PERFORMANCE ANALYSIS OF AUGMENTED SHUFFLE EXCHANGE NETWORKS

by

Viswanathan Ramachandran

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

APPROVED:

Nathaniel J. Davis IV, Chairman

Scott F. Midkiff

Walling R. Cyre

June, 1992

Blacksburg, Virginia

c.2

PERFORMANCE ANALYSIS OF AUGMENTED SHUFFLE EXCHANGE NETWORKS

by

Viswanathan Ramachandran

Committee Chairman: Nathaniel J. Davis IV
Electrical Engineering

(ABSTRACT)

This research presents an analysis of the improvement in the performance of a class of fault tolerant multistage interconnection networks. In the network discussed here, fault tolerance is achieved by providing multiple redundant paths between the source and destination. The extra paths are obtained by providing redundant links between switching elements within a stage (intra-stage links), thereby increasing the switching element complexity. The techniques used in the construction of this network, its properties, advantages, and disadvantages are discussed. While early studies focused their effort in analyzing the fault tolerant characteristics of the network and the performance in a circuit switched environment, this investigation complements the previous work by examining the performance of a packet switched network. The reasons for the choice of the architecture that include factors like hardware complexity, cost and simplicity of control algorithm are analyzed. The study concentrates on improving the run-time performance of the fault tolerant network by using these multiple paths not only in the presence of a fault, but also in a fault-free environment. The throughput of the packet switched network in the presence of a fault, congestion and when fault free are analyzed. A description of the investigation, assumptions and factors used for the study, a cost analysis, and the results of the simulation analyses is included.

Acknowledgments

Many people have helped this thesis along the way and in these few words I would like to express my appreciation for their help. Special thanks go to my advisor, Dr. N. J. Davis IV, for all the help he has given me. Dr. S. F. Midkiff and Dr. W. Cyre were kind enough to serve in my committee. I enjoyed working with R. Raines and J. Park. Thanks also go to Dr I. Jacobs and the Dept. of Statistics. For all the nice Friday evenings I have enjoyed thanks are due to Vasant, Sam, Arun, Mahesh, Rajesh and Rajat. Not to be forgotten are my parents and my sisters whose moral support is greatly appreciated.

TABLE OF CONTENTS

Page

Page

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The inherent limitation of uniprocessor systems [Dec90] and the insatiable demand for processing power, has led the computer world to look at multiprocessor systems as a possible solution for computing needs. Multiprocessor systems connect a number of processors and memory modules together. Improvements in the performance of these multiprocessor systems are now being achieved through advanced system architectures. The tremendous advances in VLSI technology and the availability of inexpensive hardware have been the major impetus for the growth of these modern day multiprocessor systems.

1.1 Multiprocessor Systems

For decades, computers used a single processor to interpret instructions one-at-a-time and process the data. Innovative techniques were developed to speed up this slow process. At first, the clock speed was increased, reducing the time between independent instructions. Then, faster components were built and the components were packed tightly to minimize the propagation delays for communication. Finally, researchers took steps towards parallelism. Rather than letting the central processor carry out every step of a task, the concept of "pipelining" was used to split the central processor into an assembly line (an array) of processing units. Each one of these units would carry out a task one step at a time and pass it on to a second unit. While the second unit started on its task, the first would start on another. The idea of parallelism was further extended by incorporating more processors into a system. In such a system, each processor carries out a separate stream of instructions and, as a whole, works in a joint fashion. This paved way to the

concept of parallel processing or concurrent processing. Concurrent processing is considered a proper approach for significantly increasing processing speed [Fen81].

Parallel processing achieves speed up by identifying routines that can be distributed among processing elements for simultaneous execution. Processing elements have to access the data stored in the memory and communicate with the memory to carry out the stream of instructions. Hence, much of the recent research interest has been focused on the interprocessor communication network that transmits data and control information to and between a system's processing elements and memory modules. The overall system performance relies heavily on this interconnection network. If the inter-processor or processor-memory communication proves to be inefficient, the time gained through the use of a multiple processor system can be lost. Since time is a determining factor in the execution of parallel programs in these highly integrated systems, the communication network must also be reliable. The time required to identify and recover from a network fault can easily offset any performance gains resulting from the system's parallelism.

1.2 Interprocessor Communication Network

The heart of today's multiprocessor systems is the communication network. In many cases, the performance of the communication network determines the performance of the overall system. Hence, the class of communication networks, called Interconnection Networks (IN), forms a crucial topic in the field of parallel and or distributed computing. INs consist of software and hardware entities that are designed to facilitate efficient interprocess and interprocessor communication in a parallel processing system. Parallel processing systems are currently available that contain 10's, 100's and even 1000's of processor and memory subsystems. Tasks are executed in parallel by distributing the work load between the many processors, thus reducing the total run time. In a basic concurrent processing system, processes are assigned to individual processors. In

many cases, the execution of a process depends upon the result of some other process executed at an earlier instant. When a situation of this nature arises, where processors are assigned tasks or when processors have to communicate with each other, an IN provides the necessary framework for communication. The communication takes place not only between the processors but also between processors and memory modules. INs have emerged as a basic issue in exploiting this parallelism. The IN greatly affects the system level control in addition to its impact on algorithmic design. Figure 1.1 illustrates the classification of MIN structures.

INs can be implemented in a wide variety of topologies based on a system's cost and its desired level of performance. A cost effective network topology is the bus or ring IN. Although an architecture like the bus is economically feasible, its communication speed is unacceptably slow. The bus is a set of wires that connects all the processing elements together. The bus allows only one source processing element to communicate with its destination at a time. As the number of processors sharing a bus or ring grows, there is increased contention and the bus system of interconnection becomes less and less attractive. For a network with N sources (inputs) and N destinations (outputs), multiple buses of the order $N^2/2$ need to be employed to overcome the problem of contention. A system of multiple buses is not cost effective, and its design structure is more complicated than the single bus architecture.

In contrast to the bus, a crossbar network is an ideal IN for interprocess communication in small systems. This is because in a crossbar network, the information needs to pass through only one switch to reach the destination. A crossbar switch provides an explicit path between the communicating elements. Since any input can be connected to any output, all permutations, i.e., every possible combination of input-output pairs is realizable in a crossbar switch. In a multiprocessor system, a crossbar switch provides one of the fastest means of connecting a source

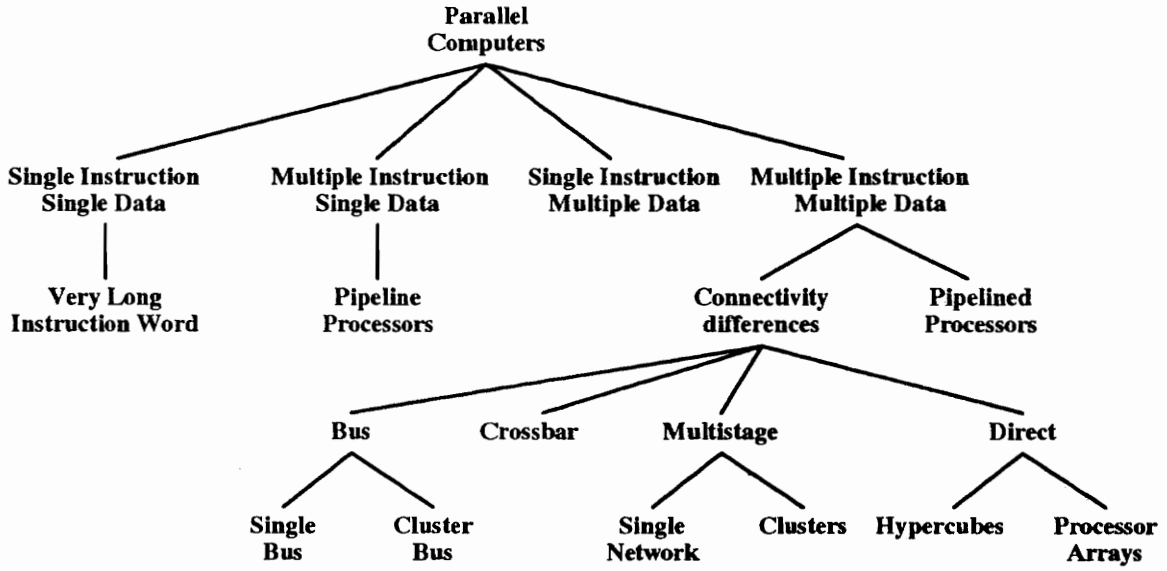

Figure 1.1: Classification of MIN Structures

to a destination because of the short transmission delay based on the IN's powerful communication capabilities. For a network with N sources (inputs) and N destinations (outputs), the required number of cross points is of the order N^2. For large values of N, implementation of the crossbar proves to be very expensive. The idea of a crossbar switch may be extended to a network that has point-to-point links. Since there is usually a bound on the maximum number of processors that can be directly connected to a single processor, it is not possible to have these dedicated links that connect the sources and the destinations [Kum85].

1.3 Multistage Interconnection Networks

Multistage interconnection networks (MINs) are a class of interconnection networks that can provide a good, efficient, distributed connection scheme in a multiprocessor system [Sto71] and [Law75]. MINs provide a good compromise between the expensive crossbar switch and the slow bus topology. A MIN consists of several stages of switching elements (SE). The number of stages depends on the network size and the complexity of the switching elements. The SEs are usually n-by-n crossbar switches (e.g., 2-by-2 or 4-by-4) and the switching elements in adjacent stages are connected by inter-stage links. Many such networks have been proposed [JeS86] [DiJ81] [KuR87] [AdS82] [PaL83] [WuF84]. In all these networks, any input port of the MIN can be connected to any one of its output ports. As shown in Figure 1.2, a connection can be established between input port 3 and any output port, through the labeled switches. As long as no two connections attempt to share a common link in a MIN, simultaneous connections can be provided. Multistage interconnection networks have the ability to provide these simultaneous connections at a hardware cost that is considerably lower than that of crossbars. This is because the cost of the MIN is of the order NlogN when compared to N^2 for that of crossbars. Although a MIN may not be able to realize all the permutations that are possible in crossbars, it is a very cost effective means to provide high bandwidth communication in multiprocessor systems.

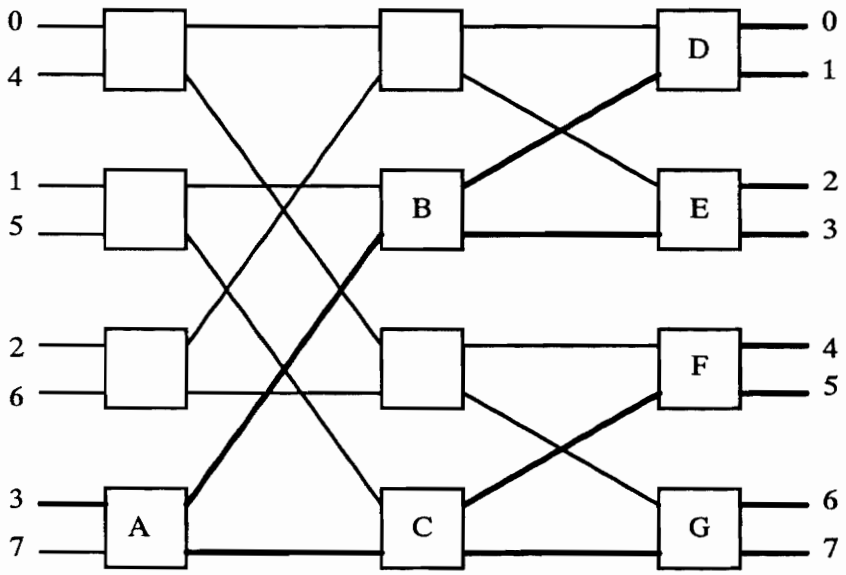

Figure 1.2: Illustration of MIN connectivity

MINs, even though a good compromise between the crossbar switch and the bus architecture, have their drawbacks. Two situations can drastically affect the performance of a MIN. First, heavy network loading will cause congestion and thus long delays in message completion. Secondly, a fault in one of the MIN links or switching elements will cause some source-destination communication failures.

A congested environment is one in which a certain group of switching elements and/or the links between stages becomes a favorite path for the transmission of messages. Congestion can also occur, when requests enter network at a rate faster than what the network can handle. The consequence is heavy traffic at the switching element, causing the message requests to queue one behind the other. There is usually a limit on the maximum number of requests that can be buffered by a SE. Hence, a queue-up can cause the SEs in the previous stage to stop transmitting, as the requests issued by the SE haven't been serviced yet. These situations are commonly referred to as "Hot Spots" [PfN85]. The net effect of a situation like this is the queuing of the requests that builds up backwards towards the network sources in a tree-like fashion.

In a faulty environment, a MIN may not be able to honor requests from a source at all. A fault in a SE or in a SE-SE link can result in one (or more) of the destinations being inaccessible. Under such conditions, performance of the MIN deteriorates and the network cannot carry out its functions. An improvement in the performance of the network can be brought about by introducing fault tolerant features. The features can be in the form of redundant links or extra stages that provide alternative ways to access the destination that are not found in the normal MIN topologies. Chapter 2 discusses several ways of achieving fault tolerance in MINs.

Unlike a crossbar switch, a MIN does not have links that exclusively serve the input ports. The absence of these links can account for certain situations in which two or more requests contend for the same output link. In such a situation, a decision is to be made whether to drop or hold one of the requests while servicing the other. A buffered request suffers a delay in reaching the destination, while a dropped request requires re-transmission and hence additional delay.

1.4 Research Goals

This thesis focuses on the run-time performance improvement brought about by the presence of redundant links in a MIN. The performance study discusses the instances in which a MIN faces the problem of congestion or encounters a faulty situation. In order to carry out the study, a software model of a fault tolerant IN, the Augmented Shuffle Exchange Network (ASEN) proposed by Kumar [Kum87] is used to analyze the performance metrics. The model includes, the average time spent in the network, the variance in the delay times, and the cost involved in the construction of the network. The primary goal of this investigation is to propose and evaluate an optimal configuration of the ASEN that is fault tolerant, efficient in performance, and minimal in cost. The effective use of the redundant links in a fault free environment to bring about an increased level of performance is also discussed. This investigation complements Kumar's work by providing a comparison of the ASEN and the multistage cube network in a packet switched environment with heavy network congestion.

1.5 Outline of Thesis

In Chapter 2, the idea of fault tolerance is discussed. A discussion of MIN structures, the background and related research done so far in this field and a comparison of several fault tolerant networks are then outlined. Chapter 3 discusses the choice of network structure, its reliability, fault tolerant properties, and the extensions in the hardware needed to overcome some of the

drawbacks present in the simple MIN structures. Later sections include a discussion of the congestion avoidance schemes, the dynamic routing capability and the utilization of the redundant links.

Chapter 4 is a comprehensive treatment of the modeling concepts of the network. The chapter begins with the details of the assumptions on modeling, the routing algorithm, and a discussion of the performance characteristics of the network. The network performance analysis and the validations follow this discussion. The concept of mathematical meta-modeling applied to the network, is presented in the later half of this chapter. A comparison of some fault tolerant architectures is also outlined. Chapter 5 presents the conclusions and the scope for future research.

CHAPTER 2

OVERVIEW OF FAULT TOLERANT NETWORKS

This chapter introduces the concept of fault tolerance in a Multistage Interconnection Network. It is followed by a discussion of the MIN structure. The possible avenues to overcome the problems of poor performance and lack of fault tolerant capability seen in a MIN are then explored. Properties of a network, including factors like performance considerations, reliability, bandwidth, and control algorithms, that determine the network performance are also discussed. A comparison of several different networks that have been proposed is then presented. This comparison highlights the advantages and the drawbacks suffered in each one of the networks.

2.1 Fault Tolerance

A fault tolerant system is one which continues to perform its specified tasks in the presence of hardware and/or software faults. A fault can be local or global, transient, intermittent, or permanent. Fault tolerance is achieved, either by fault avoidance during the manufacturing process or by introducing redundancy into the system in the form of extra hardware, software, information, or time. Redundancy is simply an addition of resources beyond what is needed for normal system operation.

Hardware redundancy is usually provided for the purpose of detecting, locating and thereby circumventing a fault. Hardware redundancy can be passive or active. The passive technique uses the concept of fault masking to hide the occurrence of a fault. The active approach detects a fault and takes some action to remove the faulty hardware from the system.

Two common software redundancy techniques are consistency checks, and capability checks. A consistency check can be done on systems where the outcome is known in advance and can, therefore confirm the correct operation of the software. A capability check is done in software to ensure that the system is working properly, e.g., testing of memory locations by processors

Information redundancy is the addition of redundant data to allow fault detection. Error detecting and error correcting codes are formed by mapping data words into new representations containing redundant information. In all these cases a valid binary code is constructed that is a combination of 1's and 0's. An error is detected if the errors introduced into the code force the code to lie in the range of illegal code words.

Time redundancy attempts to reduce extra hardware requirements at the expense of added run time. It is implemented in applications where time is of less importance when compared to hardware or hardware cost.

The need for fault tolerant features was seen in Chapter 1 in the discussion of MINs operating in a congested or a faulty environment. The necessity of fault tolerant features in a MIN is further emphasized when the different interconnection network structures are analyzed. INs, in general, can be a single stage or a multistage network based on topological considerations. A single stage network is composed of a stage of switching elements cascaded to a link connection pattern, i.e., the output links are fed back to the input. The information (data) packets pass through the same stage a number of times, until the switches in the stage route the data to its destination. The single stage network is also known as a recirculating network. A multistage network, on the other hand, contains more than one stage of switching elements and is capable of

connecting an arbitrary input terminal to any output terminal. Multistage networks are considered in this thesis.

2.2 MIN Structures

Multistage interconnection networks can either be nonblocking or blocking, according to the different permutations the network is capable of performing.

2.2.1 Nonblocking Networks

A MIN is called strictly nonblocking (SNB) if, irrespective of what state the network is in, it is able to connect an input port to an output port not already in use by another input port [Clo53] [Ben65]. On the other hand, the MIN is a rearrangable nonblocking network (RNB) if, the network rearranges some of the existing conditions to connect an input to a non-busy output [Ben62]. For a network with N inputs and outputs, the SNBs have a hardware complexity of the order N $(\log N)^2$, while the RNBs have a complexity of NlogN. It is always possible to find a permutation between a pair of input and output ports in the RNB network. The drawback is that the routing algorithm becomes more complex, of the order NlogN. Hence both SNB and RNB networks are not widely used in multiprocessor systems.

