Equivalence of Buddy Networks with Arbitrary Number of Stages*

Chiuyuan Chen, Frank K. Hwang, James K. Lan
Department of Applied Mathematics, National Chiao Tung University, Hsinchu 300, Taiwan

Equivalence of multistage interconnection networks is an important concept because it reduces the number of networks to be studied. Equivalence among the banyan networks has been well studied. Occasionally, the study was extended to networks obtained by concatenating two banyan networks (identifying the output stage of the preceding network with the input stage of the succeeding one). Recently, equivalence among the class of networks that are obtained from banyan networks by adding extra stages has also been studied. Note that all these above-mentioned networks are in the general class of buddy networks. In this article we study equivalence of buddy networks with an arbitrary number of stages. © 2005 Wiley Periodicals, Inc. NETWORKS, Vol. 46(4), 171–176 2005

Keywords: multistage interconnection networks; topological equivalence; banyan property; buddy property; bit permutation

1. INTRODUCTION

Let $N = d^n$ be the number of inputs and outputs of a network. A *d-nary s-stage network* is a network with s columns (stages) where each column consists of N/d $d \times d$ crossbars (switches) such that links exist only between crossbars of adjacent stages (note that we do not allow multilinks between crossbars). An n-stage network is a *banyan network* if each input has a unique path to each output (see Fig. 1). If a network has more than n stages, then we say such a network has *extra stages*. In all the figures, the arcs are directed from left to right.

We can associate an s-stage network with a directed graph G in which vertices represent crossbars and arcs the communication links. Throughout this article, $G_{i,j}$ denotes the subgraph of G induced by the vertices from stage i to stage j. When there is no confusion, $G_{i,j}$ also denotes the subnetwork from stage i to stage j. Set $G_i = G_{i,i+1}$ for easy writing (see Fig. 1).

Two s-stage networks are *topologically equivalent* (or simply *equivalent*) if their associated directed graphs are isomorphic. In other words, two s-stage networks are equivalent if one can be obtained from the other by permuting crossbars in the same stage. Note that equivalence in this sense preserves the connecting properties of the network. Hence, once we prove a nonblocking property for a network, it extends to all equivalent networks.

Parker [10] first established the equivalence of several n-stage banyan networks including the Baseline network. Wu and Feng [13] expanded the equivalence class. Dias and Jump [6] introduced the "buddy" notation: Let v and v' be two crossbars in stage i and let V_v and $V_{v'}$ be the two sets of crossbars in stage j that v and v' can reach, respectively. Then the network is a *buddy network* if for any i and $j = i + 1$, either $V_v = V_{v'}$ or $V_v \cap V_{v'} = \emptyset$. Agrawal [1] called a buddy network a *strict buddy network* if the buddy condition also holds for $j = i + 2$. In this article, we further generalize the strict buddy network to the *universal buddy network* by allowing j to be arbitrary. In [1], Agrawal claimed that the strict buddy property characterizes the Baseline-equivalent networks. Bermond et al. [2, 3] gave a counterexample to Agrawal's claim. Instead, they defined the $P(*, *)$ property for characterization: A network is a $P(*, *)$ *network* if for any two stages $i \leq j$, the number of components in the subgraph $G_{i,j}$ is $d^{n-1-(j-i)}$.

Siegel and Smith [12] proposed an extra stage to the Baseline-equivalent class of networks, while Shyy and Lea [11] considered the k-extra-stage version. Hwang et al. (see [8]) pointed out that the extra stage versions of Baseline-equivalent networks are not necessarily equivalent. Equivalence depends not only on the base network (Baseline or others), but also on how the extra stages are added. Previously, equivalence of extra-stage networks has been studied only for the double-concatenation type [4, 7] because it contains the famous Beneš network as a special case.

To study the equivalence of extra-stage networks for arbitrary number of stages, Chang et al. [5] proposed the class of bit permutation networks. Label the crossbars in a stage by

Received June 2003; accepted August 2005
Correspondence to: C. Chen; e-mail: cychen@mail.nctu.edu.tw
*This article was written when the author was visiting the Center of Mathematical Sciences, Zhejiang University, Hangzhou, Zhejiang, P.R. China.
Contract grant sponsor: The National Science Council of the Republic of China; Contract grant numbers: NSC93-2115-M-009-011 and NSC93-2115-M-009-013
DOI 10.1002/net.20085
Published online in Wiley InterScience (www.interscience.wiley.com).

