TOPOLOGICAL PROPERTIES OF BANYAN-HYPERCUBE NETWORKS

A. Youssef and B. Narahari

Department of Electrical Engineering and Computer Science
The George Washington University
Washington, DC 20052

Abstract

This paper discusses the topological properties of a recently introduced family of networks, called the *banyan-hypercubes* (BH), and defines a family of generalized banyan hypercubes. A banyan-hypercube, denoted BH(h,k,s), is constructed by taking the bottom h levels, from the base, of a rectangular banyan of spread s and s^k nodes per level for s a power of 2, and interconnecting the nodes at each level in a hypercube. The banyan-hypercubes can be viewed as a scheme for interconnecting hypercubes while keeping most of the advantages of the latter. In this paper, the definition of BH's will be extended and generalized to (1) allow the interconnection of an unlimited number (h) of hypercubes and (2) allow any h successive levels of the banyan to interconnect hypercubes. This leads to better extendability and flexibility in partitioning the BH. The diameter and average distance of the generalized BH will be derived and shown to provide an improvement over the hypercube for a wide range of values for h, k and s. Self-routing point to point and broadcasting algorithms will be presented and efficient embeddings of various networks, on the BH, will be shown.

§1. Introduction

Amongst the various interconnection networks that have been studied and built, hypercube networks have received much attention over the past few years. Hypercubes have a rich interconnection structure with logarithmic diameter, average distance and degree. Furthermore, it has a simple and elegant routing strategy and many toplogical structures such as rings, meshes, and trees can be efficiently embedded in the hypercube [2][7] [14]. The main disadvantage of the hypercube lies in its extendability; extendability allows for the gradual growth of systems. Extending the hypercube (i.e., adding more nodes) requires doubling its size and incrementing the node degree, thus causing cost and hardware problems. Due to the same reasons, the hypercube results in large internal fragmentation when it is partitioned. Therefore, a desirable network structure is one that retains the advantages of the hypercube while providing better, i.e., cheaper, extendability and more flexible partitioning.

A new family of hierarchical, partitionable networks that are a synthesis of rectangular banyans [6] [4] and hypercubes has been proposed and shown to greatly reduce the aforementioned disadvantages [15]. These networks, called the *banyan-hypercubes* (BH), are constructed by taking the first h successive levels (from the base) of a $(k+1)$-level rectangular banyan of spread (outdegree) and fan-out (indegree) equal to s, where s is a power of 2 and $h \leq k+1$. Each level has s^k nodes representing processing elements, labelled from 0 to $s^k - 1$ in binary, and *interconnected* as a hypercube. Such a network is denoted BH(h,k,s). A hypercube is a BH$(1,k,s)$ network and hence a special case.

BH networks are shown in [15] to combine the advantageous features of banyans and hypercubes and thus have better communication capabilities. In particular, many hypercube features in routing, embedding and partitioning are incorporated into banyan-hypercubes and new gains are achieved in diameter, average distance, embedding efficiency, partitioning flexibility and lower cost extendability. It should be noted that many multistage networks have been shown to be topologically equivalent and thus their synthesis with the hypercube results in properties identical to those of the banyan-hypercube.

The limitations of the banyan-hypercube networks, as defined in [15], are that the maximum number of levels are restricted (to $k+1$) and the inter-level connections must start from the base of the banyan. In this paper we will extend the definition of the banyan-hypercubes in two ways and thereby define the family of *generalized banyan-hypercube* networks. The first extension is to allow any h successive levels of BH$(k+1,k,s)$ to be a banyan-hypercube. This leads to more flexible and efficient partitioning, and will be justified by showing that any h successive levels of BH$(k+1,k,s)$ are isomorphic to the first h levels of BH$(k+1,k,s)$. The second extension is to lift the limitation $h \leq k+1$. This leads to unlimited extendability and enhances the flexibility and efficiency of partitioning. It will be shown that even after the second extension, it still holds that any h successive levels are isormorphic to the first h levels.

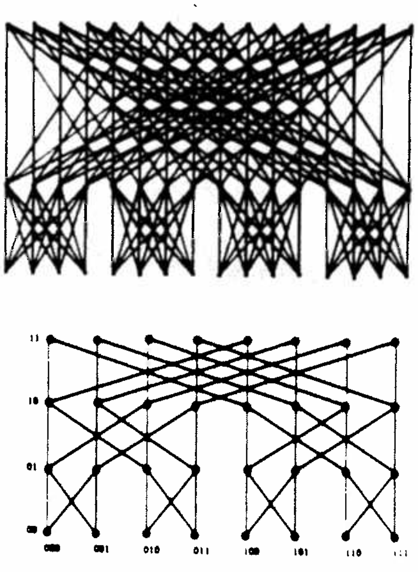
This paper makes a number of contributions to the study of banyan-hypercubes. The first is to allow the network to start from any level of the banyan thus enhancing its partitioning flexibility. Secondly, we allow the number of levels to grow arbitrarily and show that any h levels are isomorphic to any other h levels. Thirdly,

CH2908-2/90/0000/0324/$01.00 - 1990 IEEE

we derive the diameter and average distance of the generalized banyan-hypercube, and show that for reasonably large numbers of levels the performance of the BH networks provides an improvement over the hypercube. Next, optimal algorithms for point-to-point routing and one-to-all routing (i.e., broadcasting) will be presented. Finally, we note that by using the isomorphic property, the embeddings of rings, meshes, trees, pyramids and multiple pyramids, provided in [15] can be extended to the generalized BH networks. Partitioning the Banyan-Hypercube is discussed in [16].

§2. Structure of the Banyan-Hypercube

We first briefly review the definition of Banyan- hypercube networks as given in [15]. A *banyan* graph [4] [6] [10] is a Hasse diagram of a partial ordering where there is a unique path from every base to every apex. A base is any vertex having no arcs incident into it, and an apex is any nóde having no arcs incident out from it. An *L-level* banyan is a banyan whose nodes can be arranged into L levels so that the arcs are only between adjacent levels. A *regular* banyan is an L-level banyan where all the nodes except the bases have the same indegree F called the *fanout* and all the nodes except the apexes have the same outdegree S called the *spread*. A *rectangular* banyan is a regular banyan where $S = F$. In this case, all the levels have the same number of nodes, which is S^{L-1}. Figure 1 shows a rectangular banyan of spread 2 and another of spread 4.



Two banyan networks
Figure 1

Normally the base nodes represent system resources such as memory modules and processors, while the other nodes are switching elements. In this paper all the nodes of a banyan will represent processing elements, and the arcs will be treated as undirected edges representing bidirectional links. The spread will be assumed to be a power of two so that each level has a power of two number of nodes and can be interconnected in a hypercube structure.

