High-Speed Crosstalk-Free Routing for Optical Multistage Interconnection Networks

Enyue Lu and S. Q. Zheng
Department of Computer Science
University of Texas at Dallas
Richardson, TX 75083-0688, USA
{enyue, sizheng}@utdallas.edu

***Abstract*— Multistage interconnection networks (MINs) can be used to construct electro-optic switches. To implement crosstalk-free switching in such a switch, two I/O connecting paths cannot share a common switching element (SE). Thus, a permutation must be decomposed into partial permutations, each being routed through the switch without crosstalk. It was shown that any permutation can be decomposed into two semi-permutations, and each is a maximum partial permutation realizable in one pass in an optical Benes network. However, the time complexity of existing decomposition algorithms for realizing connection requests is proportional to permutation size. In this paper, we reexamine the permutation capacity of MINs, present a simpler proof for semi-permutation decomposability, and propose a parallel decomposition algorithm of logarithmic time. This algorithm is shown useful for optimally routing crosstalk-free paths in optical Benes networks in high-speed.**

***Index Terms*— Optical multistage interconnection networks (OMINs), crosstalk, semi-permutation, parallel algorithm, graph coloring.**

I. Introduction

The explosive growth of Internet is driving an increased demand for transmission rate and faster switching technologies. Optical communications with photonic switching promise to meet high bandwidth, low error probability, and large transmission capacity. To build a large IP router with capacity of 1 Tb/s and beyond, MINs will be used. Such networks can be realized either all electronically or with the introduction of optical switching. Optical routers will have better scalability than electronic routers in terms of switching capacity. Unfortunately, the required optical technologies are immature for all-optical switching to happen anytime soon. A hybrid approach in which optical signals are switched, but both switch control and routing decisions are carried out electronically, becomes more practical. Advances in electro-optic technologies provide a promising choice to meet the increasing demands for high channel bandwidth and low communication latency.

A hybrid OMIN can be built from 2×2 electro-optic SEs such as common lithium-niobate ($LiNbO_3$) SE (e.g. [3], [4], [22]). Each SE is a directional coupler with two inputs and two outputs. Depending on the amount of voltage at the junction of the two waveguides, optical signals carried on either of two inputs can be coupled to either of two outputs. An electronically controlled optical SE can have switching speed in the range from hundreds of picoseconds to tens of nanoseconds [18]. However, large OMINs built from integrating these electro-optic SEs have the problem of *crosstalk*, which is caused by undesired coupling between signals of the same (close) wavelength(s) carried in two waveguides so that two signal channels interfere with each other. Figure 1 shows an example of crosstalk in an SE. An SE has two logic states, namely, *straight* and *cross* (see Figure 1 (a)). For the straight state, a small fraction of input signal injected at the upper input may be detected at the lower output (see Figure 1 (b)). Crosstalk can also occur when an SE is in the cross state. Consequently, the input signal will be distorted at output due to loss and crosstalk accumulated along connection path.

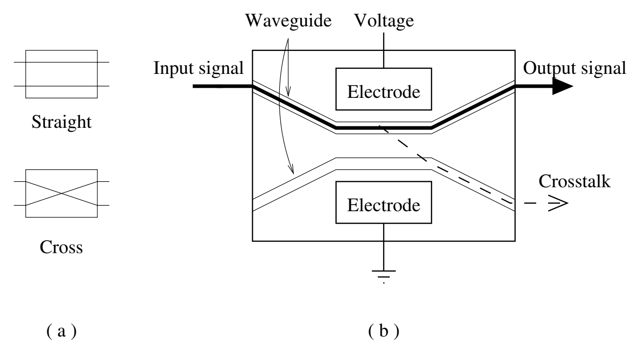

Fig. 1. Crosstalk in an electro-optic SE.

In order to reduce the crosstalk effect, three approaches, space dilation, time dilation and wavelength dilation, have been proposed. In both space and time dilation, crosstalk can be eliminated by ensuring that only one signal pass through an SE at a time. In other words, only one input and one output of an SE is used at any time instance. In a space dilation approach, an $N \times N$ OMIN is dilated into a network that is essentially equivalent to a $2N \times 2N$ network (e.g. [9], [11], [23], [24]). The space dilation trades the hardware cost that is more than 2 times of regular OMINs to achieve the same permutation capability. In a time dilation approach, crosstalk can be avoided by using the principle of *reconfiguration with time division multiplexing* (RTDM) paradigm proposed by C. Qiao *et al.* in [14]. More specifically, a set of permutation connections is partitioned into subsets so that the connections in each subset can be established simultaneously without crosstalk and the subsets can be used to form a sequence of configurations for the set of connections. Such a subset

0-7803-7945-4/03/$17.00 (C) 2003 IEEE

is called a *crosstalk-free (CF) partial permutation*. Since the paths realizing a CF partial permutation for a given OMIN do not share any SE, the time dilation approach is also useful for establishing a set of connections that would normally cause conflicts in blocking OMINs such as Banyan networks [12], [15], [21]. In the wavelength dilation approach, the crosstalk between two signals passing through the same SE is suppressed by ensuring two wavelengths to be far apart by routing (e.g. [19], [20]), or by using wavelength converters (e.g. [16], [17]), which limits the efficiency of bandwidth utilization and/or increases cost and complexity.

