Performance Analysis of Multiple Bus Interconnection Networks with Hierarchical Requesting Model

Wen-Tsuen Chen, *Senior Member, IEEE,* and Jang-Ping Sheu, *Member, IEEE*

Abstract—**In this paper, we study the performance of multiprocessor systems employing multiple buses as the interconnection networks under a nonuniform requesting model, called the hierarchical requesting model. The effective memory bandwidth is chosen as the performance measure. The multiple bus networks investigated include the multiple bus networks with full bus–memory connection, the multiple bus networks with single bus–memory connection, and the multiple bus networks with partial bus–memory connection. In this paper, we also propose one type of the multiple bus networks with partial bus–memory connection, called partial bus networks with K classes. The cost and fault-tolerant capability of each multiple bus network is also evaluated and compared to one another. It can be shown that the partial bus networks with K classes are useful in applications requiring high performance and degree of fault tolerance with moderate cost.**

Index Terms— **Effective memory bandwidth, hierarchical requesting model, interconnection networks, multiple bus, multiprocessor systems, performance analysis.**

I. Introduction

With the advent of VLSI technologies, a great deal of attention has been paid to the design of multiprocessor systems to achieve high levels of computation power. However, the performance of a multiprocessor system depends significantly on the efficiency of its interconnection network. Several interconnection networks have been proposed for the multiprocessor systems, such as the crossbar, single bus, multiple bus, multistage interconnection networks, and others [5].

For a crossbar network, all possible one-to-one simultaneous connections are allowed between the processors and the shared memory modules. Nevertheless, the network cost grows with $O(N^2)$, where N is the number of processors or memory modules. This cost prohibits the crossbar to be used in a large multiprocessor system. Single bus systems are inexpensive and easy to implement but have limited bandwidth. The multistage interconnection networks allow a rich subset of one-to-one simultaneous connections between processors and memory modules, and have a cost function that grows with $O(N \log N)$. However, a major disadvantage of the multistage interconnection networks is that they are not inherently fault tolerant.

The multiple bus networks with the following features become an attractive solution for connecting processors and memory modules in a multiprocessor system [4], [7], [9]. First, they provide a moderate throughput and cost comparing to that of the single bus networks and the crossbars. Second, they allow easy incremental expansion as the number of processors, memory modules, and buses grow. Finally, the multiple bus networks possess fault-tolerant capability. In case a bus fails, the multiprocessor system can still function with other nonfaulty ones.

This paper is concerned with studying the performance of various multiple bus multiprocessor systems containing N processors, M memory modules, and B buses, where the memory modules are shared among all processors and $B \leq \min(M, N)$. One type of $N \times M \times B$ multiple bus networks is shown in Fig. 1. There are N processors, M memory modules, and B buses. Each bus is connected to all N processors and M memory modules. Many performance measures can be used to evaluate a system. Here, we shall use the effective memory bandwidth as a performance metric. The memory bandwidth is defined as the number of successful requests per memory cycle [1]–[3], [6], [7].

In this paper, we propose an architecture of $N \times M \times B$ multiple bus networks, called partial bus networks with K classes. In this architecture, each processor is connected to all buses. However, each memory module is connected only to a subset of buses. It is more flexible and less costly than that of multiple bus networks with full connection of each processor and memory module to all buses. Under a nonuniform memory reference model, called hierarchical requesting model, the performance of the proposed $N \times M \times B$ networks and other earlier proposed ones is analyzed and compared to one another. The cost and fault tolerance of these networks are also evaluated.

In the next section, we describe various types of multiple bus networks and evaluate their cost and fault-tolerant capabilities. In Section III, we analyze the performance of various multiple bus networks under the hierarchical requesting model. Some numerical results are presented in Section IV. Finally, we conclude this paper in Section V.

Manuscript received March 7, 1988; revised January 3, 1991. This work was supported by the National Science Council of the Republic of China under Grants NSC76-0408-E007-005 and NSC76-0408-E008-06.

W.-T. Chen is with the Department of Computer Science, National Tsing Hua University, Hsinchu 30043, Taiwan, R.O.C.

J.-P. Sheu is with the Department of Electrical Engineering, National Central University, Chung-Li 32054, Taiwan, R.O.C.

IEEE Log Number 9100995.

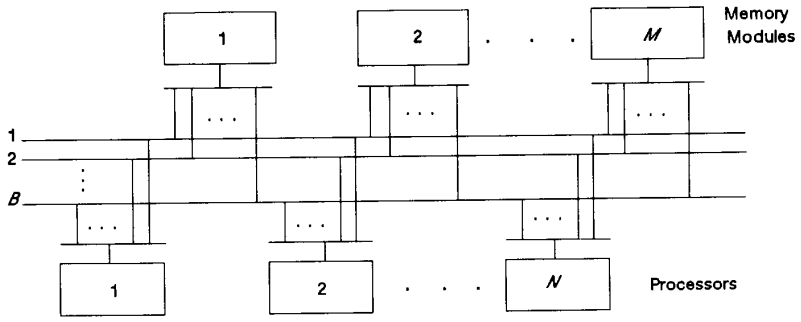


Fig. 1. An $N \times M \times B$ multiple bus network.

II. The Multiple Bus Networks and Their Cost

In this section, we first define various types of multiple bus networks, then evaluate their cost and fault-tolerant capabilities.

