined using the multiple access star couplers (MASC) described in section 3.4.1. As a quick review, the output fibers from the ports are combined together using star couplers within the MASC's. The t-ports are combined using combining MASC's. The r-ports are combined using broadcasting MASC's. This way, bit-pipes connecting two MASC's are shared by a pair of t-port r-port groups.

If only WDM is employed, then the sharing of the bit-pipe is discrete in units of a wavelength. For example, if a bit-pipe has a capacity of 4 wavelengths, then a port can use 4, 3, 2, 1, or 0 wavelengths within the bit-pipe. The discreteness of the sharing can be alleviated if TDM is employed on top of WDM. This way, fractions of a wavelength can be shared. We assume TDM in this chapter, and ignore the discrete aspect of sharing.

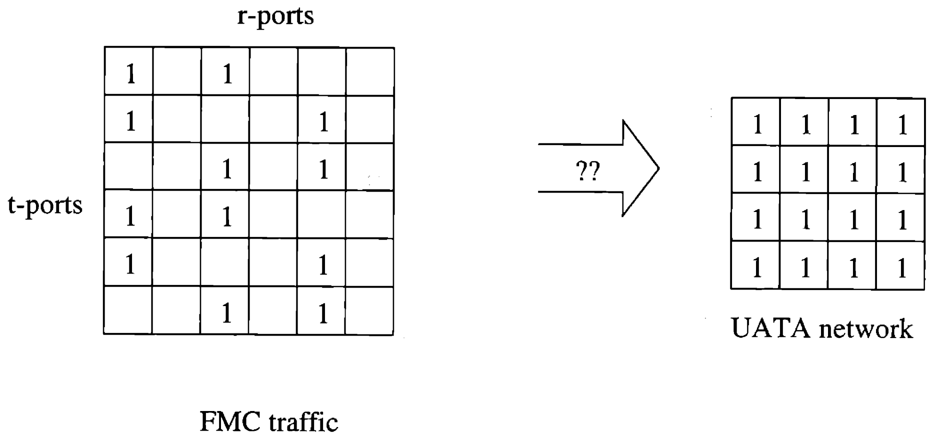
After the MASC's have combined the ports, then the MASC to MASC traffic becomes the traffic of concern. The problem is then to build a backbone that will support the MASC to MASC traffic. Sometimes it is possible to combine the ports such that the MASC to MASC traffic is close to uniform, and a UATA network can be applied. The MASC's can be viewed as another level of aggregation (done in optics rather than electronics). This is exactly what we have done in the UATA network using TDM.

6.2 Example of skewed FMC traffic

Unfortunately, some skewed FMC traffic cannot be reasonably combined to create a resulting traffic pattern that is UATA. Before going into an example, the notion of being 'reasonable' needs to be clarified. One can always combine all the t-ports using one combining MASC, and all the r-ports using one broadcasting MASC. The resulting traffic is degenerate UATA traffic. Therefore, this is not reasonable.

Also, if the total amount of traffic for the FMC is $|\Gamma|$, then it is reasonable to ask if a combination is possible such that the resulting traffic fits within an UATA network with total capacity close to $|\Gamma|$

Figure 6-1 depicts an example where the FMC traffic is skewed in such a way, that it cannot be combined to fit into a given UATA network. The FMC connection is drawn as a matrix on the left, and the UATA network on the right.



1		1			
1				1	
		1		1	
1		1			
1				1	
		1		1	

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

Figure 6-1: Example of a skewed FMC traffic that cannot be easily combined into a UATA network.

The total traffic for the FMC is 16, and the maximum throughput of the UATA network is also 16. Combining two t-ports is equivalent to adding two rows, and combining two r-ports is equivalent to adding two columns. The ports cannot be combined because if any two ports are combined together, then there will be a place where the MASC to MASC traffic is 2 units large. Note that even if the 1's in the FMC traffic matrix is replaced by $0.5 + \epsilon$, where ϵ is a small but finite positive value, the ports still cannot be combined.

The problem of transforming FMC traffic into UATA traffic is a difficult one. Next we investigate the problem of transforming FMC traffic to any given fixed matrix. We want to do this because sometimes a passive network infrastructure already exists, and we would like to use it to support the FMC traffic. As it turns out, this problem is NP-complete.

6.3 General Port assignment problem (PAP)

Suppose a passive optical network infrastructure already exists and we wish to use it as an optical backbone to interconnect local electronic networks. We can first connect the local networks to a port, and then combine the ports using MASC's, hoping that the resulting MASC to MASC traffic fits within the existing physical infrastructure. The input to the infrastructure are the MASC's. Since the existing infrastructure is passive, the capacity from MASC to MASC is fixed. Let $m(x, y)$ be the capacity from MASC x to MASC y.

This leads to the port assignment problem (PAP) which is defined as follows: Assume a situation where an AON backbone has m MASC's. The backbone supports a particular fixed multiple connection traffic from the MASC's point of view. Let $m(x, y)$ be the maximum amount of traffic supported by the backbone from MASC x to MASC y, i.e., the size of the bit pipe from x to y. Also, assume n local networks with known and fixed inter-network traffic. Each local network is connected to the AON backbone via a port. Let $tr(i, j)$ be the offered load, (traffic), from local network i to local network j, or equivalently, from port i to port j. We want to find an assignment connecting the ports to the MASC's so that the inter-network traffic can be supported by the backbone.

Although we have split a physical port into two functioning units, t-port and r-port, both units are still physically together since they both connect to the same local network. This implies symmetric assignment of the ports, i.e, grouping of the t-ports and r-ports are done in the same way. This will be assumed in the PAP.

If an assignment is such that the resulting traffic from MASC x to MASC y is

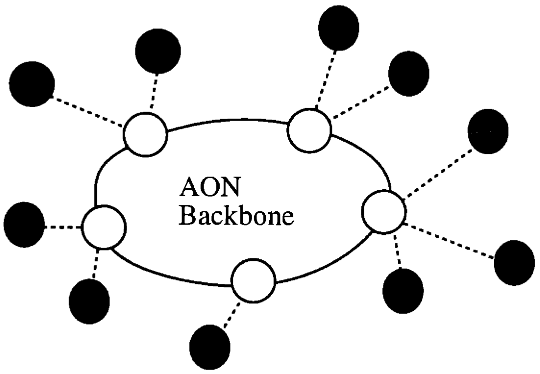


Figure 6-2: The assignment problem is to assign each pot, indicated by dark circles, to a MASC, indicated by white circles.

greater than the allowed traffic $m(x,y)$, then that assignment is not valid. In general, a valid assignment may not exist because the backbone may not be large enough to handle the inter-network traffic. A PAP is *not feasible* if no valid assignment can be found. Feasibility for the PAP is NP-complete for a general FMC traffic and a general passive backbone.

Note that the feasibility of a PAP does not consider the geographical constraints of the problem. The solution may be impractical if a port is assigned to a far away MASC. We analyze PAP feasibility because it is an interesting graph problem in its own right.

6.3.1 NP-completeness of PAP

The PAP can be modeled using two graphs. The first is a directed graph representing the ports and the inter-network traffic demands. This graph is called the *demand* graph. The second graph is another directed graph representing the existing infrastructure and the MASC to MASC traffic supported by the infrastructure. This graph is called the *support* graph. The assignment problem is equivalent to transforming the demand graph via a process called *merging* so that the resulting merged graph is *included* in the support graph. The concept of merging and inclusion will be defined shortly.

The demand graph, $G_d(V_d, E_d)$, models the inter-network traffic. The set of nodes of the graph, V_d, is the set of ports desiring connections to the MASC's. A directed edge, $(u,v) \in E_d$, with weight $tr(u,v)$, represents the amount of traffic from port u to port v. An edge with zero weight is equivalent to a non-existent edge.

The support graph, $G_s(V_s, E_s)$, models the MASC to MASC traffic supported by the passive AON backbone. Nodes of the graph, V_s, represent the MASC's. A directed edge, $(u,v) \in E_s$, with weight $m(u,v)$, represents the amount of traffic allowed from port u to port v. Note that self loops are included to indicate traffic allowed between ports connecting the same MASC. Therefore, $m(u,u)$ is the amount of traffic allowed, in the backbone, between combining MASC u and broadcasting MASC u.

Define *merging* of a group of nodes Φ into a new node x as follows: Draw a new node x. Form a directed edge (x,i) with weight $tr(x,i) = \sum_{u\in\Phi} tr(u,i)$ for each $i \notin \Phi$, and form a directed edge (j,x) with weight $tr(j,x) = \sum_{v\in\Phi} tr(j,v)$ for each $j \notin \Phi$. Also form an edge (x,x), a self loop, with weight $tr(x,x) = \sum_{u\in\Phi}\sum_{v\in\Phi} tr(u,v)$. Lastly, delete the group of nodes in Φ along with all edges adjacent to Φ.

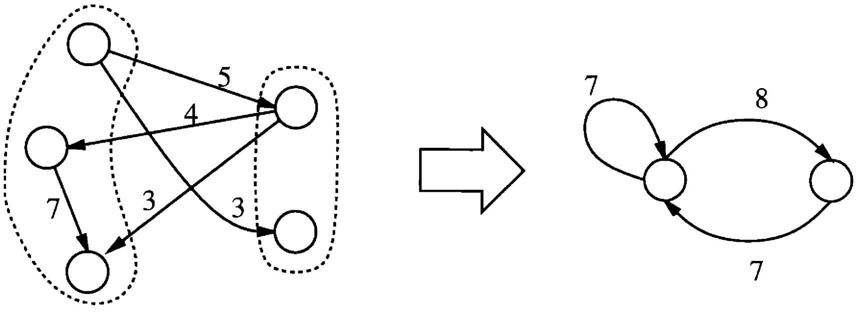


Figure 6-3: An example where a 5 node graph is transformed into a 2 node graph by merging two groups of nodes simultaneously.

Merging can be done simultaneously on multiple groups. A *merged graph* is a graph where each resulting node represents the merging of a group of original nodes. An example where a graph is transformed by merging is depicted in figure 6-3. Note that merging preserves the total weight of the graph. Merging produces different merged graphs by choosing the node groupings differently. The goal is to choose the groupings in such a way that the merged graph is included in the support graph.

A merged graph, $G_m(V_m, E_m)$ is *included* in the support graph, if and only if the following two conditions are true: First, $|V_s| \geq |V_m|$. Second, there exists a one to one assignment function $x = \Psi(u), x \in V_s$ and $u \in V_m$, such that $tr(u, v) \leq p(\Psi(u), \Psi(v))$ for all $u, v \in V_m$. (If $|V_s| \neq |V_m|$, add extra nodes without edges to G_m so the number of nodes of G_m and G_s are the same.) The above definition simply says that a graph is included in G_s if it fits inside of G_s. An example of inclusion is shown in figure 6-4.

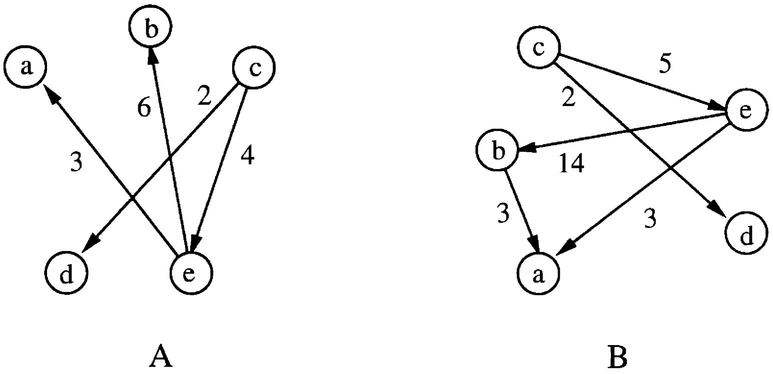


Figure 6-4: An example of inclusion, graph A is included in graph B.

The PAP is equivalent to finding groupings of the nodes of the demand graph, such that if the the groups are merged, the resulting graph is included in the support graph. The assignment function Ψ, dictates the grouping of the ports. Sometimes it is impossible to group the nodes of the demand graph so the resulting merged graph is included in the support graph. For a general demand graph and a general support graph the feasibility of the assignment problem is NP-complete.

To show NP-completeness, we will first show that PAP is polynomially verifiable. Then, we will map a known NP-complete problem into a particular instance of PAP.

PAP is polynomially verifiable. Given a possible solution to a PAP, the solution can be verified by performing the merge and checking to see if the resulting graph is included in the support graph. Assume an N node demand graph. For each group, merge creates at most N new edges. There are at most N groups. So the total number of new edges created is at most $O(N^2)$ The total number of old edges deleted during merge is less than or equal to the number of edges in the original demand graph which is less than or equal to $O(N^2)$. Hence, merging takes at most $O(N^2)$ steps. Checking of inclusion takes at most $O(N^2)$ steps. This is because one step is taken to check that each edge has not exceeded the maximum weight supported by the support graph. Therefore, PAP is verifiable in polynomial time.

To show that PAP is NP-complete, we will map, in polynomial time, a known NP-complete problem to a specific instance of PAP. The NP-complete problem is the graph k-colorability problem defined below:

> The k-coloring problem – Given an undirected graph $G(V, E)$, and a fixed positive integer k, is there a way to color the nodes of V with k colors such that no two neighboring nodes have the same color? More formally, does there exist a function $f : V \to \{1, 2, \ldots, k\}$ such that $f(u) \neq f(v)$ whenever $(u, v) \in E$?

The k-coloring problem is NP-complete [GJ 79, p.191]. We will narrow down the general PAP problem into a specific problem and show that the k-coloring problem can be mapped to an instance of this specific problem.

A directed graph is symmetric if edge (u, v) has the same weight as edge (v, u), for all nodes $u \neq v$. If both the demand and the support graphs of PAP are symmetric, then one can replace the directed graphs with undirected graphs. Specifically, in both graphs, replace each pair of directed edges (u, v) and (v, u) with an undirected edge (u, v) of the same weight. For self loops, (u, u), simply replace the directed edge with an undirected edge.

The symmetric PAP (SPAP) is a specific case of PAP. To solve SPAP, one can use a modified merge defined as follows: To merge a group of nodes $u \in \Phi$ into a new node x, draw a new node x. Form undirected edge (x, i) with weight $tr(x, i) = \sum_{u \in \Phi} tr(u, i)$ for all $i \notin \Phi$. Also form edge (x, x), a self loop, with weight $tr(x, x) = \sum_{u \in \Phi} \sum_{v \neq u \in \Phi} tr(u, v)$. Lastly, delete the group of nodes in Φ along with all edges adjacent to Φ.

Inclusion for undirected graphs is exactly the same as defined for the directed graphs, and therefore, will not be repeated here. Obviously, the SPAP is feasible if one can merge the undirected demand graph so the resultant merged graph is included in the undirected support graph. We now show that any k-coloring problem is equivalent to an instance of the SPAP.