A hypercube of dimension k, or a k-cube, is a graph of 2^k nodes labelled in k-bit binary labels such that there is an edge between every two nodes that differ by exactly one bit.

In what follows, we first give a restricted definition of a banyan-hypercube (as stated in [15]) and then give a more general definition in the next section. A *banyan-hypercube* BH(h, k, s), where $h \leq k+1$ and s is a power of two ($s = 2^r$ for some r), is a graph of h levels of nodes where each level has s^k nodes interconnected in a hypercube structure and the inter-level edges form the first h levels (from the base) of a $(k+1)$-level rectangular banyan of spread s. The nodes in each level are labelled in binary from 0 to $s^k - 1$ and the levels are numbered from 0 to $h-1$ from the base to the top. Every node is then uniquely identified by a pair (L, X) of its level number L and its cube label X. The label X can be viewed as $x_{k-1}...x_1x_0$ in the number system of base s where each x_i is an s-ary digit, or as a binary label of $k \log s$ bits $t_{kr-1}...t_0$ such that $x_i = t_{(i+1)r-1}...t_{ir+1}t_{ir}$. Clearly the total number of nodes is $N = hs^k$. Figure 2 shows two banyan-hypercube networks.

BH(h, k, s) can be specified more formally in graph theoretical terms as follows. Let $e^a_L(X) = e^a_L(x_{k-1}...x_1x_0) = x_{k-1}...x_{L+1}ax_{L-1}...x_1x_0$, where $0 \leq a \leq s-1$ and $X = x_{k-1}...x_1x_0$ in base system s. In other terms, e^a_L, when applied to X, replaces the s-ary digit of X in position L by the s-ary digit a. Let also e_i be the i-th exchange which, when applied to a binary number, complements the i-th bit (from the right). That is, $e_i(X) = e_i(t_{kr-1}...t_0) = t_{kr-1}...t_{i+1}\bar{t}_it_{i-1}...t_0$.

2.1. Definition. BH(h, k, s) is an undirected graph $G = (V, E)$ such that

$$V = \{(L, X) \mid 0 \leq L \leq h-1,\ 0 \leq X \leq s^k - 1\}$$

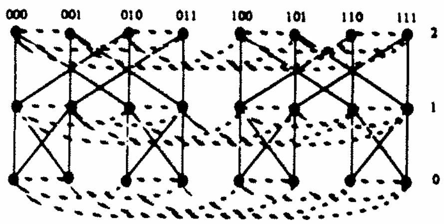
and $E = E_b \cup E_c$ where

$E_b = \{\langle (L, X), (L+1, e^a_L(X)) \rangle \mid 0 \leq L \leq h-2, 0 \leq X \leq s^k - 1, 0 \leq a \leq s-1\}$

and $E_c = \{\langle (L, X), (L, e_i(X)) \rangle \mid 0 \leq L \leq h-1, 0 \leq X \leq s^k - 1, 0 \leq i \leq k \log s - 1\}$.

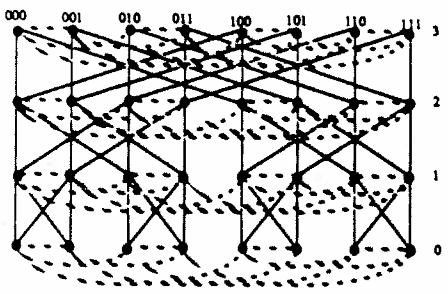
Note that E_b is the set of the banyan edges and E_c is the set of cube edges in all the levels. Note also that the connections between level L and level $L+1$ include two subsets, one being the set of "vertical" edges corresponding to $\langle (L, X), (L+1, e^{x_L}_L(X)) \rangle$'s, and the other the set of the exchange connections e_i for $(L+1)r - 1 \leq i \leq Lr$ corresponding to $\langle (L, X), (L+1, e^a_L(X)) \rangle$ where $a = x_{(L+1)r-1}...x_{i+1}\bar{x}_ix_{i-1}...x_{Lr}$.

§3. The Generalized Banyan-Hypercube



BH(3,3,2)

(a)

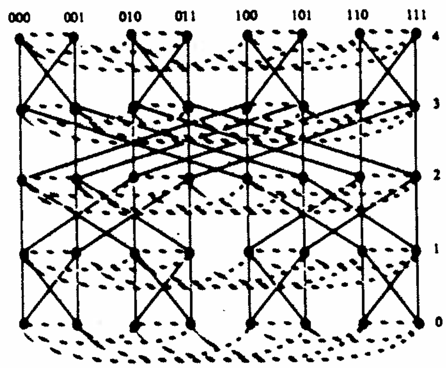


BH(4,3,2)

(b)

The dashed line show the hypercube connections
The solid lines show the Banyan connections

Banyan-Hypercubes
Figure 2



BH(5,3,2)
Figure 3

In this section we define the generalized BH networks by allowing any number of levels and any successive banyan connections between levels. We first define the two extensions and then provide the proofs to justify these definitions.

Level Extension of Banyan-hypercubes: This extension allows h to be greater than $k+1$. It is carried out simply by repeating after level k the banyan connections from the base. That is, the connection between level k and level $k+1$ is the same as the connection between level 0 and level 1, the connection between level $k+1$ and level $k+2$ is the same as the connection between level 1 and level 2, and so on. In general, the connection between level i and level $i+1$ is the same as the connection between level $i\%k$ (i.e., i modulo k) and level $(i+1)\%k$. Figure 3 shows a level-extended BH.

Formally, BH(h,k,s), where $h \geq 1$ and unbounded from above, is the same undirected graph $G=(V,E)$ as in Definition 2.1 with one modification to e_L^a so that $e_L^a(X) = e_L^a(x_{k-1}...x_1x_0) = x_{k-1}...x_{(L\%k)+1}ax_{(L\%k)-1}...x_1x_0$ that is, e_L^a is defined to be $e_{L\%k}^a$.

Extension to Inner Level Banyan-hypercubes: This extension allows any h successive levels of the level-extended banyan-hypercubes to be banyan-hypercubes. The notation of BH's has to be modified somewhat to indicate the bottom and top levels of the h successive levels. We denote by BH$(j, j+h-1, k, s)$ the subgraph consisting of the levels $i, i+1, i+2, ...j+h-1$ (along with their edges) of the level-extended BH(h', k, s), where $h' \geq j+h$. Using this notation, the BH in Figure 3 is BH(0,4,3,2), and the top three levels of that network form BH$(2,4,3,2)$.