A. Multiple Bus Networks

Performance analyses of the multiple bus networks have appeared recently in several papers [4], [6], [7], [8], [11]–[13]. Most of the authors focus their attention on the multiple bus network with full connection of each processor and memory module to all the buses. Such a multiple bus network is too costly for large multiprocessor systems. Lang *et al.* [9] proposed a less costly type of multiple bus network, called partial bus networks. In an $N \times M \times B$ partial bus architecture, the shared memory modules and buses are divided into g groups. All the processors are connected to all the buses, whereas each group of M/g memory modules is connected to a set of B/g buses. Fig. 2 shows a partial bus network with $g = 2$. The memory modules are divided into two groups. All the processors are connected to all the buses, whereas each group of $M/2$ memory modules is connected to a group of $B/2$ buses.

Here, we propose another architecture of $N \times M \times B$ multiple bus networks, called partial bus networks with K classes. In this type of network, there are K classes of memory modules, where $K \leq B$. The memory modules in class C_K are connected to B buses from bus 1 to bus B, memory modules in C_{K-1} are connected to $B-1$ buses from bus 1 to bus $B-1$. In general, memory modules in class C_j are connected to $j + B - K$ buses from bus 1 to bus $j + B - K$, for $1 \leq j \leq K$. A $3 \times 6 \times 4$ partial bus network with three classes is shown in Fig. 3.

With our proposed networks, we can have the following two principles for the memory modules being connected to the buses in order to enhance system fault tolerance and performance. One is that the memory modules which need higher fault tolerance of buses failure are connected to more number of buses than those which need lower fault tolerance of buses failure. The other is that the memory modules which are more frequently referenced are connected to more number of buses than those which are less frequently referenced. As will be clear in the following sections, the performance and cost of the partial bus networks with K classes are close to the partial bus networks with $g = 2$. However, the fault tolerance of the former is more flexible than that of the latter.

For ease of description, we give the following definitions for various types of multiple bus networks. A multiple bus network with full bus–memory connection is one with each processor and each memory module connected to all buses. Fig. 1 shows a multiple bus with full bus–memory connection. A multiple bus network with single bus–memory connection is one with each processor connected to all buses, but each memory module is only connected to a single bus as shown in Fig. 4. A multiple bus network with partial bus–memory connection means that each processor of the network is connected to all buses, but each of its memory modules can be connected to a subset of buses. The partial bus networks proposed in [9] and the partial bus networks with K classes are two examples of the multiple bus networks with partial bus–memory connection.

In the above $N \times M \times B$ multiple bus networks, two types of requesting conflicts can occur. One type of conflict arises when more than one processor attempts to access the same idle memory module simultaneously, or a referenced memory module might be busy at the requesting time. This is called memory contention or memory interference. The other type of conflict arises when one or more processors attempt to access an idle memory module but no buses are available. This is called bus contention or bus interference. A two-stage arbitration scheme proposed by Lang *et al.* [9] can be used to resolve memory and bus contentions.

In the first stage, the memory contention is resolved by M arbiters of the N-users, 1 server type. Each arbiter is associated with one memory module. Each of these arbiters selects with equal probability one of the processors which have outstanding requests to the arbiter's associated memory module. In the second stage, bus contention is resolved by a B-out-of-M arbiter which assigns the buses to the memory requests selected in the first stage. That is, at most B memory modules can be requested simultaneously in each memory cycle. The buses are assigned in a round-robin fashion to the memory modules that are requested by the processors. In the next section, we shall analyze the performance of the $N \times M \times B$ multiple bus networks under the two-stage arbitration scheme.

B. Cost Analysis

The cost and fault tolerance of various types of multiple bus networks will be evaluated in the following. The cost

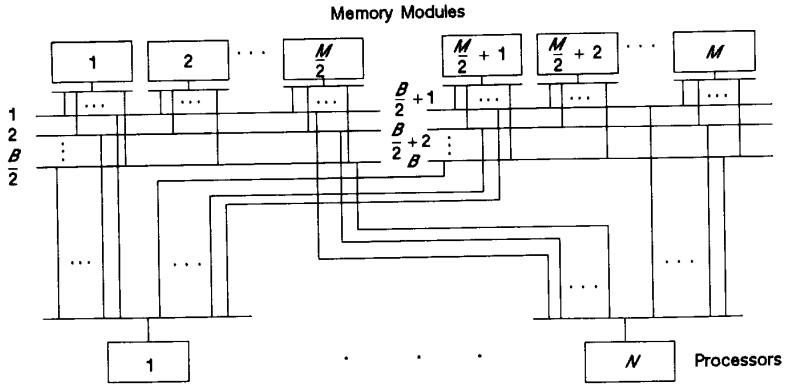


Fig. 2. An $N \times M \times B$ partial bus network with $g = 2$.

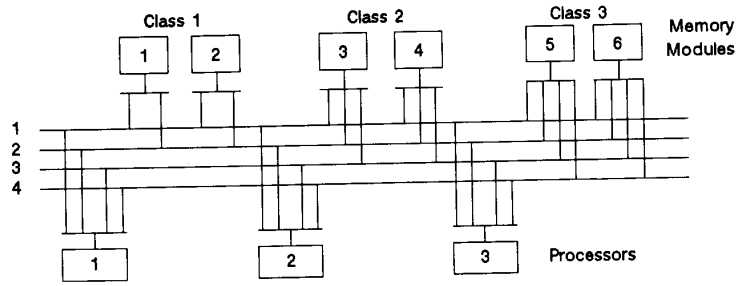


Fig. 3. A $3 \times 6 \times 4$ partial bus network with three classes.

