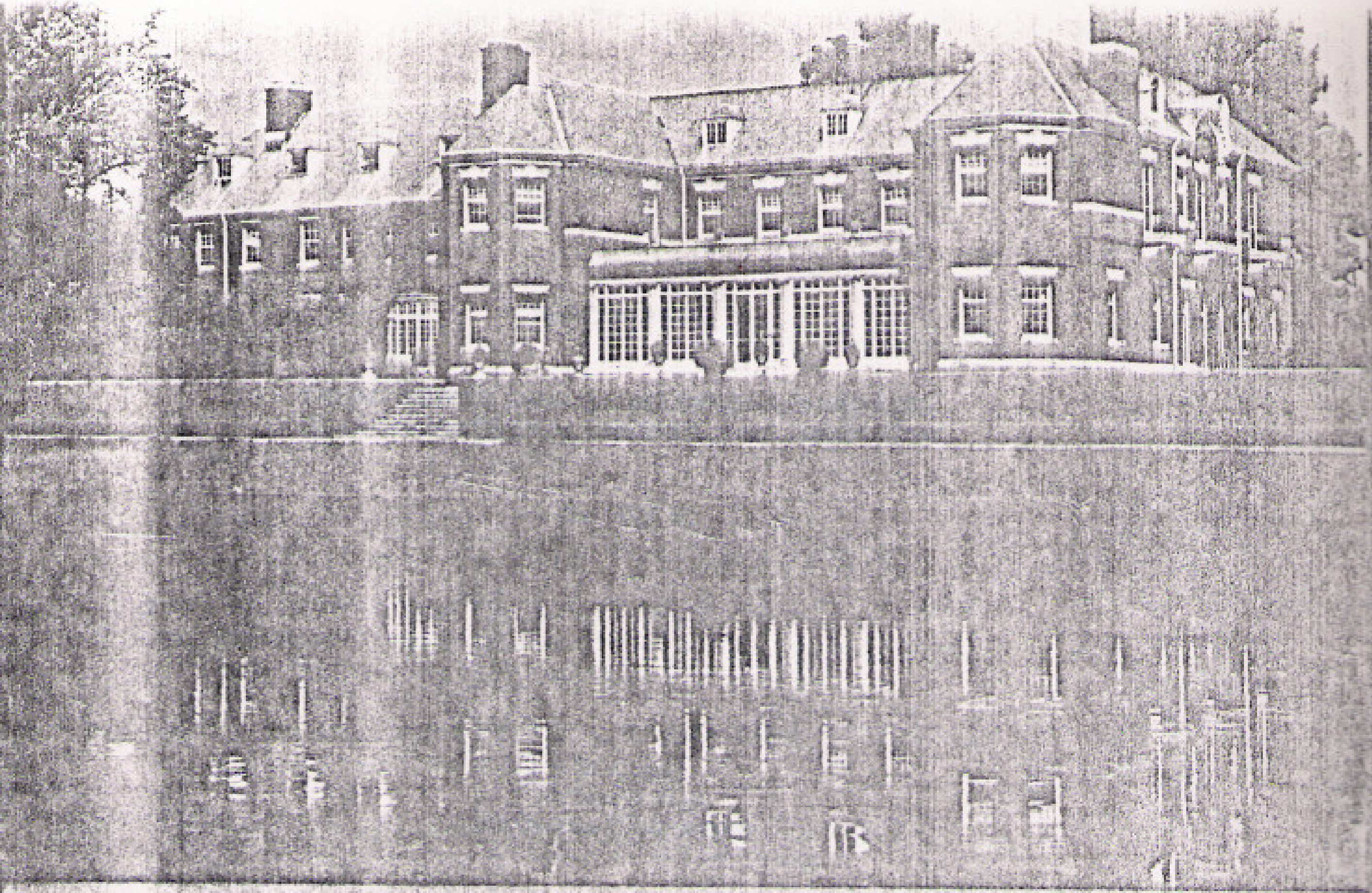


#2

Proceedings

TWENTY-FIFTH ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

September 30-October 2, 1987

Allerton House, Monticello, Illinois
Sponsored by the
Coordinated Science Laboratory and the
Department of Electrical and Computer Engineering of the
University of Illinois at Urbana-Champaign

ON THE PERFORMANCE OF SW-BANYAN NETWORKS

M.H. PEYRAVI
Department of Mathematical Sciences
Kent State University
Kent, Ohio 44242

M.R. MEYBODI
Department of Computer Science
Ohio University
Athens, Ohio 45701

ABSTRACT

A class of interconnection networks is studied as a communication medium in multi-processing systems. These networks, termed expanding and contracting SW-banyan networks, recently introduced by DeGroot. In this paper a new method for computing the number of permutations realized by this class is given. It is shown that the known method for computing this number can not be used for any SW-banyan network. Finally, we study different classes of expanding and contracting NxN SW-banyan networks with respect to three different performance measures: number of links, number of switches, and blockage of the network. We derive conditions on fanout and spread vectors of SW-banayan networks to minimize one or more of these quantities.

