MODELING AND SIMULATION

Volume 16
Part 2: Control, Robotics and Automation

Proceedings of the Sixteenth Annual Pittsburgh Conference
Held April 25-26, 1985
University of Pittsburgh School of Engineering

OPTIMAL EXPANDING AND CONTRACTING nxn

SW-BANYAN NETWORKS

M. R. Meybodi
Computer Science Department
Western Michigan University
Kalamazoo, Michigan 49008

ABSTRACT

In this report we study different classes of Expanding and Contracting nxn SW-banyan Networks
with respect to three different performance measures: number of arcs, number of switches and
blockage in the network. We derive conditions on elements of fanout and spread vectors to
minimize one or more of these quantities.

INTRODUCTION

Switching Networks are important in the area of computer architecture and communication.
Most parallel distributed architectures require some type of interconnection network to link
the various elements of the computer system. Many interconnection networks have been pro-
posed in the literature. In this report we study a class of interconnection networks called
Expanding and Contracting SW-banyan Networks.

The topology and a detailed description of the operation of a switching system using banyan
network were described by Lipovskii [1] without using the name Banyan. Goke and Lipovskii
[2], Lipovskii and Tripathi [3], Tripathi and Lipovskii [4] have given more general discus-
sions of banyan networks.

An interesting class of banyan networks is the SW-banyan wihich is being implemented as part
of the TRAC system at the University of Texas at Austin. SW-banyans is a proper subset of
banyans that can be recursively constructed from SW-banyans. A procedure for constructing on
SW-banyan network will be given later in this report. Expanding and contracting Multistage
SW-banyan were discussed for the first time in [6]. In this report we study different per-
formance measures: number of arcs in the network, number of nodes (switches), and static
blockage in the network. Conditions on elements of fanout and spread vectors are derived to
minimize one or more of these quentities.

BANYAN NETWORK

A banyan is a Hasse diagram of a partial ordering in which there is one and only one path
from any base to any apex. A base is defined as any vertex with no arcs incident into it, 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.

An L-Level banyan is a banyan in which every path from any base to any apex is of length L.
An L-level banyan is a (L+1) - partite graph in which the partitions may be linearly ordered
from 0 through L such that arcs exit only from i^{th} partition to $(i + 1)^{st}$ partition. By
convention, apexes are at level 0 and bases at level L.

In a Banyan the number of arcs exiting upwards the vertex is called spread of the vertex and
number of edges exiting below the vertex is called fanout of the vertex. A banyan is called
uniform if within each level all vertices are alike in that they have the same number of arcs
incident out from it and the same number of arcs incident into it. The arcs incident out
from the vertices of a uniform banyan are characterized by an L-component vector F called
fanout vector. Similarly, the arcs incident in are characterized by an L-component spread
vector S. We let f_i and s_i denote the fanout and spread of every node at level i, respec-
tively. We define both S_0 and f_L to be 1. We also use n_i to denote the number of nodes at

level i. The number of apexes is n_o and the number of bases is n_L.

A regular banyan is a special case of Uniform banyan, in which all vertices throughout the the network are alike. That is, the spread of every vertex except the base is equal to some constant s and fanout of every vertex except the apex are equal to a constant f. That, $f_i = f$ for all $i \in \{0, 1, \ldots, L-1\}$ and $s_i = s$ for all $i \in \{1, 2, \ldots, L\}$ where L is the number of levels in the banyan network. When S=F the banyan has the same number of nodes (switches) in each level and is called rectangular. Non-rectangular banyans have $F \neq S$. A banyan is called strongly rectangular when $F = S$. Examples of different types of banyan network are given in figure 1.

SW-BANYAN NETWORK

A banyan is an SW-banyan if and only if for any two bases b and c, or any two apexes d and e. The set of all nodes in level i that can be reached by directed path from base b and c or apex d and e are either disjoint or identical for $0 \leq i \leq L$. SW-banyans are suitable for partitioning and connection network. The Texas Reconfigurable Array Computer (TRAC) uses a 4-level regular SW-banyan with spread and fanout of 2 and 3, respectively, as interconnection network. This network connects 16 processors on one side to 81 memory and I/O modules on the other side. From the definition of Uniform Banyan, in a Uniform SW-Banyan all nodes in a given level have the same fanout values and some spread values.

