A Performance Comparison of Buffering Schemes for Multistage Switches

Bin Zhou
Dept. of Comp. Science and Comp. Engg.
LaTrobe University
Melbourne 3083, Australia
Email: binz@latcs1.lat.oz.au

M. Atiquzzaman
Dept. of Elec. & Computer Systems Engg.
Monash University, Clayton,
Melbourne 3168, Australia
Email: atiq@eng.monash.edu.au

Abstract

Multistage Interconnection Networks (MIN) are used to connect processors and memories in large scale scalable multiprocessor systems. MINs have also been proposed as switching fabrics in ATM networks in the future Broadband ISDN networks. A MIN consists of several stages of small crossbar switching elements (SE). Buffers are used in the SEs to increase the throughput of the MIN and prevent internal loss of packets. Different buffering schemes for the SEs are discussed in this paper. The objective of this paper is to study the performance of MINs with different buffering schemes, in the presence of uniform and hot spot traffic patterns. The results obtained from the study will help the network designers in choosing appropriate buffering strategies for MINs. For comparing different buffering strategies, the throughput and packet delay have been used as the performance measures.

1 Introduction

Multistage Interconnection Networks (MIN) have been found to be very suitable for interconnecting a large number of processors and memories in large scale multiprocessor systems. A MIN consists of a number of small crossbar switching elements (SE) interconnected by a permutation function. MINs can be broadly classified into two main categories, namely internally *blocking* and internally *nonblocking*. In an internally nonblocking MIN two or more packets at different input ports can be simultaneously forwarded to two different output ports. A MIN is called internally blocking if two or more packets with distinct output port destinations cannot always be transferred to the output ports due to routing conflict within the MIN. For instance, resource contentions occur in MINs when more than one packet access the same internal link. Buffers are used in the SEs to store the packets which lose the routing conflicts in an internally blocking MIN. The packets are queued in the buffers for transmission during subsequent clock cycles.

The proper placement and arrangement of buffers in the SEs have a dramatic impact on the performance of the MIN. The implementation of *input buffered* SEs, operating in the first-in first-out (FIFO) fashion, is very simple in the sense that the internal links of the MIN have to operate at the same speed as the external input/output lines of the MIN. Therefore, internal speedup of the MIN is not required, and the hardware complexity can be lower than other buffering schemes to be discussed later. However, when a packet at the head of a queue in an SE waits for transmission to its destined output link, successive packets (which may be destined to different output links) in the queue must also wait. This phenomenon, called the head-of-line (HOL) blocking, in input-buffered SEs reduces the throughput of the MIN.

In an *output-buffered* SE with separate buffers for each output link [1], a buffer must be able to receive up to d packets at a time, where d is the size of the SE. Output buffered SEs do not suffer from the above mentioned head-of-line blocking effect and hence have higher throughput than input buffered SEs. However, the main drawback of the output buffered SEs is that it needs to operate d times faster than the input/output lines of the MIN. This higher speed increases the implementation complexity and cost of the MIN. There are also SEs that combine input and output buffering techniques, and in this case, the operating speed of the SEs can be lower than in the case of the output buffered SEs [2].

The buffers can also be located at the *crosspoints* inside the SE [3]. This buffering scheme removes the blocking of packets by a packet destined to a different output of the SE. All packets arriving at the inputs of an SE can, in principle, be transferred to their target buffers within one clock cycle.

Finally, another possibility to obtain high performance is through the use of a shared buffer [4]. In a shared buffer SE of size $d \times d$, all input and output links of the SE have access to a shared buffer module which is able to write up to d incoming and read up to d outgoing packets in a clock cycle. There is no HOL blocking in shared buffer SEs and optimal throughput/delay performance is achieved. Furthermore, buffer utilization is better than input, output or crosspoint buffered SEs, thereby requiring a smaller number of buffers for

0-7803-2018-2/95/$4.00 © 1995 IEEE

the same performance. A shared-buffer MIN also has some additional features, e.g., its basic architecture can be easily modified to handle several service classes through priority control functions to meet different service requirements. Multicasting and broadcasting can also be easily implemented. The limitations of shared buffer MINs arise from technological limitations. A buffer in an SE needs to queue *d* incoming and dequeue *d* outgoing packets per clock cycle. Therefore, the bandwidth of a buffer must be at least the sum of the bandwidths of the incoming and the outgoing lines.

In addition to be being used in multiprocessor systems, MINs have also been proposed as the switching fabrics in the future Broadband Integrated Services Digital (B-ISDN) networks [5]. The CCITT has standardized the asynchronous transfer mode (ATM) as the multiplexing and switching principle for the B-ISDN network [6]. ATM will provide flexibility in bandwidth allocation and will allow a switch to carry heterogeneous services ranging from narrow-band to wide-band services requiring real time. However, the challenge is to build fast high performance switches which are able to match the high speeds of the input links.

Early work in the performance of input-buffered Banyan switches has been discussed in various publications. Jenq [7] has analyzed MINs consisting of 2 × 2 SEs with single-cell input buffers and operating in the presence of a uniform traffic in the MIN. Kim [8] reported a queuing analysis and simulation study of output-buffered Banyans with an arbitrary (finite) buffer size. All the above performance analyses were made on the assumption of the MIN operating in the presence of a uniform traffic pattern.

