Design and Analysis of a High-Performance Fault-Tolerant ATM Network

by

Ghaleb A. Al-Hashim

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER ENGINEERING

May, 1998

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

DESIGN AND ANALYSIS OF A HIGH-PERFORMANCE FAULT-TOLERANT ATM NETWORK

BY

GHALEB A. AL-HASHIM

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE
In
COMPUTER ENGINEERING

MAY 1998

UMI
300 North Zeeb Road
Ann Arbor, MI 48103

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN, SAUDI ARABIA

This thesis, written by

GHALEB A. AL-HASHIM

under the direction of his thesis committee, and approved by all the members, has been presented to and accepted by the Dean of the College of Graduate Studies, in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER ENGINEERING.

Thesis Committee

Dr. Mostafa I. Abd-El-Barr (Chairman)

Dr. Khalid M. Al-Tawil (Member)

Dr. Hasan Cam (Member)

Department Chairman

Dean, College of Graduate Studies

Date : 10/6/98

Dedicated to

my parents and my family.

Acknowledgments

I thank Allah, the Lord of worlds, for His mercy and limitless help and guidance. May peace and blessings be upon Mohammed the last of the messengers.

Acknowledgment is due to KFUPM for the support of this project.

I would like to express my deep appreciation to my Thesis advisor Dr. M. I. Abd-El-Barr for his invaluable advice and encouragement. I also wish to thank the other members of my Thesis committee Dr. K. M. Al-Tawil and Dr. H. Cam for their helpful suggestions and comments.

My thanks go also to my friends and colleagues, especially Mr. W. K. Al-Marhoon, Mr. T. M. Al-Nemer, and my cousin Mr. T. H. Al-Hashim, for their support and encouragement.

I am grateful to my family, especially my wife and children for their patience and understanding during my busy schedule.

Contents

List of Figures

List of Tables

Abstract

Name : Ghaleb A. Al-Hashim
Title : Design and Analysis of a High-Performance Fault-Tolerant ATM Network
Major Field : Computer Engineering
Date of Degree : May 1998

A high-performance fault tolerant Asynchronous Transfer Mode (ATM) switching fabric architecture called Binary-Tree Banyan Network (BTBN) is proposed. It is based on two interconnected parallel banyan networks in a binary tree form. Many ATM switching fabric architectures are proposed in the literature; however, few of them consider both the throughput performance measure and fault tolerance feature in the design. Compared to known and recent switching fabric architectures, BTBN has demonstrated very low cell loss probability and high throughput under normal operation and in the presence of faulty switching element conditions. In addition, any multistage interconnection networks (MIN) such as OMEGA, Delta, Baseline, and any topologically equivalent networks can be used in the BTBN switching fabric. BTBN requires less transmission latency than the simple baseline network, delivers all input cells in the right sequence using simple and distributed self-routing algorithm, and does not have jitter or Head-Of-Line (HOL) problems. In addition, BTBN is modular, scalable, and can be recursively expanded easily.

Master of Science Degree
King Fahd University of Petroleum & Minerals
Dhahran, Saudi Arabia
May 1998

خلاصة الرسالة

الاســـــــم: غالب عبدالرضى علي الهاشم

عنوان الرسالة: تصميم و تحليل شبكة (ATM) ذات الأداء العالي و المحتملة للخلل

التخـصـص: هندسة الحاسب الآلي

تاريخ التخرج: مايو ١٩٩٨

في هذه الرسالة، نقدم تصميم شبكة (ATM) الجديدة ذات الأداء العالي و المحتملة للخلل المسماة بـ (BTBN). يعتمد تصميم هذه الشبكة على ربط شبكتين متوازيتين من نوع (Banyan) على شكل شجرة ثنائية. لقد قدمت بحوث كثيرة في تصميم شبكات (ATM) ولكن قليل منها يأخذ بالاعتبار الظروف الغير طبيعية عندما يحدث خلل في مكونات الشبكة. أما شبكة (BTBN) فقد أثبتت كفاءتها العالية في الظروف العادية وكذلك في حالة وجود خلل في كثير من مكونات الشبكة و ذلك بالمقارنة بعدد من الشبكات المشهورة و المقترحة حديثا. اضافة الى ذلك، فانها قابلة للتصميم باعتماد اي نوع من أنواع الشبكات المترابطة و المرحلية المعروفة بـ (Multistage Interconnection Networks). مثال على ذلك (OMEGA) و (Delta) و (Baseline). والجدير بالذكر أن شبكة (BTBN) خالية من العيوب التي تشتكي منها كثير من الشبكات المقترحة حيث أنها شبكة ذات اعتمادية عالية و تحتمل لكثير من الخلل المحتمل وقوعه في كثير من مكوناتها و تضمن وصول جميع الخلايا المطلوب ارسالها في ترتيبها الصحيح من خلال نقلها داخل الشبكة مستخدمة برنامج التسيير المبسط و الموزع على جميع مكوناتها. اضافة الى ذلك، امكانية توسيع الشبكة بسهولة و بانتظامية في التصميم.

درجة الماجستير في العلوم

جامعة الملك فهد للبترول و المعادن

الظهران - المملكة العربية السعودية

مايو ١٩٩٨

Chapter 1

INTRODUCTION

Broadband-Integrated Services Digital Network (B-ISDN) is a cost-effective and service-independent network capable of transporting all different communication services and sharing all its available resources among all services. International Telecommunications Union - Telecommunication Standardization Sector (ITU-T) accepted ATM as the ultimate transfer mode solution for B-ISDN. ATM is a connection-oriented packet switching and multiplexing technique to transfer information over B-ISDN where established end-to-end paths are required prior to the beginning of information transfer. ATM has various features that extend the capabilities of current packet-switching networks toward incorporating the most desired features of circuit-switching networks to support real-time and variable bit-rate traffic most efficiently [38,40,64,65].

Several ATM switching fabric architectures have been proposed in the literature. There are two classes of ATM switching fabrics: time-division and space-division. Time-division architectures are further classified as shared-memory and shared-medium. Space-division architectures are classified based on their structures as crossbar, banyan-based and N^2 disjoint paths [12,52,53].

In time-division architectures, the traffic from all N input lines is multiplexed into a single resource of bandwidth equal to N times the bandwidth of a single line. This resource is shared by all input and output ports and can be a common memory or a shared medium such as a bus or a ring. *Coprin* is a shared-memory ATM switching fabric developed by the French CNET. ATM Output buffer Modular (ATOM) is a shared-medium ATM switching fabric developed by NEC. The switching fabric scalability of such architectures is restricted by the limitation on the bandwidth of the shared resource and on the number of input/output ports. Usually, the buffer management and control functions are centralized which increases the switch complexity [12,40,53].

In space-division architectures, multiple concurrent paths are established from the input to the output lines. Each path has the same data rate capacity as an individual line. In addition, the control of the switch can be either centralized or distributed throughout the switching fabric, thereby reducing its design complexity [12,40,52,53].

Although the design of the space-division switching fabric solves the scalability problem of the time-division architectures, some of the space-division switching fabrics have an inherent problem called internal blocking. This problem occurs when multiple disjoint paths can not be established simultaneously to route input cells to the proper destinations. The internal blocking problem limits the throughput of the switching fabric. Many switching fabric designs are proposed to minimize the effect of the internal blocking problem. For example, the crossbar switching fabric is considered as a nonblocking switching fabric [12,40,52,53].

Although, the crossbar switching fabric can successfully switch any destination permutation patterns at the input of the switching fabric to the proper output destinations, it can not switch more than one input cell destined to the same output port simultaneously. This problem is called output contention. This problem severely affects the performance of the switching fabric when the destination patterns traffic is modeled as a hot-spot input traffic where most of the input cells are destined to a specific output port that is highly on demand such as a file server. Another disadvantage of the crossbar switching fabric is the exponential increase in its size (the required number of cross-point switches) with the number of input lines. Hence, it is not recommended for large-size networks [12,29,53].

To solve the internal blocking and output contention problems in space-division switching fabrics, an N^2 disjoint paths switch architecture called the Knockout

switching fabric has been proposed. This is the most efficient, yet very expensive, NXN switching fabric where it is possible to establish N^2 disjoint paths from the input to the output lines simultaneously [82,84].

Banyan-based switch architectures are more practical than the crossbar switching fabric for large size networks. At the same time, they are not as expensive as the Knockout switching fabrics. They are based on MIN. The simplest banyan-based switching fabric is the Baseline Banyan Network. There is at most one path connecting any input to any output lines. This network has two problems: internal blocking and output contention [40].

A buffered-banyan network based on a banyan network with buffers in each switching element is effective for uniform traffic since it minimizes the blocking problem in the Baseline Banyan Network. It, however, introduces other problems for nonuniform traffic such as Head-Of-Line (HOL) blocking, large buffers requirement, and random delays within the switching fabric causing high jitter. HOL happens when one cell is waiting its turn for access to an output port and the other cells behind it are blocked and forced to wait despite the fact that their output ports are possibly idle [52,53].

To minimize HOL blocking, Double Banyan Network (DBN) is proposed. The network is based on cascading two buffered banyan networks: a distribution network

followed by a routing network. While the DBN minimizes HOL blocking, it increases the random delays within the switching fabric causing high jitter problem [52].

Another solution for HOL blocking is proposed for nonuniform traffic input. The switching fabric is called Pseudo Randomizer-Banyan Network. This network is based on a cell-scattering hardware, called pseudo randomizer; to distribute the nonuniform input traffic uniformly over the entire buffered banyan network by generating random patterns. It has been analyzed under nonuniform traffic and was found to have almost the same performance as that of a banyan network under uniform traffic [83].

Bypass Queues-Banyan Network is proposed to minimize the internal and HOL blocking problems by allowing other cells in the input buffers, called bypass queues, to be transmitted when the leading cell is blocked. It was shown that 90% throughput can be achieved for large-size switching fabric by using four banyan planes in parallel with bypass queues [80].

To minimize internal blocking without introducing the high jitter problem, Batcher-Banyan Network has been proposed [37]. The switch architecture consists of two consecutive networks: batch sorting and banyan networks. Cells are first fed to a batch sorter in which they are sorted according to their destination address, and then routed by a banyan self-routing network. Improved versions of Batcher-Banyan based switch architectures such as the *starlite* and *sunshine* switches are proposed to minimize the output contention problem [12,52,25].

Rerouting-Banyan Network was proposed to minimize cell contention efficiently. It achieves high throughput and low cell loss probability even with hot-spot traffic [62].

In general, a network is called i-fault tolerant if any set of i faults can be tolerated. A robust network can tolerate some instances of i faults. To improve the switching fabric reliability and fault tolerance, most of the banyan-based networks use one or more of the following strategies. The strategies are: switch size and internal links expansion, switching fabric duplication, additional switching elements in each stage, additional input/output ports, use of buffers in each switching element, and enhancement of the internal links speed relative to the input/output ports.

Extra-Stage Shuffle-Exchange Network uses extra stage to the basic Shuffle-Exchange Network to improve its fault tolerance. This is achieved by providing two paths for each input-output pair. The main problem with this network is that the two paths of each input-output pair are not disjoint [71].

MD-Omega Network is a single fault tolerant switch architecture that is based on a banyan network. It provides two disjoint paths throughout all network stages by using multiplexers at the input stage and demultiplexers replacing the last stage [76]. Extra Stage Cube Network is also proposed to provide two disjoint paths throughout all network stages [16]. The Benes Network minimizes the internal blocking problem and improves the fault tolerance of the switching element [7]. It consists of two baseline networks mirrored to each other sharing the middle stage. However, the

internal blocking problem and the fault tolerance are not improved throughout all stages of the switch architecture. The transmission latency is doubled and the routing complexity is increased.

Itoh Network increases the number of paths from any input to any output pair. It consists of a modified version of the baseline network with added subswitches between stages. The internal blocking problem is minimized and the switching element fault tolerance is improved at the expense of loosing the cells sequence and increasing transmission latency. In addition, the switch architecture is not modular and does not have a regular structure [3].

The parallel-banyan network provides two disjoint paths without increasing the transmission latency and loosing the cell sequence. It consists of two parallel baseline networks (planes) connected using input and output routers. Internal blocking and output contention problems are not minimized within each plane [52]. Tagle and Sharma Network minimizes the internal blocking and improves fault tolerance of each plane of the parallel-banyan network. This architecture allows routing from one plane to the other plane if there is cells contention or switching element faults [49].

Tandem-Banyan Network (TBN) has been proposed in order to minimize the internal blocking and output contention problems. It consists of multiple cascaded banyan networks. Unfortunately, the TBN achieves this at the expense of an increase in transmission latency and the out-of-order cell delivery [13,58].

Baseline-Tree Network minimizes internal blocking and output contention problems. In addition, it improves the fault tolerance of the switching elements. It is based on multiple interconnected banyan networks to provide multiple paths from any input to any output pair with minimum cell loss. The Baseline-Tree and Baseline Networks have the same transmission latency [28,27].

Pipeline-Banyan Network (PBN) is based on parallel banyan data planes controlled by a control plane. The control plane is for path reservation and the data planes are for cells routing. This switch architecture achieves a close to 100% maximum throughput, delay that is independent of the switch size, and in-sequence delivery of cells. Fault tolerance is not considered in the design of PBN [48].

Lin and Wang Banyan Network minimizes internal blocking and achieves fault tolerance by providing large number of paths between each input-output pair. This architecture provides two access points to the output ports to minimize the output contention problem. However, it does not preserve cells sequence in addition to introducing high jitter [26].

The Parallel-Tree Banyan Switching Fabric (PTBSF) minimizes internal blocking and output contention [77]. It is based on parallel banyan networks interconnected in a tree topology. This architecture consists of several levels. Cells sequence is not guaranteed when contended cells go through several levels in the tree.

Reliable And Zealous Network (RAZAN) minimizes internal blocking and output contention, and achieves fault tolerance by providing disjoint paths and large number

of redundant paths between each input-output pair. RAZAN and Baseline Networks have the same transmission latency. However, it is not scalable since the size of the switching element increase as the network size increases [71].

Most of the proposed switching fabrics attempt to minimize the internal blocking and output contention affects that are inherent problems in MIN. Those problems have a direct effect on the throughput performance. Other problems arise as a result of improving the throughput performance such as HOL problem, high jitter, out-of-order cell delivery, and complex buffer management. Some other solutions are proposed to minimize these arising problems by increasing the hardware complexity. Although this increase in hardware was not designed to cover switching element fault tolerance, some other switching fabrics attempt to improve throughput performance in the presence of faulty switching elements by increasing the hardware complexity. However, those fault tolerant switching fabrics are less concerned on the improvement of the throughput performance under normal conditions. Very few switching fabrics attempt to minimize internal blocking and output contention problems in order to achieve high throughput performance, and at the same time improve on the switching element fault tolerance. From the comprehensive literature survey, recommended criteria to design high performance and fault tolerant space-division MSI switching fabric architecture have evolved. The criteria are summarized in the following list:

1. Minimize cell loss during normal operation by
    - minimizing the effect of internal blocking

- minimizing output port cell contention by increasing access links to each output port of the switching fabric
- utilizing shared output buffer for each output port
- distributing the input load throughout the switching fabric

2. Maximize throughput performance for uniform and nonuniform input traffic during normal operation by
   - allowing multiple access to each input port of the switching fabric
   - increasing the internal speed of the switching fabric
3. Maximize throughput performance in the presence of faulty switching elements by
   - increasing the number of disjoint input-output paths
   - increasing the number of redundant input-output paths
4. Minimize the switching fabric transmission latency
   - by reducing the number of stages required for cell routing
5. Deliver cells in sequence, eliminate random delays within the switching fabric (jitter), and maintain the switching fabric synchronization by
   - making the input cells pass through the same number of switching elements in order to reach the correct output port destinations
   - keeping the self-routing switching algorithm simple
   - minimizing the use of input and internal buffering
   - avoiding cells missrouting and non progressive deflection

6. Make the switching fabric architecture modular, scalable, easily expandable, and consists of regular structures suitable for VLSI implementation

7. Increase the performance contribution of each switching element and each interconnection link in the switching fabric by

   - increasing the number of redundant paths
   - reducing the required number of interconnection links
   - reducing the required number of switching elements

During the switching fabric design of most proposed architectures, more than one criterion from the above list is considered in the design. However, few of the proposed architectures consider both the throughput performance and the fault tolerance aspect in the switching fabric design. In this thesis, a new high-performance fault tolerant switch architecture is proposed. The proposed switch architecture is carefully designed to meet most of the criteria listed above. A detailed literature survey on ATM switching fabric architectures is described in Chapter 2. Chapter 3 describes the design evolution and the architecture of the proposed switching fabric. Chapter 4 presents the results of the performance evaluation of the proposed switch architecture under fault-free conditions. The results of the fault tolerance evaluation are presented in Chapter 5. The reliability analysis of the proposed switch architecture is described in Chapter 6. Finally, the concluding remarks and a projection for the future work activities are provided in Chapter 7.

Chapter 2

LITERATURE SURVEY ON ATM SWITCH ARCHITECTURES

2.1 Motivation

At present, telecommunication networks are characterized by service specialization. Each network is dedicated to a specific application or a class of services such as telephony, telex transmission, TV distribution, and computer data transmission. Each network is designed to provide its basic services very efficiently [10,12,40,53,54].

This service specialization causes a large number of worldwide independent networks to evolve. Each network requires its own design, implementation,

installation and maintenance. A given network resources can not be shared with other networks providing different services. This is cost inefficient. In addition, there is a strong desire to share data among different public and local area networks. As a result, many telecommunication networks have been connected via network interfaces called gateways and bridges that are required for translating the different network protocols. Those translator devices slowed the communication among different networks. This led to an anxious desire to integrate all telecommunication services based on different traffic characteristics and requirements in a unified fashion in a single large-scale network type. As a result, ITU-T adopted the first set of ISDN recommendations in 1984. ISDN extends the concept of the telephone network by incorporating additional functions and features of circuit-switching and packet switching networks to provide existing and new services in an integrated manner [40,53,54,56].

2.1.1 Integrated Services Digital Network (ISDN)

ISDN was a digital end-to-end telecommunications network supporting a wide range of voice and non-voice applications in the same network. The network was characterized by its access requirements and service characteristics and provides digital access to digital transmission services, packet data services, and network-provided data services. Many companies had looked at ISDN as a potential solution to tie their widely separated organizations together, reduce their cycle times, expand their

markets, and improve their ability to interact with their customers and satisfy them [40,52,53,54,56].

ISDN was based on 64-kbps switching technology and was intended to support voice facilities, existing data services, and low-speed video. It integrated most of the required telecommunication services except the transmission of moving high-resolution images, digital TV, digital HDTV, video library, and high quality videophony. This is because these services require transmission channels capable of supporting transmission rates greater than the ISDN primary rate. Otherwise, the required transmission time would be unsatisfactory and very long when using the ISDN basic service rate. For instance, it would take over four hours to transmit a one-gigabit high-resolution graphics image using a 64-kbps-access line. The main source for the ISDN bandwidth limitation was the lack of reliable and high-bandwidth physical transmission medium. However, the progress made in fiber optic technology allows networks to operate at much higher rates, even over 155 Mbps, than the basic service rate described for ISDN. Consequently, a high-speed ISDN network called B-ISDN can provide all those services requiring high bandwidth. B-ISDN utilizes the progress in fiber optic technology, systems concepts, and speech coding and chip technology [10,40,47,53,54].

2.1.2 Broadband ISDN

B-ISDN is a service-independent network capable of transporting all different services and sharing all its available resources. There are many advantages of using B-ISDN such as the flexibility to adapt to changing or new needs, the efficiency in using available resources among all services, and the overall cost reduction of the design, manufacturing, operations and maintenance [42,73].

The term broadband is defined as a "service or system requiring transmission channels capable of supporting rates that are greater than the primary access rate [22]." The concepts of B-ISDN are summarized in [22] as follows: "B-ISDN supports switched, semi-permanent and permanent, point-to-point and point-to-multipoint connections and provides on demand, reserved and permanent services. Connections in B-ISDN support both circuit and packet mode services of a mono and multi-media type and of a connectionless or connection oriented nature and in a bi-directional and uni-directional configuration. A B-ISDN contains intelligent capabilities for the purpose of providing service characteristics, supporting powerful operation and maintenance tools, network control, and management."

B-ISDN is expected to support interactive and distributive services, bursty and continuous traffic, connection-oriented and connectionless services, and point-to-point and complex communications, all in the same network. The types of services B-ISDNs are envisaged to offer can be characterized by one or more of the following

attributes: high bandwidth, bandwidth on demand, varying quality of services parameters, guaranteed service levels, point-to-point, point-to-multipoint, and multipoint-to-multipoint connections, constant-bit-rate and variable-bit-rate services, and connection-oriented or connectionless services [40,53,54].

ITU-T classified possible broadband applications into four categories [53]:

1. Conversational services (Video/audio information transmission services)
2. Retrieval services (High-resolution image retrieval services)
3. Messaging services (Video mail services)
4. Distribution services:
   - Without user-individual presentation control (Document distribution services)
   - With user-individual presentation control (Full-channel broadcast videography)

Accordingly, B-ISDN should be capable of assigning useable capacity dynamically on demand. In addition, B-ISDN switching fabrics should be capable of switching all types of services. In recent years, large technological progress has taken place both in the field of electronics and in the field of optics. This progress allows the economical development of new telecommunication networks running at very high speeds. The evolution of highly reliable fiber systems into the access network provides the necessary high bandwidth required for B-ISDN. In fact, one type of optical fiber called monomode fiber has almost unlimited bandwidth transmission. As technology advances rapidly to meet the need for high-speed communications, the bottlenecks in communications networks are moving from transmission medium to the

communications processors. The throughput and end-to-end delay requirements of applications become limited by the processing power at network nodes, necessitating fast network protocols. The suitability of current network protocols for B-ISDN has not been fully addressed in the standardization committees. One issue that is resolved is the transfer mode: ATM principle is accepted by ITU-T as the ultimate transfer mode solution for B-ISDN. Different transfer modes exist in the telecommunication world each with different features. The following transfer modes are listed in increasing order of protocols complexity and bit rate fluctuation: circuit switching, multirate circuit switching, fast circuit switching, ATM, fast packet switching, frame relaying, frame switching, and packet switching [10,12,40,41,53,54].

Circuit switching transfer mode is used in telephone networks. To transport information from one node to another, a circuit is established by reserving all links from the source to the destination for a complete duration of the connection. The transmission is based on the STM (Synchronous Transfer Mode) technique. The principle of the STM technique is illustrated in Figure 2.1. The time axis is divided into n-slot frames with one slot dedicated to each channel. Each slot is one time-unit long and can carry a single data unit. A data unit associated with a given channel is identified by its position in the transmission frame. A connection always uses the same time slot in the frame during the complete transmission period. For this reason, all channels for different services must have the same bit rate [40].

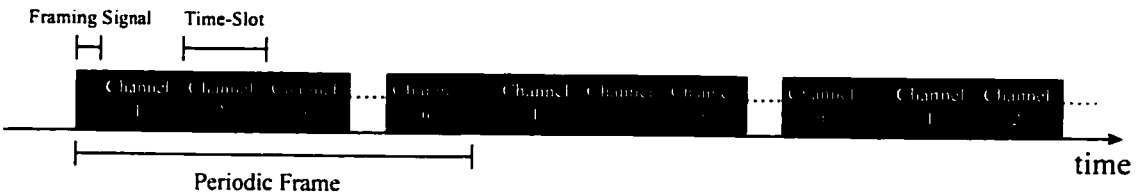


Figure 2.1 Synchronous Transfer Mode.

Multirate circuit switching transfer mode is an improved version of the circuit switching transfer mode. It uses the same periodic frame format with a fixed basic channel rate as shown in Figure 2.1. However, each connection can allocate multiple of the fixed basic channel rate. Each channel of one connection must remain synchronized. The required channel synchronization increases the complexity of the switching systems. In addition, the management and correlation of large number of basic rate channels (140,000 basic channels of 1 kbps for HDTV) becomes very complicated. By increasing the bandwidth of the selected basic rate, the channel management complexity is reduced for high bit rate applications. However, the cost is the enormous waste of redundant bandwidth for the low speed applications such as the voice transmission. An enhanced version of this transfer mode was proposed to use multiple basic rates. This enhancement did not overcome a major drawback of this transfer mode; namely its inability to cope efficiently with fluctuating and bursty character sources. Since the resources in the network, with channel bit rate equal to or greater than the peak bit rate, are reserved even when there is an idle period for the sending terminal [40].

Fast circuit switching transfer mode extends the concepts of circuit switching to sources with fluctuating and bursty nature. The resources in the network are reserved only during active transmission periods, and released when the source is idle. Services with different bit rates can be accomplished when the fast and multirate circuit switching transfer modes are combined. The resulting multirate fast circuit switching

transfer mode allows the use of different information rates efficiently. However, the complexity of designing and controlling such a system is high since it is required that the system must be able to set up and tear down connections in very short periods [40].

In ***packet switching transfer mode***, user information is encapsulated in packets. These packets contain additional information used inside the network for routing, error correction and flow control. These packets have variable length and require a complex buffer management inside the network. This transfer mode has complex protocols performing error checking and flow control on every link of the connection because of the low-quality physical transmission medium available. The protocol complexity increases the processing requirements and switching delay inside the network since link-by-link error and flow control is required to guarantee an acceptable end-to-end performance on each link of the network [40].

Frame switching transfer mode is similar to packet switching transfer mode except it has less functionality such as multiplexing of logical channels. The functional reduction in the protocols increases the network throughput four times the speed of packet switching mode [40].

Frame relaying transfer mode is similar to frame switching transfer mode except that the protocols do not have error and flow control. Only CRC checking is performed to discard erroneous frames. The functional reduction in the protocols has increased the network throughput dramatically from 8 Mbps to 140 Mbps. However,

error corrections are performed between the source and destination nodes. This requires high-quality physical transmission medium [40].

Fast packet switching transfer mode is similar to packet switching mode with minimal functionality in the network. The main aim of this transfer mode is to transmit information efficiently at a very high-speed rate [40].

Asynchronous transfer mode is the same as the fast packet switching with fixed-size packets, referred to as ATM cells. The packet-oriented nature of ATM is well suited to applications with bursty traffic characteristics. The short fixed-size cells guarantee a jitter (the variance of the delay) compatible with the constraints imposed by voice or moving image transmission. These cells offer full bandwidth flexibility at high transmission rates. They also provide the basic framework for guaranteeing the quality of service requirements of applications with a wide range of performance metrics, while allowing statistical multiplexing where several variable-bit-rate connections can share a link with a capacity less than the sum of their peak bit rate requirements. Figure 2.2 illustrates the principle of the ATM technique. The ATM approach does not require a framed transmission system. The label contained in the cell's header identifies the connection. So that, a data unit (ATM cell) associated with a specific channel may occur at any position and several connections can be multiplexed on any link. ATM is an attempt to utilize the properties of both the packet-switch and circuit-switch networks in an integrated network as shown in Figure 2.3.

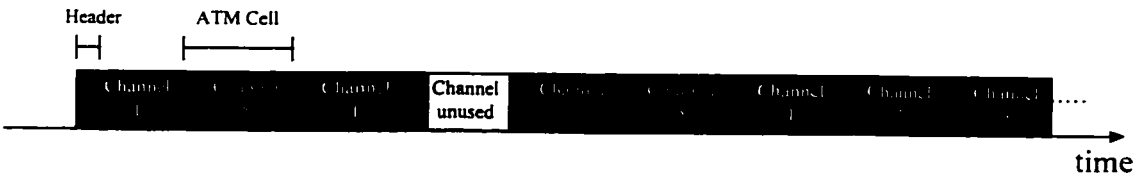


Figure 2.2 Asynchronous Transfer Mode.

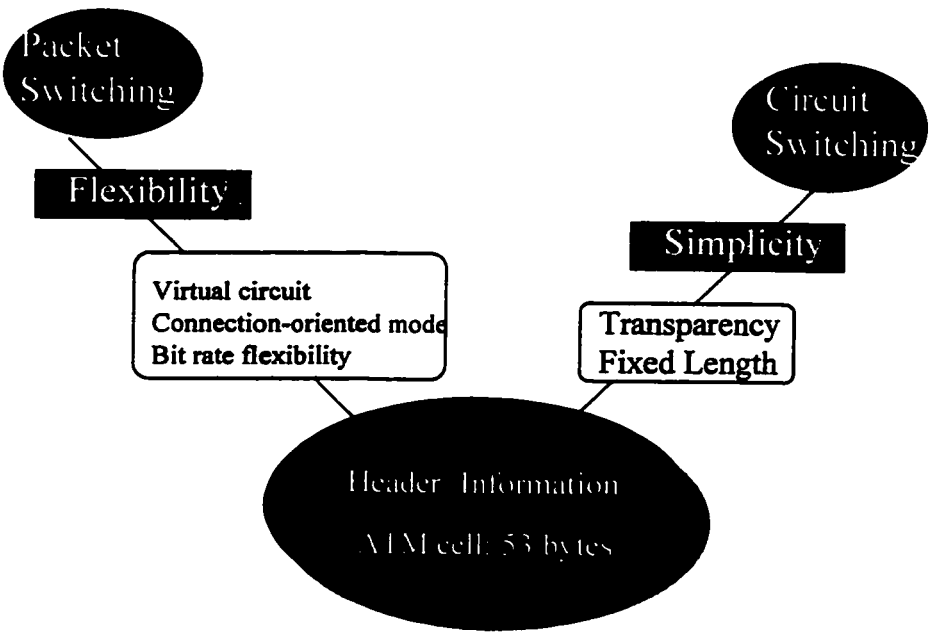


Figure 2.3 ATM combines the flexibility of packet switching with the simplicity of circuit switching.

ATM combines the simplicity of circuit switching with the flexibility of packet switching where it does not check the fixed-size cell contents and it is a connection-oriented technique using virtual channels and high bit rate flexibility. In fact, it is a compromise that allows the integration of different services with different characteristics and requirements in the same network using a unique interface. ATM is a connection-oriented, packet switching and multiplexing technique to transfer information over a B-ISDN network where established end-to-end paths are required prior to the beginning of information transfer [38,40,53,54,65,73].

In general, the transfer mode of B-ISDN is envisaged to:

- support and integrate all existing networks as well as other future applications with unknown characteristics
- minimize switching complexity
- minimize the processing load per cell at intermediate switching nodes to support very high transmission speeds
- minimize the buffer requirements at the intermediate nodes to bound the delay and buffer management complexity
- provide the basis for guaranteeing quality of service requirements for each service in the network

ATM is an attempt to meet all the above objectives. It has various features that extend the capabilities of current packet-switching networks toward incorporating the

most desired features of circuit switching to support real-time traffic most efficiently [38,40,53,54].

B-ISDN standards are being developed by a number of bodies around the world and are being finalized by ITU-T. The initial ITU-T recommendation on B-ISDN was published in 1988 [22]. There are thirteen ITU-T recommendations outlining the fundamental principles and initial specifications for B-ISDN approved in 1990. These recommendations include the B-ISDN Protocol Reference Model (PRM) as shown in Figure 2.4. The PRM consists of three planes: control, user and management. These planes use three layers: physical, ATM and ATM adaptation [38,40,53,54].

The physical layer is the underlying transport of the network. It consists of two sublayers: transmission convergence and physical medium dependent. The services offered by the transmission sublayer include cell rate decoupling and delineation (i.e. determining cell boundaries from received bit stream), transmission frame adaptation, generation and recovery, and header sequence generation and verification. The physical medium dependent sublayer is responsible for the correct transmission and reception of bits on the appropriate physical medium. This sublayer must guarantee a proper bit timing and maximum bit error rate. The transfer mode defines how the information supplied by higher layers is to be mapped onto the physical layer. The ATM layer mainly provides the switching and multiplexing of traffic. There is no awareness of the specifics of applications at this layer.

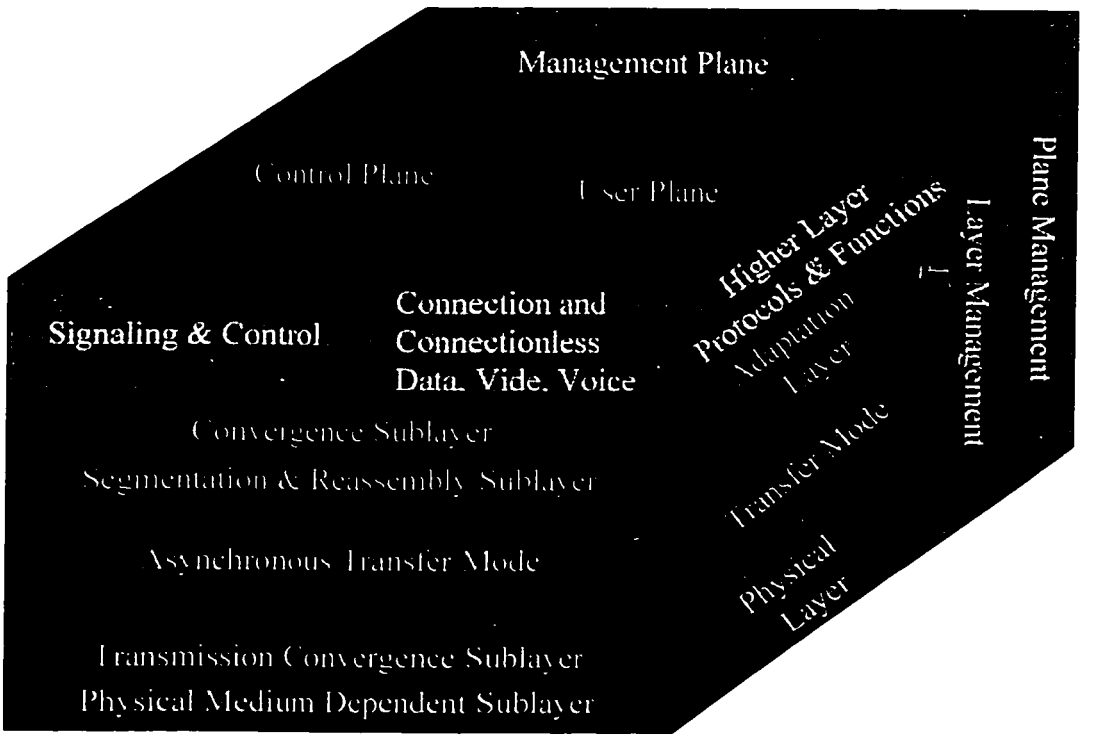


Figure 2.4 B-ISDN ATM Protocol Reference Model.

This simplicity is necessary to keep up with the high-speed network links. Application-specific services are provided at end stations by the adaptation layer (AAL) [53].

The adaptation layer supports the higher layer functions of the user and control planes. Examples of the adaptation functions include continuous bit stream-oriented services adaptation functions, connectionless services, and packet mode services.

The control plane is used for connection management, including the connection setup and release functions, addressing, and routing. These functions play a particularly important role for connections that are established dynamically on demand in the network. Once a connection is established, the user data are transmitted using one of the protocols in the user plane. The user plane transmits end-to-end user information between two or more communicating entities. Both planes use lower layers to transmit their messages and data.

The management plane provides for operations and management functions, and also provides the mechanisms to exchange information between the user and the control planes. This plane is further divided into two layers: layer and plane management. Layer management deals with layer-specific management functions that include the detection of failures and protocol abnormalities. Plane management provides management and coordination functions related to the complete system.

The ATM framework is a connection-oriented packet-switching technology that segments application data frames into 48-byte-long cell payloads and adds the

associated 5-byte-long header as shown in Figure 2.5. Then it transfers these cells through the ATM network, and assembles the cell payloads at their destination to reconstitute the original user data frames [38,40,53,54].

The ATM cell header at the network-node interface (NNI) illustrated in Figure 2.6a consists of five fields: virtual path identifier (VPI), virtual channel identifier (VCI), payload type identifier (PTI), cell loss priority (CLP), and header error check (HEC). At the user network interface point (UNI) between an ATM end station and the network, the cell header also includes a generic flow control (GFC) field. This 4-bit field is part of the VPI inside the network as shown in Figure 2.6b.

In ATM, end-to-end virtual channels are established between end stations before the traffic can start flowing. Routing of cells in the network is performed at every switch for each arriving cell. The routing information of a cell is included in the two routing fields of the header: VPI and VCI.

The two levels of routing hierarchies, virtual paths (VP) and virtual channels (VC), are defined in [22] as follows:

"VC: A concept used to describe unidirectional transport of ATM cells associated by a common unique identifier value, referred to as the VCI.

VP: A concept used to describe the unidirectional transport of cells belonging to VCs that are associated by a common identifier value, referred to as the VPI."

The PTI specifies whether the contents of a payload carry user or management data.

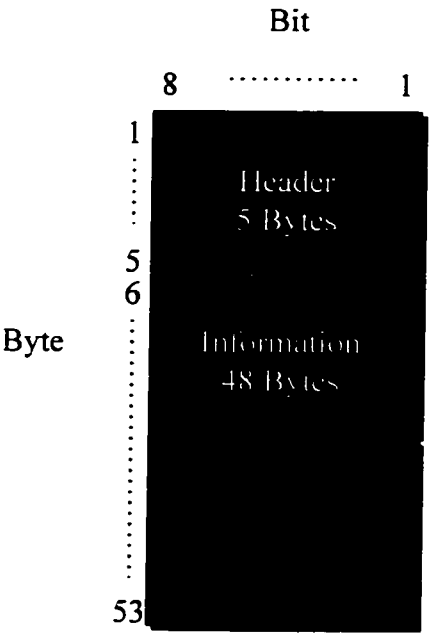


Figure 2.5 ATM Cell

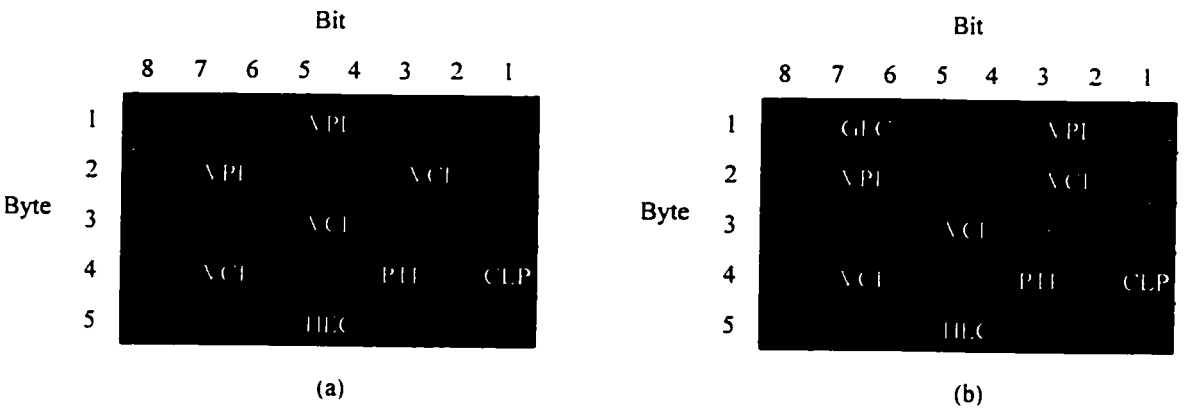


Figure 2.6 ATM Cell header structure: (a) NNI; and (b) UNI.

The management data may be used inside ATM networks, whereas the ATM layer is not concerned with the contents of ATM cells that carry user information. The CLP at the ATM cell header is a 1-bit field used for cell loss priority. Due to the statistical multiplexing of connections, it is unavoidable that cell losses will occur in ATM networks. Cells with CLP bit set (low priority) may be discarded earlier at congested switches than cells with CLP bit not set (high priority). The HEC field is used mainly for two purposes: for discarding cells with corrupted headers and for cell delineation. The 8-bit field, when used for HEC, provides single-bit error correction and a low probability corrupted cell delivery capabilities. The HEC value is equal to the reminder of the division of the product X^8 and the polynomial of order 31. The coefficients of the polynomial are given by the bit values of the 4 bytes of the cell header. The GFC mechanism provides orderly and fair access of terminals to the shared medium by supervising the cell streams and assigning capacity to contending terminals on per-cell basis [38,40,53,54].

The ATM layer transfers cells between peer ATM layer entities. It provides in sequence delivery of cells among ATM layer users by utilizing services provided by the physical layer. At the originating end station, it receives any 48-byte cell payload from an ATM layer user, adds 4 bytes of the corresponding cell header excluding the HEC byte, and passes the cell to the physical layer for HEC calculation and transmission. At the destination end stations, the ATM layer receives cells from the

physical layer, removes the cell header, and passes the cell payloads to their corresponding ATM layer users. Inside an ATM transport network, there is no ATM layer user for the user traffic, and cells are passed from the receiving ATM layer entities to the transmitting counterparts at each switching node along their paths between the source and destination end stations. Accordingly, the ATM layer mainly provides the switching function of ATM networks. Cells are discarded at intermediate nodes when they have errored headers (i.e. bit errors), or the transmission link buffer is full. The ATM layer provides its services unreliably since there is no retransmission of errored and lost cells inside the network and it is up to the end stations to ensure the integrity of the data carried in ATM cell payloads. The ATM layer deals only with the functions of the cell header, regardless of the payload contents. This simplicity is necessary to keep up with the required high-speed transmission links [38,40,53,54].

The following list summarizes the key benefits of Broadband ATM-based networks:

1. ATM provides a single network for all traffic types: voice, data and video. It improves the efficiency and manageability of integrated networks.
2. ATM enables the creation and expansion of new applications due to its high-speed transmission and the integration of different traffic types.
3. ATM is compatible with currently deployed physical networks since it is not based on a specific type of physical transport.
4. Efforts within the standards organizations and the ATM Forum continue to ensure that embedded networks will be able to gain the benefits of ATM incrementally

(upgrading portions of the network based on new application requirements and business needs).

5. ATM is evolving into a standard technology for local, campus backbone and public and private wide area services. This uniformity is intended to simplify network management by using the same technology for all levels of the network.
6. The information systems and telecommunications industries are focusing and standardizing on ATM. ATM has been designed from the beginning to be scaleable and flexible in: geographic distance, number of users, and access and trunk bandwidths. This flexibility and scalability ensures that ATM will be around for a long time.

The thesis proposal is to investigate the ATM switch architectures in the space-division class and attempt to improve on the switch architecture performance and fault tolerance.

2.2 ATM Switch Architecture

ATM is a packet oriented transfer mode based on asynchronous time division multiplexing and the use of fixed-length cells. Each cell consists of an information field and a header. The header is primarily used to identify cells belonging to the same virtual channel within the asynchronous time division multiplexing, and to perform the

appropriate routing. Cell sequence integrity is preserved per virtual channel. The information field is carried transparently through the network. No processing such as error control is performed inside the network. All communication services can be transported using ATM including connectionless services. To accommodate various services, several types of ATM adaptation layers have been defined, depending on the nature of the service, to fit information into ATM cells and to provide service specific functions such as cell loss recovery [38,40,52,53,54].

ATM is a connection-oriented technique. This technique requires end-to-end connections to be established before starting the traffic transfer flow. The header values are assigned to each section of a connection for the complete duration of the transmission, and translated when switched from one section to another. The header contains the identification of the virtual connection: VCI and VPI fields. The VCI field identifies dynamically allocatable connections, and the VPI field identifies statically allocatable connections. The physical transmission path that connects ATM network elements may consist of several virtual paths as shown in Figure 2.7. Each virtual path may consist of several virtual channels [38,40,53,54].

Several virtual channels may be concatenated to form a virtual channel connection. Similarly, concatenating several virtual paths forms a virtual path connection. An ATM switching node transports cells from the incoming links to the outgoing links based on the information stored in its routing table using the routing label at the cell header as shown in Figure 2.8.

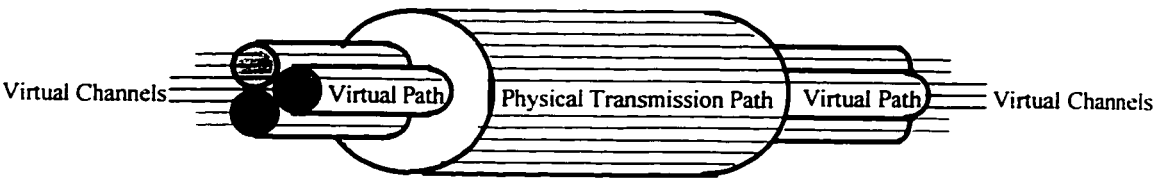


Figure 2.7 Relationship between virtual channel, virtual path and transmission path.

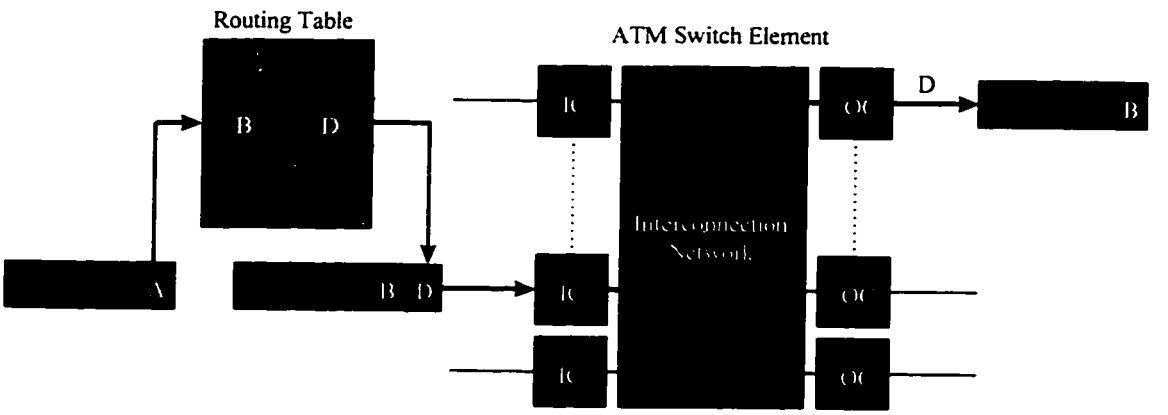


Figure 2.8 ATM Cell switching.

The routing label at the cell header is read and a table lookup is performed to determine the outgoing link and the new routing label used at that link. Then, the routing label is updated to its new value and the cell is switched from the incoming link to the outgoing link. An N x N ATM switching fabric can be viewed as a black box with N inputs and N outputs that transports cells from any incoming link to any outgoing link. As shown in Figure 2.8, the incoming links are connected to the ATM switching fabric through input ports. An intelligent input controller (IC) manages each input port and provides buffering, cells duplication, cell processing, VCI translation, traffic multiplexing, and path connection requests and reservations through the ATM switching fabric. After the cell header is processed to determine its outgoing link, it is passed to the ATM switching fabric to be delivered to its outgoing link. The interface between the switching fabric and the outgoing link is referred to as the output port. An intelligent output controller (OC) manages each output port and provides buffering, VCI translation, demultiplexing, and cells concentration. Output conflict occurs when more than one cell attempts to access a single access output port simultaneously. When this happens, only one of the contending cells can be distend to the output port and the other cells may either be stored in a buffer until they can be distend or discarded [38,40,53,54,77].

The ATM switching fabric has three queuing disciplines: input, central, and output queuing. The input queuing is located between the incoming link and the input port. This queuing strategy is used to solve a possible contention problem at the input ports.

Each input buffer stores the incoming cells until the switching arbitration logic determines the ATM cells of the input buffer to be transferred to the outlet without an internal contention. This discipline with FIFO (First-In/First-Out) queuing suffers from HOL (Head- Of-Line) blocking. This happens when one cell is waiting its turn for access to an output port and the other cells behind it are blocked and forced to wait despite the fact that their output ports are possibly idle where no cells waiting in the other queues. The central queuing is located inside the ATM switching fabric. Its queuing buffers are shared among all input and out ports. The output queuing is located between the ATM switching fabric and the output port. This strategy is used to minimize the output conflict problem [12,40,52,53,54].

2.3 ATM Switching Fabric Architectures

Several ATM switching fabrics with different architectures have been proposed in the literature and some have been developed. There are two classes of ATM switching fabrics: time-division and space-division. Time-division architectures are further classified as shared-memory and shared-medium. Space-division architectures can be classified according to different criteria: single-stage versus multistage, single-path versus multipath, blocking versus nonblocking, input/central/output queuing principles, and switch structure principles. In this thesis, a thorough survey on most of

the proposed space-division ATM switch architectures has been conducted. The various space-division ATM switch architectures are described according to the structured-based classification that consists of three types: crossbar, banyan-based and N^2 disjoint paths [12,52,53].

2.3.1 Time-Division Architectures

In time-division, the traffic from all N input lines is multiplexed into a single resource of bandwidth equal to N times the bandwidth of a single line. This resource is shared by all input and output ports and can be a common memory or a shared medium such as a bus or a ring. Multiplexing is required at the input side and demultiplexing at the output side of the switching fabric. Time-division switching fabric's scalability is restricted by the limitation on the bandwidth of the shared resource and on the number of input/output ports. Usually, the buffer management and control functions are centralized which increases the switch complexity [12,40,53].

2.3.1.1 Shared-memory Type

The switch consists of a single dual-port memory shared by all input and output lines of the switch. Figure 2.9 shows the *Coprin* shared-memory based ATM switching fabric developed by the French CNET.

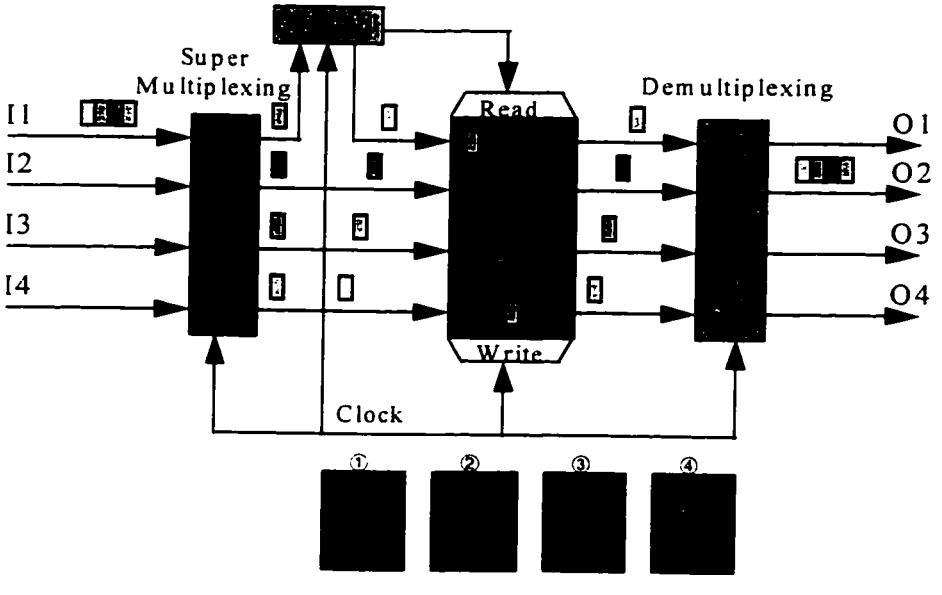


Figure 2.9 *Coprin*: A 4 X 4 shared-memory ATM switch fabric developed by the French CNET.

This switching element operated with links at 280 Mbps with 15 bytes information and 1 byte for the header. The switch consists of four basic blocks: super multiplexing, buffer memory, demultiplexing and control blocks [12,40].

The super multiplexing block is responsible for multiplexing the serial data coming in each input line into parallel streams feeding the memory queues based on instructions given by the control block when it receives the header of the cell. The buffer memory block consists of several queues based on the number of input lines plus one queue to store the header of each cell. Its function is to store all ATM cells. The demultiplexing block performs the reverse operation of the super multiplexing block, reconstructs the ATM cells, and routes the cells to the proper output ports. Figure 2.9 shows an example of routing one cell of size four bytes from input line 1 to output port 2. The four blocks at the bottom of Figure 2.9 shows the space switch states of the super multiplexing block in each time slot from 1 to 4. The demultiplexing block uses the same switch states in reverse order from 4 to 1. In this example, the focus is only on the first input line and the second output port. The cell consists of one leading byte called the cell header followed by three data bytes. At time slot # 1, the super multiplexing switch (SMS) passes the cell header to the header queue of the memory buffer through the controller. At time slot # 2, SMS passes the first data byte to the first queue of the memory buffer. At time slot # 3, SMS passes the second data byte to the second queue of the memory buffer. At time slot # 4, SMS passes the third data byte to the third queue of the memory buffer. Then the controller

decides on the time slot, depending on the load of the output port destination and the sate of the space switch of the demultiplexer, for the demultiplexer to read the cell bytes on the same order they are written in the memory buffer. In this example, at time slot # 6, the demultiplexer (DS) passes the cell header to the output port # 2 using the space switch sate (SSS-2). At time slot # 7, DS passes the first byte to the output port # 2 using SSS-1. At time slot # 8, DS passes the second byte to the output port # 2 using SSS-4. At time slot # 9, DS passes the third byte to the output port # 2 using SSS-3.

From this example, one notices that the alignment of the space switch in the super multiplexer and incoming cells at the input lines are crucial in order to perform the correct operation [12]. In other words, the cell header of input line # 4 has to come on the same time slot for the first byte in input line # 1. The cell header of input line # 3 has to come on the same time for the second byte in input line # 1. The cell header of input line # 2 has to come on the same time for the third byte in input line # 1. The remaining three bytes of each input line have to follow their cell header.

2.3.1.2 Shared-medium Type

The switch consists of a single high-speed medium or bus that is common to all input and output lines of the switch. All cells coming in the input lines are synchronously multiplexed onto this common bus.

Figure 2.10 shows the basic structure of a shared-bus switch type. It consists of five main blocks: serial to parallel multiplexers, high-speed bus, address filters, first-in-first-out queues and parallel to serial demultiplexers [12].

The serial-to-parallel blocks are responsible to multiplex incoming cells onto the shared bus. The address filter is responsible to pass the cells to the right output ports. The first-in-first-out queues are responsible to store the cells for each output port in the same order. The parallel-to-serial blocks are responsible to demultiplex and reconstruct the ATM cells.

Contrary to the shared-memory switch type, the shared-medium switch type has separate FIFO queue memory for each output port. The queue memory access speed does not have to be as fast as that of the shared-memory switch type. However, the size of each queue has to accept all cells in the shared-bus that have the same destination. Thus the utilization of the memory is not as efficient as that of the shared-memory switch type. For example, when a critical destination node receives data form many input lines, the buffer might not be enough to store all arrived cells even though the other output queues are not fully utilized.

In addition, the bus speed has to be very fast to accommodate the cells on all input lines simultaneously. The minimum bus speed is equal to the aggregate speed of all input lines. The speed requirement of the shared-bus increases linearly with the size of the switching fabric input lines.

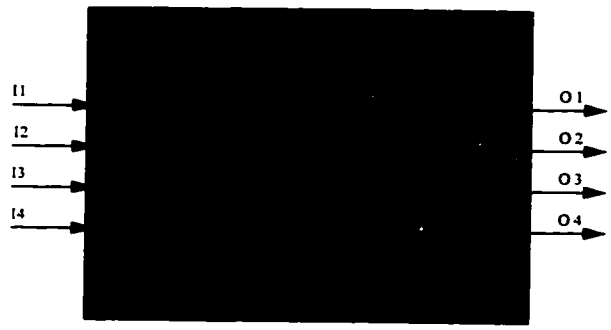


Figure 2.10 A 4 X 4 shared-bus ATM switch fabric.

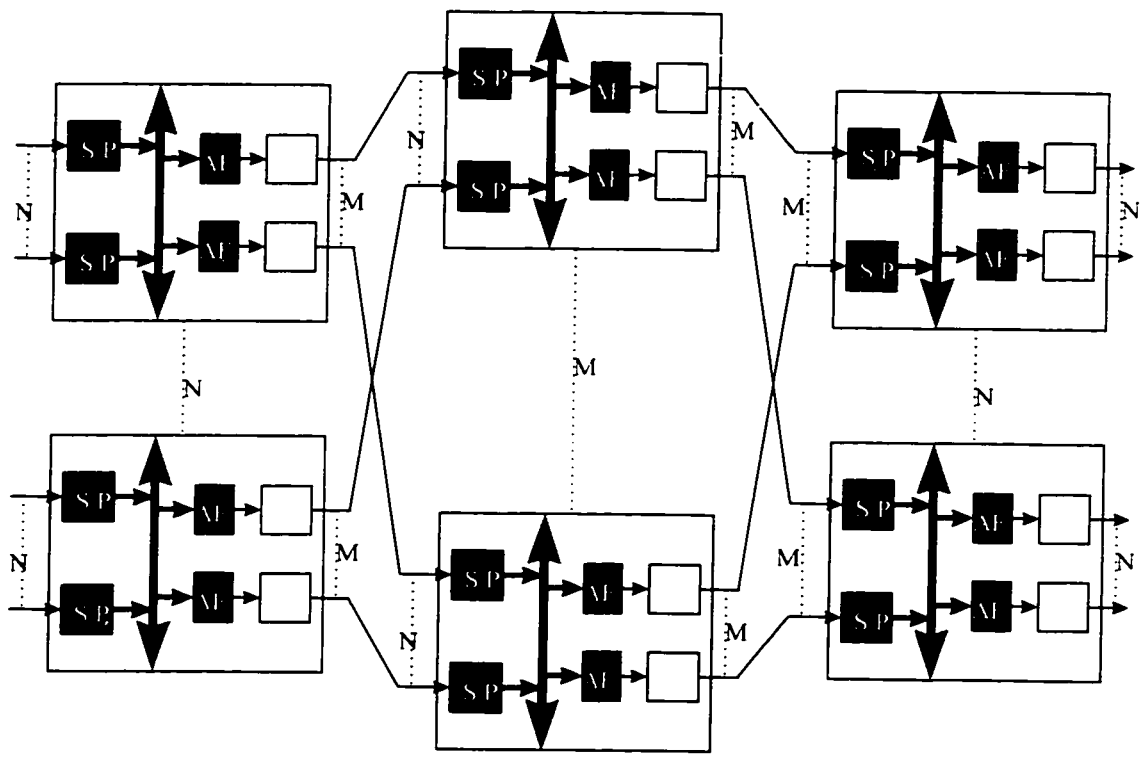


Figure 2.11 ATOM switching fabric architecture.

This problem is similar to the case of the memory access time in the shared-memory switching fabrics. The shared-memory access time increases linearly with the size of the switching fabric input ports [12].

One shared-medium architecture ATM switching fabric was developed by NEC called ATM output buffer modular (ATOM) [12,53]. The ATOM chip, switching fabric element, consists of a bus and output buffers. The input links are limited by the bus speed. The architecture achieves scalability by interconnecting a number of ATOM chips together through serial links as shown in Figure 2.11.

2.3.2 Space-Division Architectures

In the space-division class, multiple concurrent paths are established from the input to the output lines. Each path has the same data rate capacity as an individual line. In addition, the control of the switch can be either centralized or distributed throughout the switching fabric, thereby reducing its design complexity. As shown in the following sections, some types of this class have an inherent problem in the switching fabric. The problem is that it might not be possible to have multiple disjoint paths simultaneously to route the cells to the proper destinations. This problem is called internal blocking and limits the throughput of the switching fabric [12,40,52,53].

In structured-based classification, ATM space-division switch architectures have three categories: crossbar, banyan-based and N^2 disjoint paths. The three types follow a common abstract reference model as shown in Figure 2.12. The reference model consists of N data planes where N corresponds to the number of inputs and outputs (N X N switching fabric). Each input line is connected to a demultiplexer that routes the input cells to the corresponding input buffer of each data plane. Each data plane consists of N buffers. The output of all buffers of each data plane is connected to a multiplex that concentrates the output cells into the corresponding output line. Thus the abstract reference model consists of N routers, N^2 buffers, and N concentrators. The various proposed ATM switching fabrics differ in the way the routers and concentrators are implemented, and the locations of the buffers.

A thorough comparison of input versus output queuing on an N X N nonblocking space-division switches using Markov chain models, queuing theory, and simulation is conducted in [39]. In this experiment, the intuition that better performance results with output queuing than with input queuing is quantified.

The blocking problem in banyan-based networks was thoroughly studied in [74]. The banyan-based networks realize only a subset of all possible input-output permutation in a nonblocking fashion. General nonblocking permutation patterns are presented for the Inverse Omega network and its topological equivalents.

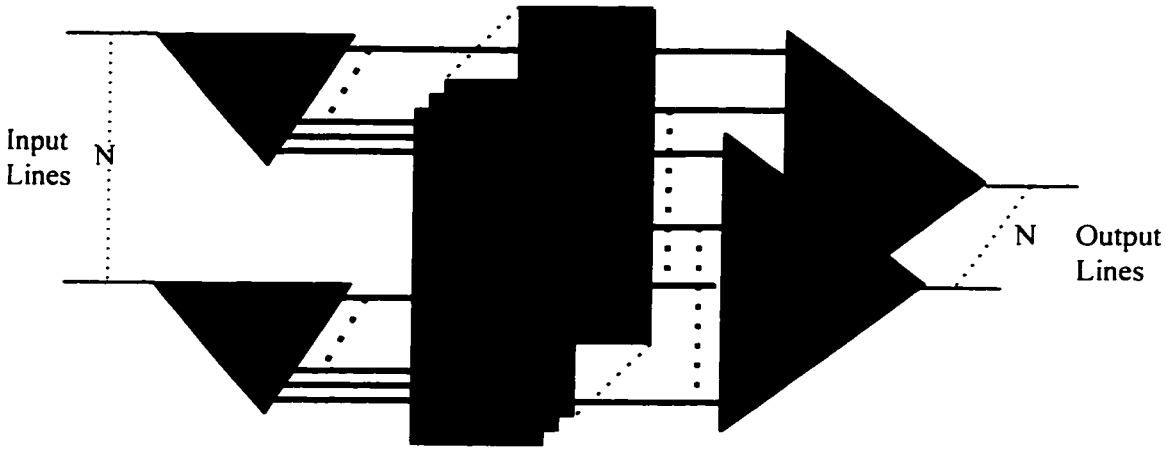


Figure 2.12 Abstract reference model for space-division ATM switching fabrics class.

These patterns are useful for routing in high speed blocking networks by breaking general connection requests into maximal nonblocking subsets.

2.3.2.1 Crossbar type

The crossbar switching fabric consists of a square array of N^2 cross-point switches where N is the number of input lines as shown in Figure 2.13. Any input can be connected to any output through the cross-point switches. The cross-point switch can have two states: bar and cross. Figure 2.13 shows only one cross-point switch on the Bar State to connect input line 3 to output line 2. It is shown in the second row and the second column. The remaining switches are on the Cross State. In this switching fabric, it is possible to establish maximum of N disjoint paths simultaneously from the input to the output lines. For this reason it is considered as a nonblocking switching fabric. In other words, it is possible to route messages simultaneously within the switching fabric from all input to output lines except for messages routed to the same destination. Output contention occurs when cells are destined to the same output port. Only one cell will be routed to the required output line and the remaining cells are discarded. Although this is a nonblocking switching fabric, it is not recommended for large size networks since the number of cross-point switches increases exponentially with the number of input lines [12].

