Memory Latency in Distributed Shared-Memory Multiprocessors

Bahman S. Motlagh
Dept. of Engineering Technology
University of Central Florida
12424 Research Parkway
Orlando, FL 32826
Tel: 407-384-2153
E-mail: bmotlagh@mail.ucf.edu

Ronald F. DeMara
Dept. of Electrical and Computer Engr.
University of Central Florida
Box 162450
Orlando, FL 32816-2450
Tel: 407-823-5916
E-mail: rfd@engr.ucf.edu

Abstract

Analytical models were developed and simulations of memory latency were performed for Uniform Memory Access (UMA), Non-Uniform Memory Access (NUMA), Local-Remote-Global (LRG), and Replicated Concurrent-Read (RCR) architectures for hit rates from 0.1 to 0.9 in steps of 0.1, memory access times of 10 nsec to 100 nsec, proportions of read/write access from 0.01 to 0.1, and block sizes of 8 to 64 words. The RCR architecture based on redundant inexpensive DRAM is shown to provide favorable performance over UMA and NUMA architectures for application and system parameters in the range evaluated. RCR outperforms LRG architectures when the hit rates of the processor cache exceed 80% and hit rates of replicated memory exceed 25%. Inclusion of a small replicated memory at each processor significantly reduces expected access time since all replicated memory READ access hits become independent of global traffic. For configurations of up to 32 processors, results show that latency is further reduced by distinguishing burst-mode transfers between isolated memory accesses and those which are incrementally outside the working set.

1 Introduction

Rapid changes in the cost and density of semiconductor memory technology have enabled new multiprocessor design approaches which were previously cost-prohibitive. In particular, traditional interconnection strategies between multiple processors and a common memory regard the storage space as a very scarce resource. These conventional approaches restrict scalability by incurring latency to transfer data whenever and wherever remote memory misses occur. Previous designs have addressed this problem by increasing the complexity of local caches or using multistage combining networks and elaborate referencing schemes, but require sophisticated hardware to maintain coherence between the physically-distinct memories. In particular, there is a need to design architectural support for *shared-memory programming model* that possesses a reasonable balance between cost and performance.

0-7803-4391-3/98/$10.00 1998 IEEE

2 Simulator Development

To evaluate Distributed Shared-Memory architectures, analytical models were developed for UMA, NUMA, LRG, and RCR architectures. Since hardware configurations of computer systems could vary among different models, consistent assumptions have been made to allow more direct comparisons of the results. The simulation code consists of a series of functions written in C programming language. The main program contains a FOR loop which allows for simulations in ten nanosecond increments per cycle. As simulation progresses, the total memory access time is recorded and expected access times are computed. A few assumptions have been made in order to resume consistency in analyzing generated data from the simulators. If there is a READ miss, then a block of data will be copied to the cache. In the case of a *shared-write* then *write-through* policy is implemented.

3 Uniform Memory Access (UMA) Architecture

In the UMA architecture, a READ hit feches data from the cache in t_c time and the probability of a cache hit is denoted as h_c. A READ miss will cause P_i to access the global memory in order to fetch data. The probability of having to access global memory is $(1-h_c)$. P_i may have to wait to access the cache if there are a number of pending WRITEs since only one invalidation can occur at a time. Since every processor has a probability of $(1-h_c)$ of accessing shared-memory, a delay in accessing the shared-memory will be inevitable. In the UMA architecture, as the number of processors increases, undesirable delays will increase average memory access time.
In the UMA architecture, since there is no local memory other than cache, the cache hit rate has a major impact on its performance. The simulation has been repeated for a various number of processors in the system. The average access time shows an improvement of over 85% as h_c increases from 10% to 90%. The effects of other parameters of simulation will be discussed as RCR, UMA, NUMA and LRG simulation results are compared.

4 Non-Uniform Memory Access (NUMA) Architecture

In the NUMA architecture, shared memory is distributed among all processors. Every processor can address its local memory or remote memories of other processors. Every P_i is also attached to a private cache. A READ hit fetches data from the cache in t_c time. The cache hit rate is denoted as h_c. A cache miss with a probability of $(1-h_c)$ will cause an access to local memory. Fetching data from local memory may be delayed if there are other pending READ or WRITES by other processors. In the case of a local memory miss, remote memories will be accessed. The cache hit rate h_c and local memory hit rate h_L are two major parameters in the performance evaluation of NUMA machines. There is a positive correlation between average access time and h_L.

