OPTICAL INTERCONNECTS FOR MULTIPROCESSORS COST PERFORMANCE TRADE-OFFS

P. Lalwaney, L. Zenou, A. Ganz, and I. Koren

Department of Electrical and Computer Engineering
University of Massachusetts, Amherst, MA 01003

Abstract

Fiber optic interconnects based on wavelength division multiplexing (WDM) are a promising candidate for future interconnection networks due to their high bandwidth, low wire density and their low power requirements. As the cost of optical communication hardware for WDM star based interconnects may be large, we introduce reduced cost structures. The performance of the optical implementations of the reduced cost structures is compared to the electronic implementations for the hypercube topology. The performance is compared in terms of the communication overhead in implementing two commonly used algorithms on these structures. Our results indicate that in most situations, the optically implemented reduced cost variations perform better than the electronic implementations. Moreover, the hardware cost-performance trade-offs show that among the optically implemented schemes, the performance degradation of the reduced cost variations is not significant in view of the hardware savings involved.

1 Introduction

Future high performance computers will consist of hundreds and thousands of processors operating in parallel. It is imperative that such systems have efficient interconnection networks, so as to minimize the communication overheads. As individual processor data rates and complexities grow, electrical interconnects will not be able to support the speeds required by large multiprocessor systems. Further, packaging constraints and high wire densities at the backplane levels may lead to a data communication bottleneck in massively parallel systems. Optical interconnections have the potential to alleviate the problems of interconnect density while providing a very fast network for data communications [5]. Investigations into the possible incorporation of optical interconnections in multiprocessors have recently been reported [4][6]. For example, in a joint Honeywell/Thinking Machines Corp. project, optical fibers are replacing thousands of wires to connect parallel computers [7].

In this paper, we investigate the feasibility of wavelength division multiplexing star based interconnects for large multiprocessor systems. Wavelength division multiplexing (WDM) is one of the prevalent techniques used to exploit the large bandwidth of optical fibers [4][9]. WDM star based networks have been extensively investigated in the context of local area networks and switching fabrics [9]. Wavelength encoded signals from the transmitting nodes are multiplexed onto the fiber using a passive star coupler. Demultiplexing is performed at the receivers by recovering the desired input port signal from the common medium. The wavelength dimension provides the flexibility of designing any logical connectivity among the system nodes, independent of the physical topology (the WDM star). The logical connectivity is obtained by the assignment of wavelengths to the system's transmitters and receivers. Figure 1 illustrates the logical topology of a nine node 2-D torus realized by a WDM passive star coupler.

The main advantage of the WDM star based systems is the reduction in packaging problems due to the drastic decrease in the wire density. This reduction is obtained by multiplexing numerous channels onto a small diameter fiber. Further, low power requirements of WDM star based interconnects make this scheme very attractive. However, the main disadvantage of this scheme is the high cost of the optical communication hardware. In this paper, we propose a number of *reduced cost* topologies that would significantly decrease the total hardware cost. As many existing massively parallel systems have been implemented as hypercubes, we introduce three reduced cost variations of the hypercube topology. We study the performance of two commonly used algorithms by considering their communication requirements, the structure of the underlying hypercube topologies (electronically or optically implemented), the relative size of the problem to the available network size and the properties of the physical links in the network. Among the optically