2.2.2 Blocking Networks

A MIN is said to be blocking if it is possible that a connection cannot be set up between an input/output pair because an interconnection link is already busy serving another input/output pair. It can be seen from Figure 2.1 that a path for communication between source-3 and destination-5 cannot be established because of an active connection between source-5 and destination-6. The blocking in these MINs occurs because of the conflicts that arise due to common SE-SE links.

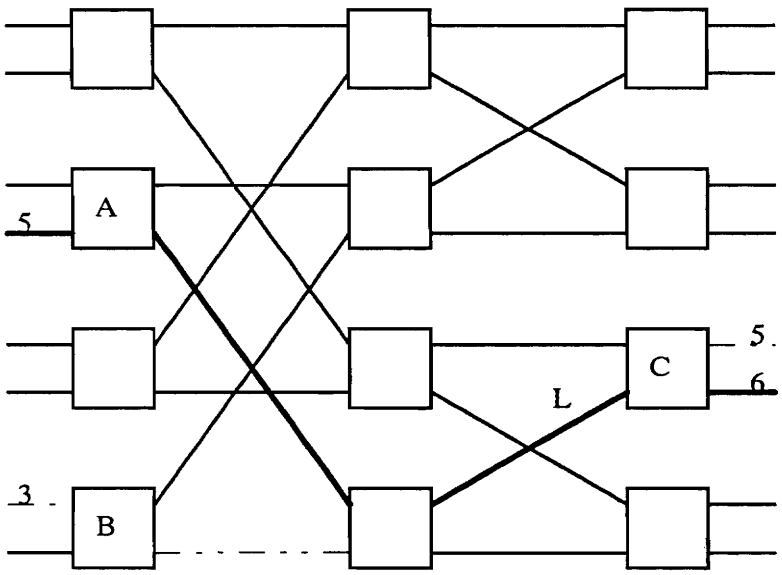

Figure 2.1: Blocking Network

From the many blocking networks proposed so far, it has been shown that the hardware complexity of the blocking network is half that of a RNB network [KrS83]. Blocking networks have only one single path from any input port to any output port and hence the path is unique. For an N-input blocking MIN with $N=m^n$, and a switch size m-by-m, the hardware complexity is $mN\log_m N$ [KuR87]. Since blocking networks have simple and distributed control algorithms they are implemented in multiprocessor applications. Blocking networks suffer from potentially poor performance and low reliability due to lack of fault tolerance capability. In the presence of a fault, the absence of alternate paths leads to poor performance as some input/output pairs can no longer be connected. A switch failure can lead to the disconnection of m^{n+1} paths for a blocking network described above. The performance of blocking networks have been studied in detail by Patel [Pat81] and [KrS83]. They show that, as the number of stages increase in the blocking network, the percentage of requests that are honored decreases. Moreover, the waiting time, i.e., the time taken by the packets at the queues while traversing the network, accounted for 75% of the communication delay for network sizes as big as 1024-by-1024. Several MINs have been proposed to overcome these drawbacks and they are discussed in Section 2.3.

2.3 Network Design Considerations

The discussion of blocking networks has highlighted the fact that unipath networks suffer from two major drawbacks because of their structure: poor performance and lack of fault tolerance capability. MINs have been proposed that overcome these disadvantages. In all these networks an alternate path is chosen when the original path has a fault or if the path is blocked. A study of the factors that play a role in designing the structure of the network is hence essential. They are:

- Control strategy
- Switching method
- Routing method

- Operation mode

2.3.1 Control strategy

A network control strategy determines operational characteristics, such as routing, blocking, protocol management, congestion control, in a MIN. These strategies can be implemented either as a centralized function or can be distributed across the network's switching elements.

2.3.1.1 Centralized Control

In a centralized control strategy, a certain processor decides on the course of action to be taken by the network. It is that processor which controls the routing, the protocol handling, and other control features of the network. The advantage of this scheme is that the switching elements are simple and less complicated. The SEs are only responsible for the transmission of messages across the link to the SE in the next stage. The disadvantage of the centralized control method is that the central processor has to keep track of the faults and take into consideration other abnormalities of the network while rerouting a packet. The control processor also becomes a hot spot in communication traffic and a potential single point failure for the entire network.

2.3.1.2 Distributed Control

Distributed control, in contrast to centrally controlled network operation, increases the complexity of the switches, so that routing and protocol functions are now handled by the switching elements. Each SE in the network has the built-in capacity and the ability to handle the protocols while the messages are being routed from one stage to another and eventually to their destination. In the event of a fault in an IN with distributed control, it is assumed that the switch

has the intelligence to inform the source and the switching elements it is connected to, about the fault. The advantage of this over the centralized control strategy is that there is no control processor whose failure paralyzes the network. Assuming similar speeds for the centralized processors and the SE's, the performance of the network with distributed control is better than a network with centralized control.

2.3.2 Switching method

A switching strategy used in a MIN determines the method in which messages are sent from the source to the destination. The message can either be transmitted as a whole, as in circuit switching, or broken down into a series of packets and sent through the network, as in packet switching.

2.3.2.1 Circuit Switching

In the circuit switching strategy, an exclusive independent virtual circuit is established for a source/destination pair on request and mutual agreement. This virtual circuit continues to exist until both the source and the destination agree to stop communicating with each other. The SE-SE links are then made available for another circuit to be established. The advantage of this switching methodology is the presence of an independent circuit exclusively for the use of an input/output pair. The disadvantage of the circuit switching method is that no communication is possible between any other source/destination pair, if that pair has to use some of the links used by an already established circuit.

2.3.2.2 Packet Switching

In the packet switching method, the information (data) is broken down into a series of small packets and transmitted from the network source to the destination. This is the most

commonly used switching method. The advantage of the packet switching method over circuit switching is that no SEs and links are exclusively serving a source-destination pair, they are available to all the sources requesting transmission of messages. In addition to reducing the overall transmission delay of the network, the throughput is also high since the effective utilization of the circuit is high. In packet switching, each packet or block of data produced at a source is prefixed with a header containing routing information that identifies the source and the destination. Intermediate switching elements in the network examine each header and decide where to send the packet to move it closer to its final destination. Since the switching elements get information about the status of the links that are busy or have failed, they have the capability to select alternate routes for passing packets without altering the packets' destination. This is an advantage of using packet switching instead of circuit switching, where new circuits would have to be setup if messages are to be rerouted. Establishment of new circuits could, in turn, block the normal flow of packets at the rerouted node.

2.3.3 Routing

The third crucial factor that plays a role in the design of a MIN is the routing capability of the network. In a MIN, due to certain faults or blockages in the links, rerouting of the messages becomes essential. This is because the overhead of re-transmitting a request is very high. The ability of a network to route its requests determines whether it has a static or dynamic topology.

2.3.3.1 Static Routing

If a network allows rerouting only at the source or at some predetermined points, it is said to be a static routing network. If a packet is blocked at some point in such a network, rerouting can be done only by backtracking the request to a SE in one of the previous stages that lie on the

path, or even back to the source. This is because alternate paths might be available for rerouting only at these intermediate points or at the source.

2.3.3.2 Dynamic Routing

In a dynamic network on the other hand, the switches in each stage of the network can make decisions to reroute the packets as and when the situation arises. The decisions to reroute the messages are made when packets encounter faulty links, switches, or a blockage due to high traffic at a particular node. Dynamic routing is possible in a multipath MIN, because for a given source destination pair, there is an alternate path available at every stage. Chapter 4 describes in detail the dynamic routing capability with emphasis on an IN with intra-stage links.

2.3.4 Operation mode

Networks can be classified into two types based on the operation mode. These modes are synchronous and asynchronous. Synchronous communication is needed for processing in which communication paths are established synchronously for data manipulating functions or instruction broadcasts. On the other hand asynchronous communication is needed for multiprocessing in which connection requests are issued dynamically [Fen81]. There are systems which are designed to facilitate both modes of operation.

Many multiple path networks have been proposed based on the above design considerations. A comparison of the multipath fault tolerant networks and unique path networks shows that the multipath MINs have higher hardware complexity in terms of the switching elements, the number of stages of switches, or the number of switches per stage. In these MINs, the communication protocols generate the necessary control settings on switching elements to ensure reliable data routing. The protocol also provides the necessary handshaking between the

SEs. The selection between static or dynamic routing is based in part on the network topology and the operating mode used. Some fault tolerant networks use static routing - static in the sense that rerouting in case of a fault can be done only at the message's source processor or at some predetermined points in the network. Dynamic routing decisions can be made by the switches in any stage when a fault occurs. In case a message faces a block, due to traffic at a particular spot being heavy, the switch can make a decision to route the message through redundant links to another less congested switch. Of the many different ways of achieving fault tolerance capabilities in networks, the most important way is providing multiple paths between the input and the output. The network which is analyzed in this thesis is a multipath, dynamic MIN with distributed control using packet switching. Multiple paths have been provided in all of the networks that are compared below. Before a survey of the networks is done, an understanding of some general forms of MINs is required. Some of them are Omega, Delta, Baseline [WuF80] and the Shuffle Exchange [Sto71] networks.

2.4 Comparison of Networks

Banyan networks are the most general class of unipath MINs [GoL71]. In these networks, there is a unique path between the source and the destination. A subclass of the banyan networks are called the delta networks [Pat81]. Delta networks have the property of simple and distributed control routing. Omega networks are a subclass of the Delta network. The Omega network is a multistage implementation of the single stage, shuffle exchange network. The properties of the shuffle exchange network have been studied in detail and their ability to realize several permutations have found wide applications in array processing. Most of the present day MIN structures have the shuffle exchange property incorporated in them. The Omega network [Law73], Generalized cube [SiM81], Indirect binary n-cube [Pea77], STARAN flip network [Bat76],

Regular SW banyan with S=F=2 [GoL71], Modified data manipulator [WuF80] are all topologically equivalent to the delta network.

2.4.1 Omega Network

The Omega network [Law73] [Law75] is illustrated in Figure 2.2. The Omega network differs from the Generalized Cube network, in the labeling convention used in numbering the switches. An N-by-N Omega network consists of logN identical stages. Each stage consists of a perfect shuffle interconnection followed by N/2 switching elements of size 2x2. Each of the switching elements can have one of the four states and is shown in Figure 2.3. The SEs may either send their output straight through, interchange, or broadcast one of the inputs to both outputs. The perfect shuffle connection has the property of moving an input with binary representation $s_0s_1s_2s_3....s_n$ to another position $s_1s_2s_3....s_ns_0$ (a left circular shift is performed). The switching function then moves the output to either $s_1s_2s_3....s_n0$ or $s_1s_2s_3....s_n1$. This depends on the bit in the destination tag. A tag is associated with every source and destination. This source and destination tag are binary representations of numbers. Switching elements in each stage of the network (say in stage i), check the i-th bit of the destination tag, i.e., d_i. The check determines the required output channel for the request. The switch that is processing the message is set to switch to the upper output if $d_i=0$ and to the lower output if $d_i=1$. This check at every stage of the network ensures that the message reaches its final destination.

2.4.2 Delta Network

A Delta network [Pat79] having N inputs and N outputs, i.e., of size N, contains n stages of mxm crossbars, where $N=m^n$. The stages are numbered 0 through n-1 each having m^{n-1} switches. The sources are connected to stage 0 and the destinations to the output of stage n-1. The requests are routed in this network by using a routing tag that is the binary representation of the

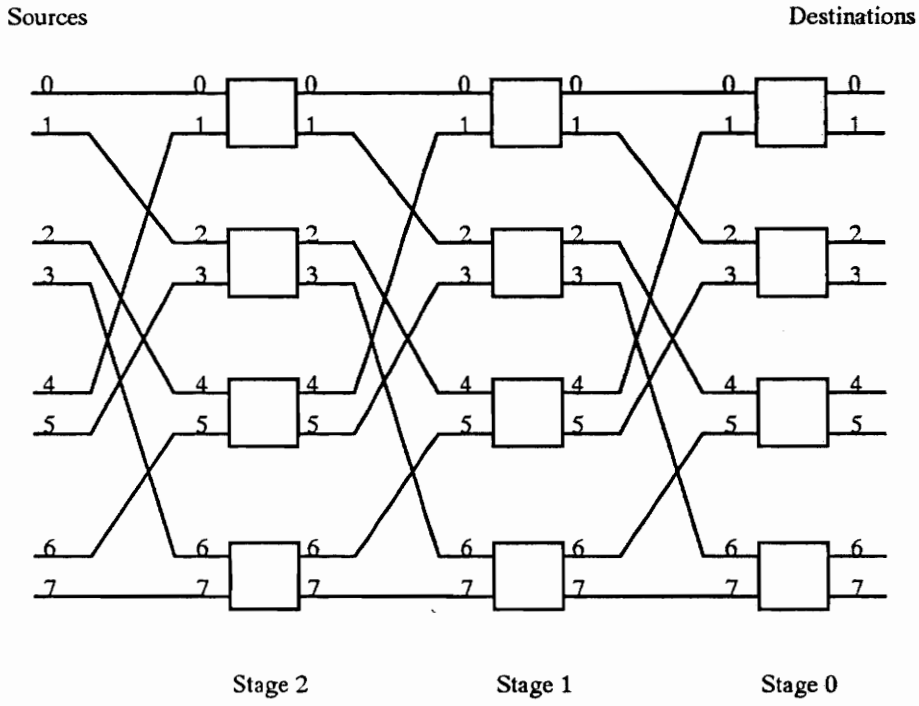

Figure 2.2: Omega Network for N = 8

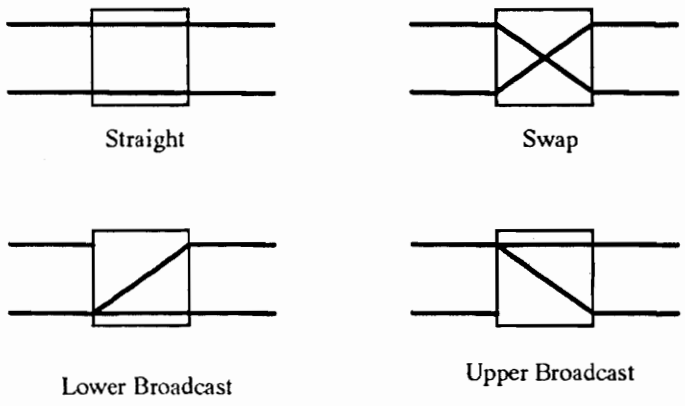

Figure 2.3: Interchange box states

destination. The routing algorithm is simillar to the routing scheme discussed in the Omega networks

2.4.3 Extra Stage Cube

The Extra Stage Cube (ESC) is formed from the multistage cube network by adding an extra stage of switches at the input [AdS82]. The switches in the extra stage and in the final stage, as shown in Figures 2.4(a) and 2.4(b), are equipped with demultiplexers at the input and multiplexers at the output. The ESC also has the bypass capability for the two stages. This extra stage (in all $1+\log_2 N$ stages, and N/2 switches/stage) provides two disjoint paths between every input/output pair. The ESC is exactly one fault tolerant. The network is, however, robust in the presence of multiple faults. In the presence of a fault, a modification of the routing tag becomes necessary for the network to function. The network employs static routing. Since a stage of switches is by-passed in a fault free environment, there exists only one path between a source/destination pair. Hence the ESC continues to fuction like a unique path MIN. The extra hardware introduced into the network is not utilized effectively. The ESC cannot handle multiple initiation requests from processors at the same time.

