TAIWANESE JOURNAL OF MATHEMATICS
Vol. 10, No. 5, pp. 1225-1243, September 2006
This paper is available online at http://www.math.nthu.edu.tw/tjm/

CROSSTALK-FREE REARRANGEABLE MULTISTAGE INTERCONNECTION NETWORKS

Xuewen Bao and Qiao Li

Abstract. In this paper, the notion of crosstalk-free rearrangeability (CF-rearrangeability) of multistage interconnection networks (MINs) is formally defined. Using the concept of line digraphs from graph theory, we show that the problem of crosstalk-free routing on any bit permutation network (BPN) is always equivalent to the classical permutation routing problem on a BPN of smaller size and with fewer stages. We also show the CF-rearrangeability and minimality (in stage number) of three families of BPNs, including the dilated Benes network. Some necessary conditions for a BPN to be CF-rearrangeable are given, and a brief discussion of CF-rearrangeable networks with dilation in time or space is included.

1. Introduction

Multistage interconnection networks (MINs) have been an important interconnection scheme to realize permutations. A MIN is a staged network connecting N inputs to N outputs, where each stage consists of a number of crossbars and consecutive stages are connected by left-to-right directed edges. To realize a permutation, N edge-disjoint paths are to be found to connect the N input-output pairs. In the more recent application of photonic switching, each crossbar can be used at most once by the paths to avoid crosstalk.

Unless otherwise stated, we assume all crossbars are $d \times d$. We will ignore the physical feature of crossbars and treat them simply as nodes. As each input (output) is connected to a unique node in the first (last) stage, it does not play a role in permutation routing. So, instead of maintaining the input and the output lines, we

Received September 29, 2004; accepted October 28, 2004.
Communicated by Xuding Zhu.
2000 *Mathematics Subject Classification*: 68M07, 68M10, 90B18.
Key words and phrases: Multistage interconnection network, Bit permutation network, Crosstalk-free, Rearrangeability.
Partially supported by NNSFC Grant No. 10371075.

directly treat the nodes in the first (last) stage as inputs (outputs). Also, we deviate from the usual symbolism a little by calling a MIN of size $N \times N$ if it contains N nodes in the input (and output) stage, for simplicity of notations.

The interest of this paper is in the ability of networks to realize permutations rather than in the design of routing algorithms for a specific network. Therefore, we do not differentiate between topologically equivalent networks. Our attention is restricted to connected bit permutation networks (BPNs) which were first proposed by Chang, Hwang and Tong [6]. BPN is a very important class of networks. It includes most of the intensively studied MINs such as the Omega equivalent networks and their extra-stage versions, the shuffle-exchange networks and Benes networks. Also, many other networks of interest use BPNs as basic building blocks, such as those networks derived by vertical stacking or by horizontal concatenating.

In a MIN of size $N \times N$, a *t-permutation* is a set of input-output pairs in which each input and output node appears exactly t times. The network is said to *realize a t-permutation conflict-free* if there are tN edge-disjoint paths in the network, each connecting an input-output pair. For $t = 1$, the network is said to *realize a 1-permutation crosstalk-free* if there are N node-disjoint paths connecting the input-output pairs. A d-nary network is *rearrangeable* if it can realize any d-permutation conflict-free, and is *crosstalk-free rearrangeable (CF-rearrangeable)* if it can realize any 1-permutation crosstalk-free.

The central idea of this paper is to use the well-known concept of line digraphs in graph theory. It was noted by several authors that the conflict-free routing on a MIN is equivalent to the crosstalk-free routing on its line digraph [9, 10, 12]. Since crosstalk-free routing is a new problem arising from photonic communication, it is desirable to convert it to the classical problem of conflict-free routing. Works dealing with this conversion can be found in the literature, but they are restricted to very limited number of networks.

Hwang and Yen [9] showed that the line digraph of any BPN is also a BPN. They also showed how to compute the line digraph of any BPN. This result gives a method to convert the conflict-free routing on BPNs to the crosstalk-free routing on their line BPNs, but is inconvenient to do the conversion in the opposite direction. In particular, it cannot answer the question if the crosstalk-free routing on an arbitrary BPN can be converted to the conflict-free routing on another BPN. In this paper, we continue Hwang and Yen's work by presenting a method which computes a MIN whose line digraph is the given BPN. This MIN is called a *root digraph* of the BPN. It is seen that the root digraph obtained by our method is either a BPN or a MIN which can be derived easily from a functionally equivalent BPN of smaller size and with fewer stages. This implies that the problem of crosstalk-free routing on a BPN is always equivalent to the classical conflict-free routing on a smaller BPN, and that, from a theoretical view of point, crosstalk-free routing is not a new problem which needs to be attacked from scratch. Based on the result, we are able to establish some general results about CF-rearrangeable BPNs. The dilated Benes network is shown to be a CF-rearrangeable network with minimum number of stages for a given network size, and two new classes of CF-rearrangeable networks with minimum number of stages are presented. We also give a brief discussion of CF-rearrangeable networks with dilation in time or space.

The rest of this paper is organized as follows. Section 2 contains the definition and basic results of BPNs. Section 3 presents a method to compute a root digraph of an arbitrary BPN. The discussion of CF-rearrangeable networks is included in Section 4 and concluding remarks are made in Section 5.

2. Bit Permutation Networks

Let G be an s-stage MIN of size $N \times N$, where $N = d^n$. The stages of G are numbered from left to right by $1, 2, \cdots, s$ and are denoted by $V_1, V_2, \cdots, V_s$ respectively. The set of edges between stage i and stage $i+1$ is denoted by E_i and is referred to as the i-th edge stage.

Bit permutation networks were first defined and studied by Chang, Hwang and Tong [6]. The definition and some basic results are given in the following.

Definition 1. Let G be an s-stage network. G is called a bit permutation network if the nodes in every stage can be labelled by the d-nary numbers of length n which admits the existence of $s-1$ permutations $p_1, p_2, \cdots, p_{s-1}$ on set $\{1, 2, \cdots, n\}$ and $s-1$ integers $a_1, a_2, \cdots, a_{s-1} \in \{1, 2, \cdots, n\}$, such that, for any $i\,(1 \le i \le s-1)$, node $(x_1, \cdots, x_n)$ in stage i is adjacent to node $(y_1, \cdots, y_n)$ in stage $i+1$ if and only if $y_j = x_{p_i(j)}$ for all j except for possibly $j = a_i$.