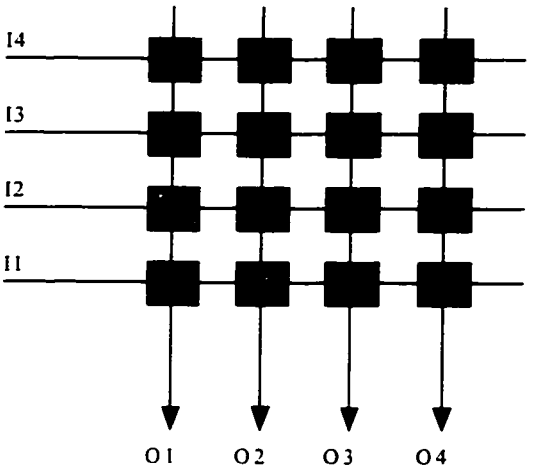


Figure 2.13 A 4 X 4 crossbar ATM switch fabric.

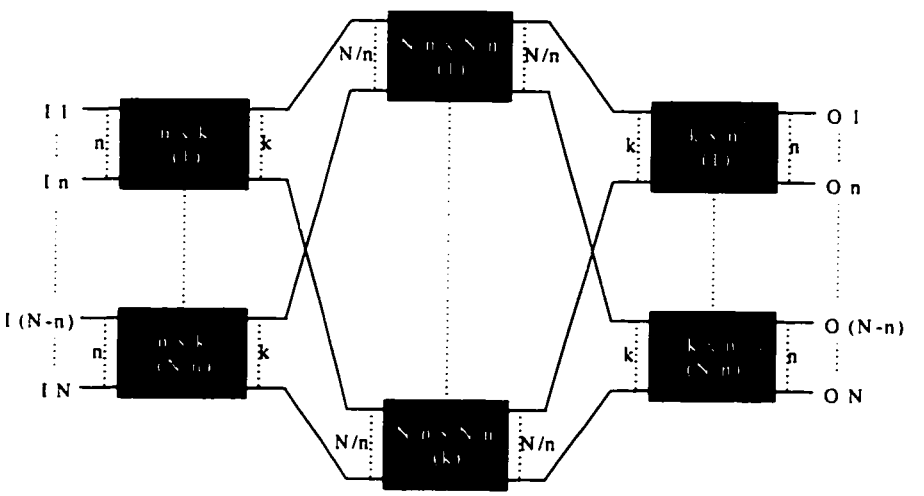


Figure 2.14 Nonblocking three-stage space-division switch fabric.

Other drawbacks of the crossbar switching fabrics are the absence of any alternate paths in the case of cross-point failure and the inefficient utilization of each cross-point. Multiple-stage switching solves all problems with crossbar switching fabric except the output contention. Figure 2.14 shows a nonblocking three-stage space-division switching fabric. The first stage consists of N/n (n x k)-crossbar matrices where $n = \log_2 N$. To achieve a strictly nonblocking switching fabric, k must be equal to or greater than 2n-1. The second stage consists of k (N/n x N/n)-crossbar matrices. The third stage consists of N/n (k x n)-crossbar matrices. The total number of cross-points in this switching fabric is equal to $4N(\sqrt{2N}-1)$. It should be noted that the required cross-points to make the three-stage space-division switching fabric nonblocking is less than that of the crossbar switch only when N is 24 or greater. As the switch size increases over 24, the saving on the number of required cross-points increases considerably compared to the crossbar switch. However, there is a cost for reducing the number of cross-points, the increase in routing complexity. The switch is no longer self-routing and requires address translation tables to establish end-to-end connections for each cell [29,53].

2.3.2.2 N^2 Disjoint Paths type

This is the most efficient, yet very expensive, switching fabric where it is possible to establish N^2 disjoint paths from the input to the output lines simultaneously where N

is the number of input lines. Figure 2.15 shows a 4 X 4 *Knockout* switch fabric. It consists of four input lines feeding four broadcast buses that operate at an equal speed, and four bus interfaces that can access all input lines, one for each output line. This is different from the shared-bus switching fabric since it has parallel input broadcast busses and does not require parallel to serial multiplexers. This is a nonblocking switching fabric since it can route messages from any input to any output lines simultaneously. In addition, it has the capability to route multiple messages simultaneously to the same output port [84].

Each bus interface consists of one concentrator, one cells' buffer, and four cell filters, one for each input line. The function of the cell filter is to route the incoming cells to the proper output line. The concentrator function is to reduce the number of cells coming to the same output port simultaneously to a specific number depending on the application requirements. The function of the cells' buffer is to store the final cells that passed the concentrator for each output port.

In this example, the concentrator selects only two good cells out of four possible cells coming from the input lines. The selection of the cells are done using 2 X 2 contention switches (CS) in which two input lines contend for a winner output and a loser. If there is only one cell in one of the inputs, then CS selects the cell as a winner. However, if each input line has a cell, CS selects the first input cell as a winner. Each concentrator consists of five CS's and one delay block.

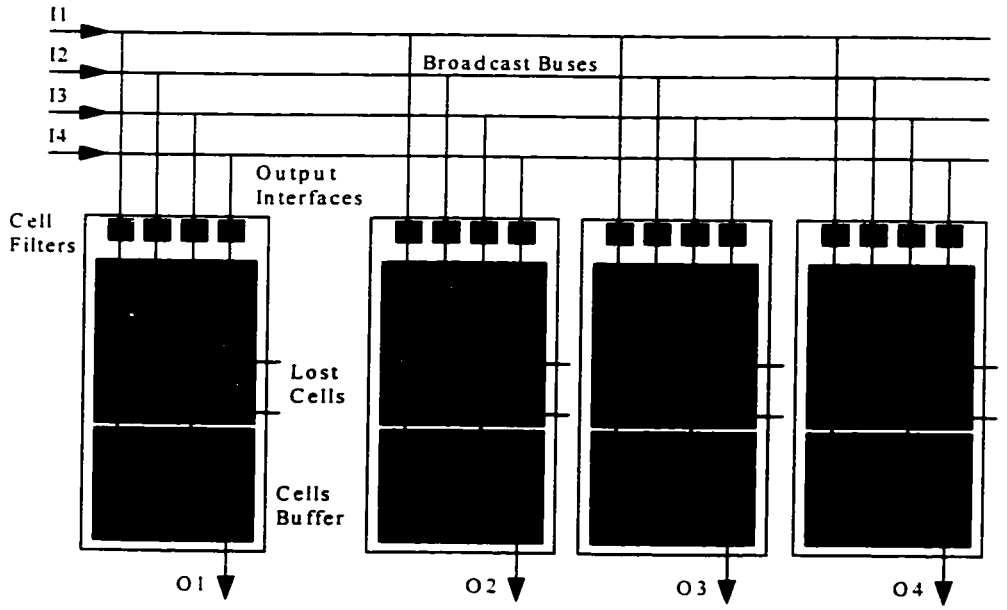


Figure 2.15 A 4 X 4 *Knockout* ATM switch fabric.

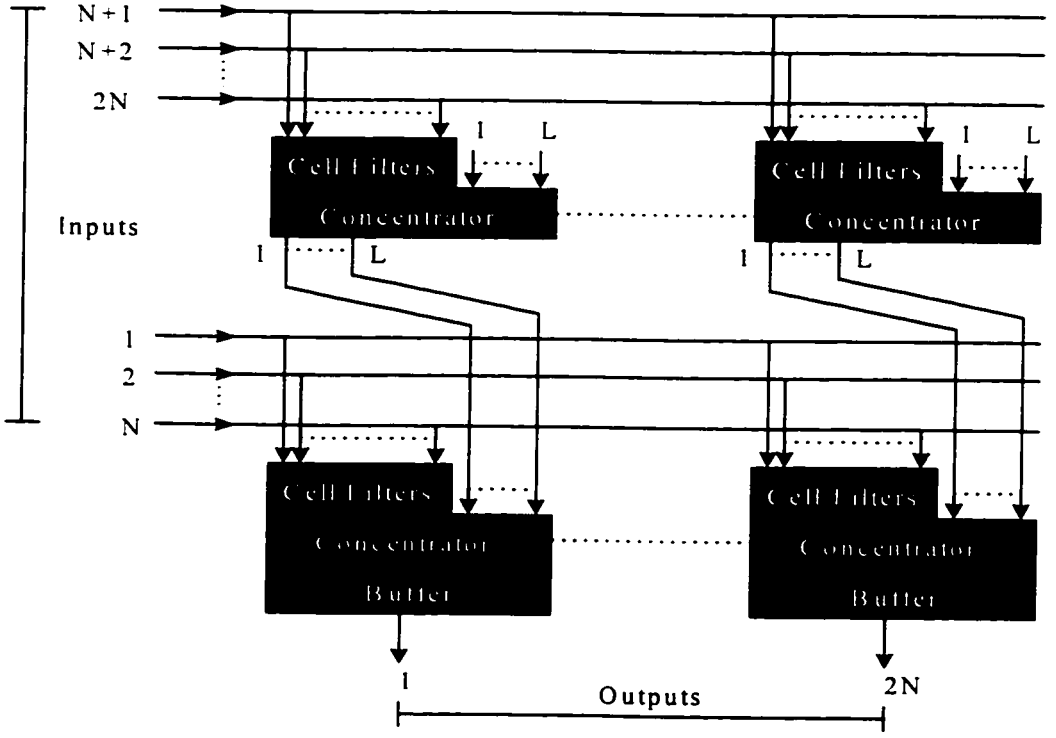


Figure 2.16 Modular growth of 2N X 2N *Knockout* switch.

The CS's are connected in three stages. The first stage includes CS-1 and CS-2. The first two input cells go to CS-1 and the last two input cells go to CS-2 simultaneously. The second stage includes CS-3 and CS-4. The winner of CS-1 competes with the winner of CS-2 through CS-3. The loser of CS-1 competes with the loser of CS-2 through CS-4. The third stage includes a block delay and CS-5. The winner of CS-3 become the first selected cell and goes to a delay block waiting for the second winner cell to leave the concentrator. In the other hand, the loser of CS-3 competes with the winner of CS-4 through CS-5. The winner of CS-5 and the first winner cell leave the concentrator to the shifter block while the other two cells (losers) are discarded.

Fault tolerant design can be achieved by providing only a single spare interface module attached to the N broadcast buses since all interface modules are identical. This spare module could take over the operation of any one of the N interface modules if a failure occurs. Service disruption in the input or output attached to the failed module only.

The *Knockout* switch can grow modularly from N X N to JN X JN where J = 2,3, ... Figure 2.16 shows one way to expand the *Knockout* switch to 2N X 2N. Notice that each concentrator has additional L inputs for a total of N+L inputs and L outputs. Two separate N-bus interfaces daisy chained together. Each interface module connected to the (N+1,2N) busses has N cell filters and one N+L to L concentrator.

Additional shared buffer structure with shifter and L FIFO buffers is included in each of the interface modules connected to the (1,N) busses.

This is a high performance ATM switching fabric and the cost of realizing such a switching fabric increases exponentially with the size of the input lines. It fits the high availability applications requirements where loosing cells within a switching fabric is not permitted. Although it is high performance switching fabric, it has a potential problem that could jeopardize its reliability due to faults in the input broadcast busses.

A *Knockout* Switch-based multistage interconnection networks (KSMIN's) is proposed in [83] to reduce the severe buffer requirements for large *Knockout* Switches through a phased address filtering. The number of stages of N X N KSMIN's based on m X m *Knockout* switches is $\lceil \log_m N \rceil$. Figure 2.17 shows a 16 X 16 KSMIN based on the 4 X 4 *Knockout* switch shown in Figure 2.15. In Figure 2.17, N = 16, m = 4 and the number of stages is equal to $\lceil \log_4 16 \rceil = 2$.

2.3.2.3 Banyan-Based type

This type of switches is practical for large network size and at the same time they are not as expensive as those of the N^2 disjoint paths type are. It is based on MIN. Using MIN would reduce the ultimate size of the switching fabric as already shown in the nonblocking three-stage switching fabric and in the KSMI's. MIN is constructed from the cross-point binary switching elements used in the crossbar switch in Figure 2.13.

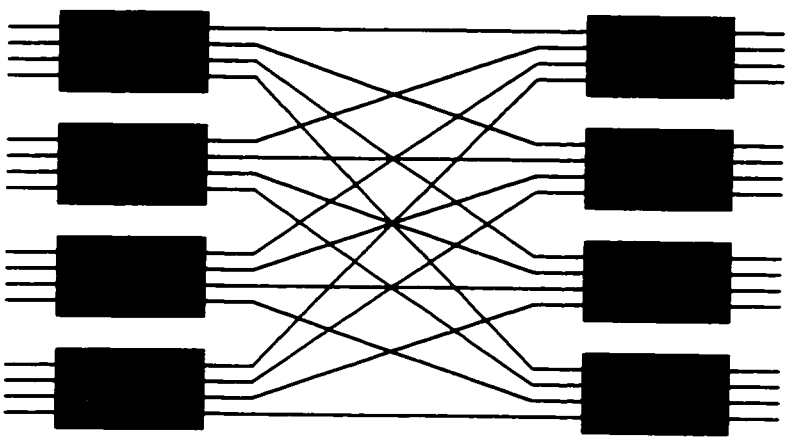


Figure 2.17 A 16 X 16 *Knockout* switch-based MIN.

The cross-point switches are interconnected in a multistage binary tree router as shown in Figure 2.18a. This interconnection allows the first input line to be connected to any output line using the two sates of the cross-point switches: bar and cross. The same binary tree can be shared to route the second input line to any output line. This introduces the internal blocking since the binary tree is shared among both input lines and would not be able to route two cells requiring one cross-point to be in both states simultaneously. Similarly, output conflict occurs when two cells are destined to the same out put line. This is because it requires one cross-point binary switch to be in both states simultaneously. Figure 2.18 shows the construction of a complete 8 X 8 multistage network. This network consists of three stages ($\log_2 8$). Each stage consists of four 2 X 2 switching elements (N/2). Each 2 X 2 switching element consists of four cross-point switches. Total of twelve 2 X 2 switching elements (48 cross-point switches) are required to build an 8 X 8 multistage switching network compared to a total of 64 cross-point switches used in the crossbar switching fabric. However, the existence of internal blocking and output conflicts introduces performance limitations and causes such architecture to be referred to as blocking.

Figure 2.19 shows an 8 X 8 shuffle-exchange network known as the OMEGA network. In this network, the interconnection of switching stages is identical.

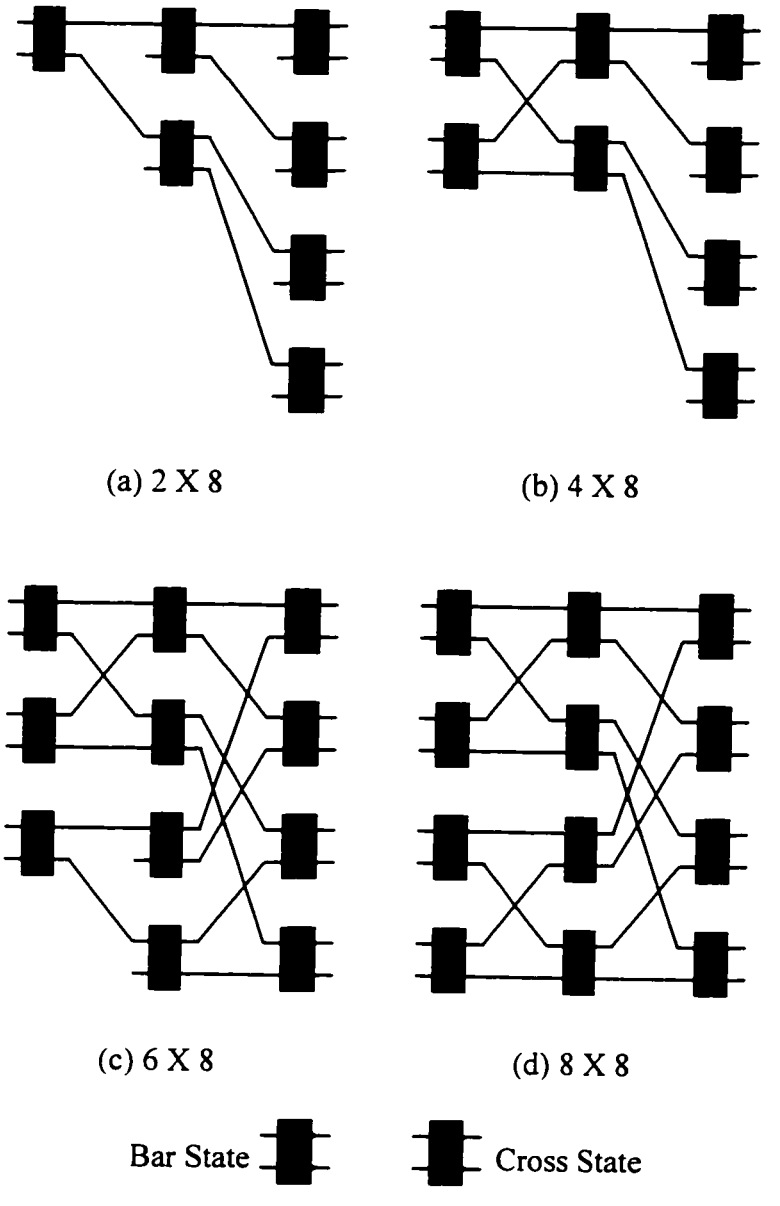


Figure 2.18 Constructing an 8 X 8 multistage network using binary switches.

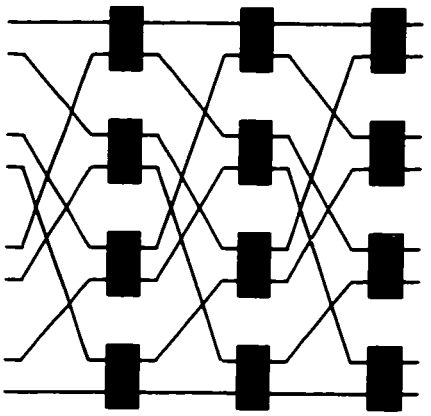


Figure 2.19 An 8 X 8 OMEGA multistage network using binary switches.

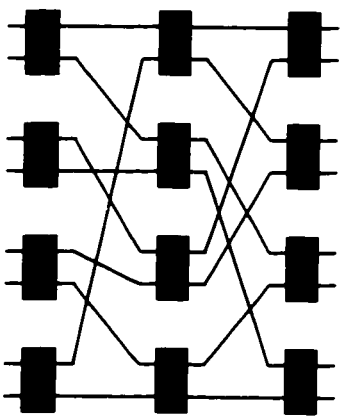


Figure 2.20 An 8 X 8 delta multistage network.

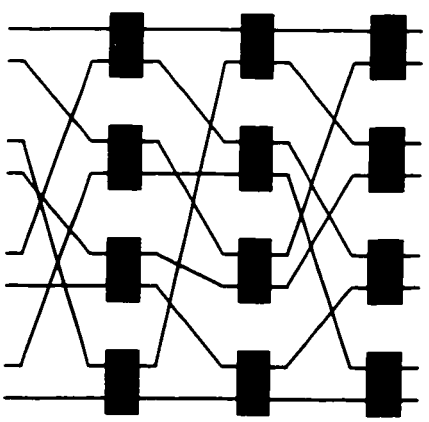


Figure 2.21 A modified 8 X 8 delta multistage network.

Figure 2.20 shows an 8 X 8 delta network and Figure 2.21 shows a modified version of the delta network. Banyan network refers to any N X N multistage interconnection network regardless of the particular form they take as shown in the previous figures.

In general, Banyan networks have the following properties:

1. provides multiple concurrent paths from the input lines to the output lines
2. reduces the switching complexity to $O(N \log_2 N)$
3. accommodates distributed self-routing as well as centralized switching control
4. usually does not require a complex cell routing algorithm
5. consists of scaleable modular building blocks allowing the construction of large switching networks easily
6. possesses regular structures suitable for VLSI implementation

Contrary to the above advantages, Banyan networks might have inherently one or more of the following problems: internal blocking, output conflict, potential cells sequence loss, and high jitter. Banyan-based network becomes a fault-tolerant network when it contains one or more permanent faulty components and continues providing its service in at least some cases. In general, a network called i-fault tolerant if any set of i faults can be tolerated. A robust but not i-fault tolerant network can tolerate some instances of i faults.

Most of the banyan-based networks use one or more of the following strategies to improve the switching fabric reliability and fault tolerance.

The strategies are:

- switch size and internal links expansion
- switching fabric duplication
- additional switching elements in each stage
- additional subswitching elements between stages
- additional stages in the network
- additional input/output ports
- use of buffers in each switching element
- enhancement of the internal links speed relative to the input/output ports.

Banyan-based networks can be classified according to the following construction methods:

1. ***Buffering***

   Adding buffers at each internal link. The buffers can be placed at the input or output ports, or inside the switch.

2. ***Sorting***

   Placing a sorting network in front of the banyan network to guarantee that all incoming cells to the banyan network are passable by such networks without blocking.

3. ***Expansion***

Expanding the number of internal links to minimize both internal and output blocking of a banyan network while preserving the self-routing and path-uniqueness properties.

4. ***Dilation***

Replacing each internal link by *d* links to minimize internal blocking.

5. ***Stage Extension***

Extending number of switches per stage to create disjoint paths through the network.

6. ***Stage Bridging***

Adding extra subswitches between stages to bypass faulty stage switching elements.

7. ***Network Augmentation***

Adding extra stages to minimize internal blocking problems and to improve network fault tolerance.

8. ***Nonuniform Traffic Regulator***

Adding a device that converts nonuniform traffic to a uniformly distributed input source in order to minimize HOL blocking and output contention problems.

9. ***Deflection Routing***

Missrouting contending cells through wrong links to be involved in the routing process again.

10. ***Load-Sharing***

Organizing the switching elements of each stage of a banyan network into a number of groups in such a way that the proper routing function is not distributed by sharing the input traffic of any two switching elements in the same group. A cell can be routed to any switching element in the same group, and still be correctly routed to its destination.

11. Multiple Banyans

Placing banyan networks in parallel or in cascade.

A lot of research has been done in the banyan-based type of switches. In this survey, examples are presented to show the evolution of this switching fabric type. All of them are banyan-based multistage interconnection networks. However, the switch architecture differs from one fabric to the other with an ultimate objective to increase the network fault tolerance, minimize the above problems, and provide high performance switching fabric.

2.3.2.3.1 Non Fault-Tolerant Banyan-Based Networks

Baseline Banyan Network

This is the simplest banyan-based network. There is at most one path connecting any input to any output lines. Figure 2.22 shows an 8 X 8 baseline network that consists of three stages. Each stage has four 2 X 2 switching elements (SE).

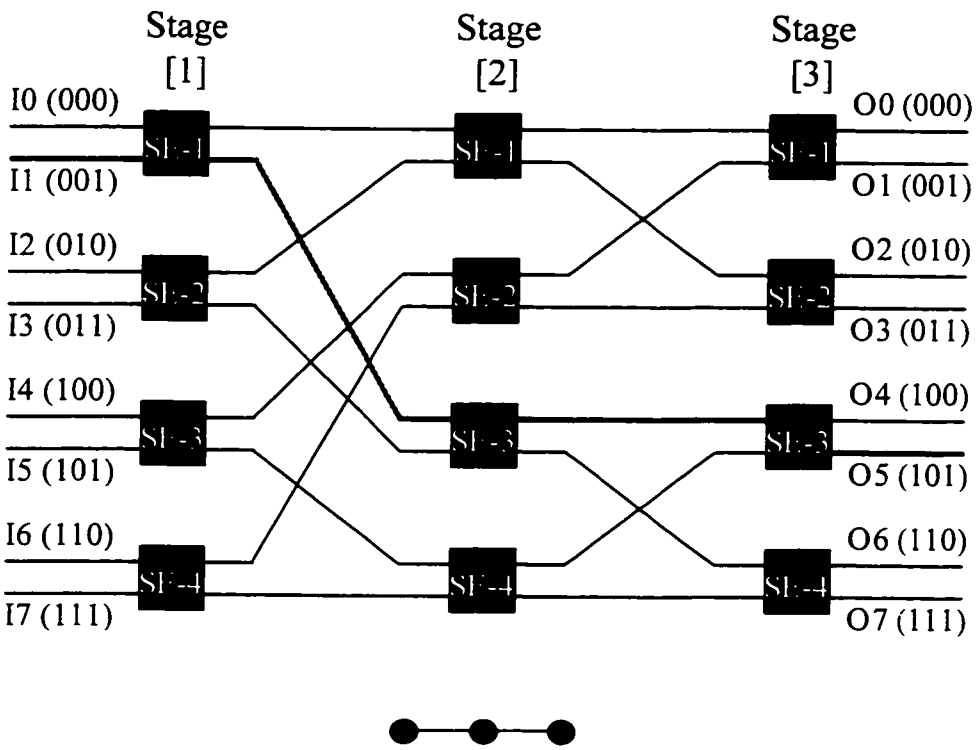


Figure 2.22 An 8 X 8 Baseline banyan network.

To route a cell in a SE, the SE checks the bit in the destination address that corresponds to the stage number. If the bit is equal to 0, the SE routes the cell using the top output port. Otherwise, it uses the lower output port [40].

The following steps are to route the ATM cell from input port 1 to output port 5 as shown in Figure 2.22 (thick line):

- the cell is in SE #1 of stage #1. SE #1 checks the third bit (MSB) of the destination address (output port 5). Since the bit is equal to 1, SE #1 routs the cell using its bottom output port. (The cell is routed to SE #3 of stage #2)
- SE #3 checks the second bit of the destination address. Since the bit is equal to 0, SE #3 routes the cell using its top output port to SE #3 of stage #3.
- SE #3 checks the first bit (LSB) of the destination address. Since the bit is equal to 1, SE #3 routs the cell using its bottom output port. (The cell is routed to output port 5)

The baseline network is fast and efficient since it uses self-routing technique and it has to go through only $\log_2 N$ stages to reach an output line. However, it is a blocking network since it is possible that some cells require the same switching element (link) to go through. Then one of the cells passes to the right destination and the other cell is either lost or deflected to the wrong destination. In addition, if one switching element is faulty, all cells routed through it are lost since there is no another alternative path to go through. This means that all switching elements have to function correctly in order to guarantee the proper function of the whole network as

illustrated by the channel graph shown in the bottom of Figure 2.22. The channel graph illustrates the possible routes of a cell from any input to any output. The circles in the channel graph represent the switching elements of each stage required to route the cell from the input source to the output port.

Buffered-Banyan Network

This architecture is based on a banyan network with buffers in each switching element as shown in Figure 2.23. This is a direct solution to the internal blocking problem [52,53]. When there is a conflict, the blocked cells remain in their buffer instead of being discarded. There are three modes to advance cells from one stage to the next: queue loss (QL), local backpressure (LB), and global backpressure (GB). In the QL operation mode, arrived cells get discarded when the buffers are full. In the other hand, with LB operation mode a cell can not be advanced to the next stage unless the next buffer along the path is currently not full. This condition is relaxed in the GB operation mode where a cell can advance to the next stage even if the next buffer along the path is currently full as long as it is not going to be full upon the arrival of the cell under consideration.

The cost of resolving internal blocking using buffered banyan networks is the introduction of new problems such as HOL blocking, large buffers requirement, and random delays within the switching fabric causing high jitter.

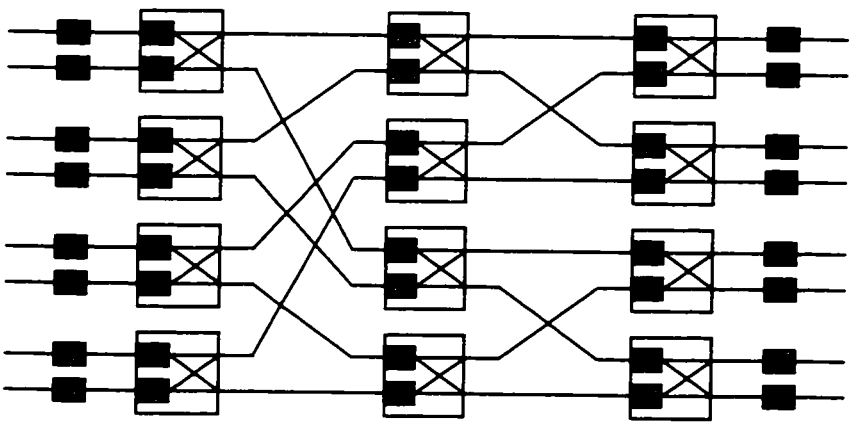


Figure 2.23 An 8 X 8 buffered Baseline banyan network.

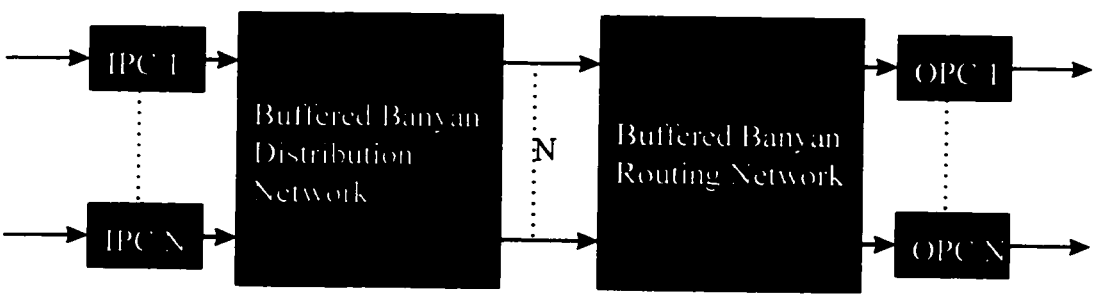


Figure 2.24 An N X N double banyan network.

Similar to the baseline network, all switching elements have to function correctly in order to guarantee the proper function of the whole network as illustrated by the channel graph shown in Figure 2.22.

Input queuing is the cause of HOL blocking problem for nonuniform input traffic. Several solutions to minimize the HOL blocking problem are proposed: Batcher sorter network, bypass queues, permutation network, randomized routing, pseudo-randomizer, and double banyan.

Double Banyan Network

This architecture is based on cascading two buffered banyan networks: a distribution network followed by a routing network, as shown in Figure 2.24. The switching elements of the distribution network ignore the destination addresses of incoming cells and route them alternately to each of their outlets. If one or both outlets are busy, the first available outlet is used.

The basic function of the first network is to regulate the bursty source input traffic. This architecture minimizes the HOL blocking problem and increases the random delays within the switching fabric causing high jitter problem. In addition, it does not guarantee the in-sequence cell delivery since cells belonging to the same connection may follow different paths and suffer different queuing delay [45]. Similar to the

baseline network, all switching elements in the first and second networks have to function correctly in order to guarantee the proper function of the whole network as illustrated by the channel graph of each network shown in Figure 2.22.

Pseudo Randomizer-Banyan Network

Buffered banyan networks are effective packet switches for uniform traffic. However, ATM switching fabric is expected to have nonuniform traffic most of the time. Therefore, a buffered banyan network is highly vulnerable since it has only a single path per network input-output pair. A packet-scattering hardware, called pseudo randomizer (PR), to distribute the nonuniform input traffic uniformly over the entire buffered banyan network by generating random patterns is proposed and analyzed in [82]. Figure 2.25 shows the proposed network block diagram.

The PR-Banyan network is a nonblocking and distributes the input stream from each of N inputs over N possible paths in the buffered banyan network. It has been analyzed under nonuniform traffic and proved to have almost the same performance as a banyan network under uniform traffic. Similar to the baseline network, all switching elements in the routing network and the pseudo randomizer hardware have to function correctly in order to guarantee the proper function of the whole network as illustrated by the channel graph shown in Figure 2.22.

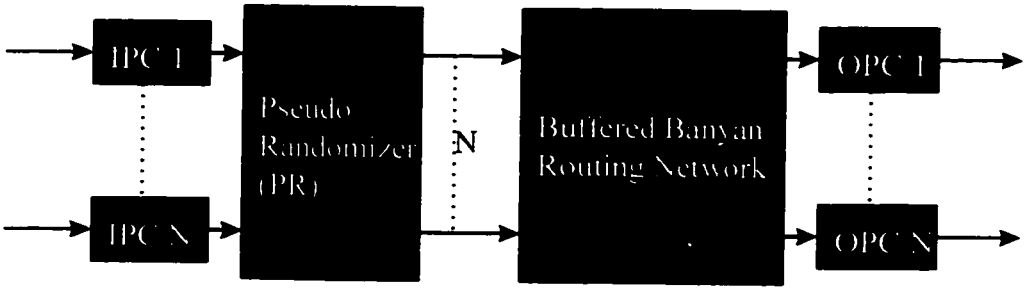


Figure 2.25 An N X N PR-banyan network with a pseudo randomizer.

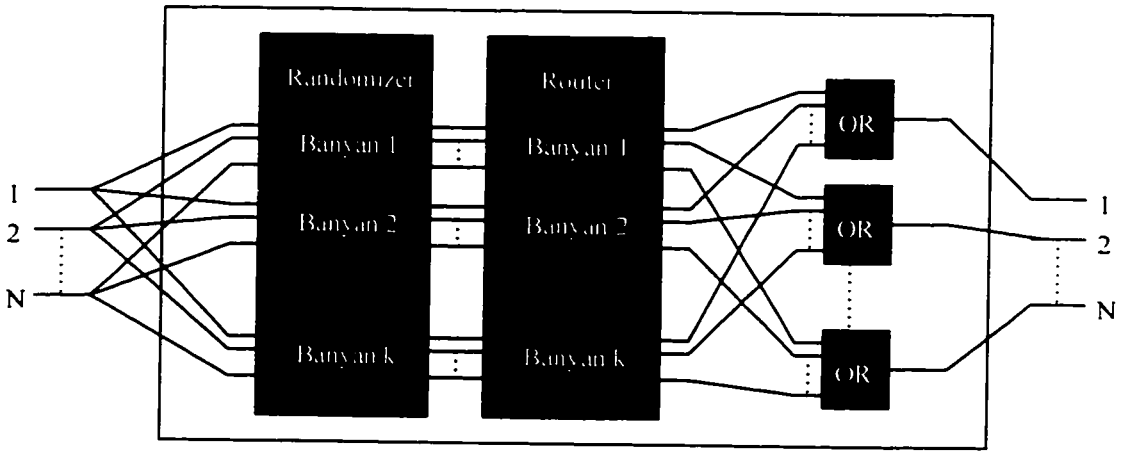


Figure 2.26 An N X N permutation-banyan network.

Randomized Routing-Dilated Banyan Network

This architecture is based on cascading two dilated banyan networks: a randomization network followed by a routing network similar to the double banyan network. However, the switching elements of the randomization network route every connection request to a random output port [69]. Since the randomization network randomizes the input traffic on a connection-basis rather than on a cell-basis like the Double Banyan network, the delivery of cells are guaranteed to be in-sequence.

Permutation-Banyan Network

This architecture is based on cascading two k-stacks of banyan networks: a randomization-stack network followed by a routing-stack network as shown in Figure 2.26. The increase in the topological complexity made this network inherently fault tolerant since there are many physical paths for each logical path. In addition, the network routing behavior gets close to the nonblocking networks when $k = \log_2 N$ [14].

Bypass Queues-Banyan Network

This architecture consists of N input buffers, N output buffers, N input port controllers, N output port controllers, and k N X N banyan network planes as shown in Figure 2.27.

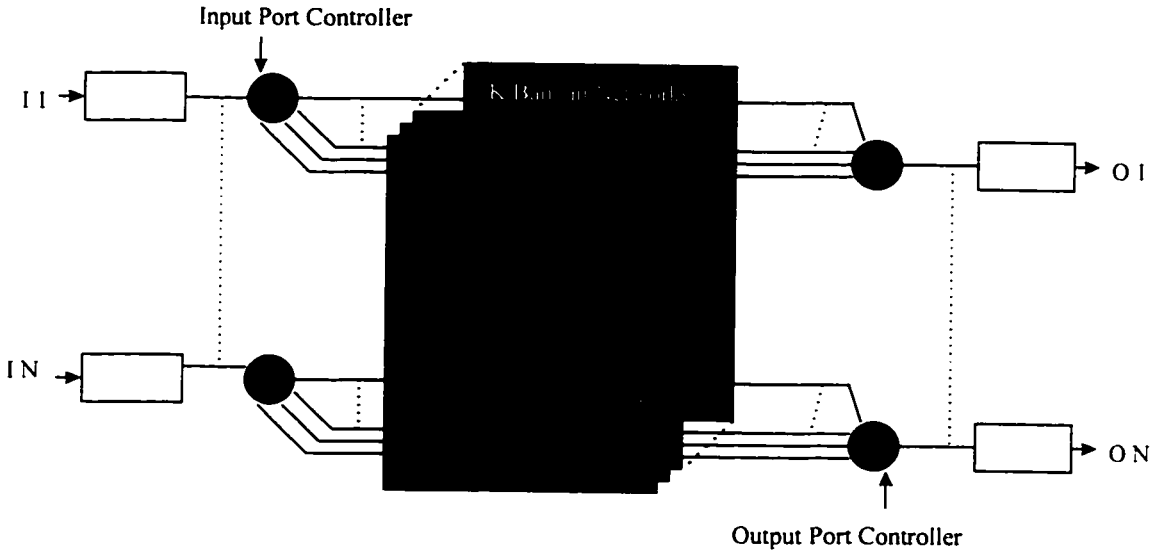


Figure 2.27 An N X N bypass queues-banyan network.

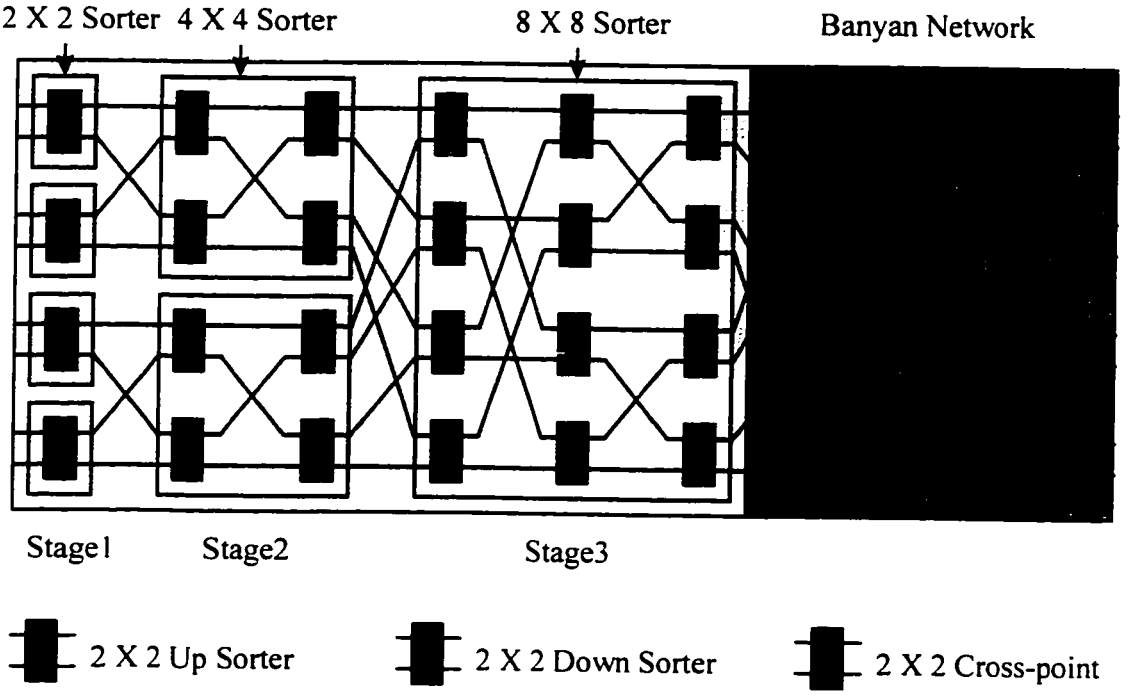


Figure 2.28 An 8 X 8 Batcher-banyan network.

This network is designed to minimize the internal and HOL blocking problems by allowing other cells in the input buffers, called bypass queues, to be transmitted when the leading cell is blocked. The input and output port controllers implement this function. It was shown that 90% throughput can be achieved for large size switch by using four banyan planes with Bypass queues. This throughput is higher than the recorded throughput of switches with output queuing [80].

Batcher-Banyan Network

This architecture consists of two consecutive networks: Batcher sorting and banyan networks as shown in Figure 2.28. Cells are first fed to a Batcher sorter in which they are sorted according to their destination address, and then routed by a banyan self-routing network [37]. Two building blocks are used in a Batcher sorter: up and down 2 X 2 cross-point sorters as shown in Figure 2.28. The up sorter sorts two numbers in descending order and the down sorter sorts two numbers in ascending order. In the first Batcher sorting stage, two pairs of numbers are sorted by two 2 X 2 sorters resulting into two sorted lists that are fed to the second sorting stage where they get sorted by a 4 X 4 sorter. In the last sorting stage, an 8 X 8 sorter then sort two sorted lists of four numbers. As shown in Figure 2.28, the sorting network consists of three stages ($\log_2 8$) of sorters. Stage-*i* sorter consists of *i* stages of 2 X 2 sorters. The total number of stages is equal to 6. In general, N X N Batcher sorter network requires

n(n+1)/2 stages, and Nn(n+1)/4 (2 X 2) sorters where $n = \log_2 N$. This network minimizes the internal blocking problem since the Batcher sorter proceeding the banyan network guarantees that all incoming cells to the banyan network follow the allowed various permutations of N concurrent input/output connections realizable in banyan networks. Furthermore, this type of network can minimize output conflicts easily by using trap and concentration networks after the batch sorter and before the banyan network as shown in Figure 2.29. If the input pattern presents output conflicts, multiple requests to the same output destination can be identified at the output of the Batcher sorter by comparing the output address requests over pairs of consecutive lines. The trap network removes all additional requests beyond the first request for each output line and the concentration network concentrates the selected cells to the top allowing various permutations of N concurrent input/output connections realizable in the banyan networks. The cells that are not selected by the concentration network are recirculated and fed back into the switching fabric at later time slots [12].

The *Starlite* switch, developed by AT&T Bell Laboratories shown in Figure 2.30, is the first sort banyan based switching fabric proposed in the literature [52]. It is similar to the Batcher banyan switch in Figure 2.29.

The *Sunshine* switch, developed by Bell Communications Research shown in Figure 2.31, achieves high performance by utilizing both internal and output buffering [25].

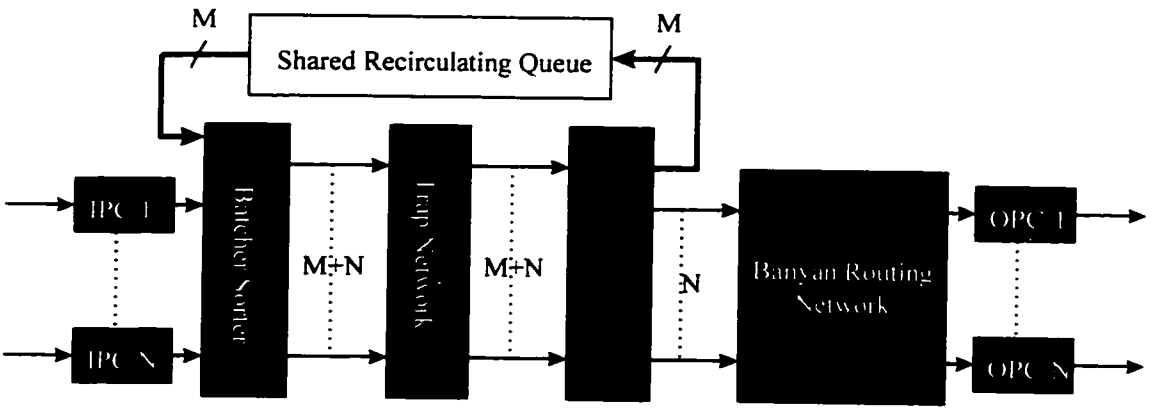


Figure 2.29 An N X N Batcher-banyan switching network.

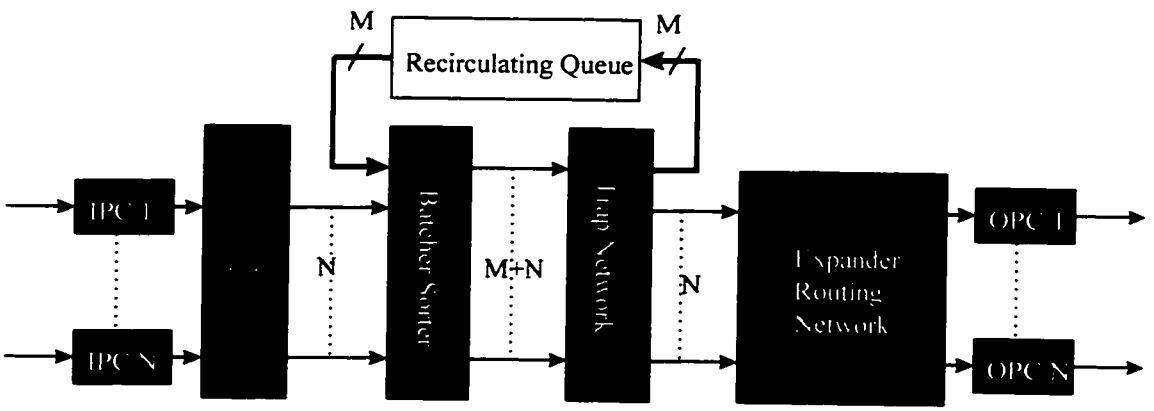


Figure 2.30 An N X N Batcher-banyan *Starlite* switching network.

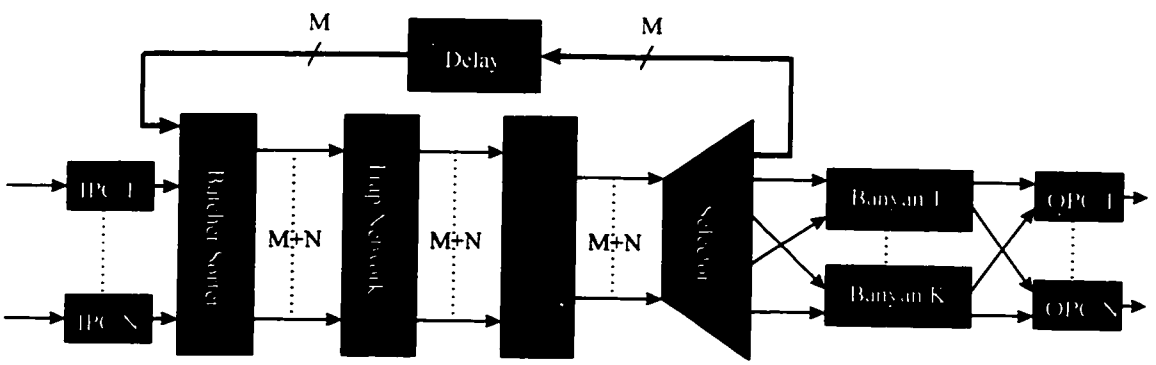


Figure 2.31 An N X N Batcher-banyan *Sunshine* switching network.

It combines a Batcher-banyan network with k multiple banyan networks used in parallel to route up to k cells to the same output. This solution decreases the rate at which cells are recirculated, and achieves a certain degree of output buffering. This queuing strategy results in an extremely robust and efficient architecture that is suitable for wide range of services requiring low cell loss probabilities. In a bursty environment where output overloads are likely, parallel banyan networks and output queues effectively handle this traffic. Without output queues, the shared queue requirements would be extremely large to support high occupancies or bursty services.

Rerouting-Banyan Network

The rerouting-banyan network is formed by adding $\log_2 N - 1$ extra stages after a banyan network with different switching element architecture as shown in Figure 2.32 [62]. The new switching element consists of 2 X 2 cross-point, 2 X 2 bypass links, contention controller, and two buffers. The routing algorithm is the same as that of banyan network except that when there is a cell contention, one cell gets routed properly and the other cell gets deflected and restarts its routing in the next stage. This means that there are partial banyan networks composed of switching elements and routing links from switching stages 1 to 4, 2 to 5, 3 to 6, and 4 to 7.

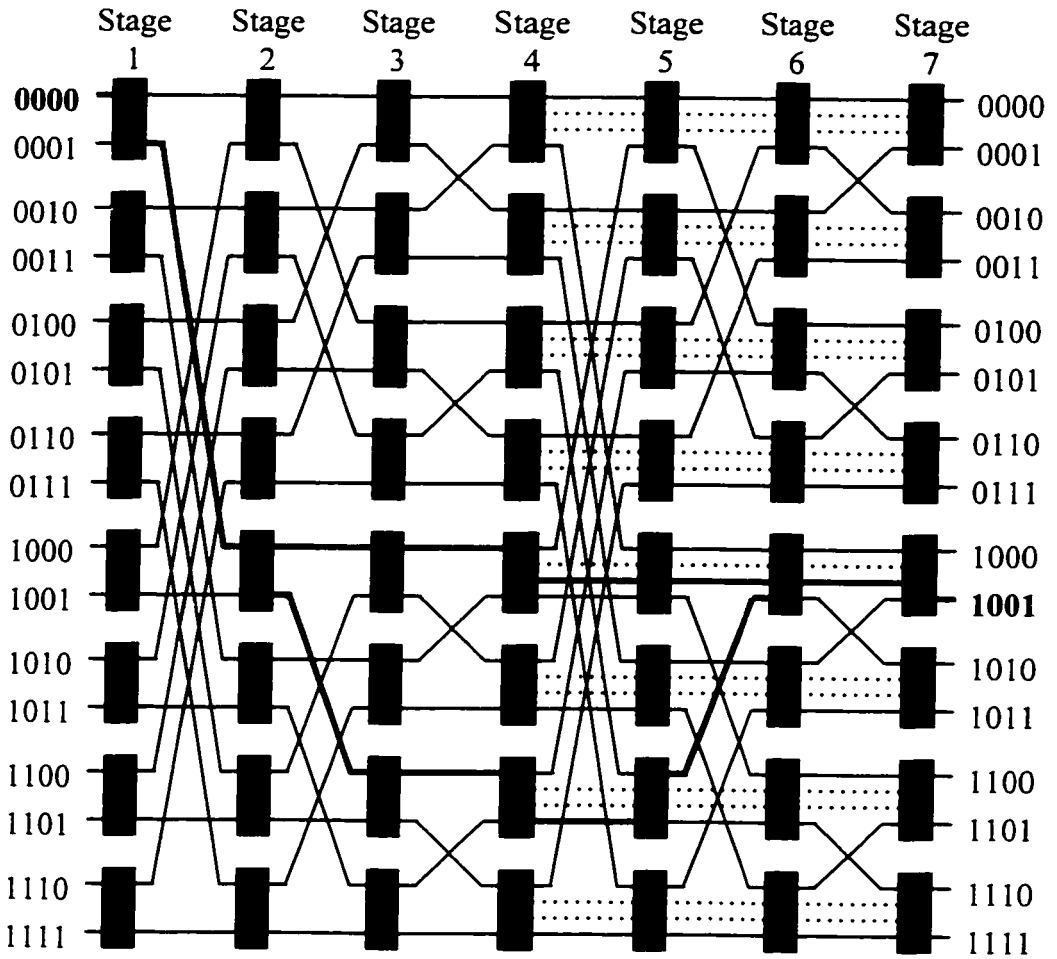


Figure 2.32 A 16 X 16 rerouting -banyan network.

For example, Figure 2.32 shows the routing algorithm to switch a cell from input 0000 to output 1001 under normal operation where there is no cell contention and when there is one cell contention at the fifth switching element of the second stage. The cell gets routed from the SE #1 of stage #1 to SE #5 of stage #2 based on the left most bit of the destination label. If there is no cell contention, the cell gets routed to SE #5 of stage #3 based on the second bit from left of the destination label. However, if there is cell contention on SE #5 of stage #2, then the cell gets deflected to SE #7 of stage #3 and the algorithm resets the routing sequence and starts the routing procedure based on the left most bit of the destination label. Notice that all the switching elements of stage #4 to stage #7 are based on the new switching element architecture.

In this network, a cell may start rerouting from any stage and finish its routing at any stage. Therefore, a cell's routing tag has an additional field indicating how many stages the cell should pass through from the present stage to its destination.

This network minimizes cell contention efficiently and it has high throughput and low cell loss probability even with hot-spot traffic. However, it does not consider fault tolerance.

Double Phase Banyan Network

The basic structure of an N X N double-phase banyan network is shown in Figure 2.33. There are N input controllers, k baseline networks, and k output controllers [4].

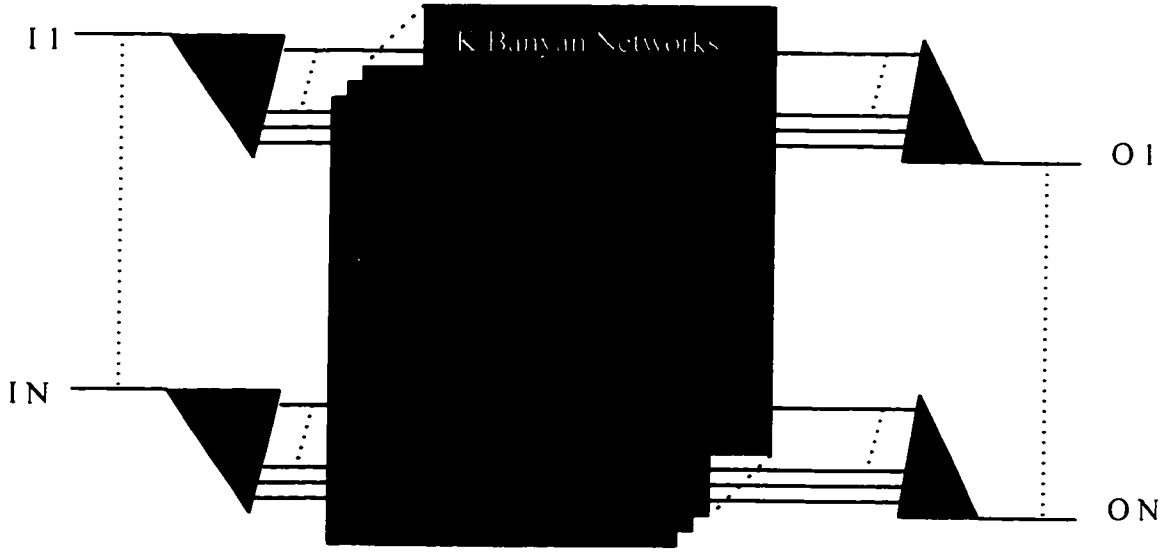


Figure 2.33 An N X N double phase banyan network.

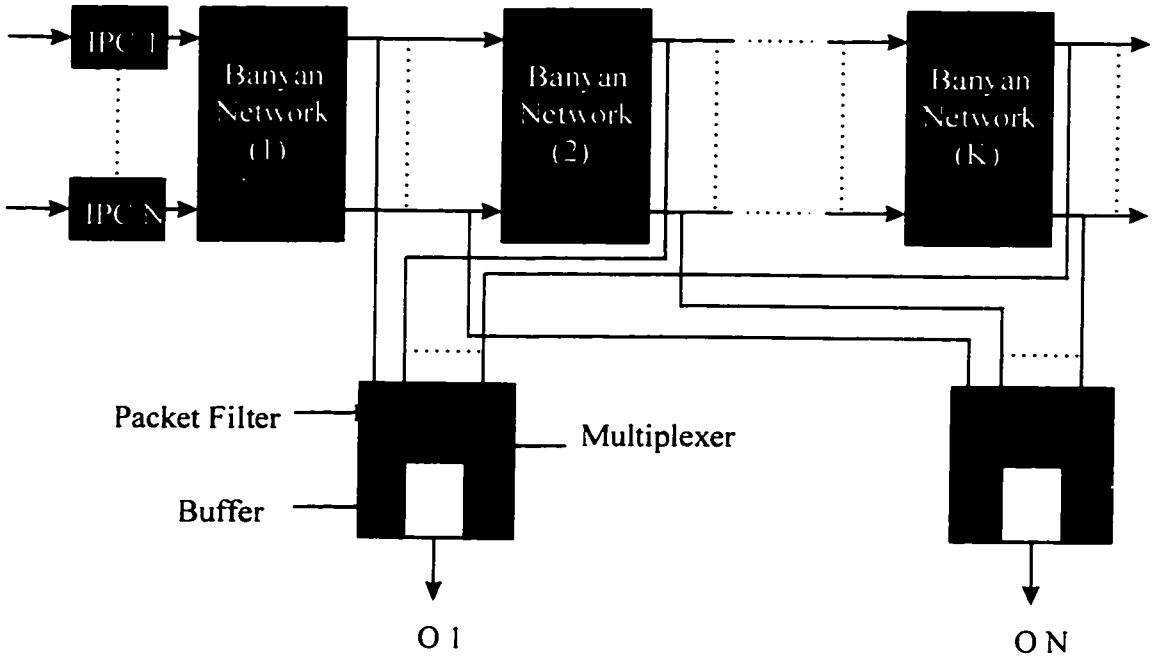


Figure 2.34 An N X N Tandem-banyan network.

The main function of the input controller is to accept cells from its corresponding input line, to issue requests through a certain baseline network, and to send cells to the baseline network selected after receiving acknowledgment. The input controller consists of local input queue buffer and a distributor or a demultiplexer to determine the appropriate baseline network for requests or cells to be routed through. The main function of the output controller is to direct requests from and to the selected baseline network, to accept cells from the baseline networks, to buffer and send them to the corresponding output line [4]. It has k cell filters to return the requests to acknowledge the original input port controllers in the arbitration phase and to allow the arrival cells to pass on the concentration in the transmission phase. In addition, it has one concentrator to select a subset l of the k output cells and to store them in l buffers through a shift register, and one shared buffer to allow sharing of the l buffers.

Tandem-Banyan Network

This Tandem-Banyan network consists of k banyan cascaded networks. The output of each banyan network is connected to the output port concentrators as shown in Figure 2.34 [11,57]. The output port concentrators are similar to those of the *Knockout* switch.

Every time slot, cells arrive at the input ports. These cells are fed to the first banyan network. When there is a cell contention, one cell is routed correctly and the other one

is marked and misrouted. The correctly routed cells at the end of the first banyan network exit to their respective output port. Those cells that are misrouted carry on into the next banyan network where the process is repeated, and so on till the last banyan network. Those misrouted cells at the end of the last banyan network are considered lost.

Pipeline Banyan Network

This architecture is based on parallel banyan data planes controlled by a control plane as shown in Figure 2.35. The control plane is for path reservation and the data planes are for cells routing. Each switching element in the control plane has select lines connecting to the corresponding switching elements in the data planes. When a cell arrives at an input port controller, its routing address header is sorted into the routing controller at the control plane. One field consists of two additional bits is added to the destination label to identify whether there is no cell, an active cell, a high-priority cell, or a broadcasted cell where it is to be sent to all output ports [47].

The time is divided into reservation slots. At each reservation slot, a data plane is selected on a round-robin basis. The HOL headers are transmitted into the control plane and are self-routed to their destinations. The 2 X 2 switching element of the control plane consists of a self-routing decision circuit and two pairs of multiplexers.

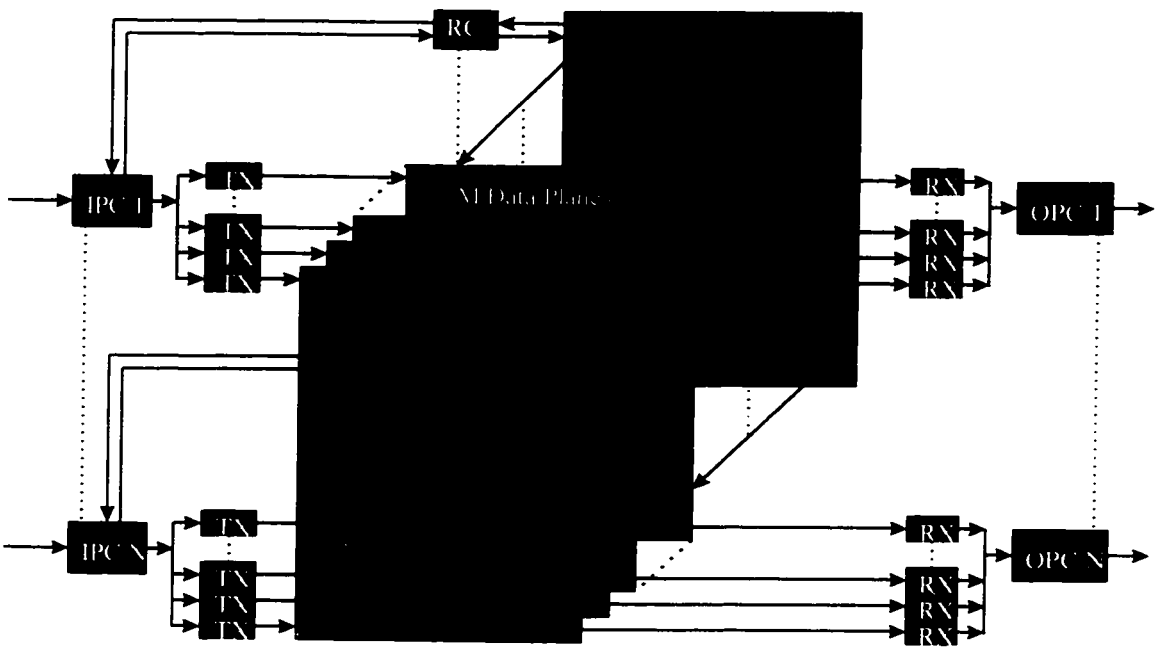


Figure 2.35 An N X N pipeline banyan network.

Based on the first three header bits, the switching element sets itself into one of the switching states: straight, cross, up-broadcast and down-broadcast, and forward the routing headers. Simultaneously, the state information is captured for setting the switching state of the reverse multiplexers and the corresponding switching element in the selected data plane. If there is a conflict between two headers, on header is selected for further routing and the other one is dropped. Eventually, a header succeeded in reaching its destination will have set up two paths: one reverse path at the control plane and one forward path at the selected data plane. A one-bit acknowledgment signal is then sent back through the reverse path to notify which routing controller has succeeded in making reservations. Upon receiving the acknowledgment signal, the routing controller removes its HOL header and notifies the corresponding input controller. While a reservation is taking place, the input controller writes a copy of the HOL cell into the transmit register of the selected data plane. When the input controller receives an enable signal from the corresponding routing controller, the cell is routed through the data plane to the output register. Cells delivered by individual data planes are merged into the same output buffer. If the output buffer is full, the destination port can stop an input port from transmission by disabling the acknowledgment signal. The input port will have to reserve again at the next reservation slot.

This architecture can give a close to 100% maximum throughput, and gives a delay that is relatively independent of the switch size. This network guarantees the cell delivery in sequence.