Index terms: *banyan networks, SW-banyan, blockage, combinatorial power.*

I. INTRODUCTION

A fundamental class of interconnection networks, in the context of multi- processing systems, is called banyan networks and were first proposed by Goke and Lipovski[1]. The banyan class contains a very rich and useful subclass of networks called L-stage banyan networks [1],[2]. A variety of special cases of L-stage banyan networks have received considerable attention in the design and in the literature. Large banyan networks can be synthesized from smaller ones. This is illustrated in Figure 1(a). The interconnection of these banyan networks can be represented by a switching network as in Figure 1(b). A cross-bar switch is a trivial one level banyan network. L-stage banyan networks are synthesized recursively from cross-bar switches as in Figure 1(b). Two of the most notable subclasses of an L-stage banyan are Delta networks [4],[5],[6], and SW-banyan networks. The well known networks such as Omega networks[7], Indirect Binary n-Cube networks[8], STARAN flip networks[9] and Baseline networks[10] are all special cases of Delta networks. The characteristic property of a Delta network is a simple decentralized routing algorithm based on "destination tag". The SW-banyan class is a proper subset of banyan networks that can be constructed recursively from smaller SW-banyans. Another interesting class of SW-banyan networks is *Expanding and Contracting* SW-banyan networks which have been studied in [2],[11],[12],[13].

In this paper we study the performance of expanding and contracting SW-banyan networks with respect to four parameters, namely, number of permutations realizable, number of links, number of switches and number of blockage in the network.

In section II different classes of banyan networks are described. In section III, we compute the number of permutations realized by a given L-stage expanding and contracting SW-banyan network. We show that the method due to Bhuyan and Agrawal [14] for computing the number of permutations realized is incorrect. We introduce a new method for computing this quantity. Finally, a trade off between path blockages and combinatorial power of a network is discussed. In section IV different classes of expanding and contracting NxN SW-banyan networks are analyzed with respect to different performance measures such as number of links, number of switches and blockage in the network. We find conditions on elements of the fanout vector and spread vector such that one or more of these parameters are minimized.

II. BANYAN NETWORKS

A banyan graph is a Hasse diagram of a partial ordering in which there is only one path from any base to any apex. A base is defined as any vertex with no arcs incident into it and an apex is defined as any vertex with no arcs incident out of it, and all other vertices are called intermediates. In a computer system, the set of bases may correspond to processors, the set of apexes to memory and I/O devices and the intermediates to switch nodes. One of the useful property of a banyan network is that there is one and only one path from any base to any apex (unique path network).

An L-level banyan network is a banyan graph in which the path from base to apex (or apex to base) is of length L. Therefore, in an L-level banyan, there are L+1 level of nodes and L-level of edges. An L-level banyan is an (L+1)-partite graph in which the partitions may be linearly ordered from 0 though L such that arcs exist only from i^{th} to $(i+1)^{th}$ partition. By convention, apexes(outputs) are considered to be at level 0 and bases(inputs) are at level L.

Definition 1: In a banyan network, the spread of a vertex is the out-degree of a node and the fanout of a vertex is the in-degree of a node (the direction is from base to apex).

Definition 2: If all vertices within the same level of a banyan network have identical spread and fanout value then the banyan is called uniform, Figure 2, otherwise it is called non-uniform.

In a uniform banyan network, the fanout values and the spread values may be characterized by L component vectors, $F = (f_0, f_1, \ldots, f_{L-1})$ and $S = (s_1, s_2, \ldots, s_L)$, the fanout vector and spread vector, respectively, where s_i and f_i denote the spread and fanout of a node at level i. Note that $s_i \times f_{i-1}$ is the size of the switches at level i, for $1 \le i \le L$ and $f_L = s_0 = 0$.

Definition 3: If $s_{i+1} = f_i$, $0 \le i \le L-1$, that is $S = F$ then the banyan network is called rectangular, Figure 3. If $s_{i+1} \neq f_i$, for some $0 \le i \le L-1$, then it is called non-rectangular.

Definition 4: If every component of S is equal to some constant s and every component of F is equal to some constant f then the banyan is called regular, Figure 4. otherwise it is irregular.

If $s = f$ this implies that F=S and the banyan network is both regular and rectangular which is called strongly rectangular.

SW-Banyan Networks:

Let A be the set of apexes (output terminals) and B be the set of bases (input terminals). Given $b \in B$ and $a \in A$, define $R_x(a)$, (x reachability set for apex a), to be a set of all nodes at level x, $(0 \le x \le L)$, which are reachable from apex a, and $R_x(b)$, (x reachability set for base b), to be a set of all nodes at level x, $(0 < x < L)$, which are reachable from base b. Clearly,

$$\begin{array}{ll} R_0(a) = \{a\} & \\ R_L(a) = B, & \text{the set of input terminals (bases).} \\ R_0(b) = A, & \text{the set of output terminals (apexes).} \\ R_L(b) = \{b\} & \end{array}$$

Definition 5: An L-stage uniform banyan is called SW-banyan if and only if either one or both of the following conditions is true.

$$\begin{array}{llll} 1) & R_x(a_i) = R_x(a_j) & \text{or} & R_x(a_i) \bigcap R_x(a_j) = \phi \\ 2) & R_x(b_i) = R_x(b_j) & \text{or} & R_x(b_i) \bigcap R_x(b_j) = \phi \end{array}$$

for $a_i, a_j \in A, b_i, b_j \in B$ and $0 \le x \le L$. In other words, A banyan network is SW-banyan if and only if for any two bases b_1 and b_2 (or any two apexes a_1 and a_2, their reachability sets at each level are disjoint or identical.

