Parallel Optical Window Algorithm Applied to Optical Multistage Interconnection Network

[1]Mohamed Othman, [2]Monir Abdullah & [3]Rozita Johari

[1]Laboratory of Computational Sciences and Informatics
Institute for Mathematical Research
Universiti Putra Malaysia

[1, 2, 3]Department of Communication Technology and Networks
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia

[1]mothman@fsktm.upm.edu.my

Abstract

The crosstalk problem is introduced in an optical multistage interconnection network caused by coupling two signals within a switching element. To avoid this crosstalk, a time domain approach is used, which is to partition the set of permutation connections into several subsets such that the connections in each subset can be established simultaneously in the network without crosstalk. Since we want to partition the messages to be sent to the network into several groups, we have to use the window method that is used for finding the conflicts among all the messages to be sent. In this paper, a new parallel algorithm of the window method is developed called the Balanced Parallel Window Method (BPWM) algorithm. The BPMW algorithm reduces the execution time by a percentage of 83% of the time compared to the sequential algorithm with seven processors.

Introduction

Multistage Interconnection Network (MIN) has been used in telecommunication and parallel computing systems for many years. As optical technology advances, there is a considerable interest in using optical technology to implement interconnection networks and switches [1,7]. A major problem called crosstalk is introduced by Optical MIN (OMIN), which is caused by coupling two signals within a Switching Element (SE). To reduce the negative effect of crosstalk, many approaches have been proposed. One way to solve crosstalk is to use a $2N \times 2N$ regular OMIN to provide a number of $N \times N$ connections [4], which is the space domain approach. However, half of the inputs and outputs are wasted in this particular approach. Another efficient solution is to route the traffic through an $N \times N$ optical network to avoid coupling of two signals by allowing only one signal be propagated within each switching element. This idea can be implemented using the time domain approach i.e. [1,2], where all the inputs are routed in several groups such that there will be no crosstalk between messages in each group. Since the messages to be sent to the network are to be distributed into several groups, a method is to be used to find out which messages should not be in the same group in the routing table since they will cause crosstalk. The window method is used to determine conflicts among all the messages to be sent. This method has already been proved their correctness by many researchers [4,7].

In this research, we are interested in a network called the Omega Network (ON), which has the shuffle-exchange connection pattern. Since many other topologies are equivalent to omega topology, performance results obtained for ON are also applicable to other MINs [5].

The rest of the paper is organized as follows. Section 2 describes the crosstalk problem in OMIN, omega network and window method. In Section 3, the BPWM algorithm is presented while Section 4 presents the experimental results. Conclusion is made in the last section.

Optical MIN

Crosstalk in Optical MIN

Fiber optic communications promise to meet the increasing demand of communication systems,

and received much attention in the parallel processing community [1]. Although OMIN has great promise and has some advantages over the electronic MIN, it leads to some other problems. One of the problems is optical crosstalk arises in OMIN. This crosstalk occurs when two signal channels interact with each other within a single SE. When crosstalk happens, a small fraction of the input signal power may be detected at another output although the main signal is injected at the right output. For this reason, when a signal passes many SEs, the input signal will be distorted at the output due to the loss and crosstalk introduced along the path [1].

Optical Omega Network (OON)

The OON connects each input and output devices through a number of switch stages. It is very useful for building parallel computers with hundreds of processors. This network consists of N inputs, N outputs, and S stages (S=$\log_2 N$). Each stage has $N/2$ SEs; with each SE having two inputs and two outputs connected in a certain pattern. Figure 1 shows a network of size N (N=8) sending all the eight inputs to the eight outputs in one time slot.

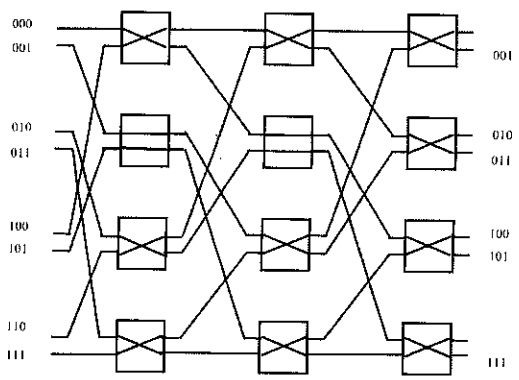

Figure 1: 8×8 Omega Network

Window Method (WM)

To distribute the messages to be sent to the network into several groups, we have to use a method to find out which messages should not be in the same group because they will cause crosstalk. In [1], WM is used for finding conflicts among all the messages to be sent. The combination of each source address and its corresponding destination address will produce a combination matrix. The optical window size is defined as M-1, where M=$\log_2 N$ and N is the size of the network. This window is applied to the produced matrix from left to right except the first column and last column. Messages that have the same bit pattern in any of the optical windows are noted, since they will cause conflict in the network.

Proposed BPWM Algorithm

Load balancing is a technique of dividing several tasks to the available processors, equally. This can easily be done using the same operations being performed by all the processes [6]. The BPWM algorithm is proposed to solve the unbalancing problem. An independent problem is found in the window itself. The problem is divided into small subproblems. In other words, each window is divided into many SubWindows (SWs). In each SWs, there is a conflict and each window in network size 8×8 can be divided into seven independent SWs. In general, each window in the network size N×N can be derived into N-1 SWs. The division of a particular window is made according to its contents, as in the figure 2.

