

Performance Evaluation of A WDMA OIDSMS Multiprocessors

I-Shyan Hwang
Department of Electronic Engineering
Van-Nung Institute of Technology
Chung-Li, Taiwan, 32045

ABSTRACT

Optically interconnected *distributed shared memory* (OIDSMS) systems offer significant performance advantages due to the fast interconnection network. The photonic network of the proposed approach is based on a wavelength-division multiplexed (WDMA) passive star-coupled configuration. *Optical self-routing* is achievable which partitions the traffic, relaxing the design constraints on the receiver subsystem since a node now only receives and processes a fraction of the network traffic. A major concern with the multi-access approach is that a media access control and a cache coherence protocol are required to provide access to a distributed arbitration of the WDMA photonic network. In particular, one class of media access protocol (TDMA-C) requires a control channel to broadcast reservation requests, and the broadcast capability is also able to support coherence level control signals such as invalidations which enable a snooping based coherence protocol. This paper evaluates how OIDSMS can ease the traffic in large-scale snooping-based shared memory multiprocessors.

Key Words : OIDSMS, WDMA, media access control, cache coherence protocol.

1. INTRODUCTION

Optically interconnected *distributed shared memory* (OIDSMS) systems offer significant performance advantages due to the fast interconnection network [1, 2, 3, 4] and complexity through a simplified memory allocation strategy [5]. The objective of this paper is to examine the fast OIDSMS interconnection network to support larger scalability multiprocessor systems.

The approach proposed in this paper eases the traffic design constraint by incorporating a wavelength-division multiplexed (WDM) photonic network to support large scale interprocessor communication. A major problem hindering the development of photonic based communication networks is the speed mismatch between the electronic and optical components [6]. This illustrates the *speed mismatch problem*: the optical media is capable of speeds far exceeding the maximum speeds of the electronic interface components. WDM is an approach that circumvents the speed mismatch problem by partitioning the bandwidth into

many, *more manageable*, high speed channels. Each channel operates at the data-rate limited by the electronic interface components. This achieves a significant improvement in bandwidth utilization, allowing concurrent transmission over the multiple channels.

The photonic network of the proposed approach [7, 8], based on a WDMA passive star-coupled configuration shown in Fig. 1 eases the heavy communication burden required to support the shared memory view and relaxes this significant design constraint. The network is no longer a principal constraint. Limitations due to network bandwidth have diminished and performance improves due to the superior performance of the interconnection network. Depending on the architecture, *optical self-routing* is achievable where a node only receives data destined to it and the system has the non-blocking connectivity characteristics of a crossbar.

The object of this paper is to evaluate how large scale snooping based shared memory multiprocessors can be used to ease the traffic in OIDSMS. A semi-Markov model is developed to study the performance of OIDSMS.

2. SYSTEM ARCHITECTURE

The following section describes the proposed architecture and the media access protocol on which the cache coherence protocol is based.

2.1 Node Organization

The OIDSMS system consists of m identical nodes. Each node possesses a local processor, a memory management unit (MMU) and a receiver/transmitter subsystem as shown in Fig. 2. A node has two levels of cache: the processor cache (PC), as with the Intel i860 and DEC Alpha (21064-AA), and an extended cache (EC) located in the physical memory of the node. The physical memory of a node is partitioned into EC and local (global) memory (LGM). LGM is mapped into the global physical address space of the system, and the EC is used to cache the global virtual memory. This allows the EC to be constructed with the same low cost memory as the LGM.

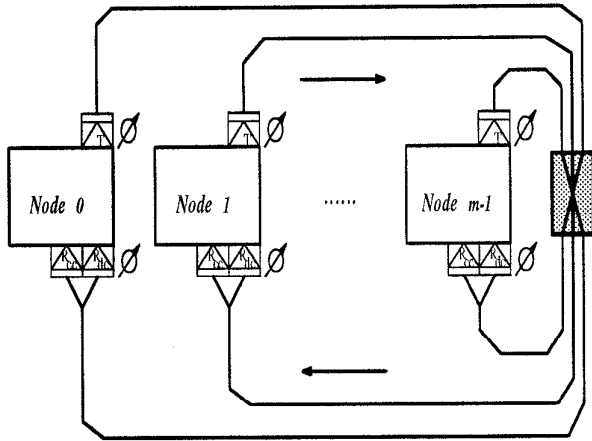


Figure 1: Passive optical star-coupled configuration with wavelength tunable devices to achieve wavelength division multiple access.

