# Impact of Switch Architectures on the Performance of Multistage Interconnection Networks

**Bin Zhou** and **M. Atiquzzaman**

Dept. of Computer Science and Engineering
La Trobe University
Melbourne 3083, Australia
atiq@LATCS1.lat.oz.au

## Abstract

*Switching elements in interconnection networks for highly parallel shared memory computer systems may be implemented with different internal buffer structures. A Multistage Interconnection Networks (MIN) consists of several stages of small crossbar switching elements (SEs). The aim of this paper is to study the performance of a multibuffered MIN with different SEs architecture, in the presence of uniform and nonuniform traffic. For the purpose of comparison, the throughput and the network delay have been used as the performance measures.*

## 1 Introduction

A multiprocessor system consists of a number of processors and memories connected together by an interconnection network. Overall system performance for a highly parallel shared memory computer system depends on the message throughput that can be achieved by the interconnection network. Early work in the performance analysis of unbuffered MINs was done by Patel [1]. Performance of unbuffered MINs in the presence of nonuniform traffic have been reported in [2, 3]. Performance of MINs using buffered SEs under uniform and non uniform traffic are presented in [4, 5, 6, 7, 8, 3].

Three different $2 \times 2$ crossbar switching elements (SE) are analyzed in [9] for the single and unbounded queue sizes. The aim of this paper is to study the performance of MINs having finite buffered SEs, in the presence of uniform and hot spot traffic patterns. The results of the research work will enable the designer to consider the options for hardware implementation of $2 \times 2$ buffered SEs in a MIN, to characterize the performance of low cost hardware implementations, to prove the throughput limitations for different SE architectures, and to quantify the performance differences between the different types of SEs. In this study the throughput and delay are considered as the performance measures.

In Section 2, the structure of the different SE architectures and their operation are described, followed by the network operating assumptions of a buffered Omega network. A traffic model for the performance evaluation of the Omega network is proposed in Section 3. The simulation methodology is presented in Section 4. In Section 5, we present the comparison of performance of buffered Omega networks with different SE architectures, in the presence of both uniform and hot spot traffic pattern, followed by concluding remarks in Section 6.

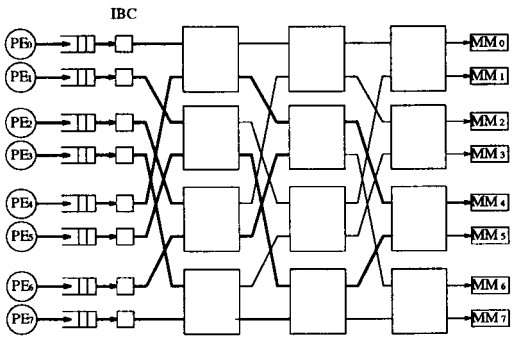


Figure 1: An $8 \times 8$ Omega network under hot spot traffic pattern

## 2 Omega Network and Assumptions

Figure 1 shows an Omega network using several stages of SEs. This section describes the different SE architectures and network operating assumptions under which the research has been carried out.

### 2.1 Switch Element Architectures

Three possible arrangements of the buffers inside an SE are illustrated in Figure 2 . As discussed in [9], $2 \times 2$ buffered SEs used in a MIN can be classified into three different types. The first design, Type $A$, contains two output buffers, each buffer capable of accepting two packets simultaneously. This design has been analyzed in [10, 11, 4], but has the disadvantage of being more difficult to implement than the designs of Type $B$ and $C$ given below. The capability of inserting two packets into a queue during a cycle adds considerable complexity and may increase the cycle time of the component, which may in turn, increase the network cycle time.

Type $B$ and $C$ SEs have single-input and output buffers. They are simpler to design and need fewer hardware resources per buffer than the Type $A$ design. Each switch of type $B$ has four of these simpler buffers, one for each input/output pair, and is the one that has been used to implement the NYU Ultracomputer switching node [9]. Each switch of type $C$ has a single input buffer at each of its input links.