In this paper, we focus on how to quickly configure an OMIN for realizing permutations using the approach of time dilation. A special type of partial permutation, named *semi-permutation*, has demonstrated the maximum potential to be realized in OMIN in one pass without crosstalk [25], [26]. It was shown that for the optical Benes network, any semi-permutation is realizable in one pass and any permutation can be routed in two passes, which is the minimum number of passes needed for a permutation in an OMIN. However, the existing permutation decomposition algorithms have $O(N)$ time complexity and the crosstalk-free routing algorithms in an optical Benes network take $O(N \log N)$ time[1], which is too time-consuming even for circuit switching in optical domain. In this paper, we reexamine the permutation capacity of OMINs, show the decomposability of semi-permutation based on edge coloring of a bipartite graph, and propose a fast parallel decomposition with time complexity $O(\log K)$ for avoiding crosstalk in OMINs. We then present a routing algorithm based on our parallel decomposition with improved time complexity $O(\log^2 K + \log N)$ for realizing any one-to-one I/O mapping with $K(\leq N)$ active inputs in an $N \times N$ optical Benes network.

II. Decomposing a Permutation into Two Semi-Permutations

In this section, we will present our main result, a parallel decomposition algorithm to decompose a permutation into two semi-permutations. This decomposition algorithm is equivalent to find a 2-edge coloring of a bipartite graph where every vertex has degree of 2.

A. Decomposability

Let $I = \{I_0, I_1, \cdots, I_{N-1}\}$ and $O = \{O_0, O_1, \cdots, O_{N-1}\}$ be the sets of inputs and outputs, respectively, of an $N \times N$ OMIN. Let $\pi : I \longmapsto O$ be an I/O *mapping* that indicates connection requests from inputs to outputs. Input I_i is *active* if there is a connection request from I_i to O_j, and in this case, $\pi(i) = j$ and $\pi^{-1}(j) = i$.

An I/O mapping π is a *permutation* if π is a one-to-one I/O mapping and all inputs are active. A one-to-one I/O mapping involving $K(< N)$ active inputs is called a *partial permutation*. Clearly, a permutation can not be realized in a single pass in an $N \times N$ OMIN without crosstalk. Hence, we are interested in a type of partial permutation that can be passed through OMIN without crosstalk. Y. Yang *et al.* [25] introduced a concept called *semi-permutation*, which is a partial permutation that ensures only one active input in each SE of the first and last stages of an OMIN at the same time. Formally, we have the following definition.

Definition 1: For any permutation π of $\{0, 1, \cdots, N-1\}$, a partial permutation with $N/2$ active inputs, $x_0, x_1, \cdots, x_{N/2-1}$, is called a *semi-permutation* of π, denoted as π^s, if it satisfies:

$$\{\lfloor x_0/2 \rfloor, \lfloor x_1/2 \rfloor, \cdots, \lfloor x_{N/2-1}/2 \rfloor\} =$$
$$\{\lfloor \pi(x_0)/2 \rfloor, \lfloor \pi(x_1)/2 \rfloor, \cdots, \lfloor \pi(x_{N/2-1})/2 \rfloor\} =$$
$$\{0, 1, \cdots, N/2-1\}.$$

□

Clearly, a semi-permutation is a maximum potential partial permutation that can be realized in one pass of an $N \times N$ OMIN built with 2×2 SEs.

For any permutation π with N inputs, we can construct a bipartite graph G, named *I/O mapping graph*, as follows. The vertex set consists of two parts, A and B. Each part has $N/2$ vertices corresponding to I (resp. O), i.e. a pair of two inputs (resp. outputs) $2i$ and $2i+1$ with $i \in \{0, 1, \cdots, N/2-1\}$, called *dual* inputs (resp. outputs), is represented by a vertex in A (resp. B). There is an edge between vertex $\lfloor i/2 \rfloor$ in part A and vertex $\lfloor j/2 \rfloor$ in part B if $j = \pi(i)$. An I/O mapping graph may consist of parallel edges, which have the same ends. However, there is a one-to-one correspondence between inputs/outputs in a permutation and edges in an I/O mapping graph G. Hence, we can label each edge by its corresponding input/output. Since each vertex in G represents two inputs/outputs, the degree of each vertex in G is 2. Consider the following lemma proved in [2].

