[20] H. Yoon, K. Y. Lee, and M. T. Liu, "***Performance Analyis of Multibuffered Packet Switching Networks in Multiprocessor Systems***", IEEE Transactions on Computers, vol. 39, no. 3, pp. 319 - 327, March 1990.

[21] B. Zhou and M. Atiquzzaman, "***Performance of Output Multibuffered Multistage Interconnection Networks Under Non-Uniform Traffic Patterns***", International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'94), North Carolina, pp. 405 - 406, Jan. - Feb. 1994.

The simulations have demonstrated that multistage blocking networks have the unfortunate property that a sufficient concentration of references to one particular output (a hot-spot) can degrade the response of the network to all references, and not just those to the hot-spot link. The potential degradation is sufficient enough to cripple system performance.

Due to the increase in the contention at the hot link with an increase in the hot-spot probability, normalized cell delay increases significantly with an increase in the buffer size. A significant linear increase in cell delay caused by an increase of buffer size can degrade the performance of the MIN and hence override any potential throughput gain. Higher hot request probability also contributes to a faster increase rate of cell delay.

References

[1] 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, October 19-21, 1992.

[2] M. Atiquzzaman and M. S. Akhtar, "***Performance of buffered Mutistage Interconnection Networks in Non-Uniform Traffic***", 7th International Parallel Processing Symposium, California, pp. 762 - 767, April 1993.

[3] M. Atiquzzaman and M. S. Akhtar, "***Effect of Non-Uniform Traffic on the Performance of Multistage Interconnection Networks***", 9th International Conference on System Engineering, Las Vagas, pp. 31 - 35, July 1993.

[4] M. Atiquzzaman and M. S. Akhtar, "***Effect of Non-Uniform Traffic on the Performance of Unbuffered Multistage Interconnection Networks***", IEE Proceedings Part-E, 1994.

[5] S. Bassi, M. Decina, and A. Pattavina, "***Performance Analysis of the ATM Shuffleout Switching Architecture under Non-Uniform Traffic Patterns***", INFOCOM '92, pp. 734 - 742.

[6] C. K. Chen, "***Performance Evaluation of Finite Input Buffered Multistage Interconnection Networks with Arbitrary Sizes***", Master Thesis, LaTrobe University, 1993.

[7] D. M. Dias and J. R. Jump, "***Analysis and Simulation of Buffered Delta Networks***", IEEE Transactions on Computers, vol. C-30, no. 4, pp. 271-282, April 1981.

[8] S. H. Hsiao and R. Y. Chen, "***Performance Analysis of Single-Buffered Multistage Interconnection Networks***", 3rd IEEE Symposium on Parallel and Distributed Processing, pp. 864 - 867, December 1-5, 1991.

[9] M. N. Huber, E. P. Rathgeb, and T. H. Theimer, "***Self Routing Banyan Networks in an ATM Environment***", Computer Communication Technologies for the 90's (North-Holland), pp. 167 - 174, ICCC 1988.

[10] Y. C. Jenq, "***Performance Analysis of a Packet Switch based on Single-Buffered Banyan Network***", IEEE Journal on Selected Areas in Communications, vol. SAC-1, no. 6, pp. 1014 - 1021, December 1983.

[11] H. Kim and A. Leon-Garcia, "***Performance of Self-Routing ATM Switch under Non-uniform Traffic Patterns***", Proc. INFOCOM '90, pp. 140 -147.

[12] M. Kumar and G. Pfister, "***The Onset of Hot Spot Contention***", Proc. 1986 International Conference on Parallel Processing, August 1986.

[13] R. Lee, "***On 'Hot-Spot' Contention***", ACM Computer Architecture, vol. 13, pp. 15 - 20, December 1985.

[14] Y. S. Mun and H. Y. Youn, "***Performance Analysis of Finite Buffered Multistage Interconnection Networks***", Supercomputing 1992.

[15] J. H. Patel, "***Performance of Processor-Memory Interconnections for Multiprocessors***", IEEE Transactions on Computers, vol. C-3-, no. 10, pp. 771 - 780, October 1981.

