IEICE TRANS. INF.&SYST., Vol.E86-D, No.9, Sept.2003.

PAPER *Special Issue on* *****

# Crosstalk-free Permutation in Photonic Rearrangeable Networks Built on a Combination of Horizontal Expansion and Vertical Stacking of Banyan Networks

Xiaohong Jiang, Hong Shen, Md. Mamun-ur-Rashid Khandker and Susumu Horiguchi, *Regular Member*

**Summary**
Crosstalk in optical switch is an intrinsic drawback of optical networks, and avoiding crosstalk is important for making optical network work properly. Horizontal expansion and vertical stacking are two basic techniques for creating nonblocking multistage interconnection networks (MINs). Rearrangeable (nonblocking) optical MINs are feasible since they have lower complexity than their strictly nonblocking counterparts. In this paper, we study the crosstalk-free permutations in rearrangeable optical MINs built on a combination of horizontal expansion and vertical stacking of banyan networks, and provide a scheme for realizing crosstalk-free permutations in this kind of optical MINs. The basic idea of this scheme is to first decompose a permutation into multiple partial permutations by using Euler Split technique, then route and realize each of these partial permutations crosstalk-free in one plane (stacked copy) of a MIN based on both the Euler Split technique and self-routing property of a banyan network. The tradeoff between the overall time complexity and hardware cost of this class of MINs is also explored in this paper.
***Key words:*** *Optical switch, banyan network, rearrangeably nonblocking, optical crosstalk.*

## 1. Introduction

An optical switching element can be implemented by using directional-coupler (DC) that consists of two waveguides closer to each other [1,2]. DC can switch multiple wavelength at same time which is important for the future optical cross-connects. DC suffers from the crosstalk which occurs between two signals carried in the two waveguides of the coupler. Due to the stringent bit-error rate requirement in optical transmission systems, elimination of crosstalk in a DC-based switching system has been widely studied [1,6,7,8,9,10,11,12,13].

Available optical multistage interconnection networks (MINs) were constructed mainly from banyan-type networks (e.g. *banyan* [3], *baseline* and *omega* [4] [5]) because they have a small number of switches between an input-output pair and crosstalk and signal attenuation in a DC-based optical switching network are proportional to the number of couplers that a light signal passes through. A typical banyan-type network is shown in Fig.1. Banyan-type networks have a unique path between an input-output pair, and this makes them blocking networks. Nonblocking network can be constructed by either appending some extra stages to the back of a banyan network (Horizontal expansion technique) or vertically stacking multiple copies of a banyan network (Vertical Stacking technique) [15]. Rearrangeable is an interesting choice for optical switching networks. This kind of network can always rout any idle input to any unused output, but one or more existing connections may have to be rerouted to establish the path. The rearrangeable optical networks are attractive because the cost and signal degradation of a rearrangeable optical network are lower than its strictly nonblocking and wide-sense nonblocking counterparts. Based on a combination of horizontal expansion technique and vertical stacking technique, the condition for a banyan-type network to be rearrangeable and free of crosstalk in SEs (we refer to this as crosstalk-free hereafter) has been determined [8].

In this paper, we will look into the crosstalk-free permutation in this kind of rearrangeable optical MINs, and present a scheme for realizing permutations in this kind of optical MINs crosstalk-free by first decomposing a permutation into multiple crosstalk-free realizable partial permutations (CRPPs) and then routing each of these CRPPs in a stacked plane of the MIN crosstalk-free. The main contributions of our work are the following:

(1) We present a rigorous proof of CRPP decomposability based on a combinatorial theorem of P.Hall [16] and the treatments established in [13].

(2) We develop an efficient routing algorithm for realizing a CRPP in a stacked plane of the MIN crosstalk-free based on both the *Euler Split* technique for edge coloring of bipartite graph [14] and the self-routing property of banyan networks.

(3) We study in detail the time complexity of our scheme and also the possible tradeoff between overall time complexity and hardware cost.

The rest of the paper is organized as follows: Section 2 describes briefly the rearrangeable MINs built on a combination of horizontal expansion and vertical stacking of banyan networks. The scheme for realizing a permutation crosstalk-free in this class of MINs is presented in Section 3. Section 4 discusses the tradeoffs

Manuscript received Dec. 5, 2002. Manuscript revised Feb.26, 2003.
The authors are with Graduate School of Information Science, Japan Advanced Institute of Science and Technology, JAIST, Tatsunokuchi, Ishikawa 923-1292, Japan.

between time complexity and hardware cost. Section5 summarizes the results of this paper.

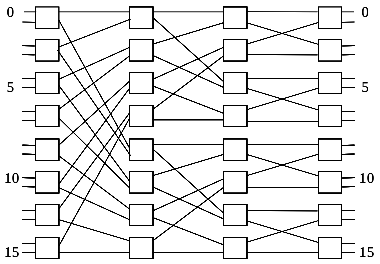


Fig.1 16×16 banyan-type network.

## 2. Rearrangeably nonblocking networks built on banyan networks

Based on an $N \times N$ banyan network composed of $n = \log_2 N$ stages, an $N \times N$ rearrangeable network can be built by using either Vertical Stacking (VS) technique or Horizontal Expansion (HE) technique [15]. A VS network is constructed by vertically stacking $K$ copies of a banyan network with each input (output) is connected to a $1 \times K$ splitter ($K \times 1$ combiner). A HE network can be built by appending the reverse of first $m$ ($m \le n$-1) stages of a regular $N \times N$ banyan network to the back of the network and can be described by the symbol H($n$, $m$). More generally, a rearrangeable network can also be derived by the combination of both VR and HE techniques in which an H($n$, $m$) network is first created from a regular banyan network and the H($n$, $m$) network is then vertically replicated $K$ times [8]. The overall network can be described by the symbol VH($n$, $m$, $K$). Fig.2 illustrates a network VH(4,2,2).

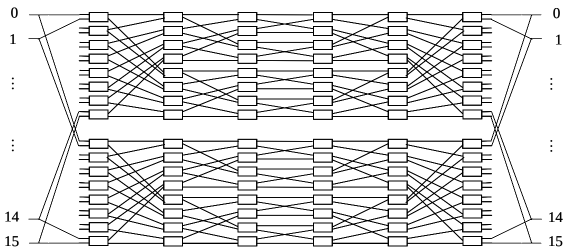


Fig.2 The construction of a VH (4,2,2) from banyan networks.

Ideally, we are interested in designing a network without any crosstalk. We have the following result concerning the rearrangeable conditions for VH($n$, $m$, $K$) networks without crosstalk [8].