Lemma 1: Every bipartite graph G is $\Delta(G)$-edge colorable, where $\Delta(G)$ is the maximum degree of vertex in G. That is, we can color edge set $E(G)$ with $\Delta(G)$ colors so that the adjacent edges have different colors.

By Lemma 1, we can establish the following theorem.

Theorem 1: Any permutation can be decomposed into two semi-permutations.

Proof: We know that the I/O mapping graph constructed from a permutation is 2-edge colorable by Lemma 1. If we color $E(G)$ with two colors, then two edges with ends corresponding two dual inputs/outputs incident at a vertex in G must have different colors. Thus, each of the subgraphs induced by the edges of the same color contains all of vertices in $V(G)$ and half of edges in $E(G)$. Therefore, each subgraph is corresponding to a semi-permutation. □

Example 1: For the permutation

$$\pi = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & 6 & 0 & 5 & 3 & 2 & 7 & 4 \end{pmatrix}$$

a 2-edge coloring of its corresponding I/O mapping graph is shown in Figure 2, with one color represented by solid edges and the other by dashed edges. The solid edges correspond to the semi-permutation

$$\pi_1^s = \begin{pmatrix} 0 & 3 & 4 & 6 \\ 1 & 5 & 3 & 7 \end{pmatrix}$$

[1]In this paper, all logarithms are in base 2.

and the dashed edges correspond to the semi-permutation

$$\pi_2^s = \begin{pmatrix} 1 & 2 & 5 & 7 \\ 6 & 0 & 2 & 4 \end{pmatrix}$$

Clearly, $\pi = \pi_1^s \circ \pi_2^s$. □

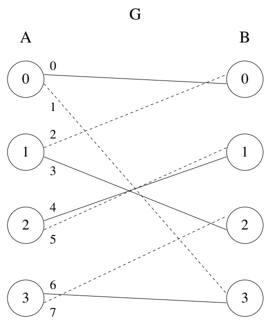

Fig. 2. A 2-edge coloring of bipartite graph G, where each edge is labeled by its corresponding input, and solid and dashed edges are colored with different colors.

B. Parallel Decomposition

Our parallel decomposition algorithm is presented for a completely connected multiprocessor system, which consists of a set of N processor elements (PEs) connected in such a way that there is a direct connection between every pair of PEs. The PEs are labeled beginning with 0 and placed as an array according to their labels in nondecreasing order. We assume that each PE can communicate with at most one processor during a communication step.

To facilitate the description of our algorithm, we introduce some notations. Let $b_v b_{v-1} \cdots b_1 b_0$ be the binary representation of a. We use $\bar{a}$ to denote the integer that has the binary representation $b_v b_{v-1} \cdots b_1(1-b_0)$. We use operator ":=" to denote an assignment local to a PE or to the control unit, and use operator "←" to denote an assignment requiring some interprocessor communication.

Initially, each PE$_i$ reads $\pi(i)$ from inputs, assigns value of $m(i)$ as i, and sets value of π^{-1} in PE$_{\pi(i)}$ as i. The pointer $p(i)$ of PE$_i$ will be set to point to a PE with index of $\pi^{-1}(\overline{\pi(\bar{i})})$, which is actually done by two steps. In the first step, PE$_i$ computes $\bar{i}$ and reads value of $\pi(\bar{i})$ from the PE with index $\bar{i}$. In the second step, PE$_i$ computes $\overline{\pi(\bar{i})}$ and reads value of $\pi^{-1}(\overline{\pi(\bar{i})})$ from the PE with index of $\overline{\pi(\bar{i})}$. Then, by $\lceil \log(N/2) \rceil$ times of pointer jumping [5], each PE$_i$ computes its $m(i)$ to be the minimum index of a PE it ever points to. Finally, the parity of $m(i)$ decides in which semi-permutation input I_i is, i.e. all inputs with the same parity are in the same semi-permutation. The detailed implementation is given in Algorithm 1.

Theorem 2: For any permutation, Algorithm 1 correctly computes two semi-permutations in $O(\log N)$ time on a completely connected multiprocessor system of N PEs.