[16] G. F. Pfister and V. A. Norton, "***Hot-Spot Contention and Combining in Multistage Interconnection Networks***", IEEE Transactions on Computer, vol. C-34, no. 10, pp. 943 - 948, October 1985.

[17] T. H. Theimer, E. P. Rathgeb, and M. N. Huber, "***Performance Analysis of Buffered Banyan Networks***", IEEE Transactions on Communications, vol. 39, no. 2, pp. 269 - 277, February 1991.

[18] L. T. Wu, "***Mixing Traffic in a Buffered Banyan Network***", Computer Communication Review, vol. 15, pp. 134 - 139, September 1985.

[19] P. C. Yew, N. F. Tzeng, and D. H. Lawrie, "***Distributing Hot-Spot Addressing in Large Scale Multiprocessors***", IEEE Transactions on Computers, vol. C36, no. 4, pp. 338 - 395, April 1987.

By considering non-uniform hot-spot traffic reference, the simulations have demonstrated that multistage blocking networks have an unfortunate property. A sufficient concentration of references to one particular output (a hot-spot) can degrade the response of the network to all references, and not just those to the hot-spot link. The potential degradation is sufficient to cripple system performance. Figure 13 shows the degradation in system bandwidth with different hot-spot probabilities and different buffer sizes under various request loads. It is rather interesting to observe that increased request load does not contribute to system throughput but the bandwidth is affected more by a change in the hot request probability.

Due to an increase in the contention at the hot link with an increase in the hot-spot probability, the normalized cell delay increases significantly with an increase in the buffer sizes. Figure 14 shows the increase in cell delay for various hot request probability with buffer size equals to 4. Figure 15 shows the increased rate of cell delay with various hot request probabilities. Although the increase in delay is linear, higher hot request probability contributes to faster rate of increase of the cell delay.

5. Conclusion

Our simulation results illustrate that the minimum throughput of any SE size under uniform traffic reference is 0.3. Observation has been made on the normalized delay (the average total number of network cycles a packet required to propagate through all the stages in the network) of the packet under uniform traffic reference. With an increasing load, delay increases linearly when the number of stages are two and three. However, a further increase in the number of stages causes an exponential increase in delay between request rates of 0.1 to 0.5. The normalized delay due to increasing the number of stage is also exponential.

For a fixed, large network size (4096 x 4096 in this case), the delay to propagate a packet through numerous stages plays a more significant role in network throughput than the delay caused by packet conflicts inside an SE. Hence, for better performance, it is better to implement a network with as large as possible SE sizes to reduce the number of stages. In general, under full uniform traffic load, higher throughput and shorter cell delay can be achieved with larger SE sizes (better implementation efficiency as well).

An alternative way to improve system bandwidth can also be achieved by increasing the SE's buffer size. However, it is interesting to observe that there is no significant increase in the normalized throughput when the request load is full. When the request load is between 0.4 to 0.8, there is a significant increase in system bandwidth. A further investigation on the effect caused by increased buffer size shows an exponential increase in cell delay when the request load is between 0.4 to 0.8. It is also observed that for the same request load, an increase in buffer size causes a linear increment in cell delay.

Extensive simulation runs show that for a fixed network size, larger SEs have higher throughput when the buffer size is small. This is due to lower propagation delay. Although the delay of the packet is linearly dependent on the SE's buffer size, the rate at which the delay increases is faster for smaller size SEs as compared to the larger size ones.

Simulation results have clearly demonstrated that an increase in the buffer size does increase the normalized throughput of the network. However, to avoid the cell delay penalty caused by the increased buffer sizes, larger size SEs are recommended over the smaller size one since the rate of increased of the delay is slower for larger size SE. A buffer size of two to four seems to be a reasonable compromise.

By considering non-uniform hot-spot traffic reference, it is rather interesting to observe that increased request load does not contribute to system throughput but the bandwidth is affected more by a change in the hot request probability.

internal conflict resolutions and the delay due to propagation through the network stages must be balanced.

For a fixed, large network size (4096 x 4096 in this case), Figure 6 shows how each different size SE performs. It is clear that the delay to propagate a packet through numerous stages plays a more significant role in network throughput than the delay caused by packet conflicts inside an SE. Hence, for better performance, it is better to implement a network with as large as possible SE sizes to reduce the number of stages.