A first approach to the analysis of single buffered MINs in a nonuniform traffic environment is described in [9]. It is shown that certain nonuniform traffic patterns can have a crucial influence on the performance of the MIN.

Pfister [10] discussed the phenomenon of tree saturation arising as a result of a hot spot in a buffered MIN. Atiquzzaman [11] proposed an efficient Markov chain model for the performance evaluation of a single buffered Omega switch in the presence of a hot spot.

Three different buffering schemes for 2 × 2 SEs are analyzed in [12] for the unbounded queue size and queue size equal to one. The *aim* of this paper is to study the performance of MINs with four different buffering schemes in the presence of *uniform* and *hot* spot traffic patterns. The results of this research work will enable the network designer to consider the buffering options for hardware implementation of buffered SEs in a MIN, to characterize the performance of low cost hardware implementations, to obtain an insight into the throughput limitations for different SE architectures, and to quantify the performance differences between the different types of SEs. Designers may use the results to weigh a higher cost implementation with higher performance SE against a lower cost implementation with lower performance SE. In this study, the *normalized throughput* and *mean delay* have been used as the performance measures.

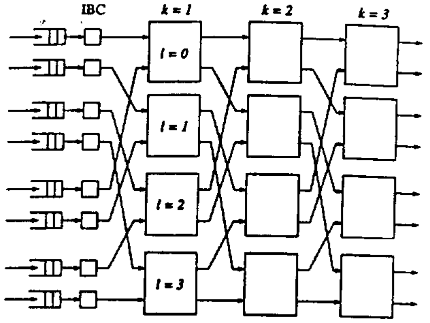

Figure 1: A three stage MIN.

The paper is organized as follows. In Section 2, the four different buffering schemes in SEs are described, followed by the operating assumptions of buffered internally blocking MINs. The simulation methodology is presented in Section 3. In Section 4, we present the performance results of the MINs using different buffering schemes in the SEs, in the presence of both uniform and hot spot traffic patterns, followed by concluding remarks in Section 5.

2 ATM Switches and Assumptions

A MIN connects N inputs to N outputs using $n = \log_2 N$ stages of $N/2$ SEs per stage. We use a perfect shuffle permutation to connect adjacent stages as shown in Figure 1 for $N = 8$. Each SE is a 2 × 2 crossbar allowing any input link to be connected to any output link. An SE has a finite number of buffers.

A request generated from a packet generator is bundled into a packet consisting of data and the destination address. The destination address is an n-bit number represented by $D = (d_1 d_2 \ldots d_{n-1} d_n)_2$. Destination tag routing is used to route a packet through the MIN. An SE at stage k inspects bit d_k, and in the case of no conflict routes the packet to the upper or lower output of the SE depending on d_k being 0 or 1 respectively. A unique path of constant length exists between any input-output pair of the MIN, thereby rendering the MIN a blocking type of switch. In addition to the buffers in the SEs, the MIN has input buffer controllers (IBCs) at every input of the MIN. To prevent packet loss in a MIN having finite-sized buffers at the SEs, IBCs with large buffer space are required in a MIN employing back-pressure as the flow control mechanism.

2.1 Different Switch Element Architectures

Four possible arrangements of the buffers in an SE are illustrated in Figure 2. As discussed in [12], any 2 × 2 buffered SE used in a MIN can be classified as one of these four types. *Input* buffering with buffers located at the inputs of the SEs, is the simplest to implement. When two or more packets contend for the same output port of an SE, only one of them is allowed to move to the next stage. The packets losing the contention wait at the head of the input queues and block other cells (waiting in the same queue) which may be destined to other idle output ports. This phenomenon is called head-of-line (HOL) blocking.

In the case of *output* buffering, buffers are placed at the output ports. In this type of buffering, only the unavoidable blocking effects introduced by the contention for the output links of the MIN are present [1]. Consequently, this is the optimum buffer placement. In a given clock cycle, the two inputs of an SE may access the same output buffer. Thus, multi-port buffers are required to support output buffering. This design has been analyzed in [8] but has the disadvantage of being more difficult to implement than the input or crosspoint buffering. The capability of inserting two cells into a queue of an SE during a clock cycle adds considerable complexity and may increase the clock cycle time.

In *crosspoint* buffering, there is a separate buffer for every input-output pair. Cells arriving at the inputs are enqueued in the appropriate buffer according to the destination tag. Buffers corresponding to a particular output of an SE are served in round robin or some other predetermined fashion. The multiplexor at an SE output selects one buffer if both buffers contain cells. The choice of a buffer is carried out according to a selection policy described in Section 2.2. Since separate buffers exist for each input-output pair, only one read and one write operation needs to be performed on a buffer in a single clock cycle. A disadvantage from the performance point of view is that there are many small buffers, each of which is dedicated to a particular input/output pair, and no buffer sharing is possible. Therefore, buffers cannot be used as efficiently as in the output or shared buffer case.

In *shared* buffering, cells destined to different outputs share the same buffer resulting in a higher buffer utilization. Shared buffer SEs are, therefore, very attractive for the industry, and the majority of implemented prototypes use them as the building blocks for larger switching fabrics [13]. The main reasons are:

- The previous experience in this kind of devices acting as space division modules in conventional packet switches, or as time-division switching stages in the current circuit switching environment, where the principal component of the packet switch is a random access memory (RAM).