If an L-stage uniform banyan network satisfies either 1) or 2) but not both then it is called one-way SW-banyan network. If condition (1) is true then it is called input SW-banyan and if condition (2) is true then it is called output SW-banyan. If both 1) and 2) are true, then it is called two-way SW-banyan.

From the definition of the L-stage uniform NxM banyan networks it follows [2] that

$$N = \prod_{i=1}^{L} s_i \qquad M = \prod_{i=0}^{L-1} f_i.$$

Given this factorization for N and M, an L-stage input SW-banyan network is defined recursively as follows: The input stage (stage L) is made of N/s_L cross-bar switches, each of size $s_L * f_{L-1}$. This stage is followed by f_{L-1} blocks. Each of these blocks is an (L-1)-stage block structured SW-Banyan network and its input stage contains $N/s_L s_{L-1}$ cross-bar switches of size $s_{L-1} \times f_{L-2}$. The recursion stops at stage 1. The output terminal of switches in stage i are connected to the input terminals of blocks in stage $i+1$. Below we state some theorems and corollaries which discuss topological feature of SW-banyan networks [2].

Theorem 1: For any level i in an SW-banyan $0 \le i \le L-1$, $n_{i+1}s_{i+1} = n_i f_i$, where n_i is the number of nodes at level i.

Theorem 2: For any uniform L-level SW-banyan $n_i < n_{i+1}$ if and only if $f_i > s_{i+1}$. Furthermore, $n_i > n_{i+1}$ if and only if $f_i < s_{i+1}$, and $n_i = n_{i+1}$ if and only if $f_i > s_{i+1}$.

Theorem 3: In a rectangular SW-banyan n_i is constant and equal to $|B|$, the number of bases in the banyan, for $0 \le i \le L$.

Corollary 3.1: If $n_i = n_{i+1}$ for all i, $0 \le i \le L-1$, then $F = S$.

Theorem 4: Each base of an L-level SW-banyan reaches $s_{i+1}s_{i+2}\ldots s_L$ nodes at level i, $0 \le i \le L-1$.

Corollary 4.1: The number of apexes in a uniform SW-banyan is $s_0 s_1 s_2 \ldots s_L$

Corollary 4.2: Each node at level i, $0 \le i \le L$, reaches $s_0 s_1 s_2 \ldots s_i$ apexes.

Theorem 5: Every apex in a uniform L-level SW-banyan reaches $f_0 f_1 \ldots f_{i-1}$ nodes at level i, $0 \le i \le L$.

Corollary 5.1: The number of bases in a uniform SW-banyan is $f_0 f_1 f_2 \ldots f_L$.

Corollary 5.2: Each node at level i, $0 \le i \le L$, reaches $f_i f_{i+1} \ldots f_L$ bases.

Theorem 6: In a SW-banyan network, the number of apexes equals the number of bases if and only if $f_0 f_1 \ldots f_{L-1} = s_1 s_2 \ldots s_L$.

Expanding and Contracting NxN SW-banyan Networks:

Consider a non-prime integer $N > 2$. Let $N = n_1 n_2 \ldots n_L$ be a factorization of N, where all factors are not necessarily distinct. Let $n = (n_1, n_2, \ldots, n_L)$ be a vector of the L factors. Let $F = (f_0, f_1, \ldots, f_{L-1})$ and $S = (s_1, s_2, \ldots, s_L)$ be two not necessarily distinct permutations of the components of n. Given S and F, consider an L-stage SW-banyan network where the i^{th} stage is made up of complete cross-bar switches of size $s_i \times f_{i-1}$ [1,2]. Clearly, the number of switches in stage 1 is N/s_L and in stage i is given by $N s_1 s_2 \ldots s_{i-1}/f_1 f_2 \ldots f_i$ for $1 \le i \le L$. Let r be the number of distinct permutations of the L factors of N, then there are r^2 such networks. By exhausting the set of all factors of N, a class of all SW-banyan networks with N inputs and N outputs can be obtained.

Example: If $N = 2 \times 2 \times 4$ then there are three distinct permutations, $(2,2,4)$, $(4,2,2)$ and $(2,4,2)$ and hence there are nine networks. For $N = 16$ there are 16 different SW-banyan networks including the 16×16 cross-bar switch which are illustrated in Table 1. This class of networks includes all the well-known Delta networks such as Omega networks, indirect binary networks as well as the class of expanding and contracting SW-banyan networks [2, 5].

If $F = S$ that is $s_i = f_{i-1}$ for $1 \le i \le L$, then each stage is made of square cross-bar switches. If $S \ne F$, that is there exist $j, 1 \le j \le L$ such that $s_j \ne f_j$, then there is at least one stage made of rectangular cross-bar switch. For example, consider $F = (2,2,4)$ and $S = (2,4,2)$, then stage one is made of 2×2 switches and the second the third stages are made of switches of size 2×4 and 4×2, respectively, Figure 5. If $S \ne F$, the network is called Expanding-Contracting SW-banyan Network [2].

