Reconfigurable Networks For Fast Packet Switching

Shih-Chian Yang

CENG Technical Report: 91-15

A Dissertation Presented to the
FACULTY OF THE GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(Electrical Engineering)

Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, CA 90089-2562
(213)740-4579

To my wife and my parents

Acknowledgements

I would like to express my sincere gratitude to Dr. John Silvester, my dissertation committee chairman, for his advice and support. Without his encouragement, this research might never have been completed. I am also grateful to other committee members, Professors Cauligi S. Raghavendra, Ming-Deh Huang, Kai Hwang, and Michel Dubois for their useful suggestions and constructive criticisms.

Many thanks to my friends and colleagues in the Electrical Engineering Department at USC for their valuable discussions and support: Dr. David Chang, Dr. Kuo-Hui Liu, Dr. Sye-Je Wang, Dr. Chin Yuan, Dr. Jonathan Wang, Dr. Howard Liu, Thomas Papavassiliou, Dr. Arthur Lin, Dr. Anastasios Economides, Stanley Wang, K. Ramakrishnan and Nelson Fonseca. Special thanks to William Bates and Lea Vasquez for their friendship and administrative help.

Lastly, I am thankful to my parents, my brothers and my sister for their moral support. I am deeply grateful to my wife, Ai-Lin for her love, endless patience and unconditional support. Also, the cute smiles of my two little boys have been a source of inspiration in numerous hard working evenings.

Table of Contents

List of Figures

List of Tables

Abstract

The dramatic progress of fiber optics and VLSI technologies in past decade have stimulated the researchs into high speed networks. A very aggressive high speed network development is in Broadband Integrated Service Digital Networks (BISDN) or Asynchronous Transfer Mode (ATM) networks. An important question in ATM network design is how to provide fast packet switching functions.

We propose a large scale fast packet switching network that can be constructed with currently feasible technologies. A modular design that is carefully matched to technological design constraints makes large fast packet switching networks ($> 1000 \times 1000$) feasible. Based on our analysis of the technology, we find that module interconnection becomes the bottleneck for a large fast packet switch rather than the topology of the interconnection network.

For a large network, reliability becomes critical, so particular attention is paid to fault-tolerance which is achieved by dynamic reconfiguration of the module interconnection network. The proposed design significantly improves system reliability with relatively low hardware overhead. An abstract model of the replacement problem for our design is presented and the problem is transformed into a well known assignment problem. The maximum fault tolerance is found and a fast run time replacement algorithm is presented.

The reconfiguration capability can also be used to ameliorate unbalanced traffic flows. We formulate this traffic flow assignment problem for our switch fabric and we show that the problem is NP-hard. We then propose a simple heuristic algorithm.

Another scenario is the case that the switching speed is not critical. The conventional fault tolerant multistage interconnection network design techniques proposed by many researchers can be used. The performance under failure mode is not addressed. A serious performance degradation due to the irregularity of failure mode network is discovered and a new class of networks is then proposed. We have shown that the new networks improve the failure mode performance significantly without any hardware overhead and with a insignificant reliability degradation in the range of interest.

Chapter 1

Introduction

1.1 Communication Network Systems

Historically, the invention of a new technology creates demands for new services. The invention of the telephone in 1876 created the telecommunication industry which is indispensable to all the societies in the world today. The invention of the digital computer led to the need for data networks. The progress in fiber optics and VLSI technologies in the past decade has created high performance transmission and information processing systems resulting in the evolution of a new service Broadband Integrated Service Digital Networks (BISDN). BISDN is an *integrated service* for all types of communication. It provides a *shared facility* to reduce the cost of communication network and *high bandwidth* network access with a low delay over a wide area.

A communication network system consists of two major facilities: *i*) The transmission system; and *ii*) the switching system. As shown in Figure 1.1, a message generated from the information source A is transmitted by the transmission system and then switched to the desired destination by the switching system.

The focus of this dissertation is the switching system for BISDN services. Intelligent equipment creates a need for high performance communication networks. The transmission component of such a network will soon be available based on the international standard SONET which provides tremendous transmission bandwidth. It remains to provide high speed switching systems. We examine the requirements and survey previous research in high speed switching and then propose a reconfigurable switching network that pushes the boundaries of electronic technology, thus providing the highest possible switching rate.

The reconfiguration capabilities can be used to provide fault tolerance. When a failure occurs, the network can be reconfigured such that the failure is replaced by a spare. The network achieves great reliability with even a small number of spares.

The reconfiguration capabilities can also be used to alleviate the traffic imbalance problem which may occur with integrated broadband services. The interconnections of the network can be reconfigured so that the traffic load on a bottleneck link is reduced.

While BISDN is still in the research phase, an interim solution which has lower switching speed is required. In addition, some applications do not need very high speed switching. For these cases, conventional Fault Tolerant Multistage Interconnection Networks (FTMIN) can be used to overcome the reliability problem. However, most research in conventional FTMIN is concerned with only fault tolerance [2]. We find that the performance of a conventional

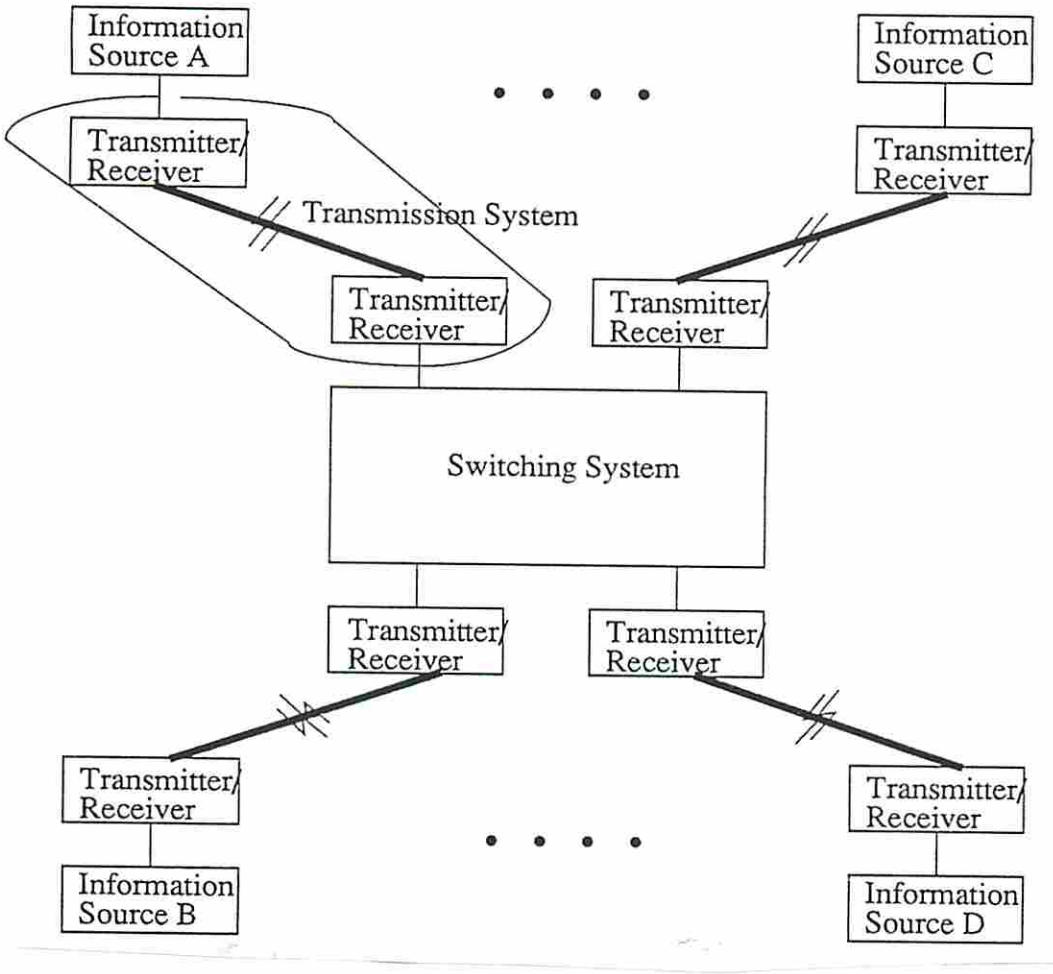


Figure 1.1: A Communication Network System

FTMIN may be seriously degraded under a failure mode operation and thus propose a new class of FTMINs to solve this problem.

1.2 Dissertation Outlines

In Chapter 2, we survey advances in fiber optic and VLSI technologies and show the needs for fast packet switching. The characteristics for very high speed switching are studied and a reconfigurable network that possesses these characteristics is proposed. How the reconfiguration capabilities are used to solve the reliability and unbalanced traffic problems is briefly discussed.

In Chapter 3, we study the fault tolerance of the proposed reconfigurable network with a replacement bipartite graph. Based on this model, the maximum fault tolerance, a replacement algorithm and hardware overhead are studied. Two implementations based on different interconnection techniques are given.

In Chapter 4, the problem of unbalanced traffic patterns due to integrated services is formulated as a topology design problem. It is shown that the problem is *NP*-hard. An efficient heuristic algorithm is given to solve the problem.

In Chapter 5, performance degradation of the conventional FTMIN under a failure mode operation is shown. A new class of FTMIN is proposed to alleviate this problem. The new

networks significantly improve the network performance in failure mode without any hardware overhead, yet sacrifice the network reliability only a little in the range of interest. Simulation and numerical techniques are used to show the tradeoff between the performance improvement and the reliability degradation.

We summarize the results and point out many interesting future research topics raised by this research in Chapter 6.

Chapter 2

Fast Packet Switching Networks

2.1 Technology Advancement

The progress in single mode fiber and VLSI technology provides significant cost reduction and quality improvement for communication networks and computer systems. These technology advances provide tremendous opportunities for new applications.

2.1.1 Fiber Optic Transmission System

Fiber optics transmission systems have become the choice for long haul telecommunications [104, 72]. Low transmission loss single mode fibers, high power lasers and sensitive receivers increase the transmission distance and hence reduce the number of repeaters required [84]. The minimum chromatic dispersion of single mode fibers allows a very high transmission rate which corresponds to a superior capacity. The high signal to noise ratio of single mode fibers induces a low bit error rate. Fiber optics transmission systems are also very reliable [25, 24]. These facts indicate that the fiber optics transmission systems are cost effective solutions for providing better quality telecommunication services.

In terrestrial telecommunications, single mode fiber optic transmission systems have been successfully deployed for interoffice trunking in North America [58], Europe [82] and Japan [99] and can achieve transmission rates of Gigabits per second (Gbit/s) and repeater spacing of 40 kilometer. The undersea fiber system TAT-8 that operates at 296 Megabits per second (Mbit/s) with 50 kilometer repeater spacings [26], provides transatlantic links between the United States and Europe. Deployment of higher bandwidth, better quality and lower cost fiber optic transmission systems is occurring at a fast pace all over the world.

In 1988, the American National Standard Institute (ANSI) and Consultative Committee for International Telephone and Telegraph (CCITT) approved the Synchronous Optical NETwork (SONET) standard [7, 8, 16, 102] (called the Synchronous Digital Hierarchy (SDH) within CCITT [40]). The SONET is a synchronous transport signal standard for high speed fiber optics transmission systems. It defines a basic rate of 51.84 Mbit/s, the STS-1 for the electronic signals and the OC-1 for the photonic signals. As shown in Table 2.1, higher rates are defined as a multiple of the basic rate up to 2.488 Gbit/s. The existing lower rate services, e.g. DS1, DS2, CEPT-1, are covered by Virtual Tributaries (VTs) which use a fraction of the basic rate.

The SONET STS-N frame format is shown in Figure 2.1 where N is an integer. There are four layers in the SONET standard. The photonic layer defines the signal conversion between

Synchronous Transport Signal	Optical Carrier Level	Line Rate (Mbit/s)
STS-1	OC-1	51.84
STS-3	OC-3	155.52
STS-9	OC-9	466.56
STS-12	OC-12	622.08
STS-18	OC-18	933.12
STS-24	OC-24	1244.16
STS-36	OC-36	1866.24
STS-48	OC-48	2488.32

Table 2.1: Standard SONET Rates

VT size	VT Rate (Mbit/s)	Mapping Example	Rate (Mbit/s)
VT-1.5	1.728	DS1	1.544
VT-2	2.304	CEPT-1	2.048
VT-3	3.456	DS1C	3.152
VT-6	6.912	DS2	6.312

Table 2.2: SONET Virtual Tributary Mapping for Asynchronous, Floating Mode

the photonic and electronic signals. The section layer regenerates the signals, e.g. in a repeater. The line layer provides the synchronization and multiplexing functions. The path layer maps services into the SONET format. Each SONET equipment has to be terminated at a layer as shown in Figure 2.2. All the overhead bytes corresponding to the terminated layer have to be examined and regenerated. There are no overhead bytes for the photonic layer. The overheads bytes for the section layer, line layer and path layer are shown in Figure 2.1. The remaining shaded area is the payload that can be used by an application.

The synchronous framing structure provides the foundation for the signal hierarchy. The signal hierarchy covers the rates from the low speed existing rates used in the United States, Europe and Japan to the rates of the future, e.g. broadband ISDN at 150 Mbit/s or 600 Mbit/s. Therefore, a wide variety of services can be transported on high bandwidth fiber optics transmission systems with the same interface throughout the world.

The multiplexing functions use only the synchronous Add/Drop Multiplexer (ADM) and crossconnect switch [91, 79] and the transport network does not need any knowledge from the services that use SONET. Hence, synchronous multiplexing equipment can be built with a relative low cost. A large part of the overhead defined in the three layers is devoted to the network management functions. Convenient Operations, Administration, Maintenance and Provision (OAM&P) functions are expected for SONET transmission systems. The costly network management problem will be significantly simplified [49]. Trial and deployment of the SONET networks are progressing [97] and SONET services will be available soon. High transfer rate and low cost worldwide transport networks are evolving.

The rate of a fiber optics transmission system is usually no greater than 1 or 2 Gbit/s, due to the so-called electronic bottleneck [47]. The photonic logic device technology is still in its infancy. Even a very simple logic function is very difficult to implement in optics. Control

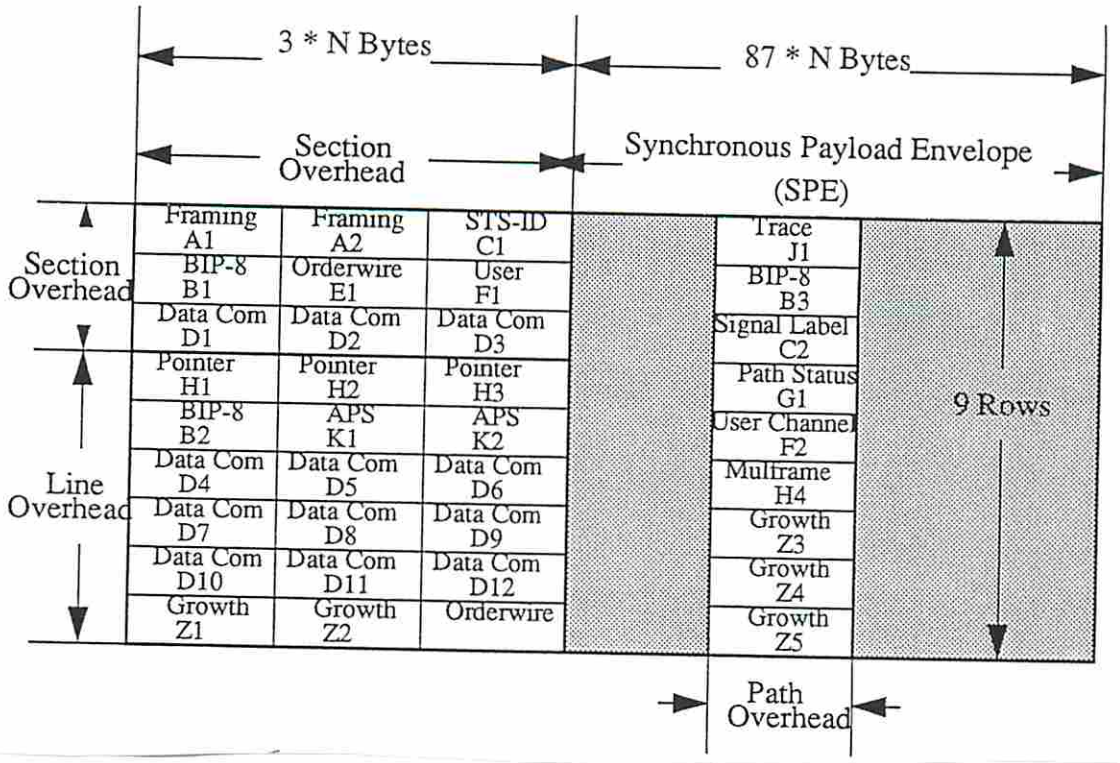


Figure 2.1: STS-N frame format

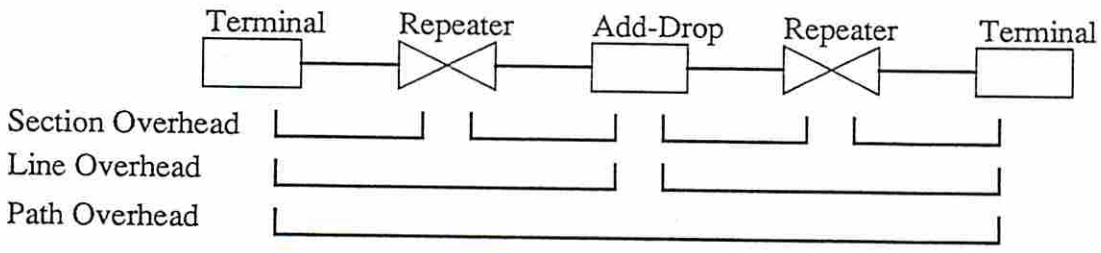


Figure 2.2: SONET Layered Overheads and Terminations

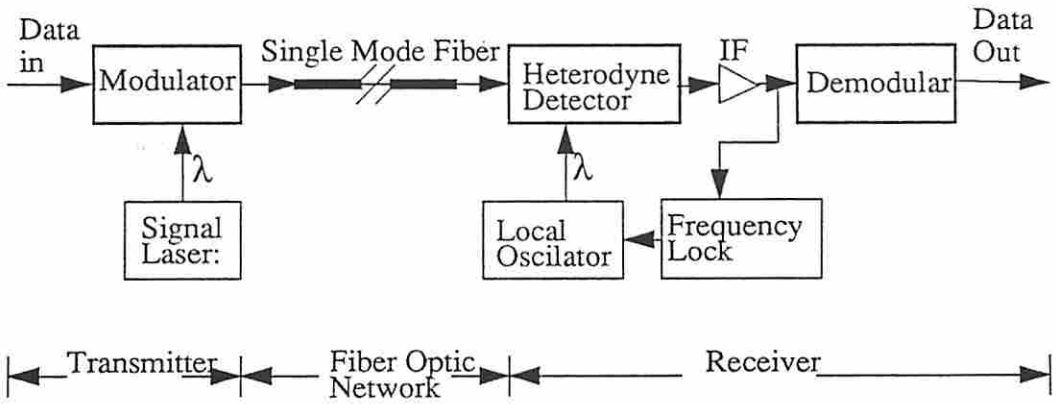


Figure 2.3: Basic Coherent Lightwave System

of photonic devices relies entirely on electronic techniques. However, the switching speed of electronic devices is limited by the mobility of electrons. It is very difficult to switch an electronic signal faster than 1 or 2 Gbit/s, whereas, the potential optical signal rate for a single mode fiber can reach more than 10 Terabits per second (Tbit/s), several orders of magnitude larger than the electronic counterpart [18].

Wavelength Division Multiplexing (WDM) can potentially provide a solution to the problem [18, 31, 44, 75, 74, 107]. As shown in Figure 2.3, the carrier signal, wavelength λ, is optically modulated by the input data. In the heterodyne receiver, the received signal is combined with the signal generated by a local oscillator. The Intermediate Frequency (IF) filter and demodulator demodulate the information signal. The carrier wavelength λ forms a channel between the transmitter and receiver.

Furthermore, several wavelength channels with distinct carrier wavelengths can be combined, using a star coupler, and transmitted on the same single mode fiber. As shown in Figure 2.4, the N local oscillators of the receiver are tuned to wavelengths λ_1 to λ_N. A tunable laser is used in each transmitter which is tuned to a wavelength channel of the desired receiver. The advantage of this approach is that although the transmission rate of the individual transmitter is still limited by the electronic device, the total transmission rate is much higher. The multiplexing function is automatically provided by properly tuning the wavelength of the transmitter. It has been shown in [70] that up to a hundred Gbit/s channels are achievable with current device technologies.

2.1.2 VLSI Technology

VLSI technology has progressed dramatically in the past decade. Semiconductors are the driving force for high technology products [98]. The possibility of a million-transistor chip has been demonstrated by the INTEL N10 design team [62]. This chip, i860, is a 64 bit Reduced Instruction Set Computer (RISC) coprocessor in an one-micron CMOS die for graphical computing. It has a 32-bit integer unit, a 64-bit 3-D graphics unit, a memory management unit, a 4 kilobyte instruction cache, an 8 kilobyte data cache and an ANSI/IEEE standard 754 floating point unit in an 168-pin ceramic pin grid array chip. It indicates that a functionally sophisticated block is feasible on a single silicon chip, thus reducing system cost and increasing system reliability.

A sub-nano second gate delay is possible with today's technology. CMOS VLSI chips running

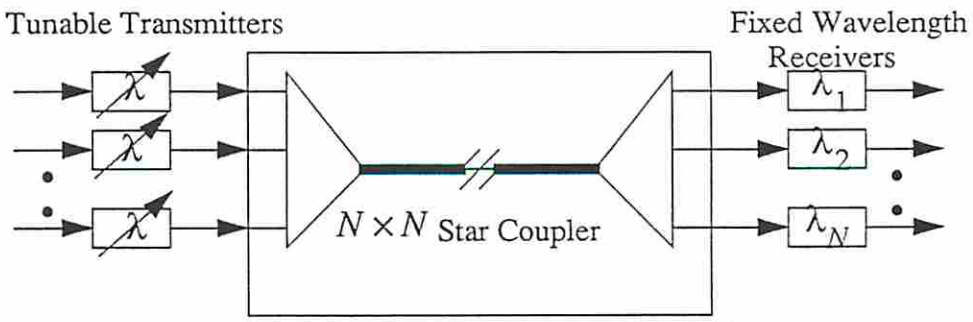


Figure 2.4: A Wavelength Division Multiplexing Network with Tunable Transmitter

at 25-50 mega hertz (MHz) are commonplace. Hitachi has announced the first 64 Megabit Dynamic RAM (DRAM) in mid-1990 and expects to go into volume production in 1995 [109]. GaAs devices that run ten times faster than the CMOS counterparts [68] provide a promising solution for high speed applications. The Multi-Chip Modules (MCM) that interconnect dies on substrates are appealing for military applications [57] and will be found in commercial applications when they becomes cost effective. Sophisticated Computer Aided Design (CAD) tools and Application Specific IC (ASIC) have matured and significantly shorten the design turnaround time.

Low cost Personal Computers (PC) and high performance Work-Stations (WS) that saw explosive growth in late 80's are the direct results of VLSI technology advancement. The VLSI based Local Area Networks (LAN) built around the IEEE 802.3 Ethernet or IEEE 802.5 token ring standard are another advancement made possible by this technology. They provide a 1-16 Mbit/s data network among PCs and WSs for resource sharing and distributed computing. The interconnected PCs and WSs form a robust computing facility for the corporate environment.

LANs are restricted to a relatively small geographical area. The number of attachable stations is also limited. There is a clear need to provide an internetwork LAN backbone [17] as shown in Figure 2.5. Metropolitan Area Networks (MAN) and High Speed LANs (HSLAN) are proposed for this application [100]. FDDI provides a 100 Mbit/s token ring network. DQDB uses a unidirectional dual bus to provide 155 Mbit/s access. The METROCORE Network proposed by BELLCORE is also a 150 Mbit/s unidirectional dual bus.

Bridges and Routers developed in late 80's provide internetworking capability among multi-vendor LANs [77]. TCP/IP or X.25 interfaces to Wide Area Networks (WAN) are also available.

As shown in Figure 2.6, the bridges and routers are connected via mesh interconnections and lack congestion control mechanisms, network management tools, etc. As the the network becomes large or the traffic flow becomes heavy, it becomes difficult to manage.

Furthermore, there were no standard data network protocols for rates higher than 64 Kilobits per second (Kbit/s) in WAN. The frame relay proposed by equipment vendors uses DS1 to provide 1.5 Mbit/s internetworking connections [76]. The Switched Multimegabit Data Service (SMDS) is an attempt from the Regional Bell Operation Companies (RBOC) to provide a public connectionless data service for wide area internetworking with DS1 and DS3 [46]. It uses a subset of the IEEE 802.6 (DQDB) standard as the Subscriber Network Interface (SNI) and uses MAN technology for the switching system, Figure 2.6. It is expected to be available in mid-1992.

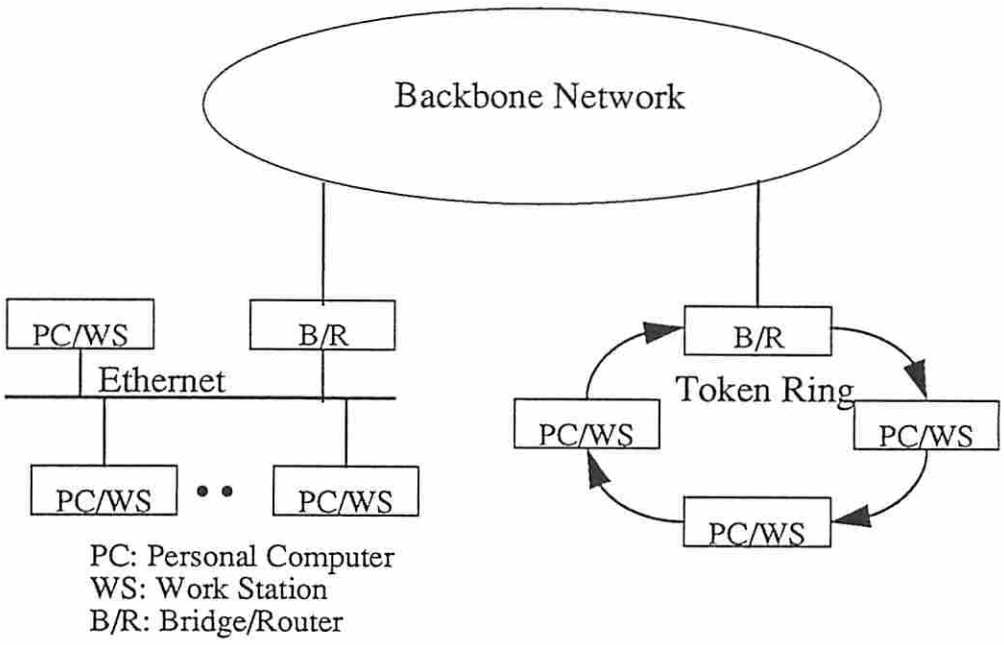


Figure 2.5: LAN interconnections

Graphics and image systems also benefited from VLSI technology advancement. High performance processors, high density storage device and high resolution color graphics systems are commercially available [55]. Graphics and image applications, e. g. CAD/CAM graphics, electronic publishing, color graphic presentation systems and facsimiles are common today. It is anticipated that these image systems and other multimedia applications will be networked in the near future [60]. Video applications, e.g. teleconferencing, video phone, high definition TV are under study [108]. These applications require much higher transfer rates, e.g. 150 Mbit/s and no existing network systems can satisfy their needs [39, 112].

2.2 Broadband Integrated Service Digital Networks

The technology advances discussed in the previous section have indicated three important aspects:

1. Highly integrated VLSI technology has generated very high performance information processing equipment: the PCs, WSs, network equipment, image/graphic systems and video systems which are widely used by business and residential users.
2. Several Gbit/s worldwide standard fiber optics transmission systems will soon be commonly available as the SONET deployment progresses. The cost for bandwidth will be reduced significantly.
3. Sophisticated and high speed function can be integrated into a single VLSI chip which becomes the building block of future high technology products.

The obvious need for and availability of technologies for the high speed networks have provided the impetus for development of Broadband Integrated Service Digital Networks (BISDN).

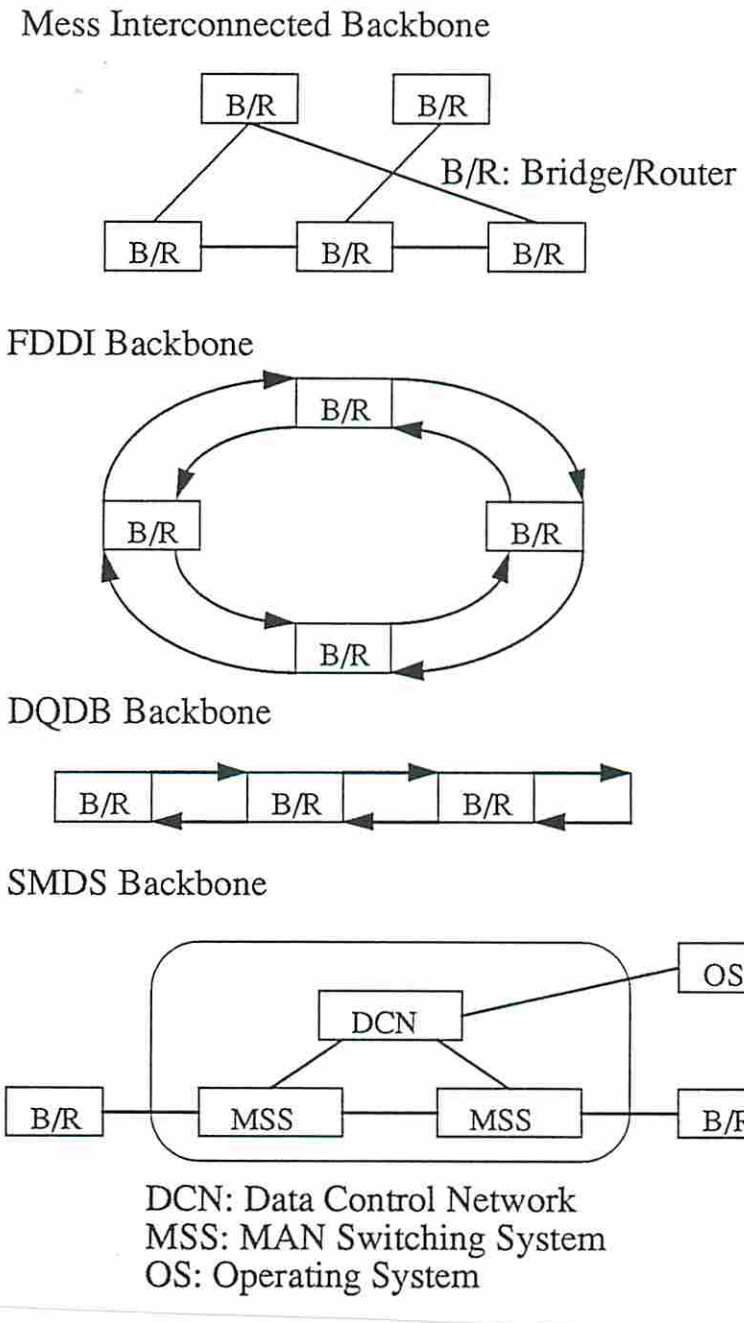


Figure 2.6: Backbone Networks for LAN interconnections

The need for BISDN to provide integrated services for applications from low rate Plain Old Telephony Services (POTS) to high rate video services has been widely recognized [6, 5, 11, 45, 61, 103]. Two standards bodies, the CCITT Study Group XVIII and U.S. T1S1, are actively developing standards for BISDN [92].

2.2.1 Asynchronous Transfer Mode Network

The major objectives for BISDN are to provide a worldwide standard interface for integrated voice, data, graphic, image and video services [92] using SONET as the transmission system.

The traffic characteristics of potential services include: [12, 83]

1. Both connection oriented and connectionless services
2. Both interactive and distributed services
3. Both bursty and continuous traffic.
4. Both broadband and narrowband rates.
5. Both point to point and point to multipoint connections

Accurate prediction of future needs is not possible [28, 112].

Hence, the protocol for the BISDN needs to be *flexible* to accommodate a wide range of applications and *simple* to minimize implementation overhead and to achieve highest possible performance. Also, it must be *application independent* so that service portability and multi-vendor equipment manufacturing are possible.

A synchronous transfer mode network that can properly transport only continuous data streams is not feasible for the diverse BISDN services. On the contrary, an asynchronous transfer mode network that transfers data on the basis of small size packets provides great flexibility to satisfy the diverse needs of BISDN. A cost reduction for all applications is possible by the network sharing. Also, data can be transported on a demand basis which offers the users flexibility to optimize their network.

Therefore, a consensus has been reached in the standards committees that the Asynchronous Transfer Mode (ATM) is the appropriate transport structure for BISDN.

2.2.2 ATM Protocol and Architecture

The ATM network is based on fast packet switching technology [92]. A user data stream is packetized into fixed length cells. Each cell has its own routing information and is switched asynchronously. A cell consists of a 48 byte user data segment and a 5 byte header as shown in Figure 2.7. Note that the destination tag in the figure is for the switch implementation and is not a part of the ATM cell format. The header contains the information for the flow control, routing, error detection and correction, and other cell control functions. There are no assumptions on the 48 byte user data segment for ATM equipment.