**Theorem 1**: Under the constraint of crosstalk-free, a VH($n$, $m$, $K$) network is rearrangeable (nonblocking) if and only if

$$K = 2^{\lfloor (n-m+1)/2 \rfloor} \quad (1)$$

Hereafter, we will use RVH($n$, $m$) to refer to the rearrangeable network VH($n$, $m$, $K$) with $K = 2^{\lfloor (n-m+1)/2 \rfloor}$. Note that under the constraint of crosstalk-free, the complexity of a rearrangeable banyan-type optical MIN is much lower than its strictly nonblocking counterpart [6] and its wide-sense nonblocking counterpart [12].

## 3. Crosstalk-free permutation in RVH (*n*, *m*)

The above result indicates that any permutation can be realized crosstalk-free in a RVH($n$, $m$) network. In this section, an efficient scheme will be developed to actually realize a permutation crosstalk-free in a RVH($n$,$m$) network. The basic idea of realizing a full permutation crosstalk-free in such a network is to decompose it into multiple crosstalk-free realizable partial permutations each of which can be realized crosstalk-free in a single plane of the vertically stacked network.

### 3.1 Crosstalk-free Realizable Partial Permutation

To realize a permutation in a RVH ($n$, $m$) network, we need to distribute a full permutation to $T$ stacked H ($n$, $m$) networks, here $T = 2^{\lfloor (n-m+1)/2 \rfloor}$. To do so, we can decompose the permutation into $T$ partial permutations and realize each of these partial permutations in one H($n$,$m$) of the RVH ($n$, $m$) network. For convenience, we introduce the following definition.

**Definition 1** A partial permutation $P = \begin{pmatrix} x_0, x_1, \cdots, x_{N/T-1} \\ y_0, y_1, \cdots, y_{N/T-1} \end{pmatrix}$ for an $N \times N$ H($n$,$m$) network, where input $x_i$ is mapped to output $y_i$, with $x_i$, $y_i \in \{0,1,\ldots,N\text{-}1\}$ and $x_0 < x_1 < \cdots < x_{N/T-1}$, is referred to as a crosstalk-free realizable partial permutation (CRPP) to the H($n$,$m$) network if:

$$\left\{\left\lfloor \frac{x_0}{T} \right\rfloor, \left\lfloor \frac{x_1}{T} \right\rfloor, \cdots, \left\lfloor \frac{x_{N/T-1}}{T} \right\rfloor\right\} = \left\{0,1,\cdots,\frac{N}{T}-1\right\} \text{ and}$$

$$\left\{\left\lfloor \frac{y_0}{T} \right\rfloor, \left\lfloor \frac{y_1}{T} \right\rfloor, \cdots, \left\lfloor \frac{y_{N/T-1}}{T} \right\rfloor\right\} = \left\{0,1,\cdots,\frac{N}{T}-1\right\} \quad (2)$$

Note that if we divide both the inputs and outputs of an H ($n$, $m$) network into $T = 2^{\lfloor (n-m+1)/2 \rfloor}$-element set $I_j = \{T \cdot j,\ T \cdot j+1,\ \cdots,\ T \cdot j + T - 1\}$ and $O_j = \{T \cdot j,\ T \cdot j+1,\ \cdots,\ T \cdot j + T - 1\}$ for $0 \le j \le \frac{N}{T} - 1$, with $I_j$ and $O_j$ correspond to inputs and outputs, respectively, partial permutation $P$ is a CRPP to the network if and only if each input and each output of the partial permutation falls within a distinct set $I_j$ and $O_j$ for $0 \le j \le \frac{N}{T} - 1$ ,respectively. Since we always have $T = 2^{\lfloor (n-m+1)/2 \rfloor} \ge 2$ for any H($n$, $m$) network, it is easy to see that a CRPP eliminates crosstalk at least in the first and the last stages of the network.

**Example 1** For an H(4,1) network, partial permutation $\begin{pmatrix} 3 & 4 & 9 & 13 \\ 12 & 4 & 1 & 10 \end{pmatrix}$ is a CRPP, since $N$ =16, $T = 2^{\lfloor (n-m+1)/2 \rfloor} = 4$ and we have

$$\left\{ \left\lfloor \frac{3}{4} \right\rfloor, \left\lfloor \frac{4}{4} \right\rfloor, \left\lfloor \frac{9}{4} \right\rfloor, \left\lfloor \frac{13}{4} \right\rfloor \right\} = \{0, 1, 2, 3\} \text{ and}$$

$$\left\{ \left\lfloor \frac{12}{4} \right\rfloor, \left\lfloor \frac{4}{4} \right\rfloor, \left\lfloor \frac{1}{4} \right\rfloor, \left\lfloor \frac{10}{4} \right\rfloor \right\} = \{3, 1, 0, 2\} = \{0, 1, 2, 3\} \tag{3}$$

## 3.2 Decomposition of a permutation into CRPPs

Recall that any permutation can be realized crosstalk-free in a RVH($n$, $m$) network consisting of $2^{\lfloor (n-m+1)/2 \rfloor}$ vertically stacked planes (copies) of H($n$, $m$) network. We first prove in this section that any permutation can be evenly decomposed into $T = 2^{\lfloor (n-m+1)/2 \rfloor}$ CRPPs, and then present an algorithm for such decompositions.

### 3.2.1 CRPP decomposability

The CRPP decomposability of a permutation is summarized as the following lemma.

**Lemma 1:** Any permutation of an $N$-element set {0,1,…, $N$-1} can be decomposed into $T = 2^{\lfloor (n-m+1)/2 \rfloor}$ CRPPs for an H($n$, $m$) network.

*Proof.* We will prove the lemma by applying the P.Hall's distinct system representatives theorem [16] and using the treatments similar to those established in [13].

Let the permutation be the form

$$\begin{pmatrix} x_0, x_1, \dots, x_{N-1} \\ y_0, y_1, \dots, y_{N-1} \end{pmatrix}$$

where $\{y_0, y_1, \dots, y_{N-1}\} = \{0,1,\dots,N-1\}$ and $x_l = l$, $0 \le l \le N-1$.