In graph theoretical terms, the banyan-hypercube BH$(j, j+h-1, k, s)$, where $j \geq 0$ and $h \geq 1$, is an undirected graph $G=(V,E)$ such that

$V = \{(L,X) \mid j \leq L \leq j+h-1,\ 0 \leq X \leq s^k-1\}$

and $E = E_b \cup E_c$ where

$E_b = \{\langle (L,X), (L+1, e_L^a(X))\rangle \mid j \leq L \leq j+h-2,\ 0 \leq X \leq s^k-1,\ 0 \leq a \leq s-1\}$

and

$E_c = \{\langle (L,X), (L, e_i(X))\rangle \mid j \leq L \leq j+h-1,\ 0 \leq X \leq s^k-1,\ 0 \leq i \leq k \log s - 1\}$.

Note that here too e_L^a is the same as $e_{L\%k}^a$.

Observe that BH$(0, h-1, k, s)$ is identical with BH(h, k, s). We will continue using the notation BH(h,k,s) when the bottom level is level 0, and use the other notation when the bottom level is not level 0. It will be shown in the later sections that the extended, i.e., generalized, banyan-hypercubes have the same optimality of routing and the same embedding capabilities as the original banyan-hypercube. Also, they can be shown to have more partitioning flexibility and efficiency.

In the remainder of this section we will show the isomorphism between BH$(j, j+h-1, k, s)$ and BH$(0, h-1, k, s)$. To this effect, let U be the unshuffle permutation of $\{0, 1, ..., s^k-1\}$ defined on the k-digit s-ary representations of integers. That is,

$$U(x_{k-1}...x_1x_0) = x_0x_{k-1}x_{k-2}...x_1$$

where $x_{k-1}...x_0$ is a k-digit number in the base system s. The following two lemmas will prove useful in the proof of the ismorphism theorems. Their proofs are straightforward and hence omitted.

3.1 Lemma. *$Ue_L^a = e_{L-1}^aU$ for every $L \geq 1$ and every a, $0 \leq a \leq s-1$.*

3.2 Lemma. *If the binary representations of X and X' differ in only one bit, then the binary representations of $U(X)$ and $U(X')$ also differ in only one bit.*

3.3 Theorem. *$BH(j, j+h-1, k, s)$ is isomorphic to $BH(j-1, j+h-2, k, s)$ for every $h \geq 1$ and $j \geq 1$.*

Proof. Let $\mathrm{BH}(j, j+h-1, k, s) = (V, E = E_b \cup E_c)$ and $\mathrm{BH}(j-1, j+h-1, k, s) = (V', E' = E'_b \cup E'_c)$ where

$V = \{(L, X) \mid j \leq L \leq j+h-1,\ 0 \leq X \leq s^k - 1\}$

$E_b = \{\langle (L, X), (L+1, e^a_L(X))\rangle \mid j \leq L \leq j+h-2,\ 0 \leq X \leq s^k - 1,\ 0 \leq a \leq s-1\}$

$E_c = \{\langle (L, X), (L, e_i(X))\rangle \mid j \leq L \leq j+h-1,\ 0 \leq X \leq s^k - 1,\ 0 \leq i \leq k \log s - 1\}$

$V' = \{(L, X) \mid j-1 \leq L \leq j+h-2,\ 0 \leq X \leq s^k - 1\}$

$E'_b = \{\langle (L, X), (L+1, e^a_L(X))\rangle \mid j-1 \leq L \leq j+h-3,\ 0 \leq X \leq s^k - 1,\ 0 \leq a \leq s-1\}$

$E'_c = \{\langle (L, X), (L, e_i(X))\rangle \mid j-1 \leq L \leq j+h-2,\ 0 \leq X \leq s^k - 1,\ 0 \leq i \leq k \log s - 1\}$.

Clearly, $|V| = |V'|$ and $|E| = |E'|$. Let f be the following mapping from V to V':

$$f((L, X)) = (L-1, U(X)).$$

It can be seen that f is one-to-one and onto. It remains to be shown that if $\langle u, v\rangle$ is an edge in E, then $\langle f(u), f(v)\rangle$ is an edge in E'.

Let $\langle u, v\rangle = \langle (L, X), (L+1, e^a_L(X))\rangle$ be an edge in E_b. Then $\langle f(u), f(v)\rangle = \langle (L-1, U(X)), (L, U(e^a_L(X)))\rangle = \langle (L-1, U(X)), (L, e^a_{L-1}(U(X)))\rangle$ (making use of Lemma 3.1).

Thus, $\langle f(u), f(v)\rangle$ is an edge in E'_b since $j-1 \leq L-1 \leq j+h-1$. Similarly, let $\langle u, v\rangle = \langle (L, X), (L, e_i(X))\rangle$ be an edge in E_c. $\langle f(u), f(v)\rangle = \langle (L-1, U(X)), (L-1, U(e_i(X)))\rangle$. As the binary representations of X and $e_i(X)$ differ only in one bit, it follows from Lemma 3.2 that $U(X)$ and $U(e_i(X))$ differ in only one bit. Consequently, $\langle (L-1, U(X)), (L-1, U(e_i(X)))\rangle$ is an edge in E'_c. Hence, f is an isomorphism from $\mathrm{BH}(j, j+h-1, k, s)$ to $\mathrm{BH}(j-1, j+h-2, k, s)$. ∎

3.4 Theorem. *$BH(j, j+h-1, k, s)$ is isomorphic to $BH(0, h-1, k, s)$ for every $h \geq 1$ and $j \geq 1$.*

Proof. This theorem follows from a repeated use of the previous theorem. That is, After Theorem 3.3, we have $BH(j, j+h-1, k, s) \simeq BH(j-1, j+h-2, k, s) \simeq BH(j-2, j+h-3, k, s) \simeq \ldots \simeq BH(0, h-1, k, s)$ where $\simeq$ is the sign for isomorphism. Moreover, the isomorphism from $\mathrm{BH}(j, j+h-1, k, s)$ is f^j where

$$f^j((L, x_{k-1} \ldots x_0)) = (L-j, x_{j-1} x_{j-2} \ldots x_0 x_{k-1} x_{k-2} \ldots x_j).$$

∎

The above theorem proves that any two subgraphs of h levels (of a full banyan-hypercube of $k+1$ levels) are isomorphic to $\mathrm{BH}(h, k, s)$ and thus to each other. This isomorphism justifies the inner-level extension of the banyan-hypercube definition. As the isomorphism holds even after the level extension, the above theorem shows then that our level extension "preserves" the topological properties of the original banyan-hypercubes. In particular, all the embeddings with the minimum dilation costs of meshes, rings, trees, pyramids and multiple pyramids on $\mathrm{BH}(h, k, s)$ for $h \leq k+1$ [15] can be inherited by any $\mathrm{BH}(j, j+h-1, k, s)$ using the ismorphism of Theorem 3.4. The generalized BH is seen to have more partitioning flexibility [16] since any h levels form a BH. A detailed discussion of the partitioning of the BH network is discussed in [16].