Proof: After initialization of N pointers, a set of directed cycles (including loops) are formed (see Figure 3 for an example). It is easy to see that two dual inputs and two inputs mapped to a pair of dual outputs are in different directed

Input: A permutation
Output: Two semi-permutations
for all PE_i, $0 \le i \le N-1$, **do**
  $m(i) := i$;
  $\pi^{-1}(\pi(i)) \leftarrow i$;
  $p(i) \leftarrow \pi^{-1}(\overline{\pi(\bar{i})})$; /* pointer initialization */
  for $t := 1$ **to** $\lceil \log(N/2) \rceil$ **do**
    $m(i) \leftarrow$ min $\{m(i), m(p(i))\}$; /* comparison */
    $p(i) \leftarrow p(p(i))$; /* pointer jumping */
  end for
  if $m(i)$ is even **then**
    I_i is in the first semi-permutation;
  else
    I_i is in the second semi-permutation;
  end if
end for

Algorithm 1: A Parallel Decomposition

cycles. Since the length of each directed cycle is at most $N/2$, after $\lceil \log(N/2) \rceil$ times of pointer jumping, each PE$_i$ maintains the minimum index of the input in the directed cycle/loop to which I_i belongs. Hence, each PE$_i$ has $m(i) \equiv 0$ or $1 \bmod 2$. Therefore, two dual inputs/outputs are in different semi-permutations. Clearly, the algorithm takes $O(\log N)$ time since pointer jumping dominates the time complexity. □

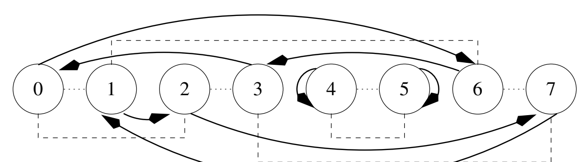

Fig. 3. Pointer initialization in Algorithm 1. By initializing N pointers, two directed cycles, $(0 \rightarrow 6 \rightarrow 3)$ and $(1 \rightarrow 2 \rightarrow 7)$, and two loops, 4 and 5, are formed, where each edge is represented by a circle, two dual inputs are connected by a dotted line, and two inputs mapped to two dual outputs are connected by a dashed line.

From the proof of theorem 2, we know that the decomposition of a permutation may not be unique, since we can assign one of dual inputs/outputs to either of two semi-permutations. In fact, if the number of cycles in its corresponding I/O mapping graph is c, $1 \le c \le N/2$, there are 2^{c-1} ways to decompose a given permutation into a pair of semi-permutations. Two semi-permutations can be used to set SEs in the first and last stages of $N \times N$ OMINs for crosstalk-free switching. In order to route a semi-permutation in a single pass without crosstalk, we need to assure there is only one active input of each SE in every stage of OMINs. In the next section, we will present a parallel crosstalk-free routing algorithm to realize semi-permutations in optical Benes networks.

III. A Parallel Crosstalk-Free Routing Algorithm for Optical Benes Networks

The Benes network [1] is a rearrangeable nonblocking permutation network and one of the most efficient switching architectures in terms of the number of 2×2 SEs used.

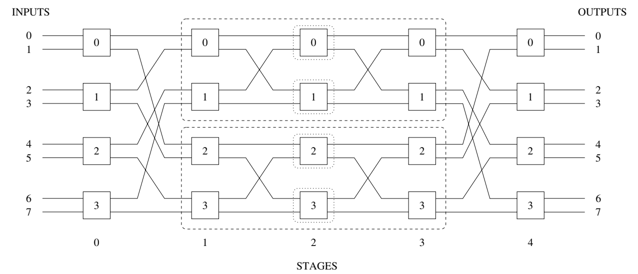

Fig. 4. A $B(8)$ contains 2 $B(4)$s within dashed boxes, each containing 2 $B(2)$s within dotted boxes.

An $N \times N$ Benes network is denoted by $B(N)$. It is constructed recursively. A $B(2)$ is a 2×2 SE. A $B(N)$ consists of a switching stage of $N/2$ SEs, an $N/2\times2$ shuffle connection (i.e. O_i is connected to I_j with $j \equiv N/2 \cdot i + \lfloor i/2 \rfloor \mod N$ in two adjacent stages [13]), followed by a stack of two $B(N/2)$s, a $2\times N/2$ shuffle connection, and another switching stage of N SEs. Each $B(N)$ contains 2 subnetworks that are $B(N/2)$s, and 4 subnetworks that are $B(N/4)$s, and so on. Thus, a $B(N)$ consists of $2\log N - 1$ stages labeled by $0, 1, \cdots, 2\log N - 2$ from left to right, and each stage consists of $N/2$ SEs labeled by $0, 1, \cdots, N/2 - 1$ from top to bottom. A pair of SEs i and $\bar{i}$ is called a pair of *dual SEs*. A $B(8)$ is shown in Figure 4. It was shown that any permutation can be realized in an Optical Benes network in two passes under the constraint of crosstalk-free [25]. The algorithm for routing a semi-permutation in an optical Benes Network is given below as Algorithm 2.