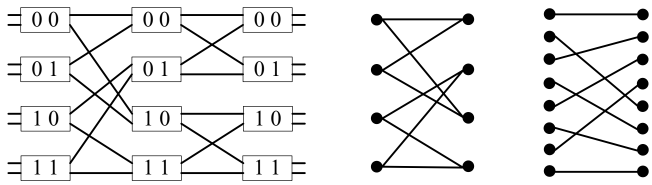

FIG. 1. A binary three-stage banyan network (the Baseline network), G_1, and $\mathcal{G}_1$.

distinct d-nary $(n-1)$-sequences $x_1x_2\ldots x_{n-1}$. A *bit-i group* (or simply an *i-group*) consists of the d crossbars whose labels differ only in bit i (there are d^{n-2} bit-i groups). An s-stage network is a *bit permutation network* if for every $G_i, 1 \le i \le s-1$, the links always go from bit-u_i groups G' of stage i to bit-v_{i+1} groups G'' of stage $i+1$ for some u_i, v_{i+1}, where G'' is a permutation of G'. (A detailed definition of bit permutation networks is given in Section 2.) They proved that a bit permutation network is equivalent to one whose G_i has the property that $v_{i+1} = u_i$ for all i. Such a network can be characterized by the vector $(u_1, u_2, \ldots, u_{s-1})$.

Recently, Li [9] proposed the bit permuting network. He views the outputs of stage i and the inputs of stage $i+1$ as the vertices of a bipartite graph $\mathcal{G}_i$ and labels the outputs of stage i (inputs of stage $i+1$) by distinct d-nary n-sequences; see Figure 1. Then $\mathcal{G}_i$ gives a bijection from the d^n outputs to the d^n inputs, and hence can be treated as a permutation. Such a permutation is called an *bit permutation* if it can be characterized by a permutation σ_i of the n bits. A network is a *bit permuting network* if each $\mathcal{G}_i$ corresponds to a σ_i. Li gave an elegant "guide" algorithm to route any n-stage bit permuting network.

The notions of universal buddy (*UB*), bit permutation (*BP*), and bit permuting (*BPT*) are applicable to networks with any number of stages. Because $P(*,*)$ is defined only for n-stage networks, we generalize it to the power-of-d networks. An s-stage network is a *power-of-d network* if for any $i,j, 1 \le i \le j \le s$, the number of components in $G_{i,j}$ is a power of d. An s-stage network is a *power-of-d universal buddy network* if it is both power-of-d and universal buddy. The notion of power-of-d $\left(d^P\right)$ and power-of-d universal buddy $\left(d^P UB\right)$ are applicable to networks with any number of stages. In this article, the notations of UB, BP, BPT, d^P, and $d^P UB$ also denote their corresponding classes of networks.

Let $A \supset B$ denote that A properly contains B. Let $A = B$ denote that *A is equal to B*, meaning any network in class A is a network in class B (no permutation of crossbars allowed), and vice versa. Let $A \sim B$ denote that *A is equivalent to B*, meaning any network in class A is topologically equivalent to a network in class B (permutations of crossbars allowed), and vice versa. Note that the permutation of crossbars is neither unique nor one-to-one. Hence $A \sim B$ does not imply $|A| = |B|$. In particular, $A \supset B$ does not preclude $A \sim B$. In this article, we will establish:

$$\begin{matrix} UB \supset \\ d^P \supset \end{matrix}\; d^P UB \supset BP = BPT. \qquad (1.1)$$

$$d^P UB \sim BP, \text{ but } UB \not\sim d^P, UB \not\sim d^P UB, \text{and } d^P \not\sim d^P UB. \qquad (1.2)$$

Because the *BP* network has the vector characterization and is defined for any number of stages, it is of interest to know whether this very useful class can be further extended with all connecting properties preserved. Relation (1.1) shows that $d^P UB$ generalizes *BP* and (1.2) shows that they are equivalent.