5 Local-Remote-Global (LRG) Architecture

Local-Remote-Global Architecture is a combination of UMA and NUMA machines. Every cluster contains two processors and a shared local memory. Each processor on the cluster is attached to a private cache. LRG also provides a global memory accessible by all processors. A READ hit with a probability of h_c fetches data directly from the cache in t_c time. In the case of a cache miss, the local memory is referenced. Let h_L denote the local memory hit rate. The probability of accessing global memory is $(1-h_c)(1-h_L)$. As a result, the average access time is a function of h_c and h_L. This simulator was used to study and analyze the impact of various ratios of h_c and h_L. For a complete analysis of the effects of h_c and h_L on expected memory access time, various percentages of h_c and h_L were studied. The cache hit rate has a more drastic effect on expected access time than the local memory hit rate.

6 Replicated Concurrent-Read (RCR) Architecture

In the RCR, a READ miss will cause a reference to replicated memory to access the requested word. If the address requested is not held by replicated memory then the auxilary memory will be referenced. The probability of a cache hit is h_c and the replicated memory hit is h_L. Processor P_i with probability of $(1-h_c)$ will face a cache miss and has $(1-h_L)$ chance of replicated memory miss. Therefore, the chance of having to access auxilary memory is $(1-h_c)(1-h_m)$. P_i will have to wait until this data is transfered to P_i's private cache C_i. During this time P_i will be inactive. P_i has to compete with other processors to access the global bus. As a result, P_i may have to wait for its turn to access the auxilary memory. A WRITE access is treated differently in the RCR architecture. Every *shared-write* access is broadcasted to all replicated memories. The simulator also generated the number of other memory references pending for bus access in order to compute the wait time for P_i. Various cache hit rates, with respect to different replicated memory hit rates, were studied. This experiment has been conducted for 8, 16, 32, and 64 processor systems. Average access time decreases as the replicated memory hit rate increases. Let N denote the total number of processors. When N=8, there is more than a 64% improvement in access time as replicated memory hit rates increase from 10% to 80%. Systems with 16, 32, and 64 processors also demostrate an improvement in average access time by over 68%. As more memory accesses are satisfied by cache and replicated memory, better average access time and more CPU utilization results.

7 Comparison of Distributed Shared-Memory Architectures Performance

For the purpose of comparing these machines, the effect of numerous varying parameters will be examined. As shown in Figure 1, the effect of various cache hit rates is illustrated. As expected, the NUMA machine has shown great improvement in average access time, with respect to varying cache hit rates. The reason being that the delays caused by the interconnection network are decreased. All of the other machines have shown improvement as h_c is increased from 10% to 90%. When h_c is above 75%, RCR delivers a reduced memory access time. When h_c is below 75%, RCR's memory access time is comparable to LRG's average access time. Figure 2 shows the effect of varying local memory hit rates on expected memory access time. Since the UMA machine does not have local memory, its average memory access time does not vary. RCR demonstrates a direct effect as a result of increasing the replicated memory hit rate. The NUMA machine demonstrates a better performance as the hit rate increases. The LRG machine is affected less by the local memory hit rate than the RCR and NUMA machines. Figure 3 illustrates the impact of varying local memory hit rates when the cache hit rates increase. The effect of varying local memory hit rates, accompanied by higher cache hit rates, is more pronounced with the RCR and NUMA machines. The other machines did show improvement, but not as significantly as that of the rate of the RCR machine. Figure 4 also illustrates the effects of varying local memory hit rates in conjunction with a 90% cache hit rate. The RCR, LRG, and NUMA machines demonstrates an improvement as local memory hit rates increase. The effect of varying shared-write percentages on these machines have been analyzed. Let P_{shared_write} denote the probability of a shared-write memory access such that $P_{shared\_write} + P_{private\_write} + P_{read} = 1$. The results, shown in Figure 5, illustrate a slight increase in the average memory access time as the probability of shared writes increases from 0.0 to 0.5. Figure 6 shows the effect of varying block sizes on expected memory access time on the RCR, UMA, NUMA, and LRG machines. The NUMA machine demonstrates a drastic increase in memory access time as the block sizes increase from 8 to 64. The main reason for this increase in memory access time is due to the transfer of blocks of data from remote memories. The UMA machine also experienced a significant increase in memory access time due to the transfer of blocks of data from global memory. The RCR and LRG machines demonstrate a lesser effect as block size increases.

8 References

[BISIANI90] R. Bisiani and M. Ravishankar, "PLUS: A Distributed Shared-Memory System," *Proceedings of the 17th International Symposium on Computer Architecture*, May 1990.

[HARRIS95] Harris Computer Systems Corporation, *Night Hawk 6800 Functional Specification*, Brochure, 1995.

[HWANG84] Hwang, Briggs, *Computer Architecture and Parallel Processing*, McGraw-Hill, Inc., 1984.