Take an undirected and connected graph $G(V, E)$, and replace each edge $(u, v) \in E$ with a new node w and two new edges, (u, w) and (w, v). Denote W as the set of all the new nodes. Furthermore, add edges so the nodes in W form a complete graph, i.e., create edge (x, y) for all $x, y \in W$ and $x \neq y$. Denote $G_n(V_n, E_n)$ as the new graph. Note that $V_n = V \bigcup W$. Let all the edges have unity weight. An example of a G_n created from G is depicted in figure 6-5. Note that G_n can be created in polynomial time.

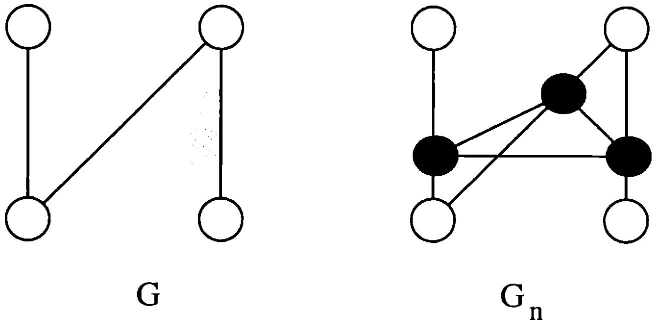


Figure 6-5: To show the equivalence between merging and k-coloring, a new graph, G_n, is created from G by replacing each edges with a new node and two edges. New edges are also added so the new nodes form a complete graph. New nodes are denoted by dark circles.

Let $G_s(V_s, E_s)$ be a complete graph of $k + |E|$ nodes with edges of unity weight. (Note that a complete graph has no self loops). Claim:

The SPAP with G_n as the demand graph and G_s as the support graph is equivalent to the k-coloring problem of G.

Proof:

Fact 1: Two nodes in G_n cannot be merged if they are adjacent to each other or adjacent to a common node. This is because if two adjacent nodes are merged, then the resulting merged graph will have a self loop. If two nodes that are adjacent to a common node are merged together, then the resulting graph will have an edge of weight 2. In both cases, the merged graph cannot be included in the support graph G_s because G_s has unity weight edges and no self loops.

Fact 2: In $G_n(V_n, E_n)$, (remember $V_n = V \bigcup W$), only nodes in V can be merged together. This is a consequence of fact 1. All the nodes in W are adjacent to each other, and therefore, cannot be merged together. A node $u \in W$ cannot be merged with any nodes in V because u is a neighbor to all nodes in W and a node in W can always be found next to any node in V. In other words, nodes in V are always at most 2 edges away from any node in W.

Fact 3: If two nodes in G_n can be merged, then the corresponding nodes in G can have the same color. Only nodes of V matter because no nodes in W can be merged. If two nodes are merged, that means they are at least 3 edges apart in G_n which means they are at least 2 edges apart in G, which further implies that they can have the same color.

Fact 4: The nodes of G_n in V are at least 2 edges away from each other. This fact is due to construction. If nodes are in V, then they are in the original graph G. These nodes cannot be next to each other because all the edges in G are replaced by two edges, and no new edges connecting nodes in V is added during the construction of G_n.

Fact 5: If two nodes in G have the same color, then the corresponding two nodes in G_n can be merged. We will show this by contradiction. Suppose two nodes in G have the same color but the corresponding two nodes in G_n cannot be merged. Because they cannot be merged, these two nodes must be at most 2 edges away from each other. This, together with fact 4 implies that these two nodes are exactly 2 edges away. This means these two nodes must be adjacent to each other in the original G. However, two adjacent nodes cannot have the same color. This is a contradiction, therefore, the assumption is false, and the fact is true.

Facts 3 and 5 shows that if we know how to color G, then we know how to merge G_n and vice versa. Hence, k-coloring of G is equivalent to merging of G_n into k subsets. $\square$

There are no restrictions on G, except that it has to be connected. Therefore, any instance of k-coloring can be transformed, in polynomial time, to an instance of SPAP, which is an instance of PAP. The claim implies that the PAP is at least as hard as k-coloring. Since k-coloring is NP-complete and PAP is P verifiable, PAP is NP-complete.

Chapter 7

Variable Multiple Connection (VMC) Network

In practice, even with aggregation, it is often unavoidable that the expected port to port traffic will vary with time. These variations are not statistical deviations from the mean. Instead, the actual traffic statistics can change due to factors such as the time of day. We call this type of traffic *Variable Multiple Connection (VMC)* traffic. Both the UATA and the FMC backbones do not handle VMC traffic well because the bit-pipes have constant capacity.

In this chapter, we design a backbone such that the port to port capacity can vary according to need. The backbone is flexible enough that resources not being used by a port pair can be utilized by another port pair. The basic channel for this chapter is assumed to be one wavelength (i.e, WDM without TDM). This is because time slot alignment is difficult in the type of network we will be designing. However, if timing problems can be solved, the results in this chapter can be easily extended to networks using TDM.

7.1 Traffic model

As before, let $tr(i, j)$ be the amount of traffic going from port i to port j. For VMC traffic, $tr(i, j)$ depends on i and j, and varies slowly with time. Therefore, a more accurate traffic representation is $tr(i,j,t)$. The changes occur slowly enough that there is ample time for the backbone to reconfigure itself to match the changing traffic. Also, since we are working in a WDM environment, $tr(i,j,t)$ will be a non-negative integer number of wavelengths.

We assume that there is some nominal traffic between each port pair. Hence $tr(i,j,t) \geq tn(i,j)$, where $tn(i,j)$ is a fixed constant depending on i, and j. Let $v(i,j,t) = tr(i,j,t) - tn(i,j)$. Then $v(i,j,t)$ is the amount of traffic that varies over time from port i to port j.

The fixed portion of the traffic $tn(i,j)$ can be handled by a FMC network (FMC networks are described in chapter 6). Therefore, we propose a two part network, where one part handles the fixed portion of the traffic and the other part handles the variable portion of the traffic. Effectively, we are augmenting a FMC network with a VMC network so the variation in traffic can be handled by the VMC network. (See figure 7-1) We assume that the FMC network is much larger than the VMC network. This is reasonable if the amount of aggregation is large.

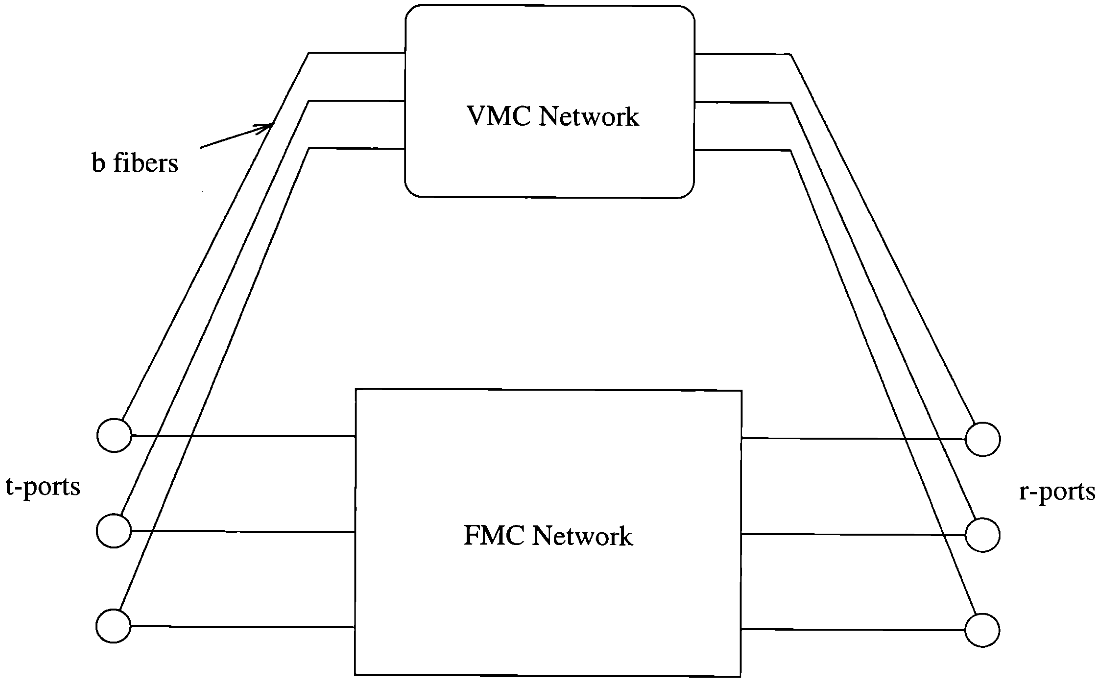


Figure 7-1: An FMC network augmented by a VMC network. The fixed portion of the traffic is handled by the FMC network, while the variable portion of the traffic is handled by the VMC network.

We can now focus our attention on the variable traffic $v(i,j,t)$ and the VMC part of the network. Therefore, we assume that the port to port traffic is $v(i,j,t)$, and ignore the fixed portion of the traffic, and also ignore the FMC portion of the network.

Figure 7-1 closely resembles figure 5-1. One may want to use the construction in chapter 5, i.e., shared resource bit-pipes (SRB) to deal with variable traffic. This is possible except that the SRB is shared among different port pairs and no guaranteed bandwidth is allocated. Therefore, one port pair is able to occupy all the available bandwidth on a SRB and prevent other port pairs from using the SRB. In this chapter, we would like to guarantee that each port is allowed to send and receive a given amount of traffic, regardless of what other ports are doing.

Let b be the number of fibers connecting each port to the VMC network. Then, bF, where F is the number of wavelengths per fiber, is the maximum number of wavelengths over which a port can send and receive data. We impose the following restrictions on $v(i,j,t)$:

$$\sum_i v(i,j,t) \leq bF, \qquad \text{for all } j,t \tag{7.1}$$

and

$$\sum_j v(i,j,t) \leq bF, \qquad \text{for all } i,t. \tag{7.2}$$

The above equations simply state that, at any given time t, the total amount of variable traffic coming out or going into a port must not exceed bF. If a traffic assignment satisfies the above two equations, we call it a *valid* traffic assignment. As a direct consequence,

$$0 \leq v(i,j,t) \leq bF, \qquad \text{for all } i,j. \tag{7.3}$$

Our goal is to build a network that supports all valid traffic assignments.

On average, there are $\frac{bF}{N}$ wavelengths between each port pair, where N is the total number of ports. In the case that $\frac{bF}{N}$ is an integer, the VMC network is able to implement an UATA network with $\frac{bF}{N}$ wavelengths in each bit-pipe.

7.2 VMC Network design

We are interested in designing a VMC network that supports the VMC traffic. There are two backbone requirements. 1) Capacity requirement: each t-port must have bF transmitters and each r-port must have bF receivers. (We are only counting the transmitters and receivers used for the VMC portion of the network.) The maximum throughput of the VMC network is NbF. 2) Flexibility requirement: the backbone must be able to support any valid traffic assignment, i.e., any set of $v(i, j, t)$ such that equation 7.1 and equation 7.2 are satisfied.

The major components of the backbone are the wavelength specific spatial switches (WSSS) and the AON ports. The WSSS's connect the ports together and set up bit-pipes of various sizes between port pairs. We now describe the WSSS.

7.2.1 Wavelength specific spatial switch

The WSSS is, as the name suggests, a spatial switch that is wavelength specific, i.e., the switch setting for different wavelengths are independent of one another. An easy, though perhaps not the most efficient, way of building a WSSS is to use multiplexers, demultiplexers, and regular spatial switches. This is depicted in figure 7-2

The multiplexers and demultiplexers effectively divide the device into wavelength layers. Each wavelength layer is connected together using a regular spatial switch. The setting of the switch for each layer is independent of other layers. The number of wavelength layers is at most F, where F is the number of wavelength channels per fiber in the WDM system. Not all F layers need to be used. So WSSS's can be designed to have between 1 to F different wavelengths.

Figure 7-2 shows a 2 x 2 switch. In fact, larger switches can be built using the 2 x 2 WSSS's as the basic building blocks. There are a variety of methods to do this, and they are analogous to building large non-blocking spatial switches. For example, one can use construction methods for a Beneš network or a Clos network.

A non-blocking switch supports permutation traffic. Permutation traffic is a traffic assignment such that at any given moment each t-port is connected to exactly

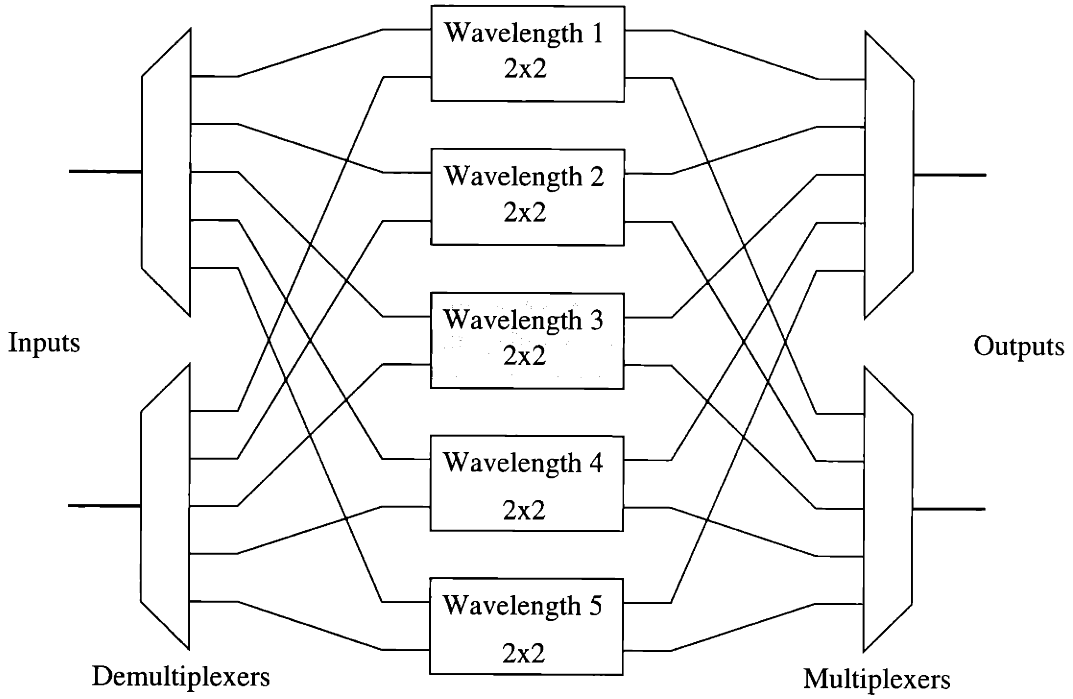


Figure 7-2: A possible implementation of a 2 x 2 wavelength specific spatial switch. The inputs are demultiplexed and connected by spatial switches. The outputs are multiplexed back together using a wavelength multiplexer. Each wavelength layer can individually controlled.