0-8186-2772-7/92 $3.00 © 1992 IEEE

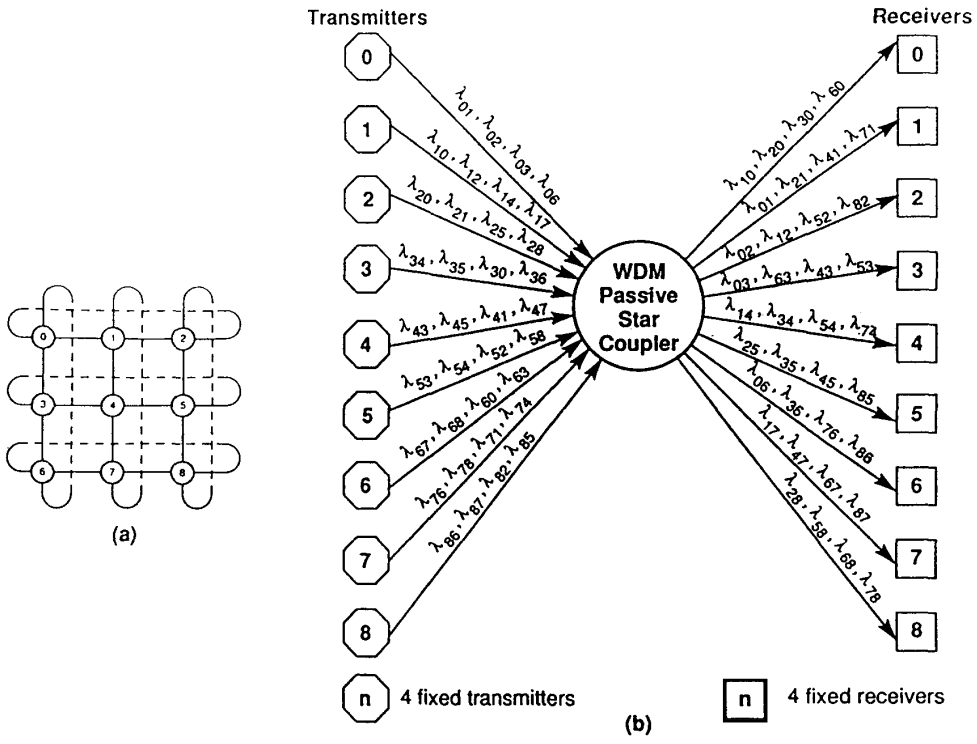

Figure 1: (a) A nine node 2-D torus; (b) WDM star embedding of the torus. λ_{ij} is the wavelength assigned to a transmitter of processor i and to a receiver of processor j.

implemented schemes, the hardware cost-performance trade-offs are studied.

Our investigation indicates that in most situations, the optically implemented reduced cost variations perform better than the electronically implemented topologies. Moreover, the hardware cost-performance trade-offs show that among the optically implemented schemes, the performance degradation of the reduced cost variations is not significant in view of the hardware savings involved. In Section 2, we introduce some reduced cost topologies for hypercubes. The performance of two hypercube based algorithms on these topologies is presented in Section 3. This is followed by a discussion on the obtained results and the cost-performance trade-offs for these systems.

2 Reduced Cost Variations of the Hypercube Topology

In this section, we study the variations of the hypercube topology and their properties. These variations are referred to as minimal, extended minimal and asymmetric incomplete hypercubes.

Fully Connected Hypercube

A fully connected hypercube consists of N nodes interconnected according to the hypercube topology. The degree of each node is denoted as n, where $N = 2^n$. An n dimensional hypercube has links along n levels. A link is said to be at *level* i if it connects two nodes whose binary addresses differ by 2^{i-1}. In a fully connected hypercube, the links at all the n levels are bidirectional. Let ω_{full} denote the number of wavelengths required to implement this system. Then, $\omega_{full} = n \cdot 2^n$. The diameter of the network is n and the average distance of the fully connected binary n-cube, denoted by $\tau_{full}(n)$, is $\tau_{full}(n) = \frac{n \cdot 2^{n-1}}{2^n - 1}$.

Minimally Connected Hypercube

A minimally connected hypercube is defined as a structure that can be implemented using minimum number of wavelengths with the constraints that the geometrical shape of the structure is preserved and that each node in the resulting network has the same

degree. These two constraints dictate that each edge of the hypercube has at least one unidirectional link. As a result

$$\frac{n \cdot 2^n}{2} \leq \omega(n) \leq n \cdot 2^n$$

where $\omega(n)$ is the total number of wavelengths required to implement this structure. Considering the above two constraints, $\omega(n)$ can assume $\lfloor \frac{n}{2} \rfloor + 1$ possible values. We define the *minimal structure* for a given n, as the regular structure built with the minimum number of wavelengths, denoted by $\omega_{min}(n)$, and given by: $\omega_{min}(n) = \lfloor \frac{n+1}{2} \rfloor \cdot 2^n$. The minimal structure for $n=3$ is depicted in Figure 2.

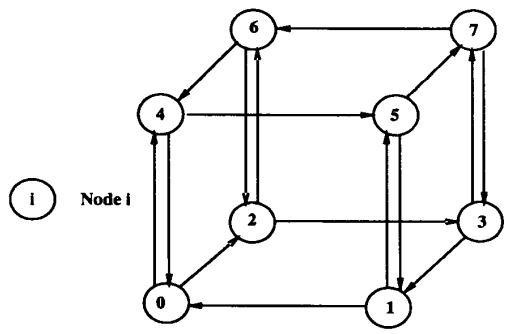

Figure 2: Minimal structure for an 8 node hypercube.

