PERGAMON

Computers and Electrical Engineering 25 (1999) 77–94

Computers and Electrical Engineering

Multiple-bus multiprocessor under unbalanced traffic

M.A. Sayeed [a], M. Atiquzzaman [b,*]

[a] Cabletron Systems, 50 Minuteman Road, Andover, MA 01810, USA
[b] Department of Electrical and Computer Engineering, Univeristy of Dayton, Dayton, OH 45469-0226, USA

Received 6 January 1998; received in revised form 31 July 1998; accepted 25 August 1998

Abstract

Performance evaluation of multiple-bus multiprocessor systems is usually carried out under the assumption of uniform memory reference model. Hot spots arising in multiprocessor systems due to the use of shared variables, synchronization primitives, etc. give rise to non-uniform memory reference pattern. The objective of this paper is to study the performance of multiple bus multiprocessor system in the presence of hot spots. Analytical expressions for the average memory bandwidth and probability of acceptance of prioritized processors have been derived. Two new phenomenon, coined as *bumping* and *knee effect*, have been observed in the acceptance probabilities of the processors. The results are validated by simulation results.

Keywords: Interconnection networks; Performance evaluation; Memory bandwidth; Blocking probability; Multiprocessors; Parallel processing

1. Introduction

An increasing demand for large amounts of computing power has led to the design and analysis of multiprocessor systems, where several processors are employed to speed up the execution of algorithms and parallel memory modules are used to increase the memory bandwidth of such systems. An interconnection network is used to connect the processors and memories together.

A variety of interconnection networks have been proposed and analyzed [1, 2]. Because of modularity and fault tolerance of multiple bus systems, such systems have been widely investigated [3]. Conflicts in a multiple-bus system arise due to a number of processors trying

* Corresponding author. Tel.: +1-937-229-3183; Fax: +1-937-229-4529; E-mail: atiq@engr.udayton.edu.

0045-7906/99/$ - see front matter
PII: S0045-7906(98)00028-7

to access several memory modules in parallel. Bus conflicts arise due to the limited number of buses. Moreover, more than one processor requesting the same memory module results in a memory conflict. Different types of conflict resolution strategies are adopted depending on applications.

Performance of a multiprocessor system degrades due to the above mentioned conflicts. Commonly used criteria for performance measurement are the average memory bandwidth of the system and the probability of acceptance of processor requests. Other criteria like processor utilization, throughput and blocking probability, etc. are easily derived from the above two criteria. In analyzing the performance, most authors assume a *uniform memory reference* (URM) model where all the memories are equally likely to be accessed by the processors [4–14], i.e. the probability with which a processor requests a memory module is the same for all the modules.

The uniform memory reference model is rather restrictive in real-world situations. Memory references in a multiprocessor system are not necessarily uniform. Non-uniformities arising due to locality of references have been analyzed by several authors. Das [15] and Bhuyan [16] have analyzed the case where each processor has a different favorite memory which it accesses with a higher probability than other memories. A model with local referencing, where the probability of successive requests to the same memory module is higher, has been analyzed by Sethi [17] and Irani [18]. The above type of local referencing arises in cases like array accesses by user programs. Siomalas [19] has analyzed a system where each processor has a local reference pattern and a local memory module which it references immediately after requesting another memory module. This has been called the immediate return reference pattern. Mudge [20] has used a favorite memory assumption similar to that used in Ref. [15]. Bandwidth of a multiple bus system for uniform and favorite memory references has been determined by modeling the system as a t-out-of-s system [21]. Other performance criteria, like the probability of acceptance, have not been considered in Ref. [21].

Another type of non-uniform memory reference arises in multiprocessor systems due to the use of variables for locking, global and barrier synchronization, pointers to shared queues, etc. These are indivisible primitives and must be stored in a single shared memory. The primitives are accessed by all the processors, giving rise to an increased number of reference to the memory module where they are stored. This type of memory module is called a *hot memory* and the phenomenon as *hot-spot contention*. Pfister [22] first noticed that in buffered multistage interconnection networks (MIN), hot spot contention gives rise to tree saturation which severely degrades the performance of such networks. Performance of crossbar multiprocessors, unbuffered MINs and buffered MINs in the presence of hot spots have been developed by Atiquzzaman [23–26]. Combining and feedback schemes have been suggested as solutions to the problem in multistage interconnection networks [22, 27, 28]. The bandwidth of a crossbar system under hot spot traffic conditions have been determined in Ref. [23]. The model assumes processors with uniform priority.

Analytical modeling permits one to obtain a quick evaluation of the expected performance of the system as a function of the different parameters. The objective of this paper is to determine the performance of a multiple bus interconnection network under hot spot conditions. Analytical expressions for *memory bandwidth* and the *probability of acceptance* of processor requests are derived for processors having *different priorities* in the case of conflicts.

Note that a uniform memory reference model is a special case of the hot spot request pattern. It is shown that the bandwidth is affected by a change in the hot spot probability for high request rates. Moreover, for high request rates and a small number of buses, the degradation at low hot spot probabilities is mainly due to bus conflicts and memory conflicts play an insignificant role in the degradation. We have observed two new phenomenon in the acceptance probabilities of the requests from prioritized processors. We call these the *bumping effect* and the *knee effect*. With an increase in the hot spot probability, the bumping effect results in an increase and decrease in the acceptance probabilities of the low and high priority processors respectively. The knee effect explains the fact that a system having prioritized processors may appear to be a crossbar system for some processors, while it may appear as a multiple-bus system for the rest of the processors.

To determine the amount of degradation due to non-uniform memory references, performance results are compared with those from the uniform memory reference model.

The rest of the paper is organized as follows. The modeling assumptions are given in Section 2. Analytical models for determining the average memory bandwidth and the probability of acceptance of processor requests are presented in Section 3, followed by results in Section 4.

2. Assumptions

We model the multiple-bus system under the following assumptions.

- The system consists of N identical processing elements (PE), M identical memory modules (MM) and $B \leq \min(N,M)$ common buses (see Fig. 1). Contention in the system is due to bus conflicts and/or memory conflicts. The processing elements and memory modules will be represented by $\{PE_0, PE_1, \ldots, PE_{N-1}\}$ and $\{MM_0, MM_1, \ldots, MM_{M-1}\}$, respectively. For $B = \min(N, M)$, the system reduces to a crossbar multiprocessor.
- The system operates *synchronously*, i.e. each processor generates a request at the start of a memory cycle with a probability r. All memory modules, for which a bus is available, are accessed simultaneously.
- Processor requests are *spatially independent*, i.e. requests from the different processors are independent of each other.

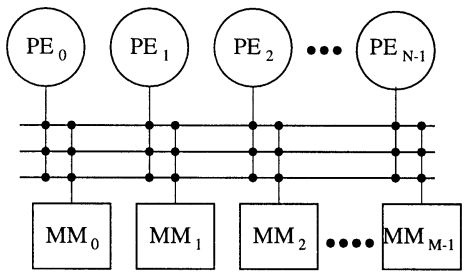


Fig. 1. Multiple-bus multiprocessor systems.