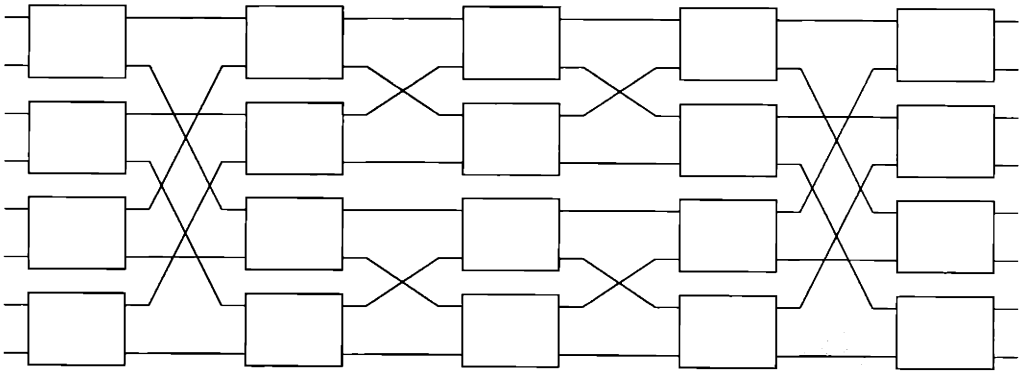


Figure 7-3: Scaling a WSSS from building blocks. Here, we show how an 8 x 8 switch can be build using 2 x 2 switches.

one r-port. The actual pairing of the ports can change over time. We say a switch performs permutation routing if it supports the permutation traffic, even as the port pairings change over time. Non-blocking switches perform permutation routing. Using construction methods for large non-blocking spatial switches, we can construct large non-blocking WSSS. Since each WSSS contains independent wavelength layers, a non-blocking WSSS is equivalent to many (up to F) layers of independent non-blocking switches, and each layer support a permutation traffic.

Methods of building large non-block switches are well known and spelled out in references such as [Hui 90]. Therefore, they will not be expanded further here. Interested readers are encouraged to consult the references. The important aspect is that a non-blocking WSSS separates the inputs into wavelength layers. Each layer is a non-blocking and independently controlled spatial switch. In [Pan 92], the concept of looking at wavelength layers using WSSS's is more fully developed. The author further investigated the interaction between different layers by using wavelength changers. Here, we simply take the layering concept and apply it to our VMC network.

7.2.2 Network design and its validity

The VMC network must support all valid VMC traffic. At any given instant, say $t = t_o$, a valid VMC traffic, $v(i, j, t_o)$, can be represented by a bipartite graph $G(L, R, E)$ as shown in figure 7-4a. The nodes on the left column belong to set L, and they represent the t-ports. The nodes on the right belong to set R, and they represent the r-ports. We assume that the number of t-ports equal the number of r-ports. So $|L| = |R|$. The set E represents the set of edges. Each edge is one wavelength connection between the connecting ports.

Equation 7.1 and equation 7.2 force the degree of each node to be $\leq bF$. For simplicity, modify the graph by adding extra edges such that the degree of each node is exactly bF. We can always do this because if a t-port node exists with degree less than bF, then there must be an r-port node also with degree less than bF. An extra edge can then be added between those two nodes. Note that there may be more than one edge connected between a pair of nodes.

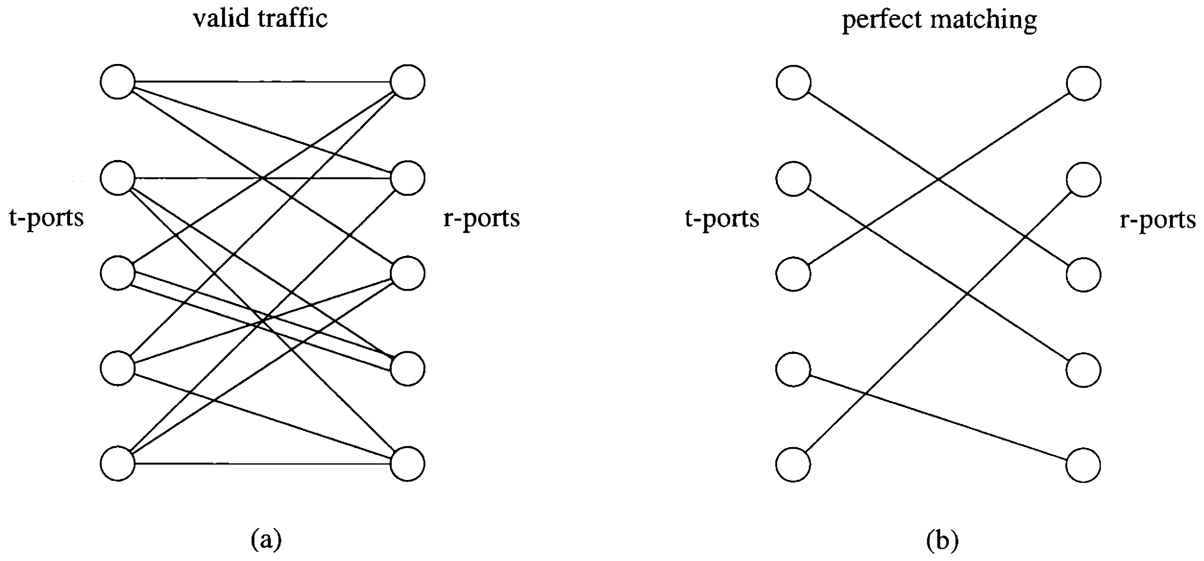


Figure 7-4: (a) VMC traffic represented by a bipartite graph. The left column of nodes represent the t-ports, and the right column of nodes represent the r-ports. The degree of each node is at most bF. (b) a perfect matching extracted from the bipartite graph.

If we build a network that support the traffic described by the modified graph, then that same network can support the traffic in the original graph.

Define a *perfect matching* of a bipartite graph $G(L, R, E)$ as a set of edges, P, such that $|P| = |L| = |R|$, and every node in G touches exactly one edge in P. Figure 7-4b is an example of a perfect matching extracted from the bipartite graph in figure 7-4a. Note that a perfect matching represents a specific instance of permutation traffic because each t-port is connected to exactly one r-port. Permutation traffic can be supported by a non-blocking switch.

Claim: A bipartite graph with nodes of degree d is made up of d layers of perfect matchings.

Before proving the claim, we will first describe the consequence of the claim. If the claim is true, then our traffic, represented as a bipartite graph with node degree bF, can be separated into bF layers of permutation traffic. Then, the required network is one that can support bF independent layers of permutation traffic. Since each N x N WSSS can support up to F independent layers of permutation routing, all we need are b of these N x N WSSS's. Each t-port has one fiber connection to one input of each of the b WSSS's, and each r-port has one fiber connection to one output of each of the b WSSS's. Therefore, if the claim is true, then the required network is simply b N x N WSSS's.

We will prove the claim by induction. If $d = 1$, then the claim is true by definition. Next, assume that the claim is true for $d - 1$; we must show that it is also true for d. To do this, we invoke Hall's theorem [Hal 35] which is stated below:

Hall's Theorem: A $2N$-node bipartite graph $G = (L, R, E)$ has a perfect matching if for all subsets $S \subseteq L$, $|\mathcal{A}(S)| \geq |S|$, where $\mathcal{A}(S)$ denotes the nodes in R that are adjacent to a node in S.

In our case, we have a bipartite graph where the degree of each node is exactly d. We will show that the condition for perfect matching as stated in Hall's Theorem is satisfied. We do this by contradiction. Suppose $|S| > |\mathcal{A}(S)|$. There are $d\,|S|$ edges incident on S. Each of these $d\,|S|$ edges also incident on $\mathcal{A}(S)$. The average number of edges per node in $\mathcal{A}(S)$ is $\frac{d|S|}{|\mathcal{A}(S)|}$. This implies that some node in $A(S)$ must have at least $\frac{d|S|}{|\mathcal{A}(S)|} > d$ edges. This contradicts the fact that all nodes have degree d. Therefore, the condition in Hall's theorem is met, and there is a perfect matching in the bipartite graph of degree d.

Continuing with the induction proof, delete the edges of the perfect matching in the degree d graph. We are then left with a graph where all nodes have degree $d - 1$. By the induction hypothesis, this graph has $d - 1$ perfect matchings. Therefore, the bipartite graph with degree d nodes has d perfect matchings, and our claim is proved. Furthermore, the network with b N x N WSSS's can support any valid traffic.

7.2.3 VMC design is reasonable

We have shown that a network consisting of b layers of N x N WSSS's can support any valid traffic $v(i, j, t)$. However, the network appears to be more complicated than necessary. In this section, we show that in some sense, our design is reasonable.

As proved, at any given moment, any valid traffic can be separated into bF layers of permutation traffic. Each permutation assignment is supported using one layer of the non-blocking switch. We show that each layer must perform all possible permutations.

Let $\mathcal{T}$ be any valid traffic assignment, and $\mathcal{P}_i$ be a particular permutation assignment. Then $\mathcal{T} = \sum_{j=1}^{bF} \mathcal{P}_{i_j}$ because the valid traffic assignment can be split into bF layers of permustation assignments. If $\mathcal{P}_{i_j} = \mathcal{P}_x$ for all i_j, then the only way to support $\mathcal{T}$ is to have each layer support $\mathcal{P}_x$. This is because the only perfect matching is $\mathcal{P}_x$. $\mathcal{P}_x$ can be any permutation traffic. Therefore, each layer must perform all such $\mathcal{P}_x$.

7.3 VMC conclusions

The VMC traffic, $tr(i,j,t)$ can be represented as $tr(i,j,t) = tn(i,j) + v(i,j,t)$, where $tn(i,j)$ is fixed, and $v(i,j,t)$ varies with time. The fixed portion can be handled by a FMC network, and the variable portion by a network consisted of WSSS's. The variable traffic can be split into bF layers of permutation traffic, and each permutation traffic is supported by a wavelength layer within a N x N WSSS. A total of b WSSS's are required.

Chapter 8

Point to Point AON

So far, the thesis has investigated an AON backbone network that supports traffic resulting from the aggregation of many users. This remaining chapter concentrates on an AON network that supports point to point high rate traffic. Since the concept of local network connected to a backbone does not exist here, the AON will simply be called a network. The end users attach directly onto the network. Alternatively, a local area network could send point to point traffic to some given user or some other local area network.

Since optical packet switching technology is still immature in its developing stages, the most sensible way of dealing with point to point connections is to set up circuits. Other devices we exclude from the network are dynamic wavelength changers, optical memory, and optical time slot interchangers.

We allow the following devices in the network: wavelength routers, wavelength multiplexers and demultiplexers, wavelength filters, frequency selective and non-selective spatial switches, and static wavelength changers. The virtual circuits are determined by the wavelength and the network state. The network state is determined by the state of the included devices.

The traffic model assumed in this chapter is the permutation traffic. In chapter 7, we designed a network supporting many independent layers of permutation traffic. Here, we are interested in only one layer. Therefore, we can use the tuning ability of the transmitters and receivers to perform switching. In fact, we are interested in supporting as many users as possible using just transmitter and receiver tuning. Or, equivalently, use as small number of wavelengths as possible to build a non-blocking network without using spatial switches.

Networks that perform permutation routing are called *non-blocking networks.* Furthermore, they are *rearrangeably* non-blocking if existing connection might need to be rearranged during new connection set up. In this chapter we report on three constructions of rearrangeably non-blocking networks using $O\left(N^{2/3}\right)$ wavelengths. The first construction is adapted from a 2-stage switching network proposed by [FFP 88]. We show this construction to specifically illustrate how the network can be built using known optical components. The second uses the concept of broadcast, and is a new design. Both of these constructions require static wavelength changers. Lastly, we show a third construction, also new, that does not require any wavelength changers.

Using $O\left(N^{2/3}\right)$ wavelengths is not optimal because [Bar 93] proved the existence of a non-blocking network using $O\left(\sqrt{N \log N}\right)$ wavelengths. However, no explicit construction was given. Our constructions are explicit. Also, constructions using $\sqrt{N} 2^{(\log N)^{(4/5+o(1))}}$ can be derived using methods outlined in [Ag+ 94] and in [WZ 93], and this beats our constructions asymptotically. However, our construction uses $4N^{2/3}$, and this beats $\sqrt{N} 2^{(\log N)^{(4/5+o(1))}}$ for all $N < 2^{7765}$. Therefore, for all practical N, our construction is better.

8.1 Connection matrix

A matrix used by the constructions is described in this section. This matrix will be called the *connection matrix* and is denoted by C. The matrix has 1's and 0's as its elements. When we use the matrix in the constructions, a "1" will denote a connection, and a "0" will denote an open circuit.

Let $q = p^k$, where $p > 1$ is prime and k is any positive integer. There exists a finite field with q elements, (See Theorem 4.417 of [Ber 84]), which is denoted by $GF(q)$.

Let each row of the matrix represent a distinct polynomial of degree at most d over the finite field $GF(q)$. Denote the polynomial of the i^{th} row by $f_i(x)$. There are a total of q^{d+1} such polynomials. Therefore, there can be at most $N = q^{d+1}$ rows.

Each column of the matrix is labelled by an ordered pair, (x, y), where $(x, y) \in GF(q)^2$. Hence, there are exactly q^2 columns.

The matrix element at row i and column (x, y) is a "1" if and only if $f_i(x) = y$, otherwise, the element is a "0". Therefore, each row contains exactly q 1's; one for each possible value of x and the corresponding $y = f_i(x)$.

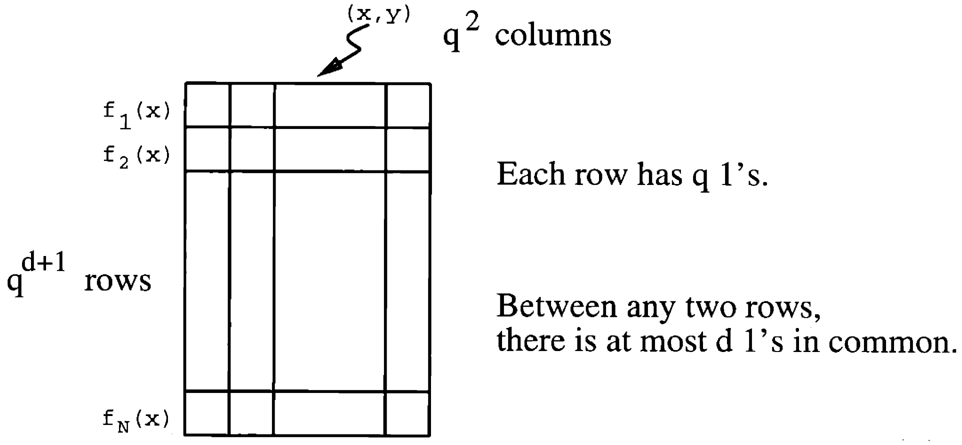


Figure 8-1: The connection matrix used in both constructions.

Define the *overlap* between two rows to be the number of columns such that both row entries are 1's. Also, we say row i and j overlap at column (x, y) if $f_i(x) = y$ and $f_j(x) = y$.

Property 1 (P1): There are at most d overlaps between any two rows.

