ANALYSIS OF MULTIPLE BUS INTERCONNECTION NETWORKS[1]

T.N. Mudge, J.P. Hayes, G.D. Buzzard and D.C. Winsor

University of Michigan
Computing Research Laboratory
Room 1079 East Engineering Bldg.
Ann Arbor, MI 48109 USA

Abstract

A new analytic performance model is presented for multiprocessor systems employing multiple bus interconnection networks. The system bandwidth is analyzed as a two stage process taking into account conflicts arising from memory and bus interference. The analysis covers multiple bus systems in which each memory is connected to every bus, and systems in which each memory is connected to a subset of the buses. The model is compared to previously published simulation data and is shown to be in close agreement.

I. Introduction

A great deal of attention has been paid to the design and analysis of interconnection networks for multiprocessor systems. Most of the previous research has dealt with crossbar networks or multistage networks [1]. While these networks are attractive for applications where high bandwidth is required, their high cost and special implementation requirements have prevented them from being used for the full range of multiprocessor applications. Most commercial systems containing more than one processor employ a single bus; consider, for example, the design philosophy advocated for the iAPX86 family in which the Multibus (IEEE 796 standard bus) provides all the intrasystem communication [2]. Single bus systems are inexpensive and easy to implement but have limited bandwidth and lack fault tolerance. A natural extension is to employ several shared buses to increase bandwidth and fault tolerance at moderate cost. Figure 1 shows typical systems in which B buses are used to interconnect N processors to M memory modules ($B \leq N$). Unlike a crossbar or multistage network, a multiple bus interconnection scheme allows easy incremental expansion of the number of processors and memories in the system. Furthermore, the buses can be configured in a variety of ways to provide a range of trade-offs between bandwidth, connection cost, and reliability.

Recently, Lang, et al. [3,4] have investigated multiple bus systems of the kind depicted in Figure 1. Using simulation they determined the bandwidth characteristics of two representative bus configurations, complete and partial. In the complete case, which is illustrated in Figure 1(a), every processor and memory module is connected to every bus; in the partial case, which is illustrated in Figure 1(b), each memory need only be connected to a subset of the buses. In particular, Lang et al. [3] showed that a complete multiple bus configuration with $B \approx N/2$ has almost the same bandwidth as an $N \times M$ crossbar, as well as higher fault tolerance. Similar advantages can be obtained at lower cost using a partial bus configuration.

This paper presents an analytic model of the bandwidth of multiple bus systems. Our results are shown to agree closely with the experimental data presented in [3]. Section II defines the underlying assumptions and develops the bandwidth model. Section III then compares the analytic model to the previous simulation results. Finally, some possible extensions of this model are mentioned in Section IV.

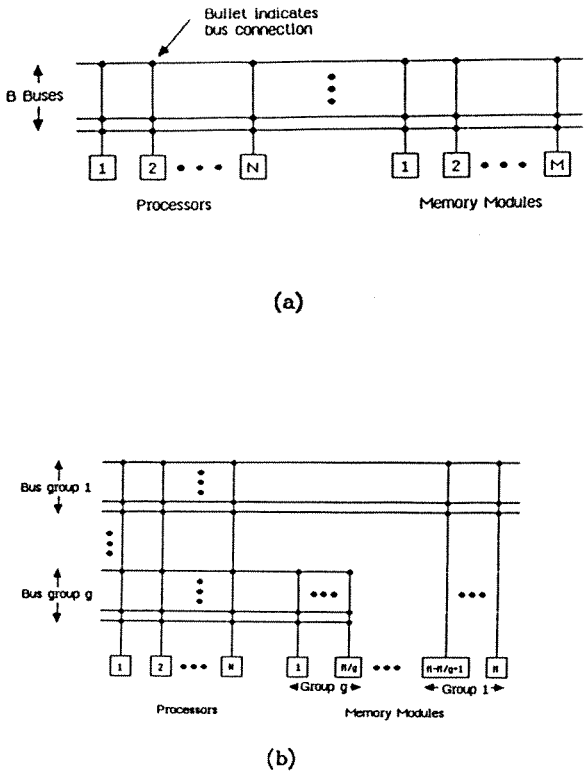

Figure 1. Two multiprocessor systems with multiple bus interconnection networks: (a) complete; (b) partial.

[1]This work was supported by the National Science Foundation, under the Grants No. ECS-8214709 and MCS-8009315.

Proceedings of the 1984 Conf. on Parallel Processing, pp. 228-232.