Although most of the study carried out in this paper applies to BH's for any values of s, in practice s is preferred to be 2 or 4 because the resulting banyan-hypercube meets all the requirements of partitioning and embedding without incurring a high degree cost.

§4. Properties of Banyan-Hypercubes

In this section we shall evaluate the degree, diameter and average distance of BH networks. The degree of $\mathrm{BH}(i, i+h-1, k, s)$ can be easily seen to be $2s + k \log s$, the sum of the degree $2s$ of the banyan of spread s and the degree $k \log s$ of the hypercube of s^k nodes. The degree is independent of the number of levels. The degree of the hypercube of hs^k nodes (when h is a power of two) is $\log hs^k$ which is equal to $\log h + k \log s$. The difference between the degree of $\mathrm{BH}(j, j+h-1, k, s)$ and the degree of the hypercube is $2s - \log h$. Therefore, the degree of the hypercube is asymtotically larger (e.g., $h \geq 4^s$) than that of the banyan-hypercube of the same size, assuming fixed s. However, for practical values of s (2 or 4), the degree of the BH is slightly larger. The corresponding extra hardware cost is nevertheless justified by the smaller diameter and average distance of the banyan-hypercube, as well as its added embedding capabilities and its flexible extendability.

To facilitate the analysis of banyan-hypercubes and the evaluation of the diameter and average distance, a class of networks, called product networks, is first introduced and state some results that we use in the following sections. A full treatment of product networks is given in [17].

4.1 Product Networks

Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two undirected graphs. The *product network* of G_1 and G_2, denoted $G_1 G_2$, is a graph $G = (V, E)$ such that

$$V = V_1 V_2 = \{xx' \mid x \in V_1,\ x' \in V_2\}$$

and

$$E = \{(xx', xy') \mid x \in V_1,\ (x', y') \in E_2\} \cup \{(xx', yx') \mid x' \in V_2,\ (x, y) \in E_1\}.$$

That is, (xx', yy') is an edge in the product network if and only if either $x = y$ and (x', y') is an edge in G_2, or $x' = y'$ and (x, y) is an edge in G_1.

Intuitively, $G_1 G_2$ is constructed by taking $|V_2|$ 'copies' of G_1, and interconnecting in G_2-structure the sibling nodes in the copies of G_1, where two nodes xx' and yy' are siblings if $x' = y'$.

Several existing networks are product networks. For instance, a mesh is the product network of two lines of nodes, a torus is the product network of two rings, and a hypercube is the product network of two subcubes. Also, as was pointed out in section 2, if $h < k+1$, BH(h,k,s) can be constructed from s copies of BH$(h,k-1,s)$ by interconnecting the sibling nodes in a hypercube of s nodes. It follows that BH(h,k,s)= BH$(h,k-1,s)$CUBE(s), where CUBE(s) is a hypercube of s nodes. Hence the relevance of product networks to banyan-hypercubes.

While a full treatment of product networks is outside the scope of this paper, several of their salient features that are relevent to the analysis of banyan-hypercubes will be stated and the proofs are included in [17].

4.1 Lemma. *Let $d_G(x,y)$ denote the distance between the nodes x and y in a graph G. Then, $d_{G_1G_2}(xx',yy') = d_{G_1}(x,y) + d_{G_2}(x',y')$.*

Proof. In [17]. ∎

4.2 Theorem. *Let $D(G)$ denote the diameter of the graph G. Then, $D(G_2G_2) = D(G_1) + D(G_2)$.*

Proof. In [17] ∎

4.3 Theorem. *Let $\overline{d}_G$ denote the average distance of the graph G, and $S(G)$ the sum of distances between all the pairs of nodes. Then, $S(G_1G_2) = |V_2|^2S(G_1) + |V_1|^2S(G_2)$, and $\overline{d}_{G_1G_2} = \overline{d}_{G_1} + \overline{d}_{G_2}$.*

Proof. In [17] ∎

4.2 Diameter of BH's

Due to the ismorphism theorem, the diameter and average distance of BH$(j, j+h-1, k, s)$ are respectively identical to those of BH(h,k,s). Hence, the discussion in this and the next subsection is limited to BH(h,k,s).

To evaluate the diameter of BH(h,k,s), we will make use of the product networks and the results in [15]. The following theorem from [15] gives the diameter of BH$(k+1,k,s)$.

4.4 Theorem. *The diameter of BH$(k+1,k,s)$ is k for $s=2$, and $2k$ for $s \geq 4$.*

The next theorem gives the diameter of BH(h,k,s) in its full generality.

4.5 Theorem. *If $h \leq k+1$, the diameter of BH(h,k,s) = k for $s=2$, and it is equal to $k\log s - (h-1)\log\frac{s}{4}$ for $s \geq 4$. If $h \geq k+1$, the diameter of BH(h,k,s) is $h-1$ for $s=2$, and it is $\max(h-1, 2k)$ for $s \geq 4$.*

Proof. If $h = k+1$, the theorem follows from Theorem 4.4.

If $h < k+1$, BH(h,k,s) can be easily seen to be the product network BH$(h,h-1,s)$CUBE(s^{k-h+1}). Therefore, D(BH(h,k,s)) = D(BH$(h,h-1,s)$) + D(CUBE(s^{k-h+1})). Since D(CUBE(s^{k-h+1})) $= \log s^{k-h+1} = (k-h+1)\log s$, and D(BH$(h,h-1,s)$) is $(h-1)$ for $s=2$, and $2(h-1)$ for $s \geq 4$ (after Theorem 4.4), it follows that for $s=2$, D(BH(h,k,s)) $= h-1+(k-h+1)\log 2 = k$, and for $s \geq 4$, D$(BH(h,k,s))$ $= 2(h-1)+(k-h+1)\log s = k\log s - (h-1)\log\frac{s}{4}$.

If $h \geq k+1$, it can be seen that D(BH(h,k,s)) = max$(h-1, D(BH(k+1,k,s))$ since the distance between two nodes that are at most $k+1$ levels apart is $\leq$ D(BH$(k+1,k,s)$), while if the two nodes are more than $k+1$ levels apart, their distance is the difference between their level numbers because there is a "pure" banyan path between the two nodes. Using now the previous theorem for the value of D(BH$(k+1,k,s)$), the theorem follows for the case $h \geq k+1$. ∎

As the diameter reflects only the worst case communication time, the average distance conveys better in practice the actual performance of the network.