Parallel-Tree Banyan Switching Fabric (PTBSF)

This architecture is based on multiple banyan networks interconnected in binary tree structure as shown in Figure 2.36 [74]. Each banyan plane uses nonblocking 3 X 4 switching elements. Two outputs are used for correct routing and the other two outputs are used to move cells to the next level in the binary tree. In other words, this switching element can route only one cell correctly and the remaining two cells will be handled in the next level. Starting from the second level, nonblocking 1 X 2 demultiplexers are used in the first stage of each banyan plane. The input cells arrive at the inputs of the topmost banyan plane. Then the cells are routed using the self-routing algorithm for simple banyan network. When a conflict occurs between two cells in the first level, one of the cells is routed correctly, while the other cell is moved to the next lower level. In other words, the other cell will be given another chance for correct routing in the next lower level. So that, internal cell contention is minimized by locally distributing the conflicting cells over different banyan planes. From the second level onwards, there is a possibility to have three cells arrive at the input of each 3 X 4 switching element. One cell is routed correctly in the same banyan plane.

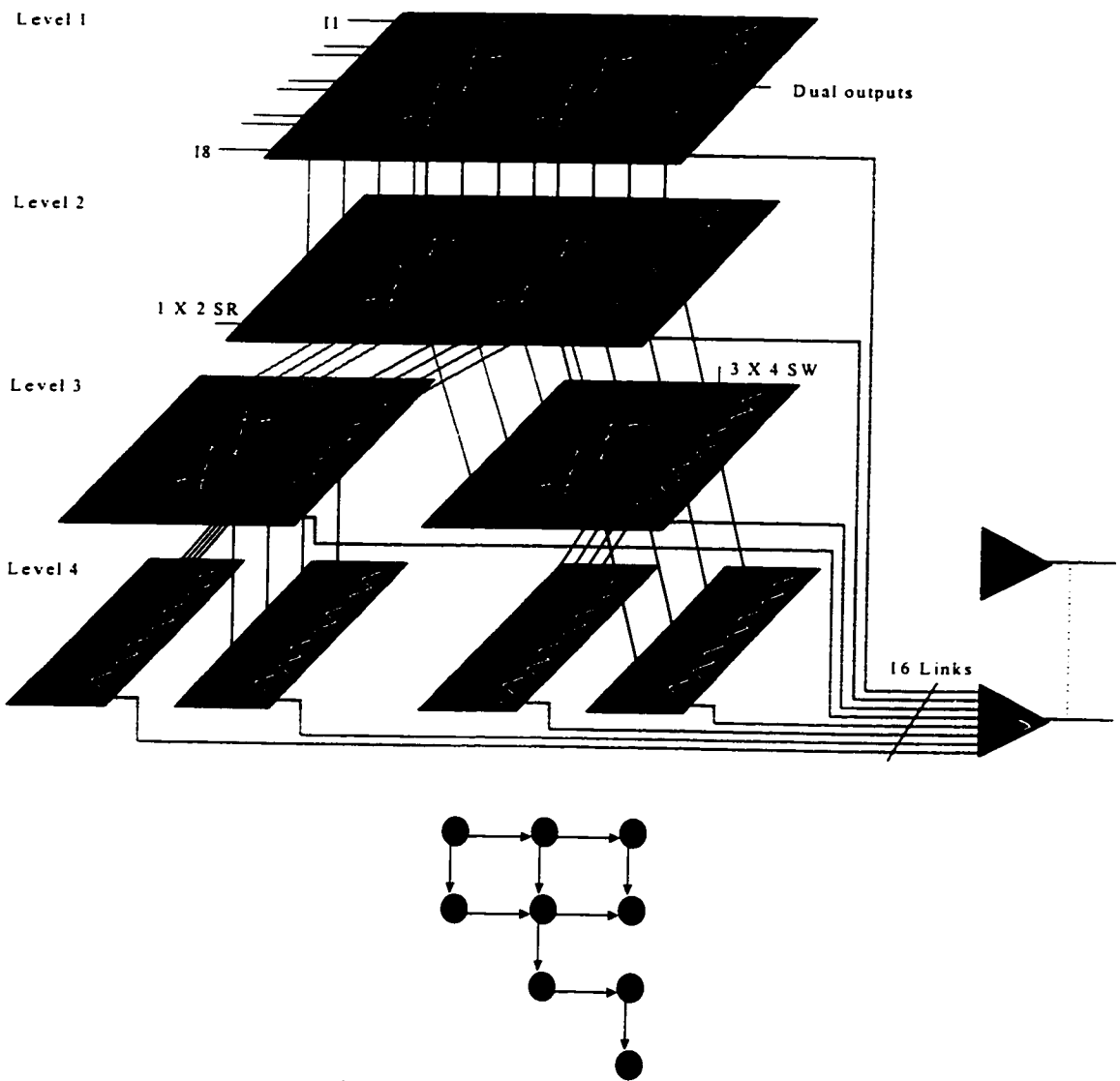


Figure 2.36 An 8 X 8 PTBSF.

The second cell is moved to the left banyan plane in the next lower level. The third cell is moved to the right banyan plane in the next lower level. In the second level, the banyan plane has the demultiplexers to replace the switching elements in the first stage. In the third level, the banyan planes have the demultiplexers to replace the switching elements in the second stage. Hence, from level three onwards, one stage reduces at each level for all banyan planes in that level. At each level, the switching elements in the last stage of each banyan plane have two access links to the output ports. Therefore, output contention problem is solved using many access links to each output port. However, there is a possibility to loss cells sequence when cells go through vertical levels which introduce high jitter.

Fault tolerance is not considered since the first stage of the topmost level is critical. However, from the first stage onwards, the faulty switches are bypassed by moving the cells to the next lower level as illustrated by the channel graph in the bottom of Figure 2.36.

2.3.2.3.2 Fault-Tolerant Banyan-Based Networks

MD-Omega Network

This architecture is based on a banyan network, Omega, with N multiplexers added to the input side of the Omega network, and N demultiplexers replacing the N/2 switching elements in the last stage as shown in Figure 2.37.

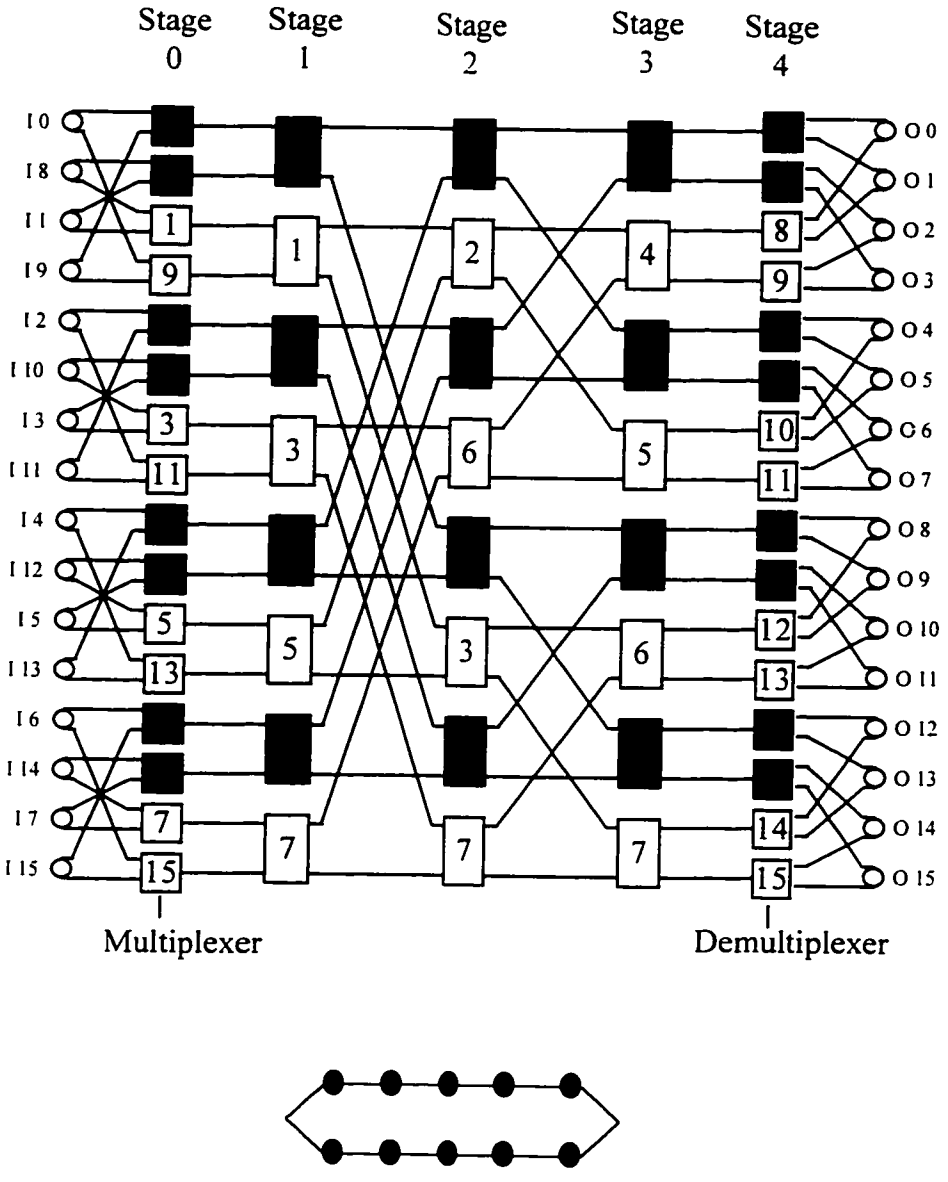


Figure 2.37 A 16 X 16 MD-Omega network.

The network is divided into two planes as indicated in Figure 2.37 by different shading colors of the switching elements. The MD-Omega network provides two disjoint paths, one in each plane, for every input-output pair. As a result, this network is a single fault tolerant as shown in the channel graph in the bottom of Figure 2.37 [76]. Notice that the multiplexers do not use the routing label. However, the demultiplexers use the last bit in the destination label. The internal blocking problem is not solved.

Extra Stage Shuffle-Exchange Network

The Extra Stage Shuffle-Exchange network is formed by adding an extra stage in front of the input side of a shuffle-exchange network as shown in Figure 2.38. The addition of this stage is to improve the fault tolerance of the shuffle-exchange network by providing two paths for each input-output pair. However, the two paths are disjoint in the middle stages and they share the same switching elements in the input and output stages as shown by the thick links and the channel graph in Figure 2.38 [71].

Extra Stage Cube Network

The Extra Stage Cube network is formed by adding an extra stage to the input side of a banyan network along with multiplexers at the input stage and demultiplexers at the output stage as shown in Figure 2.39.

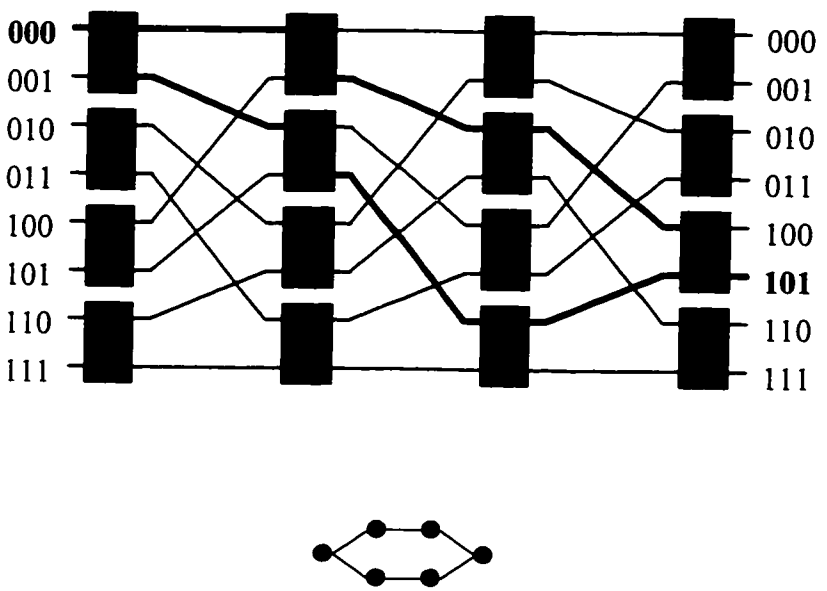


Figure 2.38 An 8 X 8 extra stage shuffle-exchange network.

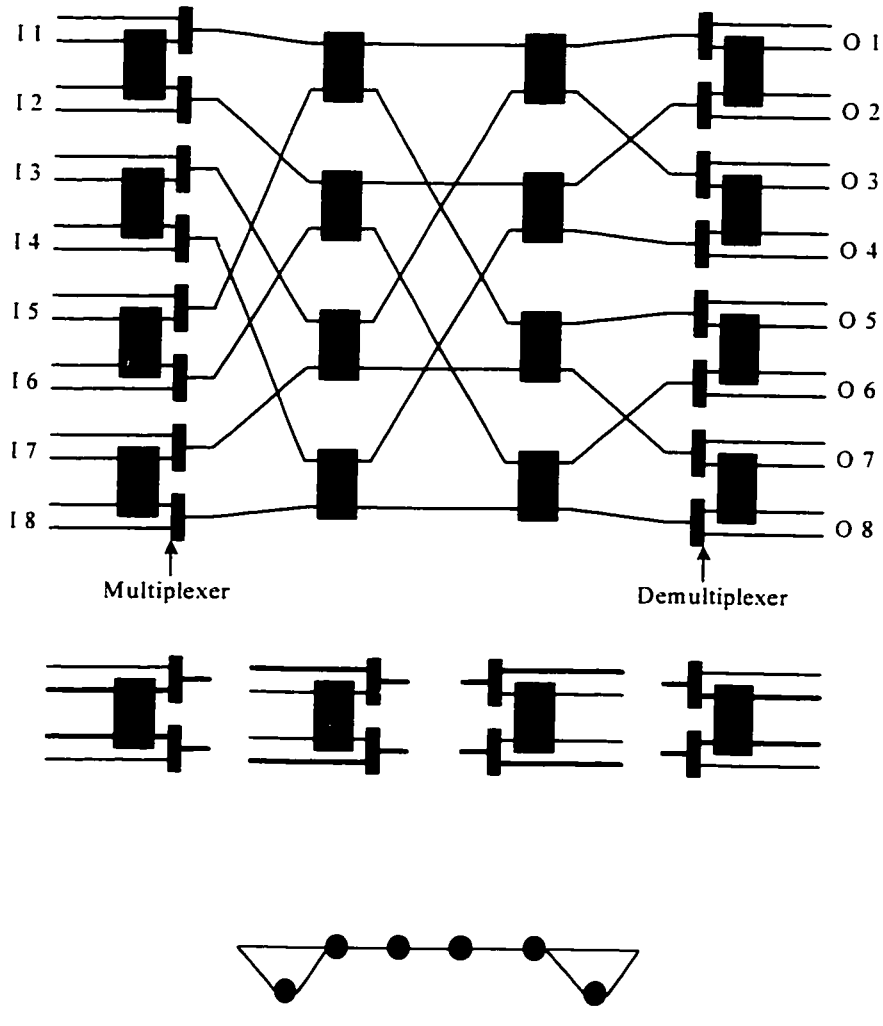


Figure 39 An 8 X 8 extra stage cube network.

The multiplexers at the input stage are used to enable or disable the input stage. On the other hand, the demultiplexers are used to enable or disable the output stage. A stage is enabled when its switches are being used to provide interconnection and it is disabled when its switches are being bypassed as shown in the bottom of Figure 2.39. During normal operation, the input stage is disabled and the output stage is enabled forming a regular banyan network [16].

The network reconfigures itself when it finds a fault after running the fault detection and location tests. If a fault occurs in an output stage, the input stage is enabled and the output stage is disabled. If a fault occurs in the input stage, the output stage is enabled and the input stage is disabled. Faulty switches in the middle stages are not tolerated as shown in the channel graph in the bottom of Figure 2.39. Multiple-fault tolerance can be achieved by individually enabling and disabling the switching elements in the input and output stages.

Benes' Network

This architecture consists of two baseline networks mirrored to each other sharing the middle stage as shown in Figure 2.40. This network has four possible routes from any input to any output line. The blocking problem and the fault tolerance of switching elements are solved in the first $(\log_2 N) - 1$ stages.

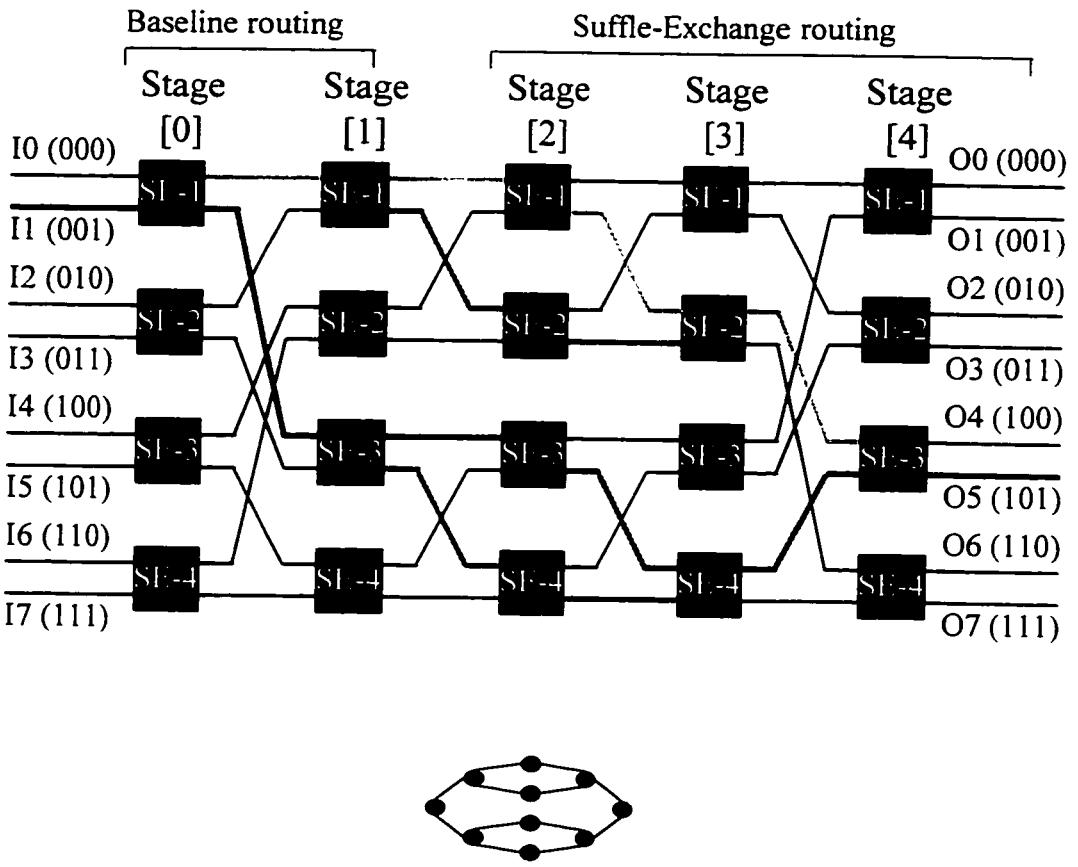


Figure 2.40 An 8 X 8 Benes' network.

However, the remaining $\log_2 N$ stages are still having the blocking problem and if there is a fault in any of the switching elements, cells will be lost. Each cell has to be routed through $2(\log_2 N)-1$ stages. It is almost double the time it takes through the previous networks. The routing procedure through the network is divided into two sections. The first $(\log_2 N)-1$ stages use the baseline routing and if there is an internal contention through a switching element, it deflects one cell to the wrong output port of the SE. The remaining stages use the shuffle-exchange routing algorithm and if there is an internal contention through a SE, it either discards one cell or deflects it to the wrong output port of the SE, then the cell goes to the wrong destination [7].

The shuffle-exchange routing procedure is as follows:

- the cell is in SE #2 of stage #2. SE #2 checks the third bit (MSB) of the destination address. Since it is equal to 1, it routes the cell through its lower output port. (Cell goes to SE #2 of stage #3)
- SE #2 checks the second bit of the destination address. Since the bit is equal 0, it routes the cell through its top output port. (Cell goes to SE #3 of stage #4)
- SE #3 checks the first bit (LSB) of the destination address. Since it is equal to 1, it routes the cell through its lower output port. (Cell goes to output line 5)

Figure 2.40 shows the four possible routes (thick links) from input line 1 to output line 5. The following steps describes the normal route from line 1 to output line 5 if there is no internal contention or faulty switching elements in the path:

- the cell in SE #1 of stage #0 has two routes to the next stage. Either it goes to SE #1 or SE #3 depending on the status of those switching elements and the internal contention (two cells going to the same switching element in the next stage). Following the baseline routing, SE #1 routes the cell to SE #3 of stage #1 since the third bit (MSB) of the destination address is equal to 1.
- the cell in SE #3 has two routes to the next stage. Either it goes to SE #3 or SE #4. Normally, the cell goes to SE #3 since the second bit of the destination address is equal to 0.
- the cell in SE #3 has only one route to the next stage using the shuffle-exchange algorithm. The cell goes to SE #4 since the third bit (MSB) of the destination address is equal to 1.
- the cell in SE #4 has only one route to SE #3 of the next stage since the second bit of the destination address is equal to 0.
- SE #3 routes the cell to the output line 5 since the first bit (LSB) of the destination address is equal to 1.

The other three routes as follows:

- if SE #3 of stage #1 is faulty, then SE #1 of stage #0 routes the cell to SE #1 of the next stage. Then the cell is routed to SE #1 of the next stage. Then it is

routed to SE #2 of the next stage. Then it is routed to SE #3 of the next stage. Then it is routed to output line 5.

- if SE #3 of stage #2 is faulty, then SE #1 of stage #0 routes the cell to SE #3 of the next stage. Then the cell is routed to SE #4 of the next stage. Then it is routed to SE #4 of the next stage. Then it is routed to SE #3 of the next stage. Then it is routed to output line 5.
- if SE #3 of stage #1 and SE #1 of stage #2 are faulty, then SE #1 of stage #0 routes the cell to SE #1 of the next stage. Then the cell is routed to SE #2 of the next stage. Then it is routed to SE #2 of the next stage. Then it is routed to SE #3 of the next stage. Then it is routed to output line 5.

Figure 2.40 shows the channel graph of an input-output connection. Notice that all four paths start from the same switching element and merge in the last switching element. The first and last stages have to work correctly in order for the network to route the cells to the proper destinations. In addition, all switching elements of the last $\log_2 N$ stages have to function correctly in order to guarantee the proper cells routing for the whole network.

SEROS Switching Element

The SEROS switching element can be used in any banyan network. It is an intelligent 2 X 2 fault tolerant switch [43]. The switching element consists of three circuits:

bypass, error detection, and routing as shown in Figure 2.41a. Figure 2.41b shows the normal operation of the switching element. It routes the cell according to the left most bit of the destination label. The bypass circuit is enabled when the routing circuit is faulty. If the bypass circuit is enabled and there is only one cell at input ports as shown in Figure 2.41c, the input cell is broadcasted to both switching element outlets. In addition, "0" is appended to the error field of the cell in the upper switch outlet, and "1" is appended to the error field of the cell in the lower switch outlet. When there are two cells at the switch inputs, the upper input cell is selected for broadcasting and the other cell waits in the queue for broadcasting in the next cycle. The error detection circuit checks the error bits in the error field of both cells. If the error field is empty, the cell is passed to the routing circuit. Otherwise, it compares the left most bit of the error field with the left most bit of the destination label and discards the cell if the bits are different. The circuit eliminates the compared bits of the destination label and the error field and repeat the same operation on the left most bit of the error field until either the cell is discarded or the error field becomes empty; then the cell is passed to the routing circuit. The routing circuit consists of local buffers and contention controller. It stores the input cells in the local buffer of each switch inlet. Then, two input cells from the local buffers are routed. When there is an output contention, one cell is routed and the other cell is recirculated to the local buffer by the contention controller.

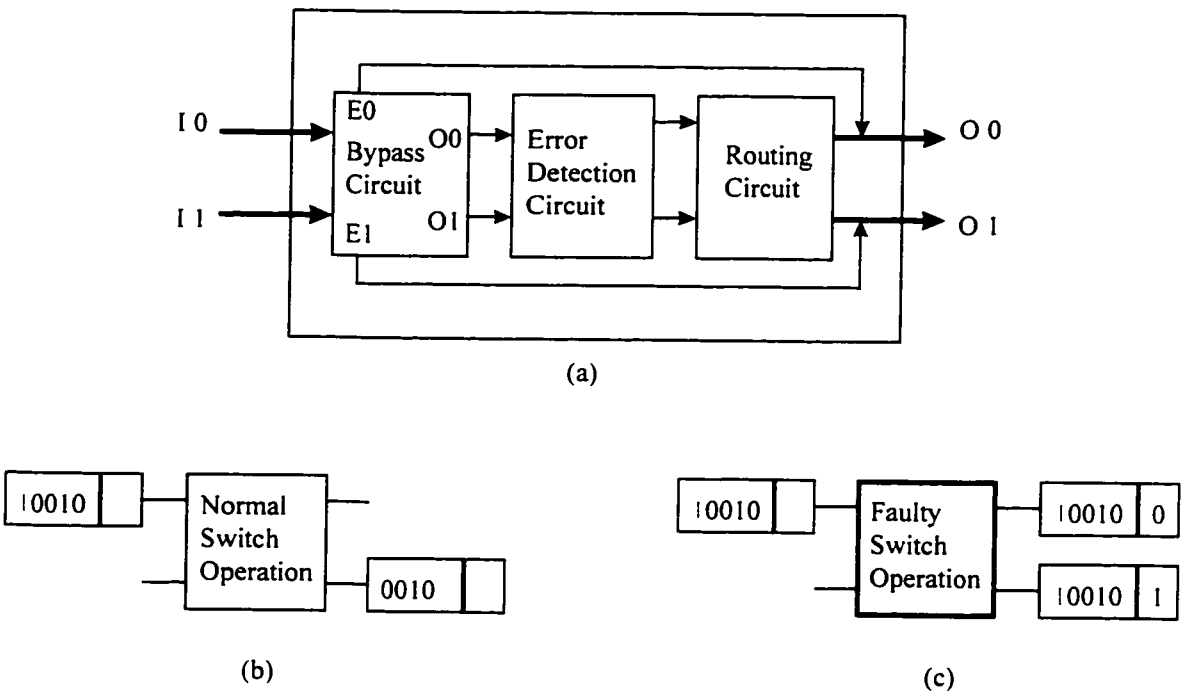


Figure 2.41 A 2 X 2 SEROS switching element.

This switching element minimizes the cell loss probability, yet it introduces high jitter, increases network complexity, and dose not preserve cell sequence.

Itoh's Network

This architecture consists of a modified version of the baseline network with added subswitches between stages in order to increase the number of paths from any input to any output lines as shown in Figure 2.42. In an 8 X 8 network, there are five possible paths from any input to any output line. This is a non-blocking network with one exception at the last stage of the network when there is more than one cell routed to the same destination. Normally the cell would be routed through $\log_2 N$ stages [3]. Figure 2.42 shows that in the worst case the cell would be routed through five links in an 8 X 8 network.

There is an inherent problem with this architecture. The problem is that cells might get out of order at the destination. Normally, the routing procedure is the same as that of the baseline network. However, if there is an internal contention in one SE or if the next SE is faulty, the cell is routed to the subswitch of rank-1 (R1 subswitch). If there is also contention in R1 or if the next SE is faulty, the cell is routed to the subswitch of rank-2 (R2 subswitch). R1 and R2 checks the same destination bit for routing as the other SE's in the same stage.

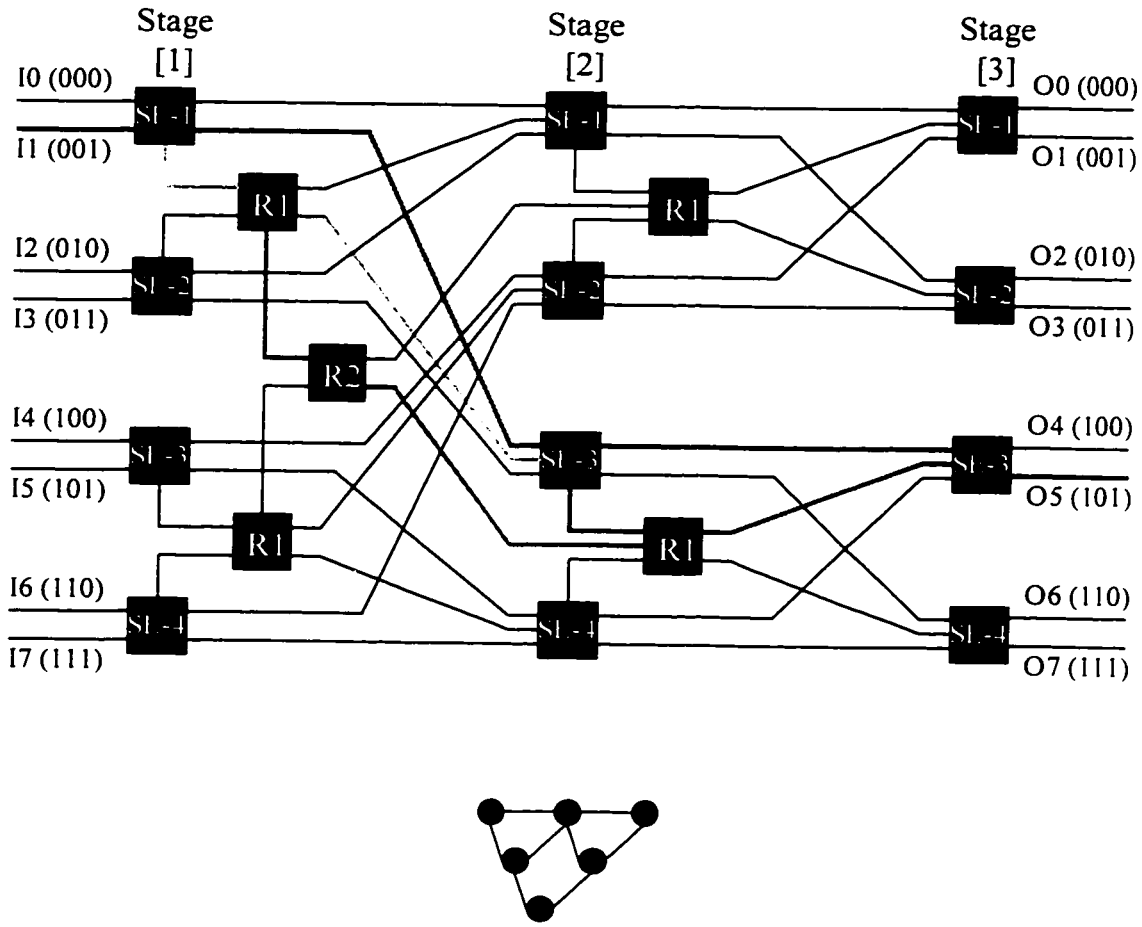


Figure 2.42 An 8 X 8 Itoh's network.

The basic problem with this architecture is the usage of different switching elements. In addition, the rank subswitch set-up is different in each stage. There are five possible routes to route cells from input line 1 to output line 5. The first one is the normal route based on the baseline network routing. The cell traverses through the normal route in the following order: SE #1 of stage #1, SE #3 of stage #2, SE #3 of stage #3, and then the output line 5.

The following list describes the remaining four routes:

- if the link between SE#1 of stag#1 and SE#3 of satge#2 is faulty, then SE#1 of stage#1 routes the cell to the upper R1 subswitch of stage#1. Then it is routed to SE#3 of stage#2. Then it is routed to SE#3 of the next stage. Then it is routed to output line 5.
- if SE#3 of stage#2 is faulty, then the cell is routed to the upper R1 subswitch of stage#1. Then it is routed to R2 subswitch of stage#1. Then it is routed to the lower R1 subswitch of stage#2. Then it is routed to SE#3 of the next stage. Then it is routed to output line 5.
- if the link between SE#3 of stage#2 and SE#3 of satge#3 is faulty, then SE#1 of stage#1 routes the cell to SE#3 of the next stage. Then it is routed to the lower R1 subswitch of the same stage. Then it is routed to SE#3 of the next stage.
- if the link between SE#1 of stage#1 and SE#3 of satge#2 is faulty and the link between SE#3 of stage#2 and SE#3 of satge#3 is faulty , then SE#1 of stage#1

routes the cell to the upper R1 subswitch of stage#1. Then the cell is routed to SE#3 of the next stage. Then it is routed to the lower R1 of the same stage. Then it is routed to SE#3 of the next stage. Then it is routed to output line 5.

Notice that all five paths start from the same switching element and merge in the last switching element as shown in Figure 2.42. The first and last stages have to work correctly in order for the network to route the cells to the proper destinations. The intermediate stages can have five routes for any input to any output line. So that, the problem of internal contention and the problem of having single faulty link or switching element faulty is minimized at the expense of adding subswitches, using 3 X 3 switching elements, and delivering cells out-of sequence.

Parallel Banyan Network

This architecture consists of two parallel baseline networks connected using input and output routers as shown in Figure 2.43 [52]. Each baseline network is called a plane. Depending on the function of the routers, this architecture could double the throughput of a single baseline by routing two cells concurrently in two planes. The switching fabric can tolerate any switch faults in the first stage by routing the cell to the second plane if the switching element of the first plane is faulty. Each plane is still a blocking network. This switching fabric is fast since cells pass through only $\log_2 N$ stages to reach the output line.

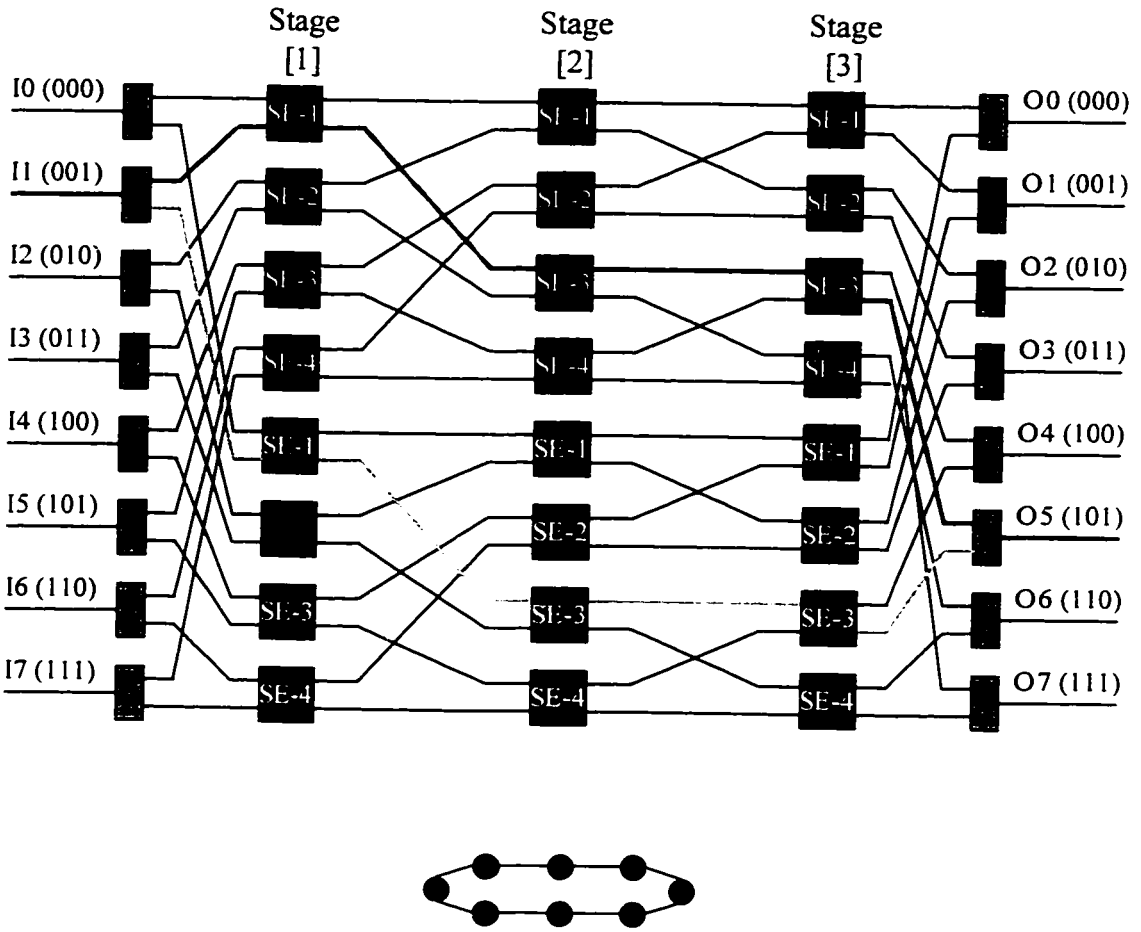


Figure 2.43 An 8 X 8 parallel banyan network.

The number of disjoint paths is doubled, as shown in the bottom of Figure 2.43, at the expense of doubling the hardware of the baseline network and adding the input multiplexers and output demultiplexers. The routing procedure is the same as that of a single baseline network. In Figure 2.43, the dashed lines show the only two possible routes from input line 1 to output line 5.

Tagle & Sharma's Network

This architecture consists of two baseline networks similar to that of the parallel baseline network. However, 4 X 4 switching elements are used to allow routing from one baseline network (plane) to the other one if required as shown in Figure 2.44. In an 8 X 8 network, there are 8 possible paths from any input to any output line as shown by the channel graph in the bottom of Figure 2.44. This network is simple and minimizes the internal blocking problem in the middle stages and the problem of faulty switching elements within a plane. However, the last stage can route only two cells to one output destination. So that, when there is more than one cell going to the same destination, only two would reach the destination and the remaining cells are lost [49,50].

This switching fabric is fast since cells pass through only $\log_2 N$ stages to reach the output destination.

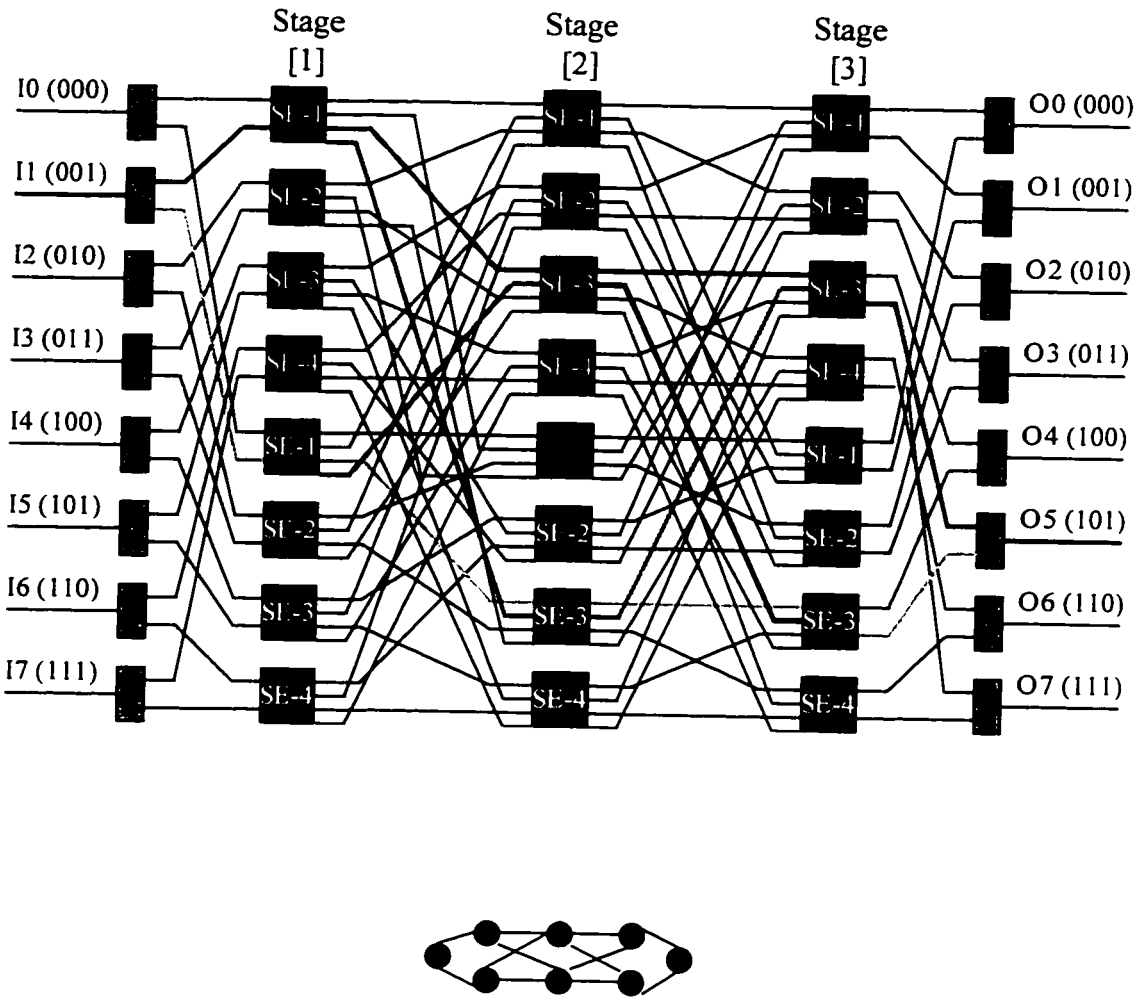


Figure 2.44 An 8 X 8 Tagle & Sharma's network.

In addition, it could double the throughput of a single baseline by routing two cells concurrently in two planes unless there is a faulty switch or cell congestion in one of the planes where one cell would be lost. The routing procedure is the same as that of the baseline network with the exception that if there is an internal contention in one of the SE's in the first $(\log_2 N) - 1$ stages, it routes the cell to the next stage in the other plane.

Baseline-Tree (B-Tree) Network

This architecture is based on multiple interconnected baseline networks to provide multiple paths from any input to any output line with minimum cell loss. Figure 2.45 shows the network connection of an 8 X 8 network and the possible eight paths from input line 1 to output line 5 (thick links). This switching fabric is fast since cells pass through only $\log_2 N$ stages to reach the output destination. It basically provides N different routes and $2(\log_2 N)$ access links to each output port. In other words, if eight cells are routed to the same destination, only two cells will be lost and the remaining cells will reach the destination port concentrator [27,28].

Figure 2.45 shows the connection of the network stages, (six stages: each stage consists of four 4 X 4 switching elements), and output port 5 only. The remaining output ports are not shown to simplify the drawing.

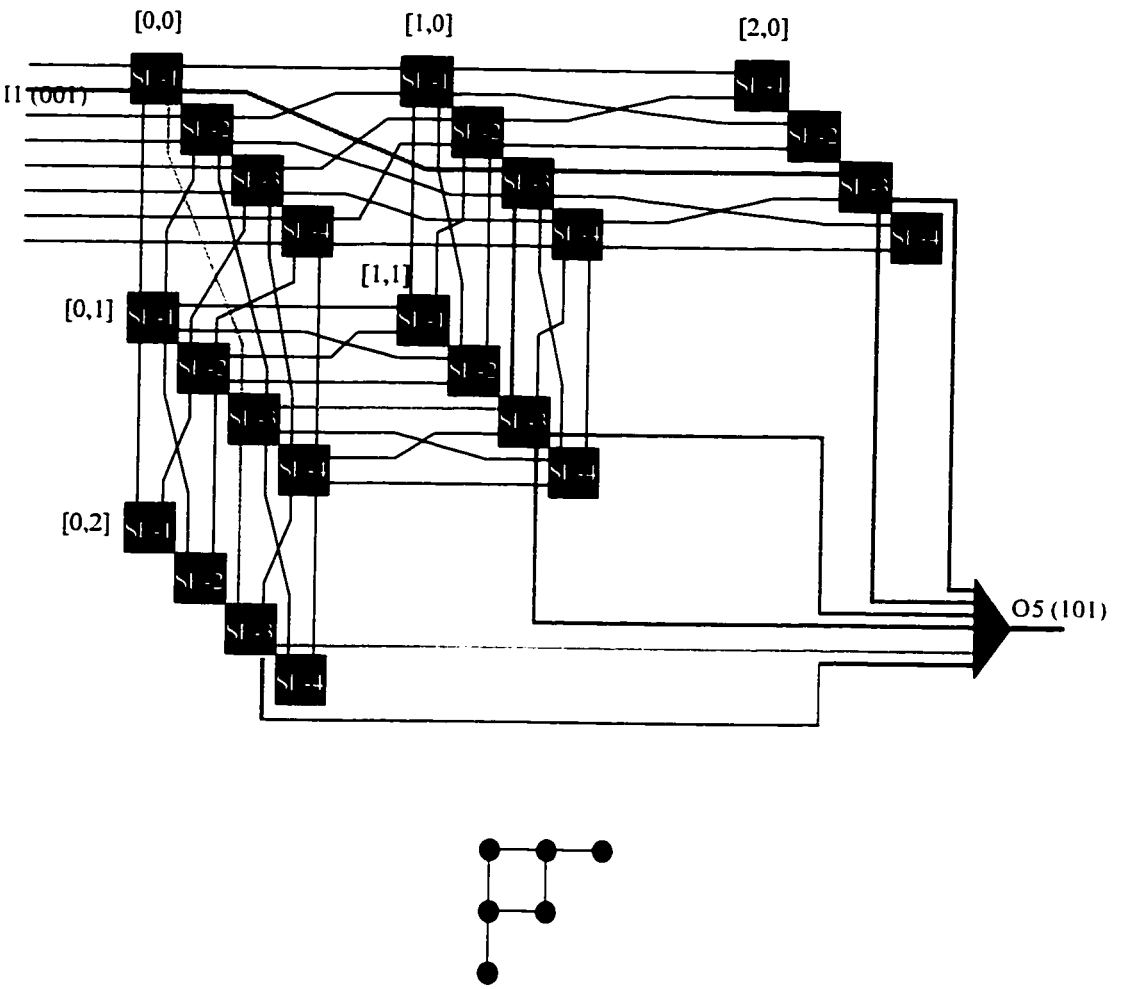


Figure 2.45 An 8 X 8 baseline-tree network.

Each switching element has four inputs, two formal outputs, and two redundant outputs. The switching element is capable of routing four different cells to four different routes if there is no fault in any of the switching elements of the next stages and the routing destination bit is 0 for two cells and 1 for the other two cells. Otherwise, some cells would be lost. This capability is great compared to the other network architectures.

The following is a description of the normal route from input line 1 to output line 5:

1. the cell in SE#1 of the upper left stage has two possible routes: horizontal and/or vertical. Normally SE#1 routes the cell to the next horizontal stage unless there is a faulty switching element or congestion. SE#1 routes the cell to SE#3 of the next horizontal switch since the third destination bit is equal to 1.
2. SE#3 routes the cell to SE#3 of the next horizontal stage since the second destination bit is equal to 0.
3. SE#3 routes the cell to output port 5 through the horizontal link since the first destination bit is equal to 1.

The following is a description of the remaining seven routes from input line 1 to output line 5:

1. if there were a fault in the horizontal link of SE#3 of the stage on the first row and third column, then the route would be the same as the normal route with the exception on the last routing link. SE#3 routes the cell to the output port 5 through the vertical link.

2. if there is a fault in SE#3 of the first row and third column, then SE#3 routes the cell to SE#3 on the next horizontal stage. Then SE#3 routes the cell to SE#3 of the stage on the first row and first column. Then SE#3 routes the cell to the output port 5 through the horizontal link.
3. if there were a fault, in addition to the previous case, in the horizontal link of SE#3 of the stage on the first row and first column, then the route would be the same as the previous case with the exception of the last routing link. SE#3 routes the cell to the output port 5 through the vertical link.
4. if there is a fault in SE#3 of the stage on the first row and second column, then SE#1 routes the cell to SE#3 of the stage on the second row and first column. Then SE#3 routes the cell to SE#3 of the stage on the second row and second column. Then SE#3 routes the cell to the output port 5 through the horizontal link.
5. if there were another fault, in addition to the previous case, in the horizontal link of SE#3 of the stage on the second row and second column, then the route would be the same as the previous route with the exception on the last routing link. SE#3 routes the cell to the output port 5 through the vertical link.
6. if there is another fault, in addition to the first case, in SE#3 of the stage on the second row and second column, then SE#1 routes the cell to SE#3 of the stage on the second row and first column. Then SE#3 routes the cell to SE#3 of the stage on the third row and first column. Then SE#3 routes the cell to the output port 5 through the horizontal link.

7. if there were another fault, in addition to the previous case, in the horizontal link of SE#3 of the stage on the third row and first column, then the route would be the same as the previous route with the exception on the last routing link. SE#3 routes the cell to the output port 5 through the vertical link.

This switching fabric is fast since cells have to go through only $\log_2 N$ stages to reach the output port. In addition, it could double the throughput easily by routing two cells concurrently in two disjoint paths. The other six routes are to handle internal congestion and faulty switching elements. The routing procedure is the same as that of the baseline network with the exception that if there is an internal contention in one of the SE's, it routes the cell to the next stage using the redundant output links. The channel graph of the B-Tree network, shown in the bottom of Figure 2.45, indicates that the input stage is critical and all switches have to function correctly in order to use the available redundant paths in the network. A fault tolerant version of the B-Tree is called B-Tree (1) and shown in Figure 2.46. This version has more redundant paths and more access links to the output port. The network channel graph, illustrated in the bottom of Figure 2.46, shows the resolution of the critical stage problem in the B-Tree network.

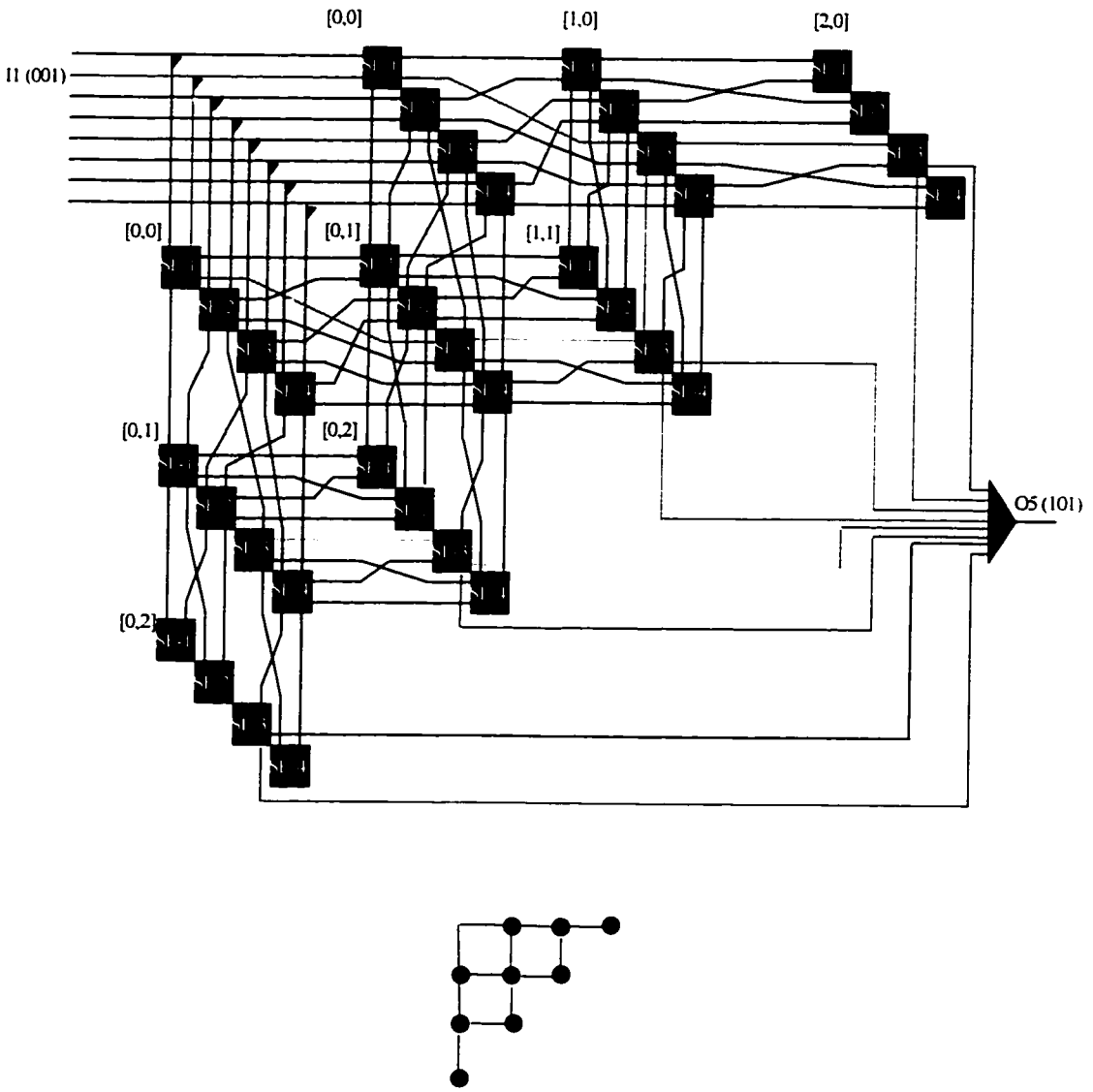


Figure 2.46 An 8 X 8 baseline-tree(1) network.

LIN and WANG's Banyan Network

The architecture is based on banyan network with additional redundant switching elements as shown in Figure 2.47 [25]. A 16 X 16 switch architecture consists of one demultiplexer for each input port, one multiplexer for each output port, and four stages. Each stage has two types of switching elements: base and redundant. The base switching elements are connected as in baseline network. On the other hand, the redundant switching elements are connected to support the fault-tolerance feature. The switching elements are labeled by $[P_{n-1}P_{n-2}...P_k]_x^i$ where $1 \le k \le n-1$, i is the stage number, and x denotes whether the switching element is a base switch or not. If the switching element is a base switch, $x = 0$; if the switching element is a redundant switch in the medium stages, $x = 1$; if the switching element is a redundant switch in the first and last stages, $x = 2$. The switching element has three types of links: formal, redundant, and standby. As there is no contention; only the formal links are used in the routing process. If only one contention occurs, both the formal and the redundant links are used in the routing process. If more than one contention occurs, all links are used to minimize the contention problem. This switching fabric has many redundant paths as illustrated by the channel graph shown in the bottom of Figure 2.47. There are 120 redundant paths between any input port and output port. There are two access links to each output port to minimize output contention. Overall, the switching fabric has good performance under normal conditions.

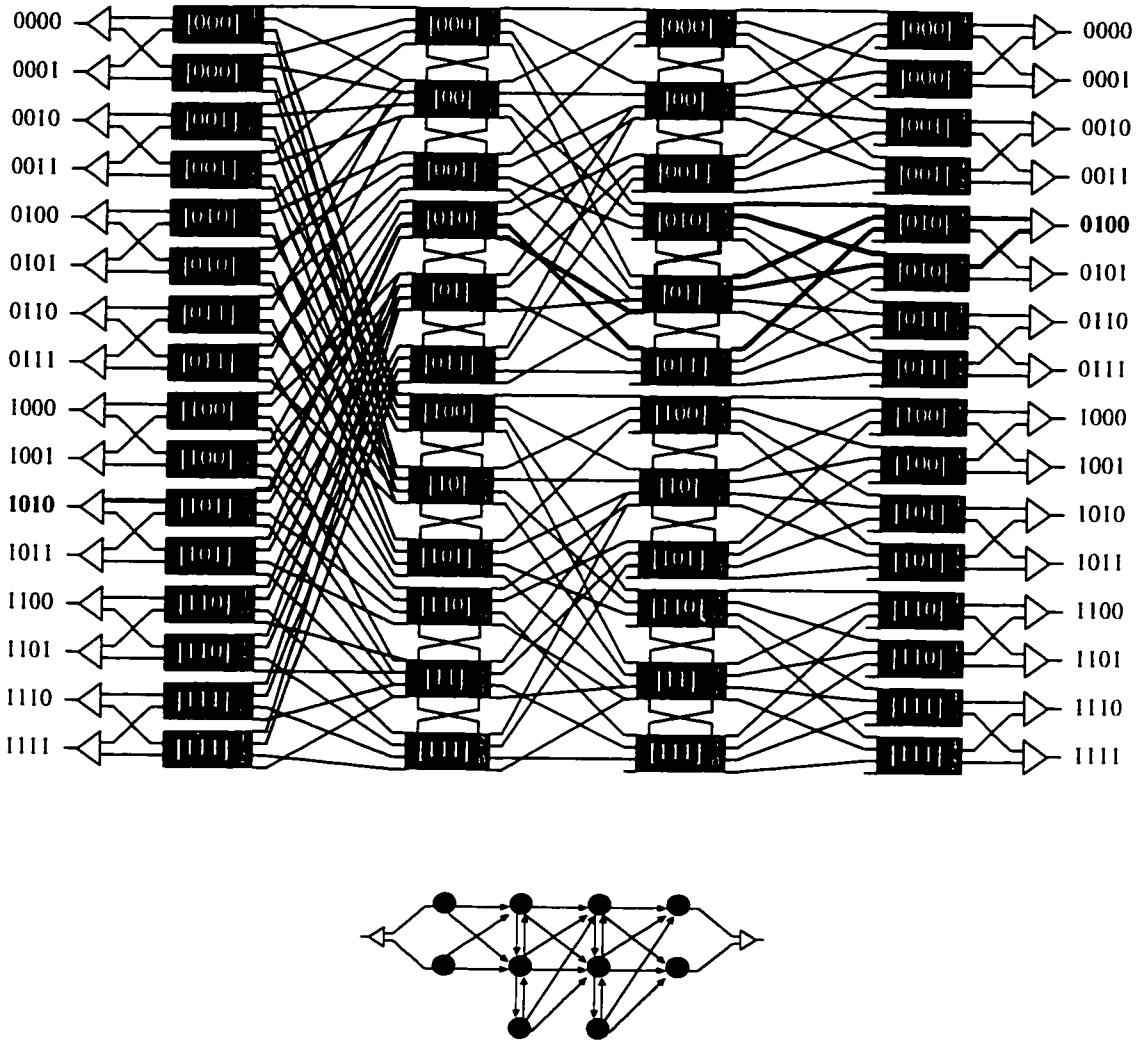


Figure 2.47 A 16 X 16 LIN and WANG's banyan network.

In the presence of faulty switching elements, the switching fabric performance decreases very slowly as the network size increases. The main problem with this switching fabric is the cells out of sequence problem when the standby links are used. This problem also introduces high jitter. This switching fabric is not modular and the interconnection does not follow a regular pattern throughout the stages of the switch.

Reliable And Zealous Network (RAZAN)

This architecture is based on the Improved Logical Neighborhood network as shown in Figure 2.48 [70]. An N X N switching network consists of n stages where $n = \log_2 N$. Each stage consists of N (n+1 X n+1) switching elements. Every switching element is connected to n+1 neighboring switching elements that their binary addresses is the same or differ by at most one bit in the next stage. The routing algorithm forwards the cell to the neighboring switch whose binary address has the minimum hamming distances from the destination's binary address. Once a hamming distance of zero is achieved, the cell is routed to the switch with the same binary address in the next stage. The switching fabric has n+1 disjoint paths. Hence, RAZAN can tolerate n faults on the path between a source and a destination. There are more than (n+1)! redundant paths which make the switching fabric very reliable. However, The routing algorithm is very complex compared to simple banyan routing algorithm.

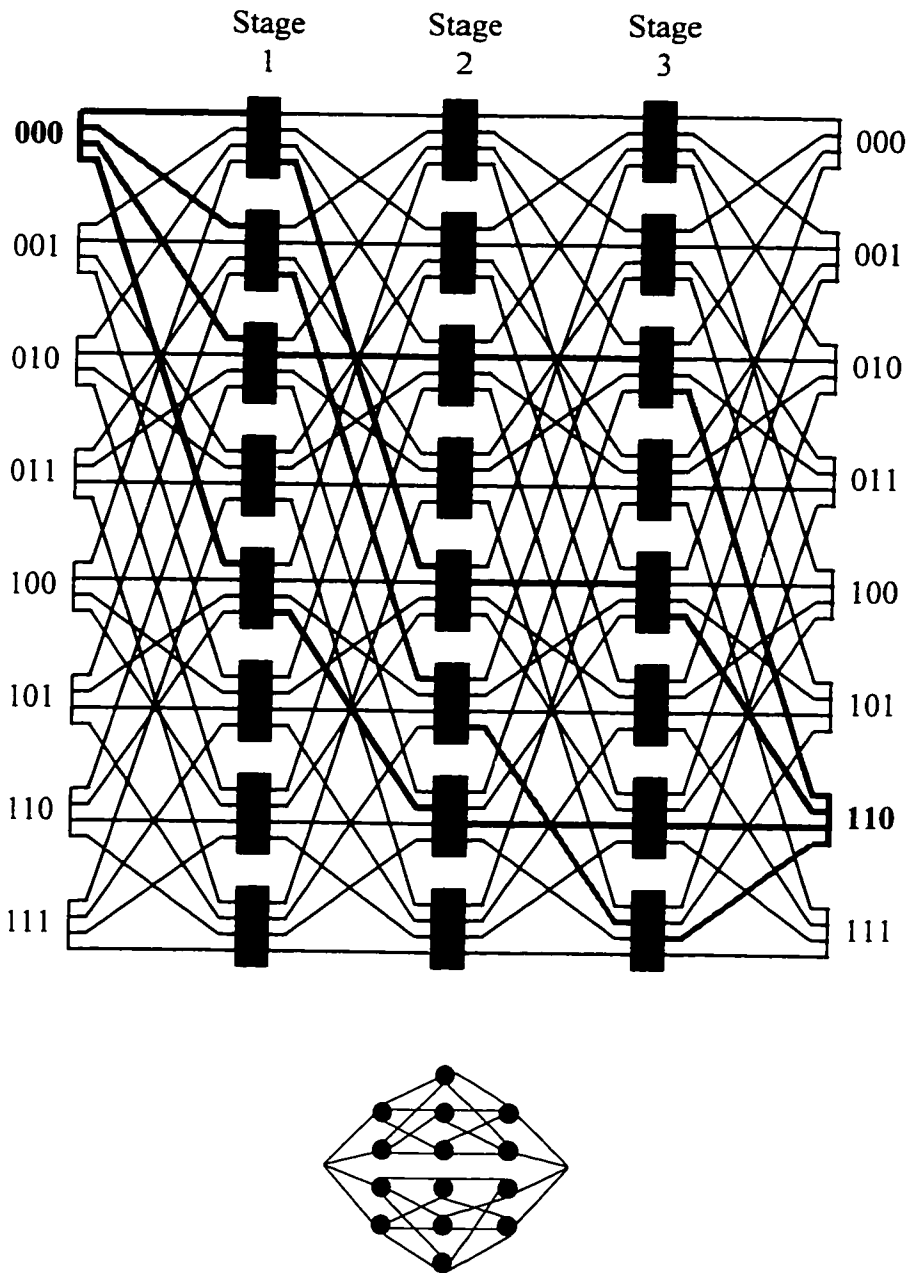


Figure 2.48 An 8 X 8 RAZAN network.

In addition, the switching fabric is not easily scalable since the switching element changes with the network size. The complexity of the switching element increases as the network size increases and the required time variation to select a link for routing a cell increases leading to switch synchronization problem. The channel graph of RAZAN is shown in the bottom of Figure 2.48.