- *Temporal independence* of requests is assumed, i.e. requests from a processor at different memory cycles are independent, implying that requests rejected due to memory conflict or unavailability of buses are simply discarded.
- Processors have *priority*. PE_i, $0 \le i \le N-1$, has a higher priority than PE_{i+1}. The priority is used to resolve both bus and memory conflicts.
- Requests from processors to the different memory modules are *not uniformly distributed*. Memory module M_h is a hot memory for all processors. The probability of PE_i, $0 \le i \le N-1$, requesting MM_h is p_h, and of requesting MM_j, $j \ne h$, is given by $p_0 = (1-p_h)/(M-1)$, where $p_h > 1/M$. Note that $p_h = 1/M$ is the uniform reference model widely used in the literature.

3. Performance evaluation

Performance models of a multiprocessor system operating under the above assumptions will be developed in this section. Average memory bandwidth and probability of acceptance will be used as the measures of performance.

3.1. Average memory bandwidth

Average memory bandwidth (AMBW) of a multiprocessor system is defined as the average number of memory modules accessed simultaneously during a memory cycle. The average memory bandwidth of an N processor, M memory, B bus system with a processor request rate of r will be denoted by AMBW(N, M, B, r). AMBW of a multiple-bus system under the assumptions mentioned in Section 2 for $r = 1$ is given by

$$\mathrm{AMBW}(N, M, B, 1) = \sum_{k=1}^{B} k \Pr(k) + \sum_{k=B+1}^{\min(N,M)} B \Pr(k) \tag{1}$$

where, $\Pr(k)$ is the probability that exactly k memory modules are accessed in a cycle. For a URM, $\Pr(k)$ is given by [4]

$$\Pr(k) = \frac{k! S(N, k)}{M^N} \binom{M}{k} \tag{2}$$

where $S(N, k)$ is the Stirling number of the second type [29]. In the rest of this section, $\Pr(k)$ for a hot spot reference model (HSRM) will be derived, which will be used to determine the bandwidth using Eq. (1).

3.1.1. Request probability = 1

First, let's consider $r = 1$. Let ${E_k}^{p,q}$ be the event that k distinct memory modules are requested during a cycle, where p and q are the number of references for MM_h and MM_js, $j \ne h$, respectively. Hence, $p + q = N$ for $r = 1$. Depending on the distribution of requests to hot and non-hot memories, we can divide ${E_k}^{p,q}$ into two classes.

Class 0: all the requests are to non-hot memory modules. This event will be denoted by $E_k^{0,N}$.

Class 1: i requests ar to the hot memory and $N-i$ requests to the non-hot memories. This event will be denoted by $E_k^{i,N-i}$, $i \neq 0$.

The probability that k distinct memory modules are requested during a cycle is, therefore, given by the sum of the probabilities of the two classes. It can be shown [23] that

$$\Pr(k) = k!S(N, k)\binom{M-1}{k}p_0^N + (k-1)!\left(\binom{M}{k} - \binom{M-1}{k}\right)\sum_{i=1}^{N-k+1} S(N-i, k-1)$$

$$\binom{N}{i}p_h^i p_0^{N-i} \tag{3}$$

Substituting Eq. (3) in Eq. (1) gives the expression for the average memory band-width, AMBW(N, M, B, 1), of an $N \times M \times B$ system for $r = 1$.

3.1.2. Request probability < 1

Lets now consider the case when a processor does not request memories at every cycle, i.e. the processor request rate $r < 1$. The probability of having exactly n processors requesting memory modules at the beginning of a memory cycle is $\binom{N}{n}r^n(1-r)^{N-n}$. The conditional probability of having k distinct memory modules being requested given that n memory requests have been generated in a cycle is given by

$$\Pr(E_k^{0,n} | n \text{ requests have been generated}) = k!S(n, k)\binom{M-1}{k}p_0^n \tag{4}$$

Therefore, class 0 probability for $r < 1$ is given by

$$\Pr(E_k^{\text{class 0}}) = \sum_{n=1}^{N} k!S(n, k)\binom{M-1}{k}p_0^n\binom{N}{n}r^n(1-r)^{N-n} \tag{5}$$

Similarly, the class 1 probability of requesting k distinct MMs such that MM_h is one of them is given by replacing N by n in Eq. 6 in Ref. [23] and multiplying it by the probability of n processors requesting memories and $(N-n)$ processors not requesting.

$$\Pr(E_k^{\text{class 1}}) = \sum_{n=1}^{N}\left(\binom{M}{k} - \binom{M-1}{k}\right) \times \sum_{i=1}^{n-k+1} (k-1)!S(n-i, k-1)\binom{n}{i}p_h^i p_0^{n-i}$$

$$\binom{N}{n}r^n(1-r)^{N-n} \tag{6}$$

Average memory bandwidth is, therefore, given by

$$\begin{aligned}
\mathrm{AMBW}(N, M, B, r) &= \sum_{k=1}^{B} k \Pr(k) + \sum_{k=B+1}^{\min(N,M)} B \Pr(k) \\
&= \sum_{k=1}^{B} k(\Pr(E_k^{\text{class } 0}) + \Pr(E_k^{\text{class } 1})) + \sum_{k=B+1}^{\min(N,M)} B(\Pr(E_k^{\text{class } 0}) \\
&\quad + \Pr(E_k^{\text{class } 1})) \\
&= \sum_{k=1}^{B} k \Bigg\{ \sum_{n=1}^{N} k! S(n, k) \binom{M-1}{k} p_0^n \binom{N}{n} r^n (1-r)^{N-n} + \sum_{n=1}^{N} \Big(\binom{M}{k} - \\
&\quad \binom{M-1}{k} \Big) \times \sum_{i=1}^{n-k+1} (k-1)! S(n-i, k-1) \binom{n}{i} p_h^i p_0^{n-i} \\
&\quad \binom{N}{n} r^n (1-r)^{N-n} \Bigg\} + \sum_{k=B+1}^{\min(N,M)} B \Bigg\{ \sum_{n=1}^{N} k! S(n, k) \binom{M-1}{k} p_0^n \\
&\quad \binom{N}{n} r^n (1-r)^{N-n} + \sum_{n=1}^{N} \Big(\binom{M}{k} - \\
&\quad \binom{M-1}{k} \Big) \times \sum_{i=1}^{n-k+1} (k-1)! S(n-i, k-1) \binom{n}{i} p_h^i p_0^{n-i} \\
&\quad \binom{N}{n} r^n (1-r)^{N-n} \Bigg\} = \sum_{n=1}^{N} \binom{N}{n} r^n (1-r)^{N-n} \mathrm{AMBW}(n, M, B, 1)
\end{aligned} \tag{7}$$

If the average memory bandwidth of a system with $r = 1$ is known, Eq. (7) can be used to calculate the average memory bandwidth of a system with $r < 1$.