We can see from the definition that, if the bits in the labels of all nodes in V_i are permuted by p_i: $(x_1, \cdots, x_n) \rightarrow (x_{p_i(1)}, \cdots, x_{p_i(n)})$, then E_i connects a node in V_i to a node in V_{i+1} if and only if their labels are different at most at the a_i-th bit. Obviously the Omega equivalent networks, the shuffle-exchange networks and Benes network all fit into this definition, therefore they are all BPNs. In [6], Chang, Hwang and Tong also proved that the permutations of bits, $p_1, p_2, \cdots, p_{s-1}$, can always be set to identity by wisely labelling the nodes.

Theorem 2.1. *G is a BPN of s-stage if and only if the nodes in every stage can be labelled by the d-nary numbers of length n which admits the existence of $s-1$ integers $a_1, a_2, \cdots, a_{s-1} \in \{1, 2, \cdots, n\}$ such that, for any $i\,(1 \le i \le s-1)$, node $(x_1, \cdots, x_n)$ in stage i is adjacent to node $(y_1, \cdots, y_n)$ in stage $i+1$ if and only if $y_j = x_j$ for all j except for possibly $j = a_i$.*

Chang, Hwang and Tong used the vector $(a_1, a_2, \cdots, a_{s-1})$ to represent the BPN described in Theorem 2.1, called the *sequence notation*. The sequence notation of a BPN is not unique. It is implied in [6] that each sequence naturally defines a partition of the set $\{E_1, E_2, \cdots, E_{s-1}\}$ in which E_i and E_j belong to the same subset if and only if $a_i = a_j$. Chang, Hwang and Tong proved that two BPNs are topologically equivalent if and only if their sequences induce the same partition. Theorem 2.3 below gives a characterization of BPNs which does not depend on the existence of a labelling system of nodes [14]. It is also a generalization of a theorem by Bermond et al. [4], a well-known result which first characterizes the Omega equivalent networks graph-theoretically. More characterizations can be found in [2, 14].

Denote by $G_{i,j}$ the subnetwork of G induced by the nodes in V_i through V_j. Let π be a partition of the set $\{E_1, E_2, \cdots, E_{s-1}\}$, and let $t_{i,j} = |\{[E_i], [E_{i+1}], \cdots, [E_{j-1}]\}|$, where $[E_i]$ is the equivalence class containing E_i and $|A|$ stands for the number of different elements in A. The following definitions are derived from [1, 4. 14].

Definition 2. Let $i \leq j$.

(i) G is said to satisfy the Banyan property if each pair of input and output nodes is connected by a unique path.

(ii) G is said to satisfy the $P(i, j)$ property if $G_{i,j}$ contains exactly 2^{n-j+i} connected components, and is said to satisfy the $P(*, *)$ property if it satisfies $P(i, j)$ for all possible ordered pairs (i, j).

(iii) G is said to satisfy the $P_\pi(i, j)$ property if $G_{i,j}$ contains exactly $d^{n-t_{i,j}}$ connected components, and is said to satisfy the $P_\pi(*, *)$ property if it satisfies $P_\pi(i, j)$ for all possible ordered pairs (i, j).

(iv) G is said to satisfy the buddy property if any edge stage consists of disjoint 4-cycles.

(v) G is said to satisfy the extended buddy property $Q(i, j)$ if for any $u, v \in V_i$, the sets of nodes in V_j reachable from u and v are either the same or disjoint. G satisfies $Q(*, *)$ if it satisfies $Q(i, j)$ for all possible ordered pairs (i, j).

Theorem 2.2. ([4]) *G is topologically equivalent to the binary Omega network if and only if G satisfies the Banyan property and property $P(*, *)$.*

Theorem 2.3. ([14]) *G is a bit permutation network if and only if there exists a partition π of $\{E_1, E_2, \cdots, E_{s-1}\}$ such that G satisfies properties $P_\pi(*, *)$ and $Q(*, *)$.*

When $d = 2, s = n$, and each equivalent class of π contains exactly one element, then the $P_\pi(i,j)$ and $P_\pi(*,*)$ property are reduced to the $P(i,j)$ and $P(*,*)$ property of Theorem 2.2, respectively.

In [14], it was also proved that two BPNs are topologically equivalent if and only if they have the same partition of edge stages. Simple analysis reveals that the partition in Theorem 2.3 and the partition induced by sequence notations are identical.

By Theorem 2.3, the number of subsets in π is exactly n in an $N \times N$ connected BPN. When a BPN is considered, we always assume a proper labelling already exists such that the sequence notation of Theorem 2.1 is applicable. For example, we will use $(1, 2, \cdots, n)$, $(1, \cdots, n, n, \cdots, 1)$ and $(1, \cdots, n-1, n, n-1, \cdots, 1)$ to denote the $N \times N$ Omega network, Benes network and the dilated Benes network [13]. Fig. 1 shows four binary BPNs and their sequence notations, where nodes in each stage are labelled by 0000, ..., 1111 from top to bottom.

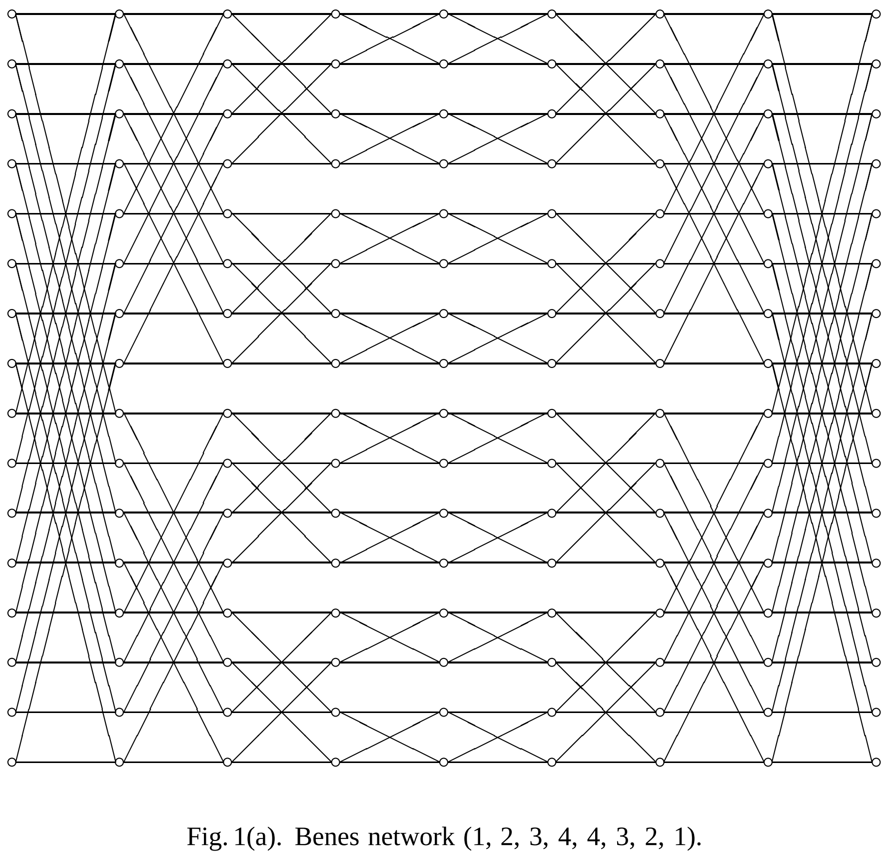

Fig. 1(a). Benes network (1, 2, 3, 4, 4, 3, 2, 1).