2. THE *BP* AND *BPT* CLASSES

We now give a detailed definition of *BP* networks; this definition is from [5]. An s-stage network is a *bit permutation network* if for every $G_i, 1 \le i \le s-1$, there exists a permutation ρ_i on $\{1, 2, \ldots, n\}$ such that $\rho_i(n) \ne n$ and each crossbar $x_1x_2\ldots x_{n-1}$ is adjacent to crossbar $x_{\rho_i(1)}x_{\rho_i(2)}\ldots x_{\rho_i(n-1)}$, where $x_n \in \{0, 1, \ldots, d-1\}$. Note that x_n has d values, and whenever it appears in the coordinates, d sequences are generated by running x_n through the set $\{0, 1, \ldots, d-1\}$. For example, the network in Figure 1 is a bit permutation network with $\rho_1 = (132)$ and $\rho_2 = (23)$. Because $\rho_1 = (132), x_1x_2x_3$ is mapped to $x_3x_1x_2$ and the links go from bit-2 groups of stage 1 to bit-1 groups of stage 2. In particular, crossbars 00 and 01 at stage 1 are adjacent to crossbars 00 and 10 at stage 2. Because $\rho_2 = (23), x_1x_2x_3$ is mapped to $x_1x_3x_2$, and the links go from bit-2 groups of stage 2 to bit-2 groups of stage 3. Thus, crossbars 00 and 01 at stage 2 are adjacent to crossbars 00 and 01 at stage 3.

The stages in Figure 2 and Figure 3 are drawn horizontally to save space. These two figures are the same (they have the same connections between crossbars) except their labels. The labels in Figure 2 are outputs of stage i and inputs of stage $i+1$. The labels in Figure 3 are crossbars of stage i and crossbars of stage $i+1$. The permutation in Figure 2 illustrates a bit permutation $\sigma_i = (1234)$ in $\mathcal{G}_i$, while the permutation in Figure 3 illustrates a permutation $\rho_i = (1234)$ in G_i.

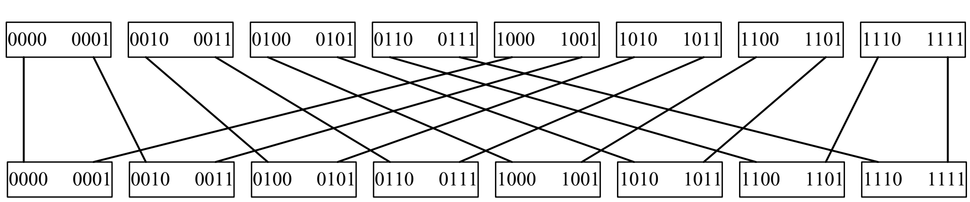

FIG. 2. A bit permutation σ_i in $\mathcal{G}_i$.

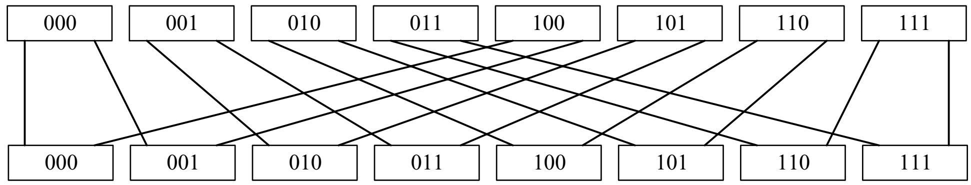

FIG. 3. A permutation ρ_i of crossbars in G_i.

We now prove

Theorem 1. *BPT* = *BP*.

Proof. First consider a *BPT* network. For every $\mathcal{G}_i$, there exists a bit permutation σ_i on $\{1, 2, \ldots, n\}$ such that each output $x_1x_2 \ldots x_n$ of stage i is adjacent to input $x_{\sigma_i(1)}x_{\sigma_i(2)} \ldots x_{\sigma_i(n)}$ of stage $i + 1$. Note that the label of a crossbar of stage i $(i + 1)$ can be obtained from the labels of its d outputs (inputs) by dropping the last bit. Thus, crossbar $x_1x_2 \ldots x_{n-1}$ is adjacent to crossbar $x_{\sigma_i(1)}x_{\sigma_i(2)} \ldots x_{\sigma_i(n-1)}$. Note that $\sigma_i(n) \neq n$; otherwise, there are multilinks between crossbar $x_1x_2 \ldots x_{n-1}$ and crossbar $x_{\sigma_i(1)}x_{\sigma_i(2)} \ldots x_{\sigma_i(n-1)}$. Because $\sigma_i(n) \neq n$, crossbar $x_1x_2 \ldots x_{n-1}$ is adjacent to crossbar $x_{\sigma_i(1)}x_{\sigma_i(2)} \ldots x_{\sigma_i(n-1)}$, where $x_n \in \{0, 1, \ldots, d - 1\}$. Thus, a *BPT* network is a *BP* network. On the other hand, consider a *BP* network. For every G_i, there exists a permutation ρ_i on $\{1, 2, \ldots, n\}$ such that $\rho_i(n) \neq n$ and each crossbar $x_1x_2 \ldots x_{n-1}$ of stage i is adjacent to crossbar $x_{\rho_i(1)}x_{\rho_i(2)} \ldots x_{\rho_i(n-1)}$ of stage $i+1$, where $x_n \in \{0, 1, \ldots, d-1\}$. Thus, each output $x_1x_2 \ldots x_n$ of stage i is adjacent to input $x_{\rho_i(1)}x_{\rho_i(2)} \ldots x_{\rho_i(n)}$ of stage $i + 1$. Because a permutation on $\{1, 2, \ldots, n\}$ is a bit permutation, a *BP* network is a *BPT* network. Theorem 1 now follows. ■