- As single stage MINs, they not only perform better, but also use less hardware, have smaller size, are suitable for VLSI implementation, and have the highest efficiency in hardware utilization because the two required functions of every packet switch, viz., queuing and switching, are carried out via buffering.

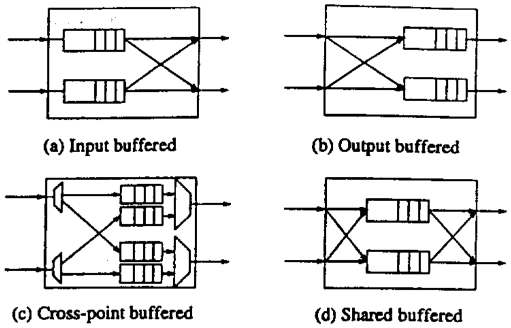
(a) Input buffered
(b) Output buffered
(c) Cross-point buffered
(d) Shared buffered

Figure 2: Four buffering schemes in SEs

The design of multi-port buffers in shared buffering is somewhat more complex than the buffers in input or crosspoint buffering, which can be simple FIFO's. Input and cross point buffering have single input buffers and are, therefore, simpler to design and need fewer hardware resources per buffer item than the output or shared buffer design where each buffer has two inputs. The hardware implementation of shared buffer SEs have been described in [14].

2.2 Windowed Service

If the shared multi-buffer switch architecture operate under the first-in-first-out (FIFO) rule, the head-of-line (HOL) blocking can arise to causes a deterioration in throughput or the rate of utilization of the output ports. To avoid this blocking, we employ a windowed access protocol [15] which however relaxes the FIFO rule. Under the windowed service rule, at the beginning of each clock cycle, the first w cells in each buffer sequentially contend for access to the switch outputs. The cells at the heads of the buffers contend first for access to the switch outputs. All the buffers then contend with their second cells for access to any remaining idle outputs (i.e., outputs not yet assigned to receive cells in this clock cycle). The contention process is repeated up to w times at the beginning of each clock cycle, sequentially allowing the w cells in an buffer's window to contend for any remaining idle outputs, until a cell is selected to transmit. A window size of $w = 1$ corresponds to the switch architecture with FIFO buffers.

2.3 Operating Assumptions

We make the following assumptions regarding the operation and the environment of the MINs [11, 16].

1. The MIN operates *synchronously* implying that packets move from one stage to the next only at the beginning of a clock cycle and thus the time axis is considered to be discrete. This reflects the situation in an ATM environment where all cells have a fixed length and fit exactly into one clock cycle.

2. A *back-pressure* mechanism ensures that no cells are lost within the MIN. Thus, a cell can only leave its buffer if the corresponding destination buffer at the next stage is able to accept it.

3. The arrival process at each input of the MIN is a simple *Bernoulli* process, i.e., the probability that a cell arrives within a clock cycle is constant and the arrivals are independent of each other. This implies that the inter-arrival time between two cells is geometrically distributed with a minimum distance of one clock cycle.

4. Each input link of the MIN is offered the same *traffic* load. The probability that an input link generates a request at the beginning of a clock cycle is r.

5. There is *no blocking* at the output links of the MIN. This means that the output links have at least the same speed as the internal links.

6. The *conflict resolution* logic at each SE is fair for input, output and shared buffer schemes, i.e., routing conflicts among cells at the inputs of a SE are randomly resolved. We consider the following three packet selection policies for the cross-point buffering scheme:

   - *Random selection* (RS): the multiplexer randomly selects a cell from the buffer of contending cells for the given output.
   - *New cell Selection* (NS): in this selection policy, the multiplexer selects a cell from that buffer which has a new cell at the head of the queue. If there is no such cell, it selects a cell using the RS policy.
   - *Blocked cell Selection* (BS): the multiplexer selects a cell from the buffer which has a blocked cell at the head of the queue. If there is no such cell, it selects a cell on RS basis.

7. The minimum possible *delay* of a cell is equal to $n+1$, where n is the number of stages. It includes the delay at the IBC buffer, since at least one time unit is spent in each buffer even when there is no waiting.

8. The total amount of buffer per SE is $2m$. Therefore, the total amount of buffer per input or output of an input, output or crosspoint buffered SE is m.

9. For a uniform traffic pattern, a request is equally likely to be directed to any output link of the MIN. For a hot spot traffic pattern, the probability that a generated request will be directed to a non-hot or hot output are $(1-h)/N$ and $h+(1-h)/N$ respectively, where h is defined to be the hot spot probability.

3 The Simulation Method

The assumptions mentioned in Section 2.3 were implemented in the simulator as follows.

1. At each clock cycle, a cell generator generates a cell with probability r (offered traffic load) at an input of the MIN. The cell generation is independent of cells generated at previous clock cycles and those at the other input ports.

2. The destination of a generated cell is taken from a uniform random number generator in the case of a uniform traffic, and in the case of a hot spot traffic, from a nonuniform random number generator which generates requests according to the hot spot probability (h) distribution mentioned in Section 2.3.

3. If there is a routing conflict among cells within an SE, a cell is selected randomly by another random number generator for input, output and shared buffered SEs. In the case of cross-point buffered SEs, either the randomly selection (RS) or the blocked packet selection (BS) is used (see Section 2.3).