We illustrate the construction of an n-dimensional minimal hypercube using a recursive procedure. Assuming even n, an n dimensional minimal hypercube is obtained by interconnecting four $(n-2)$ minimal subcubes. The subcubes are interconnected in a clockwise or counter clockwise manner. We denote by CW-r and CCW-r, the clockwise and counterclockwise directions, respectively, for an r dimensional minimal hypercube. A CW-r structure is obtained when the first and the third $(r-2)$-subcubes are CW-$(r-2)$ structures, the second and fourth are CCW-$(r-2)$ structures, and the four $(r-2)$-subcubes are interconnected using clockwise directed links. This is illustrated in Figure 3a for a CW-4 structure. Links drawn by hollow arrows indicate the CW and CCW interconnections for the two dimensional subcubes. Note that the bold arrows represent the clockwise interconnections between corresponding nodes in the four subcubes to give the CW-4 minimal structure. To obtain a CCW-r structure, the first and third $(r-2)$-cubes are CCW-$(r-2)$ structures, the second and fourth cubes are CW structures, and the four $(r-2)$-subcubes are interconnected with counterclockwise links. This is illustrated in Figure 3b for a CCW-4 structure. We have proved that the CW-n minimal structure constructed using the above procedure has the minimum average distance. Also note that the above procedure can be applied to odd degree networks with the modification that all links in one dimension are implemented as bidirectional links. The minimal network can be obtained by applying the above recursive procedure to the remaining even degree network.

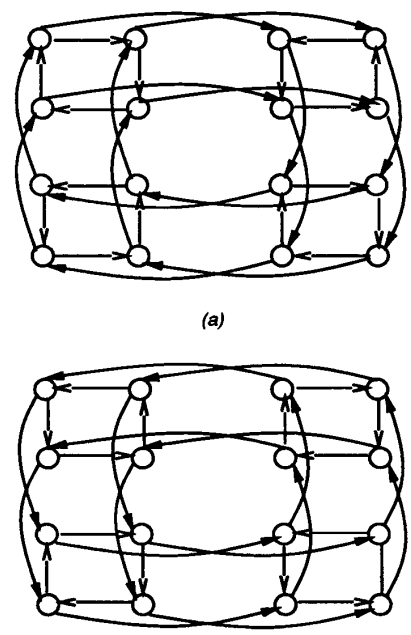

Figure 3: Construction of the minimal structure: (a) CW-4; (b) CCW-4.

Applying this recursive definition of the minimal scheme, we obtain the following closed form expression for the average distance with even n:

$$\tau_{min}(n) = \frac{n2^{n-1}}{2^n - 1} + \frac{2}{3} = \tau_{full}(n) + \frac{2}{3} \qquad (2.1)$$

Notice that while using *half* the number of links, the average distance is increased only by $\frac{2}{3}$ over the fully connected one. For example, for a 10 dimensional hypercube the average distance increases from 5.005 to 5.672 in its minimal implementation.

The minimal structure uses fewer wavelengths than a completely connected structure. A lower number of wavelengths reflects lower hardware costs, thereby making it an attractive scheme for optically intercon-