It is shown through examples [2] that Expanding and Contracting SW-banyan networks have less path blockage, i.e., when one base is connected to one apex by a direct communication path, no other new base-apex connection can be made if the new connection requires a link in use by the first connection, the second connection is blocked by the first one. As a result less blockage could imply more permutations to be realized by the network. Table 1 illustrates the blockage for the set of all 16x16 SW-banyan networks.

III. NUMBER OF PERMUTATIONS REALIZED BY SW-BANYAN NETWORKS

While there is a reason to believe that networks with less blockage may realize more permutations, computing the latter quantity for the expanding and contracting SW-banyan networks is by no means trivial and in fact is equivalent to a class of enumeration problems which are of independent interest. Two different cases will be analyzed, one for $S = F$ and the other for $S \ne F$.

Case A: If $F = S$, then all the switches of each stage have the same size, i.e., $(s_i \times s_i), 1 \le i \le L$, where L is the number of stages of the network. Each switch is capable of realizing $s_i!$, then the number of permutations $\alpha(N)$ realized by an $N \times N$ SW-banyan network is given by:

$$\alpha(N) = \prod_{i=1}^{L} (s_i!)^{m_i}$$

where m_i is the number of switches in stage i. The instances 2, 3, 5, 8, 12, 15 and 16 in Table 1 correspond to this case.

Case B: If $F \ne S$, then it is not always necessary that $\alpha(N) \ge 0$. For example if $F = (8,2)$ and $S = (2,8)$, then stage 1 is made of two copies of 2×8 switches and stage 2 contains two copies of 8×2 switches. Figure 6 illustrate the network. Since the output of stage one contains only four output ports, 12 of the 16 inputs are blocked by stage 1 and hence $\alpha(N) \ge 0$ Consequently, the following theorem is immediate.

Theorem 7: Let the j^{th} stage of an L-stage $N \times N$ SW-banyan network is made of cross-bar switches of size $s_j \times f_{j-1}$, $1 \le j \le L$. Then a necessary and sufficient condition for $\alpha(N) \ge 0$ is that $m_j s_j \ge N$, where m_j is the number of switches in stage j.

The above condition is equivalent to the following statement:

$$s_1 s_2 \ldots s_j \ge f_1 f_2 \ldots f_j$$

It is easy to see that $\alpha(N) = 0$ for instances 6,9,13 and 14. The remaining four instances namely 4,7,10 and 11 satisfy the condition of the above theorem. The computation of $\alpha(N)$ for later instances of the SW-banyan network gives rise to an interesting class of enumeration problems. We illustrate using the example in Figure 5, where $F = (2,2,4)$ and $S = (2,4,2)$. Consider the first block which consists of switches A_1 to A_4 and B_1 to B_4. The structure of the network induces the following natural constraints. Each of the switches $A_i (1 \le i \le 4)$ has four output ports, since there are only two input ports to A_i, only two out of four output ports can actually carry the two inputs. Likewise, each switch $B_j (1 \le j \le 4)$ through the link permutation can receive four inputs, since it has only two output links. To avoid the blockage within the switches $B_j (1 \le j \le 4)$, it is required that B_j receives exactly two inputs from any two of the four input ports. A similar requirement holds for the switches in the second block. In other words, each switch in stage 2 and 3 receives exactly two inputs and acts as though they were 2×2 switches. Thus, within the first block, the number of permutations realized by the network crucially depend on the number of ways in which each of the switches A_i sends its two outputs to the switches B_j under the constraint that each B_j can receive exactly two inputs. This number, as it can be seen, is the same as the number of distinct 4×4 matrices where the elements of the matrices belong to the set $\{0, 1\}$ and the sum of each row and the sum of each column is equal to 2.

The following matrix corresponds to the first (second) block,

$$\begin{array}{ccccc} & A_1 & A_2 & A_3 & A_4 \\ B_1 & 1 & 0 & 1 & 0 \\ B_1 & 0 & 1 & 0 & 1 \\ B_1 & 1 & 1 & 0 & 0 \\ B_1 & 0 & 0 & 1 & 1 \end{array}$$

where the i^{th} row corresponds to the i^{th} switch B_i and the j^{th} column corresponds to the j^{th} switch A_j. The $(i, j)^{th}$ element is 1 if and only if B_i receives an input from A_j.

Let $f(a, A, b)$ be the number of distinct matrices whose elements belong to the set A and every row and column sum is b. For the above matrix, this number is $f(4, \{0, 1\}, 2)$. It can be easily seen that this number is 90. Notice the first block and the second block of Figure 5(b) can be controlled independently. With this constraint, each of the 4×2 and 2×4 switches acts as a 2×2 switch only. Thus each of the switches at stage 2, and 3 realizes two permutations. There are 24 switches involved in the network; therefore the number of permutations realized by the network is:

$$\alpha(N) = 2^{24} f^2(4, \{0, 1\}, 2).$$

Referring to Table 1, the network corresponding to the instance 4 and 13 give rise to the computation of $f(8, 0, 1, 2)$ and $f(4, 0, 1, 4)$, respectively. These later enumeration problems are of intrinsic interest.