This property is true because an overlap implies $f_i(x) - f_j(x) = 0$. $f_i(x) - f_j(x)$ is a polynomial of degree at most d over $GF(q)$ which has at most d roots (see Theorem 2.15 in [Ber 84]). Therefore, there are at most d ordered pairs (x, y) such that $f_i(x) = f_j(x) = y$.

View each row of the connection matrix C as a set containing a collection of columns. A column belongs to a row-set if and only if the column entry at that row is a "1". Hence, each row-set has exactly q members. A *representative* of a row-set is a column belonging to the set selected to represent that set. A collection of sets is said to have a *system of distinct representatives* if and only if the representatives can be selected such that all representatives are unique.

A necessary and sufficient condition for a collection of k sets to have a system of distinct representatives is: for $1 \leq i \leq k$, the union of any i sets out of that k contains at least i distinct elements [Hal 86].

Property 2 (P2): Any collection of $\leq \frac{q^2}{2d}$ rows has a system of distinct representatives. This property will be used for the first and the third construction.

To prove this property, all we have to show is the following condition (C1): for any $1 \leq i \leq \frac{q^2}{2d}$, the union of any i row-sets contains at least i columns. For $i = 1$, any given row contains q columns. Therefore, (C1) is satisfied for $i = 1$. For $1 < i \leq \left\lfloor \frac{q}{d} \right\rfloor$ the collection of i sets contains at least $q + (q - d) + (q - 2d) + \ldots + (q - (i - 1)d)$ columns. This is because the first row contains exactly q columns, and because of (P1), the second row adds at least $(q-d)$ new columns, and the third row adds at least another $(q - 2d)$ new columns, etc. Since there are i terms in the sum, and each term is greater than 1, (C1) is satisfied for $i \leq \left\lfloor \frac{q}{d} \right\rfloor$. For $i > \left\lfloor \frac{q}{d} \right\rfloor$, the number of columns contained in the union of i row-sets is at least $q+(q-d)+(q-2d)+\ldots+(q-\lfloor q/d \rfloor\, d)$. We want to find the largest i such that (C1) is satisfied. (C1) is satisfied if

$$i \leq q + (q - d) + (q - 2d) + \ldots + (q - \left\lfloor \frac{q}{d} \right\rfloor d) \qquad (8.1)$$

The right hand side can be re-written as:

$$\begin{aligned} & d\left\{\frac{q}{d} + (\frac{q}{d} - 1) + (\frac{q}{d} - 2) + \ldots + (\frac{q}{d} - \left\lfloor \frac{q}{d} \right\rfloor)\right\} \\ &= d\left\{\frac{q}{d}\left(\left\lfloor \frac{q}{d} \right\rfloor + 1\right) - \frac{1}{2}\left\lfloor \frac{q}{d} \right\rfloor \left(\left\lfloor \frac{q}{d} \right\rfloor + 1\right)\right\} \\ &= d\left(\left\lfloor \frac{q}{d} \right\rfloor + 1\right)\left(\frac{q}{d} - \frac{1}{2}\left\lfloor \frac{q}{d} \right\rfloor\right), \end{aligned}$$

which can be lower bounded by

$$d\left(\left\lfloor \frac{q}{d} \right\rfloor + 1\right)\left(\frac{q}{2d}\right) \geq \frac{q^2}{2d}.$$

Therefore, as long as,

$$i \leq \frac{q^2}{2d}, \tag{8.2}$$

(P2) holds.

Property 3 (P3): Given any collection of $l \leq \frac{q-1}{d} + 1$ row-sets, every row-set in the collection contains a column that is not a member of any of the other $l-1$ row-sets. This property will be used in the second construction.

To prove (P3), take a collection of l row-sets. Observe any one row-set. At most d members of this row-set can be in common with any other row-set due to (P1). Remember that each row has q members. Therefore, as long as $q - d(l-1) \geq 1$, then at least one member of that row-set does not belong to any of the other $l-1$ row-sets. Hence, (P3) is true for $l \leq \frac{q-1}{d} + 1$.

8.2 A network construction adapted from a two stage switching network

We want to transform a switching network to a switchless all optical network. To do so, we can use the tuning capability of a transmitter in place of a 1 x F switch. This is shown in figure 8-2. Depending on the wavelength that the transmitter is tuned to, the data goes out on the corresponding output of the switch. Similarly, a receiver can be viewed as an F x 1 switch. The tuning of the receiver corresponds to listening to one of the inputs to the switch. Note that the middle of the network operates on one frequency only, namely f_1. Note that the multiplexer can be constructed by using a star coupler.

The construction dictates how the outputs of the transmitter switches are connected to inputs of the receiver switches. Without any active switching elements, an all optical network can be viewed as a 2 stage switching network.

Both demultiplexing and multiplexing can be done in stages, (see figure 8-3.) For the demultiplexer, the first stage separates the input into bundles of wavelengths. The next stage demultiplexes the result into individual wavelengths. Although figure 8-3

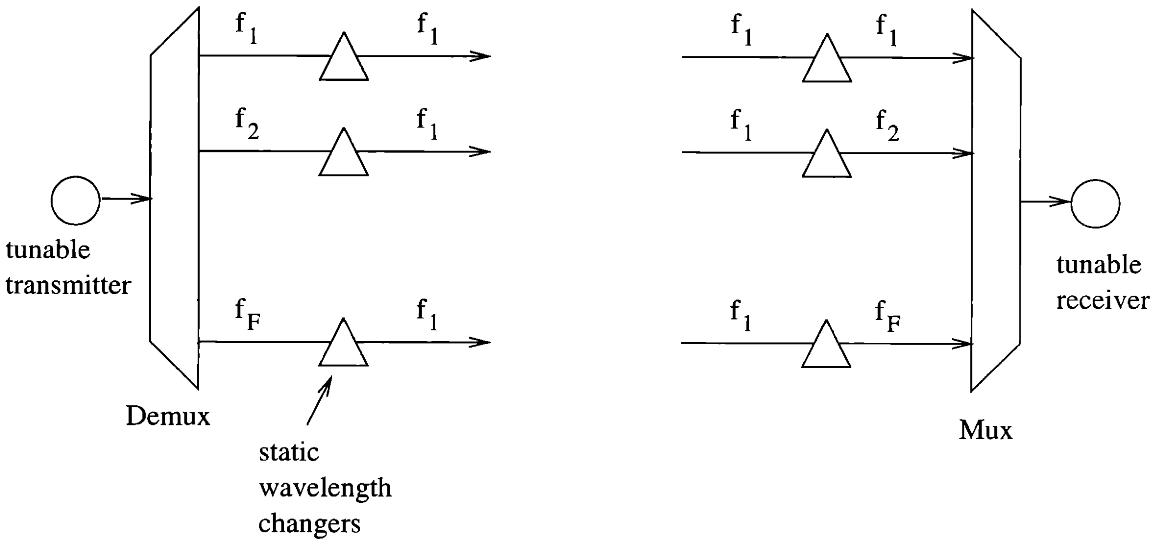


Figure 8-2: Tunable transmitter viewed as a 1xF switch, and tunable receiver viewed as an Fx1 switch.

shows only two stages, this concept can be generalized using many stages where in each successive stage, the demultiplexer separates the input wavelengths into smaller and smaller bundles. Note that each wavelength comes out of exactly one output of the last stage. Similarly, multiplexing can be done in multiple stages. In each stage, all inputs to the multiplexer are gathered together.

The first construction is an adaptation of a 2 stage switching network first described in [FFP 88]. Here we show how it can be converted to a passive optical network without switches. The validity proof shown here is slightly different than the proof in the original paper.

The network is built from smaller sub-network blocks. (See figure 8-4.) The blocks on the left are N x q^2 sub-networks each using the connection matrix C. Call these sub-network blocks $N1$. Identify the rows of C with the inputs of $N1$. (Note that the number of inputs is $N = q^{d+1}$). Identify the columns of C with the output of $N1$. The element "1" in C indicates a connection between the input and the output, the element "0" indicates an open circuit. Each input of an $N1$ block is connected to exactly q outputs. This fan out connection to q outputs is accomplished by the second stage of demultiplexers depicted in figure 8-3. Using transmitter tuning, the data from an input can be switched to any one of the q connecting outputs on any

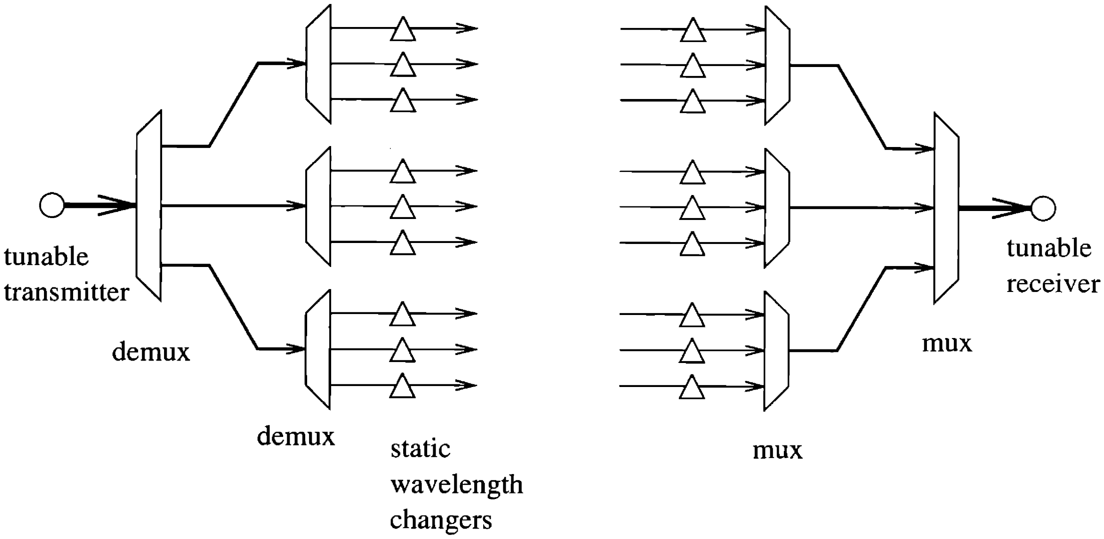


Figure 8-3: The Fx1 switch implemented using multiple stages of multiplexers and demultiplexers.

one of the $N1$ blocks.

On any of the $N1$ blocks, as long as there are $\leq \frac{q^2}{2d}$ inputs active, Property (P2) of the matrix guarantees that each of these active inputs can be switched to a unique output.

The next set of blocks are q^2 x $\frac{q^2}{2d}$ sub-networks denoted by $N2$. $N2$ changes the inputs into unique wavelengths and bundles all inputs together into a star coupler. One receiver resides at each output of $N2$, and all the receivers are attached to the star. The basic function of $N2$ is to allow each of the receivers to listen to one of the q^2 inputs. Note that q^2 wavelengths are required for $N2$, and that the wavelengths used for one $N2$ can be re-used by another $N2$. Therefore, a total of q^2 wavelengths are required for all the $N2$ blocks.

Let sub-network $N3$ be the cascade of an $N1$ and $N2$ block. $N3$ is a non-blocking N x $\frac{q^2}{2d}$ switch as long as the number of active transmitters is at most $\frac{q^2}{2d}$. This is because property (P3) ensures that all the active transmitters can be switched to a unique input to $N2$. The receivers then select the desired transmitter.

To complete the full N x N network, put enough $N3$'s in parallel to cover all N possible receivers. Since each $N3$ connects to only $\frac{q^2}{2d}$ receivers, a total of $\left\lceil N/\left(\frac{q^2}{2d}\right)\right\rceil$ $N3$'s are required. Each transmitter is connected to each $N3$ exactly once. The

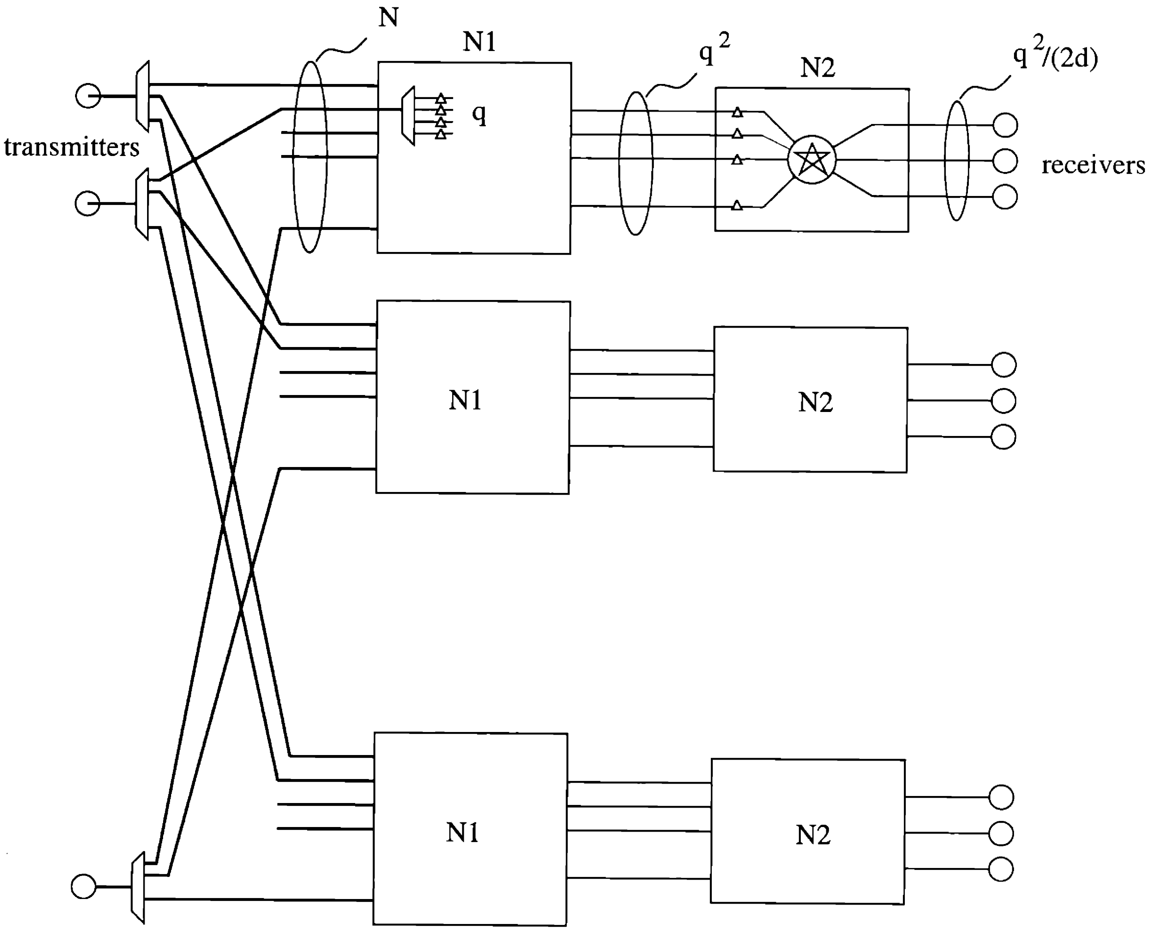


Figure 8-4: A non-blocking network adapted from a 2 stage switching network. There are $N/\left(\frac{q^2}{2d}\right)$ $N1$'s and $N2$'s. The transmitter uses $N/\left(\frac{q^2}{2d}\right)$ wavelengths, and the receiver uses q^2 wavelengths.

fan out emanating from the transmitter is accomplished by the first stage of the demultiplexer depicted in figure 8-3.