Input: A semi-permutation
Output: A setting of SEs of $B(N)$ without crosstalk
Step 1. If the size of the semi-permutation is 1, set up $B(2)$ according to the connection request, and exit.
Step 2. Decompose the semi-permutation into 2 parts, named upper- and lower-semi-permutation, satisfying that two active inputs/outputs in a pair of dual SEs in the first/last stage are in different parts.
Step 3. Set SEs in the first and last stages so that the active inputs and outputs in the upper/lower-semi-permutation are connected with the upper/lower subnetwork.
Step 4. Recursively call this algorithm in the upper/lower subnetwork with the input of the upper/lower-semi-permutation.

Algorithm 2: A Crosstalk-Free Parallel Routing Algorithm in Optical Benes Networks

The correctness and complexity of Algorithm 2 are given in the following theorem.

Theorem 3: For any semi-permutation of an optical $B(N)$, Algorithm 2 correctly computes the crosstalk-free I/O paths and sets the SEs on the paths in $O(\log^2 N)$ time on a completely connected multiprocessor system of N PEs.

Proof: By the topology of $B(N)$, we know that every pair of dual SEs in stage i (resp. $2\log N - 2 - i$), $0 \le i \le \log N - 2$, is connected with two SEs in stage $i+1$ (resp. $i-1$) and these two SEs are in different subnetwork $B(N/2^{i+1})$s. In order to satisfy the crosstalk-free constraint in each stage of $B(N)$, two active inputs (resp. outputs) belonging to a pair of dual SEs of stage i (resp. $2\log N - 2 - i$) must be connected with the SEs in different subnetwork $B(N/2^{i+1})$s. It is equivalent to assign a 2-edge coloring to a bipartite graph G, where 2 active inputs (outputs) belonging to a pair of dual SEs of stage i ($2\log N - 2 - i$) compose a vertex and each connection is corresponding to an edge. Thus, by using parallel decomposition algorithm recursively, the SEs are set to be crosstalk-free for any given semi-permutation. By Theorem 2, the time complexity of Step 2 in Algorithm 2 is $O(\log N)$. Since there are total $2\log N - 1$ stages and every parallel decomposition step can decide the setting of SEs of two stages (i.e. the first and last stages of a subnetwork) in $B(N)$, the time complexity for Algorithm 2 is $O(\log^2 N)$. □

By Theorem 3, given any permutation, we first call Algorithm 1 to decompose the permutation into 2 semi-permutations, then call Algorithm 2 twice to route two semi-permutations without crosstalk. Thus, the time complexity to route a permutation in an optical $B(N)$ is $O(\log^2 N)$, which is the same as the time complexity of the best known parallel routing algorithms for realizing a permutation in an electronic $B(N)$ [7], [10].

IV. Extending the Crosstalk-Free Routing for Partial Permutations

The decomposition algorithm presented in Section II is used to decompose a full permutation into two semi-permutations. We can generalize this algorithm to one that can decompose any partial permutation with $K(< N)$ active inputs into two partial permutations, each a subset of semi-permutation, named *partial semi-permutation*. The decomposition of partial permutation is equivalent to a 2-edge coloring of a bipartite graph with $\Delta(G) \le 2$. The extended parallel decomposition can be done in $O(\log K)$ time on a completely connected multiprocessor system of N PEs as follows.

Initially, each PE_i is associated with edge i. Let $p(i)$ be a pointer of PE_i, which is set to point to the PE with index of $\pi^{-1}(\overline{\pi(\bar{i})})$ if $\bar{i}$ is active and $\pi^{-1}(\overline{\pi(\bar{i})})$ exists (i.e. there is an active input j so that $\pi(j) = \overline{\pi(\bar{i})}$), and otherwise it is set to point to itself. For a partial permutation with K active inputs, its corresponding I/G mapping graph G is the union of a set of paths and cycles since $\Delta(G) \le 2$. For cycles, the case is the same as Algorithm 1. For paths, there are two directed paths formed for each path by the pointer initialization. By pointer jumping, the edges corresponding to the vertices in the same directed path can be colored with the same color. The two directed paths formed from a path can be colored with two different colors by comparing the indices of the end edges. For the detailed implementation, refer to [8], which contains parallel decomposition as a special case of 2-colorings of bipartite graphs.

Since a partial semi-permutation is a subset of some semi-permutation, it can be routed in an optical Benes network in one pass without crosstalk. By applying the extended parallel decomposition in step 2 of Algorithm 2, the total time for routing any partial permutation with K active inputs in an optical $B(N)$ takes $O(\log N \log K)$ since $B(N)$ has $2\log N-1$ stages and there are $\log N$ iterations in Algorithm 2.