We first decompose the permutation evenly into $N/T$ partial permutations each of which has $T$ elements

$$\begin{pmatrix} x_0 \dots x_{T-1} \\ y_0 \dots y_{T-1} \end{pmatrix}, \begin{pmatrix} x_T \dots x_{T+T-1} \\ y_T \dots y_{T+T-1} \end{pmatrix}, \dots, \begin{pmatrix} x_{(N/T-1)T} \dots x_{N-1} \\ y_{(N/T-1)T} \dots y_{N-1} \end{pmatrix} \tag{4}$$

Then for $j^{\text{th}}$ partial permutation $\begin{pmatrix} x_{j\cdot T} \dots x_{j\cdot T+T-1} \\ y_{j\cdot T} \dots y_{j\cdot T+T-1} \end{pmatrix}$ ($0 \le j \le N/T-1$), we have:

$$\begin{pmatrix} \left\lfloor \frac{x_{j\cdot T}}{T} \right\rfloor \dots \left\lfloor \frac{x_{j\cdot T+T-1}}{T} \right\rfloor \\ \left\lfloor \frac{y_{j\cdot T}}{T} \right\rfloor \dots \left\lfloor \frac{y_{j\cdot T+T-1}}{T} \right\rfloor \end{pmatrix} = \begin{pmatrix} j & \dots & j \\ \left\lfloor \frac{y_{j\cdot T}}{T} \right\rfloor & \dots & \left\lfloor \frac{y_{j\cdot T+T-1}}{T} \right\rfloor \end{pmatrix}$$

Denote $\left\{ \left\lfloor \frac{y_{j\cdot T}}{T} \right\rfloor, \left\lfloor \frac{y_{j\cdot T+1}}{T} \right\rfloor, \dots, \left\lfloor \frac{y_{j\cdot T+T-1}}{T} \right\rfloor \right\}$ as $B_j$ for $0 \le j \le N/T-1$, then we know from $\{y_0, y_1, \dots, y_{N-1}\} = \{0,1,\dots,N-1\}$ that there are $T$ 0's, $T$ 1's, … , and $T$ ($N/T$ -1)'s distributed in sets $B_0, B_1, \dots, B_{N/T-1}$. Thus, for any $K$ sets ($1 \le K \le N/T$), $B_{i_1}, B_{i_2}, \dots, B_{i_K}$, there are a total of $T \cdot K$ elements that form a multiset with the multiplicity of each element no more than $T$. Therefore, the cardinality of the union of these $K$ sets satisfies:

$$\left| B_{i_1} \bigcup B_{i_2} \bigcup \dots \bigcup B_{i_K} \right| \ge K \tag{5}$$

From P.Hall's distinct system representatives theorem [16], we know that (5) is the necessary and sufficient condition for these $K$ sets $B_0, B_1, \dots, B_{N/T-1}$ have a set of distinct system representatives, so there exist $b_0 \in B_0$ , $b_1 \in B_1$ , … , $b_{N/T-1} \in B_{N/T-1}$ such that $b_i \ne b_j$ for any $i \ne j$ ($0 \le i, j \le N/T$ -1). This indicates that

$$\{b_0, b_1, \dots, b_{N/T-1}\} = \{0, 1, \dots, N/T - 1\} \tag{6}$$

Since $b_j \in B_j = \left\{ \left\lfloor \frac{y_{j\cdot T}}{T} \right\rfloor, \left\lfloor \frac{y_{j\cdot T+1}}{T} \right\rfloor, \dots, \left\lfloor \frac{y_{j\cdot T+T-1}}{T} \right\rfloor \right\}$, we denote $b_j$ as $\left\lfloor \frac{y_{j\cdot T+d_j}}{T} \right\rfloor$. Then we can get the following partial permutation based on the selection of $b_j$ ($0 \le j \le N/T$ -1):

$$\begin{pmatrix} x_{d_0} & x_{T+d_1} & x_{2T+d_2} & \dots & x_{(N/T-1)T+d_{N/T-1}} \\ y_{d_0} & y_{T+d_1} & y_{2T+d_2} & \dots & y_{(N/T-1)T+d_{N/T-1}} \end{pmatrix} \tag{7}$$

For subscripts $d_j$, it easy to see that

$$0 \le d_j \le T - 1,\ j = 0, 1, \dots, N/T - 1. \tag{8}$$

We will show that (7) is a CRPP. Note that

$$\left\{ \left\lfloor \frac{x_{d_0}}{T} \right\rfloor, \left\lfloor \frac{x_{T+d_1}}{T} \right\rfloor, \left\lfloor \frac{x_{2T+d_2}}{T} \right\rfloor, \dots, \left\lfloor \frac{x_{(N/T-1)\cdot T+d_{N/T-1}}}{T} \right\rfloor \right\} =$$

$$= \left\{ \left\lfloor \frac{d_0}{T} \right\rfloor, \left\lfloor \frac{T+d_1}{T} \right\rfloor, \dots, \left\lfloor \frac{(N/T-1)\cdot T + d_{N/T-1}}{T} \right\rfloor \right\} \overset{\text{by (8)}}{=} \{0, 1, \dots, N/T - 1\}$$

$$\left\{ \left\lfloor \frac{y_{d_0}}{T} \right\rfloor, \left\lfloor \frac{y_{T+d_1}}{T} \right\rfloor, \left\lfloor \frac{y_{2T+d_2}}{T} \right\rfloor, \dots, \left\lfloor \frac{y_{(N/T-1)T+d_{N/T-1}}}{T} \right\rfloor \right\}$$

$$= \{b_0, b_1, b_2, \dots, b_{N/T-1}\} \overset{\text{by (6)}}{=} \{0, 1, \dots, N/T - 1\}$$

Therefore, by Definition 1, partial permutation (7) is a CRPP, and we can take it as the first CRPP decomposed from the permutation.

After we delete the inputs and outputs corresponding the first CRPP from the $N/T$ $T$-element partial permutations in (4), we can get $N/T$ ($T$-1)-element partial permutations, and the above decomposition process can then be applied to these $N/T$ ($T$-1)-element partial permutations again to obtain the second CRPP. Applying the above process recursively, we can finally decompose the original permutation into $T$ CRPPs for an H($n$, $m$) network. QED.

### 3.2.2 CRPP decomposition algorithm

Lemma 1 guarantees the correctness of the CRPP decomposition of a permutation. As indicated in [8], a CRPP decomposition algorithm can be easily obtained by the repetition of the simple bi-partite graph coloring procedure with two colors [14]. A high-level description of the complete CRPP decomposition algorithm can be summarized as:

**CRPP Decomposition Algorithm for H ($n$, $m$) Networks:**

**Initiate:** $i$ = 0 and take the permutation as the 0-level partial permutation.

**Step 1:** If $i = \lfloor (n-m+1)/2 \rfloor$, exit.

**Step 2:** For each $i$-level partial permutation, do step 3 and step 4.

**Step 3:** Construct the bipartite graph $G = (V_1,V_2\,;\,E)$ for the $i$-level partial permutation. The vertex sets of $G$ are defined as

$$V_1 = \left\{I_0, I_1, \cdots, I_{\frac{N}{2^{i+1}}-1}\right\},\ V_2 = \left\{O_0, O_1, \cdots, O_{\frac{N}{2^{i+1}}-1}\right\}$$

Here $I_j = \{2^{i+1}\cdot j,\ 2^{i+1}\cdot j+1,\ \cdots,\ 2^{i+1}\cdot j+2^{i+1}-1\}$ and $O_j = \{2^{i+1}\cdot j,\ 2^{i+1}\cdot j+1,\ \cdots,\ 2^{i+1}\cdot j+2^{i+1}-1\}$ for $0 \le j \le \frac{N}{2^{i+1}}-1$, with $I_j$ and $O_j$ correspond to inputs and outputs, respectively. The edge set $E$ is defined as: for any input-out pair ($x_j$, $y_j$) in the partial permutation, if $x_i \in I_{j_1}$ and $y_i \in O_{j_2}$, then there is an edge between vertex $I_{j_1}$ and vertex $O_{j_2}$ in $E$.