Below we state some theorems and corollaries that describe topological features of SW-banyan [6].

Theorem 1: For any level i in an SW-banyan, $0 \leq i \leq L-1$, $n_{i+1}s_{i+1} = n_i f_i$.

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 $0 \leq i \leq L$.

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

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

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

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

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

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

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

Theorem 6: In a uniform SW-banyan, 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 NETWORK

In a rectangular nxn SW-banyan there are n switches (nodes) in every stage of the network. Expanding and contracting nxn SW-banyan are nxn SW-banyan networks for which the number of switches at different stages may be different. These networks are certain type of non-rectangular nxn network. Given a rectangular nxn L-level SW-banyan with particular fanout and spread vectors F and S, it is possible to construct another nxn SW-banyan by permuting the components of F or S (or both). From Theorem 1 it is evident that derived nxn SW-banyans are not rectangular unless $f_i = s_{i+1}$ for $0 \leq i \leq L-1$. It is shown is [6] that Expanding and Contracting SW-banyan possess significantly less inherent blockage than the corresponding rectangular SW-banyans. Although they need more switches and may result in more network stages than some strongly rectangular SW-banyans. They retain the advantages of a cost function of only $o(n \log n)$.

The following recursive procedure can be used to construct SW-banyan network with given fanout and spread vectors.

Procedure to construct nxn L-level SW-banyan network with $F=(f_1, f_2, \ldots, f_{(L-1)})$ and $S=(s_1, s_2, \ldots, s_L)$.

Step 1: Let $p = \dfrac{n_o}{f_o}$ and $q = \dfrac{n_o}{s_1}$. We assume $n_1 = f_0 f_1, \ldots, f_{L-1} = s_1 s_2 \ldots s_L$

Step 2: Construct f_0 p x q (L-1)-level SW-banyan networks with $F = (f_1, f_2, ..., f_{L-1})$ and $S = (s_2, s_3, ..., s_L)$

Step 3: For i=1 to N_0 do

If (i mod $(\frac{n_0}{S_1})$ = 0) then

connect i^{th} apex of the L-level banyan to the last apex of all the pxq (L-1)-level banyan constructed in Step 2.

Else

Connect i^{th} apex of the L-level banyan to the $(i \bmod (\frac{n_0}{S_1}))^{th}$ apex of all the pxq (L-1)-level banyan constructed in Step 2.

The above procedure can be used to construct regular, irregular, rectangular, strongly rectangular, weakly rectangular nxn SW-banyans, as well as Expanding and Contracting nxn SW-banyans which is the topic of this report.

OPTIMAL EXPANDING AND CONTRACTING NXN SW-BANYAN

In this section we study different classes of Expanding and Contracting nxn SW-banyan with respect to three different performance measures: number of arcs in the network, number of nodes (switches) and blockage in the network. We try to find conditions on elements of vectors F and S such that one or more of these quantities are minimized. We use function $BP(i) = (a_i-1)(b_i-1)$ suggested in [6] to measure the amount of blockage in the network. This function counts the number of blocked paths that pass through a busy node at level i. The sum of all B_i for $1 \leq i \leq L-1$ which is denoted by DP can be used to measure the blockage in the network. Both BP(0) and BP(L) are defined to be zero. 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. In the above sum, many blocked paths would be counted more than once ans 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 counting. 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 $\sum_{i=0}^{L} BL(i)$ gives the total blockage created by a single connection.