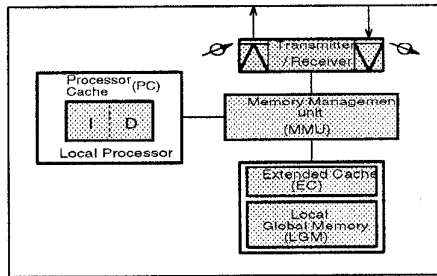


Figure 2: Node organization

2.2 Photonic Media Access Protocol

This section describes the media access protocol being considered. TDMA-C [9, 2, 4] is a reservation-based protocol with a time-division multiplexed control channel. TDMA-C requires one fixed and one tunable receiver at each node. The systems consider m nodes and C wavelength channels numbered $\{c_0, c_1, \dots, c_{C-1}\}$.

2.2.1 TDMA-C

A control cycle consists of m control slots as shown in Fig. 3. Every node has an assigned control slot it uses to reserve access on a data channel if backlogged. In Fig. 3, node P transmits a control packet in control slot T_P . The control slot includes the time required for the source node to check the status tables, build and transmit the control packet. The transmitter then waits for L time slots before transmitting the data packet on the selected data channel, where L is the *switching latency* which includes the wavelength tuning time of optical devices and the proto-

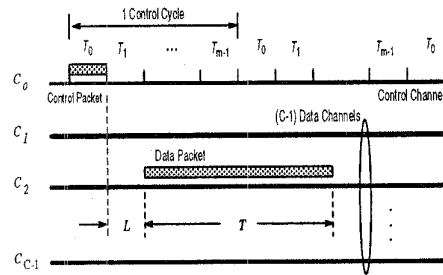


Figure 3: Time-space diagram for TDMA-C access protocol illustrating access to the control channel and transmission on the data channels.

col processing time. The switching latency is defined as $L = \max\{t_s, t_r\}$, where t_s is the time required by the transmitter of the source to switch to the selected wavelength, and t_r is the time required by the target node to receive and process the control packet and switch the tunable receiver to the selected data channel.

Collisionless transmission is achieved by this protocol through the use of status tables. Each node maintains two tables: a table to track the status of the data channels to eliminate data channel collision, and a table to avoid destination conflict by tracking the status of the R_{dc} receiver at each node. This is why each node has receiver R_{cc} parked on the control channel: all transmitted control packets are received by all nodes (including the node that transmitted the control packet). R_{cc} updates the two status tables at the end of each control slot after receiving and decoding a control packet. If node m_j transmits a control packet targeting m_i on data channel c_k , all nodes add $T + L$ against entry i in their node status table and entry k in the channel status table. The entries indicate the number of time slots that the resources will be busy. All positive entries of each table are decremented at the end of every control time slot to update the remaining busy control slots.

A backlogged node checks its status tables at the beginning of its preallocated time slot. If the target node has a status table entry equal to 0, it is considered idle. If the target node is idle, the transmitter then checks for any available data channel. A data channel is considered idle if its status table entry is less than or equal to L . This control packet is then formed with the source, target, selected data channel and packet length identifiers. If a node is not backlogged, its control slot remains idle during that cycle. In case the target is busy or an idle data channel is not available, the transmitter waits until the next cycle to

attempt transmission.

2.4 Cache Coherence Protocol

The unit of transfer is a block, made up of k lines, but the unit of invalidation is a single line. This allows us to take advantage of spatial locality and alleviate the problem of *false sharing* [10, 11]. The snooping-based protocol investigated in this paper employs a *write-invalidate* policy. The action that takes place following a read miss and write miss depends on the media access protocol.