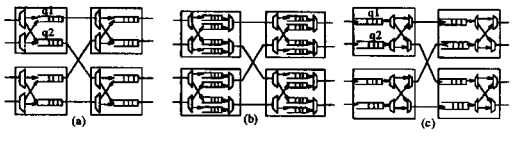


Figure 2: Three types of $2 \times 2$ buffered SE: (a)Type $A$; (b) Type $B$; (c) Type $C$

In the Type $A$ switch, $q_1$ and $q_2$ refer to two buffers paired at an output port. A Type $C$ switch with two single-input buffers, is the simplest hardware implementation.

### 2.2 Network Assumptions

We make the following assumptions regarding the operation and the environment of the interconnection network in which the above mentioned switches are used [10, 3] .

1. There are $N = 2^n$ processors and $N$ memory modules in the system, where $n$ is an integer. The processors and memories will be represented by $PE_i$, $0 \leq i \leq N-1$, and $MM_j$, $0 \leq j \leq N-1$ respectively.
2. The network operates synchronously.
3. A backpressure mechanism ensures that no packets are lost within the network.
4. The arrival process at each input of the network is a simple Bernoulli process. Each input link of the network is offered the same traffic load.
5. There is no blocking at the output links of the network.
6. The conflict resolution logic at each SE is fair, i.e., routing conflicts among packets at the inputs of an SE are randomly resolved.
7. In addition to the buffers in the SEs, the network has input buffer controllers (IBCs) at every input of the network.
8. The minimum possible delay of a packet is equal to $n+1$, where $n$ is the number of stages. It includes the delay at the IBC buffer.

## 3 Traffic Modeling

In this study, we have considered uniform and hot spot traffic patterns.

### 3.1 The Traffic Matrix of a Switch

A traffic matrix denotes the probability with which a packet arriving at an input port of a MIN is destined to the different outputs of the network. Let $P = \{p_{ij}, 0 \leq i,j \leq N-1\}$ denote the traffic matrix of an $N \times N$ MIN. The sum of the elements of row $i$ of the traffic matrix, given by $\rho_i = \sum_{j=0}^{N-1} p_{ij}$, is the traffic load to input port $i$. The sum of column $j$ of the traffic matrix, given by $r_j = \sum_{i=0}^{N-1} p_{ij}$, is the arrival rate of packets, at any network input, which are destined for output port $j$. Letting $q_{ij}$ be the probability of a packet arriving at input port $i$ being destined for output port $j$, we have $p_{ij} = \rho_i q_{ij}$. We can represent the traffic matrix $P$ by the product of the traffic load matrix $P_L$ and the traffic destination matrix $P_D$. The traffic matrix is therefore expressed as $P = P_L P_D =$

$$\begin{bmatrix} \rho_0 & 0 & \dots & 0 \\ 0 & \rho_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \rho_{N-1} \end{bmatrix} \begin{bmatrix} q_{00} & \dots & q_{0(N-1)} \\ q_{10} & \dots & q_{1(N-1)} \\ \vdots & \vdots & \vdots \\ q_{(N-1)0} & \dots & \dots \end{bmatrix}$$

### 3.2 Uniform Traffic Pattern

Uniform traffic satisfies the following conditions: (i) $\rho_i = \rho, 0 \leq i \leq N-1$ and (ii) $q_{ij} = 1/N, 0 \leq i,j \leq N-1$. Under condition (i), the traffic load matrix is uniform in that each diagonal element equals $\rho$, i.e., $P_L = \text{diag}[\rho, \rho, \dots, \rho]$. Under condition (ii), the traffic distribution matrix becomes uniform in that every element is equal to $1/N$. The elements of the traffic matrix $P$ are all equal, i.e., $p_{ij} = \rho/N, 0 \leq i,j \leq N-1$. Thus, under uniform traffic, the average packet arrival rate to any output is equal to $r_j = \rho, 0 \leq j \leq N-1$.