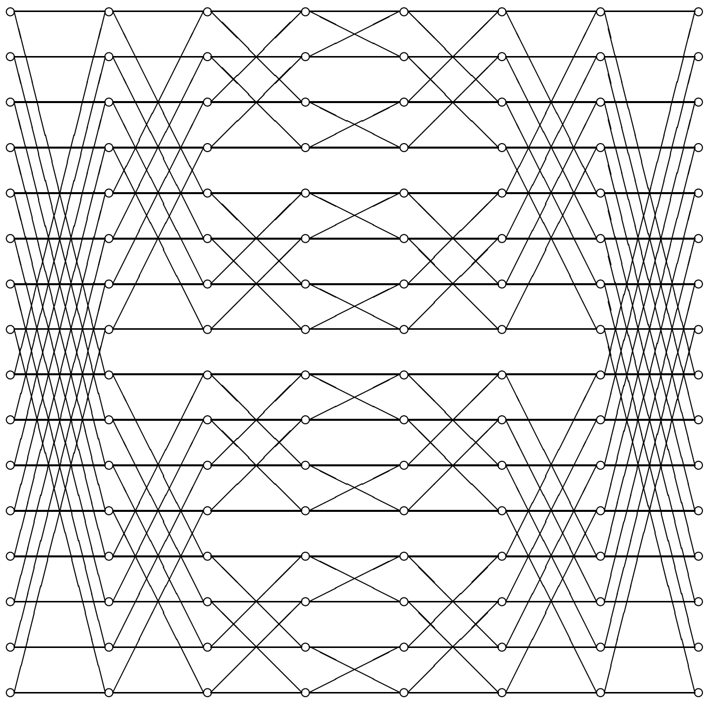

Fig. 1(b). Dilated Benes network (1, 2, 3, 4, 3, 2, 1).

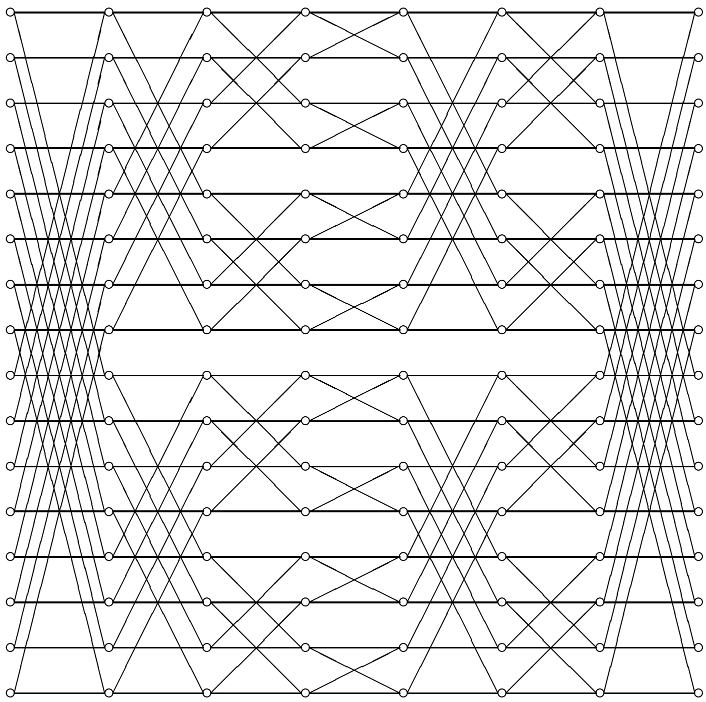

Fig. 1(c). BPN (1, 2, 3, 4, 2, 3, 1).

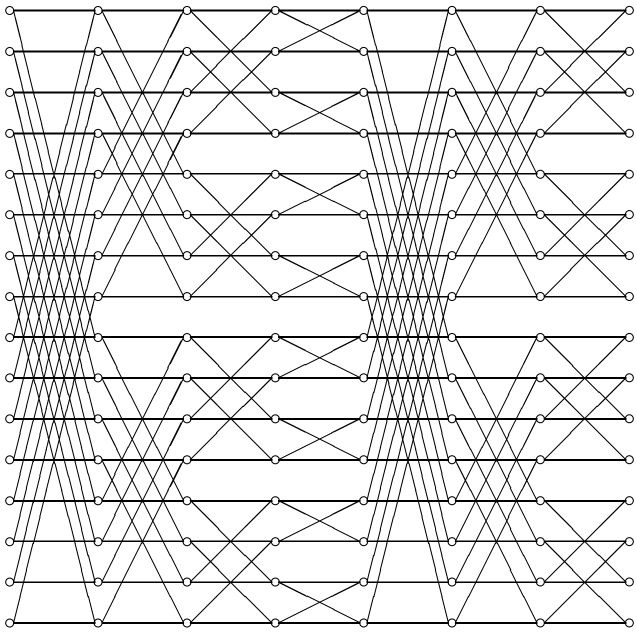

Fig. 1(d). BPN (1, 2, 3, 4, 1, 2, 3).

3. Root Digraph of BPNs

Let G be a digraph. The *line digraph* of G, denoted by $L(G)$, has the edges of G as its nodes and there is an edge from (u_1, v_1) to (u_2, v_2) in $L(G)$ if and only if $v_1 = u_2$. If $H = L(G)$, we say G is a *root digraph* of H and denote it by $G = L^{-1}(H)$.

Recall that all MINs considered in this paper do not have input and output lines. But we make an exception when defining their line digraphs. For a d-nary MIN G, we obtain a digraph by adding d edges incident to each input (output) node in the first (last) stage. This is equivalent to adding the input (output) lines back, and treating them as ordinary edges. The line digraph of this digraph is called the *line digraph of MIN G*.

Readers can easily check that the difference in defining the line digraph of a general digraph and the line digraph of a MIN does not affect the correctness of all results in the remaining part of this paper. Thus we still denote the line digraph of a MIN G by $L(G)$. Obviously, $L(G)$ is a MIN with the input (output) lines of G becoming its input (output) nodes.