2.4.4 Dynamic Redundancy Network

The Dynamic Redundancy (DR) network proposed by Jeng and Siegel [JeS86] supports multiprocessor systems using dynamic redundancy for fault tolerance and is shown in Figure 2.5. In the DR network, the system contains spare processing elements (PEs) that are used to replace faulty PEs and hence provide the functionality of a generalized multistage cube network. For a network with N PEs, N I/O ports and S spare PEs, the decoupling network is used to connect N of the N+S PEs to the N I/O ports. The DR network contains $\log_2 N$ stages with N+S switches and 3(N+S) links in each stage. The extra hardware involved in this type network are the S extra

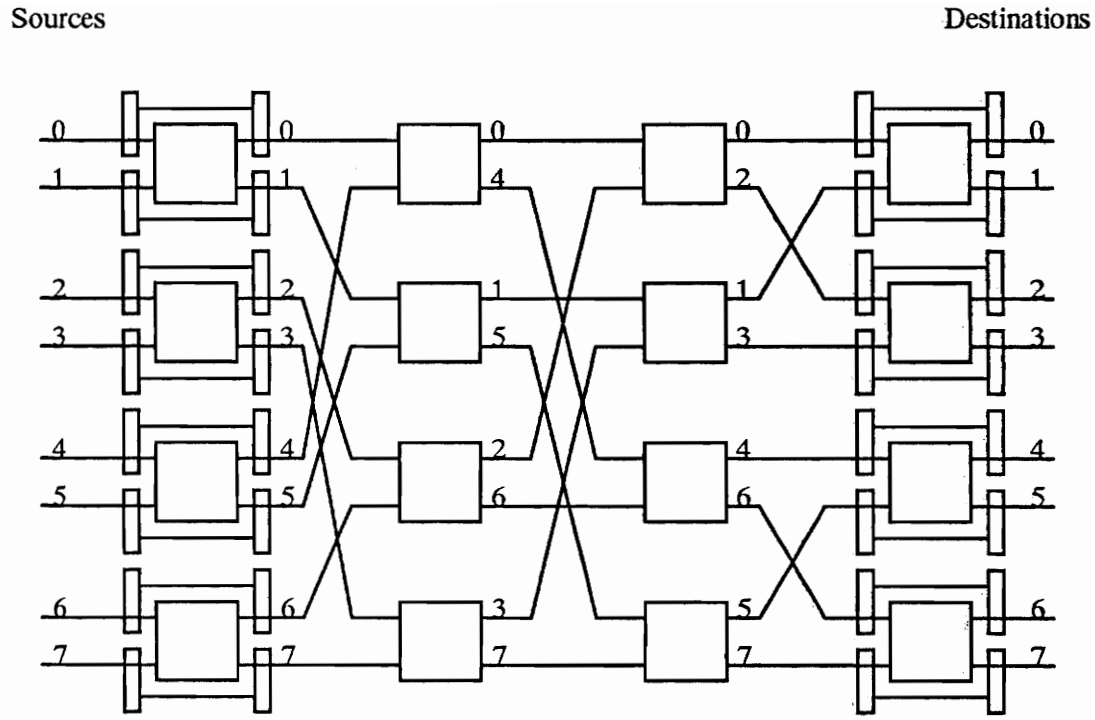

Figure 2.4 (a): Extra Stage Cube for N = 8

Interchange box enabled

Interchange box disabled

Figure 2.4 (b): Interchange box for first and last stage in ESC

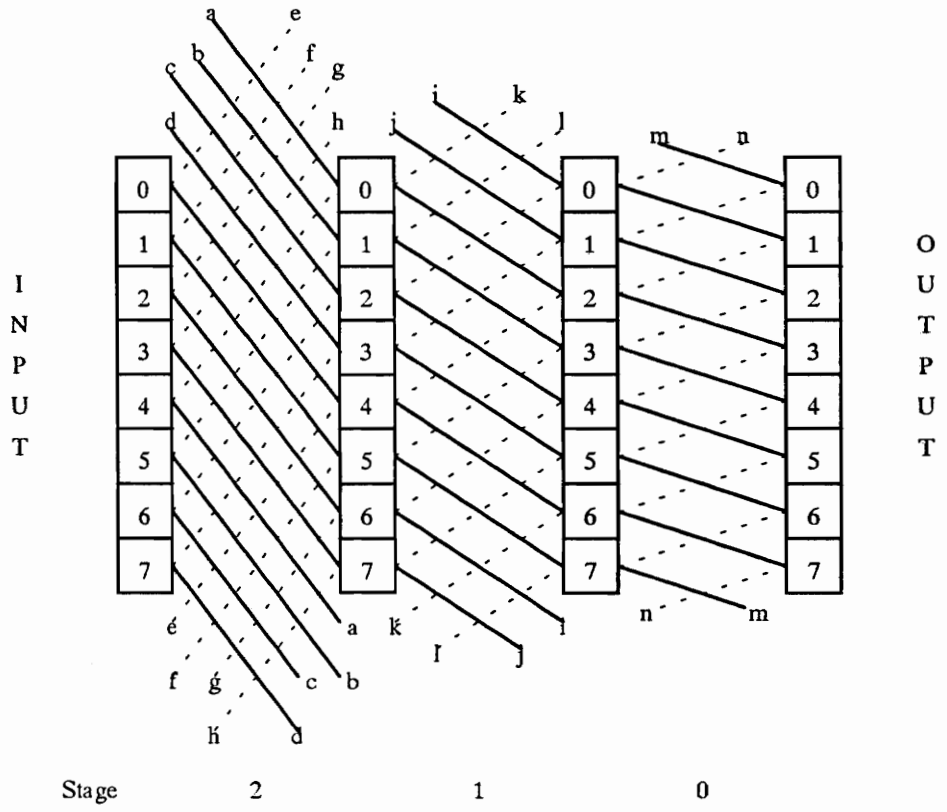

Figure 2.5: Dynamic Redundancy Network

switches in each stage and their associated links, as well as the 3 links from each stage instead of the normal 2 links (for the 2x2 crossbar switch). For an NxN network, where N is large, this extra hardware is quite substantial. Moreover, there is additional circuit delay in the decoupling network, and the problem of designing a fault tolerant decoupling network. Basically, this network is still only single fault tolerant. The advantage of the DR is that it is fault tolerant for faults in the first and the last stages and the link faults in the I/O ports.

2.4.5 Augmented Delta Network

The Augmented Delta Network (ADN) proposed by Dias and Jump [DiJ82] is constructed by adding extra stages to the Delta network. This network has the advantage that it has k disjoint connection paths if kxk switches are used. This makes the number of stages 1+logkN with N/k switches/stage. As before, excluding input and output stage failures, it can tolerate (k-1) component failures. This network can be viewed as an extension of the extra stage cube network.

2.4.6 Baseline Network

The Modified Baseline Network proposed by Wu, Feng and Lin [WuF80] is similar to the ADN discussed above. It is constructed from the baseline network by adding an extra stage at the input side of the network. In this network, the routing algorithm is extended to full communication which allows connections between terminals on the same side of a network. The mathematical model of this network facilitates the possibility of calculating the set of switching elements (or links) on a path, if the two terminals to be connected are known. Its fault tolerant capability is like the Augmented Delta Network for switches of similar structure.

2.4.7 INDRA Network

The INDRA Network [RaV84], an IN designed for reliable architectures, has multiple copies of the basic omega network and is shown in Figure 2.6. The multiple copies of the network provide the redundant paths. They can be considered as the union of R parallel networks each with $\log_R N$ stages. There is an initial distribution stage at the input. The INDRA networks use R-by-R switches, and hence have $(1+\log_R N)$ stages and RN links/stage. The N switches in each stage are connected by links in an Rx(N/R) shuffle. Since there are C copies of the omega network, it requires C-I/O ports per device using the network. Without using the R redundant links to stage 0, there exists R paths between any source/destination pair. Using the redundant links one gets R^2 paths although they are not disjoint. This network is (R-1) fault tolerant and it can tolerate link faults. Even though the routing complexity is comparable to the omega networks, the hardware complexity is much greater.

2.5 Conclusion

In this chapter, the concept of fault tolerance was discussed with special emphasis on multistage interconnection networks. Several unique path and the multipath networks were described. The MINs were compared using cost, complexity of the switches, number of stages, and number of links as design factors. Most of the networks described here achieved fault tolerance by the addition of extra stage(s) of switching elements. The network comparisons were made to highlight the requirements for a fault tolerant network in terms of the design factors mentioned above. The network considered in this investigation achieves fault tolerance by the use of intra-stage links, i.e., links between switching elements in the same stage. The network was chosen after careful considerations of several design factors. The network is described in greater detail in Chapter 3.

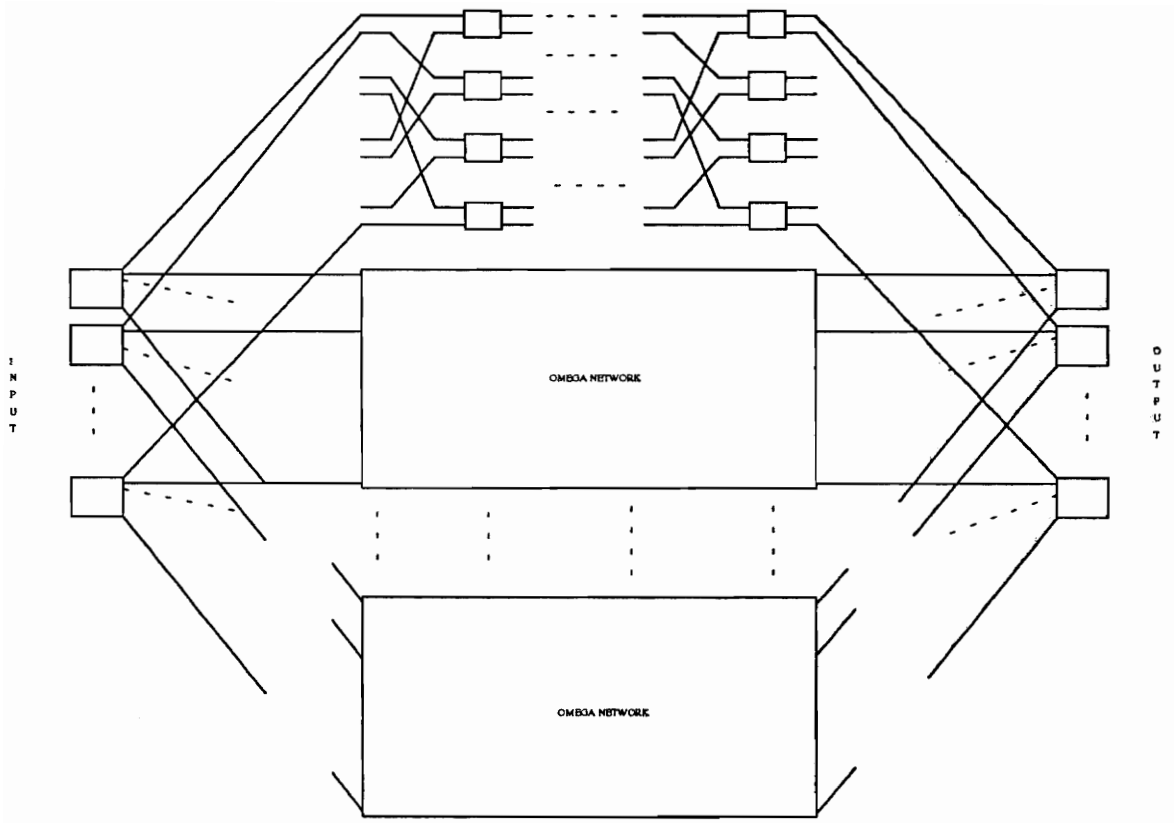

Figure 2.6: INDRA Network

CHAPTER 3

PERFORMANCE CONSIDERATIONS OF ASEN

In this chapter, the Augmented Shuffle Exchange Network (ASEN) is introduced. In Section 3.1, the structure of the ASEN and the early work done by Kumar is highlighted. The properties of the network including the control algorithm, the switching methodology, and the switch modeling is also presented in this section. Section 3.2 outlines the congestion avoidance schemes. An in-depth discussion of the dynamic routing capability follows in Section 3.3. The use of redundant links to overcome the problems of fault in links and switches is also explained in this section. In Section 3.4 the need for queuing mechanisms is explained. A summary is in the last section.

3.1 ASEN Structure

The ASEN is a variation of the multistage cube network. The network is built upon the Omega network which is an isomorphic form of the MSC. Both networks require the same number of stages of switching elements $n=\log_s N$ and N/s switching elements per stage, where N is the number of processing elements in the network and s is the basic switch size. The principle differences in the two networks are in the construction of the switches, the fault-tolerant capabilities, and the routing of messages. The switch sizes used in implementations of ASEN and multistage cube networks for a given network size, differ by one input and one output, except in the last stage where they use the same switches. The ASEN network and its switch structure are shown in Figure 3.1. The augmentation procedure requires the switch size to be scaled from s-by-s for the MSC to s+1-by-s+1 for the ASEN. The increase in the switch size provides redundant

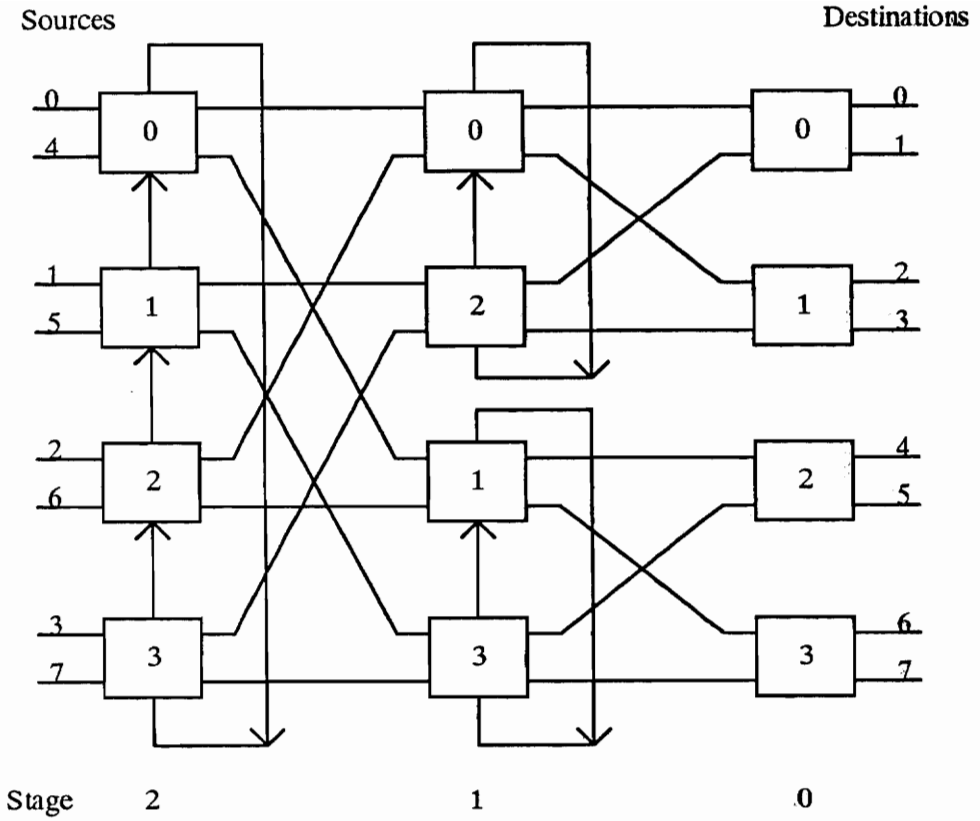

Figure 3.1(a): ASEN network for N = 8

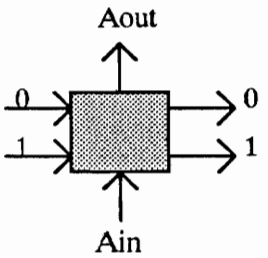

Figure 3.1(b): ASEN switch structure

paths between the switching elements in the same stage while the number of inter-stage links, i.e., the links between two stages of switching elements, remains the same. For an s-by-s network, the number of paths between the source and the destination vary anywhere from two to (s-1). The ASEN uses redundant links within a stage of switching elements to reroute messages when a faulty switch/link or congestion is encountered. Messages are redirected in a cyclic manner within the stage preceding the stage where the fault or congestion is detected. The presence of extra links between SEs in a stage, or intra-stage links in the ASEN increases the number of ways in which a source can reach the destination. These intra-stage links provide the dynamic rerouting capability to the network. Isomorphic forms of the network exist [KuR87] in which switching elements in a stage are chained together to form clusters of two or four switches. These different forms of the ASEN network are known as ASEN-2 and ASEN-4 respectively. The ASEN-2 is shown in Figure 3.2. In all these isomorphic forms, fault tolerance is provided not only to a switching element but to the entire group of switching elements that are in the loop.

Consider the 8-node ASEN shown in Figure 3.1. Suppose that a packet is to be routed from source node 2 to destination node 0. The normal path would be through switches (2,2), (0,1), and (0,0) where (x, y) indicates the switch number, x, and the stage number y. Suppose switch (0,1) is faulty, Switch (2,2) detects the faulty switch and must reroute the packet around the fault. Using the augmented link out of Switch (2,2), the packet is redirected through switches (1,2), (2,1), and then to (0,0).

Kumar's research in circuit switched networks analyzes the delay characteristics of the ASEN network as well as the effects of faults within it [KuR87]. The results of his research show that the ASEN, with longer path lengths, quantitatively outperforms the multistage cube network in a circuit switched environment in terms of message delay. His research analyzes the effects of

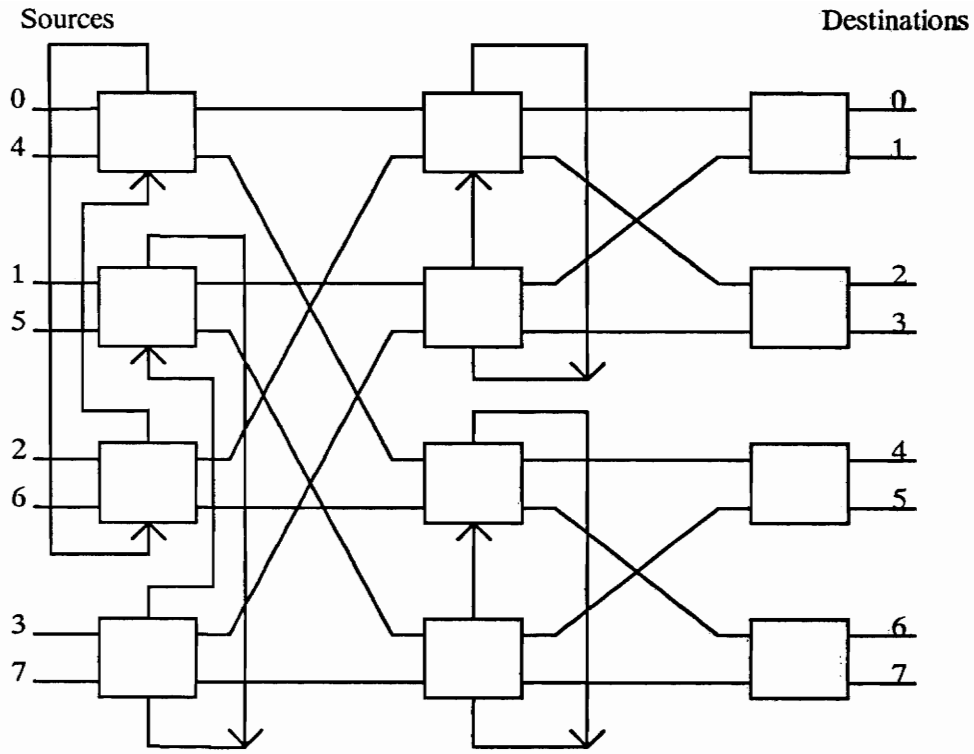

Figure 3.2: ASEN isomorphic form - ASEN-2

component failures on the ASEN performance and the fault recovery characteristics of the network. For fault free networks, the network bandwidth is used as a performance metric for the analysis. The bandwidth is analyzed under symmetric traffic assumptions. In addition to the bandwidth, the performance is also quantified by considering the *probability of acceptance*. This measure is the probability that a request is accepted by a destination without being blocked by other requests or connections in the network. The probability of acceptance is used to compare the ASEN with the crossbar and the other forms of the network. Kumar's research concludes that provision needs to be made for dynamic scheduling, resource allocation, and load balancing. This investigation complements Kumar's work by providing a comparison of one of the ASEN-max and the multistage cube network in a packet switched environment and considers the impact of heavy network congestion.