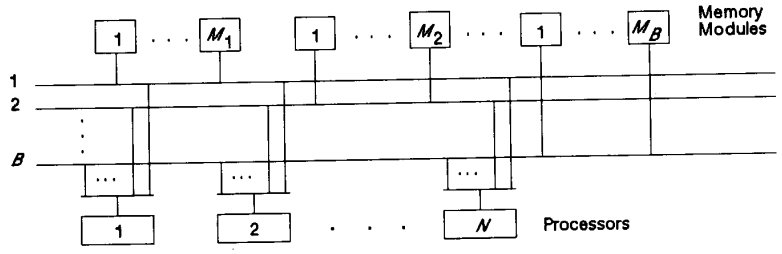


Fig. 4. An $N \times M \times B$ network with single bus–memory connection.

of the multiple bus networks is measured by the number of connections and the load of each bus. It is obvious that the cost of a multiple bus network is proportional to the number of connections in the network. The capacitive loads and drive requirements of a bus are proportional to the number of connections on the bus. The number of connections of the $N \times M \times B$ multiple bus with full bus–memory connection is equal to $B(N + M)$. The load of each bus is proportional to $M + N$. It is fault tolerant because each processor can still access to all the memory modules as long as there exists at least one nonfaulty bus and hence the degree of fault tolerance is $B - 1$.

The number of connections of the $N \times M \times B$ multiple bus network with single bus–memory connection is equal to $BN + M$. The load of bus i, for $1 \leq i \leq B$, is proportional to $N + M_i$, where M_i is the number of memory modules connected to bus i. The single bus–memory connection scheme is less fault tolerant than the full bus–memory connection one because a failure in one of the buses, say bus i, results in inaccessibility of M_i memory modules which are connected to bus i. The number of connections in the $N \times M \times B$ partial bus network is equal to $B(N + M/g)$, where g is the number of groups. The load of each bus is proportional to $N + M/g$. The degree of fault tolerance of this network is $B/g - 1$.

The number of connections of the $N \times M \times B$ partial bus network with K classes is equal to

$$BN + \sum_{j=1}^{K} M_j(j + B - K)$$

where M_j is the number of memory modules in class C_j, for $1 \leq j \leq K$. According to the connection scheme of this type of network, bus B is connected by the memory modules in class C_K and bus $B - 1$ is connected by the memory

modules in class C_K and class C_{K-1}. In general, bus i is connected by the memory modules which belong to classes $C_K, C_{K-1}, \cdots, C_{\max(i+K-B,1)}$. Thus, the load of each bus i is proportional to

$$N + \sum_{j=\max(i+K-B,1)}^{K} M_j, \qquad \text{for } 1 \le i \le B.$$

Because the memory modules in class C_1 are only connected to $B-K+1$ buses, the degree of fault tolerant of this network is equal to $B-K$. However, accesses to the memory modules in class C_j are more fault tolerant than those to class C_{j-1}, for $j \le K$.

The cost and fault-tolerant capability of the above networks are summarized in Table I. In view of the results of Table I we conclude that the cost and fault-tolerant capability of networks with partial bus–memory connection scheme are intermediate between the networks with full bus–memory connection and the networks with single bus–memory connection. In Section IV, the performance–cost ratios of various types of multiple bus networks will be compared to one another.

TABLE I
THE COST AND FAULT TOLERANCE OF VARIOUS MULTIPLE BUS NETWORKS

Connection schemes	No. of connections	Load of each bus i	Degree of Fault-Tolerance
Multiple bus with full bus–memory connection	$B(N+M)$	$N+M$	$B-1$
Multiple bus with single bus–memory connection	$BN+M$	$N+M_i$	0
Partial bus network	$B(N+M/g)$	$N+M/g$	$B/g-1$
Partial bus network with K classes	$BN+\sum_{j=1}^{K} M_j\,(j+B-K)$	$N+\sum_{j=\max(i+K-B,1)}^{K} M_j$	$B-K$

III. PERFORMANCE ANALYSIS

Most of the previous performance analyses of the multiple bus networks are based on the assumption that a processor addresses any one of the shared memory modules with the same probability. In Das and Bhuyan [4], they assumed that each processor is likely to address a particular memory module more frequently than others. The equally likely requesting case is a special case of the Das's model. In this paper, a general memory reference model, called hierarchical requesting model, is proposed. Under this model, the performance of various bus–memory connection schemes of the $N \times M \times B$ multiple bus networks is analyzed. In the following, we shall describe the hierarchical requesting model.

A. The Hierarchical Requesting Model

In the multiprocessing environment, a job to be run on the system usually consists of a set of communicating tasks. To execute these tasks efficiently, the system should be organized in such a way that communication overhead among these tasks is minimized. The task assignment procedure should assign those tasks that have large amount of communications to the same processor or to a cluster of processors with low communication cost. It leads to that the traffic between a processor and other processors belonging to the same cluster is higher than that with those processors belonging to other clusters.

To model such a system, a hierarchical requesting model is proposed [3]. We assume that a cluster of processors have a cluster of memory modules as their favorite memory modules. These memory modules may be used for storing the tasks assigned for these processors. Besides, the relations of the processors with their favorite memory modules can be classified into n-level hierarchy. Each processor has different fractions of requests to the memory modules belonging to different levels of subclusters. In the following, we shall describe the n-level hierarchy for the $N \times N \times B$ multiple bus networks and $N \times M \times B$ multiple bus networks.

For an $N \times N \times B$ multiple bus network, we assume that $N = k_1 k_2 \cdots k_n$. Each processor P_i has a memory module MM_i as its favorite memory module, for $1 \le i \le N$. These processors and memory modules are organized into an n-level hierarchy. First, the N pairs of processors and memory modules are partitioned into k_1 clusters in the first level, each cluster contains $k_2 k_3 \cdots k_n$ pairs of processor–memory. In the second level, each of k_1 clusters is partitioned into k_2 subclusters with equal size, and so on. Finally, each subcluster in the $(n-1)$th level contains k_n pairs of processor–memory. Note that, in the nth level each subcluster only contains one processor and its favorite memory module. For example, with a three-level hierarchy, an $N \times N \times B$ network, where $N = k_1 k_2 k_3$, can be partitioned into k_1 clusters, and each cluster can be partitioned into k_2 subclusters again and each subcluster contains k_3 pairs of processor and its favorite memory module.