### 3.3 Hot-spot Traffic Pattern

The nonuniform traffic pattern to be considered in this paper is the hot spot traffic pattern. In the hot spot traffic pattern, there is an output destination port which is accessed more than other output ports. Such a hot-spot traffic can be characterized by a single hot-spot of a higher access rate, superimposed on a background of uniform traffic [6]. Let $h$ be the fraction of packets directed to the hot-spot output $j_H$. Then we have, for all $i$

$$q_{ij} = \begin{cases} h + \frac{1-h}{N} & \text{if } j = j_H \\ \frac{1-h}{N} & \text{if } j \neq j_H \end{cases} \quad (1)$$

Figure 1 shows an $8 \times 8$ Omega network under a hot spot traffic pattern. The switching elements and links that carry hot traffic are shown by thick lines. The traffic matrix for the hot spot pattern is given below. $MM_4$ will be assumed to be a hot memory module for all processors. Processor $PE_i$ generates packets to memory $MM_j$ with rate $q_{ij}$ at each cycle. Assume that the traffic load on each input is equal to $\rho$, so $p_{ij} = \rho q_{ij}$.

$$P = \begin{bmatrix} \frac{(1-h)\rho}{N} & \dots & \rho h + \frac{(1-h)\rho}{N} & \dots & \frac{(1-h)\rho}{N} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \frac{(1-h)\rho}{N} & \dots & \rho h + \frac{(1-h)\rho}{N} & \dots & \frac{(1-h)\rho}{N} \end{bmatrix}$$

The average packet arrival rate for the $j$-th output is

$$r_j = \begin{cases} [1 + (N-1)h]\rho & \text{if } j = j_H \\ (1-h)\rho & \text{if } j \neq j_H \end{cases} \quad (2)$$

As long as $h > 0$, we have $r_j \geq \rho$ for $j = j_H$,and $r_j \leq \rho$ for $j \neq j_H$. Only one packet can be transmitted to an output link in one time slot, so in order to guarantee a stable output queueing, the traffic load $\rho$ must be limited such that $r_j \leq 1$ for $j = j_H$. Therefore, from Eqn.(2),

$$\rho \leq \frac{1}{1 + (N-1)h} \quad (3)$$

## 4 The Simulation Method

We carried out simulation of three types of buffered Omega networks with arbitrary buffer size and uniform and hotspot traffic patterns. The assumptions mentioned in Section 2.2 were implemented in the simulator as follows.

1. At each stage cycle, a random request generator generates a packet with probability $\rho$ (input load) at an input port.

2. The destination of a generated packet is taken from a uniform random number generator in the case of a uniform traffic, and in the case of hot spot traffic, from a nonuniform random number generator which generates requests according to the distribution mentioned in Section 3.

3. If there is a routing conflict among packets within an SE, a packet is selected randomly by another random number generator.

4. First-in-first-out (FIFO) queueing policy was used at the buffers in the SEs.

5. The throughput and the delay were measured at each output port of the network and averaged over the network size and simulation time span (typically 50,000 cycles) to get the normalized throughput and the normalized delay of the network. The outputs for the first 500 cycles were discarded to allow the network to reach a steady state.

The simulator, written in $C$, has the following components: main routine to control the flow of the program; switex( ) to implement the switching operation; generate( ) to generate random requests; conflict( ) to resolve the conflicts randomly; shuf( ) to give the shuffled form of a link; unshuf( ) to give the unshuffled form of a link; shuffle( ) to actual shuffle operation on a set of requests; rotate( ) to give the destination bit to be tested at a stage; count( ) to count the number of requests to a memory; shift( ) to shift the packets forward in the buffer, and the report generator.

We use a three-dimensional array to represent the buffers at the SEs of the network. The first dimension is the stage number, the second is the number of the SE in the stage and the third is the buffer in the SE. A two-dimensional array contains the address of the current empty location in the queue. The following input data values were varied each run to have a comprehensive picture of the network behavior:

1. Length of the simulation: The number of cycles for which the simulations were performed were large, typically 50,000.

2. Seed for the random number generator: The simulator required two independent streams of numbers, one for the generation of the requests and the other for the resolution of the conflicts.

3. System size: Networks of sizes up to 128 were simulated.

4. $\rho$ and $h$ were varied.

### 4.1 Parameters evaluated

The parameters evaluated for the comparison of performance include the normalized throughput and the average time delay. When the network reached a steady state after $t_1$ clock cycles , the number of valid packets at the outputs of the network were counted at the end of each cycle after $t_1$. These were averaged over a large number of cycles upto $t_2$ to give the average memory bandwidth. The normalized throughput ($\lambda$) is therefore given by

$$\lambda = \frac{1}{N(t_2 - t_1)} \sum_{l=0}^{N-1} \sum_{t=t_1}^{t_2} \lambda_{l,t} \qquad (4)$$

where, $\lambda_{l,t}$ is the throughput at the $l$-th output link during cycle $t$.