Chapter 3

PROPOSED SWITCHING FABRIC ARCHITECTURE

3.1 Design Evolution

The proposed switching fabric architecture has evolved through several phases starting from the basic Shuffle-Exchange Network (SEN) shown in Figure 3.1. N X N SEN consists of n stages where $n = \log_2 N$. Each stage consists of *N/2* 2 X 2 switching elements (SE). The cell-switching algorithm is distributed throughout the switching fabric in each SE. Each switch has only two states: cross and bar. The following is a description of the routing algorithm in each SE.

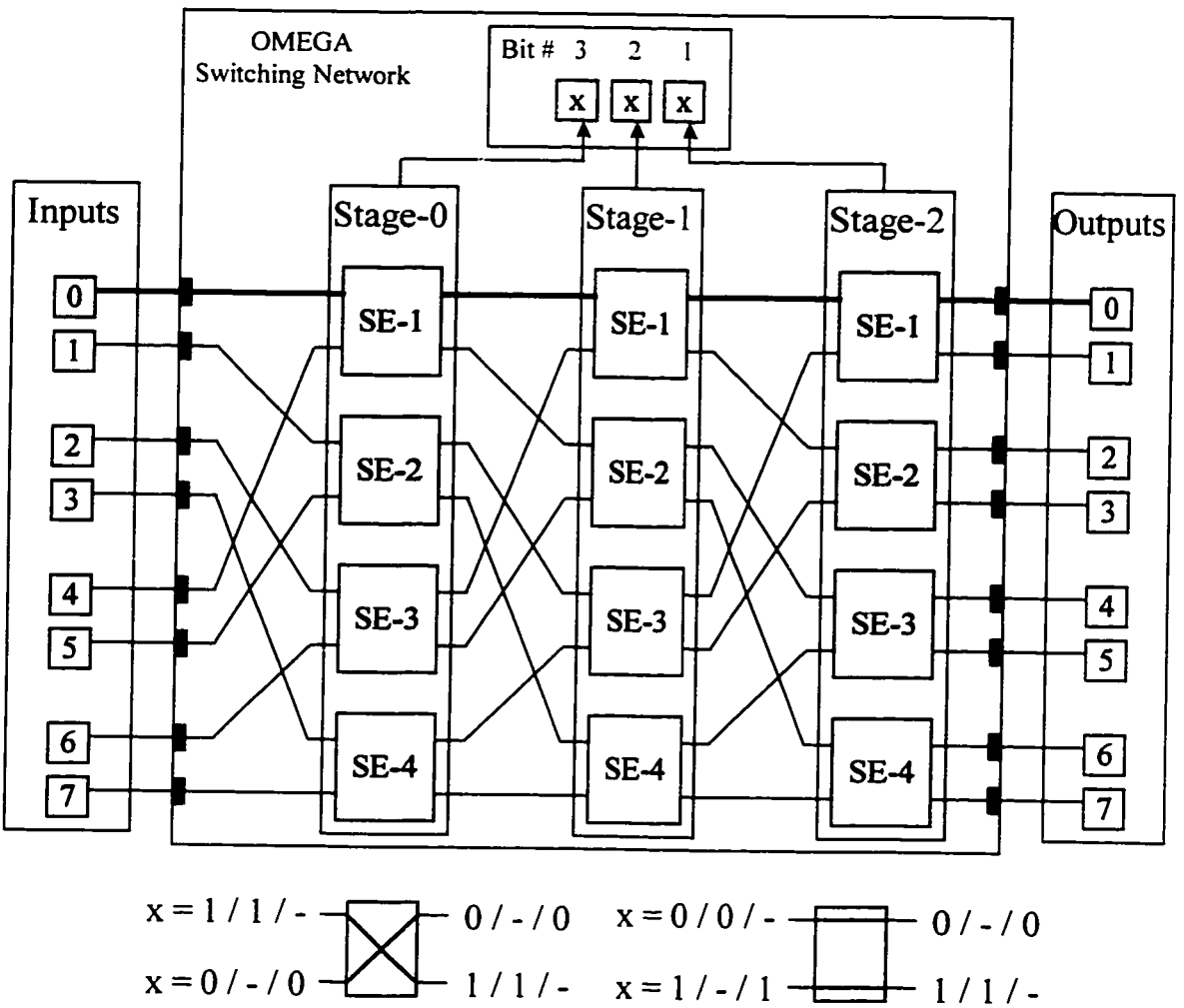


Figure 3.1 8 X 8 shuffle-exchange (Omega) network.

The SE routing algorithm of stage-0 depends on the most significant bit (routing bit n) of the binary representation of the cell destination address. The routing bit of stage-1 is n-1; stage-2 is n-2, etc. The routing bit of stage-(n-1) is the least significant bit (routing bit 1) of the binary representation of the cell destination address as shown in the top of Figure 3.1. The input-output routing path of a cell consists of n SE. Each SE in a given stage checks its corresponding routing bit of both inputs and changes its state accordingly. Each SE has two states: cross and bar. If the routing bit of the input cell were equal to 0, the cell would be routed through the top output link. Otherwise, it would be routed through the bottom output link. There are six nonblocking routing-bit patterns shown in the bottom of Figure 3.1 where "–" means there is no cell available at the input link. When both routing bits at the input links are the same, the SE randomly selects one cell for routing and discards the other cell. This is called internal blocking problem. Figure 3.1 shows the cell routing path (thick links) from input port 0 to output port 0. Although SEN is a low performance and not fault tolerant switch architecture, it has low transmission latency, delivers cells in sequence, and its architecture is suitable for VLSI implementation. J. H. Patel has derived the following recurrence relation to calculate the throughput performance for non buffered banyan networks under uniformly distributed and independent random input traffic:

$p_i = 1 - \left(1 - \frac{p_{i-1}}{2}\right)^2$, where $1 \le i \le n$, $p_0 = 1$ (full load cell arrival rate), and p_i is the probability that an output link of stage-(i-1) carries a cell [52]. The throughput is

$TP = p_n$. This formula is developed for 2 X 2 switching elements. This analytical result is quite useful to validate the experimental design simulation software that is developed to estimate the throughput performance of SEN. In many simulation studies a great deal of time is spent on model development and programming, but little effort is made to analyze the simulation output data appropriately. A very common mode of operation is to make a single simulation run of somewhat arbitrary length and then to treat the resulting simulation estimates as the "true" model characteristics. Since random samples from probability distributions are just particular realizations of random variables that may have large variances. As a result, these estimates could differ greatly from the corresponding true characteristics for the model. The major effect is that there could be a significant probability of making erroneous inferences about the switching fabric under study. Therefore, appropriate statistical techniques must be used to design and analyze the simulation experiments [4].

Let $Y_{11}, Y_{12}, ..., Y_{1m}$ be the performance measure of a switching fabric resulting from making a simulation run of length m observations using the random numbers $u_{11}, u_{12}, ..., u_{1m}$. Assume u_{ji} as a set of N independent and uniformly distributed random numbers representing the cell destination addresses of the *i*th observation in the *j*th simulation run. By running the simulation with a different set of random numbers $u_{21}, u_{22}, ..., u_{2m}$, a different realization $Y_{21}, Y_{22}, ..., Y_{2m}$ of the switching fabric performance measure will be obtained. The two realizations are not the same since the

different random numbers used in the two runs produce different samples from the input probability distributions. Suppose that one makes n independent replications (simulation runs) of length m, assuming that different random numbers are used for each replication and each replication uses the same initial conditions, resulting in the observations:

$$\begin{matrix} Y_{11}, & Y_{12}, & \cdots, & Y_{1m} \\ Y_{21}, & Y_{22}, & \cdots, & Y_{2m} \\ \vdots & \vdots & & \vdots \\ Y_{n1}, & Y_{n2}, & \cdots & Y_{nm} \end{matrix}$$

The observations from a particular replication (row) are clearly not identical and identically distributed (IID). However, $Y_{1i}, Y_{2i}, \ldots, Y_{ni}$ from the ith column are IID observations of the random variables Y_i, for $i = 1,2,\ldots, m$. This independence across simulation runs provides an unbiased estimate of the mean or the expected value, $E(Y_i)$, of the random variables Y_i, for $i = 1,2,..,m$. The steady states mean reaches the true mean value as i approaches infinity ($v = \lim_{i \to \infty} E(Y_i) = \lim_{m \to \infty} \frac{\sum_{i=1}^{m} Y_i}{m}$).

The experimental design is based on a statistical analysis of steady-state parameters for stochastic processes (switching fabric model). The *replication/deletion approach* is used for obtaining a point estimate of the throughput performance of a given switching fabric model. Let $Y_1, Y_2, \ldots, Y_n$ be the average throughput performance for the switching fabric model from n independent replications. Each simulation run has

fixed-sample-size observations (for instance, processing a full load of N cells through the switching fabric 1000,000/N times). Then, the average throughput performance is equal to $\frac{\sum_{i=1}^{n} Y_i}{n}$. This average value approaches the true mean value as n approaches infinity. Minimum of ten replications is recommended for good approximation of the mean.

All experimental designs are based on the algorithm shown in Figure 3.2. The pseudo random number generator used in the experimental design is the Prime Modulus Multiplicative Linear Congruential Generator (PMMLCG) which is provided and thoroughly tested in [4]. This algorithm is applied to the Shuffle-Exchange switch architecture to estimate the steady-state mean of the throughput performance for the network size of $n = 2$ to $n = 10$. Ten replications are used in the simulation algorithm. Of course, the more replications performed, the closer the measured performance to the true value is. Each replication uses different stream for the pseudo random number generator. The algorithm basically loads the switching fabric with at least 1000,000 cells in each replication. The input cells are simplified to represent the destination address only. Then, those cells are processed through the switching fabric model and monitored at the output ports. The cell loss and the throughput performance are calculated at the end of the switching fabric process.

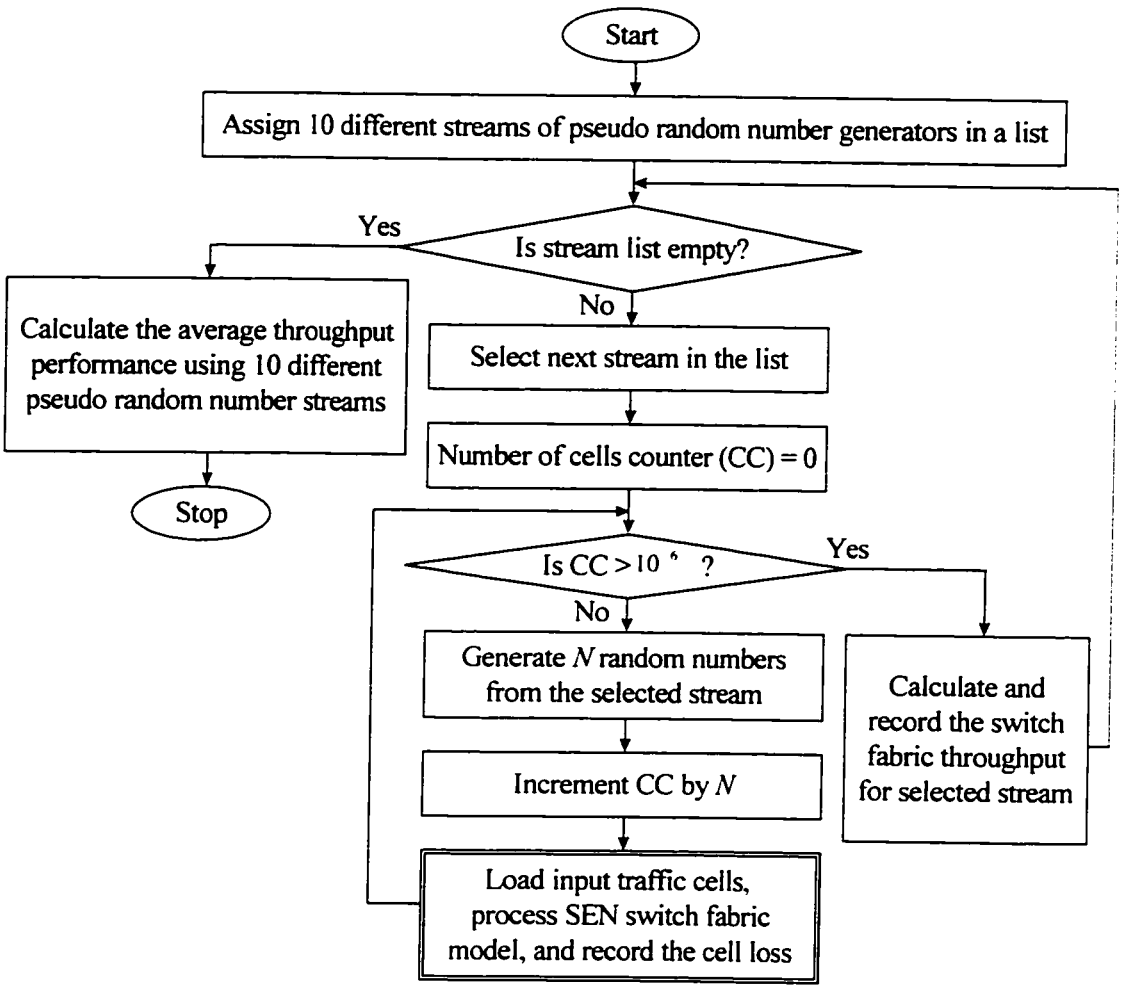


Figure 3.2 Experimental design simulation model for SEN.

This process is repeated ten times using different pseudo random number streams. The overall throughput is given by the average throughput over the ten replications.

The results of the experimental design for simulating the SEN is very close to the analytical results generated by J. H. Patel's recurrence relation as shown in Table 3.1 and Figure 3.3. This result validates the experimental design and the uniform input traffic source that will be used as the basis for the remaining experiments. The technique of using a common input traffic for all switching fabrics is called Common Random Number Variance-Reduction Technique. This technique is used to compare the alternative switching fabrics under similar experimental conditions so that one can be more confident that any observed differences in performance are due to differences in the switching fabrics rather than to fluctuations of the experimental conditions [4]. SEN has low performance due to the internal blocking and output contention problems. Most of the solutions proposed to solve the internal blocking problem of a simple banyan network, for instance SEN, generate side effects that create other problems such as increasing the switching fabric transmission latency, out-of-sequence cells delivery, high jitter and HOL problems. One method not considered before is to divide the input traffic into two parallel banyan networks. This method resolves more than 50% of the internal blocking problem. This new method is experimented in phase-1 with slight architectural modification to decrease transmission latency, to improve switching element fault tolerance, and to minimize the output contention problem as well as to resolving the internal blocking problem.

Table 3.1 SEN throughput performance comparison between analytical formula and experimental design simulation.

Network Size	SEN Formula	SEN Simulation
n=2	0.609375	0.609289
n=3	0.516541	0.516663
n=4	0.449837	0.450106
n=5	0.399249	0.399354
n=6	0.359399	0.359485
n=7	0.327107	0.327175
n=8	0.300357	0.300290
n=9	0.277804	0.277872
n=10	0.258510	0.258535

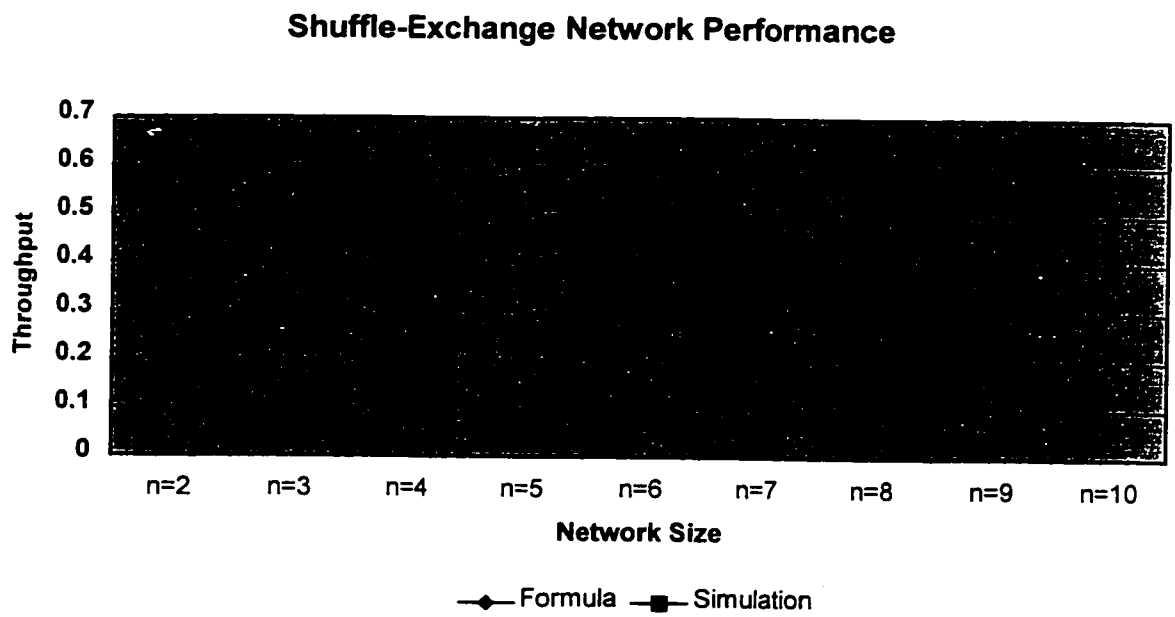


Figure 3.3 SEN throughput performance comparison.

Figure 3.4 shows a block diagram of an 8 X 8 network. The main idea is to put two banyan networks in parallel. The input traffic is balanced by dividing the interconnection of the input ports into two sections: Odd (i1, i3, ..., iN-1) and Even (i0, i2, ..., iN-2). Each section is connected to one banyan network. The first stage of both banyan networks is eliminated and the first stage routing is performed at the input ports. Hence there are two links for each input port: f0 and f1. If the routing bit is 0, f0 is selected to route the cell; otherwise f1 is selected. The routing in the remaining stages is the same as the routing in SEN. Two experimental designs are simulated for the parallel banyan network shown in Figure 2.43 and for phase-1 switching network using the same algorithm used to simulate SEN that is shown in Figure 3.2. Only the double border box is affected by replacing the SEN simulation model with phase-1 simulation model and parallel banyan simulation model. The input traffic used for simulation is identical in both experiments. About 62% increase in the throughput is gained as shown in Table 3.2 and Figure 3.5, and the transmission latency is decreased by the elimination of the first stage. To compare the switching fabrics, one assumption is made. It is assumed that all switching elements used in the switching fabrics are crossbar. The number of the cross-point switches is used as the complexity measurement in addition to the number of interconnection links used in the switching fabric. Table 3.3 and Figure 3.6 show that phase-1 has less cross-point switches than the parallel banyan network.

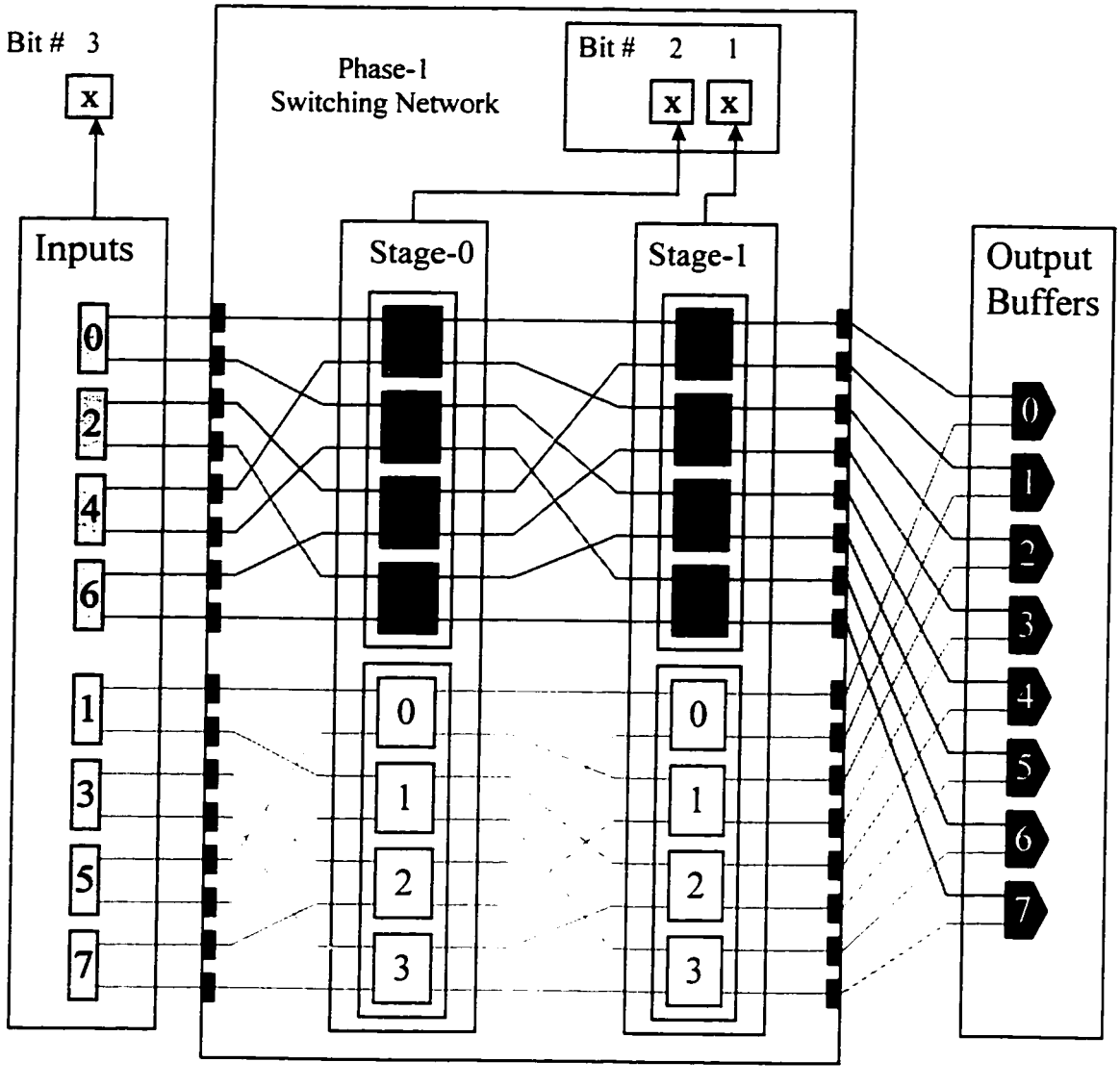


Figure 3.4 Phase-1: 8 X 8 switching network.

Table 3.2 Throughput performance comparison between Phase-1 and parallel banyan networks.

Network Size	Parallel Banyan Simulation	Phase-1 Simulation
n=2	0.609289	0.874695
n=3	0.516663	0.779580
n=4	0.450106	0.703105
n=5	0.399354	0.641245
n=6	0.359485	0.590352
n=7	0.327175	0.546532
n=8	0.300290	0.509266
n=9	0.277872	0.476885
n=10	0.258535	0.448427

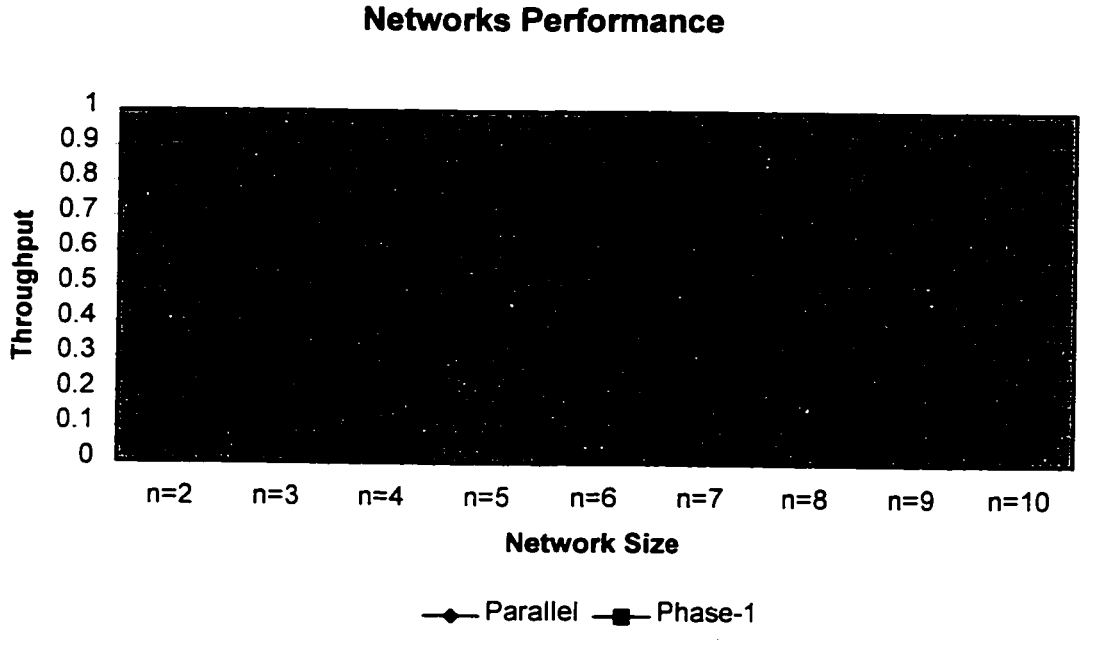


Figure 3.5 Throughput performance of phase-1 and parallel banyan networks.

Table 3.3 Cross-point switches comparison for Phase-1 and parallel banyan networks.

Network Size	Parallel Banyan	Phase-1
n=2	32	16
n=3	96	64
n=4	256	192
n=5	640	512
n=6	1536	1280
n=7	3584	3072
n=8	8192	7168
n=9	18432	16384
n=10	40960	36864

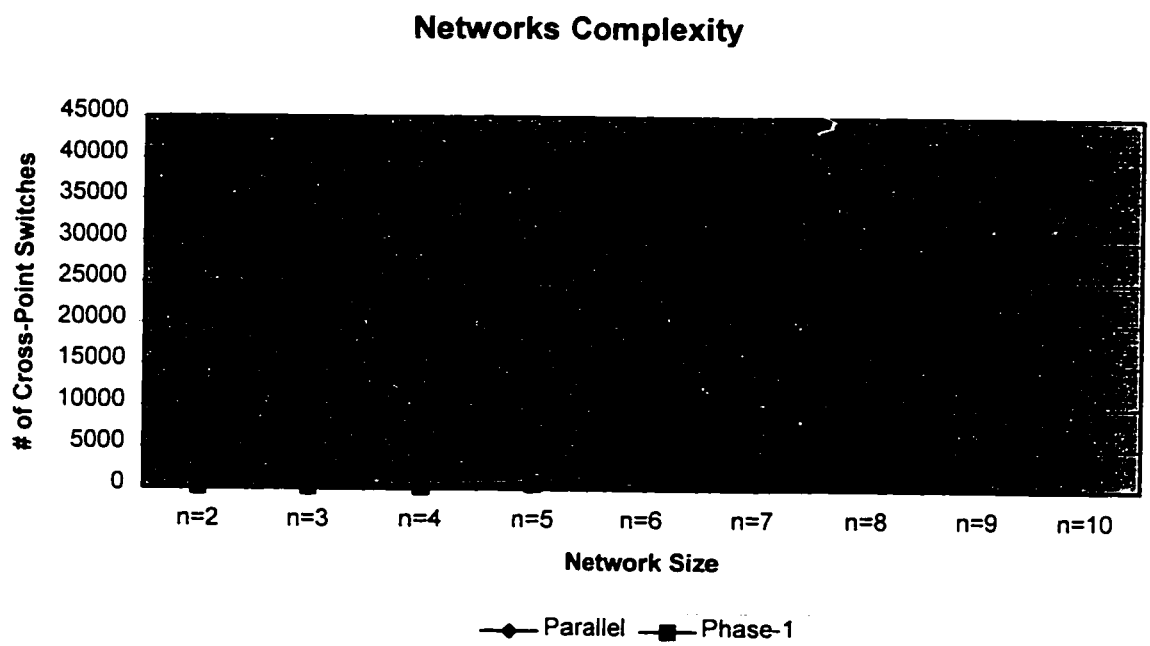


Figure 3.6 Cross-point switches complexity of phase-1 and parallel banyan networks.

The difference increases as the switching network increases. The interconnection links complexity of phase-1 switching network is also less than that of the parallel banyan networks as shown in Table 3.4 and Figure 3.7.

It is noticed in Figure 3.5 that the throughput performance decreases as the network size increases. The main reason is the increase of number of stages that increases the distance to the output ports. As the number of switching stage increases, the probability of internal blocking increases. In order to improve the throughput performance, a new modification to phase-1 is required to resolve the internal blocking problem within each banyan network.

The second phase is to improve the throughput performance and the fault tolerance of the phase-1 switching fabric by interconnecting the two parallel banyan networks as shown in Figure 3.8. The input port has four access links to the network as shown in Figure 3.9. One pair of links for each banyan network. The formal links are used to connect the input port to the switching elements in one banyan network and the redundant links are used to connect the input port to the switching elements in the other banyan network. This means if there is a faulty switching element in the first stage of a banyan network, the input port routes the cells to the other banyan network using the other pair of links. The switching fabric uses 4 X 4 switching elements as shown in Figure 3.10.

Table 3.4 Interconnection links comparison for phase-1 and parallel banyan networks.

Network Size	Parallel Banyan	Phase-1
n=2	32	16
n=3	80	48
n=4	192	128
n=5	448	320
n=6	1024	768
n=7	2304	1792
n=8	5120	4096
n=9	11264	9216
n=10	24576	20480

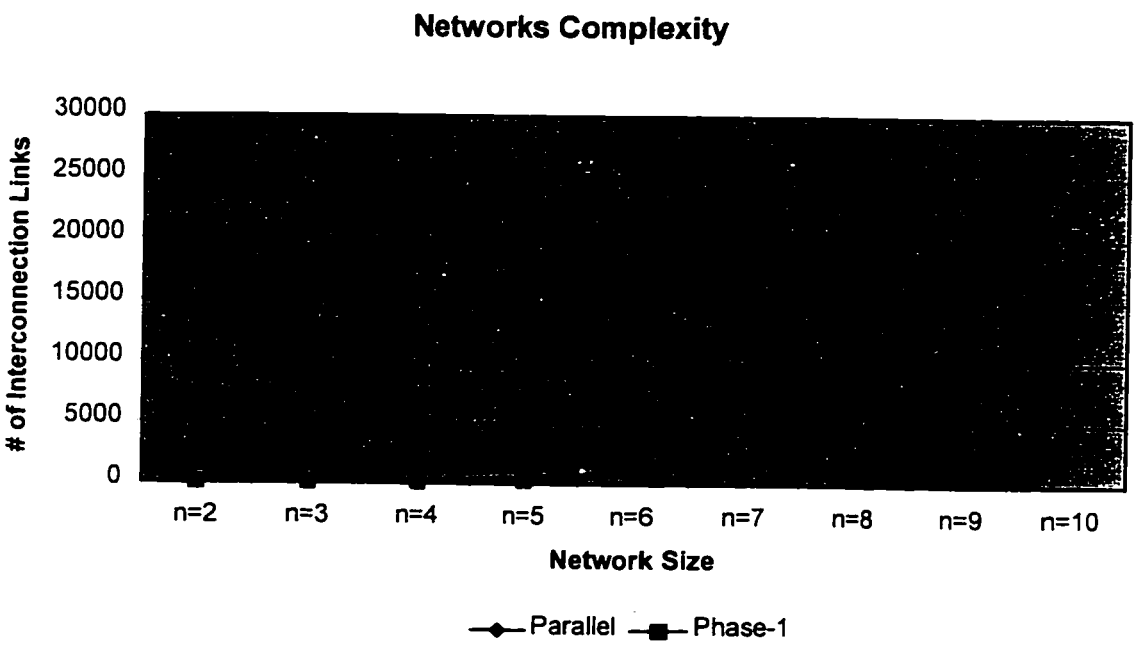


Figure 3.7 Interconnection links complexity of phase-1 and parallel banyan networks.

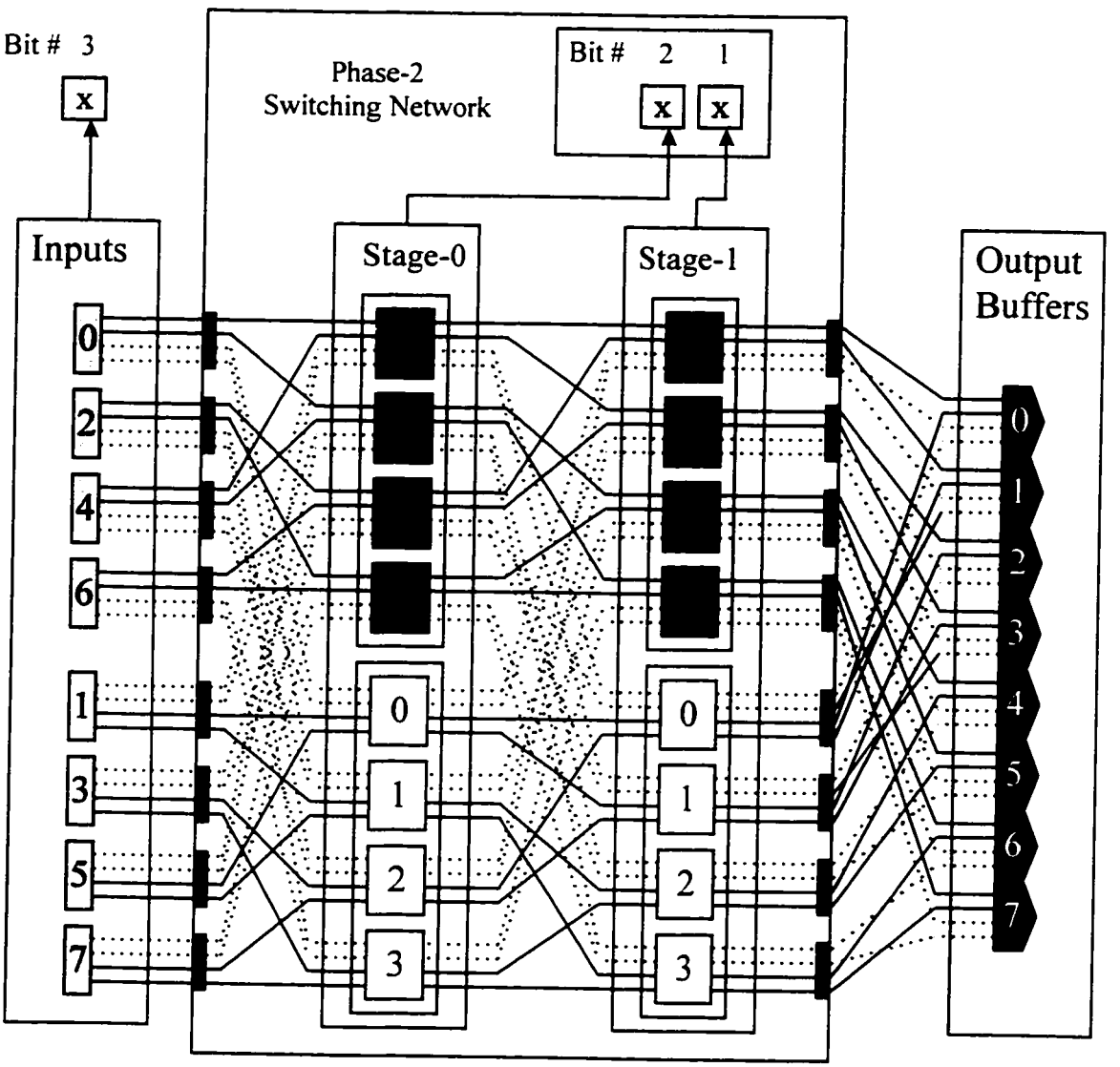


Figure 3.8 Phase-2: 8 X 8 switching network.

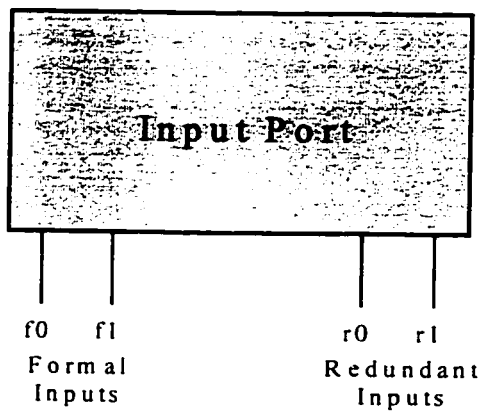


Figure 3.9 Input port links.

Figure 3.10 Phase-2: 4X4 switching element.

The switching element in one stage is linked with two switching elements in the next stage of the same banyan network using formal links and two switching elements in the next stage of the other banyan network using redundant links. The cells coming from the main input links have higher priority for routing than the cells coming from the secondary input links. Normally when there is no cell contention and the switches connected to the formal output links are not faulty, the formal output links are used for routing; otherwise, the redundant output links are used. If both the formal and the redundant links are not useable for routing, the cells are discarded. There are four access links available for each output port to minimize the output contention problem. The routing algorithm is the same as that of first phase. Two experimental designs are simulated for the Tagle and Sharma's network shown in Figure 2.44 and for phase-2 switching network using the same algorithm shown in Figure 3.2. About 17% increase in the throughput is gained as shown in Table 3.5 and Figure 3.11. Table 3.6 and Figure 3.12 show that phase-2 has less cross-point switches than Tagle and Sharma's network. The interconnection links complexity of phase-2 switching network is also less than that of Tagle and Sharma's networks as shown in Table 3.7 and Figure 3.13.

The main drawback of phase-2 switching network is that the number of access links to each output port is fixed for all network sizes. Therefore, the output contention problem increases as the network size increases.

The third phase is to resolve both the internal blocking and output contention problems simultaneously.

Table 3.5 Throughput performance comparison between phase-2 and Tagle & Sharma's networks.

Network Size	Tagle&Sharma Simulation	Phase-2 Simulation
n=2	0.843777	1.000000
n=3	0.845606	0.993409
n=4	0.839863	0.984516
n=5	0.833505	0.975376
n=6	0.826193	0.966053
n=7	0.819917	0.956829
n=8	0.813185	0.947851
n=9	0.806736	0.939150
n=10	0.800824	0.930637

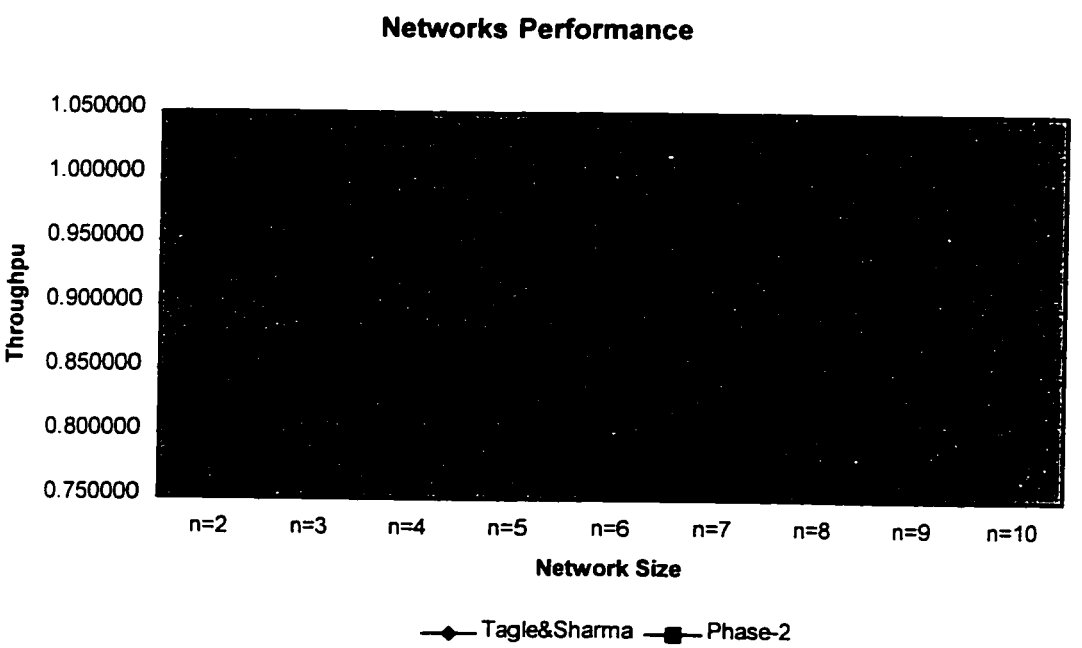


Figure 3.11 Throughput performance of phase-2 and Tagle & Sharma's networks.

Table 3.6 Cross-point switches complexity comparison between phase-2 and Tagle & Sharma's networks.

Network Size	Tagle&Sharma	Phase-2
n=2	128	64
n=3	384	256
n=4	1024	768
n=5	2560	2048
n=6	6144	5120
n=7	14336	12288
n=8	32768	28672
n=9	73728	65536
n=10	163840	147456

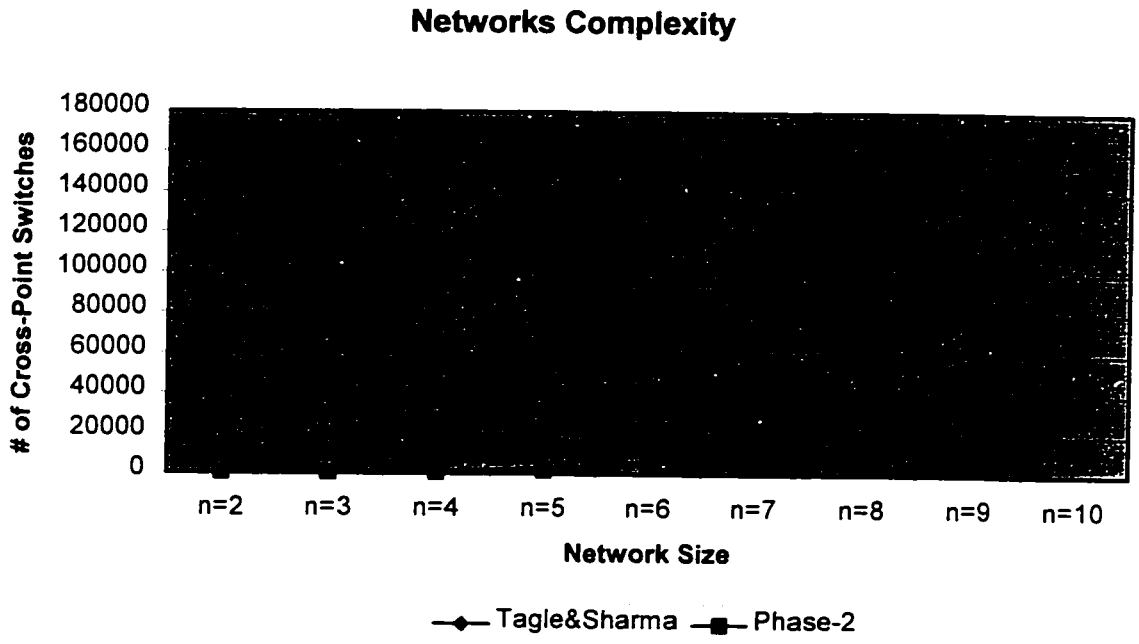


Figure 3.12 Cross-point switches complexity of phase-2 and Tagle & Sharma's networks.

Table 3.7 Interconnection links complexity comparison between phase-2 and Tagle & Sharma's networks.

Network Size	Tagle&Sharma	Phase-2
n=2	40	32
n=3	112	96
n=4	288	256
n=5	704	640
n=6	1664	1536
n=7	3840	3584
n=8	8704	8192
n=9	19456	18432
n=10	43008	40960

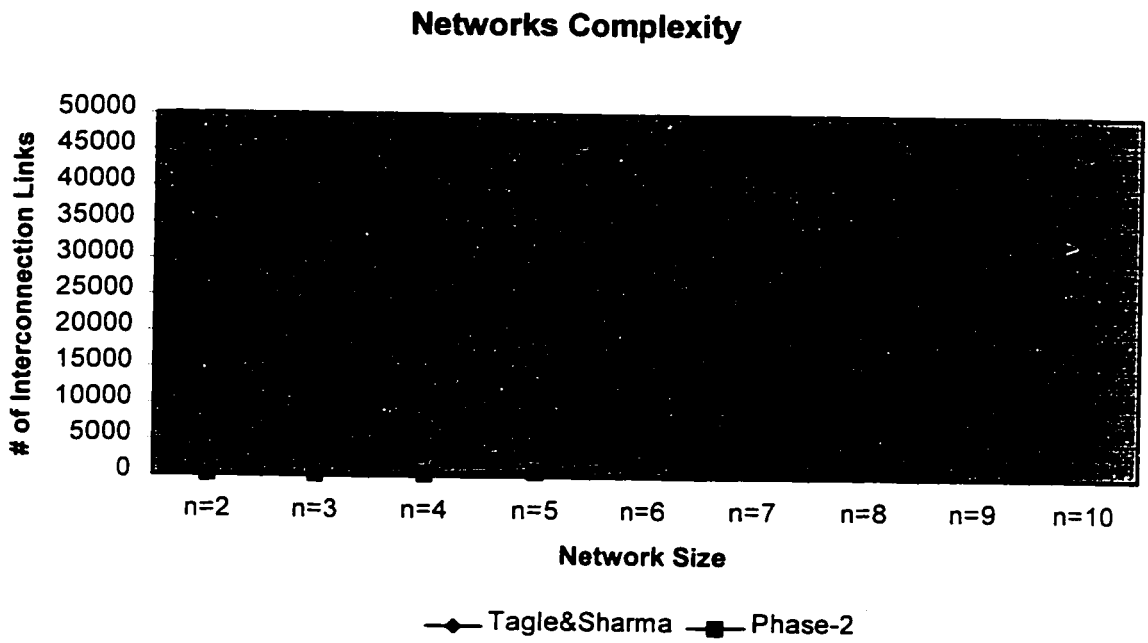


Figure 3.13 Interconnection links complexity of phase-2 and Tagle and Sharma's networks.

Therefore, this modification would improve the throughput performance and the fault tolerance of the switching fabric of phase-1 with a new method to increase the output access links to each output port linearly as the network size increases. The modification is to interconnect the two parallel banyan networks using additional expansion stages interconnected in a binary tree form as shown in Figure 3.14. The output contention problem is resolved by providing *2n* output access links to each output port. This architecture is structured in a binary tree form where the input ports in the root. The two parallel banyan networks form the two expanding edges of the binary tree triangle. Both banyan networks start from the first level and extend to the last level where their switching elements are connected to the output ports. The binary tree expansion starts in the second level and continues to expand to the last level. The number of expansion stages in each level is one stage more than the number of expansion stages in the previous level. The tree level is considered as the switching stage for routing purposes. Appending an additional level to the binary tree would double the network size. The switching fabric uses 4 X 4 switching elements similar to those switching elements used in phase-2. The two parallel banyan networks are placed in the left and right edges of the binary tree. The switching elements of each network use the formal output links to connect to the next stage of the same banyan network in the next level. The redundant output links are used to connect to the closest expansion stage in the next level.

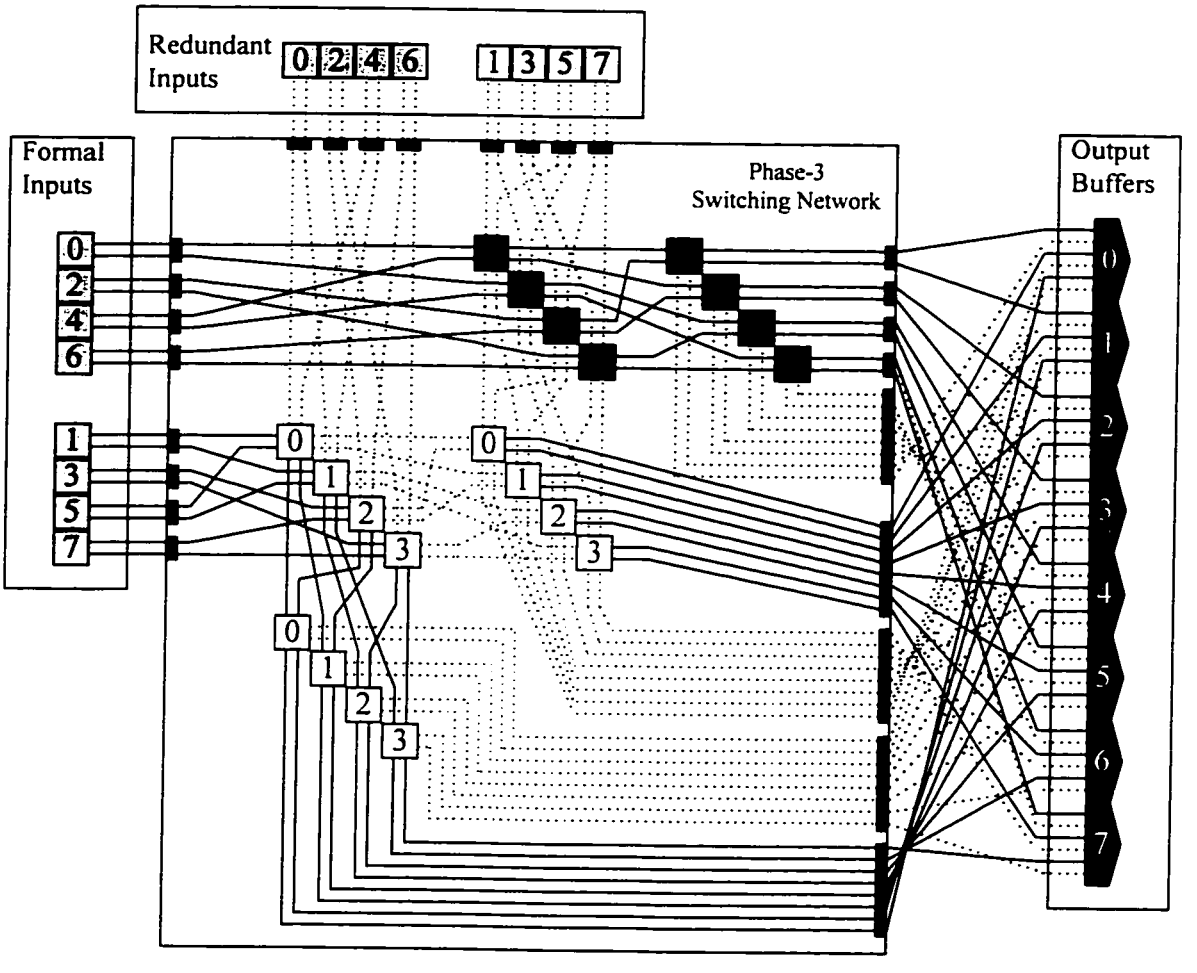


Figure 3.14 Phase-3: 8 X 8 switching network.

Each expansion stage uses the formal output links to connect to the right expansion stage in the next level, and the redundant output links to connect to the left expansion stage in the next level. The routing algorithm is the same as that of the second phase where the formal output links have higher priority. Normally all cells would be routed through the two parallel banyan networks. However, the expansion stages would be used when there is cell contention or faulty switching element in the parallel banyan networks. Notice that the switching elements of the two parallel banyan networks are nonblocking since each switching element receives only two cells and the other two input links are not used. On the other hand, the switching elements in the expansion stages are 50% blocking. This means maximum of two cells would be routed if the four input cells have the same routing bit. In other words, when there is cell contention, one cell gets routed to the left expansion stage and the other cell gets routed to the right expansion stage of the next level. Two experimental designs are simulated for the B-Tree (1) network shown in Figure 2.46 and for phase-3 switching network using the same algorithm shown in Figure 3.2. About 0.3% increase in the throughput is gained as shown in Table 3.8 and Figure 3.15. Table 3.9 and Figure 3.16 show that phase-3 has less cross-point switches than B-Tree (1) network. The interconnection links complexity of phase-3 switching network is also less than that of B-Tree (1) networks as shown in Table 3.10 and Figure 3.17.

Combining the modification done in phase-2 and phase-3 forms the proposed switching fabric architecture that is called the Binary Tree Banyan Network (BTBN).

Table 3.8 Throughput performance comparison between phase-3 and B-Tree (1) networks.

Network Size	B-Tree(1) Simulation	Phase-3 Simulation
n=2	1.000000	1.000000
n=3	0.996239	0.996707
n=4	0.992429	0.993720
n=5	0.989418	0.991362
n=6	0.986490	0.989063
n=7	0.984185	0.987013
n=8	0.982084	0.985143
n=9	0.980224	0.983527
n=10	0.978594	0.982069

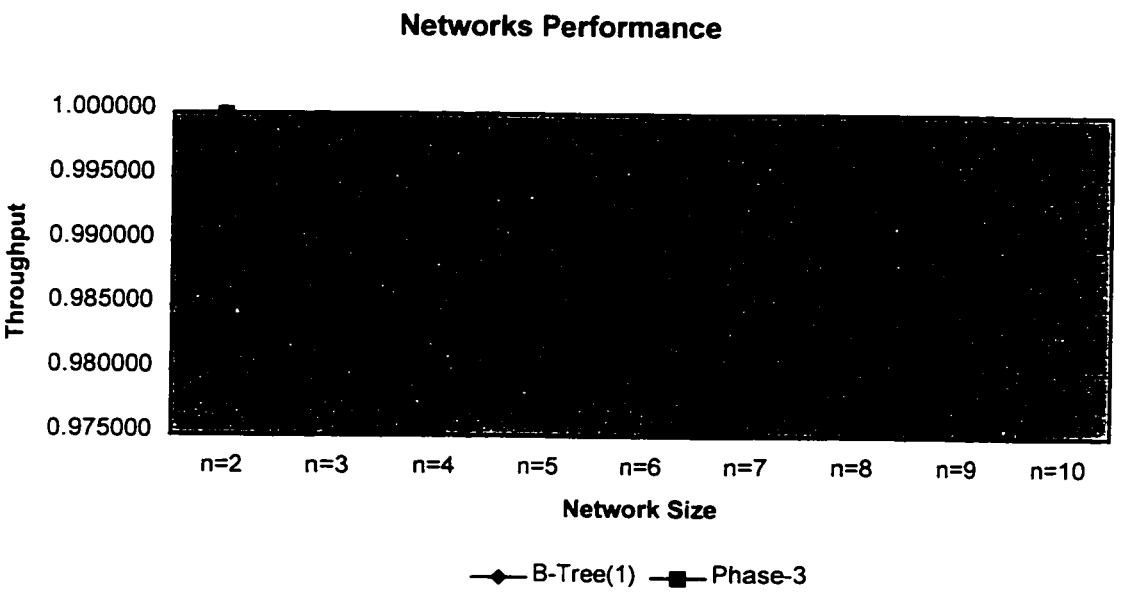


Figure 3.15 Throughput performance of phase-3 and B-Tree (1) networks.

Table 3.9 Cross-point switches complexity comparison between phase-3 and B-Tree (1) networks.

Network Size	B-Tree(1)	Phase-3
n=2	160	64
n=3	576	320
n=4	1792	1152
n=5	5120	3584
n=6	13824	10240
n=7	35840	27648
n=8	90112	71680
n=9	221184	180224
n=10	532480	442368

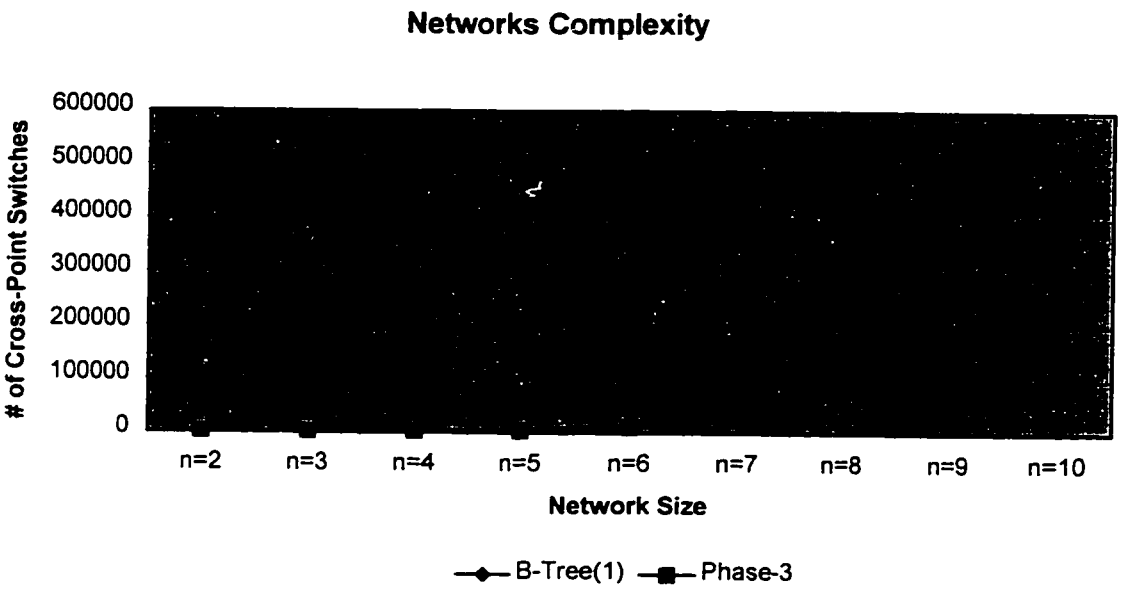


Figure 3.16 Cross-point switches complexity of phase-3 and B-Tree (1) networks.

Table 3.10 Interconnection links complexity comparison between phase-3 and B-Tree (1) networks.

Network Size	Phase-1	Phase-3
n=2	48	32
n=3	160	112
n=4	480	352
n=5	1344	1024
n=6	3584	2816
n=7	9216	7424
n=8	23040	18944
n=9	56320	47104
n=10	135168	114688

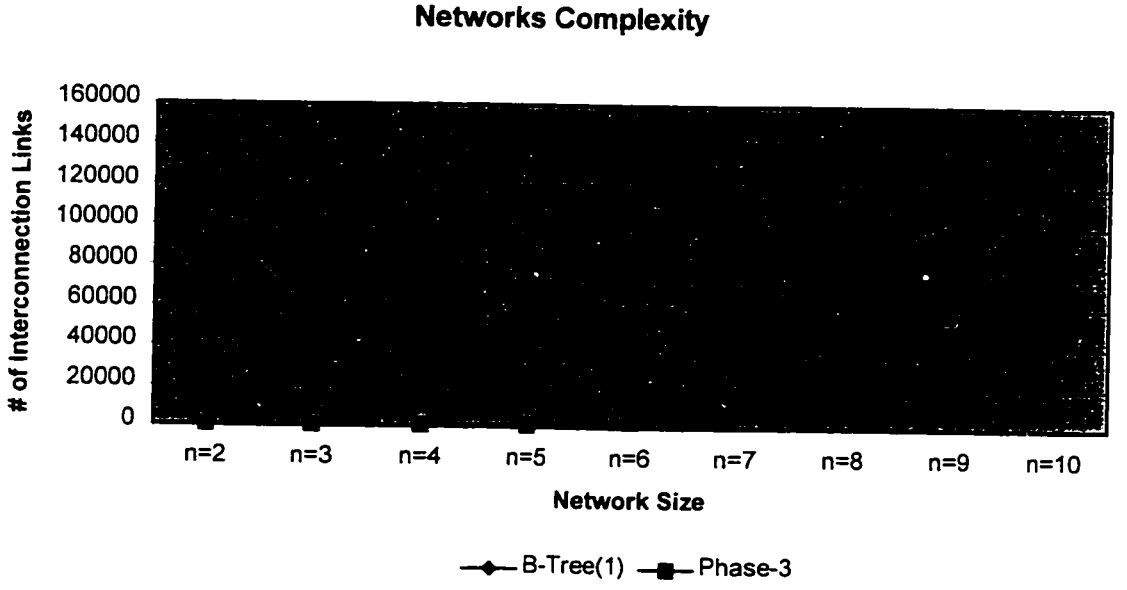


Figure 3.17 Interconnection links complexity of phase-3 and B-Tree (1) networks.

To evaluate the performance of the BTBN under normal conditions, an experimental design is simulated using the same input traffic offered to SEN. Comparing the results to those of B-Tree (1), about 2% increase in the throughput performance is gained as shown in Figure 3.18 and Table 3.11. It is clear that there is only little impact on the throughput performance as the network size increases. This is an excellent feature for large-size ATM networks.

In the following sections, BTBN will be described in details and compared to the parallel banyan, Tagle and Sharma's, and B-Tree (1) networks.

3.2 Binary Tree Banyan Network

BTBN is a high performance fault tolerant ATM switch architecture. It is carefully designed to achieve very low cell loss probability and high performance under normal conditions and in the presence of faulty switching elements.

3.2.1 BTBN Architecture

BTBN consists of two parallel banyan networks interconnected in a similar manner to the modification done in phase-2. In addition, they are also interconnected using the expansion stages, used in phase-3, in a binary tree form as shown in Figure 3.19.

Table 3.11 Throughput performance comparison between BTBN and B-Tree (1) networks.

Network Size	BTBN Simulation	B-Tree (1) Simulation
n=2	1.000000	1.000000
n=3	0.999917	0.996239
n=4	0.999735	0.992429
n=5	0.999579	0.989418
n=6	0.999358	0.986490
n=7	0.999166	0.984185
n=8	0.998943	0.982084
n=9	0.998759	0.980224
n=10	0.998587	0.978594

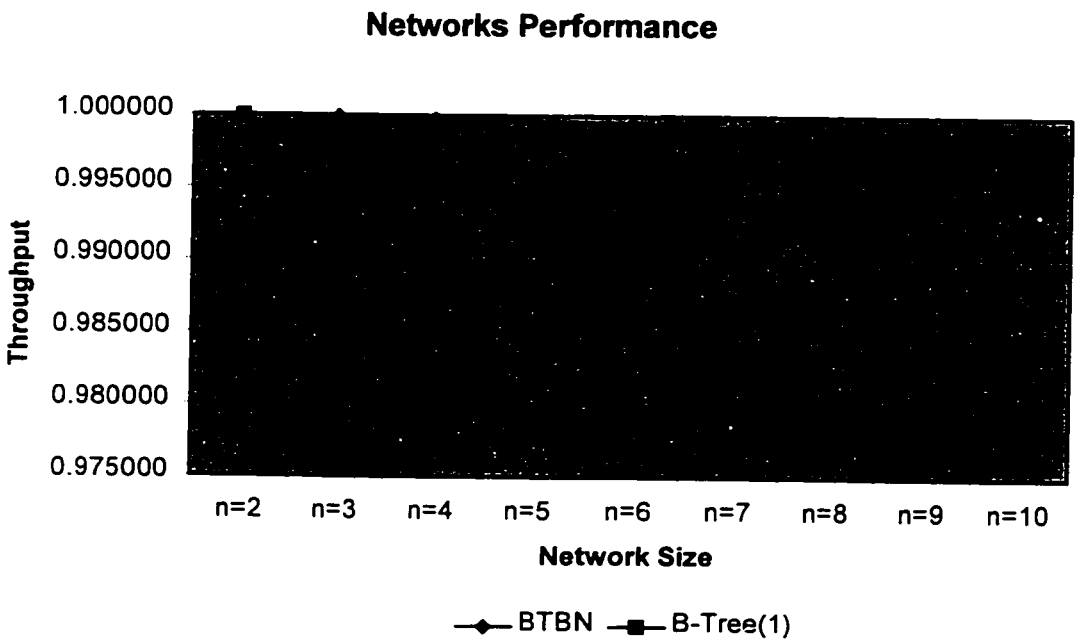


Figure 3.18 Throughput performance of BTBN and B-Tree (1) networks.