Although a network implemented with large sized SE, will enhance the throughput, the cost for the large size SE need not to justify the performance gain. Figure 7 shows an interesting example of how the network will perform when the size of the SE is doubled from 2 x 2 to 4 x 4. The maximum gain obtained is only approximately a marginal 13% where the cost to use a 4 x 4 SE could be twice of the 2 x 2 SE. However, for a network size of 4096 implemented with 2 x 2 SE, 24,576 SEs will be needed. When implemented with 4 x 4 SE, a four times saving in the number of components required (6,144 SEs) could be a cheaper alternative depending on the actual market value of the different sizes SE.

Observations made from Figure 4-7 show that, in general, under full uniform traffic load, higher throughput and shorter cell delay can be achieved with larger SE sizes (better implementation efficiency as well).

An alternate way to improve system bandwidth is to increase the SE's buffer size. Figure 8 shows the performance of a crossbar SE under uniform traffic reference with various buffer sizes. It is interesting to observe that there is no increase in the normalized throughput when the request load is full. However, for request load between 0.4 to 0.8, there is a significant increase in system bandwidth. A further investigation on the effect caused by increased buffer size in Figure 9 shows an exponential increase in cell delay when the request load is between 0.4 to 0.8. It is also observed that for the same request load, an increase in buffer size causes a linear increment in cell delay.

Figure 10 illustrates the fact that for a crossbar SE under full load uniform traffic reference, an increase in the buffer size will not improve the system's normalized throughput.

Figure 11 shows that for a fixed network size, larger SEs have higher throughput when the buffer size is small (due to less propagation delay). However, once the buffer size starts to increase, normalized throughput increases with the increase in the number of stages. This is due to the fact that for a smaller size SE, the probability of internal conflict also decrease. Since there are finite buffers in the following stage, this means that the probability of the packet being forwarded is also higher, which is directly proportional to the normalized throughput of the network.

Although the delay of the packet is linearly dependent on the SE's buffer size, the rate which the delay increases is faster for smaller size SEs as compared to the larger size one. Figure 12 illustrates the rate of increase in cell delay for a fixed network size implemented with different size SEs.

Simulations on multiple input buffers MIN demonstrate that the increase in buffer size(s) does increase the normalized throughput of the network. This is due to the fact that as the number of the stages increases, the blocking probability of the packet also increases. With larger buffer size, more packets can be accommodated when there are packet conflicts, and hence the throughput of the network implemented with larger buffer size SEs is significantly better than the network implemented with smaller buffer size SEs. However, to avoid the cell delay penalty caused by the buffers, larger size SEs are recommended over the smaller size one since the rate of delay increase is slower for larger size SE and a buffer size of two to four seems to be a reasonable compromise.

request is not generated with a rate of one, then the effective hotspot probability equals e = Hot_Request * (1 - x).

Thus, if the generated number k is such that $x \leq k \leq (x + e)$, then the request is for the hot module. If the request falls outside the above range, then it is for any one of the (N - 1) non-hot modules and the distribution of the requests over these (N - 1) should be uniform. Hence, (1 - x - e) / (N - 1) gives the probability of referencing a non-hot module.

An example illustrating the above procedure is shown below:

range of random numbers generated	0 - *rand_max*
request rate	0.8
(valid range of requests is)	0.2 - 1.0
range of the hot output request	0.2 - 0.6
(hotspot probability is)	0.4 / 0.8 = 0.5
range of the non-hot output request	0.6 - 1.0
non-hot output request probability for	(0.4 / 0.8) / 7
a 8x8 network is therefore	= 0.0714

Routing conflicts are resolved randomly (as explained earlier in this section) and the packet which loses the contention remains in the queue waiting for the next cycle. A tag is associated with each valid packet at the end of the switching operation.

The various parameters evaluated for the simulated networks include the average system bandwidth and the average packet delay. At the outputs of the network, the number of valid packets are counted at the end of each cycle. These are averaged over the total number of cycles to give the average system bandwidth. For the calculation of packet delays, the number of requests blocked in each cycle are calculated, summed and averaged over the total number of cycles to give the average packet delay for a generated request.