II. The Model

The systems shown in Figure 1 are assumed to be synchronous, and processor-memory transactions are assumed to occur during discrete time intervals termed bus cycles (continuous time analogues of such systems are discussed in [5,6]). For the purposes of this paper, bandwidth will be defined as the expected number of buses in use during a bus cycle. Apart from the configuration of the system, i.e., the values of B, M, N, and the buses grouping used, the most important factors affecting bandwidth are the rate at which memory requests are made by processors, and the degree of conflict that those requests experience.

There are two sources of conflict due to memory requests in a multiple bus system. First, more than one request can be made to the same memory module, resulting in memory interference. Second, there may be an insufficient number of buses available to accommodate all the memory requests, resulting in bus interference. In [3] a two-stage arbitration scheme is used to resolve these conflicts. In the first stage, memory interference is resolved by M 1-out-of-N arbiters each of which selects at most one outstanding request per memory module. In the second stage, bus interference is resolved by a B-out-of-M arbiter which assigns the buses to the memory requests selected in the first stage. The assignment is done on a round robin basis by each arbiter. In a realistic system requests that are blocked by either memory or bus interference are resubmitted during the following bus cycle. This policy for handling rejected requests is implemented in the simulation model of [3]. Analytic models that capture this feature appear to be intractable except in those cases where B, M, and N are very small [7].

The basic assumptions underlying our model follow those of [3]. Each processor is assumed to generate independent requests (Bernoulli trials) for memory with probability p at the start of each bus cycle. This value of p will be referred to as the request rate. Modeling the memory access process as a Bernoulli process has been validated empirically in [8-10], and is widely used as a basis for memory interference models. The memory requests are assumed to be uniformly distributed across all the memories with probability $1/M$; this is a reasonable assumption when address interleaving based on the low-order address bits is used. Hence, the probability that processor P_i requests memory M_j is p/M for all i and j. Note that the foregoing assumptions imply that the rejected requests are in effect discarded. As we will show later, this simplifying assumption yields results quite close to simulations in which blocked requests are resubmitted during the following bus cycle.

The analysis can be treated in two parts corresponding to memory interference and bus interference.

Memory interference analysis: As noted earlier the probability that processor P_i requests memory M_j is given by p/M. It follows that the probability that P_i does not request M_j is given by $(1-p/M)$, and further, that the probability that none of P_i $(i=1, \ldots, N)$ requests M_j is given by $(1-p/M)^N$. This last expression can also be interpreted as the probability that the 1-out-of-N arbiter associated with M_j has no input requests from which to choose. Conversely, the probability that there is at least one request for M_j is given by

$$q = 1-(1-p/M)^N \tag{1}$$

From the behavior of the arbiters, we can conclude that the probability that one request gains access to M_j is q for all j.

Bus interference analysis: Only the requests from at most B of the M 1-out-of-N memory request arbiters can be handled during any bus cycle, since there are only B buses. The probability that exactly i of the M memory-request arbiters output a memory request is given by

$$f(i) = \binom{M}{i} q^i (1-q)^{M-i} \tag{2}$$

The probability that B or more of the M memory-request arbiters output a memory-request can be written

$$F(B) = \sum_{i=B}^{M} f(i) \tag{3}$$

This is the probability that all B buses are in use, i.e., the interconnection network is saturated. From equations (2) and (3), the following expression can be derived for the expected number of buses in use during a bus cycle:

$$BW = B\,F(B) + \sum_{i=1}^{B-1} i\,f(i) \tag{4}$$

By our earlier definition, BW also represents the bandwidth of a complete multiple bus system.

As will be shown, the above expression for BW is in close agreement with the simulation results presented in [3]. The major source of error arises from the assumption that blocked requests are discarded. In reality, and also in the simulations, blocked requests are repeatedly resubmitted or queued until the memory they request allows them access. Equation (4) can be refined by taking this into account in the manner described below.

The probability that a memory request is accepted in the bus cycle in which it is made, is given by

$$P_a = \frac{BW}{Np} \tag{5}$$

The numerator of (5), i.e., the bandwidth, measures the number of requests that obtain memory access during a bus cycle. The denominator of (5) measures the total number of requests made by all the processors during a bus cycle. It is convenient to define an "adjusted" request rate α, that accounts for resubmission of rejected requests, where $0 \le p \le \alpha \le 1$. By assumption, the memory request process is a Bernoulli trail with success probability p or, in the case of the adjusted rate, α. It follows that the mean number of bus cycles before a request (trial) is $1/p-1$, or in the

case of the adjusted rate, $1/\alpha - 1$ [11]. Thus, the ratio of the number of successful requests to the total number of requests, i.e., P_a, is given by

$$P_a = \frac{\frac{1}{\alpha} - 1}{\frac{1}{p} - 1} \tag{6}$$

Equations (5) and (6) can be used in an iteration scheme to get an improved estimate for BW due to the adjusted rate α, as follows:

$$\alpha_{k+1} = \left[1 + \frac{BW(\alpha_k)}{Np}\left(\frac{1}{p} - 1\right)\right]^{-1} \tag{7}$$