To understand the logic behind the structure and highlight the dynamic routing capability of the ASEN, the concept of subset of switches [KuR87] is used. This concept is best explained with the help of a tree structure, as shown in Figure 3.3. The tree structure shows that any of the output ports of the network can be reached by any of the input ports. If a tree structure is drawn from a destination node to all of the source nodes, the nodes that are equidistant from the destination node (or the root node) are switching elements in the same stage of the network. In the ASEN network, these switching elements are linked together to form a conjugate subset, through the use of augmented links in the switches. These links enable a blocked message to be rerouted to another switch in the same stage and from there, be routed to the desired destination. The process of connecting the switches that comprise the conjugate subset of destinations to form a loop is continued up to a point when the condition has been satisfied for all the tree structures drawn from the destinations towards the sources. Two interesting observations are obtained from the tree structure. First, any request that arrives at any one of the switches in stage 'i' can be routed to its

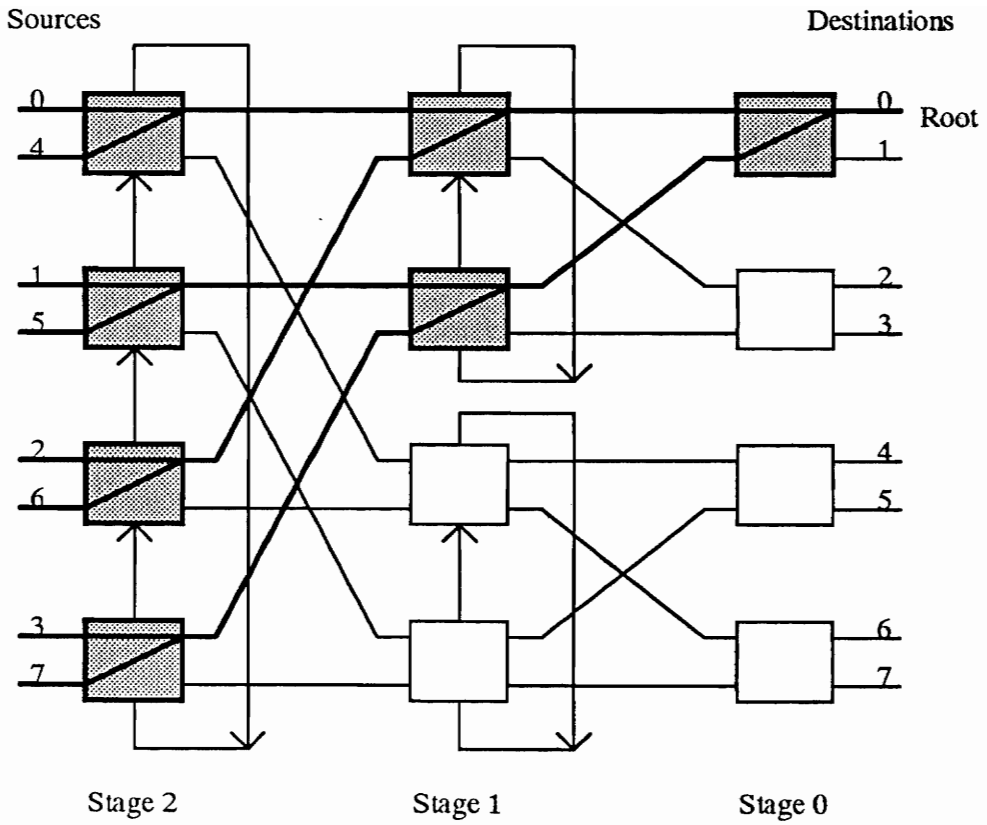

Figure 3.3: ASEN Tree Structure

final destination by any one of the switches that are present in the associated conjugate subset. The second observation is that as one moves along the tree structure towards the source nodes, there are twice as many switching elements connected to the destination node in stage i as there are in stage i-1. The loop formed by connecting the switches in a stage together via the SE (n+1)st augmenting inputs and outputs enables the request to be rerouted to another link in the same stage.

3.2 Congestion Control in an ASEN

In a packet switched network, each packet establishes its own path through the network. This in turn leads to path establishment delays. The logic that is needed to control the routing of the packets and the settings of the switching elements is far more complex than that of a circuit switched environment. A situation can arise when two or more packets vie for the same output port of a SE. Since the output port can handle only one request at an instant, only one request is serviced. In this case, the request that comes in first takes priority. If two or more requests were to come in at the same time, there is another problem. Two alternatives exist here: 1) service one request and drop the other, or 2) service one and hold the other. If the requests were to be dropped the source would have to be informed to regenerate the dropped packet again. On the other hand, holding a packet that lost arbitration requires that buffering mechanisms be provided in the network such that those packets that have been buffered can be serviced as soon as the output port is free. Most networks implement the hold algorithm to avoid the regeneration problem. In the ASEN, the hold algorithm also helps in tracking the worst case delay statistics of packets in addition to optimizing the ease of analysis. The problems seen in congestion situations can easily be overcome if alternate paths are available for rerouting the packets (requests). The rerouting procedure requires a dynamic routing capability, which is present in the ASEN. The next section describes this feature of the network.

3.3 Redundant Links and Dynamic Routing Capability

The ability to dynamically route packets in a network significantly improves the network throughput [Rai87]. This ability is utilized in the ASEN model. It was shown in Chapter 2 that routing decisions for a packet can be made either at the source (static routing), or at an intermediate stage in the network (dynamic routing). In the MSC type of networks, if routing decisions are made at the source, a packet can get stuck at an intermediate location either temporarily due to congestion or permanently if a fault were to occur in a switch or an output link. Since packets are processed on a first-come-first-served basis at an SE, servicing of requests coming in at later instants of time is delayed indefinitely. Meanwhile, as more packets arrive, they start queuing one behind the other. This blockage, even a temporary one, can grow backward towards the source nodes and completely fill the input buffers of switches in preceding stages. Consequently, the network performance falls [PfN85]. In short, as long as decisions are made at the source regarding packet routing, the presence of alternate paths to reroute requests does not improve network throughput. In order to utilize the redundant paths, it is imperative that switches in intermediate stages make intelligent decisions regarding packet routing.

The redundant links of the ASEN network increase the number of alternate paths that are available from a source to a destination. In the presence of a fault, the redundant paths provide graceful degradation and the ASEN is robust for more than one fault. For a fault in a link between stage i and stage i-1 there are s^i-1 additional ways in which a packet can reach the destination that do not use the faulty link. Alternate paths for rerouting packets are available at every switch in an ASEN, except the last stage where the conjugate subsets consist of only one SE. Consequently, the switches in the ASEN are provided with the intelligence to make the rerouting decisions. Moreover, a packet that loses arbitration is assigned a priority and rerouted to another switch in the same conjugate subset. At this second switch, the rerouted packet has priority in accessing the

required output port over other packets that might be queued there. The priority option is to ensure that delayed packets do not suffer unnecessary additional delay at other switches.

3.4 Queuing Mechanisms

Providing buffers in the network is yet another way of achieving improvement in the throughput of the network. Research has been done to study the effects of buffers in an interconnection network [DiJ81]. It has been shown that the network throughput increases by over 50% when two buffers are provided at each input of a switch instead of one. The improvement in the network throughput continues up to a point beyond which increasing the buffer size produces diminishing returns. For varying network sizes, five buffers appear to prevent overflow 99 percent of the time in a simulation study done on the MSC [Rai87]. No significant change in the network performance is seen for larger buffer sizes to warrant the cost of their inclusion.

Using the results of Raines' infinite buffer MSC simulations [Rai87] and pilot simulations of the ASEN as starting points, the models of the ASEN use finite length buffers within each SE. Finite buffer limits allow the cost of building a network to be estimated and provide accurate performance estimates that can be used to select optimal buffer lengths for the ASEN switching element. Since the ASEN has redundant links, the queuing model also takes into consideration the buffer requirement of the augmented links. The queuing mechanism for these augmented links is explained in greater detail in the next chapter.

3.5 Conclusion

This chapter provided a brief introduction to the ASEN network. The earlier work done by Kumar was highlighted. The chapter also gave an idea of the ASEN structure and the capabilities offered by the network. Packet switched environment was chosen to complement

previous research results. The need for buffering mechanisms and congestion avoidance schemes were discussed. The use of redundant links to utilize the dynamic routing capability was then presented. The next chapter details the modeling assumptions for the ASEN. The validation and analysis of the results follow the modeling assumptions. Mathematical models that characterize the ASEN performance from the simulation results are also presented in Chapter 4.

CHAPTER 4

MODELING THE AUGMENTED SHUFFLE EXCHANGE NETWORK

Chapter 4 describes the modeling aspects, results of the simulation analysis, validations and a performance comparison of the ASEN and MSC networks. Section 4.1 describes the use of SLAM as a modeling tool for the multistage cube network. The next section details the assumptions that went into the modeling of the network. Section 4.3 explains the routing algorithm and the extensions of the MSC to the ASEN. Section 4.4 presents the basis on which the network performance comparisons are done. Section 4.5 gives a detailed discussion of the primary performance metric under consideration -- the average time a packet takes to traverse the network. Both uniform and normal distributions of message destinations are considered. Sections 4.6 and 4.7 explain the two other performance criteria -- delay variance and network cost. Section 4.8 introduces mathematical metamodels developed to describe the ASEN performance. Section 4.9 highlights the validation of the ASEN network against the results of previously published work. The last section is a summary of the chapter.

4.1 SLAM as a Modeling Tool

The simulation language used in the modeling of the ASEN is SLAM (Simulation Language for Alternative Modeling) [Pri86]. The justifications for using SLAM are many; SLAM provides the user a simulation environment to cast general systems as a queuing model. Networks fall in this class of queuing systems. The time required to maintain and write concise code when compared to writing the simulation in a high level language such as 'C' is small, and has been justified by its earlier use [Rai87] [McH90]. It is very easy to write and maintain code in SLAM.

The validation procedure focuses on the model itself. The underlying simulator is known to be correct. SLAM, an event driven simulator, has the capability of providing snap shots of data and/or statistics at user defined intervals. Moreover, the simulator is structured for simulating discrete and continuous event models.

To simulate a packet moving through a network, SLAM uses an "entity" object. Attributes associated with each packet entity keep track of relevant data: source and destination addresses, packet length, time of entry into system, packet position in system, output channels, queue number, etc. SLAM uses internal files (different from computer system files) to keep track of the packet entities and their attributes as they traverse the network. All entities (packets) created, and entered into the IN system will leave the network at some point of time. At that instant, SLAM provides the user the option of obtaining statistics about that packet. During a simulation run, the user can specify the entity generation rate, the number of entities, and the intervals at which packet statistics are collected.

4.2 Modeling Assumptions For The ASEN Network

The simulation model of the ASEN is based on the earlier work for the multistage cube network, principally from [RaD88]. The MSC models that form the basis for the ASEN model have been validated against previously published research, including [KrS83] [DiJ81] [AbP86], and have been shown to be highly accurate with their descriptive power [ShD92]. As such, these models provide an efficient way to develop and verify the ASEN simulation models. The models also provide a basis for comparing the performance of the MSC and the ASEN. Central to the earlier research on the MSC is the development, simulation, and validation of the crossbar switch model and network routing algorithms. Model development for the ASEN network modifies the MSC crossbar switch model by adding additional augmented links (scaling the switch size) and

priority queues at each switch to form conjugate set linkage, as shown in Figure 4.1. The underlying simulation algorithm for the switch itself does not change. Because the switch model for the multistage cube network is known to be valid, the ASEN model requires only validation of the new augmented link portions. The verification of the ASEN switch model and packet routing algorithm is accomplished by the use of simulation traces inherent to the SLAM language. These traces provide statistical data for specified packets and queues, as well as for overall network performance.

The modeling and the operating assumptions used in this research are detailed below.

1. The network is assumed to be operating in an MIMD environment with the message passing architecture supported by a set of processor-memory pairs of processing elements (PEs).
2. Packet-switching is used as the method of inter-PE communications. First-in-first-out (FIFO) packet buffers are used to store packets at each input to a switching element in the network.
3. The network load is determined by the message inter arrival rate which is a Poisson process. The source and destination addresses are chosen from uniform or normal distributions depending on the simulation experiment.
4. Messages are assumed to be single packets in length.
5. The unit of measure to determine the average message delay is the packet cycle time. This term is defined as the total time taken by a packet to move from the front of a buffer in a switch to another buffer in the next switch (in the same stage or in the next stage), as specified by the routing algorithm. It includes both SE processing delays and the required transmission delay. Without loss of generality, the packet cycle time is normalized to 1.

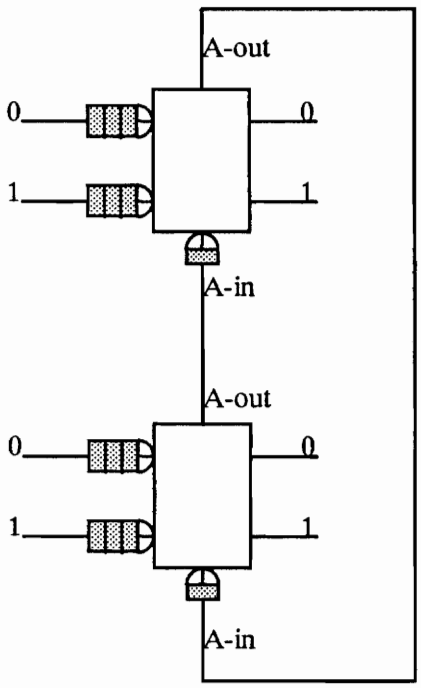

Figure 4.1: Priority Queuing Structure with conjugate subset linkage of the ASEN

6. When a link blockage due to congestion or a fault is encountered, the switching logic determines if the packet should be buffered or routed through an augmented link within the same stage. In the case of a link fault, the switch that is connected to the faulty link treats the link as if it is always in use and therefore, never available. This is to ensure that the packets are sent to another switch in the same conjugate subset, except in the last stage. Transmission or reception across the faulty link is not attempted.
7. A switch fault is modeled by assuming that all of its input links are faulty. As a result, no switch in a previous stage will attempt to route a packet through a faulty switch.
8. Packets that have been sent over an augmented ASEN switching element link are assumed to have a priority over the other packets in the straight links vying for an output port at the next switch.
9. At the network inputs, packet buffers for switches in stage n-1 are modeled as being of infinite length. This assumption ensures that the total time delay from the generation of a packet to its delivery at the network's output is tracked. Finite buffer models can artificially restrict the message generation rates of the external network sources, thus biasing the overall network performance data. Packet buffers for all other stages within the network are modeled as finite buffers. These buffer lengths are set based on the probability of buffer overflow being less than one percent at heavy, non-saturation loading.
10. For a particular simulation run, the source and destination addresses for packets are chosen based on either a uniform or a normal distribution. The normal distribution is defined to have a standard deviation of 0.25N, where N is the number of PEs in the system, about a chosen mean. One mean is chosen for the entire network. The standard deviation is chosen so that network "hot-spots" will be generated while maintaining the full range of source-destination addresses.

11. Network loading factors range from three to 100 percent. A hundred percent loading factor is equivalent to a mean Poisson packet generation rate every 1/N second, where N is the number of PEs in the network [Rai87].
12. The switching function has the capability to decide whether to route the packet to another link in the same stage or to buffer the packet until the output link is free.

4.3 Routing Algorithm

Packet entities created by the Poisson process are prefaced with headers containing vital statistics and routing information. The SE in each stage examines the header and calculates the output port to be used to move the packet closer to its final destination. The routing algorithm is explained with the help of the flow chart shown in Figures 4.2(a) and 4.2(b). Routing can be described as follows in mathematical terms.

- For each switch of size 2-by-2, in stage i ($0 \leq i \leq n-1$), the i^{th} bit of the destination tag is used to specify the output port number. For larger switch sizes two, three, or four bits (in general, for switch size s, $\log_2 s$ bits) are respectively used to locate the output port. For a packet in stage 1 through n-1, if the required output port is busy or is faulty, the packet is assigned a priority and pushed into the augmented out-link to another switch above it and in the same stage (packet rerouting is limited to switches within the same conjugate subset of switches). The packet is held for a unit duration of time in the auxiliary link. A packet in the last stage continues to be buffered until the output port is free. For a faulty switch or link in the first or last stages of the network, multiplexing and demultiplexing switches external to the ASEN can be used to provide fault tolerance [KuR87] [AdS82].