We now show that a bit permutation σ_i of $\mathcal{G}_i$ defines a mapping from u-groups of stage i to v-groups of stage $i + 1$. In fact, we can pinpoint u and v.

Lemma 2. *Suppose $\mathcal{G}_i$ is represented by the bit permutation σ_i. Then $\mathcal{G}_i$ induces a mapping from $\sigma_i(n)$-groups of stage i to $\sigma_i^{-1}(n)$-groups of stage $i + 1$.*

Proof. Note that each output $x_1x_2 \ldots x_n$ of stage i is adjacent to input $x_{\sigma_i(1)}x_{\sigma_i(2)} \ldots x_{\sigma_i(n)}$ of stage $i + 1$. The label of a crossbar of stage i $(i + 1)$ can be obtained from the labels of its d outputs (inputs) by dropping the last bit. Because $x_{\sigma_i(n)}$ is the last bit and gets dropped in the crossbar label of stage $i+1$, the d stage-i crossbars differing only in bit $\sigma_i(n)$, that is, the $\sigma_i(n)$-group, are mapped to the same set of stage-$(i + 1)$ crossbars. On the other hand, the stage-i crossbar containing d outputs whose labels differ only in bit $\sigma_i(n)$ is mapped to the $\sigma_i^{-1}(n)$-group of stage $i + 1$. Lemma 2 is proved. ■

For the example in Figure 2, the mapping is from $(\sigma_i(4) = 1)$-groups of stage i to $\left(\sigma_i^{-1}(4) = 3\right)$-groups of stage $i + 1$. We now give a vector characterization of a *BPT* network. First a lemma.

Lemma 3. *Suppose $\mathcal{G}_i$ corresponds to a bit permutation σ_i which maps $\sigma_i(n)$-groups of stage i to $\sigma_i^{-1}(n)$-groups of stage $i+1$ and suppose $\mathcal{G}_{i+1}$ corresponds to a bit permutation σ_{i+1}. Suppose we permute the crossbars of stage $i + 1$ such that the j-th crossbars of the $\sigma_i^{-1}(n)$-groups are lined up with the j-th crossbars of the $\sigma_i(n)$-groups, $j = 0, 1, \ldots, d - 1$. Then after the lining-up operation, $\mathcal{G}_i$ corresponds to the bit permutation $(u_i\ n)$ and $\mathcal{G}_{i+1}$ corresponds to the bit permutation $\left(\sigma_{i+1}^{-1}\left(\sigma_i^{-1}(n)\right)\ \sigma_{i+1}^{-1}(\sigma_i(n))\right) \circ \sigma_{i+1}$.*

Proof. Take a $\sigma_i^{-1}(n)$-group of stage $i + 1$. The j-th crossbar in this group is mapped (lined up) to the j-th crossbar of the corresponding $\sigma_i(n)$-group of stage i under this lining-up operation; see Figure 4. Then the only difference is that before lining up, the bit permutation σ_i maps $\sigma_i(n)$-groups to $\sigma_i^{-1}(n)$-groups, while after lining up, the mapping is from u_i-groups to u_i-groups. Note that the mapping from u_i-groups to u_i-groups corresponds to the bit permutation $(u_i\ n)$. After lining up, $\sigma_i^{-1}(n)$-groups of stage $i + 1$ become $\sigma_i(n)$-groups. Because σ_{i+1} maps $\sigma_{i+1}^{-1}\left(\sigma_i^{-1}(n)\right)$ to $\sigma_i^{-1}(n)$ and $\sigma_{i+1}^{-1}(\sigma_i(n))$ to $\sigma_i(n)$, swapping bit $\sigma_i^{-1}(n)$ with bit $\sigma_i(n)$ corresponds to applying $\left(\sigma_{i+1}^{-1}\left(\sigma_i^{-1}(n)\right)\ \sigma_{i+1}^{-1}(\sigma_i(n))\right)$ on σ_{i+1}. Thus, after lining up, $\mathcal{G}_{i+1}$ corresponds to the bit permutation $\left(\sigma_{i+1}^{-1}\left(\sigma_i^{-1}(n)\right)\ \sigma_{i+1}^{-1}(\sigma_i(n))\right) \circ \sigma_{i+1}$. ■