4.3 Average Distance of BH's

The sum S of the distances between all the pairs of nodes in BH$(k+1,k,s)$ was derived in [15] and shown to be

$$S = s^{2k}[(\frac{\sigma_s}{s^2} + \frac{\log s}{2} + 1)\frac{k(k+1)(2k+1)}{6} + \frac{k(k+1)}{2}]$$

where σ_s is the sum of all distances of CUBE(s). It is well-known that the average distance of a hypercube is half the logarithm of its number of nodes. Hence, $\sigma_s = \frac{s^2\log s}{2}$. This simplifies the value of S to

$$S = s^{2k}[\log(2s)\frac{k(k+1)(2k+1)}{6} + \frac{k(k+1)}{2}].$$

Dividing S by $(s^{2k}(k+1)^2$, which is the square of the number of nodes of BH$(k+1,k,s)$, we get the average distance of BH$(k+1,k,s)$. This is summarized in the following theorem

4.6 Theorem. *The average distance of BH$(k+1,k,s)$ is $\frac{k(2k+1)}{6(k+1)}\log(2s) + \frac{k}{2(k+1)}$.*

This theorem will be used to derive the average distance of BH(h,k,s) for all values of h, k, and s.

4.7 Theorem. *For $h < k+1$, the average distance of BH(h,k,s) is $\frac{h^2-1}{3h}[1-\frac{\log s}{2}] + k\frac{\log s}{2}$.*

Proof. It was pointed out earlier that BH(h,k,s) is equal to the product network BH$(h,h-1,s)$CUBE(s^{k-h+1}). Using Theorem 4.3 for the average distance of product networks, and Theorem 4.6 for the average distance of BH$(h,h-1,s)$, and the fact that the average distance of $CUBE(s^{k-h+1})$ is $(k-h+1)\frac{\log s}{2}$, the theorem immediately follows. ∎

4.8 Theorem. *For $h \geq k+1$, the average distance of BH(h,k,s) is*

$$\frac{h^2-1}{3h} + \frac{k(k+1)(5k+1)\log s}{6h^2} - \frac{k^2\log s}{2h}.$$

Proof. Let S_h be the sum of all distances in BH(h,k,s). S_h will be shown to satisfy the following recurrence relation:

$$S_h = S_{h-1} + [h(h-1) + k^2\frac{\log s}{2}]s^{2k} \qquad (1)$$

The recurrence relation can then be solved using standard linear recurrence relations techniques and the average distance will follow.

To derive relation (1), let L be the set of nodes in level $h-1$ (i.e., top level) of BH(h,k,s), T the set of

nodes of BH($h-1, k, s$), and D the set of nodes of the whole network BH(h, k, s). Let also x and y represent two arbitrary nodes in BH(h, k, s), and $d(x,y)$ the distance between x and y. As $D \times D = (T \times T) \cup (L \times D) \cup [(D \times L) - (L \times L)]$, it follows that
$S_h = \sum_{(x,y)\in D\times D} d(x,y) = \sum_{(x,y)\in T\times T} d(x,y) +$
$2\sum_{(x,y)\in L\times D} d(x,y) - \sum_{(x,y)\in L\times L} d(x,y)$ (2).

The first term of the right hand side of (2) is S_{h-1} because T =BH($h-1, k, s$), and the third term is the sum of all distances in a hypercube of s^k nodes, and hence is equal to $s^{2k}k\frac{\log s}{2}$.

To compute $\sum_{L\times D} d(x,y)$, let D_1 be the set of nodes of the top $k+1$ levels of BH(h, k, s), and D_2 the set of nodes in the remaining $h-k-1$ levels.

$$\sum_{(x,y)\in L\times D} d(x,y) = \sum_{(x,y)\in L\times D_1} d(x,y) + \sum_{(x,y)\in L\times D_2} d(x,y)$$

Note that $\sum_{(x,y)\in L\times D_2} d(x,y) = |L|^2 \sum_{0\le j\le h-k-2}(h-1-j)$ because the distance between any node in the top level L and any node in level j in D_2 is $h-1-j$. It follows that

$$\sum_{(x,y)\in L\times D_2} d(x,y) = \frac{s^{2k}}{2}[h(h-1) - k(k+1)]$$

Let $a_k = \sum_{(x,y)\in L\times D_1} d(x,y)$.
$a_k = \sum_{(x,y)\in L\times D_1} d(x,y)$
$= \sum_{(x,y)\in L\times L} d(x,y) + \sum_{(x,y)\in L\times D_1'} d(x,y)$
$= s^{2k}k\frac{\log s}{2} + \sum_{(x,y)\in L\times D_1'} d(x,y)$
where D_1' is $D_1 - L$. Recall that D_1 is isomorphic to BH($k+1, k, s$) and D_1' isomorphic to BH(k, k, s) which consists of s copies of BH($k, k-1, s$). Call these copies $E_1, E_2, \ldots E_s$. In the computation of our terms here, we can safely assume that D_1 is identical to BH($k+1, k, s$) and D_1' to BH(k, k, s). Let L' be the top level of E_1. As x has the same relationship to each of the E_i, it follows that:
$\sum_{(x,y)\in L\times D_1'} d(x,y) = s\sum_{(x,y)\in L\times E_1} d(x,y)$
$= s\sum_{(x',y)\in L'\times E_1} s(1 + d(x',y))$
$= ks^{2k} + s^2\sum_{(x',y)\in L'\times E_1} d(x',y)$

In the second equality, we made use of $d(x,y) = 1 + d(x',y)$, where x' is the neighbor of x in L'. We also used the fact that x' has s neighbors in L.
Observing that $a_{k-1} = \sum_{(x',y)\in L'\times E_1} d(x',y)$, we conclude

$$a_k = ks^{2k}(1 + \frac{\log s}{2}) + s^2 a_{k-1}. \qquad (3)$$

Letting $c_k = \frac{a_k}{s^{2k}}$ transforms (3) to $c_k = c_{k-1} + k(1 + \frac{\log s}{2})$. This recurrence relation is linear and its solution is $c_k = (1 + \frac{\log s}{2})\frac{k(k+1)}{2}$, making use of the initial value $c_0 = a_0 = 0$. Therefore,

$$a_k = s^{2k}(1 + \frac{\log s}{2})\frac{k(k+1)}{2}.$$

Now that the 3 sums of the right hand side of (2) are known, substituting them will result in equation (1). This equation is a simple linear recurrence relation, valid for $h-1 \ge k+1$, and can thus be easily solved, yielding $S_h = S_{k+1} + [\frac{h(h^2-1)}{3} - \frac{k(k+1)(k+2)}{3} + (h-k-1)k^2\frac{\log s}{2}]s^{2k}$.