3.2. Probability of acceptance

The second performance criterion to be evaluated is the probability of acceptance, denoted by P_a. It is defined as the probability that a processor's request is accepted. P_a for both crossbar and multiple bus system under HSRM will be determined. If all the PEs have equal priority i.e. each processor has equal probability of being accepted in case of bus or memory conflicts, P_a is given by

$$P_a = \frac{\mathrm{AMBW}(N, M, B, 1)}{Nr} \tag{8}$$

Processors in a multiprocessor system are usually prioritized. P_a for processors having different priorities will be determined. The probability of acceptance for the nth processor will

be denoted by $P_a(n)$. We assume that PE_j has a higher priority than PE_{j+1}, for $j = 0, \ldots N-2$. Priorities will be used for resolution of bus and memory conflicts. PE_n can, therefore, be blocked only by processors of higher priority, PE_i, $0 \le i \le n-1$. Let's assume that PE_n and i other processors of higher priority than PE_n generate requests during a memory cycle. The analysis will be divided into two cases:

Bus sufficient system (BSS): there is no possibility of PE_n being blocked because of bus conflicts, i.e. the system is either a crossbar, or the system is not a crossbar but $i < B$. The only possible conflict due to which PE_n may be blocked by a processor of higher priority is because of memory conflicts.

Bus deficient system (BDS): PE_n may be blocked due to bus and/or memory conflicts, i.e. the system is not a crossbar and $i \ge B$.

3.2.1. Analysis of bus sufficient system (BSS)

Since there is no bus conflict in this case, the probability that PE_n requests a memory module and is accepted, $P_a(n)$, is equal to the probability that no other processor(s) of higher priority request the particular memory module requested by PE_n. We can further subdivide this case into two subcases:

Case BSS-H: PE_n requests the hot MM and the request is accepted.
Case BSS-NH: PE_n requests a non-hot MM and the request is accepted.

$P_a(n)$ for the BSS case will, therefore, be the weighted sums of the probabilities of cases BSS-H and BSS-NH. Let MM_h be a hot MM. Therefore,

$$P_a(n)|_{BSS} = p_h \cdot P_a(n)|_{BSS\text{-}H} + (M-1) \cdot p_0 \cdot P_a(n)|_{BSS\text{-}NH}$$
$$= p_h \cdot P_a(n)|_{BSS\text{-}H} + (1-p_h) \cdot P_a(n)|_{BSS\text{-}NH} \quad (9)$$

where $P_a(n)|_{BSS\text{-}H}$ and $P_a(n)|_{BSS\text{-}NH}$ are evaluated in Sections 3.2.1.1 and 3.2.1.2.

3.2.1.1. Analysis of case BSS-H. The probability that PE_n requests MM_h and gets it is equivalent to the probability that none of the n higher priority processors request MM_h. The probability that a particular set of i higher priority processors do not request MM_h and the rest $(n-i)$ higher priority processors do not generate a memory request is $r^i(1-p_h)^i(1-r)^{n-i}$. The set of i processors can be chosen out of n processors in $\binom{n}{i}$ ways and i ranges from 0 to n. Therefore, the probability of acceptance for case BSS-H is given by

$$P_a(n)|_{BSS\text{-}H} = \sum_{i=0}^{n} \binom{n}{i} r^i (1-r)^{n-i} (1-p_h)^i \quad (10)$$

3.2.1.2. Analysis of case BSS-NH. As in the case of BSS-H, let a set of i, $0 \le i \le n$, higher priority processors request memories and the other $(n-i)$ higher priority processors do not request any memory during a cycle. The probability that PE_n requests a non-hot MM and gets it is equivalent to the probability that none of the i higher priority requesting processors request the particular non-hot MM requested by PE_n. Of the set of i requests, let a set of k,

$0 \le k \le i$, be directed towards a set of j, $1 \le j \le k$, distinct non-hot MMs and the rest $(i-k)$ towards the MM_h. The probability of $(n-i)$ processors not requesting, k processors requesting non-hot MMs and $(i-k)$ processors requesting the hot MM is

$$(1-r)^{n-i}(r \cdot p_h)^{i-k}(r \cdot p_0)^k$$

The i processors can be selected out of the n processors in $\binom{n}{i}$ ways and the remaining $(i-k)$ requests can be selected out of i requests in $\binom{i}{i-k}$ ways. The j distinct non-hot MMs can be selected out of the $(M-2)$ non-hot MMs (i.e. excluding the non-hot MM requested by PE_n) in $\binom{M-2}{j}$ ways. The k requests can be distributed among the j MMs, such that none of them is empty, in $j!S(k, j)$ ways. Therefore, the probability that PE_n requests a non-hot MM and is not blocked by any processor of higher priority is given by

$$\begin{aligned} P_a(n)|_{\text{BSS-NH}} &= \sum_{i=1}^{n}\sum_{k=0}^{i}\sum_{j=1}^{k}\binom{n}{i}(1-r)^{n-i}(r \cdot p_h)^{i-k}(r \cdot p_0)^k\binom{i}{i-k}\binom{M-2}{j}j!S(k, j) \\ &= \sum_{i=0}^{n}\binom{n}{i}r^i(1-r)^{n-i}\sum_{k=0}^{i}\binom{i}{i-k}p_h^{i-k}p_0^k\sum_{j=1}^{k}j!S(k, j)\binom{M-2}{j} \end{aligned} \tag{11}$$

Substituting Eqs. (10) and (11) in Eq. (9) and rearranging terms gives

$$\begin{aligned} P_a(n)|_{\text{BSS}} = \sum_{i=1}^{n}\binom{n}{i}r^i(1-r)^{n-i}\Bigg\{ & p_h(1-p_h)^i \\ & + (1-p_h)\left(\sum_{k=0}^{i}\binom{i}{i-k}p_h^{i-k}p_0^k\sum_{j=1}^{k}j!S(k, j)\binom{M-2}{j}\right)\Bigg\} \end{aligned} \tag{12}$$

Eq. (12) gives the probability of acceptance of prioritized processors in a crossbar system, or in a multiple bus system having low processor request rates such that there is no bus conflict, or for a PE_n, where $n \le B + 1$.

3.2.2. Analysis of bus deficient system (BDS)

In this case the system is not a crossbar and $i \ge B$. PE_n is not blocked if it can be assigned a bus to access the requested MM and none of the i higher priority processors (requesting memory) request the particular MM requested by PE_n. As before, we can sub-divide the analysis into two sub-cases:

Case BDS-H: PE_n requests the hot MM and gets it.
Case BDS-NH: PE_n requests a non-hot MM and gets it.