The ATM protocol model is shown in Figure 2.8 [92, 95]. The Physical Medium Dependent (PMD) layer uses the SONET STS-3n and STA-3nc as its transmission medium. The payload of SONET shown in Figure 2.1 contains a sequence of ATM cells. In the other words, an ATM equipment that terminates the path layer of SONET (or terminates the PMD layer) produces a sequence of ATM cells. The adaptation layer maps the information from users or control functions into ATM cells. It adapts different services to a common ATM format which greatly

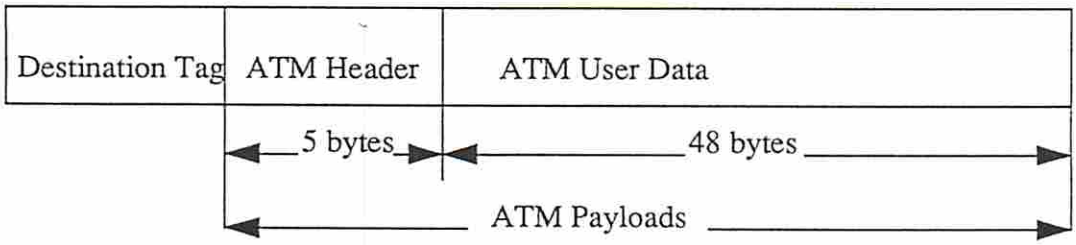


Figure 2.7: ATM Cell Format with Switch Fabric Overhead

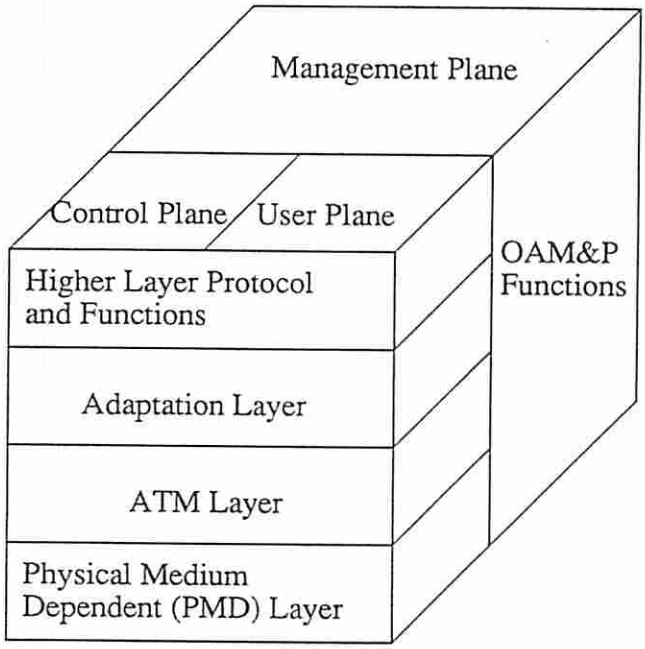


Figure 2.8: ATM Layers

simplifies the ATM equipment design. ATM equipment needs only to check the ATM header. The user data will be examined at the service termination. Therefore, the ATM layer provides a common cell transfer capability which is independent of any service related functions. It also provides the minimum functions necessary to transport asynchronous mode data.

An example of ATM network architecture is shown in Figure 2.9 [11, 20, 14, 101]. The Customer Premise Equipment (CPE) generates and converts service messages to the ATM format. They are multiplexed into a single data steam by the Remote Multiplexer (RM) and are then fed into a Remote Access Node (RAN) via a SONET transmission system. The RANs send ATM cells to a Exchange Node (EN) in the Central Office (CO) for switching. There are two types of CPE: the single sourced CPE e.g. the video equipment of a residential user and the Sharing Network (SN) e.g. the LAN of a business user. The users of a SN share the same high speed ATM network for their WAN access.

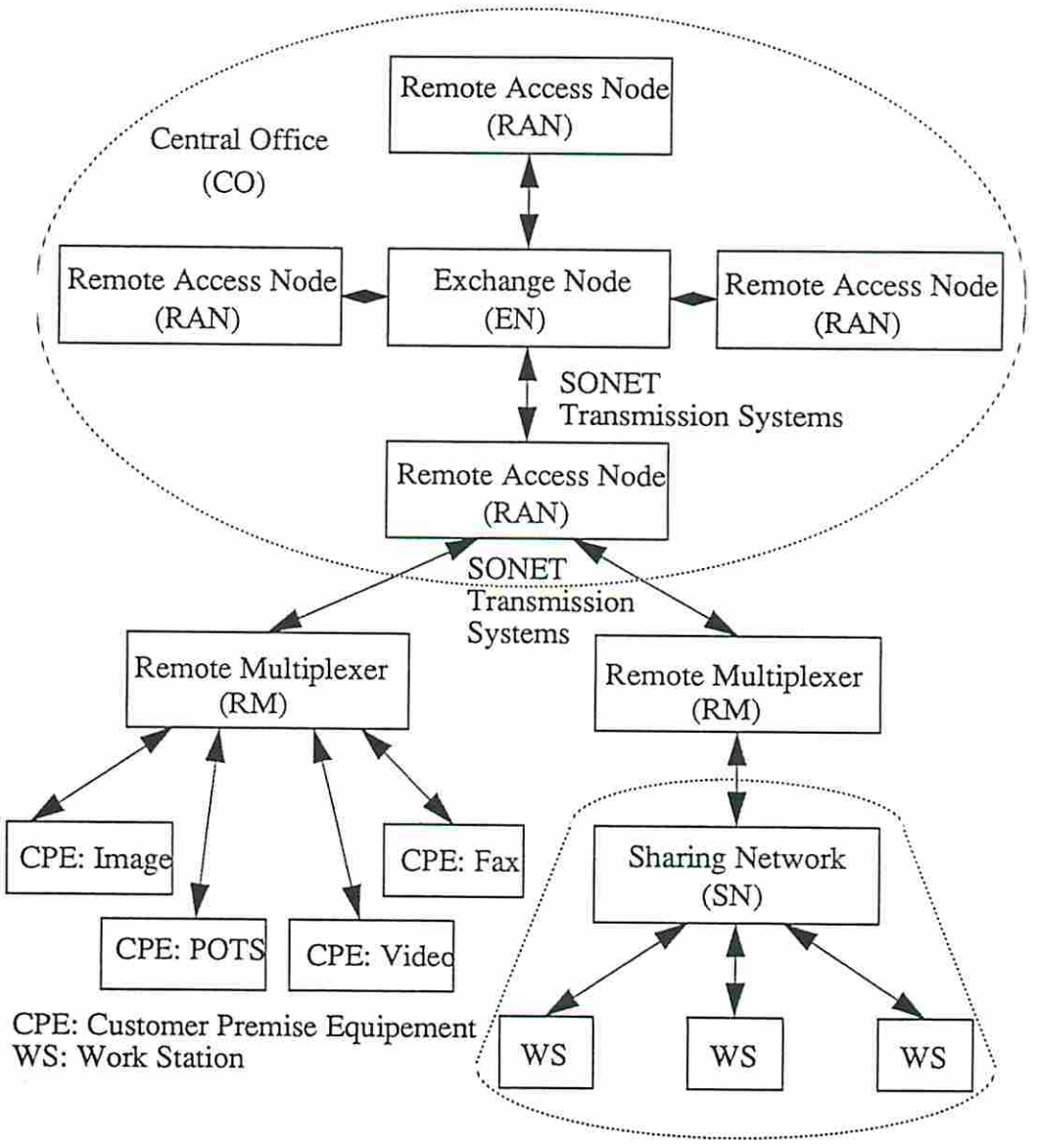


Figure 2.9: An Example of ATM Network

2.3 Fast Packet Switching Techniques

One major function of ATM networks is to provide switching capabilities for the high speed cells. There are two places that require fast packet switching techniques:

1. The centralized switch fabric for the exchange node: The aggregated bursty traffic from RAN is at a SONET rate, currently 150 Mbit/s or 600 Mbit/s and several Gbit/s in the future. Therefore, the exchange node needs to switch the ATM cells very fast.
2. The distributed switching for the sharing network: A sharing network typically operates at around 10 Mbit/s range with today's LAN technology or 100 Mbit/s with proposed MAN technology. However, as more graphic, image and video equipment is attached to the sharing network and more powerful machines are able to manipulate the generated data, a much higher switching rate will be needed.

2.3.1 Centralized Switch Fabric

Although photonic switching may eventually provide a solution, current photonic logic cannot support the complexity for the ATM switch design [78, 87, 90]. Thus, it is reasonable to assume that the fundamental switch modules for the medium term will continue to be electronic.

There are four basic switching techniques [29, 52].

1. A *space switching* switch selects one of the physical links to route data, e.g. a cross bar switch.
2. A *time switching* switch delays a cell a certain amount of time until the desired time slot corresponding to the destination. Time division multiplexing is a typical example [53].
3. A *frequency (wavelength) switching* switch selects one of the carrier frequencies to route the data. The multichannel multihop lightwave networks [1] use WDM technique is an example.
4. A *code switching* switch uses different codes to distinguish the desired destinations. The code division multiplexing technique on frequency channels [38, 96, 106] and address filtration (e.g. Ethernet, Token ring) are examples.

The time switching [36] and address filtration techniques usually require a broadcasting channel which imposes a limitation on high speed switching. On the other hand, for the frequency switching and code division multiplexing, it is fairly difficult to synchronize among carrier channels at the ATM cell level. The channel conflict problem is very hard to solve. They can be used in a relative long period, e.g for circuit switching or network reconfiguration. Therefore, space switching is the only viable technique for very high speed switching systems.

A fundamental problem in switch design is the so-called Head Of Line (HOL) blocking which occurs when more than one input line has a cell for the same output line. There are two types of HOL blocking: the internal blocking and external blocking. *Internal blocking* occurs when two cells are sent through the same internal link (to the switch) simultaneously even though their destinations are different. An example for a unique path Multistage Interconnection Network (MIN) is shown in Figure 2.10 [113]. Sources 1 and 5 have a cell for different destinations, 8 and 11 respectively, but they both require the upper outnode link of switch module 3 at switch module stage 1. *External conflict* occurs when several sources request the same destination

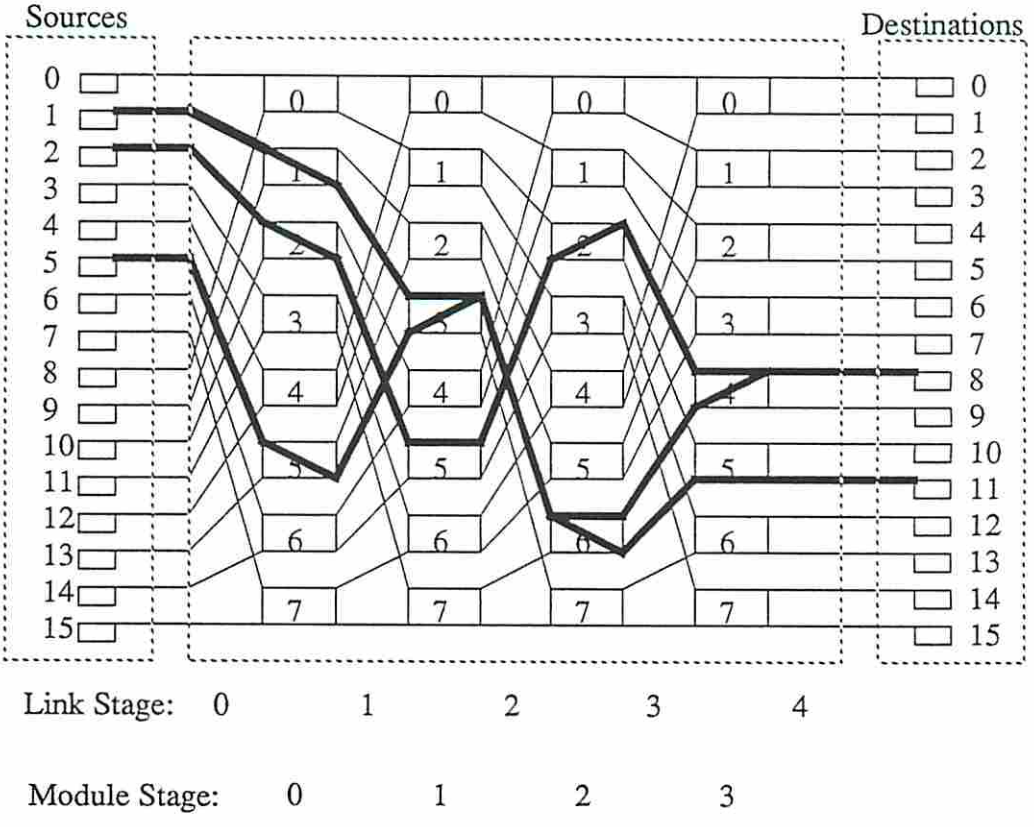


Figure 2.10: A Unique Path MIN

simultaneously. An example is when sources 1 and 2 both have a cell to the same destination 8. Note that they do not incur blocking internal to the switch.

Internal blocking can be avoided by the use of a non-blocking network. External blocking is more difficult to solve, however. The Integrated Service Packet Network (ISPN) [19] uses an internally buffered Banyan network to overcome both blocking problems. A buffer is required at each innode of all switch modules. Not only is more hardware overhead required but also more queueing delay is introduced.

Many switch designs use the internally non-blocking Batcher Banyan networks [10] to avoid internal blocking and therefore, internal buffers are not required. There are various ways to resolve the external blocking problem. The Starlite switch [51] circulates the blocked cells through reentry links which results in a serious sequencing problem. To solve this problem, the 3-phase algorithm in [54] uses arbitration and acknowledgement phases to resolve external blocking before cells are sent. A common problem for these two designs is the necessity for feedback links which restrict network size.

Another interesting design, the Knockout switch [35, 117], uses the knockout principle: it is unlikely that more than L sources request the same destination simultaneously. All the sources broadcast their HOL cells to all the destinations but only up to L cells are collected at each destination. The extra externally blocked cells are simply dropped. The cell loss probability is very small for reasonably large L. A fully connected bus interconnection is used for broadcasting which inhibits expandability for high speed networks. The Photonic knockout switch [34] uses the Wavelength Division Multiplexing (WDM) technique to switch the ATM cells. Not only

does this design require a wide range agile tunable laser diode, but the slot synchronization and the output contention prevent it from running at a very high speed. The Growable switch architecture [36] generalizes the knockout principle to a group of destinations. It overcomes the expandability problem that is inherent in the other designs.

The blocked cells need to be buffered at the innodes of a switch module which is referred to as *input buffering*. Blocking implies that some outnodes are idle and therefore, the achievable throughput is degraded. It has been shown that the achievable throughput can be as low as $2 - \sqrt{2}$ for a big switch module even under a uniform traffic pattern assumption [48]. To alleviate this problem, bypass queueing discipline [19], window input buffering [48] or decoding first buffering scheme [116] can be used, all provide a similar solution. On the other hand, *output buffering* does not have this problem but it requires a very fast scheme to allow the externally blocked cells to enter the output buffer simultaneously [73, 81, 117].

Another important issue is the switch module control. The stored control memory can only work for circuit switching networks since it is not efficient to manipulate a centralized control store on a per cell basis. On the other hand, the routing table mapping in packet switching computer communication networks is not feasible for a high speed switch design since there is not much processing time for each cell. The *self-routing* scheme used in many MIN designs provides a good solution. The destination of a cell is translated to a destination tag that contains the minimum information about the destination within this switch fabric as shown in Figure 2.7. Each switch module decodes only the necessary part of the destination tag so that the switching speed at each switch module is maximized. An example is to use a single bit for each 2×2 switch module for a MIN design.

At a high line rate, switching with a single line implementation is not feasible. A parallel implementation that process the incoming cells over parallel lines is required. The parallel implementation suffers from two main limitations:

1. Maintaining synchronization of parallel lines between VLSI chips is very difficult at a high speed.
2. The number of pins per VLSI chip is limited.

Thus, we choose a VLSI switch module chip that is parallel internally but serial externally as shown in Figure 2.11. For each switch module, the serial incoming signals are converted to a parallel form with serial to parallel converters. Internally a $k \times k$ cross bar switch module routes the incoming cells to the desired destination with the parallel lines. Parallel to serial converters convert them back to serial form. The serial signal lines are much easier for the layout of interconnections and require fewer pins. With a reasonably large number of internal parallel lines, the speed bottleneck becomes the serial interconnections. Under distributed control, this approach provides very high speed even for a large switch fabric and is probably the best alternative until full photonic switching matures.

As shown in Figure 2.12, the switch modules are mounted on PC boards and then integrated into a cabinet. It is still possible to keep the interconnects short on the same PC board and thus the low cost solution is to use copper wires for the on-board interconnections. As the switch fabric size grows beyond the PC board size, the interconnects become longer and harder for high speed design. Fiber optics are used for the interconnect. Note that the on-board interconnections are not restricted to be in copper, they can be any cost effective technology.

The Multicast switch fabric and Growable switch architecture do not really suffer from an expandability problem. The major drawbacks of the Multicast switch fabric (or the Banyan

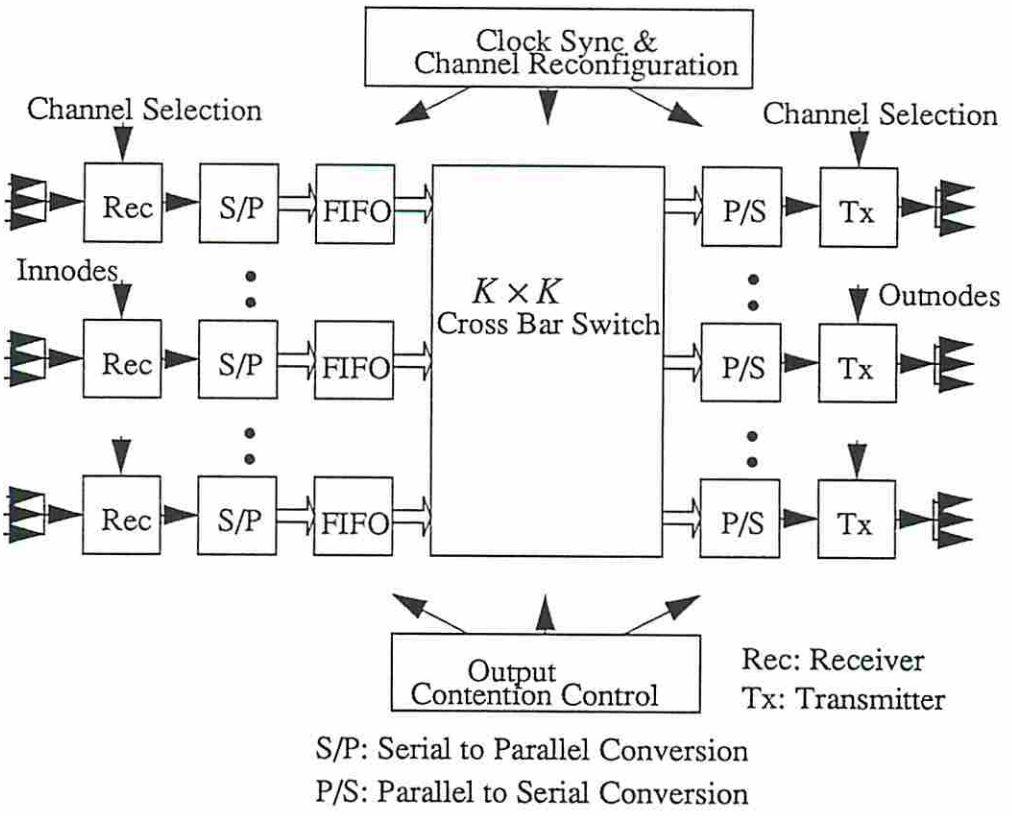


Figure 2.11: Switch Module for the Centralized Switch Fabric

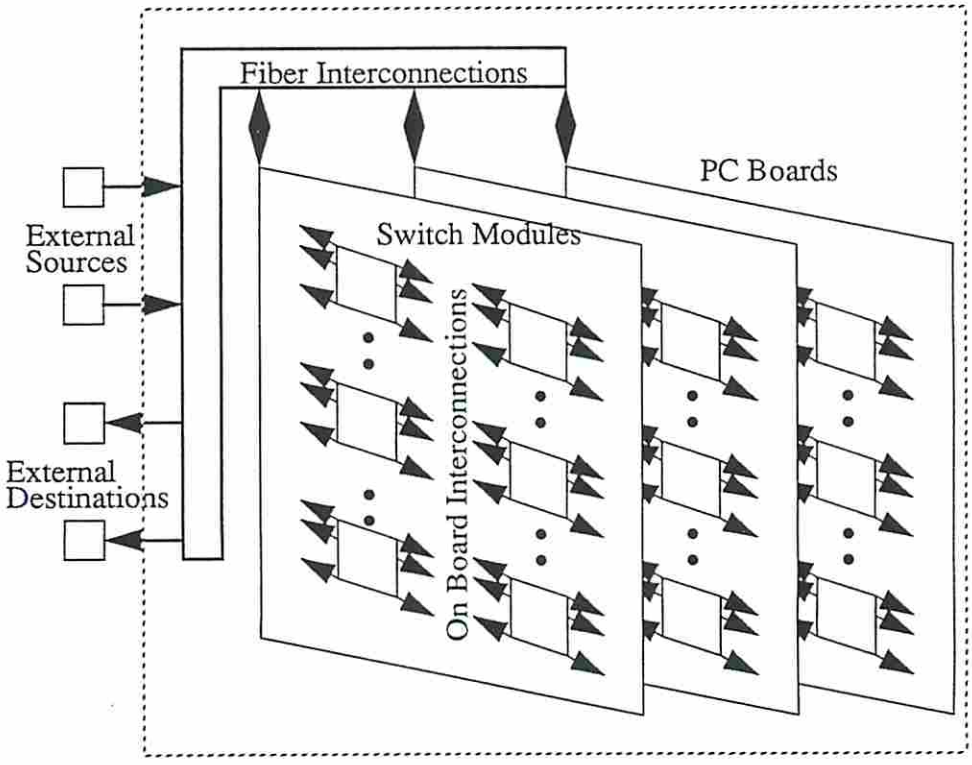


Figure 2.12: A Centralized Switch Fabric

network with internal buffers) are:

1. A large memory requirement
2. A longer queueing delay.
3. The lower achievable bandwidth when compared to output buffering.

Although the output buffering scheme can achieve a better throughput, it requires more switch modules for a internal non-blocking network and also requires the switch to effectively operate at a higher speed. The advances in CMOS VLSI technology allow very high gate density. Thus, providing a reasonable amount of memory inside the VLSI chip for internal buffering is not difficult. 8×8 to 32×32 single chip modules are feasible which significantly reduces the number of stages. Therefore, the queueing delay due to memory at each stage is no longer a serious problem. As mentioned before, the achievable bandwidth can be improved by using the bypass queueing discipline, the window input buffering or the decoding first buffering scheme.

As an example, consider a 4096×4096 switch fabric utilizing 8×8 switch modules with 32 internal parallel lines. Four stages of 512 switch modules are required. Based on the 53 byte ATM cell size and a reasonable buffer size of 25 cells per input, 10 Kbyte of memory buffer is required in each chip (switch module) which is feasible for a CMOS VLSI implementation. VLSI chips running at 20 MHz are required for a 600 Mbit/s line rate. A 700 *nsec* per cell switching time (or 14 clocks for the 20 MHz chip) is allowed for a bursty traffic. Note that the serial to parallel and parallel to serial converters have to run at at least 600 MHz and are asynchronous.

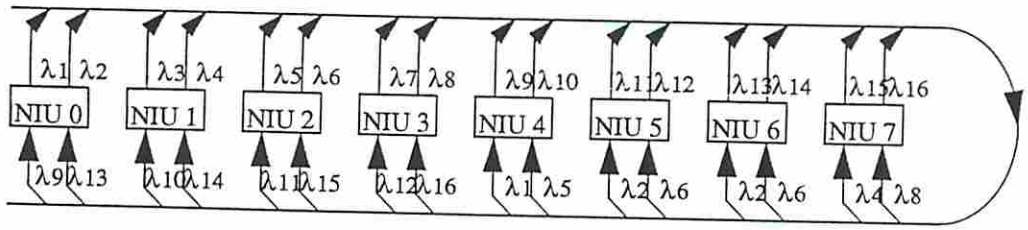


Figure 2.13: Multichannel Multihop Lightwave Network

Another important requirement for ATM switch fabric design is to keep cells in sequence, since resequencing overhead is extremely large in high speed networks. Multiple paths for the same source are better avoided. Therefore, we use the unique path MIN based on a single chip $k \times k$ switch module with an internal buffer at each switch module for its simplicity and regular switch module design. Any network that is equivalent to the unique path MIN, such as the Omega network [113] etc, can also be used.

2.3.2 Distributed Sharing Network

Resource sharing and distributed computing are two major functions for distributed computer systems on a sharing network. The successful LAN systems that play a indispensable role in the corporate environment today have proved the need for a sharing network. As discussed in section 2.1.2, more bandwidth hungry applications (e.g. image, video and multimedia etc.) are appearing, and much higher bandwidth will be needed for the sharing network.

It is interesting to note that the topologies of most LAN/MAN systems are broadcast type (the Ethernet, token ring, FDDI, DQDB and etc). The problem for a broadcasting media is that a single, bandwidth limited resource is shared by all the stations. As the bandwidth demand increases, it soon reaches electronic device speed limitations.

The WDM technology described in section 2.1.1 allows many wavelength channels to co-exist in a single mode fiber. As in the LAN/MAN counterpart, it is also a broadcasting media. Yet, many wavelength channels, each electronic technology limited, are carried on one single mode fiber. The total useful bandwidth carried on the single media is many times the electronic limitation. Around one thousand 100 Mbit/s channels or one hundred Gbit/s channels are possible with the tunable laser technology demonstrated in research labs [70]. The only new device is the tunable laser diode.

The multichannel multihop lightwave network [1], (Figure 2.13), is an example that make use of the WDM technique. As shown in Figure 2.13, the stations in a sharing network attach to a broadcasting media (a shared bus, a shared tree or a star coupler). Each station uses two incoming and two outgoing WDM channels and has a buffered electronic cross bar switch module. It receives cells from the two incoming channels and transmits cells to the two outgoing channels. Also, it has to forward the cells whose destination is not this station.

By a proper assignment of wavelengths, a logical topology of the network can be drawn. The ShuffleNet [48] is an example of the logical topology as shown in Figure 2.14. It is very similar to a unique path MIN which characterizes the essentials of switching function. The only difference is the external access to *all* switch modules as comparing to the unique path MIN in Figure 2.10. Therefore, the topology of multichannel multihop lightwave network is a single stage interconnection network [111].

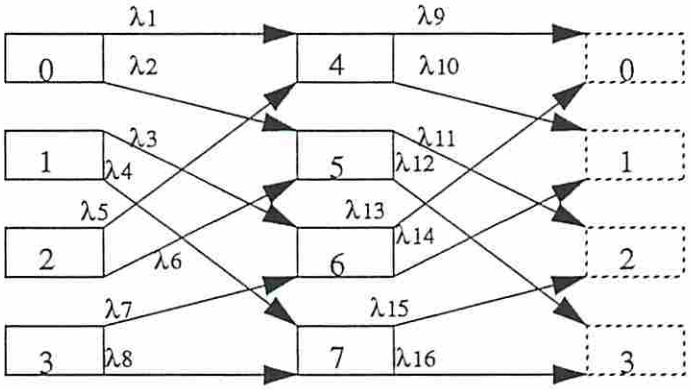


Figure 2.14: ShuffleNet: Topology for Multichannel Multihop Lightwave Network

Several problems are associated with the multichannel multihop lightwave network architecture. First, the maximum achievable bandwidth of an external node is limited to $O(\frac{1}{\log N})$ due to the channel sharing (and the switch module sharing) among external and internal traffic, (where N is the size of the ShuffleNet and there are $\frac{N}{2}(\log N - 1)$ switch modules in the network). This increased load also causes more queueing delay for a similar amount of traffic. Secondly, since each message potentially traverses the fiber $O(\log N)$ times, it is not practical to spread the switch modules over a wide area [9]. Thirdly, each station relies on the other stations to forwarding its cells. Distributed network management functions are required.

An alternative approach is to collect all these switch modules to a centralized location. All access stations transmit cells to and receive cells from the centralized switch fabric through WDM channels. The centralized switch topology can be the ShuffleNet or the unique path MIN for a higher line rate [116].

2.4 Network Reconfiguration

Although the internally buffered unique path MIN or the ShuffleNet is capable of providing all necessary switching functions, they are subject to the following two problems:

1. *Reliability*: One of the main goals of design for fast packet switching is to push the boundaries of electronic technology to best match the abundant bandwidth of fiber optics. One problem associated with the very high speed switching devices is the concomitant reduction of component reliability. The reliability problem is especially severe when the size of network becomes very large.
2. *Unbalanced traffic pattern*: Each incoming line carries diverse services which may create unbalanced traffic to the network. For example, a business area has much higher traffic flow during business hours while a residential area has higher traffic flow after hours. It has been shown that the unbalanced traffic pattern suffers from a serious performance degradation problem [33, 71, 86].

The use of tunable laser diodes [70] and/or tunable optical filters [59] provides the potential for a rich set of redundant links between the switch modules. It is possible to reconfigure the links by retuning the transmitter or receiver. While the tunable optical devices are still in the

FTMINs	Hardware cost	Bandwidth	# of paths per sourc/dest. pair
unique path MIN	$\frac{N}{2}logN$	1	1
extra stage (e.g. ESC)	$\frac{N}{2}logN + \frac{N}{2}$	1	2
Beneš	$NlogN - \frac{N}{2}$	1	N
chained MIN,ASEN	$\frac{9}{8}NlogN$	1	v. large
two copy MIN	$NlogN + N$	2	2
INDRA	$R\frac{N}{2}logN + \frac{NR^2}{4} + \frac{NR}{2}$	R	R^2
ACN	$2NlogN + N$	2	2N
2-MDN	$2NlogN - N$	2	2N
F-net	$4NlogN$	4	N

Table 2.3: Comparison of hardware cost, Bandwidth, number of paths per source/destination pair

research stage, it is possible to have redundant links with the redundant physical connections. The cost overhead is not significant for the centralized switch. Similarly, it is possible to have redundant links for copper connections without too much cost impact. The capability to reconfigure the network topology with redundant links provides a potential solution to both the reliability and unbalanced traffic pattern problems.

2.4.1 Fault Tolerance

Similar reliability problems have been discussed in the design of Fault Tolerant Multistage Interconnection Networks (FTMIN) [2]. A FTMIN provides multiple paths between sources and destinations by introducing redundant switches and links to a MIN topology. In Table 2.3, we compare several FTMINs found in the literature, looking at their hardware cost, bandwidth, and number of paths for each source/destination pair. The intention of comparison is to give a qualitative feel for the performance and reliability tradeoffs in terms of number of active devices (the switch module) rather than a precise quantitative comparison.

The *hardware cost* of a network is calculated based on the cost of a 2×2 switch module. A $k \times k$ switch module costs $\frac{k^2}{4}$ and an $1 \times k$ or $k \times 1$ switch module costs $\frac{k}{4}$. The numbers may be slightly different from the ones in the original papers due to the different assumptions or definitions. The *bandwidth* of a network is the maximum throughput it can provide (assume the infinite, output buffering). The exact solution of bandwidth is dependent on the operating mode of network and is in general a hard problem [30]. Rough estimates are given and are normalized to the bandwidth of unique path MIN. We consider only symmetrical FTMINs, thus the improved IADM[80], the Gamma network [89] and other similar networks are excluded for this comparison.

The bandwidth differences of extra stage MIN [3], Beneš network [13], chained MIN [105] and ASEN [64] are insignificant in comparing with the unique path MIN. The additional switch modules are mainly devoted to the reliability. A similar conclusion that much higher reliability can be achieved with these types of topology can be found in [23, 64].

The bandwidth is proportional to the hardware cost if the number of paths per source/destination pair remains *constant*, e.g. the two copy MIN, which will be defined in Chapter 5, and INDRA [93]. In these cases, the extra hardware is used for the performance improvement. For the rest

in the table, F-net [23], ACN [94] and 2-MDN [94], the hardware is used to improve both the reliability and performance.

Another question is how to derive the redundancy: link redundancy or switch module redundancy. For example, the ASEN and the Dilation network [63] use link redundancy. The ASEN solves only the reliability problem while the Dilation network provides extra bandwidth. On the other hand, the ESC, the Replication network [63], INDRA, F-net and ACN 2-MDN rely mostly on switch module redundancy. ESC concentrates only on the reliability while the Replication network is concerned with the performance improvement. The other networks allow both reliability and performance improvement. Another case is to make use of both types of redundancy, e.g. the Dynamic Redundancy (DR) network [56]. The DR network allow for both reliability and performance improvement.

As mentioned previously, the self routing scheme is very critical in designing a fast switching network. A change of routing scheme due to failures may introduce a significant complexity for a switch module which slows down the switching speed. Therefore, it is very desirable not to change the *functional topology* of a network under the failure mode. A network is running in a *Topology Invariant* (TI) mode if it does not change its functional topology with the fault tolerable failures. Clearly, the performance is not degraded by the fault tolerable failures for a network running in TI mode. Most networks mentioned above that address performance improvement can run in the TI mode. However, the reliability is reduced since many originally fault tolerable failures are now not tolerable. The fault tolerance for networks running in TI mode is considered in Chapter 3.

2.4.2 Topology Design

Consider the multichannel multihop lightwave network. In theory, it is possible to set up a direct link between any pair of stations due to the broadcasting nature of the WDM channels. This implies a fully connected network base topology. The problem in fast packet switching environment is that changing a connection (a WDM channel or a link) to another station can not be done at the cell level due to the difficulty of synchronizing different WDM channels in a short period of time. Hence, the retuning capability is not efficient for the cell switching. However, channel switching by retuning the lasers can be done in a longer period. This can be used to reconfigure the network topology to reflect a change of traffic patter, for example. This is a topology design problem that selects a permissible subgraph from the fully connected base topology such that each transmitter can connect to exactly one receiver and vice versa. In general, the topology design problem is to find an optimal topology for a given objective function from a *permissible topology space*. The permissible topology space reflects the design restrictions.