**Step 4:** Since each vertex of $G$ has an even degree (degree 2), then the bipartite graph only consists of closed paths (connected component), an *Euler split* for edge coloring of bipartite graph [14] split $G$ into two bipartite graphs $G_1 = (V_1,V_2\,;\,E_1)$ and $G_2 = (V_1,V_2\,;\,E_2)$ where $E_1$ and $E_2$ are formed as follows: for each connected component of $G$, start from a vertex of this component in $V_1$, traverse through an unvisited edge to the neighbor vertex in $V_2$, back and forth until return to the starting vertex. During the traversing, a visited edge will be placed into $E_1$ if the traverse direction on this edge is from $V_1$ to $V_2$; and placed into $E_2$ if the direction is opposite.

**Step 5:** Take all one-pair mappings corresponding to the edges in $E_1$, to form one ($i$+1)-level partial permutation corresponding to the $i$-level partial permutation; let the remaining one-pair mappings, corresponding to the edges in $E_2$, form another ($i$+1)-level partial permutation corresponding to the $i$-level partial permutation.

**Step 6:** $i \Leftarrow i+1$. Go to Step 1.

It is clear that after running Steps 2-5 for a $i$-level partial permutation of the permutation, two ($i$+1)-level partial permutations will be obtained correspondingly with each input and output of a ($i$+1)-level partial permutation within a distinct set $\{2^{i+1}\cdot j,\ 2^{i+1}\cdot j+1,\ \cdots,\ 2^{i+1}\cdot j+2^{i+1}-1\}$ for $0 \le j \le \frac{N}{2^{i+1}}-1$. Thus, after running the decomposition algorithm for a permutation in an H ($n$, $m$) network, the permutation will be decomposed into $T = 2^{\lfloor (n-m+1)/2 \rfloor}$ partial permutations with each input (output) of a partial permutation within a distinct set $\{T\cdot j,\ T\cdot j+1,\ \cdots,\ T\cdot j+T-1\}$ for $0 \le j \le \frac{N}{T}-1$. So these partial permutations obtained are just the CRPPs for the H($n$, $m$) network. It is easy to see that the time to construct the bipartite graph is proportional to the number of pairs in the permutation, i.e., $O(N)$, and the time to traverse all edges is $O(N)$, so Steps 2-5 take $O(N)$ steps and these steps will repeat $O(\lfloor (n-m+1)/2 \rfloor) = O(\log_2 N-m)$ times. Therefore, the time complexity of the decomposition algorithm is $O(N(\log_2 N-m))$, where $0 \le m \le \log_2 N-1$.

**Example 2** Decomposition of the following permutation into CRPPs for H(4,2) network

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 8 & 2 & 0 & 12 & 4 & 13 & 3 & 11 & 9 & 1 & 6 & 7 & 5 & 10 & 15 & 14 \end{pmatrix} \quad (9)$$

Since $n$ = 4 and $m$ = 2, we need $\lfloor (n-m+1)/2 \rfloor = 1$ level decomposition to decompose the permutation into $2^{\lfloor (n-m+1)/2 \rfloor} = 2$ CRPPs for an H(4,2) network.

Take the permutation (9) as the 0-level partial permutation, the bipartite graph $G = (V_1,V_2\,;\,E)$ and edge traverses are shown in Fig.3, where $V_1 = \{I_0, I_1, \cdots, I_7\}$, $V_2 = \{O_0, O_1, \cdots, O_7\}$, and $e_j = \binom{j}{-}$ for $0 \le j \le 15$.

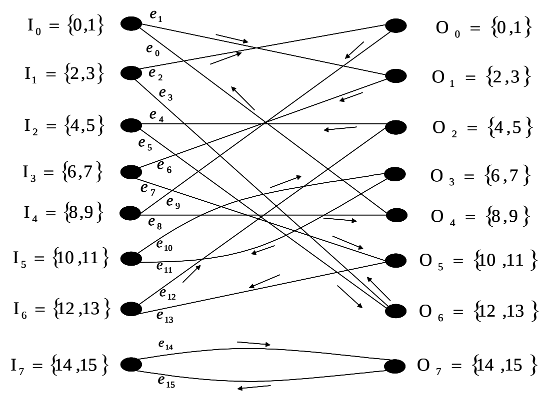


Fig.3 The bipartite graph and edge traverses of CRPP decomposition.

Then the set of all the edges with traversing direction being from $V_1$ to $V_2$ is a perfect matching of the bipartite graph $G$, and so is the set of all the edges with traversing direction being from $V_2$ to $V_1$. Take the pairs $e_1$, $e_2$, $e_5$, $e_7$,

$e_8$, $e_{10}$, $e_{12}$ and $e_{14}$ that correspond to the edges with traversing direction being from $V_1$ to $V_2$ to form

$$\begin{pmatrix} 1 & 2 & 5 & 7 & 8 & 10 & 12 & 14 \\ 2 & 0 & 13 & 11 & 9 & 6 & 5 & 15 \end{pmatrix} \quad (10)$$

and take the pairs $e_0$, $e_3$, $e_4$, $e_6$, $e_9$, $e_{11}$, $e_{13}$ and $e_{15}$ that correspond to the edges with traversing direction being from $V_2$ to $V_1$ to form

$$\begin{pmatrix} 0 & 3 & 4 & 6 & 9 & 11 & 13 & 15 \\ 8 & 12 & 4 & 3 & 1 & 7 & 10 & 14 \end{pmatrix} \quad (11)$$

Since we need only one level decomposition to complete the CRPP decomposition for H(4,2) network, then the two partial permutations (10) and (11) obtained are just the two CRPPs for H(4,2) network.

## 3.3 Realization of a CRPP in an H ($n$, $m$) network crosstalk-free

The problem remaining unsolved is to show that a CRPP can be actually realized crosstalk-free in an H($n$,$m$) network in one pass. We first need the following lemma concerning the crosstalk-free property of a banyan network.