With an n-level hierarchy, there are $n+1$ different requesting rates for a processor accessing to the memory modules in different subclusters. On the other hand, each memory module is requested by different processors with different requesting rates. Each processor has $n+1$ types of requests: namely, requests to its favorite memory module in the nth level with fraction m_0, requests to each of the other memory modules in the same subcluster of the $(n-1)$th level except its favorite memory module with fraction m_1, requests to each of the memory modules in the same subcluster of the $(n-2)$th level excluding the memory modules in the $(n-1)$th level with fraction m_2, requests to each of the memory modules in the same subcluster of the $(n-3)$th level excluding the memory modules in the $(n-2)$th level with fraction m_3, and so on. We assume that the 0th level includes the whole network.

For example, with a two-level hierarchy, a multiprocessor system can be partitioned into k_1 clusters, and each cluster contains k_2 pairs of processor and its favorite memory module, when $N = k_1 k_2$. Then a processor in a particular cluster C has three types of requests: namely, requests to its favorite memory module with fraction m_0, requests to each of other memory modules in cluster C with fraction m_1, and requests to each of the memory modules in other clusters with fraction m_2.

In general, the fraction of a processor requesting connection to its favorite memory module is higher than that of requesting connection to nonfavorite ones. Furthermore, the fraction of requests to a memory module within the same subcluster is higher than that of requests to a memory module in other subclusters. That is, $m_0 > m_1 > \cdots > m_n$. Let N_i be the number of processors or memory modules belonging to the same subcluster in the $(n-i)$th level, excluding those in the $(n-i+1)$th level, for $0 \le i \le n$. Then N_i's can be computed as follows

$$N_i = (k_{n-i+1} - 1)k_{n-i+2} \cdots k_{n-1}k_n, \qquad \text{for } 1 \le i \le n,$$
$$\text{and} \quad \sum_{i=0}^{n} m_i N_i = 1, \qquad \text{where } N_0 = 1. \tag{1}$$

For example, with a three-level hierarchy, let $N = k_1k_2k_3$. From formula (1), we have $N_0 = 1$, $N_1 = k_3 - 1$, $N_2 = (k_2 - 1)k_3$, and $N_3 = (k_1 - 1)k_2k_3$.

In the following, let us consider an $N \times M \times B$ multiple bus network, N and M are restricted to $N = k_1k_2 \cdots k_{n-1}k_n$ and $M = k_1k_2 \cdots k_{n-1}k'_n$, respectively. The partitioning way of the $N \times M \times B$ networks is the same as that of the $N \times N \times B$ networks. However, each subcluster in the $(n-1)$th level of the $N \times M \times B$ networks contains k_n processors and k'_n memory modules. The k'_n memory modules in the $(n-1)$th level are used as the favorite memory modules of the k_n processors with the same subcluster. We assume that each processor with equal probability requests to any one of its favorite memory modules.

For example, with a three-level hierarchy, an $N \times M \times B$ network, where $N = k_1k_2k_3$ and $M = k_1k_2k'_3$, can be partitioned into k_1 clusters, and each cluster can be partitioned into k_2 subclusters again. Finally, each subcluster contains k_3 processors and k'_3 favorite memory modules. Then a processor in a particular subcluster has three types of requests: namely, requests to each of its favorite memory modules with fraction m_0, requests to each of the memory modules in other subclusters, which belong to the same cluster, with fraction m_1, and requests to each of the memory modules in other clusters with fraction m_2. With an n-level hierarchy, there are n different requesting rates for a processor accessing to the memory modules in different subclusters. In the following, we only consider the case of $N \times N \times B$ multiple bus networks in the performance analysis. The performance of the $N \times M \times B$ networks can be obtained similarly from the formulas derived in the case of $N \times N \times B$ networks.

Based on the hierarchical requesting model, the performance of various types of $N \times N \times B$ multiple bus networks is analyzed with the following assumptions [4].

1) The multiple bus networks operate in a synchronous mode. The requests of all processors are issued at the same time and each processor has an identical memory cycle time.

2) Each processor P_i generates random and independent requests.

3) At the beginning of every memory cycle, each processor generates a new request with probability r. Thus, r is also the average number of requests generated per memory cycle by each processor.

4) The propagation delays and arbitration times associated with the multiple bus network are included in the memory cycle time.

5) The requests which are blocked (not accepted) are ignored. That is, the requests issued at the next cycle are independent of the previous cycle.

B. The Multiple Bus Networks with Full Bus–Memory Connection

The performance of the multiple bus networks with various bus–memory connection schemes can be analyzed by considering memory interference and bus interference in the networks. First, we consider the memory interference. Let X be the probability that there is at least one request for a particular memory module MM_j. Let p_0 be the probability of a processor P_j requesting a connection to its favorite memory module MM_j. Let p_i be the probability that at least one request is generated by those processors which request connection to memory module MM_j with fraction m_i, for $1 \le i \le n$. The number of processors with fraction m_i requesting connection to MM_j is equal to N_i as given in formula (1). It follows that

$$p_i = 1 - (1 - rm_i)^{N_i}.$$

Hence, the probability of at least one processor requesting connection to MM_j is

$$\begin{aligned} X &= 1 - (1 - p_0)(1 - p_1) \cdots (1 - p_n) \\ &= 1 - (1 - rm_0)(1 - rm_1)^{N_1} \cdots (1 - rm_n)^{N_n}. \end{aligned} \tag{2}$$