The importance of line digraphs is in that a set of edge-disjoint paths in a digraph always corresponds to a set of node-disjoint paths in its line digraph. Since the conflict-free routing on an electronic MIN is to find edge-disjoint paths and the crosstalk-free routing on a photonic MIN to find node-disjoint paths, it is justifiable to unify them as one problem via line digraphs, either for theoretical analysis or even for design of routing algorithms. This suggests that the classical theory in electronic MINs might not become old-fashioned as a result of technology advancement.

Next we will restrict to connected BPNs. To bridge the gap between the two problems, two questions naturally arise: (1) what is the line digraph of a BPN? and more importantly, (2) does a BPN always have a BPN as its root digraph? if not, what can its root digraph be like? The idea of using line digraphs to tackle crosstalk-free routing problem was researched by several authors (sometimes under different names) [9, 10, 12, 13]. But almost all these works were in the line of the first question. A complete answer to the first question was by Hwang and Yen [9]. They proved that the line digraph of any BPN is also a BPN. They also gave a method to compute a sequence of $L(G)$ from the sequence of G.

Theorem 3.1. *If G is a BPN of size $N \times N$ with sequence $(a_1, a_2, \cdots, a_{s-1})$, then $L(G)$ is a BPN of size $dN \times dN$ with sequence $(b_1, b_2, \cdots, b_s)$, where $b_1 = n+1$ and, for $t = 2, \cdots, s$, $b_t = a_{t-1}$ if $a_{t-1} \notin \{a_1, \cdots, a_{t-2}\}$, or $b_t = b_i$ if $a_{t-1} \in \{a_1, \cdots, a_{t-2}\}$ and $i = max\{j \mid a_j = a_{t-1}, 1 \le j \le t-2\}$.*

Let the edge stages of G and $L(G)$ be $\{E_1, E_2, \cdots, E_{s-1}\}$ and $\{F_1, F_2, \cdots, F_s\}$, respectively, and let π_G, $\pi_{L(G)}$ be their partitions. We implement the above method by procedure LINEGRAPH in terms of π_G and $\pi_{L(G)}$.

Procedure LINEGRAPH
Input: $\pi_G = A_1 \cup A_2 \cup \cdots \cup A_n$, where each A_k ($1 \le k \le n$) is an ordered set in which E_i precedes E_j if and only if $i < j$.
Output: $\pi_{L(G)} = B_1 \cup B_2 \cup \cdots \cup B_{n+1}$.
begin
  Set $B_k = \emptyset$ ($1 \le k \le n+1$);
  Put F_1 into B_1;
  FOR $i = 2$ TO s
    IF E_{i-1} is the first element in $[E_{i-1}]$, put F_i into the first empty set among $B_1, \cdots, B_{n+1}$.
    ELSE IF E_k is the predecessor of E_{i-1} in $[E_{i-1}]$, put F_i into the same set as F_k belongs to.
end

If one wants to get a sequence notation from the partition of edge stages, just let a_k be a same integer in $\{1, 2, \cdots, n\}$ for all subscripts of E_k in one subset. When the above procedure ends, for example, let $a_k = j$ for $F_k \in B_j$, $k = 1, 2, \ldots, s$. Then $(a_1, a_2, \cdots, a_s)$ is a *canonical* sequence of $L(G)$ in the sense defined in [6], i.e., $i < j$ implies i appears before j in that sequence. We exemplify the method with the same BPN as presented in [9]: (1, 3, 3, 2, 2, 3, 1, 3, 1, 1, 2, 3, 2, 2, 1). It is of size $d^3 \times d^3$ and has 16 stages. The partition π_G of its edge stages is:

$$\begin{aligned} A_1 &= \{E_1, E_7, E_9, E_{10}, E_{15}\} \\ A_2 &= \{E_2, E_3, E_6, E_8, E_{12}\} \\ A_3 &= \{E_4, E_5, E_{11}, E_{13}, E_{14}\}. \end{aligned}$$

Using Procedure LINEGRAPH, we get the partition $\pi_{L(G)}$:

$$\begin{aligned} B_1 &= \{F_1, F_8, F_{13}, F_{15}\} \\ B_2 &= \{F_2, F_4, F_6, F_9, F_{11}, F_{14}\} \\ B_3 &= \{F_3, F_7, F_{10}, F_{16}\} \\ B_4 &= \{F_5, F_{12}\}. \end{aligned}$$

$L(G)$ is of size $d^4 \times d^4$ and has 17 stages. A sequence notation of it is (1, 2, 3, 2, 4, 2, 3, 1, 2, 3, 2, 4, 1, 2, 1, 3).

To answer the second question, one needs to first answer when a BPN has a root digraph. In 1964, Heuchenne [8] gave a characterization as follows.

Theorem 3.2. *Let H be a digraph. $H = L(G)$ for some digraph G if and only if*

(1) *H has no multiple edges, and*

(2) *if (u_1, v_1), (u_2, v_1) and (u_2, v_2) are edges of H, then (u_1, v_2) is also an edge of H.*

From Theorem 2.3, it is clear that any BPN satisfies Heuchenne's conditions. So every BPN is a line digraph. It should be noted that a line digraph may have several nonequivalent root digraphs [7]. Since this paper only concerns the problem of converting crosstalk-free routing to conflict-free routing on BPNs, we are interested in finding root digraphs which have the "closest" relation to BPNs. It will be seen that every root digraph obtained in the following is a MIN functionally equivalent to a BPN (in realizing permutations).

Lemma 3.3. *Let $s \geq 3$. If G is the line digraph of a BPN, then G is a BPN in which any two consecutive edge stages belong to different subsets of π_G.*

Proof. Let H be a BPN whose line digraph is G. Let u be a node in the i-th stage of H. Then u is connected to d nodes, $v_1, v_2, \cdots, v_d$, in the (i+1)-th stage of H. For each j, $1 \leq j \leq d$, v_j has d incoming edges and d outgoing edges. These $2d$ edges (as nodes of G) form a complete bipartite graph, $K^{(j)}_{d,d}$, which is a connected component in $G_{i+1,\,i+2}$. Any incoming edge of u in H is connected to $K_{d,d}^{(j)}$ in G. This means the d connected components $K_{d,d}^{(1)}, \cdots, K_{d,d}^{(d)}$ of $G_{i+1,\,i+2}$ belong to one connected component in $G_{i,\,i+2}$. So $G_{i,\,i+2}$ has fewer connected components than $G_{i+1,\,i+2}$ does. By Theorem 2.3, the i-th and the $(i+1)$-th edge stage of G belong to different subsets of π_G. ■

Fig. 2 illustrates the proof (for $d = 2$), where nodes and edges of H are represented by circles and lines, and nodes and edges of G are represented by solid circles and dotted lines.