3. PERFORMANCE MODEL

This section defines the model used to evaluate the OIDS system performance. The model, based on a semi-Markov process, is used to determine the impact of the media access protocol incorporating cache coherence protocol on system performance. The model predicts the behavior of a process that resides at each of the m nodes.

3.1 Model Assumptions

The OIDS system is described in terms of normalized time. Time is normalized to the average time between successive EC accesses. Since the nodes in the system are assumed to possess an on-chip processor cache, the basic time unit could be viewed as the average time between a processor cache miss. The memory access time of a block from LGM or NLGM is assumed to be M time units. The time to transmit a block, not including any queuing or overhead due to the media access protocol, from source to target node is denoted as T .

The behavior of the system is described in terms of the state of the process executing at a local processor. The process is viewed as residing at the processor (extended cache hit), local memory (LGM hit), the communication network during transfer, or external access (NLGM hit). The process may be *blocked* due to contention at a memory module or network. Network latency is highly dependent on the media access protocol. The system is assumed to possess C multiple access channels created through WDM in the photonic network. System operations will be characterized by the following assumptions:

- \mathcal{A}_1 : *i.i.d.* behavior of the nodes.
- \mathcal{A}_2 : Each processor submits a global memory request when an extended cache miss occurs with a hit ratio of α .
- \mathcal{A}_3 : The probability that more than one node requests arrival or release of a memory module or a data channel between time t and time $t + \Delta t$ is $o(\Delta t)$.

\mathcal{A}_4 : Packet-switched system. A source node sends a request message (a single packet) to the target node requesting a copy of a particular memory block. The target node receives the packet, decodes the request, accesses its local memory to satisfy the global memory request, then sends the requested data to the source node via a single packet along the OIDS network.

\mathcal{A}_5 : Random selection for service (RSS) queue discipline for network and memory access. When more than one request is queued for a resource, the requests are selected for service in random order independent of the time of arrival.

\mathcal{A}_6 : No context switching. A processor waits for the response to an external request rather than context switching.

\mathcal{A}_7 : The memory access time is M time units. Access to memory is slotted on unit time. Time is slotted for network access based on the block transmission time T (there is an initial synchronization delay for network access to align to the packet slot boundaries).

\mathcal{A}_8 : Request and response packets have the same size.

The following section derives a performance model of the proposed system based on a semi-Markov model.

3.2 The Semi-Markov Model

A semi-Markov process is developed to approximate the behavior of a node. The state diagram of the model is depicted in Fig. 4. Note that the self-loops in this model could be eliminated by specifying the average sojourn time to be the mean of the appropriate distribution. However, the self-loops have been retained since we feel they aid in clarity and add little additional complexity.

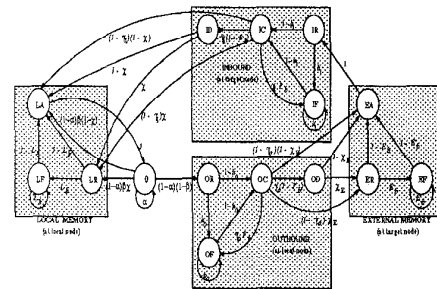


Figure 4: State diagram of the semi-Markov model for the OIDS system.

The state diagram of the model is depicted in Fig. 4. The states of the semi-Markov process define the behavior of a node in the system. In addition to S_0 ,

the active state, S_{yz} denotes a state with an average sojourn time of τ_{yz} , where $y \in \mathcal{Y}$ and $z \in \mathcal{Z}$. The two sets, \mathcal{Y} and \mathcal{Z} , are defined as follows. Let $\mathcal{Y} = \{L, E, O, I\}$, where L , E , O , and I denote local memory access, external memory access (either owner or dirty memory), outbound memory request via network, and inbound memory response via network, respectively. Let $\mathcal{Z} = \{R, F, C, D\}$, with the elements defined as residual wait (R), full wait (F), access resource to control channel (C), and data channels (D).