3.2.2.1. Analysis of case BDS-H. In a bus deficient system, if PE_n generates a request for the hot MM, it will be accepted if the i higher priority requesting processors request no more than $(B-1)$ distinct non-hot MMs out of the $(M-1)$ non-hot MMs. Let the i requests be for j distinct non-hot MMs. The probability of this event is $(1-r)^{n-i}(r \cdot p_0)^i$. The i processors can be selected out of n processors in $\binom{n}{i}$ ways. Also, i requests can be partitioned into j distinct non-

hot MMs in $j!S(i,j)$ ways and j, $1 \le j \le \min(i,B-1)$, distinct non-hot MMs can be selected out of $(M-1)$ MMs in $\binom{M-1}{j}$ ways.

Therefore, the probability that PE_n requests MM_h and is accepted, is given by

$$P_a(n)|_{BDS\text{-}H} = \sum_{i=0}^{n} \sum_{j=1}^{\min(i,B-1)} (1-r)^{n-i}(r\cdot p_0)^i \binom{n}{i}\binom{M-1}{j} j!S(i,j)$$
$$= \sum_{i=0}^{n} \binom{n}{i} r^i (1-r)^{n-i} p_0^i \sum_{j=1}^{\min(i,B-1)} j!S(i,j)\binom{M-1}{j} \tag{13}$$

3.2.2.2. Analysis of case BDS-NH. In a bus deficient system, if PE_n generates a request for a non-hot MM, the request will be accepted if the i higher priority requesting processors do not request the particular non-hot MM requested by PE_n and the i requests are directed to no more than $(B-1)$ distinct MMs. These $(B-1)$ MMs may or may not include the hot MM. Of the i requests, let k, $0 \le k \le i$ of them be for j distinct non-hot MMs and $(i-k)$ are for MM_h. The probability of this event is

$$(1-r)^{n-i}(r\cdot p_h)^{i-k}(r\cdot p_0)^k$$

The i processors can be selected out of n processors in $\binom{n}{i}$ ways and the $(i-k)$ requests can be selected out of i requests in $\binom{i}{i-k}$ ways. The j distinct non-hot MMs can be selected out of $(M-2)$ non-hot MMs in $\binom{M-2}{j}$ ways and the k requests for non-hot MMs can be partitioned into j distinct MMs in $j!S(k,j)$ ways.

For $k < i$, the k requests for non-hot MMs should be to no more than $(B-2)$ distinct non-hot MMs which do not include the particular non-hot MM requested by PE_n. For $k = i$ (i.e. no requests to MM_h) the k requests for non-hot MMs should be to no more than $(B-1)$ distinct non-hot MMs.

Therefore, the probability that PE_n requests a non-hot MM in a bus deficient case and gets accepted is given by

$$P_a(n)|_{BDS\text{-}NH} = \sum_{i=0}^{n}\sum_{k=0}^{i}\binom{n}{i} r^i (1-r)^{n-i}\binom{i}{i-k} p_h^{i-k} p_0^k \times \sum_{j=1}^{\min(k,B-2)} j!S(k,j)\binom{M-2}{j}, \quad k<i,$$
$$P_a(n)|_{BDS\text{-}NH} = \sum_{i=0}^{n}\sum_{k=0}^{i}\binom{n}{i} r^i (1-r)^{n-i}\binom{i}{i-k} p_h^{i-k} p_0^k \times \sum_{j=1}^{\min(k,B-1)} j!S(k,j)\binom{M-2}{j}, \tag{14}$$
$$k = i$$

The expressions in Eq. (14) can be combined for $k \le i$ and can be written as

$$P_a(n)|_{BDS\text{-}NH} = \sum_{i=0}^{n}\sum_{k=0}^{i}\binom{n}{i} r^i (1-r)^{n-i}\binom{i}{i-k} p_h^{i-k} p_0^k \times \sum_{j=1}^{\min(k,B-2+\lfloor k/i \rfloor)} j!S(k,j)\binom{M-2}{j}, \tag{15}$$
$$k \le i$$

Substituting Eqs. (13) and (15) in Eq. (9) and rearranging terms, the probability of acceptance for the nth processor in a bus deficient system is given by

$$P_a(n)|_{BDS} = p_h\left\{\sum_{i=0}^{n}\binom{n}{i}r^i(1-r)^{n-i}p_0^i\sum_{j=1}^{\min(i,B-1)} j!S(i,j)\binom{M-1}{j}\right\} + (1-p_h)\left\{\sum_{i=0}^{n}\binom{n}{i}r^i(1-r)^{n-i}\sum_{k=0}^{i}\binom{i}{i-k}p_h^{i-k}p_0^k \times \sum_{j=1}^{\min(k,B-2+\lfloor k/i\rfloor)} j!S(k,j)\binom{M-2}{j}\right\},$$

$$k \le i, \tag{16}$$

$$P_a(n)|_{BDS} = \sum_{i=0}^{n}\binom{n}{i}r^i(1-r)^{n-i}\left\{p_h p_0^i\sum_{j=1}^{\min(i,B-1)} j!S(i,j)\binom{M-1}{j} + 1 - p_h) \times \sum_{k=0}^{i}\binom{i}{i-k}p_h^{i-k}p_0^k \sum_{j=1}^{\min(k,B-2+\lfloor k/i\rfloor)} j!S(k,j)\binom{M-2}{j}\right\},$$

$$k \le i$$

Note that when evaluating Eq. (16) on a digital computer, 0° should be interpreted as 1. When the upper limit in the summation $\Sigma_{j=1}^{\min(i,B-1)} j!S(i,j)\binom{M-1}{j}$ takes the value i, the summation term can be shown to be simply $(M-1)^i$. This is also true for the other summation on j.

4. Results

An analytical model for performance evaluation of a multiple-bus system operating under hot spot conditions has been developed. In this section, we present analytical results for the bandwidth and the probability of acceptance of prioritized processors of a multiple-bus system with 10 processors, 10 memory modules and a variable number of buses. Later, we validate our analytical model by comparison with simulation results.

In Fig. 2, the bandwidth is plotted against hot spot probability for various processor request rates. The bandwidth is found to decrease with increasing p_h, the reason being the increased contention for MM_h. The degradation is significant for high processor request rates. For a fixed value of p_h, as processor request rate decreases, the bandwidth also decreases, since there are fewer number of processors which are requesting memories in a cycle. For $p_h = 1.0$ all the active processors requests are for MM_h and only one can be accepted, thus the bandwidth approaches unity for high values of r. For small r, e.g. $r = 0.1$, the average bandwidth is less than 1, since during some cycles no processor may be requesting the hot MM. Note that $p_h = 0.1$ corresponds to the URM case.