To derive the average distance $\overline{d}(BH(h,k,s))$, divide S_h by $(hs^k)^2$, make use of the fact that $\overline{d}(BH(k+1,k,s)) = \frac{S_{k+1}}{(k+1)^2 s^{2k}}$, and replace $\overline{d}(BH(k+1,k,s))$ by its value given in Theorem 4.6. This results in the desired expression of the average distance of BH(h, k, s). ∎

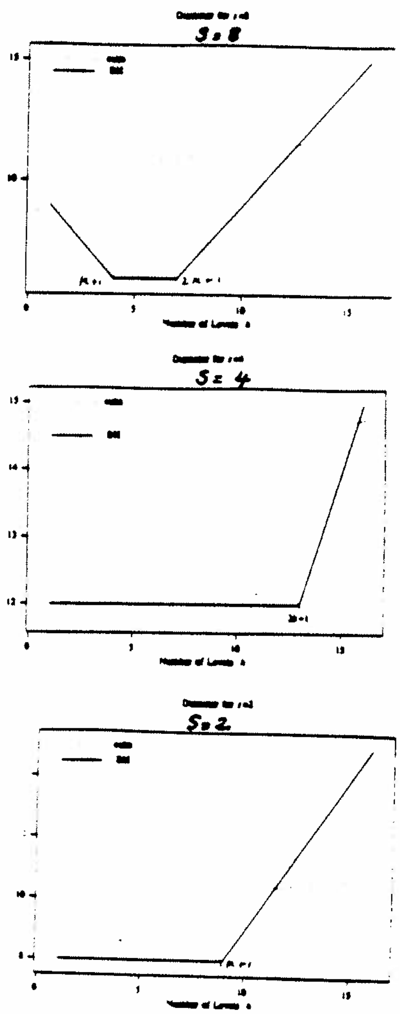
4.4 Behavior of the Diameter and Average Distance

As the BH family is parametrized, it is of interest to study the behavior of the diameter and average distance with respect to the three parameters h, k and s. From the practical standpoint, the BH's can be viewed as a scheme of interconnecting a number (h) of hypercubes. Therefore, it is important to know the range of values for the parameters, particularly the number of levels h, for which the BH's yield an optimal or nearly optimal communication performance. It is also useful to determine the advantages and disadvantages of the banyan-interconnected hypercubes vis-a-vis mere hypercubes.

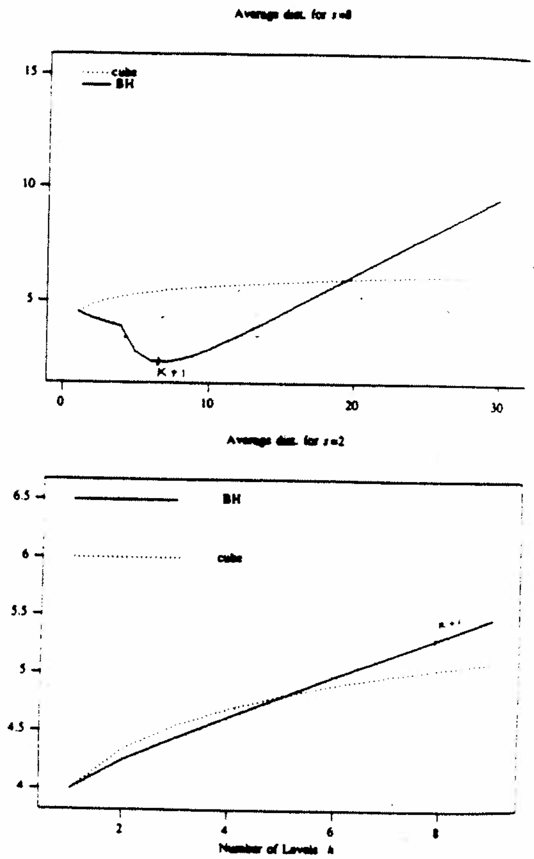
Figure 4 and Figure 5 plot the diameter and average distance of BH(h, k, s) and the hypercube of the same number of nodes hs^k. The plots are presented as if h were a continous variable in order to better illustrate the behavior. For example, the diameter, $\log h + k\log s$, of the hypercube of hs^k nodes is ploted as a continuous function in h even for values of h that do not correspond to a proper hypercube size. Note that some of the values shown in the plots were derived through mathematical analysis of the formulas of the diameter and average distance given in the previous two subsections. This analysis is omitted due to space limitation.

It can be seen from the plots that the banyan- hypercubes provide an improvement over the hypercubes for a wide range of values of h and k, particularly when $s \ge 4$. It can also be seen that extending the number of levels beyond the previously imposed limit of $k+1$ decreases the performance of BH's, but still offers an improvement over the hypercubes for a fairly wide range of h when $s \ge 4$. In particular, for $s = 4$, the diameter of BH($h, k, 4$) remains equal to k for $h \le 2k+1$, and remains smaller than that of the hypercube for $h \le 2k+1+\log(2k+1)$. The diameter of the BH when $s = 8$ initially decreases when we start increasing h and stays constant before increasing again. This is due to the banyan connections, i.e., by using the banyan edges to go between levels the distance between two nodes at the same level (in a 8^k hypercube) decreases (i.e., it is no longer $\log(8^k)$).

The average distance of BH($h, k, 4$) is equal to k for $h \le k+1$, and then decreases slightly before increasing steadily at a small rate, remaining smaller than the average distance of the hypercube for $h \le (1+\sqrt{6})(k+1)$. This is due to the fact that we are adding 4^k nodes when



Diameter Behavior of BH(h, k, s) and CUBE(hs^k)
Figure 4



Average Distance Behavior of BH(h, k, s) and CUBE(hs^k)
Figure 5

we add a level but since the diameter does not increase the average distance drops. A similar situation occurs when $s = 8$ as shown in Figure 5 .A detailed analysis is omitted due to the space but it can be seen, from the equations for the average distance, that there is a minimum in the curve. Finally, it should be noted that the performance of the BH improves as s increases. However, as pointed out earlier, the higher s, the larger the node degree and hence the higher hardware cost. Therefore, $s = 4$ seems to be a happy compromise, especially since the embedding requirements of various topological structures will be met on BH$(h, k, 4)$ as will be seen in section 6.