As shown in Fig. 4, states often appear as triples: the access state, the full wait state, and the residual wait state. The residual wait state represents a synchronization delay: the delay until the beginning of the next access cycle of the resource. The full wait state represents queueing (with a RSS service discipline) for the resource, once the initial residual time has been met, until access is obtained.

The traffic is highly dependent on the actions of cache coherence protocol. There are two types: *requests* and *responses*. A *request packet* is generated upon the local node needs the target's data when Read Miss or Write Miss occurs. A *response packet* can be either the requested data or the *write permission* in response to a request packet received by the local node.

The semi-Markov process considered in this paper has an irreducible embedded Markov chain with ergodic states so the above equation is always applicable.

4 PERFORMANCE ANALYSIS

The model developed in the previous sections is now used to analyze the behavior of the OIDS system. The analytical model is validated through a comparison with simulation. The performance of the system is analyzed in terms of the following metrics: average transaction time per access, memory module utilization, and data channel utilization.

4.1 Validation of the model

This section compares the values predicted by the analytical model to simulation results. The simulator is based on a stochastic self-driven discrete event model, written in the C programming language with a C -based library of routines that provided discrete-event and random variate facilities. A trace-based simulation was performed using the *fft64* trace, which features heavy sharing between all processors (the shared data is often referenced [12]). It is intended to stress the system with reservation media access protocol to give the results in a worst case scenario.

The maximum deviation between the analytical and simulation model is shown in Fig. 5 to be less

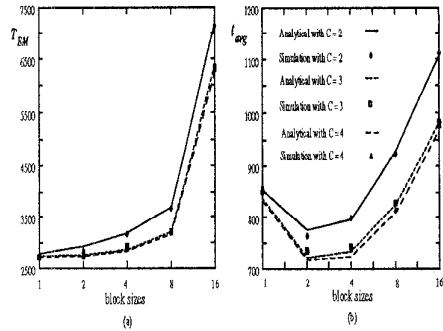


Figure 5: Comparison of the analytic model to *fft64* trace-based simulation with varying the block sizes $\in \{1, 2, 4, 8, 16\}$ times of line size. Points and lines represent simulation and analytic model values, respectively. (a) T_{EM} , (b) t_{avg} .

than 2% and 4% for T_{EM} and t_{avg} at block size = 2 times of line size and block size = 4 times of line size, respectively. The results show a high degree of correlation between the simulation and analytical models. The graphs show how system performance is impacted by variations in channels and block sizes. In general, the graphs show that T_{EM} increases when block size increases due to the problem of *false sharing* [11, 13], or the number of channels is decreased, and t_{avg} depends on the block size is minimized.

Based on the simulation results, the deviations for T_{EM} and t_{avg} when the numbers of channels are greater than 3 are less than 4%, respectively. It shows that only two or three high speed channels can support a 64 processor OIDS. The following sections investigate the performances of OIDS by varying the parameters based on *fft64* trace. for system $\in \{8, 16, 32\}$ that only one high speed data channel can support the systems. No impact on performance is shown until the data channel is reduced to one. Since some metrics have the same trend, we will show the one typical metric and explain the other metrics. Unless stated, the block size = 2 time of lines is chosen in the following sections.

4.2 Variation in hit ratio α

This section investigates the impacts of the hit ratio α on system performance and the other parameters are fixed.