The N x N network operates as follows: By tuning to the appropriate wavelength, each transmitter selects a connection to exactly one output of one particular $N1$. Each receiver also tunes and listens to exactly one output of $N1$. To show that this is a non-blocking network, we will show that for any permutation routing problem, every session can be assigned.

We have already argued that $N3$, the cascade of $N1$ and $N2$, is non-blocking as long as the number of active inputs is $\leq \frac{q^2}{2d}$. Therefore, all we need to show is that at most $\frac{q^2}{2d}$ transmitters are active on each $N3$ for any permutation. This is true because each $N3$ is connected to $\frac{q^2}{2d}$ receivers. Therefore, the network is non-blocking.

Now we will calculate the number of wavelengths required for this network. Transmitter tuning selects the appropriate $N1$ as well as the appropriate output of $N1$. There are a total of $\left\lceil N/\left(\frac{q^2}{2d}\right)\right\rceil$ $N1$'s and each input to $N1$ is connected to q outputs. Therefore, the number of wavelengths required by the transmitter is

$$F_t = q\left\lceil \frac{N}{\frac{q^2}{2d}} \right\rceil, \tag{8.3}$$

N is restricted to be $\leq q^{d+1}$. We pick the smallest valid q so that $N \approx q^{d+1}$. Therefore,

$$F_t = 2dN^{\frac{d}{d+1}}\left(1 + o(1)\right). \tag{8.4}$$

Where $o(1)$ approaches 0 as N approaches infinity. The last equation comes from the fact that $N = q^{d+1}$.

On the receiver side, each receiver tunes to one of the q^2 inputs of $N2$. Therefore, the number of frequencies required by the receiver is

$$F_r = q^2 = N^{\frac{2}{d+1}}. \tag{8.5}$$

The number of frequencies required overall is max(F_t,F_r). This value is minimized when $F_t = F_r$. This implies $d = 2$. Therefore, the number of frequencies required, using this particular construction, is $4N^{2/3}\,(1+o(1))$.

8.3 A network construction utilizing broadcast

The second construction described here utilizes the broadcasting capability of optical networks. Even though this construction is fundamentally different from the one described in the previous section, the number of wavelengths required is of the same order. The construction use the notion that if sessions are broadcasted then less switching is needed to connect the sessions. However, broadcast will cause contention at the receivers. The goal here is that contentions will only appear at don't-care wavelengths at each of the receivers.

Like the first construction, the network is formed from a collection of smaller sub-networks, as indicated in figure 8-5. Call the blocks on the left $N4$. Each transmitter is connected to each N4 exactly once, using the configuration in figure 8-2. Therefore, by wavelength tuning, the transmitter selects the $N4$ to which it transmits. Note that all inputs to a single N4 use the same frequency.

$N4$ is a hard wired network with N inputs and q^2 outputs. Like $N1$, the inputs of $N4$ are identified with the rows of the connection matrix C, and the outputs are identified with the columns. However, unlike $N1$, where the transmitter tunes to one of the q connecting outputs, each input of $N4$ broadcasts to all the q connecting outputs. The fan out is accomplished using a star coupler, (see figure 8-5. $N4$ is not capable of switching. Therefore, an active input will be heard simultaneously on q connecting outputs.

Let l be the number of active inputs to an $N4$. Each of these active inputs will broadcast its data session to q outputs. Some outputs will receive one session, others more than one sessions, and the rest will be idle. The outputs that receive more than one session are useless because the data from different sessions clash with one another. The outputs that receive only one session are called *contention free* outputs. The goal is to make sure that every active input is transmitting to at least one contention free output.

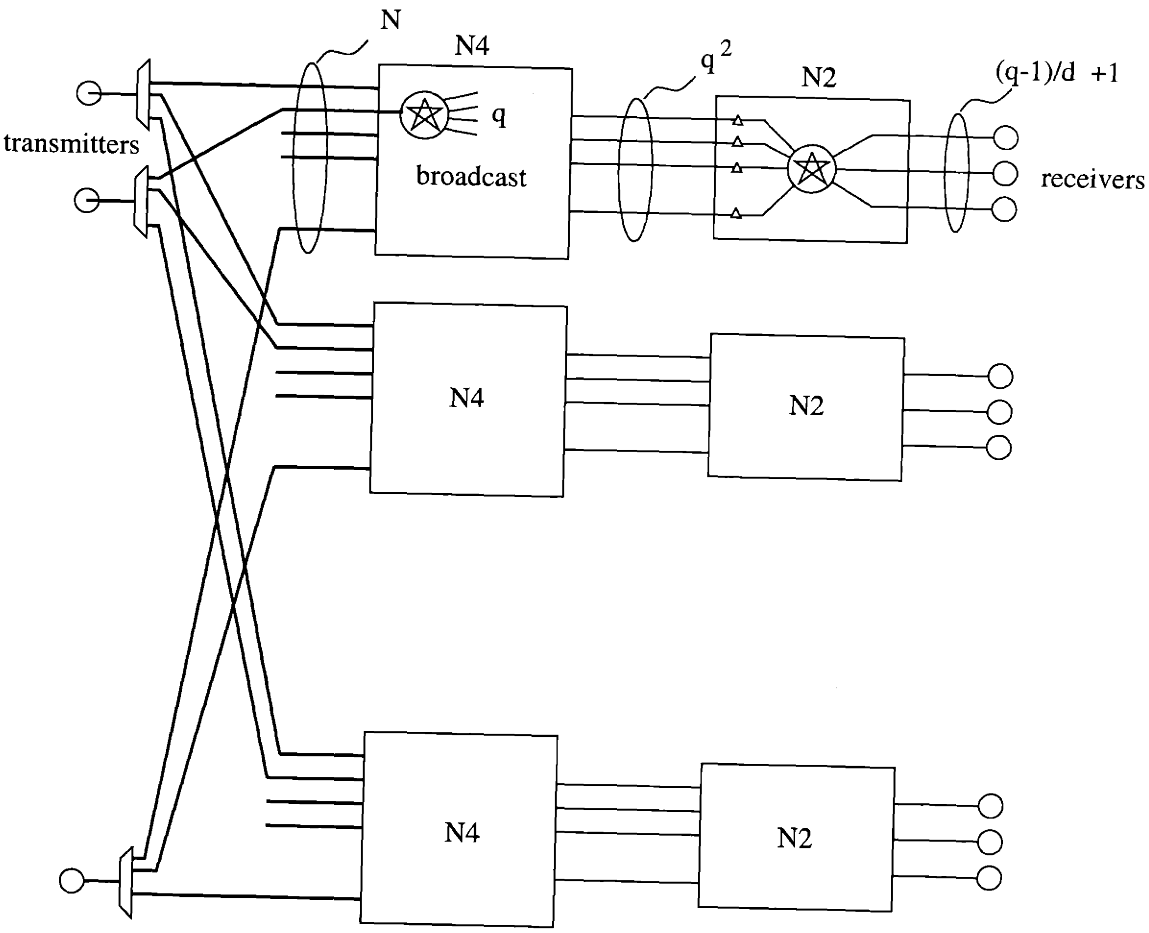


Figure 8-5: Non-blocking network using broadcast.

If the number of active inputs to each $N4$ is $l \leq \frac{q-1}{d} + 1$, and $N = q^{d+1}$, then property (P3) guarantees that for every active input i, there exist a contention free output connected to i. That particular output can only hear i and no other active input. Therefore, $N4$ acts as a concentrator, able to concentrate any l active inputs from the N different possible source locations to the q^2 outputs.

To ensure that no more than $\frac{q-1}{d} + 1$ inputs are active on each $N4$, allow each $N4$ to connect to at most $\frac{q-1}{d} + 1$ receivers. This way, no more than $\frac{q-1}{d} + 1$ transmitters will choose one particular $N4$. This connection is done using $N2$'s. $N2$ is described in the previous section. However, the number of outputs this time is $\frac{q-1}{d} + 1$ see figure 8-5. Each receiver is connected to the star. The routing is completed by having each receiver tune to the desired contention free $N4$ output.

Lastly, if each $N4$ is connected to $\frac{q-1}{d} + 1$ receivers, then $\left\lceil \frac{N}{(q-1)/d + 1} \right\rceil$ $N4$'s are needed to complete the network. The network is non-blocking because each $N4$ will contain $\frac{q-1}{d}+1$ active inputs and all these active inputs will be routed to a contention free output.

The number of frequencies required on the transmitter side is

$$F_t = \left\lceil \frac{N}{(q-1)/d+1} \right\rceil . \tag{8.6}$$

Therefore,

$$F_t = \frac{Nd}{q}\left(1+o(1)\right). \tag{8.7}$$

Remember that the transmitter only needs to select the $N4$ to use, once the data reaches the input to an $N4$, it is broadcasted to q different $N4$ outputs.

The number of frequencies required on the receiver side is

$$F_r = q^2 = N^{\frac{2}{d+1}}. \tag{8.8}$$

This is because each receiver must choose among the q^2 possible $N4$ outputs.

Once again, the overall network requires $\max(F_t,F_r)$, which is minimized for $d=2$. Therefore,

$$F = 2N^{2/3}\left(1+o(1)\right). \tag{8.9}$$

8.4 A construction without wavelength changers

The above two constructions require a large number of static wavelength changers. Specifically, the first construction requires $5N^{\frac{5}{3}}$ static wavelength changers, and the second construction requires $4N^{\frac{5}{3}}$ static wavelength changers. By using clever wavelength allocation schemes, one can reduce the number of static wavelength changers by half. However, we will not dwell on the description of these schemes. Instead, in this section, we will describe another construction that does not use wavelength changers at all. This construction is illustrated in figure 8-6.

Let k and b be integers. This construction will require $F=kb$ wavelengths. First,

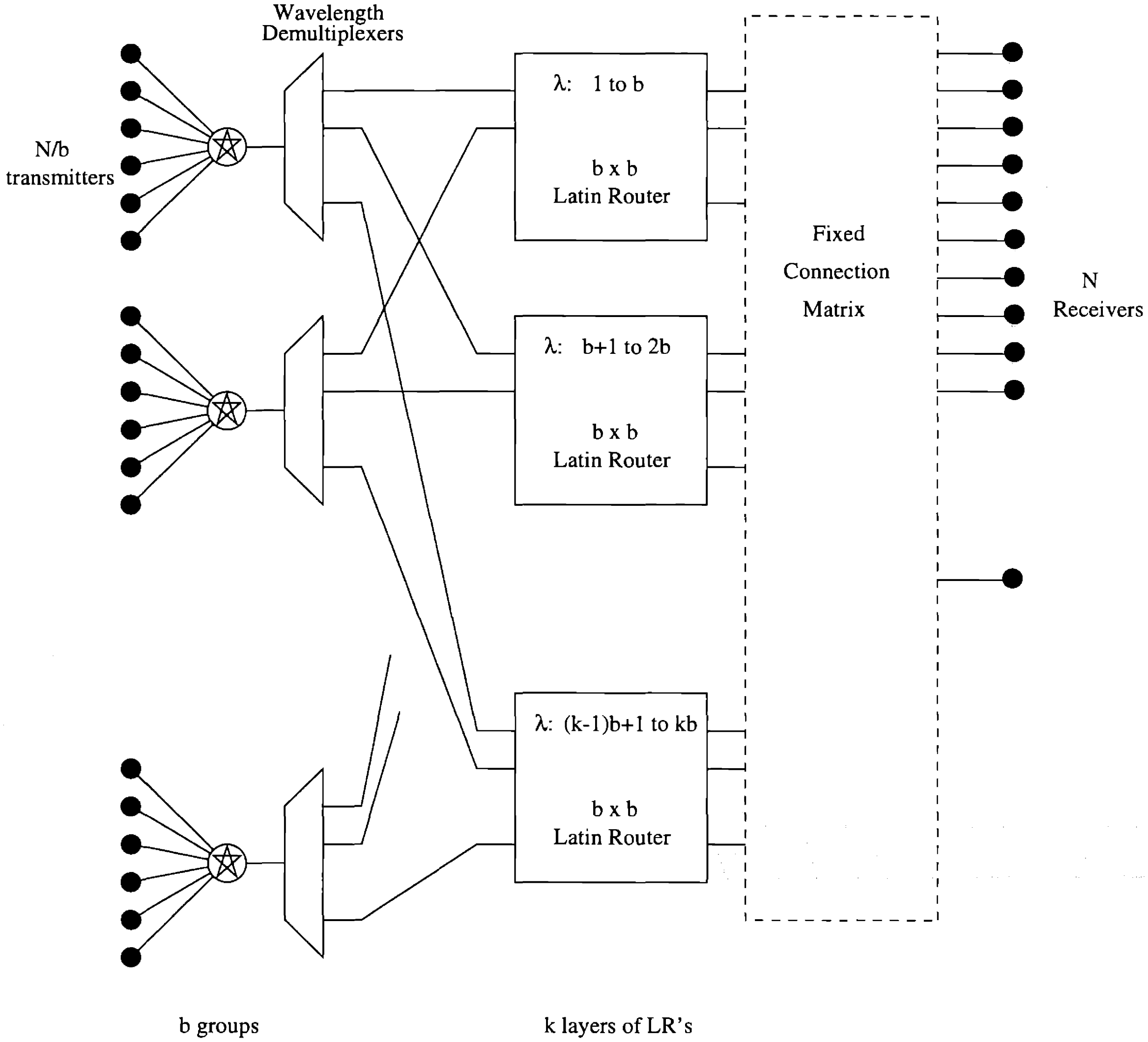


Figure 8-6: Non-blocking network without wavelength changers. For clarity not all connections are drawn.

group the transmitters into b clusters, each of size no more than $\lceil N/b \rceil$. Combine all the transmitters of the same cluster by a star coupler. Take one output of the star coupler, and demultiplex the output into k different wavelength groups, each containing b wavelengths. Let wavelength 1 through b be in the first group, $b+1$ through $2b$ be in the second group, etc. Each wavelength will belong to one and only one group and there are exactly kb wavelengths. As it turns out, we will choose $k = b = q$. However, we leave k and b as free parameters to be optimized later.

Take k Latin routers (LR's) of coarseness 1 and size b x b. LR's are described in section 3.1. Connect the outputs of the wavelength demultiplexers to the LR's such that all the outputs of the same wavelength group are connected to the same LR. Therefore, the first LR operates on the first group of wavelengths, the second LR operates on the second group of wavelengths, etc. Since the inputs to a given LR are restricted to operate only within a group of b wavelengths, the LR's can be viewed as having coarseness 1.

One of the most useful properties of the LR's is that no contention can occur within the LR. Therefore, all the contention that occurs must occur before the input of the LR, i.e., in the star coupler. This means contention occurs only between transmitters of the same cluster, and each cluster of transmitters can tune independently of other clusters.