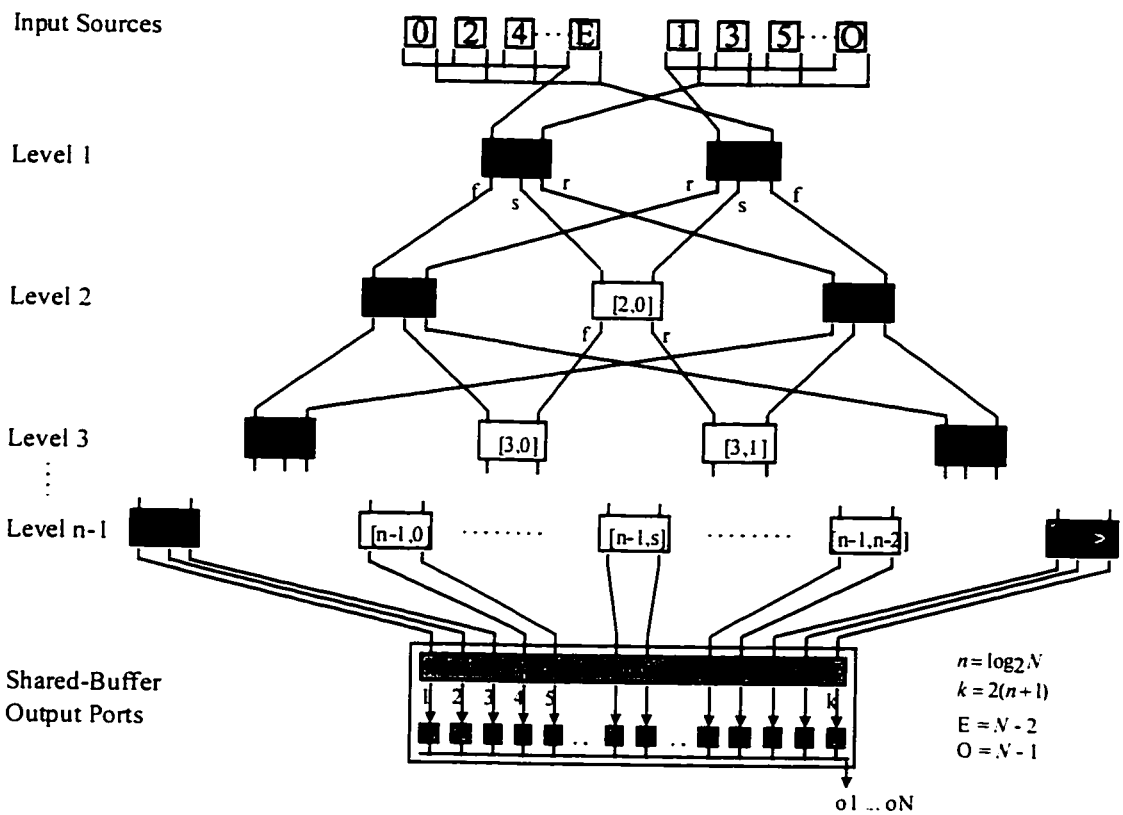


Figure 3.19 N X N BTBN.

Figure 3.20 shows the 4 X 6 switching element used in the parallel banyan networks and the 4 X 4 switching element used in the expansion stages. The 4 X 6 formal output links are used for connecting the parallel banyan network in the same way it is done in phase 2 and phase-3. The 4 X 6 redundant output links are used to interconnect the two parallel banyan networks in the same way it is done in phase-2. The 4 X 6 standby output links are used to connect to the expansion stages in the same way it is done in phase-3. The expansion stages use the same 4 X 4 switching elements used in phase-3. The solid box in Figure 3.19 represents a stage consisting of N/2 4 X 6 switching elements. The empty box represents an expansion stage consisting of N/2 4 X 4 switching elements.

N X N BTBN consists of n-1 levels where $n=\log_2 N$. Increasing the number of levels of the binary tree by one doubles the size of the switching network. Each level is considered as a switching stage. The last level is connected to *N* shared-buffer output ports. The interconnection links between stages represent MIN such as OMEGA, Delta, Baseline, and any topologically equivalent networks. In this thesis, the OMEGA (Shuffle-Exchange) network is selected for demonstration.

Notice that the 4 X 6 switching element shown in Figure 3.20 is 75% nonblocking since it can route three cells out of four cells that are destined to the same destination. The formal link is used when it is available. The redundant link is used if the formal link is busy or connected to a faulty switching element.

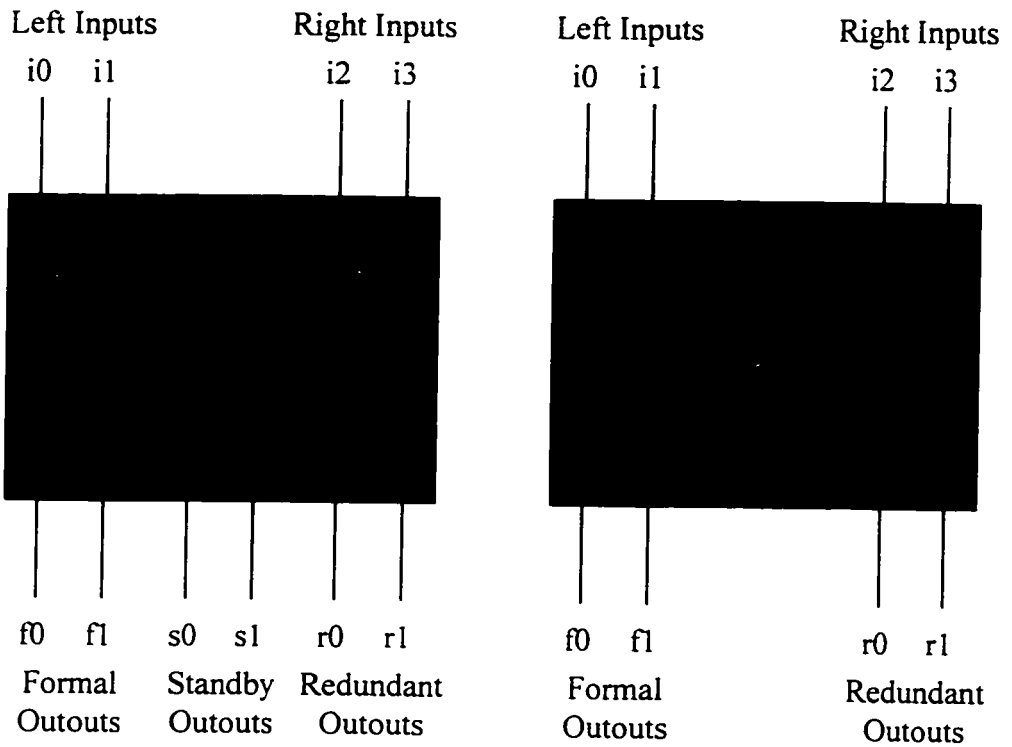


Figure 3.20 BTBN switching elements.

The standby link is used if both the formal and the redundant links are busy or connected to faulty switching elements. The 4 X 4 switching element shown in Figure 3.20 is 50% nonblocking since it can route two cells out of four cells that are destined to the same destination. The sequence in selecting the formal and redundant links is the same as in the 4 X 6 switching element. Figure 3.21 shows a 4 X 4 BTBN network. It consists of one level that consists of two 4 X 6 switching stages. Each switching stage consists of two 4 X 6 switching elements numbered as L/S/s where L indicates the level number, S indicates the stage number, and s indicates the switching element number. The two 4 X 6 switching stages are connected to four shared-buffer output ports. Each shared-buffer output port consists of a shifter and six queues.

Figure 3.22 shows an 8 X 8 BTBN network. It consists of two levels. The first level consists of two 4 X 6 switching stages. Each 4 X 6 stage consists of four 4 X 6 switching elements. The second level consists of two 4 X 6 switching stages and one 4 X 4 switching stage that consists of four 4 X 4 switching elements. The switching stages in the second level are connected to eight shared-buffer output ports. Each shared-buffer output port consists of a shifter and eight queues. Output contention is resolved by increasing the output access links to the shared buffer output ports as the network size increases. There are 2(n+1) access links for each shared buffer output port. Output queuing is proved to be more effective than input or internal queuing [39]. This fact is used in the BTBN where the shared buffer of each output port consists of a shifter and N buffers.

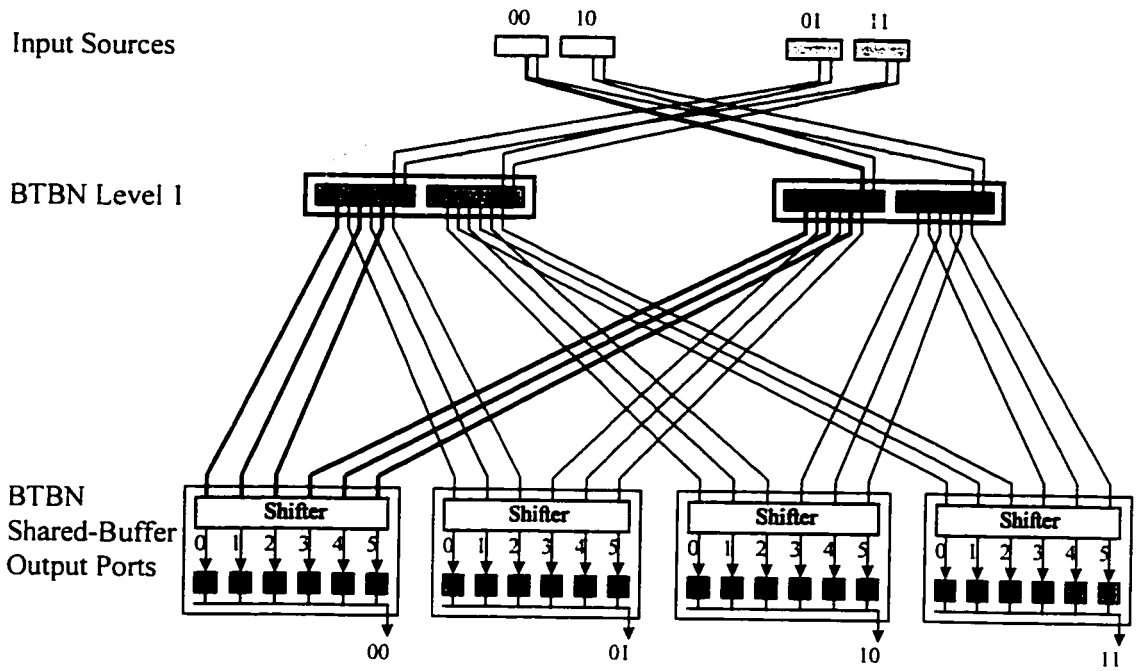


Figure 3.21 4 X 4 BTBN.

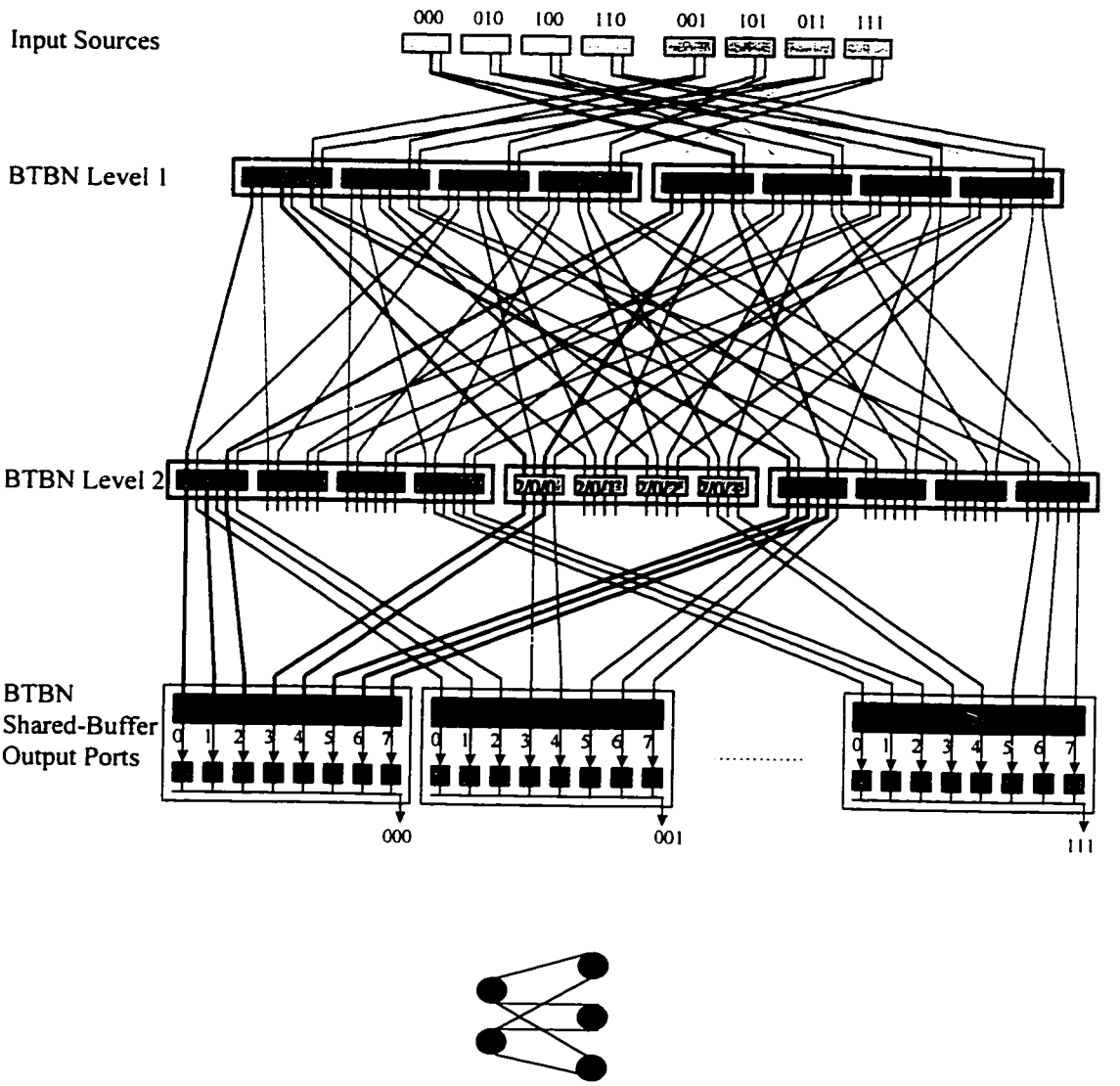


Figure 3.22 8 X 8 BTBN.

The shifter is used to balance the load on the buffers. The cell sequence is preserved and there is no jitter or HOL problems since the switching fabric uses neither input nor internal buffers.

3.2.2 BTBN Routing

The routing algorithm is the same as the basic routing algorithm used in SEN with minor modification regarding the resolution of internal congestion and tolerance of faulty switching elements. Figure 3.23 shows the general routing algorithm used by all switching elements. The routing of the 4 X 6 output links is shown in Figure 3.24 and Figure 3.25. The routing of the 4 X 4 output links is shown in Figure 3.26 and Figure 3.27. Figure 3.21 shows the possible routes to switch a cell from input port 0 to output port 0 in a 4 X 4 BTBN. There are six routes for routing a cell to output port 0 through the thick links. The following is the routing procedure:

- input port 0 checks the most significant bit (MSB) of the cell destination address label. Input f0 is selected since the MSB is 0. If the switching element connected to input f0 were faulty, input r0 would be selected.

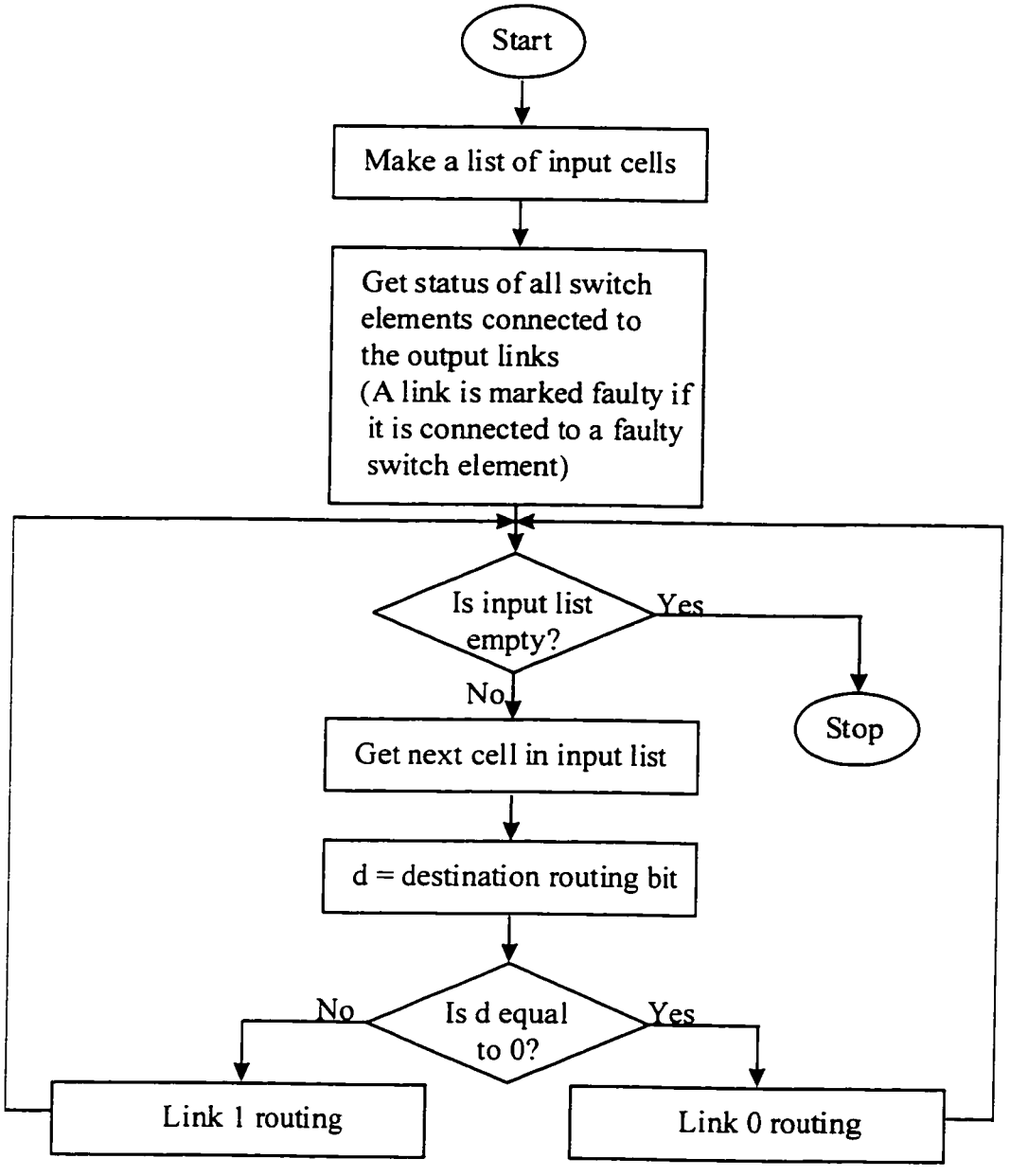


Figure 3.23 BTBN routing algorithm.

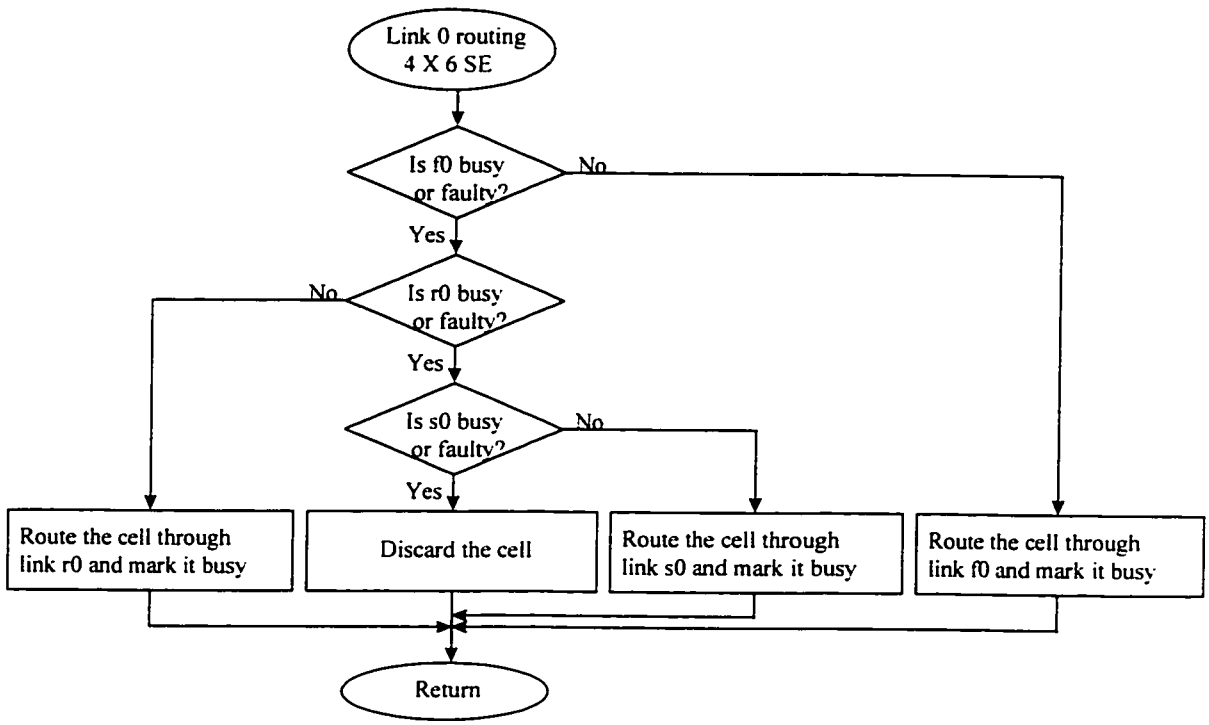


Figure 3.24 Output link 0 routing algorithm for 4 X 6 switching element.

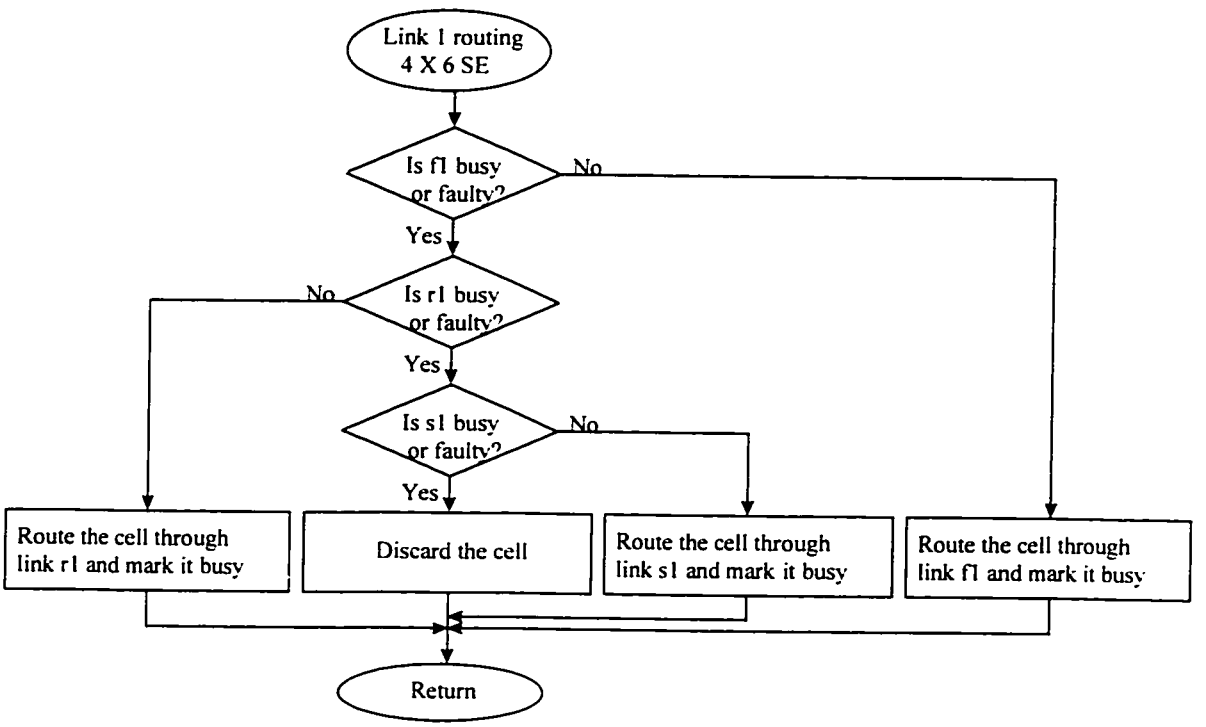


Figure 3.25 Output link 1 routing algorithm for 4 X 6 switching element.

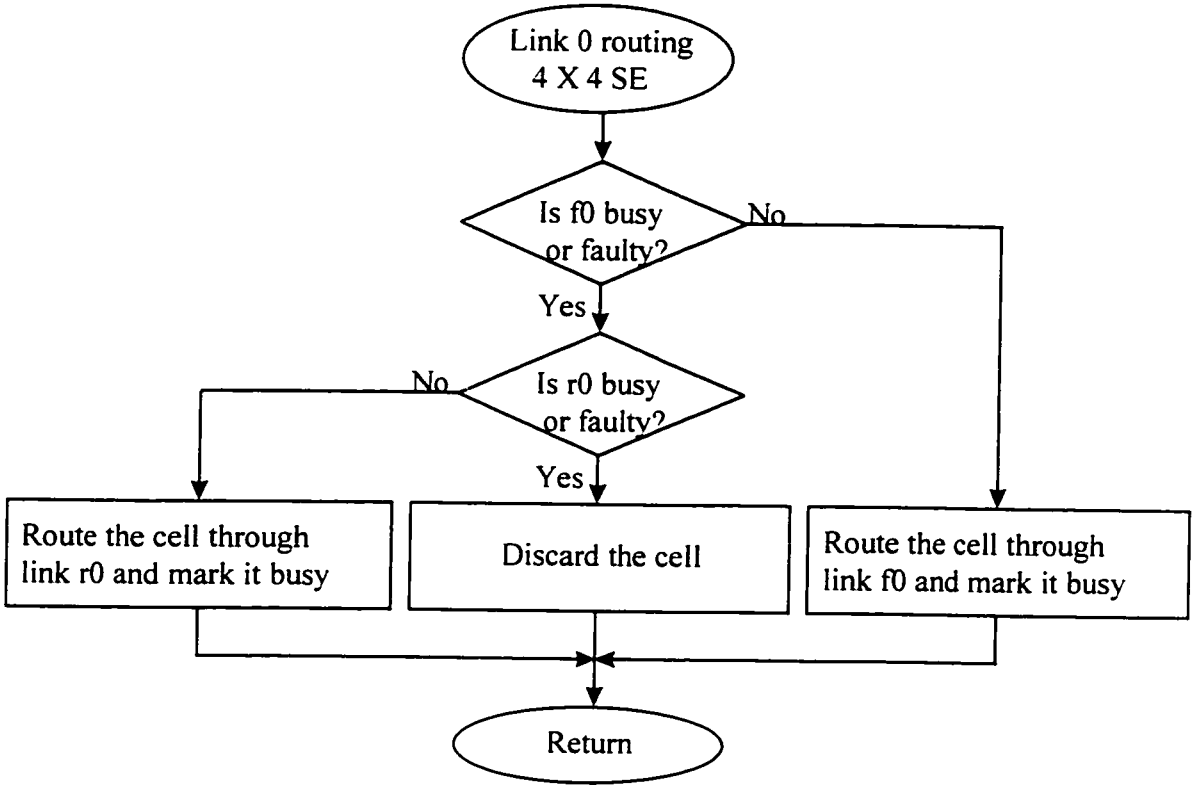


Figure 3.26 Output link 0 routing algorithm for 4 X 4 switching element.

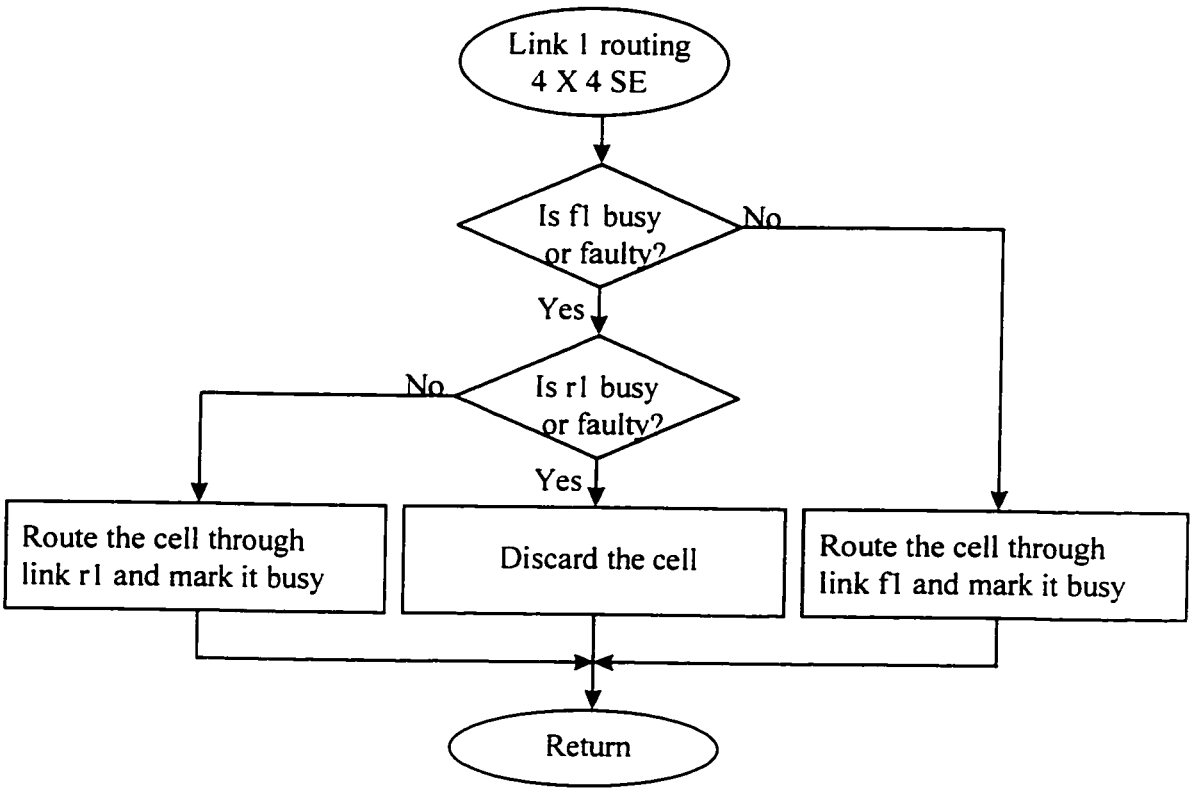


Figure 3.27 Output link l routing algorithm for 4 X 4 switching element.

- the cell is routed to switching element 1/0/0. The first bit of the cell destination address label is checked. Output f0 is selected since the first bit is 0. If the output link f0 were faulty, output link r0 would be selected. If the output link r0 were faulty, output link s0 would be selected.
- if the cell were routed to switching element 1/1/0 through input r0, the cell would be routed through output f0, r0 or s0.

Figure 3.22 shows the possible routes to switch a cell from input port 0 to output port 0 in an 8 X 8 BTBN. There are sixteen routes for routing the cell to output port 0 through the thick links. Table 3.12 lists the possible routes. In general, there are multiple redundant paths available to resolve internal congestion and to tolerate faults in the switching elements and links. The channel graph of an 8 X 8 BTBN is shown in the bottom of Figure 3.22.

3.2.3 BTBN Complexity

In the literature, there are two methods to compare the switching fabric complexity. The first method is to compare the required interconnection links and cross-point switches assuming that the switching elements are crossbar switches. The number of cross-point switches is calculated by multiplying the number of inputs by the number of outputs of each switching element. The second method is to compare the

contribution of the switching elements and the interconnection links. This comparison is calculated as follows. Let RP equal to the total redundant paths, SE equal to the total switching elements, and IL equal to the total interconnection links. The switching element contribution is equal to RP / SE, and the interconnection link contribution is equal to RP / IL.

The following formula shows the cross-point switches complexity of each switching fabric assuming $n=\log_2 N$:

BTBN (X-Point) = $24N(n-1) + 4N(n-1)(n-2)$

B-Tree (1)(X-Point) = $4N[(n+1)(n+2) -2]$

Tagle and Sharma (X-Point) = $16Nn$

Parallel (X-Point) = $4Nn$

Figure 3.28 and Table 3.13 show the cross-point complexity for the switching fabrics for the network size of n = 2 to n = 10. BTBN network has less cross-point switches than B-Tree (1) network. The following formula shows the interconnection links complexity of each switching fabric:

BTBN (Links) = $N[(n+4)(n-1)+4]$

B-Tree (1)(Links) = $N(n+1)(n+2)$

Tagle and Sharma (Links) = $N(4n+2)$

Parallel (Links) = $2N(n+2)$

Table 3.12 8 X 8 BTBN routes from input port 0 to output port 0.

Cell Destination Address Label (000)	Third Bit (MSB) 0	Second Bit 0	First Bit (LSB) 0
Route # 1	Input Port 0, f0	SE-4X6 1/0/0, f0	SE-4X6 2/0/0, f0
Route # 2	Input Port 0, f0	SE-4X6 1/0/0, f0	SE-4X6 2/0/0, r0
Route # 3	Input Port 0, f0	SE-4X6 1/0/0, f0	SE-4X6 2/0/0, s0
Route # 4	Input Port 0, f0	SE-4X6 1/0/0, r0	SE-4X6 2/1/0, f0
Route # 5	Input Port 0, f0	SE-4X6 1/0/0, r0	SE-4X6 2/1/0, r0
Route # 6	Input Port 0, f0	SE-4X6 1/0/0, r0	SE-4X6 2/1/0, s0
Route # 7	Input Port 0, f0	SE-4X6 1/0/0, s0	SE-4X4 2/0/0, f0
Route # 8	Input Port 0, f0	SE-4X6 1/0/0, s0	SE-4X4 2/0/0, r0
Route # 9	Input Port 0, r0	SE-4X6 1/1/0, f0	SE-4X6 2/1/0, f0
Route # 10	Input Port 0, r0	SE-4X6 1/1/0, f0	SE-4X6 2/1/0, r0
Route # 11	Input Port 0, r0	SE-4X6 1/1/0, f0	SE-4X6 2/1/0, s0
Route # 12	Input Port 0, r0	SE-4X6 1/1/0, r0	SE-4X6 2/0/0, f0
Route # 13	Input Port 0, r0	SE-4X6 1/1/0, r0	SE-4X6 2/0/0, r0
Route # 14	Input Port 0, r0	SE-4X6 1/1/0, r0	SE-4X6 2/0/0, s0
Route # 15	Input Port 0, r0	SE-4X6 1/1/0, s0	SE-4X4 2/0/0, f0
Route # 16	Input Port 0, r0	SE-4X6 1/1/0, s0	SE-4X4 2/0/0, r0

Table 3.13 Cross-point switches complexity of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	96	160	128	32
n=3	448	576	384	96
n=4	1536	1792	1024	256
n=5	4608	5120	2560	640
n=6	12800	13824	6144	1536
n=7	33792	35840	14336	3584
n=8	86016	90112	32768	8192
n=9	212992	221184	73728	18432
n=10	516096	532480	163840	40960

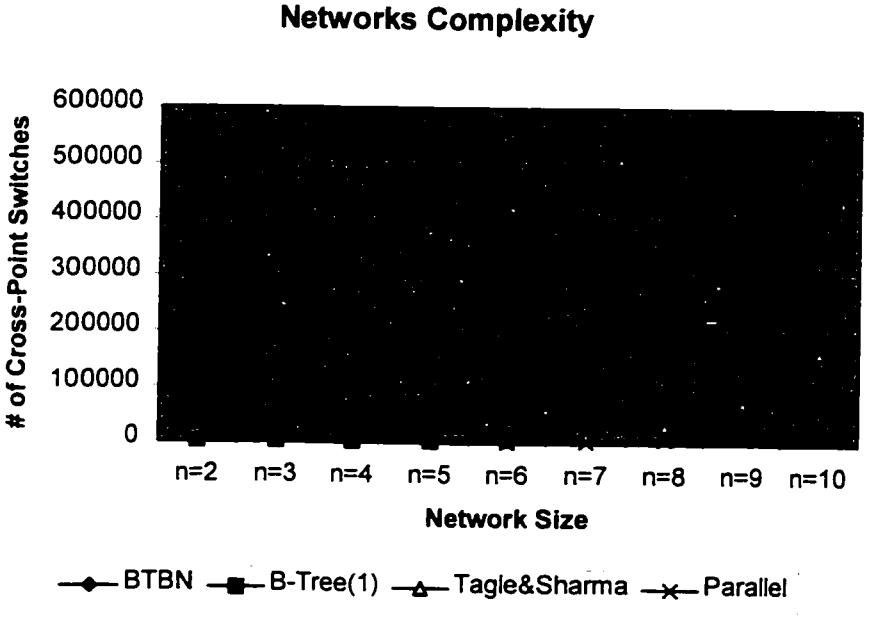


Figure 3.28 Cross-point switches complexity of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Figure 3.29 and Table 3.14 show the interconnection links complexity for the switching fabrics for the network size of n = 2 to n = 10. BTBN network has less interconnection links than B-Tree (1) network.

The second comparison method requires the number of switching elements and the number of redundant paths available in the switching fabrics. The following formula shows the number of switching elements in each switching fabric:

BTBN (SE) = N(n-1)(n+2)/4

B-Tree (1)(SE) = Nn(n+3)/4

Tagle and Sharma (SE) = Nn

Parallel (SE) = Nn

Figure 3.30 and Table 3.15 show the number of switching elements in the switching fabrics for the network size of n = 2 to n = 10. BTBN network has less switching elements than B-Tree (1) network. The following formula shows the number of redundant paths in each switching fabric:

BTBN (RP): $P_n = 2 * P_{n-1} + 2^{n-1}$ where $P_2 = 6$ and $n > 2$.

B-Tree (1)(RP) = 2^{n+1}

Tagle and Sharma (RP) = 2^n

Parallel (RP) = 2

Figure 3.31 and Table 3.16 show the number of redundant paths in the switching fabrics for the network size of n = 2 to n = 10.

Table 3.14 Interconnection links complexity of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	40	48	40	32
n=3	144	160	112	80
n=4	448	480	288	192
n=5	1280	1344	704	448
n=6	3456	3584	1664	1024
n=7	8960	9216	3840	2304
n=8	22528	23040	8704	5120
n=9	55296	56320	19456	11264
n=10	133120	135168	43008	24576

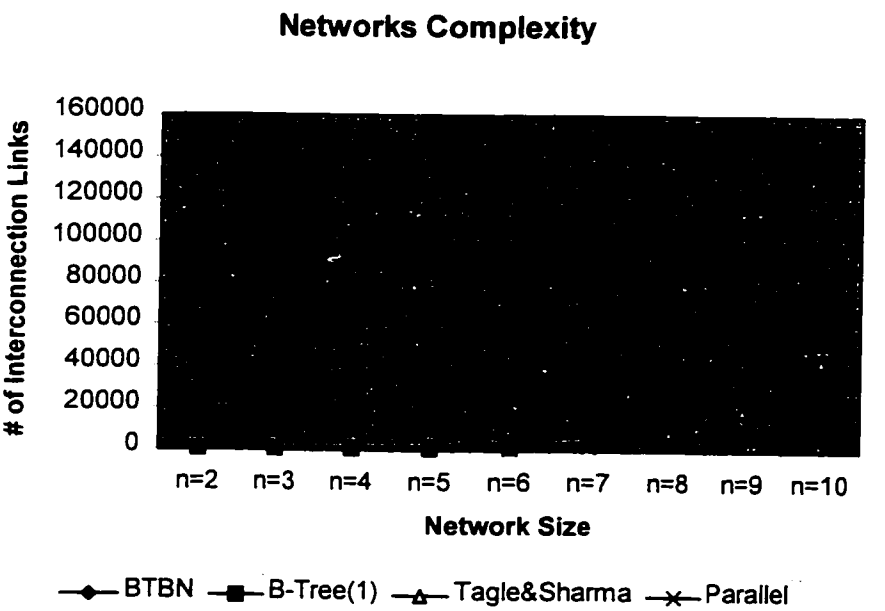


Figure 3.29 Interconnection links complexity of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Table 3.15 Switching network complexity of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	4	10	8	8
n=3	20	36	24	24
n=4	72	112	64	64
n=5	224	320	160	160
n=6	640	864	384	384
n=7	1728	2240	896	896
n=8	4480	5632	2048	2048
n=9	11264	13824	4608	4608
n=10	27648	33280	10240	10240

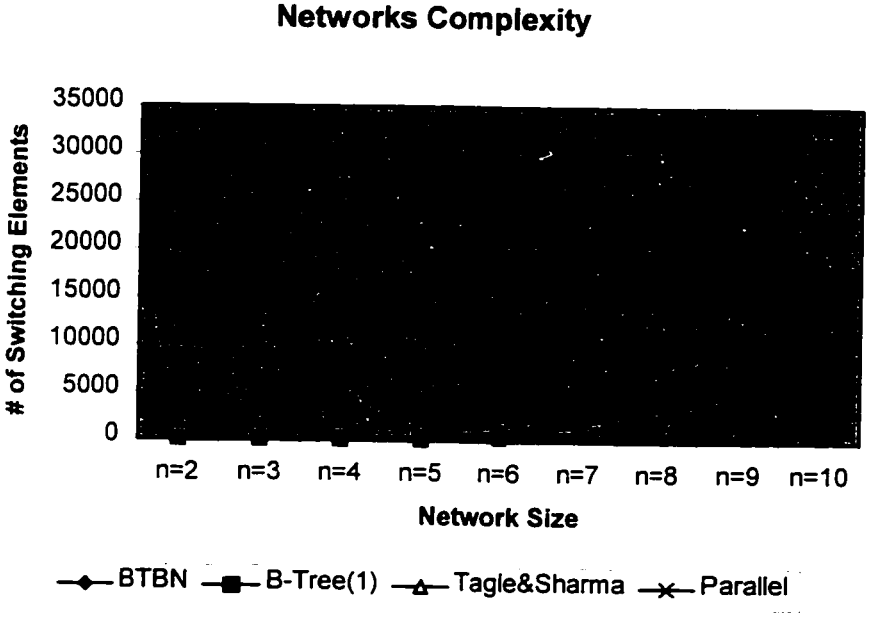


Figure 3.30 Switching network complexity of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Table 3.16 Redundant paths of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	6	8	4	2
n=3	16	16	8	2
n=4	40	32	16	2
n=5	96	64	32	2
n=6	224	128	64	2
n=7	512	256	128	2
n=8	1152	512	256	2
n=9	2560	1024	512	2
n=10	5632	2048	1024	2

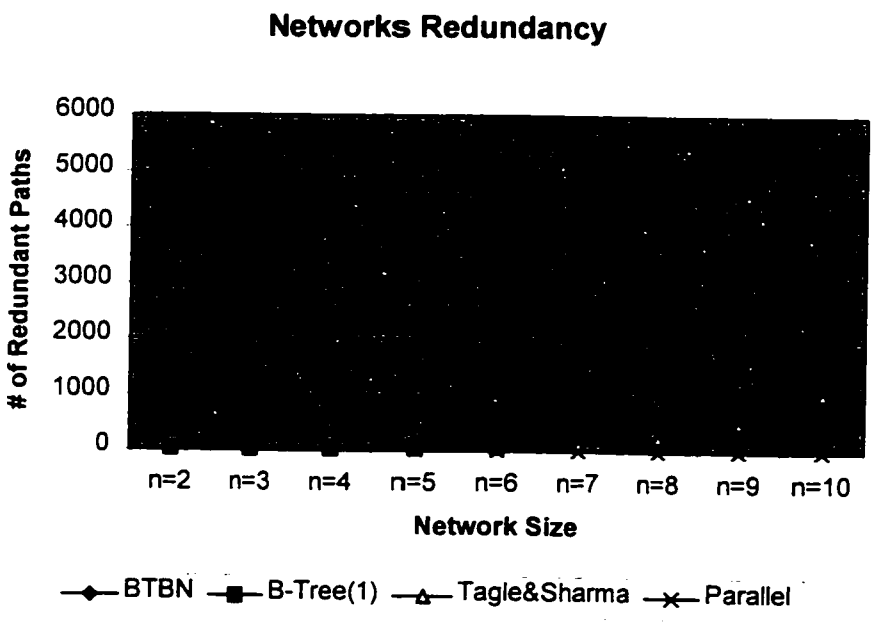


Figure 3.31 Redundant paths of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

BTBN network has the highest number of redundant paths. The number of redundant paths available in BTBN network of size n = 10 is almost three times more than the B-Tree (1) network. Figure 3.32 and Table 3.17 show the switching element contribution of the switching fabrics for the network size of n = 2 to n = 10. BTBN network has the highest switching element contribution. Figure 3.33 and Table 3.18 show the interconnection link contribution of the switching fabrics for the network size of n = 2 to n = 10. BTBN network has the highest interconnection link contribution.

3.2.4 BTBN Expandability

BTBN is modular and easily expandable. Figure 3.34 illustrates the method to build a 32 X 32 BTBN from a 4 X 4 BTBN. Put two 4 X 4 BTBN on top of each other. One 4 X 4 BTBN is connected to input ports 0, 1, 4, and 5. The other 4 X 4 BTBN is connected to input ports 2, 3, 6, and 7. An additional switching stage is added to interconnect both 4 X 4 BTBN to form an 8 X 8 BTBN. A copy of 8 X 8 BTBN is put on top of each other and an additional switching stage is added to interconnect both 8 X 8 BTBN to form a 32 X 32 BTBN. This feature is excellent for constructing large size networks.

Table 3.17 Switching element contribution of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	1.500000	0.800000	0.500000	0.250000
n=3	0.800000	0.444444	0.333333	0.083333
n=4	0.555556	0.285714	0.250000	0.031250
n=5	0.428571	0.200000	0.200000	0.012500
n=6	0.350000	0.148148	0.166667	0.005208
n=7	0.296296	0.114286	0.142857	0.002232
n=8	0.257143	0.090909	0.125000	0.000977
n=9	0.227273	0.074074	0.111111	0.000434
n=10	0.203704	0.061538	0.100000	0.000195

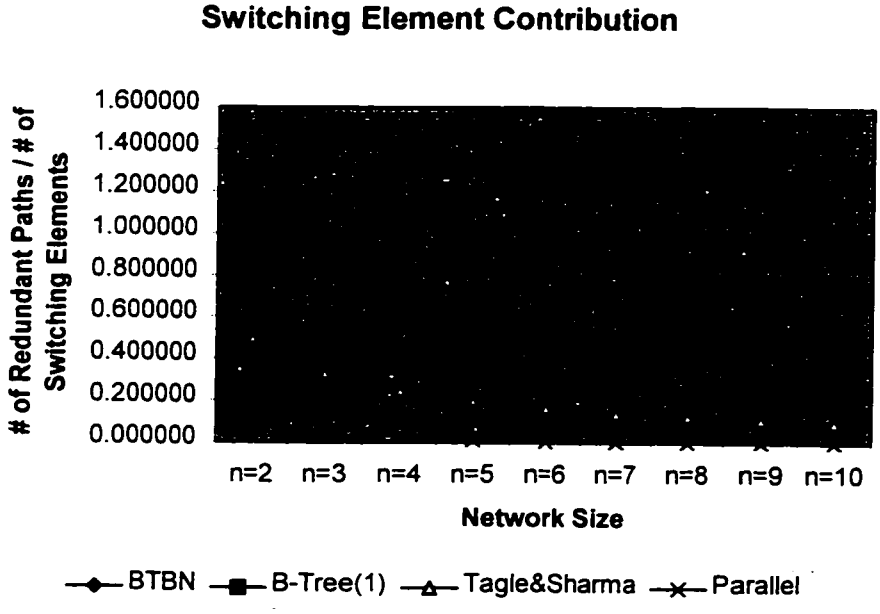


Figure 3.32 Switching element contribution of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Table 3.18 Interconnection link contribution of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	0.150000	0.166667	0.100000	0.062500
n=3	0.111111	0.100000	0.071429	0.025000
n=4	0.089286	0.066667	0.055556	0.010417
n=5	0.075000	0.047619	0.045455	0.004464
n=6	0.064815	0.035714	0.038462	0.001953
n=7	0.057143	0.027778	0.033333	0.000868
n=8	0.051136	0.022222	0.029412	0.000391
n=9	0.046296	0.018182	0.026316	0.000178
n=10	0.042308	0.015152	0.023810	0.000081

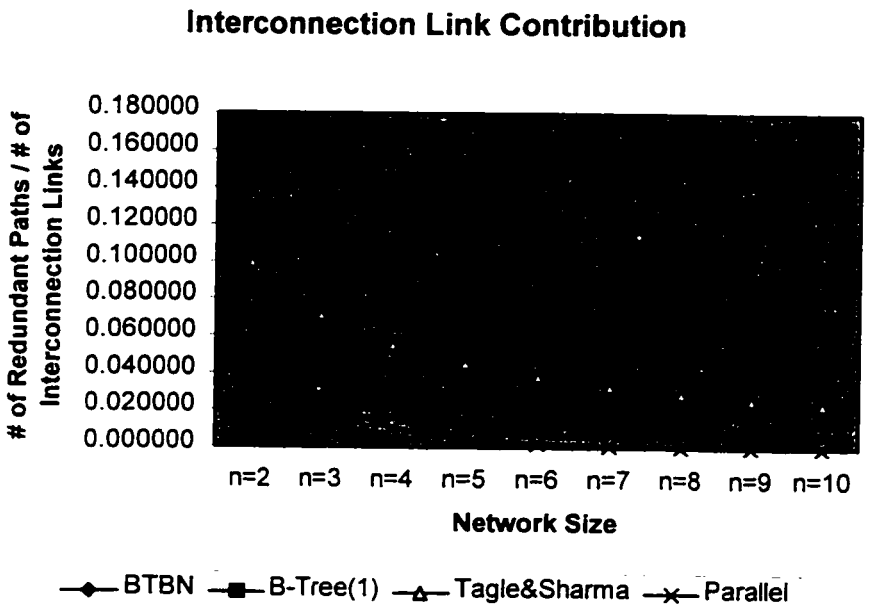


Figure 3.33 Interconnection link contribution of BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

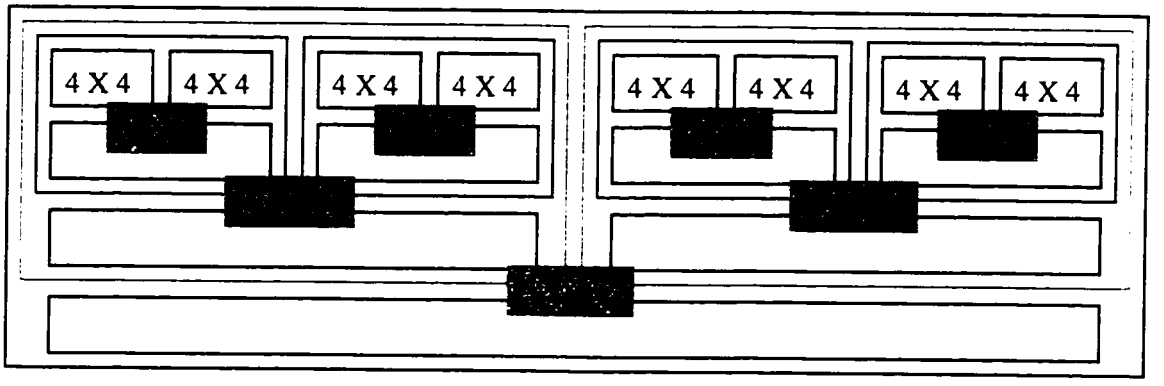


Figure 3.34 Illustration of BTBN modularity and expandability

As an example, Figure 3.35 illustrates the method used to build an 8 X 8 BTBN using two 4 X 4 BTBN networks. Figure 3.36 shows the final presentation of the 8 X 8 BTBN network.

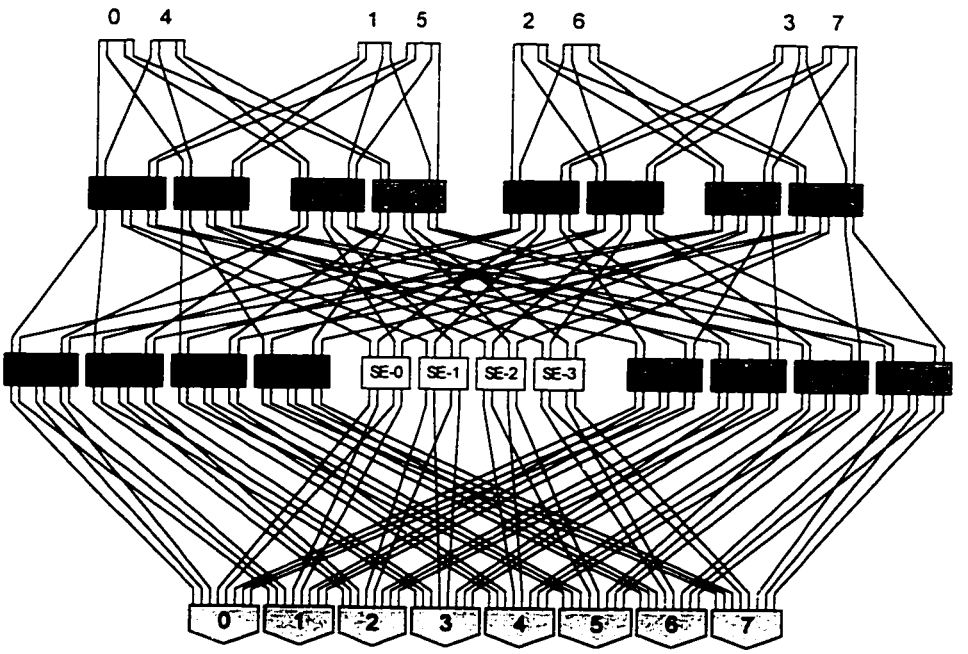


Figure 3.35 Building 8 X 8 BTBN using two 4 X 4 BTBN networks

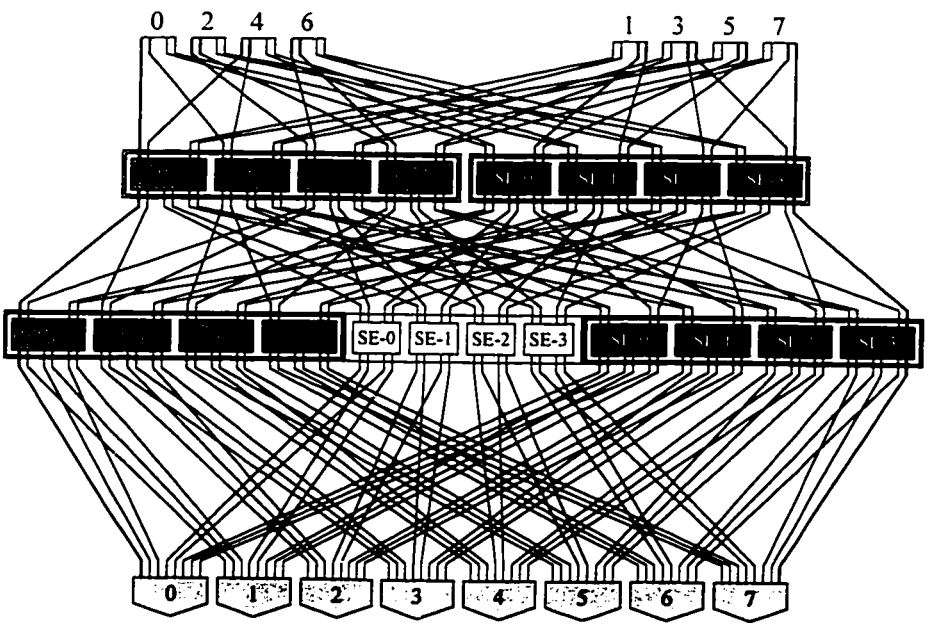


Figure 3.36 8 X 8 BTBN

Chapter 4

FAULT-FREE BTBN

PERFORMANCE EVALUATION

In this chapter, a through throughput performance analysis is conducted to evaluate the switching fabrics under several traffic models with the assumption that all switching fabrics are properly working. The traffic models are permutation, uniform, hot spot, and variable bit rate.

4.1 Permutation Traffic

This traffic is encountered in circuit switching applications. The input traffic to the switching fabric is basically the permutation patterns of the destination addresses of

the switching fabric. In any time slots, the input ports are fully loaded with cells that have unique destination address. The number of permutation patterns is equal to the factorial of the number of inputs. For example, a 4 X 4 network has 24 permutation patterns, an 8 X 8 network has 40320 permutation patterns, and a 32 X 32 network has more than $2X10^{13}$ permutation patterns. Figure 4.1 and Table 4.1 show the throughput performance of the switching fabrics under permutation traffic for the network size of n = 2 and n = 3. 4 X 4 and 8 X 8 BTBN networks are nonblocking since they passed all permutation patterns. However, the other switching fabrics are considered to be blocking for the network size of n = 3.

The permutation patterns are an excellent input traffic to measure the blocking problem since those input patterns do not have any output contention affect. In other words, the throughput performance degradation is due to pure internal blocking problem. However, the number of permutation patterns increases exponentially as the network size increases. For example, a 128 X 128 network has $3.8562X10^{215}$ different input patterns. To simulate such a network size under permutation traffic, it would take months. Of course this is not practical and that is the reason for limiting the network size in the design experiments in this thesis to only 8 X 8 networks.

Table 4.1 Throughput performance under permutation traffic for BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	1.000000	1.000000	1.000000	0.833333
n=3	1.000000	0.999975	0.999975	0.688078

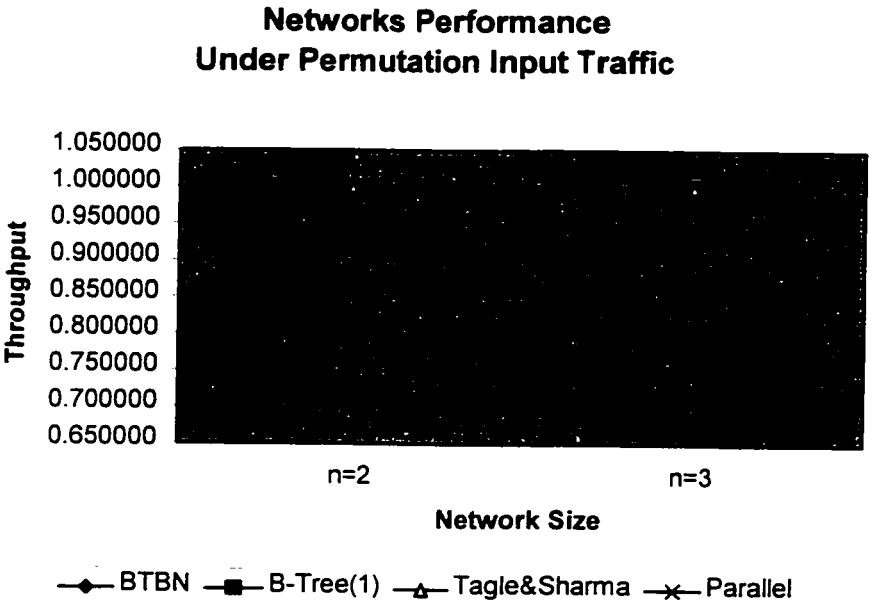


Figure 4.1 Throughput performance under permutation traffic for BTBN, B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

4.2 Uniform Traffic

This traffic is commonly used for evaluating ATM switching fabrics despite the fact that it is not an accurate representation of ATM traffic. This is because it is a practical traffic to measure the internal blocking problem of a given switching fabric despite the fact that the throughput performance degradation is due to random mix of both internal blocking and output contention problems. In addition, full load uniformly distributed traffic is much harder traffic for testing the switching fabric internal blocking problem than the actual ATM traffic. This issue will be clear when ATM traffic models are discussed in Section 4.4.

4.2.1 Analytical Model

There are two assumptions in the analysis of the switching fabrics. The first assumption is that the cells arrive at each input ports in a Bernoulli process with parameter λ, independent from all other input ports. The second assumption is that the input loads are uniformly distributed over all output ports of the switching fabric. The analytical model for 2 X 2 switching element is discussed in Section 3.1. This analytical model is extended for the 4 X 4 and 4 X 6 switching elements used in Tagle and Sharma's, B-Tree (1), and BTBN networks.

Parallel Banyan Network

When an internal conflict occurs between two arriving cells in a switching element, one cell is randomly selected for routing and the other cell is discarded. The throughput performance of this network is the same as the performance of the SEN described in Section 3.1 since the second parallel banyan is used only when there is faulty switching element in the main banyan network. Figure 4.2 and Table 4.2 show the analytical results for the throughput performance under various loads starting from 10% load to full load. For all various loads, as the network size increases, the throughput performance decreases. In fact, the throughput performance decreases much faster as the network size and the input load increase.

Tagle and Sharma's Network

When an internal conflict occurs among k arriving cells in a switching element where $1 < k < 5$, one cell is randomly selected for routing using the formal output link. Another cell from the remaining $k - 1$ is randomly selected for routing using the redundant output link, and the remaining $k - 2$ cells are discarded. Figure 4.3 shows the load labels for the 4 X 4 switching elements used in this network. The average load on the input links of the switching element is labeled by $\lambda_0, \lambda_1, \lambda_2$, and λ_3.

Table 4.2 Analytical throughput performance under various loads of uniform traffic for parallel banyan network.