By Lemma 2, we know that in every $\mathcal{G}_i$ of a *BPT* network, the links go from u_i-groups to v_{i+1}-groups for some u_i, v_{i+1}. The lining-up operation enables us to permute the crossbars of stage $i+1$ so that the links go from u_i-groups to u_i-groups. For example, in Figure 4, the links go from 2-groups to 1-groups. After lining up the stage-$(i + 1)$ crossbars, the links go from 2-groups to 2-groups.

Theorem 4. *Consider an s-stage BPT network. By permuting the crossbars of stage $2, 3, \ldots, s$, each $\mathcal{G}_i$ corresponds*

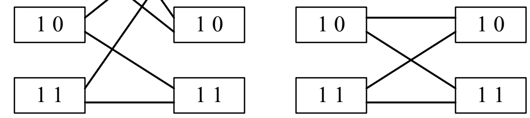

FIG. 4. Lining up stage-$(i + 1)$ crossbars.

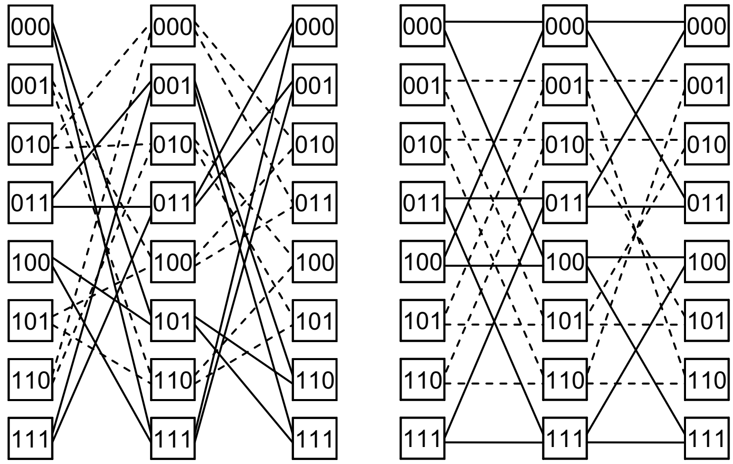

FIG. 5. (a) Before lining up and (b) after lining up.

to a bit permutation that maps u_i'-groups to u_i'-groups, $i = 1, 2, \ldots, s-1$.

Proof. We prove this theorem by induction on s. This theorem is trivially true for $s = 2$ because we can permute the crossbars of stage 2 to line up with their mates in stage 1. Then $\mathcal{G}_1$ corresponds to a bit permutation which maps u_1-groups to u_1-groups. Suppose this theorem holds for up to $s-1$ stages. We now prove the result for s stages. Again, permute the crossbars of stage 2 to line up with their mates in stage 1. By Lemma 3, $\mathcal{G}_2$ corresponds to a bit permutation. Thus, we may apply induction on this $(s-1)$-stage *BPT* network such that $\mathcal{G}_i$ is characterized by a bit permutation that maps u_i'-groups to u_i'-groups, $i = 1, 2, \ldots, s-1$. ∎

Because $BPT = BP$, the above characterization is also a vector characterization of a *BP* network, but our proof is simpler than the original proof in [5]. Recall that an s-stage network is a *BP* network if for every G_i, the links always go from u_i groups G' of stage i to v_{i+1} groups G'' of stage $i+1$ for some u_i, v_{i+1}, where G'' is a permutation of G'. If we drop the requirement that G'' is a permutation of G', then the lining-up operation would not yield a vector characterization. See Figure 5 as an example. In Figure 5(a), the links in G_1 go from 1-groups to 2-groups and the links in G_2 go from 1-groups to 3-groups. In Figure 5(b), the links in G_1 go from 1-groups to 1-groups, but the links in G_2 do not go from u-groups to u-groups for any u.