Theorem 7: Total number of switches in a SW-banyan network with $F = (f_0, f_1, ..., f_{L-1})$ $S = (S_1, S_2, ..., S_L)$ is

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

Theorem 8: Total number of arcs in a SW-banyan network with $F = (f_0, f_1, ..., f_{L-1})$ and $S = (s_1, s_2, ..., s_L)$ is

$$TL = 2n_0 \left(f_0 + \sum_{i=0}^{L-1} \frac{\prod_{j=0}^{i+1} f_j}{\prod_{j=1}^{i+1} s_j} \right)$$

Class 1: $(H_{F, S, n, L})$

We use $H_{F, S, n, L}$ to denote the set of all nxn L-level SW-banyan which can be derived by permuting the components of F or S (or both). The third subscript of $H_{F, S, n, L}$ indicates that $\prod_{i=1}^{L} f_i = \prod_{i=1}^{L} s_i = n$ and the fourth subscript indicates that both F and S have L components. Figure 2 illustrates set H when $F = (2,2,4)$ and $S = (2,2,4)$. Consider Figure 2b, this SW-banyan is obtained by exchanging the first and the last component of vector F. We let $h_{F_i, S_i, n, L}$ where $F_i = (f_{i0}, f_{i1}, ..., f_{i(L-1)})$ and $S_i = (s_{i1}, s_{i2}, ..., s_{iL})$ to denote the ith member of set $H_{F, S, L, n}$. We state the following theorems for class H.

Theorem 9: Banyan network $h_{F_i, S_i, n, L} \in H_{F, S, n, L}$ has minimum number of switches if and only if $f_{10} < f_{11} < f_{12} < \ldots < f_{1(L-1)}$ and $s_{11} > s_{12} > s_{13} > \ldots > s_{1L}$

Theorem 10: Banyan network $h_{F_i, S_i, n, L} \in H_{F, S, n, L}$ has minimum number of arcs if and only if $f_{10} < f_{11} < f_{12} < \ldots < f_{1(L-1)}$ and $s_{11} > s_{12} > s_{13} > \ldots > s_{1L}$

Theorem 11: Banyan network $h_{F_i, S_i, n, L} \in H_{F, S, n, L}$ has minimum BP if and only if $f_{10} > f_{11} > f_{12} > \ldots > f_{1(L-1)}$ and $s_{11} < s_{12} < s_{13} < \ldots < s_{1L}$

Theorem 12: $|H_{F, S, n, L}| = \dfrac{(L!)^2}{\prod_{i=1}^{L} (g_i)! \prod_{i=1}^{L} (1_i)!}$

g_i is the number of times that f_i is repeated in F and l_i is the number of times that s_i is repeated in S. $|H|$ denotes the cardinality of set H.

Problem of finding $h_{F_i, S_i, L, n}$ for which BL is minimum can be reduced to minimizing

$$\sum_{i=0}^{L} (s_0 s_1 s_2 \ldots s_i - s_0 s_1 \ldots s_{i-1})(f_{(i+1)} f_{(i+2)} \ldots f_L - 1)$$ subject to the following

constraints: 1. $F \in \{F_1, F_2 \ldots F_x\}$ where $x = \dfrac{L!}{\prod_{i=1}^{L} (g_i)!}$ and $y = \dfrac{L!}{\prod_{i=1}^{L} (1_i)!}$

2. $S \in \{S_1, S_2 \ldots S_y\}$

Non linear programming method may be used to solve the above minimization problem.

Class 2: $(H^*_{n, L})$

Let $P = \{P_1, P_2, \ldots, P_r\}$ be a set whose elements represent different ways of factoring n into L integers. Each set P_i has L elements $P_{i1}, P_{i2}, \ldots, P_{i1}$ and the product of these elements is equal to n. Set P will be singleton if n is either a prime number or can be factored into L primes uniquely. We assume $P_{ij} = 1$ for all $1 \leq i \leq r$ and $1 \leq j \leq L$. We also assume that no member of set P can be derived by permuting the members of other members of set P. A L-level nxn SW-banyan can be obtained by choosing a pair of elements from set P and using them as fanout and spread vectors. If $H^*_{n, L}$ denotes the set of all nxn L-level SW-banyan networks for which no node has a unit fanout and spread value then we have

$$H^*_{n, L} = \bigcup_{i=1}^{r} \bigcup_{j=1}^{r} H_{F_{P_i}, S_{P_j}, n, L}$$

F_{P_i} and S_{P_j} are fanout and spread vectors that correspond to P_i and P_j, respectively.