The cells in a multichannel multihop lightwave network need to be hopped through several intermediate stations before reaching the destination. A cell for a neighbor station may be unnecessarily hopped back and forth all over the network which induces a big transmission delay. In [9], an objective function that minimize the mean hopping transmission delay for a given traffic flow pattern is discussed. It is well known that a link with the maximum link traffic flow is the network bottleneck [15]. An objective function that minimizes the maximum link traffic flow is studied in [67].

The base topology is a fully connected network due to the unlimited number of available channels. Any permissible subgraph subject to the single link constraint is in the permissible

	Objective Function		
Topology Restriction	min-max link load	min hopping transmission delay	max network capacity
WDM	Labourdette, Acampora 90 [67]	Bannister, Gerla 89 [9]	
WDM / Small Star Coupler	Labourdette, Acampora 90 [66]		
Given Functional Topology	Yang, Silvester 91 [114]		Chlamtac, Ganz, Karmi 90 [22]
Multiple Link Group		Gerla, Monteiro, Pazos 89 [43]	Chlamtac, Ganz, Karmi 89 [21]
Hardware Configuration			Nassehi, Tobagi, Marhic 85 [85]

Table 2.4: Category of Topology Design Problem

topology space. If the tunable laser has a limited tuning range, the number of stations that are fully connected may be limited. In this case, the base topology is not fully connected and hence, the permissible topology space is further restricted. This problem is studied in [66].

Consider a trunk system where a trunk between two central offices may have a group of channels. Each channel can be represented by a weighted link (in terms of channel capacity). Therefore the permissible topology space is defined by the weight links. The objective function of minimizing the transmission delay is considered in [43]. Another similar problem is formulated base on a pure light path network and the objective function is to maximize the network capacity [21].

The topology design problem presented in [85] studies the permissible topology space due to different hardware configuration, especially the power loss due to the star coupler that connects all the distributed stations. The objective functions is to provide the maximum number of stations for a given set of hardware components.

The topologies resulting from the algorithms for solving the above mentioned problems may no longer retain the desired functional topology. As discussed in the previous section, this may degrade the performance of a high speed network. Consider the TI mode for a network implementation where a fixed underlying topology (functional topology) is given. This restricts the permissible topology space. The topology reconfiguration come from mapping the access networks. All these approaches can be categorized by their design restrictions for the permissible topology space and the objective function as shown in Table 2.4 This problem with an objective function to minimize the maximum link traffic flow is studied in Chapter 4.

2.4.3 Reconfigurable Networks

The switch fabric or sharing network can be modeled as shown in Figure 2.15. The external sources send cells to the network and external destinations receive cells from the network. The ATM cell address is translated to an internal destination tag as shown in Figure 2.7. The Access Networks (AN) map the external sources and destinations to the internal sources and destinations by one to one mappings. A functional topology for the internal network is designed to satisfy all the desired switching functions.

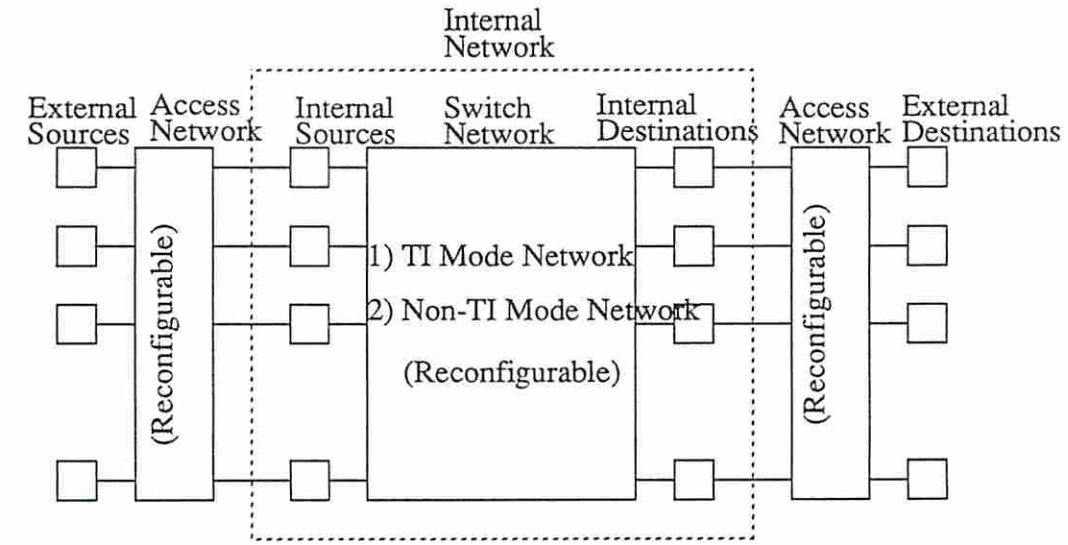


Figure 2.15: Network Model for Reconfigurable Network

As discussed previously, it is desirable to have the network running in TI mode. Base on this assumption, two areas for the reconfiguration are identified:

1. Fault tolerance: The reconfiguration of redundant links and spare switch modules for the internal network leads to a class of fault tolerant networks.
2. Topology design: Although the functional topology due to the switching functions are fixed, the reconfiguration of access network provides a great opportunity to balance the given traffic flows. The topology is actually changed in terms of the given external traffic flows even though the functional topology is kept the same.

On the other hand, if we *relax* the restriction due to the TI mode, the FTMIN proposed in literature can be used to improve the reliability and performance. The fault tolerance of these FTMINs has been studied extensively as surveyed in section 2.4.1. Yet, the performance degradation under failure mode has not been addressed. The tradeoffs between the reliability and performance for FTMINs are discussed in Chapter 5.

For a centralized switch fabric, the external sources and destinations are the incoming and outgoing access lines of ATM network. A unique path MIN is assumed for the functional topology of internal network. For a distributed sharing network, the external sources and destinations are the access stations attached to the network. ShuffleNet is assumed for the functional topology of internal network.

The underlying structure of ShuffleNet is similar to the unique path MIN. The only difference is that all the switch modules are accessible to the sources and destinations. Therefore, we use the unique path MIN as a reference functional topology for both cases and distinguish them only when necessary. A unique path MIN of size $N = k^n$ is shown in Figure 2.10. There are n *switch module stages* and $n+1$ *link stages*. There are $\frac{N}{k}$ switch modules in each stage. The switch module stages are labeled from 0 to $n-1$ while the link stages are labeled from 0 to n. For the unique path MIN, we label the external sources, external destinations, internal sources and internal destinations 0, 1, 2,.., $N-1$ as shown in Figure 2.10. For the ShuffleNet, the

external sources, external destinations, internal sources and internal destinations are labeled according to their corresponding switch module.

Chapter 3

Network Fault Tolerance with Topology Invariant Mode

For a network running in the topology invariant mode, the reconfigured network with a fault tolerable failure set maintains the functional topology. The switching function of a failed switch module has to be replaced by a live switch module. Therefore, there always exists a switch module replacement relationship. We model this relationship and study the properties of the network running in the topology invariant mode accordingly.

3.1 Replacement Model

A *switch module* is a block of components that performs certain *switching functions*. For the studies in this thesis, a switch module is a $k \times k$ cross bar switch. A switch module x is *replaceable* by a switch module y if when x fails, y can assume all the switching functions of x. The links required for supporting the normal switching functions are called the *functional links* and the spared links are called *redundant links*. All these links are called the *reconfigurable links*. Note that the definition is applicable to the ShuffleNet too once there are reconfigurable links for the external sources and destinations of switch module.

A simple replacement is shown in Figure 3.1. The switch module x is actively connected to the switch modules a, b, c and d (solid lines). The switch module y is a replacement for x, shown by the line with an arrow. Similarly, the switch modules a, b, c and d are replaceable by the switch modules a_r, b_r, c_r and d_r respectively. When the switch module x fails, the switch module y replaces it by activating the dotted links to the switch modules a, b, c and d. If the switch module a fails, the dotted links to a_r are used. Therefore, if all the dotted links are available, the switch module x is replaceable by y independent of the state of other switch modules. Replacements considered in this chapter are always such independent replacements.

A failed switch module needs to be replaced by a spare one that assumes its switching functions. If there are no spares that can directly replace the failed switch module, a sequence of replacements is required until a spare switch module is reached. A convenient way to represent this replacement process is by means of the replacement graph, where a *directed edge* from $y \rightarrow x$ indicates that module y can replace module x. The replacement graph contains all such edges. The sequence of replacements thus corresponds to a (directed) path in this graph.

As a more complicated design example: the interconnections at stage 2 of a 4 stage unique path MIN (in Figure 2.10) based on 2×2 switch modules are shown in Figure 3.2. The solid

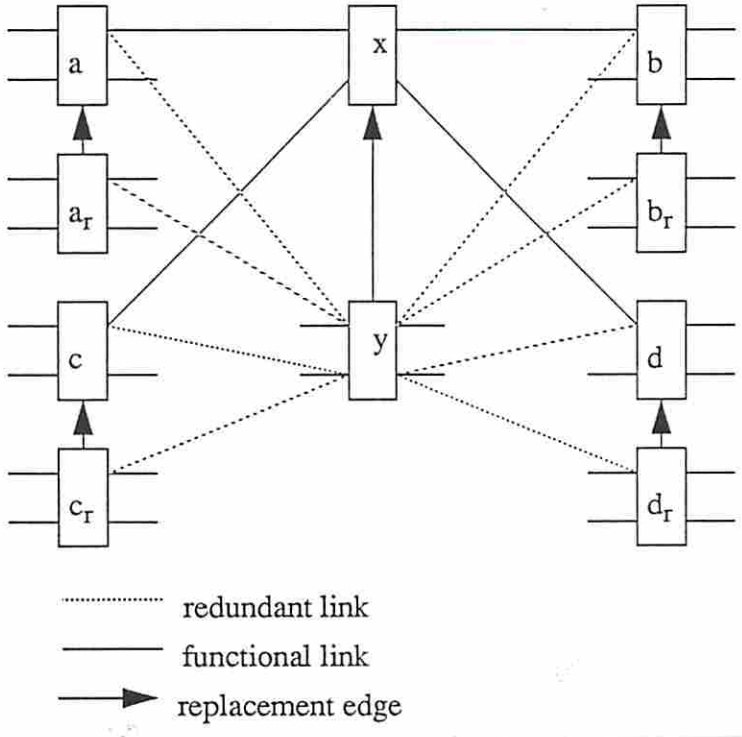


Figure 3.1: x is Replaceable by y

lines are the functional interconnections for the unique path MIN while the dotted lines are the redundant links. The switch module (2, 0) is replaceable by the switch module (2, 1). The dotted line from the upper innode of switch module (2, 1) to the switch module (1, 0) is for the failure of switch module (2, 0) while the dotted line to the upper outnode of switch module (1, 1) is for the failure of both switch modules (2, 0) and (1, 0). The last dotted link to the lower node of switch module (1, 1) is for the replacement of switch module (1, 0) by the switch module (1, 1) (The switch module (2, 0) has not failed).

Similar arguments can be applied to the upper and lower nodes of all the switch modules. Hence, the switch module (2, 1) is replaceable by the switch module (2, 2), the switch module (2, 2) is replaceable by the switch module (2, 3), and so on. Edges of the replacement graph are shown with the arrow lines. The replacement graph for the above example is shown in Figure 3.3a. To simplify the notation, we remove the stage label (2) from now on. The switch modules 0 to 7 are functionally required since, together with the functional links (solid lines in Figure 3.2), they perform all the switching functions of that stage. Hence they have to be either live or replaced by some other switch module. Furthermore, all the switch modules, including the functional switch modules and the spare switch modules are available for the replacement process.

The replacement graph can be transformed into a bipartite graph as follows. Place all the *available switch modules* in one set, Y, and place all the *functional switch modules* in the other set, X. Draw the replacement edges from the available switch modules to the functional switch modules. Since each functional module can perform the switching functions for itself, an edge is included. For example, the switch module 1 can perform the switching function for itself and for the switch module 0, thus we have two edges (1, 0) and (1, 1) as shown in Figure 3.3b.

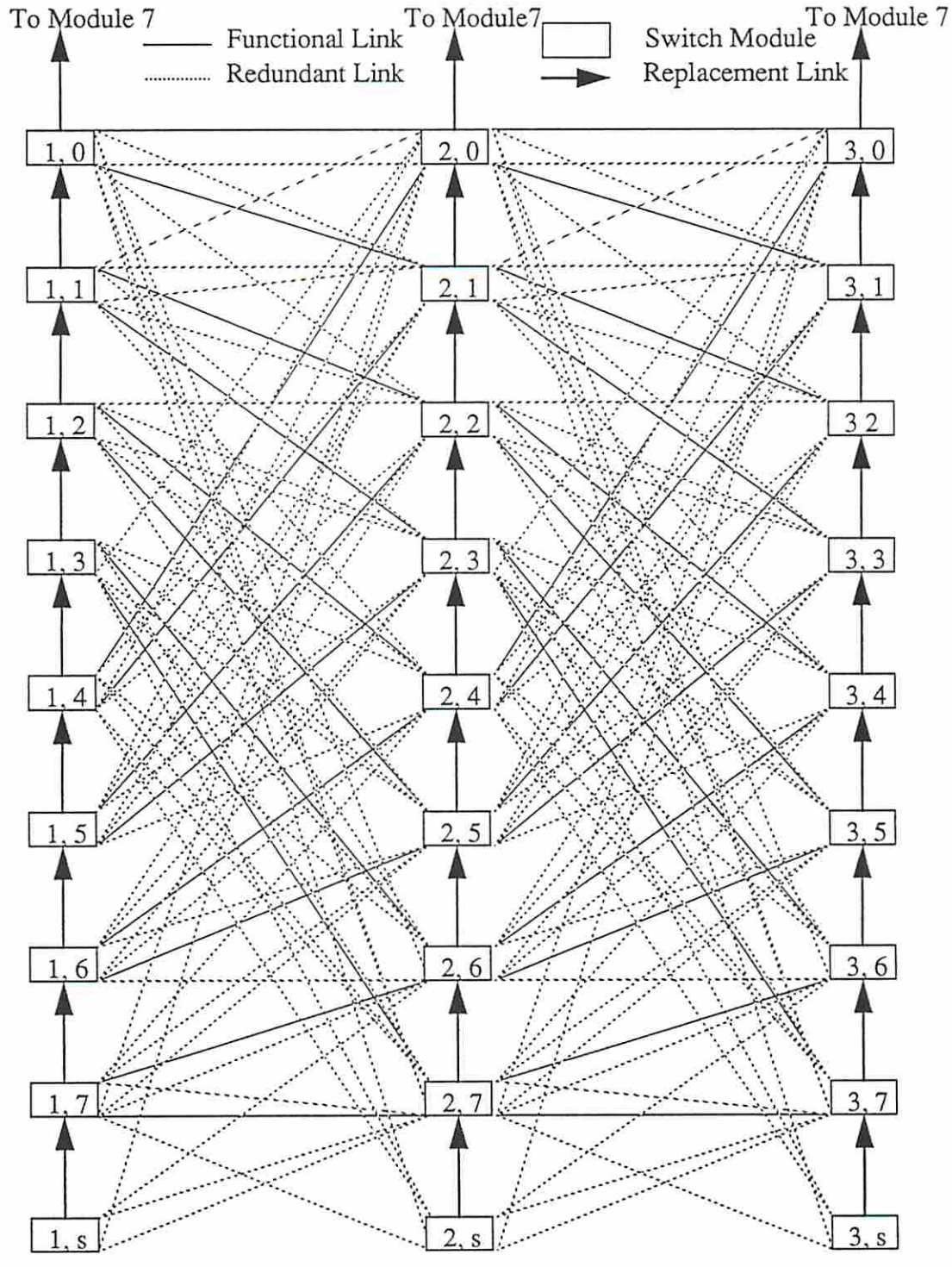


Figure 3.2: Redundant links for the unique path MIN

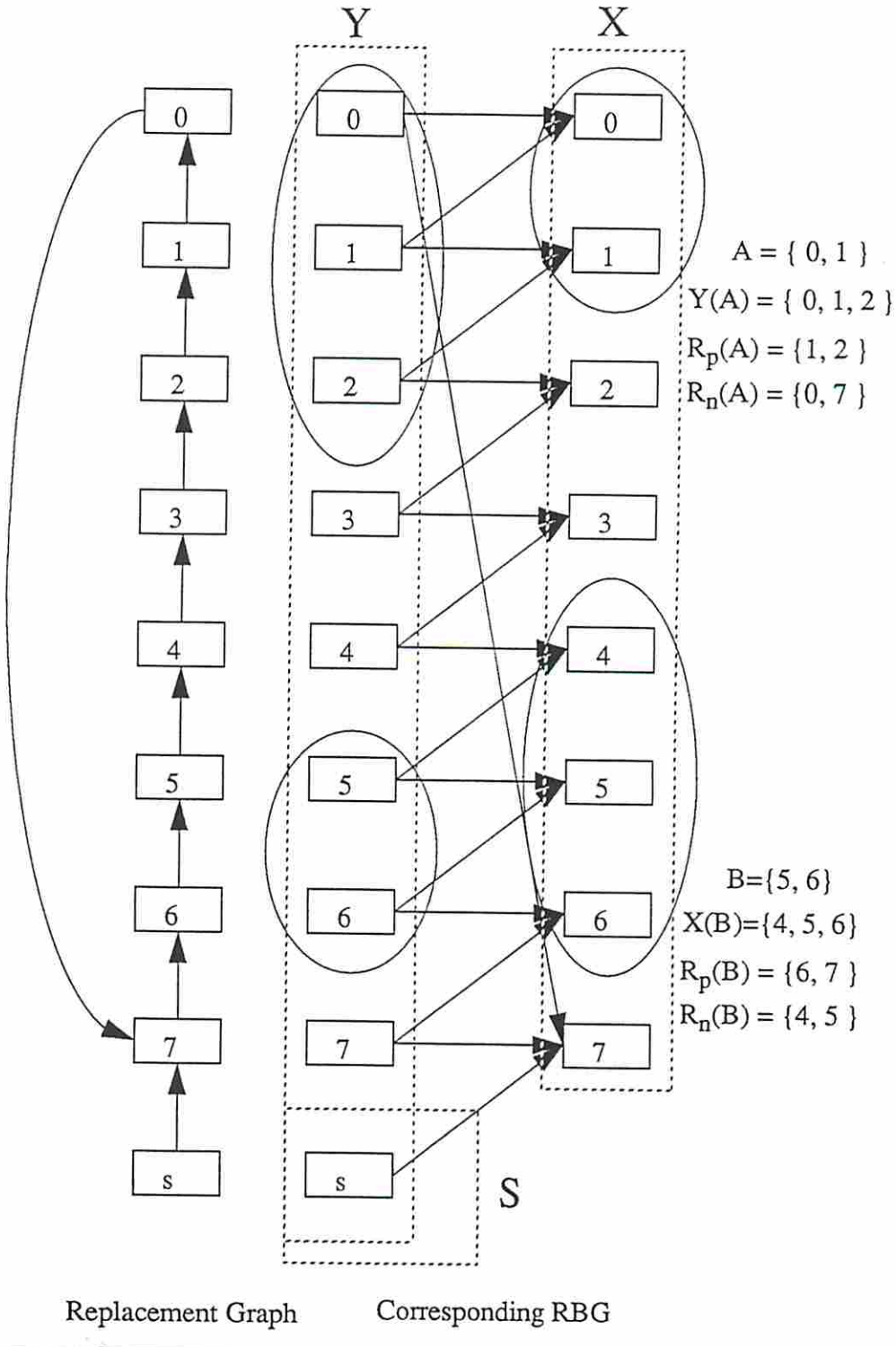


Figure 3.3: An Example of RBG

It is not hard to see that the reconfiguration of switch modules is related to the well known *Matching Problem* in graph theory [37]. A similar replacement graph (called the covering graph) was used to discuss the fault tolerance for a tree architecture multiprocessor system [32]. The concept of TI mode is also used. A different bipartite graph model has been used in the design of reconfigurable two dimensional VLSI arrays [65], where due to strong VLSI routing restrictions, a special dependent replacement model is used to model the two dimensional array replacement.

Let us formalize the bipartite graph model. Let X be the set of functional switch modules and Y be the set of available switch modules. Clearly, $X \subset Y$ and the set of *spare switch modules* is

$$S = Y - X$$

Definition 1 *A* Replacement Bipartite Graph *(RBG) is denoted by* $(Y,\ X,\ E)$ *such that the edge set*

$$E\ =\ \{(y,\ x)\ |x \in X \textit{ is replaceable by } y \in Y\}$$

□

An edge in E is called a *replacement edge.*

The matching problem for bipartite graphs has been studied extensively and many useful tools and results have been found. We adopt a terminology similar to that in [88]. A *matching* of the RBG is an edge subset $M \subset E$ such that no two edges of M share the same switch module. Edges in M are called *matched edges* and the other edges are called *free edges.* The switch modules that are incident upon a matched edge are called *matched switch modules* and the other switch modules are called *exposed switch modules.* A *complete matching* is a matching such that all the switch modules in X are matched. The complete matching can be used to characterize the fault tolerance as follows. Let a *Failure Set* $F \subset Y$ be the set of all failed switch modules and E be the remaining replacement edges after removing all the failure switch modules.

Lemma 1 *A failure set* F *is Fault Tolerable if and only if there exists a complete matching of the RBG:* $(Y - F,\ X,\ E)$. □

Proof:
If there exists a complete matching, then there is a replacement edge $(y\ , x)$ for every $x \in X$ with distinct $y \in Y - F$. Replace x with y and the reconfigured network is functional.

If the network is fault tolerable, then there exists a set $Z \subset Y - F$ such that a replacement exists for all functional switch modules in X. Place replacement edges from X to Z accordingly and it is a complete matching. QED

Based on the RBG example in Figure 3.3b, a fault tolerable failure set $\{1\}$ and the corresponding complete matching are shown in Figure 3.4.

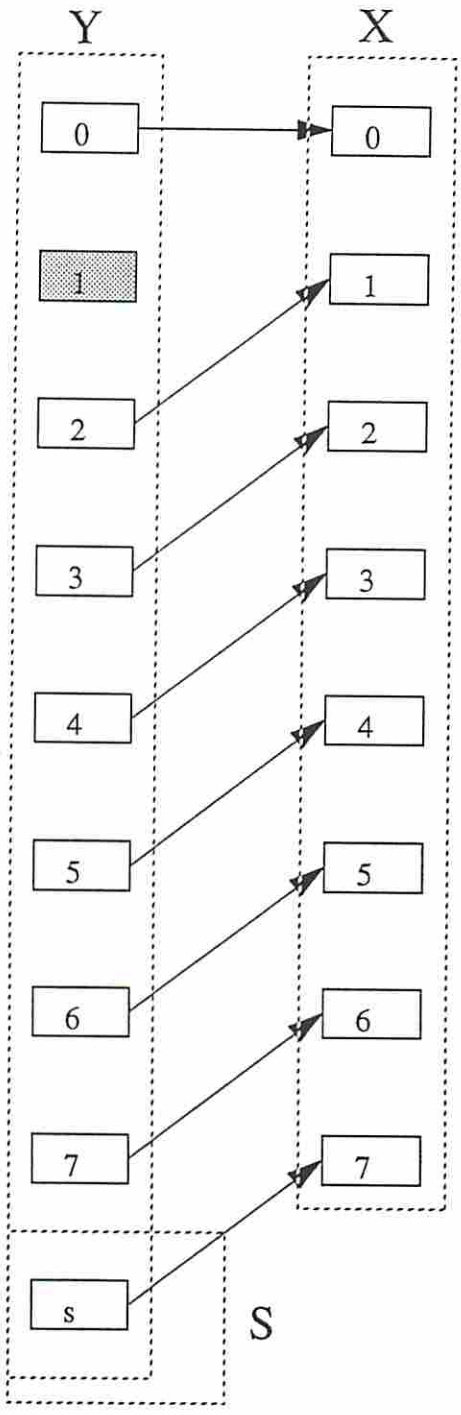


Figure 3.4: Complete Matching for a Failure set {1}

3.2 Maximum Fault Tolerance

An important system performance index is the maximum fault tolerance, since this specifies the maximum number of failures for which the network is guaranteed to be functional. Also, the switch module failure probability is usually very small and hence states with a small number of failures are the dominant terms in calculating system reliability.

Definition 2 *The maximum fault tolerance of a set of switch modules is* t *if*

1. *A failure set* F *is fault tolerable* $\forall |F| \leq t$.
2. $\exists F; |F| = t + 1$ *which is not fault tolerable.* □

For a RBG: $(Y,\ X,\ E)$, let A be a subset of Y, then the mapping set $X(A)$ is defined by:

$$X(A) = \{x | \exists y \in A \text{ such that } (y,\ x) \in E\ \}$$

Similarly, let A be a subset of X, then the mapping set $Y(A)$ is defined by:

$$Y(A) = \{y | \exists x \in A \text{ such that } (y,\ x) \in E\ \}$$

Examples from Figure 3.3b are:

$$\begin{aligned} Y(\{0,\ 1\}) &= \{0,\ 1,\ 2\} \\ X(\{5,\ 6\}) &= \{4,\ 5,\ 6\} \end{aligned}$$

The conditions for the existence of a complete matching (in our context) are repeated here without proof [37].

Complete matching theorem: [37]

A bipartite graph $(Y,\ X,\ E)$ has a complete matching if and only if

$$|Y(A)| \geq |A| \qquad \forall A \subset X$$

□

The condition for the maximum fault tolerance of a RBG can then be easily derived from the Complete matching theorem. It is essentially an extension of the Complete matching theorem.

Theorem 1 *The maximum fault tolerance of a* $RBG(Y,\ X,\ E)$ *is*

$$t = \min_{A \subset X}(|Y(A)| - |A|)$$

□

Proof:

Let

$$a = \min_{A \subset X}(|Y(A)| - |A|)$$

There exists a nonempty subset $A_1 \subset X$, such that

$$\begin{aligned} |Y(A_1)| - |A_1| &= a \\ |Y(A_1)| &= a + |A_1| \geq a + 1 \end{aligned}$$

Hence, there exists a failure set $F \subset Y(A_1)$ of size $|F| = a + 1$. For the new $RBG(Y - F,\ X,\ E)$,

$$\begin{aligned} |(Y - F)(A_1)| - |A_1| &= |Y(A_1)| - |F| - |A_1| \\ &= -1 \end{aligned}$$

which violates the Complete matching theorem. Therefore, by the definition of maximum fault tolerance:

$$t \leq a$$

Consider an arbitrary failure set F such that

$$|F| \leq a$$

then,

$$\begin{aligned} |Y(A)| - |Y(A) \cap F| &\geq |Y(A)| - a \\ &\geq |A| \qquad\qquad \forall A \subset X \end{aligned}$$

From the Complete matching theorem, there is a complete matching from X to $Y - F$. This concludes this theorem. QED

It is interesting to determine the achievable maximum fault tolerance for some given design parameters. Suppose the replacement capability of a switch module is r_n, i.e. it can replace at most r_n switch modules other than itself. r_n specifies the amount of reconfiguration overhead that a module has. Suppose the number of spare switch modules is s, which defines the cost overhead in terms of switch module redundancy.

Corollary 1 *The maximum fault tolerance is:*

$$t \leq \min\left(s,\ \left\lfloor \frac{r_n(s + |X|)}{|X|} \right\rfloor\right) \tag{3.1}$$

□

Proof:

By Theorem 1 and the fact that $A = X$, we have:

$$\begin{aligned} Y(X) &= Y \\ t &\leq |Y(X)| - |X| \\ &= s \end{aligned}$$

Since the spares do not have links to themselves, there are at most

$$r_n s + (r_n + 1)|X|$$

reconfigurable links from the set Y. There exists at least one switch $x \in X$ such that

$$Y(\{x\}) \leq \left\lfloor \frac{r_n s + (r_n + 1)|X|}{|X|} \right\rfloor$$

Then,

$$\begin{aligned} |Y(\{x\})| - |\{x\}| &\leq \left\lfloor \frac{r_n s + (r_n + 1)|X|}{|X|} \right\rfloor - 1 \\ &= \left\lfloor \frac{r_n(s + |X|)}{|X|} \right\rfloor \end{aligned}$$

QED

Although not all the designs have a maximum fault tolerance as in Equation 3.1, it is not difficult to find one for a given set of design constraints. Usually, maximum fault tolerance is achieved when the failures are uniformly tolerated and a regular fault tolerant structure is provided. In addition to providing higher fault tolerance, a regular structure is easier to realize.

3.3 A Fast Run Time Replacement Algorithm

For a given failure set, it is desirable to know if it is fault tolerable and what the replacements are. From Lemma 1, a trivial replacement algorithm is to find a complete matching, which thus answers both questions. There are polynomial time algorithms for finding the complete matching of a bipartite graph. Essentially, the fastest known algorithm is $O(|X + Y|^{1/2} \cdot |E|)$ [88].

It seems that we have the solution and we may not do better. Yet, let us look at the failure and replacement process more closely. The time between two failures must be much longer than the reconfiguration time for a failure since otherwise the network is very unreliable. The most critical timing is between the *occurrence* of a new failure and the *completion* of the replacement process which is the *down time* for the network. It directly affects the service quality of ATM network.

When a failure occurs, a centralized *configuration processor* computes a new complete matching and distributes the reconfiguration commands to all the switch modules. Each switch module then reconfigures its own interconnections accordingly. Due to the advances in VLSI technology, it is possible to have some computing power within each switch module. It is better to parallelize the on-line computations as much as possible. In [41], an efficient parallel algorithm that has speedup factor (in terms of the best known serial algorithm) almost equal to the number of parallel processors is given for the bipartite matching problem. We concentrate only on the on-line computing time in the following discussion and derive a better on-line parallel replacement algorithm. The replacement process is divided into four phases:

1. The *initial phase* with initial time T_i:
   The configuration processor prepares the algorithm initially.
2. The *on-line computing phase* with on-line computing time T_c:
   The configuration processor computes and distributes the reconfiguration commands to the live switch modules for a reconfiguration.
3. The *switch module computing phase* with switch module computing time T_m:
   Each switch module locally reconfigures its own interconnections.
4. The *off-line preparation phase* with off-line preparation time T_p: The configuration processor prepares the network for the next failure.

Our goal is then to find an algorithm that minimizes T_c and T_m (since T_i and T_p are concurrent with normal switch operation). In order to proceed, we define an auxiliary graph.

Definition 3 *The* Auxiliary Digraph $G(R,\ M) = (Y,\ Z)$ *for a RBG:* $R = (Y,\ X,\ E)$ *and a complete matching* M *has edges*

$$\forall\ (u,\ v) \in Z,\ \exists x \in X \text{ such that } (u,\ x) \in E - M \text{ and } (v,\ x) \in M$$

□

The edge $(v,\ x) \in M$ indicates that the switch module v is currently performing the switching functions for the functional switch module x. On the other hand, the edge $(u,\ x) \in E - M$ indicates that the switch module v is capable of performing the switching functions for x. Therefore, once the switch module v fails (or it is used to replace some other switch module), the switch module u can assume its role and perform the switching functions for x.

Consider the RBG shown in Figure 3.5a. The dotted arrow lines in the RBG are the complete matching. Since two solid lines (replacement edges) of switch module 1 in Y are connected to the switch modules 0 and 7 in X (which are matched to the switch modules 0 and 7 in Y), therefore, (1, 0), (1, 7) are two edges of the auxiliary digraph in Figure 3.5b. Replacements can be characterized by the following theorem.

Theorem 2 *Let* $G = (Y - F,\ Z)$ *be the auxiliary digraph of the RBG* $(Y - F,\ X,\ E)$ *and the complete matching* M *for a fault tolerable failure set* F. *A new failure* $x \in Y - F$ *is fault tolerable if and only if it is reachable from an exposed switch module in* G *or itself is an exposed module.* □

Proof:

Assume that x is reachable from an exposed switch module p_1 along the path $p_1,\ p_2, ...,\ p_k$ where $x = p_k$. For the edge (p_i, p_{i+1}) of the path, there exist a functional switch module q_i such that $(p_i,\ q_i)$ is a replacement edge of the RBG and $(p_{i+1},\ q_i)$ is an edge of the complete matching M. For the edges $(p_{i+1},\ q_i)$ in the complete matching, replace them with the edges $(p_i,\ q_i)$. Since p_1 is an exposed switch module, the replaced edges form a complete matching.

Assume that x is fault tolerable. Then, there exist a complete matching M' without the module x. If x is an exposed switch module, we conclude the theorem. If it is not, let the switch module x_1 be the matched module of x in M. Since x is not in

M', the switch module y matched with x_1 in M' is different from x. Then $(y,\ x)$ is an edge in the auxiliary digraph. If y is not an exposed switch module, the switch module y_1 it matches in M must be different from x_1. Similarly, the switch module z that is matched with y_1 in M' is different from the switch module y or x and hence $(z,\ y)$ is an edge in auxiliary digraph. Since x is not in M', there is at least one exposed switch module in M'. By repeating the above argument, a path to an exposed switch module can be established. QED

A *reachable auxiliary digraph* is the subgraph of an auxiliary digraph obtained by deleting all the switch modules (and the associated edges) that are not reachable from any exposed switch module. A direct result from Theorem 2 is that the next single failure is fault tolerable if and only if it is in the reachable auxiliary digraph. An easy way to traverse all of the switch modules of the reachable auxiliary digraph is the Depth First Search (DFS) from the exposed switch modules which results in the spanning forest. For the reconfiguration of an arbitrary single failure, we only need to find a single path from an exposed switch module for each switch module, and therefore, a spanning forest of the auxiliary digraph is enough for finding the replacements. Since the exposed switch modules are reachable only by themselves, the roots of spanning forest are exactly the exposed switch modules.