The mean network delay is obtained by averaging the delay experienced by the packets over a large number of cycles. It is given by

$$\tau = \frac{1}{s} \sum_{l=0}^{N-1} \sum_{t=t_1}^{t_2} \tau_{l,t} \qquad (5)$$

where, $\tau_{l,t}$ is the delay experienced by a packet (if there is one) at the $l$-th output link during cycle $t$, and $s$ is the total number of packets that have arrived at the output links during the interval $(t_2 - t_1)$.

## 5 Results and Discussions

We have simulated $8 \times 8, 16 \times 16, 32 \times 32, 64 \times 64$, and $128 \times 128$ multibuffered Omega network using type $A, B$ and $C$ SEs under uniform and hot spot traffic patterns.

Figure 3 shows the normalized throughput versus input load ($\rho$) for three different $64 \times 64$ Omega networks using the three types of SEs. Figure 4 and 5 plot the normalized throughput versus the hot spot probability for the three types of buffered Omega networks with different network sizes. $h$ was varied from 0 to 0.25. The normalized throughput is calculated by offering the network a load of one.

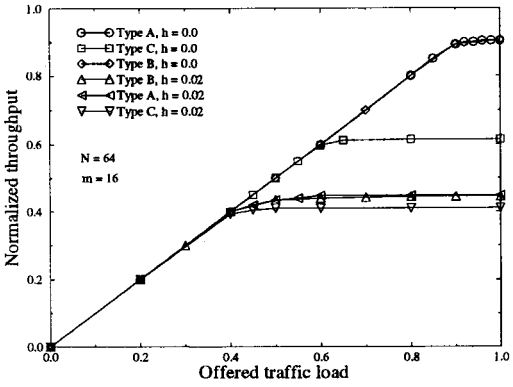


Figure 3: Normalized throughput versus input load

Figure 6 shows the normalized throughput versus buffer size for a $64 \times 64$ Omega network with three types of SEs under hot spot traffic pattern. Figure 7 shows the delay versus input load ($\rho$) for an $8 \times 8$ Omega network under different hot spot probabilities. It shows tree saturation occurring at $\rho = 0.8$ for type $A$ and $B$ buffered network, when $h$ equals 0.02, and at $\rho = 0.6$ for type $C$ buffered network. When $h$ equals 0.14, the tree saturation occurs approximate at 0.45 for networks with type $A$ and $B$ SEs. Figure 8 shows the mean transfer time versus