Theorem 13: $|H^*_{n,L}| = (L!)^2 \sum_{j=1}^{r} \sum_{i=1}^{r} \dfrac{1}{\prod_{k=1}^{L} (g_{ik})! \prod_{k=1}^{L} (g_{jk})!}$

g_{ij} is the number of times that P_{ij} is repeated in set P_i and $r = |P|$.

Based on Theorem 9 we propose the following procedure for finding a L-level nxn SW-banyan network that has minimum TC and is also a member of $H^*_{n,L}$.

    Step 1: Compute set P for integer n.

    Step 2: For all the possible pairs of elements from set P do

        1. Sort the elements of the first set in the pair into increasing order
        2. Sort the elements of the second set in the pair into decreasing order.
          (Call the set of all sorted pairs X)

    Step 3: Choose a pair from set X that minimizes TC.

    Step 4: Use the first element of the pair found in step 3 as fanout vector and second element as spread vector.

A similar procedure

Class 3: $(H'_{F, S, n,}$

Let $H'_{F, S, n, L}$ deno
F or S (or both) and
the elements of vect

Theorem 14: $|H'_{F,S,L}$

g_i is the number of

Theorem 15: Set H'_{F}

Theorem 16: A netwo

    $f_0 < f_1$

Theorem 17: A netwo

    $f_0 > f_1$

CONCLUSION

Different classes of
to a number of arcs,
cient conditions are

REFERENCES

[1] Lipovoski, C. J
    Conf., May 1970

[2] Goke, L. R. and
    Systems, Proc.

[3] Lipovoski, C. J
    Proc. of 1977 I

[4] Tripathi, A. R.
    the 16th Ann. S

[5] Lawrie, D. H.
    IEEE, Vol. C-24

[6] DeGroot, D. Ex
    Parallel Proces

[7] DeGroot, D. Pa
    Conf. on Parall

[8] Broomell G. and
    Circuit Switchi

[9] Feng, T., A Sur
    pp. 12-27.

of switches if and

$.3 > \ldots > s_{iL}$

of arcs if and

$s_{13} > \ldots > s_{iL}$

and only if

$< s_{iL}$

times that s_i is

minimizing

e following

$\dfrac{L!}{\prod_{-1}^{L} (1_i)!}$

roblem.

ys of factoring n

duct of these

ber or can be
$\leq j \leq L$. We also
ther members of
nts from set P
1 nxn L-level SW-

e have

ectively.

nxn SW-banyan

sing order
asing order.

tor and second

A similar procedure can be written to find a network with minimum TS or BP.

Class 3: $(H'_{F, S, n, L})$

Let $H'_{F, S, n, L}$ denote the set of all networks that can be obtained by permuting elements of F or S (or both) and F = Perm (S) (F=Perm(S)) means that vector F can be obtained by permuting the elements of vector (S). For set $H'_{F,S,n,L}$ we can state the following theorems.

Theorem 14: $\left|H'_{F,S,L,n}\right| = \left(\dfrac{L!}{\prod_{i=1}^{L} (g_i)!} \right)^2$

g_i is the number of times that f_i is repeated in set F (or S)

Theorem 15: Set $H'_{F,S,L,n}$ contains $\dfrac{L!}{\prod_{i=1}^{L} (g_i)!}$ rectangular nxn SW-banyan networks.

Theorem 16: A network belonging to set $H'_{F,S,L,n}$ has minimum TS and TC if and only if

$$f_0 < f_1 < f_2 < \ldots < f_{L-1} \quad \text{and} \quad s_1 > s_2 > s_3 > \ldots > s_1$$

Theorem 17: A network belonging to the set $H'_{F,S,L,n}$ has minimum BP if and only if

$$f_0 > f_1 > f_2 > \ldots > f_{L-1} \quad \text{and} \quad s_1 < s_2 < s_3 < \ldots < s_L$$

CONCLUSION

Different classes of Expanding and Contracting nxn SW-banyan have been studied with respect to a number of arcs, number of switches and blockage in the network. Necessary and sufficient conditions are given to minimize one or more of these quantities.