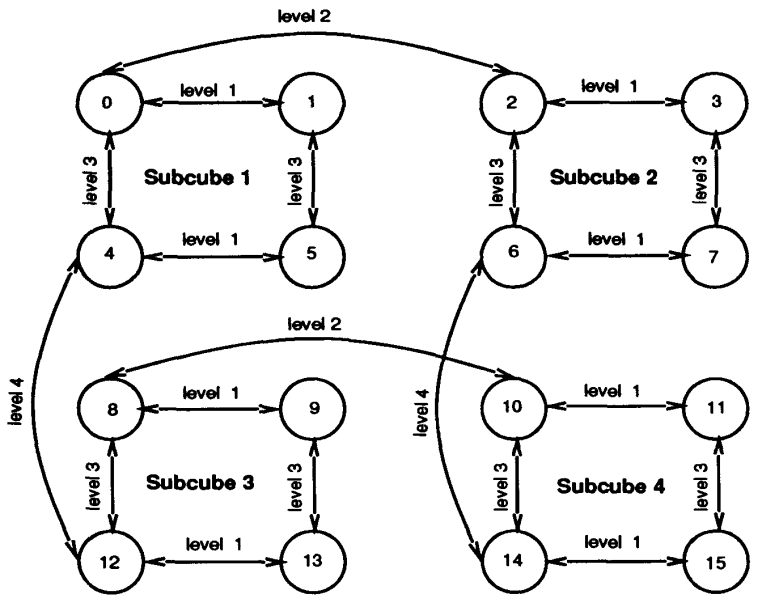

Figure 4: A 16 node Asymmetric Incomplete Hypercube (2,4) configured as four fully connected two dimensional subcubes.

nected multiprocessors. The performance penalty incurred due to the lower hardware costs can be seen as the increase in the average distance.

Extended Minimal Hypercube

An extended minimal hypercube of order (l, n) is a minimal hypercube of order n in which l of the n levels have bidirectional links. The total number of wavelengths needed to implement this structure is $w_{emin}(l, n) = 2^n \cdot (n/2 + l/2)$. The levels to be bidirected may be selected depending on the application to be run on the structure. As we will see in Section 3, judicious selection of levels to be bidirected, considerably increases the performance over the minimal structure at a small increase in hardware cost.

Asymmetric Incomplete Hypercube

The asymmetric incomplete hypercube attempts to exploit the locality of references frequently found in many applications. An asymmetric incomplete hypercube of order (l, n), consists of 2^{n-l} subcubes, where each subcube is a fully connected hypercube of order l. Thus, every node has at least l bidirectional links. These links are referred to as internal links as they are used for communication within a subcube. Communication between any two processors in distinct subcubes requires $(n - l)$ unidirectional or bidirectional links per processor in the fully connected, minimal and extended minimal hypercubes. In this scheme, we provide $(n - l)$ bidirectional links *per subcube* for external interconnections. The total number of wavelengths to implement this structure is

$$w_{aih}(l, n) = 2^n \cdot l + 2^{n-l} \cdot (n - l)$$

Every subcube has a designated processor that transmits and receives messages along the external links at a particular level. Note that the structure is not a regular one, as the degree of a node varies between l and $l + 1$.

Figure 4 shows one possible implementation of an asymmetric incomplete hypercube of order (2,4). In this implementation, links at levels 1 and 3 are internal, and links at levels 2 and 4 are external. As seen in Figure 4, processor 8 transmits (receives) messages at level 2 originating from (destined to) any processor in subcube 3. Communication at level 4 for processors in subcube 3 takes place through processor 12. The increase in the number of links traversed for communication at external levels comes at a considerable reduction in the hardware cost.

Figure 5 shows the variation in hardware cost as a function of the hypercube dimension, for the fully connected, minimally connected, extended minimal and asymmetric incomplete hypercube schemes. Four lev-

els were bidirected in the examined extended minimal and asymmetric incomplete hypercube schemes. As seen in this figure, a fully connected nine dimensional hypercube requires 4608 wavelengths, the minimal structure requires 2560 wavelengths (a 50% reduction) while the asymmetric incomplete (4,9) hypercube requires 2208 wavelengths (a 52% reduction). In the current technology, a single passive star can handle about a hundred nodes [9]. Multiple stars will have to be used for implementing networks with large number of nodes. The use of multiple stars for handling thousands of nodes has been demonstrated in switching systems [3].

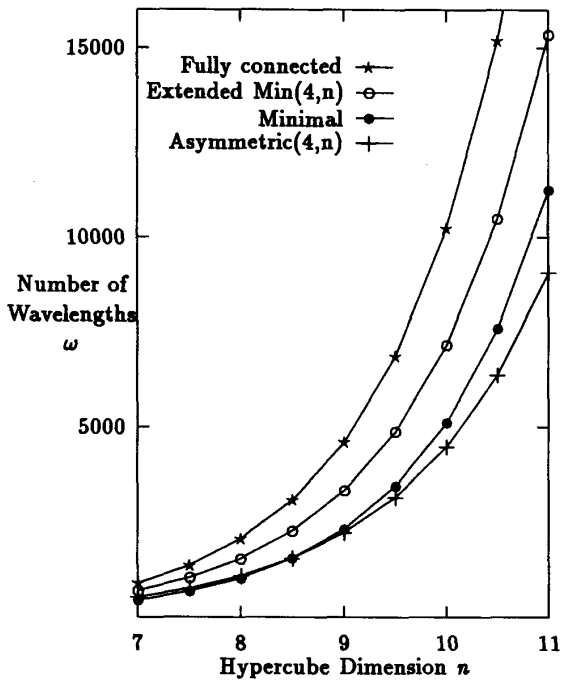

Figure 5: The required number of wavelengths for the four implementations.