[LENOSKI95] Lenoski, Weber, *Scalable Shared-Memory Multiprocessing*, Morgan Kaufmann Publishers, Inc., 1995.

[MOTLAGH97] Bahman S. Motlagh, *A Replicated Concurrent-Read Architecture for Scalable Shared-Memory Multiprocessing*, Ph.D. dissertation, University of Central Florida. 1997.

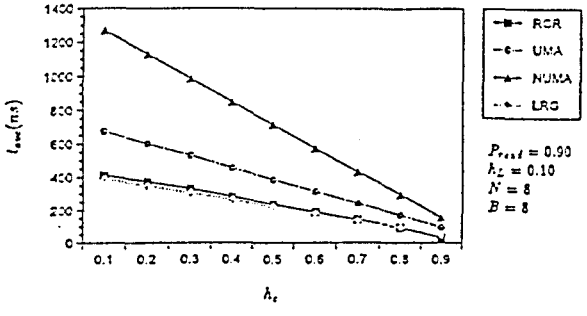

Figure 1: Expected access time for various cache hit rate percentages.

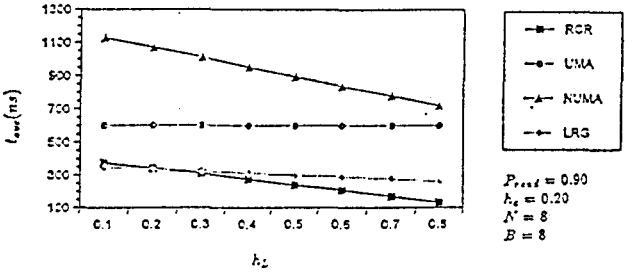

Figure 2: Expected access time for various h_L when $h_c = 0.25$.

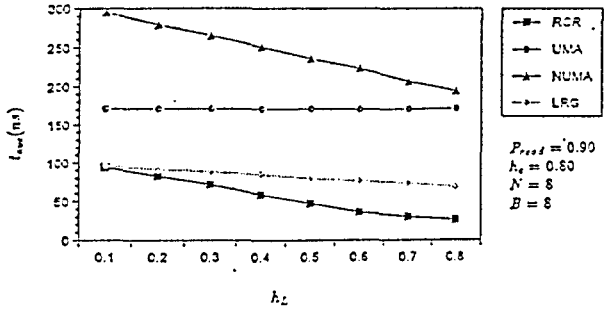

Figure 3: Expected access time for various h_L when $h_c = 0.80$.

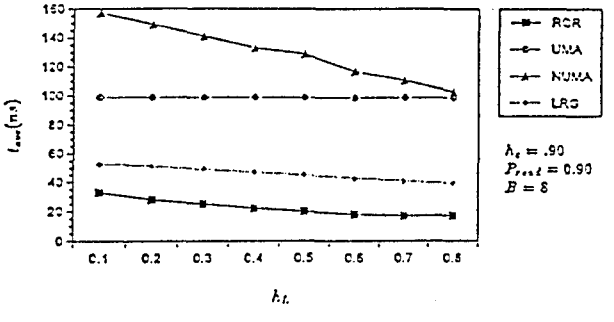

Figure 4: Expected access time for various h_L when $h_c = 0.90$.

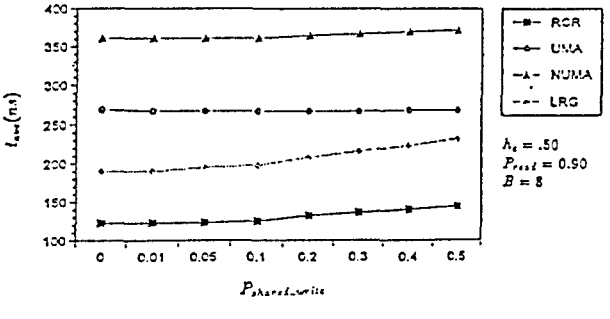

Figure 5: Expected access time for various shared-write percentages.

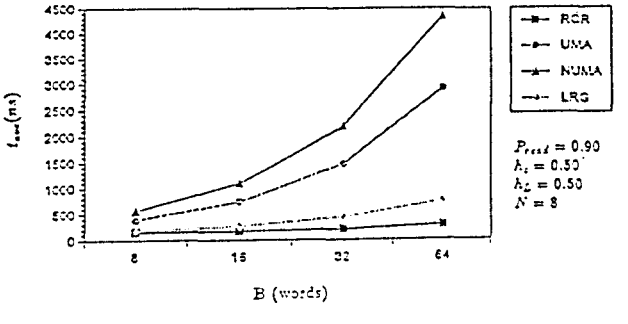

Figure 6: Expected access time for various block sizes.