For an $N \times N$ banyan network, define sets:

$$SW_j^{[1]} = \{u_{2^{\lfloor (n+1)/2 \rfloor -1}\cdot j}, u_{2^{\lfloor (n+1)/2 \rfloor -1}\cdot j+1}, \cdots, u_{2^{\lfloor (n+1)/2 \rfloor -1}\cdot j+2^{\lfloor (n+1)/2 \rfloor -1}-1}\},$$

$$SW_j^{[2]} = \{v_{2^{\lfloor (n+1)/2 \rfloor -1}\cdot j}, v_{2^{\lfloor (n+1)/2 \rfloor -1}\cdot j+1}, \cdots, v_{2^{\lfloor (n+1)/2 \rfloor -1}\cdot j+2^{\lfloor (n+1)/2 \rfloor -1}-1}\},$$

$0 \le j \le \frac{N}{2^{\lfloor (n+1)/2 \rfloor}} - 1$, where $u_0$, $u_1$,…, $u_{N/2-1}$ are the $N$/2 inputs switches and $v_0$, $v_1$ ,…,$v_{N/2-1}$ are the $N$/2 outputs switches. Then we have [8,17]:

**Lemma 2:** For the two inputs and outputs of any two one-pair mappings in an $N \times N$ banyan network, if their corresponding two input switches and output switches belong to two different input sets and output sets defined above, the two mappings will be crosstalk-free in the entire network.

We can now establish the following theorem regarding the crosstalk-free realization of a CRPP in an H($n$, $m$) network.

**Theorem 2**: Any partial permutation $P = \begin{pmatrix} x_0, x_1, \cdots, x_{N/T-1} \\ y_0, y_1, \cdots, y_{N/T-1} \end{pmatrix}$ for an $N \times N$ H($n$,$m$) network can be realized crosstalk-free in the network in one pass if the partial permutation satisfies the condition (2).

**Proof.** Note that an $N \times N$ H($n$, $m$) network can be built by appending the reverse of first $m$ ($m \le n$-1) stages of a regular $N \times N$ banyan network to the back of the network, and the stages of a H ($n$, $m$) network will be numbered from left (stage 1) to right (stage $n+m$). It is easy to see that in an H ($n$, $m$) network, the first $m$ stages and the last $m$ stages are symmetric about central $n$-$m$ stages and those $n$-$m$ central stages form a column of $2^m$ banyan networks of size H ($n$-$m$, 0). When $m$ = 0, an H($n$, $m$) network become a banyan network and it is easy to verify that the CRPPs for the network just meet crosstalk-free requirement specified in Lemma 2 and thus crosstalk-free realizable in the network taking advantage of self-routing property of banyan network. In the following, we only consider the case of $m \ge 1$. When $m \ge 1$, an H($n$, $m$) network can be defined in a recursive way shown in Fig.4.

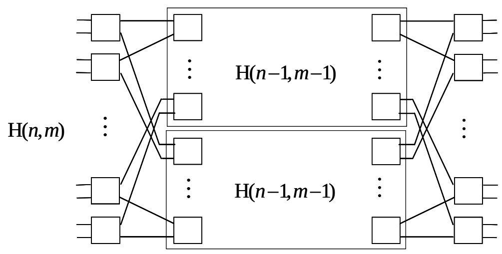


Fig.4 Recursive definition of an H($n$, $m$) network ($m \ge 1$).

In an H($n$, $m$) network with $m \ge 1$, the upper (lower) output of each input switch is connected to an input of the upper (lower) H($n$-1, $m$-1) network, respectively, and the upper (lower) input of each output switch is linked from an output of the upper (lower) H($n$-1, $m$-1) network, respectively. This recursive definition of an H($n$, $m$) network will end at the central column of $2^m$ banyan networks H($n$-$m$, 0).

We divide both the $N$/2 input switches and $N$/2 output switches of an H ($n$, $m$) network into $T = 2^{\lfloor (n-m+1)/2 \rfloor}$-element set $SW_j^{[1]} = \{u_{T\cdot j},\ u_{T\cdot j+1},\ \cdots,\ u_{T\cdot j+T-1}\}$ and $SW_j^{[2]} = \{v_{T\cdot j},\ v_{T\cdot j+1},\ \cdots,\ v_{T\cdot j+T-1}\}$ for $0 \le j \le \frac{N}{2T} - 1$, where $u_0$, $u_1$,…, $u_{N/2-1}$ are the $N$/2 inputs switches and $v_0$, $v_1$ ,…,$v_{N/2-1}$ are the $N$/2 outputs switches. From the definition of H($n$,$m$) network, $T$ input switches in $SW_j^{[1]}$ are linked to $T$ switches in the next stage ($T$/2 is in the upper H($n$-1, $m$-1) network and the other $T$/2 is in the lower H($n$-1, $m$-1) network), and no other input switches are linked to any of those $T$ switches in the next stage. By the symmetry of H($n$, $m$) network, $T$ output switches in $SW_j^{[2]}$ are linked from $T$ switches in the previous stage ($T$/2 is in the upper H($n$-1, $m$-1) network and the other $T$/2 is in the lower H($n$-1, $m$-1) network), and no other output switches are linked from any of those $T$ switches in the previous stage. For a CRPP to the network, we know from the definition of CRPP that there are just two active outputs from each set $SW_j^{[1]}$ and there are just two active inputs to each set $SW_j^{[2]}$. Note that a pair of input/output mapping in an H($n$, $m$) network can be realized by starting from the corresponding input switch via either upper or lower H($n$-1, $m$-1) network and ending at the corresponding output switch. We will show here that a CRPP to the H($n$,$m$) network can be realized crosstalk-free in the network if a proper routing algorithm is used to evenly distribute the CRPP to the upper and lower sub networks H($n$-1,$m$-1), such that the routing process can be performed in a recursive way. To achieve this, we construct a bipartite graph $G = (V_1,V_2 ; E)$ for the CRPP as follows. The vertex sets of $G$ are defined as:

$$V_1 = \left\{SW_0^{[1]}, SW_1^{[1]}, \cdots, SW_{\frac{N}{2T}-1}^{[1]}\right\},\ V_2 = \left\{SW_0^{[2]}, SW_1^{[2]}, \cdots, SW_{\frac{N}{2T}-1}^{[2]}\right\}$$