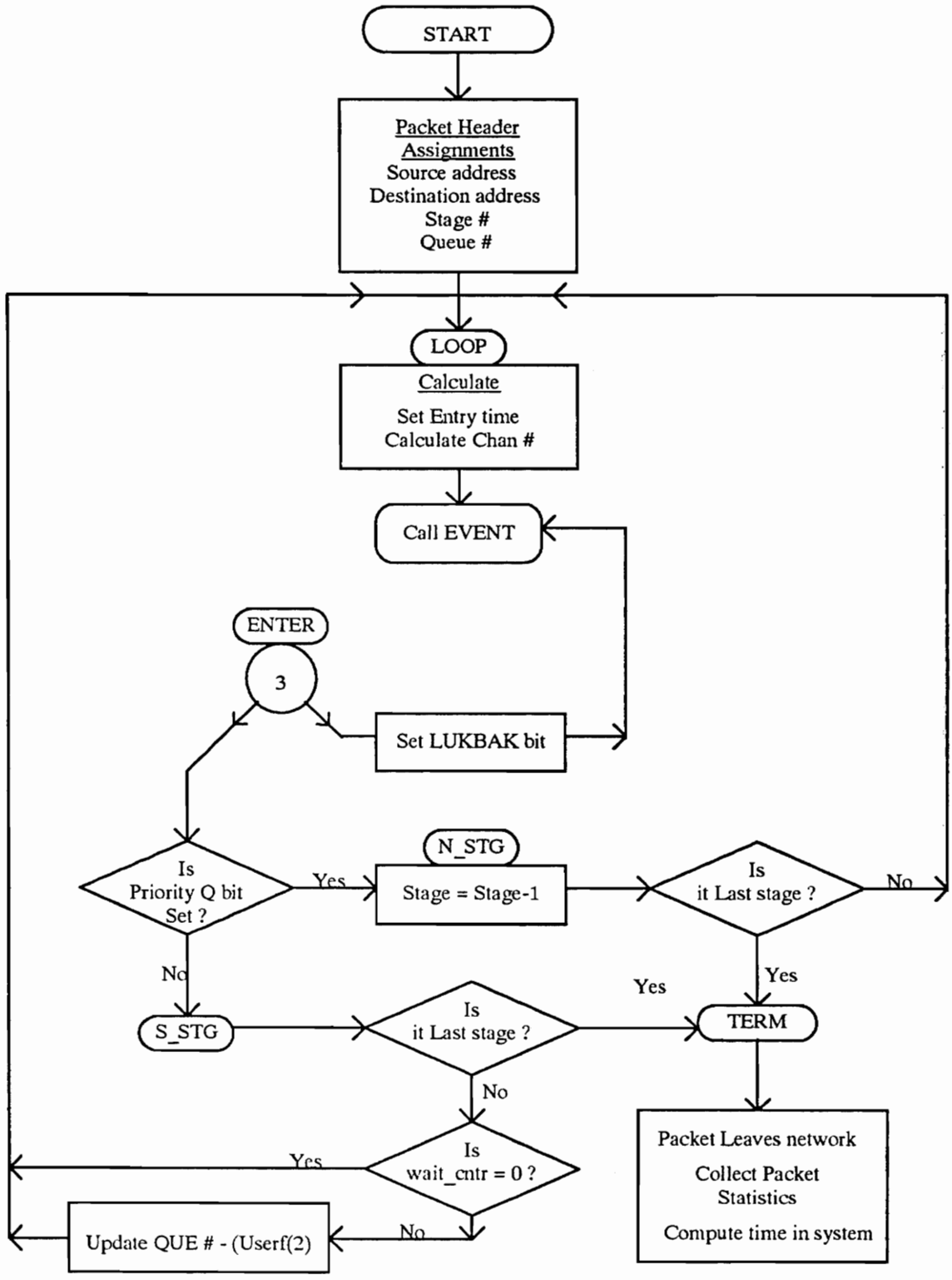

Figure 4.2 (a): Flow Chart of Routing Algorithm

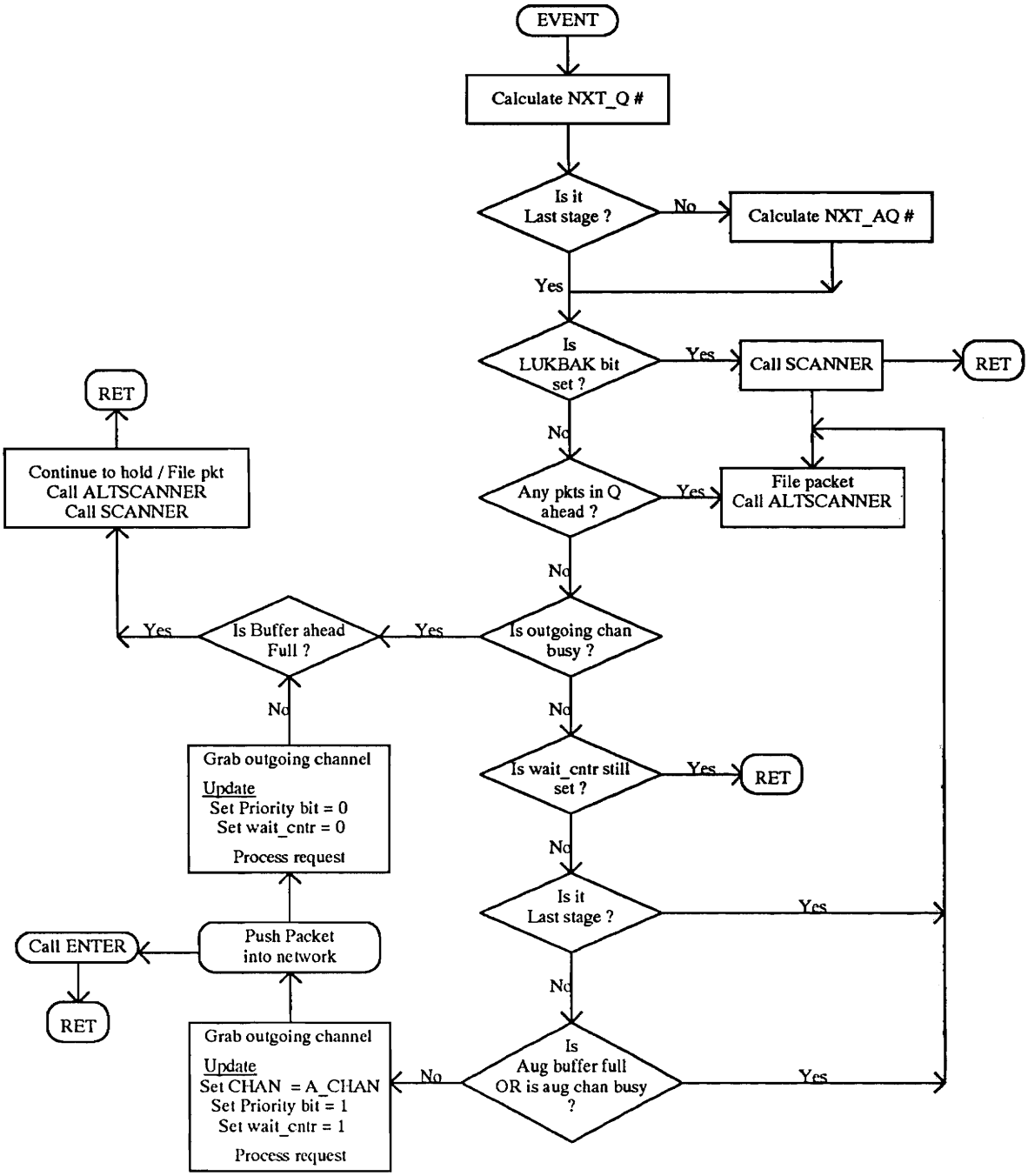

Figure 4.2(b): Flow Chart of Routing Algorithm

- For those packets in stage i ($1 \leq i \leq n-1$) that are sent through the augmented links, if the required output port at the second switch is busy after the wait of one time unit, the output port at the second switch is assumed to be faulty. The packet is then sent to the next switch above it, in the same conjugate subset. The one time unit delay coupled with the packet's priority insures the port being idle under fault free conditions.

- The SEs in each stage process any packets in the augmented links first and then other packets from the regular queues, on a first-come-first-served basis.

- A rerouted packet loses its routing priority after it is transmitted through a regular output port.

It is important that the difference in the routing schemes between the circuit switched as given in [KuR87], and the packet switched environment be highlighted. When using circuit switching, the duration for which a link would be busy serving a request is not known before hand. Hence it is imperative that a new request be rerouted to another switch in the same chain structure to prevent undue delay. If the same idea is implemented in a packet switched network, a situation may arise where packets continue to move in a run-around fashion within the switches in a conjugate subset. To avoid this, the one time unit wait state is added, i.e., the packet is buffered for a unit time at the auxiliary input link of a switch. The packets' priority then ensures that it is placed on the outgoing link in the next scanning sequence of the switch. If the outgoing link is faulty, the packet will continue to wait at the augmented link indefinitely. Pilot runs have verified that a one time unit wait at the augmented link is sufficient and adequate to ensure minimum packet delay. This method of routing packets did improve the run-time performance of the network.

4.4 Network Performance Comparisons

In modeling the ASEN, three network sizes are considered: N=64, N=256, and N=1024. These network sizes are representative of multiprocessor systems that are implementable using current microprocessor technology. Each of these ASEN networks can be constructed using crossbar switches of different sizes. For a 64-PE ASEN, the switch sizes not counting the augmenting links, of 2-by-2, 4-by-4, and 8-by-8 can be used to implement the network. For N=256, permissible switch sizes are 2-by-2, 4-by-4, and 16-by-16. The ASEN of size N=1024 can be implemented using 2-by-2, 4-by-4, or 32-by-32 switches. Multistage cube networks of various sizes can be implemented with switches of the same size used by the ASEN. Both the MSC and the ASEN network models were evaluated using loading values ranging from 3% to 100%. It is shown below that the network performance depends primarily on the size of the network, the size of the switch used, the loading factor, and the distribution used to generate the packets. Because of these interdependent parameters, it is necessary to jointly vary these factors when analyzing the network performance. Numerous performance characteristics are obtainable from a network modeling study. This research focuses on three measures: the average delay a packet experiences while traversing the network, the delay variance, and the cost associated with providing "sufficient" buffer space within the network. Mathematical models that characterize the network performance result from the analysis of the packet delays. These models are presented and discussed later in this chapter. Analysis of the ASEN and multistage cube network simulations reveals trends that are consistent across various network sizes. Because of this consistency, only results for a representative network of size N=256 are discussed here. Performance plots for other network sizes are summarized in Appendix A and B.

The most important performance parameter of this investigation is the average time required for a packet to traverse the network. The minimum delay with which a packet can

traverse the network is logN (the number of stages). A network can either be in a steady state operating condition or in saturation where the packet delay increases without bound. To accurately analyze and represent the network performance through mathematical or simulation models, the network must be studied under steady state conditions. If the network is not in steady state, the figures obtained through simulation are not representative of the network's true capabilities. Consequently, the analysis presented is only for the steady-state performance characteristics of the network. Steady state delays depend primarily on the network loading, the number of buffers provided in the network, and the number of servicing links and queues.

4.5 Average Time in System

The average time spent by a packet in a network is one of the most important metrics used to characterize the performance of the overall network. This figure represents the capability of the network to handle traffic either in normal or burst mode across varying loading values . The average time can be examined under different combinations of loading factors, source-destination distributions, network sizes, and switch sizes. Figures 4.3 to 4.5 show the time spent by a packet in an ASEN of size N=256 for variations in the factors mentioned above. The trends in the behavior of the network across varying network sizes are the same. Several performance characteristics of the network can be inferred from these trends. The performance comparison between the MSC and the ASEN networks is done under two different distribution functions: uniform distribution and normal distribution. The normal distribution function simulates the condition of "hot-spots."

4.5.1 Uniform Distribution

Looking at Figure 4.3, it can be seen that for loading values from 0 percent to 33 percent both MSC and ASEN (for a switch size of 2-by-2) have similar performance. As the loading

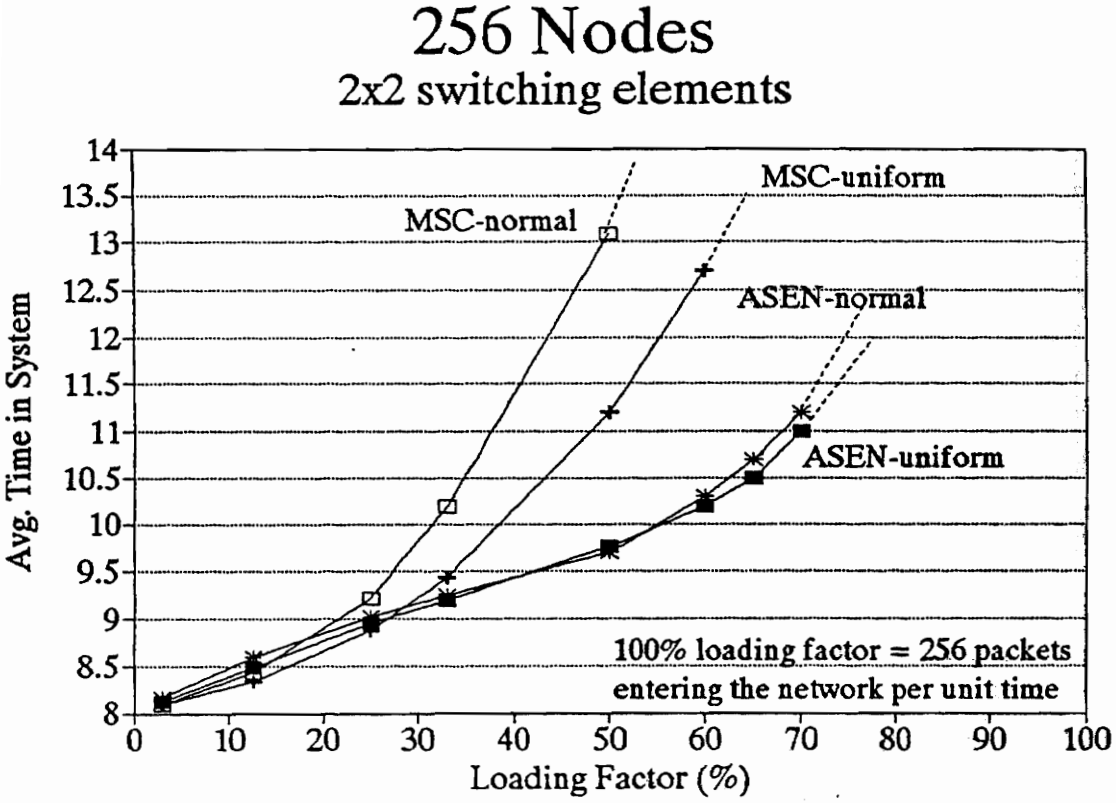

Figure 4.3: Average Message Delay vs. Network Loading

(256-PE network using 2-by-2 switches)

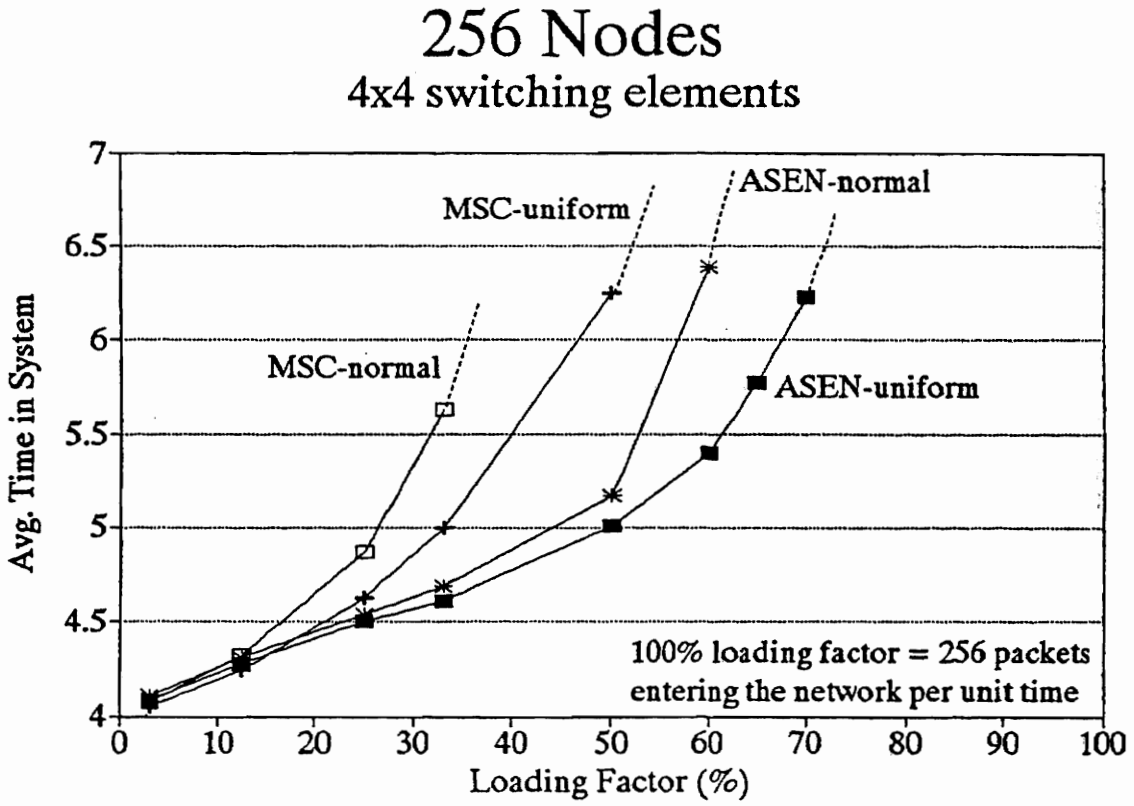

Figure 4.4: Average Message Delay vs. Network Loading

(256-PE network using 4-by-4 switches)

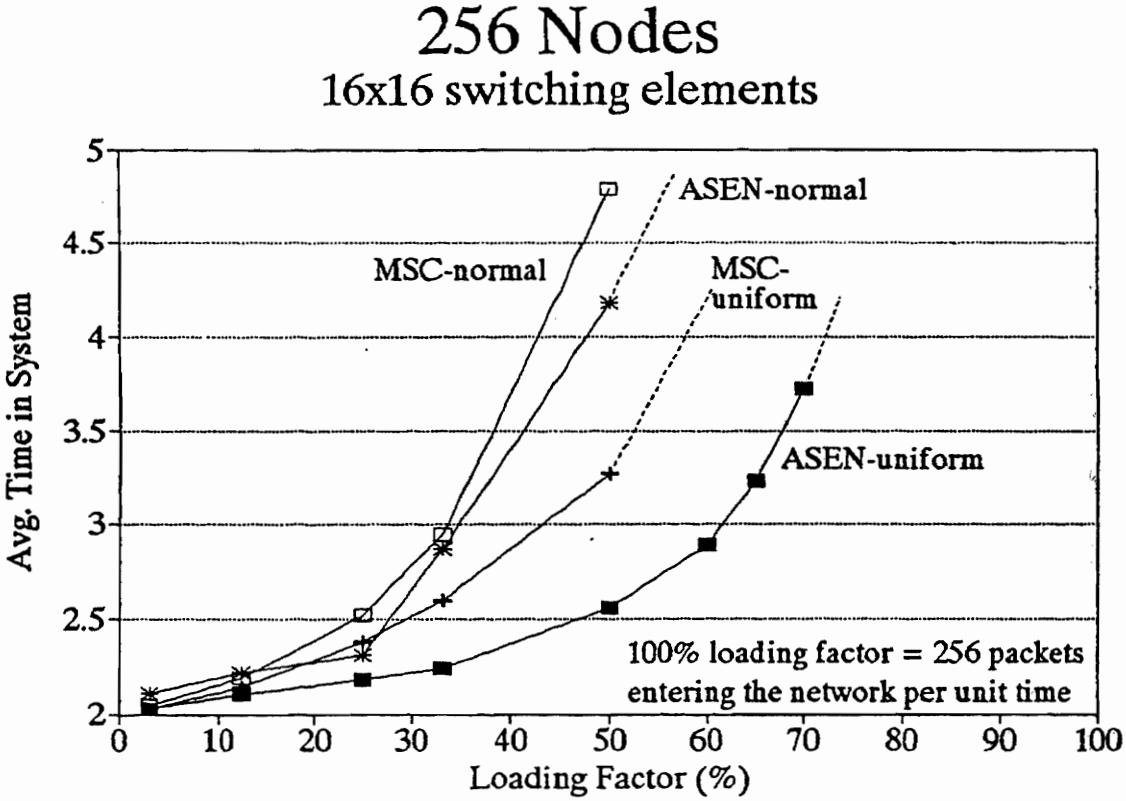

Figure 4.5: Average Message Delay vs. Network Loading

(256-PE network using 16-by-16 switches)

values increase, the performance curves tend to diverge. The MSC saturates between loading values of 55 percent to 60 percent. Under similar conditions, the ASEN withstands greater loads and continues to route packets through the network without saturation and experiences average delay times which are approximately 20 percent less than the MSC. For the case where 2-by-2 switches are used, the divergence of the delay curves is quite dramatic, with the ASEN saturating at 70 percent loading value. The non-saturation performance increase in the ASEN is approximately 17 percent.