Second, we consider the bus interference. The multiple bus network with B buses can allow at most B requests per memory cycle. With the probability X of at least one processor requesting connection to a memory module given by (2), the probability that exactly i of the N memory-request arbiters output a memory request is given by

$$Pf(i) = \binom{N}{i} X^i (1 - X)^{N-i}. \tag{3}$$

The network gets saturated when more than B requests are generated and allows only B processor–memory connections simultaneously. As a result, the memory bandwidth MBW_f of the multiple bus networks with full bus–memory connection is given by

$$\begin{aligned} \mathrm{MBW}_f &= \sum_{i=1}^{B} iPf(i) + \sum_{i=B+1}^{N} BPf(i) \\ &= \sum_{i=1}^{N} iPf(i) - \sum_{i=B+1}^{N} (i - B)Pf(i) \\ &= NX - \sum_{i=B+1}^{N} (i - B)Pf(i). \end{aligned} \tag{4}$$

The memory bandwidth MBW's of the multiple bus networks with single bus–memory connection can be derived as follows. Let M_i be the number of memory modules connected

to the bus i, for $1 \le i \le B$. Then the probability that there is at least one memory service in bus i is given by

$$Y_i = 1 - (1 - X)M_i \tag{5}$$

where X is the probability that there is at least one processor requesting to a particular memory module and is given by (2). Thus, the memory bandwidth of the network is expressed as

$$\text{MBW's} = \sum_{i=1}^{B} Y_i. \tag{6}$$

C. The Partial Bus Networks

In this subsection, we shall derive the performance of the partial bus networks under the hierarchical requesting model. In fact, the memory interference analysis for the partial bus networks is the same as before, since it is independent of the bus configuration. However, the bus interference analysis needs some modification.

Assume that an $N \times N \times B$ partial bus network is partitioned into g equal groups. Each of N/g memory modules is connected to all B/g buses. The equation resulted from the memory interference is the same as (2). Consider the bus interference in each group of buses. Equation (3) can be rewritten as follows

$$Pg(i) = \binom{N/g}{i} X^i (1 - X)^{N/g - i}. \tag{7}$$

Because each group of B/g buses allows at most B/g requests, the memory bandwidth of each subnetwork with B/g buses and N/g memory modules is given by

$$\begin{aligned} \text{MBW}g &= \sum_{i=1}^{B/g} iPg(i) + \sum_{i=B/g+1}^{N/g} B/gPg(i) \\ &= \sum_{i=1}^{N/g} iPg(i) - \sum_{i=B/g+1}^{N/g} (i - B/g)Pg(i) \\ &= (N/g)X - \sum_{i=B/g+1}^{N/g} (i - B/g)Pg(i). \end{aligned} \tag{8}$$

The memory bandwidth MBWp of the partial bus networks can be obtained from the summation of g groups of subnetworks. Hence, we obtain

$$\text{MBW}p = g(\text{MBW}g) = NX - \sum_{i=B/g+1}^{N/g} (gi - B)Pg(i). \tag{9}$$

If $g = 1$, then (9) is equal to (4).

D. The Partial Bus Networks with K Classes

In this subsection, we shall evaluate the memory bandwidth of the partial bus networks with K classes. Before presenting the memory bandwidth of the $N \times N \times B$ partial bus networks with K classes, we give a simple and fair bus-assignment procedure for assigning the requested memory modules in each class C_j to their $j + B - K$ connected buses. The bus-assignment procedure can be divided into two steps that are described in [10].

In the first step, it concerns to select the requested memory modules from each class C_j and assign them to the $j + B - K$ connected buses. For each class C_j, $\min(j + B - K, R)$ memory modules from the R memory modules which have at least one request are selected. The selected memory modules in class C_j are assigned to the buses from bus $j+B-K$ to bus $j+B-K-\min(j+B-K,R)+1$. For example, let $B = 4$ and $K = 3$. The memory modules in class C_2 are connected to buses 3, 2, and 1. If there are two requested memory modules selected from class C_2, then the buses 3 and 2 will be assigned to the selected memory modules. After the first step, a bus i may be requested by several memory-request arbiters from different classes. In the second step, each bus arbiter makes assignment in a random selection or cyclic fashion, i.e., on a round-robin basis. Based on the bus-assignment procedure, the memory bandwidth of the network can be derived by the following method.

Assume that each class C_j contains M_j memory modules, for $1 \le j \le K$. In the part of memory interference, let X be the probability that there is at least one request for a particular memory module and X is derived from (2). In the part of bus interference, let Y_i be the probability that there is at least one memory-request arbiter output a request to bus i. Then the memory bandwidth of the partial bus network with K classes is equal to

$$\text{MBW}p' = \sum_{i=1}^{B} Y_i.$$

The formulas of Y_i's are derived as follows. Given the X, the probability that exactly m memory services are requested to the memory modules in class C_j is given by

$$Q_j(m) = \binom{Mj}{m} X^m (1 - X)^{Mj - m}, \qquad \text{for } 1 \le j \le K. \tag{10}$$

From the connection scheme of the network, the bus i is connected by the memory modules which belong to classes C_K, $C_{K-1}, \cdots, C_{\max(i+K-B,1)}$. According to the bus-assignment procedure, the bus B will be requested if there is at least one memory service in the class C_K. Thus,

$$Y_B = 1 - Q_K(0).$$

The case that bus $B - 1$ is not requested is given by the conditions of no memory service in class C_{K-1} and at most one memory service in class C_K. That is,

$$Y_{B-1} = 1 - Q_{K-1}(0)(Q_K(0) + Q_K(1)).$$

In general, the case that bus i is not requested is given by the conditions of no memory service in class C_{i+K-B}, at most one memory service in class $C_{i+K-B+1}$, at most two memory services in class $C_{i+K-B+2} \cdots$, and at most $B - i$ memory services in class C_K. Notice that, we assume that if the subscript d of a class C_d is smaller than one, then this

class is a dummy (empty) class. Then (10) for the dummy class C_d is

$$Q_d(0) = 1, \quad \text{and}$$
$$Q_d(m) = 0 \quad \text{for } m > 0, \quad \text{where } d \leq 0.$$