Fig. 3 shows the variation of bandwidth versus hotspot probabilities for $r = 0.8$ and different number of buses. When $B = 10$ the system behaves like a crossbar and the bandwidth is maximum. When the number of buses is reduced to 8, bandwidth degradation is so small

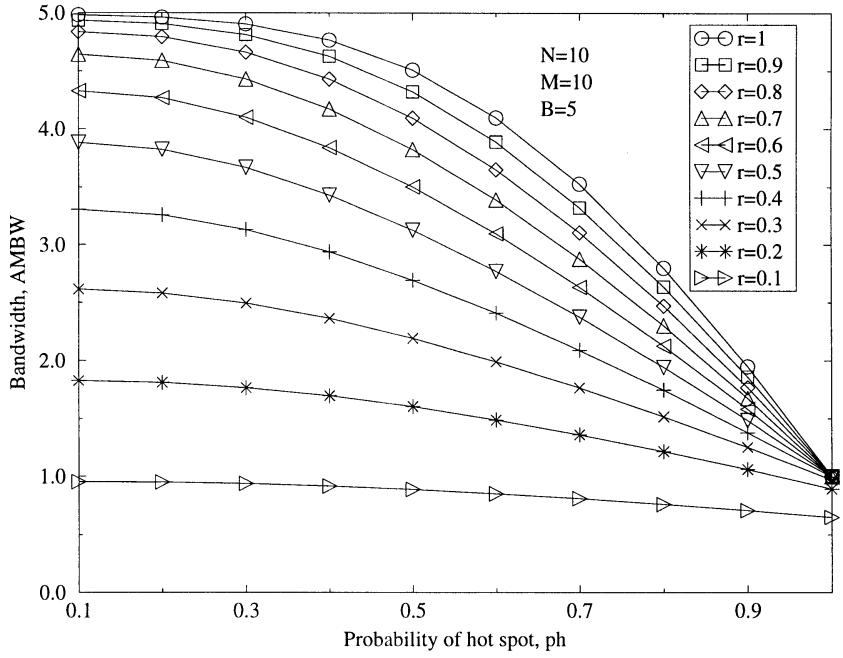


Fig. 2. Bandwidth versus probability of hot spot for different request rates.

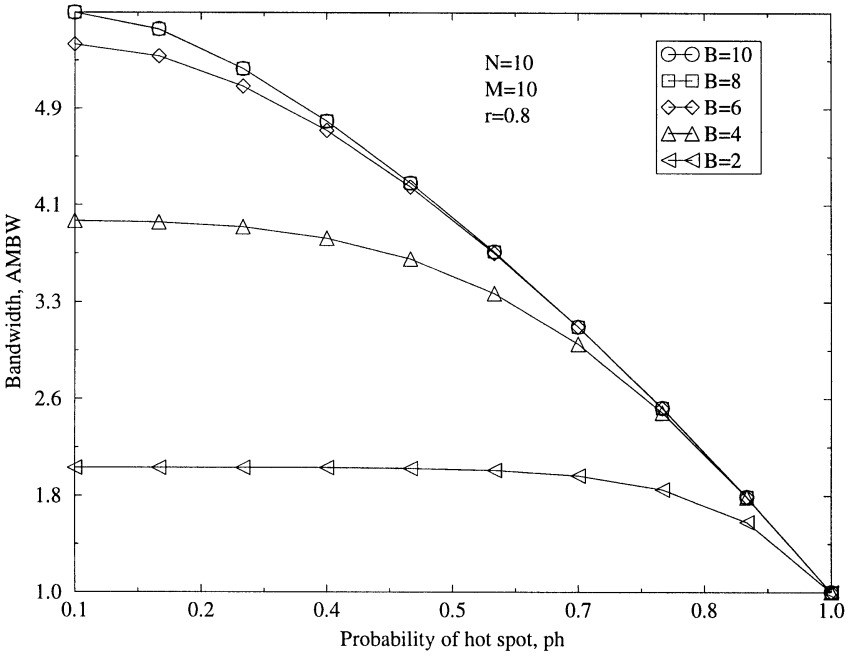


Fig. 3. Bandwidth versus probability of hot spot for different number of buses.

that the curves for $B = 10$ and $B = 8$ overlap in the graph and appears as one curve, although the numerical values differ in the third decimal place. However, when B is reduced to 6 the degradation in bandwidth is significant and when B is further reduced to 4 and 2 the degradation is worse. The reason for this is that no more than 4 or 2 MMs can be active at a time when $B = 4$ or 2 respectively. Thus the bandwidth is never greater than 4 or 2 for $B = 4$ or 2 respectively. Note again that when $p_h \rightarrow 1$, AMBW $\rightarrow 1$. Again $p_h = 0.1$ corresponds to URM case. For a fixed number of buses, the bandwidth decreases with increasing p_h. The degradation is significant when the number of buses is large, because in such cases the degradation is mainly due to memory contention. In systems with fewer number of buses, the contention is mainly for buses, and increased p_h does not have significant effect until p_h becomes very high when memory contention contributes to the degradation.

Fig. 4 is a plot of probability of acceptance of different processors versus the hot spot probability for $B = 5$ and $r = 0.8$. There is an interesting phenomenon to be observed in Fig. 4. Since the processors are in a priority hierarchy, we know that PE_0 has the highest priority and is never blocked. Thus the probability of acceptance for PE_0 is always 1.0. In general, the probability of acceptance of PE_n decreases with increasing n. But, an interesting behavior to be noted is that for large n, i.e. $n = 7$, 8 or 9, the probability of acceptance increases first and then decreases. This behavior can be explained as follows.

As the hotspot probability increases initially, the memory contention among processors increases and lower priority processors stand a better chance of getting a bus. When the hotspot probability is considerably higher, even the low priority processors may request MM_h with a higher frequency and thus the probability of acceptance decreases. This sort of *bumping*

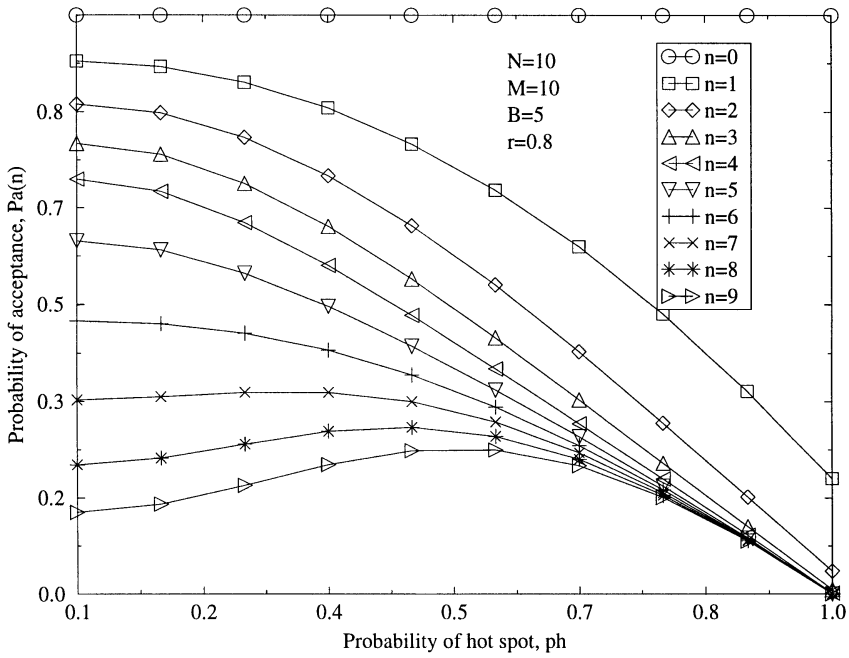


Fig. 4. Probability of acceptance for different processors versus probability of hot spot.