REFERENCES

[1] Lipovoski, G. J. The Architecture of a Large Associative Processor, Proc. of AFIPS Conf., May 1970, pp. 385-396.

[2] Goke, L. R. and Lipovoski, G. J. Banyan Networks for Partitioning Multiprocessor Systems, Proc. of 1st Ann. Symp. on Computer Architecture. Dec. 1973 pp. 21-28.

[3] Lipovoski, G. J. and Tripathi, A. A reconfigurable varistructure array processor. Proc. of 1977 Int. Conf. on Parallel Processing. Aug. 1982, pp. 165-174.

[4] Tripathi, A. R. and Lipovoski, G. J. Packet Switching in Banyan Networks. Proc. of the 16th Ann. Symp. on Computer Architecture, April 1979, pp. 160-167.

[5] Lawrie, D. H. Access and Alignment of Data in an Array Processor, Trans. on Computers, IEEE, Vol. C-24, No. 12, Dec. 1975, pp. 1145-1155.

[6] DeGroot, D. Expanding and Contracting SW-Banyan Networks. Proc. of 1983 Int. Conf. on Parallel Processing Aug. 1983, pp. 19-24.

[7] DeGroot, D. Partitioning Job Structures for SW-Banyan Networks, Proc. of 1983 Int. Conf. on Parallel Processing Aut. 1983, pp. 106-113.

[8] Broomell G. and Heath J. R. Classification Categories and Historical Development of Circuit Switching Topologies, Computing Surveys, Vol. 15, No. 2, June 1983, pp. 95-133.

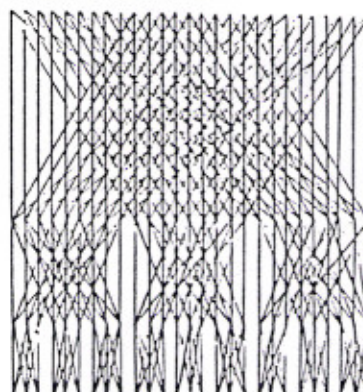[9] Feng, T., A Survey of Interconnection Network, IEEE Trans. on Computers. Dec. 1981, pp. 12-27.
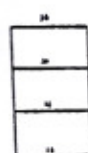
Regular

Figure 1a

Irregular

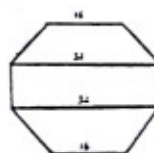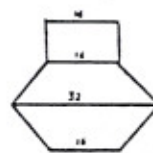Figure 1b

Somewls Rectangular

Figure 1c

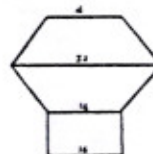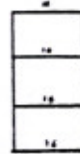| F=(2,2,4) S=(2,2,4) (a) | F=(4,2,2) S=(2,2,4) (b) | F=(2,4,2) S=(2,2,4) (c) |
| F=(2,2,4) S=(2,4,2) (d) | F=(2,4,2) S=(2,4,2) (e) | F=(4,2,2) S=(2,4,2) (f) |
| F=(2,2,4) S=(4,2,2) (g) | F=(2,4,2) S=(4,2,2) (h) | F=(4,2,2) S=(4,2,2) (i) |

Figure 2

638

ABSTRACT

A self-tuning con
contains a proces:
are compared. Th
It is proved that
control system as
vergence.

SMITH CONTROL SYS'

The Smith Control
dead times [1].
properties of the
is designed by co

  (i) control
  (ii) control

The resulting con:
model in parallel
resulting signal

$$X(t) =$$

where q^{-1} is the

$$X(t-1)$$

The polynomial D(

$$D(q^{-1}) =$$

where $q^{-k}P(q^{-1})$ a
respectively, and

$$z(t) =$$

$$v(t) =$$

In this paper it
to that of the pr
equation

$$\hat{X}(t+\ell) =$$