Table 1 illustrates $\alpha(N)$, path blockage [2] and the cost of each network. The cost of a network is measured in terms of the total number of switching elements.

It follows that less blockage means more combinatorial power measured in terms of the number of permutations realized. Further, among the networks with the same cost (instances 2, 8, 12, 15, and 16) instance 2 has the minimum number of blockages and maximum value of $\alpha(N)$ and instance 16 has maximum number of blockage and minimum value for $\alpha(N)$. Another fact that emerges from this analysis is that among the networks with the same cost, those with the larger switches have less blockage and larger value for $\alpha(N)$. Another immediate consequence of our analysis is that the formula for the number of permutations given in Theorem 3 of [14] is incorrect, as the following example illustrates.

Example: Consider the network of block A of Figure 5(b). It consists of two stages of 2×4 and 4×2 switches, respectively. Bhuyan and Agrawal [14] showed that the number of permutations realized by a generalized interconnection network can be obtained by

$$P = \prod_{i=1}^{r} S_i^{k^i}$$

where k_i is the number of switches at the i^{th} stage and

$$S_i = \binom{n_i}{m_i} m_i \text{ for } m_i \leq n_i$$

$$S_i = \binom{m_i}{n_i} n_i \text{ for } m_i \geq n_i$$

where S_i is the number of permutations achievable by an $m_i \times n_i$ cross-bar switch at the i^{th} stage and r is the number of stages of the network.

For the network of block A of Figure 5(b), $m_1 = 2, n_1 = 4, m_2 = 4, n_2 = 2, r = 2$ and $k^i = 4$, for $i = 1, 2$ then

$$S_1 = \binom{4}{2} 2! = 12$$

$$S_2 = \binom{4}{2} 2! = 12$$

$$P = \prod_{i=1}^{r} S_1^{k_i} = (12)^4 (12)^4 = 12^8$$

while this number is computed to be 90 according to the method discussed earlier.

IV. OPTIMAL EXPANDING AND CONTRACTING NxN SW-BANYAN NETWORKS

In this section we study different classes of Expanding and Contracting $N \times N$ SW-banyan with respect to three different performance measures: number of links, number of switches and blockage in the network. We find conditions on elements of vector F and S such that one or more of these parameters are minimized. We use function $BP(i) = (a_i - 1)(b_i - 1)$ suggested in [2] to measure the amount of blockage in the network. In this equation, b_i is the number of bases reachable by a node at level i and a_i is the number of apexes reachable by a node at level i. This function counts the number of blocked paths that pass through a busy node at level i. The sum of all $BP(i)$ for $1 \leq i \leq L-1$ which is denoted by TDP can be used to measure the blockage in the network. Both $BP(0)$ and $BP(L)$ are defined to be zero. In the above sum, many blocked paths would be counted more than once and so they would not give the total number of blocked paths generated by a single active path. Function $BL(i) = (a_i - a_{i-1})(b_i - 1)$ avoids multiple blocked paths that are encountered. This function counts the number of additional blocked paths that are encountered as a communication path is followed from an apex down to a base. The sum $TBL = \sum_{i=0}^{L-1} BL(i)$ gives the total blockage created by a single connection.

Theorem 8: Total number of nodes in an SW-banyan graph with $F = (f_0, f_1, \ldots, f_{L-1})$ and $S = (s_1, s_2, \ldots, s_L)$ is

$$TN = n_0 \left(1 + \sum_{i=0}^{L-1} \frac{\prod_{j=0}^{i} f_j}{\prod_{j=1}^{i+1} S_j} \right)$$

Theorem 9: Total number of edges in an SW-banyan graph with $F = (f_0, f_1, \ldots, f_{L-1})$ and $S = (s_1, s_2, \ldots, s_L)$ is

$$TE = n_0 f_0 \left(1 + \sum_{i=0}^{L-1} \frac{\prod_{j=1}^{i+1} f_j}{\prod_{j=1}^{i+1} S_j} \right)$$

Theorem 10: Total number of switches in an SW-banyan network corresponding to the SW-banyan graph with $F = (f_0, f_1, \ldots, f_{L-1})$ and $S = (s_1, s_2, \ldots, s_L)$ is

$$TS = n_0 \sum_{i=0}^{L-1} \frac{\prod_{j=0}^{i-1} f_j}{\prod_{j=1}^{i+1} S_{j+1}}$$

Theorem 11: Total number of links in an SW-banyan network corresponding to the SW-banyan graph with $F = (f_0, f_1, \ldots, f_{L-1})$ and $S = (s_1, s_2, \ldots, s_L)$ is

$$TL = n_0 \left(\sum_{i=0}^{L-2} \frac{\prod_{j=0}^{i} f_j}{\prod_{j=0}^{i} S_{j+1}} \right)$$