3 Algorithm Performance - Hardware Cost Trade-offs

The performance of the reduced cost topologies is considered in this section. The algorithm completion time on a multiprocessor consists of the data processing time at each node and the communication time between nodes. In this section, estimates for the communication overhead are derived for implementing two commonly used algorithms, the bitonic sort and matrix multiplication, on the fully connected and reduced cost variations of the hypercube. In deriving expressions for the communication overheads we have considered the effect of the communication requirements of the algorithm, the communication structure offered by the topology, the relative size of the problem to the available network size and the properties of the link implementation. The latencies and speeds of the links are expressed as two parameters in the analysis. The link latency is denoted as α, and the time for a message transfer over a link is denoted as β. It should be noted that the link latency in optical implementations α_o, is greater than the link latency α_e, of the electronically implemented network. On the other hand, due to the speed advantage of optical interconnects, the time for a message transfer in the optical implementations β_o, is less than the time β_e for the electronic implementations.

Batcher's Bitonic Merge Sort

To sort an unsorted sequence of N elements, the merge sort procedure is applied recursively. Every two element sequence is bitonic. Proceeding from $N/2$ two element sequences, larger bitonic sequences are constructed and sorted using the algorithm presented in [2]. The complete sort of an N element sequence takes $\log N \cdot (\log N + 1)/2$ steps where each step involves data exchange and a compare operation. When performing the sort on a hypercube with $N = 2^k$ nodes, level i links are used in $(k - i + 1)$ steps [8]. This implies that the lower level links are used more often as compared to the higher level hypercube links.

We now consider the more likely case when the problem size $N = 2^k$, is greater than the number of available processors, $p = 2^m$; $m \leq k$. Each physical processor thus contains a list of 2^{k-m} elements. As the lowest level links are the most frequently used, these are mapped internal to the processor. Assuming sequential message transfer over the links, the required number of message exchanges over external level i links is $2^{k-m} \cdot (k - i + 1)$. If t_{int} denotes the time of an internal exchange and compare operation, then the time to sort 2^{k-m} elements in a processor equals $(k - m) \cdot (k - m + 1)/2 \cdot t_{int}$. The algorithm communication time for the four schemes was calculated and the results are summarised below:

$$T_{full} = \frac{m(m+1)}{2} \cdot \alpha + 2^{k-m} \cdot \frac{m(m+1)}{2} \cdot \beta + \frac{(k-m)(k-m+1)}{2} \cdot t_{int} \qquad (3.1)$$

$$T_{min} = 3 \cdot \frac{m(m+1)}{2} \cdot \alpha + 3 \cdot 2^{k-m} \cdot \frac{m(m+1)}{2} \cdot \beta$$

$$+\frac{(k-m)(k-m+1)}{2}\cdot t_{int} \qquad (3.2)$$

In the extended minimal scheme, if the lowest l levels of the minimal m cube are made bidirectional, then,

$$T_{emin} = (m(m+1)/2 + (m-l)(m-l+1))\cdot\alpha$$

$$+2^{k-m}\cdot(m(m+1)/2 + (m-l)(m-l+1))\cdot\beta$$

$$+\frac{(k-m)(k-m+1)}{2}\cdot t_{int} \qquad (3.3)$$

When implementing the algorithm on an asymmetric incomplete hypercube of order (l, m) the communication time is bounded by the following expression:

$$T_{aih} \leq (m(m+1)/2 + l(m-l)(m-l+1))\cdot\alpha$$

$$+2^{k-m}\cdot(m(m+1)/2 + l(m-l)(m-l+1))\cdot\beta$$

$$+\frac{(k-m)(k-m+1)}{2}\cdot t_{int} \qquad (3.4)$$

Note that the subscripts *full, min, emin* and *aih* refer to fully connected, minimally connected, extended minimal and asymmetric incomplete hypercube structures, respectively. The first term in each of the above expressions is the overhead due to the communication setup time. This term includes the optical-electronic and the electronic-optical signal conversion times. The second term represents the sum of message transmission times and the last term is the time for internal transfers.