Table 1 shows a sample output of the simulation program. The size of the simulated network, the SE size, the total number of cycle runs, different request rates, the minimum throughput, maximum throughput, average throughput, the bandwidth of the network, the minimum cell delay, maximum cell delay and the average cell delay are all listed in detail to facilitate the analysis.

Depending on the measurement criteria, relevant data were extracted from the simulation outputs and plotted to show the effect of changing the size of the SEs and the network sizes under uniform traffic with different request load. In the following section, an analysis of the simulation data will be performed and the results discussed.

4. Results and Discussions

The performance of networks configured with 2x2 SE is investigated and plotted in Figure 4. It is rather interesting to observe the sharp decrease in the network's normalized throughput as the network size increases (from 0.6 for 4 x 4 network size down to 0.3 for 4096 x 4096 network size). This reduction in performance can be explained with Figure 5, which shows a sharp increase in cell delay for 4096 x 4096 network size as compared to 4 x 4 network size (more than 3 times). This increase in cell delay is due to the large number of stages the packets have to propagate through in order to get to its destination (two stages for 4 x 4 network size but 12 stages when network size equals to 4096).

Observations based on the above plots indicate that for a fixed network size, an increase in SE sizes also increases the probability of internal conflicts and hence the time to resolve these conflicts is also increased. However, as the size of the SE increases, the number of stages required decreases, which helps to propagate the packet through the network faster. Therefore, in order to obtain a MIN's optimum throughput performance, the delay due to

the MSW is used for storing the cycle number during which the packet is generated, the maximum allowable network size to be simulated is 32K (and must be the power of 2). Network size can be expanded to 2G by simply changing the variables to '*long*' data type (provide there are enough computing resources in the running machine).

- SE Size: Must be a power of 2. Maximum allowable value is 4Gx4G.
- Buffer Size: Any value from 0 (1 for single buffer) to 4G.
- Random Number Generator Seed: The 'C' language library provides a random number generator (sadly, just one only). Since the simulation runs over a large number of cycles, the seed chosen was the largest seven digit prime number (*9999991*) to avoid any possible correlation.
- Random Number Generator: The simulation required two independent random number generators. One for the generation of the requests and the other for the conflict resolutions. Since the 'C' compiler supports only one generator, using the same stream for both could introduce dependency in the form of periodicity of the numbers. To take this into account, a large independent stream of random numbers were generated using another seed (which was found to be nearly as accurate as the other seed used for generating the numbers) and stored in a separate file. A number from this file was used to resolve the conflicts. The range of numbers is from *0* to *rand_max* and there are *a* packets between which this range is divided. If the generated number was between *x* / *a*, the packet on the *x* link will be selected.
- Normalized Throughput: The average number of packets put out in a unit of time (network cycle).
- Normalized Delay: The average interval between the time a packet was placed in a buffer at a network's input link and the time at which the packet left the network's output link.
- Hot Request: The probability that a packet at a network input link will request the hot output link of the network. For instance, if the hot request of the network input is *h* and the size of the network is *N*, then the probability that the packet will be destined for the hot output link of this network is equal to $\left(h+\frac{1-h}{N-1}\right)\cdot r$ (where *r* is the load at the input) and that particular output is termed as the *hotspot* output. The probability that the input packet will destined to a non-hot output is therefore $\left(\frac{1-h}{N-1}\right)\cdot r$.

Random request generation has been implemented in a separate routine to generate destination address at the beginning of each cycle and the generation process was allotted according to the input parameters (request load and the probability of accessing the hot module). Figure 3 shows the demarcation process.

The random number generated is between *0* and *rand_max*. If the request rate is *1*, then the whole range of *0* to *rand_max* represents the range of valid requests.

To simulate a rate of request less than unity (let say by an amount *x*, $0 \le x \le 1$), the range *0* to *x* has been marked as the invalid requests and no destination address was assigned to this range. The remaining *x* to *1* represents the range of valid requests.