§5. Routing

Due to the highly recursive structure of the banyan-hypercubes and the exchange inteconnections between the levels, the routing turns out to be simple and doable in optimal time and space. The main idea is to view routing a message from a source node to a destination node as a sequence of communication steps equivalent to a sequence of changes made to the source address label to become the destination address label.

Point-to-Point Routing(One-to-One Communication)

We first outline the routing policy and the algorithm is given later. Let (L, X) be the address label of a node holding a message to be routed to the destination (L', D), where $X = x_{k-1}...x_1x_0$ and $D = d_{k-1}...d_1d_0$. Since the interconnections at all the levels are parallel, the message can be sent first to the level of the destination node using banyan edges and then to the destination itself using hypercube edges in the same level, without incurring any unnecessary communication steps. In label terms, this corresponds to increasing L (if $L' > L$) or decreasing it (if $L' < L$) by one every time the message is sent to a new level, and then X is transformed to D bit by bit using hypercube routing. On the way from level L to level L' in BH$(j, j + h - 1, k, s)$ the s-ary digits of X that differ from the corresponding digits of D can be changed (using e^a_L connections) to agree with those of D if their positions lie between $L\%k$ and $(L'\%k) - 1$ if $L' > L$ or between $L'\%k$ and $(L\%k) - 1$ if $L' < L$. This routing policy can be shown to route along shortest paths. The time spent at every node is constant and the information needed is just the destination address, and hence this algorithm is optimal.

Consequently, the following routing algorithm which implements the above routing policy routes messages from sources to destinations along shortest paths when executed by every node. The time spent at every node is constant and the information needed is just the destination address. Hence, this algorithm is optimal.

Routing Algorithm for the Banyan-Hypercube

Route from Current node X with address $(L, x_{k-1} \dots x_1 x_0)$ to Destination Node D with address $(L', d_{k-1} \dots d_1 d_0)$

```
Algorithm Route(X,D);
begin
  Case:
    L' > L: /* go up */
      Send to (L + 1, x_{k-1}...x_{L+1} d_L x_{L-1}...x_0)
    L' < L: /* go down */
      Send to (L - 1, x_{k-1}...x_L d_{L-1} x_{L-2}...x_0)
    L' = L: /* horizontally */
     Route in Hypercube
  End Case
end
```

Broadcasting (One-to-all Communication)

The policy for broadcasting, from a source node with address (L, X) to all nodes, combines the hypercube broadcasting algorithms and broadcasting using banyan edges. We discuss the idea of the routing policy in this paper and the complete algorithm is omitted. Let (L, X) be at level j in a $BH(i, i + h - 1, k, s)$, i.e., $L = j$. Note that sending a message to all nodes at the same level is the case of broadcasting in a s^k node hypercube and this can be done in $\log s^k$ steps. Using the s banyan edges it can send the message to s nodes at level $j + 1$ and to s nodes at level $j - 1$. The s nodes at level $j + 1$ $(j - 1)$ belong to s distinct hypercubes of dimension $\log s^{k-1}$ and the message can be simulataneously broadcast in each subcube in $\log s^{k-1}$ steps. By using the banyan edges and going up (or down) h' levels the message can be sent to $s^{h'}$ nodes at level $j + h'$ which in turn can be broadcast in a hypercube of dimension $\log s^{k-h'}$. When x has sent the message k levels up (or down) to level $j + k$ then all nodes at level $j + k$ have recieved the message and they can forward the message to all nodes at the level above (or below) using the vertical edges (among the banyan edges). Thus broadcasting from a node at level j to level $j + h$ when $h > k$ takes h steps. The worst case is clearly when we need to send a message from level 0 to level $h - 1$ in a h level banyan hypercube. This takes h steps, and in general the time required is the maximum of h and the time to broadcast in the hypercube at level j; i.e., $\max\{h, \log s^k\}$. This algorithm is optimal since it takes time in the order of the shortest distance from the source to the farthest node.

§6. Embedding on the Banyan-Hypercubes

An embedding is a one-to-one mapping of the guest graph (corresponding to the algorithm) on the host graph (corresponding to the network graph). The efficiency of an embedding is defined in terms of the dilation cost. The dilation cost of an embedding of a guest graph on a host graph is the largest distance between nodes in the host graph which correspond to neighboring nodes in the guest graph. We have shown efficient embeddings [14] of various structures on the restricted banyan-hypercube (i.e., when we restrict the number of levels to be at most $k + 1$ and only the banyan connections from the base of banyan are used). Rings and Meshes (whose sizes are powers of 2) can be embedded with unit dilation cost. Full binary trees can be embedded with dilation cost $\frac{h}{4}$ using tree embeddings on twisted cubes [5]. This dilation cost is smaller than the dilation cost 2 of the tree embedding on the hypercube for $h \leq 4$ and is equal to it for $h = 8$, which is a practical value of h for $s = 4$. By showing that any $BH(j, j+h-1, k_1, s_1)$ can be embedded in a $BH(j, j+h-1, k_2, s_2)$ with dilation cost $\log(2\lceil s_1/s_2 \rceil)$, the embeddings can be extended for any value of s.

From Theorem 3.4, any h level $BH(i, i+h-1, k, s)$ is isomorphic to any other h level $BH(j, j+h-1, k, s)$, these embeddings can be extended to the generalized banyan-hypercube networks. The following theorems, whose proofs are omitted since they follow from Theorem 3.4 and a simple extension of the embeddings provided in [14], summarize the embedding results.

We first present the following theorem which gives the dilation cost of the embedding between two banyan-hypercubes with the same number of levels h and same number of nodes at each level but with different node degrees, i.e., s. Recall from the definition of BH networks that s is always assumed to be a power of 2.

6.1 Theorem. *A $BH(j, j + h - 1, k_1, s_1)$ can be embedded on a $BH(j, j + h - 1, k_2, s_2)$, where $s_1^{k_1} = s_2^{k_2}$ with dilation cost $\log(2\lceil (s_1/s_2) \rceil)$.*

Proof. The proof is fairly straightforward and follows from the definition of BH networks and hypercubes, and is omitted in this paper.∎

From the result of the above theorem it follows that any $BH(j, j + h - 1, k_1, s_1)$ can be embedded with unit dilation cost (i.e., it is a subgraph) in a $BH(j, j + h - 1, k_2, s_2)$ when $s_1 < s_2$ (and number of nodes in a level are equal, i.e., $s_1^{k_1} = s_2^{k_2}$. Also when $s_1 = 4$ and $s_2 = 2$ it is seen that the dilation cost is $\log(2 \cdot 4/2) = 2$.