We compared the communication time for the optical implementation of the above four schemes to that of an electronic fully connected hypercube. Figure 6 depicts the variation in the total communication time required to sort lists of length 2^{12} up to 2^{20} on a hypercube with 2^{10} nodes. Here we assumed that for optical implementations, the time for message transfers over external links is a factor of 5 lower than the internal communication time. For the curves in Figure 6, we have expressed the values of α and β in terms of t_{int}, and assumed that t_{int} equals one time unit. It should be noted that the latencies of optically implemented topologies is higher than their electronic counterparts. The ratio of the latency of the electrical to the optical implementation is taken as 0.2 [1]. We also assumed that the link speeds of optical networks are ten times greater than that of the electronic implementations. Even with these conservative estimates, the optical implementations perform better than the electronic ones for large problem sizes. When the size of the list in a single processor is small, the volume

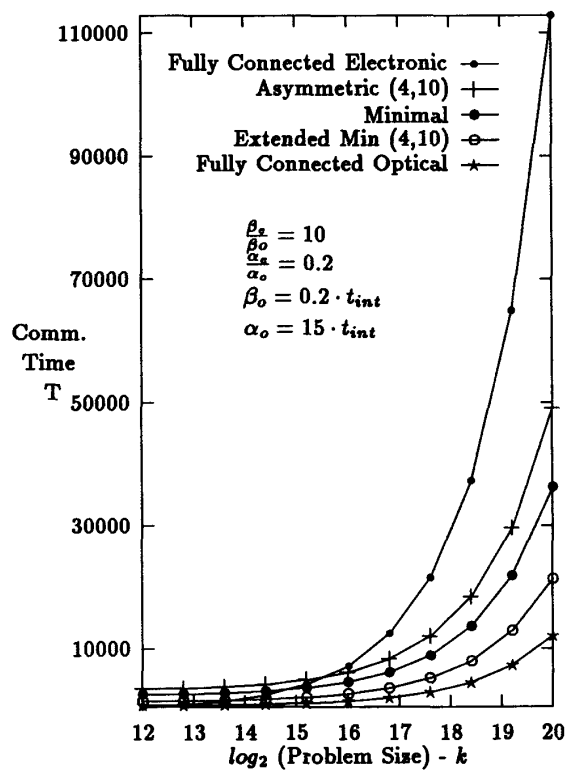

Figure 6: The communication time in the bitonic sort algorithm for five implementations as a function of the problem size on a 2^{10} node hypercube.

of data transferred between nodes and the number of external exchanges is small. The high latency of the optical interconnect dominates, and consequently, the electronic implementation performs better. For the set of α and β parameters used in Figure 6, optical implementations perform better than the fully electronic hypercube implementations when the size of the list per node is larger than 64.

As an example, it takes 14273 time units to sort a list of 2^{17} elements on the ten dimensional cube using the electronic fully connected network. It takes 9081 time units on the optically implemented asymmetric incomplete (4,10) hypercube (36.37% reduction), 6727 time units on the optical implementation of the minimal scheme (53% reduction), 3966 time units on the optical implementation of the extended minimal (4,10) hypercube (72% reduction) and 2261 time units (84% reduction) on the fully connected optical hypercube.

Matrix Multiplication

A divide and conquer algorithm is considered for the multiplication of two $N \times N$ matrices. The algorithm recursively divides a $d \times d$ matrix into four submatrices of order $d/2 \times d/2$ until the submatrices correspond to a single element. The 2×2 matrix multiplication algorithm is applied at each step. The details of the algorithm are presented in [8].

If $k = \log N$, and a hypercube of dimension $2k$ is used to implement the above algorithm, every processor contains a single data element of the matrices being multiplied. If $C = A \times B$, where A and B are $N \times N$ matrices, then the recursive division corresponds to the bidirectional exchange of the A matrix coefficients over the horizontal links (at levels 1 through k) and the B matrix coefficients over the vertical links (at levels $k+1$ through $2k$). Assuming simultaneous exchange of data in the horizontal and vertical directions, the number of messages over level i links equals the number of messages over level $(k+i)$ links. This number equals 2^{k-i}, $1 \leq i \leq k$. Note that links at the lowest levels in the horizontal and vertical directions have the highest traffic. After all the data exchanges, the computation phase begins. This involves N multiplications and N additions in each processor.