The only part of the network yet to be specified is the exact connection between the receivers and the outputs of the LR's. Here, we want to connect fibers from the outputs of the LR's to the inputs of the receivers in a specific way. These fiber connections are not frequency selective. Only two properties are needed: 1) Each receiver is connected to each LR at most once. 2) Any group of $\lceil N/b \rceil$ receivers has a system of distinct representatives (SDR) with respect to the outputs from all the LR's. Before describing the connection that has the above properties, we will show why the properties are sufficient to build a non-blocking network.

The first property ensures no contention at the receivers. Each receiver is connected to each LR at most once. Since each LR uses a disjoint set of wavelengths, each fiber into the same receiver carries a disjoint set of wavelengths. Therefore, there is no contention at the receiver, and by tuning to the proper wavelength, a receiver picks out the desired session.

The second property ensures that each transmitter of the same cluster will choose a unique wavelength to set up its connection. The $\lceil N/b \rceil$ transmitters in a cluster can choose any $\lceil N/b \rceil$ receivers as destinations. If these receivers have a SDR with respect to the given LR outputs, then each of these connections can be assigned to a unique LR output. A unique LR output implies unique wavelengths because each LR operates as a coarseness 1 LR using different wavelengths. Therefore, the connections can be set up such that no contentions will occur within each transmitter cluster. Since contentions in between transmitters of different clusters cannot happen, all connections can be set up without contention, and by definition, this is a non-blocking network.

Now, we will describe the connection between the receivers and the LR outputs. The connection utilizes the matrix described in section 8.1. Identify the receivers with the rows of the matrix, i.e., label receiver i by $f_i(x)$, a unique d degree polynomial in $GF(q)$. There can at most be q^{d+1} receivers. Label the LR's from 0 to $k-1$, and for each LR, label the outputs from 0 to $b-1$. Connect receiver i to LR x on output y if and only if $f_i(x) = y$. This ensures that each receiver is connected to each LR at most once. To ensure that all the outputs exist, let $b = q$ and restrict k to be $\leq q$. Call the above connection matrix C'.

If $k < q$, C' is a sub-matrix of the matrix described in section 8.1. C' has only qk columns, and each row contains exactly k 1's. However, property (P2) still holds if k is substituted for q in the derivation. Therefore, any set of $\frac{k^2}{2d}$ receivers have a SDR with respect to the columns of the matrix.

The requirement is that $\frac{N}{b}$ receivers has a SDR with respect to the LR outputs. Therefore, we need

$$\frac{N}{b} \leq \frac{k^2}{2d}. \tag{8.10}$$

Letting $N = \frac{q^{d+1}}{2d}$, and remembering that $b = q$, equation 8.10 becomes

$$q^d \leq k^2. \tag{8.11}$$

Since $k \leq q$, the only valid values for d are 1 or 2.

So far, we have developed a matrix, C' such that each receiver is connected to each LR at most once and any group of $\frac{N}{b}$ receivers has a SDR with respect to the LR outputs. This means all connections can be set up without contention in the network, and the network is non-blocking.

The number of wavelengths required for this network is $F = kb$. We choose the smallest k possible, which according to equation 8.11 is $q^{d/2}$. This means,

$$F = q^{\frac{d}{2}+1} = (2dN)^{\frac{d/2+1}{d+1}} \tag{8.12}$$

Since d can be either 1 or 2, we choose 2 to minimize F and get,

$$F = (4N)^{2/3} \tag{8.13}$$

8.5 Summary of the constructions

We have described three different constructions of non-blocking networks using only $O\left(N^{2/3}\right)$ frequencies. The best lower bound known so far is $O(\sqrt{N})$ [Bar 93]. The existence of $O\left(\sqrt{Nlog(N)}\right)$ has also been shown [Bar 93], but no known construction exists.

The above networks belong to the class of rearrangeably non-blocking networks. This is because the rearrangement of other existing outputs is required during a new session set up.

Chapter 9

Conclusions

9.1 Summary of results

Present day applications are usually too small to fill up one wavelength. Therefore, the most effective way to use an all optical network (AON) is to first aggregate the end users together using local electronic networks, and then connect the local networks together using high bandwidth wide area AON's. The local networks access the AON via interfaces called ports. The architectural design of the AON depends on the port to port traffic.

Because the port to port traffic is the result of aggregation of many small users, it is reasonable to assume it to be uniform all to all (UATA). Chapter 3 assumes this case, and designs the AON backbone accordingly. The design takes advantage of traffic symmetry, and uses recently developed wavelength routing devices called the Latin Routers (LR's) to set up fixed bit-pipes from port to port. The resulting AON is efficient because all LR's are used to capacity, and the AON is scalable because new ports and connections can be added without tearing down existing connections.

The UATA traffic models the port to port traffic under the assumption of large aggregation. To analyze the performance of the network built under this assumption, we need a traffic description from the end user's point of view. Chapter 4 develops this end user traffic description and measures the performance of the network by calculating the bit-pipe congestion probability. Various network parameters can be traded off to obtain the same performance measure. Therefore, we defined a cost function relating the cost of the network to the various network parameters. A minimum cost network can be found using the performance measure as a constraint.

In chapter 5 we augmented the UATA network with a shared resource so sessions that overflow the dedicated bit-pipe in the UATA portion may attempt to use the shared resource bit-pipes (SRB's). We analyzed the congestion probability of the SRB's. The analysis gives the optimal SRB size for the smallest congestion probability. In the case when overflow events are rare, the shared resource becomes one giant star allowing the maximum possible sharing.

Another port to port traffic model we studied (in chapter 6) is the fixed multiple connection (FMC) model. Sometimes, due to geographical constraints, uniformity cannot be achieved although large aggregation is still possible. Therefore, the amount of traffic between one port pair could be very different from the traffic between another port pair. Because the port to port traffic is fixed, the backbone required again consists of static bit-pipes. Unfortunately, without any symmetry in the traffic, the best way to build the backbone is simply to lay down fiber wherever needed. We also investigated the possibility of supporting FMC traffic using an existing infrastructure as the backbone. The feasibility of fitting a FMC traffic into an existing infrastructure is NP-complete.

The last type of port to port traffic we considered is the variable multiple connection (VMC) traffic. In this scenario, the port to port traffic varies slowly with time. These variations are due to factors such as the time of day. The VMC traffic can be supported using wavelength specific switches (WSSS's). Each wavelength layer of a WSSS is independent, and the network supports many layers of permutation routing.

As bandwidth become more available and affordable, large rate users will start to emerge. Therefore, AON's must also directly support these large users. In chapter 8 we constructed point to point non-blocking networks that support these large users. We described three constructions, of which two are new. The number of wavelengths required for the two constructions is $O(N^{2/3})$ where N is the number of these large users. There are other constructions in the literatures that have better asymptotic bounds, but our construction is better for all practical values of N, i.e., for $N \leq 2^{7765}$. One of the new constructions does not require wavelength changers. This is significant because currently, optical wavelength changers are expensive.

9.2 Future research

In this thesis, we developed AON backbones for hierarchical networks to support low rate users, and developed point to point non-blocking AON networks to support high rate users. These two types of network can coexist on the same physical structure to simultaneously transport both types of traffic. A possible area of future research would be to see how the two types of network interact with each other, i.e., how to design and analyze sharing schemes so the two networks can efficiently utilize common backbone resources. For example, point to point circuits not utilized by the high rate users can be used by aggregated sessions that overflow their assigned bit-pipe.

Also, consider mid rate users. There are two options of supporting mid rate users. One is to split them up to smaller pieces and consider them as small rate users. However, this leads to burstiness in the aggregated traffic. The other option is to consider them as high rate users, in which case, the network resource will be wasted because they are not large enough to fill up a wavelength channel. An interesting problem is to define the trade off between the two options. Perhaps a new network architecture can be designed to span the gap between the backbone network and the point to point network.

Appendix A

Slot Alignments

Although the use of TDM in frequency channels can provide more flexibility in a WDM network, one major problem of TDM is the misalignment of time slots due to propagation delay. The effective length of a fiber varies with temperature and wavelength. This phenomenon is more acute in wide area networks than local area networks due to the long distance fibers. This problem could be solved by actively changing the pathlength from node to node so the time slots are aligned.

In this appendix, we investigate the problem of time alignment in networks. We are not interested in the actual physical implementation of time alignment, but rather, the connectivity issue. Therefore, we assume the network is able to perform time alignment, either through variable delay lines or some form of optical buffer.

We define three types of slot alignment and investigate their relative merits. A graph models the network. A node denotes a place where merging or splitting of edges occurs. An edge represents a fiber connecting neighboring nodes. In TDM, each wavelength is divided into T time slots labelled from 1 to T. Slot 1 is the beginning of a *frame*, and each frame has T consecutive time slots. The frames follow one another as the data traverse through the network.

If there are no wavelength changers in the network, then the network can be viewed in layers. Each layer represents one wavelength, and the layers do not interact. In this case, slot alignment can be done independently on each layer. In this chapter, we will investigate the merit of various slot alignment schemes for a single layer.

A.1 Definitions

In this sections, we define three types of slot alignment.

- *Weak Sense Slot Alignment*: Two nodes connected by a link are weak sense slot aligned if data at the beginning of a slot in one node, after propagating through the link, appears at the beginning of some slot on the other node. The slot number where the data appears need not be the same for both nodes. In fact, the actual pairing may change from time to time. For example, if slot i at node a is aligned to slot j at node b, then at a later time, slot i at a may be aligned to some other slot k at b where $k \neq j$. When a change in alignment happens, we call it slot *slipping*.

- *Strict Sense Slot Alignment*: This is the same as weak sense slot alignment except that slipping is not allowed.

- *Frame Alignment*: In this type of alignment, slots with the same numbers are aligned together. Consequently, the frames are also aligned.

A.2 Complexity of slot alignments

To achieve slot alignment, the network must continuously monitor the pathlength difference between adjacent nodes. The network must also have the ability to actively change the effective pathlength between the two nodes by δd, where δd is half a slot length for weak sense alignment, and half a frame length for strict sense and frame alignment. In any case, δd does not need to be larger than the largest differential variation of the fiber length.

The system must keep a table that maps the slots from node to node. If the alignment is weak sense, this table needs to be updated whenever slipping occurs. An update of the table implies reconfiguration of the network state, and therefore, possible re-routing of existing sessions. This increases the control complexity.

A network has a cycle if the nodes and the links in the network form a cycle, where the links are viewed as undirected links. For example, networks in figure A-1, figure A-2 and figure A-3 all have cycles. Strict sense alignment is equivalent to frame alignment only for networks without cycles. Here equivalent means that one can renumber the slots in each node such that the network become frame aligned. We show this by describing a renumbering process that changes strict sense alignment to frame alignment: First, pick any node as the reference node and renumber the slots of all the neighbors of the reference node so the frames are aligned. Next, renumber the slots of all neighbors of those nodes that have been frame aligned. Continue the process of renumbering the slots for all successive neighbors until all nodes are frame aligned. The renumbering will be consistent (i.e., no node is required to have two different renumbering schemes) because there are no cycles in the network.

Renumbering may not be consistent for networks with cycles. Figure A-1 shows a network with a three node cycle. Each of the edges in the cycle has a pathlength of $T + 1$ slots. No consistent renumbering scheme can be used unless one of the pathlength is changed. Therefore, strict sense alignment is not always equivalent to frame alignment. However, one can always change a strict sense aligned network to a frame aligned network by adding extra delays in the links.

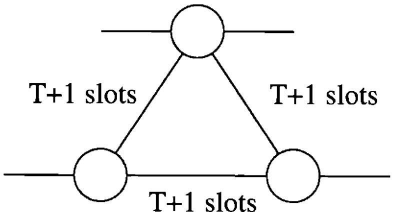


Figure A-1: Frame alignment is not possible with this network if all edges have length $T + 1$. There is no consistent renumbering scheme.

In terms of connectivity, weak sense slot alignment suffers the most because of slot slipping. For a connection covering many nodes, the slot occupied on each node can change over time. Therefore, slot conflicts can develop due to slipping even when no new session is set up. For ease of setting up sessions and avoiding new slot conflicts during a transmission, one generally desires either strict sense slot alignment or frame alignment. Because strict sense alignment is not generally equivalent to frame alignment, the next section demonstrates the difference in connectivity.

A.3 Strict sense versus frame alignment

In this section, two example networks will be given such that one will benefit from strict sense alignment and the other from frame alignment. These two example shows that neither of the two alignment schemes is superior to the other. The alignment strategy that best benefits a network depends on the topology of the network and the traffic requirements. In both examples, the number of slots, T, is two. The nodes represent time multiplexed space switches. Capital letters, A, B, C, D, denote transmitters and lower case letters, a, b, c, d, denote receivers.

Figure A-2 illustrates a case where frame alignment is needed. In this example, all pathlengths are multiples of a frame (i.e., frame aligned) except for edge (x,y), for which the pathlength is 1 slot off. Slot 1 at node x will appear as slot 2 at node y. We want to satisfy the following four sessions: (A,a), (B,c), (C,b), and (D,d). Without loss of generality, we can choose slot 1 for session (A,a). This implies session (B,c) must use slot 2. When session (B,c) arrives at node y, it must use slot 1 because of the delay in edge (x,y). Session (C,b) must use slot 2 to avoid contention with session (A,a). Session (D,d) cannot be set up because the only slot available before node y is slot 1, the only slot available after node y is slot 2, and node y cannot perform slot interchange.

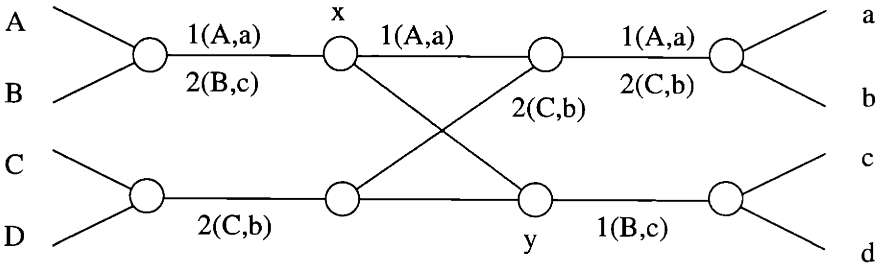


Figure A-2: An example where contention is the result of frames not being aligned. Session (D,d) is not allowed because there is no time slot interchange.

Note that session (D,d) is disallowed because of slot mismatch, and not because of capacity constraints. In fact, this problem would not have occurred if the frames were aligned. We leave it for the reader to verify that this network supports permutation routing if the frames are aligned.

This, however, does not mean frame alignment is superior. Figure A-3 illustrates an example where frame alignment causes contention. Assume we have a frame aligned 3 x 3 network with the topology shown in figure A-3. We want to satisfy sessions (A, a), (B, b), and (C, c). Without loss of generality, choose slot 1 for session (A, a). This implies session (B, b) must use slot 2. This means session (C, c) cannot be set up because it contends with both existing sessions. However, this problem can be avoided by purposefully misaligning node x and y by exactly one slot. (This can be done by adding an extra one slot delay in edge (x, y)). The reader can verify that by adding the extra delay, this network supports permutation routing.