Here we are using equation (1) for q with α replacing p. Solution of (7) for α_{k+1} yields an improved value, $BW(\alpha_{k+1})$, for the bandwidth. Any remaining deviations from the simulated bandwidth occur because α does not take into account the fact that resubmissions are all directed to the same memory. This iterative technique is an adaptation of a method first proposed by Hoogendoorn [10] (for details see [12]). For large systems, i.e., large M or N, a higher order iterative scheme may be used in place of equation (7) to reduce the number of steps to solution.

It is also possible to derive a analytic expression for the change in bandwidth, ΔBW, due to the removal (or loss) of one bus. Let

$$\Delta BW = BW(B) - BW(B-1) \tag{8}$$

Then from equation (4) it can be shown that

$$\Delta BW = F(B) \tag{9}$$

Equation (3) shows that $F(B)$ is the sum of the last $M-B+1$ terms of a binomial series. This can be approximated to the tail of a normal distribution with high accuracy if M is large [11]. In fact, the approximation works well even if M is as small as 10. By approximating ΔBW in this way, it is possible to show that $\Delta BW \approx 2$ percent if the following holds:

$$B > Mq + 2\sqrt{Mq(1-q)} \tag{10}$$

For example, for $M = N = 16$ and $p = 0.5$, a value of $B > 10$ yields a bandwidth that changes by no more than 2 percent if a bus is removed.

We now generalize equation (4) for the case of partial buses (Figure 1(b)). The memory interference analysis is the same as before, since it is independent of the bus configuration, i.e., equation (1) continues to apply. However, the bus-interference analysis needs modification. If the B buses are grouped into g equal groups (assuming g is a factor of B), equations (2) and (3) become the following:

$$f_g(i) = \binom{M/g}{i} q^i (1-q)^{\frac{M}{g} - i} \tag{11}$$

$$F_g(B) = \sum_{i=B/g}^{M/g} f_g(i) \tag{12}$$

Consequently, the bandwidth can be written as

$$BW_g = g\left[\frac{B}{g} F_g(B) + \sum_{i=1}^{\frac{B}{g}-1} i\, f_g(i)\right] \tag{13}$$

which is simply g times the bandwidth of any one of the g subsystems formed from N processors, B/g buses, and M/g memories. Equation (13) can also be incorporated into the iterative scheme of equation (7), as follows:

$$\alpha_{k+1} = 1 - \frac{BW_g(\alpha_k)}{Np}(1-p) \tag{14}$$

As before, equation (14) yields an improved value, $BW_g(\alpha_{k+1})$, for the bandwidth.

III. Evaluation of Results

In this section we briefly compare the results obtained from our analytic model with the simulation data of Lang et al. [3]. The same $N \times N$ multiprocessor configurations are employed, including systems with complete buses, and systems with two group of partial buses.

Table 1 shows the simulation results presented in [3] for complete bus systems. The bandwidth BW is calculated for various values of B and N, with p, the independent processor request rate, assigned the values 1.0 and 0.5. The data here clearly indicates that BW changes very little after N reaches $B/2$. Table 2 shows BW as predicted by equation (4). The difference between the simulated and analytic values of BW is presented in Table 3. It can be seen that, except for small values of N and p, this difference is less then about 10 percent, indicating reasonable good agreement between our results and those of [3]. Table 4 presents the bandwidth values obtained by using the iterative method. The corresponding percentage deviations from the simulated values (Table 1) appear in Table 5. The maximum difference has now been reduced to less than 7 percent in all cases.

Lang et al. also simulated the partial bus organization of Figure 1(b) with $g = 2$; their results are tabulated in Table 6. The corresponding analytic data, obtained using the iterative method of Section II, is given in Table 7. Comparison of these two sets of results shows good agreement (less than 7 percent) between the analytic and empirical data; see Table 8.