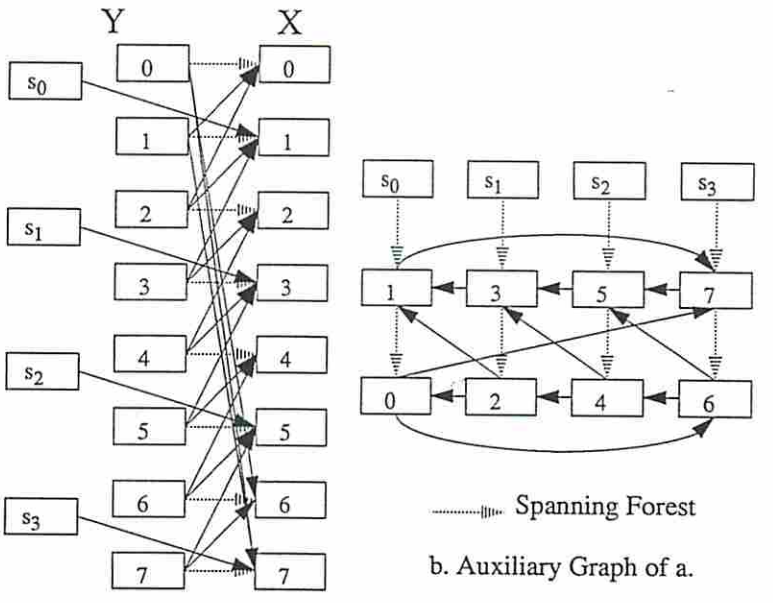
A spanning forest is shown in Figure 3.5b by the dotted arrow lines. When a failure occurs, the path of its parent switch modules up to the root which is an exposed switch module can be used for the replacements. An example is shown in Figure 3.6a and Figure 3.6b for the failed switch of module 4. The switch module 5 replaces the switch module 4 by performing the switching functions for it as shown by an edge in the spanning forest in Figure 3.5b or a dotted arrow edge in Figure 3.6a. Similarly, the spared switch module s_2 replaces the switch module 5. After the reconfiguration, a new auxiliary digraph is formed which is very different from the original auxiliary digraph. The complete matchings and auxiliary digraphs for additional failure 5 and 7 are shown in Figure 3.7 and Figure 3.8 respectively.

For a given spanning forest, each switch module needs to decide if a failure is its descendant and under which of its children so that it can replace this child for a reconfiguration. The DFS algorithm traverses through all the switch modules one by one, with the descendants of a switch module being traversed after it. All of the descendants of a sibling are traversed before the next sibling is traversed. Hence, a labeling of DFS sequence can be used to decide the replacements. For each switch module, remember its label and the label of its next sibling. If the label of the failure is between the two labels of a switch module's child, then this child needs to be replaced by its parent.

In summary, the replacement algorithm finds a spanning forest of the reachable auxiliary digraph rooted at the exposed switch modules and the corresponding DFS labeling off-line. Each switch module stores its label and the label of its next sibling (or the label of its parent's next sibling if it is the last child). When a failure occurs, the switch module for which the label of the failed switch module is between the two labels it has stored is replaced by its parent switch module. Details of the algorithms are given below. There are three arrays: $REACHABLE(x)$ indicates whether the switch module x is reachable from an exposed switch module, $LABEL(x)$ holds the DFS label of x, and $NEXT.LABEL(x)$ holds the label of its next sibling.

Algorithm 1 *Initial phase algorithm.*

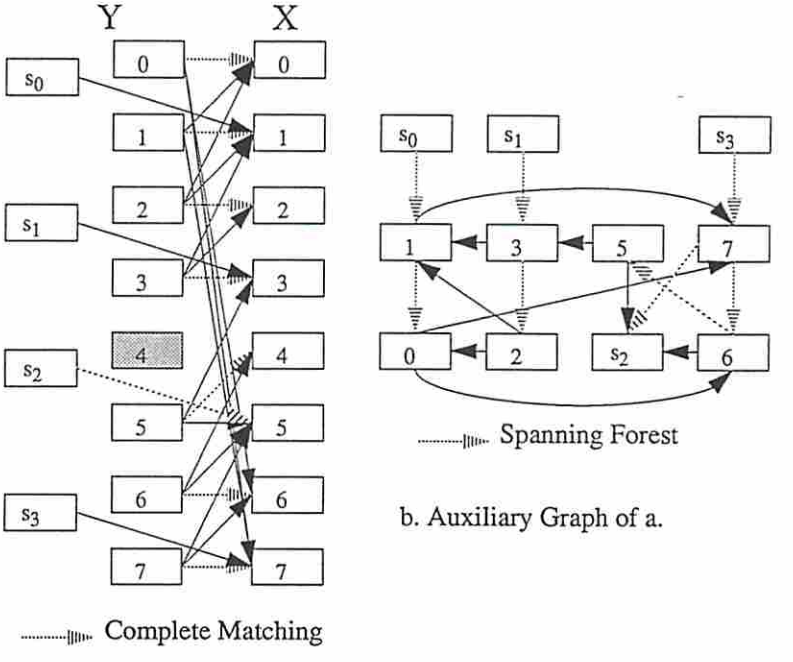
Given: $RBG\ (Y,\ X,\ E)$.
Algorithm:



a. Example of a RBG with no Failures

b. Auxiliary Graph of a.

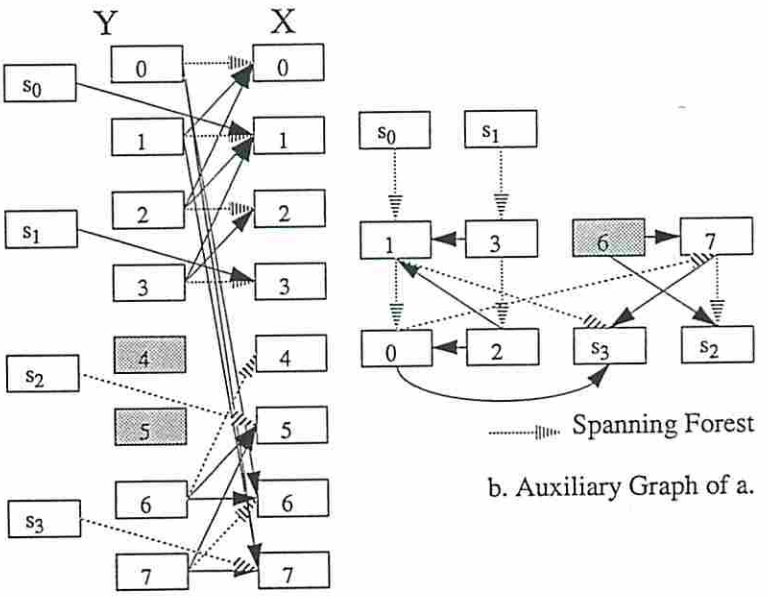
Figure 3.5: A RBG and its Auxiliary Digraph



a. Example of a RBG with 1 Failure

b. Auxiliary Graph of a.

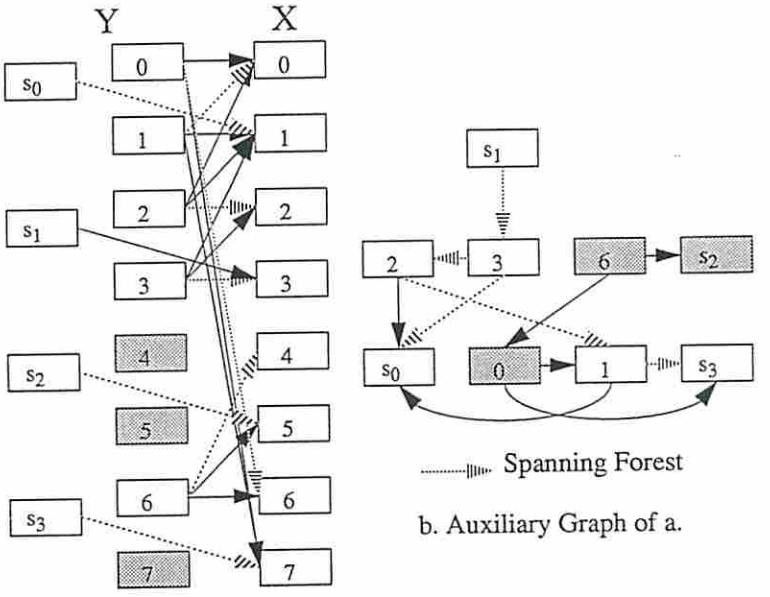
Figure 3.6: A RBG and its Auxiliary Digraph with Failure Set {4}



a. Example of a RBG with 2 Failures

b. Auxiliary Graph of a.

Figure 3.7: A RBG and its Auxiliary Digraph with Failure Set {4, 5}



a. Example of a RBG with 3 Failures

b. Auxiliary Graph of a.

Figure 3.8: A RBG and its Auxiliary Digraph with Failure Set {4, 5, 7}

1. *Since there are no failures initially, the edges corresponding to the identity map from X to Y form a complete matching. The spare switch modules are exposed switch modules.*
2. *With the above complete matching, use Algorithm 3 to construct the spanning forest of the reachable auxiliary digraph.*

Algorithm 2 *Switch module computing phase algorithm.*

Given: *a RBG (Y, X, E), a fault tolerable set F with a complete matching M and a (new) failure f.*
Question: *Find a new complete matching M' if f is fault tolerable.*
Algorithm: *For every module u:*

1. *If $REACHABLE(f) = false$, f is not fault tolerable.*
   Otherwise, search $LABEL(f)$ from the children of u.
2. *For the child v and its matched switch module y in M such that*

$$LABEL(v) \leq LABEL(f) < NEXT.LABEL(v)$$

   match u to y in M'.
   Otherwise, no change.

Algorithm 3 *Off-line preparation phase algorithm.*

Given: *a RBG $(Y - F, X, E)$, a complete matching M and a fault tolerable failure set F.*
Question: *Establish the auxiliary digraph $(Y - F, Z)$ and the DFS spanning forest.*
Algorithm:

1. *Auxiliary graph: For each replacement edge (u, x) not in M, add an edge (u, v) to Z if (v, x) is in M.*
2. *Spanning tree: Starting from each exposed switch module, use the Depth First Search (DFS) algorithm to build the spanning forest and label the forest according to its sequence [50]. Denote the label of a switch module x as $LABEL(x)$. For each switch module, remember the next sibling label of the same switch module by $NEXT.LABEL(x)$.*
3. *If a switch module x is not reachable, set $REACHABLE(x) = false$. Otherwise, set $REACHABLE(x) = true$.*

There are three arrays of size $|Y|$ required in the algorithms, hence the storage requirement is $O(|Y|)$. Let the degree of the spanning forest be L which is bounded by the number of replacement links per switch module. During the on-line computation phase, the centralized processor needs only to broadcast the label of the failed module to all the switch modules. During the module computing phase, the worst case is the search algorithm in line 2 which can be easily implemented with a binary search in $O(\log L)$ time. Hence,

$$\begin{aligned} T_c &: O(1) \\ T_m &: O(\log L) \end{aligned}$$

During the initial phase or the off-line preparation phase, the time is determined by the computation of the auxiliary graph or DFS algorithm. Hence,

$$T_i,\ T_p: \quad O(max(|Y|,\ |E|)$$

We have proposed a fast ($O(\log L)$) run time parallel replacement algorithm which impacts the service quality of ATM network directly. Although the degree of the spanning forest L is $O(|E|)$ for an arbitrary design (which is only possible for an extremely irregular design), it is usually a very small number for regular designs. The reasons are:

1. The number of replacement edges is restricted by the link redundancy (or the cost overhead) of a switch module.
2. The replacement edges are uniformly distributed to all the switch modules (or to the best) for a regular design.
3. It is the degree of the spanning forest, not of the auxiliary digraph that has more edges.
4. The DFS tree tentatively has much smaller degree (more levels) on the average.

It is also very easy to implement the $O(|Y|)$ (linear) storage for the centralized reconfiguration processor and the $O(L)$ storage for each switch module.

3.4 Link Requirement

The replacement model has been shown to be an effective tool to analyze the relationship between the reconfigurable links and spares. We must now determine how many reconfigurable links are required and how they should be arranged to provide the desired replacement edges for a particular class of functional topology designs. If the number of links can be minimized so that they are organized similar to physical interconnections, more useful replacement edges are available leading to a more reliable network. The functional links are functional interconnections for both the non-failure mode or failure mode of operation and, therefore, are the basis for reconfiguration.

To simplify our discussion, a network that consists of interconnected 2×2 cross bar switch modules is assumed. However, it is easy to extend the results to a larger switch module. There are two innodes: (upper and lower) and two outnodes: (upper and lower) for each switch module. Let A be a set of functional switch modules in X.

Definition 4 *The node sets are defined as follows:*

1. *Backward upper node set: $B_{bu}(A)$ = Set of outnodes that link to an upper innode of a switch module in A with an operational link during non-failure mode.*
2. *Backward lower node set: $B_{bd}(A)$ = Set of outnodes that link to an lower innode of a switch module in A with an operational link during non-failure mode.*
3. *Forward upper node set: $B_{fu}(A)$ = Set of innodes that link to an upper outnode of a switch module in A with an operational link during non-failure mode.*
4. *Forward lower node set: $B_{fd}(A)$ = Set of innodes that link to an lower outnode of a switch module in A with an operational link during non-failure mode.* □

For each switch module, there are switch modules that are able to replace it and switch modules that are potentially replaced by it.

Definition 5 *The replacement sets are defined as follows:*

1. *Previous replacement set:* $R_p(A)$ = *Set of switch modules that can replace a switch module in* A.
2. *Next replacement set:* $R_n(A)$ = *Set of switch modules that are replaceable by a switch module in* A. □

Examples can be found from Figure 3.3:

$$\begin{aligned} R_p(\{0,\ 1\}) &= \{1,\ 2\} \\ R_n(\{0,\ 1\}) &= \{0,\ 7\} \end{aligned}$$

Since each innode or outnode of a switch module is replaced by the corresponding node of its replacement switch module, we can apply the above definition to nodes without ambiguity. For example: if A is a set of upper innodes, $R_p(A)$ and $R_n(A)$ are the corresponding upper innodes of the replacement switch module sets.

Recall that the replacement capability of a single switch module is r_n which is the maximum number of switch modules that it can replace. Similarly, the maximum size of the previous set for a single switch module is r_p. If a switch module x is replaceable by a switch module y, a minimum requirement is that switch module y must have reconfigurable links to all innodes and outnodes functionally connected to switch module x. However, if a switch module a which was connected to x fails, it is replaced by another switch module and the original functional link is reconfigured to another reconfigurable link. To guarantee that x is still replaceable by y under other reconfigurable failures, more reconfigurable links are required, in particular y must have a link to a's replacement

Theorem 3 *For a single switch module set* $A = \{a\}$, *link requirements for its innodes and outnodes are*

$$\begin{aligned} L_{ij}(A) &= B_{ij}(A) \cup R_p(B_{ij}(A)) \cup B_{ij}(R_n(A)) \cup R_p(B_{ij}(R_n(A))) \\ &\qquad \forall i \in \{b,\ f\},\ j \in \{u,\ d\} \end{aligned}$$

□

Proof:

In Figure 3.9, consider all possible failures that may cause reconfiguration to A.

1. Normal mode of operation: All nodes in the set : $B_{ij}(A)$ require a functional link to A.
2. Switch modules in $B_{ij}(A)$ failed: Switch modules in the previous replacement set: $R_p(B_{ij}(A))$ are required to replace the failed switch modules.
3. Switch modules in the next replacement set failed: A may be needed to replace one of the failed switch modules. All switch modules that have a reconfigurable link, functional or reredundant link, to that replaced switch module need to be linked to switch module A. With the same reasoning as in the previous two cases, nodes in $B_{ij}(R_n(A))$ and $R_p(B_{ij}(R_n(A)))$ require a reconfigurable link to A.

Other failures will not affect the link requirement set, hence, we conclude the theorem. QED

Base on the unique path MIN shown in Figure 2.10, the reconfigurable links required for the replacement graph of Figure 3.2 can be derived. Dotted links are the reconfigurable links that support the desired replacement graph. Note that the number of reconfigurable links is 4 which is the upper bound for $r_n = r_p = 1$. It is also interesting to find upper and lower bounds on the link requirements in terms of r_n and r_p. They are given by following corollary.

Corollary 2 *The number of links required for a node of a switch module is*

$$r_n + 1 \leq l_{req} \leq (r_p + 1)(r_n + 1)$$

□

Proof:

Let us evaluate the size of $L_{bu}(\{a\})$ for a single switch module a.

$$\begin{aligned} |B_{bu}(A)| &= 1 \\ |R_p(B_{bu}(A))| &\leq r_p \\ |B_{bu}(R_n(A))| &\leq r_n \\ |R_p(B_{bu}(R_n(A)))| &\leq r_p r_n \\ |A \cup R_n(A)| &\geq r_n + 1 \end{aligned}$$

Therefore,

$$r_n + 1 \leq |L_{bu}(A)| \leq (r_p + 1)(r_n + 1)$$

Similarly, bounds based on the other three equations can be found.

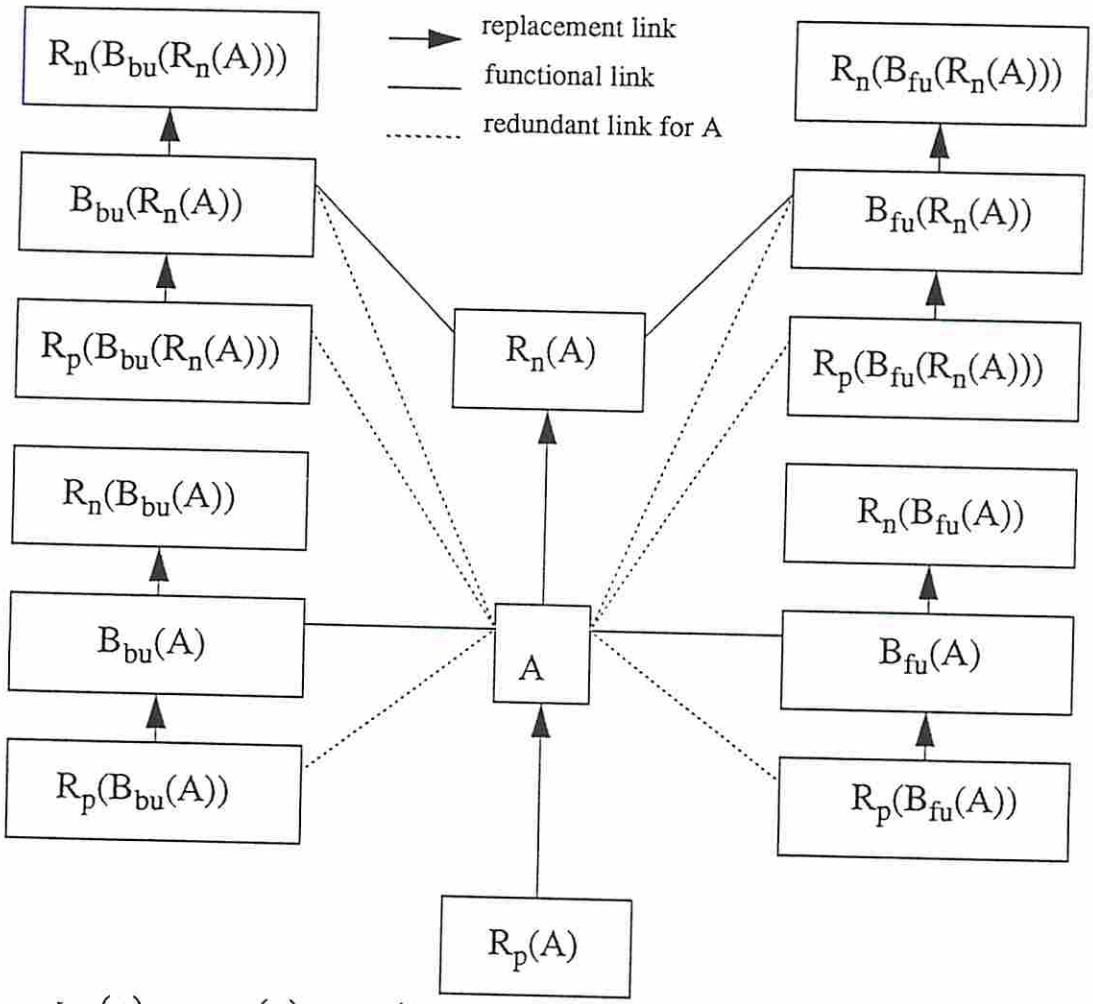
For a switch module of an RBG such that elements of the previous sets of $B_{bu}(A \cup R_n(A))$ are all distinct, the upper bound holds with equality.

If there are adequate reconfigurable links, the RBG is fully connected and $r_n = |X| - 1$ Hence, the lower bound holds with equality. QED

3.5 Two Design Examples

No physical device constraints have been assumed so far. Due to different interconnection implementations, the achievable maximum fault tolerance and reliability may be totally different. In this section, the maximum fault tolerance and reliability of different implementations are considered versus the hardware overhead. Switch modules which are active devices are usually more expensive and less reliable. On the other hand, passive links are usually cheaper and less likely to fail, therefore, they are not considered in the analysis. It is assumed that the cost is the same for every switch module.

The Omega network [69] with 2×2 switch modules is used as an example for the functional topology. The results can be easily extended to many other useful interconnection networks, for example the internally non-blocking Batcher Banyan network [10]. The network in Figure 2.10 is an example for the Omega network. Switch modules in each stage are labeled from 0 to $\frac{N}{2} - 1$ such that:



$$L_{fu}(A) = B_{fu}(A) \cup R_p(B_{fu}(A)) \cup B_{fu}(R_n(A)) \cup R_p(B_{fu}(R_n(A)))$$

$$L_{fd}(A) = B_{fd}(A) \cup R_p(B_{fd}(A)) \cup B_{fd}(R_n(A)) \cup R_p(B_{fd}(R_n(A)))$$

$$L_{bu}(A) = B_{bu}(A) \cup R_p(B_{bu}(A)) \cup B_{bu}(R_n(A)) \cup R_p(B_{bu}(R_n(A)))$$

$$L_{bd}(A) = B_{bd}(A) \cup R_p(B_{bd}(A)) \cup B_{bd}(R_n(A)) \cup R_p(B_{bd}(R_n(A)))$$

Figure 3.9: Upper Node Link Requirement Diagram

1. The upper outnode of switch module i is connected to the upper innode of switch module $2i$ in the next stage for $i < \frac{N}{4}$.
2. The lower outnode of switch module i is connected to the upper innode of switch module $2i+1$ in the next stage for $i < \frac{N}{4}$.
3. The upper outnode of switch module $i + \frac{N}{4}$ is connected to the lower innode of switch module $2i$ in the next stage for $i < \frac{N}{4}$.
4. The lower outnode of switch module $i + \frac{N}{4}$ is connected to the lower innode of switch module $2i+1$ in the next stage for $i < \frac{N}{4}$.

3.5.1 Arbitrary Replacement Interconnection

For a centralized network, the links are typically short and hence multiple physical links are possible. An implementation is shown in Figure 3.10a. The DMUX attached to a outnode selects a functional link. Appropriate selections are made such that only one link of the MUX of an innode is functional. For an electronic interconnection implementation, simple off-line (reconfiguration time) selectable MUXs, DMUXs and redundant links are required. For an optical interconnection implementation, lasers with electronically selectable optical paths are required [90]. There is a non-negligible overhead for both designs but they preserve the functional topology under failure mode which is very important to a high speed switching network design.

The logical model for an *arbitrary* replacement interconnection is that each outnode has l links, which can be interconnected (reconfigured) to an innode of *any* selected switch module. Similarly, each innode has l incoming links, which can be interconnected (reconfigured) to an outnode of *any* selected switch module. There are no constraints on which outnode is connected to which innode. The only restriction is the number of reconfigurable links that each node has.

In section 3.2, an upper bound on the maximum fault tolerance was given based only on r_n, and the limitations due to r_p were not considered. In reality, both the next and previous replacement sets are restricted by the number of available links l. Designs considered in this section sacrifice some degree of maximum fault tolerance but pay more attention to balancing the limits of both replacement set constraints.

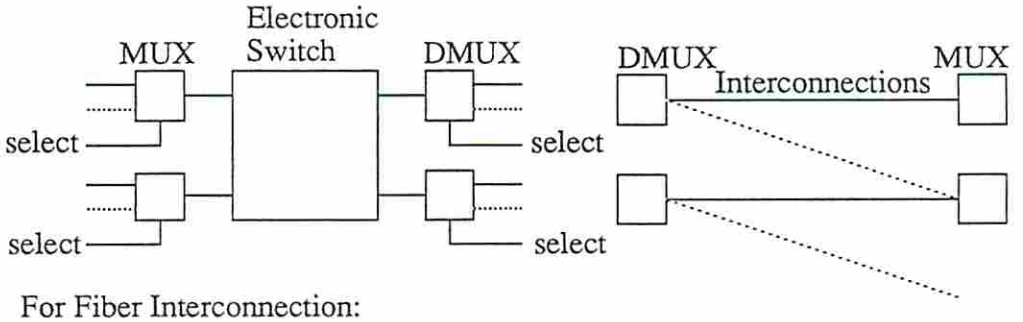
Since the switch modules in the functional topology are organized in stages and are structured similarly at each stage, it is better to design the replacements accordly. To simplify the network replacement, it is assumed that the spares are equally distributed to all the stages and that the number of spares per stage $s_s = s/\log N$ is an integer. Each group of s_s switch modules can replace the switch modules in the same stage, i.e., cross stage replacement is not allowed.

We label the functional switch modules according to the Omega labeling described above and label the spares in each stage sequentially from 0 to s_s. Assume that there are r_n replacement edges for each functional switch module and there are k replacement edges per spare, $1 \leq k \leq r_n$ This reconfigurable network is referred as a *Stagewise Arbitrary Replacement Interconnection* with k links per spare ($SARI/k$). Let

$$a = \frac{N}{2ks_s} \qquad if\ ks_s \leq N/2$$

$$b = \frac{2ks_s}{N} \qquad if\ ks_s \geq N/2$$

a) Arbitrary Replacement Interconnection



For Fiber Interconnection:

DMUX=Optic Transmitter+Power Splitter

MUX=Optic Receiver+Star Coupler

b) Star Coupler Interconnections

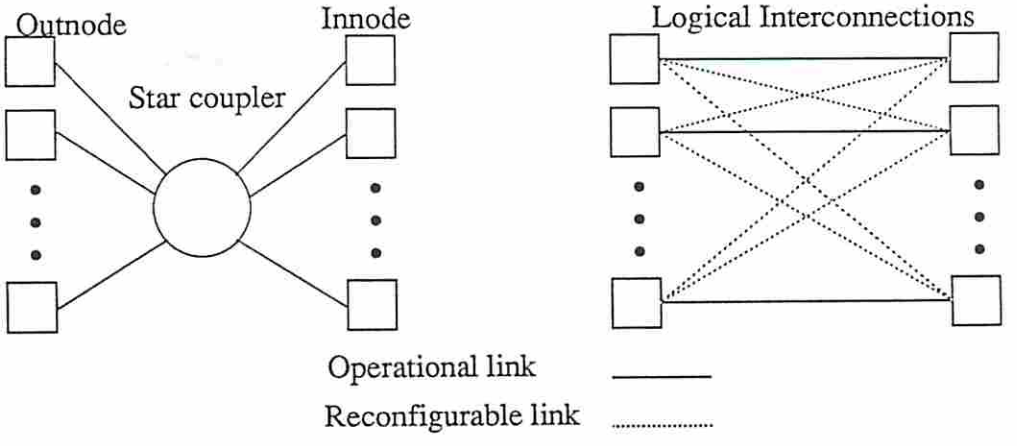


Figure 3.10: Two Redundant Interconnection Implementations

$$c = \frac{N}{2s_s}$$

For a regular design, we restrict the above parameters to be integers when applicable and $s_s \leq N/2$.

Let the *mod* function be the division remainder. Then the replacement edges of the SARI/k are as follows:

1. As shown in Figure 3.11, the $r_n + 1$ replacement edges of an *functional switch module* x are

$$(x,\ x),\ (x,\ (x+1)mod(\frac{N}{2})),\ ..,\ (x,\ (x+r_n)mod(\frac{N}{2}))$$

2. Assign replacement edges of the *spare switch modules* according to the size of ks_s:

   (a) If $ks_s \leq N/2$:
   As shown in Figure 3.12, replacement edge m of spare i is connected to functional switch module $(ia + ms_s a)mod(N/2)$, i.e.

$$(i,\ m) \leftrightarrow (ia + ms_s a)mod(N/2) \qquad \begin{aligned} \forall m &\in (0..k-1) \\ i &\in (0..s_s - 1) \end{aligned}$$

   (b) If $ks_s > N/2$:
   As shown in Figure 3.11, the k replacement edges of each switch module are grouped to b groups of c replacement edges.
   Replacement edge m of group j of spare i is connected to a switch module $(ms_s + i + j)mod(N/2)$, i.e.

$$(i,\ j,\ m) \leftrightarrow (ms_s + i + j) mod(N/2) \qquad \begin{aligned} \forall m &\in (0..c-1) \\ j &\in (0..b-1) \\ i &\in (0..s_s - 1) \end{aligned}$$

□

The replacement links for a switch due to functional switches are from itself and the r_n switches after it. One benefit for this design is that consecutive switches (according to the Omega labeling) share common switches in the previous and next replacement sets. This helps to reduce the number of required reconfigurable links. Note that for a group of q *consecutive* switches, there are $q + r_n$ consecutive operational switches in the previous or next replacement set.

For a small number of spares (case 2a), every a switches share one spare. Only the first one of each group has a link to a spare and is referred to as the marked switch as shown in Figure 3.12. For the larger number of spares (case 2b), replacement links of each switch are divided into b groups . Only replacement links in the first group are shown in Figure 3.13 and they cover all switches in X. The other groups are not shown for simplicity. They are connected to the switches in X by rotating one switch per group, i.e., the first replacement link of the second group is connected to the second switch in X. The last replacement link of this group is connected to the first switch. The third group begins with the third switch in X etc.

Theorem 4 *The maximum fault tolerance of SARI/k is*

$$t_{SARI/k} = min\left(s_s,\ r_n + \left\lfloor \frac{2ks_s}{N} \right\rfloor\right)$$

□

Proof:

Similar to the proof in 1, we can always find a set larger that s_s or $r_n + \left\lfloor \frac{2ks_s}{N} \right\rfloor$ that fails the network. Hence,

$$t_{SARI/k} \leq min\left(s_s,\ r_n + \left\lfloor \frac{2ks_s}{N} \right\rfloor\right)$$

Let A be an arbitrary switch module set and

$$\begin{aligned} |A| &= q \\ Y_x(A) &= Y(A) \cap (Y - S) \\ Y_s(A) &= Y(A) \cap S \\ Y(A) &= Y_x(A) \cup Y_s(A) \end{aligned}$$

$|Y_x(A)|$ due to a non-consecutive A is always greater than that due to the consecutive set of the same size, hence

$$\begin{aligned} |Y_x(A)| \geq q + r_n \qquad & if \quad q \leq |X| - r_n \\ Y_x(A) = X \qquad & if \quad q \geq |X| - r_n \end{aligned}$$

Note that for both cases, equality holds for all consecutive sets.

Consider the case that $ks_s \leq \frac{N}{2}$. For $q \leq |X| - r_n$, we have

$$\begin{aligned} |Y(A)| - |A| &\geq q + r_n - q + |Y_s(A)| \\ &\geq r_n \end{aligned}$$

For $q \geq |X| - r_n$, we have

$$\begin{aligned} |Y(A)| - |A| &\geq N/2 - q + \left\lfloor \frac{qs_s}{N/2} \right\rfloor \\ &= N/2 - q + \left\lfloor s_s - \frac{(N/2 - q)s_s}{N/2} \right\rfloor \\ &= s_s + (N/2 - q) - \left\lfloor (N/2 - q)\frac{s_s}{N/2} \right\rfloor \\ &\geq s_s \end{aligned}$$

Consider the case that $ks_s \geq \frac{N}{2}$. For $q \leq |X| - r_n$, we have

$$\begin{aligned} |Y(A)| - |A| &\geq q + r_n - q + b\left\lceil \frac{q}{c} \right\rceil \\ &\geq r_n + \left\lfloor \frac{2ks_s}{N} \right\rfloor \end{aligned}$$

For $\lceil \frac{q}{c} \rceil + b - 1 \geq s_s$ then all spares are in $Y_s(A)$ and we have

$$\begin{aligned} |Y(A)| - |A| &\geq q + r_n - q + s_s \\ &\geq s_s \end{aligned}$$

For the case that

$$\begin{aligned} q &\geq |X| - r_n \\ \left\lceil \frac{q}{c} \right\rceil + b - 1 &< s_s \end{aligned}$$

then

$$\begin{aligned} |Y(A)| - |A| &\geq \frac{q}{c} + b - 1 + N/2 - q \\ &= s_s + \frac{2(q-1)s_s}{N} + \left(\frac{N}{2} - q - 1\right)\left(1 - \frac{2s_s}{N}\right) \\ &\geq s_s \end{aligned}$$

From Theorem 1, we can conclude that

$$t_{SARI/k} \geq min\left(s_s,\ r_n + \left\lfloor \frac{2ks_s}{N} \right\rfloor\right)$$

QED

The link requirement of the SARI/k is derived in the following theorem.

Theorem 5 *An upper bound of the link requirement for SARI/k is*

$$\begin{aligned} l_{SARI/k} &\leq 3r_n + 1 + \min\left(s_s, \left\lceil r_n + \frac{2ks_s}{N} \right\rceil\right) \quad if\ ks_s \geq \frac{N}{2} \\ &\leq 3r_n + 1 + \min\left(s_s, \left\lceil \frac{2ks_s(r_n+1)}{N} \right\rceil\right) \quad if\ ks_s \leq \frac{N}{2} \end{aligned}$$

□

Proof:

According to Theorem 3 and referring to Figure 3.9, the switch modules in $R_n(A) \cup A$ for a single switch module set A are consecutive and are of size $r_n + 1$.