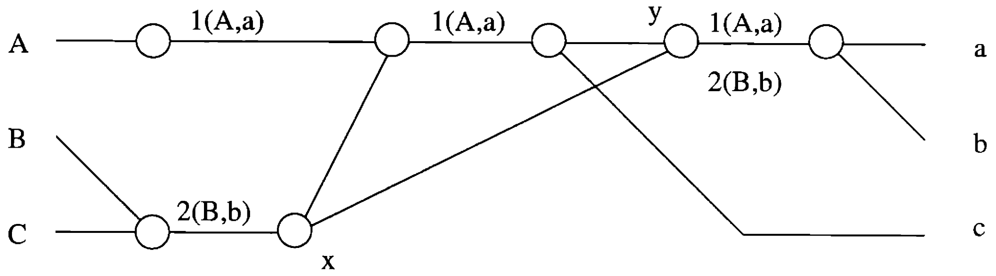


Figure A-3: An example where contention is the result of frame alignment. Session (C,c) is not allowed because there is no time slot interchange.

We have shown that strict sense alignment is better for the network in figure A-3. On the other hand, frame alignment is better for the network in figure A-2. Therefore, which is better depends on the topology of the network.

A.4 Slot alignment conclusions

In a general network using TDM, time slot alignment can help to increase connectivity of the network. Of the three types of alignment defined, weak sense slot alignment is the easiest to achieve, but worst in terms of connectivity because of slot slipping. Strict sense slot alignment is not always equivalent to frame alignment, although they require the same type of hardware to implement. Depending on the actual network topology, one type of alignment could be better than the other type.

Appendix B

Merit of Slot Re-Use in Multiple Bus Networks

We have argued that TDM is difficult in long distance AON's because temperature variation causes fiber length changes. The problem can be solved by actively changing the pathlengths of the fiber links so slots between neighboring nodes are aligned. However, this is difficult to do in practice. On the other hand, the problem can be avoided if the transmission layer consists of data buses that do not interact with each other. Two buses do not interact with each other if they are not connected to one another except through the connecting users. In other words, data cannot travel directly from one bus to another. Therefore, the buses do not interchange slots with each other, and each bus can have its own timing reference. Figure B-1 shows a network with non-interacting buses.

This appendix considers the merit of slot re-use in a multiple bus TDM network. We have *stations* that desire to transmit data to one another, and they do so using TDM connections on the buses. (These stations can be either end users, or in the context of this thesis, AON ports.) Each station is connected to multiple buses via add/drop filters. Each add/drop filter is used by a single station to receive and transmit data. (Note that this set up is slightly different than those described in section 3.4.5. There, a t-port and its corresponding r-port need not connect to a given bus at the same physical locations. Here, both are connected using the same

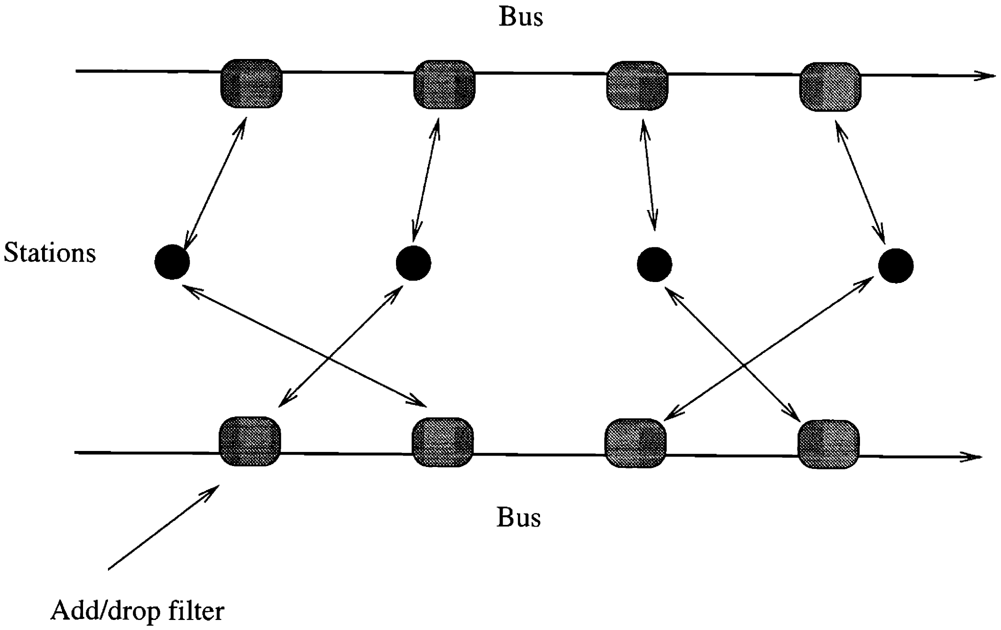


Figure B-1: Network consist of non-interacting buses. The buses do not interchange slots with each other.

add/drop filter.) If a slot is received by a station, then that slot can be re-used by another station further down the bus.

The architecture attempts to optimize slot re-use by strategically placing the stations in such a way as to minimize the expected distance between two randomly chosen stations. Here, distance is measured in number of links. A link is the part of the bus between two adjacent add/drop filters. The merit of slot re-use is analyzed by comparing the link load of the above strategy with one that does not incorporate slot re-use.

B.1 Model, architecture of the bus network

We have N stations that desire to send TDM traffic to each other. Each station is equipped with transmitters and receivers. The transmitters and the receivers work independently so coordination between them is not required. We also have directional buses that support TDM traffic. Each station may connect to a multiple number of buses. However, we assume that 1) each station has at most one connection to each bus, and 2) each station has at most k connections. The first restriction will not affect the lower bound derived later, and serves as a more fair comparison between our strategy and one that does not use time slot re-use. (Without time slot re-use, having multiple connections on the same bus is useless.) The motivation behind the second assumption is that each station has, or can simulate, at most k transmitter/receivers, where each transmitter/receiver is operating on an independent time reference according to the attached bus.

The transmitting station transmits the data onto an empty slot on a chosen bus via an add/drop filter. The add/drop filter is perfect in the sense that when the data slot reaches the destination, the data is dropped to the receiving station and the slot can be used again further down the bus. Each add/drop filter can accommodate exactly 1 station connection, and therefore, each connection requires a unique add/drop filter.

Each station has to keep track of its own timing reference to each connecting bus. We assume this is possible. Also, slot conflicts can occur within the stations unless each station has enough resources to simultaneously listen and transmit to each bus. We assume this is the case and ignore possible conflicts that occur within the station because we want to focus the analysis on the advantage of slot re-use.

We assume uniform traffic. Station i is as likely to communicate with station j as with station l provided $j \neq i$ and $l \neq i$. We do not allow a station to communicate with itself. This is because self communicating traffic does not enter the bus, and therefore, can be disregarded when analyzing this architecture.

A good architecture has low congestion on the links. Since data can only be added or removed from the bus at the add/drop filters, the length of a link is inconsequential. The amount of congestion on a link is related to the total link usage by the network. Let $d(i, j)$ be the distance in number of links between station i and station j. Then, at a given time, the total link usage is $D = \sum d(i, j)$, where the summation is over all active sessions. Let Γ be the total number of links in the system. Then the most congested link carries at least D/Γ simultaneous sessions. Therefore, a desirable parameter for network construction is to reduce the average distance of a session. Define $E[d]$, to be the expected distance between station i and station j, where i and

j are chosen uniformly over all stations such that $i \neq j$.

To communicate with station j, station i will always choose the bus such that $d(i,j)$ is minimized. If i and j do not reside on a common bus, then $d(i,j) = \infty$. In calculating $E[d]$ the smallest $d(i,j)$ is used for each pair (i,j) if there is more than one possible path. The goal is to assign the stations to the buses such that $E[d]$ is minimized

B.2 Lower bound on average distance

For any given station at most 1 other station on the same bus is exactly l links away. This is true for all $l \leq 1$. Consequently, given any k buses, at most k stations are 1 link away from a given station, at most k stations are 2 links away, and at most k stations are l links away. This implies,

$$E[d] \geq \frac{k + 2k + 3k + \ldots + \left\lfloor \frac{N-1}{k} \right\rfloor k + \left\lceil \frac{N-1}{k} \right\rceil \left(N - 1 - k \left\lfloor \frac{N-1}{k} \right\rfloor\right)}{N-1} \tag{B.1}$$

If we assume $\frac{N-1}{k}$ to be an integer, then the above simplifies to,

$$E[d] \geq \frac{N+k-1}{2k} \tag{B.2}$$

A trivial necessary condition to meet the lower bound is that each station must be connected to the network exactly k times, and that at least k buses are used.

In order to meet the above lower bound, each station must have exactly k stations that are 1 link away, k stations that are 2 links away, etc. However, this is not possible because there must be some stations that are attached to the end of a bus. This difficulty is circumvented if the buses can be wrapped around, i.e., if we use rings instead of buses. A ring can simulate a bus and provides more possibilities. Hence, we will investigate the merit of slot re-use on a network consisting of rings, and note that we cannot hope to do better for networks consisting of buses. Note that equation B.1 and equation B.2 hold for both ring and bus networks.

B.3 Relationship between $E[d]$ and other network costs

There is a trade off between $E[d]$ and the cost of the network. The significant costs of the network are the number of rings (or buses), the number of add/drop multiplexers, and the complexity of each station. The total number of rings is at least k and the total number of add/drop filters is Nk. Each station must simulate k transmitters/receivers, each operating on an independent time reference. The station also needs to choose the appropriate transmitter for each active session. Therefore, the cost of the network is directly proportional to k. One can decrease $E[d]$ at the expense of increasing k, i.e., a more costly network. On the other hand, one can sacrifice $E[d]$ in order to have a less expensive network.

B.4 The merit of time slot re-use

The lower bound derived in B.2 provides a limit on the usefulness of time slot re-use. The lower bound dictates that the best network, if it exists, satisfies the bound using at least k buses. The network must be uniform so that there is no bottleneck under uniform traffic. (Section B.5 describes the construction of these networks.) If a network uses exactly k buses and satisfies the lower bound, we call this network *optimal.* Section B.5 describes the construction of optimal networks. In this section, we assume that optimal networks exist, and compare them with a networks that do not utilize slot re-use. Call these types of networks *simple* networks.

Let S_{tot} be the total number of active sessions for a given uniformly distributed traffic matrix. Then the average total link use is $S_{tot}E[d]$. There are at most Nk links. Therefore, the average loading of a link for the network is $\geq \frac{S_{tot}E[d]}{Nk}$. By definition, $E[d] = \frac{N+k-1}{2k}$ for an optimal network. Therefore, the average link load for an optimal network is,

$$\mathrm{L}_{opt} = \frac{S_{tot}(N+k-1)}{2Nk^2}. \tag{B.3}$$

Note that L_{opt} is the same even if the optimal network uses more than k buses. This is because the total number of links do not change.

In the simple network, once a slot is used on a bus, that slot may not be used again. Therefore, the parameter of interest for the simple network is the average loading on a bus. (Note that in this case, the ordering of the stations on a bus is inconsequential). If there are b buses, then the average loading per bus is

$$L_{sim} = \frac{S_{tot}}{b}. \tag{B.4}$$

There are two ways of making the comparison. One way is to equate the number of buses and the other is to equate the number of bus connections per station, (i.e., keep the degree of the station node the same). In both methods, we want to compare the average loading per link in the optimal network to the average loading per bus in the simple network. If loading is uniform, then a smaller load implies a higher possible throughput.

First, we will compare by equating the number of buses. In other words $b = k$. Therefore,

$$L_{sim} = \frac{S_{tot}}{k}. \tag{B.5}$$

For large N and constant k, the optimal network is, at best, a factor of $2k$ less loaded. In other words, slot re-use can improve the throughput at most $2k$ times.

We now compare the two networks by equating the number of bus connections per station. This is a more fair comparison if the bus cost is small in comparison to the station complexity cost. The degree of every station node is k. Therefore, L_{opt} is still the same as equation B.3. Since L_{sim} is inversely proportional to b, we want to use the maximum number of buses possible in order to have small loading. Therefore, we first calculate the maximum b subject to the constraint that the network still have full connectivity, and try to construct a fully connected simple network using as many buses as possible. We then compare L_{sim} of the construction with L_{opt}.

The number of buses a station can reach is k. And each station must reach all other stations. Therefore, the average number of stations connected per bus must be

$\geq \frac{N-1}{k} + 1$. (For simplicity, we ignore all integer constraints.) The average number of station per bus is also equal to $\frac{Nk}{b}$. Therefore,

$$b \leq \frac{N}{N-1+k} k^2. \tag{B.6}$$

We now construct a uniform simple network using $\frac{k(k+1)}{2}$ rings. Group the stations into $k+1$ groups. Each ring is connected to two groups of stations. There are $\binom{k+1}{2}$ ways of choosing the two connecting groups. In total, we have $\frac{k(k+1)}{2}$ rings. Each group is connected to exactly k rings because there are only k other distinct groups to be coupled with. Also, all stations are connected to all other stations, because all group pairs can be found on a ring. Therefore, we have succeeded in making a totally connected network using $\frac{k(k+1)}{2}$ rings.

The loading of the simple network just constructed is

$$L_{sim} = \frac{2S_{tot}}{k(k+1)} \tag{B.7}$$

Remembering that L_{opt} is a lower bound, this means that slot re-use provides at most a factor of 4 less congestion.

In section 3.4.5 we performed a similar calculation and found that slot re-use provides at most a factor of 2 less congestion. The discrepancy lies within the fact that a station here sends and receives data using the same add/drop filter on a given ring, while in section 3.4.5, a t-port and its corresponding r-port connections on a given bus need not necessarily be the same.

B.5 Constructions meeting the lower bound

As noted previously, the lower bound can only be met if the directional buses are replaced by directional rings. Constructions that meet the lower bound have not been found for all N, and k. This section describes the construction for some values of N and k.

Even if a construction meets the lower bound, the advantage of slot re-use can be greatly compromised if the construction has bottlenecks in its topology. (Note that this is possible because even if the average link load meets the lower bound, the actual link load for some links can be much higher than the average.) Therefore, we look for uniform constructions that satisfy the lower bound.

For all the constructions, assume $(N-1)/k$ to be an integer so that the bound in equation B.2 can be used.

Define the array, $R_i = \{a, b, c, ..., x\}$, as the station assignment on ring i such that station b follows station a, station c follows station b, and so forth to station x. Station x is followed by station a because of the wrap around property of the ring. The ring is directional and one must traverse the ring from left to right. For example, station a is the furthest station away from station b.

B.5.1 Construction with ≤ 2 rings

For $k = 1$, the construction is trivial. Take one ring and connect all the stations to the ring exactly once in any order. The bound is trivially satisfied.