As in the previous algorithm, we consider the communication time for the general case, when the problem size is larger than the available number of processors. We denote the number of available processors by $p = 2^m$; $m \leq 2k$. In this case, each physical processor contains 2^{2k-m} logical nodes. This corresponds to $(k-m/2) \times (k-m/2)$ submatrices stored in each physical node. As before, we assume sequential transfer of messages between nodes. The expressions for the total communication times of the algorithm for the four schemes considered are summarized below:

$$T_{full} = m \cdot \alpha + (2^{2k-m/2} - 2^{2k-m}) \cdot \beta + (2^{k-m/2} - 1) \cdot t_{int} \quad (3.5)$$

$$T_{min} = 4m \cdot \alpha + 4 \cdot (2^{2k-m/2} - 2^{2k-m}) \cdot \beta + (2^{k-m/2} - 1) \cdot t_{int} \quad (3.6)$$

In the extended minimal scheme, a cube of order $(2l, m)$ was considered. The lowest l levels in the horizontal and vertical directions of the minimal m cube were made bidirectional. The communication time is then given by the following equation:

$$T_{emin} = (4m - 3l) \cdot \alpha + 4 \cdot (2^{2k-m/2-l} - 2^{2k-m}) \cdot \beta + (2^{2k-m/2} - 2^{2k-m/2-l}) \cdot \beta + (2^{k-m/2} - 1) \cdot t_{int} \quad (3.7)$$

When implementing the algorithm on an asymmetric incomplete hypercube of order $(2l, m)$, with the lowest l levels in the horizontal and vertical directions bidirected, the communication time is bounded by the following expression:

$$T_{aih} \leq m \cdot (4l+1) \cdot \alpha + (4l+1) \cdot (2^{2k-m/2-l} - 2^{2k-m}) \cdot \beta + (2^{2k-m/2} - 2^{2k-m/2-l}) \cdot \beta + (2^{k-m/2} - 1) \cdot t_{int} \quad (3.8)$$

Figure 7 shows the variation in communication time for multiplying two $N \times N$ matrices (problem size equals N^2), on a ten dimensional hypercube. The

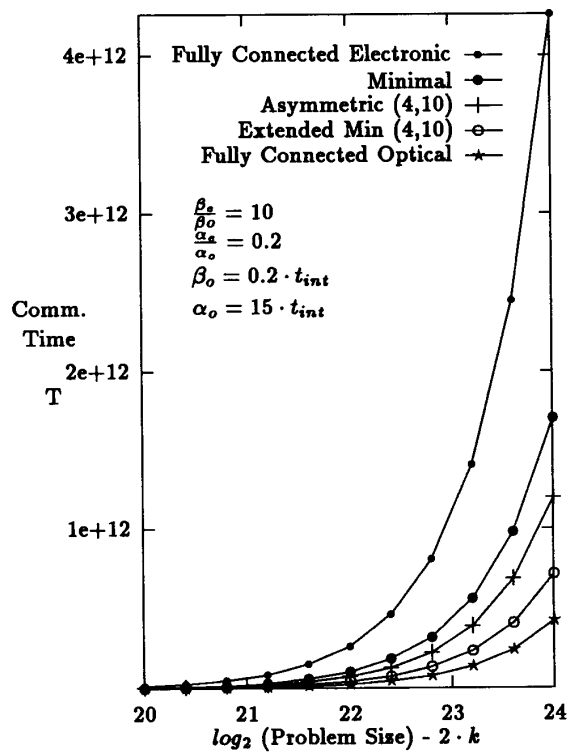

Figure 7: The communication time in the matrix multiplication algorithm for five implementations as a function of the problem size on a 2^{10} hypercube.

total communication time for the optical implementations of the fully connected, minimal, extended minimal and asymmetric incomplete hypercubes are compared against the electronically fully connected hypercube. The parameters used in the sorting application are also used in Figure 7. It is observed that the reduced structures outperform the fully electronic scheme. As an example, multiplication of two

2048×2048 matrices (problem size 2^{22}), on a ten dimensional cube takes 26.6×10^{10} time units in the electronic fully connected network. It takes 10.65×10^{10} time units on the minimally connected hypercube, 4.03×10^{10} time units on the asymmetric incomplete (4,10) hypercube, 2.92×10^{10} time units on the extended minimal (4,10) hypercube, and 2.66×10^{10} time units on the optical fully connected hypercube. These numbers represent a reduction of 60% for minimally connected, 85% for the asymmetric incomplete hypercube, 89% for the extended minimal and 90% for the optical fully connected hypercube over the electronic fully connected hypercube.