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2	1.97	1.65	2.00	2.00	2.00	2.00	2.00	2.00
3	2.55	1.77	3.00	2.87	3.00	3.00	3.00	3.00
4	2.62	1.77	3.93	3.33	4.00	3.95	4.00	4.00
5			4.62	3.45	4.99	4.67	5.00	4.98
6			4.90	3.47	5.93	5.03	6.00	5.85
7			4.94	3.47	6.68	5.13	6.98	6.43
8			4.95	3.47	7.12	5.16	7.92	6.70
9					7.27	5.16	8.72	6.82
10					7.28	5.16	9.27	6.83
11					7.30	5.16	9.53	6.83
12					7.30	5.16	9.61	6.83
13							9.63	6.84
14							9.63	6.84
15							9.63	6.84
16							9.63	6.84

Table 1. Bandwidth BW obtained by simulation for the complete bus.

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1	0.99	0.88	1.00	0.98	1.00	1.00	1.00	1.00
2	1.89	1.43	2.00	1.88	2.00	1.98	2.00	2.00
3	2.52	1.63	2.97	2.57	3.00	2.89	3.00	2.98
4	2.73	1.66	3.87	2.99	3.99	3.67	4.00	3.91
5			4.59	3.16	4.97	4.23	5.00	4.74
6			5.04	3.22	5.88	4.57	5.99	5.41
7			5.22	3.23	6.66	4.72	6.97	5.87
8			5.25	3.23	7.24	4.78	7.89	6.15
9					7.58	4.80	8.72	6.29
10					7.73	4.80	9.39	6.35
11					7.77	4.80	9.86	6.37
12					7.78	4.80	10.13	6.37
13							10.25	6.37
14							10.29	6.37
15							10.30	6.37
16							10.30	6.37

Table 2. Bandwidth BW calculated from equation (4).

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1	-1.00	-12.00	0.00	-2.00	0.00	0.00	0.00	0.00
2	-4.06	-13.33	0.00	-6.00	0.00	-1.00	0.00	0.00
3	-1.18	-7.91	-1.00	-10.45	0.00	-3.67	0.00	-0.67
4	4.20	-6.21	-1.53	-10.21	-0.25	-7.09	0.00	-2.25
5			-0.65	-8.41	-0.40	-9.42	0.00	-4.82
6			2.86	-7.20	-0.84	-9.15	-0.17	-7.52
7			5.67	-6.92	-0.30	-7.99	-0.14	-8.71
8			6.06	-6.92	1.69	-7.36	-0.38	-8.21
9					4.26	-6.98	0.00	-7.77
10					6.18	-6.98	1.29	-7.03
11					6.44	-6.98	3.46	-6.73
12					6.58	-6.98	5.41	-6.73
13							6.44	-6.87
14							6.85	-6.87
15							6.96	-6.87
16							6.96	-6.87

Table 3. Percentage difference between calculated (Table 2) and simulated values (Table 1) of BW.

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1	0.99	0.97	1.00	1.00	1.00	1.00	1.00	1.00
2	1.89	1.58	2.00	1.98	2.00	2.00	2.00	2.00
3	2.52	1.77	2.97	2.80	3.00	2.99	3.00	3.00
4	2.73	1.79	3.87	3.27	3.99	3.91	4.00	3.99
5			4.59	3.46	4.97	4.60	5.00	4.95
6			5.04	3.51	5.88	4.99	5.99	5.79
7			5.22	3.52	6.66	5.16	6.97	6.37
8			5.25	3.52	7.24	5.23	7.89	6.71
9					7.58	5.25	8.72	6.88
10					7.73	5.25	9.39	6.95
11					7.77	5.25	9.86	6.98
12					7.78	5.25	10.13	6.98
13							10.25	6.98
14							10.29	6.98
15							10.30	6.98
16							10.30	6.98

Table 4. Bandwidth BW with adjusted rate α calculated from equation (7).

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1	-1.00	-3.00	0.00	0.00	0.00	0.00	0.00	0.00
2	-4.06	-4.24	0.00	-1.00	0.00	0.00	0.00	0.00
3	-1.18	0.00	-1.00	-2.44	0.00	-0.33	0.00	0.00
4	4.20	1.13	-1.53	-1.80	-0.25	-1.01	0.00	-0.25
5			-0.65	0.29	-0.40	-1.50	0.00	-0.60
6			2.86	1.15	-0.84	-0.80	-0.17	-1.03
7			5.67	1.44	-0.30	0.58	-0.14	-0.93
8			6.06	1.44	1.69	1.36	-0.38	0.15
9					4.26	1.74	0.00	0.88
10					6.18	1.74	1.29	1.76
11					6.44	1.74	3.46	2.20
12					6.58	1.74	5.41	2.20
13							6.44	2.05
14							6.85	2.05
15							6.96	2.05
16							6.96	2.05

Table 5. Percentage difference between calculated (Table 4) and simulated values (Table 1) of BW.