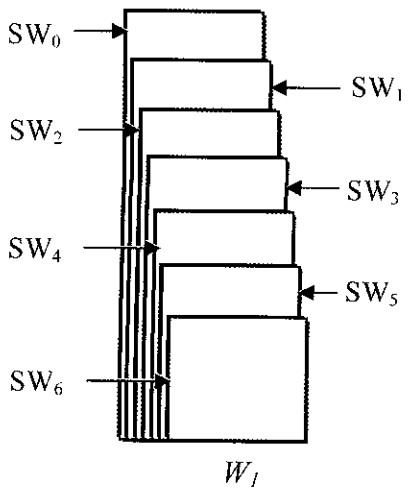

Figure 2: Window Decomposition in the BPWM Algorithm

Now each SW can be assigned to the slave processors. In this case, the communication overhead will be increased but a broadcast command is used to reduce this overhead, arising another problem in how the slave can determine the SWs. To solve this problem, the master processor will send a flag to each slave to determine the demand SWs. When the master processor sends a flag to the slave processor, thus yield communication overhead increase. For

solving such problems the master process will only send the flag in the beginning to all the slaves to start their work only after broadcasting the combination matrix.

In this algorithm, the slaves themselves recognize their subwindows using the same subwindow algorithm. Master process will broadcast the whole combination buffer to the slaves. In turn, the slave will do the comparison of all their subwindows at one time. Eventually, they will send the final results to the master again. As compared to the last strategy, communication overhead in this strategy is significantly lower. All processes in the communicator including master process work in this algorithm.

Experimental Results & Discussions

The algorithm is implemented on the Sun Fire V1280 machine with eight processors and network sizes of 512, 1024, and 2048 nodes. The running time with the number of processors is shown in the next figure 3.

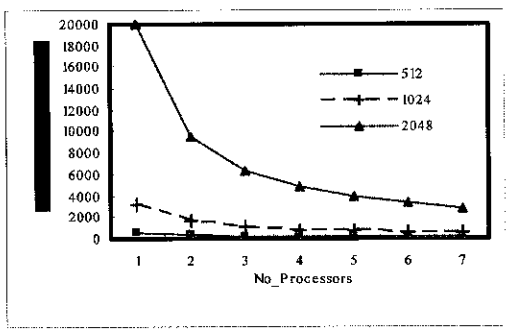

Figure 3: Running Time of the BPWM Algorithm

The time is reduced by 83%, decomposing the computation into seven processors with network size of 2048 nodes in the network. The speedup of BPWM algorithm is calculated according to the better sequential algorithm, without taking into account the initialization and completion phases. In fact, the main contribution of this algorithm is having achieved the super linear speedup as illustrated in figure 4.

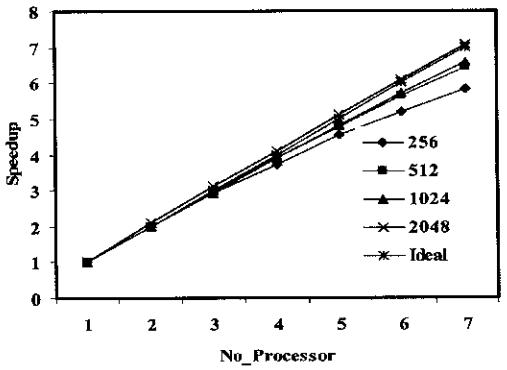

Figure 4: Speedup of the BPWM Algorithm

The main contributions of BPWM with low communication overhead are the master process works the same as any slave and there is a super linear speedup with the specific network size 2048 for the following reason: when size of network is huge (N=2048), the combination and conflict matrices are huge as well (2^{22} bytes for each). The SWM algorithm swaps in cache memory with conflict matrices buffer and combination matrix buffer. However in parallel, the combination matrix is divided among processors. As a result, a small buffer size on every processor is needed to save the conflict data and the master process does not need this matrix. The BPWM is empirically shown to be both efficient and scalable in terms of speedup for the given number of available processors.

Conclusion and Future Work

In this research, the balanced parallel window method has successfully reduced 83% of time compared to the sequential algorithm. Efficient message routing algorithms directly affected the performance of communication networks. Our algorithm is more scalable when the network size increased.

References

[1] Varma, A. and Raghavendra, C.S. 1994. *Interconnection networks for Multiprocessors and Multicomputers: Theory and Practice*, IEEE Computer Society Press.

[2] Gu, Q.P. and Peng, S. 2000. Wavelengths requirement for permutation routing in all-optical multistage interconnection networks, *Proceedings of the 2000 International Parallel and Distributed Processing Symposium*, Cancun, Mexico, 761-768.

[3] Pan, Y., Qiao, C. and Yang, Y. 1999. Optical Multistage Interconnection Networks: New Challenges and Approaches. *IEEE Communications Magazine, Feature Topic on Optical Networks, Communication Systems and Devices* **37**(2): 50-56.

[4] Padmanabhan, K. and Netravali, A.N. 1987. Dilated networks for photonic switching. *IEEE Trans. Communications* **35** (12):1357-1365.

[5] Wilkinson, B. and Allen, M. 2000. *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*. NJ 07458, USA, Prentice Hall.

[6] Abed, F. and Othman, M. 2007. A Review of Message Routing and Scheduling Algorithms in Omega Networks, *Conference on Information Technology Research and Applications*, Selangor, Malaysia: 291-296.