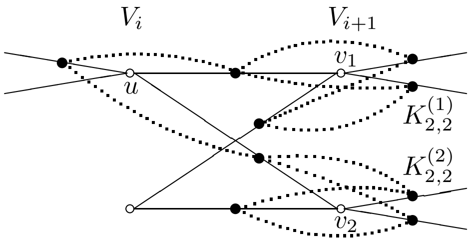

Fig. 2. Illustration of the proof of Lemma 3.3.

From Lemma 3.3, any two consecutive edge stages in the output BPN of Procedure LINEGRAPH belong to different subsets. Next we present a converse procedure, ROOTGRAPH, which recovers the input BPN of Procedure LINEGRAPH from its output. The validity proof is included in the proof of Lemma 3.4. Let the edge stages of G and $L^{-1}(G)$ be $\{E_1, E_2, \cdots, E_s\}$ and $\{F_1, F_2, \cdots, F_{s-1}\}$ respectively, and π_G, $\pi_{L^{-1}(G)}$ be their partitions.

Procedure ROOTGRAPH
Input: $\pi_G = A_1 \cup A_2 \cup \cdots \cup A_n$, where each A_k $(1 \le k \le n)$ is an ordered set in which E_i precedes E_j if and only if $i < j$.
Output: $\pi_{L^{-1}(G)} = B_1 \cup B_2 \cup \cdots \cup B_{n-1}$.
begin
Set $B_k = \emptyset$ $(1 \le k \le n-1)$;
FOR $i = s-1$ DOWNTO 1
IF E_i is the last element in $[E_i]$, put F_i into the first empty set among $B_1, \cdots, B_{n-1}$.
ELSE IF E_k is the successor of E_i in $[E_i]$, put F_i into the same set as F_{k-1} belongs to.
end

Lemma 3.4. *Let $s \ge 3$. If G is a BPN in which any two consecutive edge* *stages belong to different subsets of* π_G*, then* G *is the line digraph of the BPN obtained by Procedure ROOTGRAPH.*

Proof.

(1) If any two consecutive edge stages belong to different subsets of π_G, no E_i is followed immediately by E_{i+1} in a subset of π_G. So in Procedure ROOTGRAPH, k is never equal to $i+1$ and the subset which F_i is going into is well-defined. Therefore, Procedure ROOTGRAPH terminates.

(2) Obviously, the output $\pi_{L^{-1}(G)}$ of Procedure ROOTGRAPH is a partition of $\{F_1, \cdots, F_{s-1}\}$. Except for E_s, the last element of each subset in π_G accounts for an nonempty subset of $\pi_{L^{-1}(G)}$, so the number of nonempty subsets in $\pi_{L^{-1}(G)}$ is exactly $n-1$.

(3) Let the output of Procedure ROOTGRAPH be input to Procedure LINEGRAPH, and output π be got. We now show $\pi = \pi_G$. Remember that all subsets of π, π_G and $\pi_{L^{-1}(G)}$ are ordered sets in which E_i (F_i) precedes E_j (F_j) if and only if $i < j$.

If E_i is the predecessor of E_j in π_G, then $j > i+1$ since no two consecutive edge stages of G belong to a same subset. By the definition of ROOTGRAPH, when it is the turn of F_i to go into a subset, E_j is E_i's successor, so F_i goes into the same subset as F_{j-1} belongs to and becomes the predecessor of F_{j-1} in $\pi_{L^{-1}(G)}$. Again, by the definition of LINEGRAPH, when E_j is to be placed into a subset of π, F_i is F_{j-1}'s predecessor. So, F_j is put into the same subset as F_i belongs to and becomes the successor of F_i in π.

It is proved that, if E_i is the predecessor of E_j in π_G, E_i is the predecessor of E_j also in π. To show π and π_G are the same partition, it is sufficient to show that both have the same number of subsets. But this is readily got from the conclusion of (2). ■

By analysis similar to the proof above, we see also that, if the output of Procedure LINEGRAPH is the input of Procedure ROOTGRAPH, the original partition is retrieved. So the two procedures are converse to each other.

Theorem 3.5. *Let* $s \geq 3$*.* G *has a BPN as its root digraph if and only if* G *is a BPN in which any two consecutive edge stages belong to different subsets of* π_G*, or equivalently, the subnetwork* $G_{i,i+2}$ *has exactly* d^{n-2} *connected components for any* i*,* $1 \leq i \leq s-2$*.*

Proof. This can be easily derived from Lemma 3.3, Lemma 3.4, and Theorem 2.3. ■

When $s = 1$, G has no edge stage. The only 1-stage connected BPN consists of a single node, so its root digraph is not BPN. It is not difficult to see that there is only one 2-stage connected BPN either. That is the complete bipartite graph $K_{d,d}$, which has the single node as its root digraph.

Fig. 3(b)–(d) show the root digraphs (obtained by Procedure ROOTGRAPH) of the BPNs in Fig. 1(b)–(d). By Theorem 3.5, the root digraph(s) of any BPN which has two consecutive edge stages belonging to a same subset cannot be a BPN. For example, Fig. 3(a) shows a root digraph of the Benes network of Fig. 1(a) [10], which is not a BPN since any BPN contains no multiple edges between two nodes.

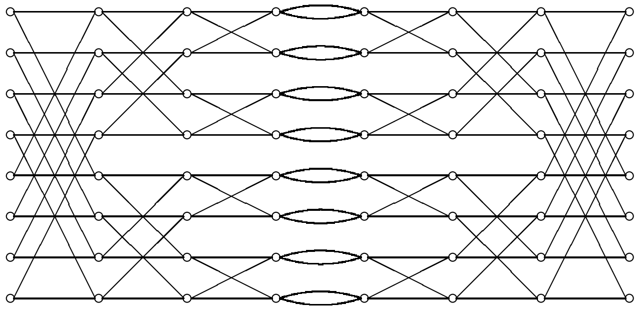

Fig. 3(a). (1, 2, 3):(3, 2, 1), root digraph of (1, 2, 3, 4, 4, 3, 2, 1).