3. THE $d^P UB$ CLASS

We now show that neither $d^P \subseteq UB$ nor vice versa; hence, the definition of $d^P UB$ makes sense. Figure 6(a) shows a 2^P network, which is not a *UB* network because C reaches $\{C', D', F', G'\}$ and E reaches $\{C', E', F', H'\}$; the two sets intersect but are not identical. Figure 6(b) shows a *UB* network, which is not a 2^P network because $G_{1,3}$ has three components.

We first quote a result of [5].

Theorem 5. *Suppose an s-stage d-nary BP network has d^n inputs, d^n outputs, and is characterized by the vector*

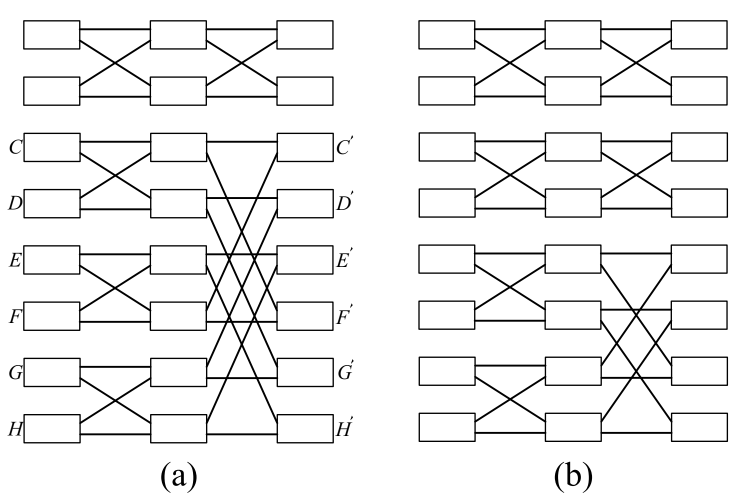

FIG. 6. (a) A 2^P network and (b) a *UB* network.

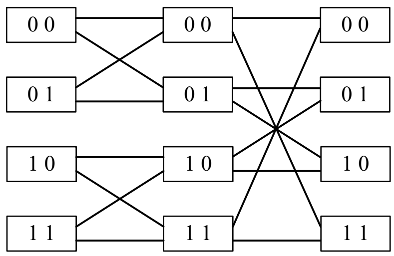

FIG. 7. A $d^P UB$ network that is not a BP network.

$(u_1, u_2, \ldots, u_{s-1})$*, which contains k distinct elements. Then the network has* d^{n-1-k} *components.*

Corollary 6. $BP \subseteq d^P$.

Proof. It is not difficult to see that every subnetwork $G_{i,j}$ of a BP network is still a BP network. By Theorem 5, the number of components in $G_{i,j}$ is a power of d. Because i, j are arbitrary, the network is in d^P. ∎

Theorem 7. $BP \subseteq UB$.

Proof. Consider an s-stage BP network characterized by $(u_1, u_2, \ldots, u_{s-1})$. Let v be a crossbar in stage i, which reaches a set $V_j(v)$ of crossbars in stage j. Then $V_j(v)$ consists of crossbars whose labels are the same in bits in the set $I = \{1, 2, \ldots, n-1\} \setminus \{u_i, u_{i+1}, \ldots, u_{j-1}\}$. Let v' be another crossbar in stage i. If v' differs from v in a bit in I, then clearly, $V_j(v') \cap V_j(v) = \emptyset$; if not, then $V_j(v') = V_j(v)$. Because i, j, v, v' are arbitrary, the network is in UB. ∎

Theorem 8. $BP \subset d^P UB$.

Proof. That $BP \subseteq d^P UB$ follows from Corollary 6 and Theorem 7. That the containment is strict follows from Figure 7 (crossbars 00 and 11 in stage 2 are connected to crossbars 00 and 11 in stage 3; so, the links from stage 2 to stage 3 do not go from u_i-groups to v_{i+1}-groups for any u_i, v_{i+1}). ∎

Theorem 9. $d^P UB \sim BP$.

Proof. Because $d^P UB \supset BPT$, it suffices to prove that a $d^P UB$ network is equivalent to a BP network. We prove this by induction on the number s of stages.