Consider the reconfigurable links due to the functional switch modules first as shown in Figure 3.11. The size of backward upper node set in the previous stage is

$$|B_{bu}(A) \cup B_{bu}(R_n(A))| = r_n + 1$$

Referring to Figure 2.10, the shuffle interconnections of Omega network require that the backward upper node set is divided into two subsets: the subset for upper outnodes and the subset for lower outnodes. The switch sets containing these two subsets are both consecutive. Let the size of switch set for upper outnodes be e. Then the size of switch set for lower outnodes is $r_n + 1 - e$. Hence, the size of $R_p(B_{bu}(A) \cup B_{bu}(R_n(A)))$ due to functional switches is

$$(e + r_n) + (r_n + 1 - e + r_n) = 3r_n + 1$$

Similarly, the size of forward upper node set in the next stage is

$$|B_{fu}(A) \cup B_{fu}(R_n(A))| = r_n + 1$$

These r_n+1 switches are every other switch rather than consecutive. The r_n switches in between are not contained in the forward upper node set (actually containing the forward lower set) but are in the previous replacement set. Thus the $2r_n+1$ switches are consecutive. Therefore, the size of $R_p(B_{fu}(A) \cup B_{fu}(R_n(A)))$ due to functional switches is

$$2r_n + 1 + r_n = 3r_n + 1$$

The results can be derived for the backward and forward lower node sets similarly.

Consider the required links due to the spares. All the spares that have replacement links to $B_{fu}(A) \cup B_{fu}(R_n(A))$ are in the previous set. There are no previous sets for the spares. For the case $2ks_s \geq N$, there are $r_n + 1 + b - 1 = r_n + b$ (but no more than s_s) spares in the previous replacement set. For the case $2ks_s \leq N$, there are $\lceil (r_n+1)/a \rceil$ (but no more than s_s) spares in the previous replacement set. QED

Although there are some failure sets larger than the maximum fault tolerance that fail the network, some do not. The determination of network reliability by counting such failure sets for an arbitrary graph is known to be NP-hard [27]. However, the larger fault tolerable failure sets typically make a small contribution to the network reliability since the component failure probability is very small in most practical designs. Approximations can be obtained by considering only the small failure sets.

Assume that the failure probability of each switch module is identical and independent. Let the switch module failure probability be $1-p$ and network failure probability be $1-R$. The total number of failure set instances for i failures among Y is $C(|Y|,\ i)$. Let the number of fault tolerable instances be $K(i)$. Then,

$$R = \sum_{i=0}^{|Y|} K(i)(1-p)^i p^{|Y|-i}$$

Since the failure sets of size less than or equal to the maximum fault tolerance are always fault tolerable and the failure sets of size greater than the number of spares are not fault tolerable, then

$$\begin{aligned} K(i) &= C(|Y|,\ i) && i \leq t \\ &= 0 && i > s \end{aligned}$$

For the case $s_s \leq t$, the network reliability can be found exactly. Let

$$\begin{aligned} Y_y^F(A) &= Y(A) \cap F \\ Y_y^{\overline{F}}(A) &= Y(A) - Y(A) \cap F \\ Y_x^F(A) &= Y_x(A) \cap F \\ Y_x^{\overline{F}}(A) &= Y_x(A) - Y_x(A) \cap F \\ Y_s^F(A) &= Y_s(A) \cap F \\ Y_s^{\overline{F}}(A) &= Y_s(A) - Y_s(A) \cap F \end{aligned}$$

Since the functional switch modules of SARI/k are sequentially replaceable, an interesting property of its failure set is as follows.

Theorem 6 *A failure set F of SARI/k is fault tolerable if and only if for any consecutive $A \subset X$,*

$$|Y_y^{\overline{F}}(A)| \geq |A|$$

□

Proof:

If F is fault tolerable, it is easy to see from Complete matching theory that above equation holds for any subset A.

Conversely, assume that the above equation holds for any consecutive subset and we want to show that F is fault tolerable. Let A_0 be a subset of X that is not

a consecutive set. Consider the case that A_0 consists of two consecutive subset $(A_1,\ A_2)$ such that

$$\begin{aligned} A_0 &= A_1 \cup A_2 \\ A_1 \cap A_2 &= \emptyset \end{aligned}$$

Let the failure set between A_1 and A_2 be W and

$$\begin{aligned} U_x &= Y_x^{\overline{F}}(A_1) \cap Y_x^{\overline{F}}(A_2) \\ U_s &= Y_s^{\overline{F}}(A_1) \cap Y_s^{\overline{F}}(A_2) \\ D_x &= Y_x^{\overline{F}}(A_a) - Y_x^{\overline{F}}(A_1 \cup A_2) \\ D_s &= Y_s^{\overline{F}}(A_a) - Y_s^{\overline{F}}(A_1 \cup A_2) \\ A_a &= A_1 \cup W \cup A_2 \end{aligned}$$

Note that A_a is a consecutive set. We conclude that:

- If $U_x \neq \emptyset$ then $D_x = \emptyset$.
- if $U_s \neq \emptyset$ then $D_s = \emptyset$.
- $|D_x + D_s| \leq |W|$

Consider the case that $U_x = \emptyset$ and $U_s = \emptyset$.

$$\begin{aligned} |Y_y^{\overline{F}}(A_1 \cup A_2)| - |A_1| - |A_2| &= |Y_x^{\overline{F}}(A_1)| + |Y_s^{\overline{F}}(A_1)| - |A_1| \\ &\quad + |Y_x^{\overline{F}}(A_2)| + |Y_s^{\overline{F}}(A_2)| - |A_2| \\ &\geq 0 \end{aligned}$$

Consider the case that $U_x \neq \emptyset$ or $U_s = \emptyset$. Since

$$\begin{aligned} |Y_y^{\overline{F}}(A_1 \cup W \cup A_2)| - |A_1| - |W| - |A_2| &= |Y_x^{\overline{F}}(A_1 \cup A_2)| + |Y_s^{\overline{F}}(A_1 \cup A_2)| \\ &+ |D_x| + |D_s| - |A_1| - |W| - |A_2| \\ &\geq 0 \end{aligned}$$

we have

$$\begin{aligned} |Y_y^{\overline{F}}(A_1 \cup A_2)| - |A_1| - |A_2| &\geq |W| - |D_x| - |D_s| \\ &\geq 0 \end{aligned}$$

By induction on the number of non-consecutive subsets, we can conclude this theorem. QED

We consider only the case of $ks_s \leq N/2$. In this case,

$$t = min(s_s, r_n)$$

Consider a set W between two marked switch modules 1 and 2 which have replacement links from two consecutive spares respectively as shown in Figure 3.12. For a failure set $F \subset Y(W)$ and $|F| = r_n + 1$:

$$\begin{aligned} |W| &= a - 1 \\ |Y(W) - Y(W) \cup F| &= a - 1 + r_n - (r_n + 1) \\ &< a - 1 \end{aligned}$$

Hence, F is not fault tolerable.

From the Theorem 6, the above failure sets are the only fault intolerable cases. There are s_s of them and $|Y(W)| = a - 1 + r_n$. Hence,

$$K(r_n + 1) = C(|Y|,\ r_n + 1) - s_s C(a - 1 + r_n,\ r_n + 1)$$

Similarly, $K(r_n + 2)$ can be derived as follows:

$$\begin{aligned} K(r_n + 2) &= C(|Y|,\ r_n + 2) - s_s[C(2a + r_n, r_n + 2) - C(a - 1 + r_n, r_n + 2)] \\ &\quad - (|Y| - 3a - r_n - 1)s_s C(a - 1 + r_n,\ r_n + 1) \end{aligned}$$

For the higher values of $K(i)$, approximations and bounds based on $K(r_n + 1)$ and $K(r_n + 2)$ can be used.

The network reliability of SARI/1 networks is plotted in Figure 3.14. The four stage and eight stage networks with a single replacement edge and several spares are considered. The corresponding networks without any spares and replacement edges are also plotted for comparison. A significant improvement can be observed even with a single spare per stage. In Figure 3.15, the maximum fault tolerance *versus* the number of spares per stage is shown. The maximum fault tolerance can be increased linearly with the number of spares per stage provided that there are enough replacement edges.

3.5.2 Star Coupler Interconnections

Another possible implementation is to interconnect outnodes and innodes between two consecutive stages by a star coupler. Each outnode has a tunable laser diode that can be tuned to any channel of an innode on the same star coupler as shown in Figure 3.10b. Although this implementation requires tunable laser diodes which are still very expensive today, for applications that use WDM [1], the reconfigurable links are already present in the design.

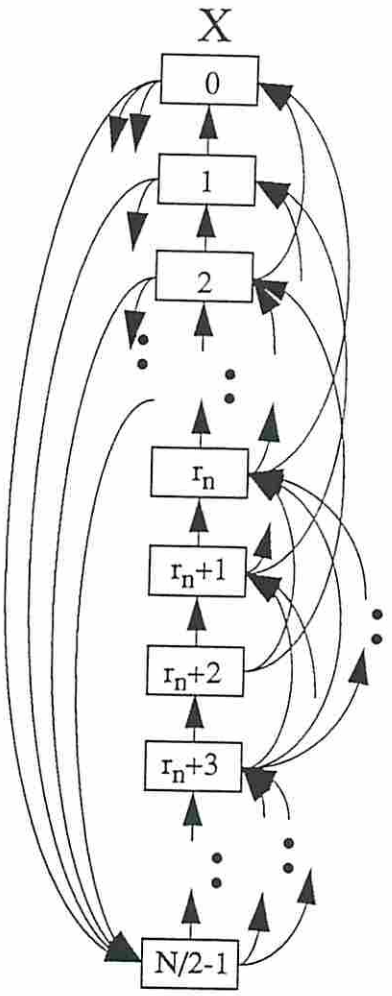


Figure 3.11: Replacement due to functional switches for SARI/k

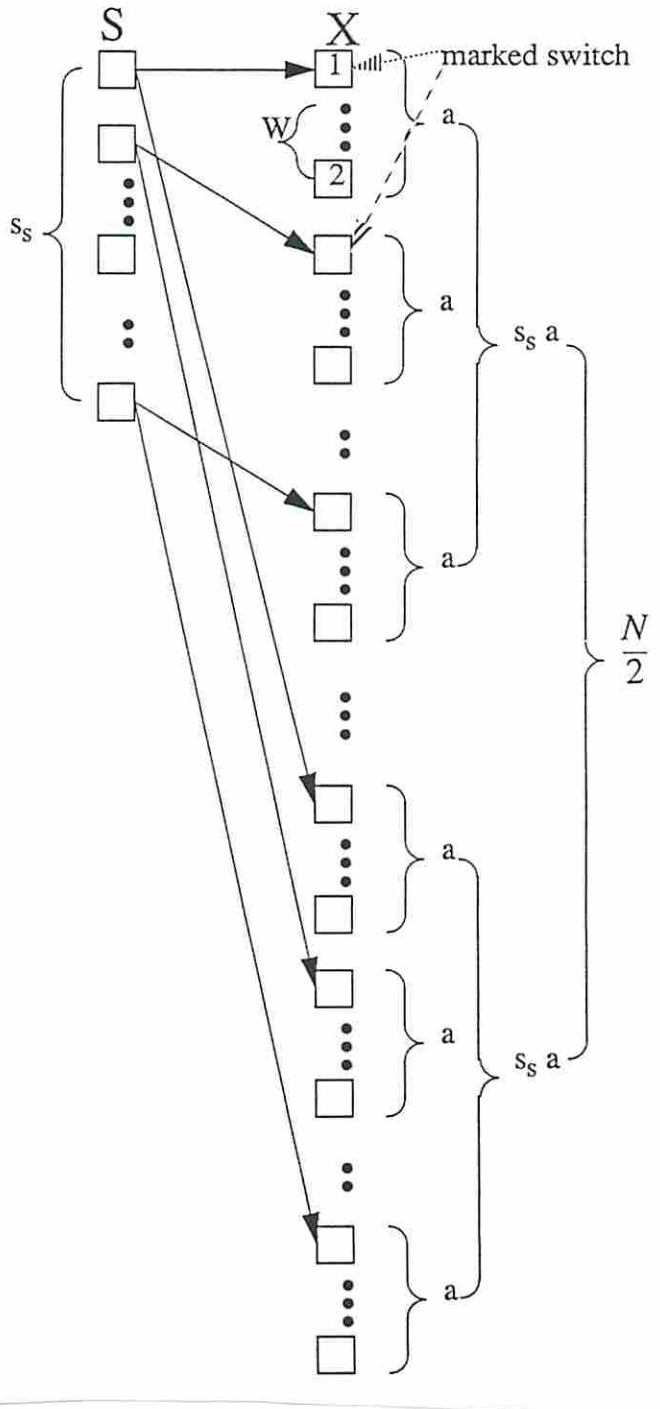


Figure 3.12: Replacement of Spares for SARI/k: $ks_s \leq \frac{N}{2}$

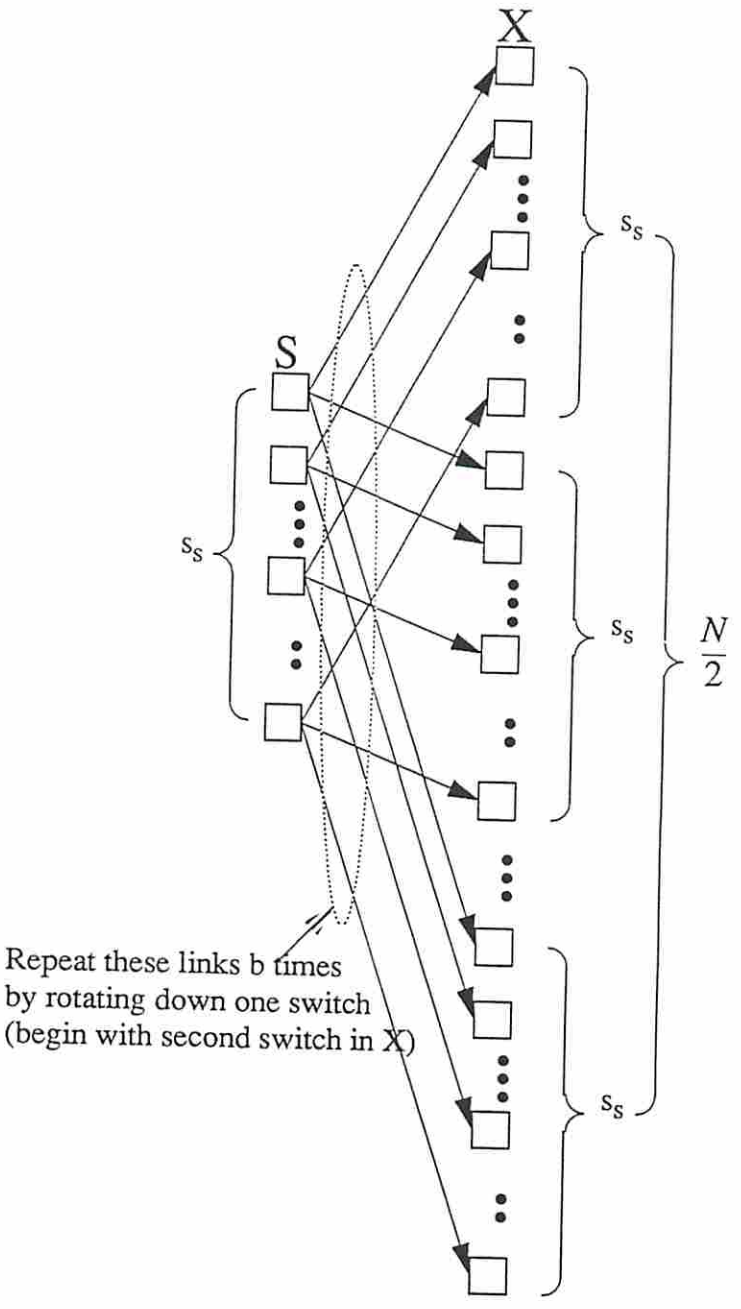


Figure 3.13: Replacement of Spares for SARI/k: $ks_s \geq \frac{N}{2}$, $s_s \leq \frac{N}{2}$

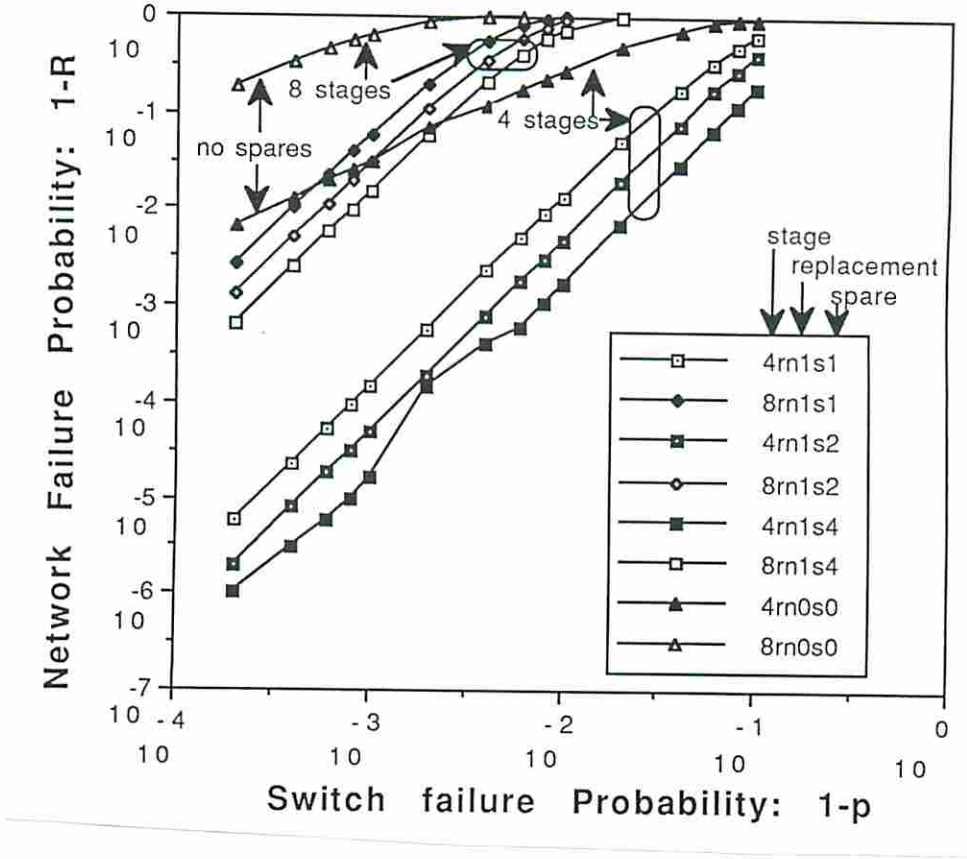


Figure 3.14: Network Reliability for SARI/1

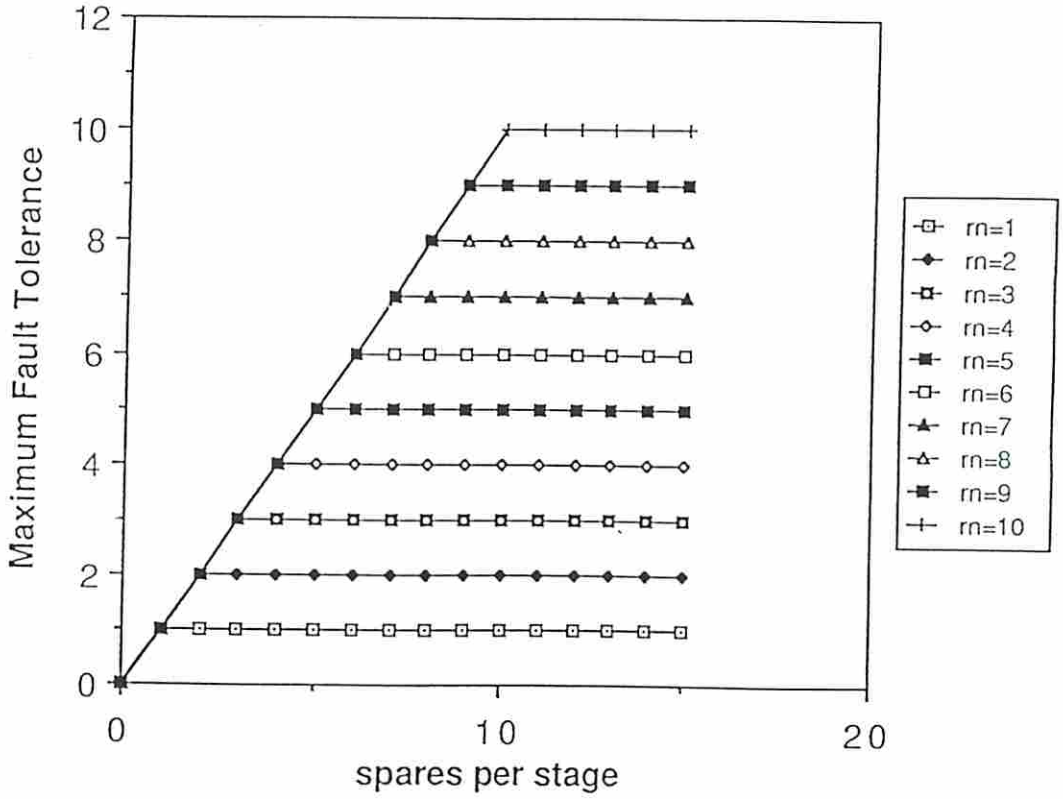


Figure 3.15: Maximum Fault Tolerance for SARI/1, 5 Stages

The innodes and outnodes on the same star coupler are logically fully connected. The upper innodes, lower innodes, upper outnodes or lower outnodes of a group of switch modules are attached to the same star couplers as shown in Figure 3.16. On the other hand, no other switch modules are accessible to them. This type of network is referred to as *Star Coupler Interconnection networks (SCI).*

Since only the switch modules on the same star coupler are replaceable by each other, spares are needed for each star coupler. Any failed switch module can be replaced by any of these spares. The two innodes or two outnodes of a switch module may be connected to different star couplers. There are l switch modules on each side of a $l \times l$ star coupler. An example of SCI implementation for the four stage Omega network of Figure 2.10 is shown in Figure 3.16. The outnodes of switch modules $\{0, 1, 2, 3\}$ in the first stage are connected to the innodes of switch modules $\{0, 2, 4, 6\}$ in the second stage via two star couplers.

The structure of Omega network (Figure 2.10) requires that in later stage, the upper and lower innodes for a switch module be on the *same* star coupler. Similarly, the outnodes for this stage will be on the same star coupler (of the next stage) and the number of nodes per star coupler is reduced to $\frac{l}{2}$, see Figure 3.16. For example: the upper outnodes of the switch modules $\{0, 2, 4, 6\}$ in the second stage are connected to both the upper and lower innodes of the switch modules $\{0, 4\}$ in the third stage (refer to Figure 2.10).

Since the smallest number of spares in the star coupler groups determines the maximum fault tolerance, we implement x spares in each star coupler group, both the full size group and

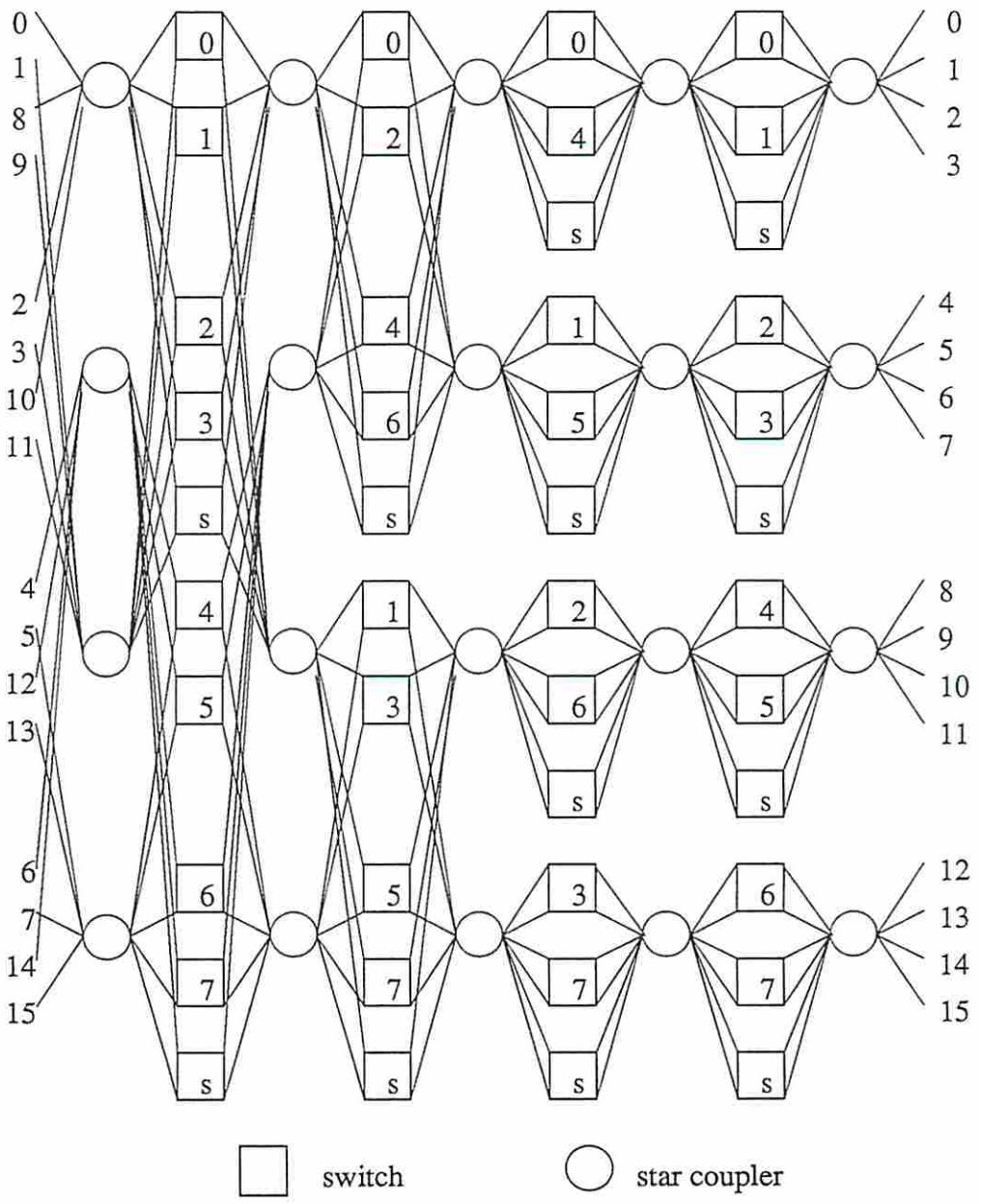


Figure 3.16: Star Coupler Interconnection for the Omega Network

the half size group. It is easy to see that the maximum fault tolerance is

$$t = x$$

Let g_1 and g_2 be the largest integers that divide the total number of functional switch modules, $\frac{N}{2}$, such that

$$\begin{aligned} g_1 &\leq l - x \\ g_2 &\leq \frac{l-x}{2} \end{aligned}$$

For simplicity of analysis, assume that $l - x$ is a multiple of 2. Group the switch modules in the first stage so that they have the same high order $\log N - \log(l - x)$ bits. Then, from the stage 0 to the stage $\log N - \log(l - x) - 1$, the switch modules are in the full groups and the rest of switch modules are in the half groups.

A group of switch modules will fail if the number of failures is more than the number of spares, x. The network fails if any of the groups fails. Let

$$G(l, x) = \sum_{i=0}^{x} C(l,\ i) p^{l-i} (1-p)^i$$

Thus, the network reliability is

$$R = G^{[\log N - \log(l-x)]\frac{N}{2g_1}}(g_1, x) G^{\log(l-x)\frac{N}{2g_2}}(g_2, x)$$

The network reliability of SCI networks is plotted in Figure 3.17. Five stage and eight stage networks with a single spare and various star coupler sizes are considered. A significant improvement can be observed with a single spare per star coupler group.

Although both the SARI/k and SCI networks improve reliability significantly, SCI usually requires many more spares for a large network since the number of spares for SARI/k is $O(\log N)$ while the number of spares for SCI is $O(N \log N)$.

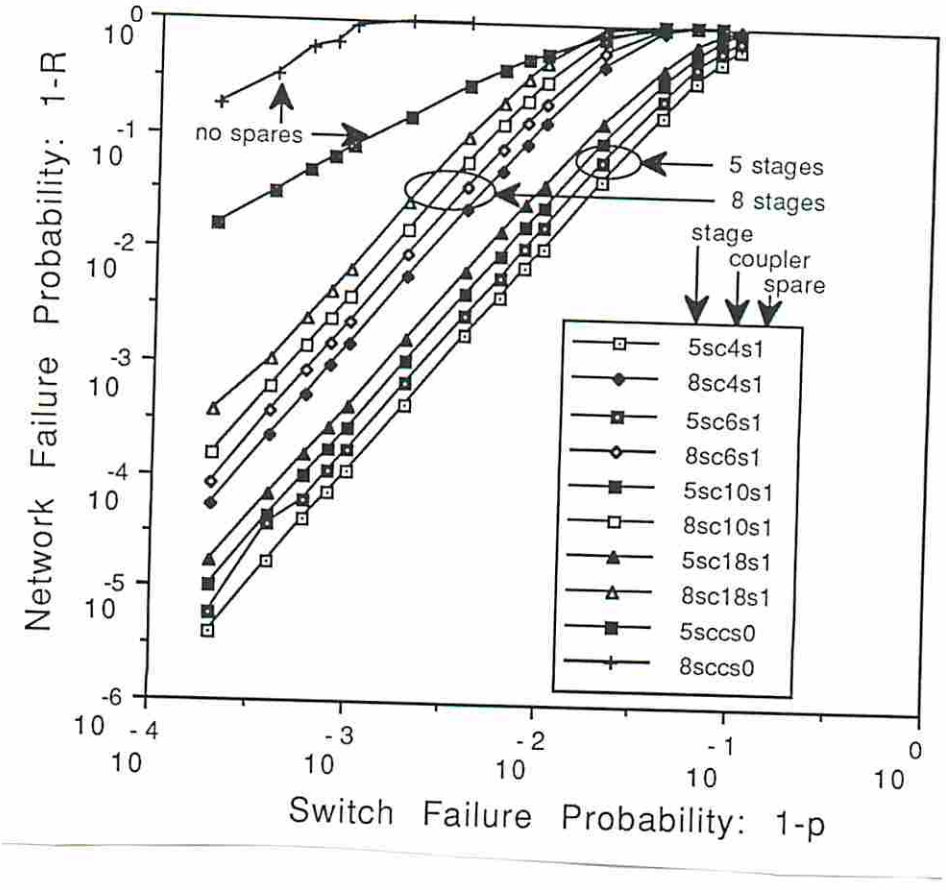


Figure 3.17: Network Reliability for SCI

Chapter 4

Topology Design with Topology Invariant Mode

In this chapter, the topology design problem is addressed as an attempt to reduce traffic imbalance. As shown in Figure 2.15, the external sources and destinations access the internal network via the access network. The internal network can be characterized by the functional topology. The functional topology defines how the cells are decoded and how the switch modules are shared. In other words, it defines how the traffic flows are forked and how they are combined. No matter how the internal network is reconfigured, the same traffic flows exist at some internal link. Therefore, the functional topology can be used to represent the internal topology.

For a given traffic flow pattern and a given network functional topology, we want to minimize the mean delay and maximize the overall throughput from the permissible access networks which generate the permissible topology space. The fundamental case is the functional topology of unique path MIN since it has only the basic switching functions.

Consider the ShuffleNet. Assume that the only external traffic flows are from the sources in first stage and they only send to the destinations in the last stage. Then, the problem for the ShuffleNet is exactly the same as for the unique path MIN. As we will see later, the problem is very difficult even for the unique path MIN case and an exact solution for either case is intractable. We therefore focus on the case of a unique path MIN in a centralized network.

Delay is mainly a function of propagation delay, queueing delay and channel access delay. Since all channels are logically dedicated, there is no conflict on channel access. The propagation delay due to the distributed switch modules has been considered in [9]. We consider a centralized network and hence the propagation delay (through the switch module) is relatively insignificant. Therefore, queueing is the major factor. The overall throughput is a function of the buffer size and the uniformity of traffic flows. The cell loss probability of each queue must be kept extremely low to maintain a good quality of service. We assume that there are adequate buffers so that we may ignore the impact of this on overall throughput. Then, it has been shown that *minimizing* the *maximum link flow* is a reasonable approach to this problem [15]. Hence, the topology design problem is equivalent to the traffic flow assignment problem [67]

4.1 Traffic Flow Assignment

The *traffic flow* t_{ij} from the external source i to the external destination j is defined by the number of cells per unit time generated from the customers. and the *traffic flow matrix*

$$T = [t_{ij}] \qquad \forall\, 0 \leq i,\ j \leq N-1$$

specifies the customer requirements.

Referring to Figure 4.1, the external sources and destinations are connected to the internal sources and destinations via the access networks. Each external source can be assigned to one of the internal sources and similarly each external destination can be assigned to one of the internal destination. The only requirement is that the access network assignment is one to one. We model the one to one mapping as a permutation [110]. A permutation can be represented as a set of cycles. Thus the permutation

$$\begin{pmatrix} 0\,1\,2\,3\,4\,5\,6\,7 \\ 2\,3\,4\,5\,0\,1\,7\,6 \end{pmatrix}$$

consists of three *cycles* and can be rewritten as

$$(0,\ 2,\ 4)(1,\ 3,\ 5)(6,\ 7)$$

Define the set of all source permutations, Ω_s, and the set of all destination permutations, Ω_d. Denote a *source permutation* and a *destination permutation* as $\pi_s \in \Omega_s$ and $\pi_d \in \Omega_d$. The source and destination permutations define the permissible topology space in a different form.