4. First-in-first-out (FIFO) queuing policy is used at the buffers in the SEs of the input, output, and cross-point buffered SEs. Window selection policy is employed in the shared-buffered SEs.

5. The throughput and delay were measured at each output of the MIN, and averaged over the MIN size and simulation time span (typically 50,000 clock cycles) to get the normalized throughput and the mean delay of the MIN. The outputs for the first 500 clock cycles were discarded to allow the MIN 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 implement the 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 packets to an output; shift( ) to shift the packets forward in the internal buffer and the IBC buffers. We use a three-dimensional array to represent the buffers inside the MIN. The first dimension is the stage number, the second is identify an SE within a stage, and the third identifies 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 time to have a comprehensive picture of the switch behavior:

1. Number of simulation cycles (t_2). 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 request and the other for the resolution of the conflicts.

3. System size (N): Different MIN sizes were simulated.

4. Offered traffic load (r).

5. Probability (h) of a cell being destined for the hot output.

6. Internal buffer size (m) and IBC buffer size (f).

3.1 Request Generation

The built-in random number generator in the 'C' language library is used to obtain random requests at the beginning of each clock cycle. The random number generator is appropriately divided to generate requests according to the input parameters (i.e., rate of request generation and the probability of accessing the hot module). The actual demarcation process is portrayed in Figure 3. We define the following:

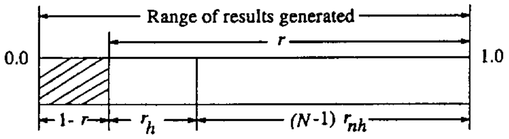

Figure 3: A pseudo-random generator.

r = portion which is valid request.

$1 - r$ = portion which is invalid request.

h = probability of hot spot.

$(N-1)r_{nh}$ = portion which is uniformly distributed among $N-1$ non-hot memories.

The effective hot spot probability is given by $r_h = rh + r_{nh}$ and $r_{nh} = \frac{(1-h)r}{N}$ gives the probability of accessing a non-hot output.

3.2 Parameters evaluated

Normalized throughput and *mean delay* were used as the criteria for the comparison of performance of the different buffering schemes. When the MIN reached a steady state after t_1 clock cycles, the number of valid cells at the outputs of the MIN were counted at the end of each clock cycle. These were averaged over a large number of cycles to give the normalized throughput (μ) as follows:

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

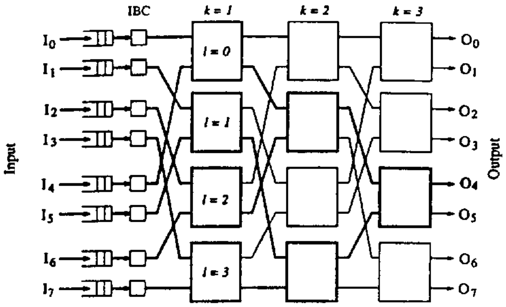

Figure 4: A MIN under a hot spot traffic pattern.

where, $\mu(l,t)$ is the throughput at the l-th output of the MIN during clock cycle t.

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

$$\tau = \frac{1}{N(t_2 - t_1)} \sum_{l=0}^{N-1} \sum_{t=t_1}^{t_2} \tau(l,t) \qquad (2)$$

where, $\tau(l,t)$ is the delay experienced by a packet (if there is one) at the l-th output of the MIN during clock cycle t, where t_1 is the number of initial simulation cycles allowed for the MIN to stabilize.

4 Results and Discussion

Four simulators have been developed for the simulation of MINs using input, output, crosspoint and shared buffered SEs. In this study, we have considered two types of traffic pattern, i.e., *uniform* and *hot spot* traffic. Figure 4 shows an 8 × 8 Omega switch under a hot spot traffic pattern. The SEs and links that carry hot traffic are shown in bold. We have simulated various MIN sizes under uniform and hot spot traffic patterns. Due to space limitations, we only show the results for switches of size 64 × 64. Figures 5 and 6 show the normalized throughput and mean delay versus input offered traffic load (r) for 64 × 64 MINs using the four types of SEs under uniform traffic pattern ($h = 0$). For input buffered SEs, the maximum normalized throughput of a MIN with buffer size six is limited to 0.56 under uniform input traffic pattern. This bottleneck, due to the head of the line (HOL) contention at each SE, is intrinsic to input queuing . When a cell at the head of a queue loses a contention, it impedes the rest of the cells in the same buffer from progressing forward, if cells are served on a FCFS basis. Another bottleneck arises when two or more cells contend for the same buffer in an SE. Since only one cell can be admitted to the buffer in one clock cycle, one of them is blocked and will have to retry in the next clock cycle. When buffers are placed at the output port of each SE or are shared, a very high throughput can be achieved. From Figure 5 and

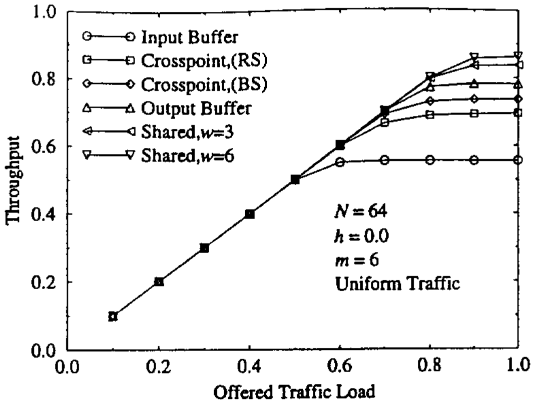

Figure 5: Normalized throughput versus offered load with m=6.