Fig. 6 shows the performances of the OIDS system by varying the hit ratio $0.8 \leq \alpha \leq 1.0$. The metrics show that t_{avg} and u_C both are

decreased when α is increased. Fig. 6(a) shows the average transaction time per access, t_{avg} . The t_{avg} is decreased by 9.8% when the number of channel

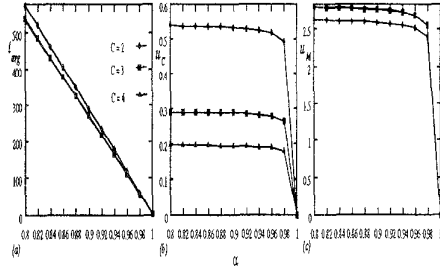


Figure 6: The performances of OIDS system by varying the hit ratio $0.8 \leq \alpha \leq 1.0$. (a) The average transaction time per access, t_{avg} . (b) The utilization per data channel, u_C . (c) The total memory utilization, U_M

decreases from 3 to 2 at $\alpha = 0.8$, and are all saturated when the number of channels is greater than 3. The U_C, U_M , and u_M have the same trend as t_{avg} . Fig. 6(b) shows the utilization per data channel, u_C . The data channel utilization is relaxed by 46% when the number of channels decreases from 3 to 2 at $\alpha = 0.8$, and by 30% when the number of channels decreases from 4 to 3, respectively.

4.3 Variation in β

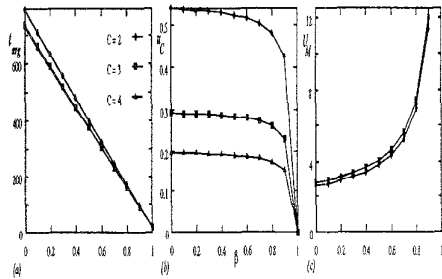


Figure 7: The performances of OIDS system by varying the $0.0 \leq \beta \leq 1.0$ (a) The average transaction time per access, t_{avg} . (b) The utilization per data channel, u_C . (c) The total memory utilization, U_M .

Fig. 7 shows the performance of the OIDS system by varying β . The direct impact on system performance is U_M and u_M . U_M or u_M increases when β is increased, and the rest of the metrics decrease when β is increased. Fig. 7(a) shows that the average transaction time per access, t_{avg} , is an inverse linear proportional relationship by varying β from 0.0 to 1.0. The t_{avg} is decreased 9% from the number of channel $C = 2$ to $C = 3$ at $\beta = 0$, and no impact on the number of channel when $\beta = 1.0$. T_{EM} shows the same result as t_{avg} . Fig. 7(b) shows the utilization per data

channel, u_C , with varying the β . u_C is decreased when β increases. The u_C decreases 47% from the number of channels $C = 2$ to $C = 3$ at $\beta = 0$, 33% from the number of channels $C = 3$ to $C = 4$ at $\beta = 0$, respectively. Fig. 7(c) shows the total memory utilization, U_M , by varying a . When β increases, the total memory utilization, U_M , increases. The U_M is sensitive to the variation of β when β approaches 1.0. The U_M is 44 when $\beta = 1.0$.

4.4 Variation in γ_O and γ_I

This section investigates the variation in γ_O and γ_I on the OIDS system performance.

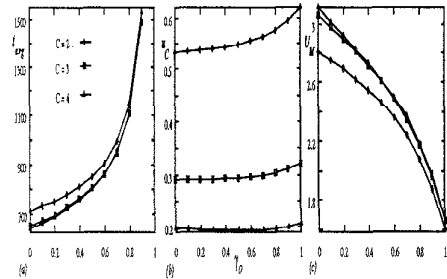


Figure 8: The performances of OIDS system by varying the $0.0 \leq \gamma_O \leq 1.0$. (a) The average transaction time per access, t_{avg} . (b) The utilization per data channel, u_C . (c) The total memory utilization, U_M .

Fig. 8 shows the OIDS system performance with varying the γ_O . Fig. 8(a) shows the average transaction time per access, t_{avg} , with varying the γ_O . The t_{avg} is increased when γ_O increases since the higher γ_O , the less possibility the local request can be sent out.

The T_{EM} decreases 10% from the number of channels $C = 2$ to $C = 3$ at $\beta = 0$, and decreases 4% from the number of channels $C = 3$ to $C = 4$ at $\beta = 0$, respectively. t_{avg} also shows the same results as T_{EM} . Fig. 8(b) shows the utilization per data channel, u_C , which has the same trend as Fig. 8(a). The higher γ_O , the more data traffic on the channels. On the contrary, U_M in Fig. 8(c) shows a different trend. The more time spent on a channel, the less time spent on memory, relatively.