Let $a = i + K - B$, then

$$\begin{aligned} Y_i &= 1 - Q_a(0)\,(Q_{a+1}(0) + Q_{a+1}(1)) \\ &\quad \cdots (Q_K(0) + \cdots + Q_K(B-i)) \\ &= 1 - \prod_{j=a}^{K} \sum_{m=0}^{j-a} Q_j(m) \end{aligned} \tag{11}$$

where $Q_j(m)$ is equal to zero if $m > M_j$. Hence, the memory bandwidth of the partial bus network with K classes is

$$\begin{aligned} \text{MBW}p' &= \sum_{i=1}^{B} \left(1 - \prod_{j=a}^{K} \sum_{m=0}^{j-a} Q_j(m)\right) \\ &= B - \sum_{i=1}^{B} \prod_{j=a}^{K} \sum_{m=0}^{j-a} Q_j(m), \\ &\quad \text{where } a = i + K - B. \end{aligned} \tag{12}$$

IV. Numerical Results

In this section, we give some numerical results obtained from our analyses for the $N \times N \times B$ multiple bus networks with various bus–memory connection schemes. The results are evaluated under the two-level hierarchy and uniform requesting model for $r = 1.0$ and 0.5. In the uniform requesting case, each processor requests connection to all the memory modules with equal probability. In the two-level hierarchy, we assume that an $N \times N \times B$ network is partitioned into four clusters, and each cluster contains $N/4$ processors and memory modules. Each processor is with probability 0.6 addressing to its favorite memory module, probability 0.3 addressing to other memory modules within the same cluster, and probability 0.1 addressing to the memory modules in other clusters.

Table II and Table III list the memory bandwidth of the $N \times N \times B$ networks with full bus–memory connection for various values of N and B. The results show that the memory bandwidth in the two-level hierarchy is higher than that in the uniform requesting model for various values of N and B. Under the two-level hierarchy with $r = 1.0$, a processor requests its favorite memory module most of the time, thereby reducing the memory access conflicts. The memory bandwidth of the system is then most affected by the number of buses. If a processor generates a request in every cycle, then the network should have at least $N/2$ buses to provide comparable performance with that of the network with N buses or a crossbar network. However, for $r = 0.5$, Table III shows that the network with $B = N/2$ buses performs close to that of network with $B = N$ buses. This indicates that when r is less than 1, the network with large number of buses is underutilized. The results indicate that the number of buses for the networks should be determined by taking both requesting rate r and requesting pattern into consideration.

TABLE II
Memory Bandwidth of $N \times N \times B$ Networks with Full Bus–Memory Connection for $r = 1.0$

No of Buses	$N = 8$		$N = 12$		$N = 16$	
	Hier.	Unif.	Hier.	Unif.	Hier.	Unif.
1	1.0	1.0	1.0	1.0	1.0	1.0
2	2.0	2.0	2.0	2.0	2.0	2.0
3	3.0	2.97	3.0	3.0	3.0	3.0
4	3.97	3.87	4.0	3.99	4.0	4.0
5	4.85	4.59	5.0	4.97	5.0	5.0
6	5.52	5.04	5.98	5.88	6.0	6.0
7	5.88	5.22	6.91	6.66	7.0	6.97
8	5.98	5.25	7.73	7.24	7.99	7.89
9			8.34	7.58	8.95	8.72
10			8.70	7.73	9.85	9.39
11			8.84	7.77	10.62	9.86
12			8.86	7.78	11.20	10.13
13					11.56	10.25
14					11.72	10.29
15					11.77	10.30
16					11.78	10.30
$N \times N$ Crossbar	5.98	5.25	8.86	7.78	11.78	10.30

TABLE III
Memory Bandwidth of $N \times N \times B$ Networks with Full Bus–Memory Connection for $r = 0.5$

No. of Buses	$N = 8$		$N = 12$		$N = 16$	
	Hier.	Unif.	Hier.	Unif.	Hier.	Unif.
1	0.99	0.98	1.0	1.0	1.0	1.0
2	1.91	1.88	1.99	1.98	2.0	2.0
3	2.67	2.57	2.93	2.89	2.99	2.98
4	3.15	2.99	3.76	3.67	3.95	3.91
5	3.38	3.16	4.41	4.23	4.83	4.74
6	3.46	3.22	4.83	4.57	5.59	5.41
7	3.47	3.23	5.04	4.72	6.15	5.87
8	3.47	3.23	5.13	4.78	6.52	6.15
9			5.16	4.80	6.73	6.29
10			5.16	4.80	6.82	6.35
11			5.16	4.80	6.85	6.37
12			5.16	4.80	6.86	6.37
13					6.87	6.37
14					6.87	6.37
15					6.87	6.37
16					6.87	6.37
$N \times N$ Crossbar	3.47	3.23	5.16	4.80	6.87	6.37

Table IV lists the memory bandwidth of the multiple bus networks with single bus–memory connection. The memory bandwidth of the network is evaluated under the case that N memory modules are distributed over the B buses. That is, each bus is connected by N/B memory modules. Consequently, the load of each bus is proportional to $N + N/B$. In the uniform requesting model, if $r = 1.0$, the ratio of the memory bandwidth of the network with N buses to that of the network with $N/2$ buses is nearly 1.5. If $r = 0.5$, the ratio is reduced to 1.2. In the two-level hierarchical requesting model, the ratio of the memory bandwidth of the network with N buses to that of the network with $N/2$ buses is almost 1.6 when $r = 1.0$. However, in the case of $r = 0.5$, the ratio is reduced to 1.28. These results show that the memory bandwidth of the network is very much dependent on the bus configuration as requesting rate $r = 1.0$. Notice that, the memory bandwidth of the single bus–memory connection network is also equal to that of $N \times N$ crossbar when $B = N$.