	L=0.1	L=0.2	L=0.3	L=0.4	L=0.5	L=0.6	L=0.7	L=0.8	L=0.9	L=1.0
n=2	0.951234	0.904875	0.904875	0.819000	0.779297	0.741625	0.705891	0.672000	0.639859	0.609375
n=3	0.928613	0.863935	0.863935	0.751924	0.703384	0.659124	0.618691	0.581683	0.547740	0.516541
n=4	0.907055	0.826616	0.826616	0.695385	0.641540	0.593957	0.551705	0.514012	0.480236	0.449837
n=5	0.886486	0.792451	0.792451	0.647029	0.590094	0.541039	0.498439	0.461170	0.428345	0.399249
n=6	0.866840	0.761052	0.761052	0.605164	0.546567	0.497131	0.454962	0.418635	0.387062	0.359399
n=7	0.848055	0.732092	0.732092	0.568542	0.509225	0.460060	0.418738	0.383584	0.353353	0.327107
n=8	0.830075	0.705294	0.705294	0.536218	0.476811	0.428312	0.388054	0.354156	0.325260	0.300357
n=9	0.812849	0.680422	0.680422	0.507465	0.448393	0.400794	0.361701	0.329071	0.301456	0.277804
n=10	0.796331	0.657274	0.657274	0.481713	0.423261	0.376699	0.338806	0.307414	0.281009	0.258510

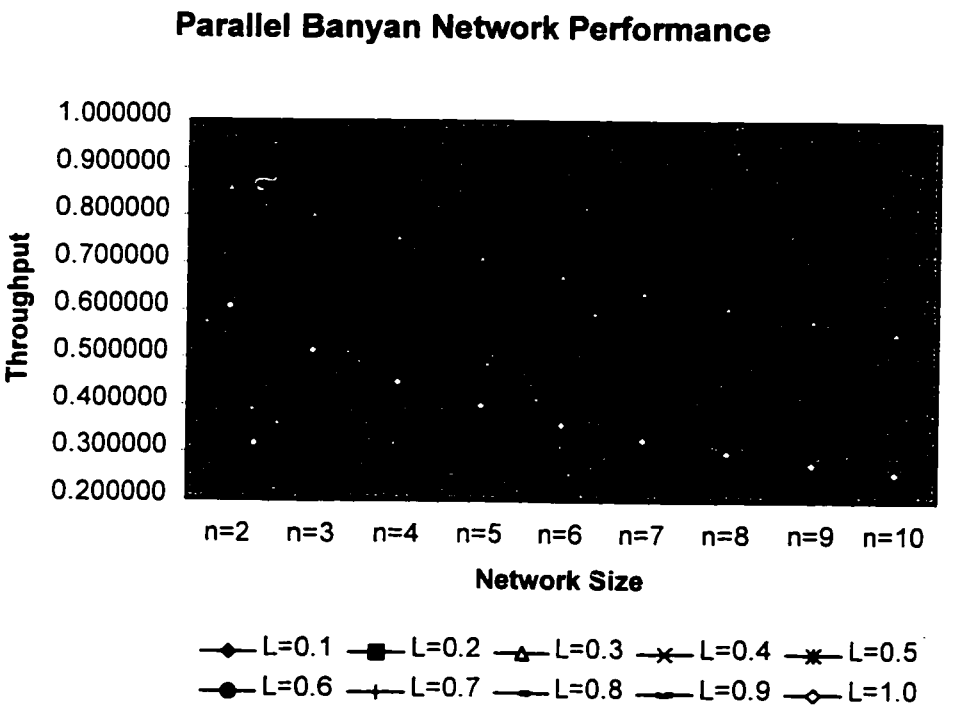


Figure 4.2 Analytical throughput performance under various loads of uniform traffic for parallel banyan network.

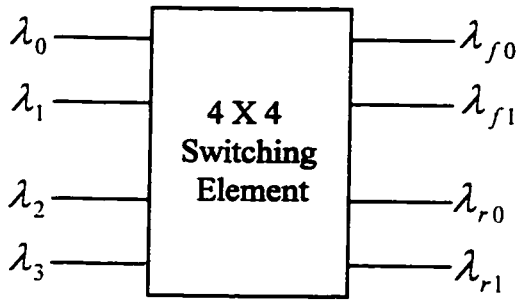


Figure 4.3 Load labeling for a 4 X 4 switching element.

The average load on the output links of the switching element is labeled by λ_{f0} / λ_{f1} for the formal output links, and λ_{r0} / λ_{r1} for the redundant output links. The formal output links have the highest priority of all output links. If at least one cell arrives at the switching element and the routing bit is equal to 0 or 1, then the formal output link $f_0(f_1)$ has one cell routed through. The loads on the output links of each switching element are obtained from the input loads. The following formula is derived based on the assumption that the four inputs are independent from each other:

$$\lambda_f = \lambda_{f0} = \lambda_{f1} = 1-(1-\lambda_0/2)(1-\lambda_1/2)(1-\lambda_2/2)(1-\lambda_3/2).$$

This formula represents the probability that at least one of the input cells selects the formal output link which is equal to one minus the probability that none of the arriving cells selects the formal output link (probability of no-cell arrival).

The redundant output links have the second priority of all output links. The redundant output link $r_0(r_1)$ is selected only when at least two cells arrive at the input links of the switching element and their routing bits are the same. The loading formula of the redundant output links is equal to the probability that at least one cell has arrived minus the probability that only one cell has arrived. This result is defined in the following formulas:

$$\lambda_r = \lambda_{r0} = \lambda_{r1} = \lambda_f - \lambda_{one-cell}$$

$$\lambda_{one-cell} = (\lambda_0/2)(1-\lambda_1/2)(1-\lambda_2/2)(1-\lambda_3/2)$$
$$+(1-\lambda_0/2)(\lambda_1/2)(1-\lambda_2/2)(1-\lambda_3/2)$$
$$+(1-\lambda_0/2)(1-\lambda_1/2)(\lambda_2/2)(1-\lambda_3/2)$$
$$+(1-\lambda_0/2)(1-\lambda_1/2)(1-\lambda_2/2)(\lambda_3/2)$$

Figure 4.4 and Table 4.3 show the analytical results for the throughput performance under various loads starting from 10% load to full load. For the input loads of 10% to 40%, as the network size increases, the throughput performance increases. For the input loads of 50% and 60%, as the network size increases up to n = 8, the throughput performance increases, then it starts slowly decreasing. For the input load of 70% to full load, as the network size increases up to n = 3, the throughput performance increases, then it starts slowly decreasing. In fact, the throughput performance decreases much faster as the network size and the input load increase.

B-Tree (1) Network

The probability of selecting the formal and redundant output links is the same formula defined for the switching element in Tagle and Sharma's network since they use the 4 X 4 switching element with the same functionality. Figure 4.5 and Table 4.4 show the analytical results for the throughput performance under various loads starting from 10% load to full load. For all various loads, as the network size increases, the throughput performance slowly decreases. In fact, the throughput performance decreases faster as the network size and the input load increase.

Table 4.3 Analytical throughput performance under various loads of uniform traffic for Tagle and Sharma's network.

	L=0.1	L=0.2	L=0.3	L=0.4	L=0.5	L=0.6	L=0.7	L=0.8	L=0.9	L=1.0
n=2	0.976219	0.954750	0.935406	0.918000	0.902344	0.888250	0.875531	0.864000	0.853469	0.843750
n=3	0.977288	0.958364	0.942174	0.927837	0.914634	0.902003	0.889520	0.876882	0.863895	0.850451
n=4	0.978251	0.961261	0.946942	0.933824	0.920936	0.907695	0.893797	0.879133	0.863721	0.847654
n=5	0.979119	0.963601	0.950363	0.937592	0.924323	0.910126	0.894885	0.878656	0.861577	0.843823
n=6	0.979902	0.965489	0.952805	0.939905	0.925968	0.910756	0.894308	0.876780	0.858371	0.839284
n=7	0.980608	0.967011	0.954535	0.941266	0.926588	0.910435	0.892948	0.874333	0.854808	0.834591
n=8	0.981244	0.968238	0.955746	0.942004	0.926607	0.909618	0.891230	0.871666	0.851154	0.829918
n=9	0.981818	0.969224	0.956578	0.942334	0.926274	0.908543	0.889354	0.868931	0.847506	0.825322
n=10	0.982336	0.970014	0.957134	0.942396	0.925737	0.907338	0.887414	0.866188	0.843902	0.820821

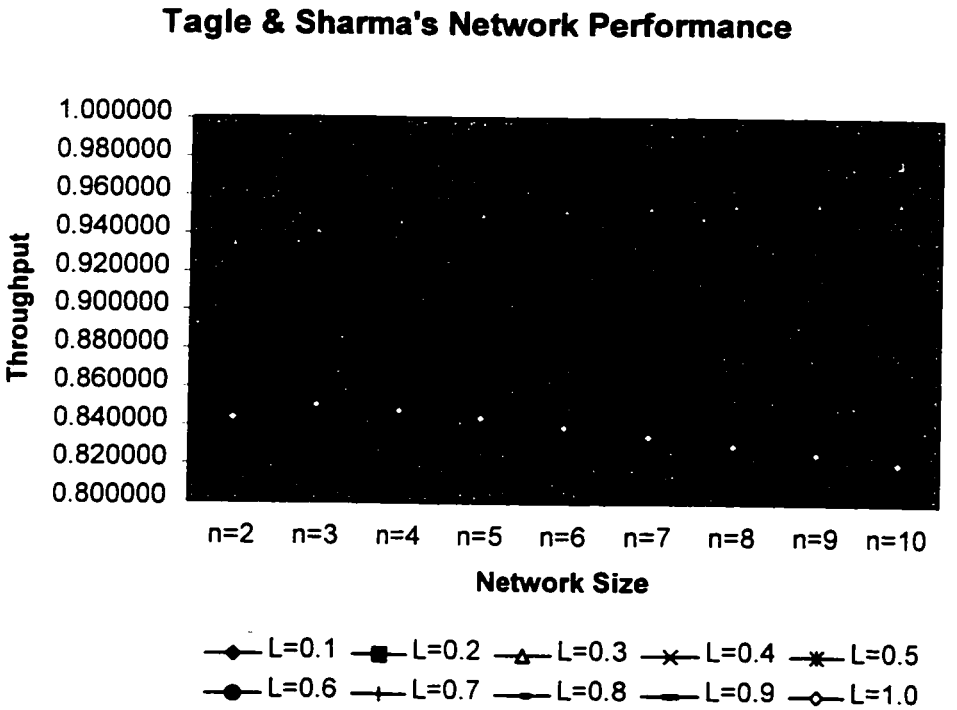


Figure 4.4 Analytical throughput performance under various loads of uniform traffic for Tagle and Sharma's network.

Table 4.4 Analytical throughput performance under various loads of uniform traffic for B-Tree (1) network.

	L=0.1	L=0.2	L=0.3	L=0.4	L=0.5	L=0.6	L=0.7	L=0.8	L=0.9	L=1.0
n=2	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
n=3	1.000000	0.999998	0.999985	0.999943	0.999841	0.999642	0.999303	0.998780	0.998037	0.997046
n=4	1.000000	0.999993	0.999953	0.999830	0.999560	0.999074	0.998310	0.997220	0.995777	0.993974
n=5	0.999999	0.999984	0.999904	0.999677	0.999214	0.998432	0.997276	0.995713	0.993735	0.991355
n=6	0.999999	0.999972	0.999843	0.999501	0.998843	0.997793	0.996307	0.994370	0.991983	0.989161
n=7	0.999998	0.999958	0.999773	0.999314	0.998475	0.997192	0.995433	0.993191	0.990468	0.987277
n=8	0.999997	0.999941	0.999698	0.999125	0.998123	0.996639	0.994651	0.992148	0.989133	0.985618
n=9	0.999996	0.999922	0.999619	0.998940	0.997793	0.996135	0.993945	0.991212	0.987936	0.984134
n=10	0.999995	0.999901	0.999540	0.998761	0.997486	0.995674	0.993302	0.990360	0.986851	0.982797

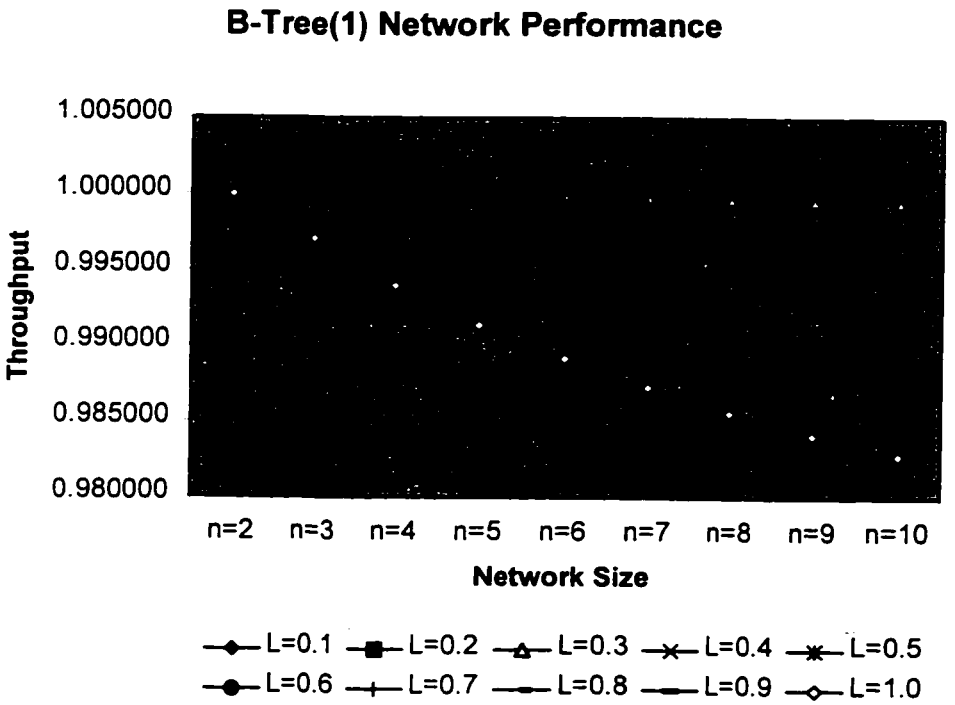


Figure 4.5 Analytical throughput performance under various loads of uniform traffic for B-Tree (1) network.

BTBN Network

BTBN has two switching element type 4 X 4 and 4 X 6. Figure 4.3 and Figure 4.6 show the load labels for the 4 X 4 and 4 X 6 switching elements used in this network. The average load on the input links of the switching element is labeled by $\lambda_0, \lambda_1, \lambda_2$, and λ_3. The average load on the output links of the switching element is labeled by λ_{f0} / λ_{f1} for the formal output links, λ_{r0} / λ_{r1} for the redundant output links, and λ_{s0} / λ_{s1} for the standby output links. The formal output links have the highest priority of all output links. The redundant output links have the second priority of all output links. The standby output links have the third priority of all output links. The 4 X 4 switching element has the same functionality as those switching elements in B-Tree (1). The same formulas also can be applied to the formal and redundant output links of the 4 X 6 switching elements. The standby output links are used only when more than two cells have arrived at the input links of the switching element and their routing bits are the same. The loading formula of the standby output links is equal to the probability that at least two cells have arrived minus the probability that only two cells have arrived. This result is defined in the following formulas:

$$\lambda_s = \lambda_{s0} = \lambda_{s1} = \lambda_r - \lambda_{two-cells}$$

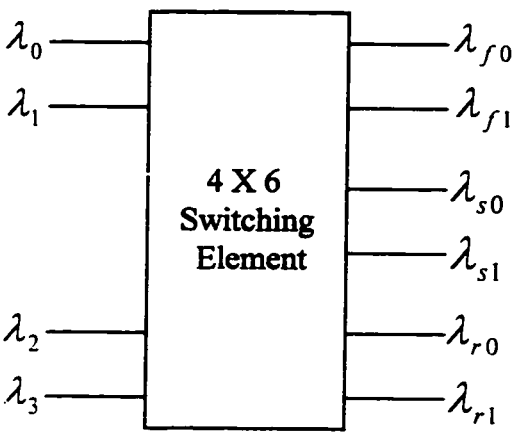


Figure 4.6 Load labeling for a 4 X 6 switching element.

$$\begin{aligned}\lambda_{two-cells} = &(\lambda_0/2)(\lambda_1/2)(1-\lambda_2/2)(1-\lambda_3/2)\\ &+(\lambda_0/2)(1-\lambda_1/2)(\lambda_2/2)(1-\lambda_3/2)\\ &+(\lambda_0/2)(1-\lambda_1/2)(1-\lambda_2/2)(\lambda_3/2)\\ &+(1-\lambda_0/2)(\lambda_1/2)(\lambda_2/2)(1-\lambda_3/2)\\ &+(1-\lambda_0/2)(\lambda_1/2)(1-\lambda_2/2)(\lambda_3/2)\\ &+(1-\lambda_0/2)(1-\lambda_1/2)(\lambda_2/2)(\lambda_3/2)\end{aligned}$$

Figure 4.7 and Table 4.5 show the analytical results for the throughput performance under various loads starting from 10% load to full load. For all various loads, as the network size increases, the throughput performance very slowly decreases. In fact, the throughput performance decreases faster as the network size and the input load increase. Overall, for 1024 X 1024 BTBN network, more than 99.9% throughput is achieved under full load of uniform traffic.

Analytical Comparison

Figure 4.8 shows the analytical throughput performance for the switching fabrics under 50% load of uniform traffic. BTBN and the B-Tree (1) have much better throughput performance than the parallel banyan and Tagle and Sharma's networks. The performance of Tagle and Sharma's network increases as the network size increases. The reason is that the number of redundant paths increases as the network size increases to resolve the internal blocking problems. Figure 4.9 shows the analytical throughput performance for the switching fabrics under 100% load of uniform traffic.

Table 4.5 Analytical throughput performance under various loads of uniform traffic for BTBN network.

	L=0.1	L=0.2	L=0.3	L=0.4	L=0.5	L=0.6	L=0.7	L=0.8	L=0.9	L=1.0
n=2	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
n=3	1.000000	1.000000	1.000000	0.999999	0.999997	0.999992	0.999983	0.999968	0.999943	0.999907
n=4	1.000000	1.000000	0.999999	0.999998	0.999993	0.999982	0.999963	0.999929	0.999876	0.999796
n=5	1.000000	1.000000	0.999999	0.999996	0.999989	0.999973	0.999942	0.999891	0.999809	0.999689
n=6	1.000000	1.000000	0.999999	0.999995	0.999985	0.999963	0.999922	0.999853	0.999744	0.999585
n=7	1.000000	1.000000	0.999998	0.999994	0.999981	0.999954	0.999902	0.999816	0.999682	0.999485
n=8	1.000000	1.000000	0.999998	0.999992	0.999977	0.999944	0.999883	0.999780	0.999621	0.999389
n=9	1.000000	1.000000	0.999998	0.999991	0.999973	0.999935	0.999864	0.999745	0.999561	0.999297
n=10	1.000000	1.000000	0.999998	0.999990	0.999969	0.999926	0.999845	0.999710	0.999504	0.999208

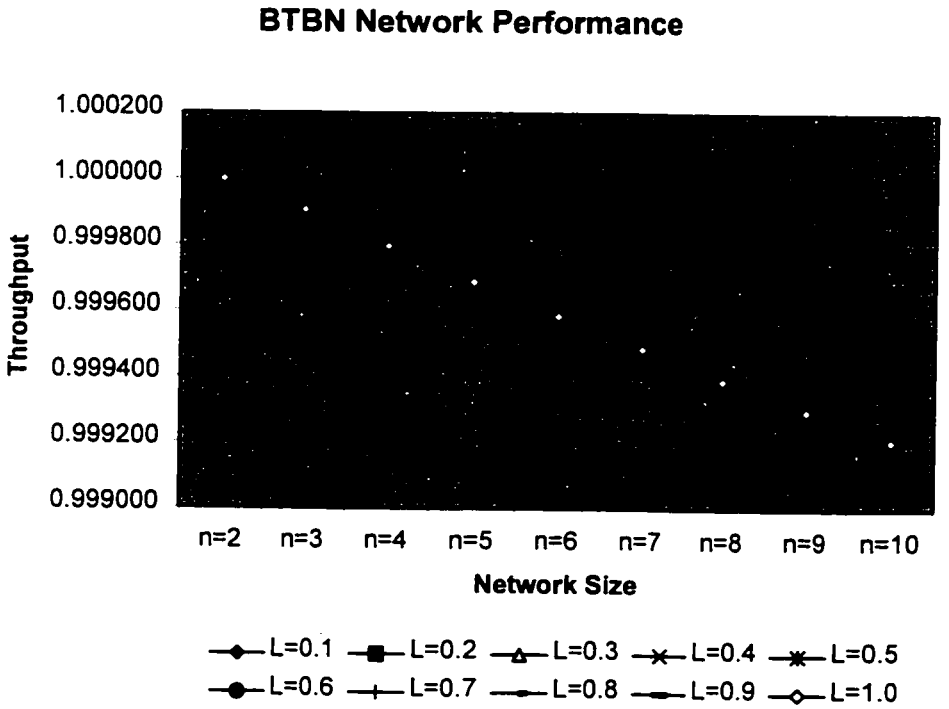


Figure 4.7 Analytical throughput performance under various loads of uniform traffic for BTBN network.

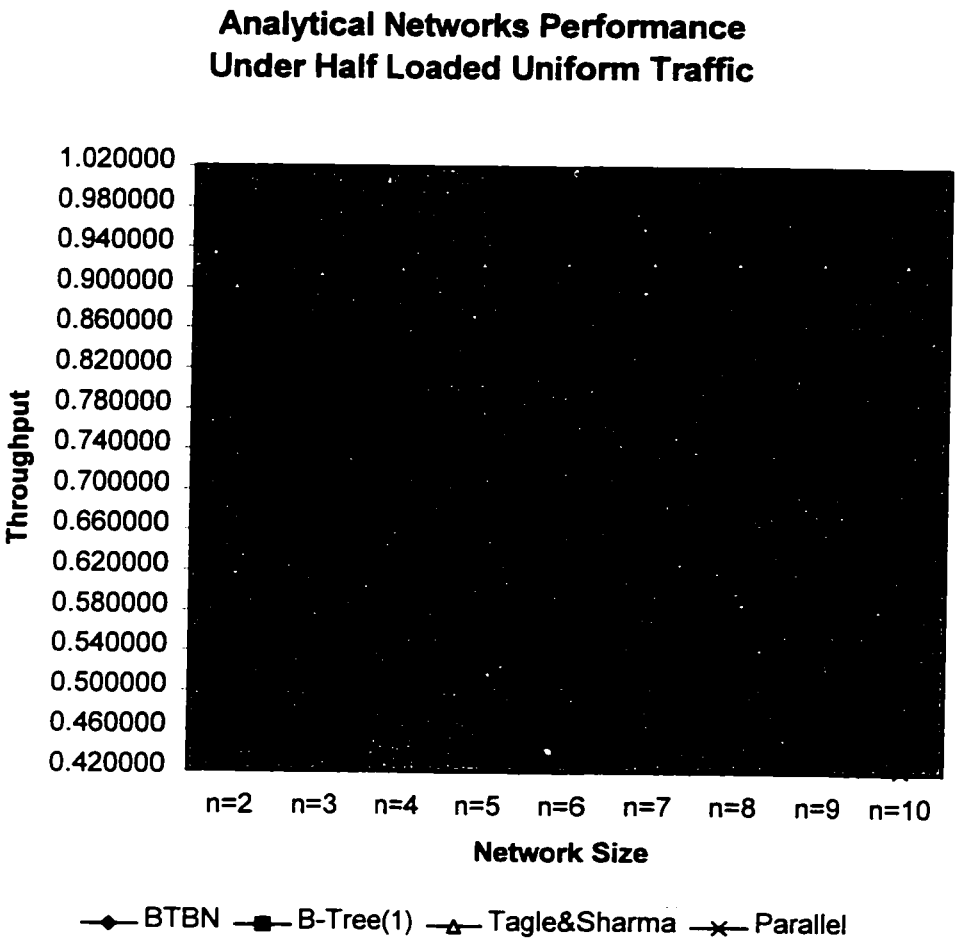


Figure 4.8 Analytical throughput performance under 50% load of uniform traffic.

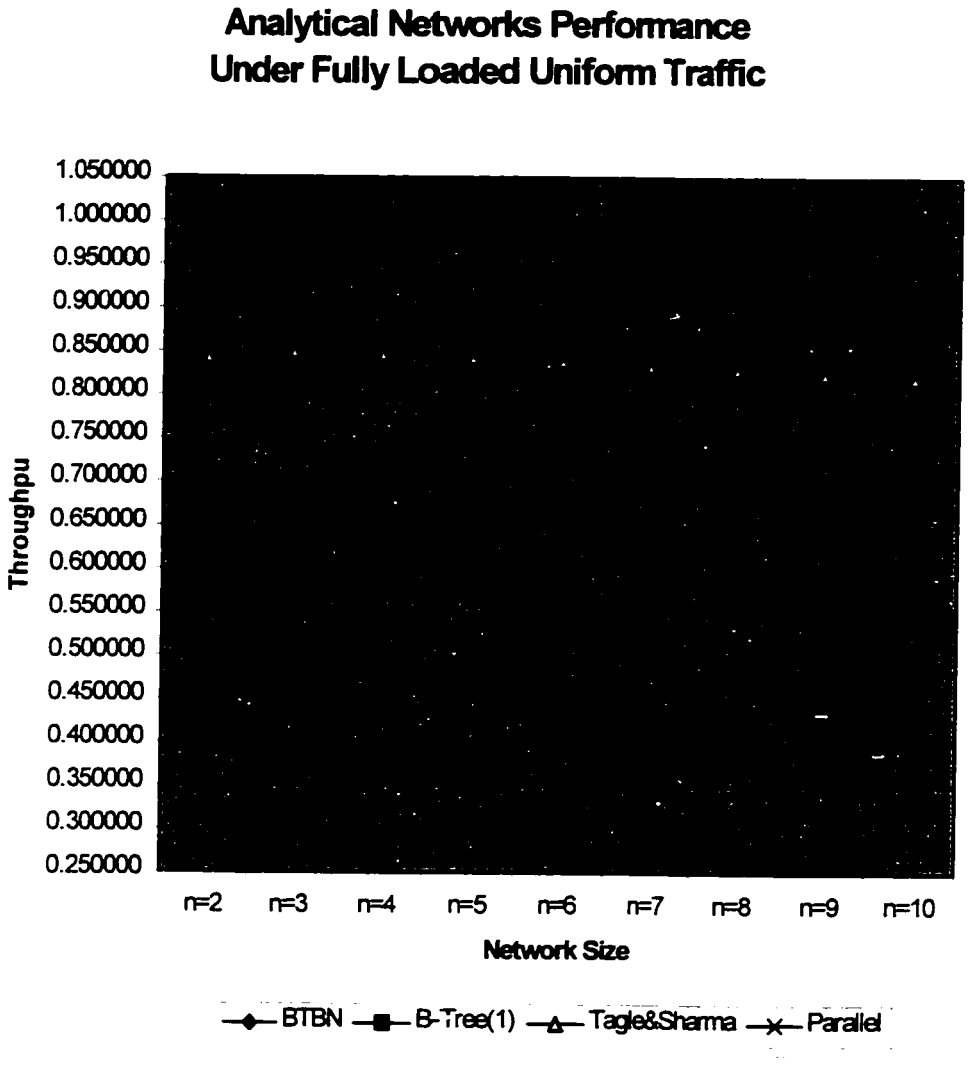


Figure 4.9 Analytical throughput performance under 100% load of uniform traffic.

The BTBN and the B-Tree (1) have much better throughput performance than the parallel banyan and Tagle and Sharma's networks. Under full load of uniform traffic, BTBN network performs better than the B-Tree (1) network. This result is shown very clearly in Figure 4.10. There is a very small impact on the throughput performance of BTBN network as the network size increases.

The performance of Tagle and Sharma's network under full load of uniform traffic increases as the network size increases to only n = 3 due to the increase in the number of redundant paths in order to minimize the internal blocking problems. However, as the network increases beyond n = 3, the throughput performance decreases slowly due to pure output contention problems since the network has only two access links to the output ports.

The B-Tree (1) network uses only 2n access links to each output port under normal conditions and 2(n+1) access links in the presence of faulty switching elements in the first switching stage. In the other hand, BTBN uses 2(n+1) access links to each output port all the time. For this reason, BTBN minimizes output contention very efficiently. The huge number of redundant paths in BTBN network as shown in Figure 3.31 and Table 3.16 reduces the internal blocking problem efficiently. For these two reasons, BTBN has the highest throughput performance by minimizing the internal blocking and output contention problems very efficiently as its size increases.

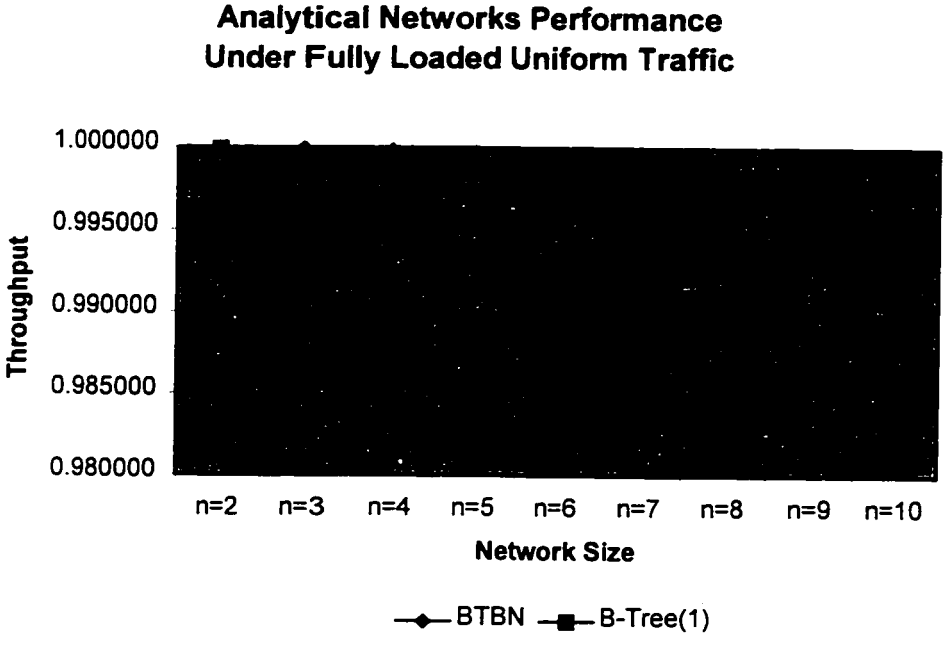


Figure 4.10 Analytical throughput performance under 100% load of uniform traffic for BTBN and B-Tree (1) networks.

4.2.2 Analytical Versus Simulation

Design experiments are simulated for the switching fabrics under full load of uniform traffic. Figure 4.11 shows the throughput performance results. The throughput performance values are presented in the following tables: Table 3.2 for the parallel banyan network, Table 3.5 for the Tagle and Sham's network, Table 3.11 for the B-Tree (1) and the BTBN networks.

Figure 4.12 and Table 4.6 show the throughput performance comparison between the analytical and the simulation results for B-Tree (1) and BTBN networks. Figure 4.13 and Table 4.7 show the throughput performance comparison between the analytical and the simulation results for parallel banyan and Tagle and Sharma's networks. Both results follow the same pattern as the network size increases. Very small difference between the analytical and the simulation results. This difference increases slowly as the network size increases. The analytical results are more optimistic than the simulation.

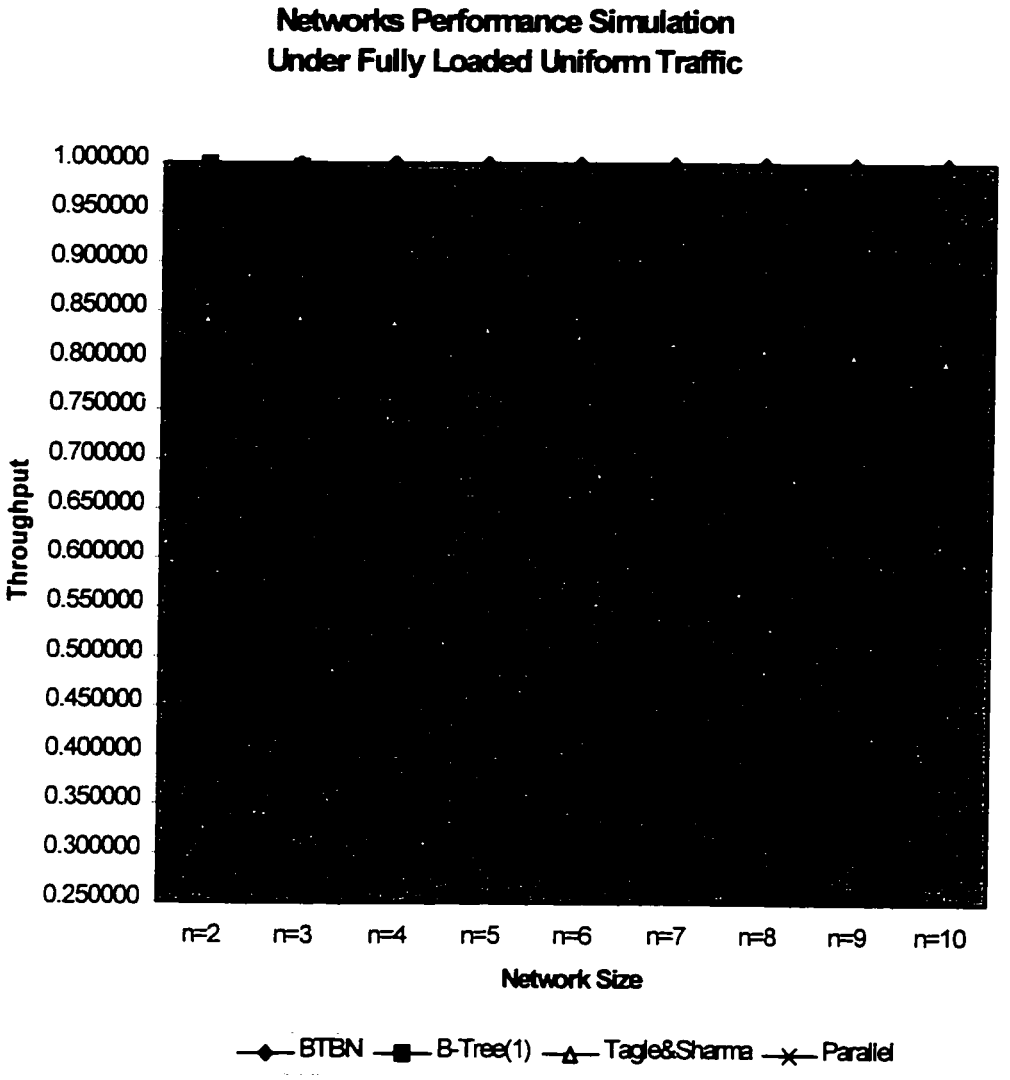


Figure 4.11 Throughput performance under 100% load of uniform traffic (Simulation).

Table 4.6 Throughput performance under 100% load of uniform traffic for BTBN and B-Tree (1) networks. (Analytical vs. Simulation)

Network Size	BTBN Analytical	BTBN Simulation	B-Tree (1) Analytical	B-Tree (1) Simulation
n=2	1.000000	1.000000	1.000000	1.000000
n=3	0.999907	0.999917	0.997046	0.996239
n=4	0.999796	0.999735	0.993974	0.992429
n=5	0.999689	0.999579	0.991355	0.989418
n=6	0.999585	0.999358	0.989161	0.986490
n=7	0.999485	0.999166	0.987277	0.984185
n=8	0.999389	0.998943	0.985618	0.982084
n=9	0.999297	0.998759	0.984134	0.980224
n=10	0.999208	0.998587	0.982797	0.978594

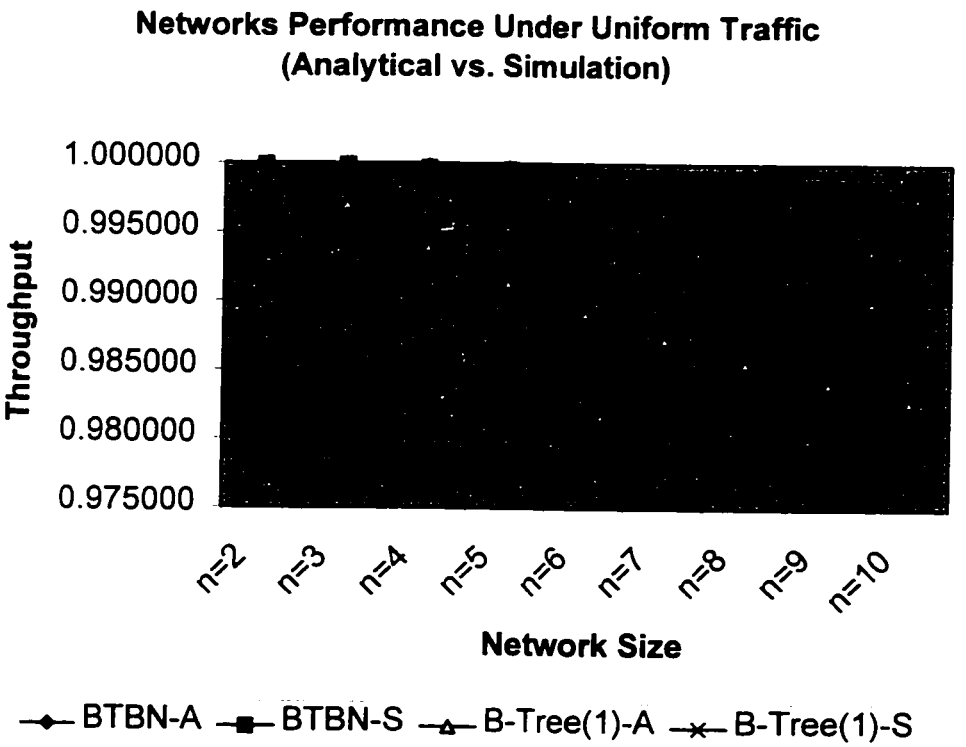


Figure 4.12 Throughput performance under 100% load of uniform traffic for BTBN and B-Tree (1) networks. (Analytical vs. Simulation)

Table 4.7 Throughput performance under 100% load of uniform traffic for parallel banyan and Tgale and Sharma's networks. (Analytical vs. Simulation)

Network Size	Tagle&Sharma Analytical	Tagle&Sharma Simulation	Parallel Analytical	Parallel Simulation
n=2	0.843750	0.843777	0.609375	0.609289
n=3	0.850451	0.845606	0.516541	0.516663
n=4	0.847654	0.839863	0.449837	0.450106
n=5	0.843823	0.833505	0.399249	0.399354
n=6	0.839284	0.826193	0.359399	0.359485
n=7	0.834591	0.819917	0.327107	0.327175
n=8	0.829918	0.813185	0.300357	0.300290
n=9	0.825322	0.806736	0.277804	0.277872
n=10	0.820821	0.800824	0.258510	0.258535

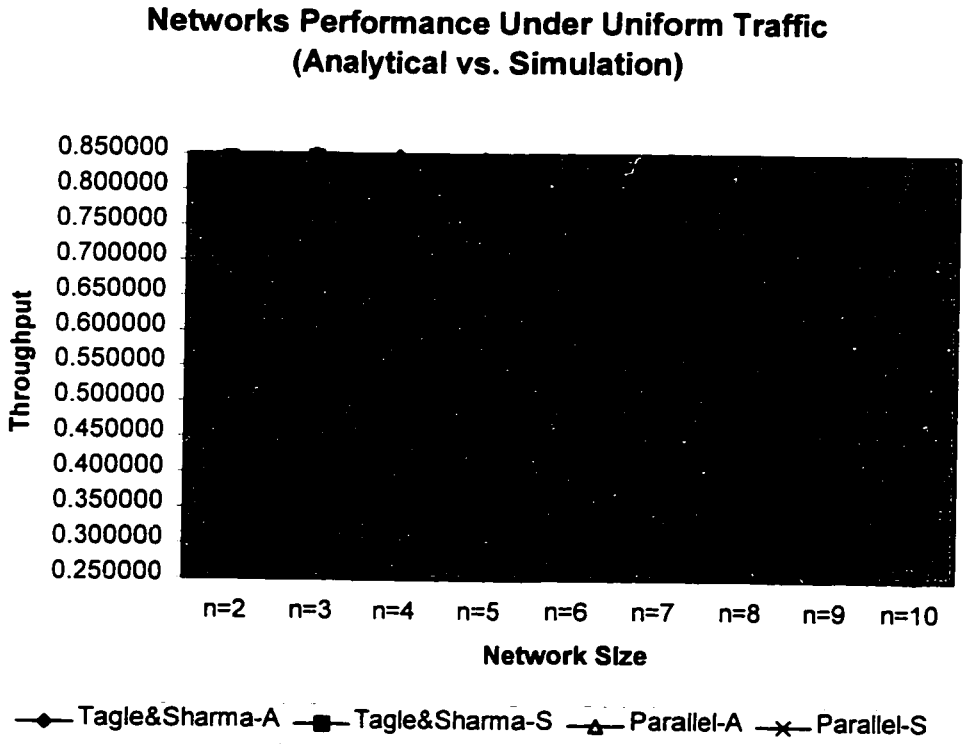


Figure 4.13 Throughput performance under 100% load of uniform traffic for parallel banyan and Tagle and Sharma's networks. (Analytical vs. Simulation)

4.3 Hot Spot Traffic

This type of traffic is encountered in data communication networks where one output port is highly in demand compared to other output ports. Hot spot traffic is an excellent input traffic for measuring the switching fabric output contention problem. There are two methods to simulate this traffic.

The first method is to assign a probability for selecting a designated hot spot output port. For example, assume that output port number zero is the designated hot spot output port. Assume that there is a probability of 25% that each input port will send cells to the hot spot output port (25% Hot Spot Traffic). Each input port generates a uniformly distributed random number, if the random number were equal to or less than 25%, then the cell would be sent to the hot spot output port. Otherwise, another uniformly distributed random number is generated for selecting the output port. Notice that the second number could also be the hot spot output port.

The second method to assign a probability for selecting the number of input ports constantly sending cells to a designated hot spot output port. For example, assume that the output port number zero is the designated hot spot output port. Assume that the first 25% of the input ports are constantly sending cells to the hot spot out put port (25% Hot Spot Traffic). The remaining input ports generate uniformly distributed

random number for selecting the out put port. Notice that the second number could also be the hot spot output port.

The second method is selected for the hot spot simulation for three reasons. The first reason is that the second method closely realizes the actual traffic in data communication networks where some work stations are dedicated for maintaining a file server and the other work stations in the network access the file server as needed. The second reason is the second method is much harder input traffic to test the output contention problem than the first method. The third reason, it is easier to generate the random numbers for selecting the output ports in the second method since the first method requires two random numbers to be generated for each input port.

Figure 4.14 and Table 4.8 show the throughput performance of the BTBN network for various percentages of the hot spot traffic. Figure 4.15 and Table 4.9 show the throughput performance of the B-Tree (1) network for various percentages of the hot spot traffic. Figure 4.16 and Table 4.10 show the throughput performance of the Tagle and Sharma's network for various percentages of the hot spot traffic. Figure 4.17 and Table 4.11 show the throughput performance of the parallel banyan network for various percentages of the hot spot traffic.

0% hot spot traffic is the same as the uniform traffic. On the other hand, 100% hot spot traffic is a hard condition where all input ports constantly sending cells to the same output port. Figure 4.18 shows the throughput performance of the switching fabrics under 25% hot spot traffic.

Table 4.8 Simulation throughput performance under various percentages of hot spot traffic for BTBN network.

	0% HS	25% HS	50% HS	75% HS	100% HS
n=2	1.000000	1.000000	1.000000	1.000000	1.000000
n=3	0.999170	0.999781	0.996104	0.968739	0.750000
n=4	0.999735	0.996212	0.871157	0.843702	0.625000
n=5	0.999579	0.933606	0.801135	0.593812	0.375000
n=6	0.999358	0.897072	0.703837	0.437473	0.218750
n=7	0.999166	0.849367	0.610347	0.343769	0.125000
n=8	0.998943	0.809620	0.555693	0.289142	0.070313
n=9	0.998759	0.778528	0.524630	0.257776	0.039063
n=10	0.998587	0.760892	0.506912	0.240199	0.021484

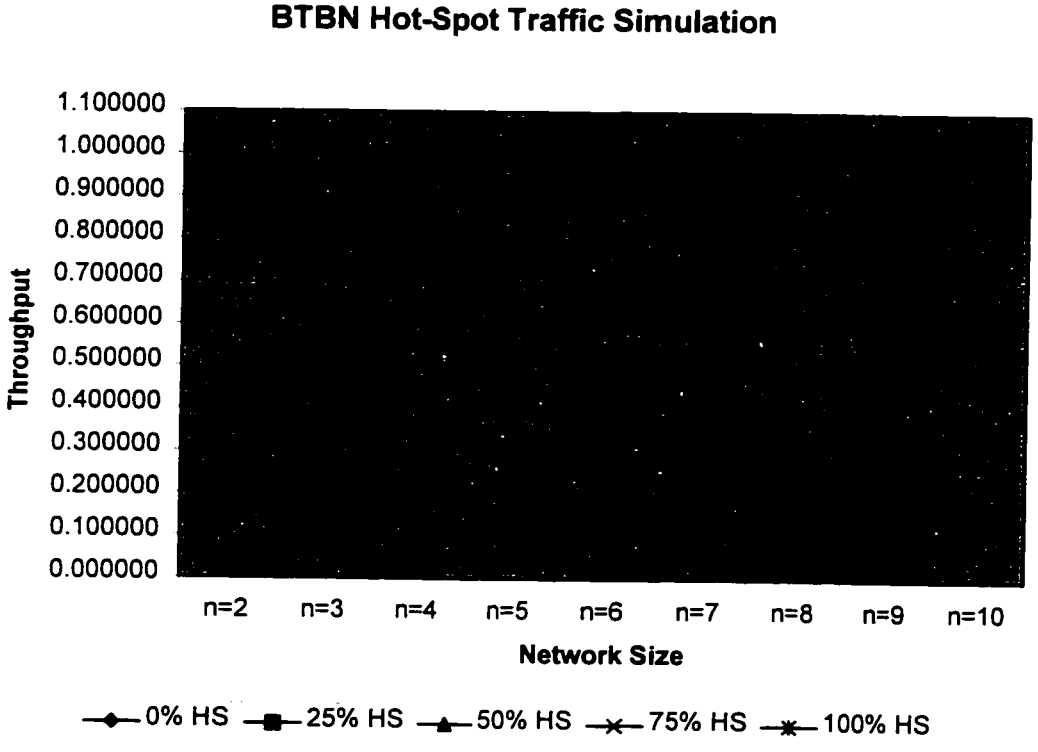


Figure 4.14 Simulation throughput performance under various percentages of hot spot traffic for BTBN network.

Table 4.9 Simulation throughput performance under various percentages of hot spot traffic for B-Tree (1) network.

	0% HS	25% HS	50% HS	75% HS	100% HS
n=2	1.000000	1.000000	1.000000	1.000000	1.000000
n=3	0.996239	0.993801	0.953065	0.851530	0.750000
n=4	0.992429	0.977098	0.842001	0.613304	0.500000
n=5	0.989418	0.910921	0.717005	0.489696	0.312500
n=6	0.986490	0.845020	0.623199	0.396652	0.187500
n=7	0.984185	0.797542	0.560770	0.334529	0.109375
n=8	0.982084	0.765711	0.521598	0.295791	0.062500
n=9	0.980224	0.745574	0.498346	0.272856	0.035156
n=10	0.978594	0.733387	0.485062	0.259926	0.019531

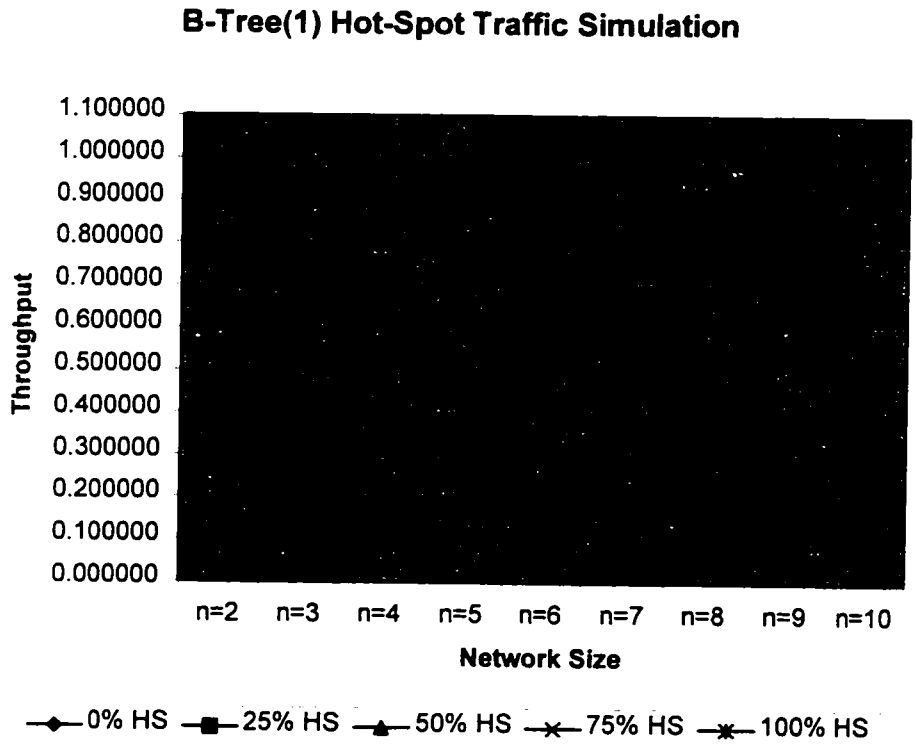


Figure 4.15 Simulation throughput performance under various percentages of hot spot traffic for B-Tree (1) network.

Table 4.10 Simulation throughput performance under various percentages of hot spot traffic for Tagle and Sharma's network.

	0% HS	25% HS	50% HS	75% HS	100% HS
n=2	0.843777	0.843852	0.874947	0.687470	0.500000
n=3	0.845606	0.835881	0.653392	0.468719	0.250000
n=4	0.839863	0.743756	0.528518	0.335211	0.125000
n=5	0.833505	0.678397	0.465511	0.272739	0.062500
n=6	0.826193	0.644625	0.434194	0.241527	0.031250
n=7	0.819917	0.626366	0.418366	0.225856	0.015625
n=8	0.813185	0.615917	0.409822	0.218307	0.007813
n=9	0.806736	0.609060	0.405700	0.214235	0.003906
n=10	0.800824	0.604804	0.402877	0.212465	0.001953

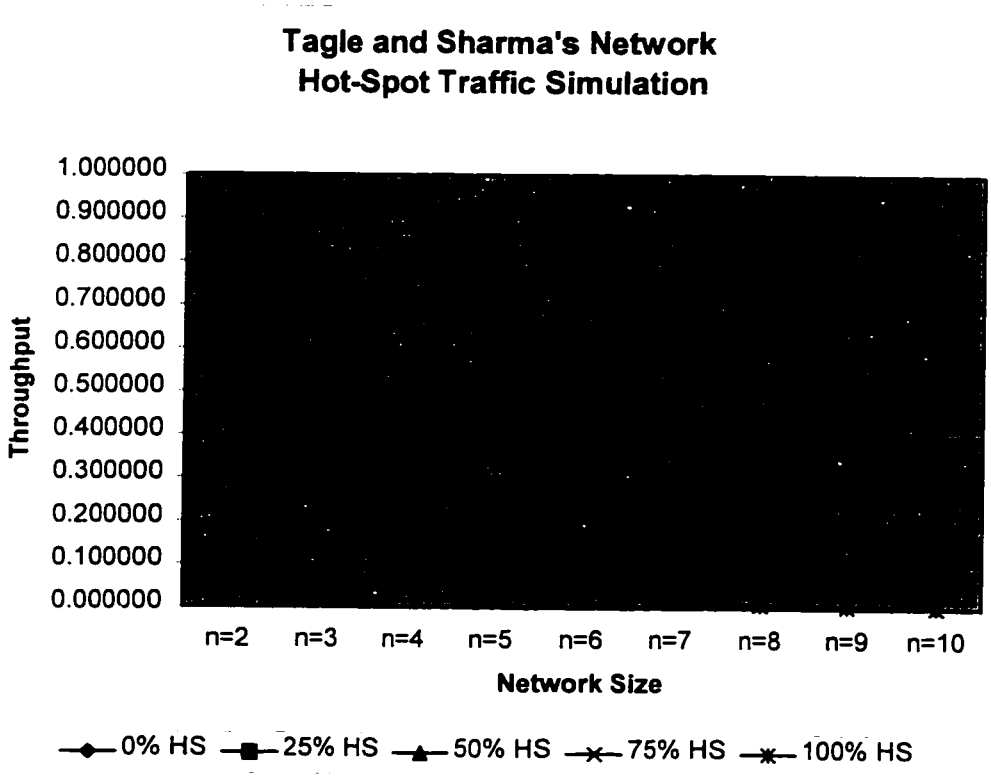


Figure 4.16 Simulation throughput performance under various percentages of hot spot traffic for Tagle and Sharma's network.

Table 4.11 Simulation throughput performance under various percentages of hot spot traffic for parallel banyan network.

	0% HS	25% HS	50% HS	75% HS	100% HS
n=2	0.609289	0.609372	0.531211	0.374883	0.250000
n=3	0.516663	0.489407	0.379871	0.265578	0.125000
n=4	0.450106	0.388160	0.295486	0.196537	0.062500
n=5	0.399354	0.326801	0.246242	0.159367	0.031250
n=6	0.359485	0.285879	0.215368	0.138326	0.015625
n=7	0.327175	0.257131	0.194190	0.125511	0.007813
n=8	0.300290	0.235346	0.178558	0.116849	0.003906
n=9	0.277872	0.217970	0.166553	0.110672	0.001953
n=10	0.258535	0.203709	0.156363	0.105859	0.000977

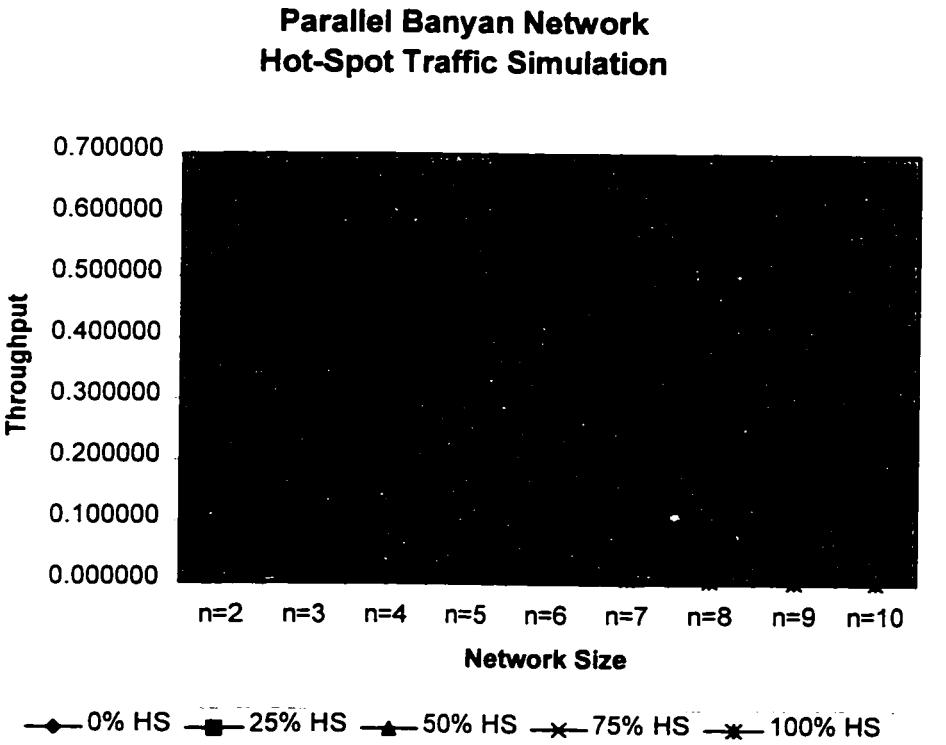


Figure 4.17 Simulation throughput performance under various percentages of hot spot traffic for parallel banyan network.

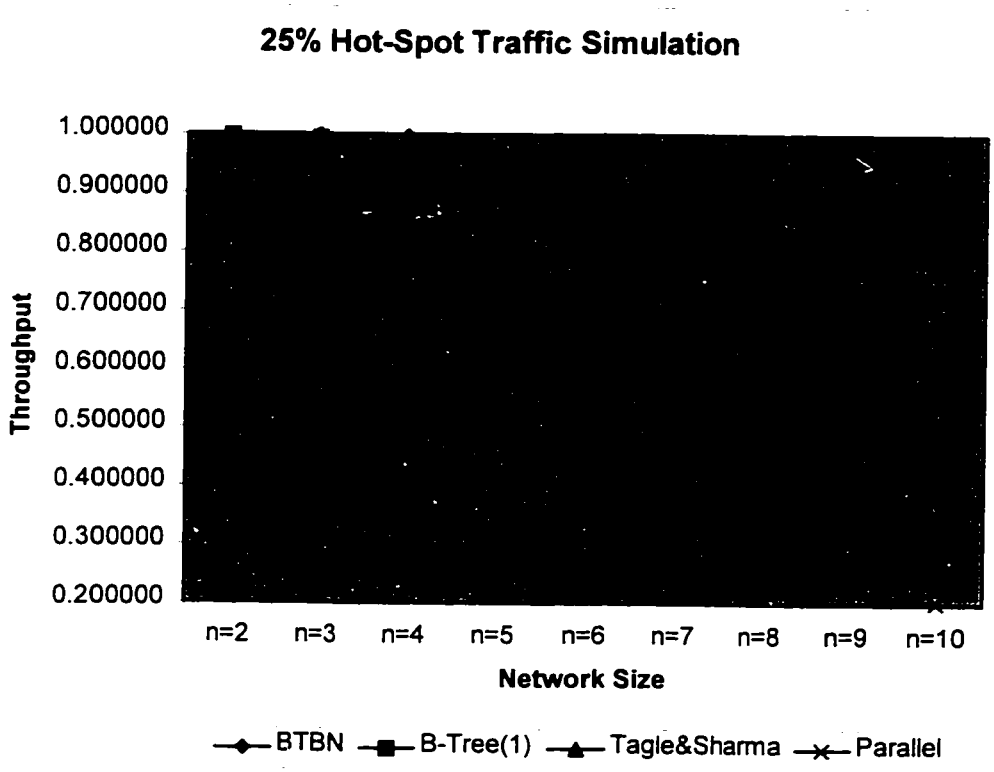


Figure 4.18 (Simulation) Throughput performance under 25% hot spot traffic.

Figure 4.19 shows the throughput performance of the switching fabrics under 50% hot spot traffic. Figure 4.20 shows the throughput performance of the switching fabrics under 75% hot spot traffic. BTBN has the highest throughput performance since it has the highest number of access links to each output port.

4.4 ATM Input Traffic

4.4.1 ATM Service Requirements

ATM-based networks are designed to support a wide range of service requirements in order provide acceptable service to all types of traffic. The supported services can be classified as follow [31]:

Interactive and distributive – Interactive services include telephone conversations and database retrieval. These services generally have a medium to low delay tolerance. Distributed services include television broadcast and text news services. These services require special routing techniques in order to avoid transmitting multiple copies along the same link.

Broadband and narrowband rates – Bit rate requirement ranges form 16 Kbps for telephone traffic to 155 Mbps for HDTV.

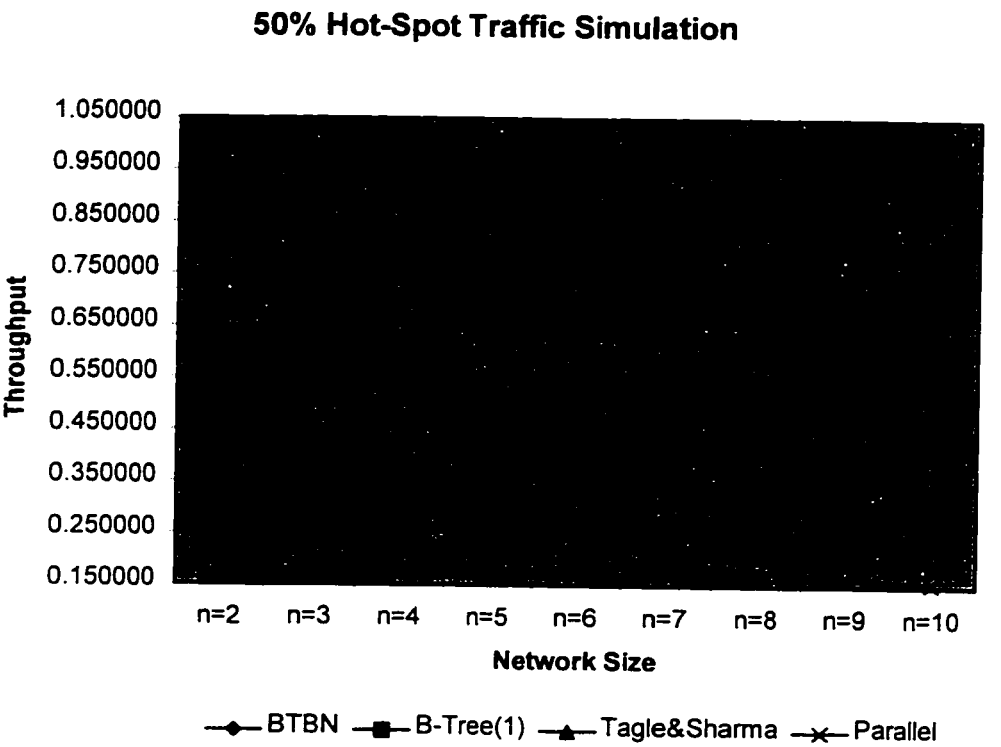


Figure 4.19 (Simulation) Throughput performance under 50% hot spot traffic.

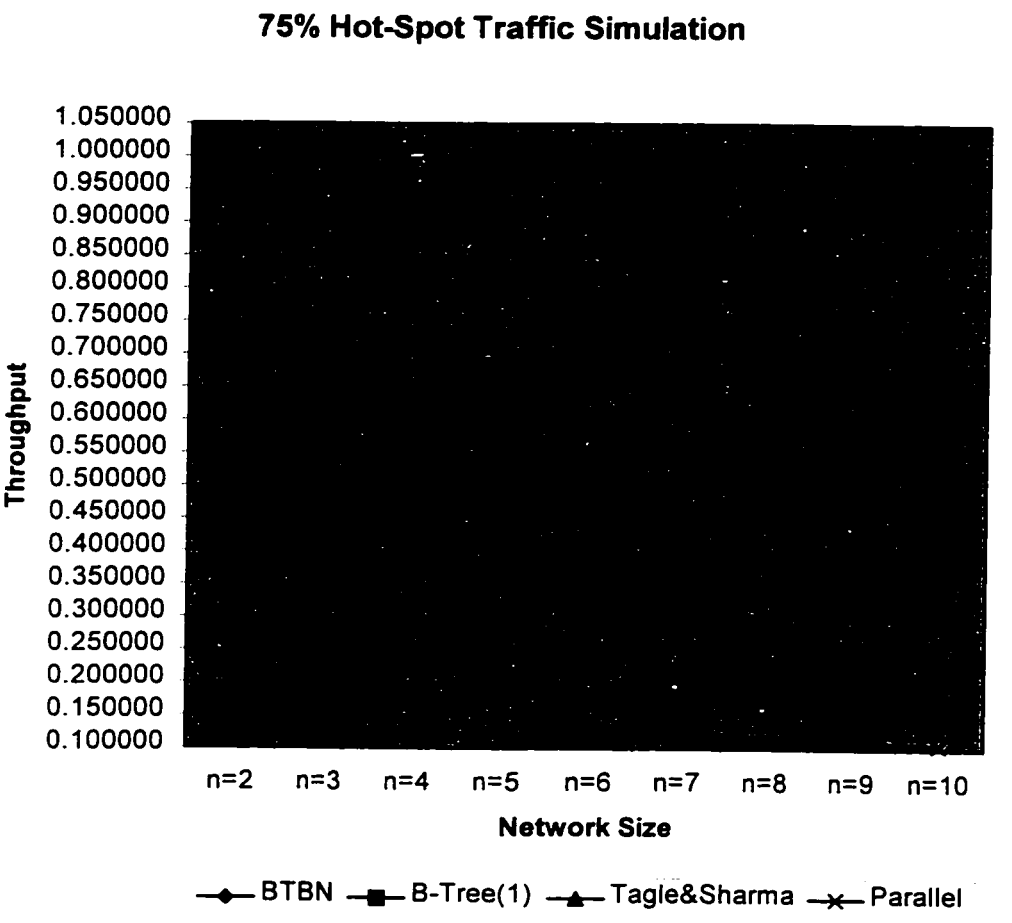


Figure 4.20 (Simulation) Throughput performance under 75% hot spot traffic.