For $k = 2$, label the stations from 0 to $N-1$. The following construction meets the bound using only two rings:

$$R_1 = \{0, 1, 2, \ldots, N-1\}$$
$$R_2 = \{N-1, \ldots, 2, 1, 0\}$$

To check that this indeed meets the bound, observe that for each station, there are exactly 2 stations that are 1 link away, 2 other distinct stations that are 2 links away, etc.

B.5.2 Construction with $E[d] = 1$

If $k = N-1$, then equation B.2 states that $E[d] = 1$. This bound can be met for all N except for $N = 4$ and $N = 6$.

Construction for prime N

First, the construction for prime N will be described. The construction is

$$R_i = \{0, i, 2i, \ldots\},$$

for all $1 \leq i \leq N - 1$.

For example, the construction for $N = 11$ and $k = 10$ is:

$$\begin{aligned}
R_1 &= \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \\
R_2 &= \{0, 2, 4, 6, 8, 10, 1, 3, 5, 7, 9\} \\
R_3 &= \{0, 3, 6, 9, 1, 4, 7, 10, 2, 5, 8\} \\
R_4 &= \{0, 4, 8, 1, 5, 9, 2, 6, 10, 3, 7\} \\
R_5 &= \{0, 5, 10, 4, 9, 3, 8, 2, 7, 1, 6\} \\
R_6 &= \{0, 6, 1, 7, 2, 8, 3, 9, 4, 10, 5\} \\
R_7 &= \{0, 7, 3, 10, 6, 2, 9, 5, 1, 8, 4\} \\
R_8 &= \{0, 8, 5, 2, 10, 7, 4, 1, 9, 6, 3\} \\
R_9 &= \{0, 9, 7, 5, 3, 1, 10, 8, 6, 4, 2\} \\
R_{10} &= \{0, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1\}
\end{aligned}$$

Note that R_i contains all elements in the integer field mod N and that neighboring elements differ by i. To see why $E[d] = 1$ in this case, first, observe that for $E[d] = 1$ to be true, $d(i, j)$ must equal 1 for all i and j. Take any i and any j. j is 1 link away from i at ring $R_{j-i(ModN)}$.

The above construction does not work for non-prime N's because if $d > 1$ is a divisor (mod N) of N, then R_d will contain only N/d distinct elements in mod N. This violates the necessary condition that each stations must be connected to the network k times.

Construction for odd N

By definition, a *Tuscan-k square* is an n x n array where the symbols, 1,2,...,n, each appear exactly once in each row, and furthermore, for any two symbols a and b, and for each m from 1 to k, there is at most one row in which b is the mth symbol to the right of a. Similarly, a *Circular Tuscan-k square* is an n x $(n+1)$ array where each of the symbols 0,1,2,...,n appears exactly once in each row, and the symbols are arranged such that the Tuscan-k square property holds when the rows are taken to be circular. Both of these definitions are taken from [GET 90]. Clearly, if R_i's are the rows of a Circular Tuscan-1 square, then they implement a network with $N = n+1$ and $E[d] = 1$.

Tuscan-1 squares are known to exist for all n except $n = 3$ and $n = 5$ [Til 80]. A simple construction algorithm is known for even n's [DK 74]. Furthermore, n x $(n+1)$ Circular Tuscan-1 arrays can be constructed from n x n Tuscan-1 squares. Therefore, constructions with $E[d]$ exists for all N except for $N = 4$ and $N = 6$, and explicit constructions are known for all odd N's.

B.5.3 Construction with $E[d] = 1.5$

For certain prime values of N, one can take the construction from section B.5.2, and reduce k in half and still meet the bound. To satisfy the bound, all stations must be 1 or 2 links away from any given station.

Before describing the construction, an example is shown for clarity. For $N = 11$, the construction for $k = 5$ can be derived from the construction for $k = 10$. (Remember that $R_i = \{0, i, 2i, \ldots\}$.) By taking only R_1, R_3, R_4, R_5, and R_9, we have,

$$\begin{aligned}
R_1 &= \{0,1,2,3,4,5,6,7,8,9,10\} \\
R_3 &= \{0,3,6,9,1,4,7,10,2,5,8\} \\
R_4 &= \{0,4,8,1,5,9,2,6,10,3,7\} \\
R_5 &= \{0,5,10,4,9,3,8,2,7,1,6\} \\
R_9 &= \{0,9,7,5,3,1,10,8,6,4,2\}
\end{aligned}$$

One can visually verify that all stations are either 1 or 2 links away from any given station. Therefore, the above example satisfies the bound.

The general construction is as follows: Create a pool of $N-1$ rings, $\{R_i : 1 \leq i \leq N-1\}$, where $R_i = \{0, i, 2i, \ldots\}$. The new network selects $\frac{N-1}{2}$ rings from this pool of R_i's. Take any integer x. Choose for the network all the rings of the form R_{x2^q} for all even integer q (including $q = 0$), and delete from the pool all the rings of the form $R_{x2^{(q+1)}}$. (All the indices should be calculated using mod N arithmetic; this fact is omitted in the notation to avoid clutter.) After the initial selection and deletion, if there are no more R_i's left in the pool, then the selection is done. Otherwise, repeat the selection with a new value of x such that R_x is still in the pool. The selection is guaranteed to be consistent unless $2^{(r+1)} = 1 \bmod N$ for some even integer r. In this case, one is forced to pick both R_x and to delete $R_{x2^{(r+1)}} = R_x$. Therefore, this construction works only for prime N and there must not be an even integer r such that $2^{(r+1)} = 1 \bmod N$.

To see why this construction satisfies the bound, observe first that the selection process chooses $\frac{N-1}{2}$ rings. This is because for each unique R_{x2^q} picked, there is a unique $R_{x2^{(q+1)}}$ deleted. So exactly half of the rings are chosen, and the other half deleted. Secondly, by construction, each station has exactly one connection with each of the selected rings. Lastly, let $X = \{x_1, x_2, \ldots\}$ be the set of x's used during the ring selection process. Take any two nodes i and j. If we can show that j is either 1 or 2 links away from i, then we are done. We know that $j - i = x_i 2^p$ for some $x_i \in X$ and some integer p. If not, there are some rings that are left over in the pool, and we have not finished the selection process yet. If p is even, then j is 1 link away from i on $R_{x_i 2^p}$, which is a selected ring. If p is odd, then $R_{x_i 2^{p-1}}$ is a selected ring, and j is 2 links away from i on that ring. (Remember that in $R_{x2^{p-1}}$, neighboring stations have labels that differ by $x2^{p-1}$, and stations that are two links apart have labels that differ by $x2^{p-1} + x2^{p-1} = x2^p$.)

B.6 Construction with $E[d] = \frac{N}{k+1}$

The construction to meet the bound for other values of N and k becomes increasingly difficult if not impossible. However, one can achieve $E[d] = \frac{N}{k+1}$ with the following construction.

Let p be any prime number > 2. Group the stations into p groups, each of size

N/p. Assume N/p is an integer. (One can always add dummy stations so that N/p is an integer and see that the following analysis is still valid.) Label the groups from g_0 to g_{p-1}. Place these groups on the rings in a similar manner as the stations in section B.5.2. i.e.,

$$R_i = \{g_0, g_i, g_{2i}, \ldots\}.$$

In the above notation, each g_i is a string of N/p stations. The order of the stations within a group is flexible and may be different on each ring. However, it is important that the arrangement is such that any station can communicate with any other station of the same group without traversing more than N/p links, (i.e. outside of the grouping) on some ring. One way to accomplish the above is to arrange the stations within each group in increasing order on one ring and in decreasing order on another.

There are two types of traffic: inter-group and intra-group. The inter-group traffic consists of communications between two stations of different groups, while the intra-group traffic consists of communications between two stations within the same group. The size of the inter-group traffic is larger than the size of the intra-group traffic. Therefore, E[d] will be determined chiefly by the inter-group traffic. To calculate $E[d]$, notice that for any pair of groups, there is a ring such that the second group is adjacent to the first group. Therefore, $E[d]$ for the inter-group traffic is equal to N/p. $E[d] \leq N/p$ for the intra-group traffic. There are $k = p - 1$ rings. Therefore, $E[d] \leq \frac{N}{k+1}$.

B.7 Random construction

For N stations and k rings, $E[(d,j)] = \frac{N-1}{k+1} + 1$ can be achieved using random constructions. On each ring, independently attach all of the stations in random order. Let $d_l(i,j)$ be the distance from station i to j on ring l. Hence,

$$d(i,j) = min\{d_1(i,j), d_2(i,j), \ldots, d_k(i,j)\}. \qquad \text{(B.8)}$$

$d_l(i,j)$ is a discrete valued random variable uniformly distributed between 1 and $N-1$ inclusively. Then, since the d_l's are independent,

$$Prob(d(i,j) \geq x) = (Prob(d_l(i,j) \geq x))^k = \left(\frac{N-x}{N-1}\right)^k, \tag{B.9}$$

and,

$$E[d] = \sum_{x=1}^{N-1} Prob(d(i,j) \geq x) = \sum_{x=1}^{N-1} \left(\frac{N-x}{N-1}\right)^k. \tag{B.10}$$

We can bound this by,

$$\int_1^N \left(\frac{N-x}{N-1}\right)^k dx \leq \sum_{x=1}^{N-1} \left(\frac{N-x}{N-1}\right)^k \leq \int_1^{N-1} \left(\frac{N-x}{N-1}\right)^k dx + \left(\frac{N-1}{N-1}\right)^k. \tag{B.11}$$

Simplifying, we have,

$$\frac{N-1}{k+1} \leq \sum_{x=1}^{N-1} \left(\frac{N-x}{N-1}\right)^k \leq \left(\frac{N-1}{k+1}\right)\left(1 - \left[\frac{1}{N-1}\right]^{k+1}\right) + 1. \tag{B.12}$$

Therefore, $E[d]$ is about N/k, and more precisely,

$$E[d] \leq \left(\frac{N-1}{k+1}\right) + 1. \tag{B.13}$$

B.8 Summary

Slot re-use can improve network throughput for networks consisting of non-interacting buses or rings. The improvement is limited. Specifically, for networks consisting of rings, the improvement is at most a factor of $2k$ if the number of rings is kept constant, and a factor of 4 if the number of connections per station is kept constant.

The improvement is maximized when the network minimizes the number of links a session occupies, i.e., when the lower bound for $E[d]$ is satisfied. Also, the topology needs to be uniform in order to avoid bottlenecks in the network. Uniform constructions, using ring networks that meet the lower bound, exist. These ring networks achieve the maximum improvement. The improvement is projected to be less for networks consisting of buses because buses are more restrictive than rings. (A ring can simulate a bus and is more flexible than a bus.)

Even for ring networks, constructions that satisfy the bound are hard to find. For large N and k, the few constructions described are not physically practical. However, one practical and useful construction is for $k = 2$. This construction contains two rings in reverse direction. The improvement here is $4\frac{N}{N+1}$ if the number of rings is kept at 2. Bi-directional rings are already popular in network designs. This analysis further support their use.

Bibliography

[Ag+ 94] A. Aggarval, et al. Efficient routing and scheduling algorithms for optical networks. *Proceedings of the 5th Annual ACM SIAM symposium on Discrete Algorithms,* 1994.

[Al+ 93] S. B. Alexander, et al. A precompetitive consortium on wideband all-optical networks. *Journal of Lightwave Technology,* 11(5/6):714-735, May/June 1993.

[AMO 93] R. K. Ahuja, T. L. Magnanti, J. B. Orlin. *Network Flows.* Prentice-Hall Inc. New Jersey, 1993. pp. 649-694.

[Bar 93] R. A. Barry. *Wavelength Routing for All-Optical Networks.* PhD Thesis, EECS, Massachusetts Institute of Technology, 1993.

[Ber 84] E. R. Berlekemp. *Algebraic Coding.* Aegean Park Press, CA, 1984.

[BG 92] D. P. Bertsekas, R. G. Gallager. *Data Networks.* Second Edition, Prentice Hall, New Jersey, 1992.

[BLM 93] Y. Birk, N. Linial, and R. Meshulam. On the uniform-traffic capacity of single-hop interconnections employing shared directional multichannels. *IEEE Transactions on Information Theory,* 39(1), January 1993.

[Dr+ 89] C. Dragone, et al. Efficient multichannel integrated optics star coupler on silicon. *Photonics Technology Lettters,* 1(8):241-243, Aug. 1989.

[DK 74] J. Denes, A.D. Keedwell, *Latin Squares and Their Applications.* Academic Press, NY, 1974, pp. 81-85.

[FFP 88] P. Feldman, J. Friedman, and N. Pippenger. Wide-sense non-blocking networks. *SIAM Journal of Discrete Mathematics*, 1(2):158-173, May 1988.

[GET 90] S. W. Golomb, T. Etzion, H. Taylor. Polygonal path construction for tuscan-k squares. *Ars Combinatoria,* 30:97-140, 1990.

[GJ 79] M. R. Garey, D. S. Johnson. *Computers and Intractability, A Guide to the Theory of NP-Completeness.* W. H. Freeman and Company, San Francisco, 1979.

[Gl+ 89] B. Glance, U. Koren, C. A. Burrus, and J. D. Evankow. Discretely tuned N-frequency laser for packet switching applications based on WDM. *Electronics Letters.* 27(15), July 1991.

[Hal 86] M. Hall. *Combinatorial Theory.* Wiley, NY, 1986.

[Hal 35] P. Hall. On representative of subsets. *Journal of the London Mathematical Society,* 10(1):26-30, 1935.

[Hui 90] J. Hui. *Switching and Traffic Theory For Integrated Broadband Networks.* Kluwer Academic Publishers, Boston, MA 1990.

[Lub 66] D. Lubell. A short proof of sperner's lemma. *Journal of Combinatorial Theory,* 1(2):402, September 1966.

[Lue 73] D. G. Luenberger. *Introduction to Linear and Nonlinear Programming.* Addison-Wesley Publishing Company, Reading, MA, 1973.

[Pan 92] R. K. Pankaj. *Architectures for Linear Lightwave Networks.* PhD Thesis, EECS, Massachusetts Institute of Technology, 1992.

[Pip 78] N. Pippenger. On rearrangeable and non-bBlocking switching networks. *Journal of Computer and System Sciences,* 17:145-162, 1978.

[Ros 83] S. M. Ross. *Stochastic Processes.* John Wiley & Sons, Inc., NY, NY, 1983.

[ST 91] B. E. A. Saleh, M. C. Teich. *Fundamentals of Photonics.* John Wiley & Sons, Inc., NY, 1991.

[Tom 66] J. A. Tomlin. Minimum-cost multicommodity network flows. *OR,* 14(1):45-51, February 1966.

[Til 80] T. W. Tillson. A hamiltonian decomposition of K^*_{2m}, $2m$ greater than or equal to 8. *Journal of Combinatorial Theory,* Series B 29:68-74, 1980.

[WZ 93] A. Wigderson and D. Zuckerman, Expanders that beat the eigenvalue bound: explicit construction and applications. *25th ACM Symposium on Theory of Computing,* 245-251, 1993.