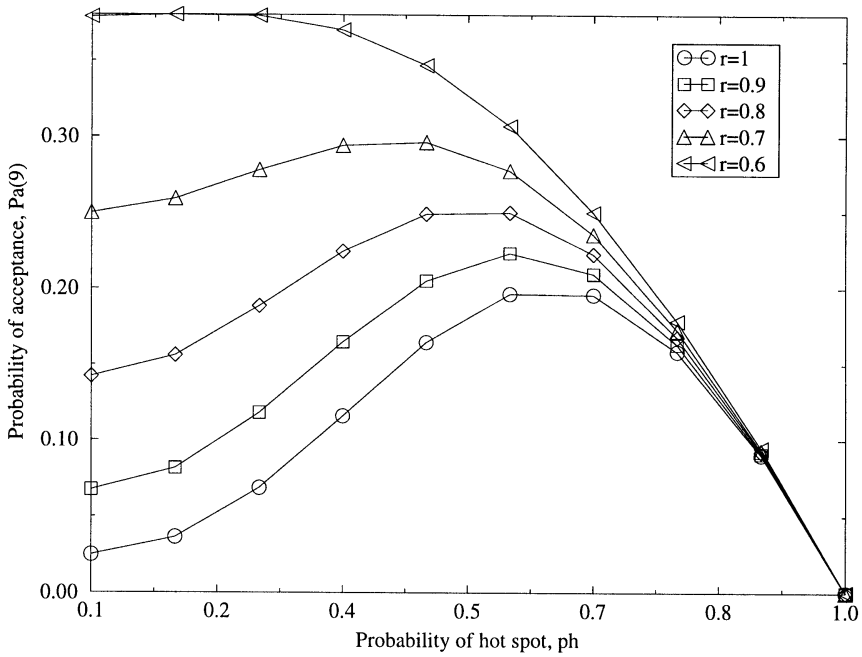


Fig. 5. Probability of acceptance of the lowest priority processor for different request rates.

effect for lower priority processors was found for request rates between 1 and 0.6. With decreasing r, the bump shifts towards the left as can be seen in Fig. 5 for the lowest priority processor element. For low request rates ($r < 0.6$), the bumping effect is not observed.

Fig. 6 shows the probability of acceptance versus request rate r for different processors when $p_h = 0.8$ and $B = 5$. As expected, the probability of acceptance decreases with increasing r. This can be attributed to the fact that when r is low, fewer number of processors request MMs during a cycle and, thus, the chances of a processor being accepted is higher. On the contrary, when r is high, more memory requests are generated by the processors in a cycle, implying more contention and consequently smaller probability of acceptance. It is observed that, at low values of r, the low priority processors are affected more with an increase in r. For high values of r, the degradation is insignificant for lower priority processors since they have already reached the bottom line. This gives rise to exponential behavior of the curves for probability of acceptance of low priority processors. The lower the priority of the processor, the faster its probability of acceptance reaches saturation as is evident by comparing the curves for PE_1 and PE_9.

Fig. 7 shows the probability of acceptance for different processors versus the number of buses for $r = 0.8$ and $p_h = 0.5$. We notice an interesting phenomenon, what we call as the *knee effect*. Again PE_0 is the highest priority processor and is, therefore, always accepted, but PE_1 is accepted approximately 80% of the time. Requests from PE_n, $n \leq B + 1$ encounters only memory contention, and hence the system appears to PE_n as a crossbar system. Therefore, as the number of buses crosses $n - 1$, $P_a(n)$ experiences a sharp rise because of the apparent transition of the system from bus deficient case (multiple-bus) to the bus sufficient case

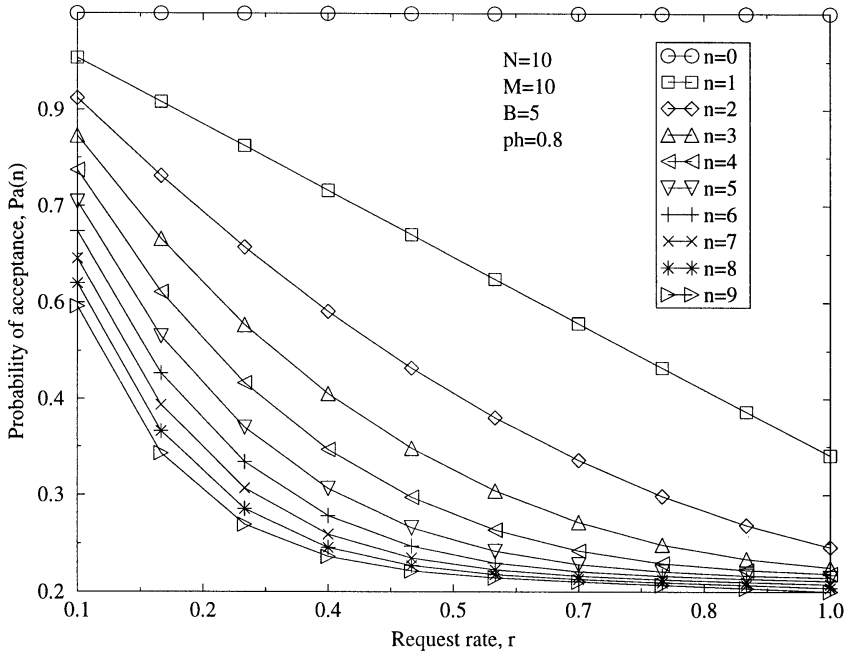


Fig. 6. Probability of acceptance for different processors versus processor request rates.

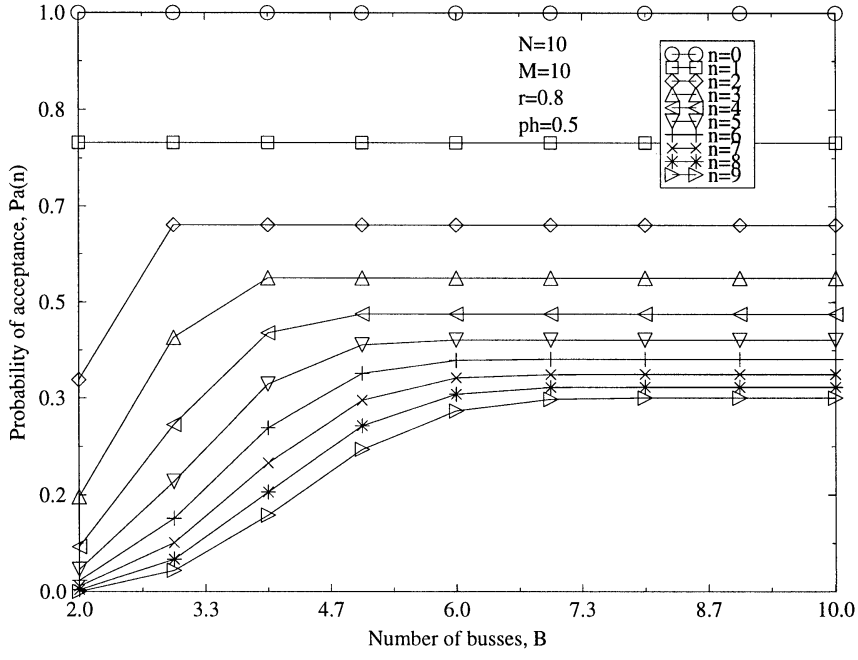


Fig. 7. Probability of acceptance for different processors versus number of buses.