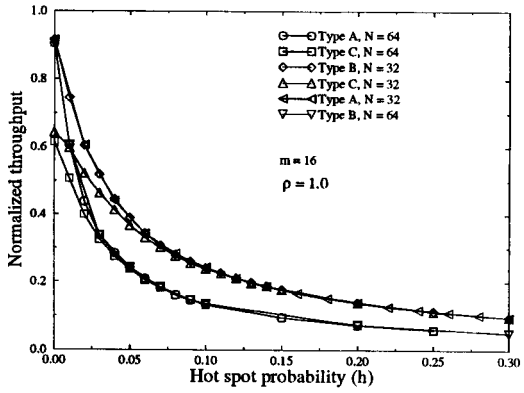


Figure 4: Normalized throughput versus hot spot

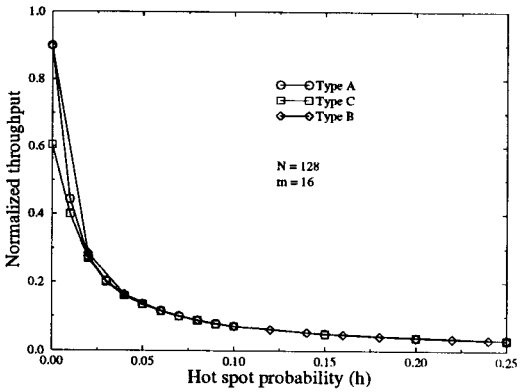


Figure 5: Normalized throughput versus hot spot

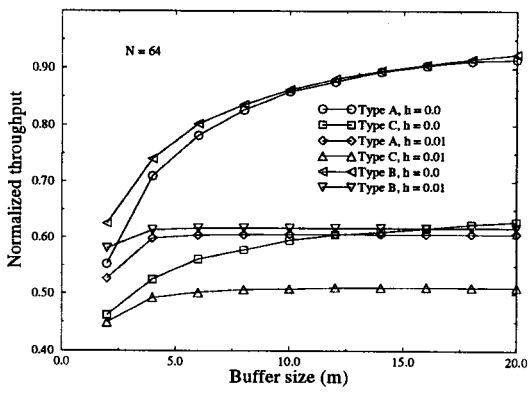


Figure 6: Normalized throughput versus buffer size

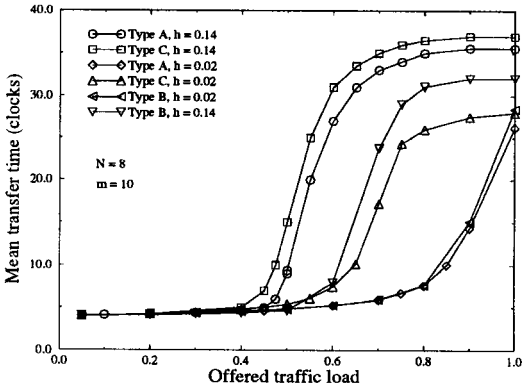


Figure 7: Delay versus input load ($\rho$)

normalized throughput for networks using type $A$, $B$ and $C$ SEs.

When buffers are located at the input links (type $C$), the maximum normalized throughput of a large network is limited to 0.62 under uniform input traffic pattern (Figure 4). This bottleneck, due to the head of the line (HOL) contention at each SE, is intrinsic to input queueing. When a packet at the head of a buffer loses a contention, it impedes the rest of the packets in the same buffer from progressing forward, if packets are served on a FCFS basis. Another bottleneck arises when two or more packets contend for the same buffer in an SE. Since only one packet can be admitted to the buffer in one clock cycle, the other one is blocked and will have to retry in the next clock cycle. When buffers are placed at the output links of each switching element (type $A$ and type $B$), a maximum throughput of unity can be achieved. From Figure 4, we see that the maximum normalized throughput of 0.9 is achieved when $h = 0$ and $N = 32$. As the hot spot probability increases, the normalized throughput decreases. The larger the network size, the faster the throughput drops off (Figures 4, and 5). At low hot spot probabilities, the output-buffered networks (using types $A$ and $B$ SEs) have higher throughput than the input-buffered network (type $C$). That is due to the HOL contention in the case of input buffered SEs. And in high hot spot probability (say larger than 0.1 for a $32 \times 32$ network in Figure 4), the normalized throughput curve is the same for input and output buffered networks. That is because of tree saturation mentioned in Section 1. Comparison of results from simulation studies show that the performance of an output-buffered network is much better than an input-buffered network when the hot request ($h$) is low. But the performance is the same for both networks when the level of hot requests is medium and high. This is due to tree saturation. Certain general conclusions can be drawn:

- Type $A$ and type $B$ network with large buffer size are very close in normalized throughput performance. The parameters illustrated in Figure 3 shows type $A$ values are identical to type $B$. In Figure 6, the normalized throughput performance of type $B$ is better than type $A$ for small buffer size.

- For loads under 60%, the throughput performance of type $C$ switches is reasonably close to that of types $A$ and $B$, and because of its

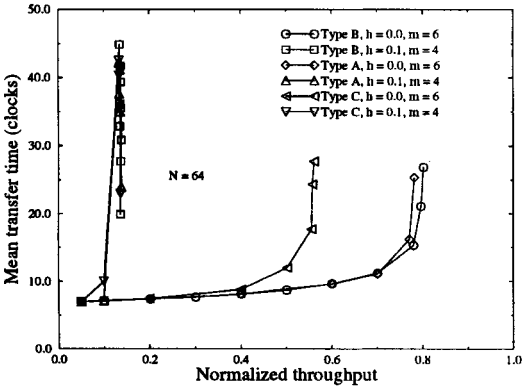


Figure 8: Mean transfer time versus normalized throughput

lower cost, type $C$ may be the implementation of choice. However, in addition to the throughput limitation, type $C$ switches show a significant increase in mean time delay even at loads as low as 60% (Figure 7).