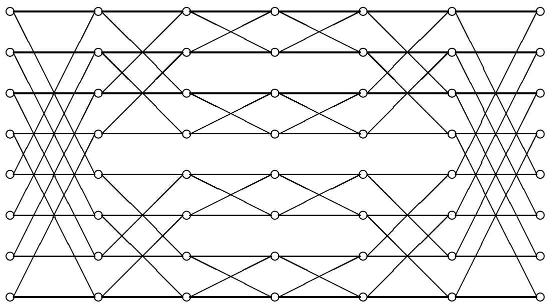

Fig. 3(b). (1, 2, 3, 3, 2, 1), root digraph of (1, 2, 3, 4, 3, 2, 1).

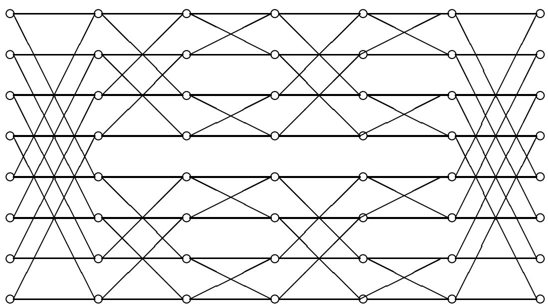

Fig. 3(c). (1, 2, 3, 2, 3, 1), root digraph of (1, 2, 3, 4, 2, 3, 1).

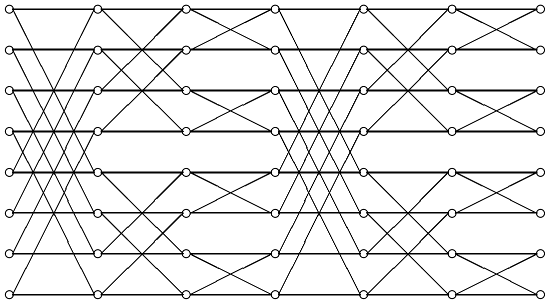

Fig. 3(d). (1, 2, 3, 1, 2, 3), root digraph of (1, 2, 3, 4, 1, 2, 3).

Now we consider the root digraph of BPNs not covered by Theorem 3.5. If edge stage E_i and E_{i+1} of G belong to the same subset, $G_{i,\,i+2}$ has exactly d^{n-2} connected components. Each component is the 3-stage BPN (1,1) (Fig. 4(a)). It is not difficult to see that BPN (1,1) has a unique root digraph, consisting of two nodes connected by d parallel edges. One node has d incoming edges and the other has d outgoing edges (Fig. 4(b)). This observation leads us to introduce a new class of MINs as follows.

(a) (b)

Fig. 4. $N = d = 3$. (a) BPN (1, 1); (b) root digraph of (1, 1).

Let G_1 and G_2 be two BPNs of the same size. The *bridged BPN* of G_1 and G_2, denoted by $G_1 : G_2$, is the MIN obtained from their union by adding d parallel edges from each output node of G_1 to the input node of G_2 with identical label. The edge stage containing parallel edges in $G_1 : G_2$ is called a *bridging edge stage*. We can define the bridged BPN of more than two BPNs similarly.

Fig. 3(a) shows the bridged BPN of (1, 2, 3) and (3, 2, 1). Note that (1, 2, 3):(1, 2, 3) and (1, 2, 3):(3, 2, 1) are not topologically equivalent, though (1, 2, 3) and (3, 2, 1) are the same network. It results from the fact that the two parts have been correlated to each other by the bridging edge stage.

As before, it is required that every bridged BPN be connected, though it may be obtained by bridging disconnected BPNs. If one of the BPNs has only one stage, its sequence notation is the empty vector, (). For example, both (1, 2, 3):():(2, 1) and ():(1, 2):(2, 3) are obtained by bridging three BPNs.

All BPNs can be considered as special bridged BPNs. In a BPN, if $E_i, E_{i+1}, \cdots, E_{i+t}$ are all in the same subset of the partition, we say $E_{i+1}, \cdots, E_{i+t}$ are *repeating edge stages*. By *removing* a bridging (or repeating) edge stage we mean deleting all its edges and identifying the pairs of nodes in the two stages orderly. *Adding* a bridging (or repeating) edge stage is the reverse operation.

Lemma 3.6. *Let* $G = (a_1, \cdots, a_{s_1-1}) : (b_1, \cdots, b_{s_2-1})$, $G' = (a_1, \cdots, a_{s_1-1}, b_1, \cdots, b_{s_2-1})$, *and* $L(G') = (c_1, c_2, \cdots, c_{s_1+s_2-1})$. *Then* $L(G) = (c_1, \cdots, c_{s_1}, c_{s_1}, c_{s_1+1}, \cdots, c_{s_1+s_2-1})$. *That is,* $L(G)$ *is obtained from* $L(G')$ *by adding a repeating edge stage after the* s_1*-th edge stage.*

Proof. Let $H = (c_1, \cdots, c_{s_1}, c_{s_1}, c_{s_1+1}, \cdots, c_{s_1+s_2-1})$ and $H' = L(G')$. We need to show that $L(G)$ is a BPN with the same partition as H. It is easy to check that $L(G)$ satisfies the $Q(*,*)$ property of Theorem 2.3. So it remains to show that $L(G)$ and H have the same numbers of components in corresponding partial networks.

Let c be the function of number of components in a digraph. Let i, j be integers such that $1 \leq i < j \leq s_1 + s_2 + 1$. Suppose u and v are two nodes of $L(G)$ in the i-th and j-th stage respectively. Then u and v are two edges of G (may be input or output edges). If $i \neq s_1+1$ and $j \neq s_1+1$, neither u nor v is an edge in the bridging edge stage of G. So they also represents two edges, u' and v', of G'. Obviously, v is reachable from u in G if and only if v' is reachable from u' in G'. Therefore, $c(L(G)_{i,j}) = c(H'_{i,j-1})$ for $i < s_1 + 1 < j$. If $j \leq s_1 + 1$, we only consider the left part of G and delete all the stages on the right of the bridging edge stage (but leave the edges in the bridging edge stage pendant), because $c(L(G)_{i,j})$ is determined only by this part. However, after deleting from G' the stages $s_1 + 1$ through $s_1 + s_2 - 1$ (leaving the last edge stage pendant), we get an equivalent network. This implies $c(L(G)_{i,j}) = c(H'_{i,j})$ for $j \leq s_1 + 1$. Similarly, $c(L(G)_{i,j}) = c(H'_{i-1,j-1})$ for $i \geq s_1 + 1$. So $c(L(G)_{i,j}) = c(H_{i,j})$, $1 \leq i < j \leq s_1 + s_2 + 1$. ∎

Lemma 3.6 can be extended to bridged BPNs with any number of bridging edge stages. Therefore, we can derive the line digraph of a bridged BPN by first removing its bridging edge stages, applying Procedure LINEGRAPH, and then adding repeating edge stages in appropriate positions. We can also revise Procedure LINEGRAPH so that it computes the line digraph of any bridged BPN directly.