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1+1	1.74	1.50	1.87	1.83	1.92	1.90	1.93	1.93
2+2	2.62	1.77	3.61	3.11	3.81	3.64	3.86	3.79
3+3			4.72	3.44	5.52	4.76	5.73	5.43
4+4			4.93	3.47	6.77	5.10	7.48	6.42
5+5					7.24	5.16	8.79	6.77
6+6					7.28	5.16	9.43	6.83
7+7							9.59	6.84
8+8							9.63	6.85

Table 6. Bandwidth BW_2 obtained by simulation for the partial bus case (2 groups).

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1+1	1.80	1.49	1.97	1.92	2.00	1.99	2.00	2.00
2+2	2.73	1.79	3.73	3.12	3.95	3.76	3.99	3.95
3+3			4.88	3.47	5.71	4.80	5.94	5.58
4+4			5.25	3.52	7.00	5.16	7.71	6.50
5+5					7.63	5.24	9.10	6.86
6+6					7.78	5.25	9.92	6.96
7+7							10.24	6.98
8+8							10.30	6.98

Table 7. Bandwidth BW_2 calculated from equation (14).

Number of Buses, B	Number of processors N $(=M)$							
	4		8		12		16	
	p=1	p=.5	p=1	p=.5	p=1	p=.5	p=1	p=.5
1+1	3.45	-0.67	5.35	4.92	4.17	4.74	3.63	3.63
2+2	4.20	1.13	3.32	0.32	3.67	3.30	3.37	4.22
3+3			3.39	0.87	3.44	0.84	3.66	2.76
4+4			6.49	1.44	3.40	1.18	3.07	1.25
5+5					5.39	1.55	3.53	1.33
6+6					6.87	1.74	5.20	1.90
7+7							6.78	2.05
8+8							6.96	1.90

Table 8. Percentage difference between calculated (Table 7) and simulated values (Table 6).

IV. Conclusion

We have presented new analytic formulas for BW and BW_g. Although they are fairly simple, they are in close agreement with previous simulation results. Possible extension to this work include examining the effects of different arbitration schemes on BW and fault tolerance. Also of interest is obtaining a simple approximation for $BW - BW_g$ to evaluate the bandwidth degradation of different partial bus

groupings. After this paper was submitted a result similar to that in equation (4) (*BW* for a complete system) was derived independently in [13].

V. References

(1) *Computer*, vol.14, no.12, Dec. 1981.

(2) Intel Corp., *iAPX 86,88 User's Manual*, Santa Clara, Calif., 1981.

(3) T. Lang, M. Valero and I. Alegre, "Bandwidth of Crossbar and Multiple-Bus Connections for Multiprocessors," *IEEE Trans. on Computers*, vol.C-31, no.12, pp.1227-1233.

(4) T. Lang, M. Valero and M.A. Fiol, "Reduction of Connections for Multibus Organization," *IEEE Trans. on Computers*, vol.C-32, no.8, pp.707-716.

(5) M.A. Marsan and M. Gerla, "Markov Models for Multiple Bus Multiprocessor Systems," *IEEE Trans. on Computers*, vol.C-31, no.3, pp.239-248, March 1982.

(6) I.H. Onyüksel and K.B. Irani, "A Markovian Queueing Network Model for Performance Evaluation of Bus-Deficient Multiprocessor Systems," *Proc. 1983 Int'l Conf. on Parallel Processing*, pp. 437-439, Aug. 1983.

(7) C.E. Skinner and J.R. Asher, "Effects of Storage Contention on System Performance," *IBM Systems Journal*, vol.8, no.4, pp.319-333, 1969.

(8) D.P. Bhandarkar, "Analysis of Memory Interference in Multiprocessors," *IEEE Trans. on Computer*, vol.C-24, no.9, pp.897-908, Sep. 1975.

(9) F. Baskett and A.J. Smith, "Interference in Multiprocessor Computer Systems with Interleaved Memory," *Comm. of ACM*, vol.19, no.6, pp.327-334, June 1976.

(10) C. H. Hoogendoorn, "A General Model for Memory Interference in Multiprocessors," *IEEE Trans. on Computers*, vol.C-26, no.10, pp.998-1005, Oct. 1977.

(11) W. Feller, *An Introduction to Probability Theory and its Applications*, vol.1, 3rd ed., Wiley, New York, 1968.

(12) D.W.L. Yen, J.H. Patel, and E.S. Davidson, "Memory Interference in Synchronous Multiprocessor Systems," *IEEE Trans. on Computers*, vol.C-31, no.11, pp.1116-1121, Nov. 1982.

(13) 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.