There are many ways to define the details of the interconnection topology. Traditionally, the physical interconnections between stages are specified [69] or a graph theoretic model is used [4]. These models have been successfully used in many MIN applications. Another simpler model that is particularly useful for the traffic flows among the links and switch modules is the *traffic flow model* used in [115]. Since the focus here is similar, we adopt this model.

The *source set*, $S(i,\ g)$ is a subset of internal sources that may send cells through the link i in the link stage g. Similarly, the *destination set*, $D(i,\ g)$ is a subset of the internal destinations that may receive cells from the link i in the link stage g. In a general topology, the traffic flows passing through a link may be the traffic flows indexed by an arbitrary subset of $S(i,\ g) \times D(i,\ g)$. However, a more complicated routing scheme is required which is undesirable in a high speed network. Therefore, we consider only designs with full set traffic flow, i.e., the traffic flows passing through a link are indexed by the full set of $S(i,\ g) \times D(i,\ g)$. This actually is the case for the self-routing scheme.

The unique path MIN with $k \times k$ crossbar switch modules can be easily modeled by recursively defining the source and destination sets (assume that all the links are properly labeled and the labeling is irrelevant in our discussions).

$$\begin{aligned}
S(i,\ 0) &= \{i\} \\
D(i,\ n) &= \{i\} \\
S(i,\ g+1) &= \cup_{m=0}^{k-1} S(i - (i) mod(k^{g+1}) + m,\ g) \qquad \forall\, 0 \leq g \leq n-1 \\
D(i,\ g-1) &= \cup_{m=0}^{k-1} D(i - (i) mod(k^{n-g+1}) + m,\ g)\ \forall\, 1 \leq g \leq n
\end{aligned}$$

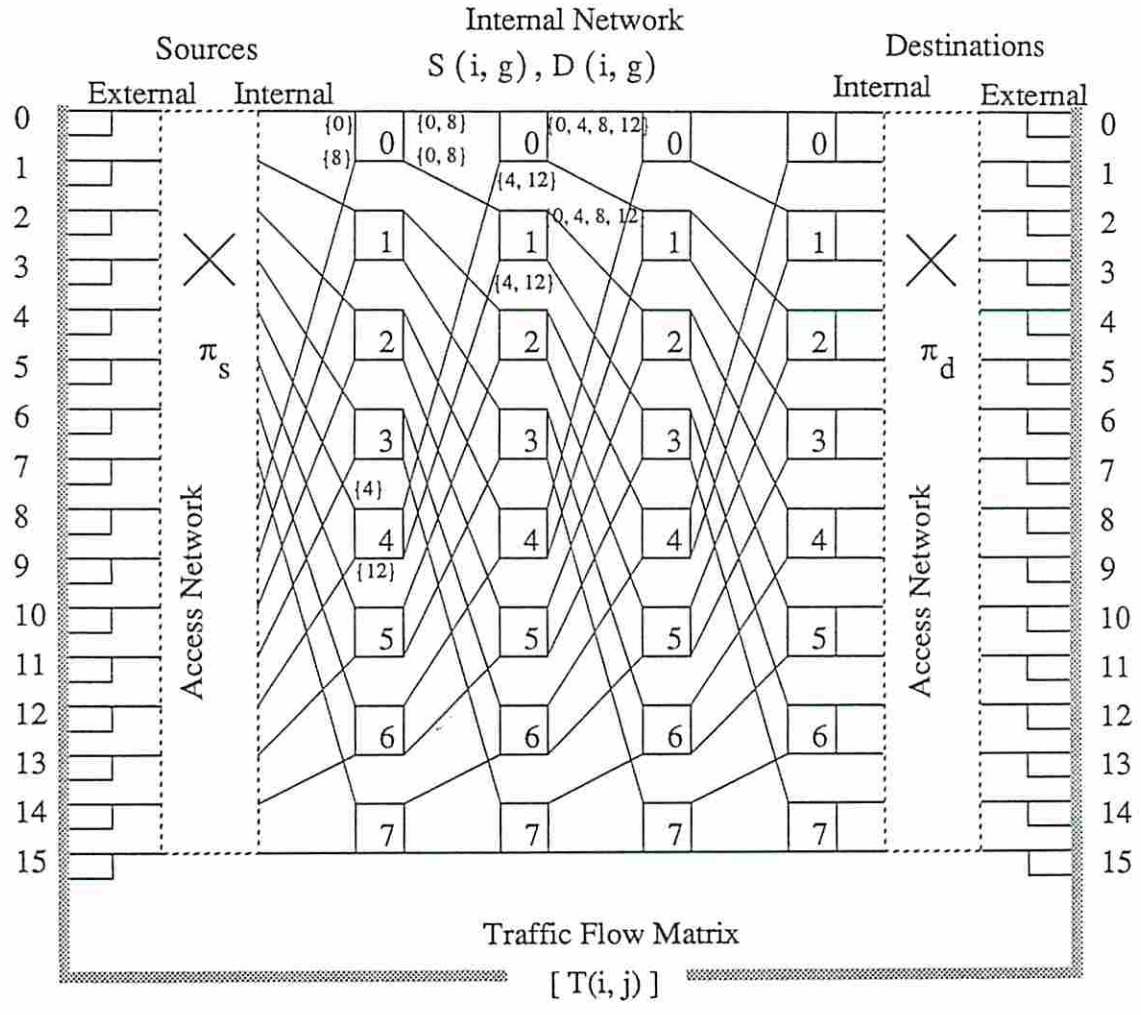


Figure 4.1: Traffic Flow Assignment Problem Model

Refer to Figure 4.1. The source set for a link at the link stage 0 is defined by the internal source attached to it. The source set for a link at link stage 1 is defined by the union of the source sets of the links attached to the same switch module at link stage 0, which defines the possible internal sources that may access this link. For the example shown in Figure 4.1, the source set of the upper input of switch modules 0, 1, 2, 3 at the switch module stage 2 is $\{0, 4, 8, 12\}$.

Similarly, destination sets can be found by working backwards from the last stage. The internal destinations are assigned to the links in the link stage n that they attach to. The rest of the link stages are generated by the backward recursion. The source and destination sets defined in the above Equations correspond to the Least Significant Bit (LSB) first decoding for the sources and the Most Significant Bit (MSB) first decoding for the destinations [115]. Since all the cells passing through a link are for the same set of destinations, there is no need for source address decoding. Also, each destination set is equally divided into k destination sets at the next link stage, only a single bit $(0..k-1)$ is required for decoding.

The traffic flow on a link then is

$$l(i,\ g) \ = \sum_{s \in S(i,\ g),\ d \in D(i,\ g)} t_{\pi_s^{-1}(s),\ \pi_d^{-1}(d)}$$

where π_s^{-1} and π_d^{-1} are the inverse mappings of the permutation. The objective function of the reconfiguration is to minimize the maximum (min-max) link flow and the problem can summarized as follows:

Problem 1 Traffic Flow Assignment Problem *(TFAP):*
Given*:*

> *A traffic flow matrix T, source and destination permissible permutation spaces Ω_s, Ω_d, and a network topology specified by $S(i,\ g)$, $D(i,\ g)$.*

Question*:*

> *Find the π_s and π_d that achieves:*

$$\min_{\pi_s \in \Omega_s,\ \pi_d \in \Omega_d} \left(\max_{0 \le i \le N-1,\ 0 \le g \le n} l(i,\ g) \right)$$

4.2 Computation Complexity

Consider the case that external sources can be arbitrarily assigned to any internal source. Then the set of permissible source permutations is the permutation group S_N [110] of all possible permutations of N sources. The same argument applies to the set of permissible destination permutations. Hence, the problem with

$$\Omega_s = \Omega_d = S_N$$

is called the *full* TFAP.

With the technology in the foreseeable future, full access to all internal sources and destinations is still difficult for a large size network. Therefore, we assume that only some of the internal sources or internal destinations are accessible from a external source or a external

destination. The permissible permutation spaces for the source and destination sets are thus subsets of the permutation group S_N. This problem is called the *partial* TFAP. Clearly, the full TFAP is a restriction of partial TFAP.

Determination of the min-max link traffic is mathematically equivalent to bounding all link traffic $l(i,\ g)$ by an arbitrary constant C. We can think of C as the link capacity of all links. Thus, finding the mini max link traffic flow is the same as determining whether the link traffic flows can be bounded by an *arbitrary* capacity C. Therefore, TFAP is equivalent to the following problem:

Problem 2 Traffic flow assignment problem with bounding*: (TFAP/B)*
Given*:*

A traffic flow matrix T, source and destination permissible permutation spaces Ω_s, Ω_d, network topology specified by $S(i,\ g)$, $D(i,\ g)$, and an arbitrary constant C.

Question*:*

Find π_s and π_d such that

$$l(i,\ g) \leq C \qquad 0 \leq i \leq N-1,\ 0 \leq g \leq n$$

□

This is a combinatorial optimization problem. Many of these problems are NP-hard [42]. If a known NP-hard problem polynomially transforms to a given problem, it is also NP-hard. *Equal Partition Problem* (EPP) is a known NP-hard problem (actually a NP-complete) [42].

Problem 3 Equal partition problem*:*
Given*:*

A finite set A and a size $s(a) \in Z^+$ for each $a \in A$.

Question*:*

Is there a subset $A' \subset A$ of size $|A'| = |A|/2$ such that

$$\sum_{a \in A'} s(a) = \sum_{a \in A - A'} s(a)$$

□

We will show that EPP is a restricted form of TFAP/B and therefore is also NP-hard [42].

Theorem 7 *TFAP/B is* NP*-hard.* □

Proof:

We restrict TFAP/B to EPP. Let the size of N given objects in EPP be $a_0,\ a_1,\ ...,\ a_{N-1}$. Consider an instance of TFAP/B such that

$$\begin{aligned} t(i,\ j) &= a_j && \forall\ i = 1 \text{ or even},\ \forall\ 0 \leq j \leq N-1 \\ &= 0 && \text{otherwise} \\ k &= 2 \\ C &= \sum_{i=0}^{N-1} a_i \end{aligned}$$

Let π_e be the identity permutation. Since all traffic flows for the links at link stage 0 are either 0 or C (identical among the same group), the best assignment that has minimum bound is the partial TFAP/B such that

$$\begin{aligned} \Omega_s &= \{\pi_e\} \\ \Omega_d &= S_N \end{aligned}$$

In this case, The bottleneck link is either the link 0 or 1 at the first link stage since all other links have traffic flow less than or equal to at least one of these two links. Bounding these two links is exactly the EPP. We conclude this theorem. QED

4.3 Heuristic Algorithm for Buddy Assignment Problem

A group of internal sources (destinations) is called a *source (destination) buddy set* if it is the source (destination) set of some link. If this link is at link stage i (link stage $n-i$), there are k^i elements in the corresponding source (destination) buddy set. For example, {0, 4, 8, 12} in Figure 4.1 is a source buddy set since the link to the upper innode of switch modules 0 at link stage 2 has it as a source set. Let $P_s = k^{p_s}$ and $P_d = k^{p_d}$. Consider a design that uses a $P_s \times P_s$ star coupler to connect the external and internal sources (transmitters and receivers with tunable lasers). Each external source can access any of these P_s internal sources but cannot access other internal sources. Therefore, the external sources are fully connected to the internal sources on the same star coupler. Similarly, $P_d \times P_d$ star couplers can be used for the destination connections.

The buddy access design is a partial TFAP and the cycles of source permutation are restricted to the buddy sets of size P_s. Selecting different source permutations can affect the traffic flows of the links only up to the link stage $p_s - 1$. The links after the link stage $p_s - 1$ have traffic flows indexed by either all sources in a source buddy set or none. A similar property for the destination buddy set applies from the link stage n to link stage $n - p_d + 1$.

In general, the traffic flows of a link may be affected by changing the source permutation or by changing the destination permutation. Hence, it is a two dimensional traffic flow assignment problem. If $p_s + p_d \leq n$, then the source and destination permutations are independent. It is reduced to a one dimensional traffic flow assignment problem. Along the lines of the proof of Theorem 7, we can see that it still contains EPP, however, and is NP hard. On the other hand, if we can evenly partition the aggregate traffic flows of the sources and destinations into k subsets stage by stage for the one dimensional case, the mini max link traffic flow is guaranteed.

The two dimensional traffic flow assignment case is much more complicated. Consider the full TFAP with $k = 2$. A traffic flow matrix is shown in Figure 4.3a. A source permutation is essentially a row permutation of the matrix while a destination permutation is a column permutation. For a given row and column permutation, the traffic flow on the links in the link stage 0 is the sum of the elements in a row. The traffic flows of links in the link stage 1 are split by the dotted lines which are made by pairing the rows and dividing the columns into two groups. Bounding the flow on all links is equivalent to properly pairing and dividing the rows and columns of the matrix stage by stage simultaneously. As shown before, even the one dimensional assignment problem is very difficult. Therefore, we propose a heuristic

algorithm that considers the source and destination permutations independently. Selection of each corresponds to a one dimensional assignment problem which is solved by a heuristic partitioning algorithm.

Algorithm 4 Heuristic for the full TFAP

//X_i's are Mapping indices for the sources S_i's//
//Y_i's are Mapping indices for the destinations D_i's//
For $i = 0$ *until* $N - 1$

$$\begin{aligned} S_i &= \sum_{j=0}^{N-1} t_{ij} \\ D_i &= \sum_{j=0}^{N-1} t_{ji} \\ X_i &= i \\ Y_i &= i \end{aligned}$$

Sort(S, X) *//Sort the sources and update index, S_0 is the largest//*
Sort(D, Y) *//Sort the destinations and update index, D_0 is the largest//*
k-RECURSIVE.PARTITION(N, S, X)
k-RECURSIVE.PARTITION(N, D, Y) □

Algorithm 5 k-RECURSIVE.PARTITION(N, S, X)

if $N \geq k$
    k-PARTITION(N, S, X)
    for $i = 0$ *until* $k - 1$
        k-RECURSIVE.PARTITION(N/k, $S_{i \cdot N/k}$, $X_{i \cdot N/k}$) □

Algorithm 6 k-PARTITION(N, S, X)

// Partition S into k groups, N/k slots each group//
// B_i is the storage buffer //
// G_j is the sum of entries in group j //
// I_j is the first empty slot in group j //
For $j = 0$ *until* $k - 1$

$$\begin{aligned} G_j &= 0 \\ I_j &= j \cdot \frac{N}{k} \end{aligned}$$

For $i = 0$ *until* $N - 1$
    select j such that:
        G_j is smallest // best fit group //
        $I_j < (j+1) \cdot \frac{N}{k}$ //still has an empty slot //

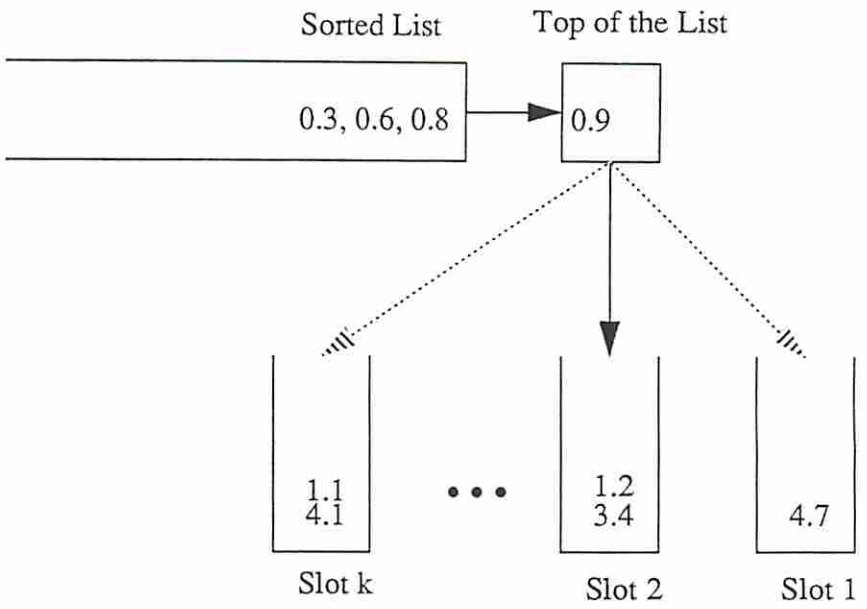


Figure 4.2: Best Fit heuristic for k-Partition problem

$$S(i) \rightarrow B_{j \cdot \frac{N}{k}+I_j}$$
$$G_j + S(i) \rightarrow G_j$$
$$I_j + 1 \rightarrow I_j$$
$$B \rightarrow S$$

□

The heuristic algorithm recursively partitions the given set into k sorted subsets. The k-PARTITION algorithm 6 can be seen as shown in Figure 4.2. The set S is sorted and then each element, largest first, is placed in the k stacks of $\frac{N}{k}$ slots. The best fit algorithm chooses the largest remaining slots for placement, similar to the well known bin packing problem. The best fit with sorted set may not be the optimal solution, but is a good one [42].

Consider the time complexity of the algorithm. The algorithm of k-PARTITION takes $O(i \cdot k)$ to find the best fit group for the size of i.
For the k-RECURSIVE.PARTITION algorithm, we use a DFS (depth first search) for a full tree. It runs k-PARTITION N/k^l times of size $i = k^l$ at the level $\log_k N - l + 1$. The algorithm spends $k \cdot N$ time at each level. Hence, the time complexity of the heuristic algorithm is $O(N \log_k N)$.

An example of a 8×8 network based on 2×2 switch modules is shown in Figure 4.3. The traffic flow matrix is given in Figure 4.3a. Figure 4.3b gives the traffic flow of all links if $\pi_s = \pi_s = \pi_e$ (identity permutation). Applying the heuristic algorithm we obtain the source and destination permutations shown in Figure 4.3c and Figure 4.3d respectively. The permuted internal traffic matrix is shown in Figure 4.3e. Finally, the traffic flows for the links are shown in Figure 4.3f. The maximum link traffic flow with no permutation is 6.0. The maximum link traffic flow with our heuristic is 5.4, which is a 10% improvement. Furthermore, observe that the maximum link traffic flow for the links at link stages 0 and 3 is 5.2. Thus the best possible

$$T = \left[\begin{array}{cccc:cccc} 0.0 & 0.8 & 0.9 & 0.1 & 0.7 & 0.3 & 0.5 & 0.4 \\ 0.7 & 0.0 & 1.2 & 0.3 & 1.0 & 0.5 & 0.7 & 0.6 \\ 0.5 & 0.1 & 0.0 & 0.1 & 0.8 & 0.9 & 0.4 & 0.2 \\ 0.9 & 0.5 & 0.2 & 0.0 & 0.3 & 1.5 & 0.8 & 0.5 \\ 0.7 & 0.1 & 1.0 & 1.0 & 0.0 & 0.4 & 0.9 & 0.7 \\ 0.2 & 0.6 & 0.9 & 0.2 & 0.7 & 0.0 & 0.5 & 1.0 \\ 0.5 & 0.3 & 0.8 & 0.8 & 0.4 & 0.7 & 0.0 & 0.7 \\ 0.1 & 0.4 & 0.2 & 0.9 & 0.1 & 0.8 & 0.4 & 0.0 \end{array}\right]$$

a. The Given Traffic Flow Matrix

Link stage: links 0, 1, 2 ,3 ,4, 5 ,6 ,7

0: 3.7 5.0 3.0 4.7 4.8 4.1 4.2 2.9
1: 4.0 2.3 4.7 4.0 4.7 5.4 4.2 3.1
2: 3.5 2.9 2.8 5.8 6.0 3.1 4.1 4.2
3: 3.6 2.8 5.2 3.4 4.0 5.1 4.2 4.1

b. Link Traffic flows without Reassignment

Old: 3.7 5.0 3.0 4.7 4.8 4.1 4.2 2.9
New: 5.0 2.9 4.2 4.1 4.8 3.0 4.7 3.7
Permutation: (0, 7, 1) (2, 5, 3, 6) (4)

c. Source Reassignment and the Permutation

Old: 3.6 2.8 5.2 3.4 4.0 5.1 4.2 4.1
New: 5.2 2.8 4.1 4.0 5.1 3.4 4.2 3.6
Permutation: (0, 2, 7) (1) (3, 4, 5) (6)

d. Destination Reassignment and the Permutation

$$\left[\begin{array}{cccc:cccc} 1.2 & 0.0 & 0.6 & 1.0 & 0.5 & 0.3 & 0.7 & 0.7 \\ 0.2 & 0.4 & 0.0 & 0.1 & 0.8 & 0.9 & 0.4 & 0.1 \\ 0.8 & 0.3 & 0.7 & 0.4 & 0.7 & 0.8 & 0.0 & 0.5 \\ 0.9 & 0.6 & 1.0 & 0.7 & 0.0 & 0.2 & 0.5 & 0.2 \\ 1.0 & 0.1 & 0.7 & 0.0 & 0.4 & 1.0 & 0.9 & 0.7 \\ 0.0 & 0.1 & 0.2 & 0.8 & 0.9 & 0.1 & 0.4 & 0.5 \\ 0.2 & 0.5 & 0.5 & 0.3 & 1.5 & 0.0 & 0.8 & 0.9 \\ 0.9 & 0.8 & 0.4 & 0.7 & 0.3 & 0.1 & 0.5 & 0.0 \end{array}\right]$$

e. The Reassigned Internal Traffic Flow Matrix

Link stage: links 0, 1, 2 ,3 ,4, 5 ,6 ,7

0: 5.0 2.9 4.2 4.1 4.8 3.0 4.7 3.7
1: 3.5 5.4 2.9 4.3 4.4 2.9 4.9 4.1
2: 4.4 3.6 4.5 3.6 4.2 4.3 3.1 4.7
3: 5.2 2.8 4.1 4.0 5.1 3.4 4.2 3.6

f. Link Traffic flows after Reassignment

Figure 4.3: An Example for the Traffic Assignment Heuristic

maximum link traffic flow (which may not even be achievable) is 5.2, since the link traffic flows of these links are not changed by any source or destination permutation. Thus, for this example the heuristic algorithm has achieved *at least* 75% of the potential improvement possible.

Chapter 5

Performance/Reliability Tradeoffs for Multi-path MIN

Although ATM networks and SONET transmission systems will eventually provide a very high speed network, there is a long lead time for equipment development and network deployment. In the interim, relatively slower fast packet switching networks, such as SMDS and Frame Relay based on existing transmission systems, (DS1 and DS3), are closer to realization. On the other hand, there are always some applications that require only lower speed switching. In these cases, the TI mode and unique path property are no longer mandatory requirements for the switching network. Thus, in this chapter, we consider designs that relax the restrictions of unique path and TI mode.

Conventional FTMINs which allow more efficient use of hardware redundancy can be used to solve the reliability problem. Most research in conventional FTMINs addresses only the reliability problem by considering the network connectivity. The performance under failure mode which is an intrinsic factor to the network design is seldom discussed.

The networks proposed in Chapter 3 preserve the functional topology and hence the performance under failure mode is maintained. However, as we will see later, the performance of conventional FTMINs may be severely degraded under failure mode. In this Chapter, we address the performance issue under failure mode for the network without the TI mode and unique path restriction.

Consider the unique path MIN based on the $k \times k$ cross bar switch as shown in Figure 2.10. Every *outnode* of a switch module is connected to an *innode* of a switch module in next stage such that there exists a unique path between any external source and destination pair. The unique path MIN forms the building block of the networks to be discussed.

Consider the case of switch module redundancy. There are two basic ways to provide redundant paths.

1. Provide an extra stage: e.g. ESC, Beneš networks.
   These type networks provide only fault tolerance.
2. Provide an extra copy of the unique path MIN: e.g. INDRA. These types of networks provide fault tolerance and extra performance improvement.

Most of the FTMINs use a mix of these techniques, e.g. ACN, 2-MDN, F-net, etc. and they have mixed benefits for the fault tolerance and performance improvement too. To further demonstrate this point, we describe two simple networks that demonstrate this relationship.

5.1 Performance Degradation of Two path MINs

Although multi-path MINs can provide fault tolerance, the failures cause irregularity in the network topology. The traffic flows of the irregular network are no longer balanced. Hence, the network performance is degraded under fault tolerable failures [33].

5.1.1 Two copy MINs

In this section, the aspect of providing multiple paths by extra copies of unique path MIN is discussed. A very simple network is defined to reveal how the performance is improved in this scenario.

Definition 6 *A two copy MIN connects N sources and destinations via two copies of the unique path MIN. In between, use N 1×2 switch modules and N 2×1 switch modules such that there are exactly two paths for each source/destination pair.* □

An example is shown in Figure 5.1. Although a single failure does not destroy the connectivity of a two copy MIN, the network performance is degraded. For simplicity, we assume that a new cell is randomly directed to either copy with probability $\frac{1}{2}$. Both the sources and destinations are assumed to be fast, i.e. they insert and remove cells in twice speed as the two sub-unique path MINs do.

A *normal two copy MIN* with two identical unique path subnetworks is shown in Figure 5.1. The same internal sources of these two sub-unique path MINs are attached to the same external source. The destinations are handled similarly.

Consider the case that all sources are generating cells identically and independently. Also, each source distributes the cells to all its destinations equally and independently. If there is no failure, the traffic distribution of all links at the same stage are identical. Hence the traffic flows for all the links are identical and are given by:

$$A = N \cdot t_{ij} \qquad \forall i, j$$

If a switch module of one subnetwork is broken, the cells through that switch module have to be rerouted to the other subnetwork and a nonuniform traffic pattern is created. In Figure 5.1, the second switch of stage 1 is broken, hence the traffic flows through the innodes and outnodes of that switch, which consist of the traffic flows from sources 0, 1, 2, 3 to destination 4, 5, 6, 7, are now zero. One half of the original traffic from sources 0, 1, 2 and 3 will still be directed to this subnetwork. The other half is rerouted to the second subnetwork. If the topologies of these two subnetworks are identical, the switch corresponding to the failed switch in the second subnetwork becomes the bottleneck, since all rerouted traffic flows pass through it. The total amount of traffic flow at the duplicate switch is twice as much as that in the original network as shown in Figure 5.1.

5.1.2 Dual MINs

The bottleneck can be removed by distributing the rerouted packets to other switches. For the example shown in Figure 5.2, the interconnections in the link stage 0 and link stage n are connected in different patterns. As this design example shows, the bottleneck is shifted to the link stage 0 which has a worst case traffic flow of $\frac{3A}{2}$ instead of the maximum of $2A$ found in

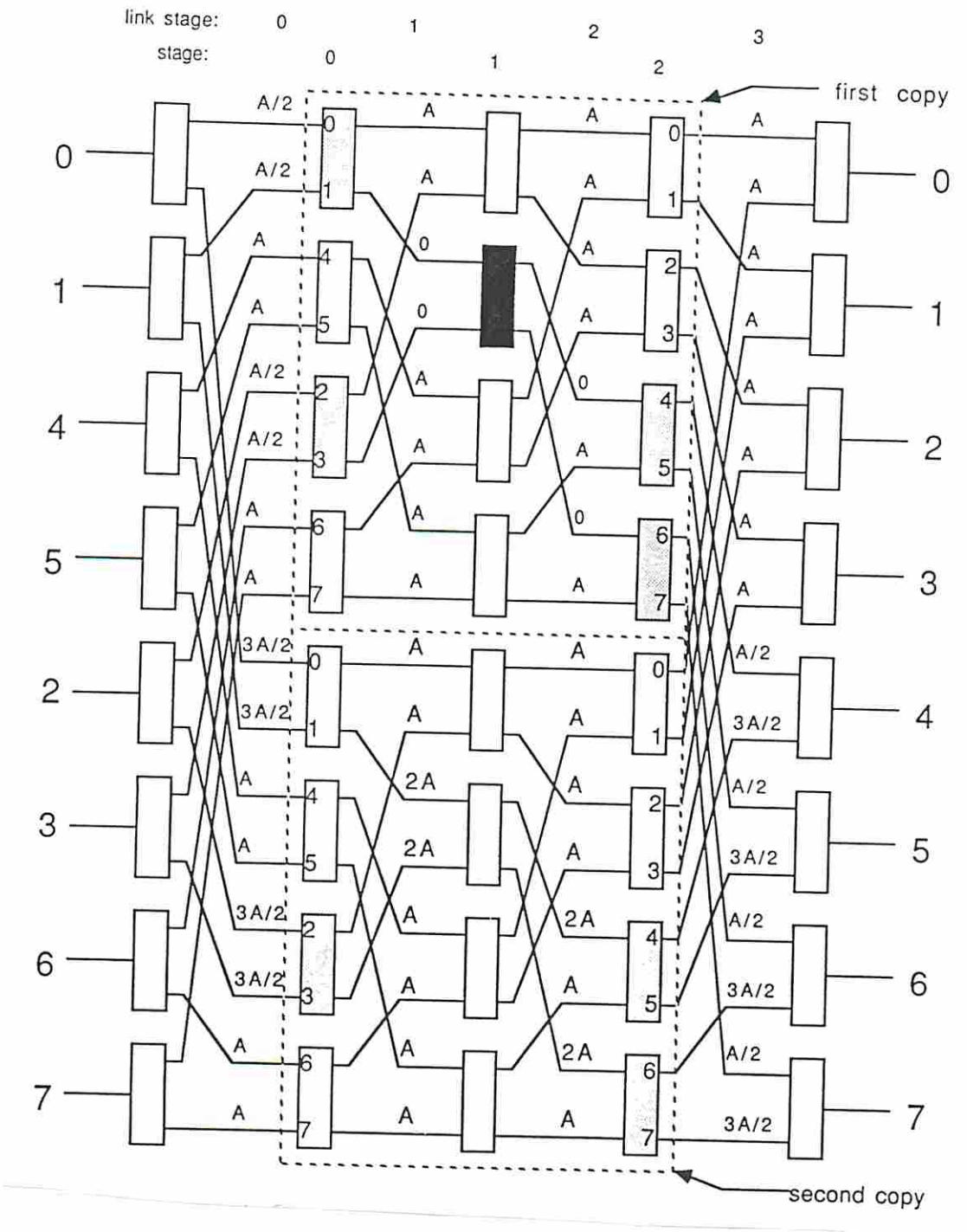


Figure 5.1: Two Copy MIN of Size 8

the original design. Also, note that the extra traffic flows due to rerouting are shared by more links at all stages.

The routing sequence of the unique path MIN is represented by the log N bits of destination tag. The switches in each stage use one bit information of destination tag to decide where to send for the destination. As show in section 4.1, the traffic flows on the link i in the link stage g are the cross product of the source set $S(i,\ g)$ and the destination set $D(i,\ g)$ for a unique path MIN. The union operation used in the definitions for source sets and destination sets is corresponding to the routing function of unique path MIN.

Therefore, the traffic flow merging and forking sequences can be defined in a similar way to the cell routing sequence. In the link stage 0, the links contain all destinations for the destination set and only one source for the source set. As the stage progresses toward the link stage n, each destination set is forked to two equal size subsets while every two source sets are merged into a bigger set.

Definition 7 *Source merging operation:*

The source sets associated with the innode links of a switch module are merged to the source set associated with any outnode link of the same switch module. □

Definition 8 *Destination forking operation:*

The destination sets associated with an innode link of a switch module forks into the destination sets associated with all outnode links of the same switch module. □

Definition 9 *Source merging sequence:*

The sequence of merging source set in n stages. □

Definition 10 *Destination forking sequence:*

The sequence of forking destination set in n stages. □

The bottleneck of a normal two copy MIN operating under a single failure is due to the rerouted cells all passing through the same duplicate switch module corresponding to the failed switch module. If a different source merging sequence or destination forking sequence is assigned in the second copy subnetwork such that both the sequences are reversed as shown in Figure 5.2, the rerouted traffic flows from sources 0, 1, 2, 3 are not merged in the switch module stage 0. Furthermore, the rerouted traffic for each source to the destinations 4, 5, 6, 7 are forked to different links in the next link stage.

Actually, the rerouted cells from different sources are merged at the latest possible stage and the rerouted cells to different destinations are forked at the earliest possible stage. By the above observation, we define a new class of networks that suffer much less from the bottleneck problem due to the unbalanced traffic pattern under the failure operation mode.