Theorem 12: The amount of blockage in an SW-banyan network is inversely proportional to the number of switches

$$TBL = \frac{n_0}{TS} \left(\sum_{i=0}^{L-1} \left(\frac{s_0}{s_{i+1}} - \frac{s_0}{s_i s_{i+1}} \right) + \sum_{i=0}^{L-1} \prod_{j=0}^{i} s_{i+1}(1 - s_i) \right)$$

Class 1: Let $H_{F,S,N,L}$ denotes the set of all $N \times N$ L-level SW-banyan networks which can be derived by permuting the components of F or S (not both). Clearly, $N = \prod_{i=1}^{L} s_i = \prod_{i=0}^{L-1} f_i$. Figure 2 illustrates an element of set H where $F = (2, 2, 4)$ and $S = (2, 2, 4)$. Consider Figure 5, this SW-banayn is obtained by exchanging the second and the last component of vector S. Let $h_{F_i,S_i,N,L}$ where $F_i = (f_{i0}, f_{i1}, \ldots, f_{i(L-1)})$ and $S_i = (s_{i1}, s_{i2}, \ldots, s_{iL})$ denote the i^{th} member of set H. We state the following theorems for class H.

Theorem 19: Set $H'_{F,S,N,L}$ contains $\frac{L!}{\prod_{i=1}^{L}(g_i)!}$ rectangular SW- banyan networks.

Theorem 20: A network belonging to set $H'_{F,S,N,L}$ has minimum TS and TL if and only if $f_0 < f_1 < \ldots < f_{L-1}$ and $s_1 > s_2 > \ldots > s_L$.

Theorem 21: A network belonging to set $H'_{F,S,N,L}$ has minimum TBP if and only if $f_0 > f_1 > \ldots > f_{L-1}$ and $s_1 < s_2 < \ldots < s_L$.

V. CONCLUSION

Different classes of Expanding and Contracting NxN SW-banyan have been studied with respect to the number of permutations realized, number of switches involved, number of arcs, number of nodes and blockage in the network. The main result of this report consists in computing the number of permutations realized by the SW-banyan networks. It is shown that the known method for computing this number is incorrect and the correct method is given. This method gives rise to an interesting class of enumeration problems. Necessary and sufficient conditions are given to minimize one or more of these parameters.

TABLE 1

PROPERTIES OF SW-BANYAN NETWORKS WITH N=16.

Instances	F	S	Number of Switching Elements	Path Blockage	$\alpha(N)$, Number of Permutation Realized
1	(16)	(16)	256	0	16!
2	(4,4)	(4,4)	128	9	$2^{24} \times 3^8$
3	(2,8)	(2,8)	160	7	$(8!)^2 \times (2!)^8 = (2^{22} \times 992, 25$
4	(2,8)	(8,2)	256	1	$2^{16} \times f(8, \{0,1\}, 2), = 187530840$
5	(8,2)	(8,2)	160	7	$(8!)^2 \times (2!)^8 = 2^{22} \times 992, 25$
6	(8,2)	(2,8)	64	49	0
7	(2,4,2)	(4,2,2)	160	9	$2^{24} \times f^2$ $f(4, \{0,1\}, 2), = 90$
8	(2,4,2)	(2,4,2)	128	13	$2^{16} \times 4! = 2^{28} \times 3^4$
9	(2,4,2)	(2,2,4)	96	25	0
10	(2,2,4)	(2,4,2)	160	9	$2^{24} \times f^2$ $f(4, \{0,1\}, 2), = 90$
11	(2,2,4)	(4,2,2)	192	5	$2^{16} \times f(4, \{0,2,2\}, 4)$
12	(2,2,4)	(2,2,4)	128	13	$2^{28} \times 3^4$
13	(4,2,2)	(2,2,4)	80	33	0
14	(4,2,2)	(2,4,2)	96	25	0
15	(4,2,2)	(4,2,2)	128	13	$2^{28} \times 3^4$
16	(2,2,2,2)	(2,2,2,2)	128	17	2^{32}

REFERENCES

[1] Goke, L. R. and Lipovski, G. J., 1973, "Banyan Networks for Partitioning Multiprocessor Systems," *Proceedings of the First Annual Symposium on Computer Architecture*, pp. 21-28.

[2] DeGroot, D., 1983, "Expanding and Contracting SW-banyan Networks," *Proceedings of the International Conference on Parallel Processing*, pp. 771-780.

[3] Goke, L. R., 1976, "Banyan Networks for Multiprocessor Systems," *Ph.D. Dissertation, University of Florida.*

[4] Dias, D. M., 1981, "Analysis and Simulation of Buffered Delta Networks," *IEEE Transactions on Computers*, Vol. 30, pp. 273-282.

[5] Kruskel, C. P. and Snir, M., 1982, "Some Results on Multi-stage Interconnection Networks for Multiprocessors," *Proceedings of the Princeton Conference on Information Sciences and Systems*, pp. 305-310.

[6] Patel, J., 1981, "Performance of Processor Memory Interconnections for Multiprocessors," *IEEE Transactions on Computers*, Vol. 30, pp. 252-264.

[7] Lawrie, D. H., 1975, "Access and Alignment of Data in an Array Processor," *IEEE Transactions on Computers*, Vol. 24, pp. 1145-1155.