Fig. 9 shows the OIDS system performance with varying the γ_I . Fig. 9 shows different results than Fig. 8 for t_{avg} . The higher γ_I , the greater the possibility that the data can be sent back. Fig. 9(a) shows the average transaction time per access, t_{avg} , with varying the γ_I . When $\gamma_I = 0$, t_{avg} approaches infinity, which means there is no possibility that the data can be sent back. Fig. 9(b) and (c) show the similar

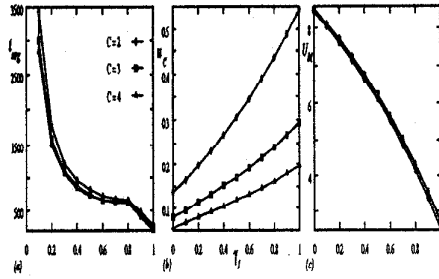


Figure 9: The performances of OIDS system by varying the $0.0 \leq \gamma_I \leq 1.0$. (a) The average transaction time per access, t_{avg} . (b) The utilization per data channel, u_C . (c) The total memory utilization, U_M .

results as Fig. 8(b) and (c).

4.5 Variation in memory service time and data channel service time

This section shows the impacts of memory service time and data channel service time on OIDS System performance.

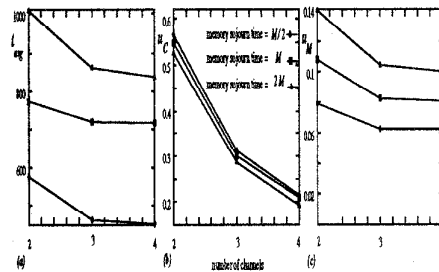


Figure 10: The performances of OIDS system by varying the memory service time $\in \{0.5, 1.0, 2.0\}M$. (a) The average transaction time per access, t_{avg} . (b) The utilization per data channel, u_C . (c) The memory module utilization per node, U_M .

Fig. 10 shows the performance of the OIDS system by varying the memory service time $\in \{0.5, 1.0, 2.0\}M$. Fig. 10(a), (b), and (c) show the average transaction time per access, t_{avg} , the utilization per data channel, u_C , and memory module utilization per node, u_M , with varying the memory service time $\in \{0.5, 1.0, 2.0\}M$. Fig. 10(c) shows that u_M increases 20% when memory service time is increased from $M/2$ to M and increases 42% when memory service time is increased from M to $2M$ and the number of channels is 2. T_{EM} and t_{avg} have the same results as u_M . u_C increases 3% when memory service time is decreased from M to $M/2$, and increases 5% when memory ser-

vice time is decreased from $2M$ to M .

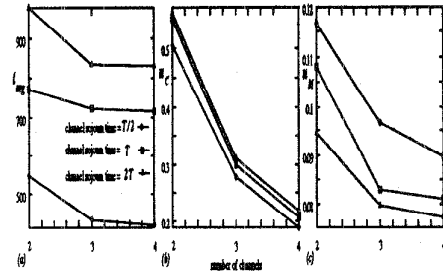


Figure 11: The performances of OIDS system by varying the data channel service time $\in \{0.5, 1.0, 2.0\}T$. (a) The average transaction time per access, t_{avg} . (b) The utilization per data channel, u_C . (c) The memory module utilization per node, U_M .

Fig. 11 shows the performance of the OIDS with varying the channel service time $\in \{0.5, 1.0, 2.0\}T$. It shows the similar results as Fig. 10.