Effect of Loading and Switch Size

An interesting observation about the ASEN is the effect of switch size on the delay performance curves as evident from Figures 4.3 to 4.5. The ASEN curves start diverging from the MSC at earlier loading factors (25 percent for 4-by-4 switch size and around 15 percent for 16-by-16 switch size). Figure 4.4 shows that the use of 4-by-4 switches allows the ASEN to route packets with an average delay that is 25 percent less than the MSC at a loading factor of 50 percent, using a uniform source-destination distribution. The performance improvement is more dramatic for the same distribution at 60 percent loading, where the ASEN average packet delay is approximately 40 percent less than the MSC. With the increase in the switch size the ASEN can no longer withstand a loading factor of 70 percent. The major attributing factor to the ASEN's superior performance when using 2-by-2 switches is that the redundant links provide a 50 percent increase in the number of queues for the stages that use the conjugate subsets. When the 4-by-4 switch size is used instead the of 2-by-2, there is only a 25 percent increase in the number of extra queues, while the increase for the 16-by-16 switch size is only 17 percent over the corresponding figure for the MSC. The extra links and queues help in easing the load on the switch where either contention for an output port, or congestion, or even a fault, is present. Though the percentage

increase in augmented links and priority queues decreases as the ASEN switch size is increased, the ASEN performance relative to the MSC does not degrade.

4.5.2 Normal Distribution

Hot-spots can be modelled by the generation of packets over whole a range of destination addresses with a preferential concentration around a mean destination address. In this investigation, packet destinations are chosen using a normal distribution with a specified mean and standard deviation. The value chosen for the standard deviation was 1/4 the network size. This value assured that enough packets were created around the mean to simulate a congestion and/or hot-spot and to ensure at the same time that message generation spread over the whole range of destination addresses. The mean was any integer chosen over the range of addresses and was consistent across both MSC and ASEN networks.

The effect of larger switch sizes on the delay performance of the ASEN using a normal distribution to simulate the effect of hot-spots can also be seen in Figures 4.3 to 4.5. Hot-spots, as shown by Pfister and Norton [PfN85], cause a disproportionate number of packets to be routed to a subset of the destinations, causing the switch buffers along the paths to quickly fill. At higher loading factors, the packet generation rate is greater than the rate at which packets are removed from the buffers, leading to network saturation. This effect is clearly seen in the MSC. Under similar operating conditions, the ASEN performance was found to be better than the MSC. Performance improvement is a direct result of the redundant intra-stage links, buffers provided, and the alternate routing paths that they create. These links ease the load on those switches whose buffers are filled, by routing packets to other switches in the same conjugate subset. To appreciate the full impact of this, one needs to look at the performance curves for a 2-by-2 crossbar switch of the ASEN see Figure 4.3. The curve for the normal distribution with hot-spots very closely

approximates the curve for the uniform loading. On the other hand, there exists a vast difference in the performance curves for the MSC with uniform and normal distribution. The difference is due to the absence of alternate paths and the dynamic rerouting capability.

Effect of Loading and Switch Size

The performance of the ASEN for the same switch size under normal distribution is even better than that of the MSC. The increase in performance is approximately 25 percent. The curve for the statistically normal source-destination distribution follows the curve for the uniform distribution. Saturation begins around 70 percent loading, whereas the MSC saturates at 50 percent loading. At a loading factor of 50 percent, where the MSC saturates, there is nearly a 40 percent difference in the performance of the two networks. As in the case of the uniform distribution, the ASEN's superior performance when using 2-by-2 switches is directly attributed to the alternative routing paths and associated queues. The intra-stage links ease the load on the ordinarily blocked paths by routing packets to less congested links within the same conjugate subset.

A comparison of the two networks using a normal packet distribution reveals that the ASEN packet delay is approximately 40 percent less than the MSC delay at 50 and 60 percent loading. Figure 4.5, for 16-by-16 switches, reflects the same trends seen in Figures 4.3 and 4.4. At loading factors greater than 50 percent, the packet delay performance of the two networks differs from 25 to 40 percent, depending upon the source-destination distribution chosen. As the switch sizes are increased in the ASEN, the delay curves for the uniform and normal distributions tend to diverge at higher loading factors. The divergence stems from the fact that the percentage of augmented links decreases as the switch size increases and thus the number of alternate paths is reduced. The onset of congestion is not as easily avoided. The ASEN, even under a normal

distribution, outperforms the MSC of same size operating under a uniform source-destination distribution for a switch size of 2-by-2 or 4-by-4. For larger switch sizes, the ASEN normal distribution curve tends to overlap with the MSC uniform distribution curves.

4.5.3 Effect of Buffer Size

Another factor that affects the performance of the networks under both uniform and normal distributions is the buffer size used within the SEs. To detect the optimum buffer size of the network with a tolerable drop in the performance of the network, the buffer limit on the SE is varied from two through five. The ASEN performance reveals interesting trends over the entire range of variation. It is seen that the network saturates at smaller loading factors (approximately 45% loading) for a buffer size of two. In step with increments in buffer size, the network performance improves, i.e., the saturation occurs at higher loading factors, up to a point. For buffer sizes more than five, there was no significant effect on network performance. This result is in complete agreement with the previous findings from Dias and Jump's analysis of buffered Delta networks. The reason for the early saturation of the network for smaller buffer sizes is that packets suffer additional delay waiting at the SEs in the previous stages if the buffers in the queue ahead are filled. This delay is responsible for the decrease in the throughput of the network. Pilot simulation runs performed with an infinite buffer capacity show that a buffer size of three packets is never exceeded until the point of saturation. Table 1 shows the number of buffers required for both MSC and ASEN for various network sizes with a 1 percent chance of overflow. Subsequent simulation runs of the ASEN are hence performed with finite limits on the buffer.

4.6 Delay Variance

The delay variance is the second performance measure that was analyzed. The analysis of the delay variance follows directly from the packet delay statistics. The coefficient of variance

Table 1: Buffer Requirements for the ASEN and MSC

(Uniform Distribution)

1 buffer = unit cost

NUMBER OF BUFFERS IN THE NETWORK*				
Size	**Network**	**64**	**256**	**1024**
2x2	MSC	2304	12288	61440
	ASEN	2080	11136	55808
4x4	MSC	1152	6144	30720
	ASEN	1024	5504	26624
8x8	MSC	768	Not	Not
	ASEN	648	Used	Used
16x16	MSC	Not	3072	Not
	ASEN	Used	2576	Used
32x32	MSC	Not	Not	12288
	ASEN	Used	Used	10272

*MSC: 6 buffers per queue; ASEN: 5 buffers per queue

(CV), the ratio of the standard deviation to the average time in system, is a convenient way to gauge the variation of the average time at various loading factors. To get a small CV, either the average time in system should be large or the standard deviation small. The CV can also be expressed as a percentage to quantify the variation. From the analysis, the delay variance is found to be approximately the same for both MSC and ASEN networks across various switch sizes. According to the operating assumptions, the minimum delay encountered by a packet at any switch is one time unit. This delay occurs either when waiting at the input buffer of a queue on being blocked or when rerouted through an augmented link to another switch in the same conjugate subset. The worst case delay variance in the ASEN is 1.42 time units (for a 2-by-2 switch). The corresponding variance for the MSC is 1.46 time units. The effect of delay is more pronounced in network implementations with fewer stages. This fact can be better explained with the help of an example. Consider a network with N=256 that has a 16-by-16 switch size in which the destinations are chosen through a uniform distribution. The minimum time taken by packets to traverse the network is two time units ($\log_{16}256 = 2$). Simulation results show that on average at saturation, the additional delay suffered by a packet is 1.72 time units, which corresponds to a worst case coefficient of variance equal to 46.2 percent. On the other hand, for an identical ASEN that uses 2-by-2 switches instead of 16-by-16, the delay variance is small. The minimum delay encountered by packets while traversing this network with eight stages is eight time units, while the additional delay suffered by a packet in the worst case is just 2.5 time units. Hence the additional delay suffered by packets in networks with fewer stages is comparable to the minimum delay and is significant. The validity of the argument is also verified from the mathematical model of the network (explained in a later section). Table 2 summarizes the delay variance for the ASEN and the MSC networks.

Table 2: Coefficient of Delay Variance Figures for MSC and ASEN

Network	Min. delay	Avg. Additional delay	Variance in %
MSC (2-by-2)	8	1.46	15.43%
ASEN (2-by-2)	8	1.42	15.07%
MSC (4-by-4)	4	1.77	30.67%
ASEN (4-by-4)	4	1.58	28.31%
MSC (16-by-16)	2	2.15	51.8%
ASEN (16-by-16)	2	1.72	46.23%

Average delay values observed at saturation under uniform distribution for a 256 node network.

4.7 Network Cost

Network costs are a function of the switches, wiring and queues required to construct the network. An implementation cost comparison between the MSC network and the ASEN is considered here for the following conditions. First, space in an input queue at any switch is defined for unit packet lengths, i.e., a queue capable of storing 5 packets equates to 5 buffers. Second, the network cost is dominated by the total number of buffers used to implement the network queues. Finally, one buffer equates to unit cost. Table 1 shows the cost associated with implementing both the MSC and the ASEN of various network and switch sizes. It is to be noted that the figures for the ASEN network include the buffers provided for the augmented links. Implicit to Table 1 is the length of each switch queue. Analysis of the network packet delays with an infinite buffer capacity in steady-state show that to achieve a queue overflow rate of less than one percent, the ASEN network requires queues to be five packets in length. The effect of queue limits on the network performance is consistent across the several different network sizes that are analyzed. The algorithm adopted in routing a packet and the method by which packets are pushed into the network play a major role in the network performance. In accordance with the modeling assumptions made earlier in this chapter, the first stage essentially becomes the bottle neck for the network. Packets tend to get queued at stage (n-1) at high loading factors. The buffer limit for stage (n-1) is hence set to infinity. This limit ensures that the maximum time a packet takes to traverse the network is tracked. The queue limits for the rest of the stages of switching elements are then varied to study the effect on the performance curves. Queue lengths in the intermediate stages of the network depend on how fast the previous stages processes packets. The simulation results exhibit certain interesting facts on the network tolerance. As mentioned earlier, a queue limit of three for the inter-stage link can take the network until the point of saturation (70 percent loading). A buffer overflow of 6 percent was noticed for a queue limit of four. The queue size for

the augmented links analyzed through simulation runs found that a buffer size of one packet is adequate for all loading factors. The buffer overflow results are summarized in Table 3.

4.8 Mathematical Model for the ASEN

Another important aspect derived from the analysis of the ASEN and the MSC is a mathematical model that describes the network performance behavior. Metamodeling [Agr85] provides an avenue for mathematically expressing a system model in terms of system parameters that effect the model. Using the statistical method of Analysis of Variance (ANOVA) [LaM86], linear equations are derived which characterize the performance of the respective networks. The degrees to which network size, switch size, and loading factor interact is of importance in the analysis of the ASEN and the MSC. The time delay, D_{ijk}, that a packet will experience in the MSC is characterized by the relation in Equation 1, taken from [ShD92]. The relation describes the delay for a particular architecture as a function of:

- variations in the number of processing elements in the system, N_i
- network loading factor, L_j
- switch size, S_k
- joint interaction among factors (NL, NS, LS, NLS)
- and a lumped error term, ε_{ijk}.

Table 4 shows the experiment design data used for the packet delay analysis. The general form of the delay relation is:

$$D_{ijk} = N_i + L_j + S_k + NL_{ij} + NS_{ik} + LS_{jk} + NLS_{ijk} + \varepsilon_{ijk} \quad (1)$$

The ability of this relation to accurately capture the variation in the delay is verified through ANOVA using the SAS program. For the statistical analysis, a procedure in SAS, called

Table 3: Buffer Overflow Results for MSC and ASEN

	Loading Values	50%	60%	65%	70%
MSC	3 buffers overflow in %	5.73%	8.59% (sat)	17.45%	
	6 buffers overflow in %	< 1%	1.34% (sat)		
ASEN	3 buffers overflow in %	2.68%	4.69%	9.37%	11.16% (sat)
	5 buffers overflow in %	< 1%	< 1%	1.4%	3.68% (sat)

Table 4. Experimental ANOVA Results

(Uniform Distribution)

ANOVA for Packet Time in System			
Multistage Cube Network*			
SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
Model	89	16984.41	190.84
Error	180	2.93	0.02
Corrected Total	269	16987.34	
Model F =	11717.59	RF > F= 0.0	
R^2	C.V.	Root MSE	Mean
0.999827	1.4470	0.1276	8.82
Augmented Shuffle Exchange Network			
SOURCE	DF	SUM OF SQUARES	MEAN SQUARE
Model	53	1836.31	34.64
Error	206	1.009	0.005
Corrected Total	259	1837.31	
Model F =	7073.21	RF > F= 0.0	
R^2	C.V.	Root MSE	Mean
0.999451	1.344680	0.0700	5.2048

*[ShD92]

PROC GLM (Generalized Linear Models Procedure) is performed on the data obtained from the simulation run. GLM helps in identifying significant factors and interaction terms of the factors that play a meaningful role in describing the performance of the network. In the second step, integer values are assigned to those factors whose numerical value does not affect the computation. In this case, each switch size is assigned an integer value in the range from one to five, e.g., 1 is assigned for a 2-by-2 switch, 2 is assigned to a 4-by-4 switch, etc. The reason for assigning an integer value is that, the differences in the equation obtained with each switch size can now be analyzed. The factors N and L are assigned the values that appear in Table 4. In the GLM analysis, the R^2 parameter measures how much variation in the model's dependent variables can be accounted for by the model parameters. The closer the R^2 value is to 1.0, the better the model fits the experimental data. Evident from Table 4 is the high descriptive power of the multistage cube model (R^2 = 0.9998) and the ASEN model (R^2 = 0.9995). The drawback of using the R^2 parameter is that R^2 keeps increasing as one includes more model terms, regardless of whether these model terms are significant or not. Furthermore, there is no significant test for the R^2 value obtained. Hence, another value, the adjusted R^2 (or adjusted R^2) value is used as it compensates for the first drawback, as an increase in the adjusted R^2 signifies a good fit. Moreover the adjusted R^2 value does not necessarily increase even if additional terms are added. There are three numerical values that one needs to observe in the ANOVA results to determine the exactness of the model. They are:

- The R^2 (or adjusted R^2) value.
- The ANOVA overall F-values. A large F-value for the model signifies a good fit.
- The p-values of the individual parameter estimates

The smaller the individual parameter estimate value (p-value) is, the more significant is the model term. The individual estimates of the factors whose p-values are high (p-value > 0.0001) are considered insignificant and excluded from the model. The general model for network delay in Equation 1 is then used to formulate regression equations to predict actual delay values based on the observed simulation data. Using SAS, the time delay regression equation for the ASEN network is:

$$\tau = x_0 + x_1N + x_2L + x_3NL + x_4S + \varepsilon \qquad (2)$$

Equation 2 omits factors from Equation 1 that have been determined to be statistically insignificant by the ANOVA PROC REG (Regression Procedure) analysis. The x_i coefficients for each network model along with their adjusted R-square values are shown in Table 5. The high adjusted R-square values indicate that the equations accurately predict the network delay values. The above relation for τ, can now be used to accurately predict the delay characteristics of the network for variations in the model parameters included in the equation. From this re-constructed model, the point estimates can be calculated for variations in values of several factors.

The usefulness of metamodels is shown through the following example. Consider a 256 node ASEN with a uniform source-destination distribution implemented using 2-by-2 switches. For a loading factor of 60 percent which is equivalent to .6 x N packets per unit time, Equation 2 estimates that the packet delay will be 10.14 units. Figure 4.3 shows that the average packet delay is approximately 10.2 units for 60 percent loading, a difference of less than 1 percent. Similarly, taking the partial derivative of Equation 2 with respect to loading, the incremental change in delay given a unit change in loading is:

$$\frac{\partial \tau}{\partial L} = x_2 + x_3N \qquad (3)$$

Table 5: Least Squares Estimates for Packet Delay Models

(Uniform Distribution)

Network Architecture	Models' Adjusted R^2	Model Parameters				
		Intercept	Nodes (N)	Load (L)	NxL	Switch (S)
MSC*	0.7103	4.06375	**	0.03282	-.000030	-2.6602
ASEN	0.8469	8.21154	0.0027	0.02660	-.000014	-2.2944
** Coefficient is not significant						

*[ShD92]

Similar regression metamodels can be formulated for other dependent variables of interest such as the delay variance of delay or the maximum delay time. Figure 4.6 shows the residual plot of the difference in the actual delay value obtained from the simulation and the reconstructed value from the mathematical model of the network. The differences as seen from the plot, are scattered on either side of the zero mark, again emphasizing the exactness of the mathematical model in predicting the behavior of the ASEN network for variations of the network parameters.

4.9 Comparison of ASEN with Other IN Topologies

The simulation and analysis results for the ASEN indicate that it has the capability to perform better that the MSC networks. Consequently, it is of interest to compare the performance of the ASEN with other IN topologies that are also fault tolerant. This section details the differences between ASEN and some architectures that were proposed to overcome the drawbacks of the MSC class of unique path networks.