The edge set $E$ is defined as: for any one-pair mapping $(x_j,y_j)$ in the CRPP, if input switch corresponding to $x_i$ belongs to $SW_j^{[1]}$ and output switch corresponding $y_i$ belongs to $SW_j^{[2]}$, then there is an edge between vertex $SW_j^{[1]}$ and vertex $SW_j^{[2]}$ in $E$. We also assign each edge in $E$ a label representing the corresponding one-pair mapping in the CRPP. Note that each vertex of the graph has degree two, so the Euler Split algorithm [14] can evenly split the bipartite graph $G$ into two bipartite graphs $G_1 = (V_1,V_2 ; E_1)$ and $G_2 = (V_1,V_2 ; E_2)$. The basic idea here is, for a one-pair mapping $(x_j, y_j)$ corresponding to an edge in $E_1$, let the input $x_i$ pass the corresponding input switch, connect to the upper sub network H($n$-1,$m$-1), then pass through the corresponding output switch, and finally reach $y_i$; for a one-pair mapping corresponding to a edge in $E_2$, we implement it via the lower sub network H($n$-1, $m$-1). It is easy to see that the partial permutation corresponding to all edges in $E_1$ is also a CRPP to the sub network H($n$-1,$m$-1), so is the partial permutation corresponding to all edges in $E_2$. Then the routing algorithm for a CRPP in an H($n$,$m$) network can be obtained by using the Euler Split algorithm recursively.

**Routing Algorithm for CRPP in a H (*n*, *m*) network**

**Step 1**: If $m$ is 0, make the connection of the CRPP in the banyan network H($n$-$m$, 0) taking advantage of self-routing property of banyan network; exit.

**Step 2**: Construct the bipartite graph $G = (V_1,V_2 ; E)$ corresponding to the CRPP in the H($n$, $m$) network.

**Step 3**: Use the Euler Split algorithm to split the bipartite graph $G$ into two bipartite graphs $G_1 = (V_1, V_2 ; E_1)$ and $G_2 = (V_1,V_2 ; E_2)$.

**Step 4**: Take the partial permutation corresponding to the edges in $E_1$ to form one upper-sub-CRPP, and make the connection of this upper-sub-CRPP through the upper sub network H($n$-1,$m$-1); take the partial permutation corresponding to the edges in $E_2$ to form another lower-sub-CRPP, and make the connection of this lower-sub-CRPP through the lower sub network H($n$-1, $m$-1).

**Step 5**: Recursively call the Routing Algorithm for the upper-sub-CRPP in the upper sub network H($n$-1,$m$-1).

**Step 6**: Recursively call the Routing Algorithm for the lower-sub-CRPP in the lower sub network H ($n$-1,$m$-1).

**End**

Note that a CRPP for a H($n$, $m$) network eliminates crosstalk at least in the first and the last stages in the network, and the routing algorithm guarantee that a CRPP for a H($n$, $m$) network can be decomposed into two sub-CRPPs for its two the sub networks H($n$-1, $m$-1), and this decomposition process can be performed in a recursive way until to the central column of $2^m$ banyan networks H($n$-$m$, 0), so the CRPP for the H($n$, $m$) network can realized crosstalk-free in both the first $m$ stages and $m$ stages of the network. Since the Routing Algorithm above can finally decompose a CRPP for the H($n$, $m$) network into $2^m$ sub-CRPPs for those $2^m$ banyan networks H($n$-$m$, 0), and a CRPP for a banyan network just meet crosstalk-free requirements specified in Lemma 2 and thus crosstalk-free realizable in the banyan network taking advantage of self-routing property, thus a CRPP for the H($n$, $m$) network can also be realized crosstalk-free in the central $n$-$m$ stages of the network. QED.

In the proof of the theorem, we have actually developed a routing algorithm for realizing a CRPP crosstalk-free in an H($n$, $m$). Since Steps 2-4 take $O(N/T)$ time and Steps 5-6 call the same algorithm for network H ($n$-1, $m$-1), so time complexity of the routing algorithm for a CRPP is $O(mN/T)$. Recall that we need to route $T = 2^{\lfloor (n-m+1)/2 \rfloor}$ CRPPs in a RVH($n$, $m$) network consisting of $T = 2^{\lfloor (n-m+1)/2 \rfloor}$ vertically stacked copies of H($n$, $m$) network, so the overall all routing complexity is $O(Nm)$.

Based on the two algorithms above, we can realize a permutation crosstalk-free in a RVH ($n$, $m$) network by first decomposing a permutation into $2^{\lfloor (n-m+1)/2 \rfloor}$ CRPPs using the CRPP decomposition algorithm and then realizing each of these CRPPs in one copy of H($n$, $m$) network using the routing algorithm for CRPP.

**Example 3** Realizing the permutation (9) crosstalk-free in a RVH(4,2) network.

From Example 2 we know that the permutation (9) can be decomposed into two CRPPs (10) and (11) for the two H(4,2) networks in the RVH(4,2) network. For each CRPP, we need to call the Routing Algorithm above two times to realize the CRPP crosstalk-free in one H(4,2) network.

For CRPP (10), the bipartite graph and edge traverses are shown in Fig.5.

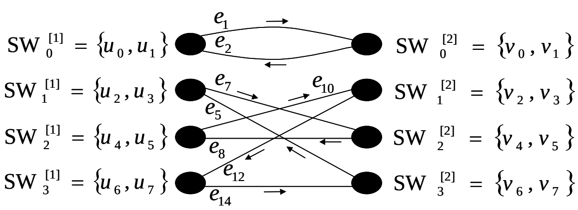


Fig.5 Decomposition of CRPP (10) for H (4,2) network.

Then the pairs $e_1$, $e_7$, $e_{10}$ and $e_{14}$ corresponding to the edges in $E_1$ form one sub-CRPP (12) for the upper H(3,1) network in the H (4,2) network,

$$\begin{pmatrix} 1 & 7 & 10 & 14 \\ 2 & 11 & 6 & 15 \end{pmatrix} \tag{12}$$

and the pairs $e_2$, $e_5$, $e_8$ and $e_{12}$ corresponding to the edges in $E_2$ form another sub-CRPP (13) for the lower H(3,1) network in the H (4,2) network.

$$\begin{pmatrix} 2 & 5 & 8 & 12 \\ 0 & 13 & 9 & 5 \end{pmatrix} \tag{13}$$

Sub-CRPPs (12) and (13) will be implemented via the upper and lower H(3,1) network in the H (4,2) network, respectively. To implement the sub-CRPP (12) via the upper H(3,1) network, we need to call the routing algorithm again. For the upper H(3,1) network and sub-CRPP (12), the bipartite graph and edge traverses are shown in Fig.6.

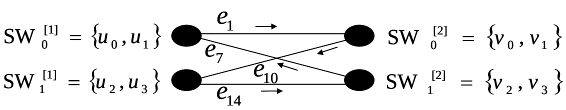


Fig.6 Decomposition of sub-CRPP (12) for upper H(3,1) network.