In order to improve the complexity of routing time to $O(\log^2 K + \log N)$, we introduce a new concept, *equitable edge coloring*. A graph G is equitable c-edge colorable if $E(G)$ can be colored with c different colors so that the adjacent edges are colored with different colors and the difference between the sizes of any two color classes is at most one, where a color class is the subset of $E(G)$ with the same color for the coloring. Clearly, both cycle and path are 2-edge colorable. For any 2-edge colorings of paths or cycles, the sizes of two color classes are equal for a cycle and an even path while the difference between the sizes of two color classes is one for an odd path. The color with which more than half edges in an odd path are colored is called *primary color*. Thus, given a partial permutation, if the I/O mapping graph G has x odd paths, we color each path and cycle in G with two different colors c_1 and c_2 so that $\lceil \frac{x}{2} \rceil$ odd paths have c_1 as primary color and the remaining $\lfloor \frac{x}{2} \rfloor$ odd paths have c_2 as primary color. These 2-edge colorings of cycles and paths compose an equitable 2-edge coloring of G.

To route a partial semi-permutation in an optical $B(N)$ without crosstalk in $O(\log^2 K + \log N)$ time, we need to do a preprocessing and apply the equitable 2-edge coloring technique in step 2 of Algorithm 2. The preprocessing is to link K PEs corresponding to K busy inputs. This preprocessing step can be done by a parallel prefix sums operation [5], which takes $O(\log N)$ time on a completely connected multiprocessors with N PEs. In the following, we show how to color x odd paths of G with 2 colors so that the difference of 2 color classes is at most 1. It is easy to see that for any odd path, the edge whose dual input is not active will be colored with primary color. We call this kind edge *primary edge*. We concatenate all primary edges by a parallel prefix sums on the K linked PEs and alternately color the primary edges with two different colors. Thus there are $\lceil \frac{x}{2} \rceil$ primary edges with one color and $\lfloor \frac{x}{2} \rfloor$ primary edges with another color. The edges in an odd path will be colored using the primary edge as reference. That is, if an edge e and a primary edge f are in the same directed cycle, then e and f have the same color, and otherwise they have different colors. Therefore, an equitable 2-edge coloring of G is found. Figure 5 shows an example. Since the operations of pointer jumping and parallel prefix sums dominate the time complexity, an equitable 2-edge coloring of G can be found in $O(\log K)$ time using a completely connected multiprocessors with N PEs.

Using equitable 2-edge coloring technique, we can decompose a partial permutation into two partial semi-permutations with the difference between the sizes of two partial semi-permutations being at most one. When we route a partial semi-permutation in an optical Benes network, by applying the equitable 2-edge coloring technique in step 2 of Algorithm 2, the size of the partial permutation entering into each subnetwork is reduced by half. Thus after $\log K$ iterations, there is at most one active input entering into one subnetwork. Therefore, the total time for setting up a partial semi-permutation in an optical $B(N)$ is $O(\log K)$ in first $\log K$ iterations. Since there are $2\log N-1$ stages in $B(N)$, there are total $\log N$ iterations.

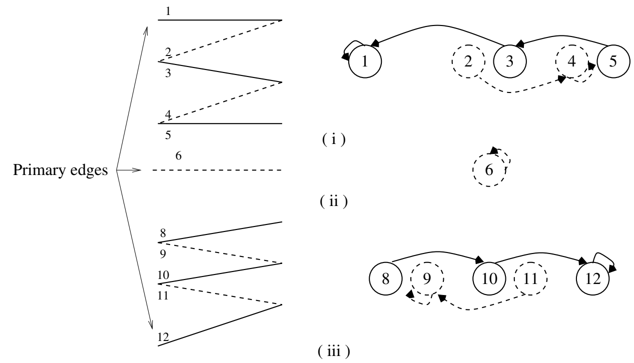

Fig. 5. An equitable 2-edge coloring of a graph consisting of 3 odd paths, where directed paths are formed by pointer initialization.

Therefore, we have the following claim.

Theorem 4: For any partial permutation with $K(< N)$ active inputs of an optical $B(N)$, it can be routed with crosstalk-free in $O(\log^2 K + \log N)$ time using a completely connected multiprocessor system of N PEs.

V. Comparisons of Three Approaches of Dilation for Optical Benes Networks

There are three approaches, time dilation, space dilation and wavelength dilation, can be used to avoid the crosstalk in OMINs.

In time dilation, the optical Benes network has the similar structure as the electronic Benes network. For any electronic Benes network, a permutation can be routed in one pass while it must be decomposed into 2 semi-permutations and routed by 2 passes in the corresponding optical Benes network to avoid crosstalk.

In space dilation, a dilated Benes network, denoted as $DB(N)$, consists of 2 copies of $B(N)$ with the corresponding two inputs and outputs are connected to a 1×2 splitter and a 2×1 combiner, respectively [6], [9] (see Figure 6 for an example). Given a permutation, we first decompose the permutation into 2 semi-permutations, then route each semi-permutation in one of copies of $DB(N)$ simultaneously.