The Extra Stage Cube (ESC), described in Chapter 2, provides the network exactly two alternate ways to route packets. The addition of an extra stage of switching elements with demultiplexing and multiplexing arrangement in the first and last stage provide the network two distinct paths for a packet to reach the destination. In a fault free environment, the network functions essentially like the MSC, since the extra stage is not included in the network. The ESC implementation structure does not take advantage of the redundant path, either during congestion or during conflicts. Hence the performance curve for the ESC fundamentally follows that of the MSC. In a similar operating environment, the ASEN provides a packet *at least* two ways, up to n-1 ways, of reaching the destination. The convincing role played by the redundant links in relieving the load on the network during congestion, contention or fault, has been demonstrated through the results in this chapter. The use of a good routing algorithm and adaptive routing capability in the

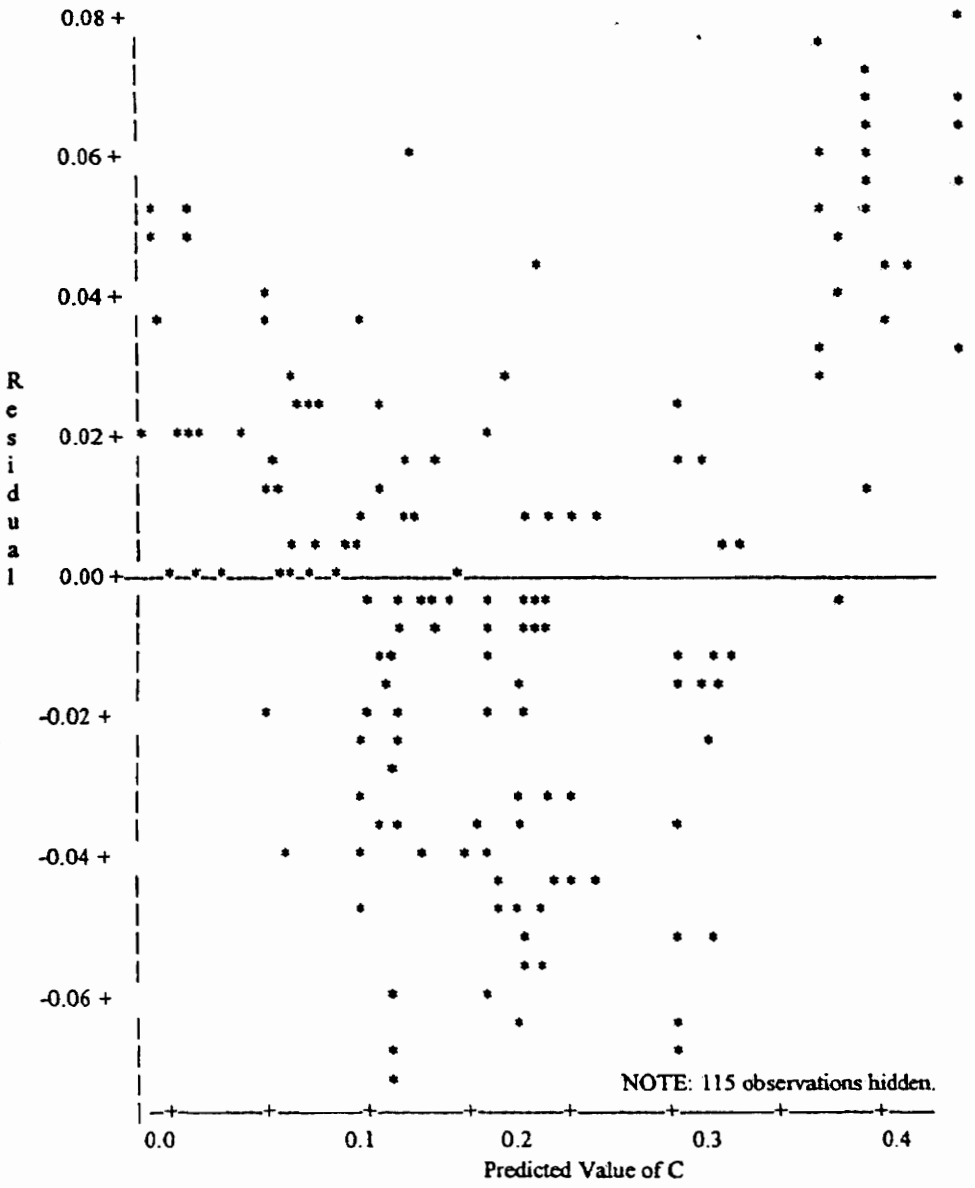

Figure 4.6: Residual Plot (Delay Value vs. Model Value)

ASEN in improving the performance of the network in a fault free environment has been verified in this investigation. A judicious choice in the number of buffers provided to the switches can result in a compromise in the cost versus performance of the network, with a marginal drop in throughput. In addition, the ASEN has a uniform switch structure that is modular in nature. The complexities in the construction of the first and the last stages and the difficulties in their switching complexity of the ESC are avoided.

Other network structures discussed in Chapter 2, like the Dynamic Redundancy Network, Augmented Delta Network, Baseline Network, INDRA Networks require a modeling study. The structure of these networks, although similar to the MSC class of networks, is different and has routing algorithms and operating conditions that may be significantly different from the assumptions made in this investigation. Hence a comparison between these networks will not be a true indication of the network's capability.

4.10 Summary

In this chapter, the modeling considerations of the ASEN were discussed. Besides being a fault tolerant network, the improvement in performance obtained with the redundant links was highlighted. This was followed by a detailed comparison of the performance of the ASEN and the MSC networks. The effect of switch size, network size, loading factor, and the source-destination distribution functions on the average time in system of a packet was analyzed in depth. Other performance criteria like delay variance, cost of buffers were explained in later sections. A mathematical model was also developed to aid the network designer.

The performance benefits of the ASEN can be attributed to the efficient use of the redundant links using dynamic routing and the use of priority queues. An optimized ASEN

network was constructed by subjecting the network to several variations through simulation. The mathematical model was developed for predicting the ASEN behavior for practical combinations of the input factors. This chapter conclusively proves the need for dynamic routing capability in a network with redundancy. It has also shown that the improvement in the ASEN performance can be achieved even in a fault free environment.

CHAPTER 5

CONCLUSIONS

This chapter presents a brief review of the ASEN network investigation. Section 5.1 discusses network validation issues. The conclusions are then addressed in Section 5.2. Improvements and suggestions for future research follow in the last section.

5.1 Network Validation

The validity of the ASEN model stems directly from previously published results. The network validation requires the comparison of the results obtained about the queue lengths and average delay times with that of previously published results. With regard to queue lengths, it was shown that a queue length of five was indeed adequate to provide a satisfactory improvement in performance. These figures are in close approximation to the buffer requirements of the MSC [Rai87]. As shown in Table 1 and Table 3, the size of buffer for the ASEN at varying loading factors has a very good correlation with that of the MSC classes of networks.

The investment in extra hardware in the form of intra-stage links plays a vital role in relieving the congestion and the contention for output ports in the network. The algorithm adopted for routing packets provides a lower average message delay. Moreover, the worst case delays for the packets to traverse the network were also found to be significantly lower than the multistage cube network. Hence, the investment in the extra hardware can be justified on the basis of an improvement in performance.

The correctness of the ASEN simulation model has been confirmed by numerous test runs made on ASEN network sizes of 8-by-8 and 16-by-16. Packets were routed through the network stage-by-stage and were verified to correctly reach their destinations. This was accomplished by having observation points in the simulation routines to provide trace functions to track the flow of packets in the network. Moreover, the ASEN simulation model is a super-structure of the MSC network, whose validity has been verified earlier [Rai87]. In addition, the minimum delay suffered by packets in both MSC and the ASEN networks until the saturation point are the same. Beyond the point of saturation, the minimum delay for the MSC networks increased because of the overwhelming number of packets being pushed into the network at higher loading factors. The delay variance was found to be the same for both networks. The network performance was to a large extent similar at early loading factors.

5.2 Conclusions

This investigation has presented a comparative analysis of performance characteristics of the augmented shuffle exchange network and the multistage cube network. Results of the research provide an extension to previous research efforts by analyzing the ASEN in a packet switched environment. These findings indicate that the ASEN provides a lower packet delay than the MSC while at the same time, providing non-saturation operation at higher network loading factors. The performance benefits of the ASEN are attributed to the use of augmented links and priority queues. It was shown that the ASEN requires fewer buffers than the MSC, and consequently has a lower implementation cost. To aid the network designer, a concise mathematical regression model was developed. This model was shown to accurately predict the delay characteristics of the ASEN. This research shows that the ASEN provides a low cost, high performance, packet switched network for parallel processing.

5.3 Improvements

The investigation on the ASEN complemented the earlier work done by Kumar [Kum87]. The packet switching aspect of the network was analyzed here. The effect of uniform and non-uniform loading on the network was simulated. The functional aspect of redundant links was examined through simulation using dynamic routing. Methods of avoiding congestion, contention and faults were also probed. Suggestions for improving the network model could include investigating the effects of

- Using redundant links as a bidirectional means for communicating messages instead of the unidirectional approach used for this investigation. The use of a link to transmit packets in each direction could also be analyzed. This could minimize the worst case delay times at higher loading values.
- The impact of multiple-packet messages on the network throughput could also be included in the study.
- Research could also focus on extending this series of investigation to encompass other fault tolerant interconnection network models to form a basis for the comparison of networks on a common ground.

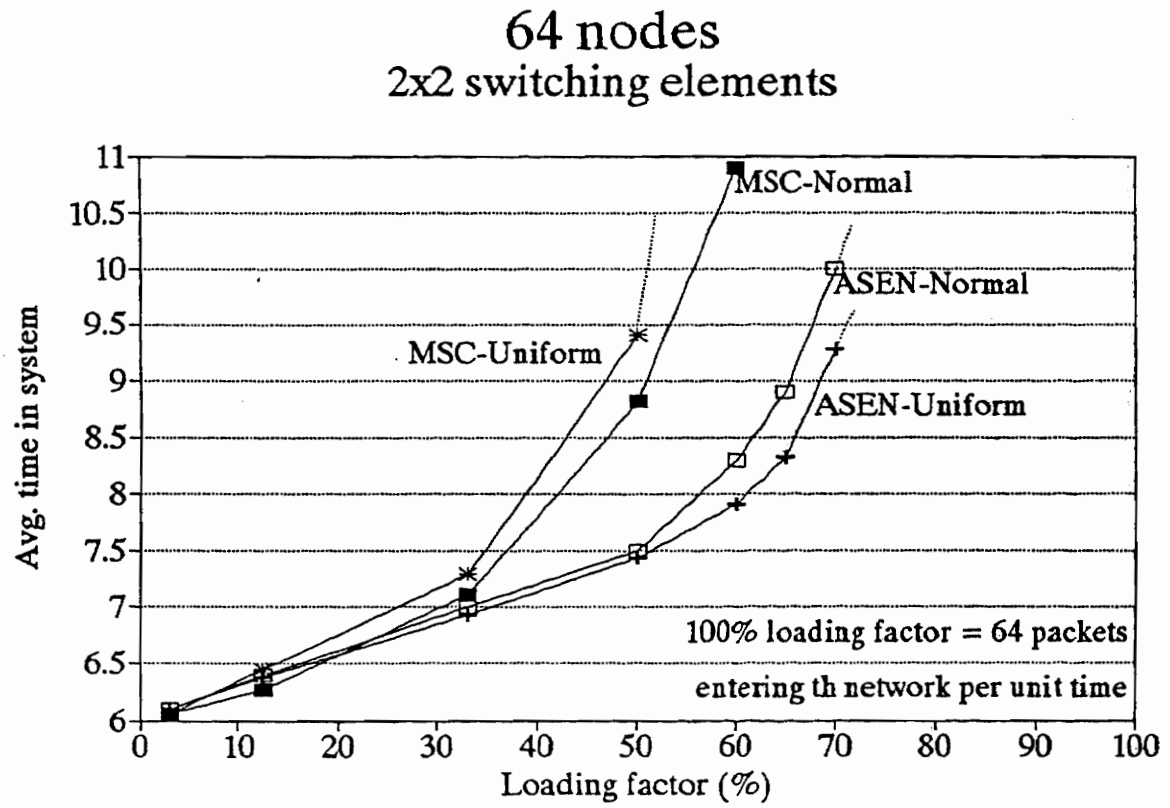

Figure A1: Performance plots of a 64 node ASEN and MSC (2-by-2 switch size)

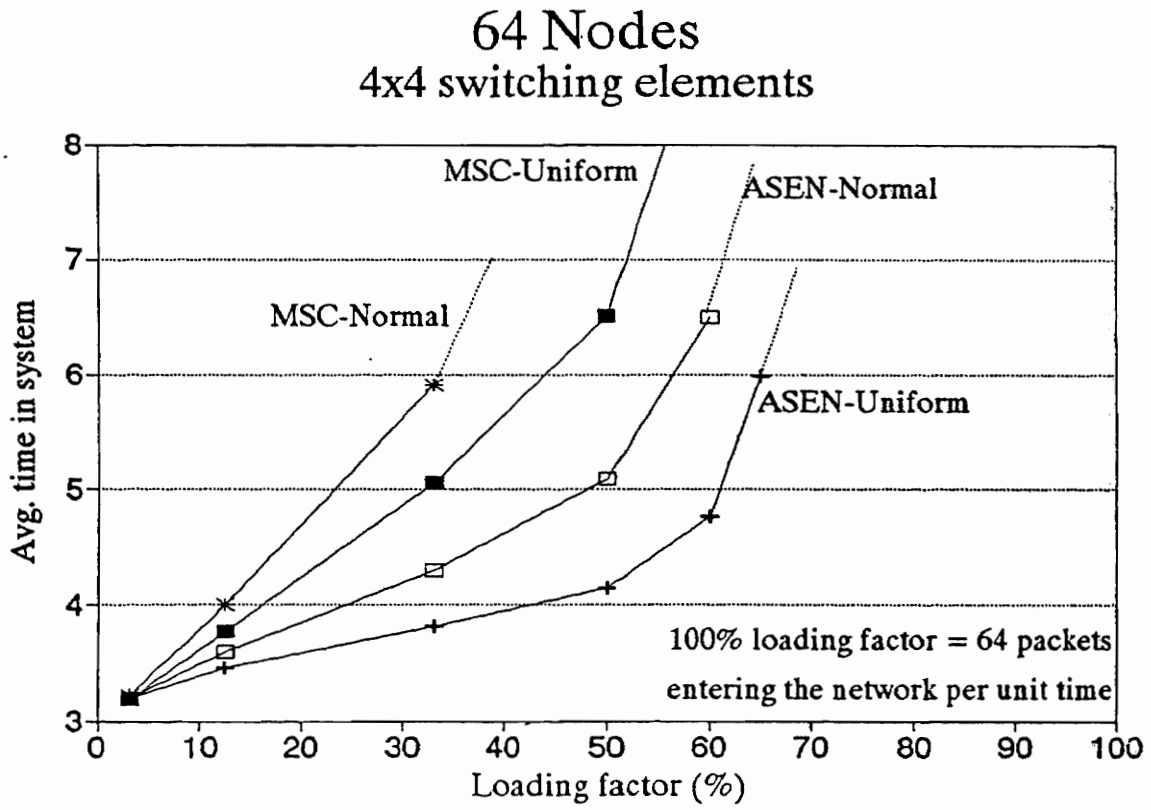

Figure A2: Performance plots of a 64 node ASEN and MSC (4-by-4 switch size)

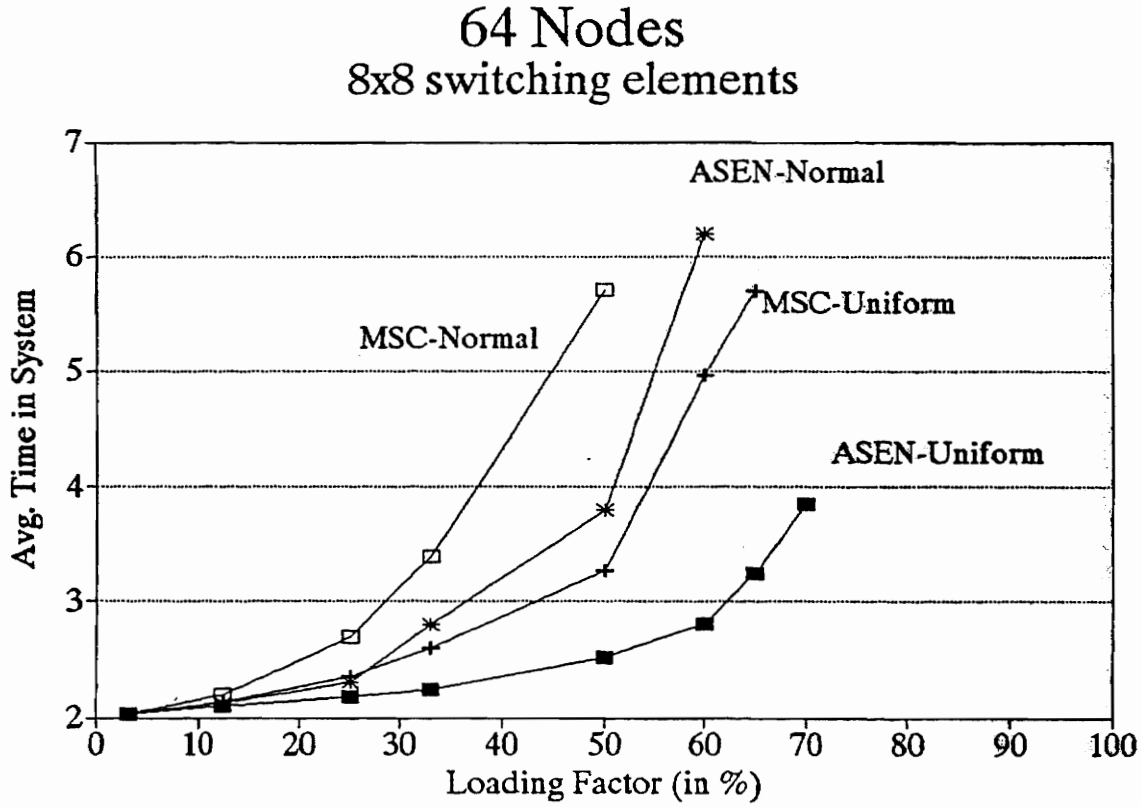

Figure A3: Performance plots of a 64 node ASEN and MSC (8-by-8 switch size)