6.2 Theorem. *A $BH(j, j + h - 1, k, s)$ embeds a ring of all the nodes with dilation cost 1.*

6.3 Theorem. *Any m dimensional mesh $h \times n_2 \times \dots \times n_m$, where each $n_i = 2^j$, $2 \leq i \leq m, j \geq 0$, can be embedded in $BH(j, j + h - 1, k, s)$ with unit dilation cost.*

6.4 Theorem. *For any power of two $h \geq 4$, $BH(j, j + h - 1, k, s)$ embeds a full binary tree of $hs^k - 1$ nodes with dilation cost $\frac{h}{4}$.*

Pyramids are very useful data structures in image processing and scientific multigrid computations [2] [13]. Therefore, it is desirable to have an efficient embedding of pyramids on the underlying network of a given architecture. A pyramid with an $n \times n$ base, where n is a power of two, has $\log n + 1$ levels of nodes, where the base level (called level 0) is a $n \times n$ mesh, and each level i is a $\frac{n}{2^i} \times \frac{n}{2^i}$ mesh such that each node is the parent of four nodes in the level below. The top level has only one node called the apex. The embedding of pyramids in hypercubes have been shown to have dilation cost 2 and cannot be done in dilation 1 [2] [13]. We showed [14] that BH$(0, k, k, 4)$ networks have pyramids as subgraphs and can thus embed them with dilation 1. Application of Theorems 6.1 and 3.4 lead to the following conclusion.

6.5 Theorem. *A pyramid with a $2^k \times 2^k$ base is a subgraph of $BH(j, j+k, k, s)$ where $s \geq 4$, that is, it can be embedded in it with dilation 1.*

Note that in BH$(j, j+h-1, k, 4)$, where $h < k+1$, several smaller pyramids can be embedded, one in each of the $4^{k-(h-1)}$ BH$(j, j+h-1, h-1, 4)$'s which are subgraphs of BH$(h, k, 4)$.

As a direct consequence of Theorem 6.5 and Theorem 6.1, we have the following corollary.

6.6 Corollary. *A pyramid with a $2^k \times 2^k$ base can be embedded in a $BH(j, j+k, 2k, 2)$ with dilation cost 2.*

A multiple pyramid of degree d is a graph made of d pyramids that have the same base but otherwise are disjoint. Such topological structures are useful in image processing where an image, stored at the base, has multiple objects of interests that can be processed simultaneously (by different pyramids), or different image processing tasks are to be done on the same image [9]. The above embedding of pyramids can be extended to embed mutiple pyramids of degree 4 on BH$(k+1, k, 4)$ with dilation 1. In general, a multiple pyramid of degree s can be embedded with dilation 1 in BH$(j, j+k, k, s)$. However, no multiple pyramid with degree $d > s$ can be embedded with dilation 1 because in such a multiple pyramid each node at the base has d parents while in BH$(j, j+k, k, s)$ each base node has only s adjacent nodes in the level above.

§7. Conclusion

This paper extended a new family of networks, called the banyan-hypercubes. These extended networks were shown to incorporate many hypercube features in routing, partitioning, and embedding of rings, meshes and trees. They were also shown to offer an improvement over hypercubes in diameter, average distance, embedding of hierarchical structures, extendability cost, and flexibility in partitioning. A detailed study on partitioning the BH networks is discussed in [15].

A useful way of viewing banyan-hypercubes is as a method of connecting various hypercubes with rich inter-level interconnections of the banyan without limitation on the number of levels, and with a small increase in degree. The extension to allow an unlimited number of levels offers better extendability, and hence more affordability and flexibility in system growth.

References

[1] S.N. Bhatt and C.F. Ipsen, "How to Embed Trees in Hypercubes," *Research Report YALEU / DCS / RR-443*, Dept. of Computer Science, Yale University, December 1985.

[2] T.F. Chan and Y. Saad, "Multigrid Algorithms on the Hypercube Multiprocessor," *IEEE Transactions on Computers*, Vol.C-35,No.11, pp.969–977, November 1986.

[3] M. Chen and K.G. Shin, "Processor Allocation in an N-Cube Multiprocessor Using Gray Codes," *IEEE Transactions on Computers*, Vol.C-36, No.12, pp.1396 – 1407, December 1987.

[4] D. DeGroot, "Mapping Communication Structures into SW-Banyan Networks," Ph.D. Dissertation, University of Texas, Austin, 1981.

[5] A. Esfahanian, L.M. Ni, and B.E. Sagan, "On Enhancing Hypercube Multiprocessors," *Proceedings of the 1988 International Conference on Parallel Processing*, pp. 86–89, August 1988.

[6] R.L. Goke, "Banyan Networks for Partitioning Multiprocessor Systems," Doctoral Dissertation, University of Florida, 1976.

[7] S.L. Johnsson, "Communication Efficient Basic Linear Algebra Computations on Hypercube Architectures," *Journal of Parallel and Distributed Computing 4*, pp. 133–172, 1987.

[8] D.H. Lawrie, "Access and Alignment of Data in an Array Processor," *IEEE Transactions Computers*, Vol C-24, pp. 1145–1155, Decemeber 1976.

[9] R. Miller and Q.F. Stout, "Simulating Essential Pyramids," *IEEE Transactions on Computers*, Vol. 37, No.12, pp. 1642–1648, December 1988.

[10] R.M. Jenevein and T. Mookken, "Traffic Analysis of Rectangular SW-Banyan Networks," *The 15th Annual International Symposium on Computer Architecture*, Computer Architecture News, Vol.16, No.2, May 30– June 2, 1988, pp. 333–342.

[11] C. Seitz, "The Cosmic Cube", *Communications of the ACM*, January 1985, pp. 22-33.

[12] H.J. Siegel, "Interconnection Networks for Large-Scale Parallel Processing," Lexington Books, 1985.

[13] Q.F. Stout, "Hypercubes and Pyramids," in *Pyramidal Systems for Computer Vision*, edited by V. Cantoni and S. Levialdi, Springer-Verlag, Berlin, 1986.

[14] A. Youssef and B. Narahari, "Banyan-Hypercube Networks", *IEEE Transactions on Parallel and Distributed Computing*, Vol.1 No.2, April 1990.

[15] A. Bellaachia and A. Youssef, "Partitioning on the Banyan-Hypercube Networks", *Proc. of 3rd Symp. on Frontiers of Massively Parallel Computation* October 1990.

[16] A. Youssef, "Product Networks: A Unified Theory of Fixed Interconnection Networks", In Preparation.