[8] Pease, M. C., 1977, "The Indirect Binary n-Cube Microprocessor Array," *IEEE Transactions on Computers*, Vol. 26, pp. 250-265.

[9] Batcher, K. E., 1976, "The Flip Network in STARAN," *Proceedings of the 1976 International Conference on Parallel Processing*, pp. R65-71.

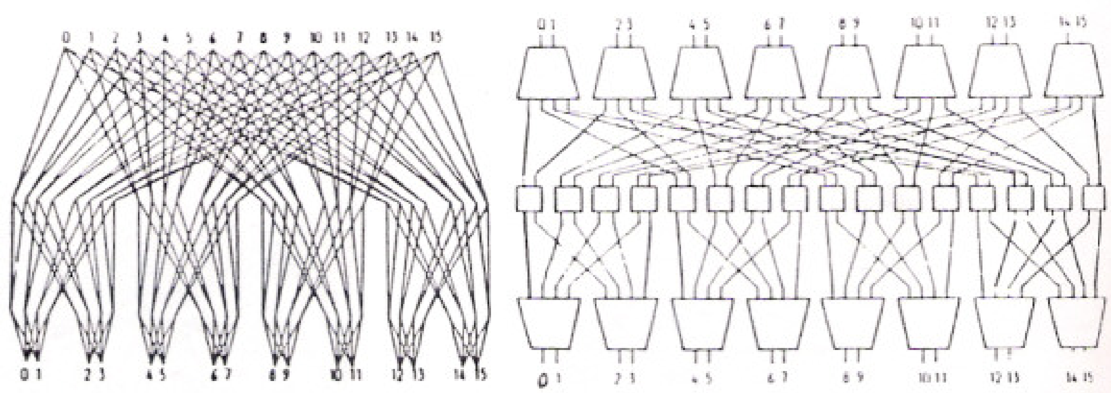
[10] Wu, C. and Feng, T., 1980, "On a Class of Multistage Interconnection Networks," *IEEE Transactions on Computers*, Vol. 29, pp. 694-702.

[11] Meybodi, M. R., 1985, "Optimal Expanding and Contracting nxn SW-banyan Networks," *Proceedings of the Sixteenth Annual Pittsburgh Conference*, pp. 633-638.

[12] Peyravi, M. H., 1985, "Study of Interconnection Networks," *Ph.D. Dissertation, University of Oklahoma, Norman, Oklahoma.*

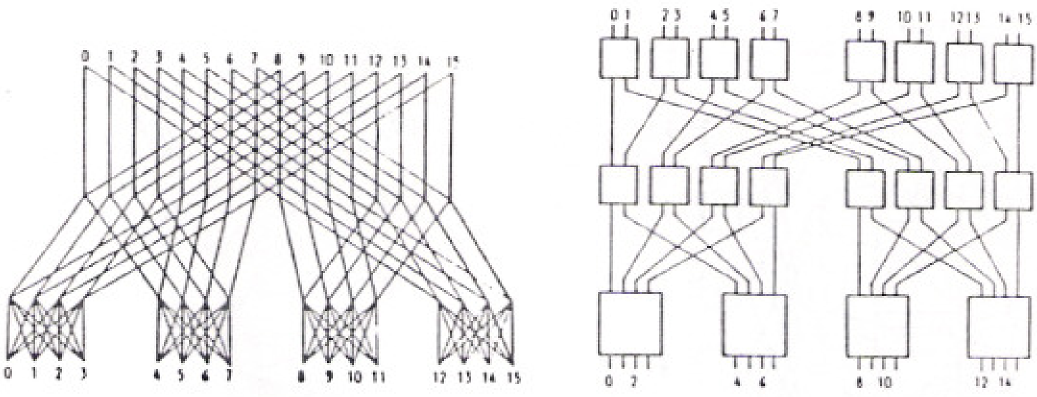
[13] Kothari, S., Lakshmivarahan, S., and Peyravi, M. H., 1984, "Computing the Number of Permutations Realizable by a Class of SW-banyan Networks," *Proceedings of the 1984 ACM Regional Conference, Austin, Texas.*

[14] Bhuyan, L. N. and Agrawal D. P., 1983, "Design and Performance of Generalized Interconnection Networks," *IEEE Transactions on Computers*, Vol. 32, pp. 1081-1090.



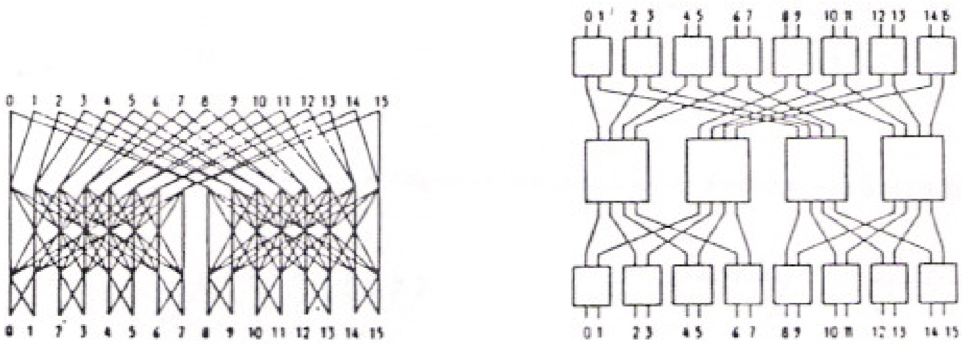
a) b)

Fig. 1. An example of L-level banyan. F=(2,2,4) and S=(4,2,2).