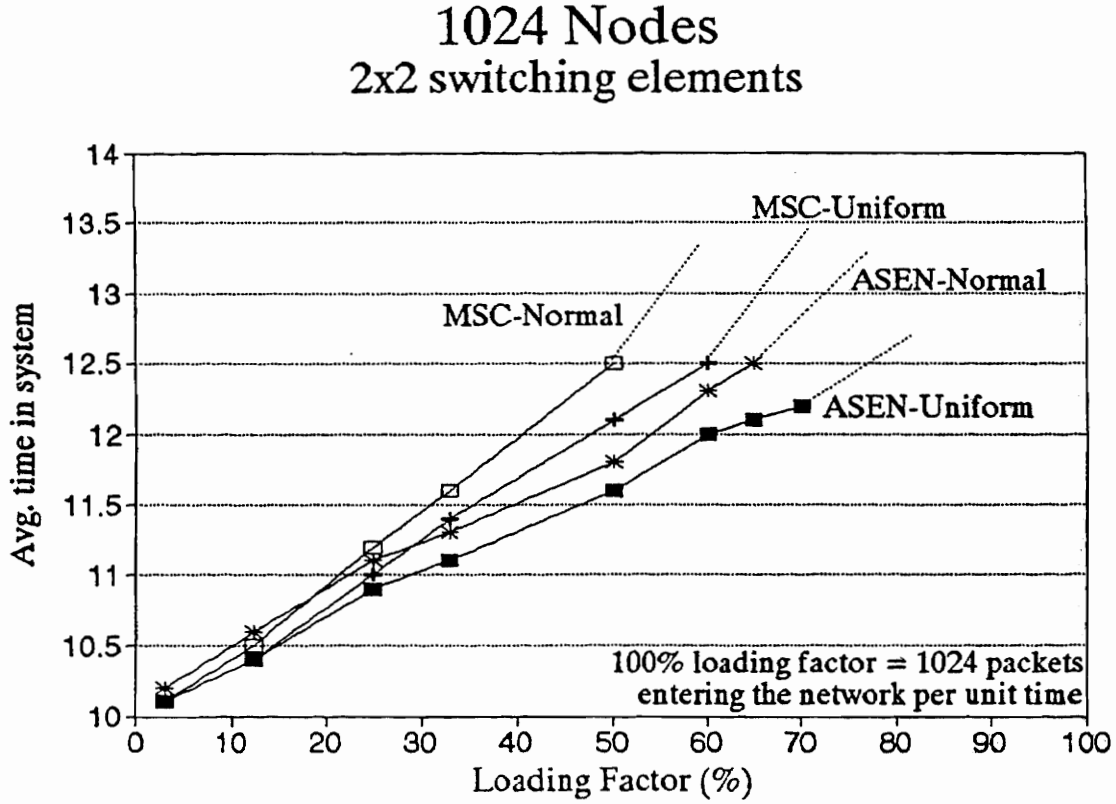

Figure A4: Performance plots of a 1024 node ASEN and MSC (2-by-2 switch size)

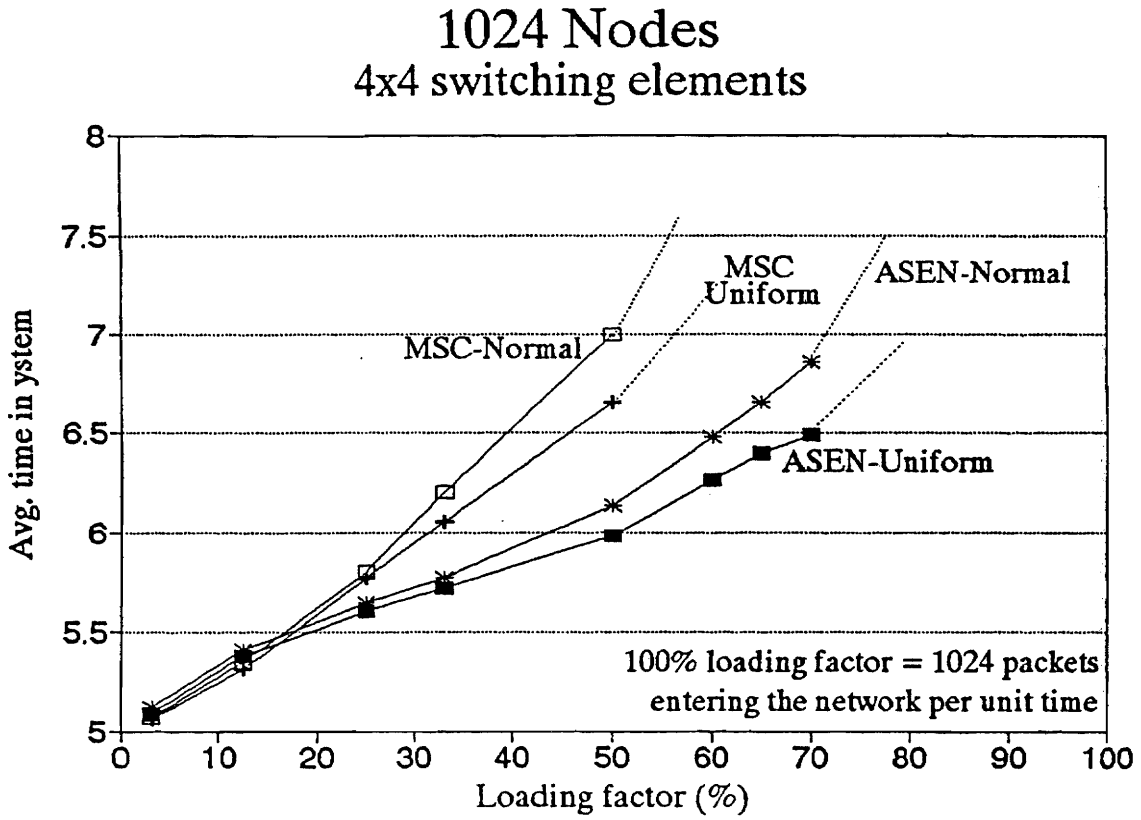

Figure A5: Performance plots of a 1024 node ASEN and MSC (4-by-4 switch size)

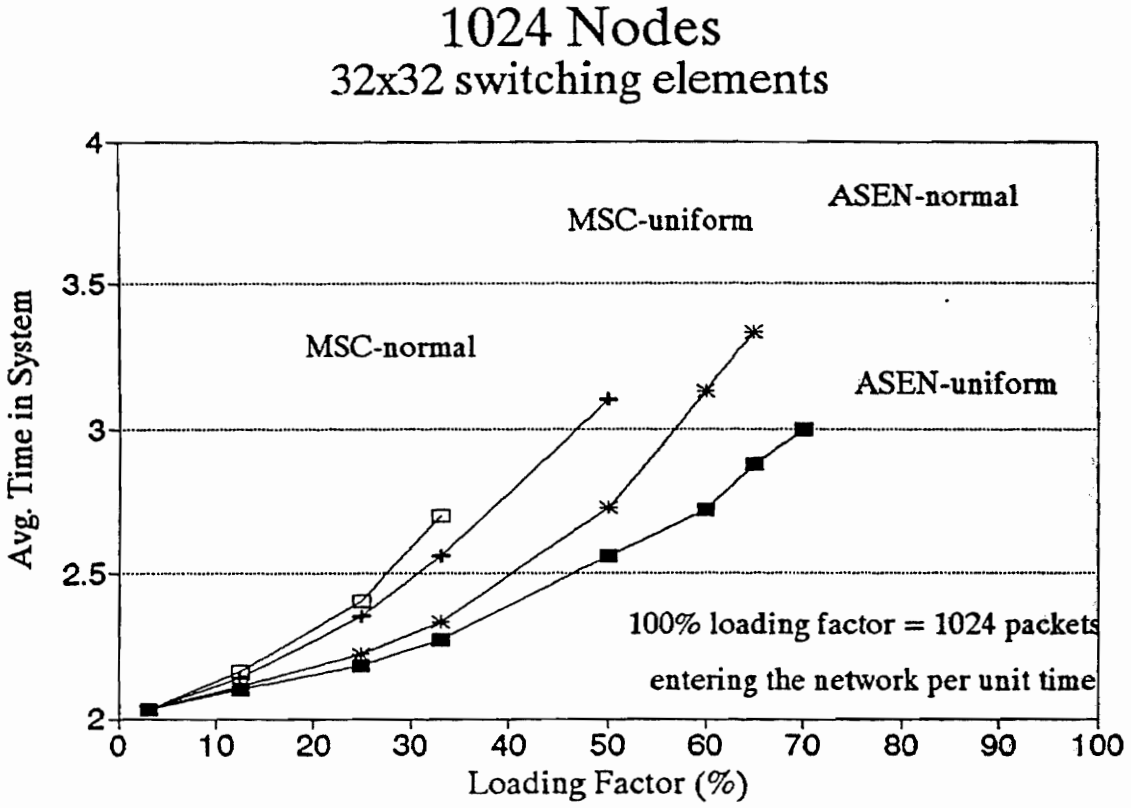

Figure A6: Performance plots of a 1024 node ASEN and MSC (32-by-32 switch size)

REFERENCES

[AbP86] S. Abraham and K. Padmanabhan, "Performance of the direct binary n-cube network for multiprocessors," *1986 International Conference on Parallel Processing,* August 1986, pp. 636-639.

[AdA87] G.B Adams III, D.P. Agrawal, and H.J. Siegel, "A survey and comparison of fault-tolerant multistage interconnection networks," *Computer,* June 1987, pp. 14-27.

[AdS82] G. B. Adams III and H. J. Siegel, "The Extra Stage Cube: A fault tolerant interconnection network for supersystems," *IEEE Transactions on Computers,* Vol. C-31, May 1982, pp. 443-454.

[AdS84] G.B. Adams III and H.J. Siegel, "Modifications to improve the fault tolerance of the extra stage cube interconnection network" *1984 Int'l Conf. Parallel Processing,* 1984, pp. 169-173.

[Agr85] S. C. Agrawal, Metamodeling, The MIT Press, Cambridge, MA, 1985.

[Bat76] K. E. Batcher, "The FLIP network in STARAN," *1976 IEEE Internatinal Conference on Parallel Processing,* August 1976, pp. 65-71.

[Ben62] V. Benes, "On rearrangeable three-stage connecting networks," The Bell System Tech. Journal, Sept. 1962, pp. 1481-1492.

[Ben65] V. Benes, "Mathematical theory of connecting networks," Academic Press, N.Y., 1965.

[BhY89] L. N. Bhuyan, Q. Yang, and D. P. Agrawal, "Performance of multiprocessor interconnection network," *IEEE Computer,* Vol. 22, No. 2, February 1989, pp. 25-36.

[Clo53] C. Clos, "A study of nonblocking switching networks," Bell System Tech. Journal, vol. 32, 1953, pp. 406-424.

[DeC90] A. L. DeCegama, The technology of parallel processing, Volume 1, Prentice Hall, Englewood Cliffs, NJ, 1990.

[DiJ81] D. M. Dias and J. R. Jump, "Analysis and simulation of buffered delta networks," *IEEE Transactions on Computers,* Vol. C-30, April 1981, pp. 273-282.

[DiJ82] D. M. Dias and J. R. Jump, "Augmented and pruned NlogN multistage networks: topology and performance," *1982 International Conference on Parallel Processing,* 1982, pp. 10-11.

[Fen81] T.Y. Feng, "A survey of interconnection networks," *IEEE Transactions on Computers,* December 1981, pp. 12-27.

[GoL71] L. R. Goke and G. J. Lipovski, "Banyan networks for partitioning multiprocessor systems," *Proc. 1st Annual Sysp. Computer Architecture,* Dec 1971, pp. 21-28.

[HwB84] K. Hwang and F.A. Briggs, Computer architecture and parallel processing, McGraw-Hill, N.Y., 1984.

[JeS86] M. Jeng and H. J. Siegel, "A fault-tolerant multistage interconnection network for multiprocessor systems using dynamic redundancy," *6th International Conference on Distributed Computing Systems,* 1986, pp. 70-77.

[KrS83] C. P. Kruskal and M. Snir, "The performance of multistage interconnection networks for multiprocessors," *IEEE Transactions on Computers,* Vol. C-32, December 1983, pp. 1091-1098.

[Kum85] V. P. Kumar, "On highly reliable, high performance multistage interconnection networks," Ph.D. Thesis, University of Iowa, Dec. 1985.

[KuP86] M. Kumar and G. F. Pfister, "The onset of hot spot contention," *1986 IEEE International Conference on Parallel Processing,* pp. 12-34.

[KuR87] V. P. Kumar and S. M. Reddy, "A fault-tolerant technique for shuffle-exchange multistage networks," *Computer,* June 1987.

[KuR89] V. P. Kumar and A. L. Reibman, "Failure dependent performance analysis of a fault-tolerant multistage interconnection network," *IEEE Transaction on Computers,* Vol. 38, No. 12 December 1989, pp. 1703-1713.

[Law73] D. H. Lawrie, "Memory-Processor connection networks," UIUCDCS-TR-73-557, University of Illinois, Urbana, Feb. 1973.

[Law75] D.H. Lawrie, "Access and alignment of data in an array processor," *IEEE Transaction on Computers,* Vol. C-24, December 1975, pp. 1145-1155.

[LaM86] R. J. Larsen and M. L. Marx, An Introduction to mathematical statistics and its applications, Second Edition, Prentice-Hall, Englewood Cliffs, NJ, 1986.

[LeK86] G. Lee, C. P. Kruskal, and D. J. Kuck, "The effectiveness of combining in shared memory parallel computers in the presence of hot spots," *1986 IEEE International Conference on Parallel Processing,* pp. 35-41.

[McH90] J. T. McHenry, "Performance evaluation of multicomputer networks for real-time cmputing," M.S. Thesis, Virginia Polytechnic Institute and State University, April 1990.

[McS82] R.J. McMillen and H.J. Siegel, "Performance and fault tolerance improvements in the inverse augmented data manipulator network," *9th Symposium on Computer Architecture*, April 1982, pp. 63-72.

[MiC86] D. Mitra and R. Cieslak, "Randomized parallel communications," *1986 IEEE International Conference on Parallel Processing,* pp. 224-230.

[PaL83] K. Padmanabhan and D. H. Lawrie, "A class of redundant path multistage interconnection networks," *IEEE Transactions on Computers,* Vol. C-32, December 1983, pp. 1099-1108.

[Pat79] J. H. Patel, "Processor-Memory interconnections for multiprocessors," *Proc. Sixth Annual Symp. Computer Architecture,* April 1979, pp. 159-163.

[Pat81] J.H. Patel, "Performance of processor-memory interconnections for multiprocessors," *IEEE Transactions on Computers,* Vol. C-30, October 1981, pp. 771-780.

[Pea77] M.C. Pease III, "The indirect binary n-cube microprocessor array," *IEEE Transactions on Computers,* Vol. C-26, May 1977, pp.458-473.

[PfN85] G. F. Pfister and V. A. Norton, "Hot spot contention and combining in multistage interconnection networks," *1985 IEEE International Conference on Parallel Processing,* pp. 790-797.

[Pri86] A. A. B. Pritsker, Introduction to Simulation and SLAM II, Systems Publishing Corporation, West Lafayette, IN, 1986.

[Rai87] R. A. Raines, The modeling, simulation and comparison of interconnection networks for parallel processing, Master's Thesis, School of Engineering, Air Force Institute of Technology, 1987.

[RaD88] R. A. Raines, N. J. Davis IV, and W. H. Shaw, "The modeling, simulation, and comparison of interconnection networks for parallel processing," *1988 Summer Simulation Conference,* July 1988, pp. 87-92.

[RaR92] V. Ramachandran, R. Raines, J. Park, N.J. Davis IV, "Performance studies of packet switched augmented shuffle exchange networks," *The 4th Symposium on the Frontiers of Massively Parallel Computation,* October 1992.

[RaV84] C.S. Raghavendra and A. Varma, "INDRA: a class of interconnection networks with redundant paths," *1984 Real-time Systems Symposium,* 1984, pp. 153-164.

[ReK84] S.M. Reddy and V.P. Kumar, "On fault-tolerant multistage interconnection networks," *1984 Int'l Conf. Parallel Processing,* 1984, pp.155-164.

[ShD92] W. H. Shaw, N. J. Davis IV, and R. A. Raines, "The application of metamodeling to interconnection network analysis," *ORSA Journal on Computing,* (submitted for publication).

[SiM81] H.J. Siegel and R.J. McMillen, "The multistage cube: A versatile interconnection network," *Computer,* Dec 1981, pp. 65-76.

[Sie90] H. J. Siegel, Interconnection networks for large-scale parallel processing: Theory and case studies, Second Edition, McGraw Book Co., NY, 1990.

[SiN89] H. J. Siegel, W. G. Nation, C. P. Kruskal, and L. M. Napolitano Jr., "Using the multistage cube network topology in parallel supercomputers," *Proceedings of the IEEE,* Vol. 77, No. 2, December 1989, pp. 1932-1953.

[Sto71] H.S. Stone, "Parallel processing with a perfect shuffle," *IEEE Transactions on Computers,* Vol. C-20, February 1971, pp. 153-161.

[WuF80] C. Wu and T. Feng, "On a class of multistage interconnection networks," *IEEE Transactions on Computers,* Aug 1980, pp. 696-702.

[WuF84] C. L. Wu and T. Y. Feng, Tutorial: interconnection networks for parallel and distributed processing, IEEE Computer Society Press, Silver Spring, MD, 1984.

VITA

V. Ramachandran was born on April 9, 1967. He finished his Undergradute degree in India and then his Masters degree in Electrical Engineering at Virginia Tech in 1992. He is now working for Vertex Semiconductor as a Network Systems Engineer in California. He can be reached at

1255 Vicente Dr, #112
Sunnyvale, CA 94086
Ph: (415) 969-8310