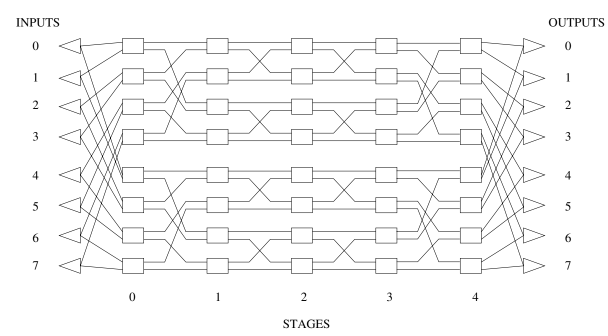

Fig. 6. A dilated Benes network $DB(8)$.

Compared with the time dilation approach, the space dilation approach uses more than double of hardware, i.e. twice of SEs and links plus splitters and combiners, and more than half of time to route a permutation, i.e. the time for decomposition and routing of one semi-permutation.

In wavelength dilation, if there is a wavelength converter available in each SE, we can convert two input signals with the same wavelength entering into the same SE to different ones. Thus, two wavelengths are necessary plus the costs of the wavelength converters. If there is no wavelength converter available, i.e. each connection will be assigned the same wavelength, then we find two wavelengths are not sufficient. An example is given as follows.

Example 2: Routing the permutation

$$\pi = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 0 & 2 & 1 & 3 \end{pmatrix}$$

in an optical $B(4)$.

In order to route the permutation π in $B(4)$, by the topology of $B(4)$, we know that inputs 0 and 1 (outputs 2 and 3) are connected with different subnetwork $B(2)$'s, which are two SEs in the second stage of $B(4)$. Since $\pi(1) = 2$ and $\pi(3) = 3$, we know that inputs 1 and 3 must be connected with different SEs in the second stage. Consequently, inputs 0 and 3 must be connected with the same SE in the second stage containing only 2 SEs. In order to avoid crosstalk, we must use different wavelengths for connections $0 \rightarrow 0$ and $3 \rightarrow 3$. We also know that the connections $0 \rightarrow 0$ and $1 \rightarrow 2$ must be carried on the signal with different wavelengths since they pass the same SE in the first stage. Thus, connections $3 \rightarrow 3$ and $1 \rightarrow 2$ must have the same wavelength if there are only two available wavelengths. However, the connections $3 \rightarrow 3$ and $1 \rightarrow 2$ pass through the same SE in the last stage of $B(4)$, which will cause crosstalk. □

From the above discussion, we know that the time dilation approach is the most cost-effective provided that the cost both in space and in wavelength are at least as high as the cost in time.

VI. Concluding Remarks

In this paper, we proposed a fast parallel decomposition algorithm with time complexity $O(\log N)$, which can decompose any permutation with size of N into two semi-permutations assuring no crosstalk in SEs of the first and last stages in OMINs. Based on this parallel decomposition, we further presented a fast crosstalk-free parallel routing algorithm, which can set up any permutation in $O(\log^2 N)$ time in an optical $B(N)$. The proposed decomposition algorithm can be generalized to any partial permutation. Using the equitable 2-edge coloring technique, any partial permutation with $K(< N)$ active inputs can be routed in $O(\log^2 K + \log N)$ time in an optical $B(N)$. In addition, the proposed algorithms run on a completely connected multiprocessor system can be easily translated to the algorithms on more realistic multiprocessor systems.

References

[1] V.E. Benes, "Permutation groups, complexes, and rearrangeable connecting networks", *The Bell System Technical Journal*, vol. 43, pp. 1619-1640, July 1964.

[2] J.A. Bondy and U.S.R. Murty, *Graph Theory with Applications*, Elsevier North-Holland, 1976.

[3] H. Hinton, "A non-blocking optical interconnection network using directional couplers", *Proceedings of IEEE Global Telecommunications Conference*, pp. 885-889, Nov. 1984.

[4] D. K. Hunter, P. J. Legg, and I. Andonovic, "Architecture for large dilated optical TDM switching networks", *IEE Proceedings on Optoelectronics*, vol. 140, no. 5, pp. 337-343, October 1993,

[5] J. Jaja, *An Introduction to Parallel Algorithms*, Addison-Wesley, 1992.

[6] C.-T. Lea and D.-J. Shyy, "Tradeoff of horizontal decomposition versus vertical stacking in rearrangeable nonblocking networks", *IEEE Transactions on Communications*, vol. 39, no. 6, pp. 899 -904, June 1991.