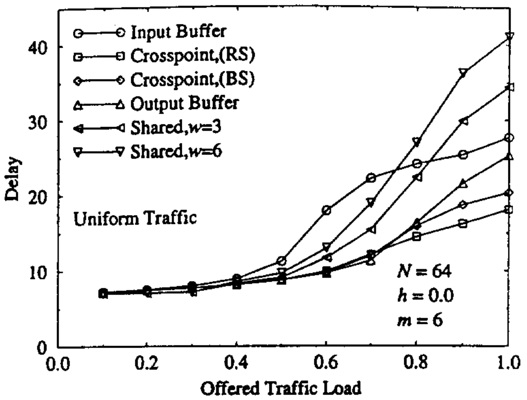

Figure 6: Mean delay versus offered traffic load with m=6.

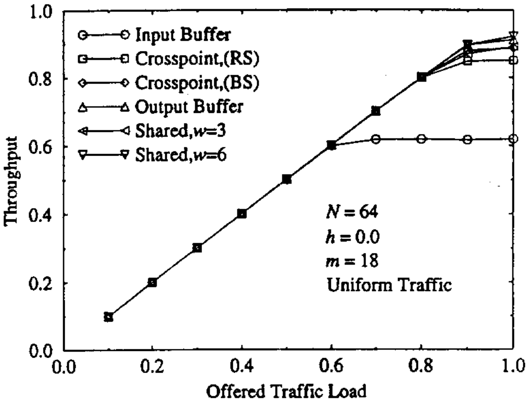

Figure 7: Normalized throughput versus offered load with m=18.

6, we see that the maximum normalized throughput of 0.78 is achieved for output buffer, 0.83 and 0.86 for shared buffer with w = 3 and 4 respectively, where w is the window size as mentioned in Section 2.2.

In Figure 7, the normalized throughput of various buffering schemes is shown as a function of the arrival rate for m = 18. The total amount of buffer space is the same for each buffering scheme. For instance, a total buffer space of 12 implies that each input buffer is of size 6, and each crosspoint buffer is of size 3. The maximum normalized throughput of the input buffered MIN built with 2 × 2 SEs is limited to about 0.62 even with a very large buffer (see Figure 9). However, with crosspoint, output, or shared buffering strategies, a normalized throughput of almost 0.9 is possible with moderately large buffer sizes. Shared buffering performs the best under uniform traffic. The window selection policy increases the performance drastically even with a small window (w). The performance of crosspoint buffering approaches the performance of output and shared buffering when the buffer size is increased to 18. Crosspoint buffering provides performance comparable to output and shared buffering under operating loads below 80%. Figures 6 and 8 also show the mean delay as a function of the arrival rate for various buffering schemes. When m = 6, reasonable delays can be achieved for both crosspoint buffering and output buffering up to a load of 0.6. However, when the buffer size is increased to 18, reasonable delays can be achieved for loads upto 0.8. Figures 9 and 10 show the normalized throughput and mean delay versus buffer size for MINs employing different buffering schemes under a uniform traffic pattern. Figures 11 and 12 plot the normalized throughput and mean delay for low hot spot probabilities (h) for four types of MINs under full load. h was varied from 0 to 0.11. Figure 13 shows the normalized throughput for high hot spot probabilities for various buffering schemes. h is varied from 0 to 1. As the hot spot probability increases, the normalized throughput decreases due to tree saturation.

Figures 14 and 15 show the normalized throughput and mean delay versus buffer size for MINs with different buffering schemes under a hot spot traffic pattern. Figures 16 - 19 show the normalized throughput and mean delay versus offered load for MINs under hot spot probabilities of 0.01 and 0.03. When h equals 0.01, the tree saturation occurs approximate at 0.6 for SEs using output, crosspoint or shared buffers, and 0.5 for input buffering (see Figures 16 and 17). At a low hot spot probability (h = 0.01), the output, crosspoint and shared buffer SEs have higher throughput than the input-buffered SE as shown in Figures 16 and 17. This is due to the HOL blocking in the case of input buffered SEs. For high hot spot probability (say h = 0.03), the normalized throughput is the same for all the buffering schemes. Comparison of results show that the performance of the shared buffer MIN is the best and an output-buffered and crosspoint MIN is much better than an input-buffered MIN when the hot request rate (h) is low. But the performance is the same for all the types of MINs

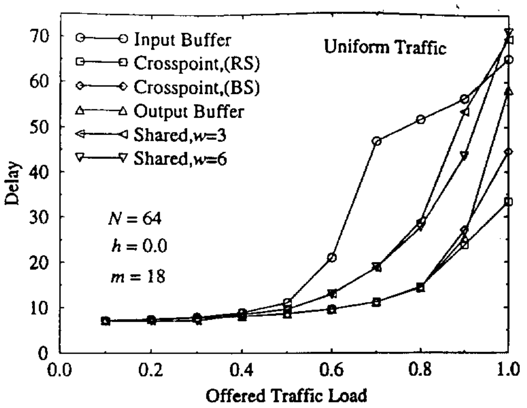

Figure 8: Mean delay versus offered traffic load with m=18.