(1) $s = 2$. Suppose v of stage 1 is connected to the set $V_2(v)$. Let v' be another crossbar in stage 1 connected to a given $w \in V_2(v)$. By the UB property, $V_2(v') = V_2(v)$. Because there are $d-1$ choices of v' from w, these v' together with v form a $d \times d$ complete bipartite graph $K_{d,d}$ with $V_2(v)$. Further, $V_2(v'') \cap V_2(v) = \emptyset$ for any $v'' \neq v, v'$ (note that there are $d-1$ choices of v'). Because v is arbitrary, $G_{1,2}$ consists of d^{n-2} $K_{d,d}$ whose equivalence to a BP network is clear.

(2) $s = 3$. By the d^P property, the network has d^{n-k} components for some $1 \leq k \leq n$. Recall that from (1) the subgraphs $G_{1,2}$ and $G_{2,3}$ must each consist of d^{n-2} $K_{d,d}$. Hence, $k = 1$ is impossible.
For $k = 2$, then no two $K_{d,d}$ in $G_{1,2}$ can be connected through $G_{2,3}$. Therefore, $G_{1,3}$ must consist of d^{n-2} copies of the concatenation of two $K_{d,d}$, with the outputs of the former identified with the inputs of the latter (see Fig. 8). Clearly, subnetwork $G_{1,3}$ is equivalent to a BP network.
For $k = 3$, first suppose $G_{1,3}$ is obtained by connecting each d-set $D = \{D_1, D_2, \ldots, D_d\}$, where each D_i is a $K_{d,d}$ in $G_{1,2}$, into one component in $G_{1,3}$. Note that the connection is done by a d-set $D' = \{D'_1, D'_2, \ldots, D'_d\}$ of $K_{d,d}$ in $G_{2,3}$. If two crossbars of the same D_i are connected to a D'_j, then one member of $D \backslash D_i$ will not be connected to D'_j, violating the UB property. Therefore, the d crossbars in a D_i must go to distinct D'_j, or all D'_j. Because we can permute the stage-2 crossbars in a D arbitrarily, and independently for each D, the stage-2 crossbars in each D can be ordered such that the k-th one goes to the k-th D', which is clearly a BP network. Figure 9 illustrates how to permute.
Suppose $G_{1,3}$ is obtained otherwise. There must exist a d'-set of $K_{d,d}$, $d' > d$, in $G_{1,2}$ connected in $G_{2,3}$ through a d'-set of $K_{d,d}$ in $G_{2,3}$. Note that an input in this component touches only d^2 among the dd' outputs. Hence, there must exist another input reaching some, but not all, of these d^2 outputs, violating the UB property.
For $k \geq 4$, then the situation described in the last paragraph must also happen.

(3) $s \geq 4$. Consider the two subnetworks $G_{1,3}$ and $G_{2,s}$. By induction, $G_{1,3}$ can be represented by a vector (u_1, u_2) and $G_{2,s}$ by $(u'_1, u'_2, \ldots, u'_{s-2})$. By Lemma 3, we can permute the crossbars in stage k, $2 \leq k \leq s$, such that $u'_1 = u_2$ and $u''_k = u'_{k-1}$ for $3 \leq k \leq s-1$. Therefore, the subnetwork $G_{1,s}$ is represented by the vector $(u_1, u_2, u''_3, \ldots, u''_{s-1})$, that is, $G_{1,s}$ is a BP network. ∎

Corollary 10. *Two* $d^P UB$ *networks are equivalent if the characterization vector of one can be obtained from the other through a permutation.*

Figure 6(a) gives an example of a d^P network, which is not equivalent to a UB network. Hence, $d^P \nsim UB$. Because $UB \supset BP$, Figure 6(a) is also an example of a d^P network, which is not equivalent to a BP network. Therefore, the UB condition cannot be dropped from Theorem 9. Because $BP \sim d^P UB$, it follows that $d^P \nsim d^P UB$. Figure 6(b) gives a UB (or strict buddy) network which is equivalent to neither a d^P nor a BP network. Hence, $UB \nsim BP$. Because $BP \sim d^P UB$, it follows that $UB \nsim d^P UB$.

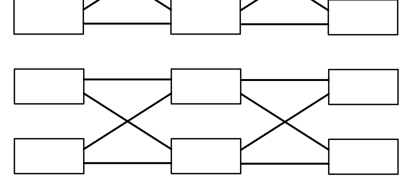

FIG. 8. Concatenation of $K_{2,2}$.