This section shows the impacts of memory service time and data channel service time on OIDS system performance. Fig. 10 shows the performance of the OIDS system by varying the memory service time $\in \{0.5, 1.0, 2.0\}M$. Fig. 11 shows the performance of the OIDS system by varying the data channel service time $\in \{0.5, 1.0, 2.0\}T$. Fig.(a), (b), and (c) show the average transaction time per access, t_{avg} , the utilization per data channel, u_C , and the memory module utilization per node, U_M , respectively. Both figures show the same results for t_{avg} , but show different trends in u_C and U_M . The more service time in memory, the more the memory module utilization; the more service time in data channel, the more utilization of the data channel.

5. CONCLUSION

This paper has considered a passive optical wavelength-division multiplexed star-coupled structure to support interprocessor communication in a distributed shared memory environment. A media access control (TDMA-C) and a cache coherence protocol *snooping* are incorporated to provide access to a distributed arbitration of the WDM photonic network. Trace-based discrete-event simulation has been used to evaluate the performance.

A semi-Markov model is developed to study the performance of OIDS systems. The performance model was developed to reflect the impact of low-level optical media access issues on high-level distributed shared memory system performance, enabling the model to predict the performance impact with a WDM

access protocol and system parameters in the underlying photonic network. The performances were compared for the variations in the number of WDM channels, the block size, memory service time, channel service time, the variations in C .

6. REFERENCES

- [1] K. Bogineni and P. W. Dowd, "An optically interconnected distributed shared memory system: Architecture and performance analysis," *International Journal on High Speed Computing*, vol. 4, pp. 179–212, Sept. 1992.
- [2] P. W. Dowd and I.-S. Hwang, "Memory and network architecture interaction in an optically interconnected distributed shared memory system," *Journal of Parallel and Distributed Computing*, vol. 25, pp. 144–161, Mar. 1995.
- [3] I.-S. Hwang and P. W. Dowd, "Media access protocol impact on a WDMA passive star photonic network with distributed shared memory system," *Journal of Information Science and Engineering*, vol. 11, Sept. 1995.
- [4] I.-S. Hwang, "Snooping based cache coherence via a reservation based WDMA protocol for large-scale OIDS multiprocessors," *Journal of The Chinese Institute of Engineers*, vol. 18, Nov. 1995.
- [5] K. Bogineni and P. W. Dowd, "Performance analysis of two address allocation schemes for an optically interconnected distributed shared memory system," in *Proc. 6th International Parallel Processing Symposium*, pp. 562–566, Mar. 1992.
- [6] C. A. Brackett, "Dense wavelength division multiplexing networks: Principles and applications," *IEEE Journal on Selected Areas of Communications*, vol. 8, pp. 948–964, Aug. 1990.
- [7] P. W. Dowd, "High performance interprocessor communication through optical wavelength division multiple access channels," in *Proc. 18th International Symposium on Computer Architecture*, pp. 96–105, May 1991.
- [8] P. W. Dowd, "Wavelength division multiple access channel hypercube processor interconnection," *IEEE Transactions on Computers*, vol. 41, pp. 1223–1241, Oct. 1992.
- [9] K. Bogineni and P. W. Dowd, "A collisionless multiple access protocol for a wavelength division multiplexed star-coupled configuration: Architecture and performance analysis," *IEEE Journal on Lightwave Technology*, vol. 10, pp. 1688–1699, Nov. 1992.
- [10] A. Gupta, W.-D. Weber, and T. Mowry, "Reducing memory and traffic requirements for scalable directory-based cache coherence schemes," in *International Conference on Parallel Processing*, pp. 1–312–I–321, 1990.
- [11] B. W. O'Krafska and A. R. Newton, "An empirical evaluation of two memory-efficient directory methods," in *Proc. 17th International Symposium Computer Architecture*, (Seattle, Washington), pp. 138–147, May 1990.
- [12] D. Chaiken, C. Fields, K. Kurihara, and A. Agarwal, "Directory-based cache coherence in large-scale multiprocessors," *IEEE Computer*, pp. 49–58, June 1990.
- [13] A. Gupta and W.-D. Weber, "Cache invalidation patterns in shared-memory multiprocessors," *IEEE Transactions on Computers*, vol. C-41, pp. 794–810, July 1992.