when the level of hot requests is medium and high. This is again due to tree saturation. In Figure 20, the mean delay in MINs using different buffering schemes as a function of normalized throughput is shown for h decreasing from 1 to 0. We note an interesting phenomenon. The mean delay first increases to a maximum and then decreases as the normalized throughput increases due to a decrease in the hot spot probability. Certain general conclusions can be drawn:

- Cross-point, output and shared buffered SEs with a large buffer size have similar throughput. The normalized throughput of the above three types of SEs is better than input buffered SEs for small buffer size. The shared buffer has the best performance, followed by the output buffer and cross-point buffer.
- For offered traffic loads under 60%, the throughput of the input buffered MIN is reasonably close to those of the other three types, and because of its lower cost, crosspoint buffering may be the choice for implementation. However, in addition to the throughput limitation, crosspoint buffered SEs show a significant increase in mean delay even at loads as low as 60% (see Figures 7 and 8).
- Adding large buffers to a crosspoint buffered SE will not bring about a substantial performance improvement, since the throughput is limited to 0.62 due to the head of the line contention. Output, crosspoint and shared buffered SEs with small buffer sizes have significantly better performance than an input-buffered SE with a large buffer size. Thus if additional hardware resources are available to improve switch performance, they are better spent on implementing output, crosspoint or shared buffer designs than on making larger buffers for an input buffered design.

In Figure 6 and 8, the mean delay is compared for the four buffering schemes. The offered load is varied from 0.01 to 1.0. It is shown that the mean delay for MINs using crosspoint buffered SEs is smaller than that for the other three types of SEs.

5 Conclusion

Simulation models have been developed to evaluate the performance of MINs having different internal buffering schemes under uniform and hot spot traffic environment. It confirms the intuition that, under uniform traffic, shared buffer SEs have better performance than SEs having buffers at the inputs, outputs or crosspoints. In addition to performance, there are other issues, such as implementation, that must be considered in designing a MIN. We also compared the performance of SEs having buffers at the inputs, outputs, crosspoints and shared between the inputs and outputs, under the hot spot traffic pattern. The results show that the performance of shared and output-buffered MINs is considerably better than input-buffered MINs when the hot request rate is low. But the performance is the same for all the buffering schemes when the hot request rate is between medium to high. This is due to the onset of tree saturation at medium offered traffic loads. A comparison of the four approaches to providing the queuing for the SEs in MINs is given below.

Input Buffer

- Simple buffering structure.
- The internal link speed of the MIN is equal to the speed of the inputs or outputs of the MIN.
- Normalized throughput is limited to 0.62 with 20 buffers at each input of an SE.

Output Buffer

- Achieves optimal throughput/delay performance.
- The buffers should operate at a speed which is equal to the sum of the speeds of the input links of an SE.

Cross-point Buffer

- Simple buffering structure.
- The internal speed can be equal to the speed of the input/output links of an SE.
- Reduces the effect of head of line blocking.
- The total buffer required is much greater.

Shared Buffer

- Achieves high utilization of the buffers.
- The total amount of buffer memory required is small.

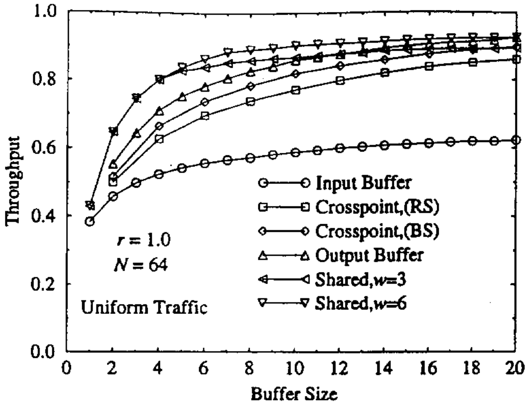

Figure 9: Normalized throughput versus buffer size under uniform traffic.

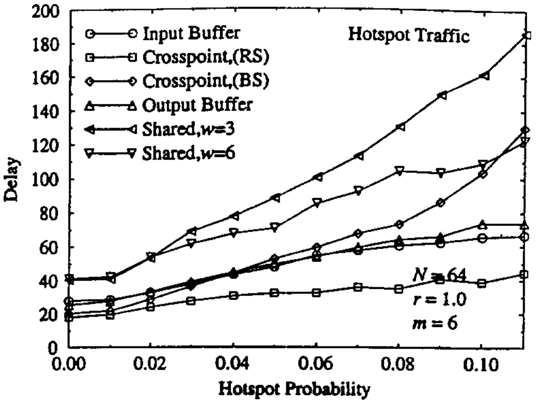

Figure 12: Mean delay versus low hotspot probabilities.

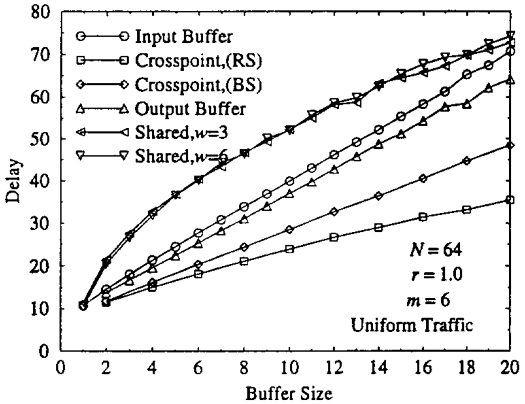

Figure 10: Mean delay versus buffer size under uniform traffic.