[7] G.F. Lev, N. Pippenger, and L.G. Valiant, "A fast parallel algorithm for routing in permutation networks", *IEEE Transactions on Computers*, vol. 30, pp. 93-100, Feb. 1981.

[8] E. Lu and S. Q. Zheng, "A Fast Parallel Routing Algorithm for Benes Group Switches", *Proceedings of the 14th IASTED International Conference on Parallel and Distributed Computing and Systems*, pp. 67-72, Nov. 2002.

[9] G. Maier and A. Pattavina, "Design of photonic rearrangeable networks with zero first-order switching-element-crosstalk", *IEEE Transactions on Communications*, vol. 49, no. 7, pp. 1268-1279, Jul. 2001.

[10] N. Nassimi and S. Sahni, "Parallel algorithms to set up the Benes permutation network", *IEEE Transactions on Computers*, vol. 31, no. 2, pp. 148-154, Feb. 1982.

[11] K. Padmanabhan and A. Netravali, "Dilated network for photonic switching", *IEEE Transactions on Communications*, vol. COM-35, no. 12, pp. 1357-1365, Dec. 1987.

[12] Y. Pan, C. Qiao, and Y. Yang, "Optical multistage interconnection networks: new challenges and approaches", *IEEE Communications Magazine*, vol. 37, no. 2, pp. 50-56, Feb. 1999.

[13] J.H. Patel, "Performance of processor-memory interconnections for multiprocessors", *IEEE Transactions on Computers*, vol. 30, no. 10, pp. 771-780, Oct. 1981.

[14] C. Qiao, R. Melhem, D. Chiarulli, and S. Levitan, "A time domain approach for avoiding crosstalk in optical blocking multistage interconnection networks", *IEEE Journal Lightwave Technology*, vol. 12, no. 10, pp. 1854-1862, Oct. 1994.

[15] C. Qiao, "Analysis of space-time tradeoffs in photonic switching networks", *Proceedings of IEEE INFOCOM*, vol. 2, pp. 822-829, March 1996.

[16] X. Qin and Y. Yang, "Permutation capacity of WDM switching networks with limited wavelength conversion", *IEEE Proceedings of International Conference on Parallel Processing Workshop on Optical Networks (ICPP '01)*, pp. 271-276, Sep. 2001.

[17] X. Qin and Y. Yang, "Nonblocking WDM switching networks with full and limited wavelength conversion", *IEEE Transactions on Communications*, vol. 50, no. 12, pp. 2032-2041, Dec. 2002.

[18] R. Ramaswami and K. Sivarajan, *Optical Networks: A Practical Perspective*, second edition, Morgan Kaufmann, 2001.

[19] J. Sharony, K.W. Cheung, and T.E. Stern, "Wavelength dilated switches (WDS)-a new class of high density, suppressed crosstalk, dynamic wavelength-routing crossconnects", *IEEE Photonics Technology Letters*, vol. 4, no. 8, pp. 933 -935, Aug. 1992.

[20] J. Sharony, K.W. Cheung, and T.E. Stern, "The wavelength dilation concept in lightwave networks-implementation and system considerations", *IEEE Journal of Lightwave Technology*, vol. 1, no. 5/6, pp. 900 -907, May-Jun. 1993.

[21] X. Shen, F. Yang, and Yi Pan, "Equivalent permutation capabilities between time-division optical Omega networks and non-optical extra-stage Omega networks", *IEEE/ACM Transactions on Networking*, vol. 9, no. 4, Aug. 2001.

[22] G.H. Song and M.Goodman, "Asymmetrically-dilated cross-connect switches for low-crosstalk WDM optical networks", *Proceedings of IEEE 8th Annual Meeting Conference on Lasers and Electro-Optics Society Annual Meeting*, vol. 1, pp. 212-213, Oct. 1995.

[23] F.M. Suliman, A.B. Mohammad, and K. Seman, "A space dilated lightwave network-a new approach", *Proceedings of IEEE 10th International Conference on Telecommunications (ICT 2003)*, vol. 2, pp. 1675 -1679, 2003.

[24] J.E. Watson et al., "A low-voltage 8×8 Ti:LiNbO$_3$ switch with a dilated Benes architecture," *IEEE Journal of Lightwave Technology*, vol. 8, pp. 794-800, May 1990.

[25] Y. Yang, J. Wang, and Y. Pan, "Permutation capability of optical multistage interconnection networks", *Journal of Parallel and Distributed Computing*, vol. 60, no. 1, pp. 72-91, Jan. 2000.

[26] Y. Yang and J. Wang, "Optimal all-to-all personalized exchange in a class of optical multistage networks", *IEEE Transactions on Parallel and Distributed Systems*, vol. 12, no. 6, pp. 567-582, June. 2001.