Bursty and continuous traffic – Few services require continuos guaranteed bandwidth and most of other services are bursty.

Connection-oriented and connectionless – Connection-oriented services such as telephone, teleconferencing, and interactive data transfer have two phases: connection establishment and information transfer. On the other hand, connectionless services such as electronic mail, updating databases, and facsimile do not need a separate connection establishment phase in a similar manner to messaging services.

Point-to-point and complex communications – Services might require a single point-to-point, point-to-multipoint, or multipoint-to-multipoint connections.

ATM traffic can be classified according to the following services: data, voice, and video.

Data traffic- It has the least stringent service requirements of the three types of services. This type of traffic is sensitive to cell loss rather than transmission delay.

Voice traffic – It is bursty traffic that has a low bit rate. This type of traffic can tolerate cell loss to a certain degree.

Video traffic – It is bursty traffic that has the highest bit rate. The bit rate ranges from 1.5 Mbps to 155 Mbps. In-sequence cell delivery is an important issue concerning the reconstruction of the video image at the destination. In addition this type of traffic is highly delay sensitive. The cells must arrive on time in the right order at the correct interval. The type of traffic is sensitive to cell loss for "live" video.

The purpose of testing a switching fabric under ATM input traffic is to observe the throughput performance under a realistic traffic load. Usually ATM input traffic is not as hard as the full load of uniform traffic since most of the ATM input traffic is bursty and at instance of time, some input ports do not send any cells to any output ports. While full load uniform traffic acts as having very long burst traffic in each input port. The input ports do not have idle time while testing the switching fabric.

4.4.2 ATM Traffic Models

Traffic flow tends to be periodic during duration of a burst since the burst generated by a source usually consists of a single piece of information such as video frame or a data file. The cells in a burst are evenly spaced since the time required to build each cell is approximately constant. The easiest method to model a bursty input traffic is to use a two-state model in which one state represents a burst or active period as shown in Figure 4.21. During the active state, the source transmits cells at some given rate. Depending on the type of the input source, the active period may be followed by an idle period during which the source is silent. This model is known as the ON-OFF model. There are other models such as the Markov Modulated Poisson Process (MMPP) model that is primarily used for modeling aggregate traffic streams, and the Generally Modulated Deterministic Process (GMDP) model.

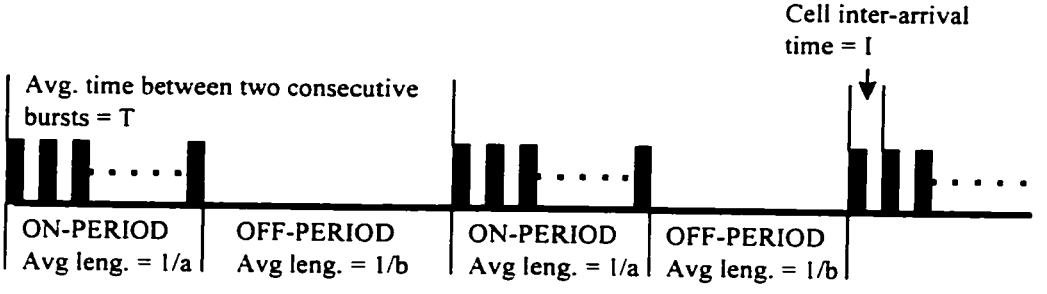


Figure 4.21 ON/OFF source model.

Table 4.12 ITU-T recommended traffic parameters for various ATM input source models.

Type of Source	B in cells	Average bit rate m x 384 b/s	Burstiness β	Cell Loss Tolerance
Constant Bit Rate (CBR)	N/A	64 Kb/s	1	10^{-4} to 10^{-6}
Connectionless data	200	700 Kb/s	1000	10^{-12}
Connection oriented data	200	25 Mb/s	1000	10^{-12}
Variable Bit Rate (VBR) video	2	25 Mb/s	2 to 5	10^{-10}
Background data/video	3	1 Mb/s	2 to 5	10^{-9} to 10^{-10}
VBR video/data	30	21 Mb/s	2 to 5	10^{-9}
Slow video	3	6 Mb/s	25	10^{-12}

On/OFF source model is simple and flexible enough to represent most of the existing traffic sources with reasonable accuracy. It is assumed that successive active and idle periods are statistically independent. ITU-T suggested that the length of the active and idle periods be exponentially distributed. The following are the required parameters to completely characterize an ON-OFF traffic source as shown in Figure 4.21.

Cell inter-arrival time (I): the time between the arrival of the first bit of a cell and the first bit of the next consecutive cell from a given source.

Active period mean value (1/a): active period is exponentially distributed with a mean value of 1/a.

Idle period mean value (1/b): idle period is exponentially distributed with a mean value of 1/b.

Peak cell arrival rate (P): the cell arrival rate when the source is active.

Average cell arrival rate (m): the average cell arrival rate when the source is active.

Traffic burstiness (β): the ratio of the peak cell arrival rate over the average cell arrival rate. Very bursty source has large value.

Active period duration (t_{on}): Average duration of the active period.

Idle period duration (t_{off}): Average duration of the idle period.

Average burst length (B): Average number of cells during active period.

Average ON-OFF cycle (T): the average time between two consecutive bursts.

Given any type of source with the three parameters: B, m and β, the active and idle period mean values 1/a and 1/b respectively ca be calculated as follows. For example, the parameters of the VBR video/data source model with B = 30 cells, m = 21 Mb/s and $\beta = 5$ are calculated as follow:

$P = m^* \beta = 21(5) = 105$ Mb/s

$$I = \frac{384\,b/cell}{P} = \frac{384}{105X10^6} = 3.657\ \mu s$$

$$1/a = B^*I = 30X3.657X10^{-6} = 109.71\ \mu s$$

$$1/b = (\beta\text{-}1)(1/a) = 4X109.71X10^{-6} = 438.84\ \mu s$$

Once the parameters *a*, *b*, and *I* are calculated, the ON/OFF input source model can be simulated. The active period is exponentially distributed with a mean value of *1/a* and the idle period is exponentially distributed with a mean value of *1/b*. The cells are generated during the active period constantly every *I* period.

4.4.3 ATM Traffic Simulation

The ATM input traffic mix used in the design experiments as follow:

10% of the input ports are connected to voice applications

20% of the input ports are connected to connectionless data applications

30% of the input ports are connected to connection-oriented data applications

40% of the input ports are connected to variable-bit-rate video and data applications

It is assumed that all switching fabrics have the same service time and transmit all cells at the input ports at 155 Mbps. The ATM switching fabric simulation slot time is equal to 384b/155Mbps = 2.4774193 microseconds. It is also assumed that each input port has enough queuing buffer to store the cells transmitted during the active period if the cell generation rate is faster than the switching fabric service time. Figure 4.22 and Table 4.13 show the throughput performance of the switching fabrics under ATM input traffic. BTBN network has the highest throughput performance. The throughput performance degradation of the switching fabrics under this traffic is very small compared to the uniform traffic. The throughput performance of 512 X 512 BTBN network under ATM input traffic is more than 99.9999%. Approximately, there is no impact in the throughput performance of the BTBN network under ATM input traffic as the network size increases. In fact, the cell loss probability is about 10^{-6}. This is an excellent feature for designing large size ATM switching fabrics.

Table 4.13 Simulation throughput performance under ATM input traffic (10% Voice, 20% Connectionless data, 30% Connection oriented data, and 40% VBR video/data).

Network Size	BTBN	B-Tree (1)	Tagle&Sharma	Parallel
n=2	1.000000000	1.000000000	0.998763185	0.964569315
n=3	1.000000000	1.000000000	0.980705801	0.939471933
n=4	1.000000000	0.999999537	0.979291149	0.910553103
n=5	1.000000000	0.999997200	0.980082220	0.888731697
n=6	0.999999985	0.999995025	0.980295067	0.869567489
n=7	0.999999919	0.999992082	0.980610195	0.850458675
n=8	0.999999913	0.999988404	0.981292135	0.832297365
n=9	0.999999890	0.999984753	0.981753484	0.815433104

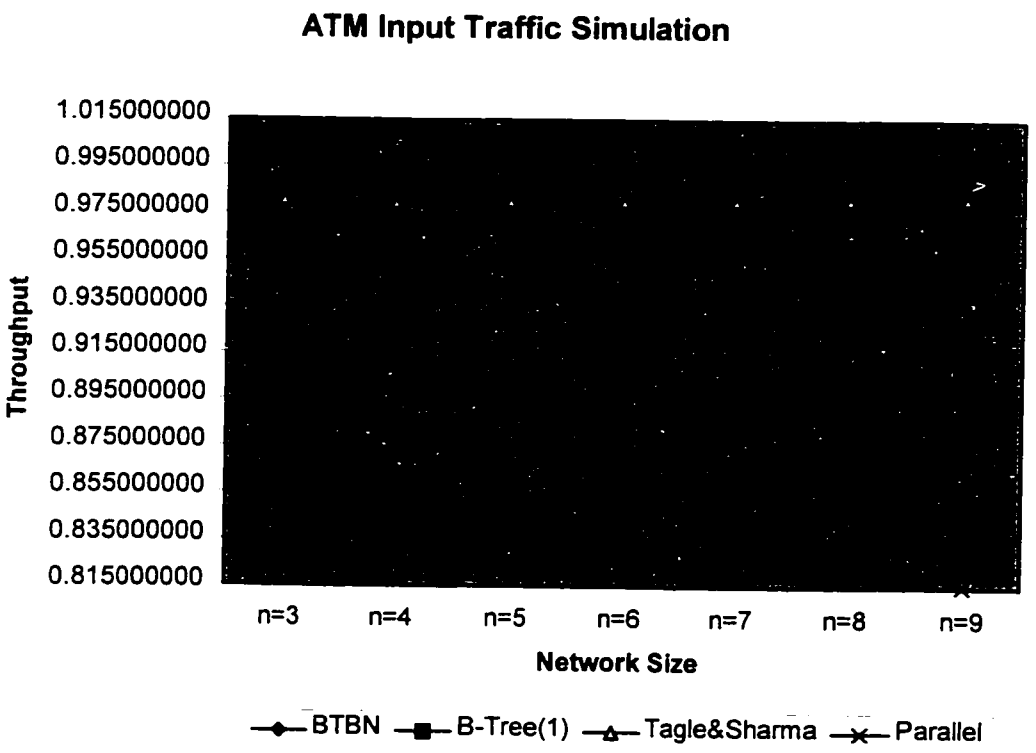


Figure 4.22 Simulation throughput performance under ATM input traffic (10% Voice, 20% Connectionless data, 30% Connection oriented data, and 40% VBR video/data).

Chapter 5

FAULTY BTBN

PERFORMANCE EVALUATION

The performance evaluation of the switching fabrics in the presence of faulty switching elements is done using only the uniform traffic. The faulty model considered in the simulation is only the failure of switching elements. It is assumed that the interconnection links, input and out ports, demultiplexers directly connected to the input ports, and multiplexers directly connected to output ports never fail. No cell can be routed through a faulty switching element. Any link connected to a faulty switching element is useless and marked faulty. Under the above conditions, the

throughput performance of the switching fabrics in the presence of faulty switching elements is evaluated using two methods.

The first method assumes that the faults of switching elements occur randomly with uniform distribution in probability, the input cells arrive at each input port independently, and the input traffic is uniformly distributed over all the output ports.

The second method assumes that the faults of switching elements occur selectively in all routes from all inputs to a designated output port. The following procedure describes this method for the simulation of k faulty switching elements:

1. Apply 100% hot spot traffic to each switching fabric assuming there is no faulty switching element.
2. Identify all switching elements used in routing the 100% host spot traffic in the first step. This means marking all switching elements in the routes to the designated hot spot output port. Assume there are m switching elements in the routes.
3. Select k marked switching elements in the routes to the hot spot output port and make them faulty.
4. Apply 100% hot spot traffic and record the throughput performance of the switching fabric in the presence of k faulty switching elements.
5. Repeat the third and fourth steps for the remaining $\binom{m}{k}$ combinations and obtain the average throughput performance.

Figure 5.1 shows the results of the design experiment to evaluate the throughput performance of the BTBN networks from the size of n = 6 to n = 10 under uniform traffic and randomly selected faulty switching elements. The purpose of this experiment is to show the affect of increasing the number of faulty switching elements on the switching fabric performance. The number of randomly selected faulty switching elements is incremented by five in each experiment. The throughput performance of a 64 X 64 BTBN is more than 80% in the presence of 200 faulty switching elements. This means that the throughput performance of a BTBN of size n = 6 is more than 80% with almost one third of the total number of switching elements in the network are faulty. BTBN networks of size n = 8 and higher have a very small effect on the throughput performance in the presence of faulty switching elements. As the BTBN network size increases, the effect of the faulty switching elements decreases as shown in Figure 5.2. In fact, BTBN networks of size n = 7 and higher have more than 99% throughput performance in the presence of 200 faulty switching elements. This high throughput performance is due to the huge number of redundant paths available in BTBN networks. This is an excellent feature for fault tolerant ATM switching fabrics.

Figure 5.3 shows the results of the design experiment to evaluate the throughput performance of the B-Tree (1) networks from the size of n = 6 to n = 10 under uniform traffic and randomly selected faulty switching elements.

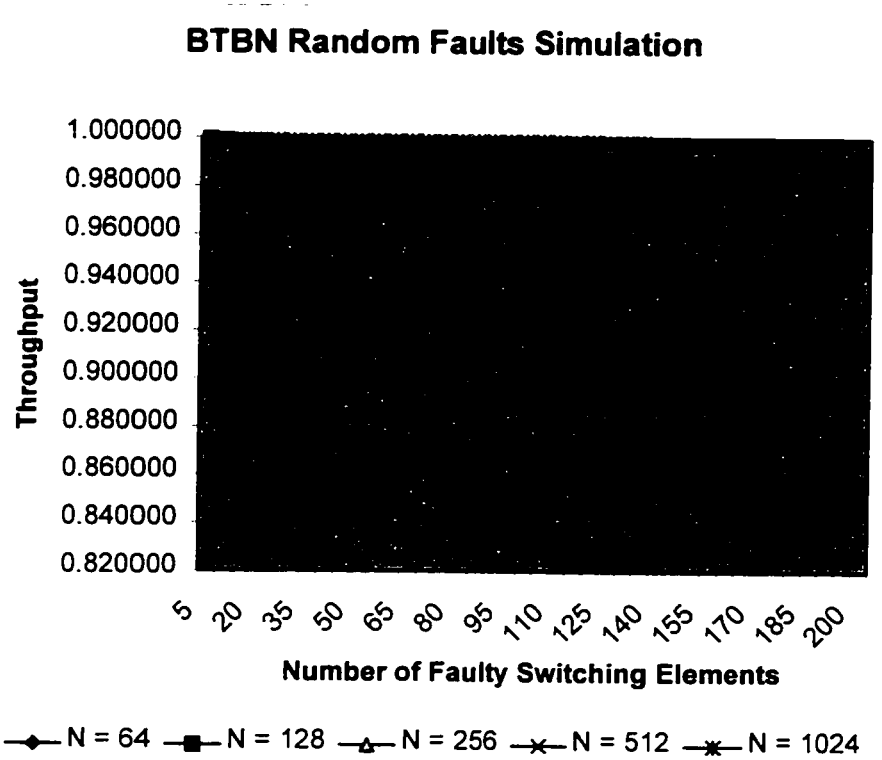


Figure 5.1 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for BTBN networks.

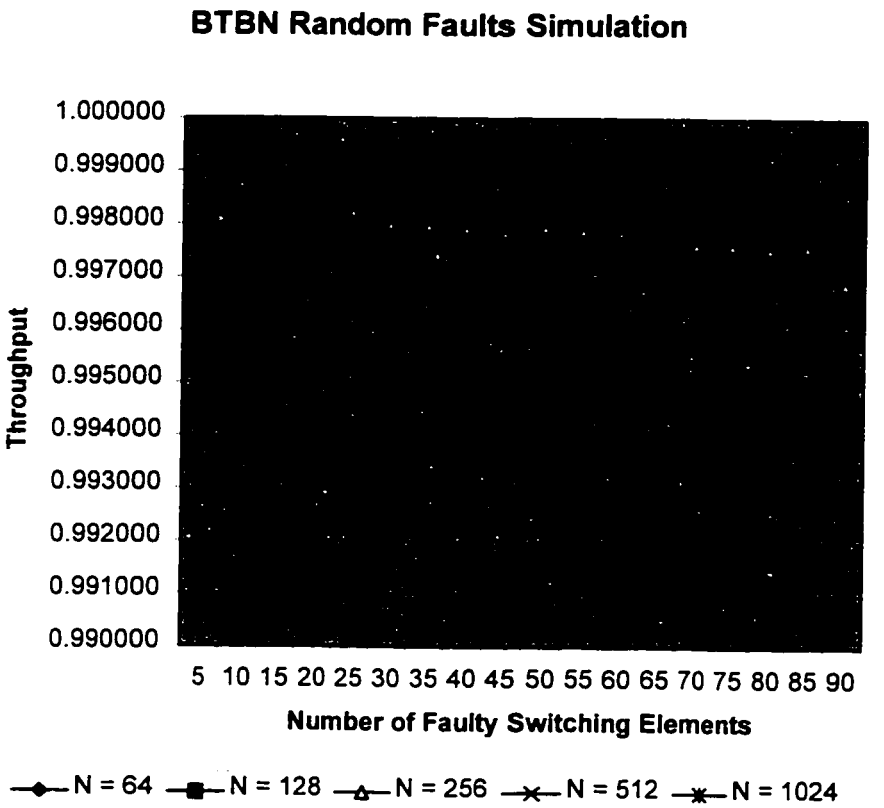


Figure 5.2 Throughput performance under uniform traffic and up to 90 randomly selected faulty switching elements for BTBN networks.

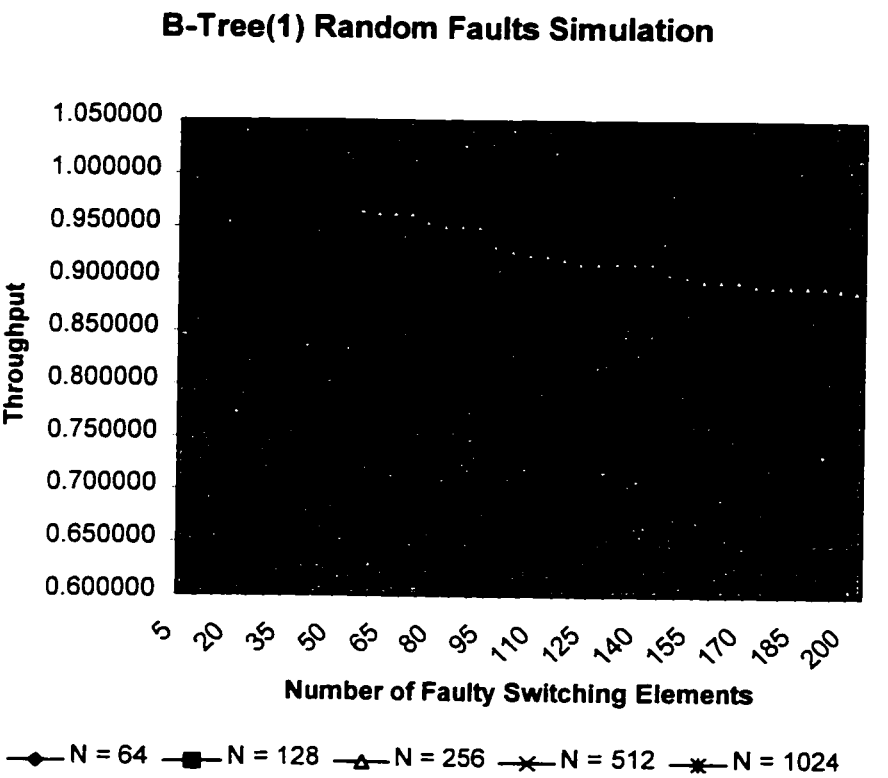


Figure 5.3 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for B-Tree (1) networks.

B-Tree (1) networks of size n = 9 and higher have a very small effect on the throughput performance in the presence of faulty switching elements. As the B-Tee (1) network size increases, the effect of the faulty switching elements decreases as shown in Figure 5.4.

Figure 5.5 shows the results of the design experiment to evaluate the throughput performance of the Tagle and Sharma's networks from the size of n = 6 to n = 10 under uniform traffic and randomly selected faulty switching elements. Tagle and Sharma's networks of size n = 10 and higher have a small effect on the throughput performance in the presence of faulty switching elements. As the Tagle and Sharma's network size increases, the effect of the presence of faulty switching elements decreases as shown in Figure 5.5.

Figure 5.6 shows the results of the design experiment to evaluate the throughput performance of the parallel banyan networks from the size of n = 6 to n = 10 under uniform traffic and randomly selected faulty switching elements. As the parallel banyan network size increases, the effect of the presence of faulty switching elements decreases as shown in Figure 5.6.

Figure 5.7 shows a throughput performance comparison among the switching fabrics of size 64 X 64 networks under uniform traffic and in the presence of faulty switching elements. BTBN network has the least effect on the throughput performance in the presence of faulty switching elements. B-Tree (1) network has better performance than Tagle and Sharma's network and the parallel banyan network.

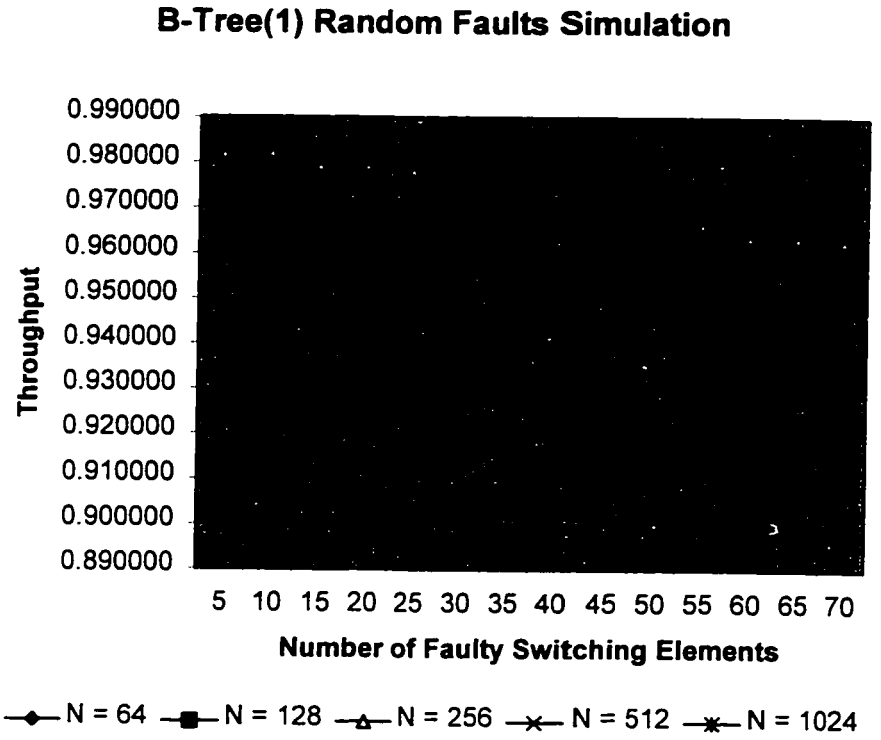


Figure 5.4 Throughput performance under uniform input traffic and up to 70 randomly selected faulty switching elements for B-Tree (1) networks.

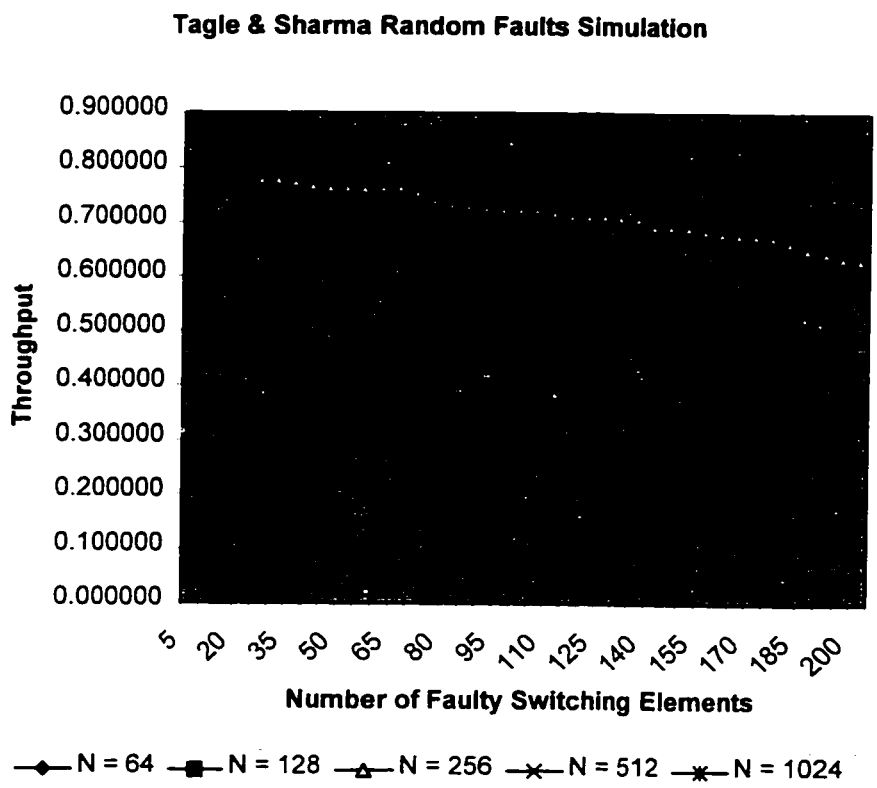


Figure 5.5 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for Tagle and Sharma's networks.

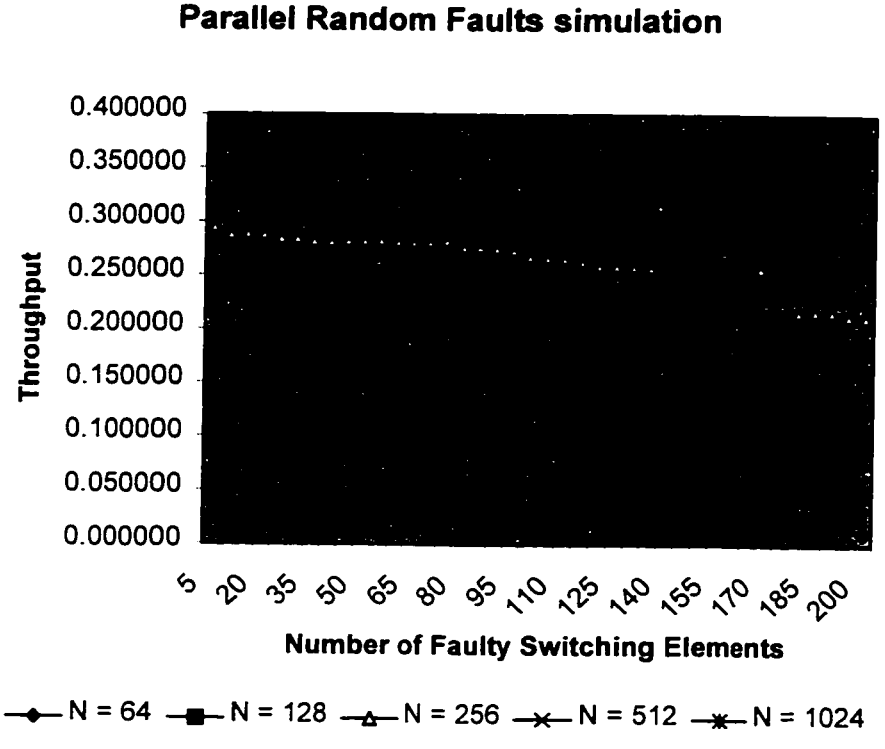


Figure 5.6 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for parallel banyan networks.

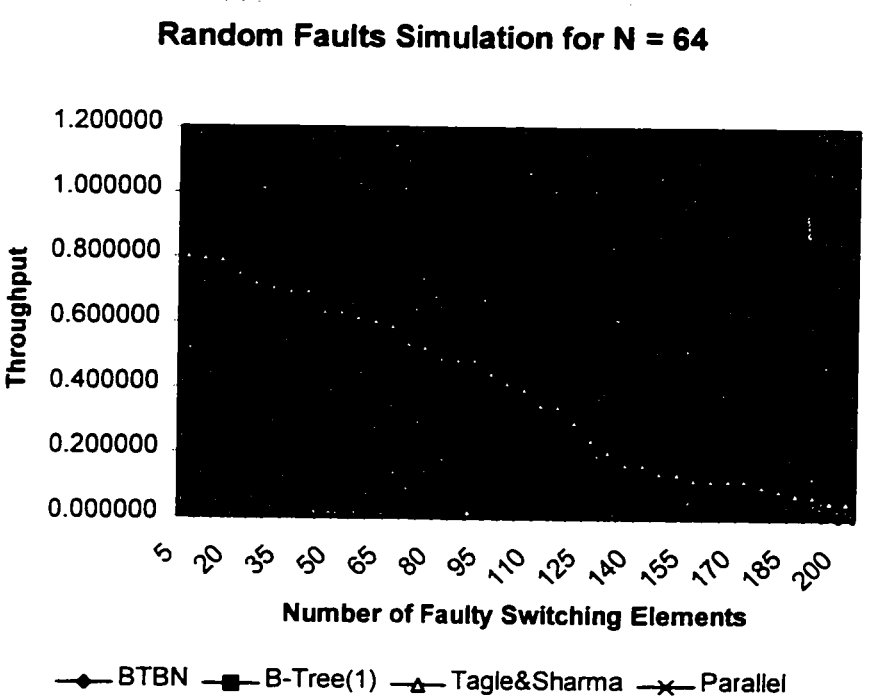


Figure 5.7 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 64 X 64 networks.

The difference on the throughput performance between the BTBN and the B-Tree (1) networks increases as the number of faulty switching element increases. This means the throughput performance of a 64 X 64 B-Tree (1) network is more sensitive than a 64 X 64 BTBN networks in the presence of faulty switching elements.

Figure 5.8 shows a throughput performance comparison among the switching fabrics of size 128 X 128 networks under uniform traffic and in the presence of faulty switching elements. BTBN network has the least effect on the throughput performance with faulty switching elements. B-Tree (1) network has better performance than Tagle and Sharma's network and the parallel banyan network. The difference on the throughput performance e between the BTBN and the B-Tree (1) networks increases fast as the number of faulty switching elements increases as shown in Figure 5.9. This means the throughput performance of a 128 X 128 B-Tree (1) network is more sensitive than a 128 X 128 BTBN networks in the presence of faulty switching elements.

Figure 5.10 shows a throughput performance comparison among the switching fabrics of size 256 X 256 networks under uniform traffic and in the presence of faulty switching elements. It is very clear that BTBN network has the least effect on the throughput performance due to faulty switching elements. B-Tree (1) network has better performance than Tagle and Sharma's network and the parallel banyan network.

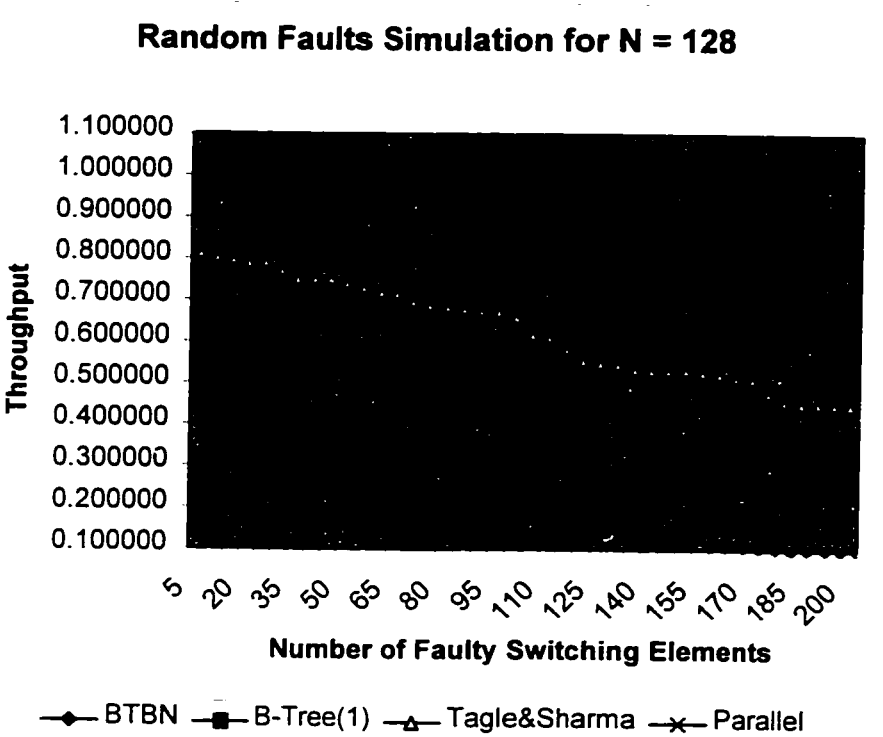


Figure 5.8 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 128 X 128 networks.

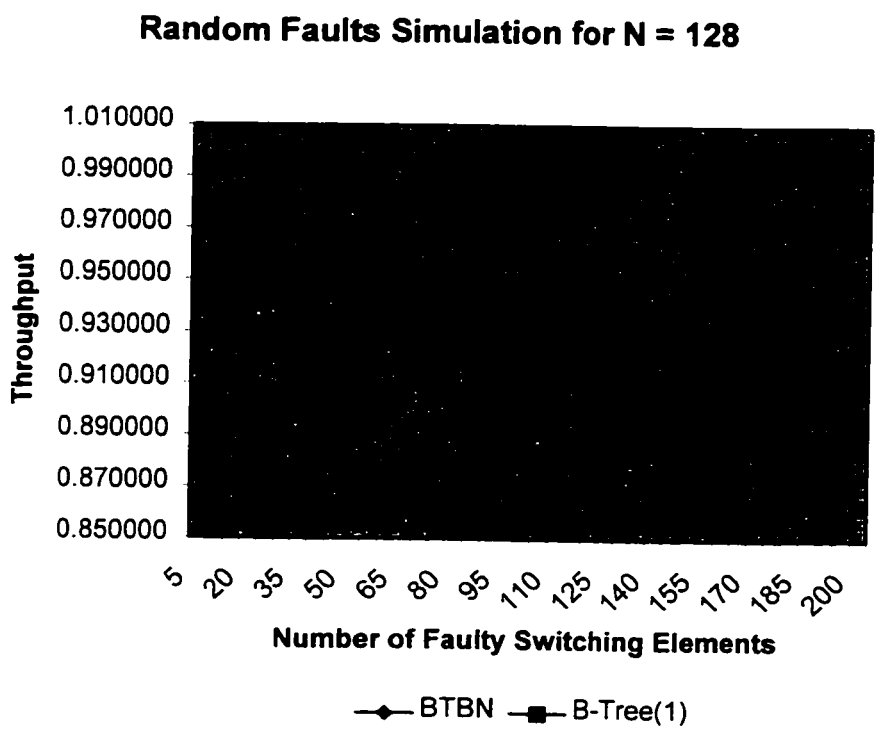


Figure 5.9 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 128 X 128 BTBN and B-Tree (1) networks.

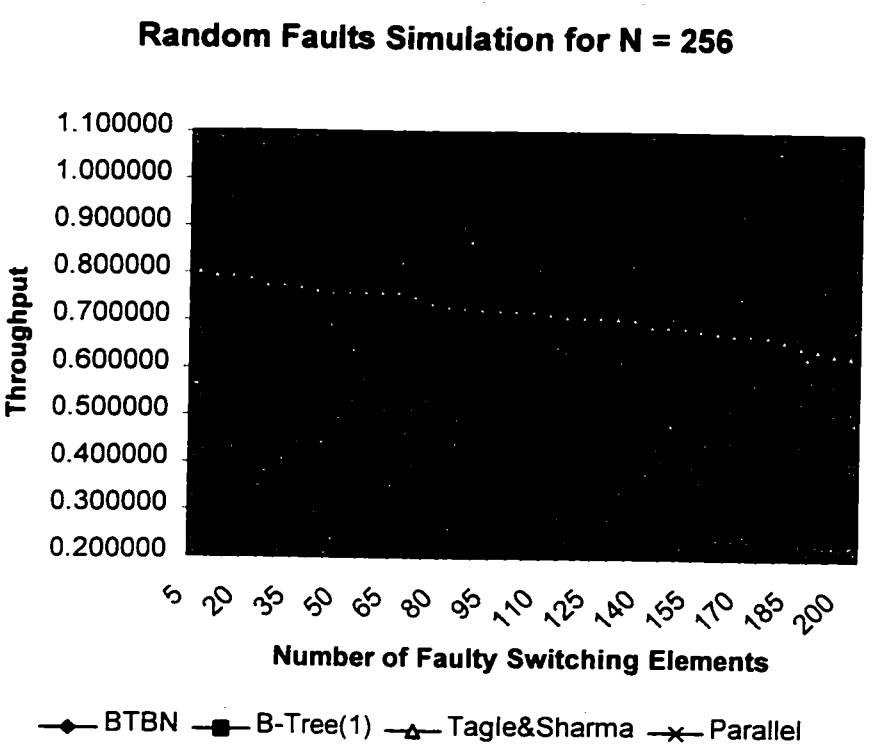


Figure 5.10 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 256 X 256 networks.

The difference on the throughput performance e between the BTBN and the B-Tree (1) networks increases fast as the number of faulty switching element increases as shown in Figure 5.11. This means the throughput performance of a 256 X 256 B-Tree (1) network is more sensitive than a 256 X 256 BTBN networks in the presence of faulty switching elements.

Figure 5.12 shows a throughput performance comparison among the switching fabrics of size 512 X 512 networks under uniform traffic and in the presence of faulty switching elements. BTBN network has the least effect on the throughput performance due to faulty switching elements. B-Tree (1) network has better performance than Tagle and Sharma's network and the parallel banyan network. The difference on the throughput performance e between the BTBN and the B-Tree (1) networks increases as the number of faulty switching element increases as shown in Figure 5.13.

Figure 5.14 shows a throughput performance comparison among the switching fabrics of size 1024 X 1024 networks under uniform traffic and in the presence of faulty switching elements. BTBN network has the least effect on the throughput performance due to faulty switching elements. B-Tree (1) network has better performance than Tagle and Sharma's network and the parallel banyan network. The difference on the throughput performance e between the BTBN and the B-Tree (1) networks increases slowly as the number of faulty switching element increases as shown in Figure 5.15.

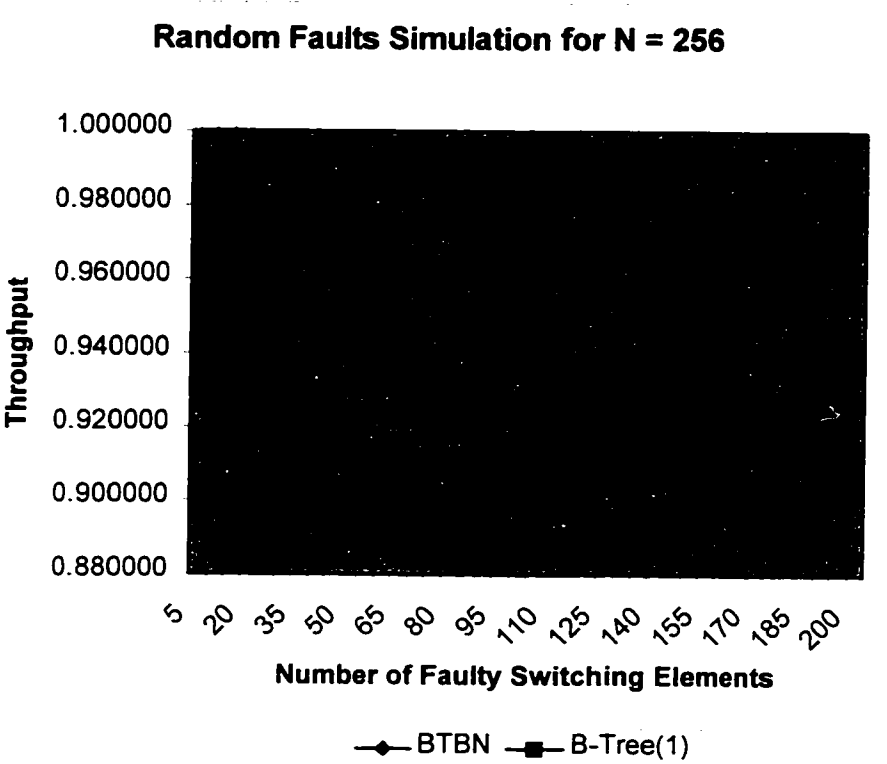


Figure 5.11 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 256 X 256 BTBN and B-Tree (1) networks.

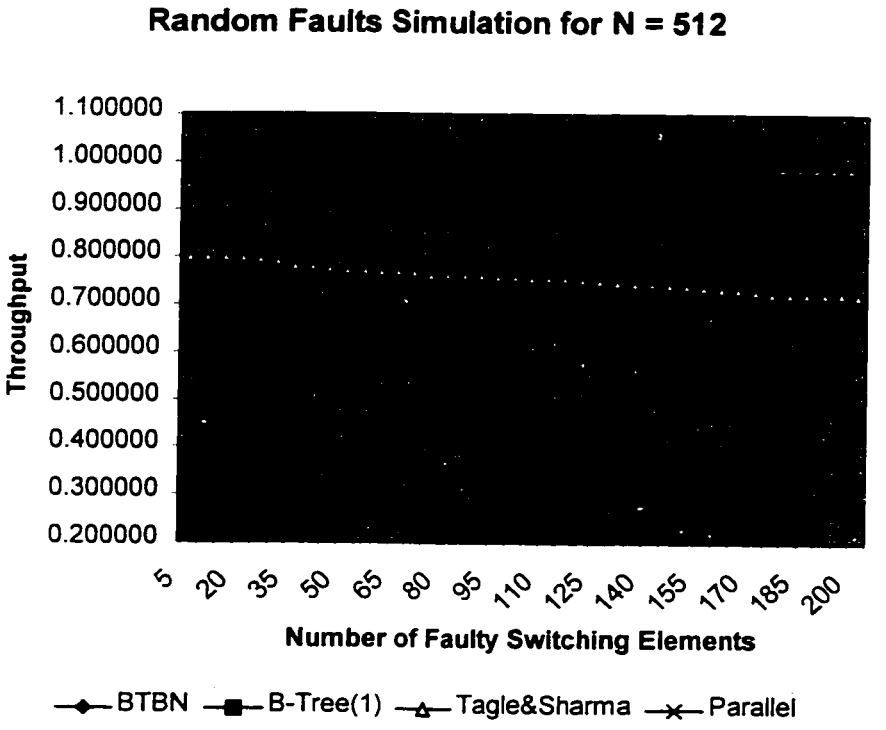


Figure 5.12 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 512 X 512 networks.

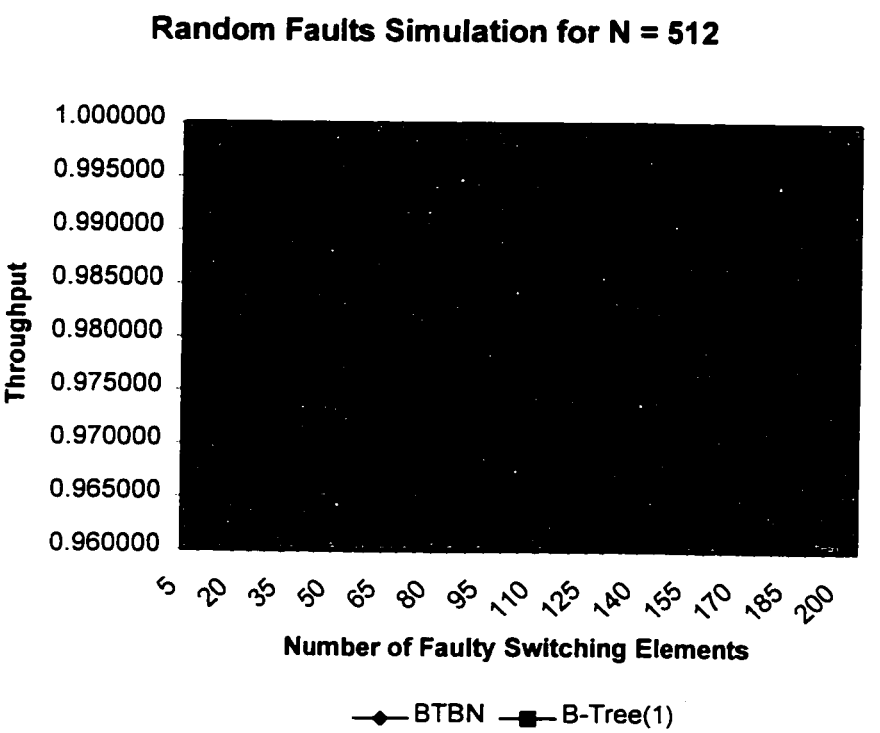


Figure 5.13 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 512 X 512 BTBN and B-Tree (1) networks.

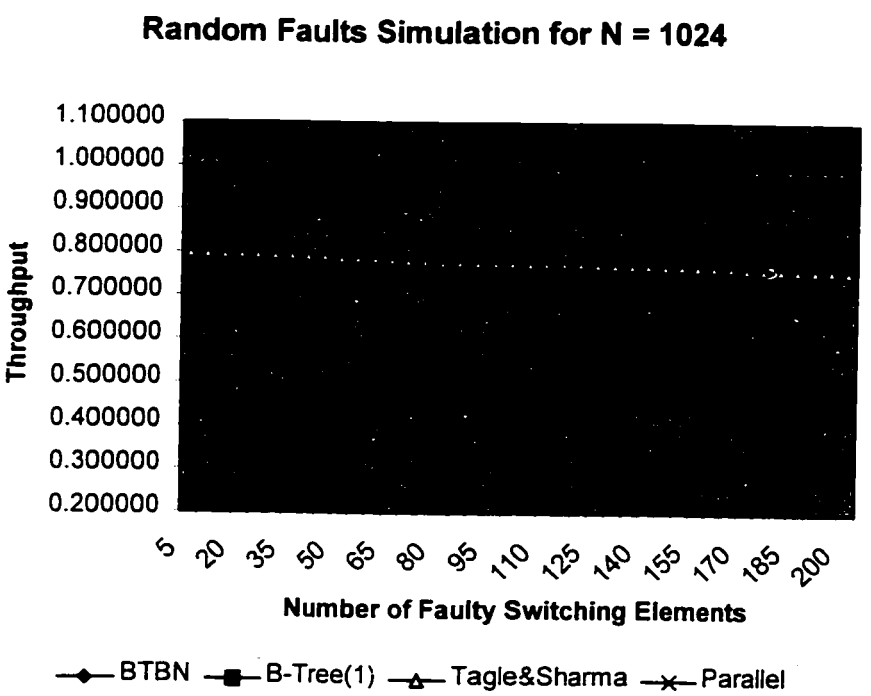


Figure 5.14 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 1024 X 1024 networks.

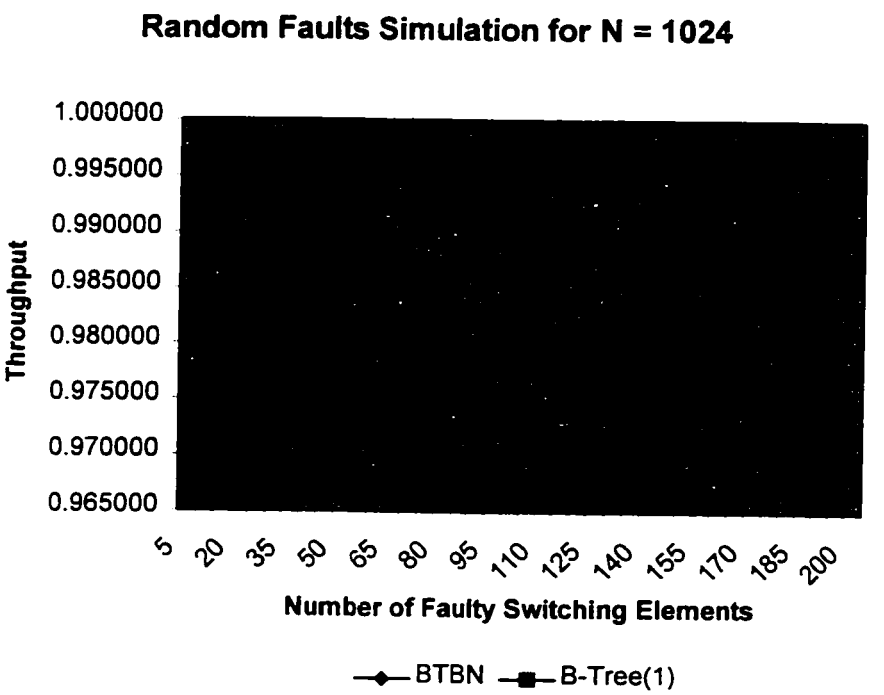


Figure 5.15 Throughput performance under uniform traffic and up to 200 randomly selected faulty switching elements for 1024 X 1024 BTBN and B-Tree (1) networks.

The throughput performance of the switching fabrics in the presence of faulty switching elements is evaluated using the second method. This method demands a lot of simulation time. The design experiments are conducted for fault-free conditions, single-fault simulation, and double-fault simulation. Figure 5.16 and Table 5.1 show the results of the design experiment to evaluate the throughput performance of the BTBN networks from the size of n = 3 to n = 10 under 100% hot spot traffic and fault-free conditions, single faulty switching element, and double faulty switching elements. The throughput performance degradation in the fault free conditions is due mainly to output contention. This result would be the upper boundary of the throughput performance in the presence of faulty switching elements. Figure 5.17 and Table 5.2 show the results of the design experiment to evaluate the throughput performance of the B-Tree (1) networks from the size of n = 3 to n = 10 under 100% hot spot traffic and fault-free conditions, single faulty switching element, and double faulty switching elements. Figure 5.18 and Table 5.3 show the results of the design experiment to evaluate the throughput performance of the Tagle and Sharma's networks from the size of n = 3 to n = 10 under 100% hot spot traffic and fault-free conditions, single faulty switching element, and double faulty switching elements. Figure 5.19 and Table 5.4 show the results of the design experiment to evaluate the throughput performance of the parallel banyan networks from the size of n = 3 to n = 10 under 100% hot spot traffic and fault-free conditions, single faulty switching element, and double faulty switching elements.

Table 5.1 Throughput performance under 100% hot spot traffic and single / double faults simulation for BTBN networks.

Network Size	Fault-Free	Single-Fault	Double-Fault
n=3	0.750000000	0.785714260	0.583333333
n=4	0.625000000	0.604166667	0.573529412
n=5	0.375000000	0.368902439	0.362347561
n=6	0.218750000	0.217684659	0.216570337
n=7	0.125000000	0.124658470	0.124311310
n=8	0.070312500	0.070249833	0.070186494
n=9	0.039062500	0.039041859	0.039021137
n=10	0.021484375	0.021480530	0.021476675

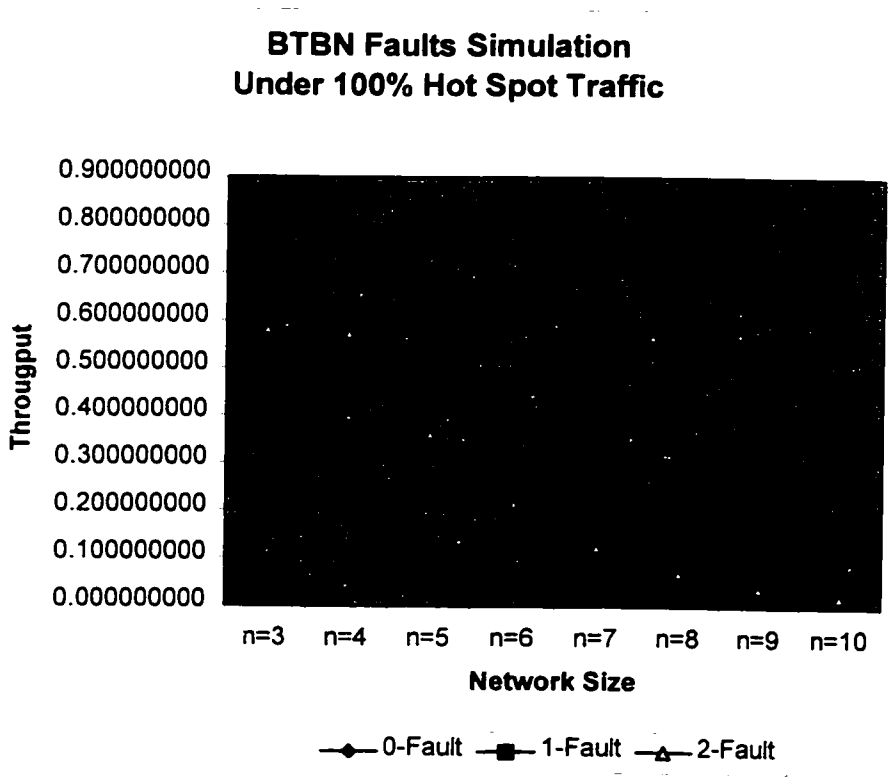


Figure 5.16 Throughput performance under 100% hot spot traffic and single / double faults simulation for BTBN networks.

Table 5.2 Throughput performance under 100% hot spot traffic and single / double faults simulation for B-Tree (1) networks.

Network Size	Fault-Free	Single-Fault	Double-Fault
n=2	1.000000000	0.857142857	0.666666667
n=3	0.750000000	0.694444444	0.645424837
n=4	0.500000000	0.481707317	0.464634146
n=5	0.312500000	0.307528409	0.302867032
n=6	0.187500000	0.185450820	0.183501096
n=7	0.109375000	0.108581217	0.107825291
n=8	0.062500000	0.062128468	0.061773126
n=9	0.035156000	0.034984519	0.034820222
n=10	0.019531000	0.019446970	0.019366245

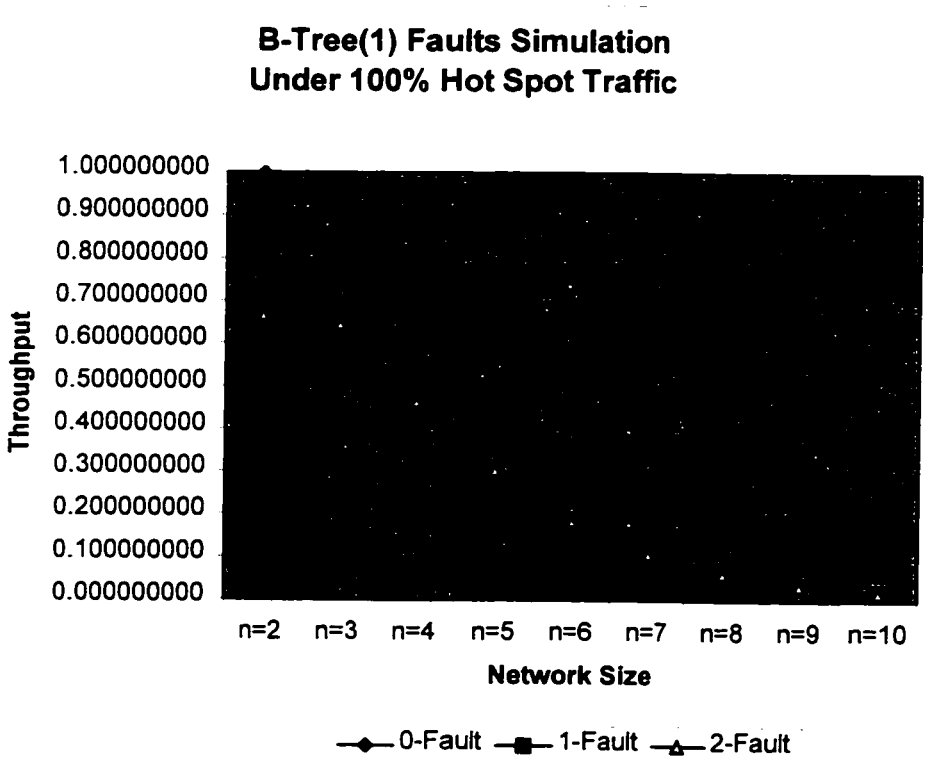


Figure 5.17 Throughput performance under 100% hot spot traffic and single / double faults simulation for B-Tree (1) networks.

Table 5.3 Throughput performance under 100% hot spot traffic and single / double faults simulation for Tagle and Sharma's networks.

Network Size	Fault-Free	Single-Fault	Double-Fault
n=2	0.500000000	0.416666667	0.333333333
n=3	0.250000000	0.232142857	0.214285714
n=4	0.125000000	0.120833333	0.116666666
n=5	0.062500000	0.061491935	0.060483871
n=6	0.031250000	0.031001984	0.030753968
n=7	0.015625000	0.015563484	0.015501969
n=8	0.007813000	0.007797181	0.007781863
n=9	0.003906000	0.003902428	0.003898606
n=10	0.001953000	0.001952170	0.001951200

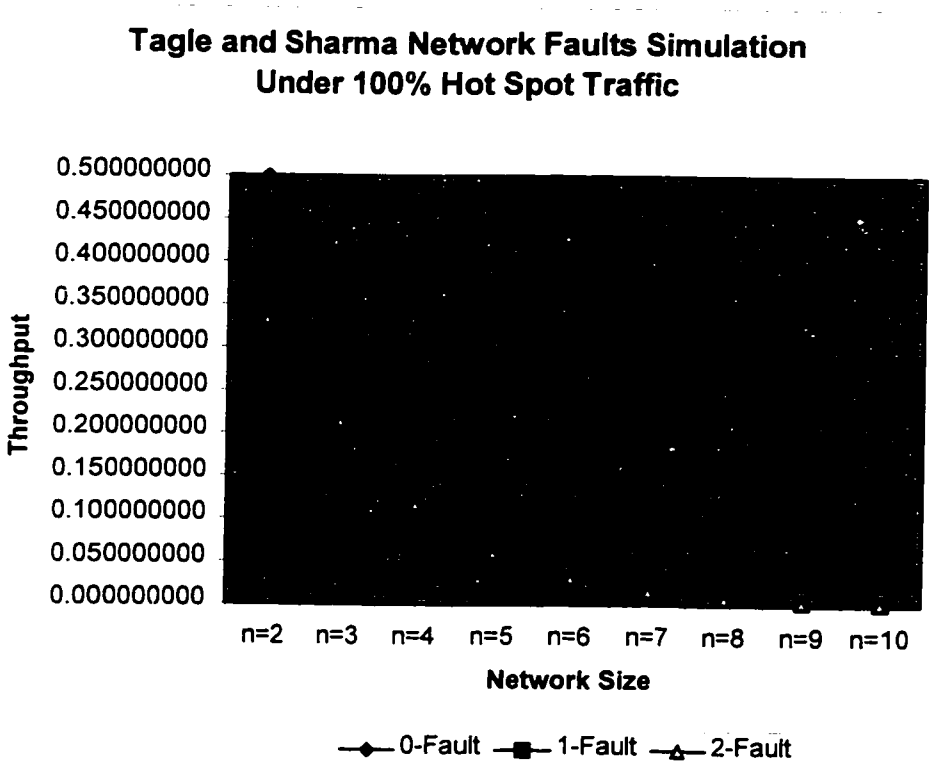


Figure 5.18 Throughput performance under 100% hot spot traffic and single / double faults simulation for Tagle and Sham's networks.

Table 5.4 Throughput performance under 100% hot spot traffic and single / double faults simulation for parallel banyan networks.

Network Size	Fault-Free	Single-Fault	Double-Fault
n=2	0.250000000	0.291666667	0.233333333
n=3	0.125000000	0.151785714	0.153846154
n=4	0.062500000	0.077083333	0.083333333
n=5	0.031250000	0.038810484	0.043164992
n=6	0.015625000	0.019469246	0.021976190
n=7	0.007813000	0.009750246	0.011096177
n=8	0.003906000	0.004878983	0.005578147
n=9	0.001953000	0.002440451	0.002797434
n=10	0.000977000	0.001220464	0.001401030

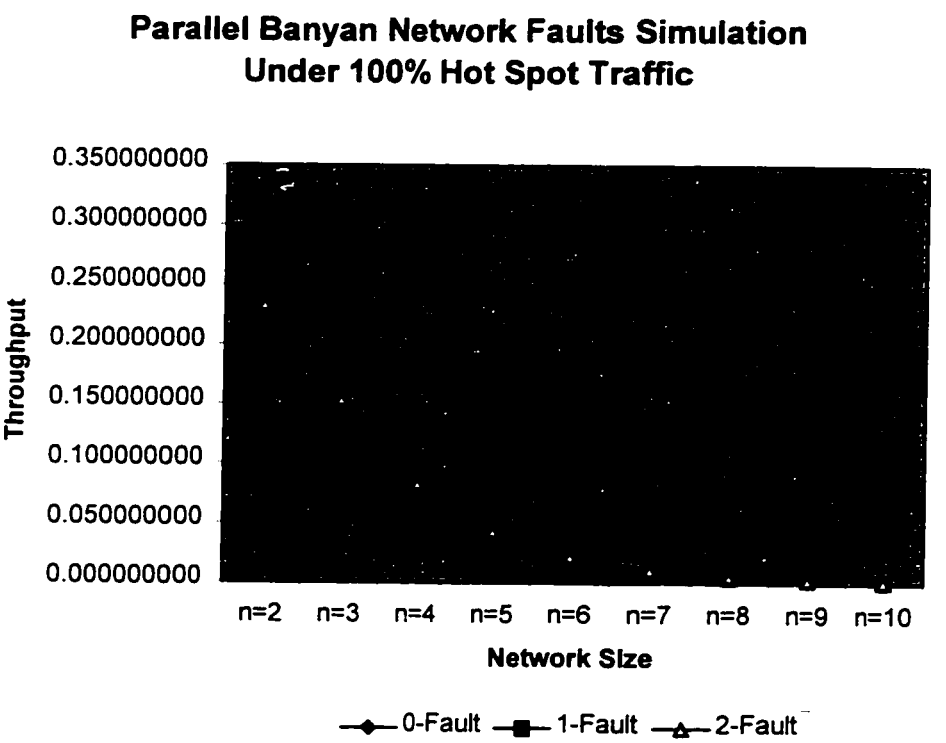


Figure 5.19 Throughput performance under 100% hot spot traffic and single / double faults simulation for parallel banyan networks.