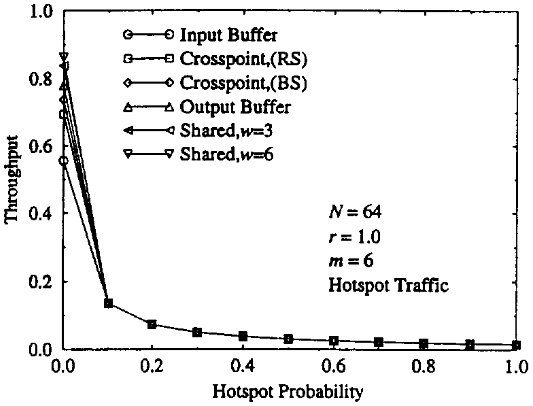

Figure 13: Normalized throughput versus high hotspot probabilities.

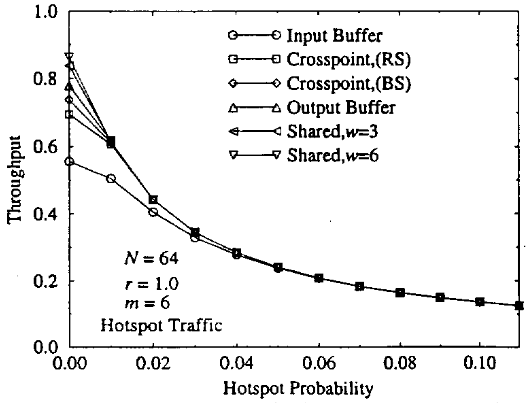

Figure 11: Normalized throughput versus low hotspot probabilities.

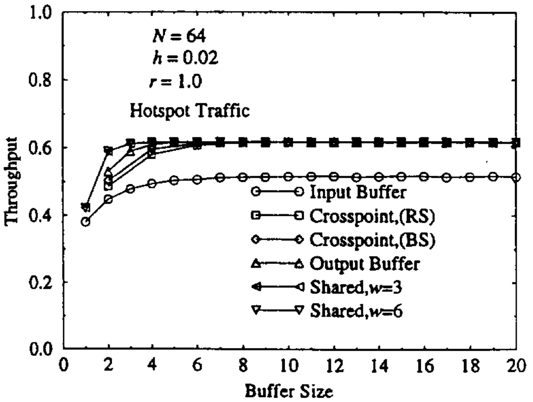

Figure 14: Normalized throughput versus buffer size under hotspot traffic.

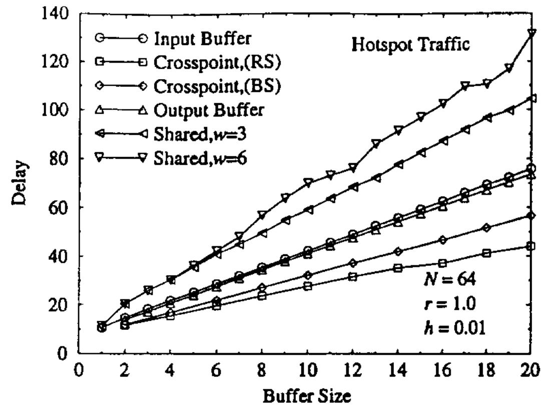

Figure 15: Mean delay versus buffer size under hotspot traffic.

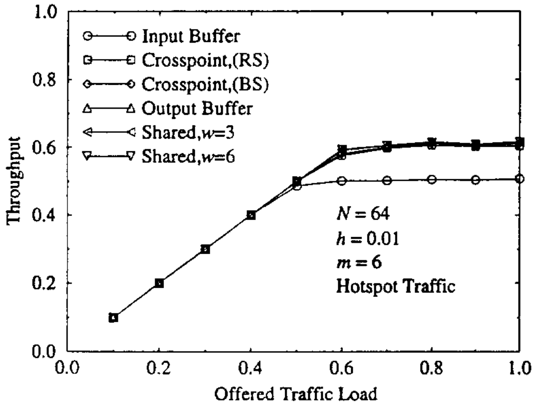

Figure 16: Normalized throughput versus offered traffic load under hotspot traffic, $h = 0.01$.

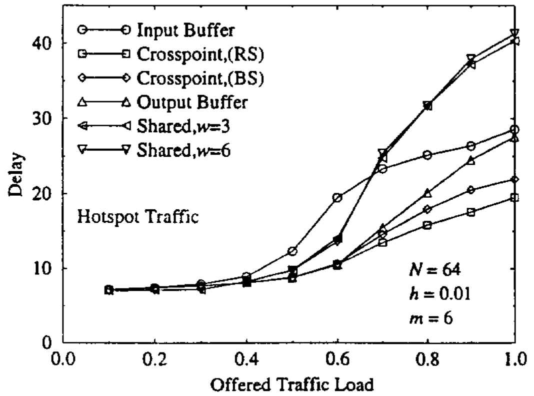

Figure 17: Mean delay versus offered traffic load under hotspot traffic, $h = 0.01$.