Definition 11 *A dual MIN is a two copy MIN with reversed memory forking sequence and reversed processor merging sequence in the second copy subnetwork.* □

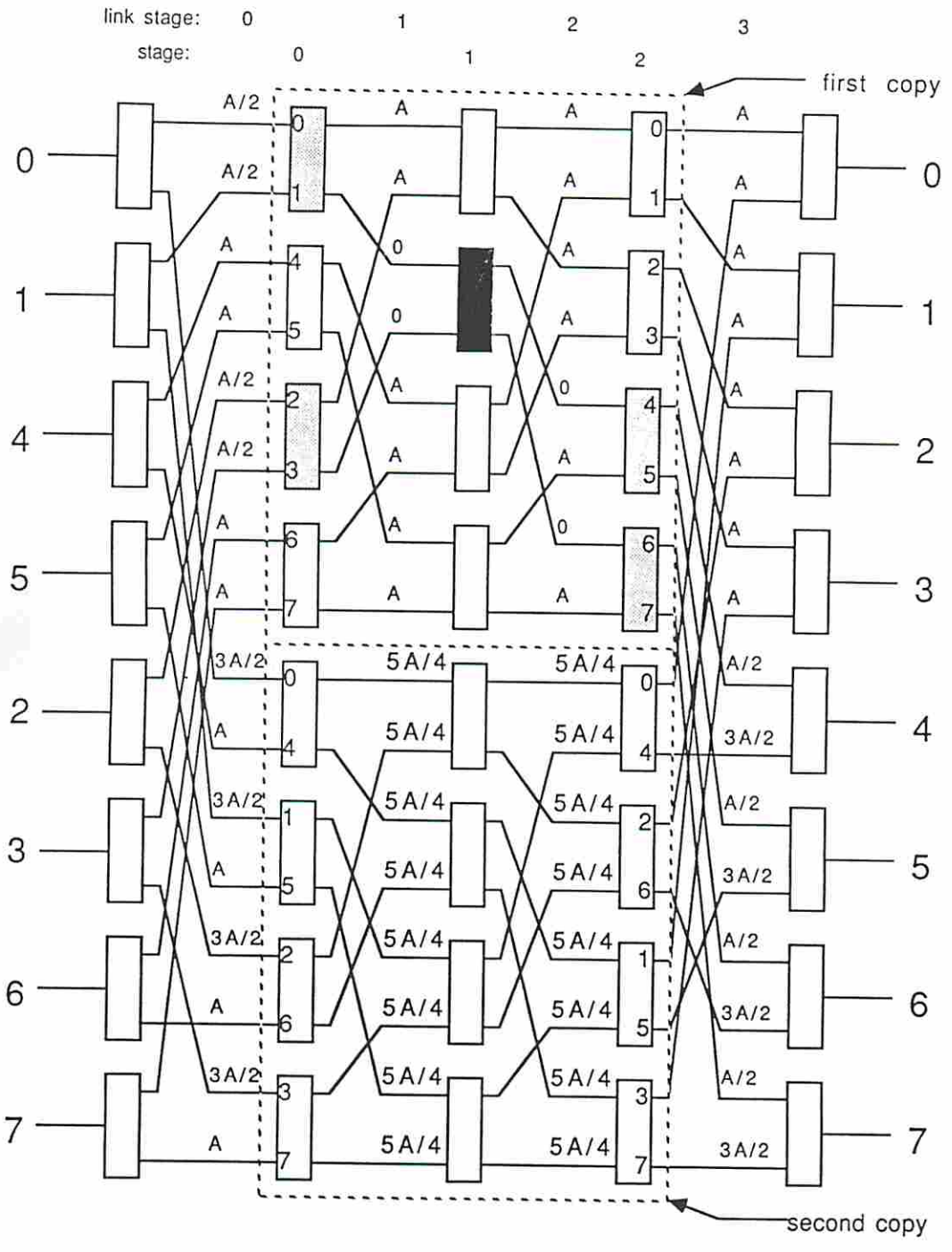


Figure 5.2: Dual MIN of Size 8

To prove that the bandwidth of the dual MIN is maximal for any single failure, we consider the failure at a particular stage and then extend it to all stages.

Lemma 2 *The bandwidth of a two copy network under a uniform independent traffic pattern with a single switch module failure, at stage s, is maximal if and only if all the destination sets forked after stage s in the failure subnetwork are forked before stage $n-s-1$ in the other subnetwork and all the source sets merged before stage s in the failure subnetwork are merged after stage $n-s-1$ in the other subnetwork.* □

Proof:

The traffic through the failed switch module has to be rerouted to the other subnetwork. The paths that need rerouting are the cross product of the source set associated with the outnode and the destination set associated with the innode of the failed switch. There are k^{s+1} sources and k^{n-s} destinations in the cross product set. The traffic flow for each path is $\frac{A}{N}$. Hence, the total rerouted traffic flow is

$$k^{s+1}k^{n-s}\frac{A}{N} = k \cdot A$$

At each link stage, the rerouted traffic has to be shared by some links while the other links carry only the original traffic. At link stage 0, only the k^{s+1} links that are in the rerouted source set carry rerouted traffic with traffic flow $k^{n-s} \cdot \frac{A}{N}$ per link.

At link stage $i \leq n-s-1$, the rerouted source traffic has not yet been merged but the rerouted destination traffic has been forked stage by stage. Hence $k^{s+1} \cdot k^i$ links share the rerouted traffic and the rerouted traffic flow is $1 \cdot k^{n-s-i} \cdot \frac{A}{N}$ per link. After stage $n-s-1$, the rerouted source traffic is merged stage by stage but the rerouted destination traffic is not forked any more. At link stage $i \geq n-s$, there are

$$k^{s+1-[i-(n-s-1)]} \cdot k^{n-s} = k^{n-i} \cdot k^{n-s}$$

links with rerouted traffic flow $k^{s+1-n+i} \cdot 1 \cdot \frac{A}{N}$ per link. Note that, in addition to the rerouted traffic, the second subnetwork still carries its original traffic. Since the rerouted packets are added on top of the original uniformly distributed traffic, the links carrying rerouted packets are more heavily loaded than the other links. Therefore these links will saturate first as more traffic is added to the network. If there are less links sharing the rerouted traffic or more rerouted traffic in these links, the network will be saturated with a smaller traffic flow.

Assume that for some stage j, $j < n-s-1$, there is a source merging operation for rerouted traffic at stage j, the number of links that share the rerouted traffic after stage j will be decreased by a factor of k and the rerouted traffic flow will be increased by a factor of k per link. Similarly, if there is a destination forking operation for rerouted traffic at stage j, the number of links that share the rerouted traffic after stage j will be decreased by a factor of k and the rerouted traffic flow will be increased by a factor of k per link. Between stage j and s, each link stage has less links sharing rerouted traffic, hence less bandwidth.

Conversely, since there are $s+1$ source merging operations required for the rerouted packets and at most $s+1$ source forking operations after stage $n-s-1$, it is not possible to move any other source forking operation to after stage $n-s-1$. Similarly, since there are $n-s$ destination forking operations required for the rerouted packets and only $n-s-1+1$ possible destination forking operations before stage $n-s-1$, it is not possible to move any other destination forking operation to before stage $n-s-1$. There is no other source merging, destination forking sequence for which the rerouted traffic is shared over *more* links. QED

Theorem 8 *The bandwidth of the two copy MIN under a uniform independent traffic pattern with a single switch failure is maximal if and only if it is a dual MIN.* □

Proof:

For the failure in either subnetwork, the blocked traffic will be routed to the other subnetwork. Since the source merging sequence and the destination forking sequence are reversed, from the lemma, the bandwidth is maximal.

Conversely, assume that the two copy network is not a dual network. First, consider the case that it is due to the source merging sequence. We can always find a source merging operation at stage i and the corresponding source merging operation of the other subnetwork at stage $n-j-1, j > i$. If the switch failure is at a stage $s, j > s \geq i$, then the source merging operation for the rerouted traffic of the other subnetwork occurs before stage $n-s-1$. From the lemma, the network is not maximal. Similarly, consider the case that it is due to the destination forking sequence. We can always find a destination forking operation at stage i and the corresponding destination forking operation of the other subnetwork at stage $n-j-1, j < i$. If the switch failure is at a stage s, $j < s \leq i$, then the destination forking operation for the rerouted traffic of the other subnetwork occurs after stage $n-s-1$. QED

5.1.3 Dual extra stage MINs

Another way to construct a multipath MIN is by introducing an extra stage[3] in front of a unique path MIN as shown in Figure 5.3. In this example, the dark switches form a unique path subnetwork with size $\frac{N}{2}$ while the shaded switches form another unique path subnetwork. Any source can access all destinations via each outnode of the switch that it is attached to, hence two paths are provided. The network reliability is increased by adding only one stage of switches instead of a complete copy of unique path MIN. The bandwidth of this network with a single switch failure can also be increased by proper topological design. In this section, only MINs using 2×2 cross bar switch modules are considered.

Definition 12 *The dual extra stage MIN is an extra stage MIN using 2×2 crossbar switch modules such that, excluding the extra and last stages, the network is divided into two subnetworks which are dual, with reversed memory forking sequence and reversed processor merging sequence.* □

An example of dual extra stage MIN is shown in Figure 5.4. The characteristic of dual extra stage MIN is very similar to the dual MIN as shown by the following theorem.

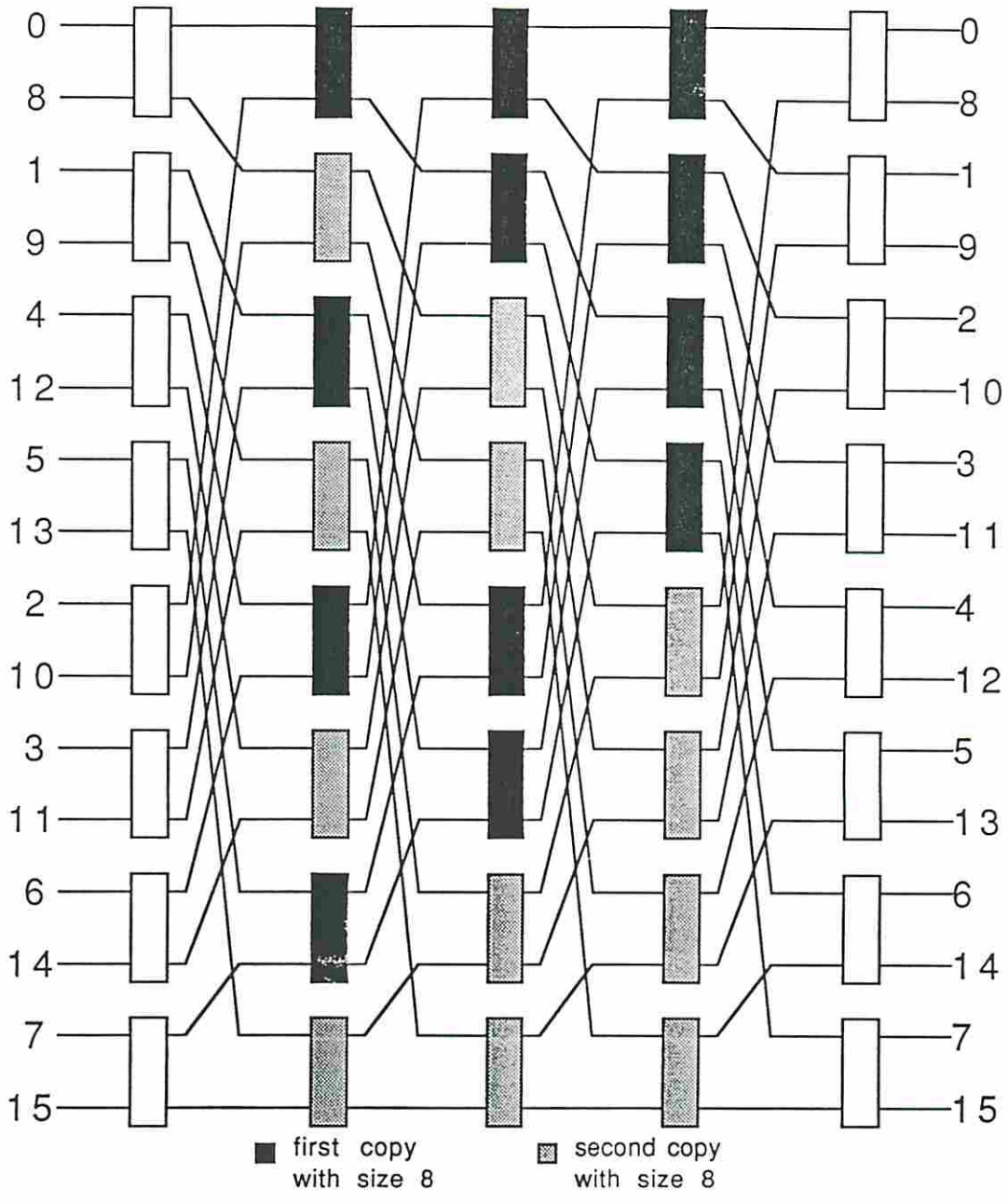


Figure 5.3: Extra Stage MIN of Size 16

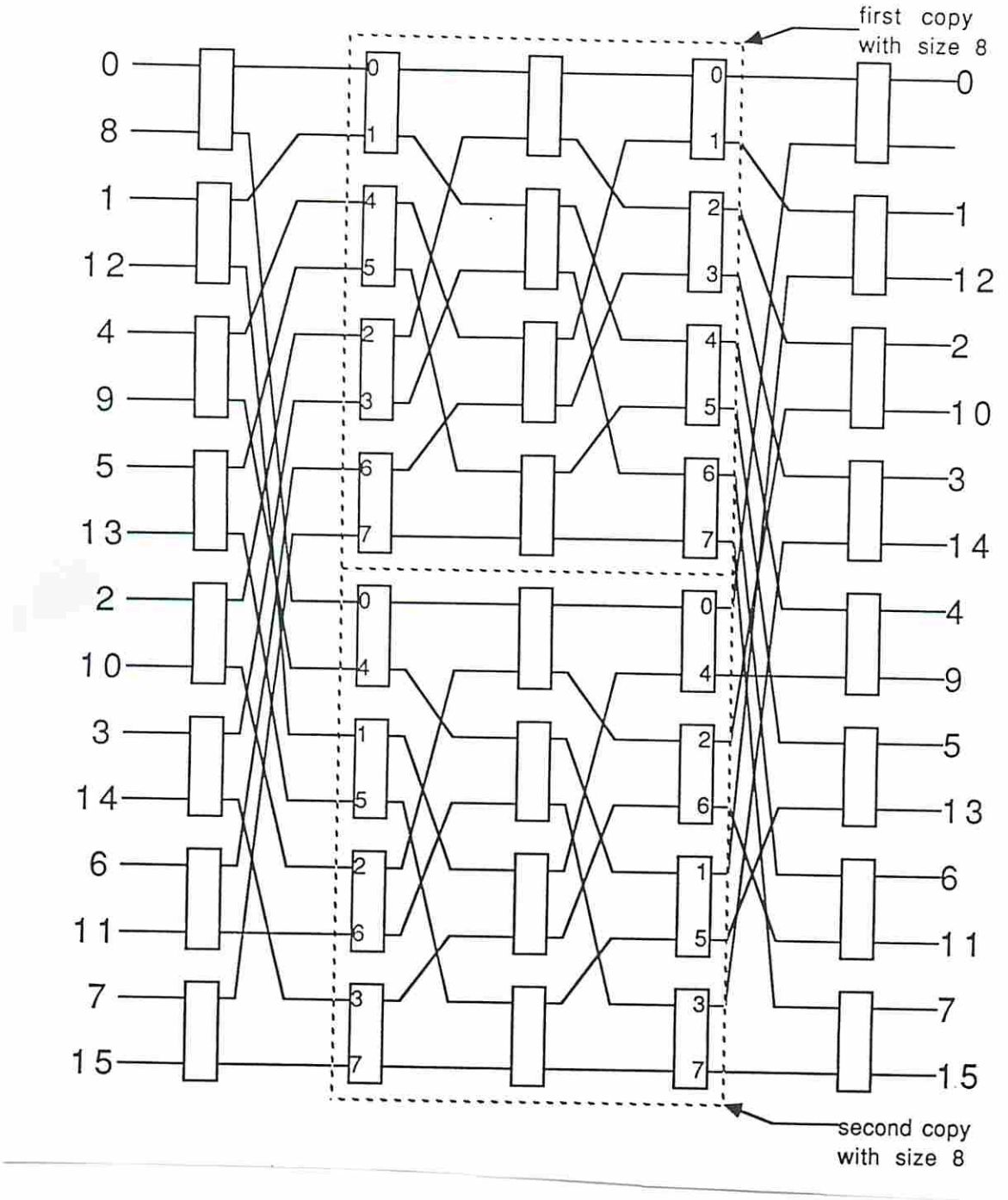


Figure 5.4: Dual Extra Stage MIN of Size 16

Theorem 9 *If a single switch failure occurs in an interior stage, the bandwidth of a dual extra stage MIN under a uniform independent traffic pattern is maximal among the extra stage MINs using* 2×2 *switches.* □

Proof:

The interior subnetwork acts like a two copy network of size $\frac{N}{2}$. The bandwidth of the interior subnetwork with a single switch failure is maximal if and only if it is a dual MIN. Hence the dual extra stage MIN is maximal. QED

As suggested by[3], if the switch failure occurs at the extra stage or the last stage, bypass links are required to maintain the connectivity. In this case, the dual extra stage MIN offers the same performance as any other extra stage MIN.

5.1.4 Simulation results for two path MINs

The *normalized bandwidth*, which is the bandwidth per copy, of the normal two copy MIN and the dual MIN under a single failure have been studied by simulation. In Figure 5.5, normalized bandwidth vs. location of failure (i.e., which stage) of a six stage two copy MIN with 2×2 switches is depicted. The switch buffer size is 4. The top curve is for the failure free case. The bottom curve is for a single failure where the two subnetworks use the same source merging sequence and the same destination forking sequence. A failure occurring near the sources is worse because blocking due to finite buffer size propagates back to the sources. The middle curve shows the normalized bandwidth for a dual MIN. If the failure occurs at interior stage, the normalized bandwidth is quite close to the failure free case.

In Figure 5.6, normalized bandwidth vs. number of sources is shown (also with switch buffer size of 4). The normalized bandwidth shown is the average over all possible single failures. It is shown that the improvement of the dual MIN is slightly larger as the network size increases. Hence, we can conclude that at least a *constant* improvement is achieved over all network size.

In Figure 5.7, normalized bandwidth vs. switch buffer size is plotted. The improvement of the dual MIN is observed over all switch buffer sizes. Although the normalized bandwidth for small buffer size is not great, there is still some observable improvement.

In Figure 5.8, bandwidth vs. location of failure of a six stage (one extra stage and five stages for the unique path MIN) extra stage MINs with 2×2 switch modules and 4 switch buffers is depicted. The failures at the extra stage and the last stage are not shown since they destroy the connectivity of the network. A similar improvement can be observed in these curves.

5.2 Network reliability of two path MINs

Full access capability, i.e. at least one path exists between any source/destination pair, is a good reliability criterion and has been used by many researchers [2]. The switch module fault model is used in this analysis, and also it is assumed that the failure probabilities of the switches are independent and identical.

In a two path MIN, a single failure can be tolerated, while multiple failures may destroy the network full access capability. However, some multiple failures do not destroy the network full access capability. The failure of any of the switch modules carrying rerouted traffic destroys the full access capability. For the normal two copy MIN in Figure 5.1, there are five such switches,

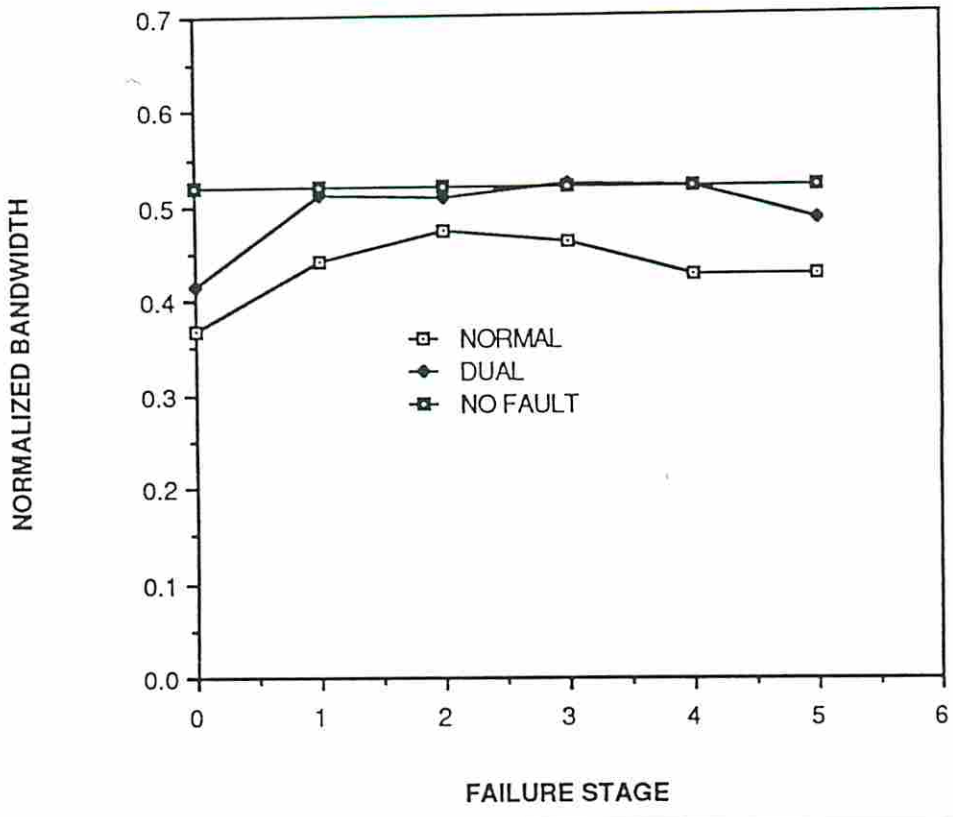


Figure 5.5: Normalized Bandwidth versus Failure Stage for Two Copy MINs

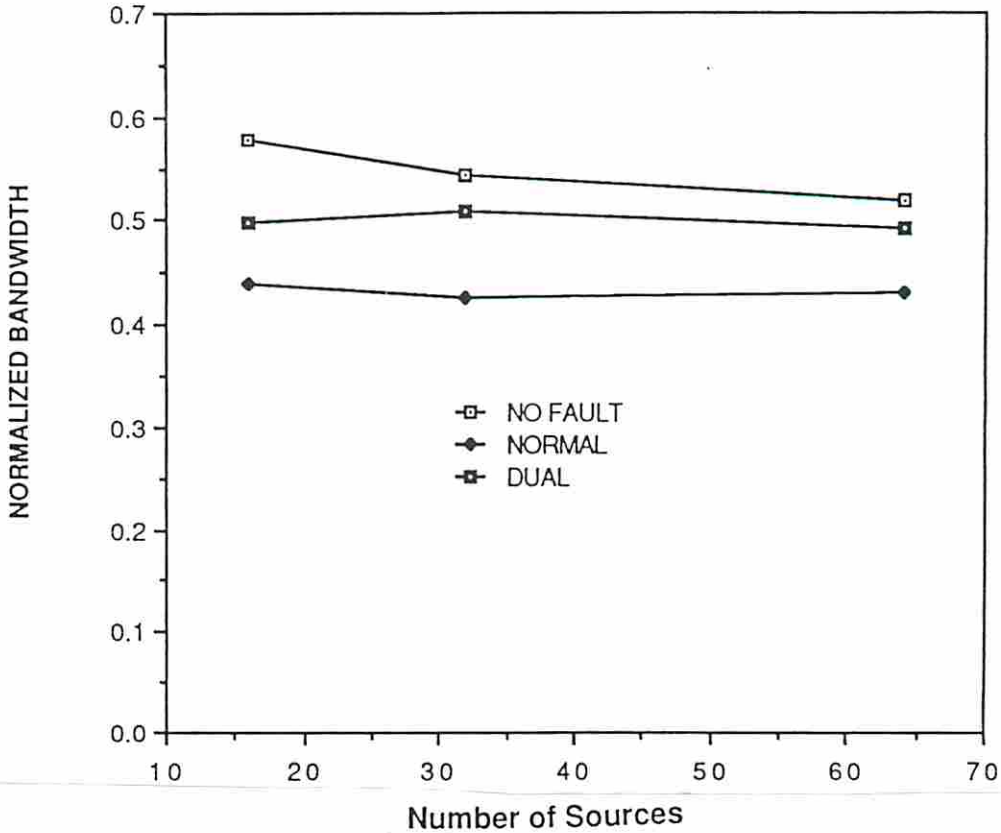


Figure 5.6: Normalized Bandwidth versus number of sources for Two Copy MINs

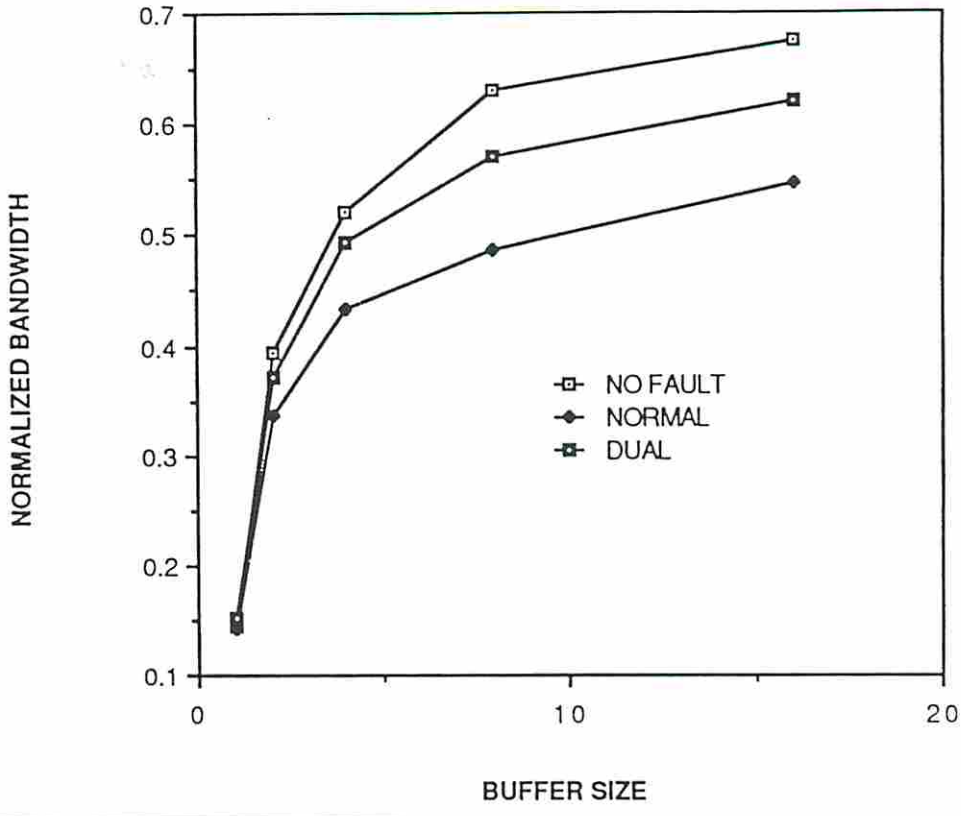


Figure 5.7: Normalized Bandwidth versus buffer size for Two Copy MINs

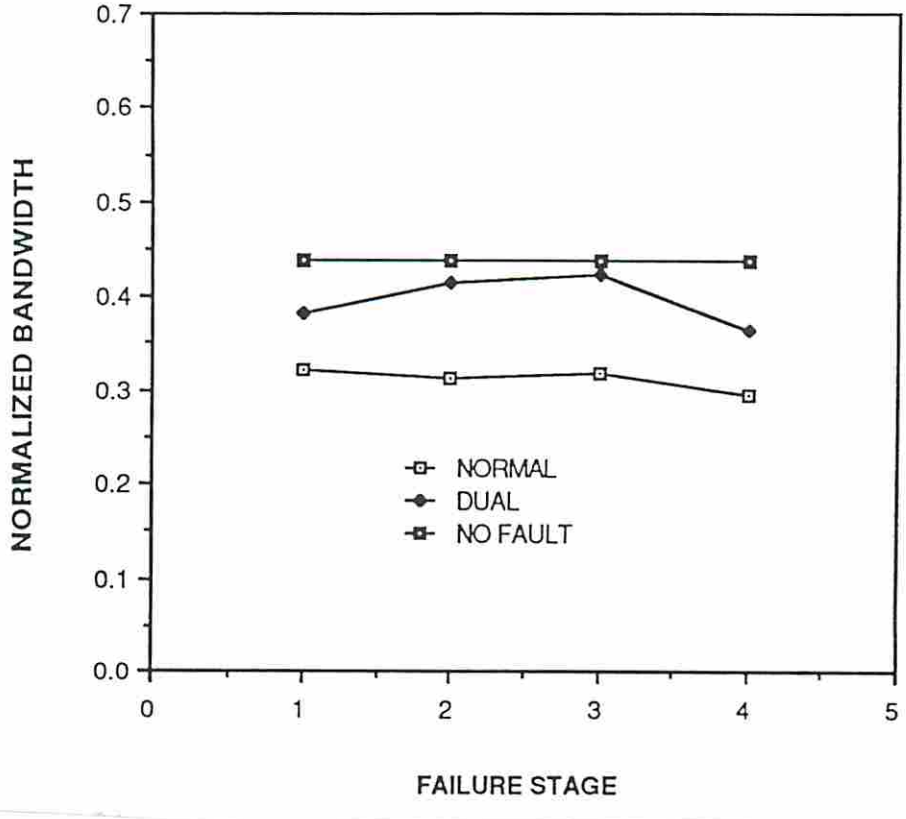


Figure 5.8: Normalized Bandwidth versus Failure Stage for Extra Stage MINs

whereas for the dual MIN in Figure 5.2, failure of any switch in the second subnetwork is a problem (they all carrying rerouted traffic).

It is obvious that the network reliability of a normal two copy MIN is better than that of a dual MIN, which is the price paid for more switches sharing rerouted traffic in order to reduce the performance bottleneck. The penalty is insignificant in the range of interest of failure probabilities, however, where the high failure count is of less importance.

5.2.1 Reliability bounds of two copy MINs

Consider a failure pattern V in the first subnetwork. The number of switch failures in the pattern V is $k(V)$ and the number of switches *covered by* the failure pattern V (i.e., the number of switches carrying rerouted traffic in the second subnetwork) is $r(V)$. Let $S = \frac{NlogN}{2}$ be the number of switches in the n stage unique path MIN. Assuming that the reliability of a switch is p, then the network reliability for a two copy MIN is

$$R^{2c} = \sum_{all\ V} p^{S-k(V)}(1-p)^{k(V)}p^{r(V)}$$

The exact solution of network reliability is quite involved, and hence only an upper bound and lower bound are studied.

The network reliability can also be computed as follows:

$$R^{2c} = \sum_{k=0}^{S} R_k$$

where R_k = Pr{full access capability exists | exactly k failures in any one subnetwork and at least k failures in the other subnetwork}. For three failures in a two copy MIN, they can be grouped as $(3,0)$, $(2,1)$, $(1,2)$ or $(0,3)$, where the first component is the number of switch failures in first subnetwork and second component is the number of switch failures in the second network. $(3,0)$, $(0,3)$ are contained in R_0 while $(2,1)$, $(1,2)$ are contained in R_1. Hence, R_0 and R_1 include all the failure patterns for less than or equal to three failures and for some higher failure counts. $R_0 + R_1$ is thus a reasonable lower bound for the reliability range of interest. (Since two copy MINs are designed to be used with only one or two failures.) The lower bound $\underline{R}^{2c}$ is:

$$\underline{R}^{2c} = R_0 + R_1 < R^{2c}$$

Evaluation of R_0 and R_1 is given in the Appendix.

Let X be Pr{full access capability exists | each copy has at least two failures} and

$$X = \sum_{k=2}^{S} R_k = R^{2c} - R_0 - R_1$$

For any failure pattern of the first subnetwork, let us arbitrarily choose one of the failed switches as a tagged switch. Let P_s = Pr{the tag switch is at stage s and there are at least two failures in the first subnetwork}. Then,

$$P_s = \frac{1 - p^S - Sp^{S-1}(1-p)}{n}$$

Let X_s be the conditional probability of the network is operational when there are at least two failures and the tagged switch is at stage s. Then

$$X = \sum_{s=0}^{n-1} P_s X_s$$

X_s is very hard to compute. A simple upper bound can be obtained by considering only those switches in the second subnetwork which carry extra traffic due to the failure of the tagged switch.

$$X_s < \overline{X}_s$$

where $\overline{X}_s$ is an upper bound given by:

$$\overline{X}_s = p^{\Gamma_s}[1 - p^{S-\Gamma_s} - (S - \Gamma_s)p^{S-\Gamma_s-1}(1-p)]$$

where Γ_s is the number of switches in the second subnetwork covered by the single failure of the tagged switch in stage s of the first subnetwork. The upper bound of the network reliability, $\overline{R}^{2c}$, is:

$$\overline{R}^{2c} = R_0 + R_1 + \sum_{s=0}^{n-1} P_s \overline{X}_s$$

By combinatorial calculation, Γ_s can be obtained and are given in the Appendix A.

The reliability of a unique path MIN, upper and lower bounds on the reliabilities of a normal two copy MIN and a dual MIN are plotted in Figures 5.9 and Figure 5.10 for 5 and 10 stages respectively. The bounds are satisfactory in the range of interest where a small number of failures is more important. The improvement over the unique path MIN is quite significant for both the two copy MIN and the dual MIN. The difference between these two networks is relatively small in the range of interest. Therefore, the penalty paid for the dual MIN is acceptable.

5.2.2 Reliability bounds of extra stage MINs

In the following analysis, it is assumed that there is no bypass link for the switches at the extra stage and the last stage, therefore, any failure in these two stages destroys the network full access capability. As shown in Figure 5.3, the interior stages form two unique path MINs with the size $\frac{N}{2}$ and reliability R^{2c}. The network reliability of an extra stage MIN, R^x, therefore, is

$$R^x = p^{2\cdot\frac{N}{2}} R^{2c} = p^N R^{2c}$$

The bounds of the network reliabilities of the normal extra stage MIN and the dual extra stage are shown in Figure 5.11. The bounds are quite satisfactory since the high failure count case is insignificant because of the fault free requirement of the extra and last stages. A clear improvement of extra stage MIN and dual extra stage MIN over the unique path MIN is observed over the range of interest, as expected.