Then the pairs $e_1$ and $e_{14}$ corresponding to the edges in $E_1$ form one sub-CRPP (14) for the upper banyan network H(2,0) in the upper H(3,1) network,

$$\begin{pmatrix} 1 & 14 \\ 2 & 15 \end{pmatrix} \tag{14}$$

and the pairs $e_7$ and $e_{10}$ corresponding to the edges in $E_2$ form another sub-CRPP (15) for the lower banyan network H(2,0) in the upper H(3,1) network.

$$\begin{pmatrix} 7 & 10 \\ 11 & 6 \end{pmatrix} \tag{15}$$

For the lower H(3,1) network and sub-CRPP (13), the bipartite graph and edge traverses are shown in Fig.7.

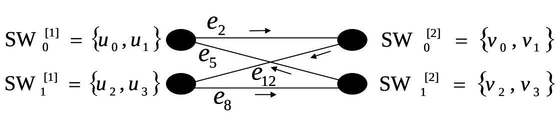


Fig.7 Decomposition of sub-CRPP (13) for lower H(3,1) network.

Then the pairs $e_2$ and $e_8$ corresponding to the edges in $E_1$ form one sub-CRPP (16) for the upper banyan network H(2,0) in the lower H(3,1) network,

$$\begin{pmatrix} 2 & 8 \\ 0 & 9 \end{pmatrix} \tag{16}$$

and the pairs $e_5$ and $e_{12}$ corresponding to the edges in $E_2$ form another sub-CRPP (17) for the lower banyan network H(2,0) in the lower H(3,1) network.

$$\begin{pmatrix} 5 & 12 \\ 13 & 5 \end{pmatrix} \tag{17}$$

Then the four sub-CRPPs (14), (15), (16) and (17) can be implemented in the four H(2,0) banyan networks of the H(4,2) network by taking the advantage of self-routing property of banyan network. The crosstalk-free routing of CRPP (11) in another H (4,2) network can be performed in a similar way. The final crosstalk-free routing of the permutation (9) in the two H(4,2) networks of the RVH(4,2) network is illustrated in Fig.8.

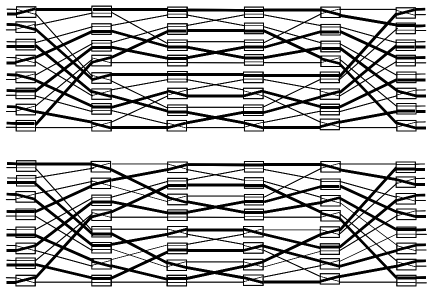

Fig.8 Crosstalk-free routing of permutation (9) in the two H (4,2) networks of a RVH (4,2) network.

## 4. Time complexity verses hardware cost

Recall that the main task in both the CRPP Decomposition Algorithm and the Routing Algorithm for CRPP is to construct and traverse bipartite graphs. Let $\alpha$ be the unit time for constructing and traversing a edge of a bipartite graph, then it is easy to see that the time complexity of CRPP Decomposition Algorithm for a RVH ($n$, $m$) network is approximately $\alpha\cdot N\lfloor(n-m+1)/2\rfloor \cong \alpha\cdot N\cdot(\log_2 N-m)/2$ and the time complexity of CRPP Routing Algorithm for a RVH ($n$, $m$) network is $\alpha\cdot N\cdot m$. After running the CRPP Decomposition Algorithm and the Routing Algorithm for CRPP, the switching setup can be performed as that of self-routing, so the complexity of switching setup in a RVH ($n$, $m$) network is proportional to the number of stages and therefore $\beta\cdot(\log_2 N+m)$, here $\beta$ is the unit time to setup the switches in one stage. Thus, the overall time complexity of realizing a crosstalk-free permutation in a RVH ($n$, $m$) network is approximately $\alpha\cdot N\cdot(\log_2 N-m)/2 + \alpha\cdot N\cdot m + \beta\cdot(\log_2 N+m) \cong \alpha\cdot N\cdot(\log_2 N+m)/2$, where $0\le m\le \log_2 N-1$.

Since the hardware cost of network are mainly determined by the number of switches in the network, and a RVH($n$, $m$) network consists of $2^{\lfloor(n-m+1)/2\rfloor} = 2^{\lfloor(\log_2 N-m+1)/2\rfloor}$ vertically stacked copies of H($n$,$m$) network with each H($n$,$m$) network having $(\log_2 N+m)\cdot N/2$ switches, so the overall hardware cost of RVH($n$,$m$) network is approximately $2^{\lfloor(\log_2 N-m+1)/2\rfloor}\cdot(\log_2 N+m)\cdot N/2$ . When $N$=4096, the time complexity and hardware cost vs. the number of extra stages of a RVH($n$,$m$) network is illustrated in Fig.9. We can see from Fig.9 that by increasing the number of extra stages in a RVH($n$,$m$) network, the hardware cost in term of number of switches in the networks can be reduced, but the overall time complexity of realizing a crosstalk-free permutation in the network will be increased correspondingly. Since the signal attenuation and transmission delay in an optical MIN are proportional to the number of couplers that a light signal passes through, so a suitable number of extra stages of a RVH($n$,$m$) network should be determined depending on the trade-off among the hardware cost, time complexity and signal attenuation of the network.

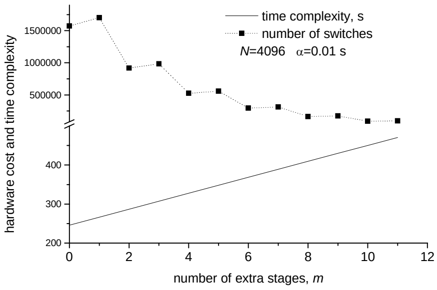


Fig.9 Hardware cost and time complexity vs. the number of extra stages in a RVH ($n$, $m$) network (notice that the two curves do not follow the same ordinate scale: one is time and the other is the number of switches).

## 5. Conclusions

A rearrangeable optical MIN can be constructed by using a combination of horizontal expansion and vertical stacking of banyan networks, and a permutation can be realized crosstalk-free in this kind of optical MIN by first decomposing it into multiple crosstalk-free realizable partial permutations (CRPPs) and then realizing each of CRPPs in a stacked plane of the MIN. In general, the necessary number of planes and thus hardware cost of this kind of optical MINs can be reduced by appending more number of extra stages to the back of banyan networks, but at the cost of increasing overall time complexity and signal attenuation. A careful trade-off among hardware cost, time complexity, transmission delay and signal attenuation is critical in designing an optical MIN built on a combination of horizontal expansion technique and vertical stacking technique.

## Acknowledgement

This work is support by Telecommunications Advancement Organization of Japan. The authors would like to thank the anonymous reviewers for their valuable comments.