TABLE IV
MEMORY BANDWIDTH OF $N \times N \times B$ NETWORKS WITH SINGLE BUS–MEMORY CONNECTION

$r = 1.0$						
No. of Buses	$N = 8$		$N = 16$		$N = 32$	
	Hier.	Unif.	Hier.	Unif.	Hier.	Unif.
1	1.0	1.0	1.0	1.0	1.0	1.0
2	1.99	1.97	2.0	2.0	2.0	2.0
4	3.74	3.53	3.98	3.94	4.0	4.0
8	5.97	5.25	7.44	6.99	7.96	7.86
16			11.78	10.30	14.87	13.90
32					23.48	20.41

(a)

$r = 0.5$						
No. of Buses	$N = 8$		$N = 16$		$N = 32$	
	Hier.	Unif	Hier.	Unif.	Hier.	Unif.
1	0.99	0.98	1.0	1.0	1.0	1.0
2	1.79	1.75	1.98	1.97	2.0	2.0
4	2.72	2.58	3.58	3.48	3.95	3.93
8	3.47	3.23	5.39	5.10	7.14	6.93
16			6.87	6.37	10.76	10.16
32					13.69	12.67

(b)

TABLE V
MEMORY BANDWIDTH OF $N \times N \times B$ PARTIAL BUS NETWORKS WITH $g = 2$

$r = 1.0$						
No. of Buses	$N = 8$		$N = 16$		$N = 32$	
	Hier.	Unif.	Hier.	Unif.	Hier.	Unif.
2	1.99	1.97	2.0	2.0	2.0	2.0
4	3.89	3.73	4.0	3.99	4.0	4.0
8	5.97	5.25	7.92	7.71	8.0	8.0
16			11.78	10.30	15.97	15.76
32					23.48	20.41

(a)

$r = 0.5$						
No. of Buses	$N = 8$		$N = 16$		$N = 32$	
	Hier.	Unif.	Hier.	Unif.	Hier.	Unif.
2	1.79	1.75	1.98	1.97	2.0	2.0
4	2.96	2.81	3.82	3.75	4.0	3.99
8	3.47	3.23	6.25	5.92	7.89	7.81
16			6.87	6.37	13.02	12.24
32					13.69	12.67

(b)

Table V lists the memory bandwidth of the partial bus networks with $g = 2$. The load of each bus is proportional to $N + N/2$. The results show that the memory bandwidth with the two-level hierarchical requesting model is higher than that with the uniform requesting model for various values of N and B. It also shows that the memory bandwidth of the network for $r = 1.0$ is more dependent on bus configuration than that of the network for $r = 0.5$. Comparison of the results in Table IV and Table V shows that the memory bandwidth of the partial bus networks with $g = 2$ is higher than that of the networks with single bus–memory connection scheme. However, the cost of the networks with single bus–memory connection is less than that of the partial bus networks.

Table VI lists the memory bandwidth of the partial bus network with K classes. The results are obtained from the case

TABLE VI
MEMORY BANDWIDTH OF $N \times N \times B$ PARTIAL BUS NETWORKS WITH $K = B$ CLASSES

$r = 1.0$						
No. of Buses	$N = 8$		$N = 16$		$N = 32$	
	Hier.	Unif.	Hier.	Unif.	Hier.	Unif.
2	2.0	1.98	2.0	2.0	2.0	2.0
4	3.85	3.68	3.99	3.98	4.0	4.0
8	5.97	5.25	7.71	7.35	7.99	7.97
16			11.78	10.30	15.44	14.70
32					23.48	20.41

(a)

$r = 0.5$						
No. of Buses	$N = 8$		$N = 16$		$N = 32$	
	Hier.	Unif.	Hier.	Unif.	Hier.	Unif.
2	1.85	1.81	1.99	1.98	2.0	2.0
4	2.90	2.75	3.78	3.70	3.99	3.98
8	3.47	3.23	5.81	5.51	7.64	7.49
16			6.87	6.37	11.66	11.02
32					13.69	12.67

(b)

of the number of classes $K = B$ and each class contains N/K memory modules. The number of connections in the network is proportional to $NB + (B + 1)N/2$. This connection cost is nearly equal to the partial bus networks with $g = 2$. The memory bandwidths of both networks are also very close for various values of N and B. In the partial bus networks with K classes, the memory modules which belong to class C_j can tolerate at least $j + B - K - 1$ bus failures, while all the memory modules in the partial bus networks can tolerate $B/g - 1$ bus failures. The fault tolerance of the former is more flexible than that of the latter.

From the performance–cost ratio comparison, the network with single bus–memory connection is more cost-effective than the partial bus networks. However, the single bus–memory connection scheme lacks fault tolerance. Similarly, comparison of the results of Table II, Table IV, Table V, and Table VI shows that the performance of the networks with full bus–memory connection is higher than that of the partial bus networks, but the multiple bus with full bus–memory connection is less cost-effective. The performance, cost, and fault-tolerant capability of the networks with partial bus–memory connection scheme are intermediate between the networks with single bus–memory connection and the networks with full bus–memory connection.