Procedure LINEGRAPH–R
Input: $A_1 \cup A_2 \cup \cdots \cup A_n \cup A_{n+1}$, where A_{n+1} contains bridging edge stages and A_k ($1 \leq k \leq n$) is an ordered set in which E_i precedes E_j if and only if $i < j$.
Output: $\pi_{L(G)} = B_1 \cup B_2 \cup \cdots \cup B_{n+1}$.

begin
  Set $B_k = \emptyset$ $(1 \leq k \leq n+1)$;
  Put F_1 into B_1;
  FOR $i = 2$ TO s
    IF $E_{i-1} \in A_{n+1}$, put F_i into the same set as F_{i-1} belongs to
    ELSE IF E_{i-1} is the first element in $[E_{i-1}]$, put F_i into the first empty set among $B_1, \cdots, B_{n+1}$.
      ELSE IF E_k is the predecessor of E_{i-1} in $[E_{i-1}]$, put F_i into the same set as F_k belongs to.
end

From the discussion above, it is seen that there is always a root digraph in the bridged BPN class for any BPN. We can find it by first removing its repeating edge stages, applying Procedure ROOTGRAPH, and then adding bridging edge stages in appropriate positions. The following is a revision of Procedure ROOTGRAPH which computes the root digraph of any BPN directly.

Procedure ROOTGRAPH–R
Input: $\pi_G = A_1 \cup A_2 \cup \cdots \cup A_n$, where each A_k $(1 \leq k \leq n)$ is an ordered set in which E_i precedes E_j if and only if $i < j$.
Output: $B_1 \cup B_2 \cup \cdots \cup B_{n-1} \cup B_n$, where B_n contains the bridging edge stages.
begin
  Set $B_k = \emptyset$ $(1 \leq k \leq n)$;
  FOR $i = s-1$ DOWNTO 1
    IF E_{i+1} is the successor of E_i in $[E_i]$, put F_i into B_n
    ELSE IF E_i is the last element in $[E_i]$, put F_i into the first empty set among $B_1, \cdots, B_{n-1}$.
      ELSE IF E_k is the successor of E_i in $[E_i]$, put F_i into the same set as F_{k-1} belongs to.
end

The main theorem of this section is derived directly from all the results above. That is the following

Theorem 3.7. *A MIN is a BPN if and only if it is the line digraph of a bridged BPN.*

As the final example for this section, removing the repeating edge stages of (1, 2, 2, 3, 1, 4, 3, 3, 3, 4, 4, 1, 1, 2) gives (1, 2, 3, 1, 4, 3, 4, 1, 2), whose root digraph, by ROOTGRAPH, is (1, 2, 1, 3, 1, 1, 3, 2). After adding bridging edge stages, we have (1):(2, 1, 3, 1):():(1):(3):(2), which is the same as the result of ROOTGRAPH–R.

4. CF-Rearrangeable BPNs

In this section, we apply the results of Section 3 to derive some results about CF-rearrangeable BPNs.

Lemma 4.2. ([10]) *The conflict-free routing of a d-permutation on a MIN is equivalent to the crosstalk-free routing of a specific 1-permutation on its line digraph. A MIN is rearrangeable if and only if its line digraph is CF-rearrangeable.*

Lemma 4.3. *In conflict-free routing, a bridged BPN G is functionally equivalent to the BPN obtained from G by removing its bridging edge stages.*

The next theorem shows that the crosstalk-free routing problem on BPNs can be solved by the older theory in electronic switching.

Lemma 4.4. *The crosstalk-free routing on any BPN is equivalent to the conflict-free routing on a BPN of smaller size and with fewer stages.*

Proof. By Theorem 3.7, any BPN G has a root digraph in the bridged BPN class. The bridged BPN can be transformed into a BPN H by removing its bridging edge stages without damaging its ability of realizing permutations. Then the crosstalk-free routing on G is equivalent to the conflict-free routing on H. ■

Theorem 4.4. *A BPN G is CF-rearrangeable if and only if the BPN obtained from G by removing its repeating edge stages is CF-rearrangeable.*

Proof. The CF-rearrangeability of G is equivalent to the rearrangeability of the bridged BPN H obtained by ROOTGRAPH–R, which is functionally equivalent to the rearrangeability of H' obtained by removing bridging edge stages. In return, the rearrangeability of H' is equivalent to the CF-rearrangeability of $G' = L(H')$, which can be derived directly from G by removing repeating edge stages. ■

It can be verified that $L(1, \cdots, n-1, n-1, \cdots, 1) = (1, \cdots, n-1, n, n-1, \cdots, 1)$ (this observation was also made in [10]). So the dilated Benes network is actually the line digraph of Benes network. Since Benes network is rearrangeable, the dilated Benes network is CF-rearrangeable. Padmanabhan and Netravali [13] introduced the dilated Benes network and gave an adapted Looping Algorithm for crosstalk-free routing. It is also seen from Theorem 4.4 that Benes network is itself a CF-rearrangeable network, because it can be derived from the dilated Benes network of same size by adding a repeating edge stage in the center part.

In the following we will consider "minimum" CF-rearrangeable networks. It is widely known that $2n+1$ is a lower bound on the number of stages for a $N \times N$

shuffle-exchange network to be rearrangeable. But whether this lower bound is tight or not is an unsolved problem. In [2], Bao, Hwang and Li showed that $2n+1$ is also a lower bound for a BPN to be rearrangeable. So Benes network is an example of rearrangeable networks with the minimum number of stages in the BPN class.

Lemma 4.5. ([2]) *The minimum number of stages for an $N \times N$ BPN to be rearrangeable is $2n+1$. Furthermore, if G is a rearrangeable BPN with $2n+1$ stages, then both of the two subnetworks, $G_{1,n+1}$ and $G_{n+1,2n+1}$, are topologically equivalent to the Omega network.*

Lemma 4.6. *The minimum number of stages for an $N \times N$ BPN to be CF-rearrangeable is $2n$. Furthermore, if G is a CF-rearrangeable BPN with $2n$ stages, then both of the two subnetworks, $G_{1,n+1}$ and $G_{n,2n}$, are topologically equivalent to the Omega network.*