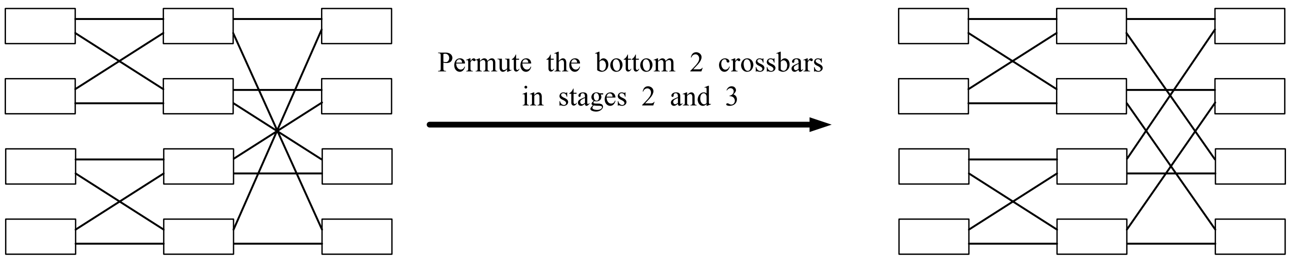

FIG. 9. A permutation to achieve *BP*.

4. CONCLUSIONS

We established the containment relation given in (1.1), and the equivalence relation given in (1.2). By so doing, we achieve three desirable generalizations:

(1) We make the logical extension of the buddy network and the strict buddy network to the universal buddy network: a network with more structure but which still includes all banyan-type networks and their extra-stage versions.
(2) We generalize the notion of *BP* to $d^P UB$, which is a larger class, yet it preserves all connecting properties of *BP*.
(3) We generalize $P(*,*)$ which is defined only for $n = \log_d N$ stages to general s stages.

The equivalence relations we established also help in simplifying some existing proofs:

(1) The proof of a vector characterization of *BP* in [5] is quite complicated. We gave a simple proof of a vector characterization of *BPT* and the equality $BPT = BP$ makes the proof valid for *BP* also.
(2) The proof that $P(*,*)$ characterizes the Baseline-equivalent class of banyan-type networks is very long, as admitted in [2]. Our proofs of Theorem 9 and Corollary 10 are much shorter and more general.

ACKNOWLEDGMENTS

The author thanks H. Zhou for providing a counter-example to a conjecture prior to our discovery of Theorem 9. We also appreciate the comments of referees that led to a better version of this article.

REFERENCES

[1] D.P. Agrawal, Graph theoretical analysis and design of multistage interconnection networks, IEEE Trans Comput 32 (1983), 637–648.
[2] J.C. Bermond, J.M. Fourneau, and A. Jean-Marie, Equialence of multistage interconnection networks, Inform Proc Lett 26 (1987), 45–50.
[3] J.C. Bermond, J.M. Fourneau, and A. Jean-Marie, A graph theoretical approach to equivalence of multistage interconnection networks, Disc Appl Math 22 (1988/89), 201–217.
[4] T. Calamoneri and A. Massini, Efficient algorithm for checking the equivalence of multistage interconnection networks, J Parallel Distrib Comput 64 (2004), 135–150.
[5] G.J. Chang, F.K. Hwang, and L.D. Tong, Characterizing bit permutation networks, Networks 33 (1999), 261–267.
[6] D.M. Dias and J.R. Jump, Analysis and simulation of buffered delta networks, IEEE Trans Comput C-30 (1981), 273–282.
[7] Q. Hu, X. Shen, and J. Yang, Topologies of combined $(2\log N - 1)$-stage interconnection networks, IEEE Trans Comput 46 (1997), 118–124.
[8] F.K. Hwang, The mathematical theory of nonblocking switching networks, World Scientific, Singapore, 2004.
[9] S.-Y.R. Li, Algebraic switching theory and broadband applications, Academic Press, New York, 2001.
[10] D.S. Parker, Notes on shuffle-exchange type of networks, IEEE Trans Comput 29 (1980), 213–222.
[11] D.J. Shyy and C.T. Lea, $\text{Log}_2(N, m, p)$ strictly nonblocking networks, IEEE Trans Commun 39 (1991), 1502–1510.
[12] H.J. Siegel and S.D. Smith, Study of multistage SIMD interconnection networks, Proc 5th Ann Symp Comput Arch, Palo Alto, CA, 1978, pp. 223–229.
[13] C. Wu and T. Feng, On a class of multistage interconnection networks, IEEE Trans Comput 29 (1980), 694–702.