- Adding bigger buffers to a type $C$ switch will not bring about a substantial performance improvement, since the output rate is limited at 75% due to the head of the line contention. Type $B$ and type $A$ switches with small size buffers have significantly better performance than type $C$ switches with large buffer size. Thus if additional hardware resources are to be applied to improving network performance, they are better spent on implementing type $B$ or type $A$ designs than on making larger buffers for a type $C$ design.

In Figure 7 and 8, the average time delay is compared. The offered load is varied from 0.01 pkt/cycle to 1.0 pkt/cycle for each processing element. It is shown the mean time delay of type $B$ is smaller than that of type $A$ and type $C$.

## 6 Conclusion

A simulation model has been developed to evaluate the performance of multibuffered Omega network with different switch element architectures under uniform and hot spot traffic environment. It is quantified the intuition that better performance results with type $B$ switch than with type $A$ and $C$ switches. Besides performance, of course, there are other issues, such as switch implementation, that must be considered in designing a switch network. We compared the performance of input-buffered MINs (type $C$) with output-buffered MINs (type $A$ and $B$) under hot spot traffic condition. The results show that the performance of output-buffered network is much better than input-buffered network when the hot request is low. But the performance is the same for both networks when the hot requests are medium and high. This is due to the onset of tree saturation at medium and high network traffic loads. Development of an analytical model for multibuffered multistage interconnection network with different switch architectures by considering the correlations between consecutive clock cycles as well as the states of the buffers in the adjacent stages is currently underway.

## References


[1] J. H. Patel, "Performance of processor-memory interconnection for multiprocessors," *IEEE Trans. Comput.*, vol. C-30, pp. 771–780, Oct. 1981.

[2] M. Atiquzzaman and M.S. Akhtar, "Effect of hot spots on the performance of multistage interconnection networks," *FRONTIERS 92: The Fourth Symposium on the Frontiers of Massively Parallel Computation*, Virginia, pp. 504–505, Oct. 1992.

[3] M. Atiquzzaman and M.S. Akhtar, "Effect of non-uniform traffic on the performance of multistage interconnection networks," *9th International Conference on Systems Engineering*, Las Vegas, Nevada, pp. 31–35, Jul. 1993.

[4] D.S. Meliksetian and C.Y.R. Chen, "A markov-modulated bernoulli process approximation for the analysis of banyan networks," *ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, pp. 183–194, May 1993.

[5] H. S. Kim and A. L. Garcia, "Performance of buffered banyan networks under nonuniform traffic patterns," *IEEE Trans. Commun.*, vol. 38, no. 5, pp. 648–658, May 1990.

[6] G.F. Pfister and V.A. Norton, "Hot spot contention and combining in multistage interconnection networks," *IEEE Trans. Comput.*, vol. C-34, no. 10, pp. 943–948, Oct. 1985.

[7] S.L. Scott and G.S. Sohi, "The use of feedback in multiprocessors and its applications to tree saturation control," *IEEE Trans. on Parallel and Distributed Systems*, pp. 943–948, Oct. 1990.

[8] P.C. Yew, N.F. Tzeng, and D.H. Lawrie, "Distributing hot-spot addressing in large-scale multiprocessors," *IEEE Trans. Comput.*, vol. C-36, no. 4, pp. 269–277, Apr. 1987.

[9] O.E. Percus and S.R.Dickey, "Performance analysis of clock- regulated queues with output multiplexing in three different 2 × 2 crossbar switch architectues," *Journal of Parallel and Distributed Computing*, vol. 16, no. 1, no. 1, pp. 27–40, 1992.

[10] B. Zhou and M. Atiquzzaman, "Performance of output-multibuffered multistage interconnection networks under nonuniform traffic patterns," *International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'94)*, Durham, North Carolina, USA., pp. 405–406, Jan 31 - Feb 2, 1994.

[11] B. Zhou and M. Atiquzzaman, "Improved performance model of multibuffered multistage interconnection network under general traffic patterns," *IEEE INFOCOM '94: Conference on Computer Communications*, Toronto, Canada, June 12-16, 1994.