The hot-spot probability (Hot_Request) is given as an input to the simulator. It is the probability that a packet requests the hot output, given that a request is generated. Hence if a

- Every packet has equal probability of winning the contention and the blocked packets are resubmitted to the original destination.
- All input requests are synchronized at the beginning of a clock cycle, and the service times of all memory modules are equal.
- The arrival of a packet during a cycle is independent of whether the packet received at the previous cycle was accepted or blocked.
- The packets received at an input is independent of the packet received by other inputs.
- Each input process generates a random request at the beginning of a memory cycle. The probability that a process requests a particular output at the beginning of a memory cycle is *r*. Thus *r* is the average number of requests generated per cycle by each input process.

Due to the changes of the network state propagating from the last stage to the first stage, the performance of a multistage interconnection network, in terms of throughput and delay, is calculated by recursively accumulating the number of packets exited from all output links within a predetermined period cycle. For a uniform traffic reference case, we typically used 50,000 cycles. Larger number of cycles are required for non-uniform hot spot traffic reference due to the lower request rate for the non hot-spot links.

The most common and significant measures for performance evaluation of any MIN is its system bandwidth and network delay. Bandwidth is measured by the throughput of the system, which is defined as the number of packets delivered by the network in unit time. Network delay is defined as the time taken by a packet to traverse the network.

3. Simulation Experiments

The simulation was performed at the packet level, assuming fixed-length packets transmitted in equal-length time slots, where a slot is the time required to forward a packet from one stage to the next.

The simulated network employs a distributed packet routing strategy and consists of separate buffers at each input port. Each packet entering the network contains header information which is used by the SEs to route it from the input to the appropriate output. Operating in a time-slotted fashion, the switch selects at the beginning of each time-slot up to *a* cells from the input buffers which act as FIFO queues and feeds them into the output port. A switch will not be able to transmit during full buffer situation and hence there will be no cell loss with the network.

The parameters for the cell traffic model are varied across simulation experiments to generate different offered loads and traffic patterns. Statistics such as packet delays and packet throughput are collected at the output ports. Estimates of the mean and variance of the delay and throughput are then readily available.

The following input data are varied each time in order to achieve a comprehensive picture of the network behavior:

- Simulation Cycles: In order to obtain time independent simulation results, the number of cycles for which the simulation have been performed must be large. In this case, *50,000* cycles.
- Stabilization Cycle: Before collecting the packet delay and network throughput statistics, the network is allowed to reach a steady state by discarding the first 1,000 cycles.
- Network Size: Due to the 32 bits length of the '*int*' data type and since the Least Significant Word (LSW) is *signed* and is used for storing destination address while

models can be verified and the analytical proofs of the various stochastic ordering properties for the steady-state queue length are also feasible.

This paper is organized as follows. Section 2 includes a review of banyan network, simulation assumptions for the measurement of the packet throughput and cell delay. Section 3 presents the actual simulation runs of the packet blocking and forward probabilities of the packet switched banyans under uniform and non-uniform traffic references with different finite input buffers. The section also illustrates the delay and throughput variance calculation for various networks implemented with different switching element sizes. Section 4 includes the analysis of the simulation data and a comparison of the performance of banyan networks implemented with different switching element architectures. Section 5 contains some concluding remarks and summary of our findings.

The results presented in this paper were obtained using a general event driven simulator for MIN written in ANSI C programming language and implemented on SUN workstations in an UNIX environment.

2. Background and Simulation Assumptions

The Banyan network is a general class of multistage interconnection networks which includes the Delta network [22] and the Omega network [16]. It is used to connect N inputs to N outputs using a multistage interconnection network. It consists of $n = log_a N$ stages of N / a SEs per stage, each SE having a input and a output lines. The i^{th} stage of switches will be denoted by S_i, $0 \leq i \leq (n - 1)$. A perfect shuffle interconnection pattern is used to connect the different stages as shown in Figure 2 for a network size of $N = 8$ and SE size of $a = 2$. Each SE is a crossbar switch allowing any input link to be connected to any one of the output links. In a typical packet switched banyan network, the data to be transferred between an input and output port is first assembled into fixed sized packet. The packet is transmitted from one stage to the next, requiring only one link per cycle, until it reaches its destination. Throughout this paper, it is assumed that each SE is built of a single integrated circuit and that all the SEs in a network are of the same size.