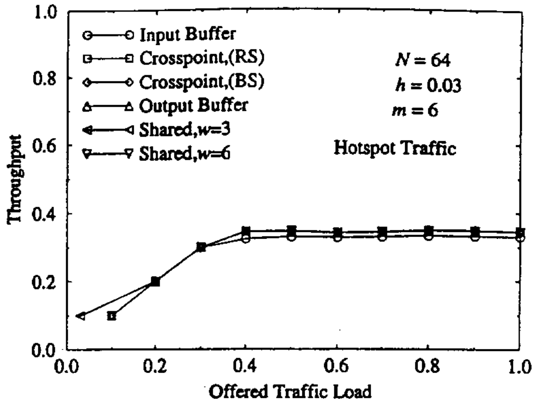

Figure 18: Normalized throughput versus offered traffic load under hotspot traffic, $h = 0.03$.

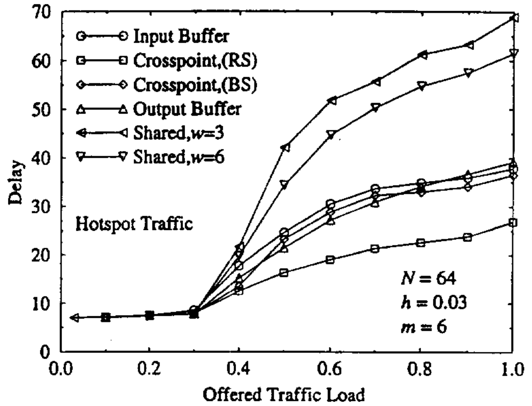

Figure 19: Mean delay versus offered traffic load under hotspot traffic, $h = 0.03$.

References

[1] B. Zhou and M. Atiquzzaman, "Performance of output-multibuffered multistage interconnection networks under general traffic patterns," *IEEE INFOCOM '94: Conference on Computer Communications*, Toronto, Canada, pp. 1448-1455, June 14-16, 1994.

[2] Y. Oie et al., "Effect of speedup in nonblocking packet switch," *ICC'89 Conf.*, Rec. Boston, MA, pp. 410-414, June 1989.

[3] P. Goli and V. Kumar, "Performance of a crosspoint buffered ATM switch fabric," *IEEE INFOCOM'92*, pp. 426-435, 1992.

[4] J.S. Turner, "Queueing analysis of buffered switching networks," *IEEE Transactions on Communications*, vol. 41, no. 2, pp. 412-420, February 1993.

[5] CCITT Recommendation I. 121, *Broadband Aspects of ISDN*. Fascicle III.7, Blue Book Geneva ed., 1989.

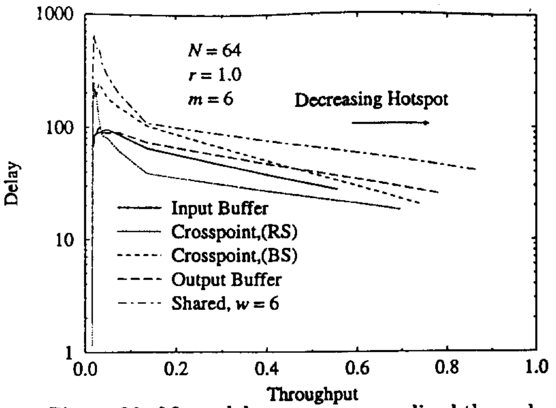

Figure 20: Mean delay versus normalized throughput under hotspot traffic.

[6] B. Zhou, K.E. Forward, and G.J. Armitage, "Simulation study of the interaction between a multi-media terminal and the ATM network," *Journal of Electrical and Electronics Engineering, Australia*, vol. 13, no. 1, pp. 41–52, Mar. 1993.

[7] Y. C. Jenq, "Performance analysis of a packet switch based on single-buffered banyan network," *IEEE J. Select. Areas Commun.*, vol. SAC-1, pp. 1014–1021, Dec. 1983.

[8] H.S. Kim, I. Widjaja, and A. Leon-Garcia, "Performance of output-buffered banyan networks with arbitrary buffer sizes," *IEEE INFOCOM '91: Conference on Computer Communications*, Bal Harbour, Florida, pp. 701–710, April 1991.

[9] 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.

[10] 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.

[11] M. Atiquzzaman and M.S. Akhtar, "Effect of nonuniform traffic on the performance of multistage interconnection networks," *IEE Proc.-Comput. Digit. Tech.*, vol. 141, no. 3, pp. 169–176, May 1994.

[12] 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, pp. 27–40, 1992.

[13] M. De Prycker, *Asynchronous transfer mode: Solution for Broadband ISDN.* Chichester, England: Ellis Horwood, second edition ed., 1993.

[14] H. Kondoh, H. Notani, H. Yamanaka, K. Higashitani, and H. Saito, "A shared multibuffer architecture for high-speed ATM switch LSIs," *IEICE Trans. Electron.*, vol. E76-C, no. 7, pp. 1094–1101, July 1993.

[15] G. Thomas, "On high speed packet switches with windowed input buffers," *IEEE GLOBECOM'93*, pp. 1406–1410, 1993.

[16] 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)*, North Carolina, USA, pp. 405–406, Jan. 31-Feb. 2, 1994.