The throughput performance is better for single or double faults since the second banyan network in standby mode routes some cells that failed to go through the main banyan network because of faulty switching elements. The throughput performance starts to decrease in the presence of more than two faulty switching elements in the parallel banyan network.

Figure 5.20 shows the percentage of throughput degradation under 100% hot spot traffic and in the presence of faulty switching elements in BTBN, B-Tree (1), and Tagle and Sharma's networks. BTBN has the smallest throughput performance degradation. Figure 5.21 shows the throughput performance improvement under 100% hot spot traffic and in the presence of faulty switching elements in the parallel banyan network. This improvement is due to the utilization of the standby network in the presence of faulty switching elements in the first stage of the main network.

Figure 5.22 and Table 5.5 show the throughput performance under 100% hot spot traffic for the switching fabrics. BTBN has the highest throughput because of large number of output access links for each output port. Figure 5.23 and Table 5.6 show the throughput performance under 100% hot spot traffic and in the presence of a single faulty switching element for the switching fabrics. BTBN has the highest throughput because of the huge number of redundant paths. Figure 5.24 and Table 5.7 show the throughput performance under 100% hot spot traffic and in the presence of double faulty switching elements for the switching fabrics. BTBN has the highest throughput performance.

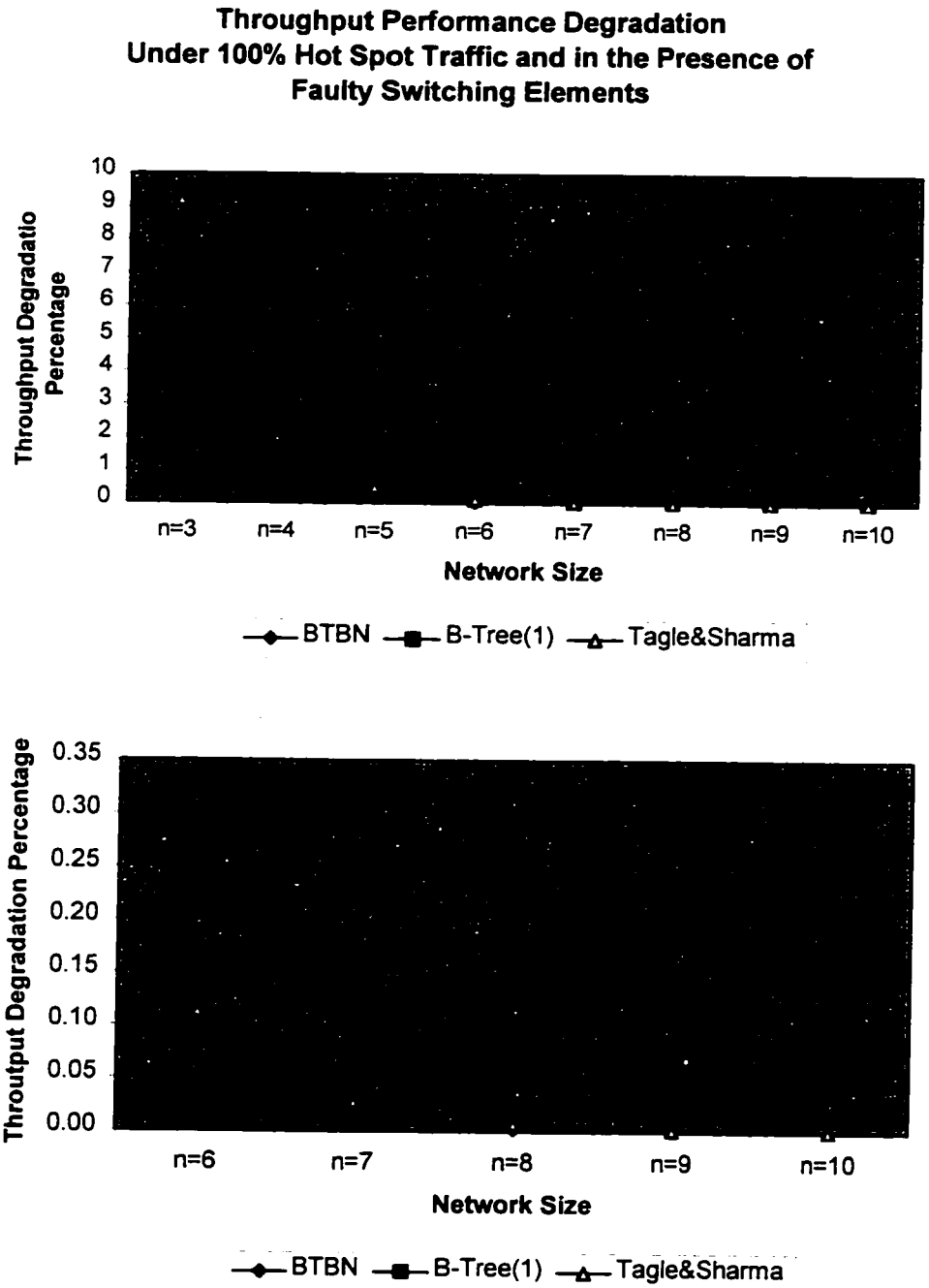


Figure 5.20 Throughput performance degradation under 100% hot spot traffic and faulty switching elements for switching fabric's networks.

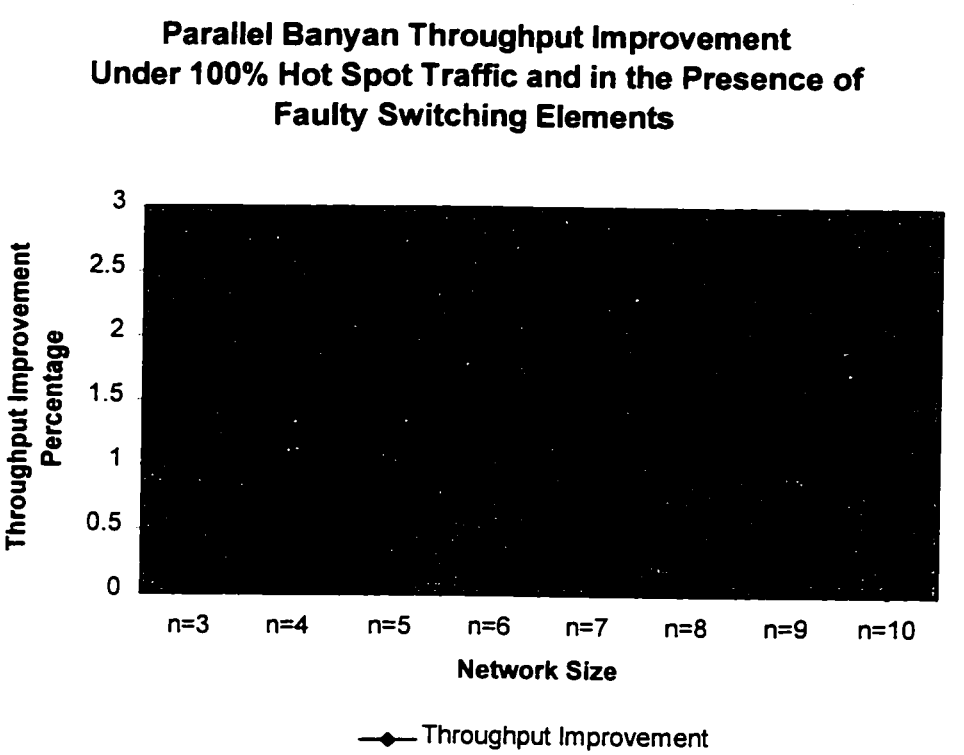


Figure 5.21 Throughput performance improvement under 100% hot spot traffic and faulty switching elements for parallel banyan networks.

Table 5.5 Throughput performance under 100% hot spot traffic simulation for switching fabric's networks.

Network Size	BTBN	B-Tree(1)	Tagle&Sharma	Parallel
n=3	0.750000000	0.750000000	0.250000000	0.125000000
n=4	0.625000000	0.500000000	0.125000000	0.062500000
n=5	0.375000000	0.312500000	0.062500000	0.031250000
n=6	0.218750000	0.187500000	0.031250000	0.015625000
n=7	0.125000000	0.109375000	0.015625000	0.007813000
n=8	0.070312500	0.062500000	0.007813000	0.003906000
n=9	0.039062500	0.035156000	0.003906000	0.001953000
n=10	0.021484375	0.019531000	0.001953000	0.000977000

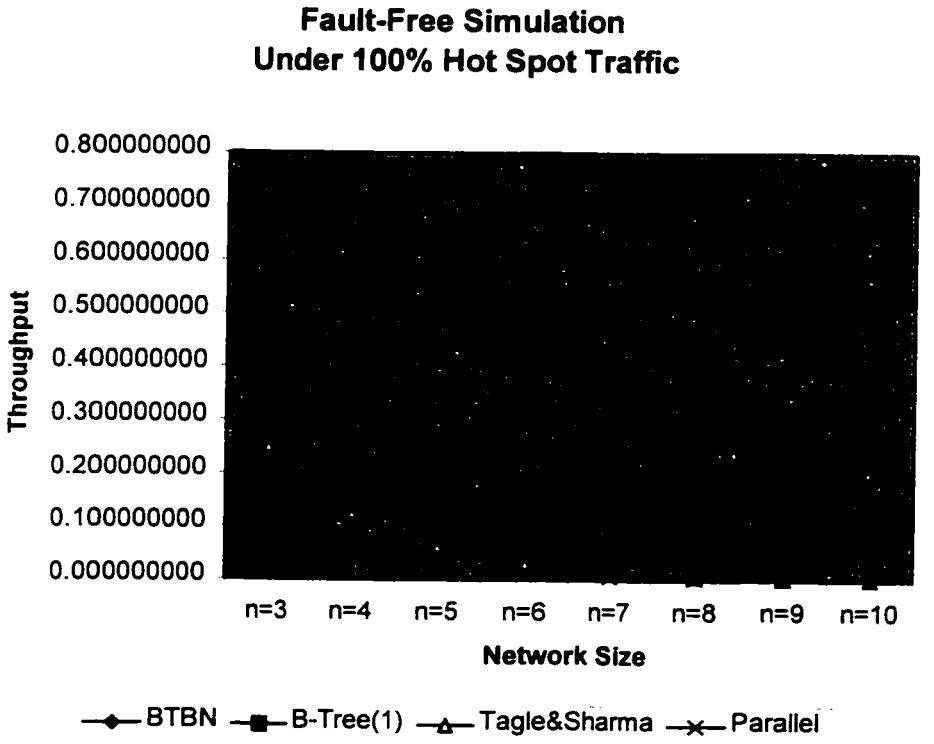


Figure 5.22 Throughput performance under 100% hot spot traffic simulation for switching fabric's networks.

Table 5.6 Throughput performance under 100% hot spot traffic and single fault simulation for switching fabric's networks.

Network Size	BTBN	B-Tree(1)	Tagle&Sharma	Parallel
n=3	0.785714260	0.694444444	0.232142857	0.151785714
n=4	0.604166667	0.481707317	0.120833333	0.077083333
n=5	0.368902439	0.307528409	0.061491935	0.038810484
n=6	0.217684659	0.185450820	0.031001984	0.019469246
n=7	0.124658470	0.108581217	0.015563484	0.009750246
n=8	0.070249833	0.062128468	0.007797181	0.004878983
n=9	0.039041859	0.034984519	0.003902428	0.002440451
n=10	0.021480530	0.019446970	0.001952170	0.001220464

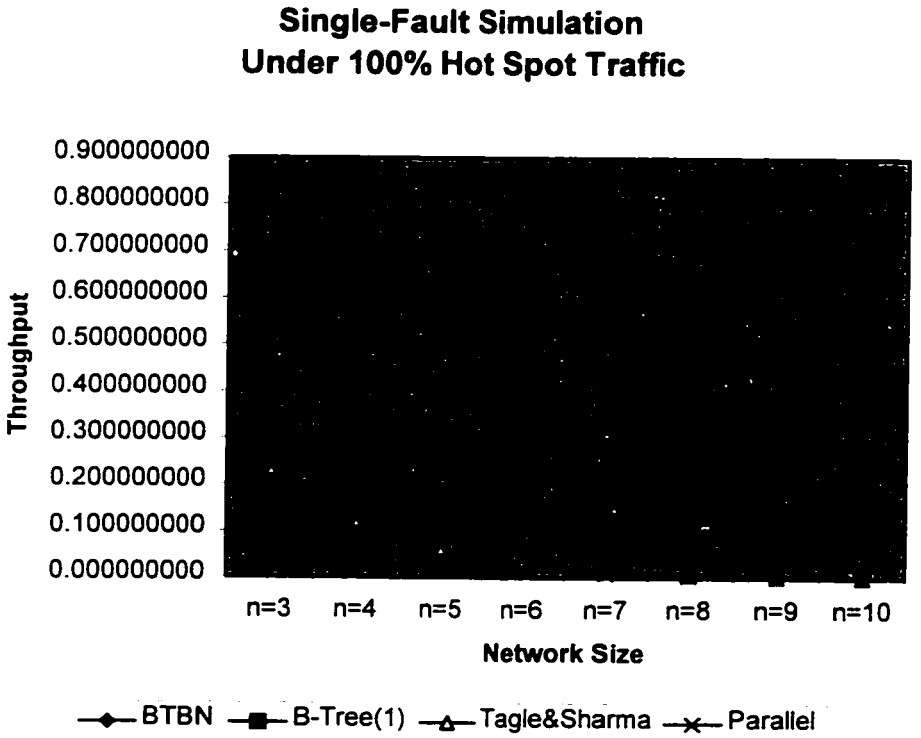


Figure 5.23 Throughput performance under 100% hot spot traffic and single fault simulation for switching fabric's networks.

Table 5.7 Throughput performance under 100% hot spot traffic and double fault simulation for switching fabric's networks.

Network Size	BTBN	B-Tree(1)	Tagle&Sharma	Parallel
n=3	0.583333333	0.645424837	0.214285714	0.153846154
n=4	0.573529412	0.464634146	0.116666666	0.083333333
n=5	0.362347561	0.302867032	0.060483871	0.043164992
n=6	0.216570337	0.183501096	0.030753968	0.021976190
n=7	0.124311310	0.107825291	0.015501969	0.011096177
n=8	0.070186494	0.061773126	0.007781863	0.005578147
n=9	0.039021137	0.034820222	0.003898606	0.002797434
n=10	0.021476675	0.019366245	0.001951200	0.001401030

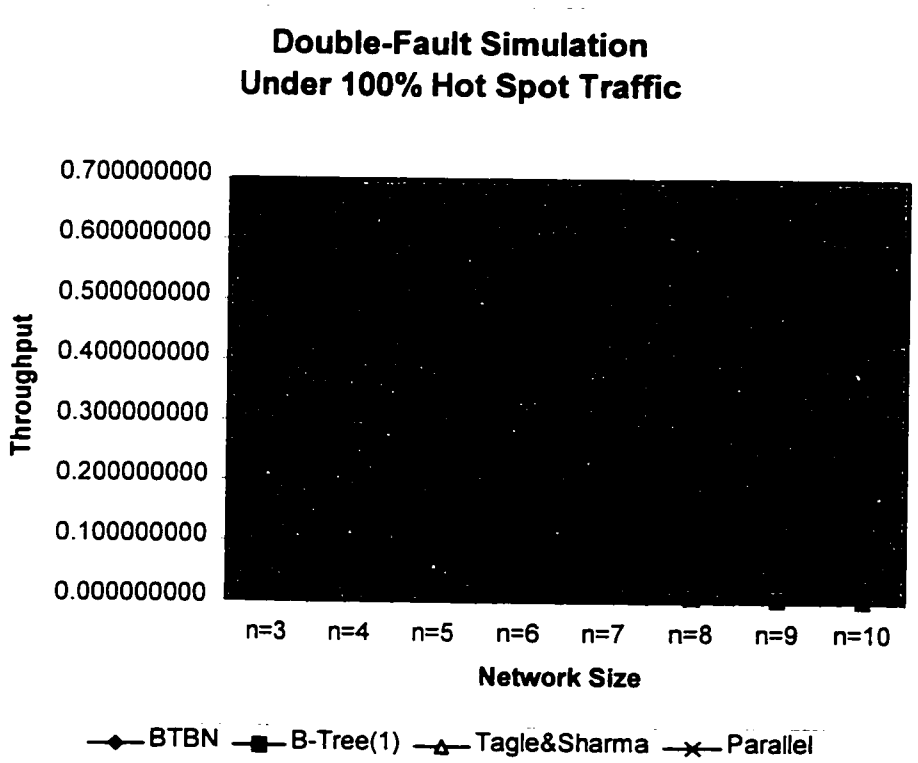


Figure 5.24 Throughput performance under 100% hot spot traffic and double fault simulation for switching fabric's networks.

Chapter 6

BTBN RELIABILITY ANALYSIS

The terminal reliability of a MIN is the probability of having at least one available path between any source to any destination. The terminal reliability is calculated using the combinatorial series and parallel models assuming that the failure probabilities of different components of the system are independent. It is also assumed that the interconnection links, input and out ports, demultiplexers directly connected to the input ports, and multiplexers directly connected to output ports never fail. No cell can be routed through a faulty switching element. Any link connected to a faulty switching element is useless. Under the above conditions, the following formulas describing the terminal reliability of the switching fabrics are developed.

Sometimes a "success" diagram is used to describe the operational modes of a switching fabric from an input source to an output destination. Figure 6.1a shows a

success diagram for an 8 X 8 B-Tree (1) network. If the success diagram becomes too complex to evaluate exactly, upper-limit approximation on the switching fabric terminal reliability can be used. An upper bound on terminal reliability is $R_{Switching-Fabric} \leq 1 - \prod_{i=1}^{i=\#RP}(1 - R_{path-i})$ where RP is the number of redundant paths available from any input source to any output destination and R_{path-i} is the serial reliability of path-i. The upper bound on terminal reliability is calculated as if all paths were in parallel. This calculation is an upper bound because the paths are not independent. That is the failure of a single switching element affects more than one path. Therefore, this approximation gets closer to the actual terminal reliability when the terminal reliability of a path is small. Placing the paths in parallel yields a reliability block diagram (RBD). Figure 6.1b shows the RBD for the success diagram of the 8 X 8 B-Tree (1) network shown in Figure 6.1a. Using the combinatorial series and parallel models, the upper bound on terminal reliability of an 8 X 8 B-Tree (1) is calculated as follows $R_{8X8B-Tree(1)} \leq 1 - \prod_{i=1}^{i=8}(1 - R_{path-i})$ where $R_{path-i} = R_{4X4SE}^3$ assuming that $R_A = R_B = R_C = R_D = R_E = R_F = R_G = R_H = R_I = R_{4X4SE}$. Therefore, the upper bound on terminal reliability of an 8 X 8 B-Tree (1) $= R_{8X8B-Tree(1)} \leq 1 - (1 - R_{4x4SE}^3)^8$. In general, the upper bound terminal reliability of B-Tree (1) is calculated as follows $R_{NXNB-Tree(1)} \leq 1 - (1 - R_{4x4SE}^n)^N$ where $n = \log_2 N$. This method would be used to calculate the upper bound terminal reliability of the remaining switching fabrics.

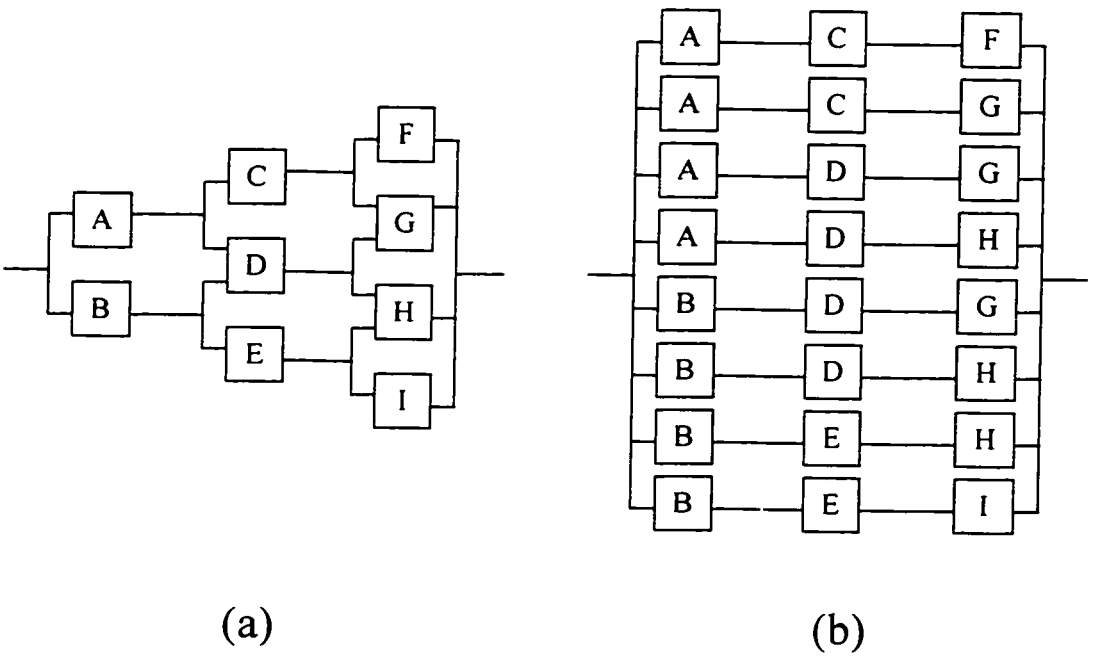


Figure 6.1 8 X 8 B-Tree (1): (a) success diagram; and (b) RBD

The upper bound reliability of the parallel banyan network is equal the actual terminal reliability since there are only two independent redundant paths between any input source and any output destination. The reliability block diagram of an 8 X 8 parallel banyan network is shown in Figure 6.2. Therefore, the parallel banyan network terminal reliability $= 1-(1-R_{path1})(1-R_{path2})$. The switching elements along each path have to work perfectly in order to establish the input to output connection. Therefore, the terminal reliability of each path = $R_{path1} = R_{path2} = R_{2X2}^{n}$ where $n = \log_2 N$. Then, the terminal reliability of parallel banyan network = $2R_{2X2}^{n} - R_{2X2}^{2n}$. For an 8 X 8 parallel banyan network, the terminal reliability = $2R_{2X2}^{3} - R_{2X2}^{6}$.

The success diagram of an 8 X 8 Tagle and Sharma's network is shown in Figure 6.3a. As shown in Figure 6.3a, each input cell has two different paths to select from in order to propagate to the next stage. The reliability block diagram of an 8 X 8 Tagle and Sharma's network is shown in Figure 6.3b. As a result, the upper bound on the terminal reliability of the Tagle and Sharma's network is the same as that of B-Tree(1) network since both have the same number of redundant paths and each path uses the same number of 4 X 4 switching elements. Therefore, with the same assumption used in calculating B-Tree(1) terminal reliability, $R_{8X8Tagle\&Sharma} \le 1-(1-R_{4x4SE}^{3})^{8}$.

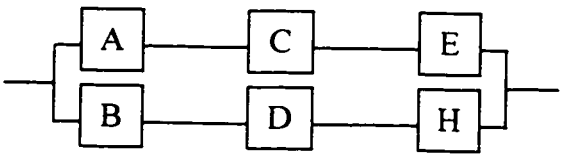


Figure 6.2 RDB of an 8 X 8 parallel banyan network

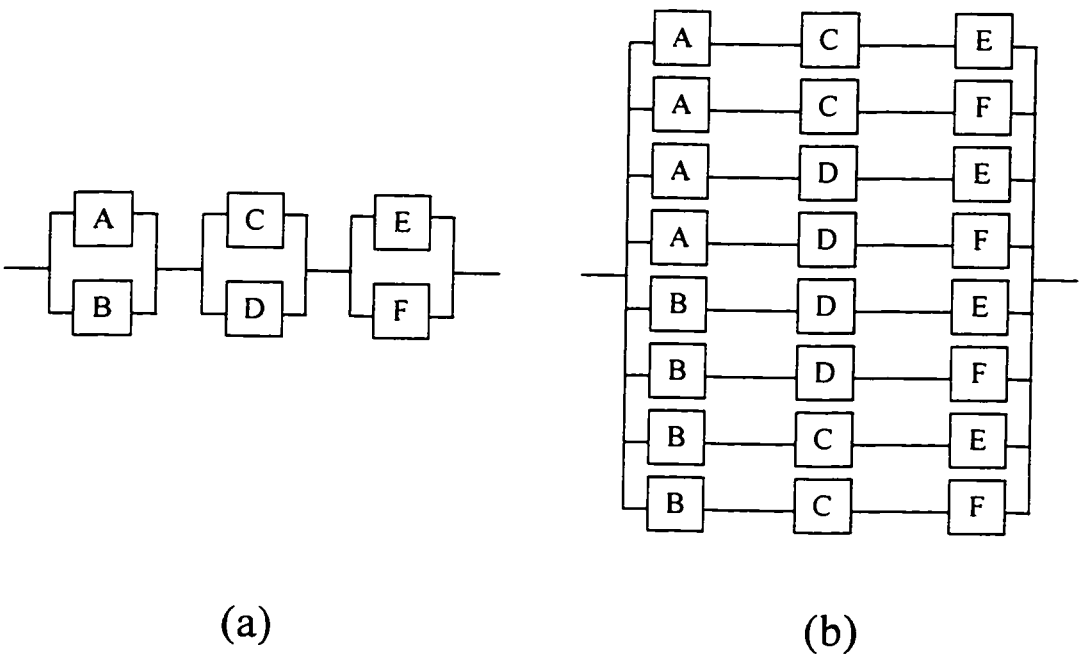


Figure 6.3 8 X 8 Tagle & Sharma's network: (a) success diagram; and (b) RBD

In general, the upper bound on terminal reliability of Tagle and Sharma's network is equal to $R_{NXNTagle\&Sharma} \le 1-(1-R_{4x4SE}^{n})^{N}$ where $n = \log_2 N$.

The success diagram of an 8 X 8 BTBN network is shown in Figure 6.4a. The reliability block diagram of an 8 X 8 BTBN network is shown in Figure 6.4b. Therefore, the upper bound on the terminal reliability of an 8 X 8 BTBN network is equal to $R_{8X8BTBN} \le 1-(1-R_{4x6SE}^{2})^{4}(1-R_{4x6SE}R_{4x4SE})^{2}$ assuming that $R_D = R_{4X4SE}$ and $R_A = R_B = R_C = R_E = R_{4X6SE}$. The upper bound on terminal reliability of a 4 X 4 BTBN network is equal the actual terminal reliability since there are only two independent redundant paths between any input source and any output destination. Therefore, $R_{4X4BTBN} \le 1-(1-R_{4x6SE})^{2}$. In general, the upper bound on terminal reliability is calculated as follows

$$R_{NXNBTBN} \le 1-(1-R_{4x6SE}^{n-1})^{2^{n-1}}(1-R_{4x6SE}^{n-2}R_{4X4SE})^{2^{n-2}}(1-R_{4x6SE}^{n-3}R_{4X4SE}^{2})^{2^{n-2}}\cdots$$
$$(1-R_{4x6SE}^{2}R_{4X4SE}^{n-3})^{2^{n-2}}(1-R_{4X6SE}R_{4X4SE}^{n-2})^{2^{n-2}}$$

where $n = \log_2 N \ge 3$.

The switching fabrics use different size of switching elements. The parallel banyan network uses 2 X 2 switching elements, the Tagle and Shama's and the B-Tree (1) networks use 4 X 4 switching elements, and the BTBN network uses two types 4X4 and 4 X 6 switching elements. To compare the terminal reliability of the switching fabrics, a relationship among these different size-switching elements has to be developed. There are two methods to develop this relation.

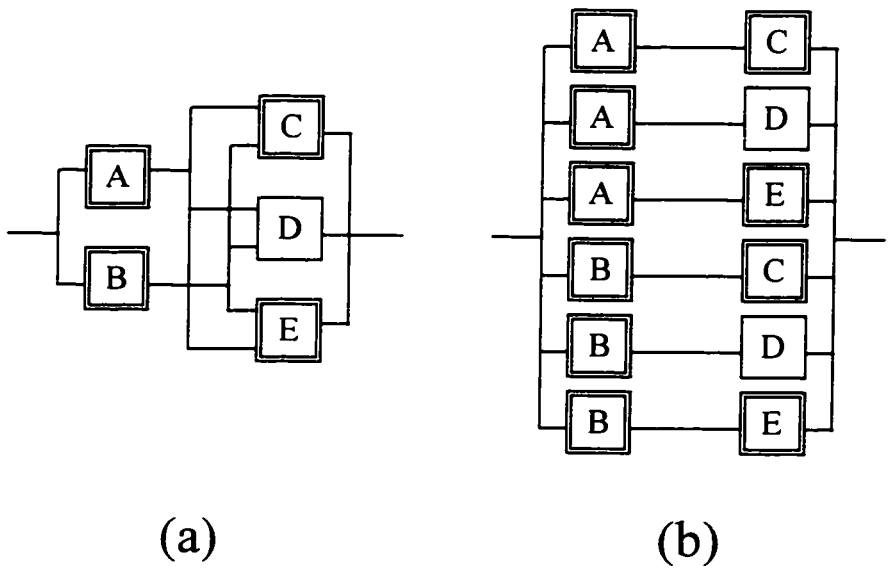


Figure 6.4 8 X 8 BTBN: (a) success diagram; and (b) RBD

The first method is to assume that all switching elements are crossbar switches and all cross-point switches and the associated interconnection links in each crossbar switch must function correctly. A single failure of one cross-point switch results in a failure of the witching element. For example, a 2 X 2 switching element has total of four cross-point switches. A single failure on these four cross-point switches makes the switching element faulty. Thus, the reliability of the 2 X 2 switching element $(R_{2X2}) = R_{X-Point}^{4}$ where $R_{X-Point}$ is the reliability of a single cross-point switch and its associated interconnection links. Therefore, $R_{4X4} = R_{X-Point}^{16}$ and $R_{4X6} = R_{X-Point}^{24}$. For example, if $R_{X-Point} = 0.964$, then $R_{2X2} = 0.863591$, $R_{4X4} = 0.556202$, and $R_{4X6} = 0.41481$.

The second method is to assume that all switching elements are crossbar switches and the switching element reliability is determined by the yield equation $e^{-\sqrt{Ad}}$ where d is the number of defects and A is the total area for the cross-point switches and their interconnection links [71]. Assuming that a single cross-point switch and its interconnection links have an area of a, then

$$R_{2X2} = e^{-\sqrt{4ad}} = e^{-2\sqrt{ad}},$$

$$R_{4X4} = e^{-\sqrt{4^2 ad}} = e^{-4\sqrt{ad}} = (e^{-2\sqrt{ad}})^2 = R_{2X2}^2, \text{ and}$$

$$R_{4X6} = R_{2X2}^{\sqrt{24}/2}.$$

Assuming $R_{2X2} = 0.8$, then $R_{4X4} = 0.6724$, and $R_{4X6} = 0.615018$.

Figure 6.5 and Table 6.1 show the terminal reliability of the switching fabrics using the first method. Figure 6.6 and Table 6.2 show the terminal reliability of the switching fabrics using the second method. In both cases the reliability of the parallel banyan network decreases as the network size increases. However, the upper bound on terminal reliability of BTBN, B-Tree (1), and Tagle and Sharma's networks increase as the network size increases. This is due to the increase on the number of redundant paths. Using both methods, the upper bound on terminal reliability of BTBN is better than that of B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

Table 6.1 Terminal reliability of switching fabrics using first method.

Network Size	Parallel	Tagle&Sharma	B-Tree(1)	BTBN
n=2	0.935377	0.772488	0.772488	0.657552
n=3	0.873305	0.779219	0.779219	0.721931
n=4	0.803043	0.800028	0.800028	0.786508
n=5	0.729944	0.826295	0.826295	0.840376
n=6	0.657552	0.853901	0.853901	0.883208
n=7	0.588126	0.880613	0.880613	0.916419
n=8	0.523017	0.905161	0.905161	0.941651
n=9	0.462947	0.926832	0.926832	0.960413
n=10	0.408205	0.945286	0.945286	0.974016

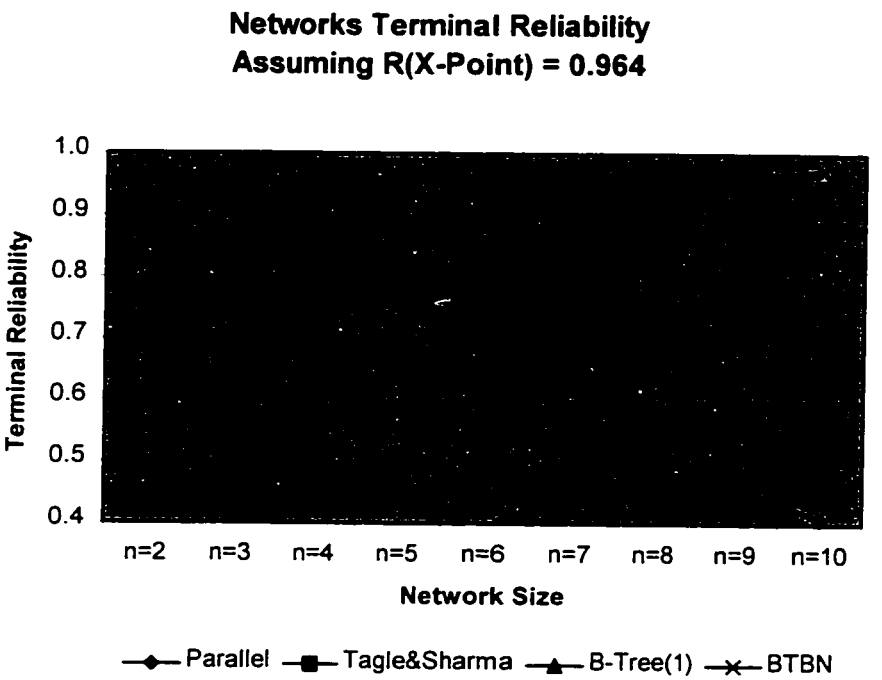


Figure 6.5 Terminal reliability of switching fabrics using first method.

Table 6.2 Terminal reliability of switching fabrics using second method.

Network Size	Parallel	Tagle&Sharma	B-Tree(1)	BTBN
n=2	0.870400	0.878497	0.878497	0.822693
n=3	0.761856	0.912144	0.912144	0.922577
n=4	0.651428	0.947049	0.947049	0.977040
n=5	0.547986	0.973611	0.973611	0.995768
n=6	0.455569	0.989501	0.989501	0.999580
n=7	0.375450	0.996840	0.996840	0.999982
n=8	0.307397	0.999331	0.999331	1.000000
n=9	0.250421	0.999909	0.999909	1.000000
n=10	0.203219	0.999993	0.999993	1.000000

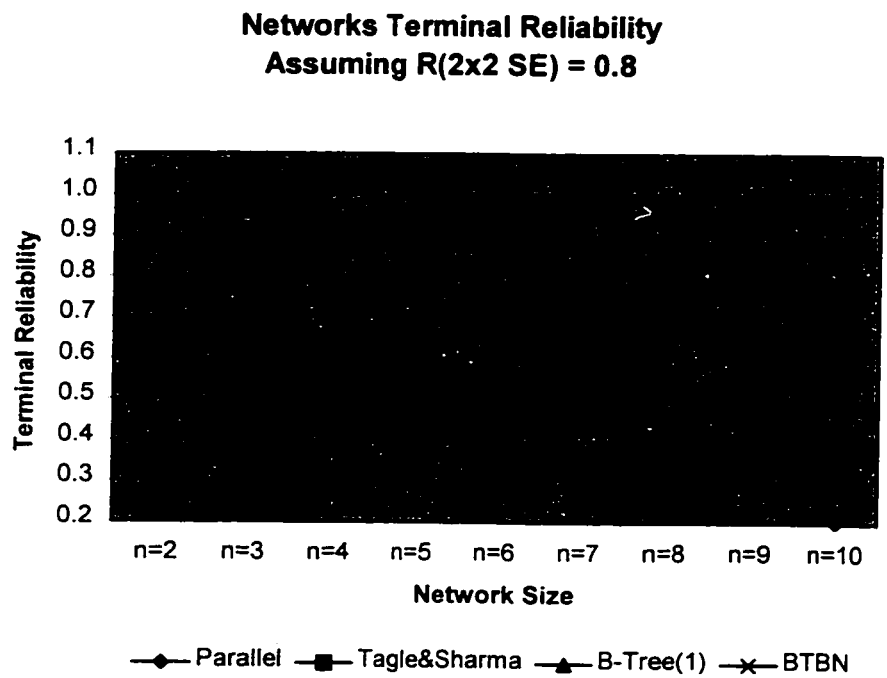


Figure 6.6 Terminal reliability of switching fabrics using second method.

Chapter 7

CONCLUSION AND FUTURE WORK

In this thesis, a high performance and fault tolerant ATM network, called Binary Tree Banyan Network (BTBN), is proposed. BTBN is carefully designed to achieve very low cell loss probability and high performance under normal conditions and in the presence of faulty switching elements. BTBN consists of two interconnected parallel banyan networks. In addition, both parallel banyan networks are also interconnected in a binary tree form.

The fault-free BTBN performance is evaluated using the permutation traffic simulation, analytical uniform traffic and simulation, hot spot traffic simulation, and ATM input traffic simulation. The throughput performance of 1024 X 1024 BTBN is

more than 99.92% under uniform traffic and more than 76% under 25% hot spot traffic. The throughput performance of 512 X 512 BTBN is more than 99.9999% under ATM input traffic, more than 77.85% under 25% hot spot traffic, and more than 99.92% under uniform traffic.

BTBN is also evaluated in the presence of faulty switching elements using two methods. The first method gives a very accurate measurement of the affect of single and double faulty switching elements on the BTBN throughput performance. The second method gives an approximate measurement of the effect of many faulty switching elements (up to 200) on the BTBN throughput performance. The throughput performance of 1024 X 1024 BTBN under uniform traffic and in the presence of 200 randomly selected faulty switching elements is more than 99.73%. The presence of 200 faulty switching elements has a small effect on the 1024 X 1024 BTBN. This is due to the huge number of redundant paths (5632 redundant paths) between any input to any output ports and due to the large number of access links (22 access links) to each output port buffer.

The BTBN reliability is calculated using two methods. In the first method, the terminal reliability is based on the cross-point switch reliability of switching elements and in the second method, it is based on the yield equation. It is demonstrated using both methods the superiority of BTBN over B-Tree (1), Tagle and Sharma's, and parallel banyan networks.

The following is a summary of the features of BTBN:

- Cell loss is minimized during normal operation and in the presence of faulty switching elements by resolving internal blocking and output contention problems. Internal blocking problem is resolved by providing huge number of redundant paths and by dividing the input load into two different banyan networks. The output contention is resolved by providing large number of access links to each output buffer.
- The number of cell routing stages is reduced. This resulted in an improvement to the switching fabric service time.
- The cells are delivered in sequence and there are no random delays within the switching fabric (no jitter).
- The switching fabric architecture is modular, easily expandable, and consists of regular structures suitable for VLSI implementation.

The projected future work activities are as follows:

- Only the second method is used to evaluate the BTBN under hot sport traffic. Design experiments will be conducted to evaluate the BTBN under hot spot traffic using the first method describe in Section 4.3.
- Only one ATM input traffic mix is used to evaluate the BTBN under ATM input traffic. Design experiments will be conducted to evaluate the BTBN under various ATM input traffic mixes.
- Design experiments will be conducted to evaluate the quality of service provided by BTBN using actual loads at the input ports such as video images.

- BTBN is a high performance and fault tolerant ATM switching fabric that is suitable for real time control applications. A complete design study will be conducted to measure the BTBN switching service time and its effect on the BTBN performance.

References

[1] A. L. Roginsky, L. A. Tomek, and K. J. Christensen, "Analysis of ATM Cell Loss for Systems with On/Off Traffic Sources," IEE Proceedings, Communications, Vol. 144, No. 3, pp. 129-134, June 1997.

[2] Andre Girard, and Redouane Zidane, "Revenue Optimization of B-ISDN Networks," IEEE Transactions on Communications, Vol. 43, No. 5, pp. 1992-1997, May 1995.

[3] Arata Itoh, "A Fault-Tolerant Switching Network for B-ISDN," IEEE Journal on Selected Areas in Communications, Vol. 9, No. 8, pp. 1218-1226, October 1991.

[4] Averill M. Law and W. David Kelton, "Simulation Modeling & Analysis," McGraw-Hill Series in Industrial Engineering Management Science, Second Edition, 1991.

[5] C. H. Ng, L. Bai, and B. H. Soong, "Modeling Multimedia Traffic Over ATM Using MMBP," IEE Proceedings, Communications, Vol. 144, No. 5, pp. 307-310, October 1997.

[6] C.-M. Weng and J.-J. Li, "Solution for Packet Switching of Broadband ISDN," IEE Proceedings-I Communications, Speech and Vision, Vol. 138, No. 5, pp. 394-400, October 1991.

[7] C. S. Raghavendra and Rajendra V. Boppana, "On Self-Routing in Benes and Shuffle-Exchange Networks," IEEE Transactions on Computers, Vol. 40, No. 9, pp. 1057-1064, September 1991.

[8] C. Y. Park, H. Chung, and C. K. Un, "Performance Analysis of an Output Queueing ATM Switch with Heterogeneous Traffic and Multiple QoS," IEE Proceedings, Communications, Vol. 143, No. 6, pp. 356-362, December 1996.

[9] Ching Yuh Jan and A. Yavuz Oruc, "Fast Self-Routing Permutation Switching on an Asymptotically Minimum Cost Network," IEEE Transactions on Computers, Vol. 42, No. 12, pp. 1469-1479, December 1993.

[10] Dimitri Bertsekas and Robert Gallager, "Data Networks," Prentice-Hall, Inc., Second Edition, 1992.

[11] Dong-Jye Shyy and Chin-Tau Lea, "Rearrangeable Nonblocking log(N,m,p) Networks," IEEE Transactions on Communications, Vol. 42, No. 5, pp. 2084-2086, May 1994.

[12] Fouad A. Tobagi, "Fast Packet Switch Architectures For Broadband Integrated Services Digital Networks," Proceedings of the IEEE, Vol. 78, No. 1, pp. 133-167, January 1990.

[13] Fouad A. Tobagi, Timothy Kwok, and Fabio M. Chiussi, "Architecture, Performance, and Implementation of the Tandem Banyan Fast Packet Switch," IEEE Journal On Selected Areas in Communications, Vol. 9, No. 8, pp. 1173-1193, October 1991.

[14] G. A. De Biase, C. Ferrone, and A. Massini, "An O(log N) Depth Asymptotically Nonblocking Self-Routing Permutation Network," IEEE Transactions on Computers, Vol. 44, No.8, pp. 1047-1050, August 1995.

[15] Gagan L. Choudhury, David M. Lucantoni, and Ward Whitt, "Squeezing the Most of ATM," IEEE Transactions on Communications, Vol. 44, No. 2, pp. 203-217, February 1996.

[16] George B. Adams III, Dharma P. Agrawal, and Howard Jay Siegel, "A Survey and Comparison of Fault-Tolerant Multistage Interconnection Networks," Computer Magazine, pp. 14-27, June 1987.

[17] G. Y. Lee and C. K. Un, "Design and Performance Analysis of an Asynchronous Banyan Network Switch With Window Policy," IEE Proceedings Communications, Vol. 142, No. 2, pp. 54-60, April 1995.

[18] H. Jonathan Chao, "A Recursive Modular Terabit/Second ATM Switch," IEEE Journal on Selected Areas in Communications, Vol. 9, No.8, pp. 1161-1172, October 1991.

[19] H. S. Kim and A. Leon-Garcia, "Decomposition of Output Queuing Switches," IEE Proceedings-I Communications, Speech and Vision, Vol. 138, No. 5, pp. 407-416, October 1991.

[20] Hamid Ahmadi and Wolfgang E. Denzel, "A Survey of Modern High-Performance Switching Techniques," IEEE Journal on Selected Areas in Communications, Vol. 7, No. 7, pp. 1091-1103, September 1989.

[21] ITU-T Recommendation I.121, "Broadband Aspects of ISDN," 1990.

[22] ITU-T Recommendation I.113, "Vocabulary of Terms for Broadband Aspects of ISDN," 1991.

[23] Iwao Toda, "Migration to Broadband ISDN," IEEE Communications Magazine, Vol. 28, No. 4, pp. 55-58, April 1990.

[24] Jagdish Kohli, "Medical Imaging Applications of Emerging Broadband Networks," IEEE Communications Magazine, Vol. 27, No. 12, pp. 8-16, December 1989.

[25] James N. Giacopelli, Jason J. Hickey, William S. Marcus, W. David Sincoskie, and Morgan Littlewood, "Sunshine: A High-Performance Self-Routing Broadband Packet Switch Architecture," IEEE Journal on Selected Areas in Communications, Vol. 9, No. 8, pp. 1289-1298, October 1991.

[26] Jeen-Fong LIN and Sheng-De WANG, "A High Performance Fault-Tolerant Switching Network for ATM," IEICE Trans. Communication, Vol. E78-B, No. 11, pp. 1518-1528, November 1995.

[27] Jiunn-Jian Li and Cheng-Ming Weng, "B-Tree: A High-Performance Fault-Tolerant ATM Switch," IEE Proceedings Communications, Vol. 141, No. 1, pp. 20-28, February 1994.

[28] Jiunn-Jian Li and Cheng-Ming Weng, "B^+-Tree: A High-Performance Switching Structure For ATM With Dual Input Buffering," Computer Networks and ISDN Systems, Vol. 27, pp. 1499-1522, 1995.

[29] Joseph L. Hammond and Peter J. P. O'Reilly, "Performance Analysis of Local Computer Networks" Addison-Wesley Publishing Company, Inc., 1988.

[30] Kai Y. Eng, Mark J. Karol, and Yu-Shuan Yeh, "A Growable Packet (ATM) Switch Architecture: Design Principles and Applications," IEEE Transactions on Communications, Vol. 40, No. 2, pp. 423-430, February 1992.

[31] Karen L. Reid, "Priority Queueing in Asynchronous Transfer Mode Switching," M.S. Thesis, University of Saskatchewan, August 1992.

[32] Kazuo Murano, Koso Murakami, Eisuke Iwabuchi, Toshi Katsuki, and Hiroshi Ogasawara, "Technologies Towards Broadband ISDN," IEEE Communications Magazine, Vol. 28, No. 4, pp. 66-70, April 1990.

[33] Kouichi Genda and Naoaki Yamanaka, "TORUS: Terabit-per-Second ATM Switching System Architecture Based on Distributed Internal Speed-Up ATM Switch," IEEE Journal on Selected Areas in Communications, Vol. 15, No. 5, pp. 817-829, June 1997.

[34] Krishnan Padmanabhan and Duncan H. Lawrie, "A Class of Redundant Path Multistage Interconnection Networks," IEEE Transactions on Computers, Vol. C-32, No. 12, pp. 1099-1108, December 1983.

[35] Laxmi N. Bhuyan and Dharma P. Agrawal, "Design and Performance of Generalized Interconnection Networks," IEEE Transactions on Computers, Vol. C-32, No. 12, pp. 1081-1090, December 1983.

[36] M. A. Henrion, G. J. Eilenberger, G. H. Petit, and P. H. Parmentier, "A Multipath Self-Routing Switch," IEEE Communications Magazine, pp. 46-52, April 1993.

[37] Madihally J. Narasimah, "The Batcher-Banyan Self-Routing Network: Universality and Simplification," IEEE Transactions on Communications, Vol. 36, No. 10, pp. 1175-1182, October 1988.

[38] Marc Boisseau, Michel Demange, and Jean-Marie Munier, "An Introduction to ATM Technology," International Thomson Publishing, Second Edition, 1996.

[39] Mark J. Karol, Michael G. Hluchyj, and Samuel P. Morgan, "Input Versus Output Queueing on a Space-Division Packet Switch," IEEE Transactions on Communications, Vol. COM-35, No. 12, pp. 1347-1356, October 1987.

[40] Martin de Prycker, "Asynchronous Transfer Mode Solution for Broadband ISDN," Ellis Horwood Limited, Second Edition, 1993.

[41] Michael Cooperman, A. Paige, and Richard W. Sieber, "Broadband Video Switching," IEEE Communications Magazine, Vol. 27, No. 12, pp. 26-30, December 1989.

[42] Mike Frame, " Broadband Services Needs," IEEE Communications Magazine, Vol. 28, No. 4, pp. 59-62, April 1990.

[43] Mohsen Guizani and Atif M. Memon, "SEROS - A Self-Routing Optical ATM Switch," International Journal of Communication Systems, Vol. 9, pp. 115-125, 1996.

[44] Mostafa Abd-El-Barr, Khalid Al-Tawil, and Osama Abed, "Fault-Tolerance and Reliability Analysis of Multi-Stage Data Manipulator Networks," 8th ISCA/IEEE International Conference on Parallel and Distributed Computing Systems, pp. 275-280, September 1995.

[45] Mustafa K. Mehmet-Ali, Mojtaba Youssefi, and Huu Tri Nguyen, "The Performance Analysis and Implementation of an Input Access Scheme in a High-Speed Packet Switch," IEEE Transactions on Communications, Vol. 42, No. 12, pp. 3189-3199, December 1994.

[46] Narayana R. Tummala, "VISTA: A Configurable Visualization and Simulation Tool for ATM Switches," M.S. Thesis, University of Texas at San Antonio, Spring 1996.

[47] Nobuhiko Kitawaki, Hiromi Nagabuchi, Masahiro Taka, and Kenzo Takahashi, "Speech Coding Technology for ATM Networks," IEEE Communications Magazine, Vol. 28, No. 1, January 1990.

[48] P. C. Wong and M. S. Yeung, "Design and Analysis of a Novel Fast Packet Switch-Pipeline Banyan," IEEE/ACM Transactions on Networking, Vol. 3, No. 1, pp. 63-69, February 1995.

[49] Pierre U. Tagle and Neeraj K. Sharma, "A High-Performance Fault-Tolerant Switching Network for B-ISDN," Phoenix Conference on Computers and Communications, pp. 599-606, March 1995.

[50] Pierre U. Tagle and Neeraj K. Sharma, "Performance of Fault Tolerant ATM Switches," IEE Proceedings, Communications, Vol. 143, No. 5, pp. 317-324, October 1996.

[51] Ra'ed Y. Awdeh and H. T. Mouftah, "Design and Performance Analysis of Input-Output Buffering Delta-Based ATM Switch With Back Pressure

Mechanism," IEE Proceedings Communications, Vol. 141, No. 4, pp. 255-264, August 1994.

[52] Ra'ed Y. Awdeh and H. T. Mouftah, "Survey of ATM switch architectures," Computer Networks and ISDN Systems, Vol. 27, pp. 1567-1613, 1995.

[53] Raif O. Onvural, "Asynchronous Transfer Mode Networks: Performance Issues," Artech House, Inc., 1995.

[54] Rainer Handel and Manfred N. Huber, "Integrated Broadband Networks: An Introduction to ATM-Based Networks," Addison Publishing Company Inc., 1991.

[55] RogerBushnell, David Morgan, and Michael Lach, "ISDN as an Enabler for Enterprise Integration," IEEE Communications Magazine, Vol. 28, No. 4, pp. 23-27, April 1990.

[56] Russell G. Dewitt, "ISDN Symposia: A Historical Overview," IEEE Communications Magazine, Vol. 28, No. 4, pp. 10-12, April 1990.

[57] Russell Roy, "ISDN Applications at Tenneco Gas," IEEE Communications Magazine, Vol. 28, No. 4, pp. 28-31, April 1990.

[58] Sandeep Sibal and Ji Zhang, "On a Class of Banyan Networks and Tandem Banyan Switching Fabrics," IEEE Transaction on Communications, Vol. 43, No. 7, pp. 2231-2240, July 1995.

[59] Sandro Bassi, Maurizio Decina, Paolo Giacomazzi, and Achille Pattavina, "Multistage Shuffle Networks with Shortest Path and Deflection Routing for

High Performance ATM Switching: The Open-Loop Shuffleout," IEEE Transaction on Communications, Vol. 42, No. 10, pp. 2881-2889, October 1994.

[60] Satoshi Nojima, Eiichi Tsutsui, Harkuki Fukuda, and Masamichi Hashimoto, "Integrated Services Packet Network Using Bus Matrix Switch," IEEE Journal on Selected Areas in Communications, Vol. SAC-5, No. 8, pp. 1284-1292, October 1987.

[61] Sema F. Oktug and Mehmet U. Caglayan, "Design and Performance Evaluation of a Banyan Network Based Interconnection Structure for ATM Switches," IEEE Journal on Selected Areas in Communications, Vol. 15, No. 5, pp. 807-816, June 1997.

[62] Shigeo Urushidani, "Rerouting Network: A High-Performance Self-Routing Switch for B-ISDN," IEEE Journal on Selected Areas in Communications, Vol. 9, No. 8, pp. 1194-1204, October 1991.

[63] Shih-Chian and John A. Silvester, "A reconfigurable ATM Switch Fabric for Fault Tolerance and Traffic Balancing," IEEE Journal on Selected Areas in Communications, Vol. 9, No. 8, pp. 1205-1217, October 1991.

[64] Stanford R. Amstutz, "Burst Switching-An Update," IEEE Communications Magazine, pp. 50-58, September 1989.

[65] Steven E. Minzer, "Broadband ISDN and Asynchronous Transfer Mode (ATM)," IEEE Communications Magazine, pp. 17-24, September 1989.

[66] Sying-Jyan Wang, "Distributed Routing in a Fault-Tolerant Multistage Interconnection Network," Information Processing Letters, No. 63, pp. 205-210, 1997.

[67] T. H. Cheng and Y. Shen, "Nonuniform Traffic Analysis on a Class of Self-Routing Networks," IEE Proceedings, Communications, Vol. 144, No. 3, pp. 143-149, June 1997.

[68] T. M. Bachtiar and M. H. Abd-Elbarr, "Logical Neighbourhood Network for Fault Tolerance in Packet Switching Networks," Proceeding of the 5th International Conference on Microelectronics, pp. 287-293, December 1993.

[69] T. Szymanski and C. Fang, "Randomized Routing of Virtual Connections in Essentially Nonblocking log(N)-Depth Networks," IEEE Transactions on Communications, Vol. 43, No. 9, pp. 2521-2531, September 1995.

[70] T.-Y. Huang and J.-L.C. Wu, "Performance Analysis of ATM Switching Using Priority Schemes, " IEE Proc.-Commun., Vol. 141, No. 4, pp. 248-254, August 1994.

[71] Talha M. Al-Jarad, "Design and Analysis of a Fault Tolerant Switch for B-ISDN," M.S. Thesis, King Fahd University of Petroleum & Minerals, January 1997.

[72] Tsern-Huei Lee, "Design and Analysis of a New Self-Routing Network," IEEE Transactions on Communications, Vol. 40, No. 1, pp. 171-177, January 1992.

[73] Uyless Black, "ATM: Foundation For Broadband Network," Prentic Hall Series in Advanced Communications Technologies, 1995.

[74] Venkatesh Chandramouli and C. S. Raghavendra, "Nonblocking Properties of Interconnection Switching Networks," IEEE Transactions on Communications, Vol. 43, No. 2/3/4, pp. 1793-1799, February/March/April 1995

[75] Vijay P. Kumar and S. M. Reddy, "Augmented Shuffle-Exchange Multistage Interconnection Networks," Computer Magazine, pp. 30-40, June 1987.

[76] Vijay P. Kumar, Joseph G. Kneuer, and Debajyoti Pal, "PHOENIX: A Building Block for Fault Tolerant Broadband Packet Switches," GLOBECOM '91, pp. 228-233.

[77] Wasif Hasan, "A Novel Fast Packet Switch Architecture for ATM Networks," M.S. Thesis, King Fahd University of Petroleum & Minerals, March 1996.

[78] Xing Chen, I. Lambadaris, and Jeremiah F. Hayes, "Queueing Analysis of ATM Multicast Switching Models," IEEE Transactions on Communications, Vol. 43, No. 12, pp. 2886-2890, December 1995.

[79] Y.-J. Cheng, T.-H. Lee, and W.-Z. Shen, "Design and Performance Evaluation of a Distributed Knockout Switch with Input and Output Buffers," IEE Proceedings, Communications, Vol. 143, No. 3, pp. 149-154, June 1996.

[80] Yo-Song Su and Jau-Huang, "Throughput Analysis and Optimal Design of Banyan Switches With Bypass Queues." IEEE Transactions on Communications, Vol. 42, No. 10, pp. 2781-2784, October 1994.

[81] Youn Chan Jung and Chong-Kwan Un, "Banyan Multipath Self-Routing ATM Switches With Shared Buffer Type Switch Elements," IEEE Transactions on Communications, Vol. 43, No. 11, pp.2847-2857, November 1995.

[82] Young Man Kim and Kyungsook Y. Lee, "PR-Banyan: A Packet Switch With a Pseudorandomizer for Nonuniform Traffic," IEEE Transactions on Communications, Vol. 41, No. 7, pp. 1039-1042, July 1993.

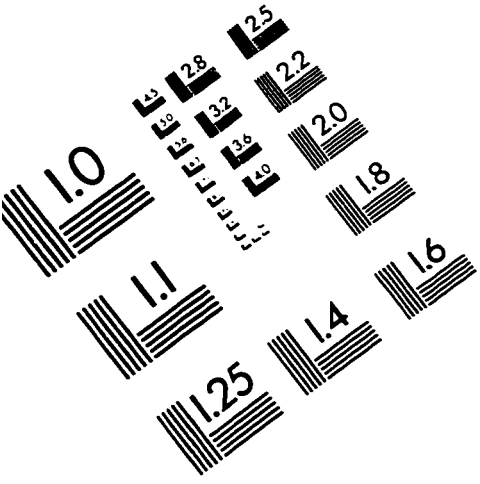
[83] Young Man Kim and Kyungsook Y. Lee, "KSMIN's: Knockout Switch-Based Multistage Interconnection Networks for High-Speed Packet Switching," IEEE Transactions on Communications, Vol. 43, No. 8, pp. 2391-2398, August 1995.

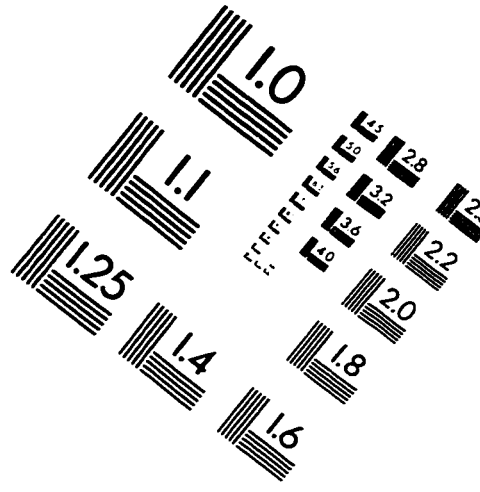
[84] Yu-Shuan Yeh, Michael G. Hluchyj, and Anthony S. Acampora, "The Knockout Switch: A Simple, Modular Architecture for High-Performance Packet Switching," IEEE Journal on Selected Areas in Communications, Vol. SAC-5, No. 8, pp. 1274-1283, October 1987.

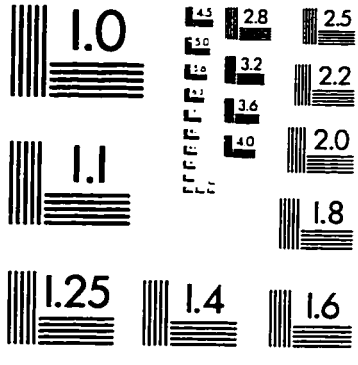
Vita

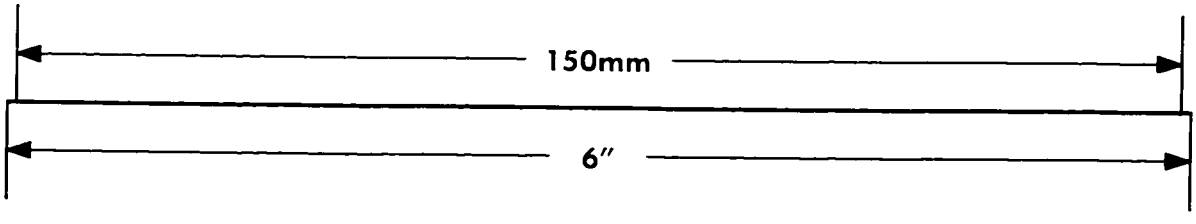
- Ghaleb A. Al-Hashim
- Born in Al-Hassa, Saudi Arabia
- Received Bachelor's degree in Computer Engineering from University of Arizona (UofA), AZ, USA in May 1988.
- Worked for the Saudi Arabian Oil Company (Saudi Aramco) for fifteen years.
- Received Master's degree in Computer Engineering from King Fahd University of Science (KFUPM), Dhahran, Saudi Arabia in May 1998.

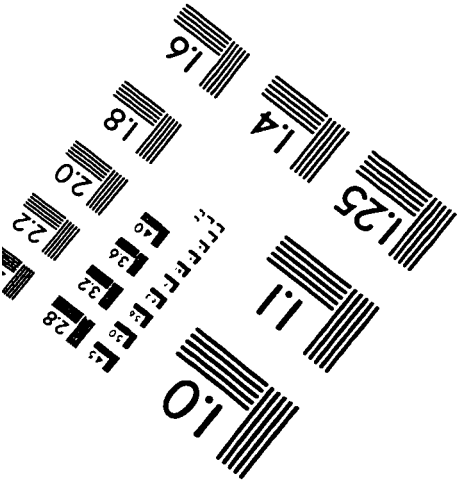
IMAGE EVALUATION TEST TARGET (QA-3)

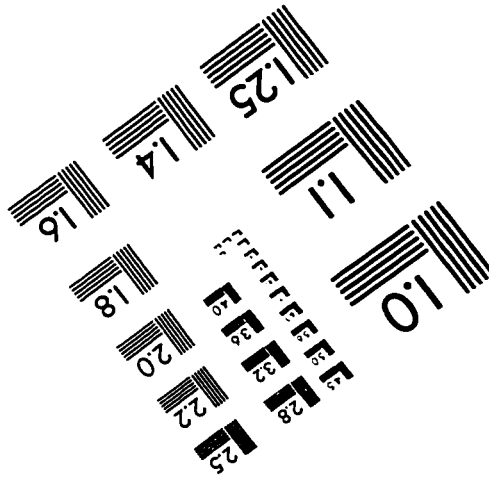












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989