Proof. Suppose G is CF-rearrangeable. Since a CF-rearrangeable BPN with minimum number of stages has no repeating edge stages, by Theorem 3.5 and Lemma 4.1, the root digraph H of G from ROOTGRAPH is a rearrangeable BPN. As the size of H is $d^{n-1} \times d^{n-1}$, H has at least $2n-1$ stages. So G has at least $2n$ stages. If G has exactly $2n$ stages, then $G_{1,n+1}$ and $G_{n,2n}$ are the line digraphs of $H_{1,n}$ and $H_{n,2n-1}$ respectively. By Lemma 4.5, both $H_{1,n}$ and $H_{n,2n-1}$ are Omega-equivalent. Therefore, both $G_{1,n+1}$ and $G_{n,2n}$ are Omega-equivalent. ■

We have seen that the dilated Benes network is a CF-rearrangeable BPN with minimum number of stages. In [5], it was shown that the $(2n-1)$-stage BPN obtained from the $(2n-1)$-stage Benes network by reversing the order of the n-th and $(n+1)$-th edge stages is rearrangeable. In [2], it was also shown that the BPN obtained by reversing the order of the n-th, $(n+1)$-th and $(n+2)$-th edge stages is rearrangeable. So, the BPN obtained from the $2n$-stage dilated Benes network by reversing its $(n+1)$-th and $(n+2)$-th [$(n+1)$-th, $(n+2)$-th and $(n+3)$-th, resp.] edge stages is also CF-rearrangeable.

Theorem 4.6. $(1,\cdots,n,n-2,n-1,n-3,\cdots,1)$ *and* $(1,\cdots,n,n-3,n-2,n-1,n-4,\cdots,1)$ *are two CF-rearrangeable BPNs with minimum number of stages.*

For $d=2$, Fig. 3(c) and (d) show the two 8×8 rearrangeable networks derive from 8×8 binary Benes network by reversing the order of edge stages. Fig. 1(c) and (d) show the two corresponding 16×16 CF-rearrangeable networks.

Rearrangeable and CF-rearrangeable networks can also realize permutations of larger size if multiple passes are permitted. In general, we have

Theorem 4.7. *An $N \times N$ rearrangeable network can realize any dt-permutation in t passes. An $N \times N$ CF-rearrangeable network can realize any t-permutation crosstalk-free in t passes.*

To prove this theorem, we only need a classical theorem in graph theory–König's Theorem [3]. In the following, we state it using the terms of this paper.

König's Theorem *Any t-permutation can be factorized into t 1-permutations.*

It is worth noting that efficient algorithms already exist in the literature to factorize a t-permutation into 1-permutations, usually in the context of perfect matching in bipartite graphs. When a t-permutation is to be routed crosstalk-free on a CF-rearrangeable network, we first factorize the t-permutation into t 1-permutations, and then route each 1-permutation in one pass. If a dt-permutation is to be realized conflict-free on a rearrangeable network, we just combine any d 1-permutations into one d-permutation, and route it in one pass.

Traditionally, realizing a permutation in multiple passes is termed as dilation in time, and realizing a permutation on a vertically stacked network as dilation in space. When only (CF-)rearrangeable networks are considered, there is no difference in the routing technic for both dilations. In [15], Yang et al. proved that any 2-permutation can be routed crosstalk-free on the binary Benes network in two passes. Jiang et al. [11] proposed to vertically stack Benes network for crosstalk-free routing. These results can be viewed as special cases of Theorem 4.8. In their cases, one stage of nodes can be saved if the dilated Benes network takes the place of Benes network.

5. Conclusions

The main contribution of this work is a line-digraph characterization of bit permutation networks. Using the concept of line digraphs from graph theory, we bridged the gap between the problem of conflict-free routing and the problem of crosstalk-free routing on bit permutation networks, and proved that crosstalk-free routing on a bit permutation network is always equivalent to the conflict-free routing on a bit permutation network of smaller size and with fewer stages. Based on these results, we gave some necessary conditions for a BPN to be CF-rearrangeable. We also showed the CF-rearrangeability and minimality (in stage number) of three families of BPNs, including the dilated Benes network. The problem of CF-rearrangeable networks with dilation in time or space was also touched upon.

References

1. D. P. Agrawal, Graph theoretical analysis and design of multistage interconnection networks, *IEEE Trans. Comput.*, **32** (1983) 637-648.

2. X. Bao, F. K. Hwang and Q. Li, Rearrangeability of bit permutation networks, *Theoretical Computer Science*, **352** (2006) 197-214.

3. C. Berge, *Graphs and Hypergraphs.*, North Holland, Amsterdam, 1976.

4. J. C. Bermond, J. M. Fourneau and A. Jean-Marie, Equivalence of multistage interconnection networks, *Info. Proc. Lett.*, **26** (1987) 45-50.

5. T. Calamoneri and A. Massini, A new approach to the rearrangeability of ($2logN$-1) stage MINs, *Proc. IASTED Int'l Symp. Applied Informatics (AI 2001)*, 2001, 365-370.

6. G. J. Chang, F. K. Hwang and L.-D. Tong, Characterizing bit permutation networks, *Networks*, **33** (1999), 261-267.

7. F. Harary and R. Z. Norman, Some properties of line digraphs, *Rend. Circ. Mat. Palermo*, **9** (1960), 161-168.

8. C. Heuchenne, Sur une certaine correspondence entre graphes, *Bull. Soc. Roy. Sci. Liège*, **33** (1964), 743-753.

9. F. K. Hwang and C.-H. Yen, Characterizing the bit permutation networks obtained from the line digraphs of bit permutation networks, *Networks*, **38** (2001), 1-5.

10. F. K. Hwang and W.-D. Lin, A general construction for nonblocking crosstalk-free photonic switching networks, *Networks*, **42** (2003), 20-25.

11. X. Jiang, H. Shen, Md. Mamun-ur-Rashid Khandker and Susumu Horiguchi, Vertically stacked Benes networks for crosstalk-free permutation, *Proceedings of the First International Symposium on Cyber Worlds (CW'02)*, 2002.

12. C.-T. Lea, Bipartite graph design principle for photonic switching systems, *IEEE Trans. Commun.*, **38** (1990), 529-538.

13. K. Padmanabhan and A. N. Netravali, Dilated networks for photonic switching, *IEEE Trans. Commun.*, **35** (1987), 1357-1365.

14. Y. Wu, X. Bao, X. Jia and Q. Li, Graph theoretical characterizations of bit permutation networks, manuscript.

15. Y. Yang, J. Wang and Y. Pan, Permutation capability of optical multistage interconnection networks, *J. Parallel and Distrib. Comput.*, **60** (2000), 72-91.

Xuewen Bao
Center for Discrete Mathematics,
Software College, Fuzhou University,
Fuzhou 350003,
P. R. China
E-mail: baoxw@163.com

Qiao Li
Department of Mathematics,
Shanghai Jiaotong University,
Shanghai 200030,
P. R. China