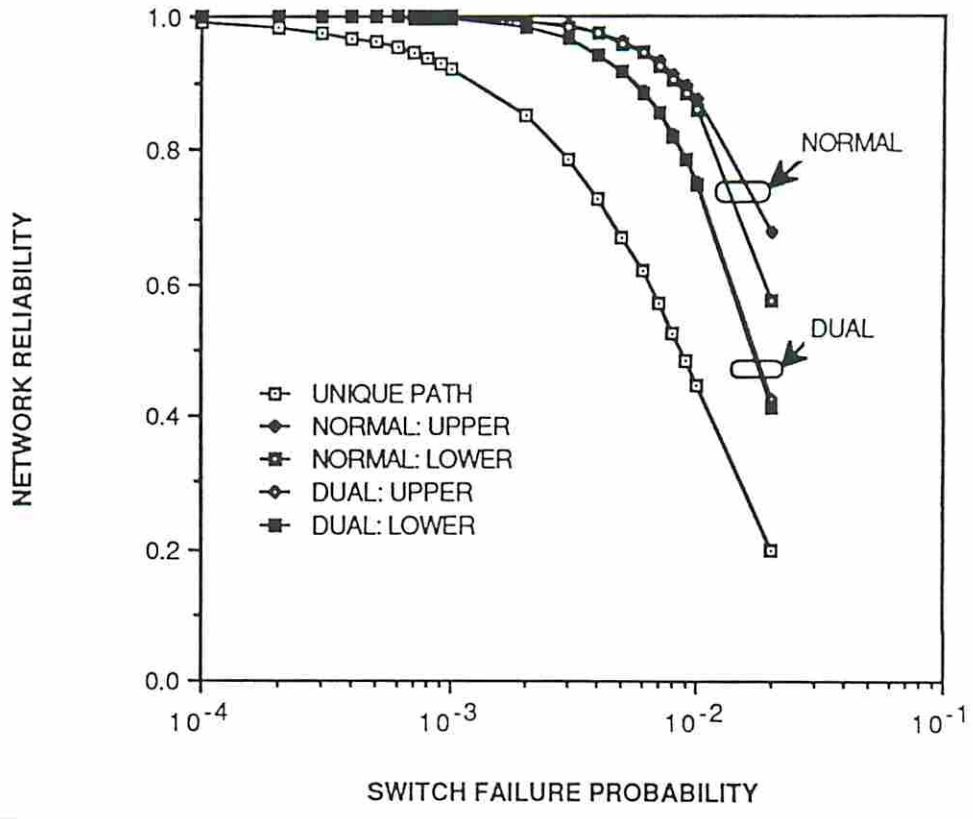


Figure 5.9: Network Reliability for Five Stage MINs

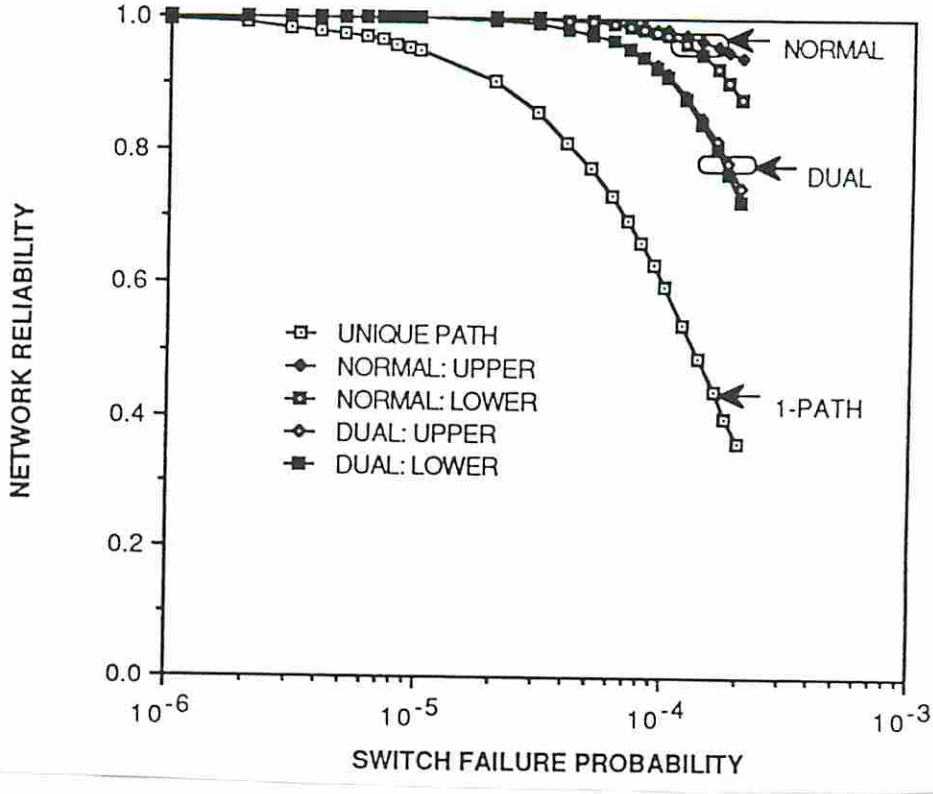


Figure 5.10: Network Reliability for Ten Stage MINs

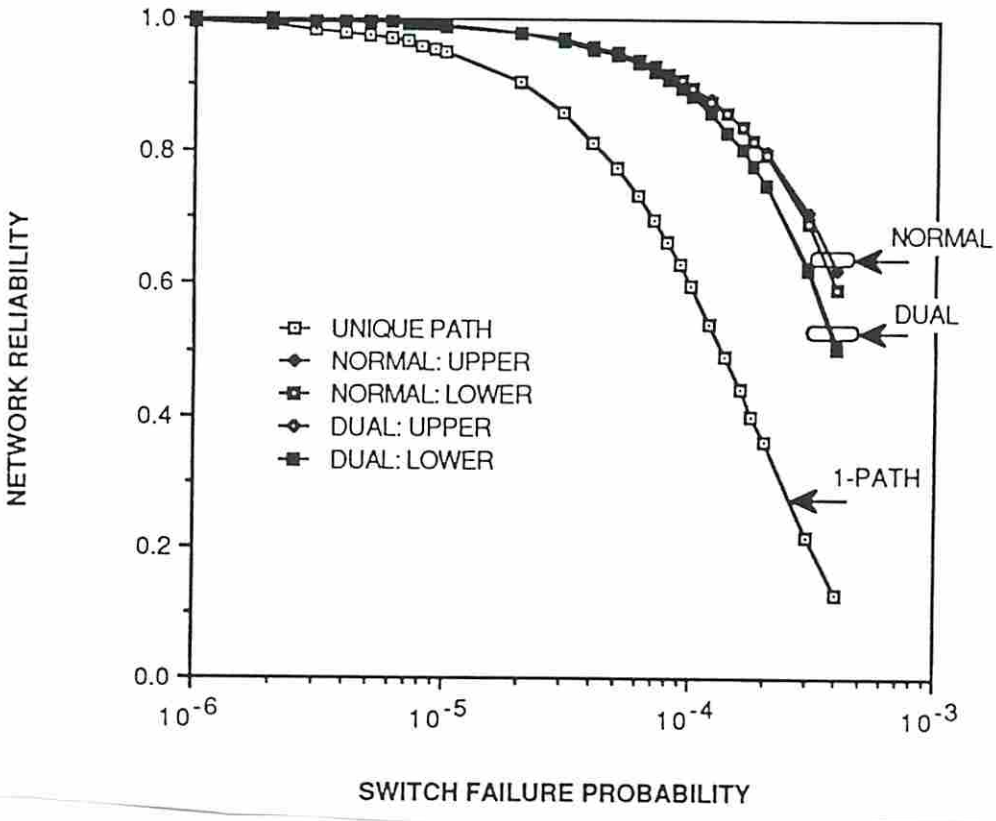


Figure 5.11: Network Reliability for 10+1 Extra Stage MINs

Chapter 6

Conclusion

6.1 Summary of the Main Results

As the speed of fast packet switching approaches Gbps, the characteristics of switch design are changed. Parallelism of the switching functions is inevitable. Synchronization and layout of parallel lines at the board level design are difficult to achieve for that speed. Pin limitations due to VLSI packaging also discourage parallel off-chip connections. Therefore, it seems appropriate to use parallel switching within a VLSI chip and serial interconnection off the chip. Since it is difficult to use copper wires for the long interconnections necessary in a large switch, fiber optics are proposed for the connections between modules or boards.

Highly integrated VLSI technology makes larger switch modules (8×8 to 32×32) with sufficient buffers feasible. Thus the number of stages of the interconnection network is significantly reduced and the large queueing delay due to the internal buffers is no longer a problem. Switch fabrics of size larger than 1000×1000 are achievable with our approach based on a module utilizing traditional internal buffered Banyan type interconnection networks.

It is clear that reliability is an important issue for a very fast, large ATM switch. Furthermore, for the speed that approaches electronic device limitations, the routing functions which are the core functions of switching system must be kept to a minimum. A change in functional topology during failure requires a complicated routing scheme. Therefore, a topology invariant mode for failure is vary desirable in fast packet switching.

We propose a novel replacement network that does not change the functional topology of network for fault tolerable failure patterns. An abstract replacement model is proposed to study this problem. The maximum fault tolerance, replacement algorithm and hardware overhead requirements are studied with this replacement model. The network is shown to achieve excellent fault tolerance with relatively a small amount of hardware overhead [2] and has short down time for recovery from fault tolerable failures with our fast run time replacement algorithm.

We also make use of this reconfiguration ability to alleviate the problem of unbalanced traffic flow. This becomes increasingly important as the network provides integrated services at higher rates. We formulate the topology design problem as a means to balance traffic flow and show that it is NP-hard even under a fixed unique path MIN functional topology. We also provide a heuristic algorithm which seems to have good performance.

The performance degradation of FTMIN design due to irregular topology under failure mode of operation is also observed. A new class of FTMINs, call dual MINs, is proposed. It is shown

that the new networks improve the failure mode network performance significantly without any hardware overhead. The tradeoffs are achieved at the expense of some reduction in network reliability. It is also shown that the degradation of network reliability is negligible in the range of interest.

6.2 Future Research

Many potential future research topics are raised during this research.

- *Dependent replacement model*:
  Many interesting FTMINs can run in the TI mode and can be modeled as a dependent replacement model. In general, the replacement algorithm is *NP*-complete [65]. It is interesting to study the mathematical structure of a dependent replacement model.
- *Topology design for single stage interconnection network*:
  It has been shown that the topology design problem for the functional topology of unique path MIN is *NP*-hard. A more complicated topology: the ShuffleNet (or a single stage interconnection network) is used in the multichannel multihop lightwave network. It is interesting to find an efficient heuristic algorithm for this class of network.
- *Topology design for any given functional topology*:
  The are many other interesting functional topologies, e.g. Batcher Banyan internal non-blocking network [10]. It is interesting to know the complexity of these problems. Many interesting classes of problems can be formulated for a special functional topology or internal structure.
- *Performance bounds for topology design heuristics*:
  While finding an efficient exact solution to the topology design problem is impossible, it is interesting to find bounds on the performance of various heuristics.
- *Reliability of network with TI mode*:
  The reliability of a generic network topology is known to be a *NP*-hard problem [27]. It is interesting to know if the reliability problem for a class of fault tolerant networks under topology invariant mode is still *NP*-hard.
- *Dual property for multi-path MIN*:
  Tradeoffs of performance and reliability of the two-path MIN have been shown. It is interesting to see how the routing scheme should be rearranged for multi-path (> 2) MINs.

Bibliography

[1] A. S. Acampora. "A multichannel multihop local lightwave network". In *GLOBECOM'87*, pages 1459–1467, 1987.

[2] G. B. Adams III, D. P. Agrawal, and H. J. Siegel. "A survey and comparison of fault-tolerant interconnection networks". *IEEE Comput. Mag.*, Vol. C-36:14–27, Jun. 1987.

[3] G. B. Adams III and H. J. Siegel. "The extra stage cube: A fault tolerant interconnection network for supersystems". *IEEE Trans. Comput.*, Vol. C-31:443–454, May 1982.

[4] D. P. Agrawal. "Graph theoretical analysis and design of interconnection networks". *IEEE Trans. Comput.*, Vol. C-32:637–648, Jul. 1983.

[5] H. Armbruster. "Applications of future broad-band services in the office and home". *IEEE J. Select. Areas Commun.*, Vol. SAC-4(No. 4):429–437, Jul. 1986.

[6] H. Armbruster and G. Arndt. "Broadband communication and its realization with broad-band isdn". *IEEE Commun. Mag.*, Vol. 25(No. 11):8–19, Nov. 1987.

[7] K. Asatani, K. R. Harrison, and R. Ballart. "CCITT standardization of network node interface of synchronous digital hierarchy". *IEEE Commun. Mag.*, Vol. 28(No. 8):15–20, Aug. 1990.

[8] R. Ballart and Y. C. Ching. "SONET: Now it's the standard optical network". *IEEE Commun. Mag.*, Vol. 27(No. 3):8–15, Mar. 1989.

[9] J. A. Bannister and M. Gerla. "Design of wavelength-division optical network". Technical report, Tech. Report, UCLA, CSD-890022, 1989.

[10] K. E. Batcher. "Sorting networks and their applications". In *Proc. 1968 Spring Joint Comput. Conf.*, pages 307–314, 1968.

[11] D. V. Batorsky, D. R. Spears, and A. R. Tedesco. "The evolution of broadband network architectures". In *GLOBECOM'88*, pages 367–373, 1988.

[12] O. Beau, J. L. Paul, and X. Penet. "Technical concept in broadband ATM". *International Journal of Digital and Analog Cabled Systems*, Vol. 1:207–211, 1988.

[13] V. E. Beneš. *"Mathematical Theory of Connecting Networks and Telephone Traffic"*. Academic Press, New York, 1965.

[14] J. E. Berthold. "Broadband electronic switching". *IEEE Lightwave Communication Systems*, Vol. 1(No. 2):35–39, May. 1990.

[15] D. Bertsekas and R. Gallager. *"Data Networks"*. Prentice-Hall, Inc., New York, 1987.

[16] R. J. Boehm. "Progress in standardization of SONET". *IEEE Lightwave Communication Systems*, Vol. 1(No. 2):8–16, May. 1990.

[17] L. Bosack and C. Hedrick. "Problems in large LANs". *IEEE Network*, Vol. 2(No. 1):49–56, Jan. 1988.

[18] C. A. Brackett. "Dense wavelength division multiplexing networks: Principles and applications". *IEEE J. Select. Areas Commun.*, Vol. SAC-8(No. 6):948–964, Aug. 1990.

[19] R. G. Bubenik and J. S. Turner. "Performance of a broadcast packet switch". *IEEE Trans. Commun.*, Vol. 37(No. 1):60–69, Jan. 1989.

[20] W. R. Byrne, T. A. Kilm, B. L. Nelson, and M. D. Soneru. "Broadband ISDN technology and architecture". *IEEE Network*, Vol. 3:23–28, Jan. 1989.

[21] I. Chlamtac, A. Ganz, and G. Karmi. "Purely optical networks for terabit communication". In *INFOCOM'89*, pages 887–896, 1989.

[22] I. Chlamtac, A. Ganz, and G. Karmi. "Lightnet: Lightpath based solutions for wide bandwidth wans". In *INFOCOM'90*, pages 1014–1021, 1990.

[23] L. Ciminiera and A. Serra. "A connecting network with fault tolerance capabilities". *IEEE Trans. Comput.*, Vol. C-35:578–580, Jun. 1986.

[24] P. Cochrane and M. Brain. "Future optical fiber transmission technology and networks". *IEEE Commun. Mag.*, Vol. 26(No. 11):45–60, Nov. 1988.

[25] P. Cochrane, R. Brooks, and R. Dawes. "A high-reliability 565 Mbit/s trunk transmission system". *IEEE J. Select. Areas Commun.*, Vol. SAC-4(No. 9):1396–1403, Dec. 1986.

[26] L. G. Cohen. "Trends in U.S. broad-band fiber optic transmission systems". *IEEE J. Select. Areas Commun.*, Vol. SAC-4(No. 4):488–497, Jul. 1986.

[27] C. J. Colbourn. *"The Combinatorics of Network Reliability"*. Oxford University Press, New York, 1987.

[28] J. P. Coudreuse. "ATM: A contribution to the debate on broadband ISDN". *International Journal of Digital and Analog Cabled Systems*, Vol. 1:213–221, 1988.

[29] G. E. Daddis, Jr. and H. C. Torng. "A taxonomy of broadband integrated switching architectures". *IEEE Commun. Mag.*, Vol. 27(No. 5):32–42, May 1989.

[30] D. M. Dias and J. R. Jump. "Analysis and simulation of buffered Delta networks". *IEEE Trans. Comput.*, Vol. C-30:273–282, Apr. 1981.

[31] N. Dono, P. E. Green, Jr, K. Liu, R. Ramaswami, and F. F. Tong. "A wavelength division multiple access network for computer communication". *IEEE J. Select. Areas Commun.*, Vol. SAC-8(No. 6):983–994, Aug. 1990.

[32] S. Dutt and J. P. Hayes. "On designing and reconfiguring k-fault-tolerant tree architectures". *IEEE Comput. Mag.*, Vol. C-39:490–503, Apr. 1990.

[33] M. Eisenberg and N. Mehravari. "Performance of the multichannel multihop lightwave network under nonuniform traffic". *IEEE J. Select. Areas Commun.*, Vol. SAC-6(No. 7):1063–1078, Aug. 1988.

[34] K. Y. Eng. "A photonic knockout switch for high-speed packet networks". *IEEE J. Select. Areas Commun.*, Vol. SAC-6(No. 7):1107–1116, Aug. 1988.

[35] K. Y. Eng and M. G. Hluchyj. "A knockout switch for variable-length packets". *IEEE J. Select. Areas Commun.*, Vol. SAC-5:1426–1435, Dec. 1987.

[36] K. Y. Eng, M. J. Karol, and Y. S. Yeh. "A growable packet (ATM) switch architecture: Design principles and applications". In *GLOBECOM'89*, pages 1159–1165, 1989.

[37] S. Even. *"Graph Algorithms"*. Computer Science Press, Maryland, 1979.

[38] G. J. Foschini. "Using spread spectrum in a high-capacity fiber-optic local network". *Journal of Lightwave Tech.*, Vol. LT-6(No. 3):370–379, Mar. 1988.

[39] M. Frame. "Broadband service needs". *IEEE Commun. Mag.*, Vol. 28(No. 4):59–62, Apr. 1990.

[40] CCITT Recommendation G.708. *"Synchronous Digital Hierarchy Bit Rates"*, 1988.

[41] H. N. Gabow and R. E. Tarjan. "Almost-optimum speed-ups of algorithms for bipartite matching and related problems". In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pages 514–527, 1988.

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

[43] M. Gerla, J. A. S. Monteiro, and R. Pazos. "Topology design and bandwidth allocation in atm nets". *IEEE J. Select. Areas Commun.*, Vol. SAC-7(No. 8):1253–1262, Oct. 1989.

[44] M. S. Goodman, H. Kobrinski, M. Vecchi, R. Bulley, and J. L. Gimlett. "The LAMBDANET multiwavelength network: Architecture, applications, and demonstrations". *IEEE J. Select. Areas Commun.*, Vol. SAC-8(No. 6):995–1004, Aug. 1990.

[45] R. Handel. "Evolution of ISDN towards broadband ISDN". *IEEE Network*, Vol. 3:7–13, Jan. 1989.

[46] C. F. Hemrick, R. W. Klessig, and J. M. McRoberts. "Switched multi-megabit data service and early availability via man technology". *IEEE Commun. Mag.*, Vol. 26(No. 4):9–14, Apr. 1988.

[47] P. Henry. "High-capacity lightwave local area network". *IEEE Commun. Mag.*, Vol. 27(No. 10):20–26, Oct. 1989.

[48] M. G. Hluchyj and M. J. Karol. "Queueing in high-performance packet switching". *IEEE J. Select. Areas Commun.*, Vol. SAC-6(No. 9):1587–1597, Dec. 1988.

[49] R. Holter. "SONET: A network management viewpoint". *IEEE Lightwave Communication Systems*, Vol. 1(No. 4):4–13, Nov. 1990.

[50] E. Horowitz and A. Sahni. *"Fundamentals of Data Structures"*. Computer Science Press, Inc., Rockville, 1982.

[51] A. Huang and S. Knauer. "Starlite: A wideband digital switch". In *GLOBECOM'84*, pages 121–125, 1984.

[52] J. Y. Hui. "Network, transport, and switching integration for broadband communications". *IEEE Network*, Vol. 3:40–41, Mar. 1989.

[53] J. Y. Hui. *"Switching and Traffic Theory for Integrated Broadband Networks"*. Kluwer Academic Publishers, 1990.

[54] J. Y. Hui and E. Arthurs. "A broadband packet switch for integrated transport". *IEEE J. Select. Areas Commun.*, Vol. SAC-5(No. 8):1264–1273, Oct. 1987.

[55] J. H. Irven, M. E. Nilson, T. H. Judd, J. F. Patterson, and Y. Shibata. "Multi-media information services: A laboratory study". *IEEE Commun. Mag.*, Vol. 26(No. 6):27–44, Jun. 1988.

[56] M. Jeng and H. J. Siegel. "A fault-tolerant multistage interconnection network for multiprocessor systems using dynamic redundancy". In *6th Int'l Conf. Distributed Computing Systems, Computer Society Press, Silver Spring, Md.*, pages 70–77, 1986.

[57] R. Johnson. "Multichip modules: Next-generation packages". *IEEE Spectrum*, Vol. 27(No. 3):34–48, Mar. 1990.

[58] P. Kaiser. "Status and future trends in terrestrial optical fiber systems in north america". *IEEE Commun. Mag.*, Vol. 25(No. 10):8–13, Oct. 1987.

[59] H. Kobrinski and K. W. Cheung. "Wavelength-tunable optical filters: Applications and technologies". *IEEE Commun. Mag.*, Vol. 27(No. 10):53–63, Oct. 1989.

[60] J. Kohli. "Medical imaging applications of emerging broadband networks". *IEEE Commun. Mag.*, Vol. 27(No. 12):8–16, Dec. 1989.

[61] J. S. Kohli, D. S. Biring, and G. L. Raya. "Emerging broadband packet-switch technology in integrated information networks". *IEEE Network*, Vol. 2:37–51, Nov. 1988.

[62] L. Kohn and Sai Wai Fu. "A 1,000,000 transistor microprocessor". In *International Solid-State Circuits Conference*, pages 54–55, Feb. 1989.

[63] C. P. Kruskal and M. Snir. "The performance of multistage interconnection networks for multiprocessors". *IEEE Trans. Comput.*, Vol. C-32(No. 12):1091–1098, Dec. 1983.

[64] V. P. Kumar and S. M. Reddy. "Augmented shuffle-exchange multistage interconnection networks". *IEEE Comput. Mag.*, Vol. 20(No. 6):30–40, Jun. 1987.

[65] S. Y. Kuo and W. K. Fuchs. "Efficient spare allocation for reconfigurable arrays". *IEEE Design and Test*, Vol. 4:24–31, Feb. 1987.

[66] J. P. Labourdette and A. S. Acampora. "Partially reconfigurable multihop lightwave networks". In *GLOBECOM'90*, 1990.

[67] J. P. Labourdette and A. S. Acampora. "Wavelength agility in multihop lightwave networks". In *INFOCOM'90*, pages 1022–1029, 1990.

[68] L. E. Larson, J. F. Jensen, and P. T. Greiling. "GaAs high-speed digital IC technology: An overview". *IEEE Comput. Mag.*, Vol. 19(No. 10):21–27, Oct. 1986.

[69] D. Lawrie. "Access and alignment of data in an array processor". *IEEE Trans. Comput.*, Vol. C-24(No. 12):1145–1155, Dec. 1975.

[70] T. P. Lee and C. E. Zah. "Wavelength-tunable and single-frequency semiconductor lasers for photonic communications networks". *IEEE Commun. Mag.*, Vol. 27(No. 10):42–52, Oct. 1989.

[71] S. Q. Li and M. J. Lee. "A study of traffic imbalances in a fast packet switch". In *INFOCOM'89*, pages 538–547, 1989.

[72] T. Li and R. A. Linke. "Multigigabit-per-second lightwave system research for long-haul applications". *IEEE Commun. Mag.*, Vol. 26(No. 4):29–35, Apr. 1988.

[73] A. Y. M. Lin and J. A. Silvester. "A critical analysis of some design issues in very high speed integrated networks (VHSIN): Recommendations for the next generation of communication systems". Technical report, University of Southern California, EE-Systems, 1990.

[74] R. A. Linke. "Frequency division multiplexed optical networks using heterodyne detection". *IEEE Network*, Vol. 3:13–20, Mar. 1989.

[75] R. A. Linke. "Optical heterodyne communication system". *IEEE Commun. Mag.*, Vol. 27(No. 10):36–41, Oct. 1989.

[76] N. Lippis. "Frame relay redraws the map for wide-area networks". *Data Communications*, pages 80–94, Jul. 1990.

[77] N. Lippis and J. Herman. "The internetwork decade". *Data Communications*, pages s2–s32, Jan. 1991.

[78] R. I. MacDonald. "Terminology for photonic matrix switching". *IEEE J. Select. Areas Commun.*, Vol. SAC-6(No. 7):1141–1151, Aug. 1988.

[79] J. S. McConnell. "The power of a SONET ADM". *Telephony*, pages 130–136, Apr. 1990.

[80] R. J. McMillen and H. J. Siegel. "Performance and fault tolerance improvements in the inverse augmented data manipulator network". In *9th Symp. Comput. Arch.*, pages 63–72, 1982.

[81] R. Melen and J. S. Turner. "Nonblocking networks for fast packet switching". In *INFOCOM'89*, pages 548–557, 1989.

[82] J. Midwinter. "Status and future trends in terrestrial optical fiber systems in europe". *IEEE Commun. Mag.*, Vol. 25(No. 10):14–17, Oct. 1987.

[83] S. E. Minzer. "Broadband ISDN and asynchronous transfer mode (ATM)". *IEEE Commun. Mag.*, Vol. 27(No. 9):17–24, Sep. 1989.

[84] S. R. Nagel. "Optical fiber-the expanding medium". *IEEE Commun. Mag.*, Vol. 25(No. 4):33–43, Apr. 1987.

[85] M. M. Nassehi, F. A. Tobagi, and M. E. Marhic. "Topological design of fiber optics local area networks with application to expressnet". Technical report, Stanford Electronics Laboratories, SEL Tech Report No. 85-271, Mar. 1985.

[86] P. Newman. "A fast packet switch for the integrated service backbone network". *IEEE J. Select. Areas Commun.*, Vol. SAC-6(No. 9):1468–1479, Dec. 1988.

[87] E. Nussbaum. "Communication network needs and technologies - a place for photonic switching?". *IEEE J. Select. Areas Commun.*, Vol. SAC-6(No. 7):1036–1043, Aug. 1988.

[88] C. H. Papadimitriou and K. Steiglitz. *"Combinatorial Optimization: Algorithms and Complexity"*. Prentice-Hall, Inc., New Jersey, 1982.

[89] D. S. Parker and C. S. Raghavendra. "The Gamma network". *IEEE Trans. Comput.*, Vol. C-33:367–373, Apr. 1984.

[90] S. D. Personick. "Photonic switching: Technology and applications". *IEEE Commun. Mag.*, Vol. 25(No. 5):5–8, May 1987.

[91] R. W. Pullen. "SONET and cross-connects: Implications for future fiber networks". *Telephone Engineer & Management*, pages 86–93, Feb. 1989.

[92] T1S1.5/90-058 R2. *"Broadband ISDN ATM Aspects - ATM Layer Functionality and Specification"*, Aug. 1990.

[93] C. S. Raghavendra and A. Varma. "INDRA: A class of interconnection networks with redundant paths". In *1984 Real-Time System Symp., Computer Society Press, Silver Spring, Md.*, pages 153–164, 1984.

[94] S. M. Reddy and V. P. Kumar. "On fault-tolerant multistage interconnection networks". In *1984 Int'l Conf. Parallel Processing, Computer Society Press, Silver Spring, Md.*, pages 155–164, 1984.

[95] M. J. Rider. "Protocols for ATM access networks". *IEEE Network*, Vol. 3:17–22, Jan. 1989.

[96] J. A. Sale. "Emerging optical code-division multiple access communications systems". *IEEE Network*, Vol. 3:31–39, Mar. 1989.

[97] N. B. Sandesara, G. Ray Ritchie, and B. Engel-Smith. "Plan and considerations for SONET deployment". *IEEE Commun. Mag.*, Vol. 28(No. 8):26–33, Aug. 1990.

[98] B. Santo and K. Wollard. "The world of silicon: It's dog eat dog". *IEEE Spectrum*, Vol. 25(No. 9):30–39, Sep. 1988.

[99] S. Shimada. "Status and future trends in terrestrial optical fiber systems in japan". *IEEE Commun. Mag.*, Vol. 25(No. 10):18–21, Oct. 1987.

[100] M. Skov. "Implementation of physical and media access protocols for high-speed networks". *IEEE Commun. Mag.*, Vol. 27(No. 6):45–53, Jun. 1989.

[101] W. E. Stephens and Kenneth C. Young. "Terabit-per-second throughput switches for broadband central office: an overview". *IEEE Lightwave Communication Systems*, Vol. 1(No. 4):20–27, Nov. 1990.

[102] ANSI T1.105-1988. *"American National Standard for Telecommunications - Digital Hierarchy Optical Rates and Formats Specification"*, 1988.

[103] I. Toda. "Migration to broadband ISDN". *IEEE Commun. Mag.*, Vol. 28(No. 4):55–58, Apr. 1990.

[104] P. R. Trischitta and D. T. S. Chen. "Repeaterless undersea lightwave systems". *IEEE Commun. Mag.*, Vol. 27(No. 3):16–21, Mar. 1989.

[105] N. F. Tzeng, P. C. Yew, and C. Q. Zhu. "The performance of a fault-tolerant multistage interconnection networks". In *1985 Int'l Conf. Parallel Processing, Computer Society Press, Silver Spring, Md.*, pages 458–465, 1985.

[106] G. Vannucci. "Combining frequency-division and code-division multiplexing in a high-capacity ottical network". *IEEE Network*, Vol. 3:21–30, Mar. 1989.

[107] R. E. Wagner and R. Linke. "Heterodyne lightwave systems: Moving towards commercial use". *IEEE Lightwave Communication Systems*, Vol. 1(No. 4):28–35, Nov. 1990.

[108] S. S. Wagner and R. C. Menendez. "Evolutionary architectures and techniques for video distribution on fiber". *IEEE Commun. Mag.*, Vol. 27(No. 12):17–25, Dec. 1989.

[109] G. Watson. "Technology 1991: The main event". *IEEE Spectrum*, Vol. 28(No. 1):30–30, Jan. 1991.

[110] H. Wielandt. *"Finite Permutation Groups"*. Academic Press, New York, 1964.

[111] J. E. Wirsching and T. Kishi. "Minimization of path lengths in single stage connection networks". In *1982 Int'l Conf. Parallel Processing, Computer Society Press, Silver Spring, Md.*, pages 563–571, 1982.

[112] D. Wright and M To. "Telecommunication applications of the 1990s and their requirements". *IEEE Commun. Mag.*, Vol. 28(No. 3):34–40, Mar. 1990.

[113] C. L. Wu and T. Y. Feng. "On a class of multistage interconnection networks". *IEEE Trans. Comput.*, Vol. C-29(No. 8):694–702, Aug. 1980.

[114] S. C. Yang and J. A. Silvester. "A reconfigurable atm switch fabric for fault tolerance and traffic balancing". submitted to jsac.

[115] S. C. Yang and J. A. Silvester. "Fault-tolerant multistage interconnection networks: Performance/reliability tradeoffs". *Comput. System Science and Engineering, Butterworths Scientific Ltd.*, Vol. 5(No. 4):233–242, Oct. 1990.

[116] S. C. Yang and J. A. Silvester. "A fault tolerant reconfigurable atm switch fabric". to appear INFOCOM'91, Florida, 1991.

[117] Y. S. Yeh, M. G. Hluchyj, and A. S. Acampora. "The knockout switch: A simple, modular architecture for high-performance packet switching". *IEEE J. Select. Areas Commun.*, Vol. SAC-5:1274–1283, Oct. 1987.

Appendix A

Calculation of R_0, R_1 and Γ_s

R_0 can be evaluated by subtracting the probability of both subnetworks being failure free from the sum of the probabilities of each being failure free.

$$R_0 = 2 \cdot p^S - p^{2 \cdot S}$$

Since single failures at the same stage have the same distribution, R_1 can be evaluated by the conditional probability that the single failure occurred at stage s.

$$\begin{aligned} R_1 &= 2 \sum_{s=0}^{n-1} 2^{n-1} p^{S-1}(1-p) p^{\Gamma_s}(1-p^{S-\Gamma_s}) \\ &- \sum_{s=0}^{n-1} 2^{n-1} p^{S-1}(1-p)(S-\Gamma_s) p^{S-1}(1-p) \end{aligned}$$

where Γ_s is the number of switches in the second subnetwork covered by the single failure of the tagged switch in stage s of the first subnetwork.

For a single failure at stage s, the set of paths corresponding to the cross product of 2^{s+1} processors and 2^{n-s} memories is rerouted. For a normal two copy MIN, the rerouted processor sets are merged stage by stage until stage s while the rerouted memory traffic sets are not forked. There are 2^{s-i} switches that carry rerouted traffic at stage i. After stage s, the rerouted processor sets are not merged any more, but the rerouted memory traffic sets start to be forked. There are $2^{n-s-1-i}$ switches that carry rerouted traffic at stage $n-1-i$. Let Γ_s^n be

the number of covered switches for a normal two copy MIN. Then,

$$\begin{aligned}\Gamma_s^n &= \sum_{i=0}^{s} 2^{s-i} + \sum_{i=0}^{n-s-2} 2^{n-s-1-i} \\ &= 2^{s+1} + 2^{n-s} - 3\end{aligned}$$

For a dual MIN, the rerouted processor sets are not merged until stage s while the rerouted memory traffic sets are forked at each stage. After stage s, the rerouted processor sets are now merged, but the rerouted memory traffic sets are no longer forked. Hence, the number of switches carrying rerouted traffic can be computed by considering the stages before and after s. Let Γ_s^d be the number of covered switches for a dual MIN. Then,

$$\Gamma_s^d = r_{before} + r_{after}$$

where

$$r_{before} = \begin{cases} \sum_{i=0}^{n-s-2} 2^{s+i+1} = 2^{n-1} - 2^{s+1} & for\ s \leq n-3 \\ 0 & otherwise \end{cases}$$

$$r_{after} = \begin{cases} \sum_{i=0}^{s-2} 2^{n-s+i} = 2^{n-1} - 2^{n-s} & for\ s \geq 2 \\ 0 & otherwise \end{cases}$$