The communication requirements of the two algorithms were different. In the sorting application, every compare operation was followed by a data exchange operation. This required the repeated use of links at a particular level in many phases of the algorithm. In the matrix multiplication case, at every level of recursion, the number of messages to be transferred increased. Links at a particular level need to be used in one phase of the algorithm. Thus, in the sorting application, the link latency was an important factor whereas in the matrix multiplication case, the speed difference in the electrical and optical implementation dominates the results. The effect of the high latency for optical networks was seen in the sorting application, as a minimum problem size was required for the optical networks to perform better than the electronic ones. We observe that in the two algorithms considered in this section, the larger the problem size (relative to the system size), the larger the performance gain when using optical interconnects. This is due to the fact that the number of messages transferred among the processors increases with the problem size, allowing the exploitation of the higher bandwidth of the optical interconnect. The performance of the network is sensitive to the ratio β_o/β_e. Reducing the ratio β_o/β_e, further enhances the performance of the optically implemented hypercubes over the electronic hypercube. As can be seen from Figures 5, 6 and 7, the optical fully connected network has the best performance and the highest hardware cost. A substantial performance improvement is still achieved for the partially connected structures, which considerably reduce the hardware cost of the optical system.

4 Conclusions

We have demonstrated the performance advantages of WDM based optical interconnects in the face of partial structures dictated by the hardware restrictions of the currently available technology. We have shown that the performance gains of the reduced cost optical implementations are a function of (1) the ratio between the size of the problem and the size of the multiprocessor system, (2) the communication requirements of the algorithm, (3) the hardware constraints in terms of the number of available wavelengths, link latencies and link speeds. We have shown that optically implemented reduced cost structures outperform the electronically implemented complete structures in the case of the hypercube topology. The analysis of the hardware cost-performance trade-offs strongly favor the reduced cost optical implementations over the complete optical implementations.

References

[1] P.J. Ayliffe, J.W. Parker and A. Robinson, "Comparison of Optical and Electrical data Connections at the Board and Backplane Levels," *Optical Interconnections and Networks*, Hartmut Bartelet, Editor, Proc. SPIE 1281, pp. 2-15, 1990.

[2] K.E. Batcher, "Sorting Networks and their Applications," *Proc. AFIPS Spring Joint Computer Conf.*, pp. 307-314, 1968

[3] A. Cisneros and C. A. Brackett, "A large ATM Switch Based on Memory Switches and Optical Star Couplers," *IEEE J. on Selected Areas in Communications*, Vol. 9, pp. 1348-1360, Oct. 1991.

[4] P. W. Dowd, "High Performance Interprocessor Communication Through Optical Wavelength Division Multiple Access Channels," *Computer Architecture*, pp. 96-105, May 1991.

[5] J.W. Goodman, F.J. Leonberger, S.Y. Kung and R.A. Athale, "Optical Interconnections for VLSI Systems," *Proc. of the IEEE*, Vol. 72, July 1984.

[6] A. Guha, J. Bristow, C Sullivan and A. Husain, "Optical Interconnections for Massively Parallel Architectures," *Applied Optics*, Vol. 29, No. 8, March 1990.

[7] B.O. Kahle, E.C. Parish, T.A. Lane and J.A. Quam, "Optical Interconnects for Interprocessor Communications in the Connection Machine," *IEEE Conf. on Computer Design*, Oct. 1989.

[8] P. A. Nelson, L. Snyder, "Programming Solutions to the Algorithm Contraction Problem," *International Conference on Parallel Processing*, pp. 258-261, Aug. 1986.

[9] Special Issue on "Dense Wavelength Division Multiplexing Techniques for High Capacity and Multiple Access Communication Systems," *IEEE J. on Selected Areas in Communications*, Aug. 1990.