The operation of the network will be synchronized to a global clock. At time $t = 0$, all buffers in the network are empty. The switching operation starts once requests are received at the inputs. The destination address in a packet is an n-bit number represented by $D = (d_0\ d_1\ d_2\ d_3 \ldots\ d_{n-2}\ d_{n-1})_2$. The operation of the SEs is controlled by the bits of the destination address. A switch at S_i inspects bits $d_{(\ln a/\ln 2)*i} \ldots\ldots d_{(\ln a/\ln 2)*(i+1)-1}$ and routes the message to its output. A unique path of constant length exists between any input-output pair. A conflict arises if both the inputs of an SE need to be connected to the same output of the SE. An arbitration scheme, based on random selection, is used to resolve such conflicts (also see Figure 2).

A network cycle is assumed to be composed of two phases. In phase 1, sending buffers check the availability of space at the destined buffers. Control signals are assumed to have passed across the network to determine whether a packet is able to move forward one stage or whether it is to be blocked. This operation is done from the output stage to the input stage.

In phase 2, depending on the availability of the buffers, the actual routing of packets will take place. The following assumptions for uniform traffic and non-uniform hot spot traffic references are employed;

- Each processor generates packets with a certain probability. The packets are uniformly distributed over all outputs for uniform traffic reference or biased toward the hot link for non-uniform hot spot traffic reference.
- All the buffers in the SEs are identical in nature.
- The queuing and dequeuing of packets in a buffer take place during the same cycle.

Performance Study of Multistage Interconnection Networks with Arbitrary Switch Sizes and Finite Input Queuing Buffers

Chang-Kai Chen
chenc@latcs2.lat.oz.au

Mohammed Atiquzzaman
atiq@latcs2.lat.oz.au

LaTrobe University, Department of Computer Science and Computer Engineering
Melbourne, Victoria 3083, Australia.

Abstract. *One of the potential uses of multistage interconnection networks (MINs) is in large parallel multiprocessors computer system. MIN provides greater flexibility in the utilization of shared memory resources. In this paper, we present a performance evaluation of multistage self-routing banyan network. The network uses axa switching elements (SEs) with finite input buffers. Uniform traffic and non-uniform destination distributions are used in the study. Traffic measurements which have significant impact on the network performance are the packet throughput and its mean time to traverse the network under different traffic conditions and the sizes of the buffer within a switching element.*

1. Introduction

Buffered non-blocking (where multiple inputs of an SE can reach an output of the SE due to packet being buffered) self-routing multistage interconnection networks are commonly regarded as the most cost effective way to utilize shared resources (e.g. memory) in parallel multiprocessors architectures. The primary purpose of buffers in a switching element is to prevent loss of packets due to routing conflicts. The winning packet is forwarded only if the output link to which it is to be routed is able to accept the packet. No packet will be lost once it enters an internally buffered network. An axa switch with all inputs and outputs operating at the same bit rate, and packets arriving in a time-slotted fashion is shown in Figure 1.

The performance evaluation of multistage interconnection networks in terms of the traffic characteristics and switch parameters is critical for comparing alternative switch designs, planning appropriate traffic measurements, and developing network congestion control strategies. Multistage interconnection networks have also been proposed as ATM switching fabrics in broadband ISDN networks.

Previously, the approximate performance of packet switched banyans based on analytical modeling under uniform traffic reference have been presented by [7-10, 14-15, 17, 20]. Approximate performance of MIN with non-uniform and hot-spot traffic reference have been presented in [1-6, 11-13, 16, 18-19, 21]. Since these analysis made some simplifying assumptions, the results obtained were having errors for heavy input traffic.

The goal of this simulation study is to investigate the performance of a MIN (as measured by packet throughput and packet delays) as a function of SE sizes, buffer sizes and network sizes. Such a study is useful in the comparative assessment on the designs of alternative switch architectures and the effects of various traffic characteristics on the network performance. Thus network congestion control strategies can be developed. In addition to these observations, the simulation study also proved helpful in giving directions for future analytical work in this area. In particular, performance parameters obtained from the various analytical