**References**

[1] V.R. Chinni, T.C. Huang, P.K.A. Wai, C.R. Menyuk and G.J. Simonis., "Crosstalk in a lossy directional coupler switch," *J.Lightwave Technol.*, vol.13, no.7, pp.1530-1535,July 1995.

[2] H.S.Hinton, *An introduction to Photonic Switching Fabrics*, New York: Plenum, 1993.

[3] G.R.Goke and G.J.Lipovski, "Banyan networks for partitioning multiprocessor systems," *Proc.1st Annu. Symp. Comp. Arch*,, pp.21-28,1973.

[4] C.Kruskal and M.Snir, "The performance of multistage interconnection networks for multiprocessors," *IEEE Trans. Commun*., vol.COM-32, pp.1091-1098, Dec.1983.

[5] J.Patel, "Performance of processor-memory interconnections for multiprocessors," *IEEE Trans. Comput*., vol.C-30, pp.771-780, Oct.1981.

[6]M.M.Vaez and C.-T. Lea, "Strictly nonblocking directional-coupler-based switching networks under crosstalk constraint," *IEEE Trans. Commun*., vol.48,no.2, pp.316-323, Feb. 2000.

[7]M.M.Vaez and C.-T. Lea, "Space-wavelength tradeoff in the design of nonblocking directional coupler based network under crosstalk constraint," *J. Lightwave Technol*.,vol.16,pp.1373-1379,Aug.1998.

[8]G.Maier and A.Pattavina, "Design of photonic rearrangeable networks with zero first-order switching-element-crosstalk, " *IEEE Trans. Commun*., vol.49,no.7, pp.1268-1279, July.2001.

[9]K.Padmanabhan and A.Netravali, "Dilated networks for photonic switching," *IEEE Trans. Commun*., vol.COM-35, pp.1357-1365, Dec.1987.

[10] D.Li, "Elimination of crosstalk in directional coupler switches," *Optical Quantum Electron.*, vol.25, no.4, pp.255-260, Apr.1993.

[11] T.-S. Wong and C-T. Lea, "Crosstalk reduction through wavelength assignment in WDM photonic switching networks, " *IEEE Trans. Commun*., vol.49, no.7,pp.1280-1287, July.2001.

[12] M.M.Vaez and C.-T. Lea, "Wide-sense nonblocking Banyan-type switching systems based on directional couplers, " *IEEE J. Select. Areas Commun*., vol.16, pp.1327-1332, Sept.1998.

[13] 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.

[14] R.Cole and J.Hopcroft, "On edge coloring bipartite graphs", SIAM J.Comput.,vol.11, pp.540-546,1982.

[15] C.Lea and D.Shyy, "Tradeoff of horizontal decomposition versus vertical stacking in rearrangeable nonblocking networks," *IEEE Trans. Commun*., vol.39, pp.899-904, June 1991.
[16] I.Anderson, *Combinatorics of Finite Sets*, Oxford Science Publications, 1987.
[17] Xiaohong Jiang, Hong Shen, Md.M.R.Khandker and Susumu Horiguchi, "Permutation in Rearrangeable Nonblocking Optical MINs with Zero First-Order Switching-Element-Crosstalk ", *Proc. of the 2002 IEEE Workshop on High Performance Switching and Routing*, pp.19-23, May.2002, Kobe, Japan.

## Biographies :

**Xiaohong Jiang** received his B.S., M.S. and Ph.D degrees in 1989, 1992, and 1999 respectively, all from Xidian University, Xi'an, China. He is currently a research associate in the Graduate School of Information Science, Japan Advanced Institute of Science and Technology (JAIST). Dr. Jiang was a JSPS (Japan Society for the Promotion of Science) postdoctoral research fellow at JAIST from Oct.1999-Oct. 2001. He was a research associate in the Department of Electronics and Electrical Engineering, the University of Edinburgh from Mar.1999-Oct.1999. Dr. Jiang's research interests include interconnection networks, optical switch networks, WDM networks, IC yield modeling, timing analysis of digital circuits, clock distribution and fault-tolerant technologies for VLSI/WSI. He has published over 30 technical papers in these areas. He is member of IEEE. His email address is: jiang@jaist.ac.jp.

**Hong Shen** received his B.Eng. degree from Beijing University of Science and Technology, M.Eng. degree from University of Science and Technology of China, Ph.Lic. and Ph.D. degrees from Abo Akademi University, Finland, all in Computer Science. He is currently a full professor in the Graduate School of Information Science, Japan Advanced Institute of Science and Technology. Previously he was a professor in Griffith University, Australia. Dr. Shen has published over 140 technical papers on algorithms, parallel and distributed computing, interconnection networks, parallel databases and data mining, multimedia systems and networking. He has served as an editor of Parallel and Distributed Computing Practice, associate editor of International Journal of Parallel and Distributed Systems and Networks, editorial-board member of Parallel Algorithms and Applications, International Journal of Computer Mathematics and Journal of Supercomputing, and chaired various international conferences. Dr. Shen is a recipient of 1991 National Education Commission Science and Technology Progress Award and 1992 Sinica Academia Natural Sciences Award. His email address is: shen@jaist.ac.jp.

**Md. Mamun-ur-Rashid Khandker** received his B.Sc. (Hons.) and M.Sc. from Rajshahi University, Bangladesh and since then he was serving as a faculty member in the department of Applied Physics & Electronics of the same university. He is currently a student of doctor course at Japan Advanced Institute of Science and Technology (JAIST), Japan. His research interest includes interconnection networks, optical switch networks and multiwavelength photonic packet switching. He is now working on multiwavelength address lookup for ultra-large optical networks. He has got a US patent. He is also a student member of IEEE. His email address is: khandker@jaist.ac.jp.

**Susumu Horiguchi** (S'79, M'81, SM'95) received the B.E., MS. and PhD degrees from TOHOKU University in 1976, 1978 and 1981 respectively. He was on a faculty of Department of Information Science at Tohoku University from 1981 to 1992. He was a visiting scientist at IBM Thomas J. Watson Research Center from 1986 to 1987. Since 1992, he has been a professor in the Graduate School of Information Science at JAIST (Japan Advanced Institute of Science and Technology) and has been conducting his research group as the chair of Multi-Media Integral System Laboratory at JAIST. He has been involved in organizing many international workshops, symposia and conference sponsored by IEEE, ACM, IASTED, IEICE and IPS. His research interest has been mainly concerned with optical switch interconnection, interconnection networks, GRIDs computing, parallel computer architecture, and VLSI/WSI architecture. He is currently serving as Editors for IEICE Transaction on Information and Systems and for Journal of Interconnection Networks. His email address is: hori@jaist.ac.jp.