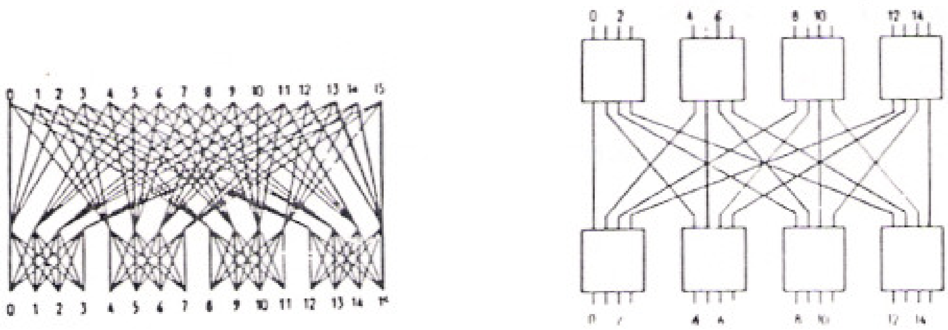
a) b)

Fig. 2. An example of uniform banyan. F=(2,2,4) and S=(2,2,4)



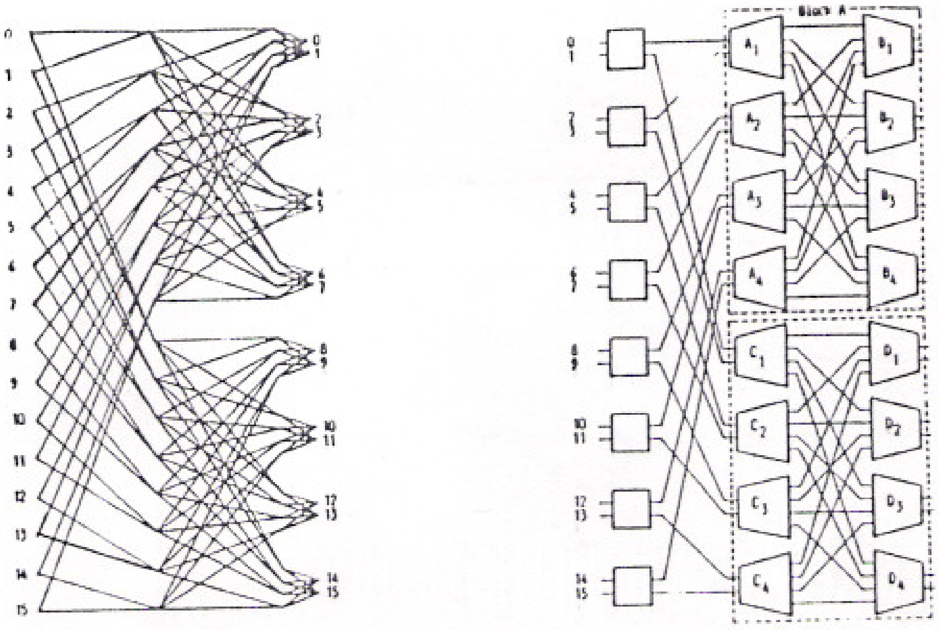
a) b)

Fig. 3. An example of rectangular banyan. F=(2,4,2) and S=(2,4,2)



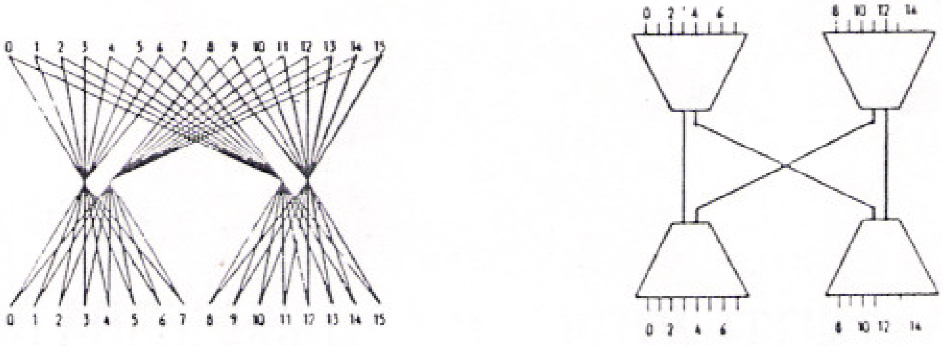
a) b)

Fig. 4. An example of regular banyan. F=(4,4) and S=(4,4)



a) b)

Fig. 5. An example of expanding and contracting banyan. F=(2,2,4) and S=(2,4,2)



a) b)

Fig. 6. An example of SW-banyan which has $\alpha(N)=0$. F=(8,2) and S=(2,8)