(crossbar) for PE_n. For PE_n, $n > B + 1$ the probability of getting a bus is less due to the number of higher priority processors being greater than the total number of buses. Therefore, for PE_n, there is a rapid change in the probability of acceptance for values between $B = n + 1$ and $B = n + 2$. As an example, for PE_2 there are two other higher priority processors PE_0 and PE_1 and hence its probability of acceptance is small because of it being blocked by bus conflicts in addition to memory conflicts. When B increases to 3 it always has a bus at its disposal and there is no bus conflict resulting in a sharp increase in the probability of acceptance. After that, the probability of acceptance is independent of the number of buses, and hence the curve remains flat. We observe the *knee effect* taking place and this knee shifts towards the right for low priority processors.

Extensive simulations were carried out to validate the analytical expressions derived in the earlier sections. The simulator was driven by a non-uniform random number generator and the number of memories accessed per memory cycle were observed for 50,000 memory cycles, the average of which gave the AMBW. To find out the probability of acceptance for the different processors, the requests generated by the processors were tagged to determine the processors' requests that have been accepted. Analytical results were found to be in close agreement to the simulation results as seen in Tables 1 and 2.

5. Conclusions

Analytical modeling offers an inexpensive and fast method of evaluating the performance of multiprocessor systems. In this paper, we have analyzed the performance of the multiple-bus system under non-uniform memory reference model, in particular, the hot spot memory reference model. For the first time, we have developed analytical expressions for average

Table 1
Average memory bandwidth obtained from analytical expressions and simulations and their percentage errors for $B = 5$

	Average memory bandwidth, AMBW								
	$r = 1$			$r = 0.7$			$r = 0.4$		
p_h	sim	ana	% err	sim	ana	% err	sim	ana	%err
0.1	4.981750	4.981450	0.006021	4.643630	4.642400	0.026500	3.243130	3.259250	–0.494591
0.2	4.965370	4.965640	–0.005435	4.603000	4.587590	0.335905	3.308500	3.305750	0.083193
0.3	4.897380	4.904220	–0.139476	4.431250	4.429220	0.045830	3.139870	3.131910	0.254153
0.4	4.759130	4.764600	–0.114801	4.148500	4.172940	–0.585674	2.949750	2.939630	0.344258
0.5	4.515000	4.506960	0.178390	3.831250	3.823600	0.200071	2.688120	2.695520	–0.274531
0.6	4.088870	4.098140	–0.226193	3.406870	3.388380	0.545684	2.424000	2.410220	0.571737
0.7	3.540750	3.525110	0.443675	2.859250	2.877660	–0.639755	2.105370	2.092110	0.633816
0.8	2.822130	2.800060	0.788195	2.292250	2.303410	–0.484505	1.749500	1.747460	0.116743
0.9	1.929120	1.950970	–1.119961	1.674870	1.675870	–0.059666	1.389000	1.380560	0.611347
1.0	1.000000	1.000000	0.000000	1.000000	0.999990	0.001001	0.992620	0.993950	–0.133811

Table 2
Probability of acceptance for different processors obtained from analytical expressions and simulations and their percentage errors for $p_h = 0.5$

	Probability of acceptance, $P_a(n)$								
	$r = 1$			$r = 0.7$			$r = 0.4$		
Proc. No., n	sim	ana	% err	sim	ana	% err	sim	ana	%err
0	1.000000	1.000000	0.000000	1.000000	1.000000	0.000000	1.000000	1.000000	0.000000
1	0.723370	0.722220	0.159233	0.804230	0.805560	–0.165105	0.890020	0.888890	0.127123
2	0.566880	0.570990	–0.719809	0.660360	0.673120	–1.895656	0.789940	0.798020	–1.012507
3	0.483000	0.483710	–0.146778	0.581200	0.581220	–0.003435	0.718600	0.723400	–0.663536
4	0.426620	0.429060	–0.568686	0.516460	0.515900	0.108545	0.653690	0.661820	–1.228433
5	0.363500	0.365550	–0.560802	0.455000	0.463730	–1.882560	0.601140	0.610440	–1.523490
6	0.310880	0.307970	0.944903	0.416480	0.417580	–0.263423	0.565000	0.566830	–0.322845
7	0.254000	0.255440	–0.563729	0.372870	0.374710	–0.491045	0.527090	0.529000	–0.361053
8	0.210750	0.207380	1.625037	0.331300	0.334290	–0.894435	0.491540	0.495430	–0.785172
9	0.161620	0.164630	–1.828336	0.300470	0.296180	1.448438	0.466030	0.464970	0.227974

memory bandwidth and probability of acceptance of prioritized processors under hot spot conditions. Effects of different parameters like hot spot probability, processor request rate, and the number of buses on bandwidth and probability of acceptance have been presented.

It has been shown that for a constant number of buses, the bandwidth decreases with increasing percentage of requests to hot memory module. Moreover, the degradation is significantly higher for high processor request rates. The reason for the degradation is bus and memory contention. We have shown that for a large number of buses, the degradation is mainly due to memory conflicts. On the contrary, for fewer number of buses, bus contention is the dominating factor contributing to the degradation.

A *bumping effect* has been noticed in the probability of acceptance for the prioritized processors. It has been shown that for a fixed number of buses, the probability of acceptance for high priority processors decreases while that of the low priority processors increases with an increase in the percentage of hot memory requests, until a point is reached after which the probability of acceptance for all the processors decrease. With the number of buses remaining constant and for low values of request rates, the low priority processors are affected more with an increase in the request rate.

The effect of varying the number of buses on the acceptance probability of the different processors has been studied. As the number of buses are increased, keeping the request rate and the hot spot probability constant, the acceptance probability of the processors reach saturation at different values. This give rise to a knee effect resulting in the high priority processors reaching saturation earlier than the low priority ones.

Results obtained from analytical expressions have been found to be in close agreement to the simulation results.

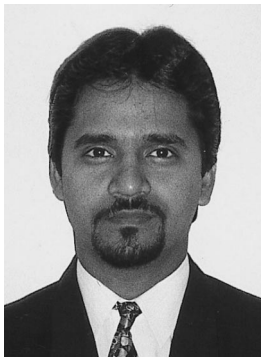
The expressions derived are for multiple bus systems. Expressions for crossbar systems under hot spot conditions follow directly and are obtained by putting the number of buses equal to the minimum of the number of processors and memories. The results can also be extended for

the case of resubmission of blocked requests; this will remove the temporal independence assumption in Section 2.