V. CONCLUSIONS

In this paper, we propose an architecture of $N \times M \times B$ multiple bus networks, called partial bus networks with K classes. It is more flexible and less costly than that of multiple bus networks with full connection of each processor and memory module to all buses. We also propose a general memory reference model, called hierarchical requesting model. Under this model, the performance of the multiple bus networks with various types of bus–memory connections is analyzed. The cost and fault-tolerant capability for various types of $N \times M \times B$ networks are also compared to one another. The

numerical results show that the memory bandwidth of all the networks in the hierarchical requesting case is higher than that in the uniform requesting case. When $r = 1.0$, the memory bandwidth is very much dependent on the number of buses. When $r = 0.5$, the networks with $N/2$ buses perform close to the crossbars.

The multiple bus networks with full bus–memory connection have higher memory bandwidth but are less cost-effective than all the other types of multiple bus networks. The multiple bus network with single bus–memory connection is the most cost-effective, but it lacks fault tolerance. The performance, cost, and fault-tolerant capability of the networks with partial bus–memory connection scheme are intermediate between the networks with single bus–memory connection and the networks with full bus–memory connection. Which type of the networks is selected should depend on the requirement of the multiprocessor systems.

REFERENCES

[1] F. Baskett and A. J. Smith, "Interference in multiprocessor computer systems with interleaved memory," *Commun. ACM*, vol. 19, no. 6, pp. 327–334, Jun. 1976.

[2] D. P. Bhandarkar, "Analysis of memory interference in multiprocessors," *IEEE Trans. Comput.*, vol. C-24, pp. 897–908, Sept. 1975.

[3] W. T. Chen and J. P. Sheu, "Performance analysis of multistage interconnection networks with hierarchical requesting model," *IEEE Trans. Comput.*, vol. 37, pp. 1438–1442, Nov. 1988.

[4] C. R. Das and L. N. Bhuyan, "Bandwidth availability of multiple-bus multiprocessors," *IEEE Trans. Comput.*, vol. C-34, pp. 918–926, Oct. 1985.

[5] T. Y. Feng, "A survey of interconnection networks," *IEEE Comput. Mag.*, pp. 12–27, Dec. 1981.

[6] A. Goyal and T. Agerwala, "Performance analysis of future shared storage systems," *IBM J. Res. Develop.*, vol. 28, no. 1, pp. 95–108, Jan. 1984.

[7] M. A. Holliday and M. K. Vernon, "Exact performance estimates for multiprocessor memory and bus interference," *IEEE Trans. Comput.*, vol. C-36, pp. 76–85, Jan. 1987.

[8] K. B. Irani and I. H. Onyuksel, "A closed form solution for the performance analysis of multiple-bus multiprocessor systems," *IEEE Trans. Comput.*, vol. C-33, pp. 1004–1012, Nov. 1984.

[9] T. Lang, M. Valero, and I. Alegre, "Bandwidth of crossbar and multiple-bus connections for multiprocessors," *IEEE Trans. Comput.*, vol. C-31, pp. 1227–1234, Dec. 1982.

[10] T. Lang, M. Valero, and M. A. Fiol, "Reduction of connections for multibus organization," *IEEE Trans. Comput.*, vol. C-32, pp. 707–716, Aug. 1983.

[11] M. A. Marsan and M. Gerla, "Markov models for multiple bus multiprocessor systems," *IEEE Trans. Comput.*, vol. C-31, pp. 239–248, Mar. 1982.

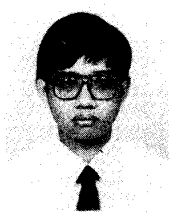
[12] T. N. Mudge and H. B. Al-Sadoun, "A semi-Markov model for the performance of multiple-bus systems," *IEEE Trans. Comput.*, vol. C-34, pp. 934–942, Oct. 1985.

[13] D. Towsley, "Approximate models of multiple bus multiprocessor systems," *IEEE Trans. Comput.*, vol. C-35, pp. 220–228, Mar. 1986.

Wen-Tsuen Chen (M'87–SM'89) was born in Taiwan on May 27, 1948. He received the B.S. degree in nuclear engineering from National Tsing Hua University, Taiwan, Republic of China, and the M.S. and Ph.D. degrees in electrical engineering and computer sciences both from University of California at Berkeley, in 1970, 1973, and 1976, respectively.

He has been with the National Tsing Hua University since 1976, and is currently a Professor of the Department of Computer Science. From 1983 to 1988 he served as the Chairman of the Department. In 1980, he was a Visiting Professor in the Department of Electrical Engineering and Computer Sciences of the University of California at Berkeley. His current research interests include computer networks, broadband ISDN, multiprocessing systems, parallel algorithms.

Dr. Chen was a recipient of the Distinguished Computer Professionals Award from the Ministry of the Economics Affairs in 1984, received the Outstanding Achievement Award from the Ministry of Defense in 1987, and the Technology Innovation Award from the Sun Yat-Sen Academics and Culture Foundation in 1989, and was a recipient of the Distinguished Research Award from the National Science Council in 1990. He is also a member of the Association for Computing Machinery, the Chinese Institute of Engineers, and Phi Tau Phi. From 1984 to 1985, he was elected as an IEEE Distinguished Visitor in region 10. Since 1988, he has been a member of the Science and Technology Advisory Board of the Ministry of Education, Republic of China, and is currently on the Board of Directors of the IEEE Taipei Section.



Jang-Ping Sheu (S'85–M'86) was born in Taiwan, Republic of China, on April 28, 1959. He received the B.S. degree in computer science from Tamkang University, Taiwan, in June 1981, and the M.S. and Ph.D. degrees in computer science from the National Tsing Hua University, Taiwan, in June 1983 and January 1987, respectively.

He joined the faculty in the Department of Electrical Engineering at the National Central University, Taiwan, as an Associate Professor in February 1987. His current research interests include parallel processing and distributed computing systems.