References

[1] Feng TY. A survey of interconnection networks. Computer 1981;14(12):12–27.
[2] Wu CL, Feng TY (eds.). Tutorial: interconnection networks for parallel and distributed processing. IEEE Computer Society, 1984.
[3] Mudge TN, Hayes JP, Winsor DC. Multiple-bus architectures. Computer 1987;20:42–8.
[4] Liu YC, Jou CJ. Effective memory bandwidth and processor blocking probability in multiple-bus systems. IEEE Trans Comput C 1987;36(6):761–4.
[5] Lang T, Valero M, Alegre I. Bandwidth of crossbar and multiple-bus connections for multiprocessors. IEEE Trans Comput C 1982;31(12):1227–34.
[6] Willis PJ. Derivation and comparison of multiprocessor contention measures. IEE Proc Part E 1978;1(3):93–8.
[7] Bhuyan LN, Agrawal DP. Generalized hypercube and hyperbus structures for a computer network. IEEE Trans Comput C 1984;33(4):323–33.
[8] Yang Q, Bhuyan LN. Analysis of packet-switched multiple-bus multiprocessors. IEEE Trans Comput 1991;40(3):352–7.
[9] Mudge TN, Al-Sadoun HB. A semi-Markov model for the performance of multiple-bus systems. IEEE Trans Comput C 1985;34(10):934–42.
[10] Yen DWL, Patel JH, Davidson ES. Memory interferenee in synchronous multiprocessor systems. IEEE Trans Comput C 1982;31(11):1116–21.
[11] Mudge TN, Hayes JP, Buzzard GD, Winsor DC. Analysis of multiple bus interconnection networks. J Parallel Distrib Comput 1986;3:328–43.
[12] Marsan MA, Gerla M. Markov models for multiprocessor systems. IEEE Trans Comput C 1982;313:239–48.
[13] Fukuda A. Equilibrium point analysis of memory interference in multiprocessor systems. IEEE Trans Comput 1988;37(5):585–93.
[14] Towsley D. Approximate models of multiple bus multiprocessor systems. IEEE Trans Comput C 1986;35(3):220–8.
[15] Das CR, Bhuyan LN. Bandwidth availability of multiple-bus multiprocessors. IEEE Trans Comput C 1985;34(10):918–26.
[16] Bhuyan LN. An analysis of processor-memory interconnection networks. IEEE Trans Comput C 1985;34(3):279–83.
[17] Sethi AS, Deo N. Interference in multiprocessor systems with localized memory access probabilities. IEEE Trans Comput C 1979;28(2):157–63.
[18] Irani KB, Onyuksel IH. A closed-form solution for the performance analysis of multiple-bus multiprocessor systems. IEEE Trans Comput C 1984;33(11):1004–12.
[19] Siomalas KO, Bowen BA. Performance of cross-bar multiprocessor systems. IEEE Trans Comput C 1983;32(7):689–95.
[20] Mudge TN, Makrucki BA. Probabilistic analysis of a crossbar switch. *9th Annual Symposium on Computer Architecture*. April 1982. p. 311–20.
[21] Prasad EV, Sarje AK. Performance evaluation of multiprocessor system modelled as *t*-out-of-*s* system. Int J High Speed Comput 1993;5(1):71–87.
[22] Pfister GF, Norton VA. Hot spot contention and combining in multistage interconnection networks. IEEE Trans Comput C 1985;34(10):943–8.
[23] Atiquzzaman M, Banat MM. Effect of hot-spots on the performance of crossbar multiprocessor systems. Parallel Comput 1993;19(4):455–61.
[24] Atiquzzaman M, Akhtar MS. Effect of non-uniform traffic on the performance of multistage interconnection networks. 9th International Conference on Systems Engineering. Las Vegas, July 14–16, 1993. p. 31–5.

[25] Atiquzzaman M, Akhtar MS. Effect of hot spots on the performance of multistage interconnection networks. FRONTIERS 92: the 4th Symposium on the Frontiers of Massively Parallel Computation. Virginia, October 19–21, 1992. p. 504–5.
[26] Atiquzzaman M, Akhtar MS. Performance of buffered multistage interconnection networks in non-uniform traffic environment. 7th International Parallel Processing Symposium. California, April 13–16, 1993. p. 762–7.
[27] Lee G, Kruskal CP, Kuck DJ. The effectiveness of combining in shared-memory parallel computers in the presence of hot spots. 1986 International Conference on Parallel Processing. August 1986. p. 35–41.
[28] Scott SL, Sohi GS. The use of feedback in multiprocessors and its applications to tree saturation control. IEEE Trans Parallel Distrib Syst 1990;1(4):385–98.
[29] Bogart KP. Introductory combinatorics. 2nd ed. Harcourt Brace Jovanovich, 1990.



Mohammed Azhar Sayeed is working as a Senior Product Manager for ATM products with Cabletron Systems. Prior to working with Cabletron he was at Digital Equipment Corporation as an ATM Technical Marketing Engineer (Aviator). He has represented Digital and now Cabletron as a speaker at major conferences such as Integrated Broadband Networks, Comdex and ATM Year. His interests include ISP/Telco network and service architectures, QoS deployments and Optical Networking. He is a member of the IEEE and actively participates in the IETF and ATM Forum discussions. He has done his Masters in Electrical Engineering from King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia and holds a Bachelors in Electronics and Communication Engineering from Osmania University, Hyderabad, India.

Mohammed Atiquzzaman received the M.Sc. and Ph.D. degrees in electrical engineering and electronics from the University of Manchester Institute of Science and Technology, England in 1984 and 1987, respectively. Currently he is a faculty member in the Department of Electrical and Computer Engineering at University of Dayton, Ohio. He is a senior editor of the IEEE Communications Magazine and member of the editorial board of the Computer Communications journal. He has been guest editor of special issues on ATM Switching and ATM Networks of the International Journal of Computer Systems Science and Engineering, special issue on Projection-based Transforms in the Image and Vision Computing journal. He has also served in the technical program committee of many national and international conferences including IEEE INFOCOM and IEEE Annual Conference on Local Computer Networks. Currently, he is guest editing a special issue on Parallel Computing on Clusters of Workstations in the Parallel Computing journal and a special issue on Enterprise Networking in the Computer Communications journal, special issue on Next Generation Internet in the European Transactions on Telecommunications, feature topics of IEEE Communications Magazine on Traffic Management and Switching for Multimedia and Optical Networks, Communication Systems and Devices. His current research interests are in Broadband ISDN and ATM networks, multiprocessor systems, interconnection networks, parallel processing and image processing. He has over 80 refereed publications in the above areas. He can be contacted at atiq@engr.udayton.edu or atiq@ieee.org